

HRoll

Alumno: Raúl Iozsef Vincze
Grado en Ingeniería Informática

Nombre Consultor/a: Gregorio Robles Martínez

09/06/2021



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/)

Licencias alternativas (elegir alguna de las siguientes y sustituir la de la página anterior)

A) Creative Commons:



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada [3.0 España de Creative Commons](#)



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-CompartirIgual [3.0 España de Creative Commons](#)



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial [3.0 España de Creative Commons](#)



Esta obra está sujeta a una licencia de Reconocimiento-SinObraDerivada [3.0 España de Creative Commons](#)



Esta obra está sujeta a una licencia de Reconocimiento-CompartirIgual [3.0 España de Creative Commons](#)



Esta obra está sujeta a una licencia de Reconocimiento [3.0 España de Creative Commons](#)

B) GNU Free Documentation License (GNU FDL)

Copyright © 2021 Raúl Iozsef Vincze

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free

Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

C) Copyright

© Raúl Iosif Vincze

Reservados todos los derechos. Está prohibido la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reprografía, el microfilme, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler y préstamo, sin la autorización escrita del autor o de los límites que autorice la Ley de Propiedad Intelectual.

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>HRoll</i>
Nombre del autor:	<i>Raúl Iozsef Vincze</i>
Nombre del consultor/a:	<i>Gregorio Robles Martínez</i>
Nombre del PRA:	
Fecha de entrega (mm/aaaa):	09/06/2021
Titulación:	<i>Grado de Ingeniería Informática</i>
Área del Trabajo Final:	<i>Desarrollo WEB</i>
Idioma del trabajo:	<i>Español</i>
<p>Resumen del Trabajo (máximo 250 palabras): <i>Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.</i></p>	
<p>El objetivo de este proyecto es utilizar las tecnologías de la información y el avance tecnológico para poder disponer de una aplicación que mejora el rendimiento de la creación de usuarios en la base de datos y el almacenamiento de la información.</p> <p>Se van a utilizar todos los conocimientos adquiridos en el grado de Ingeniería Informática a lo largo de los años cursados.</p> <p>El proyecto HRoll se encarga de realizar una solución empresarial digital tanto para dar de alta a los empleados de la empresa como una aplicación que va a permitir ver a todos los empleados existentes.</p> <p>También ofrece soluciones al departamento de Ventas y la aplicación va a tener la opción de manejar los pedidos en este departamento.</p> <p>El modo de acceso de los usuarios finales será vía browser (Chrome, Mozilla, Opera etc.) desde cualquier dispositivo (escritorio, Tablet o móvil). Solamente será imprescindible tener acceso a Internet.</p> <p>Se van a utilizar las siguientes arquitecturas: SAP BTP Cockpit – SAP Business Technology Platform Cockpit, SAP ERP Gateway, SAP Cloud Connector.</p> <p>Se van a utilizar las siguientes tecnologías: UI5, SAPUI5, Odata en el lenguaje de programación ABAP.</p> <p>Las herramientas de trabajo a utilizar son: SAP Business Application Studio, Eclipse y Github.</p>	

Abstract (in English, 250 words or less):

The objective of this project is to use information technologies and technological progress to have an application that improves the performance of the creation of users in the database and the storage of information.

All the knowledge acquired in the degree of Computer Engineering throughout the years studied will be used.

The HRoll project is responsible for creating a digital business solution both to register the employees of the company and an application that will allow to see all existing employees.

It also offers solutions to the Sales department and the application will have the option to manage orders in this department.

The access mode for end users will be via browser (Chrome, Mozilla, Opera etc.) from any device (desktop, tablet or mobile). It will only be essential to have Internet access.

The following architectures will be used: SAP BTP Cockpit - SAP Business Technology Platform Cockpit, SAP ERP Gateway, SAP Cloud Connector.

The following technologies will be used: UI5, SAPUI5, Odata in ABAP programming language.

The working tools to be used are: SAP Business Application Studio, Eclipse and Github.

Índice

Título del trabajo.....	1
1. Resumen del proyecto.....	1
1.1 Introducción.....	1
1.2 Objetivos	1
1.3 Planificación temporal	2
2. Evaluación de los riesgos.....	2
3. Requisitos.....	3
3.1 Requisitos de alto nivel funcionales	3
3.2 Requisitos de alto nivel no funcionales.....	4
4. Análisis y Diseño	5
4.1 Análisis de necesidades del proyecto.....	5
4.2 Estudio de la arquitectura y tecnologías	5
4.2.1 Arquitectura.....	5
4.2.2 Tecnologías	7
4.2.3 Herramientas de trabajo.....	7
4.3 Conceptos FIORI y SAPUI5	8
4.3.1 Fiori.....	8
4.3.2 SAPUI5	9
4.4 Casos de uso	10
4.4.1 Crear empleado	10
4.4.2 Ver empleado.....	11
4.4.3 Firmar pedido.....	11
4.5 Diseño de base de datos.....	12
4.6 Prototipos	14
4.7 Perfiles de usuarios	15
4.8 Estructura de páginas.....	15
4.8.1 Páginas en función de estructura web	15
4.8.2 Página en función de la distribución del contenido	16
5. Diseño del proyecto.....	17
6. Implementación y desarrollo.....	18
6.1 Configuración entorno de trabajo	18
6.2 Desarrollo del proyecto.....	21
6.2.1 Proyecto TFG01	21
6.2.2 Proyecto HROLL.....	24
6.3 Pruebas unitarias / Guía de usuario	27
6.3.1 Menu principal.....	27
6.3.2 Crear empleado	28
6.3.3 Información empleado.....	32
6.3.4 Dar de baja empleado.....	34
6.3.5 Ascender empleado	36
6.3.6 Ver historial empleado	38
6.3.7 Firmar pedido Ventas.....	38
7. Lanzamiento.....	47
7.1 Configuración app web.....	47
7.2 Lanzamiento app web	50
8. Cambios respecto a la planificación inicial	51

9. Objetivos conseguidos	51
10. Conclusiones.....	52
11. Anexos	52
Anexo 1. Entregables del proyecto.....	52
Anexo 2. Código fuente (extractos)	52
Anexo 3. Repositorios Git.....	57
Anexo 4. Bibliografía	58

Título del trabajo:

HRoll – creación de una plataforma digital para HR y Ventas.

1. Resumen del proyecto

En los últimos años, las nuevas herramientas tecnológicas de la información y comunicación han producido un cambio enorme en la manera de trabajar en las empresas y en la manera que las personas interactúan en el ámbito de los negocios. Todas las grandes instituciones disponen de red para afrontar los retos y migrar todos los servicios y aplicaciones a un entorno web para permitir a todos los usuarios acceder a ellos mediante Internet.

Las grandes empresas trabajan con grandes bases de datos y muchas instituciones todavía trabajan con mucha información en papel y no se han cambiado a modo digitalización.

El proyecto HRoll está planteado con el objetivo de dar una solución empresarial digital y crear una aplicación web corporativa con un menú amigable y fácil de usar para los usuarios.

A lo largo del proyecto se hará todo el análisis y el diseño, y luego en la fase de desarrollo e implementación se van a presentar más en detalle los pasos a implementar.

El objetivo de la aplicación es ofrecer la posibilidad de crear empleados en el sistema, en la base de datos y poder ver los empleados existentes.

1.1 Introducción

El proyecto HRoll se encarga de realizar una solución empresarial digital tanto para dar de alta a los empleados de la empresa como una aplicación que va a permitir ver a todos los empleados existentes.

También ofrece soluciones al departamento de Ventas y la aplicación va a tener la opción de manejar los pedidos en este departamento.

El modo de acceso de los usuarios finales será vía browser (Chrome, Mozilla, Opera etc.) desde cualquier dispositivo (escritorio, Tablet o móvil). Solamente será imprescindible tener acceso a Internet.

Este proyecto quiere hacer uso del avance de la tecnología actual y potenciar la actividad de las instituciones para poder alcanzar los objetivos.

1.2 Objetivos

El objetivo de este proyecto es utilizar las tecnologías de la información y el avance tecnológico para poder disponer de una aplicación que mejora el rendimiento de la creación de usuarios en la base de datos y el almacenamiento de la información.

Con este fin se decide crear una aplicación que facilite el acceso del personal encargado de crear nuevos empleados en el sistema, y disponer de una base de datos que almacene todos los empleados de la empresa.

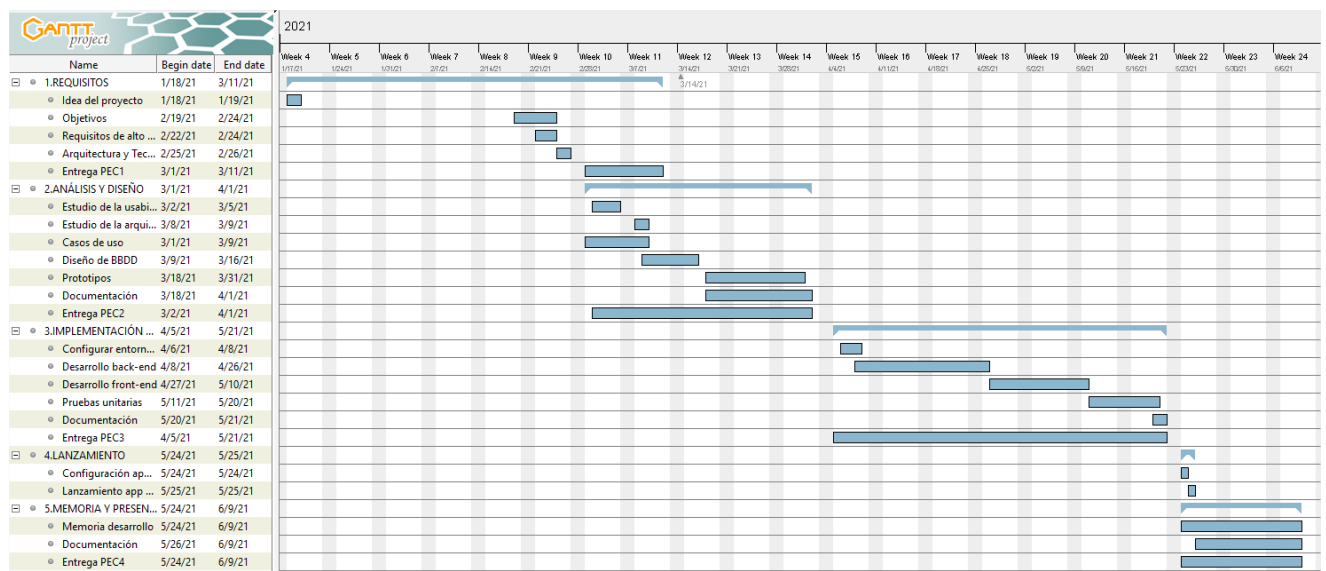
Otra funcionalidad de la aplicación será la gestión de roles y autorizaciones. No todos los usuarios tendrán el mismo acceso a la aplicación. Dependiendo del departamento de HR o Ventas, los usuarios serán limitados dependiendo de su rol y perfil.

Gracias a las nuevas tecnologías, todo será más organizado en la institución y esto aportará un gran beneficio a la empresa.

1.3 Planificación temporal

La planificación y el diagrama es temporal, van a haber cambios en la fase de desarrollo e implementación.

El proyecto empieza el día 17/02/2021 y termina el día 09/06/2021. Para construir la planificación temporal se ha utilizado la GANTT PROJECT.



2. Evaluación de los riesgos

Hay que organizarse bien para poder llegar al alcance del proyecto y evitar los posibles riesgos de alcance que son las tareas que ponen en peligro los objetivos, los resultados o el calendario del proyecto. Es un riesgo empresarial, pero se tomarán todas las medidas necesarias para evitar que estos riesgos ocurran. Se van a crear los objetivos concretos y fechas coherentes para cada fase y se van a organizar bien las tareas para cada departamento.

No habrá riesgos de coste, pero puede haber riesgos de tiempo, ya que en muchos casos se confía mucho de que el proyecto incluso puede terminar antes y luego ocurren errores imprevistos de la aplicación. Hay que organizar

bien el tiempo e intentar dejar un margen para que se pueda probar bien la aplicación y en el caso de encontrar inconsistencias, tener tiempo con antelación para arreglarlas.

También hay que evitar los riesgos de los recursos y de la comunicación de los departamentos, hay que tener en cuenta que los recursos están a disposición de acabar el proyecto y que la comunicación entre los jefes del proyecto y los desarrolladores esté clara, cada uno tiene que saber qué es lo que tiene que hacer y las fechas de cada tarea. Hay que organizarse bien y el jefe del proyecto tenga claro todos los pasos, los tiempos y las tareas que hay que hacer.

Un riesgo común también es el tecnológico porque a veces la tecnología puede fallar en algún que otro punto por lo tanto hay que programar un tiempo extra para imprevistos de este tipo como serían los retrasos en la implementación. Disponer de un equipo de seguridad informática que se encarga de solucionar estos problemas de TI y solucionar fácilmente cualquier complicación que pueda surgir en el camino.

3. Requisitos

El proyecto consiste en desarrollar una aplicación web que permita crear empleados en el sistema, en la base de datos y poder ver los empleados existentes. Se podrán dar de alta a los nuevos usuarios y también gestionar a los usuarios existentes ya en el sistema.

La aplicación dispondrá también de una opción para gestionar y manejar los pedidos del departamento de Ventas.

Como hemos mencionado, es imprescindible que los usuarios finales tengan acceso a un browser desde cualquier dispositivo, y también tener acceso a internet, para poder usar dicha aplicación web.

Otra funcionalidad de la aplicación será la gestión de roles y autorizaciones. Obviamente no todos los usuarios tendrán los mismos privilegios, habrá roles para administradores y para usuarios normales. También va a depender del departamento de ventas o del departamento de HR porque los usuarios se van a repartir según su perfil.

Tanto los requisitos funcionales como los requisitos no funcionales, tienen margen de cambio en la implementación del proyecto. Cualquier cambio en la implementación se verá reflejado en el documento final del trabajo de fin de grado.

3.1 Requisitos de alto nivel funcionales

Estos son los requisitos, que pueden sufrir cambios a lo largo de la implementación, pero que se han obtenido mediante reuniones con el cliente. Se hará un resumen de las funcionalidades de la aplicación:

- Sección Empleados
 - Crear Empleado
 - Solo el administrador podrá crear los usuarios
 - Se podrá cancelar el proceso
 - Hay 3 opciones:
 - Tipo de empleado: Interno/Autónomo/Gerente
 - Datos empleados:
 - Nombre
 - Apellidos
 - DNI/CIF (dependiendo si es Interno o Gerente)
 - Saldo bruto anual
 - Precio diario
 - Fecha de incorporación
 - Información adicional: añadir fichero, comentario con alguna información auxiliar.
 - Ver Empleado
 - Se navega a una vista de Master – Detail donde se muestran los siguientes datos:
 - Nombre
 - Apellidos
 - DNI/CIF
 - Al pulsar sobre el empleado, se muestra luego toda la información necesaria.
 - Se podrán dar de baja los empleados y ascender a los empleados, pero esto se verá en la PEC3 cuando se implementa.
- Sección Pedidos
 - Se seleccionan los pedidos para poder firmarse. Este paso también lo veremos en la PEC3 en el momento de la implementación.

3.2 Requisitos de alto nivel no funcionales

- Se debe permitir el acceso a usuarios de forma concurrente hasta 50 usuarios
- Los usuarios se tienen que familiarizar con el uso de las tecnologías web
- La aplicación tiene que ser amigable para el usuario
- La base de datos de la aplicación se alojará en Cloud de SAPUI5.
- El sistema operativo puede ser tanto Windows, como Ubuntu, Linux etc.
- El servidor será el servidor web de SAPUI5 por ahora (si hay cambios luego en la implementación se van a cambiar y se van a reflejar).

- Las contraseñas de los usuarios se almacenarán cifradas en la base de datos.
- El acceso será mediante el DNI/CIF (para el Autónomo) del usuario por ahora (si hay cambios luego en la implementación se van a cambiar y se van a reflejar).

4. Análisis y Diseño

4.1 Análisis de necesidades del proyecto

Se necesita crear una aplicación web que permita crear empleados en el sistema, en la base de datos y poder ver los empleados existentes.

Actualmente no se dispone de una aplicación para la creación de los usuarios y tampoco para la gestión de los usuarios existentes.

También se necesita crear una opción para gestionar y manejar los pedidos del departamento de Ventas.

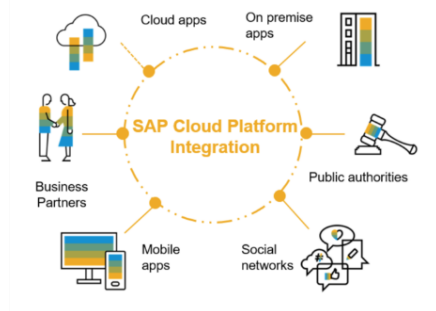
Los usuarios van a necesitar solamente que tengan acceso a un browser desde cualquier dispositivo, y también tener acceso a internet, para poder usar dicha aplicación web.

4.2 Estudio de la arquitectura y tecnologías

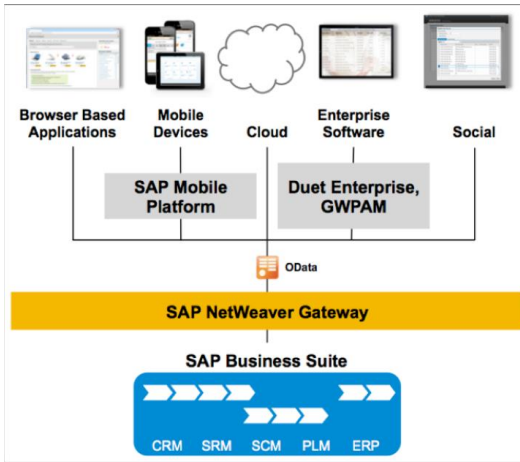
Para la arquitectura y las tecnologías del proyecto vamos a trabajar con las siguientes.

4.2.1 Arquitectura

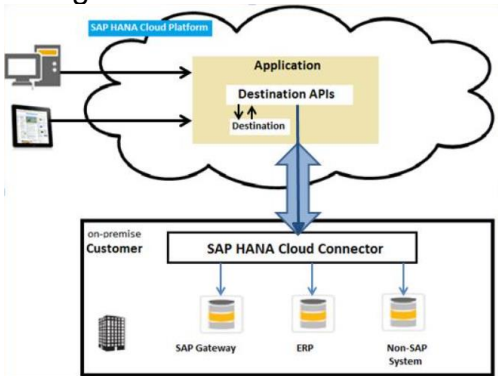
SAP BTP Cockpit – SAP Business Technology Platform Cockpit – es un cloud de SAP para el acceso de los usuarios finales con autenticación manejada por el mismo. La plataforma es la que va a servir las aplicaciones para los usuarios, es decir la capa de cliente.



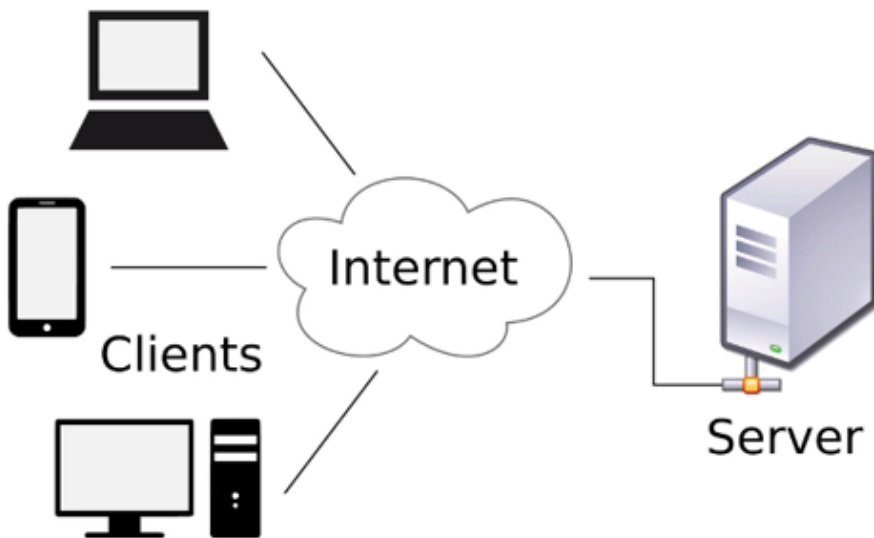
SAP ERP Gateway (Enterprise Resource Planning) – es el Sistema back-end on-premise para el manejo de los datos de negocio.



