



UNIVERSIDAD  
DE MÁLAGA



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA  
INFORMÁTICA  
GRADUADO EN INGENIERIA INFORMÁTICA

**DESARROLLO E IMPLEMENTACIÓN DE  
APLICACIONES SAP-FIORI BASADAS EN  
PROCESOS DE NEGOCIOS**

**DEVELOPMENT AND IMPLEMENTATION  
OF SAP-FIORI BUSINESS PROCESS BASED  
APPS**

Realizado por  
**Evert Urey Paz Soldan**

Tutorizado por  
**Carlos Rossi Jiménez**

Departamento  
**Departamento de Lenguajes y Ciencias de la  
Computación**

UNIVERSIDAD DE MÁLAGA  
MÁLAGA, DICIEMBRE DE 2020

Fecha defensa



# Resumen

El presente documento describe el desarrollo e implantación en un sistema productivo de un conjunto de aplicaciones SAP-Fiori. Con ello se pretende automatizar y homogenizar procesos de negocio en toda la empresa siguiendo la "*SAP User Experience Strategy*" y aprovechando las características intrínsecas de la herramienta tecnológica que SAP pone a disposición para ese fin, SAPUI5. Este framework se centra en la experiencia de usuario y la portabilidad multidispositivo.

En total se han desarrollado seis aplicaciones. Cuatro de ellas de tipo productivo, para cubrir la gestión de las solicitudes de trabajo de los clientes y dos de tipo analíticas destinadas a la evaluación y análisis de datos para ayudar en la toma de decisiones.

**Palabras clave:** SAP, SAP-Fiori, SAP Gateway, Odata, SAPUI5.



# Abstract

This document describes the development and implementation in a real production system of a set of SAP-Fiori applications. This is to automate and standardize business processes across the company by following the "SAP User Experience Strategy" and taking advantage of the intrinsic features of the technological tool that SAP makes available for this purpose, SAPUI5. This framework focuses on user experience and multi-device portability.

In total six applications have been developed. Four of them of a productive nature, to cover the management of clients' work requests and two of an analytical type destined to the evaluation and analysis of data to help in decision-making.

**Keywords:** SAP, SAP-Fiori, SAP Gateway, Odata, SAPUI5.



# Índice

<b>Resumen</b> .....	<b>1</b>
<b>Abstract</b> .....	<b>1</b>
<b>Índice</b> .....	<b>1</b>
<b>Índice de Ilustraciones</b> .....	<b>5</b>
<b>Introducción</b> .....	<b>8</b>
<b>1.1 Motivación</b> .....	<b>8</b>
<b>1.2 CEMOSA</b> .....	<b>9</b>
<b>1.3 Estado del arte</b> .....	<b>11</b>
<b>1.4 Objetivos</b> .....	<b>12</b>
1.1.1 Principal .....	12
1.4.1 Secundarios .....	12
<b>1.5 Metodología</b> .....	<b>12</b>
1.5.1 Análisis.....	13
1.5.2 Diseño.....	13
1.5.3 Implementación .....	13
1.5.4 Pruebas.....	13
1.5.5 Mantenimiento.....	14
<b>1.6 Estructura de la memoria</b> .....	<b>14</b>
<b>Arquitectura y entorno tecnológico</b> .....	<b>16</b>
<b>2.1 Enterprise Resource Planning (ERP)</b> .....	<b>16</b>
<b>2.2 Tecnologías SAP</b> .....	<b>19</b>
2.2.1 SAP-ERP .....	19
2.2.2 SAP Fiori y la evolución de la experiencia de usuario .....	22
2.2.3 SAP HANA .....	24
2.2.4 SAPUI5 .....	25
2.2.5 SAP Gateway.....	26
2.2.6 SAP Fiori Launchpad .....	27
<b>2.3 OData</b> .....	<b>28</b>
<b>2.4 Herramientas de desarrollo</b> .....	<b>28</b>
2.4.1 Diccionario de datos (SE11).....	28
2.4.2 Editor ABAP (SE38) .....	29
2.4.3 Bibliotecas de funciones (SE37).....	30
2.4.4 Navegador de Objetos (SE80).....	31
2.4.5 SAP Gateway Service Builder (SEGW) .....	31
2.4.6 Generador de perfiles (PFCG).....	32
2.4.7 SAP Web IDE.....	33
<b>2.5 Lenguajes de programación</b> .....	<b>34</b>
2.5.1 ABAP .....	34
2.5.2 JavaScript.....	34
2.5.3 XML.....	34
<b>2.6 Arquitectura del sistema</b> .....	<b>35</b>

<b>Sistema de Gestión de Avisos .....</b>	<b>37</b>
<b>3.1 Análisis .....</b>	<b>37</b>
3.1.1 Descripción funcional .....	37
3.1.2 Requisitos Generales .....	41
3.1.3 Catálogo de requisitos .....	42
3.1.4 Casos de Uso .....	44
3.1.5 Actores del Sistema .....	46
<b>3.2 Diseño .....</b>	<b>47</b>
3.2.1 Modelos entidad-relación BBDD .....	47
3.2.2 Modelado de Servicios Odata .....	49
3.2.3 Modelado aplicaciones SAPUI5 .....	52
<b>3.3 Implementación .....</b>	<b>59</b>
3.3.1 Base de datos .....	59
3.3.2 Backend - Implementación de servicios Odata .....	64
3.3.3 Frontend .....	73
<b>Aplicación de evaluación de deuda .....</b>	<b>86</b>
<b>4.1 Análisis .....</b>	<b>86</b>
4.1.1 Descripción Funcional .....	86
4.1.2 Catálogo de requisitos .....	87
4.1.3 Casos de Uso .....	89
4.1.4 Actores del Sistema .....	91
<b>4.2 Diseño .....</b>	<b>91</b>
4.2.1 Modelos entidad-relación BBDD .....	91
4.2.2 Modelado de Servicios Odata .....	92
4.2.3 Modelado aplicaciones SAPUI5 .....	93
<b>4.3 Implementación .....</b>	<b>95</b>
4.3.1 Servicios implementados .....	95
4.3.2 Frontend .....	97
<b>Aplicación Seguimiento de Conservación de carreteras .....</b>	<b>99</b>
<b>5.1 Análisis .....</b>	<b>99</b>
5.1.1 Descripción Funcional .....	99
5.1.2 Catálogo de requisitos .....	100
5.1.3 Casos de Uso .....	101
5.1.4 Actores del Sistema .....	102
<b>5.2 Diseño .....</b>	<b>102</b>
5.2.1 Modelos entidad-relación BBDD .....	102
5.2.2 Modelado de Servicios Odata .....	103
5.2.3 Modelado aplicaciones SAPUI5 .....	104
5.2.4 Wireframes .....	104
<b>5.3 Implementación .....</b>	<b>105</b>
5.3.1 Servicios implementados .....	105
5.3.2 Frontend .....	106
<b>Despliegue en SAP-Fiori Launchpad .....</b>	<b>110</b>
<b>6.1 Catálogos .....</b>	<b>110</b>
6.1.1 Mosaicos (Tiles) .....	111
6.1.2 Asignación de destino .....	114
<b>6.2 Grupos .....</b>	<b>116</b>
<b>Gestión de Roles .....</b>	<b>119</b>

7.1	Roles Backend .....	119
7.2	Roles Frontend .....	119
7.3	Perfiles.....	120
<b>Pruebas .....</b>		<b>122</b>
8.1	<b>Pruebas unitarias.....</b>	<b>123</b>
8.1.1	Servicios Odata .....	123
8.1.2	Aplicaciones SAPUI5 .....	124
8.2	<b>Pruebas funcionales.....</b>	<b>126</b>
8.3	<b>Casos de prueba .....</b>	<b>126</b>
<b>Conclusiones .....</b>		<b>130</b>
9.1	<b>Mantenimiento y Evolución .....</b>	<b>131</b>
9.2	<b>Valoración.....</b>	<b>131</b>
<b>Referencias .....</b>		<b>132</b>
<b>Apéndice A.....</b>		<b>133</b>
<b>Manual de Usuario .....</b>		<b>133</b>
Acceso a SAP-Fiori Launchpad: .....		133
App Alta Aviso:.....		134
App Avisos:.....		137
App Plan de Avisos: .....		139
App Listado de Avisos: .....		142
App Informe (Conservación carreteras): .....		145
App Evolutivo Cliente: .....		147



# Índice de Ilustraciones

Ilustración 1:Delegaciones nacionales .....	10
Ilustración 2:Delegaciones internacionales .....	10
Ilustración 3:Esquema metodología en cascada.....	12
Ilustración 4:Ranking ERP's.....	18
Ilustración 5: Estructura modular SAP-ERP.....	19
Ilustración 6:Arquitectura SAP-Fiori .....	24
Ilustración 7:Esquema simplificado de un sistema hanna .....	25
Ilustración 8:Arquitectura SAPUI5 .....	26
Ilustración 9:Patrón MVC en sapui5 .....	26
Ilustración 10:Arquitectura SAP-Gateway .....	27
Ilustración 11:SAP-Fiori Launchpad .....	27
Ilustración 12:Transacción SE11 .....	29
Ilustración 13:Transacción SE38 .....	30
Ilustración 14: Transacción SE37 .....	30
Ilustración 15:Transacción SE80 .....	31
Ilustración 16:Sap-Gateway Service Builder .....	32
Ilustración 17:Arquitectura SAP WebId .....	33
Ilustración 18:Interfaz WebId .....	33
Ilustración 19:Opciones de despliegue .....	35
Ilustración 20:Arquitectura servidores independientes .....	36
Ilustración 21: Protocolos de comunicación.....	36
Ilustración 22: Diagrama de procesos gestión de avisos .....	38
Ilustración 23:Casos de uso de la gestión de avisos .....	46
Ilustración 24: ER Avisos .....	48
Ilustración 25:MC Alta .....	50
Ilustración 26:MC Visualización .....	51
Ilustración 27:MC Agenda.....	52
Ilustración 28: MC Genérico aplicación SAPUI5.....	53
Ilustración 29:WF App Alta Aviso.....	54
Ilustración 30: WF App Avisos.....	55
Ilustración 31:Wf App Plan Avisos .....	56
Ilustración 32:WF App listado de avisos .....	57
Ilustración 33: Proceso de inicio de una aplicación SAPUI5 .....	58
Ilustración 34: Diagrama de secuencia general operaciones CRUD .....	59
Ilustración 35:SE11 Crear tabla.....	60
Ilustración 36:Conf. Parámetros de tabla .....	60
Ilustración 37:Campos de tabla .....	61
Ilustración 38:Opciones técnicas .....	61
Ilustración 39:Conf. Categoría de ampliación.....	62
Ilustración 40:Categorías de ampliación.....	62
Ilustración 41:Activar tabla.....	62
Ilustración 42:Tabla ZTLB_AVISO .....	63
Ilustración 43:Tabla ZTLB_AVITOMAS .....	63
Ilustración 44:Tabla ZTLB_AVIACI .....	64
Ilustración 45:Mapeo E-E ZGW_ALTAVI.....	65

Ilustración 46:Propiedades entity aviso.....	66
Ilustración 47:Asociaciones ZGW_ALTAVI.....	66
Ilustración 48:Artifacts ZGW_ALTAVI.....	66
Ilustración 49:Operaciones CRUD Avisoset.....	67
Ilustración 50:Metodos redefinidos en ZGW_ALTAVI.....	67
Ilustración 51:Mapeo E-E ZGW_CRUAVISOS.....	70
Ilustración 52:Asociaciones ZGW_CRUAVISOS .....	70
Ilustración 53:Propiedades entidad aviso.....	70
Ilustración 54:Metodos redefinidos en ZGW_CRUAVISOS .....	71
Ilustración 55:Mapeo E-E ZGW_CALENDARIOPLAN.....	72
Ilustración 56:Asociaciones ZGW_CALENDARIOPLAN .....	72
Ilustración 57:Metodos redefinidos en ZGW_CALENDARIOPLAN .....	72
Ilustración 58:WebId Crear proyecto.....	74
Ilustración 59:Asignación de Servicio Odata .....	74
Ilustración 60:Asignación de modelo .....	75
Ilustración 61:Conf de rutas.....	75
Ilustración 62:Crear vistas.....	76
Ilustración 63:Estructura vista XML .....	77
Ilustración 64:estructura fragmento XML.....	77
Ilustración 65: Estructura controlador .....	79
Ilustración 66:ZBSP_ALTAVISO.....	80
Ilustración 67: AltaCEW.view.xml .....	80
Ilustración 68: Interfaz gráfica Wizard .....	81
Ilustración 69:ZBSP_AVISO.....	81
Ilustración 70: Estructura Fuente: split-screen layout .....	82
Ilustración 71: App.view.xml.....	82
Ilustración 72:ZBSP_CALPLAN .....	83
Ilustración 73: Planificación.view.xml .....	83
Ilustración 74: Interfaz Planning Calendar .....	84
Ilustración 75:ZBSP_TAREASLIST.....	84
Ilustración 76: ListReportExt.controller.js.....	85
Ilustración 77: Interfaz de implementación de anotaciones.....	85
Ilustración 78:Casos de uso de deuda.....	91
Ilustración 79:ER App Deuda.....	92
Ilustración 80:MC Evolución de deuda .....	93
Ilustración 81:WF App evolución de deuda .....	95
Ilustración 82:Mapeo E-E ZGW_GESDEUDACL.....	96
Ilustración 83:Asociaciones ZGW_GESDEUDACL .....	96
Ilustración 84:Métodos redefinidos ZGW_GESDEUDACL .....	97
Ilustración 85:ZBSP_GESDEUDACL.....	97
Ilustración 86: Debdetail.view.xml.....	98
Ilustración 87:ER App informe .....	103
Ilustración 88:MC informe conservación de carreteras.....	104
Ilustración 89:WF App Informe conservación de carreteras .....	105
Ilustración 90:Mapeo E-E ZGW_INFACCT.....	105
Ilustración 91:Métodos redefinidos ZGW_INFACCT .....	106
Ilustración 92: Analytical List Page.....	107
Ilustración 93:ZBSP_INFGRA .....	107
Ilustración 94:SFLP Grupos/Catálogos .....	111
Ilustración 95:SFLP Configuración de mosaico.....	112
Ilustración 96:SFLP Catálogo Z_PS_LABORATORIO .....	112
Ilustración 97:Catálogo Z_PS_DIPGRA .....	113
Ilustración 98:SFLP Catálogo Z_PS_DIRECCION .....	114
Ilustración 99:SFLP Conf. O.S. ZCRUavisos.....	115
Ilustración 100:SFLP Asig. de destinos Z_PS_LABORATORIO .....	115
Ilustración 101:SFLP Asig. de destinos Z_PS_DIPGRA .....	116
Ilustración 102:SFLP Asig. de destinos Z_PS_DIRECCION.....	116
Ilustración 103:SFLP Grp.ZLAB .....	117

Ilustración 104:SFLP Grp. ZDEUDA.....	117
Ilustración 105:SFLP Grp. ZDIPGRADIR .....	118
Ilustración 106:Perfil Laboratorio .....	120
Ilustración 107:Perfil Dirección.....	121
Ilustración 108:Perfil Dirección de Granada .....	121
Ilustración 109:TCODE /IWFND/MAINT_SERVICE .....	123
Ilustración 110:Prueba peticiones ZGW_GESDEUDACL.....	124
Ilustración 111:Tipo de petición .....	124
Ilustración 112:Conf. mock data .....	125
Ilustración 113:Prueba mock App Inf. conservación de carreteras .....	125
Ilustración 114: CP Sis. Gestión Avisos.....	127
Ilustración 115:CP App Evolutivo cliente .....	128
Ilustración 116:CP App Informe.....	129

# 1

## Introducción

### 1.1 Motivación

En los últimos años se ha oído hablar incesantemente en el ámbito empresarial del fenómeno denominado "Transformación Digital". Se ha hablado y se sigue hablando tanto de este concepto que en muchas ocasiones se ha llegado a tergiversar su significado, asociándolo a usar herramientas ofimáticas o por ejemplo a simples procedimientos de marketing como tener presencia en las redes sociales. La realidad es que el término transformación digital es tan amplio y transversal en prácticamente todos los sectores de la sociedad que los expertos advierten de la encrucijada en el que la tecnología pone al tejido empresarial mundial en lo que han denominado "La cuarta revolución industrial".

Industria 4.0 en Europa y Smart Manufacturing o Industrial Internet en EE. UU. son los términos para definir el fenómeno de transformación digital. El término de industria 4.0 fue acuñado en Alemania el cual se extendió a toda Europa y por supuesto llegó a España. El año 2016 la firma Roland Berger junto con Siemens España publicó un estudio titulado "España 4.0 - El reto de la transformación digital de la economía" en el que se define este fenómeno de la siguiente manera:

*"La transformación digital es la adaptación de las cadenas de valor de los distintos sectores de la economía a ese efecto disruptivo que comienza con el consumidor digital. Es la conexión integral de las distintas áreas de la economía y la manera en la que los players de cada sector se adaptarán a las nuevas condiciones que imperan en la economía digital."*

Y añadía la siguiente predicción:

*"La transformación digital podría implicar un incremento de 120.000 millones de euros en los sectores analizados para el año 2025."*

En el informe se recogían las estadísticas de progreso de la adopción de la transformación digital en las empresas españolas; sus oportunidades, barreras, diferencias sectoriales y por supuesto los retos a futuro. Bajo el lema renovarse o morir, quebrar en el caso de las empresas, parte del tejido empresarial español ha recogido el testigo del reto planteado por el contexto global actual, la era de la información.

En este contexto y teniendo en mente que a transformación digital es un viaje no un destino, la empresa donde se lleva a cabo este TFG, CEMOSA, se halla inmersa en este viaje.

## **1.2 CEMOSA**

Centro de Estudios de Materiales y Control de Obra SA es una empresa especializada en servicios de ingeniería y control de calidad en el sector de la construcción. Cuenta con una gama de servicios amplia en los siguientes ámbitos: Ingeniería, Geotecnia, Seguridad y Salud, Edificación, Control de Materiales y Certificación / Mercado CE.

Fundada en 1972 como una empresa familiar, la compañía fue pionera en el sector de la construcción en Andalucía. En sus inicios, se especializó en el control en laboratorio de diferentes materiales para la construcción y en la realización de trabajos de campo y estudios geotécnicos. Posteriormente fue ampliando su abanico de servicios hasta lograr ofrecer un control integral del sector que abarca desde el proyecto hasta la certificación final de producto, tanto en obra civil como en obra pública.

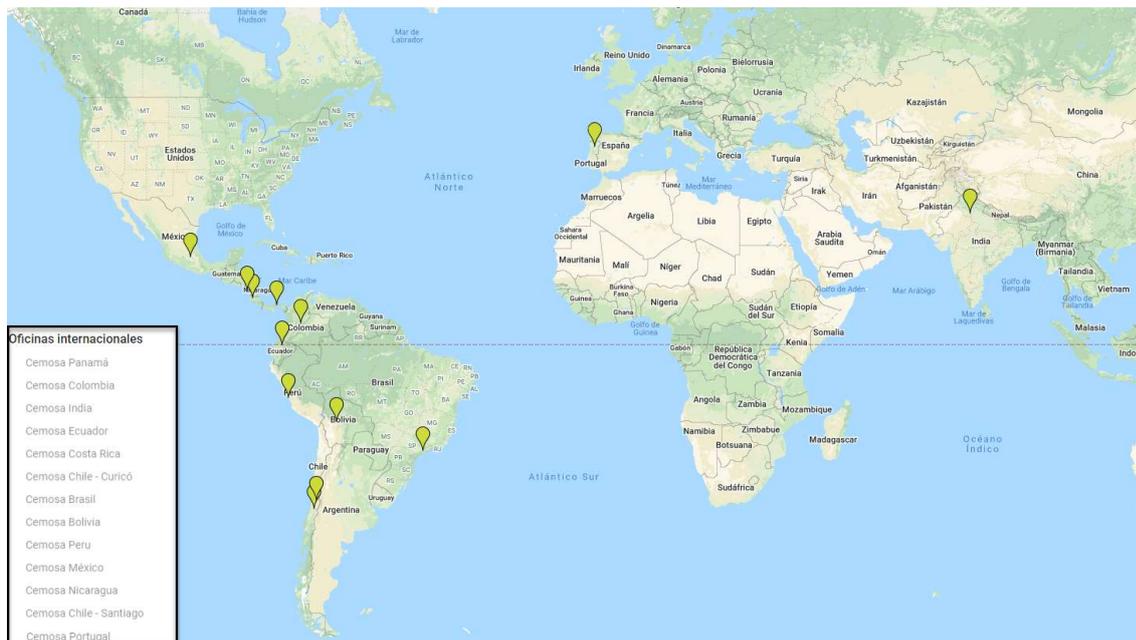
El crecimiento en su oferta de servicios fue acompañado con una expansión territorial mediante la apertura de delegaciones a lo largo y ancho de la península a nivel nacional (véase Ilustración 1) y con delegaciones estratégicamente situadas a nivel internacional (véase Ilustración 2).

Ilustración 1:Delegaciones nacionales



Fuente: <https://www.cemosa.es/>

Ilustración 2:Delegaciones internacionales



Fuente: <https://www.cemosa.es/>

CEMOSA tiene como valor de empresa el fomento del conocimiento, la innovación y la tecnología. Esto se refleja en que actualmente además de los departamentos de su modelo clásico de negocio cuenta con un departamento específico de Investigación, Desarrollo Tecnológico e Innovación I+D+I y con un departamento de Informática el cual cuenta con dos equipos:

- **Equipo de Sistemas Informáticos.** - La función principal es la de dar servicios soporte de usuario a la totalidad de la empresa y de administración y mantenimiento a la infraestructura de servidores y redes con la que cuenta la empresa.
- **Equipo de Desarrollo.** - Se enfoca principalmente en el mantenimiento y desarrollo de software sobre SAP-ERP, un sistema de gestión empresarial integral, también se realizan proyectos de desarrollo e implantación con tecnologías relacionadas con SAP, entre ellas SAP-Fiori, Adobelifecycle o SAP-HANA.