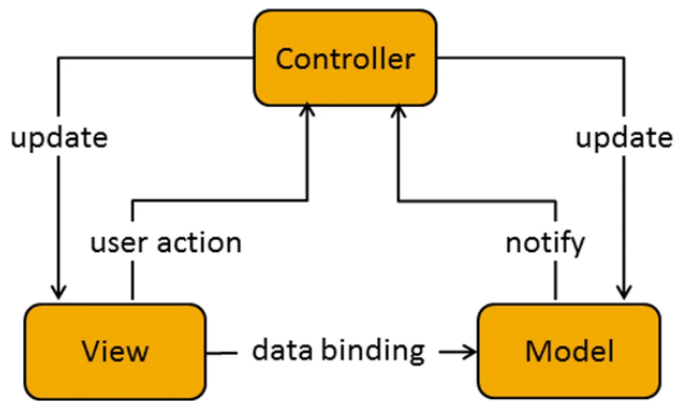
SAP Cloud Connector – es el cloud que se encarga de realizar una conexión segura entre el sistema back-end y el cloud de SAP a través de la configuración de los destinos.



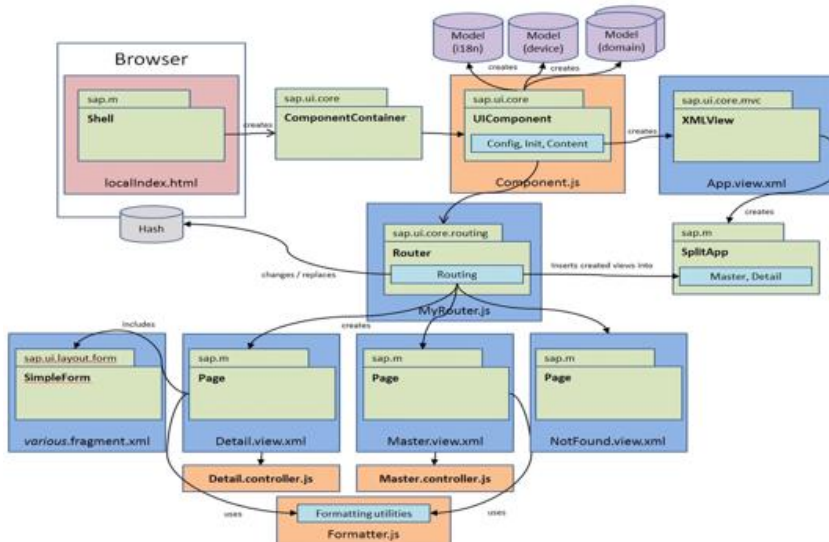
Usamos el modo Cliente-Servidor es decir, SAPUI5-Cliente Servidor:



El funcionamiento es con MVC, es decir el modelo vista controlador que es el patrón para implementar las interfaces de diseño.



Funciona de la siguiente manera:



4.2.2 Tecnologías

UI5 – librería o kit de componentes web (HTML5, JavaScript, CSS y XML)

SAPUI5 – experiencia de usuario para interactuar con las aplicaciones responsive basada en la librería UI5 para la creación de las aplicaciones empresariales.

Odata en el lenguaje de programación ABAP – para interactuar con la capa de persistencia con el propósito de exponer los datos a través del SAP Gateway en el sistema Back-end, también conocido como SAP ERP.

4.2.3 Herramientas de trabajo

SAP Business Application Studio – herramienta de desarrollo.

Eclipse- herramienta Theia

Github para respaldar el código.

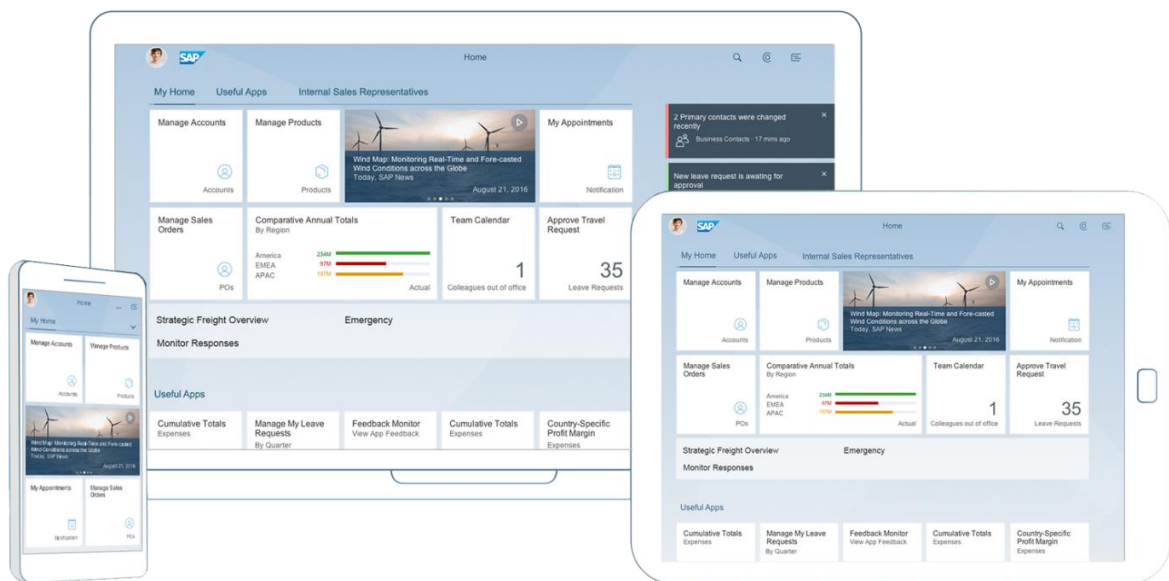
4.3 Conceptos FIORI y SAPUI5

4.3.1 Fiori

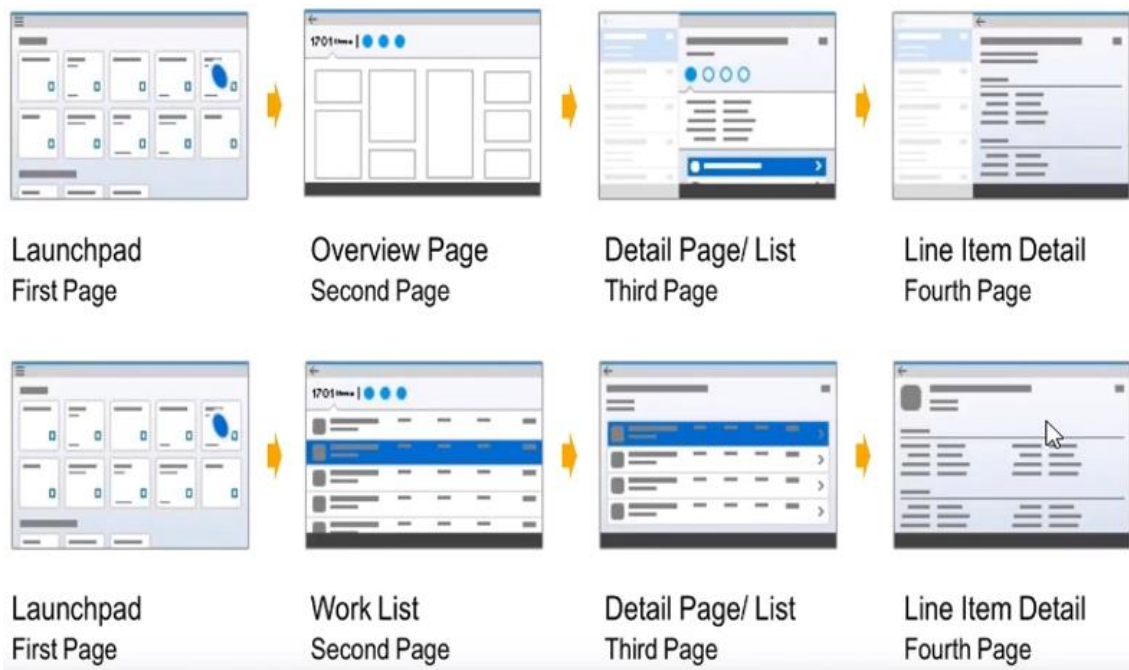
Fiori es una interfaz de usuario, es importante aclarar que no es ningún tipo de tecnología o lenguaje de programación.

Es un paradigma de programación que nos dicta unas normas específicas a la hora de desarrollar una aplicación SAPUI5 centrándose en el usuario y su experiencia.

Otra definición válida para SAP FIORI es un conjunto de aplicaciones SAPUI5 que ha desarrollado SAP siguiendo dicho paradigma de programación. En sus inicios en 2013, en la primera versión SAP solo contaba con 25 aplicaciones Fiori. Hoy en día cuenta con miles de aplicaciones en diferentes negocios.



El diseño es UX y está basado en roles, tiene la posibilidad de accederse desde cualquier dispositivo o tablet porque dispone de la opción responsive, es simple, coherente y atractiva.



4.3.2 SAPUI5

Los lenguajes y las tecnologías que usa SAPUI5 se mencionan a continuación:

- **HTML5** – contenido
 - o Controles, propiedades, agregaciones
- **JS Comportamiento**
 - o No es tipado
 - o Independiente de la plataforma y navegador
 - o Orientado a objetos y eventos
 - o No necesita compilación
- **CSS estilo**
 - o Estilo, presentación y estilo gráfico
- **jQuery Biblioteca**
- **XML Estructura**
 - o Controles, propiedades, agregaciones
- **JSON Datos**
 - o Clave-valor-nombre campo y valor es contenido del campo
 - o Estructura
 - o Colección o array



4.4 Casos de uso

A continuación, se muestran los casos de uso que va a tener la aplicación cuando se va a terminar de desarrollar e implementar.

4.4.1 Crear empleado

1. **Tipo de empleado:** Debe aparecer el texto “Seleccione el tipo de empleado para rellenar sus datos” y 3 botones centrados con los textos “Interno”, “Autónomo” y “Gerente” respectivamente. Al pulsar sobre cualquiera de ellos debe avanzar al “Paso 2 – Datos empleado”.
2. **Datos empleado:** Debe aparecer el texto “Complete los datos obligatorios para poder continuar” y un formulario con los siguientes campos:
 - a. **Nombre:** Campo obligatorio y editable
 - b. **Apellidos:** Campo obligatorio y editable
 - c. **DNI:** Solo visible si el tipo de empleado es “Interno” o “Gerente”. Campo obligatorio, editable y debe cumplir con el formato correcto de DNI español (se adjunta lógica para comprobar que es un DNI español. Ej.: 00000010X).
 - d. **CIF:** Solo visible si el tipo de empleado es “Autónomo”. Campo obligatorio y editable.
 - e. **Saldo bruto anual:** Solo visible si el tipo de empleado es “Interno” o “Gerente”. En el caso de ser interno, su valor por defecto es “24.000” y su rango de valores oscila entre los 12.000 y los 80.000. En el caso de ser gerente, su valor por defecto es “70.000” y sus valores oscilan entre 50.000 y 200.000. Campo de tipo “Slider”.

- f. **Precio diario:** Solo visible si el tipo de empleado es “Autónomo”. Su valor por defecto es 400 y sus valores oscilan entre 100 y 2000. Campo tipo “Slider”.
- g. **Fecha de incorporación:** Campo editable y obligatorio de tipo fecha.




3. Información adicional: datos adicionales sobre el usuario.

4.4.2 Ver empleado

En el menú principal al pulsar sobre “Ver empleados” se debe navegar a una nueva vista con el patrón “Master - Detail”. En la parte de la izquierda debe aparecer un listado con los empleados (se muestra su nombre, apellidos y dni) con un filtro y en la parte derecha un mensaje centrado “Seleccione un empleado”.

Al pulsar sobre un empleado, se debe cargar su información representándolo de la siguiente forma:

Una cabecera con:

1. Icono tipo empleado:
 - 0 igual a 
 - 1 igual a 
 - 2 igual a 
2. Nombre y apellidos
3. DNI
4. Fecha de incorporación
5. Tipo de empleado
 - 0 igual “Interno”
 - 1 igual “Autónomo”
 - 2 igual “Gerente”
6. Comentario

4.4.3 Firmar pedido

Dentro de esta opción vamos a tener todos los usuarios que están conectados en la base de datos del departamento de ventas.

Cada usuario va a tener pedidos asignados por lo tanto en cada pedido se va a poder realizar una firma del dicho pedido y adjuntar archivos de tipo PDF o cualquier otro tipo con informaciones.

También se va a necesitar una opción para crear algunas incidencias si pasa algo con algún pedido o si hay errores de firmas, adjunciones de ficheros etc.

4.5 Diseño de base de datos

Los datos de la aplicación de la creación de usuarios se van a guardar en una base de datos de SAP.

Para los pedidos del departamento de Ventas, se va a utilizar el servicio oData para acceder a los datos.

El servicio oData es un componente externo que nos permite enviar y recibir información. Está formado por entidades que son las que se deben especificar al realizar una acción sobre el servicio. Se tienen que especificar en el código fuente.

Una entidad es la representación de un objeto en un servicio oData y están formados por campos en donde 1 o n de ellos son campos clave, teniendo una asociación como relación entre entidades.

Al crear un modelo odata, lo primero que hace este, de forma automática es consultar al “metadata” del servicio, que contiene todas las entidades y demás información de dicho servicio.

El modelo odata tiene funciones específicas para cada una de las acciones que podemos aplicar sobre un servicio odata.

Para los pedidos del departamento de ventas se va a utilizar un servicio oData llamado Northwind.

Observamos como está el servicio accedendo la siguiente URL: <https://services.odata.org/V2/Northwind/Northwind.svc/>

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="utf-8" xml:base="https://services.odata.org/V2/Northwind/Northwind.svc/">
  <workspace>
    <atom:title>Default</atom:title>
    <collection href="Categories">
      <atom:title>Categories</atom:title>
    </collection>
    <collection href="CustomerDemographics">
      <atom:title>CustomerDemographics</atom:title>
    </collection>
    <collection href="Customers">
      <atom:title>Customers</atom:title>
    </collection>
    <collection href="Employees">
      <atom:title>Employees</atom:title>
    </collection>
    <collection href="Order_Details">
      <atom:title>Order_Details</atom:title>
    </collection>
    <collection href="Orders">
      <atom:title>Orders</atom:title>
    </collection>
    <collection href="Products">
      <atom:title>Products</atom:title>
    </collection>
    <collection href="Regions">
      <atom:title>Regions</atom:title>
    </collection>
    <collection href="Shippers">
      <atom:title>Shippers</atom:title>
    </collection>
    <collection href="Suppliers">
      <atom:title>Suppliers</atom:title>
    </collection>
    <collection href="Territories">
      <atom:title>Territories</atom:title>
    </collection>
    <collection href="alphabetical_list_of_products">
      <atom:title>Alphabetical_list_of_products</atom:title>
    </collection>
    <collection href="Category_Sales_for_1997">
      <atom:title>Category_Sales_for_1997</atom:title>
    </collection>
  </workspace>

```

Observamos las entidades y las propiedades de los pedidos.

Para poder acceder desde Cloud Foundry y representar la información desde un servicio oData, tenemos que crear un destino en nuestro servicio de BTP Cockpit.

Para ellos entramos dentro de nuestra cuenta y creamos el destino.

En SAP BTP creamos un nuevo destino a partir de nuestro sistema On-Premise con las siguientes propiedades:

Parámetro	Valor
sap-client	001
WebIDEEEnabled	true
WebIDESystem	NPL
WebIDEUsage	odata_gen,odata_abap,dev_abap
HTML5.DynamicDestination	true

Este destino se va a utilizar para la creación de los empleados.

The screenshot shows the 'Destination Configuration' interface for a destination named 'NPL'. The configuration includes:

- Name:** NPL
- Type:** HTTP
- Description:** (empty)
- URL:** http://erp13.sap4practice.com:9037
- Proxy Type:** Internet
- Authentication:** BasicAuthentication
- User:** SAPUI5
- Password:** (masked with asterisks)

Additional Properties:

- HTML5.Dyna...: true
- sap-client: 001
- WebIDEEEnabled...: true
- WebIDESyst...: NPL
- WebIDEUsage: odata_gen,odata_aba...

Buttons at the bottom: Edit, Clone, Export, Delete, Check Connection.

Comprobamos la conexión y observamos que está bien vinculada:

This screenshot is identical to the previous one, showing the configuration for the 'NPL' destination. The 'Check Connection' button is highlighted, indicating that the connection has been verified and is successful.

Tenemos que crear otro destino para acceder a los pedidos. Para ello se realiza las siguientes configuraciones:

The screenshot shows the 'Destination Configuration' interface for a destination named 'Northwind'. The configuration includes:

- Name:** Northwind
- Type:** HTTP
- Description:** Northwind
- URL:** https://services.odata.org/
- Proxy Type:** Internet
- Authentication:** NoAuthentication

Additional Properties:

- HTML5.Dyna...: true
- WebIDEEEnabled...: true
- WebIDESyst...: true
- WebIDEUsage: odata_gen

Use default JDK truststore

Buttons at the bottom: Edit, Clone, Export, Delete, Check Connection.

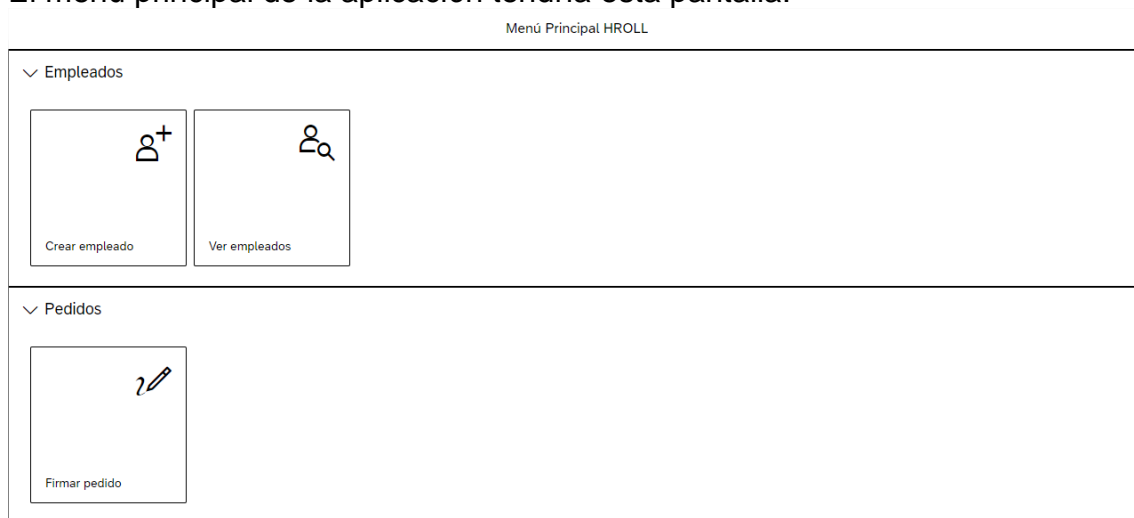
4.6 Prototipos

A continuación, se muestran los pantallazos con los prototipos de la aplicación, obviamente que la app final va a tener mejor vista.

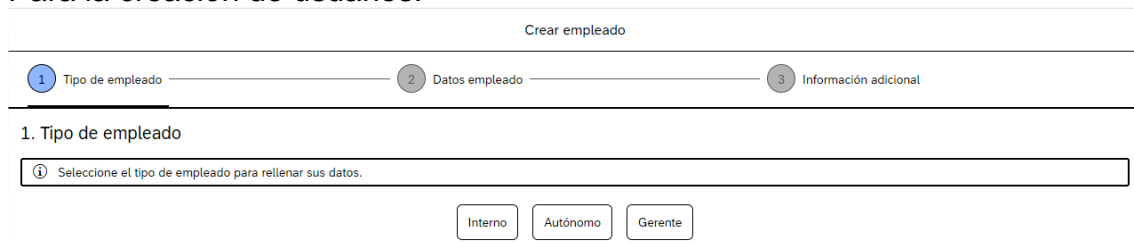
Indicar que puede haber muchos cambios en la implementación final del desarrollo. De momento se muestra una idea de lo que será la aplicación final, pero puede que se cambien algunos detalles.

Hasta que no se entrega el documento final, la capa de aplicación puede sufrir cambios en la pantalla.

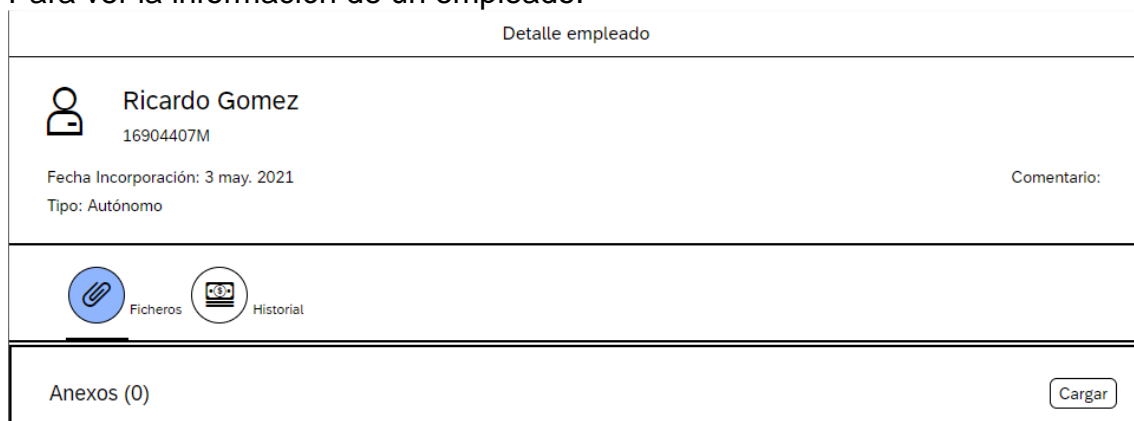
El menú principal de la aplicación tendría esta pantalla:



Para la creación de usuarios:



Para ver la información de un empleado:



Para la pantalla de la firmación de los pedidos:

Pedidos de los empleados Departamentos de Ventas HROLL

ID Empleado: País:

ID Empleado	Nombre	País	Mostrar De...
-------------	--------	------	---------------

Para la firma del pedido:

< ID Pedido 10270

ID Cliente: WARTH

<p>Fecha pedido: 1 ago. 1996</p> <p>Transporte: 136.5400</p> <p>Nombre envío: Wartian Herkku</p> <p>Dirección de envío: Torikatu 38, Oulu, Finland</p>		<input type="button" value="Limpiar"/>
--	--	--

4.7 Perfiles de usuarios

Cada usuario tendrá un rol y un perfil asignado:

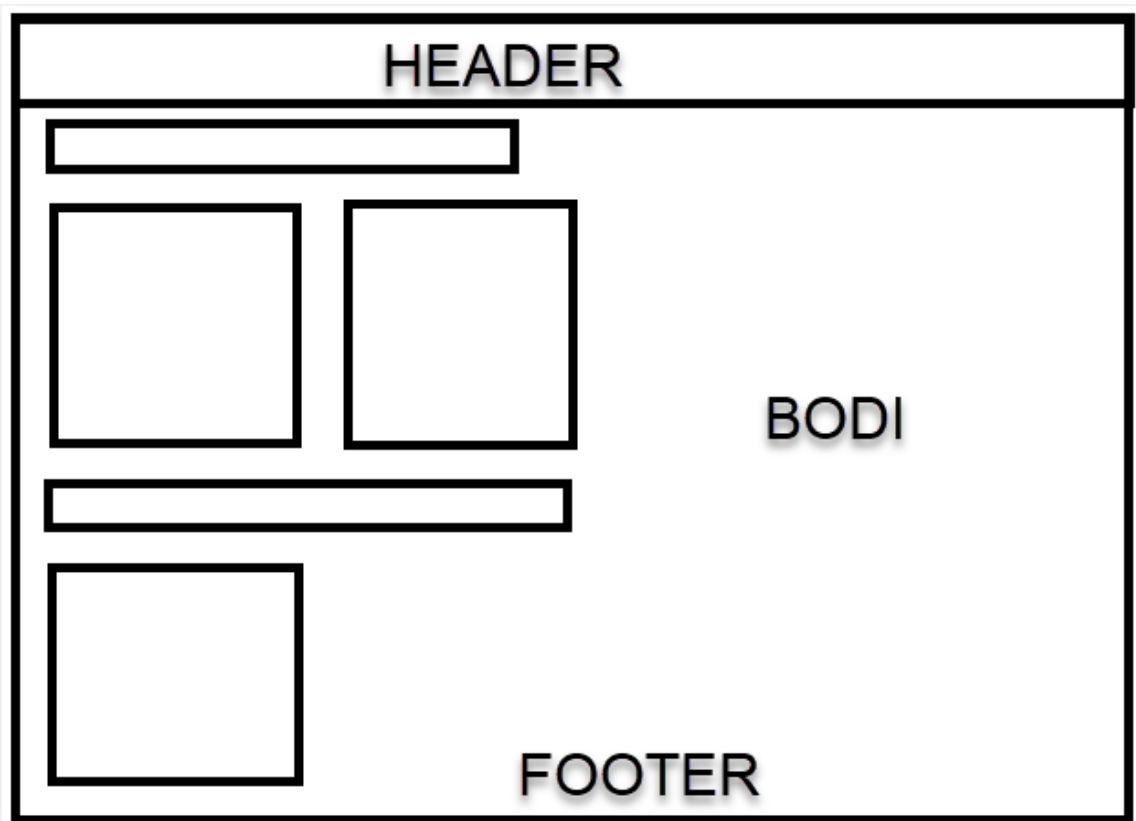
- Los administradores podrán crear y ver los empleados, darlos de baja y ascenderlos, y también firmar pedidos.
- Los empleados normales podrán ver sus usuarios, no podrán modificar nada.
- Los usuarios de HR tendrán sus actividades mientras que los usuarios de Ventas tendrán las suyas.
- Cada departamento es independiente, pero los administradores podrán ver todo, tendrán control absoluto.

4.8 Estructura de páginas

A continuación, se va a mostrar un prototipo del análisis de la estructura de páginas.

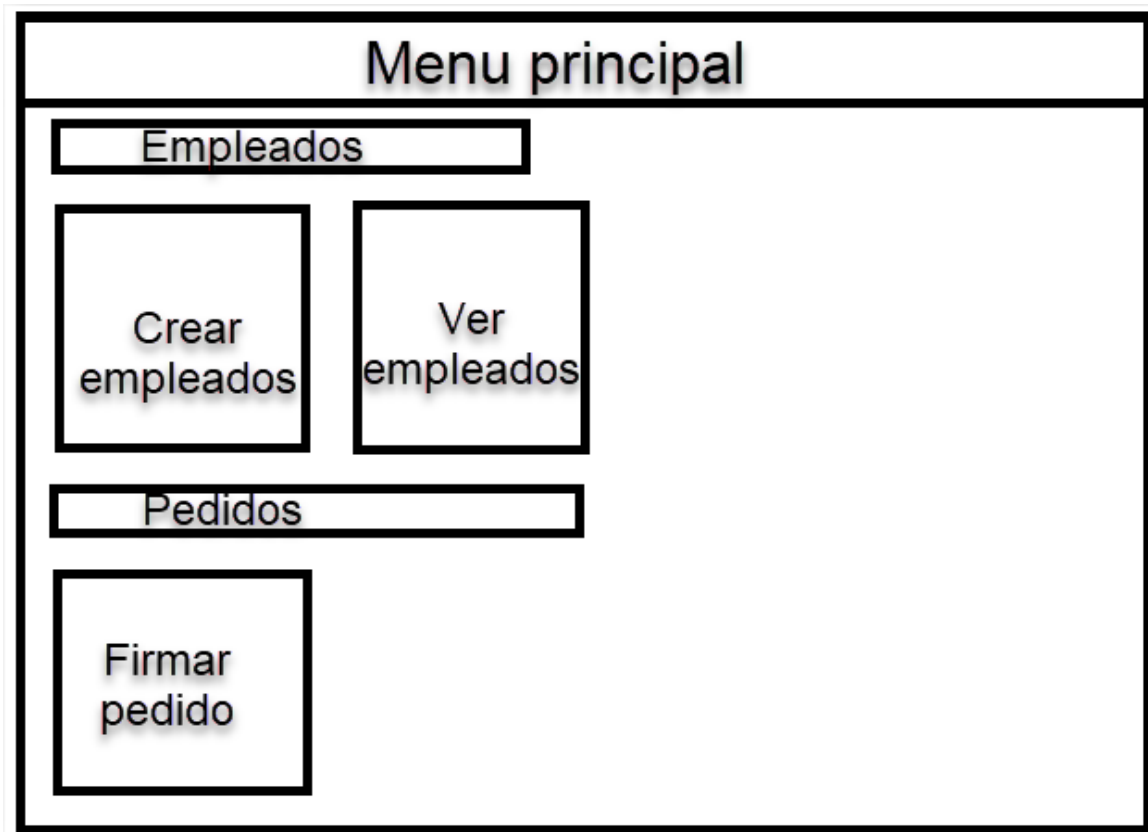
4.8.1 Páginas en función de estructura web

- Cabecera o header
- Cuerpo o body
- Pie de página o footer



4.8.2 Página en función de la distribución del contenido

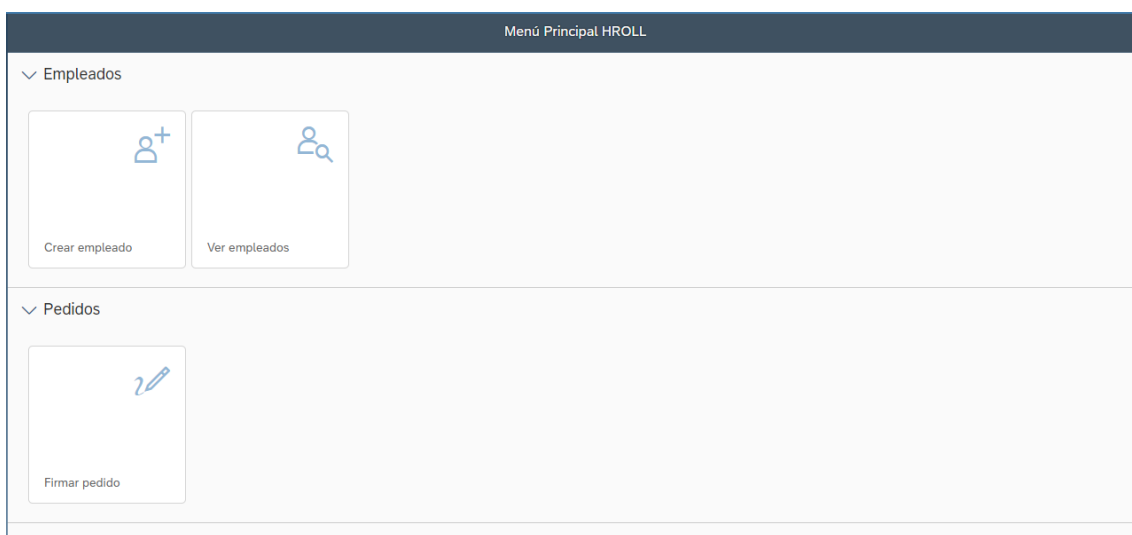
- Menú Principal
- Empleados
 - o Crear Empleado
 - o Ver Empleado
- Pedidos
 - o Firmar pedido



El pie de página para este caso, sería a la hora de cáncer la creación/visualización de un empleado, o también a la hora de cancelar el firmamento del pedido.

5. Diseño del proyecto

Observamos que se ha diseñado la pantalla final del proyecto que va a tener la siguiente vista donde se explicará en los requisitos funcionales y en los casos de uso, cada parte correspondiente de la aplicación (Empleados/Crear/Ver, Pedidos/Firmar Pedido).



A continuación, en la parte de la implementación y desarrollo del proyecto se van a detallar los pasos que se han seguido.

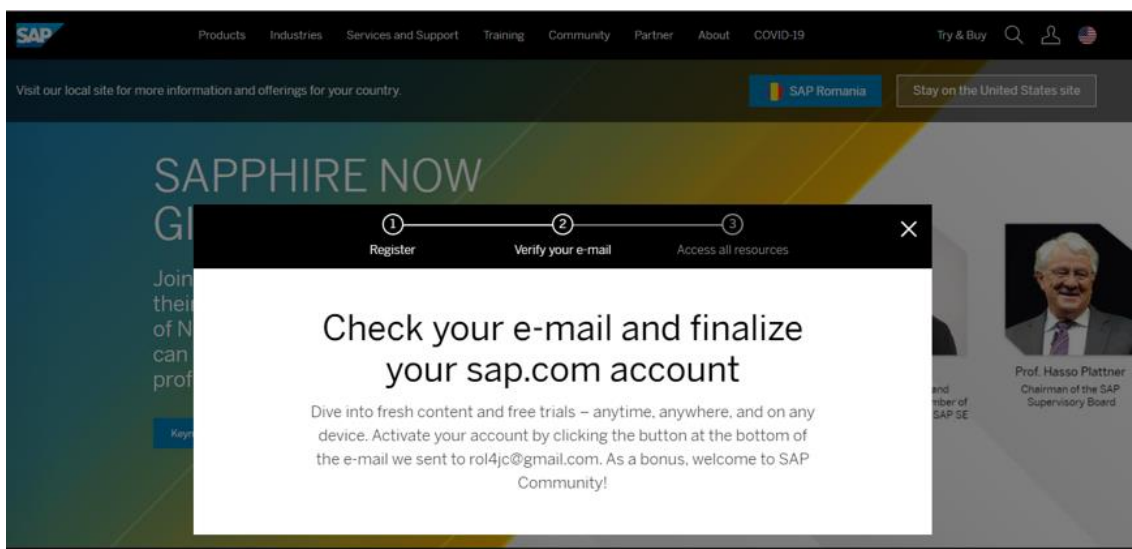
6. Implementación y desarrollo

A continuación, se van a indicar los pasos que se han necesitado para poder implementar y desarrollar este proyecto.

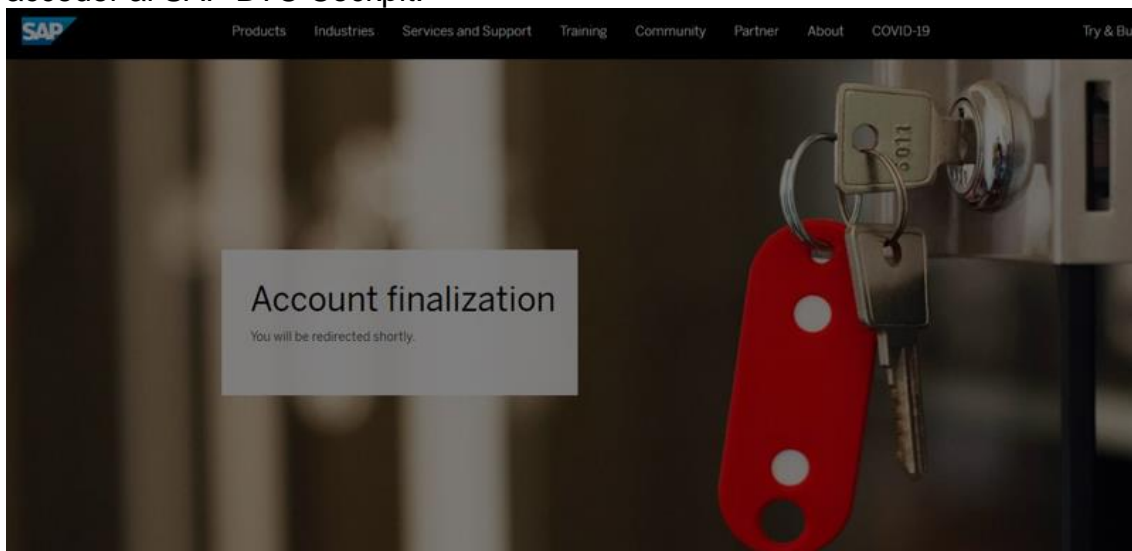
6.1 Configuración entorno de trabajo

Para el entorno de trabajo, se ha creado el usuario rol4jc@gmail.com y he tenido que crear una cuenta trial en SAP.

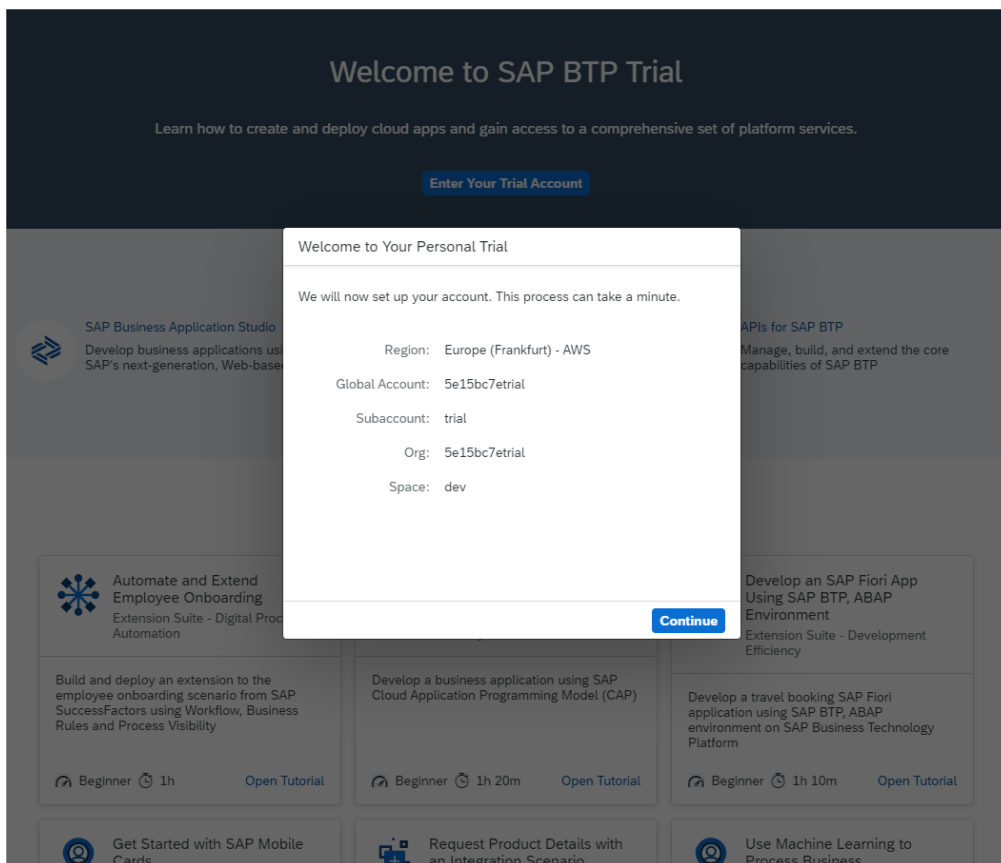
Para ello se accede a <https://sap.com> y se crea el usuario. Se indican los datos necesarios y luego se envía un correo de confirmación de la cuenta.



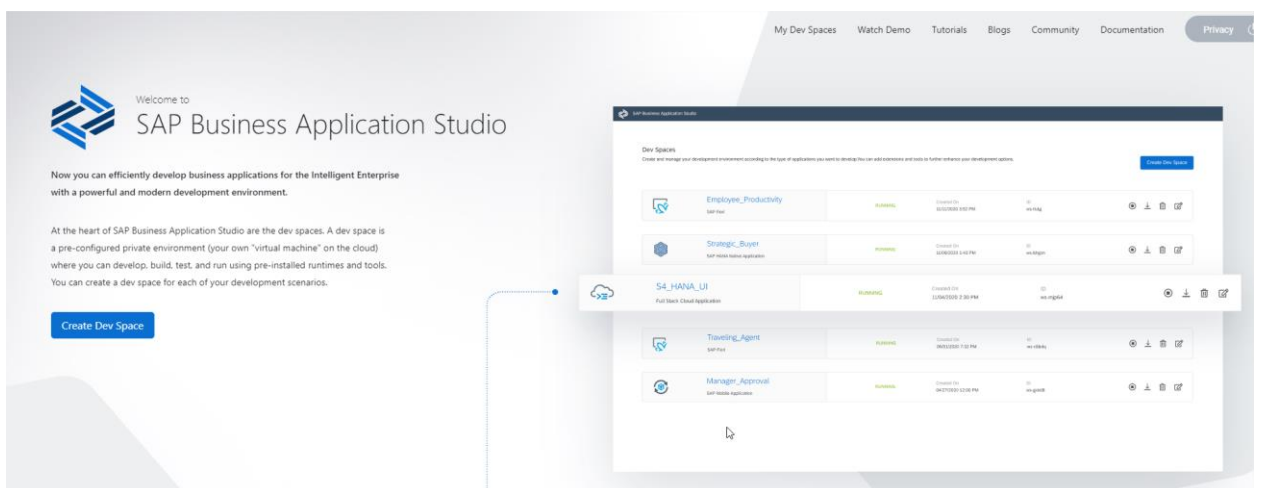
Una vez realizado el chequeo, se crea la cuenta correctamente y se puede acceder al SAP BTC Cockpit.



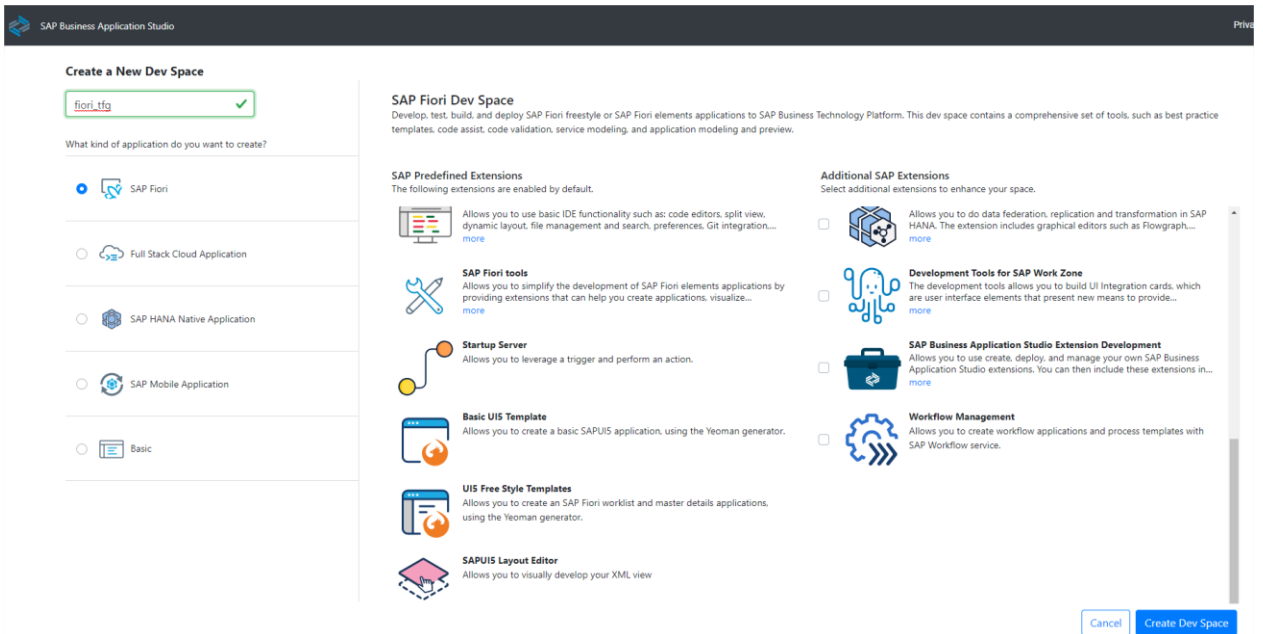
Una vez entrados dentro, observamos que la cuenta trial se ha creado correctamente:



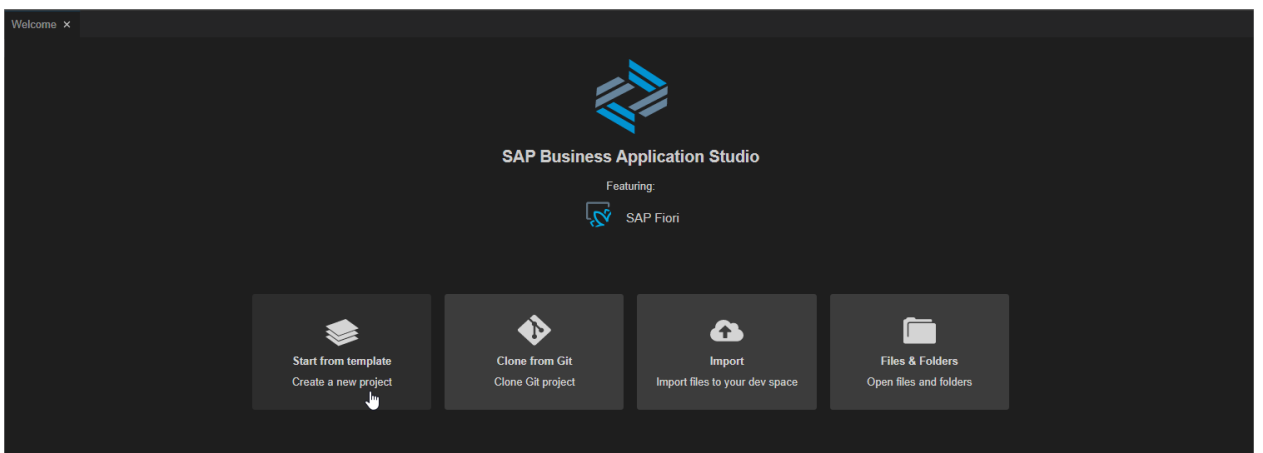
Para la creación de los proyectos y para el entorno de desarrollo, tenemos que entrar dentro del SAP Business Application Studio que tenemos dentro de nuestra cuenta:



Creamos nuestro espacio de trabajo para el proyecto de TFG llamado fiori_tfg:



Una vez creado, entramos dentro y tenemos esta vista:



Observamos la versión que tenemos del GIT:

```
user: projects $ git --version
git version 2.20.1
user: projects $ npm -version
6.14.11
user: projects $ node -v
v14.16.0
user: projects $
```

En el terminal, nos conectamos a la cuenta del BI en el cloud de SAP con los siguientes comandos:

```
~/projects ×
user: projects $ cf login
API endpoint: https://api.cf.eu10.hana.ondemand.com

Email: rol4jc@gmail.com
Password:

Authenticating...
OK

Targeted org 5e15bc7etrial.

Targeted space dev.

API endpoint: https://api.cf.eu10.hana.ondemand.com
API version: 3.99.0
user: rol4jc@gmail.com
org: 5e15bc7etrial
space: dev
user: projects $ █
```

A continuación, creamos una cuenta de Github para crear el repositorio con el código fuente del proyecto TFG. Configuramos el GIT con los siguientes comandos:

PDT

6.2 Desarrollo del proyecto

Se han creado 2 proyectos TFG01 y HROLL.

Los proyectos se pueden crear con plantilla o vía terminal. Para los proyectos que se han creado para el proyecto, se ha utilizado el terminal para crearlos.

6.2.1 Proyecto TFG01

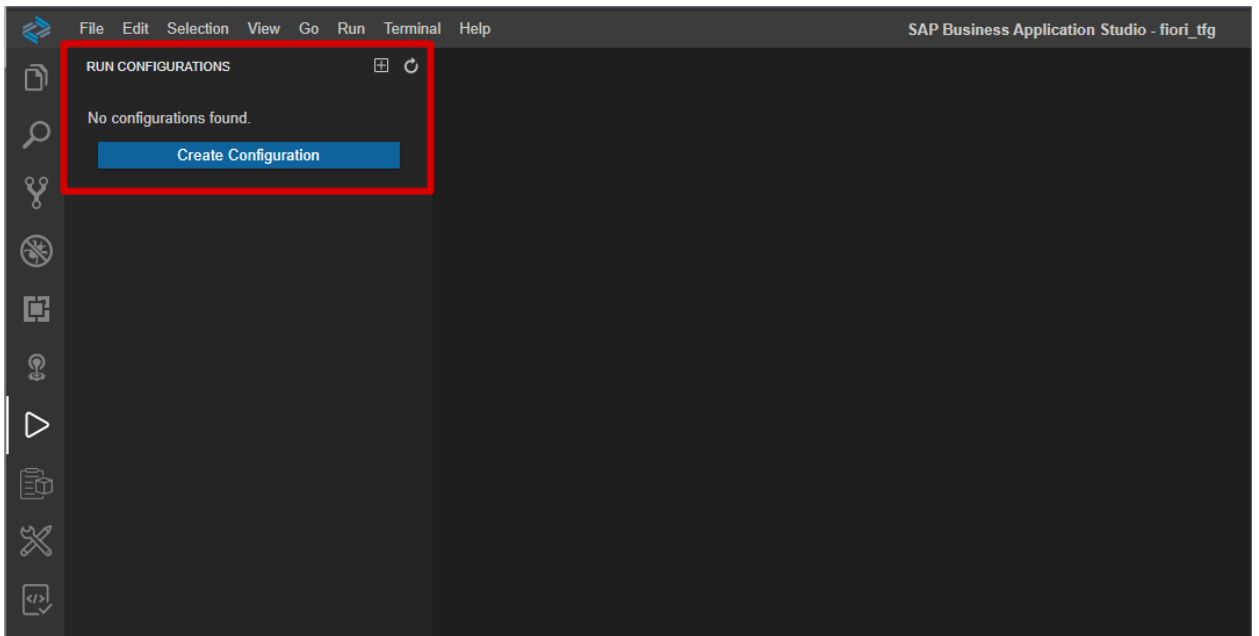
Dentro de SAP Business App, en el entorno de trabajo y de desarrollo se ha utilizado el terminal para crear el Proyecto TFG01 que se encarga de la parte de Firmar los pedidos.

```
~/projects x
user: projects $ yo
? 'Allo RaulRoll! What would you like to do? Fiori Project

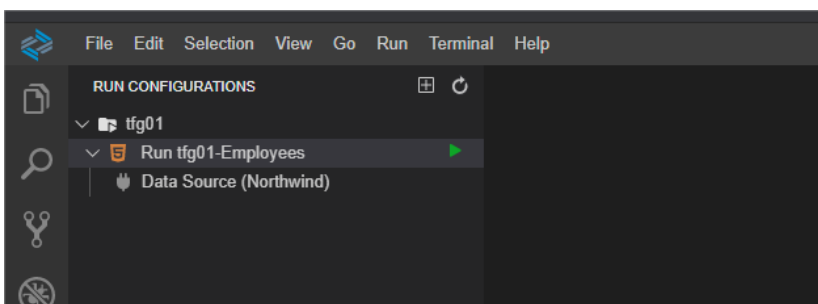
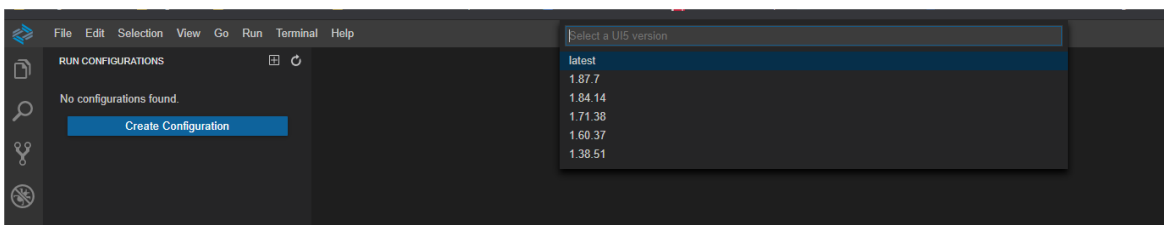
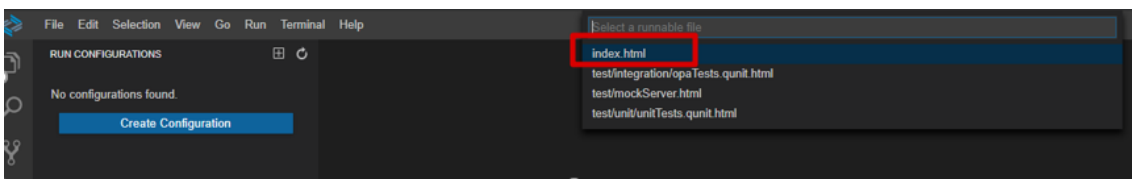
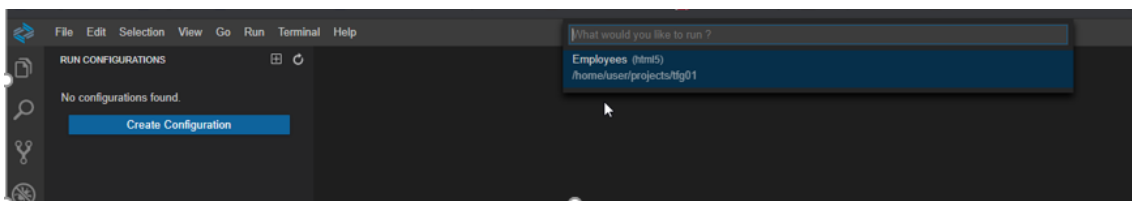
Make sure you are in the directory you want to scaffold into.
This generator can also be run with: yo fiori-project

? Select the target running environment Cloud Foundry
? Select the template you want to use SAPUI5 Application
? Enter a project name tfg01
The project will be generated in the /home/user/projects/tfg01 directory.
? Select your HTML5 application runtime Standalone Approuter
? Enter an HTML5 module name Employees
? Do you want to add authentication? No
? Enter a namespace hrolltfg
? Do you want to enable Karma tests? No
? Enter a view name MainView
? Do you want to add a data service? Yes
? Select a system My SAP systems
? Select a source Northwind (System URL)
? Enter a path to the OData service (e.g. /sap/opu/odata/sap/SERVICE_NAME) /V2/Northwind/Northwind.svc/
create SAP-Config-1623203515714.json
create tfg01/tfg01-approuter/package.json
create tfg01/tfg01-approuter/xs-app.json
create tfg01/tfg01-approuter/.gitignore
create tfg01/package.json
create tfg01/Employees/.eslintrc
create tfg01/Employees/package.json
create tfg01/Employees/tsconfig.json
create tfg01/Employees/ui5.yaml
create tfg01/Employees/xs-app.json
create tfg01/Employees/webapp/Component.js
create tfg01/Employees/webapp/index.html
create tfg01/Employees/webapp/manifest.json
create tfg01/Employees/webapp/css/style.css
create tfg01/Employees/webapp/i18n/i18n.properties
create tfg01/Employees/webapp/localService/mockserver.js
create tfg01/Employees/webapp/model/models.js
create tfg01/Employees/webapp/test/initMockServer.js
create tfg01/Employees/webapp/test/mockServer.html
create tfg01/Employees/webapp/test/integration/AllJourneys.js
```

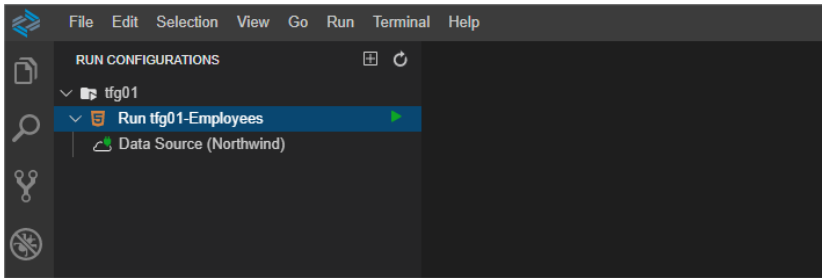
Una vez configurado el proyecto y creado el código, se ha tenido que conectar con el servicio oData explicado en otro apartado, para acceder a los datos y a los usuarios del departamento de ventas.
Para ello se ha tenido que crear una configuración:



Estos son los pasos de la creación de la configuración (solo se va a mostrar para este proyecto):

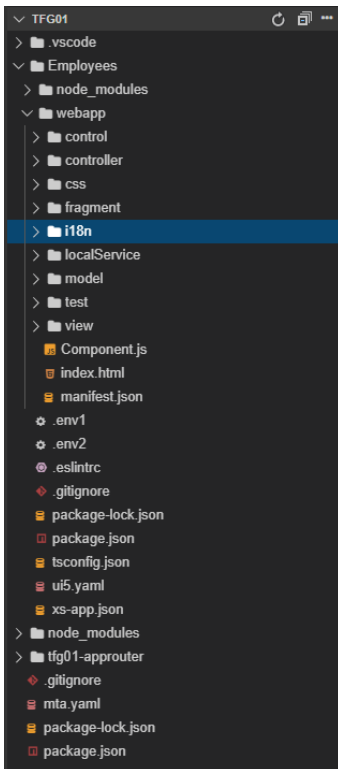


Una vez creado, hay que bindear la configuración con el servicio odata para poder utilizarlo:



6.2.1.1 Estructura de proyecto

Tenemos la siguiente estructura de datos del proyecto:



6.2.2 Proyecto HROLL

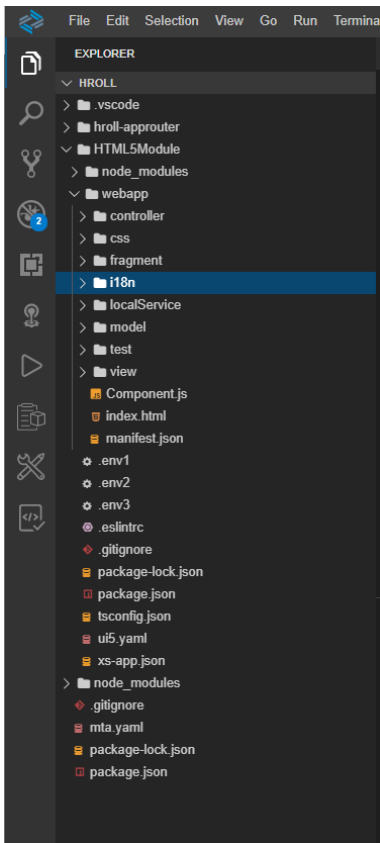
Se utiliza el terminal para crear el proyecto HROLL que es la parte donde creamos y vemos los empleados.