### 1.3 Estado del arte

En el contexto actual en plena era de la Industria 4.0 CEMOSA está inmersa en el viaje que representa la transformación digital. Así pues, es consciente de que hay cambios que debe realizar en distintos procesos asociados a varios de sus departamentos para adaptarse a los nuevos tiempos tecnológicos. Como ejemplo de estos procesos tenemos:

- **Gestionar de principio a fin el proceso de toma de materiales y actividades en obra.** Lo que implica gestionar las solicitudes de trabajo del cliente, pasar por el registro de la toma o actividades y emisión de albaranes para finalizar con la recepción de muestras en el laboratorio.
- **Realizar un análisis y seguimiento eficaz de los clientes a nivel financiero,** analizando su perfil de crédito. Esto incluye su deuda y/o impagos con la empresa a distintos niveles, volumen de contratación, entre otros parámetros. Esta información es indispensable tenerla estructurada disponible y actualizada.
- Existen necesidades específicas que surgen con la adjudicación de nuevos proyectos en la empresa. Algunos clientes incluyen el requerimiento de sus contratos un sistema que les permita **realizar el seguimiento de los trabajos** que se les va realizando.

Hemos mencionado solo los procesos que forman parte del ámbito de este TFG y que se realizan actualmente en el departamento de laboratorio, el departamento de administración y en la delegación de granada respectivamente. Estos procesos se realizan al margen del sistema integral de gestión con el que cuenta CEMOSA SAP-ERP debido a distintos factores, entre ellos:

- Disponibilidad multidispositivo. - Para trabajar con SAP-ERP es necesario un ordenador.
- Seguridad. - El acceso a SAP-ERP por parte de un cliente es inviable por la política de seguridad de la empresa.

Los procesos antes descritos se realizan en papel y de manera semi manual con apenas intervención de alguna que otra herramienta ofimática como Word o Excel. La gestión de solicitudes de trabajos presenta un problema añadido ya que se realiza de manera distinta en cada delegación.

Por lo expuesto surge la necesidad de integrar estos procesos al sistema de gestión SAP-ERP, desarrollando para ello el software necesario que nos permita superar las limitaciones actuales a las cuales nos enfrentamos y también nos permita homogenizar estos procesos en todas las delegaciones.

## 1.4 Objetivos

### 1.1.1 Principal

Como objetivo principal este TFG tiene la finalidad de desarrollar e implantar un conjunto de aplicaciones SAP-Fiori para el sistema productivo de la empresa CEMOSA.

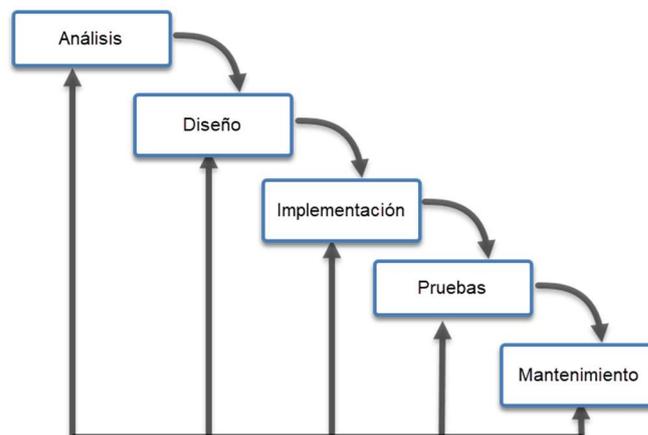
### 1.4.1 Secundarios

- Automatizar y homogenizar distintos procesos de negocio en toda la empresa.
- Avanzar en proceso de transformación digital en que lleva inmerso CEMOSA.

## 1.5 Metodología

La metodología con la que se trabaja en el departamento de informática de CEMOSA sigue el esquema de desarrollo de software en cascada con retroalimentación.

Ilustración 3:Esquema metodología en cascada



Dentro del equipo de desarrollo los proyectos los van definiendo las necesidades que transmiten el resto de los departamentos. Lo primero que se define para el proyecto es la planificación estimada y personal asignado. Una vez definido esto se empieza con las distintas fases del desarrollo de software.

### **1.5.1 Análisis**

Esta fase el personal asignado al proyecto realiza el estudio de la solicitud, se establecen los requisitos funcionales y también el alcance, llevando a cabo reuniones con las partes involucradas. También reuniones del equipo de desarrollo en las cuales se plantean soluciones a los requerimientos y se especifican estos ya sean software o de infraestructura. Una vez estudiada y definida una posible solución se intenta realizar nuevas reuniones con las partes para plantearles la solución y que esta sea aceptada.

### **1.5.2 Diseño**

Luego del análisis y el planteamiento de solución se realiza el modelado de esta mediante distintos diagramas UML según las necesidades en cada proyecto, por ejemplo, si la solución implica implementación en BBDD se realiza el modelado mediante modelos de entidad relación que definan las tablas y relaciones a crear, si la solución implica la necesidad de una interfaz gráfica de usuario se realizan modelados de plano de pantalla (wireframes). Junto con los anteriores además diagramas de clases, diagramas de flujo, diagramas de secuencia etc.

### **1.5.3 Implementación**

Esta fase es la del desarrollo propiamente dicho, incluye la parte de: programación, configuración, parametrización y despliegue de la solución. La mayor parte de proyectos del equipo son sobre SAP-ERP, estos se realizan en un entorno de desarrollo que es completamente independiente del sistema productivo.

Nuestro entorno de desarrollo es un clon del entorno productivo, pero sin algunas funcionalidades por ejemplo la integración con el módulo financiero o el servidor de correos activo. Todo esto es debido a su naturaleza ya que es un entorno pensado exclusivamente para el desarrollo.

### **1.5.4 Pruebas**

Una vez se han terminado los desarrollos en el entorno anterior se transportan y configuran nuevamente para realizar pruebas en un entorno intermedio que si cuenta con la integración con otros módulos y servicios el entorno de integración. En él se realizan pruebas funcionales y de integración con el resto del sistema.

Si se detecta algún fallo se vuelve a la fase de análisis y por tanto esa versión del desarrollo no llega al entorno de producción.

### **1.5.5 Mantenimiento**

Esta parte se incluye el soporte a usuario de los desarrollos que realiza el equipo. Los usuarios son una parte importante en el mantenimiento ya que su feedback es el que nos ayuda a detectar fallos y también a realizar ampliaciones o mejoras sobre el software desarrollado.

## **1.6 Estructura de la memoria**

La memoria se estructura en los siguientes capítulos:

### **1. Introducción.**

En este capítulo se establece la motivación y los problemas que dan origen a este TFG, así como los objetivos de este. También incluye una presentación de la empresa y descripción de la estructura del departamento de informática y la metodología que usamos.

### **2. Arquitectura y entorno tecnológico.**

En este capítulo se describe la arquitectura necesaria para implementar exitosamente la solución planteada y las tecnologías involucradas en los desarrollos necesarios para la implementación de esta.

### **3. Sistema de Gestión de Avisos.**

En este capítulo se describe el proceso de desarrollo del sistema de gestión de avisos como un conjunto de aplicaciones SAP-Fiori. Describiendo con detalle todos los procesos involucrados en el desarrollo Diseño e implementación. Se describen también todos los modelos, diagramas y código resultante de cada proceso.

### **4. Aplicación de evaluación de deuda.**

Se describe el diseño e implementación de la aplicación SAPUI5 para la Evaluación de cliente.

### **5. Aplicación Seguimiento de Conservación de carreteras.**

Se describe el diseño e implementación de la aplicación SAPUI5 informe gráfico de conservación de carreteras.

### **6. Publicación en SAP-Fiori Launchpad.**

Se describe en detalle el procedimiento para publicar las aplicaciones SAPUI5 desarrolladas en la plataforma de lanzamiento SAP-Fiori Launchpad.

### **7. Gestión de Roles.**

Se describen los roles creados para construir los perfiles de usuarios asociados a las aplicaciones SAPUI5 desarrolladas.

### **8. Pruebas.**

Se describen los tipos de pruebas realizadas a lo largo de las fases de desarrollo e implantación de las aplicaciones SAPUI5.

#### **9. Conclusiones.**

En este capítulo se resume si se han alcanzado o no los objetivos junto con la valoración y sentir al finalizar.

#### **10. Referencias.**

Donde se incluye un listado con las fuentes e información consultadas en este TFG.

# 2

## Arquitectura y entorno tecnológico

### 2.1 Enterprise Resource Planning (ERP)

Los sistemas de planificación de recursos empresariales (ERP) son un conjunto de sistemas de información que permiten el manejo e integración de la mayoría de las operaciones de una empresa. Generalmente las que tienen que ver con la producción, la logística, el inventario, los envíos y la contabilidad.

Un ERP funciona como un todo, con un acceso a una base de datos centralizada, a pesar de que su arquitectura en ocasiones puede ser la de varios sistemas integrados entre sí o estar diseñado con una estructura modular.

Estos sistemas informáticos tienen su origen como otros en el área militar. Después de la segunda guerra mundial entre 1950 y 1960 EE. UU. comenzó a promover el uso de programas informáticos para tareas de administración y organización del ejército estos se pueden considerar los precursores de los ERP actuales.

En la década de los 70 IBM lanzó al mercado el primer MRP (manufacturing resource planning), un sistema de planificación y administración normalmente asociados a la producción y control de inventario. A los cual le siguió en la siguiente década la evolución a los sistemas que se denominaron MRP-II los cuales

además de la gestión de materiales incluían la funcionalidad de administración de recursos económicos lo cual era es y será fundamental para cualquier empresa.

El término ERP (Enterprise Resource Planning) se utilizó por primera vez en la década de 1990. Se centraba en funciones empresariales como la producción, finanzas y contabilidad, recursos humanos, gestión de proyectos, etc. Estaba compuesto por sistemas modulares con características avanzadas.

A partir de comienzo de siglo los ERP se popularizaron y fueron incluyendo más funcionalidades como la gestión de las relaciones con los clientes (CRM).

Hay que resaltar en todo este periodo que el adquirir un ERP conllevaba una inversión económica importante para una empresa ya que se requería de una infraestructura tecnológica considerable entre servidores, equipos informáticos y redes. Esta realidad discriminaba a las PYMES ya que la mayoría no podía afrontar esos costes.

A partir de 2005 con la tecnología de la nube y la incorporación de los conceptos SaaS, PaaS e IaaS las principales compañías productoras de ERP se enfocan hacia estas soluciones. Esta tendencia es la que se mantiene hasta el momento y supone un punto de inflexión, desde su puesta en marcha democratizan de alguna manera el uso de los ERP ya que las PYMES no necesitan las grandes inversiones en infraestructura para poder implantar estos sistemas que suponen un requisito indispensable para una empresa en plena era de la información.

Anualmente se publican rankings de los principales proveedores de software empresarial a nivel mundial (véase Ilustración 4).

Ilustración 4: Ranking ERP's

Posición	Empresa	Desarrollos recientes
1	SAP	SAP estaba en camino de obtener más de 12,000 clientes para su ERP S / 4HANA de próxima generación a fines de 2019, y el 40% de las últimas incorporaciones se consideraban nuevas ganancias netas para el proveedor.
2	Oracle	Oracle ha obtenido a más de 25.000 clientes de ERP en la nube, incluidos casi 7.000 para Oracle ERP Cloud y 18.000 para NetSuite. Cada trimestre, Oracle está migrando al menos 200 clientes de sus sistemas ERP locales, como E-Business Suite, a la nube.
3	Intuit Inc.	Los suscriptores de Intuit QuickBooks Online superan ahora los 5 millones, frente a los 1,6 millones de 2016. Mientras tanto, Intuit fortaleció la gestión del capital de trabajo y las ofertas de pago para las pymes, junto con el procesamiento de nóminas el mismo día.
4	FIS Global	FIS sees low single-digit growth for its ERP apps for banks and FIs aided by its maintenance revenues. Cloud migration is a work in progress.
5	Fiserv	FIS prevé un crecimiento de un solo dígito bajo para sus aplicaciones ERP para bancos y entidades financieras gracias a sus ingresos por mantenimiento. La migración a la nube es un trabajo en progreso.
6	Microsoft	La migración de AX, C5, GP, NAV y SL heredados a Microsoft Dynamics 365 para la nube sigue siendo la principal prioridad para la estrategia de aplicaciones empresariales en evolución de Microsoft.
7	Cerner Corporation	Si bien Cerner se centra en la transformación digital de los proveedores de atención médica, también depende en gran medida de los planes de salud patrocinados por los proveedores para la salud de la población y las ventas gubernamentales a entidades como el Departamento de Defensa y Asuntos de Veteranos.
8	IQVIA	En junio de 2018, IQVIA lanzó IQVIA ePromo, una nueva solución de tecnología de gestión de contenido basada en la nube. ePromo facilita a las empresas de ciencias biológicas la gestión de sus materiales médicos y promocionales regulados a lo largo de todo el ciclo de contenido de creación, aprobación, difusión y retirada.
9	Constellation Software Inc.	Durante su último período de nueve meses, CSI registró un crecimiento del 24% en los ingresos totales, principalmente a través de adquisiciones. El crecimiento orgánico fue bajo de un solo dígito .
10	Infor	Infor adquirió Vivonet para aplicaciones ERP para la industria hotelera con punto de venta y funcionalidad de inventario y adquisición.

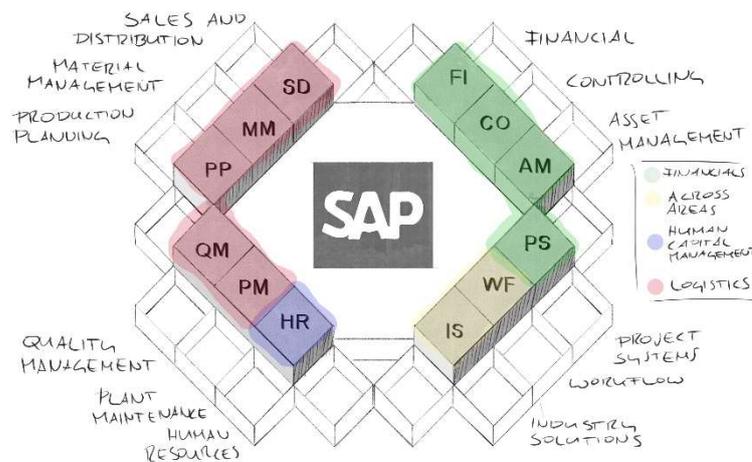
Fuente: <https://www.appsruntheworld.com/top-10-erp-software-vendors-and-market-forecast/>

## 2.2 Tecnologías SAP

### 2.2.1 SAP-ERP

Siendo la solución más antigua incluida dentro del Business suite de SAP el ERP se puede considerar el núcleo de toda instalación. Con una estructura modular (véase Ilustración 5) es esencialmente, una aplicación sobre una base de datos centralizada, que contiene toda la información de la empresa en materias de recursos humanos, gestión de materiales, finanzas, contabilidad, y otros.

Ilustración 5: Estructura modular SAP-ERP



Fuente: <https://inui.io/sap-modules-components/>

Está compuesto de tres secciones principales, compuestas a su vez de una serie de módulos:

- Sección de Financiera:
  - Finanzas
  - Contabilidad
  - Gestión de activos Fijos
  - Sistema de Proyecto
- Sección de Logística:
  - Ventas
  - Materiales
  - Planificación de producción
  - Control de Calidad
  - Gestión de Plantas
- Sección de Recursos Humanos:
  - Recursos Humanos

Todos estos módulos interactúan entre sí, dando lugar a un sistema altamente robusto, integrado y eficiente. Es un sistema complejo y por ello el coste de su implantación no es algo trivial para una empresa grande y mucho menos para una PYME de hecho y pensando en estas últimas SAP ofrece otra solución adaptada a este sector del mercado SAP Business one.

Tampoco es trivial el manejo de cada módulo ya que si nos introducimos en el de ventas por ejemplo vemos que va creciendo y se divide en submódulos que a su vez contienen las transacciones asociadas a los procesos de ventas.

- **Datos maestros**

- VD01 Crear Cliente
- VD02 Modificar Cliente
- VD05 Bloquear Cliente
- VAP1 Crear persona de contacto
- VAP2 Modificar persona de contacto
- MM01 Crear Material
- MM02 Modificar Material
- VB11 Determinación de material
- VB01 Crear Listado / Exclusión de materiales
- VK11 Crear condición de precio
- VBN1 Crear bonificación en especie
- VV11 Crear registro de condición para mensaje para documento de venta
- VDS1 Crear registro info cliente material
- VBO1 Crear acuerdo de rappel
- VB31 Crear programa de promoción
- VB21 Crear promoción de venta

- **Ventas**

- VA11 Crear Consulta
- VA21 Modificar Consulta
- VA21 Crear Oferta
- VA22 Modificar Consulta
- VA01 Crear Pedido
- VA02 Modificar Pedido
- VA31 Crear Plan de entregas
- VA32 Modificar Plan de entregas
- VA41 Crear Pedido Abierto
- VA42 Modificar Pedido Abierto
- VA05N Lista de pedidos

- **Expedición**

- VL01N Crear entrega con referencia a pedido de cliente
- VL02N Modificar entrega de salida
- VL01NO Crear entrega sin referencia a pedido de cliente
- VL06U Tratamiento colectivo de entregas de salida
- VL10A Tratamiento de entregas por pedido de cliente
- VL06G Procesamiento colectivo mediante monitor de entrega de salida

- **Facturación**

- VF01 Crear factura
- VF02 Modificar factura
- VF04 Tratar pool de facturación
- VFX3 Tratamiento de facturas bloqueadas
- VF21 Crear lista de facturas
- VF05N Lista de facturas

## **Evolución**

Bajo el nombre de “SAP Systemanalyse, Anwendungen und Programmentwicklung” SAP fue fundada en 1972 en la Ciudad de Mannheim, Alemania, por un grupo de antiguos empleados de IBM (Claus Wellenreuther, Hans-Werner Hector, Klaus Tschira, Dietmar Hopp y Hasso Plattner).

En poco más de un año lanzan SAP *R/1 System RF* (1973) orientado principalmente a ser un sistema de contabilidad financiera. No sería hasta finales de los 70 cuando SAP introdujo en el mercado su nuevo software *SAP R/2 Mainframe System* (1979) capaz de realizar la mayoría de las funciones empresariales como la cadena de suministro, los procesos de fabricación, la contabilidad y los recursos humanos en tiempo real. El R/2 se consolidó en el mercado mundial por más de una década debido a las grandes prestaciones que ofrecía para la época.

En 1992 SAP presenta su solución rediseñada *SAP R/3 Edición de Empresa 1.0* incluyendo el concepto de arquitectura cliente servidor en tres capas: presentación, aplicación y base de datos que se ejecutan en el hardware diferente. Este concepto resultó un gran éxito para la empresa la cual ya se consolidaba como el primer proveedor de software a nivel mundial. El sucesor del R/3 fue en 2004 *SAP ERP Central Component (ECC) 5.0* permitió a las empresas optimizar el rendimiento y reducir los costos de TI al ofrecer soluciones SRM, CRM, SCM, PLM y ERP en un solo paquete.

Por último, en el 2015 llegó SAP S/4 Simple Suite para HANA para integrar con el ERP lo que muchos han catalogado como la mayor evolución en el mundo del software SAP, SAP HANA es una base de datos en memoria la cual presume de ser entre 60 u 80 veces más rápida que una base de datos tradicional.

### **2.2.2 SAP Fiori y la evolución de la experiencia de usuario**

Desde la perspectiva puramente técnica SAP-Fiori es un conjunto de tecnologías que permiten el desarrollo de aplicaciones de negocio con arquitectura web. Pero el concepto engloba mucho más ya que en su concepción y diseño se incluye con un peso importante la experiencia de usuario que se plasma además en recomendaciones y buenas prácticas a la hora de desarrollar con esta tecnología.

SAP Fiori fue lanzado como concepto y producto el 2015 como respuesta de SAP SE a la necesidad de entorno empresarial actual. Para los usuarios hoy en día es imprescindible la accesibilidad a la información oportuna actualizada y precisa en cualquier momento y lugar.

Por su parte SAP Fiori UX representa un nuevo paradigma de la experiencia de usuario. Se enfoca principalmente en dos aspectos importantes; la facilidad de uso y la ejecución multiplataforma.

#### **2.2.2.1 SAP Fiori UX Principios de Diseño**

El paradigma de la experiencia de usuario de SAP Fiori está compuesto por cinco principios de diseño que de aplicarlos nos garantizan que todas las aplicaciones sigan el diseño de UX definido por SAP. Estos cinco crean una UX coherente, independientemente de su función en la organización, y ofrece la mejor experiencia del usuario en cualquier dispositivo.

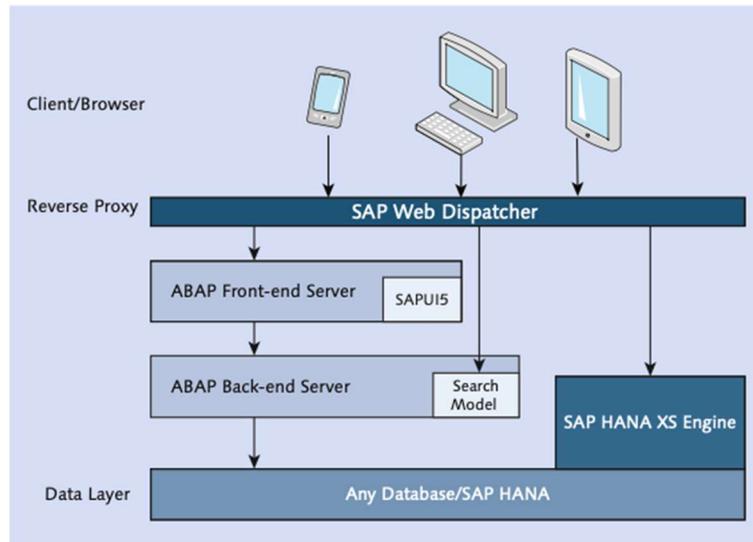
- Basado en Roles
- Responsive
- Simple
- Coherente
- Delightful

#### **2.2.2.2 Arquitectura SAP Fiori**

Desde una vista panorámica SAP Fiori se compone de una capa de cliente, proxy inverso, servidores ABAP, SAP HANA XS y una base de datos de fondo (véase Ilustración 6). Cada capa tiene un propósito claramente definido y se incluye con componentes que ayudan a cumplir ese propósito.

- **Cliente / navegador.** - La capa del cliente es donde tiene lugar el consumo real de las aplicaciones SAP Fiori. El launchpad de SAP Fiori es el punto de entrada para todas las aplicaciones de SAP Fiori en dispositivos móviles, computadoras de escritorio o tabletas.
- **Proxy inverso.** - Un proxy inverso es un tipo de servidor proxy que generalmente se encuentra detrás del firewall en una red privada y dirige las solicitudes del cliente al servidor de fondo apropiado. El proxy inverso proporciona un nivel adicional de seguridad y control para garantizar el flujo fluido del tráfico de red entre clientes y servidores. SAP recomienda usar SAP Web Dispatcher como un servidor proxy inverso. SAP Web Dispatcher solo se requiere como proxy inverso para aplicaciones analíticas y de hoja informativa.
- **Servidor ABAP front-end.** - El servidor front-end ABAP se recomienda para todos los tipos de aplicaciones. El servidor front-end ABAP contiene la capa de interfaz de usuario y los componentes de infraestructura, así como la puerta de enlace SAP con la habilitación de OData.
- **Servidor ABAP back-end.** - El servidor de back-end ABAP contiene la lógica empresarial de back-end y los modelos de búsqueda de aplicaciones de hoja informativa.
- **SAP HANA XS.** - SAP HANA XS solo se requiere para aplicaciones analíticas; replica datos del servidor de fondo y lo remodela usando modelos de datos virtuales (VDM).
- **Base de Datos.** - La base de datos de back-end puede ser cualquier base de datos para aplicaciones transaccionales o una base de datos SAP HANA para hoja informativa y aplicaciones analíticas.

Ilustración 6:Arquitectura SAP-Fiori



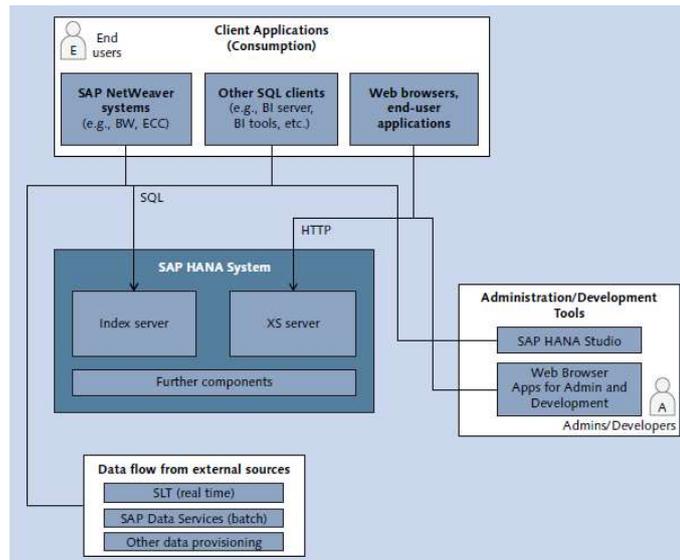
Fuente: SAP Fiori® Implementation and Development

### 2.2.3 SAP HANA

SAP HANA es una plataforma tecnológica para la nueva generación de las soluciones SAP. El sistema SAP HANA es un sistema bastante para entenderlo existen esquemas con una visión simplificada del mismo (véase Ilustración 7). Sus componentes son:

- **SAP HANA DB (o HANA DB)** se refiere a la tecnología de base de datos en sí.
- **SAP HANA Studio** se refiere al conjunto de herramientas que proporciona SAP para modelar dentro de la base de datos.
- **SAP HANA Appliance** se refiere a HANA DB como socio de Hardware presentadas en el certificado como un dispositivo. También incluye las herramientas de modelado de HANA Studio, así como herramientas de replicación y transformación de datos para mover datos a HANA DB
- **SAP HANA Aplicación en nube** se refiere a la infraestructura basada en la Computación en la nube para la entrega de aplicaciones (típicamente las aplicaciones existentes de SAP reescritas para ejecutarse en HANA).

Ilustración 7: Esquema simplificado de un sistema hanna



Fuente: SAP Fiori® Implementation and Development

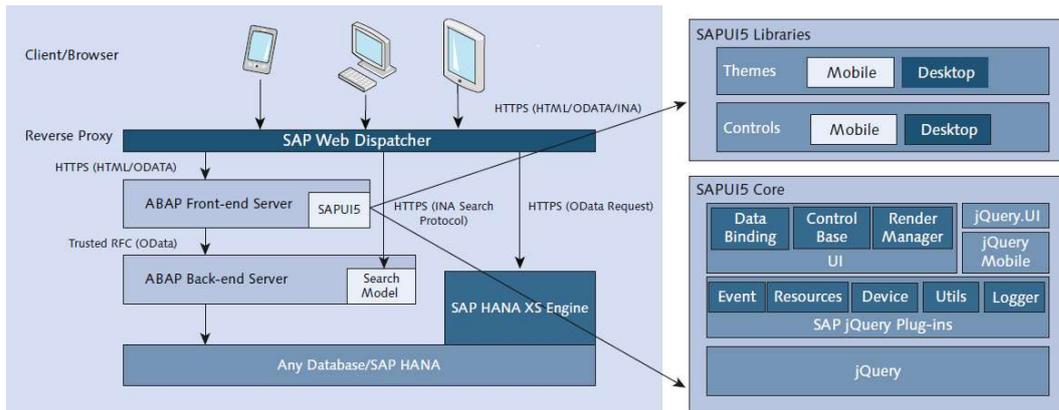
A pesar de ello para las interacciones de usuarios finales y desarrolladores en este sistema, solo dos son visibles:

**SAP HANA DB** un sistema de base de datos relacional que cumple los famosos requisitos de ACID (atomicidad, consistencia, aislamiento y durabilidad) y un **Servidor de Aplicación** de desarrollo servidor dentro del mismo entorno. Al llevar el desarrollo de aplicaciones. La característica principal es alto rendimiento que acelera la toma de decisiones y las acciones en tiempo real basadas en el procesamiento de la información, admite volúmenes importantes en cuanto a carga de trabajo.

#### 2.2.4 SAPUI5

Kit de herramientas de desarrollo de interfaz de usuario conformado por un dos componentes principales uno son las librerías SAPUI5 y un núcleo SAPUI5 para el desarrollo de aplicaciones basadas en HTML5 las cuales se pueden ejecutan en cualquier dispositivo, además nos proporciona un modelo de programación ligero para aplicaciones móviles y de escritorio usando el patrón de diseño MVC (véase Ilustración 8).

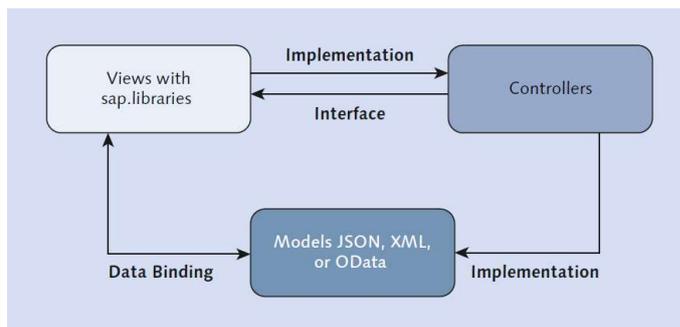
Ilustración 8:Arquitectura SAPUI5



Fuente: SAP Fiori® Implementation and Development

**Modelo Vista-Controlador (MVC).** - Es un patrón de diseño utilizado para separar la lógica de la interfaz de usuario de lógica de negocios. (véase Ilustración 9).

Ilustración 9:Patrón MVC en sapui5

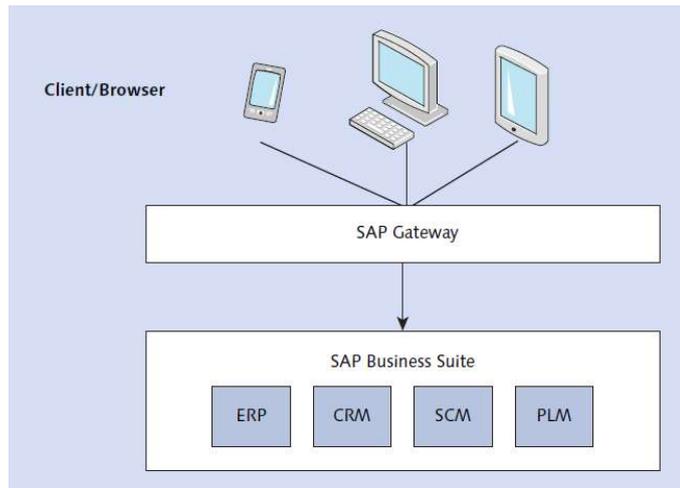


Fuente: SAP Fiori® Implementación and Development

### 2.2.5 SAP Gateway

Puerta de enlace SAP (SAP Gateway) es una tecnología que proporciona una forma sencilla de conectar dispositivos, entornos, y plataformas para aplicaciones SAP (véase Ilustración 10), utilizando cualquier lenguaje o modelo de programación sin la necesidad de conocimiento de SAP, al aprovechar los servicios REST y el protocolo OData. SAP Fiori se basa en los servicios OData para el intercambio de información entre el cliente y el servidor backend que puede ser cualquier componente de SAP Business Suite.

Ilustración 10:Arquitectura SAP-Gateway

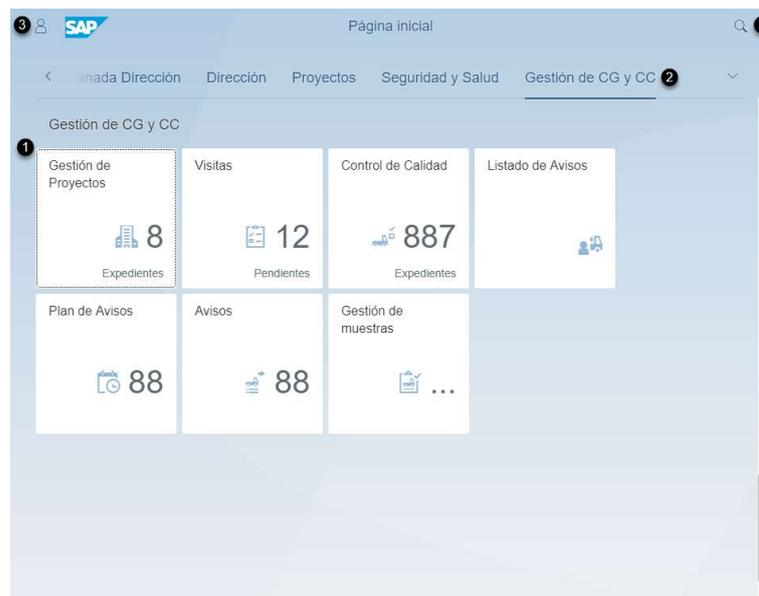


Fuente: SAP Fiori® Implementation and Development

### 2.2.6 SAP Fiori Launchpad

La plataforma de lanzamiento de SAP Fiori (SAP Fiori Launchpad) es el punto de entrada para todas las aplicaciones de SAP Fiori (véase Ilustración 11). Está basado en roles, permite la personalización y cuenta con capacidades en tiempo real. Está diseñado de acuerdo con una experiencia de usuario simple e intuitiva y no requiere mucho esfuerzo en cuanto a aprendizaje y practica para desenvolverse en el con facilidad.

Ilustración 11:SAP-Fiori Launchpad



En la Ilustración 11 podemos distinguir:

1. Aplicaciones
2. Grupos

3. Área de Usuario
4. Búsqueda

## 2.3 OData

Open Data Protocol (OData) es un protocolo basado en REST que se utiliza para exponer, acceder y modificar información de diferentes fuentes. Fue construido sobre tecnologías web como HTTP, Atom Publishing Protocol (AtomPub), XML y JSON para proporcionar acceso a información de una manera estructurada.

## 2.4 Herramientas de desarrollo

Además de la herramienta específica para el desarrollo de aplicaciones SAPUI5 "SAP WebId" para poder desarrollar para SAP-ERP se nos proporcionan diversas herramientas como transacciones:

Herramienta	Transacción	Objetos que se editan
Editor ABAP	SE38	Códigos de programas
Dictionary ABAP	SE11	Tablas y otros objetos
Bibliotecas de funciones	SE37	Módulos de función y otros grupos
Navegador de Objetos	SE80	Navegador de objetos
SAP Gateway service Builder	SEGW	Implementación de servicios
Generador de perfiles	PFCG	Administrar Roles

### 2.4.1 Diccionario de datos (SE11)

El diccionario ABAP contiene todos los metadatos acerca del sistema SAP. Está muy ligado con la plataforma de trabajo de ABAP en el que cualquier referencia a los datos se obtiene del diccionario (por ejemplo, tablas, vistas, tipos de datos, etc.).

Tipos de datos más importantes del diccionario:

- **Tablas:** son contenedores de datos que existen en la base de datos relacional subyacente.
- **Índices,** proporcionan accesos rápidos a los datos de las tablas para aquellas selecciones usadas con mayor frecuencia.
- **Vistas:** Una vista es una tabla virtual que no contiene ningún dato físicamente.
- **Estructuras:** son tipos de datos complejos que contienen varios campos.

- **Elementos de datos:** proporcionan contenido semántico a una tabla o campo estructurado.
- **Dominios:** definen las características estructurales de los elementos de datos.
- **Ayudas de búsqueda:** proporciona una estrategia de búsqueda avanzada cuando un usuario quiere ver el posible valor que tiene un campo. Objetos bloqueados implementan bloqueos a nivel de aplicación cuando se modifican datos.

- 

Pantalla inicial de la transacción (véase Ilustración 12).

Ilustración 12: Transacción SE11

#### 2.4.2 Editor ABAP (SE38)

Mediante este editor podremos crear, modificar y visualizar los programas ABAP. Esto incluye los objetos estándar del sistema (los que proporciona SAP) y también los programas "Z" (los desarrollos específicos de cada cliente de SAP)

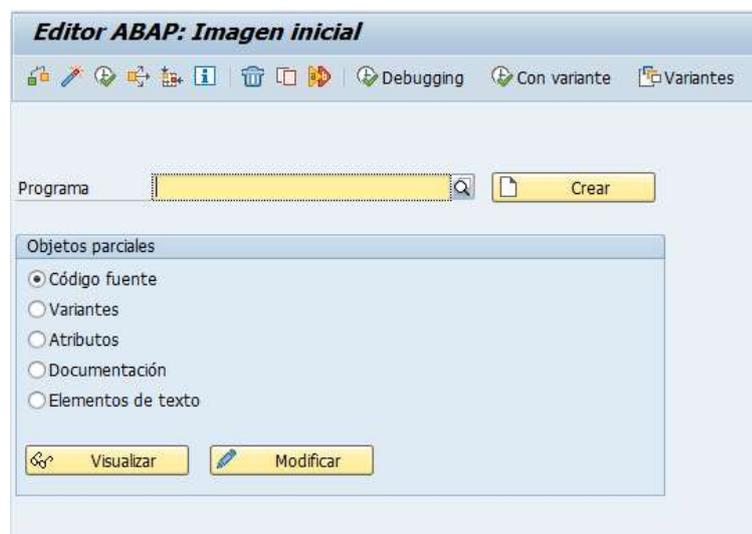
Cuando ingresamos a la transacción veremos la pantalla de la Ilustración 13, un cuadro de texto para introducir el nombre del objeto con el que deseamos trabajar y debajo de él podemos marcar la característica del objeto que deseamos visualizar o modificar (véase Ilustración 13: Transacción SE38 Ilustración 13).

Las posibilidades existentes son:

- **Código fuente:** aquí veremos el código ABAP del objeto.
- **Variantes:** aquí veremos las variantes existentes para el objeto, si es que las tiene.

- **Atributos:** aquí veremos los atributos del programa, como ser título, idioma, tipo, estatus, etc.
- **Documentación:** aquí veremos la documentación existente para el programa, si es que la tiene.
- **Elementos de texto:** aquí veremos los elementos de texto del programa.

Ilustración 13: Transacción SE38



### 2.4.3 Bibliotecas de funciones (SE37)

Mediante esta transacción podremos visualizar, modificar y crear nuevos módulos de funciones, así como también grupos de funciones.

Ilustración 14: Transacción SE37



Para crear un nuevo módulo de funciones debemos escribir el nombre del módulo de funciones en el campo de la pantalla (véase Ilustración 14). El nombre debe comenzar con Z y luego presionamos el botón "Crear".

Luego de creado el módulo de funciones, debemos especificar los parámetros de entrada, esto lo haremos en la pestaña IMPORTING.

Los parámetros de salida los especificaremos en la pestaña EXPORTING.

Si utilizamos tablas las especificaremos en la pestaña TABLES, aunque esta opción se encuentra obsoleta.

En la pestaña CHANGING podremos especificar estructuras de datos que cambiaran su contenido.

Las excepciones las declararemos en la pestaña EXCEPTIONS.

Y finalmente escribiremos el código fuente del módulo de funciones en la pestaña CÓDIGO FUENTE.

#### 2.4.4 Navegador de Objetos (SE80)

La gran ventaja que presenta esta transacción tiene que ver con la visualización clara que nos brinda del árbol de objetos (véase Ilustración 15) a la izquierda de la pantalla y la vista preliminar del objeto a la derecha de la pantalla desde el árbol se nos permite buscar y editar cualquier objeto del repositorio del ERP. Algunos de los tipos de objetos más importantes a los cuales nos da acceso el navegador son: programas, paquete, clases, grupo de funciones, interfases, entre otros.

Ilustración 15: Transacción SE80



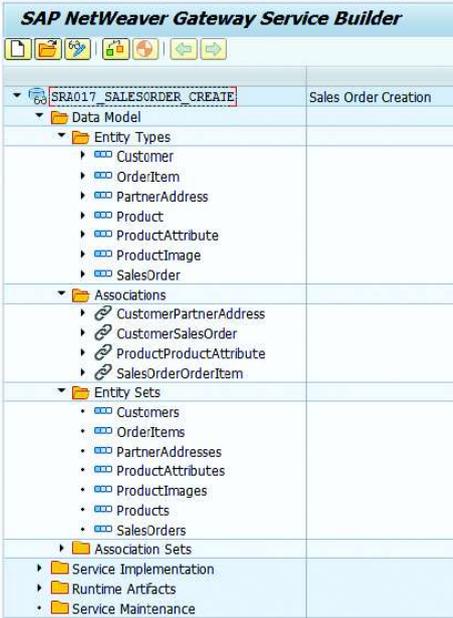
#### 2.4.5 SAP Gateway Service Builder (SEGW)

SAP Gateway Service Builder es la interfaz central de SAP Gateway (acceso a través de la transacción SEGW), incluye tanto el desarrollo basado en código de servicios OData y generación de servicios sin código.

SAP Gateway Service Builder contiene todas las funciones relevantes para el modelado y desarrollo de servicios

OData en SAP Gateway, y soporta todo ciclo de vida del desarrollo de un servicio OData. Su objetivo principal es proporcionar soporte integral para construir servicios OData de una forma declarativa o para reutilizar objetos comerciales existentes en el sistema SAP Business Suite. El modelado e implementación del servicio en el entorno sigue un enfoque basado en proyectos (véase Ilustración 16) podemos ver un ejemplo de la estructura de un proyecto en la transacción SEGW.

Ilustración 16:Sap-Gateway Service Builder



### 2.4.6 Generador de perfiles (PFCG)

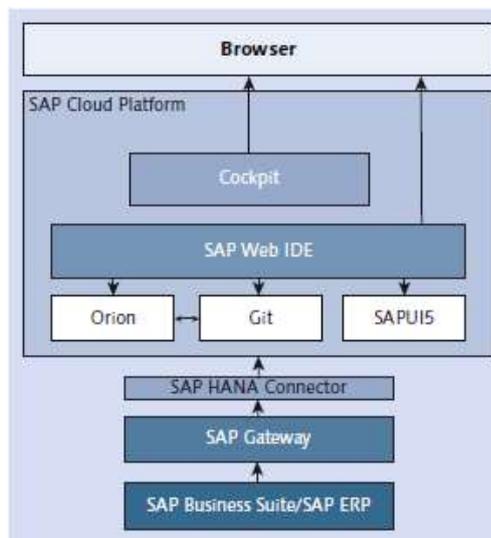
En la transacción PFCG se cubren varios aspectos: la administración de autorizaciones y perfiles, así como la definición de menús de usuario. También el mantenimiento de roles.

En la vida real, un rol corresponde al trabajo que realiza un grupo de personas dentro de una empresa o el puesto que ocupan, por ejemplo, un comercial o alguien que trabaja en soporte. En la transacción PFCG, los roles se definen coincidiendo con los roles de la vida real que desempeñan las personas en una situación laboral. Más específicamente, un rol dentro de la transacción PFCG es un conjunto de permisos que las personas dentro de roles específicos necesitarían para su trabajo diario. Por ejemplo, un vendedor puede tener un rol que requiera transacciones para agregar nuevos clientes, mostrar detalles del cliente o ingresar un nuevo cliente potencial.