```
Problems 1 ~/projects x
user: projects $ yo fiori-project
? Select the target running environment Cloud Foundry
? Select the template you want to use SAPUI5 Application
? Enter a project name hroll
The project will be generated in the /home/user/projects/hroll directory.
? Select your HTML5 application runtime Standalone Approuter
? Enter an HTML5 module name HTML5Module
? Do you want to add authentication? No
? Enter a namespace hrolltfg
? Do you want to enable Karma tests? No
? Enter a view name App
? Do you want to add a data service? Yes
? Select a system My SAP systems
? Select a source NPL (System URL)
? Enter a path to the OData service (e.g. /sap/opu/odata/sap/SERVICE_NAME) /sap/opu/odata/SAP/ZEMPLOYEES_SRV/
create SAP-Config-1623226951223.json
create hroll/hroll-approuter/package.json
create hroll/hroll-approuter/xs-app.json
create hroll/hroll-approuter/.gitignore
create hroll/package.json
create hroll/HTML5Module/.eslintrc
create hroll/HTML5Module/package.json
create hroll/HTML5Module/tsconfig.json
create hroll/HTML5Module/ui5.yaml
create hroll/HTML5Module/xs-app.json
create hroll/HTML5Module/webapp/Component.js
create hroll/HTML5Module/webapp/index.html
create hroll/HTML5Module/webapp/manifest.json
create hroll/HTML5Module/webapp/css/style.css
create hroll/HTML5Module/webapp/i18n/i18n.properties
create hroll/HTML5Module/webapp/localService/mockserver.js
create hroll/HTML5Module/webapp/model/models.js
create hroll/HTML5Module/webapp/test/initMockServer.js
create hroll/HTML5Module/webapp/test/mockServer.html
create hroll/HTML5Module/webapp/test/integration/AllJourneys.js
create hroll/HTML5Module/webapp/test/integration/NavigationJourney.js
create hroll/HTML5Module/webapp/test/integration/opaTests.qunit.html
```

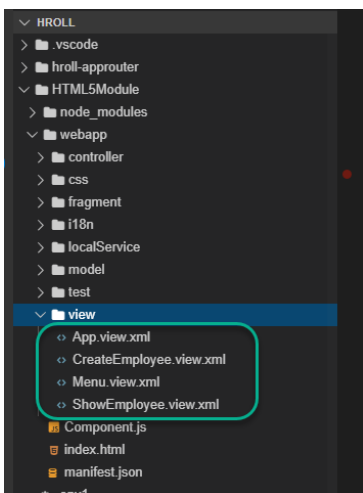
También se ha creado la configuración con el servicio oData NPL explicado en otro apartado.

6.2.2.1 Estructura de proyecto

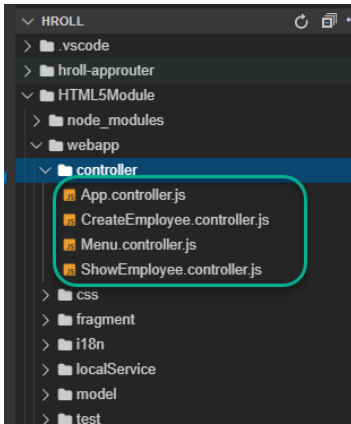
Tenemos la siguiente estructura:



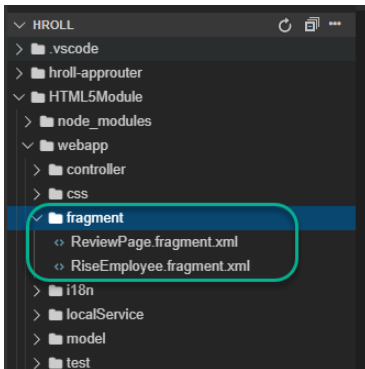
Tenemos las siguientes vistas para el alta de empleados, menú y para mostrar el detalle del empleado:



También creamos los controladores para las vistas de alta de empleados, menú y para el detalle del empleado:



Dentro del directorio “webapp” creamos la carpeta “fragmentos” con los siguientes componentes, que los usaremos para modularizar el código fuente que se va a utilizar en varios sitios.



Nota: Indicar que extractos del código fuente se van a adjuntar en los Anexos.

6.3 Pruebas unitarias / Guía de usuario

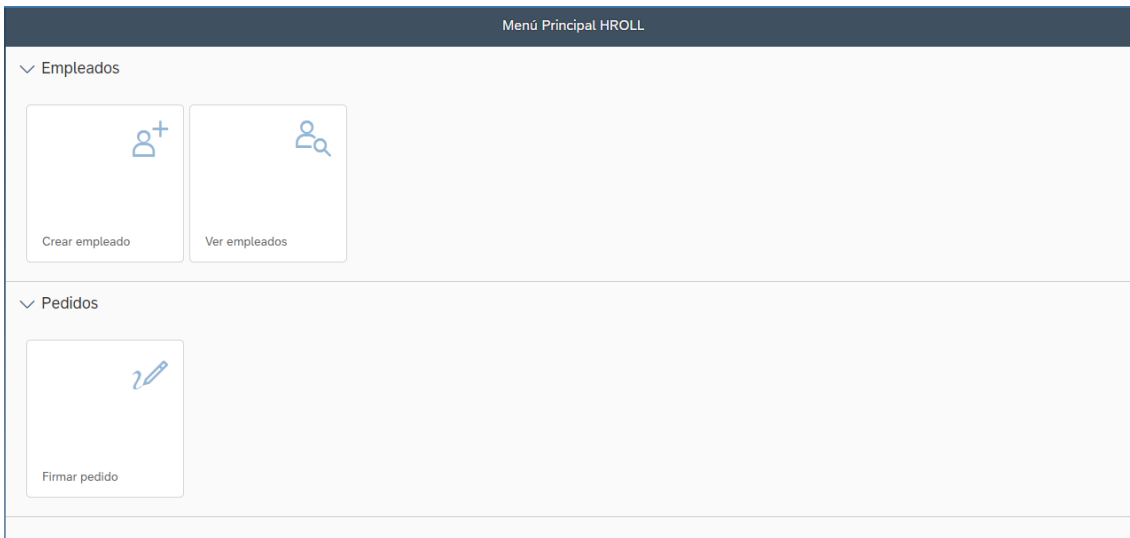
6.3.1 Menu principal

A continuación, se muestran los pantallazos de la aplicación que todavía está en la fase de análisis, pero teniendo en cuenta que tenía que hacer algún prototipo se ha empezado también la fase de implementación en la parte del código.

Todavía no está terminada la aplicación, por lo tanto, puede haber muchos cambios en la implementación final del desarrollo. De momento se muestra una idea de lo que será la aplicación final, pero puede que se cambien algunos detalles.

Hasta que no se entrega el documento final, la capa de aplicación puede sufrir cambios en la pantalla.

El menú principal de la aplicación tendría esta pantalla:

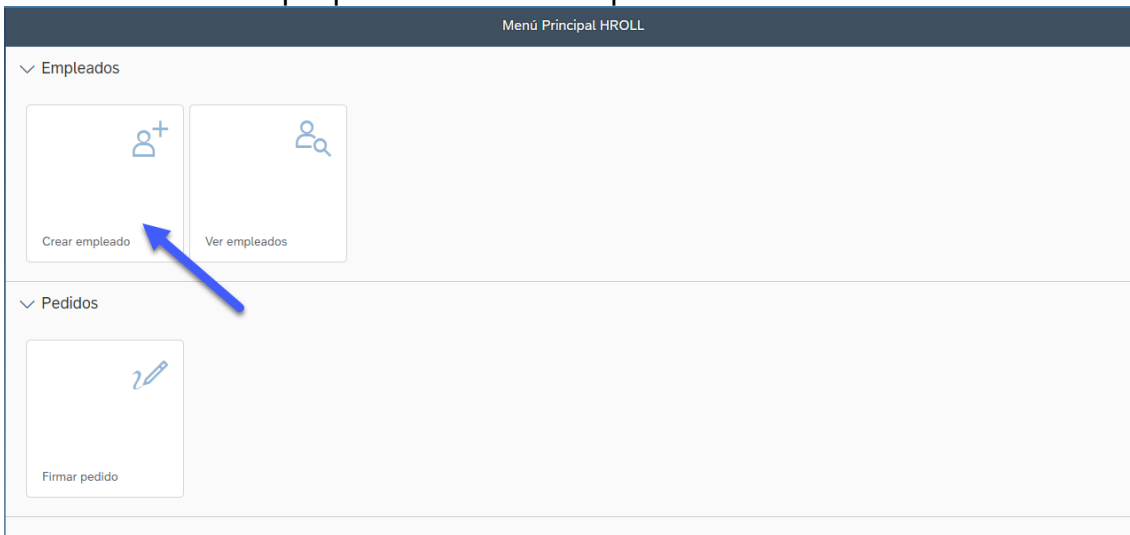


6.3.2 Crear empleado

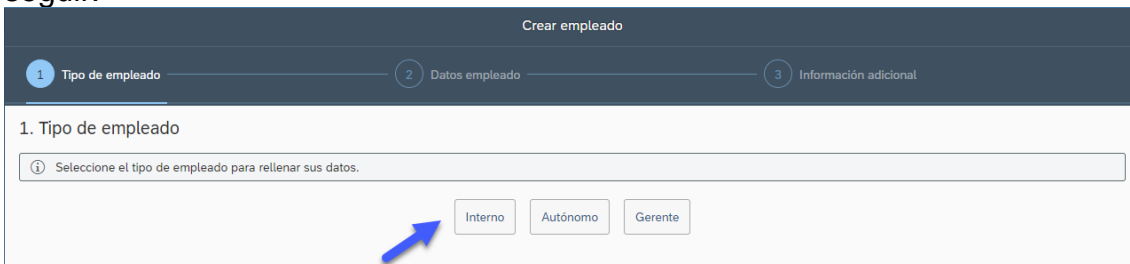
Primero se va a acceder a la creación de 3 usuarios diferentes:

- Interno (Juan Juanito)
- Autónomo (Ricardo Gomez)
- Gerente (Pedro Felipe)

Para ello tenemos que pulsar en Crear Empleado:



Se procede a crear el primer usuario (Juan Juanito). Estos son los pasos a seguir:



Tenemos que indicar el nombre, apellido, DNI (se realiza un chequeo de la validación del DNI), el salario bruto anual y la fecha de incorporación.

The screenshot shows the 'Crear empleado' form at Step 2, 'Datos empleado'. The progress bar at the top indicates three steps: 1. Tipo de empleado, 2. Datos empleado (current), and 3. Información adicional. A message box at the top of the form says 'Complete los datos obligatorios para poder continuar'. The form contains the following fields: 'Nombre:*' with the value 'Juan', 'Apellidos:*' with 'Juanito', 'DNI:*' with '82574934N', 'Salario Bruto anual:' with a slider set to '26000' (ranging from 12000 to 80000), and 'Fecha Incorporación:*' with '31 may. 2021'. A blue button labeled 'Paso 3' is located at the bottom left.

Pulsamos en Paso 3 para indicar si queremos añadir comentarios o si anexamos archivos. En este caso no lo hacemos, solamente para el Gerente vamos a adjuntar un PDF.

The screenshot shows the 'Crear empleado' form at Step 3, 'Información adicional'. The progress bar at the top indicates three steps: 1. Tipo de empleado, 2. Datos empleado, and 3. Información adicional (current). A message box at the top of the form says 'Introduce datos complementarios, si no dé al botón "Verificar"'. The form contains a 'Comentario' text area, an 'Anexos (0)' section with an 'Añadir' link, and a file upload area with a document icon and the text 'No existen archivos. Arrastrar ficheros para cargarlos'. A blue button labeled 'Revisar' is located at the bottom left.

Revisamos los datos del empleado y si todo está correcto guardamos y se realiza la creación del usuario.

Crear empleado

Resumen

1. Tipo de empleado

Tipo: Interno [Editar](#)

2. Datos empleado

Nombre: Juan
 Apellidos: Juanito
 DNI: 82574934N
 Salario Bruto anual: 26000
 Fecha Incorporación: 31 may, 2021 [Editar](#)

3. Información adicional

Comentario:
 Documentos: 0 Ficheros [Editar](#)

[Guardar](#) [Cancelar](#)

Realizamos la creación del usuario autónomo.

Crear empleado

1 Tipo de empleado — 2 Datos empleado — 3 Información adicional

1. Tipo de empleado

i Seleccione el tipo de empleado para rellenar sus datos.

Crear empleado

1 Tipo de empleado — 2 Datos empleado — 3 Información adicional

2. Datos empleado

i Complete los datos obligatorios para poder continuar

Nombre: * Ricardo

Apellidos: * Gomez

CIF: * 16904407M

Precio Diario:

Fecha Incorporación: * 4 may, 2021

Paso 3

Revisamos y creamos el empleado.

Pasamos a la creación del usuario gerente.

Crear empleado

1 Tipo de empleado 2 Datos empleado 3 Información adicional

1. Tipo de empleado

Seleccione el tipo de empleado para rellenar sus datos.

Interno Autónomo Gerente



Crear empleado

1 Tipo de empleado 2 Datos empleado 3 Información adicional

2. Datos empleado

Complete los datos obligatorios para poder continuar

Nombre: * Pedro

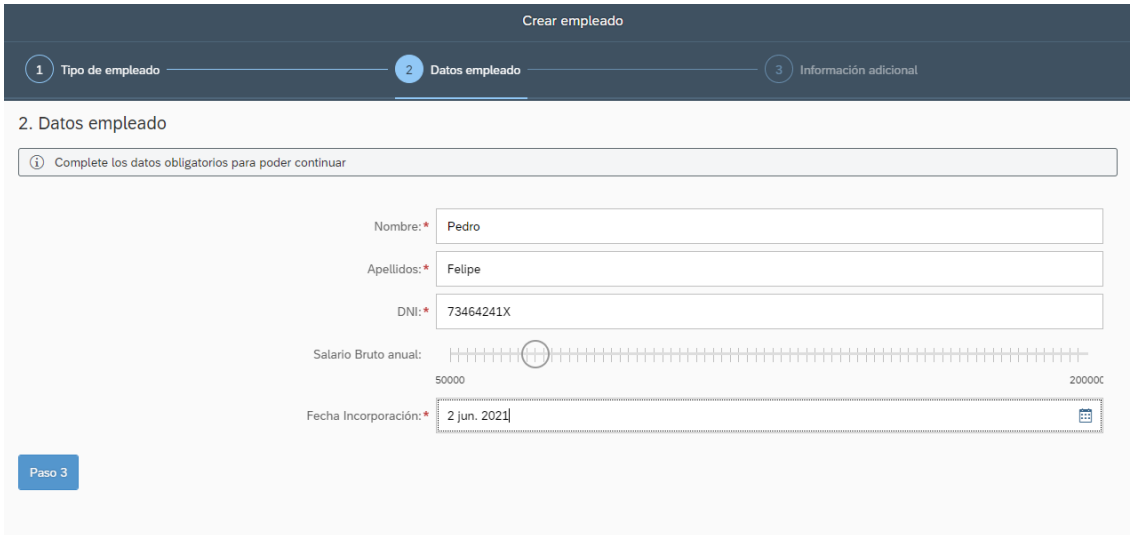
Apellidos: * Felipe

DNI: * 73464241X

Salario Bruto anual: 50000 20000€

Fecha Incorporación: * 2 jun. 2021

Paso 3



En este caso queremos añadir un comentario y añadir un PDF.

Crear empleado

1 Tipo de empleado 2 Datos empleado 3 Información adicional

3. Información adicional

Introduce datos complementarios, si no dé al botón "Verificar"

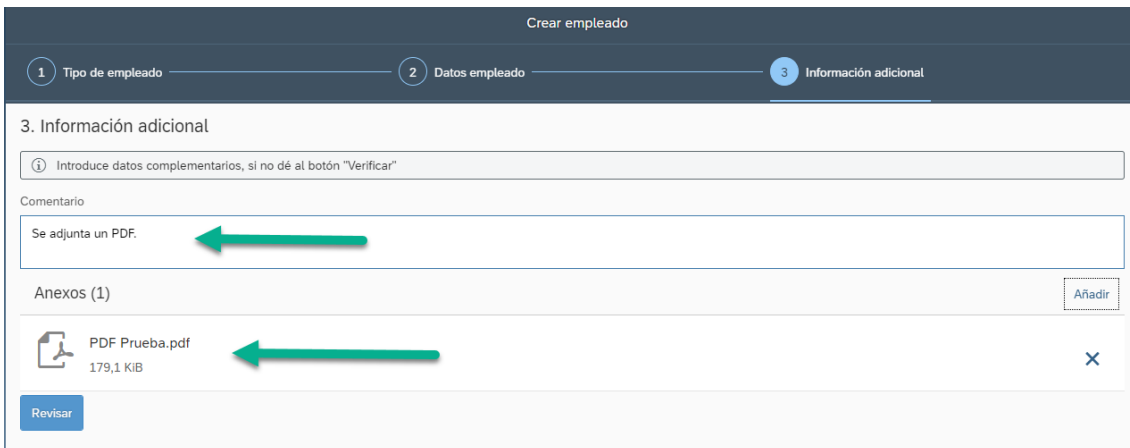
Comentario

Se adjunta un PDF.

Anexos (1) Añadir

PDF Prueba.pdf 179,1 KIB

Revisar



En el paso de la revisión de los datos observamos que se guardan bien los datos del empleado:

Resumen

1. Tipo de empleado

Tipo: Gerente Editar

2. Datos empleado

Nombre: Pedro
 Apellidos: Felipe
 DNI: 73464241X
 Salario Bruto anual: 70000
 Fecha Incorporación: 2 jun. 2021 Editar

3. Información adicional

Comentario: Se adjunta un PDF.
 Documentos: 1 Ficheros

PDF Prueba.pdf Editar

Observamos que se puede cancelar el proceso:



Al pulsar en la cancelación de un usuario, se muestra un pop-up de confirmación:

1 Tipo de empleado — 2 Datos empleado — 3 Información adicional

2. Datos empleado

i Complete los datos obligatorios para poder continuar

Nombre: * Juan
 Apellidos: * Juanilito
 CIF: * 91725915Y
 Precio Diario: 2000
 Fecha Incorporación: *

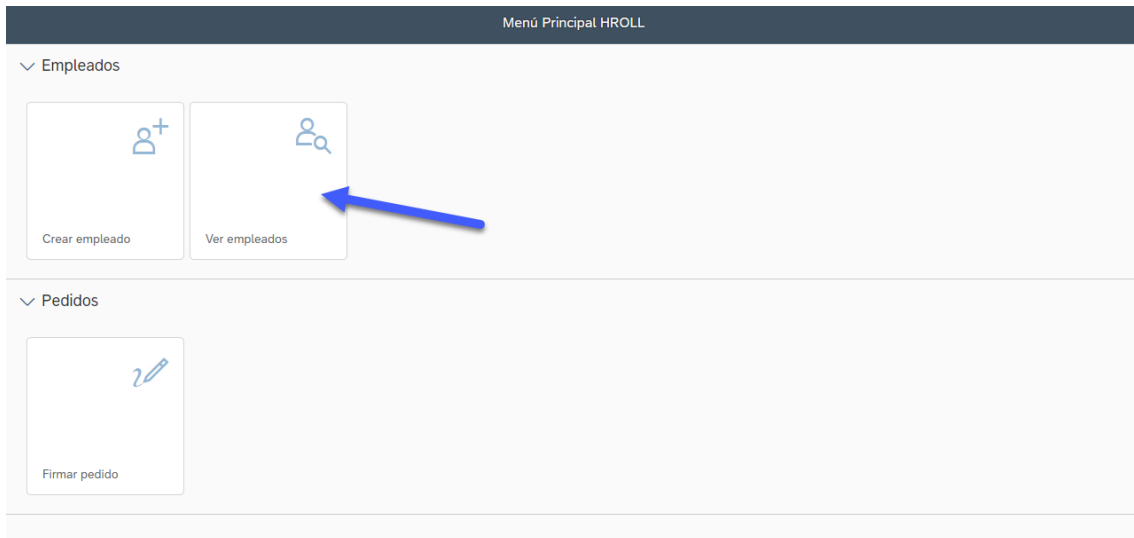
? Confirmación

¿Está seguro de cancelar la creación?

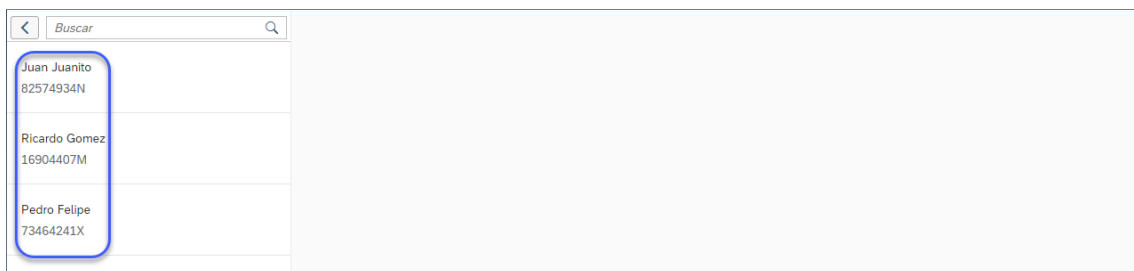
Paso 3

6.3.3 Información empleado

Pulsamos en Ver empleados, y tenemos que ver los 3 usuarios creados anteriormente:




Observamos que tenemos a los 3 empleados:





Si seleccionamos a Pedro Felipe, tenemos que tener los datos del empleado:


Detalle empleado

 **Pedro Felipe**
73464241X

Fecha Incorporación: 1 jun. 2021 Comentario: Se adjunta un PDF.
Tipo: Gerente

 Ficheros  Historial

Anexos (1) Cargar

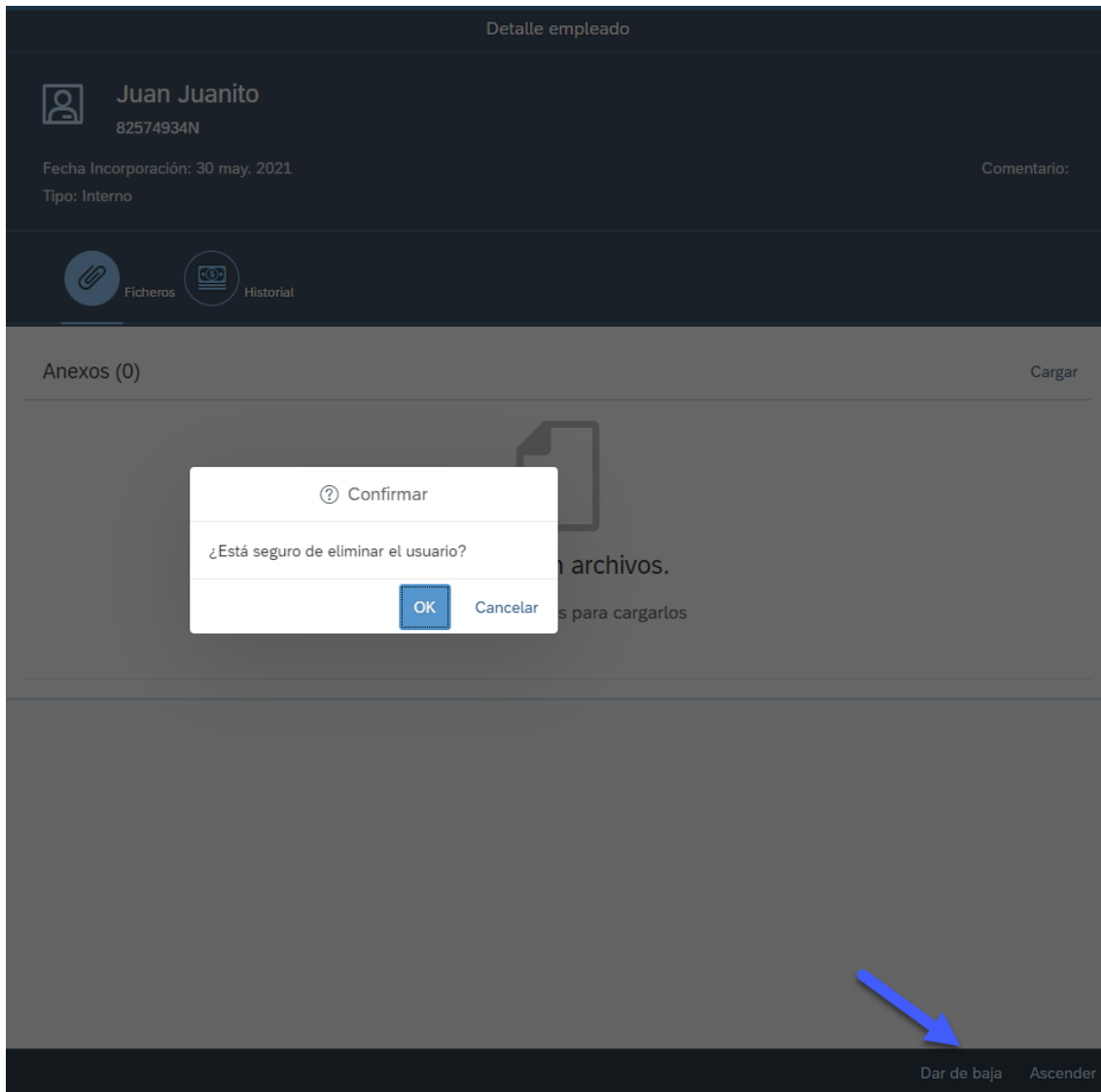
 PDF Prueba.pdf x

Dar de baja Ascender

Observamos que tenemos la opción de Dar de baja al usuario o de ascenderlo, en la parte inferior de la página.
Vamos a ver como funcionan los botones.

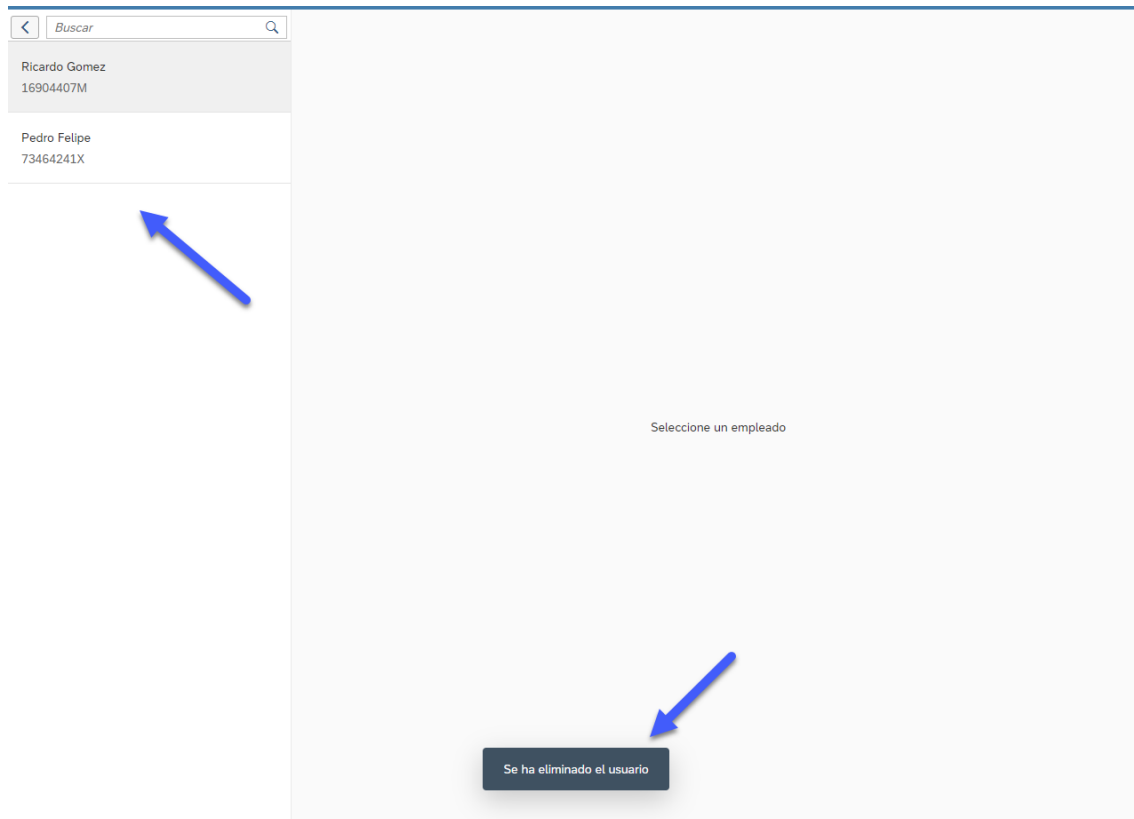
6.3.4 Dar de baja empleado

Vamos a dar de baja a Juan:
Entramos en los detalles de Juan Juanito:



Sale un mensaje de eliminación por lo tanto le damos OK y vamos a ver que el empleado se borra de la base de datos y de la aplicación.

Observamos que se ha eliminado el usuario y ya no aparece en la lista.




6.3.5 Ascender empleado



Vamos a ascender a Ricardo:

Al ascenderlo hay que añadir el nuevo salario, la fecha y algún comentario opcional:

Detalle empleado

 **Ricardo Gomez**
16904407M


Fecha Incorporación: 3 may. 2021 Comentario:
Tipo: Autónomo

 Ficheros  Historial

Anexos (0) Cargar

Nuevo ascenso


Salario:

Fecha:
 



Comentario:

[Aceptar](#) [Cancelar](#)

Detalle empleado

 **Ricardo Gomez**
16904407M


Fecha Incorporación: 3 may. 2021 Comentario:
Tipo: Autónomo

 Ficheros  Historial

Anexos (0) Cargar

Nuevo ascenso

Salario:

Fecha:
 

Comentario:

[Aceptar](#) [Cancelar](#)

6.3.6 Ver historial empleado

Podemos ver el historial del empleado y si tiene adjuntados ficheros. En este caso el empleado Ricardo no tiene ficheros, miramos el historial del ascenso:

Detalle empleado

Ricardo Gomez
16904407M

Fecha Incorporación: 3 may. 2021 Comentario:
Tipo: Autónomo

Ficheros Historial

2021

500,00 3/5/21 a las 2:00	30.000,00 6/6/21 a las 2:00 Buen trabajo.
31.000,00 Ayer a las 2:00	30.000,00 1/6/21 a las 2:00 Buen trabajo.

6.3.7 Firmar pedido Ventas

Esta es la opción que tenemos que dar para entrar dentro del Menu.