### 2.4.7 SAP Web IDE

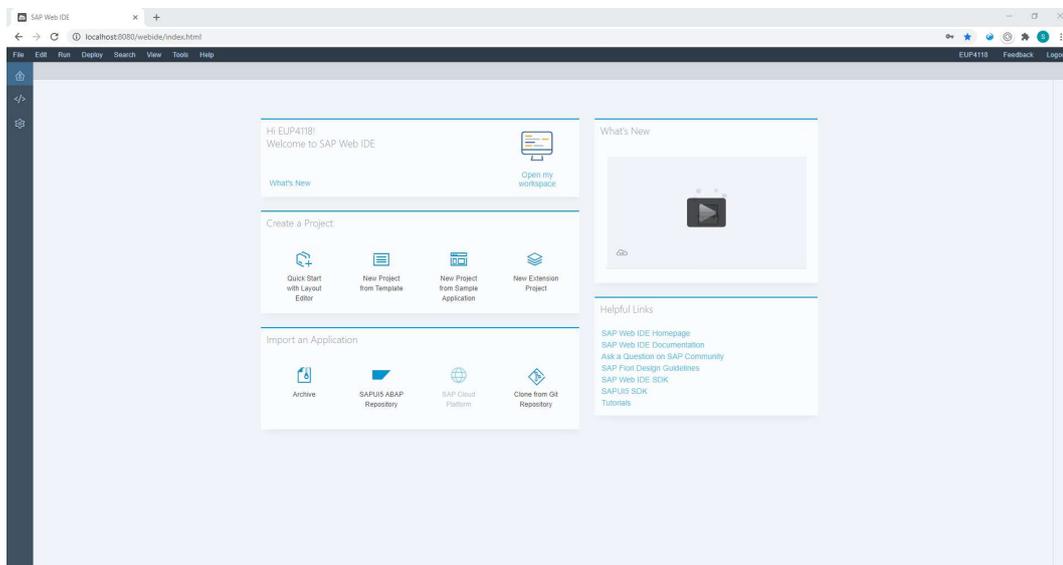
SAP Web IDE es una herramienta de desarrollo basada en arquitectura web (véase Ilustración 18) que se ejecuta en el navegador, junto con otros componentes (véase Ilustración 17) forma un entorno de desarrollo completo que permite a los desarrolladores crear aplicaciones SAP Fiori. SAP Web IDE acelera la creación de aplicaciones modernas para dispositivos móviles y de escritorio utilizando las últimas tecnologías de interfaz de usuario. Es una herramienta flexible con editores de código robustos además permite desarrollar aplicaciones de manera rápida y eficiente.

Ilustración 17:Arquitectura SAP WebId



Fuente: SAP Fiori® Implementation and Development

Ilustración 18:Interfaz WebId



## **2.5 Lenguajes de programación**

### **2.5.1 ABAP**

Advanced Business Application Programming es un lenguaje de programación estructurado de cuarta generación y está orientado a eventos es decir, no sigue la típica estructura TOP-DOWN (ejecución de arriba a abajo) de los lenguajes estructurados como por ejemplo C. Tiene sus raíces en otros lenguajes como COBOL o PASCAL. En los 90 con el auge de los lenguajes orientados a objetos se adaptó y surgió ABAP OO con esto se acercó a otros lenguajes como Java.

Hace uso de sentencias Open SQL para conectarse con prácticamente cualquier base de datos. Cuenta con miles de funciones para el manejo de archivos, bases de datos, fechas, etc. Permite conexiones RFC (Remote Function Calls) para conectar a los sistemas SAP con cualquier otro sistema o lenguaje de programación.

Los eventos en ABAP pueden ser cualquier acción ejecutada por el usuario y que implica una reacción del sistema mediante ejecución de código.

### **2.5.2 JavaScript**

JavaScript es un lenguaje de programación orientado a objetos usado sobre todo en el diseño de páginas web, ya que permite dotar a estas de una mayor interactividad y dinamismo, haciendo la experiencia del usuario mucho mejor y enriquecedora. Es reconocido junto a HTML y CSS como uno de los lenguajes de programación principales en el desarrollo web. La sintaxis usada es parecida a C, aunque adopta nombres y convenciones del lenguaje de programación java.

Una de las características de JavaScript es que es un lenguaje interpretado, esto significa que no requiere de compilación previa y sus instrucciones son interpretadas directamente, en el caso del desarrollo web por el navegador, sin necesidad de otros programas o interpretes intermedios.

### **2.5.3 XML**

El lenguaje de marcado extensible (XML) es un formato de texto simple y muy flexible derivado de SGML (ISO 8879). Originalmente diseñado para enfrentar los desafíos de la publicación electrónica a gran escala, XML también está desempeñando un papel cada vez más importante en el intercambio de una amplia variedad de datos en la Web y en otros lugares.

Dentro del TFG se usa este lenguaje para definir las vistas en las aplicaciones SAPUI5.

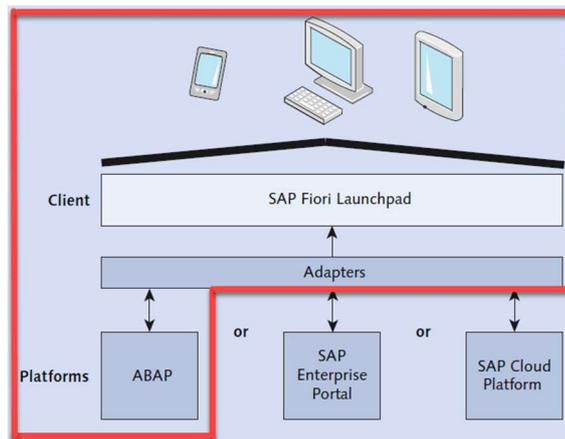
## 2.6 Arquitectura del sistema

SAP pone a disposición de sus clientes tres opciones de despliegue de SAP-Fiori Launchpad y por ende de las aplicaciones SAPUI5 objeto de este TFG. Dependiendo del landscape y de las aplicaciones que se quieran implementar tenemos:

- **ABAP backend servidor en combinación con SAP Gateway:**  
Este es el modelo tradicional de implementación y es el recomendado para clientes que no cuentan con un SAP Enterprise Portal. Esta es la única combinación que soporta todos los tipos de aplicaciones Fiori (Transactional, fact Sheet y analytical).
- **SAP Enterprise Portal:** Este tipo de despliegue es recomendado para clientes quienes cuentan y quieren reusar su SAP EP y ejecutar el SFL.
- **SAP HANA Cloud Platform:** Otra opción disponible es ejecutar SAP Fiori Launchpad en SAP HANA Cloud Platform (SAP HCP) como Software as a Service (SaaS).

En nuestro caso la implementación se realizará sobre la primera opción de arquitectura (véase Ilustración 19) ya que CEMOSA cuenta con los recursos necesarios en cuanto a componentes dentro de su SAP- Business suite.

Ilustración 19: Opciones de despliegue

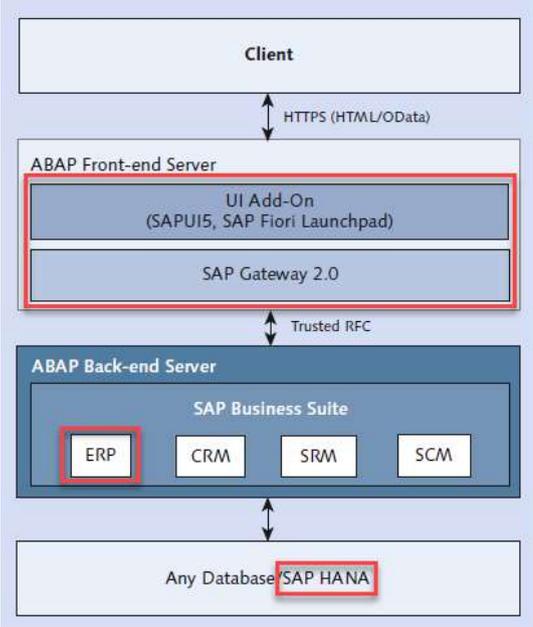


Fuente: SAP Fiori® Implementation and Development

Dentro de esta opción de despliegue existen dos variantes dependiendo si el Gateway está incrustado en el mismo servidor ABAP o está en un servidor independiente. Para nuestro desarrollo se ha escogido la alternativa de servidores independientes (véase Ilustración 20). Al tener este tipo de arquitectura tenemos

un mayor control sobre cada servidor, aunque ello implique un mayor empleo de recursos en mantenimiento.

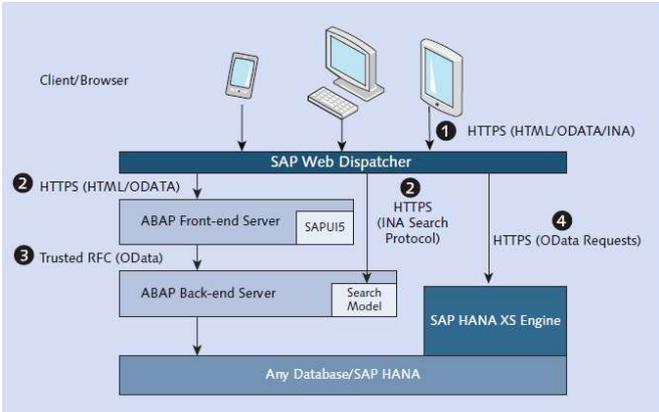
Ilustración 20:Arquitectura servidores independientes



Fuente: SAP Fiori® Implementation and Development

Tenemos en el nivel inferior una base de datos HANA que tiene comunicación directa con el servidor backend que es el componente ERP del SAP-Business Suit a su vez este se comunica con el servidor front end donde tenemos operativo el SAP Gateway, Los componentes SAPUI5 y el Launchpad de SAP-Fiori. La comunicación es algo importante al tener esta arquitectura y es aquí donde el protocolo OData toma relevancia ya que la mayoría de las comunicaciones entre nuestros componentes se realiza sobre este (véase Ilustración 21).

Ilustración 21: Protocolos de comunicación



Fuente: SAP Fiori® Implementation and Development

# 3

## Sistema de Gestión de Avisos

Este capítulo está dedicado al proceso de desarrollo de la solución planteada para dar respuesta uno de los problemas que motivan este TFG, la falta de una herramienta que automatice y homogenice el proceso de gestión de solicitudes de trabajo por parte de los clientes. Hemos denominado a esta herramienta “*sistema de gestión de avisos*” y estará formado por cuatro aplicaciones SAPUI5 independientes, las cuales implementarán todas las funcionalidades requeridas.

- APP Alta avisos
- APP Plan de avisos
- APP Avisos
- APP Listado avisos

Aunque las aplicaciones son independientes implementarán navegación entre ellas que permite la comunicación e integración entre las mismas.

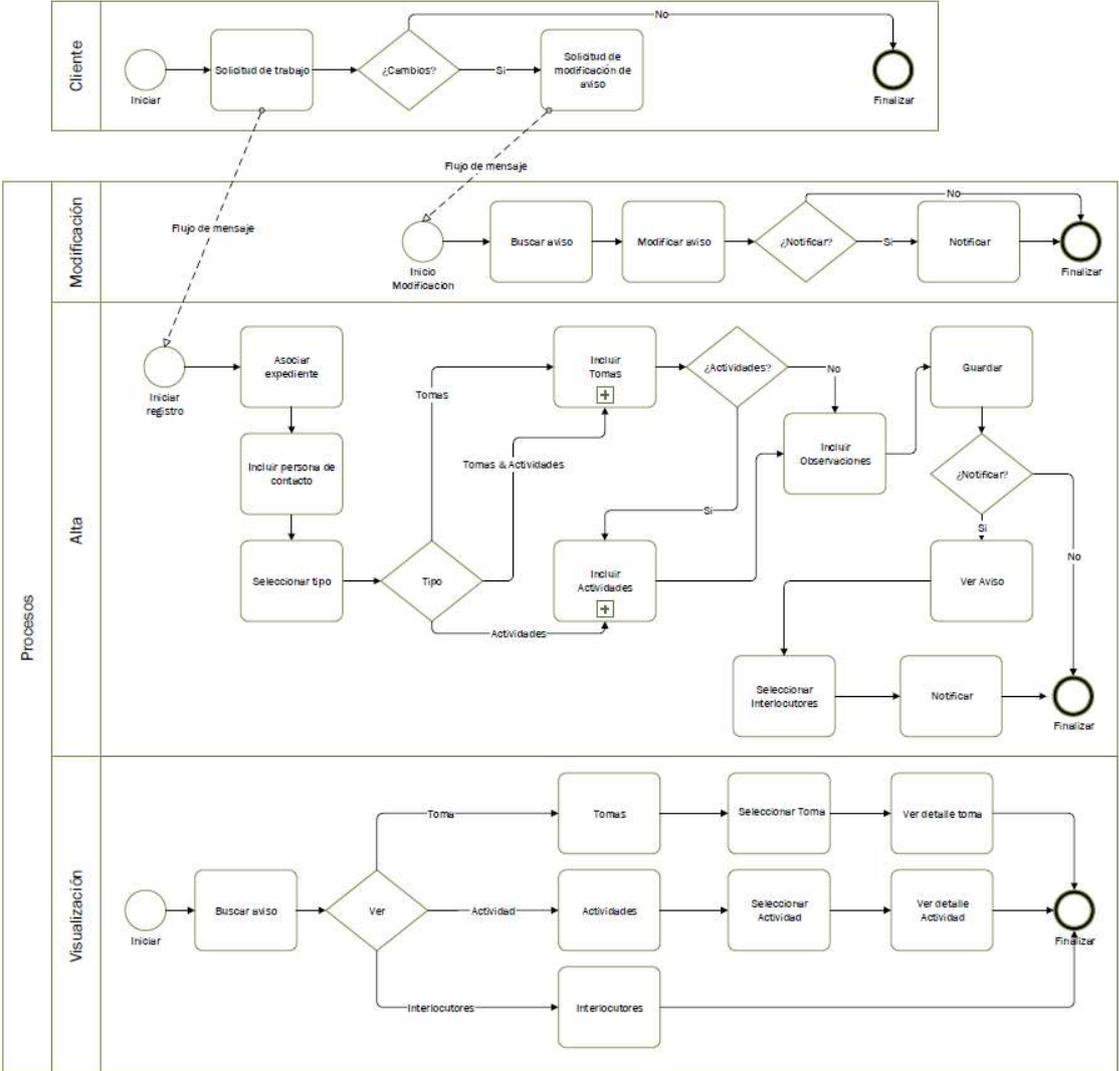
### 3.1 Análisis

#### 3.1.1 Descripción funcional

La funcionalidad que cubre el sistema de gestión de avisos de manera general es la de gestionar las solicitudes de trabajo de los clientes (Avisos). Esta gestión como tal incluye varios procesos (véase Ilustración 22). Realizara esta tarea mediante operaciones CRU (Create, Read, Update) además de funcionalidades adicionales como la notificación vía mail de alta o cambios en las solicitudes o

funcionalidades implícitas a la gestión como por ejemplo el análisis de la carga de trabajo diario de los operadores de calle.

Ilustración 22: Diagrama de procesos gestión de avisos



Seguindo el flujo normal de los procesos de gestión de avisos podemos describir como funcionalidades independientes y específicas las operaciones CRUD ya que conviven dentro de las aplicaciones que forman el sistema.

**3.1.1.1 Alta aviso (Create)**

Esta funcionalidad se implementará en la aplicación de "Alta de avisos". Es el primer paso del flujo del proceso de gestión, se refiere al registro de la solicitud de trabajo/s del cliente en el sistema la cual incluye los siguientes pasos:

1. Seleccionar expediente de obra. - El expediente de obra recoge todos los materiales y ensayos sobre estos que contrato el cliente y se contemplan las siguientes casuísticas:
  - a. El cliente aún no tiene un expediente de obra asociado (cliente nuevo/ pendiente de formalizar la oferta), en este caso se asigna al aviso un expediente genérico.
  - b. Se busca y selecciona el expediente de obra que tiene contratado el cliente.
2. Introducir los datos de la persona de contacto en obra. - Nombre, teléfono, email.
3. Seleccionar el tipo de trabajos tomas / actividades / ambos.
4. Introducir los datos de toma. - Fecha, hora, operador asignado (existe la posibilidad de dar de alta la toma pendiente de asignación a operador) y material. Al seleccionar los materiales se puede escoger entre seleccionarlos del expediente o seleccionar desde un listado de todos los materiales disponibles en el sistema además existe datos adicionales que se deben introducir si el tipo de material es Hormigón. Por último, si se han introducido los materiales del expediente se pueden asociar a muestras que tenga programadas el expediente para ese material.
5. Introducir los datos de actividad. - Fecha, hora, operador asignado (existe la posibilidad de dar de alta la actividad pendiente de asignación a operador) y material este último igual que en las tomas se puede seleccionar del expediente o del listado de todos los materiales con servicios disponibles. Por último, se asocia el servicio que se realizara in situ sobre el material.
6. Añadir observaciones para el operador y guardar el aviso.

### **3.1.1.2 Detalle aviso (Read)**

Esta funcionalidad se incluye en la aplicación "Avisos". Una vez un aviso este registrado en el sistema desde las distintas aplicaciones que lo componen los usuarios pueden visualizar en detalle todos los datos de un aviso: los datos de contacto, las tomas, las actividades y las observaciones.

Desde el punto de vista de la presentación de los avisos para cada grupo de usuarios esta funcionalidad varía así por ejemplo los operadores de calle deben poder visualizar sus avisos diarios en formato de listado desde el cual tendrían disponible los datos principales de la obra y los trabajos a realizar y el cual les permite de manera rápida acceder al detalle completo. Se debe combinar el listado

con herramientas de búsqueda por obra y filtros por fecha. También deben disponer de la presentación en formato de calendario para evaluar la carga de trabajo diaria y semanal de uno o varios operadores.

Por otra parte, los técnicos de laboratorio y el responsable de los operadores deben disponer de un listado de tomas y actividades independientes en formato de listado con distintos filtros para poder buscar un trabajo en concreto. En el listado se visualizarán los datos más relevantes para ellos como el expediente al que pertenecen, el material y el operador asignado.

Este mismo grupo de usuarios también lleva a cabo tareas de auditoria con este listado.

### **3.1.1.3 Modificar Aviso (Update)**

Esta funcionalidad se implementará en varias aplicaciones "Avisos" "Listado de avisos" y "Plan Avisos". Si surge la necesidad de modificar algún dato del aviso ya registrado el sistema permite realizar esta función, aunque no en la totalidad de los datos, solo se permite modificar los datos de fecha hora operador asignado y observaciones.

La modificación de estos datos en su conjunto se realiza de manera individual para cada aviso y como un caso especial el sistema permite modificar en masa los avisos, pero solo en cuanto a la asignación de estos a un operador esto es indispensable a la hora de que los encargados de los operadores gestionen la carga de trabajo entre ellos.

Dentro de la modificación de un aviso también se contempla la funcionalidad de anular este ya que si bien la percepción del usuario al realizar esto es que este se ha eliminado el aviso porque no lo visualizará en las aplicaciones, la funcionalidad de anular es solo el cambio de estatus del aviso ya que los datos de este no se eliminan del sistema.

### **3.1.1.4 Notificación**

Se implementará en la aplicación "Avisos". Esta es una funcionalidad adicional ya que no forma parte de las operaciones CRU, permite de manera opcional poder notificar al cliente vía correo electrónico del registro o cambios de un aviso.

### 3.1.1.5 Partes de trabajo

El listado de tomas y actividades cuenta con la funcionalidad de poder exportarlo a Excel el cual pueden imprimir y se usan como los partes diarios de trabajos de un operador.

### 3.1.2 Requisitos Generales

En este apartado se especifican requisitos generales para todas las aplicaciones que son objeto de desarrollo de este TFG. Tanto para las de este capítulo como para las del capítulo 4 y 5.

#### 3.1.2.1 Funcionales

<b>RFGS001</b>	<i>Acceso en base a roles (Apps/Datos)</i>
<b>Descripción</b>	<i>El acceso a las aplicaciones y datos se hace mediante roles definido para los grupos de usuarios</i>

#### 3.1.2.2 No Funcionales

<b>RNFGS001</b>	<i>Directrices de diseño</i>
<b>Descripción</b>	<i>El sistema debe desarrollarse siguiendo los cinco pilares establecidos en las directrices de diseño UX de SAP -Fiori.</i>

<b>RNFGS002</b>	<i>Responsive</i>
<b>Descripción</b>	<i>Las aplicaciones desarrolladas deben poseer un diseño "Responsive" a fin de garantizar la adecuada visualización en múltiples dispositivos.</i>

<b>RNFGS003</b>	<i>Usabilidad</i>
<b>Descripción</b>	<i>El tiempo de aprendizaje del sistema por un usuario deberá ser corto se tendrá en cuenta siempre a la hora del diseño e implantación la facilidad de uso como factor determinante.</i>

<b>RNFGS004</b>	<i>Ejecución desde FLP</i>
<b>Descripción</b>	<i>Todas las aplicaciones resultantes deben implementar la ejecución desde FLP (SAP Fiori Launchpad).</i>

### 3.1.3 Catálogo de requisitos

#### 3.1.3.1 Funcionales

<b>RFGA001</b>	<i>Gestión de avisos de toma/actividad</i>
<b>Descripción</b>	<i>El sistema deberá permitir a los usuarios (Actores) realizar las tareas de gestión de los avisos (solicitudes) de toma de materiales o actividades insitu solicitadas por los clientes. Las tareas de gestión incluyen operaciones CRU.</i>
<b>Requisitos hijos</b>	<b>RFGA002 RFGA003 RFGA004</b>
<b>RFGA002</b>	<i>Crear aviso</i>
<b>Descripción</b>	<i>El sistema deberá incluir la funcionalidad dar de alta un aviso con los datos de la solicitud del cliente.</i>
<b>Requisitos hijos</b>	<b>RFGA013</b>
<b>RFGA003</b>	<i>Visualizar aviso (Detalle)</i>
<b>Descripción</b>	<i>Los usuarios deben poder visualizar todos los elementos que componen a un aviso.</i>
<b>RFGA004</b>	<i>Modificar aviso</i>
<b>Descripción</b>	<i>El sistema debe incluir la posibilidad de modificar los datos de un aviso si fuera necesario.</i>
<b>RFGA005</b>	<i>Anular aviso</i>
<b>Descripción</b>	<i>Un aviso debe poder ser anulado desde el sistema mas no eliminado.</i>
<b>RFGA006</b>	<i>Visualizar interlocutores</i>
<b>Descripción</b>	<i>Los usuarios deben poder visualizar a los interlocutores del expediente asociado al aviso y sus datos de contacto.</i>
<b>RFGA007</b>	<i>Notificar trabajo</i>

<b>Descripción</b>	<i>El sistema debe permitir que los usuarios puedan notificar el trabajo realizado a los interlocutores del expediente asociado al aviso.</i>
--------------------	---

<b>RFGA008</b>	<i>Carga de trabajo</i>
<b>Descripción</b>	<i>Los usuarios deben poder visualizar la carga de trabajo en periodos diarios, semanales y mensuales de un operador o varios.</i>

<b>RFGA009</b>	<i>Búsqueda</i>
<b>Descripción</b>	<i>El sistema debe permitir a los usuarios poder buscar uno o un grupo de avisos mediante el uso de filtros.</i>

<b>RFGA010</b>	<i>Listar avisos</i>
<b>Descripción</b>	<i>El sistema debe implementar un listado de avisos</i>
<b>Requisitos hijos</b>	<b>RFGA015</b>

<b>RFGA011</b>	<i>Modificar varios avisos a la vez</i>
<b>Descripción</b>	<i>El sistema debe permitir a los usuarios el poder modificar más de un aviso al mismo tiempo.</i>

<b>RFGA012</b>	<i>Exportar listado de avisos</i>
<b>Descripción</b>	<i>El sistema debe implementar la funcionalidad necesaria para poder exportar el listado de avisos a una hoja de cálculo.</i>

<b>RFGA013</b>	<i>Asociar Expediente</i>
<b>Descripción</b>	<i>En el momento de la creación de un aviso el sistema debe permitir de manera opcional asociar el aviso a un expediente</i>
<b>Requisitos hijos</b>	<b>RFGA014</b>

<b>RFGA014</b>	<i>Buscar Expediente</i>
<b>Descripción</b>	<i>El sistema debe permitir realizar la búsqueda de un expediente.</i>

<b>RFGA015</b>	<i>Opciones de personalización</i>
<b>Descripción</b>	<i>Los usuarios deben poder personalizar el listado de avisos moviendo, agregando y/o quitando columnas, así como establecer un orden para alguna de ellas.</i>

<b>RFGA016</b>	<i>Navegar a detalle</i>
<b>Descripción</b>	<i>El sistema debe implementar la navegación al detalle del aviso desde la tarea en el calendario de trabajo y también desde el listado de avisos.</i>

### 3.1.3.2 No Funcionales

<b>RNFGA001</b>	<i>Usar código de colores</i>
<b>Descripción</b>	<i>Se debe usar un código de colores para mejorar diferenciación de las tareas asociadas a los avisos.</i>

### 3.1.4 Casos de Uso

Se han elaborado casos de uso atendiendo a los perfiles de los usuarios finales de las aplicaciones.

#### 3.1.4.1 Especificación

<b>CUGA001</b>	<i>Crear aviso</i>
<b>Actores</b>	<i>Técnico de laboratorio, Operador de calle</i>
<b>Descripción</b>	<i>Una vez llegue una solicitud de trabajo este caso de uso permitirá a los actores que sean personal de laboratorio registrar una solicitud de trabajo en el sistema. Al crear del aviso se incluirán, además de los datos generales, las tomas y actividades que lo compongan.</i>
<b>Requisito</b>	<i>RFGA002</i>

<b>CUGA002</b>	<i>Visualizar aviso</i>
<b>Actores</b>	<i>Técnico de laboratorio, Operador de calle</i>

<b>Descripción</b>	<i>Cuando se requiera consultar los datos de un aviso este caso de uso permitirá a los actores obtener los datos completos del mismo del sistema y visualizarlos.</i>
<b>Requisito</b>	<i>RFGA003</i>

<b>CUGA003</b>	<i>Modificar aviso</i>
<b>Actores</b>	<i>Técnico de laboratorio, Operador de calle</i>
<b>Descripción</b>	<i>Una vez un aviso se halle en el sistema este caso de uso permite recuperarlo y modificar los datos de esa solicitud de trabajo, un aviso se considera modificado si han modificado alguna o todas sus tomas o actividades.</i>
<b>Requisito</b>	<i>RFGA004</i>

<b>CUGA004</b>	<i>Anular aviso</i>
<b>Actores</b>	<i>Técnico de laboratorio, Operador de calle</i>
<b>Descripción</b>	<i>El caso de uso permitirá a los actores establecer como anulado el estado de una toma o actividad. Si se anulan todos los elementos de un aviso este pasa a considerarse anulado, pero si solo se eliminan algunos el aviso se considera anulado parcialmente.</i>
<b>Requisito</b>	<i>RFGA005</i>

<b>CUGA005</b>	<i>Buscar aviso</i>
<b>Actores</b>	<i>Técnico de laboratorio, Operador de calle</i>
<b>Descripción</b>	<i>Una vez una solicitud de trabajo este ya registrada en el sistema este caso de uso permitirá la búsqueda de esta mediante alguna herramienta especificando algún parámetro del aviso objeto de la búsqueda.</i>
<b>Requisito</b>	<i>RFGA009</i>

<b>CUGA006</b>	<i>Descargar listado avisos</i>
<b>Actores</b>	<i>Técnico de laboratorio, Operador de calle</i>
<b>Descripción</b>	<i>Este caso de uso permite obtener por medio de la aplicación un listado de avisos en un archivo Excel para su tratamiento fuera del sistema.</i>
<b>Requisito</b>	<i>RFGA012</i>

<b>CUGA007</b>	<i>Solicitar trabajo</i>
<b>Actores</b>	<i>Técnico de laboratorio, Operador de calle, Cliente</i>
<b>Descripción</b>	<i>El caso de uso es el que da origen a toda la gestión de avisos dentro del sistema se da en el escenario inicial para cualquier aviso el cliente realiza una solicitud de trabajo.</i>
<b>Requisito</b>	-

<b>CUGA008</b>	<i>Notificar</i>
<b>Actores</b>	<i>Técnico de laboratorio, Cliente</i>
<b>Descripción</b>	<i>Una vez se ha dado de alta un aviso en el sistema o se han modificado los datos de alguno existente o se ha anulado un aviso existente. Este caso de uso permite a los técnicos de laboratorio notificar de dichas acciones al cliente.</i>
<b>Requisito</b>	<i>RFGA007</i>

### 3.1.4.2 Diagramas

Ilustración 23: Casos de uso de la gestión de avisos



### 3.1.5 Actores del Sistema

**Cliente.** - Es la persona/empresa que inicia el proceso ya que es el que realiza la solicitud de trabajos que da lugar a todas las actividades definidas en los casos de uso.

**Técnico de laboratorio.** - Personal de laboratorio el cual se encarga entre otras cosas de atender las solicitudes de trabajo que llegan y las gestiona.

**Operador de calle.** - Los operadores de calle son aquel personal que se desplaza hasta la ubicación de la obra, realiza las tomas de muestras y las transporta hasta el laboratorio para su posterior ensayo. También realiza los ensayos insitu los cuales son ensayos que por su naturaleza se deben realizar en la obra misma.

**Responsable de operadores.** - Es la persona encargada de organizar el trabajo diario de los operadores respecto a toma de muestras y actividades.

## **3.2 Diseño**

### **3.2.1 Modelos entidad-relación BBDD**

Los datos de los avisos junto con los de las tomas y actividades que los componen deben almacenarse en base de datos. Para ello es necesario definir las tablas físicas y su estructura que es donde se llevara a cabo esta tarea de almacenamiento y posteriormente consultas. Es necesario especificar un modelo de tres tablas a implementar junto con las relaciones con algunas de tablas ya existentes en el sistema como las tablas de: proyectos, elementos peps, clientes, usuario, muestras y servicios con los cuales por definición estarán relacionados los avisos las tomas y las actividades ya que no serán entidades aisladas en el sistema.

En el modelo (véase Ilustración 24) si bien las tablas que se crearan estarán destinadas a formar parte del módulo de laboratorio intervienen otras tablas de los módulos PS, LB y HR. Esto es debido a las relaciones intrínsecas de los avisos con otros elementos del sistema.

Partimos de la premisa de que un aviso debe pertenecer a un expediente el cual a su vez pertenece a un pep definido en un proyecto. Dentro de un expediente existen distintos interlocutores tanto internos como externos asociados a la gestión y control del trabajo uno de ellos es el cliente.

Pasando a nivel de tomas y actividades, que componen un aviso, estas representan solicitudes de trabajo las cuales deben estar asociadas a las tablas de muestras y servicios respectivamente para especificar dichos trabajos. También existe una relación directa con los usuarios del sistema ya que en las tablas de avisos, tomas y actividades se almacenan datos de auditoria.



para los datos propios de las tomas y otra para los datos propios de las actividades. La tabla de tomas cuenta con la clave foránea de la de avisos que junto con su identificador de toma forman el conjunto de claves de la tabla para la tabla de actividades sucede lo mismo con el id de actividad.

El resto de las tablas que componen el modelo son las siguientes:

- **KNA1.-** Tabla maestro de clientes.
- **PA0001.-** Tabla maestro de personal.
- **IHPA.** - Tabla de interlocutores.
- **PROJ.** - Tabla maestro de proyectos.
- **PRPS.** - Tabla maestro de elementos pep.
- **ZTLB\_OBRAS.** - Tabla maestro de obras.
- **MARA.** - Tabla maestro de materiales
- **ZTLB\_TIPMATERIAL.** - Tabla de tipos de material.
- **ZTLB\_MURELSER.** - Tabla de servicios por muestra.
- **USR01.-** Tabla maestro de usuarios.

### 3.2.2 Modelado de Servicios Odata

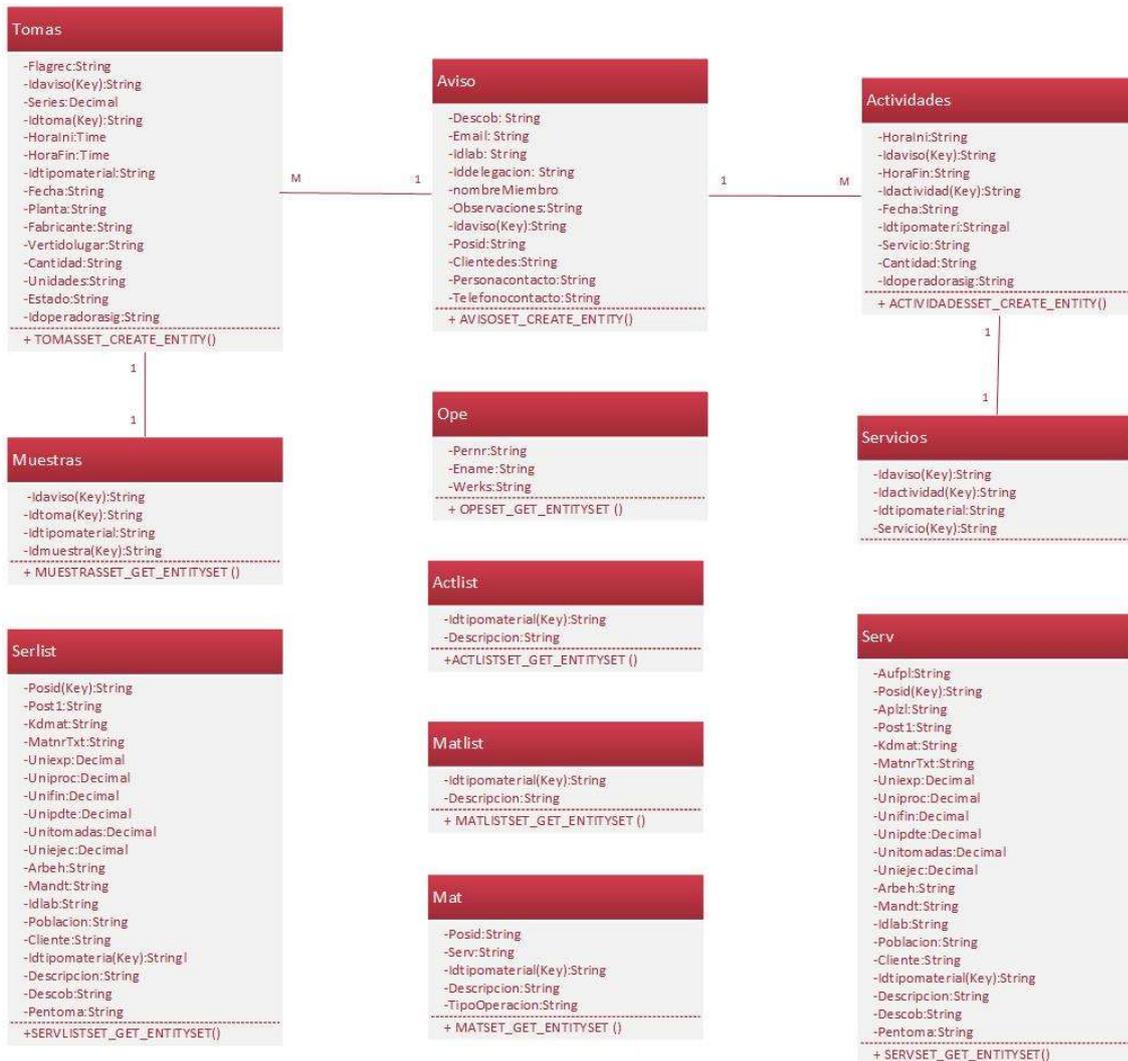
Se implementarán tres servicios Odata, para cada uno de ellos es necesario definir un modelo de entidades y asociaciones.

El primer modelo (véase Ilustración 25) está destinado a cubrir las necesidades de la aplicación de alta. Debe recoger la estructura completa necesaria para guardar todos los datos de un aviso, es decir una entidad para el aviso y como este está compuesto por tomas y actividades otras dos entidades para cada uno de estos elementos. Para completar el modelo nos hacen falta además otras dos entidades una para las muestras asociadas a las tomas y otra para los servicios asociados a las actividades.

Además de las entidades anteriores son necesarias entidades complementarias para conseguir que el procedimiento de alta no sea completamente manual. Un ejemplo es la necesidad de recuperar los operadores de un centro a la hora de asignar uno a una toma o actividad. Para poder recuperarlo desde la aplicación en el modelo incluimos una entidad denominada “Ope”.

Además de la entidad de operadores incluimos en el modelo también entidades para recuperar materiales y servicios de expediente (Servlist, Matlist) y materiales y servicios del sistema (Mat, Serv) junto con una entidad para recuperar actividades insitu existentes en el sistema (Aclist).

Ilustración 25:MC Alta

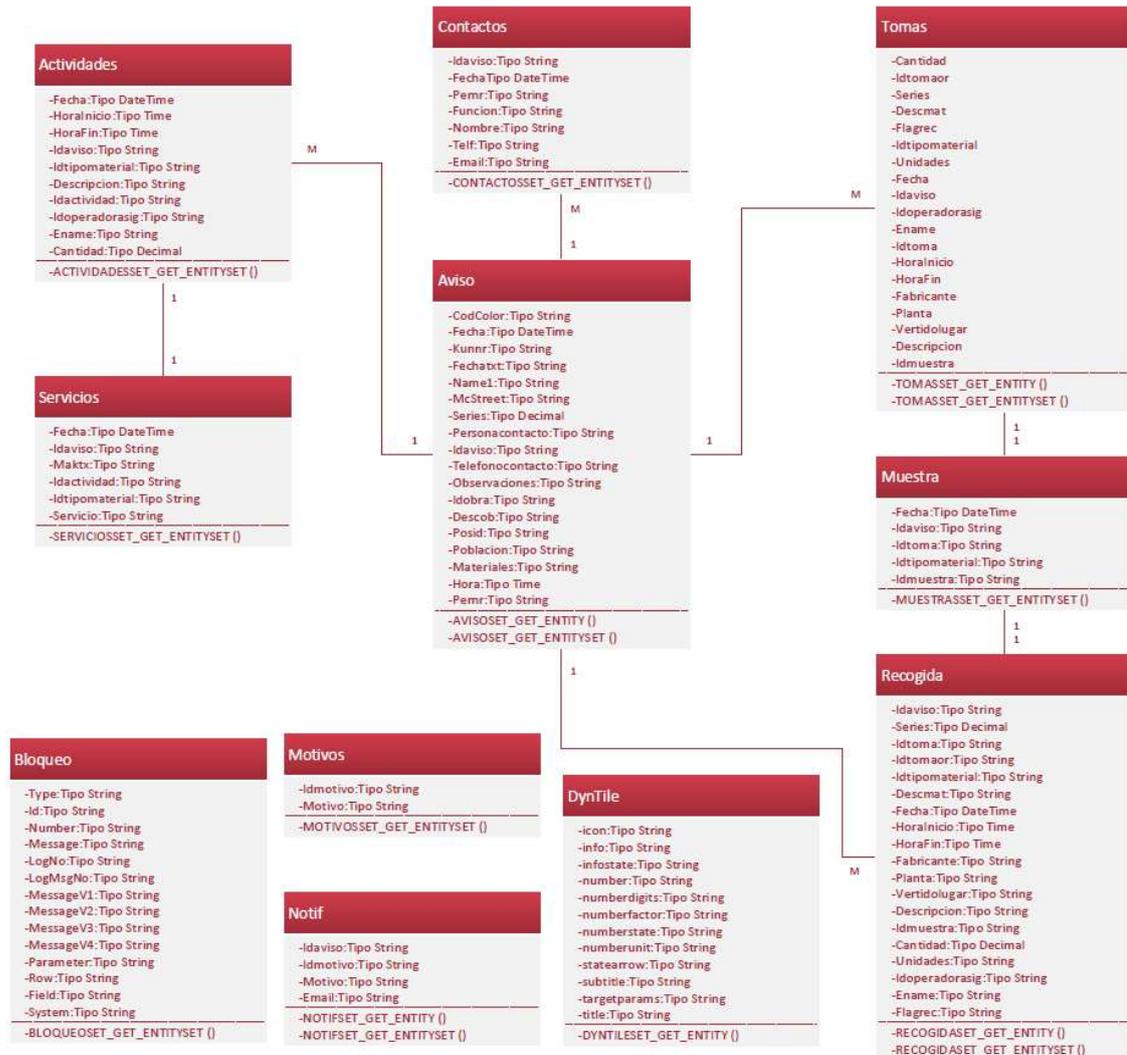


Para el modelo (véase Ilustración 26) del servicio destinado a cubrir las necesidades de la aplicación de visualización de avisos, nuevamente tenemos la estructura de entidades aviso, tomas, actividades, muestra y servicios con sus asociaciones, pero se añaden dos entidades asociadas a la entidad aviso. La primera se denomina “Contactos” que representan los interlocutores del expediente al que está asociado el aviso. La segunda es una entidad denominada “Recogida” la cual representa una actividad que se realiza de manera automática luego de registrar un tipo de material específico en una toma. Por ejemplo, hay algunos materiales por los que hay que desplazarse en dos ocasiones a obra, la primera cuando el cliente solicita el trabajo y ese material debe permanecer en obra 24 horas entonces el sistema creará un aviso de recogida a la hora de crear la toma.

Respecto a las entidades independientes las denominadas “Notif” y “Motivos” cumplirán la función de notificación especificada en los requisitos. La entidad

“DynTile” sirve para devolver los datos dinámicos del dyntile de la aplicación. Por último, la entidad “Bloqueo” es necesaria para recuperar validaciones con el cliente al ver un aviso y enviar mensajes desde el backend al frontend.

Ilustración 26:MC Visualización



Por último, el modelo de Agenda (véase Ilustración 27) es el más sencillo de todos, pues se vuelve a introducir las entidades típicas de aviso, tomas y actividades ya que esta aplicación también permite la edición de las tareas. Las tareas son quizá la entidad más relevante en este modelo ya que esta representa a los ítems que se visualizarán en el calendario y con los que se interactuará desde la aplicación de planificación.