Menú Principal HROLL

Empleados

- Crear empleado
- Ver empleados

Pedidos

- Firmar pedido

Para la segunda parte de la app, se ha creado en el Menu, una opción para firmar los pedidos de los usuarios del departamento de Ventas. Para ello se tiene que pulsar en Firmar Pedido y tendremos la siguiente vista:

Pedidos de los empleados Departamentos de Ventas HROLL

ID Empleado Pais

ID Empleado	Nombre	Pais	Mostrar De...
1	Davolio, Nancy	USA	>
2	Fuller, Andrew	USA	>
3	Leverling, Janet	USA	>
4	Peacock, Margaret	USA	>
5	Buchanan, Steven	UK	>
6	Suyama, Michael	UK	>
7	King, Robert	UK	>
8	Callahan, Laura	USA	>
9	Dodsworth, Anne	UK	>

Layout

6.3.7.1 Ver usuarios departamento de ventas

Observamos que tenemos en la cabecera la opción de poner el ID del empleado y de añadir filtros.

En la parte inferior tenemos varias formas para ver el informe. Esta parte se verá más detallada en el video de presentación cuando lleguemos a este punto.

Al pulsar por ejemplo sobre el usuario Nancy, Davolio, se nos abre una vista con los pedidos y los detalles del usuario:

Pedidos de los empleados Departamentos de Ventas H...

Detalles Empleado

Ms. Nancy Davolio

ID Empleado 1 Título Sales Representative

Código Pais USA Ciudad Seattle Código postal 98122

Incidencia United States [Crear Incidencia](#)

Pedidos

ID Pedido	Transporte	Dirección de envío
10258	140.5100	Kirchgasse 6
10270	136.5400	Torikatu 38
10275	26.9300	Via Ludovico il Moro 22
10285	76.8300	Taucherstraße 10
10292	1.3500	Av. Inês de Castro, 414
10293	21.1800	Avda. Azteca 123
10304	63.7900	Avda. Azteca 123
10306	7.5600	Gran Vía, 1
10311	24.6900	67, rue des Cinquante Otages
10314	74.1600	2817 Milton Dr.

Layout Two Columns - Mid Expanded

Observamos que en la parte derecha tenemos los detalles como el nombre, id, código país, título y código postal.

6.3.7.2 Crear incidencia

En la parte inferior tenemos la opción de crear incidencia por si el usuario tiene alguna dificultad o tarea pendiente, podemos indicar cualquier cosa.

Hay 3 tipos de incidencias que se van a ver a continuación. Vamos a crear para el usuario, los 3 tipos, pulsando “Crear Incidencia”:

Incidencia United States [Crear Incidencia](#)

Incidencia

paso 1 Fecha* 1 jun. 2021

paso 2 Motivo* Pendiente firmar

paso 3 Tipo Pending

Correct

Pending

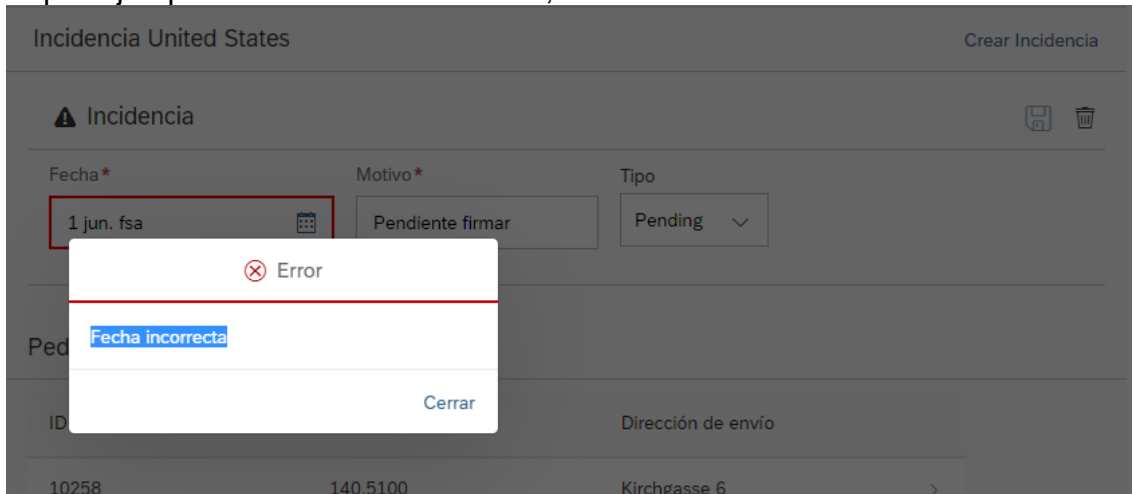
Error

Pedidos

Una vez creada la incidencia, tenemos que pulsar el boton Guardar. Si queremos eliminar, pulsamos el boton Eliminar que tenemos en la parte derecha.

El boton Guardar, solamente se habilita cuando hay algún cambio en alguna incidencia, y si el campo fecha está correctamente indicado, y si hay valor en el campo motivo.

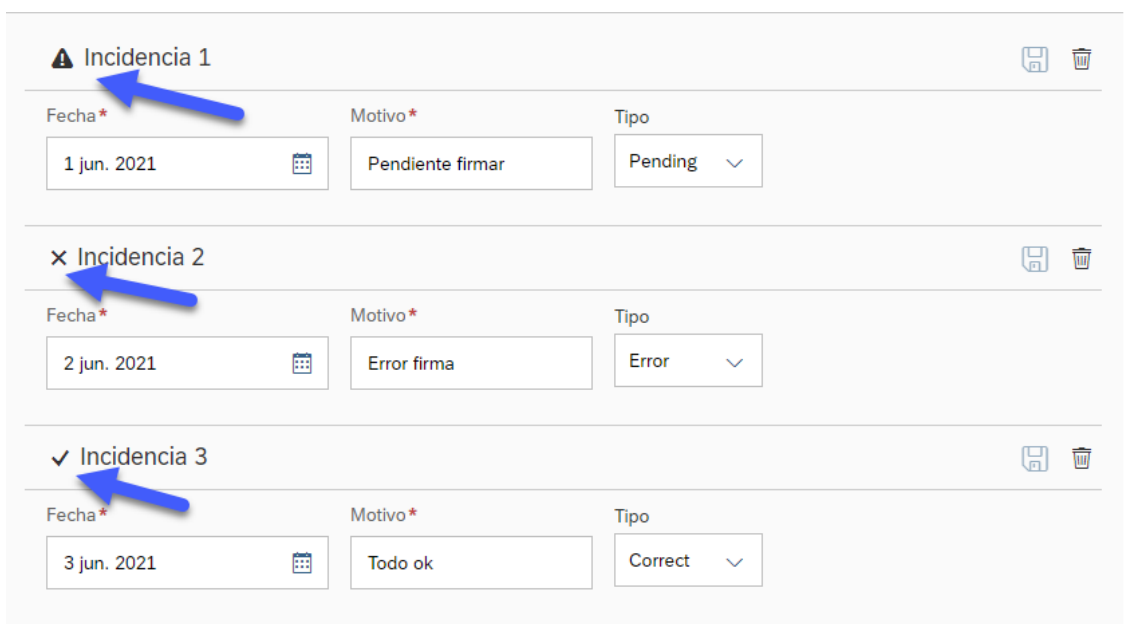
Si por ejemplo indicamos mal la fecha, tenemos error:



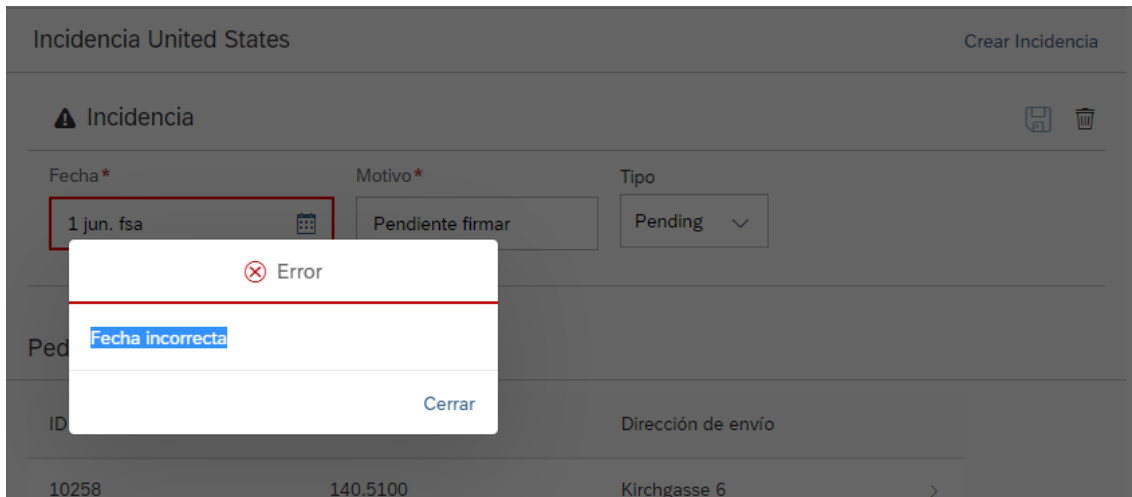
Creamos 3 incidencias:

- Pendiente firmar
- Todo ok
- Error firma

Observamos los 3 tipos de incidencia con el logo correspondiente:



Si todo está correcto se muestra el mensaje de confirmación.



6.3.7.3 Detalles pedido

El siguiente paso sería ir al primer pedido, navegar pulsando click y realizar la firma del pedido:

Pedidos			
ID Pedido	Transporte	Dirección de envío	
10258	140.5100	Kirchgasse 6	>
10270	136.5400	Torikatu 38	>
10275	26.9300	Via Ludovico il Moro 22	>

Al pulsar, vamos a ver la siguiente vista donde tenemos detalles del cliente y del pedido en la parte izquierda:

- Fecha pedido
- ID Cliente
- Nombre envío
- Dirección de envío

6.3.7.4 Firmar pedido

En la parte derecha vamos a poder realizar la firma tanto desde el movil como desde cualquier dispositivo o PC/Portátil etc.

También tenemos la parte de Anexos si queremos añadir algún documento con información.

En la parte inferior tenemos los productos del pedido y el stock disponible.

Una vez realizados los cambios, tenemos que guardar, y al refrescar o volver a entrar dentro del pedido, se queda guardada la firma y el anexo.

The screenshot displays a web interface for an order with ID 10258. The top header shows a back arrow and the text 'ID Pedido 10258'. The main content area is divided into several sections:

- Order Details:** A box on the left contains the following information:
 - ID Cliente: ERNSH
 - Fecha pedido: 17 jul. 1996
 - Transporte: 140.5100
 - Nombre envío: Ernst Handel
 - Dirección de envío: Kirchgasse 6, Graz, Austria
- Signature Area:** A large empty rectangular box on the right, with a 'Limpiar' button next to it.
- Attachments:** A section labeled 'Anexos (0)' with a 'Cargar' button.
- File Upload:** A central area with a document icon, the text 'No existen archivos.', and the instruction 'Suelte aquí los archivos para cargarlos o utilice el botón "Cargar".'
- Products Table:** A table listing items with their quantities, stock status, and prices.

Productos		
Chang (50)	Stock disponible 17	15,20
Chef Anton's Gumbo Mix (65)	Stock disponible 0	17,00
Mascarpone Fabioli (6)		25,60

The bottom right corner of the interface features a 'Guardar' button.

6.3.7.5 Adjuntar PDF pedido

Realizamos una firma virtual y añadimos un PDF de prueba. El archivo a cargar se carga desde local, igual que en el caso de los empleados, al a hora de crear el usuario.

ID Pedido 10258

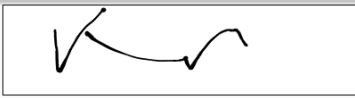
ID Cliente: ERNSH

Fecha pedido:
17 jul. 1996


Transporte:
140,5100

Nombre envío:
Ernst Handel

Dirección de envío:
Kirchgasse 6, Graz, Austria


Limpiar

Anexos (1) Cargar


PDF Prueba.pdf
×

Productos

Chang (50)	Stock disponible 17	15,20
Chef Anton's Gumbo Mix (65)	Stock disponible 0	17,00
Mascarpone Fabioli (6)		25,60

Guardar

Guardamos los cambios:

i Información

Firma guardada

OK

Volvemos a entrar al pedido para ver si se ha guardado la firma y el PDF adjuntado y observamos que si se han guardado los cambios:

← ID Pedido 10258

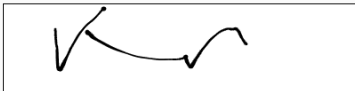
ID Cliente: ERNSH

Fecha pedido:
17 jul. 1996


Transporte:
140.5100

Nombre envío:
Ernst Handel

Dirección de envío:
Kirchgasse 6, Graz, Austria



Anexos (1) Cargar

 PDF Prueba.pdf ×

Productos

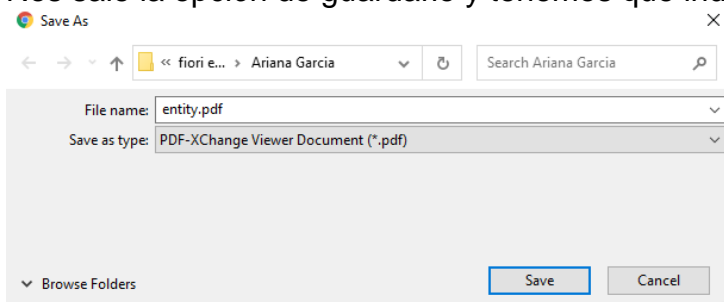
Chang (50)	Stock disponible 17	15,20
Chef Anton's Gumbo Mix (65)	Stock disponible 0	17,00
Mascarpone Fabioli (6)		25,60

6.3.7.6 Descargar fichero pedido

Observamos que tenemos habilitada la opción de descargar los ficheros que se adjuntan para los diferentes pedidos.

Pulsamos sobre el archivo:

Nos sale la opción de guardarlo y tenemos que indicar dónde:



Lo guardamos y lo abrimos y funciona correctamente.

6.3.7.7 Layout pedidos

Con el layout de la parte inferior de la página, se pueden mostrar los datos de varias formas, indicaremos a continuación 2 formas (begin expanded end hidden y mid expanded end hidden), pero esta parte también se verá en el video de presetación de la aplicación, más en detalle.

Begin expanded end hidden:

Pedidos de los empleados Departamentos de Ventas HROLL				Detalles Empleado	
ID Empleado	<input type="text"/>	País	<input type="text"/>	<input type="text"/>	<input type="text"/>
Mostrar Ciudad					
ID Empleado	Nombre	País	Mostrar De...		
1	Davolio, Nancy	USA		>	
2	Fuller, Andrew	USA		>	
3	Leverling, Janet	USA		>	
4	Peacock, Margaret	USA		>	
5	Buchanan, Steven	UK		>	
6	Suyama, Michael	UK		>	
7	King, Robert	UK		>	
8	Callahan, Laura	USA		>	
9	Dodsworth, Anne	UK		>	

ID Empleado	Salario
10529	11.0000
10537	78.8500
10542	10.9500
10546	194.7200
10558	72.9700
10562	22.9500
10567	33.9700
10579	13.7300
10587	62.5200
10591	55.9200
10598	44.4200
10604	7.4600
10605	379.1300
10612	544.0800
10616	116.5300
10618	154.6800
10626	138.6900
10630	32.3500

Layout: Three Columns - Begin Expanded End Hidden

Mid expanded end hidden:

Pedidos de los empleados Departamentos de Ventas H...				Detalles Empleado	
ID Empleado	<input type="text"/>	País	<input type="text"/>	<input type="text"/>	<input type="text"/>
Mostrar Ciudad					
ID Em...	Nombre	País	Mostrar De...		
1	Davolio...	USA		>	
2	Fuller, ...	USA		>	
3	Leverlin...	USA		>	
4	Peacoc...	USA		>	
5	Buchan...	UK		>	
6	Suyam...	UK		>	
7	King, R...	UK		>	
8	Callaha...	USA		>	
9	Dodsw...	UK		>	

ID Empleado	Salario	Dirección
10529	11.0000	11, rue des Bouchers
10537	78.8500	Starenweg 5
10542	10.9500	Maubelstr. 90
10546	194.7200	2, rue du Commerce
10558	72.9700	Brook Farm Stratford St. Mary
10562	22.9500	Strada Provinciale 124
10567	33.9700	8 Johnstown Road
10579	13.7300	87 Polk St. Suite 5
10587	62.5200	Rua da Panificadora, 12
10591	55.9200	Smagsloget 45
10598	44.4200	2817 Milton Dr.
10604	7.4600	Jardim das rosas n. 32
10605	379.1300	43 rue St. Laurent
10612	544.0800	187 Suffolk Ln.
10616	116.5300	2732 Baker Blvd.
10618	154.6800	43 rue St. Laurent
10626	138.6900	Berguvsvägen 8
10630	32.3500	Maubelstr. 90

Layout: Three Columns - Mid Expanded End Hidden

7. Lanzamiento

Una vez terminadas las aplicaciones, las construcciones y las fase de desarrollos y pruebas, tenemos que hacer el despliegue de las aplicaciones para poder usarlas vía web.

Para ver donde tenemos las aplicaciones lanzadas, tenemos que entrar en nuestro espacio de SAP BTP Cockpit.

Observamos que de momento no las tenemos dentro del espacio dev.

The screenshot shows the SAP BTP Cockpit interface. At the top, it displays 'Subaccount: trial' and 'Delete Subaccount'. Below this, it lists 'Subaccount ID: 0082458e-1db3-40a6-b455-9d21ab6360cd', 'Tenant ID: 0082458e-1db3-40a6-b455-9d21ab6360cd', and 'Subdomain: 5e15bc7etrial'. There is a '1 (16 available) Active Subscriptions' indicator. The main section is titled 'Cloud Foundry Environment' and shows 'Org Name: 5e15bc7etrial', 'API Endpoint: https://api.cf.eu10.hana.ondemand.com', and 'Org ID: 24ed0b98-54b5-4ff1-b276-afb863eb9d1c'. On the right, a 'Spaces (1)' table is visible:

Name	Applications	Service Instances
dev	0	3

7.1 Configuración app web

Entramos en las 2 aplicaciones y modificamos el ficheros “xs-app.json” añadiendo el “welcomeFile” con el name space del proyecto.

The screenshot shows a file explorer on the left with 'xs-app.json' selected. The code editor on the right shows the following JSON configuration:

```
1 {
2   "authenticationMethod": "none",
3   "welcomeFiles": "hrolltfgEmployees",
4   "routes": []
5 }
```

Tenemos que tener en cuenta los recursos que se van a utilizar. Tenemos que entrar en la parte de manifest.json para cambiar algunas configuraciones.

```

17  },
18  "dataSources": {
19    "Northwind.svc": {
20      "uri": "Northwind/v2/Northwind/Northwind.svc/",
21      "type": "OData",
22      "settings": {
23        "odataVersion": "2.0",
24        "localUri": "localService/metadata.xml"
25      }
26    },
27    "YSAPUI5_SRV_01": {
28      "uri": "sap/opu/odata/sap/YSAPUI5_SRV_01/",
29      "type": "OData",
30      "settings": {
31        "odataVersion": "2.0",
32        "localUri": "localService/YSAPUI5_SRV_01/metadata.xml"
33      }
34    }
35  },
36  "sap.ui": {
37    "technology": "UI5",
38  }

```

Tenemos que copiar los componentes de la carpeta localService, porque es una carpeta de test, que no se va a transportar en el entorno productivo. Necesitamos los 3 componentes para la construcción del proyecto.

Los copiamos en la carpeta model porque esta carpeta si que se va a transportar.

En la carpeta model creamos otra carpeta json y los guardamos ahí.

```

localService
├── mockdata
├── YSAPUI5_SRV_01
│   ├── metadata.xml
│   ├── mockserver.js
│   └── test
├── model
│   └── json
│       ├── Countries.json
│       ├── Employees.json
│       └── Layout.json
├── formatter.js
├── models.js
└── test

```

Modificamos el manifest.json añadiendo esta parte de código:

```

"jsonLayout": {
  "uri": "model/json/Layout.json",
  "type": "JSON"
}

```

También modificamos el controlador Main.controller.js


```

14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
var oView = this.getView();

// @ts-ignore
var oJSONModelEmpl = new sap.ui.model.json.JSONModel();
oJSONModelEmpl.loadData("./model/json/Employees.json", false);
oView.setModel(oJSONModelEmpl, "jsonEmployees");

var oJSONModelCountries = new sap.ui.model.json.JSONModel();
oJSONModelCountries.loadData("./model/json/Countries.json", false);
oView.setModel(oJSONModelCountries, "jsonCountries");

var oJSONModelLayout = new sap.ui.model.json.JSONModel();
oJSONModelLayout.loadData("./model/json/Layout.json", false);
oView.setModel(oJSONModelLayout, "jsonLayout");

var oJSONModelConfig = new sap.ui.model.json.JSONModel({
  visibleID: true,
  visibleName: true,
  visibleCountry: true.

```

Tenemos que construir los recursos para desplegar la aplicación con MTA – Build.

Tenemos el fichero mta.yaml que va a crear la aplicación:

```

1  _schema-version:
2  ID: tfg01
3  version: 0.0.1
4  modules:
5  - name: tfg01-app
6    type: approuter
7    path: tfg01-app
8  - name: tfg01-ui5
9    type: com.sap.application
10   path: .
11   requires:
12     - name: tfg01-app
13     parameters:
14       disk-quota: 200M
15       memory: 256M
16   build-parameters:
17     content-target-path: /
18     build-parameters:
19       content-target-path: /
20     build-parameters:
21       content-target-path: /

```

Se crea la configuración para el despliegue del proyecto:

```

Terminal will be reused by tasks.
> Executing task: mbt build -s '/home/user/projects/tfg01'; sleep 2; <
[2021-06-09 19:22:23] INFO Cloud MTA Build Tool version 1.1.1
[2021-06-09 19:22:23] INFO generating the "Makefile_20210609192223.mta" file...
[2021-06-09 19:22:23] INFO done
[2021-06-09 19:22:23] INFO executing the "make -f Makefile_20210609192223.mta p=cf mtar= strict=true mode=" command...
[2021-06-09 19:22:23] INFO validating the MTA project
[2021-06-09 19:22:23] INFO running the "before-all" build...
[2021-06-09 19:22:23] INFO executing the "npm install" command...
.....npm WARN tfg01@0.0.1 No description
npm WARN tfg01@0.0.1 No repository field.
npm WARN tfg01@0.0.1 No license field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.1.3 (node_modules/@ui5/cli/node_modules/fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@https://registry.npmjs.org/fsevents/-/fsevents-2.1.3.tgz: wanted {"os":"darwin","arch":"any"} (current: {"os":"linux","arch":"x64"})

up to date in 5.31s

50 packages are looking for funding
  run `npm fund` for details

```

Para hacer el despliegue tenemos que ir a la carpeta mta_archives. Tenemos que estar conectados a la cuenta trial.

```
user: tfg01 $ cf logout
Logging out rol4jc@gmail.com...
OK

user: tfg01 $ cf login
API endpoint: https://api.cf.eu10.hana.ondemand.com

Email: rol4jc@gmail.com
Password:

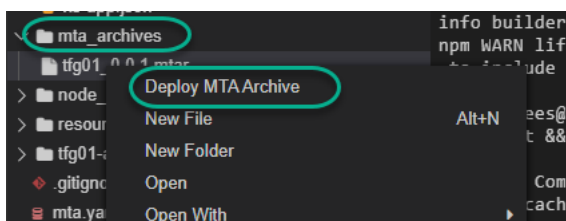
Authenticating...
OK

Targeted org 5e15bc7etrial.

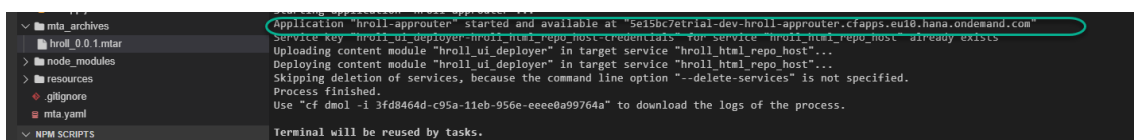
Targeted space dev.

API endpoint: https://api.cf.eu10.hana.ondemand.com
API version: 3.100.0
user: rol4jc@gmail.com
org: 5e15bc7etrial
space: dev
user: tfg01 $
```

Realizamos el Deploy MTA Archive:



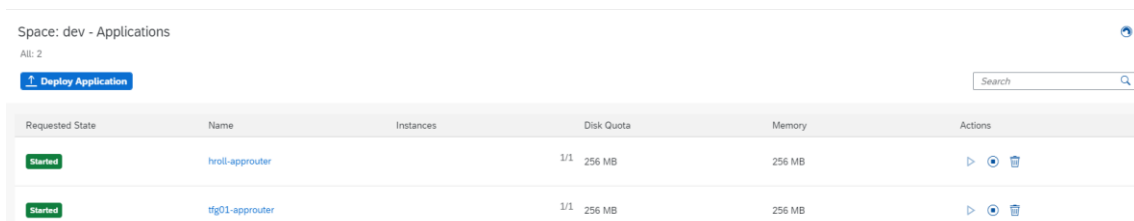
Entramos en el segundo proyecto y hacemos lo mismo.



7.2 Lanzamiento app web

Observamos que ya tenemos la app en nuestro espacio de trabajo y ya se pueden usar.

Ya tenemos las 2 aplicaciones en despliegue:



Requested State	Name	Instances	Disk Quota	Memory	Actions
Started	hroll-approuter	1/1	256 MB	256 MB	▶ ⌂ 🗑
Started	tfg01-approuter	1/1	256 MB	256 MB	▶ ⌂ 🗑

NOTA para cuando el profesor prueba el código: Para que funciones la app, en el espacio dev tienen que ser activas las 2 configuraciones. Al ser una cuenta trial, cuando no se usa dicha ejecución, se queda en stand-by y ya nos dará un error de ejecución.

8. Cambios respecto a la planificación inicial

No se han realizado muchos cambios respecto a la planificación inicial, pero a continuación indico los pocos cambios que se realizaron.

En la planificación inicial se han realizado para la etapa de “Análisis y Diseño” las siguientes tareas:

- **Análisis y Diseño**

- Estudio de la usabilidad – esta tarea será analizada y documentada en el documento final de fin de grado porque contendrá todo el uso de la aplicación, paso a paso para los usuarios finales.
- Estudio de la arquitectura de la aplicación - no ha sufrido cambios respecto a la planificación inicial.
- Casos de uso –no ha sufrido cambios respecto a la planificación inicial.
- Diseño de base de datos – esta tarea no se ha realizado como hemos indicado en la planificación inicial porque se cambia para realizarse a la hora de la implementación final del código.
- Documentación – esta tarea la tenemos en todos los pecs, porque se trata de la documentación final del TFG. Esta PEC sirve para ayuda en la realización del documento final, que se entregara cuando está programado el trabajo fin de grado.
- Prototipos – esta tarea se ha actualizado, se han añadido los prototipos de la aplicación, pero todavía faltan algunos porque se han programado para realizarse a la hora de la implementación final del código.

9. Objetivos conseguidos

En primer lugar, se ha conseguido hacer el resumen del proyecto incluyendo la introducción y los objetivos del desarrollo actual, teniendo toda esta información en este documento.

También se ha podido analizar la arquitectura y todas las tecnologías que se van a utilizar a lo largo del diseño y la implementación del desarrollo y de la creación de la aplicación.

Se ha realizado una planificación temporal que no ha sufrido muchos cambios y se han evaluado todos los posibles riesgos que se pueden dar a la hora de la creación de la aplicación.

Se ha podido realizar en la segunda fase, el análisis y el diseño incluyendo puntos como el estudio de la usabilidad de la aplicación, estudio de la arquitectura de la aplicación, los casos de uso y los prototipos.

También se trabaja sobre la documentación final del TFG y en la siguiente fase se trabajará más sobre el código fuente, la parte de back-end, Font-end y trabajar con las bases de datos de SAPUI5.

Se han mencionado los cambios respecto a la planificación inicial. Toca la fase de la implementación para poder ir terminando la aplicación final.

10. Conclusiones

Finalmente se ha concluido la creación del proyecto en los parámetros de tiempo indicados en la planificación.

Se ha podido crear una aplicación amigable y muy fácil de usar por los usuarios finales, sin necesidades de formación previa ya que es bastante fácil para usar.

Los usuarios que tienen perfiles de creador, en los roles, y los del departamento de HROLL podrán crear empleados y también modificar los usuarios existentes.

Van a poder usar la opción de dar de baja a usuarios o también la opción de ascender, en el caso en el cual algún empleado merece una mejora salarial.

Los usuarios del departamento de Ventas ya tienen la opción que necesitaban para firmar los pedidos sea desde un móvil, Tablet o cualquier PC o portátil, teniendo los eventos necesarios para la firma.

También podrán crear incidencias y ver todos los pedidos de todos los clientes del servicio oData.

Se pueden adjuntar archivos correspondientes a los pedidos y la firma se queda guardada dentro del pedido del empleado indicado.

11. Anexos

Anexo 1. Entregables del proyecto

Se van a entregar los siguientes archivos con los siguientes nombres:

RAUL_IOZSEF_VINCZE_TFG.pdf

RAUL_IOZSEF_VINCZE_TFG.pptx

Se ha realizado el video de presentación en la herramienta de UOC.

Anexo 2. Código fuente (extractos)

A continuación, se van a mostrar algunos extractos de código de la app principal de la creación de usuarios.

Controladores:

App.Controller.js

```
App.controller.js x
1  sap.ui.define([
2      "sap/ui/core/mvc/Controller"
3  ],
4  /**
5   * @param {typeof sap.ui.core.mvc.Controller} Controller
6   */
7  function (Controller) {
8      "use strict";
9
10     return Controller.extend("hrolltfg.hroll.HTML5Module.controller.App", {
11         onInit: function () {
12             }
13     });
14 });
15
16
```

CreateEmployee.controller.js – función para validar el dni.