Ilustración 27:MC Agenda

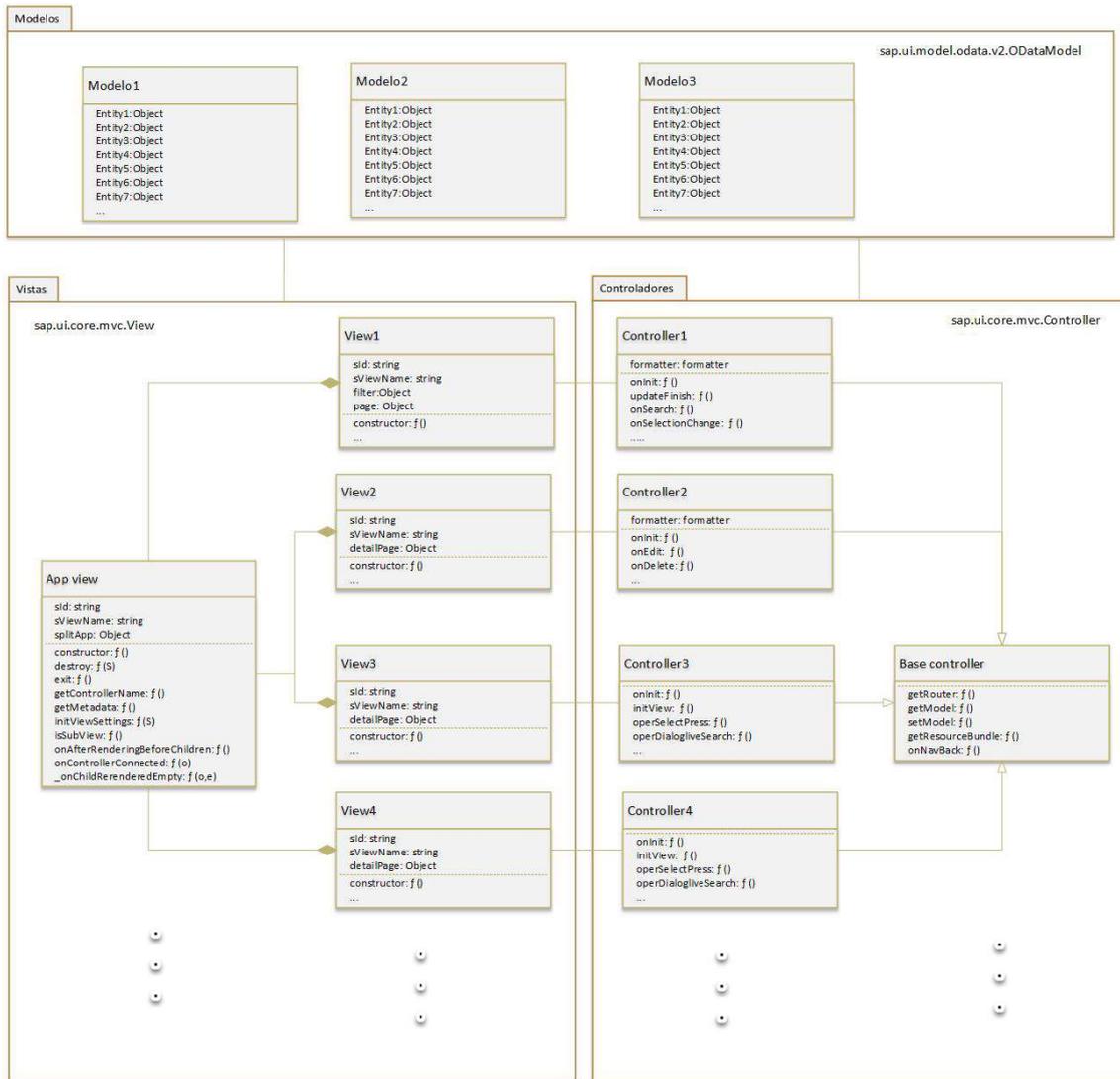


### 3.2.3 Modelado aplicaciones SAPUI5

En general las aplicaciones SAPUI5 implementan un modelo común (véase Ilustración 28) ya que la implementación de una se resume en la creación del servicio y las vistas y controladores que lo consumirán. En nuestro modelo las clases destinadas a representar los modelos vendrán definidas por las implementaciones de los servicios Odata que consume nuestra aplicación. Todas las clases del modelo son del tipo base `sap.ui.model.odata.v2.ODataModel`.

En cuanto a las vistas, una aplicación SAPUI5 siempre implementa una clase `App.vista.view` la cual es la raíz para la agregación y navegación de todas las otras vistas que definamos en nuestra aplicación. Cada vista de una aplicación SAPUI5 está asociada a un controlador del tipo base `sap.ui.core.mvc.Controller`.

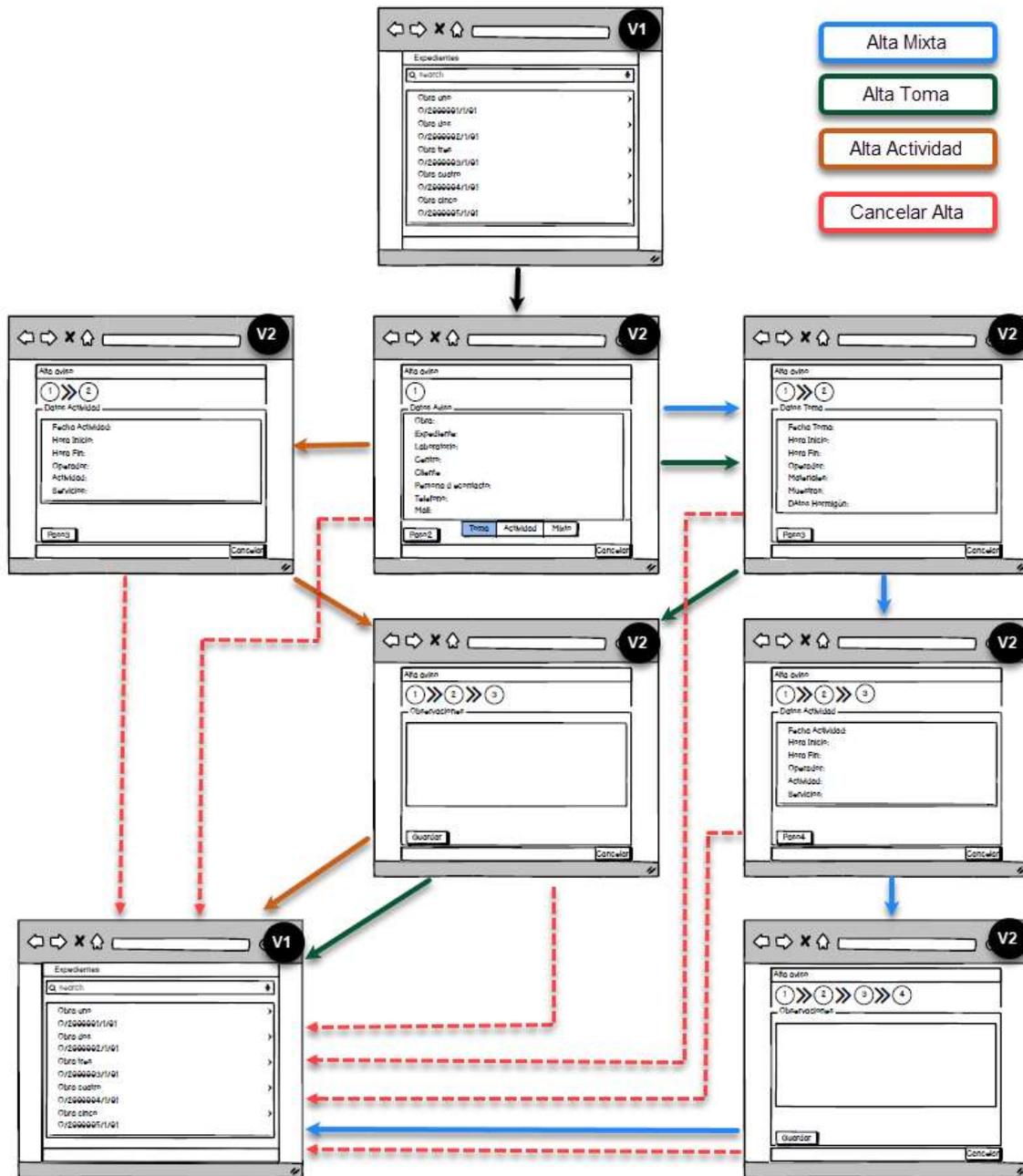
Ilustración 28: MC Genérico aplicación SAPUI5



### 3.2.3.1 Wireframes

Para diseñar la interfaz gráfica y la navegación de las aplicaciones SAPUI5 a desarrollar usaremos wireframes como herramienta. Un wireframe o prototipo, un boceto donde se representa visualmente, de una forma muy sencilla y esquemática la estructura de una página o aplicación web junto al flujo de navegación de esta.

Ilustración 29:WF App Alta Aviso

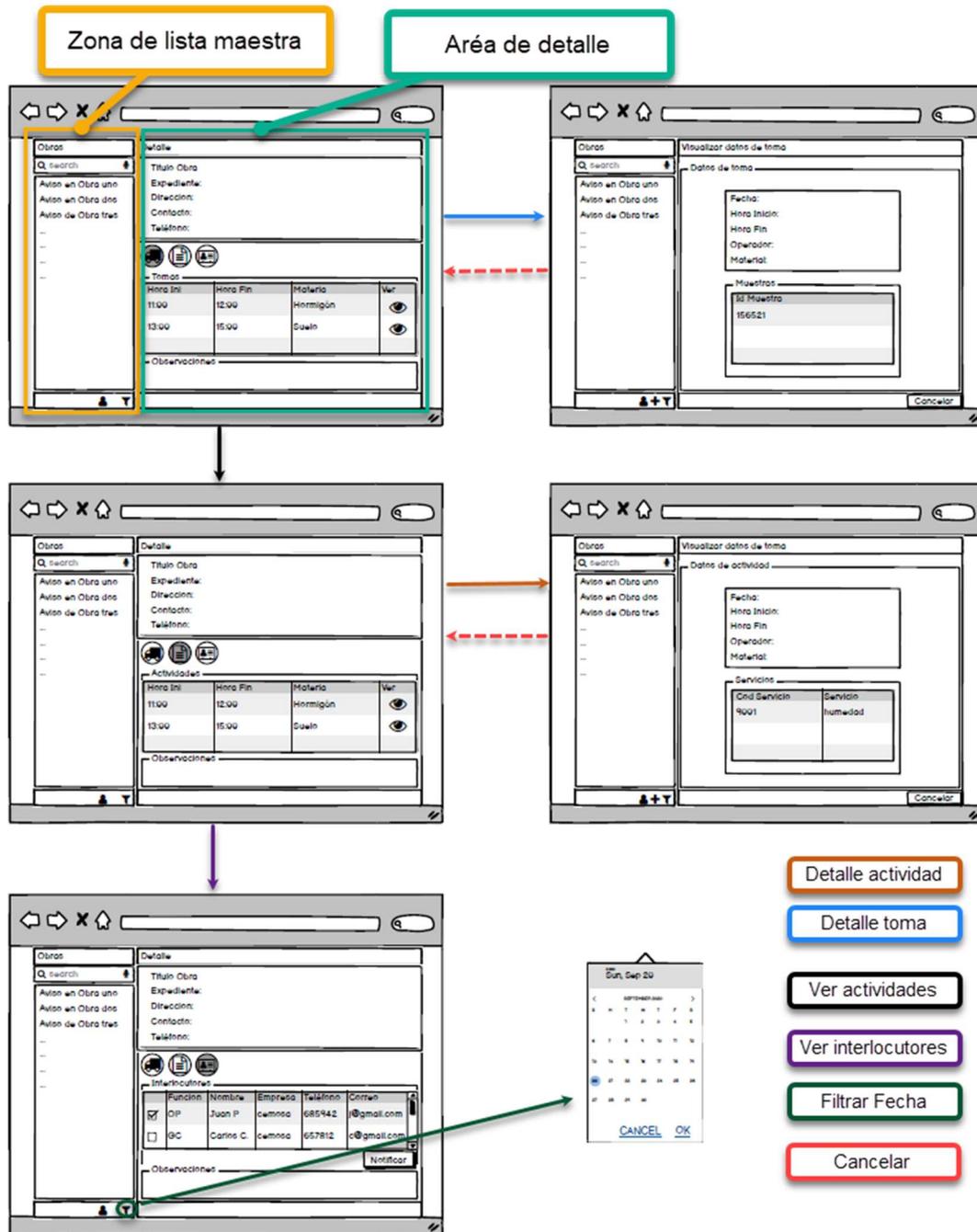


En el wireframe para la aplicación de alta de avisos (véase Ilustración 29) en el que se puede ver el flujo de navegación que deberá implementar la aplicación. Incluye tres flujos para los distintos tipos de registro que se debe poder realizar por toma, actividad o mixta. Básicamente se trata de la navegación entre dos vistas v1 y v2.

A nivel estructural la v1 está compuesta por una barra de búsqueda en la parte superior y en el cuerpo de la página un listado de expedientes que permite a nivel de registro navegación a la v2. Por su parte la v2 es una página con un componente wizard que dependiendo del tipo de aviso que se esté dando de alta

va cargando el contenido del formulario a medida que el usuario va introduciendo los datos del aviso.

Ilustración 30: WF App Avisos

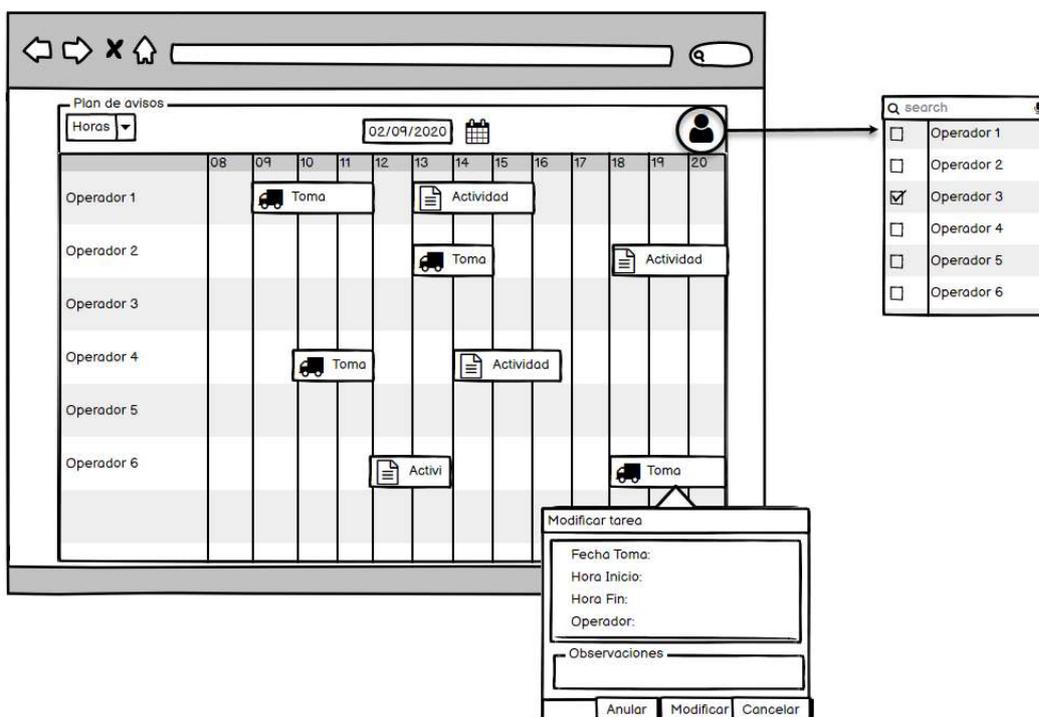


En el wireframe que especifica cómo será la interfaz de la aplicación de visualización de avisos (véase Ilustración 30) tiene una particularidad, se implementará como un Split-Screen Layout lo que significa que tendremos disponibles dos secciones en la pantalla para cargar las vistas de la aplicación.

- **Zona de lista maestra:** La lista maestra muestra los elementos disponibles para el usuario en nuestro caso un listado de avisos.
- **Área de detalles:** Esta área de contenido muestra los detalles de uno o varios elementos seleccionados en la lista maestra.

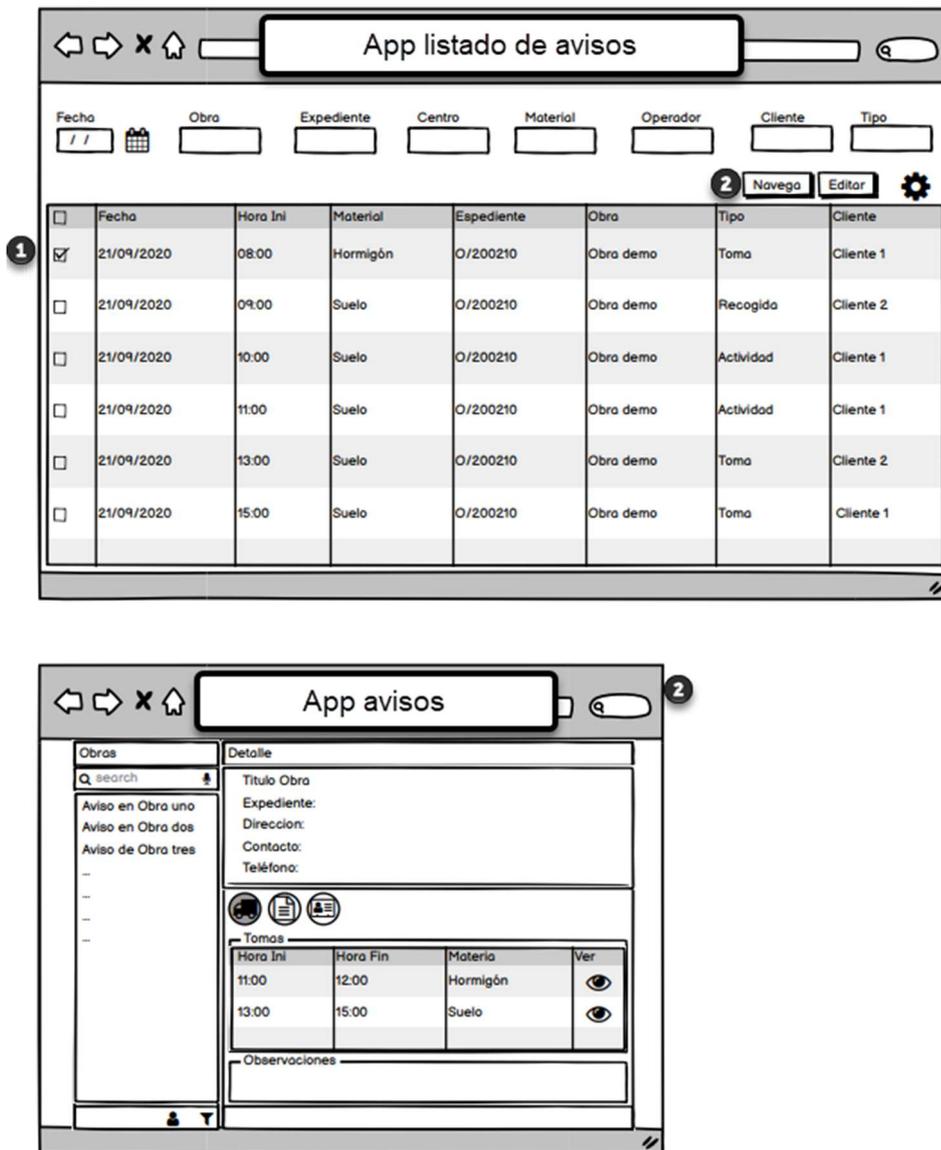
El flujo de navegación básico se resume en que se navegara al detalle del aviso cuando se seleccione uno en concreto. Una vez que la vista de detalle de aviso se visualice en el área de detalle desde esta podemos navegar al resto de vistas definidas dependiendo de lo que deseamos visualizar ya sea el detalle de una toma, actividad o los interlocutores.

Ilustración 31:Wf App Plan Avisos



El esquema que representa cómo será la aplicación de agenda (véase Ilustración 31) no contiene navegaciones ya que la aplicación está compuesta por una única vista donde se pueden ver las tareas de cada operador en un rango temporal dando lugar a especie de agenda.

Ilustración 32:WF App listado de avisos

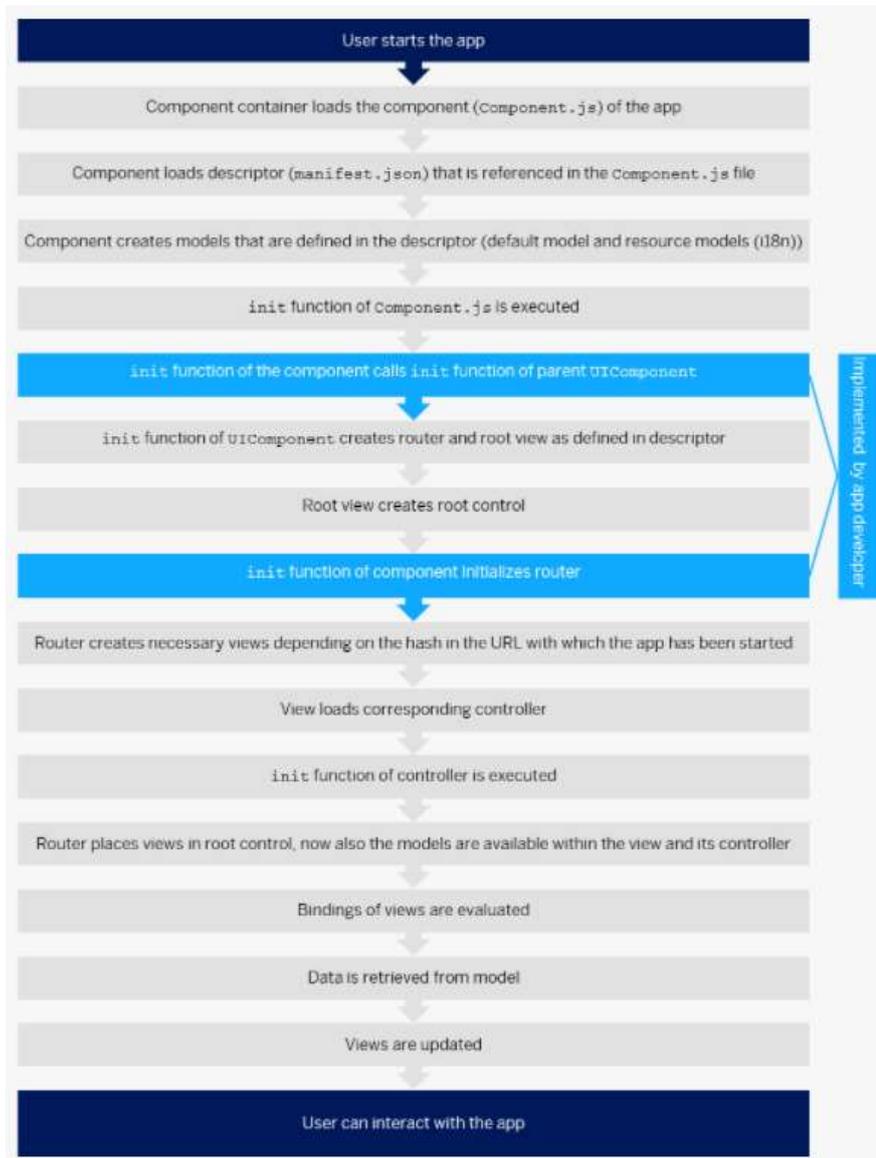


La aplicación de listado de avisos seguirá un diseño típico de List Report (véase Ilustración 32), es decir una única vista donde se muestre listado con las tomas y actividades. En la además de la representación de esta vista se ha querido mostrar el flujo de navegación externa, es decir la navegación de una aplicación a otra.

### 3.2.3.2 Diagrama de secuencia.

La interacción de los componentes en la ejecución de una aplicación SAPUI5 se puede generalizar en el esquema para las operaciones CRUD. Antes de iniciar con la interacción propiamente dicha entre el usuario y la aplicación, se producen una secuencia de pasos para iniciar la aplicación (véase Ilustración 33).

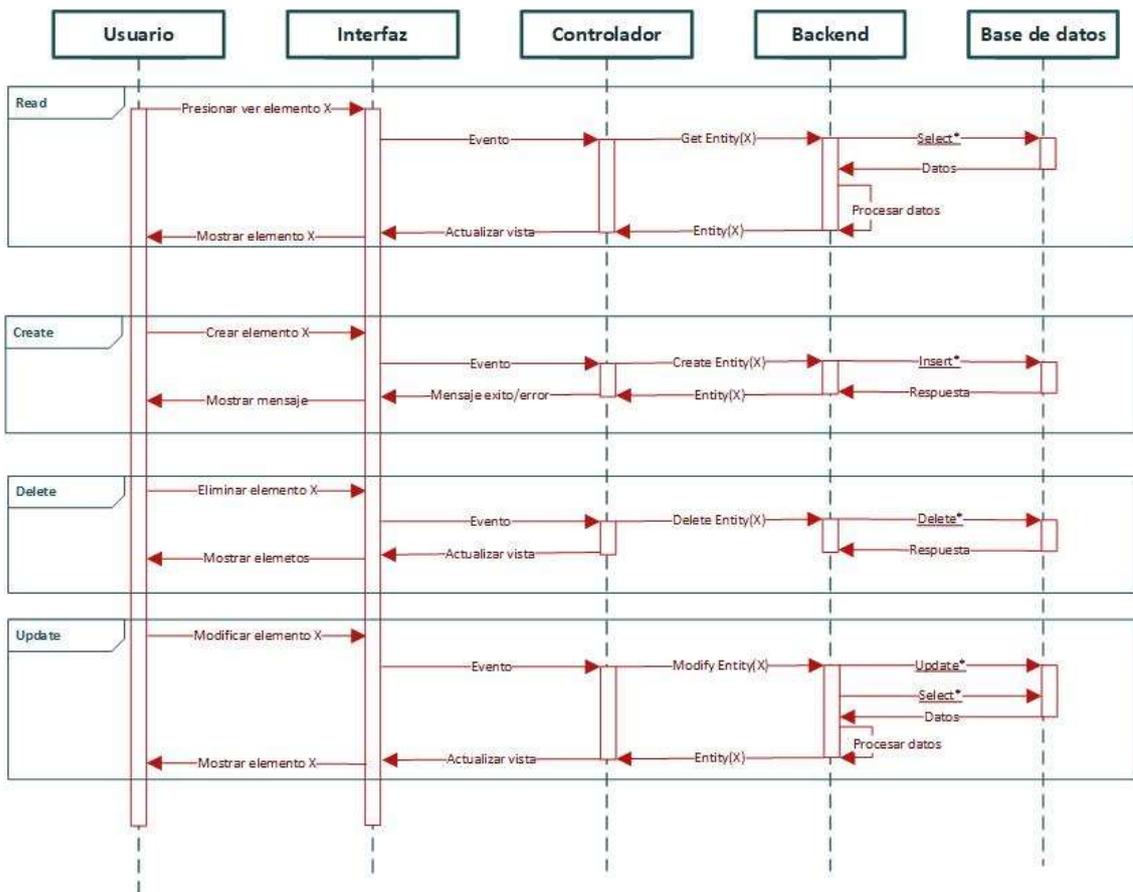
Ilustración 33: Proceso de inicio de una aplicación SAPUI5



Fuente: <https://sapui5.hana.ondemand.com/>

Una vez la aplicación se ha iniciado la mayoría de los procedimientos que se implementan sobre todo para las aplicaciones de tipo productivo. Seguirán una secuencia de interacciones común ya que los métodos a implementar son los de las operaciones CRUD (véase Ilustración 34).

Ilustración 34: Diagrama de secuencia general operaciones CRUD



### 3.3 Implementación

#### 3.3.1 Base de datos

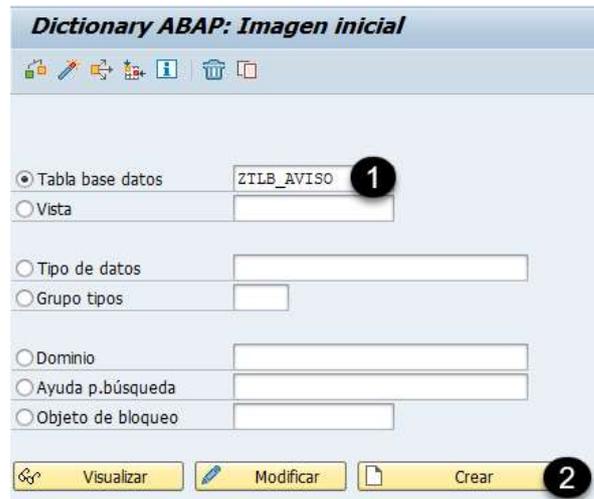
Esta tarea abarca la creación de todos los elementos del diccionario de datos necesarios para el desarrollo de este sistema, por ejemplo: dominios, elementos de datos, estructuras, tipos de tabla, tablas transparentes, vistas, etc.

##### 3.3.1.1 Tablas transparentes

En este punto se crean las tablas definidas en el modelo E-R en el punto 2.3.1. El procedimiento para crear una tabla transparente en nuestro entorno usando la herramienta del diccionario de datos integrada en SAP-ERP es el siguiente:

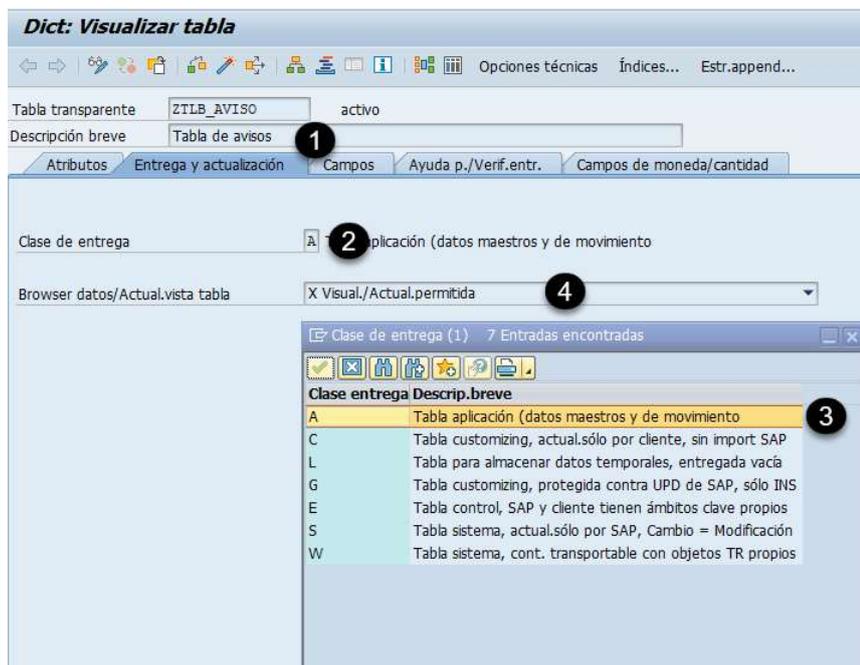
1. Accedemos a la transacción del diccionario de datos SE11 e informamos el nombre de la tabla base datos que vamos a crear.

Ilustración 35:SE11 Crear tabla



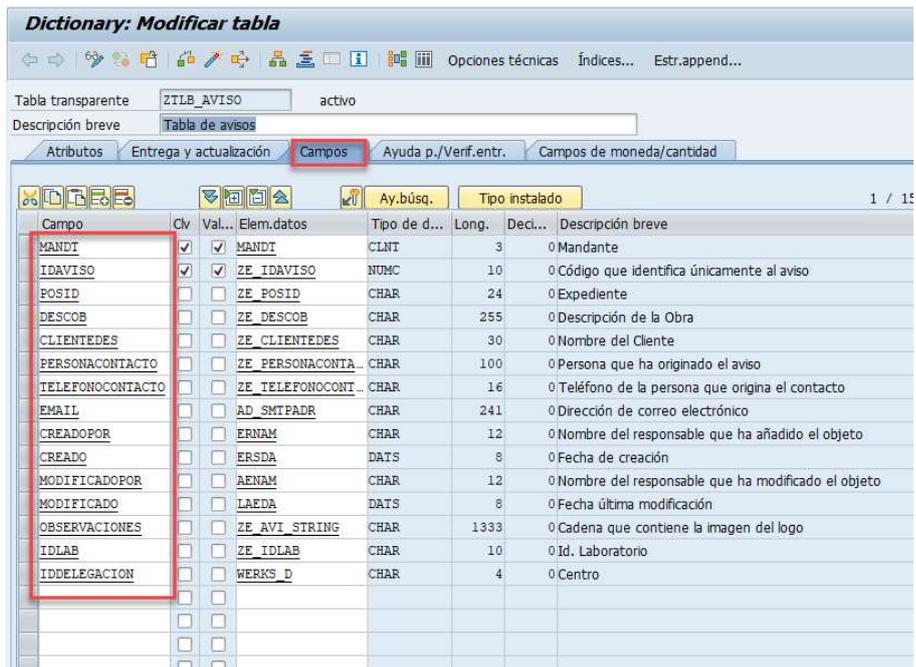
2. Debemos rellenar el campo descripción ya que este es obligatorio. En la pestaña Entrega y actualización seleccionamos la Clase de entrega A y el tipo de actualización Visualización/Actualización permitida.

Ilustración 36:Conf. Parámetros de tabla



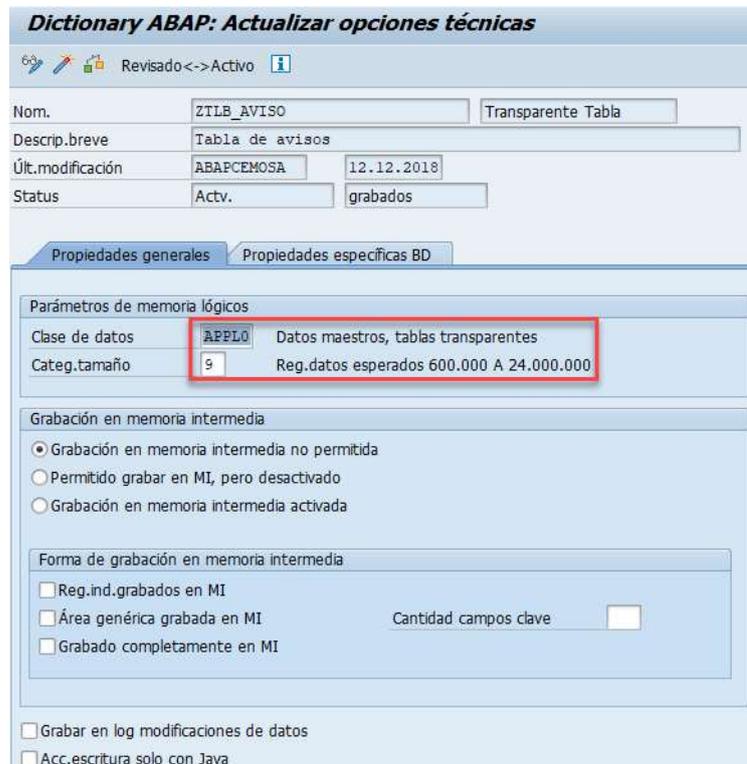
3. En la pestaña Campos añadimos los campos de la tabla.

Ilustración 37:Campos de tabla



4. Configuramos las opciones técnicas de la tabla.

Ilustración 38:Opciones técnicas



5. Configuramos las categorías de ampliación que las podemos encontrar en el menú en Detalles.

Ilustración 39: Conf. Categoría de ampliación

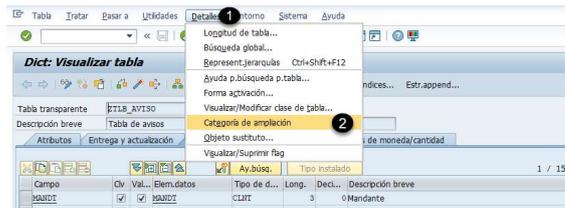
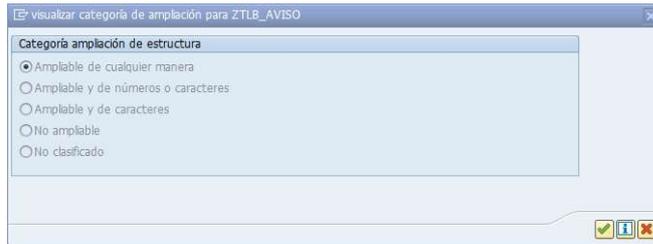
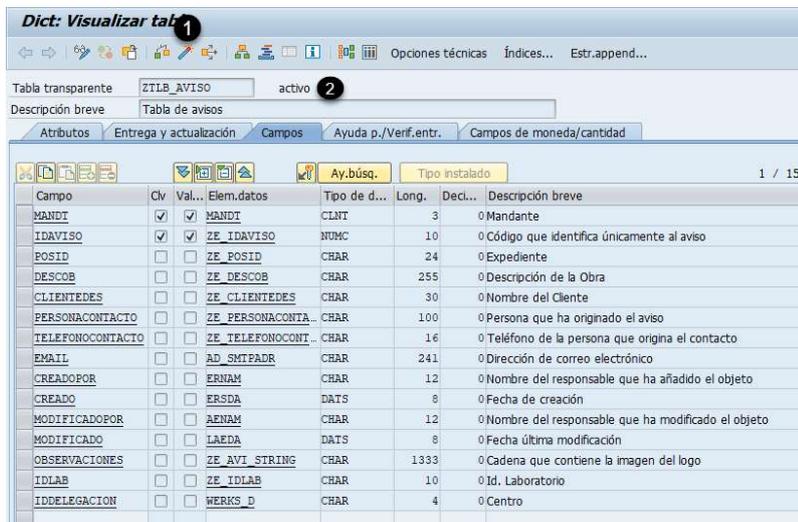


Ilustración 40: Categorías de ampliación



6. Activamos la tabla utilizando la opción de la barra de herramientas.

Ilustración 41: Activar tabla



El resultado es el siguiente:

Ilustración 42:Tabla ZTLB\_AVISO

**Dict: Visualizar tabla**

Tabla transparente: ZTLB\_AVISO activo

Descripción breve: Tabla de avisos

Atributos Entrega y actualización Campos Ayuda p./Verif.entr. Campos de moneda/cantidad

Campo	Cv	Val...	Elem.datos	Tipo de d...	Long.	Decl...	Descripción breve
MANDI	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDI	CLNT	3		0 Mandante
IDAVISO	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	ZE_IDAVISO	NUMC	10		0 Código que identifica únicamente al aviso
POSID	<input type="checkbox"/>	<input type="checkbox"/>	ZE_POSID	CHAR	24		0 Expediente
DESCOB	<input type="checkbox"/>	<input type="checkbox"/>	ZE_DESCOB	CHAR	255		0 Descripción de la Obra
CLIENTEDES	<input type="checkbox"/>	<input type="checkbox"/>	ZE_CLIENTEDES	CHAR	30		0 Nombre del Cliente
PERSONACONTACTO	<input type="checkbox"/>	<input type="checkbox"/>	ZE_PERSONACONIA	CHAR	100		0 Persona que ha originado el aviso
TELEFONOCONTACTO	<input type="checkbox"/>	<input type="checkbox"/>	ZE_TELEFONOCONTI	CHAR	16		0 Teléfono de la persona que origina el contacto
EMAIL	<input type="checkbox"/>	<input type="checkbox"/>	AD_SMTFADR	CHAR	241		0 Dirección de correo electrónico
CREADOPOR	<input type="checkbox"/>	<input type="checkbox"/>	ERNAM	CHAR	12		0 Nombre del responsable que ha añadido el objeto
CREADO	<input type="checkbox"/>	<input type="checkbox"/>	ERSDA	DATS	8		0 Fecha de creación
MODIFICADOPOR	<input type="checkbox"/>	<input type="checkbox"/>	AENAM	CHAR	12		0 Nombre del responsable que ha modificado el objeto
MODIFICADO	<input type="checkbox"/>	<input type="checkbox"/>	LAEDA	DATS	8		0 Fecha última modificación
OBSERVACIONES	<input type="checkbox"/>	<input type="checkbox"/>	ZE_AVI_STRING	CHAR	1333		0 Cadena que contiene la imagen del logo
IDLAB	<input type="checkbox"/>	<input type="checkbox"/>	ZE_IDLAB	CHAR	10		0 Id. Laboratorio
IDDELEGACION	<input type="checkbox"/>	<input type="checkbox"/>	WERKS_D	CHAR	4		0 Centro

Para crear las tablas de tomas y actividades se sigue el procedimiento de creación y configuración descrito anteriormente.

Ilustración 43:Tabla ZTLB\_AVITOMAS

**Dict: Visualizar tabla**

Tabla transparente: ZTLB\_AVITOMAS activo

Descripción breve: Tabla de tomas de avisos

Atributos Entrega y actualización Campos Ayuda p./Verif.entr. Campos de moneda/cantidad

Campo	Cv	Val...	Elem.datos	Tipo de d...	Long.	Decl...	Descripción breve
MANDI	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDI	CLNT	3		0 Mandante
IDAVISO	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	ZE_IDAVISO	NUMC	10		0 Código que identifica únicamente al aviso
IDTOMA	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	ZE_IDTOMA	NUMC	3		0 Identificador toma
IDTIPOMATERIAL	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	ZE_TIPO	NUMC	3		0 Tipos de material
IDTOMAGR	<input type="checkbox"/>	<input type="checkbox"/>	ZE_IDTOMA	NUMC	3		0 Identificador toma
FECHA	<input type="checkbox"/>	<input type="checkbox"/>	DATUM	DATS	8		0 Fecha
HORA_INI	<input type="checkbox"/>	<input type="checkbox"/>	ZE_HORA	TIMS	6		0 Hora
HORA_FIN	<input type="checkbox"/>	<input type="checkbox"/>	ZE_HORA	TIMS	6		0 Hora
PLANTA	<input type="checkbox"/>	<input type="checkbox"/>	ZE_PLANTA	CHAR	35		0 Planta
FABRICANTE	<input type="checkbox"/>	<input type="checkbox"/>	ZE_FABRICANTE	CHAR	35		0 Descripción fabricante.
VERTIDOLUGAR	<input type="checkbox"/>	<input type="checkbox"/>	ZE_VERTIDOLUGAR	CHAR	255		0 Descripción del lugar donde se produce el vertido
SERIES	<input type="checkbox"/>	<input type="checkbox"/>	ZE_SERIES	DEC	7	2	2 Numero de series
CANTIDAD	<input type="checkbox"/>	<input type="checkbox"/>	ANZHL	DEC	7		2 Cantidad
UNIDADES	<input type="checkbox"/>	<input type="checkbox"/>	MEINS	UNIT	3		0 Unidad de medida base
ESTADO	<input type="checkbox"/>	<input type="checkbox"/>	ZE_ESTADOAVISO	CHAR	10		0 Estado de la línea de aviso
MOTIVO	<input type="checkbox"/>	<input type="checkbox"/>	ZE_MOTIVO	CHAR	255		0 Motivo
IDOPERADORASIG	<input type="checkbox"/>	<input type="checkbox"/>	ZE_IDOPERADORAS	NUMC	8		0 Operador Asignado para la toma de la muestra
MODIFICADOPOR	<input type="checkbox"/>	<input type="checkbox"/>	AENAM	CHAR	12		0 Nombre del responsable que ha modificado el objeto
MODIFICADO	<input type="checkbox"/>	<input type="checkbox"/>	LAEDA	DATS	8		0 Fecha última modificación
FLAGREC	<input type="checkbox"/>	<input type="checkbox"/>	CHAR1	CHAR	1		0 Indicador de una posición

Ilustración 44:Tabla ZTLB\_AVIACI

Campo	Clv	Val...	Elem.datos	Tipo de d...	Long.	Deci...	Descripción breve
MANDI	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDI	CLNT	3	0	Mandante
IDAVISO	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	ZE_IDAVISO	NUMC	10	0	Código que identifica únicamente al aviso
IDACTIVIDAD	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	ZE_IDACTIVIDAD	NUMC	3	0	Elemento para id de actividad
FECHA	<input type="checkbox"/>	<input type="checkbox"/>	DATUM	DATS	8	0	Fecha
HORA_INI	<input type="checkbox"/>	<input type="checkbox"/>	ZE_HORA	TIMS	6	0	Hora
HORA_FIN	<input type="checkbox"/>	<input type="checkbox"/>	ZE_HORA	TIMS	6	0	Hora
IDTIPOMATERIAL	<input type="checkbox"/>	<input type="checkbox"/>	ZE_TIPO	NUMC	3	0	Tipos de material
SERVICIO	<input type="checkbox"/>	<input type="checkbox"/>	ZE_IDTIPOENSAYO	CHAR	18	0	Id .Tipo ensayo
CANTIDAD	<input type="checkbox"/>	<input type="checkbox"/>	ANZHL	DEC	7	2	Cantidad
MODIFICADOPOR	<input type="checkbox"/>	<input type="checkbox"/>	AENAM	CHAR	12	0	Nombre del responsable que ha modificado el objeto
MODIFICADO	<input type="checkbox"/>	<input type="checkbox"/>	LAEDA	DATS	8	0	Fecha última modificación
IDOPERADORASIG	<input type="checkbox"/>	<input type="checkbox"/>	ZE_IDOPERADORAS...	NUMC	8	0	Operador Asignado para la toma de la muestra
ESTADO	<input type="checkbox"/>	<input type="checkbox"/>	ZE_ESTADOAVISO	CHAR	10	0	Estado de la línea de aviso
MOTIVO	<input type="checkbox"/>	<input type="checkbox"/>	ZE_MOTIVO	CHAR	255	0	Motivo

### 3.3.2 Backend - Implementación de servicios Odata

Al contar el sistema con varias aplicaciones es necesario definir más de un servicio ya que el modelado de cara a la diversidad de elementos que se usaran en el frontend lo exige, además de seguir la recomendación de desarrollo de aplicaciones sapui5 1-1-3.