```
App.controller.js x
1  sap.ui.define([
2      "sap/ui/core/mvc/Controller"
3  ],
4  /**
5   * @param {typeof sap.ui.core.mvc.Controller} Controller
6   */
7  function (Controller) {
8      "use strict";
9
10     return Controller.extend("hrolltfg.hroll.HTML5Module.controller.App", {
11         onInit: function () {
12             }
13     });
14 });
15
16
```

Menu.controller.js

```

App.controller.js  CreateEmployee.controller.js  Menu.controller.js x
6      function onInit(){
7
8      }
9
10     function onAfterRendering(){
11         // Erro en el framework : Al agregar la dirección URL de "Firmar pedidos", el componente GenericTile debería navegar directamente a dicha URL,
12         // pero no funciona en la versión 1.78. Por tanto, una solución encontrada es eliminando la propiedad id del componente por jquery
13         var genericTileFirmarPedido = this.byId("linkFirmarPedido");
14         //Id del dom
15         var idGenericTileFirmarPedido = genericTileFirmarPedido.getId();
16         //Se vacía el id
17         jQuery("#"+idGenericTileFirmarPedido)[0].id = "";
18     }
19
20     //Función al pulsar sobre el Tile "Crear empleado". Hace un routing a la vista "createEmployee"
21     function navToCreateEmployee(){
22         //Se obtiene el conjunto de routers del programa
23         var oRouter = sap.ui.core.UIComponent.getRouterFor(this);
24         //Se navega hacia el router "CreateEmployee"
25         oRouter.navTo("CreateEmployee", {}, false);
26     }
27
28     //Función al pulsar sobre el Tile "Ver empleados". Hace un routing a la vista "showEmployee"
29     function navToShowEmployee(){
30         //Se obtiene el conjunto de routers del programa
31         var oRouter = sap.ui.core.UIComponent.getRouterFor(this);
32         //Se navega hacia el router "CreateEmployee"
33         oRouter.navTo("ShowEmployee", {}, false);
34     }
35
36     return Controller.extend("hrolltfg.hroll.controller.Menu", {
37         onInit: onInit,
38         onAfterRendering : onAfterRendering,
39         navToCreateEmployee : navToCreateEmployee,
40         navToShowEmployee : navToShowEmployee
41     });
42 });
43

```

ShowEmployee.controller.js – seleccionar un empleado

```

App.controller.js  CreateEmployee.controller.js  Menu.controller.js  ShowEmployee.controller.js x
1  sap.ui.define([
2      "sap/ui/core/mvc/Controller",
3      "sap/ui/core/routing/History"
4  ], function (Controller, History) {
5      "use strict";
6
7      function onInit(){
8          this._splitAppEmployee = this.byId("splitAppEmployee");
9      }
10
11     //Función al pulsar "<" para regresar al menú
12     function onPressBack(){
13         // vamos al menú
14         var oRouter = sap.ui.core.UIComponent.getRouterFor(this);
15         oRouter.navTo("menu", {}, true);
16     }
17
18     //Función para filtrar empleados
19     function onSearchEmployee(){
20
21     }
22
23     //Función al seleccionar un empleado
24     function onSelectEmployee(oEvent){
25         //Se navega al detalle del empleado
26         this._splitAppEmployee.to(this.createId("detailEmployee"));
27         var context = oEvent.getParameter("listItem").getBindingContext("odataModel");
28         //Se almacena el usuario seleccionado
29         this.employeeId = context.getProperty("EmployeeId");
30         var detailEmployee = this.byId("detailEmployee");
31         //Se bindea a la vista con la entidad Users y las claves del id del empleado y el id del alumno
32         detailEmployee.bindElement("odataModel>Users(EmployeeId="+ this.employeeId +"', SapId='"+this.getOwnerComponent().SapId+'')");
33     }
34 }
35

```

Modelos:

Models.js

```

App.controller.js  CreateEmployee.controller.js  Menu.controller.js  ShowEmployee.controller.js  models.js x
1  sap.ui.define([
2    "sap/ui/model/json/JSONModel",
3    "sap/ui/Device"
4  ], function (JSONModel, Device) {
5    "use strict";
6
7    return {
8
9      createDeviceModel: function () {
10       var oModel = new JSONModel(Device);
11       oModel.setDefaultBindingMode("OneWay");
12       return oModel;
13     }
14
15   };
16 });
17

```

Vistas:

App.view.xml

```

App.controller.js  CreateEmployee.controller.js  Menu.controller.js  ShowEmployee.controller.js  models.js  i18n.properties  index.html  manifest.json  Component.js  App.view.xml x
1  <mvc:View controllerName="hrolltfg.hroll.HTML5Module.controller.App" xmlns:mvc="sap.ui.core.mvc" displayBlock="true" xmlns="sap.m">
2    <Shell id="shell">
3      <App id="app">
4        <pages>
5          <Page id="page" title="{i18n}title">
6            <content></content>
7          </Page>
8        </pages>
9      </App>
10     </Shell>
11   </mvc:View>
12

```

CreateEmployee.view.xml

```

App.controller.js  CreateEmployee.controller.js  Menu.controller.js  ShowEmployee.controller.js  models.js  i18n.properties  index.html  manifest.json  Component.js  App.view.xml
1  <mvc:View xmlns:core="sap.ui.core" xmlns:mvc="sap.ui.core.mvc" xmlns="sap.m" controllerName="hrolltfg.hroll.HTML5Module.controller.CreateEmployee"
2    xmlns:html="http://www.w3.org/1999/xhtml" xmlns:form="sap.ui.layout.form" xmlns:u="sap.ui.unified">
3    <Page title="{i18n}crearEmpleado">
4      <content>
5        <NavContainer id="wizardNavContainer">
6          <pages>
7            <Page showHeader="false">
8              <content>
9                <Wizard id="wizard" complete="wizardCompletedHandler">
10                 <WizardStep id="typeEmployeeStep" title="{i18n}tipoEmpleado"
11                   validated="false">
12                   <MessageStrip class="sapUiSmallMarginBottom"
13                     text="{i18n}step1ms"
14                     showIcon="true"/>
15                   <FlexBox alignItems="center" justifyContent="center">
16                     <Button text="{i18n}interno" press="toStep2" class="sapUiSmallMarginEnd">
17                       <customData>
18                         <core:CustomData key="typeEmployee" value="interno"/></core:CustomData>
19                       </customData>
20                     </Button>
21                     <Button text="{i18n}autonomo" press="toStep2" class="sapUiSmallMarginEnd">
22                       <customData>
23                         <core:CustomData key="typeEmployee" value="autonomo"/></core:CustomData>
24                       </customData>
25                     </Button>
26                     <Button text="{i18n}gerente" press="toStep2">
27                       <customData>
28                         <core:CustomData key="typeEmployee" value="gerente"/></core:CustomData>
29                       </customData>
30                     </Button>
31                   </FlexBox>
32                 </WizardStep>
33                 <WizardStep id="dataEmployeeStep"
34                   validated="false"
35                   title="{i18n}datosEmpleado">
36                   <MessageStrip class="sapUiSmallMarginBottom"
37                     text="{i18n}step2ms" showIcon="true"/>
38                   <form:SimpleForm
39                     id="step2Form"
40

```

ShowEmployee.view.xml

```

1   <mvc:View xmlns:core="sap.ui.core" xmlns:mvc="sap.ui.core.mvc" xmlns="sap.m" controllerName="hrolltfg.hroll.HTML5Module.controller.ShowEmployee"
2   xmlns:html="http://www.w3.org/1999/xhtml" xmlns:suite="sap.suite.ui.commons" class="sapUISizeCompact">
3   <SplitApp id="splitAppEmployee">
4   <masterPages>
5   <Page showHeader="false">
6   <ToolBar class="sapUiTinyMarginTop">
7   <Button icon="sap-icon://nav-back" press="onPressBack"></Button>
8   <SearchField liveChange="onSearchEmployee" width="100%" />
9   </ToolBar>
10  <List itemPress="onSelectEmployee" items="{path:'odataModel/Users',filters:[{path:'sapId',operator:'EQ',value:'rol4jc@gmail.com'}]}">
11  <items>
12  <StandardListItem title="{odataModel>FirstName} {odataModel>LastName}" description="{odataModel>Dni}" type="Active"/>
13  </items>
14  </List>
15  </Page>
16  </masterPages>
17  <detailPages>
18  <Page id="detailSelectEmployee" showHeader="false">
19  <FlexBox alignItems="center" justifyContent="center" height="100%">
20  <Text text="{i18n>seleccioneEmpleado}"></Text>
21  </FlexBox>
22  </Page>
23  <Page
24  id="detailEmployee"
25  title="{i18n>detalleEmpleado}">
26  <ObjectHeader
27  icon="{= ${odataModel>Type} === '0' ? 'sap-icon://employee-pane' : ${odataModel>Type} === '1' ? 'sap-icon://employee' : 'sap-icon://leads' }"
28  title="{odataModel>FirstName} {odataModel>LastName}"
29  intro="{odataModel>Dni}"
30  class="sapUIResponsivePadding--header"
31  responsive="true">
32  <ObjectAttribute title="{i18n>fechaIncorporacion}" text="{ path:'odataModel>CreationDate', type:'sap.ui.model.type.Date' } />
33  <ObjectAttribute title="{i18n>tipo}" text="{= ${odataModel>Type} === '0' ? ${i18n>interno} : ${odataModel>Type} === '1' ? ${i18n>autonomo} : ${i18n>gerente}" />
34  <ObjectAttribute title="{i18n>comentario}" text="{odataModel>Comments}" />
35  </ObjectHeader>
36  <IconTabBar
37  class="sapUIResponsiveContentPadding">

```

I18n – Claves

```

App.controller.js  CreateEmployee.controller.js  Menu.controller.js  ShowEmployee.controller.js  models.js  i18n.properties x
1  title=Menú
2  appTitle=Menú
3  appDescription=App Description
4  menuPrincipal = Menú Principal HROLL
5
6  pedidos = Pedidos
7
8  #XTXT:
9  firmarPedido=Firmar pedido
10
11 empleados = Empleados
12
13 #XTXT:
14 crearEmpleado=Crear empleado
15
16 #XTXT:
17 verEmpleados=Ver empleados
18
19 #XTXT:
20 tipoEmpleado=Tipo de empleado
21
22 #XTXT:
23 interno=Interno
24
25 #XTXT:
26 autonomo=Autónomo
27
28 #XTXT:
29 gerente=Gerente
30
31 #XTXT:
32 datosEmpleado=Datos empleado
33
34 #XTXT:
35 step1ms=Seleccione el tipo de empleado para rellenar sus datos.
36

```

Index.html


```

App.controller.js  CreateEmployee.controller.js  Menu.controller.js  ShowEmployee.controller.js  models.js  i18n.properties  index.html x
1  |!DOCTYPE html|
2  <html>
3  <head>
4    <meta charset="utf-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Recursos humanos HROLL</title>
7    <script id="sap-ui-bootstrap"
8      src="https://sapui5.hana.ondemand.com/resources/sap-ui-core.js"
9      data-sap-ui-theme="sap_belize_plus"
10     data-sap-ui-resourceroots="{ 'hrolltfg.hroll.HTML5Module': './' }"
11     data-sap-ui-compatVersion="edge"
12     data-sap-ui-oninit="module:sap/ui/core/ComponentSupport"
13     data-sap-ui-async="true"
14     data-sap-ui-frameOptions="trusted">
15   </script>
16 </head>
17 <body class="sapUiBody">
18   <div data-sap-ui-component data-name="hrolltfg.hroll.HTML5Module" data-id="container" data-settings='{ "id" : "HTML5Module" }'></div>
19 </body>
20 </html>
21

```

Component.js

```

App.controller.js  CreateEmployee.controller.js  Menu.controller.js  ShowEmployee.controller.js  models.js  i18n.properties  index.html  manifest.json  Component.js x
1  |sap.ui.define([
2    "sap/ui/core/UIComponent",
3    "sap/ui/Device",
4    "hrolltfg/hroll/HTML5Module/model/models"
5  ], function (UIComponent, Device, models) {
6    "use strict";
7
8    return UIComponent.extend("hrolltfg.hroll.HTML5Module.Component", {
9
10     metadata: {
11       manifest: "json"
12     },
13
14     /**
15      * The component is initialized by UIS automatically during the startup of the app and calls the init method once.
16      * @public
17      * @override
18      */
19     init: function () {
20       // call the base component's init function
21       UIComponent.prototype.init.apply(this, arguments);
22
23       // enable routing
24       this.getRouter().initialize();
25
26       // set the device model
27       this.setModel(models.createDeviceModel(), "device");
28     },
29     Sapid: "rol4jc@gmail.com"
30   });
31 });
32

```

Anexo 3. Repositorios Git

Todo el código de las 2 aplicaciones está en los siguientes repositorios de GITHUB:

- TFG01 -> <https://github.com/RaulRoll/Proyecto-TFG01/tree/Proyecto-TFG01>
- HROLL -> <https://github.com/RaulRoll/Proyecto-HROLL/tree/Proyecto-HROLL>

Configuración GITHUB:

```

Problems 1 ~/projects/hroll ~/projects/tfg01 x
user: projects $ cd tfg01
user: tfg01 $ git init
Initialized empty Git repository in /home/user/projects/tfg01/.git/
user: tfg01 $ git config --global user.email "rol4jc@gmail.com"
user: tfg01 $ git config --globat user.name "RaulRoll"
error: unknown option `globat'
usage: git config [<options>]

Config file location
--global          use global config file
--system         use system config file
--local          use repository config file
--worktree       use per-worktree config file
-f, --file <file> use given config file
--blob <blob-id> read config from given blob object

Action
--get            get value: name [value-regex]
--get-all      get all values: key [value-regex]
--get-regexp    get values for regexp: name-regex [value-regex]
--get-urlmatch  get value specific for the URL: section[.var] URL
--replace-all  replace all matching variables: name value [value_regex]
--add           add a new variable: name value
--unset        remove a variable: name [value-regex]
--unset-all   remove all matches: name [value-regex]
--rename-section rename section: old-name new-name
--remove-section remove a section: name
-l, --list     list all
-e, --edit     open an editor
--get-color    find the color configured: slot [default]
--get-colorbool find the color setting: slot [stdout-is-tty]
..           ..

user: tfg01 $ git push
fatal: The current branch Proyecto-TFG01 has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin Proyecto-TFG01

user: tfg01 $ git push --set-upstream origin Proyecto-TFG01
Enumerating objects: 82, done.
Counting objects: 100% (82/82), done.
Delta compression using up to 8 threads
Compressing objects: 100% (73/73), done.
Writing objects: 100% (82/82), 157.98 KiB | 7.18 MiB/s, done.
Total 82 (delta 7), reused 0 (delta 0)
remote: Resolving deltas: 100% (7/7), done.
remote:
remote: Create a pull request for 'Proyecto-TFG01' on GitHub by visiting:
remote:   https://github.com/RaulRoll/Proyecto-TFG01/pull/new/Proyecto-TFG01
remote:
To https://github.com/RaulRoll/Proyecto-TFG01.git
 * [new branch]      Proyecto-TFG01 -> Proyecto-TFG01
Branch 'Proyecto-TFG01' set up to track remote branch 'Proyecto-TFG01' from 'origin'.
user: tfg01 $ █

```

Anexo 4. Bibliografía

[1]Getting to know SAP Cloud Platform Integration.[Consulta durante la creación del proyecto]

<https://www.linkeit.com/blog/getting-to-know-sap-cloud-platform-integration>

[2]Gateway SAP.Foro de SAP Community.[Consulta durante la creación del proyecto]

<https://blogs.sap.com/2014/03/20/there-is-a-gateway-for-that/>

[3]SAP Cloud Platform Cloud Connector.Foro de SAP Community[Consulta durante la creación del proyecto]

<https://blogs.sap.com/2015/07/13/cloud-connector-a-brief-guide-for-beginners/>

[4]Página oficial GITHUB.[Consulta durante la creación del proyecto]

<https://github.com/oznerjvs/rrhh>

[5]Página oficial THEIA.[Consulta durante la creación del proyecto]

<https://theia-ide.org/>

[6]Página oficial CLOUDFOUNDRY.[Consulta durante la creación del proyecto]

<https://www.cloudfoundry.org/>

[7]Cuenta SAP BTP Cockpit.[Consulta durante la creación del proyecto]

<https://account.hanatrial.ondemand.com/>

[8]Página oficial WRIKE.[Consulta durante la creación del proyecto]

<https://www.wrike.com/>

[9]App JSPAIT.[Consulta durante la creación del proyecto]

<https://jspaint.app/#local:d8733ac5d0548>

[10]App SNAGIT.[Consulta durante la creación del proyecto]

https://www.techsmith.com/store/snagit?utm_source=google&utm_medium=cpc&utm_campaign=1535514905&utm_content=56328306902&utm_term=snagit%20editor&gclid=CjwKCAjw3pWDBhB3EiwAV1c5rBXSanPKD_q86YmhHjJeBn0qSX24Za989_WJcQranRCtBZMG9WloQBoc46cQAvD_BwE

[11]Partes de pág web: estructura y contenido.[Consulta durante la creación del proyecto]

<https://www.lucushost.com/blog/partes-de-una-pagina-web-estructura-y-contenido/>

[12]Generador de números DNI para pruebas.[Consulta durante la creación del proyecto]

<https://testingdatagenerator.com/doi.html>

[13]Página oficial YEOMAN.[Consulta durante la creación del proyecto]

<https://yeoman.io/>

[14]Recursos oData SAP.[Consulta durante la creación del proyecto]

<https://sapui5.hana.ondemand.com/resources/sap-ui-core.js>

[15]Página oficial de GIT[Consulta durante la creación del proyecto]

<https://github.com/>

[16]Documentación oficial SAPUI5[Consulta durante la creación del proyecto]

<https://sapui5.hana.ondemand.com/#/topic>

[17]Imagen sap fiori design guidelines.[Consulta durante la creación del proyecto]

<https://experience.sap.com/fiori-design/>

[18]Icon Explorer.[Consulta durante la creación del proyecto]

<https://sapui5.hana.ondemand.com/sdk/test-resources/sap/m/demokit/iconExplorer/webapp/index.html>

[19]Página W3Schools para probar código.[Consulta durante la creación del proyecto]

https://www.w3schools.com/cssref/css_units.asp

[20]Página oficial de oData.[Consulta durante la creación del proyecto]

<https://www.odata.org/>

[21]Servicios oData SAP.[Consulta durante la creación del proyecto]

[https://services.odata.org/V3/Northwind/Northwind.svc/\\$metadata](https://services.odata.org/V3/Northwind/Northwind.svc/$metadata)

[22]Servicios oData SAP.[Consulta durante la creación del proyecto]

[https://services.odata.org/V3/Northwind/Northwind.svc/?\\$format=json](https://services.odata.org/V3/Northwind/Northwind.svc/?$format=json)

[23]Servicios oData SAP.[Consulta durante la creación del proyecto]

<https://services.odata.org/V3/Northwind/Northwind.svc/>

[24]Página oficial QUNITJS.[Consulta durante la creación del proyecto]

<https://qunitjs.com/>

[25]Servicios cdnjs de cloudflare.[Consulta durante la creación del proyecto]

https://cdnjs.cloudflare.com/ajax/libs/signature_pad/1.5.3/signature_pad.js

[26]Wikipedia, artículo sobre Cross-site_request_forgery.[Consulta durante la creación del proyecto]

https://es.wikipedia.org/wiki/Cross-site_request_forgery

[27]Página oficial de GANT PROJECT.[Consulta durante la creación del proyecto]

<https://www.ganttproject.biz/download/gp30>

[28]Página Oficial OBS.[Consulta durante la creación del proyecto]

<https://obsproject.com/es/welcome>