El proceso de implementación de servicios OData tiene tres fases diferenciadas:

- **Fase 1: Definición del modelo de datos**  
En esta fase, se define el modelo (es decir, define los tipos de entidad, conjuntos de entidades y asociaciones que utilizará su servicio).
- **Fase 2: Implementación del servicio**  
En esta fase, implementa las operaciones soportadas por el servicio. Si eres usando el RFC / BOR Generator, luego la implementación se lleva a cabo mapeando el modelo OData a los métodos del módulo de función RFC un objeto BOR.
- **Fase 3: Mantenimiento del servicio**  
En esta última fase se publica el servicio. Esta suele ser una actividad única (es decir, no tiene que volver a visitar este paso, incluso si actualiza el modelo o la implementación)

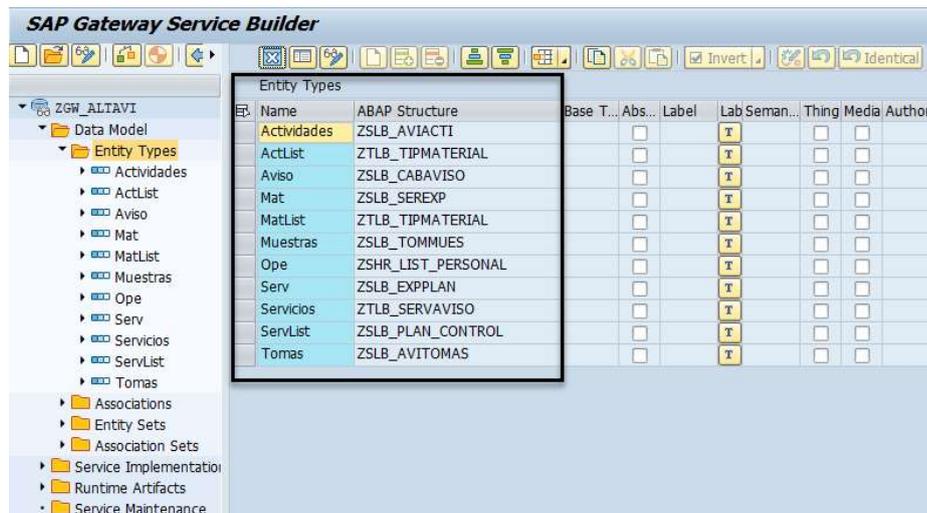
#### 3.3.2.1 Implementación de servicios Odata

En este apartado vamos a describir con detalle los pasos a seguir en cada fase del proceso de implantación para el servicio ZGW\_ALTAVI que nos servirá para generalizar este procedimiento ya que se realiza de la misma manera para el resto de los servicios Odata que se implementaran en este TFG.

## Definición del modelo

Basándonos en modelo de clases definido en el apartado 3.2.1. Empezamos definiendo las entidades de nuestro modelo de datos y asociándole una estructura del diccionario de datos, si bien se nos permite crear las entidades sin asociarles ninguna estructura y definir sus propiedades en el propio servicio vamos a trabajar con estructuras porque aportan bastantes ventajas al desarrollo tanto en la mantenibilidad y escalabilidad del modelo y de las operaciones de servicio.

Ilustración 45: Mapeo E-E ZGW\_ALTAVI



El siguiente paso es definir las propiedades de cada entidad, al estar usando estructuras contamos con la ventaja de que al definir las bastara con seleccionar los elementos de la estructura y añadirlos como propiedades a nuestras entidades. Podemos seleccionar todos los elementos de la estructura o algunas de ellas, una vez añadidos a la entidad se habrán definido componentes como tipo de datos, longitud, escala, label, etc. todo ello asociado al elemento de datos del campo de la estructura que seleccionamos como propiedad.

Es necesario definir, además de las características que se heredan de la estructura, las propiedades que serán clave dentro de la entidad y otras características que definirán los metadatos de servicio como, por ejemplo: si la propiedad puede ser nula, si se puede modificar la propiedad con peticiones del frontend, si se puede filtrar por la propiedad, etc.

Estas características son interesantes sobre todo de cara al fast develop en el frontend ya que unos metadatos enriquecidos más las anotaciones hacen que los desarrollos se terminen en menos tiempo.

Ilustración 46:Propiedades entity aviso

Name	Key	Edm Core Type	Prec.	Scale	Max...	Unit	Property Name	Crea...	Upd...	Sort...	Null...	Flit.	Label
Descob	<input type="checkbox"/>	Edm.String	0	0	255			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Obra
Email	<input type="checkbox"/>	Edm.String	0	0	241			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Dir.cor.elec.
Idlab	<input type="checkbox"/>	Edm.String	0	0	10			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Id. Laboratorio
Iddelegacion	<input type="checkbox"/>	Edm.String	0	0	4			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Centro
Observaciones	<input type="checkbox"/>	Edm.String	0	0	0			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Observaciones
Idaviso	<input checked="" type="checkbox"/>	Edm.String	0	0	10			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ID de Aviso
Posid	<input type="checkbox"/>	Edm.String	0	0	24			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Expediente
Cientedes	<input type="checkbox"/>	Edm.String	0	0	30			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Nombre del Cliente
Personacontacto	<input type="checkbox"/>	Edm.String	0	0	100			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Persona de contacto
Telefonocontacto	<input type="checkbox"/>	Edm.String	0	0	12			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Teléfono de contacto

Para finalizar la implementación del modelo de datos es necesario definir las asociaciones entre las entidades que se especifican en el diseño de clases.

Ilustración 47:Asociaciones ZGW\_ALTAVI

Name	E	Principal Entity	Principal Entity Cardinality	Dependent Entity	Dependent Entity Cardinality
ActivdadToServiciosAss	<input type="checkbox"/>	Activdades	1	Servicios	N
AvisoToActivdadesAss	<input type="checkbox"/>	Aviso	1	Activdades	N
AvisoToTomasAss	<input type="checkbox"/>	Aviso	1	Tomas	N
TomasToMuestrasAss	<input type="checkbox"/>	Tomas	1	Muestras	N

Una vez tenemos definido el modelo podemos generar el servicio.

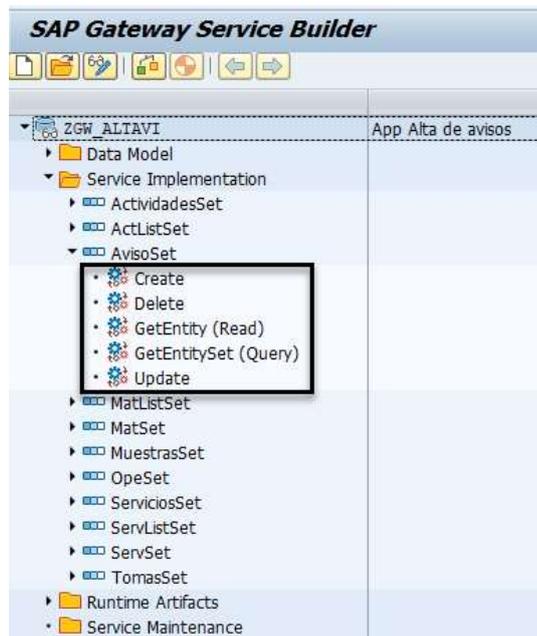
## Implementación del servicio

Luego de generar el servicio automáticamente se crea una estructura de clases para implementar los métodos definidos en el modelo de entidades *Figura 24* que son básicamente métodos asociados a operaciones CRUD. Específicamente para la implementación de dichos métodos se nos proporciona dos interfaces, que heredan de las clases principales proveedor de datos y proveedor de modelo, en ellas podemos realizar redefiniciones de los métodos necesarios para implementar nuestra lógica de negocio.

Ilustración 48:Artifacts ZGW\_ALTAVI

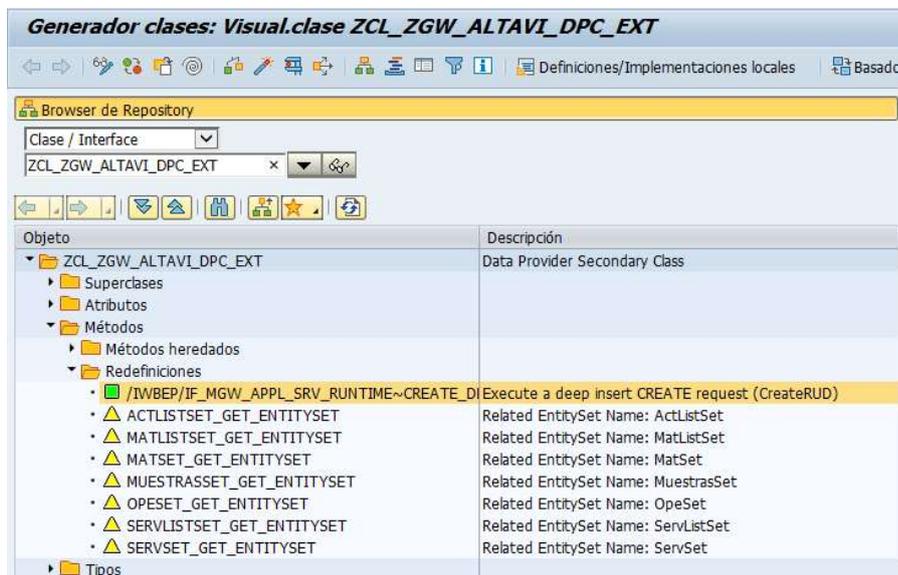
Name	Generated Artifact Type	ID programa	Tipo objeto	Nombre de objeto
ZCL_ZGW_ALTAVI_DPC	Data Provider Base Class	R3TR	CLAS	ZCL_ZGW_ALTAV...
ZCL_ZGW_ALTAVI_DPC_EXT	Data Provider Extension Class	R3TR	CLAS	ZCL_ZGW_ALTAV...
ZCL_ZGW_ALTAVI_MPC	Model Provider Base Class	R3TR	CLAS	ZCL_ZGW_ALTAV...
ZCL_ZGW_ALTAVI_MPC_EXT	Model Provider Extension Class	R3TR	CLAS	ZCL_ZGW_ALTAV...
ZGW_ALTAVI_MDL	Registered Model	R3TR	IWMO	ZGW_ALTAVI_MDL
ZGW_ALTAVI_SRV	Registered Service	R3TR	IWSV	ZGW_ALTAVI_SRV

Ilustración 49:Operaciones CRUD Avisoset



De acuerdo con los requerimientos no necesitamos implementar todas las operaciones para cada entidad, implementamos solo las necesarias. Esto se realiza redefiniendo los métodos generados por el servicio. en la clase ZCL\_ZGW\_ALTAVI\_DPC\_EXT.

Ilustración 50:Metodos redefinidos en ZGW\_ALTAVI



En la redefinición de métodos en este servicio en particular nos encontramos con una peculiaridad. Odata como sabemos es un protocolo que no soporta a priori la inserción profunda es así como la forma de hacer un update de una de ellas es usando proceso batch. Si bien podemos plantearnos esa solución no es óptima por lo que optamos en implementar la inserción de estructuras complejas

redefiniendo el método:

```
/IWBEF/IF_MGW_APPL_SRV_RUNTIME~CREATE_DEEP_ENTITY
```

De tal manera que cuando exista una petición PUT/POST de la entidad aviso se ejecute un método Z que hemos implementado para mapear el objeto JSON que nos devuelve el proveedor de datos a los campos correspondientes de las tablas de avisos, tomas y actividades.

```
METHOD /iwbef/if_mgw_appl_srv_runtime~create_deep_entity.
```

```
DATA: wl_entityset_name TYPE string,
      wa_avisoo         TYPE zslb_avisoo,
      wa_deep_avisoo   TYPE zslb_avisoo_e.

wl_entityset_name = io_tech_request_context->get_entity_set_name(
).

CASE wl_entityset_name.
  WHEN 'AvisoSet'.
    io_data_provider->read_entry_data( IMPORTING es_data =
wa_deep_avisoo ).

    CALL FUNCTION 'Z_LB_GUARDAR_AVISOO'
      EXPORTING
        pe_avisoo = wa_deep_avisoo.

    copy_data_to_ref( EXPORTING is_data = wa_deep_avisoo
                     CHANGING cr_data = er_deep_entity
).

ENDCASE.
ENDMETHOD.
```

El detalle de código de la implementación de cada método se encuentra en el Anexo "Código Fuente Back-end".

### Mantenimiento del servicio

Debido a nuestra arquitectura los procedimientos de esta fase se realizan en el servidor frontend. Desde la transacción IWFND/MAINT\_SERVICE debemos primero activar el nodo IFC y registrar un alias de sistema para la comunicación entre backend y frontend.



Ilustración 51: Mapeo E-E ZGW\_CRUAVISOS

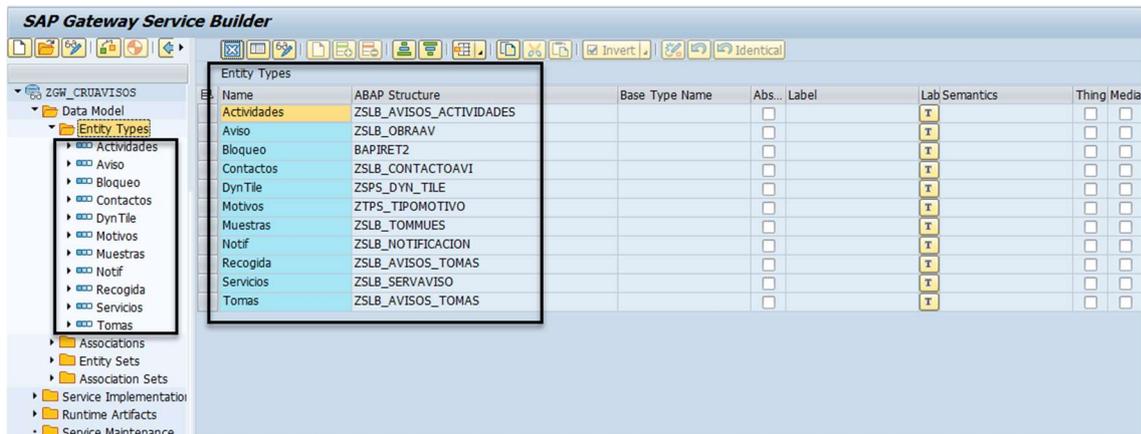
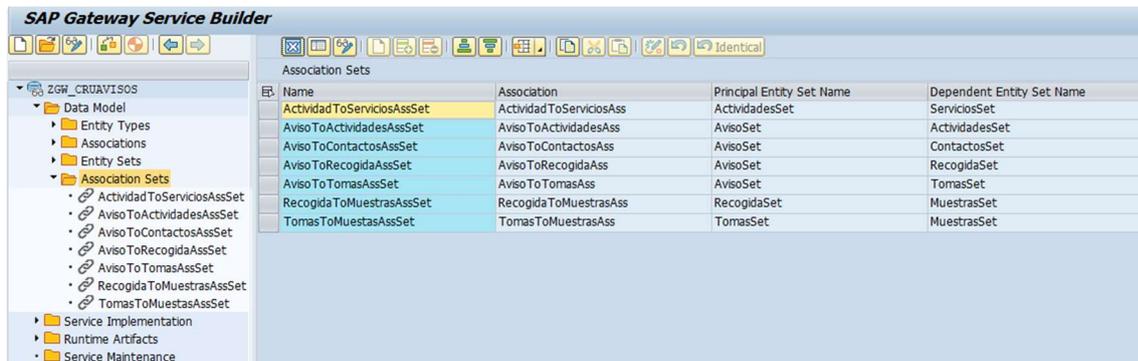


Ilustración 52: Asociaciones ZGW\_CRUAVISOS

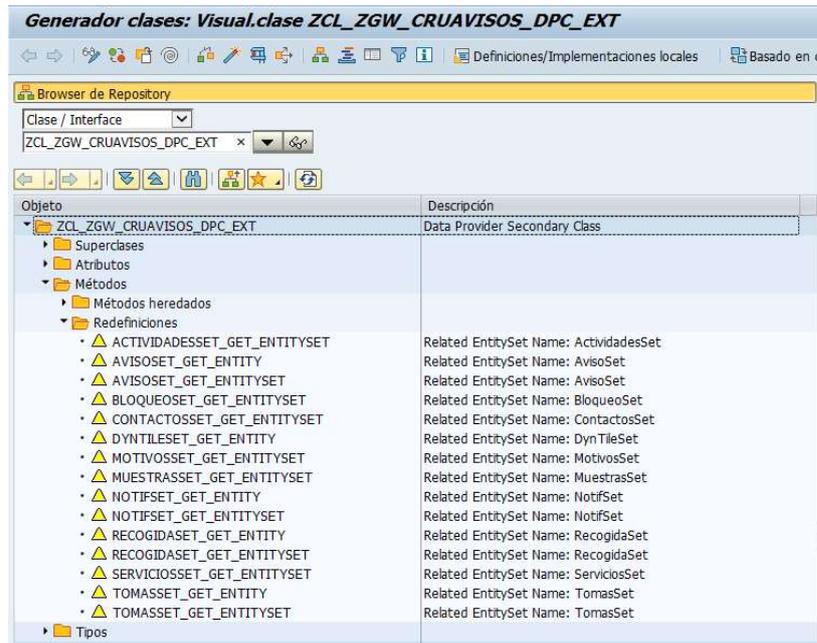


La configuración de la entidad principal del modelo es la siguiente

Ilustración 53: Propiedades entidad aviso

Name	Key	Edm Core Type	Prec.	Scale	Max...	Unit Property Name	Cre...	Upd...	Sort...	Null...	Flt.	Label	Lab
CodColor	<input type="checkbox"/>	Edm.String	0	0	1		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Carácter 1	<input type="checkbox"/>
Fecha	<input checked="" type="checkbox"/>	Edm.Date Time	7	0	0		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Fecha	<input type="checkbox"/>
Kunnr	<input type="checkbox"/>	Edm.String	0	0	10		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Cliente	<input type="checkbox"/>
Fechaabt	<input type="checkbox"/>	Edm.String	0	0	10		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Fecha	<input type="checkbox"/>
Name1	<input type="checkbox"/>	Edm.String	0	0	35		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Cliente	<input type="checkbox"/>
McStreet	<input type="checkbox"/>	Edm.String	0	0	25		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Calle	<input type="checkbox"/>
Series	<input type="checkbox"/>	Edm.Decimal	7	2	0		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Nº Series	<input type="checkbox"/>
Personacontacto	<input type="checkbox"/>	Edm.String	0	0	100		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Persona de contacto	<input type="checkbox"/>
Idaviso	<input checked="" type="checkbox"/>	Edm.String	0	0	10		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	ID de Aviso	<input type="checkbox"/>
Telefonocontacto	<input type="checkbox"/>	Edm.String	0	0	16		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Teléfono de contacto	<input type="checkbox"/>
Observaciones	<input type="checkbox"/>	Edm.String	0	0	0		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Observaciones	<input type="checkbox"/>
Idobra	<input type="checkbox"/>	Edm.String	0	0	6		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Código de Obra	<input type="checkbox"/>
Descob	<input type="checkbox"/>	Edm.String	0	0	255		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Obra	<input type="checkbox"/>
Posid	<input type="checkbox"/>	Edm.String	0	0	24		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Expediente	<input type="checkbox"/>
Poblacion	<input type="checkbox"/>	Edm.String	0	0	40		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Población	<input type="checkbox"/>
Materiales	<input type="checkbox"/>	Edm.String	0	0	255		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	char255	<input type="checkbox"/>
Hora	<input type="checkbox"/>	Edm.Time	0	0	0		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Hora	<input type="checkbox"/>
Pennr	<input checked="" type="checkbox"/>	Edm.String	0	0	8		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Operador Asignado	<input type="checkbox"/>

Ilustración 54:Metodos redefinidos en ZGW\_CRUAVISOS



En cuanto a las redefiniciones de métodos hay uno en particular en este servicio que merece la pena comentar si bien se ha implementado redefiniendo un método get y debería estar asociado a una operación READ la implementación es un procedimiento de envío de mail usando el MF

```
CALL FUNCTION 'SO_NEW_DOCUMENT_SEND_API1'
EXPORTING
  document_data          = wa_subject
  document_type          = 'HTM'
  put_in_outbox          = abap_true
  commit_work            = abap_true
TABLES
  object_content         = t1_msg
  receivers              = t1_mailrecipients
EXCEPTIONS
  too_many_receivers    = 1
  document_not_sent     = 2
  document_type_not_exist = 3
  operation_no_authorization = 4
  parameter_error       = 5
  x_error                = 6
  enqueue_error         = 7
  OTHERS                 = 8.
```

Esto es debido a que la interfaz que nos proporciona la clase solo contiene métodos de operaciones CRUD y la notificación no es una de ellas, la entidad notificación no existe como tal en el sistema es un objeto abstracto que hemos usado para poder enviar notificaciones vía email desde el frontend

## ZGW\_CALNENDARIOPLAN

En este apartado se describe la implementación del servicio para la aplicación de Plan Avisos, siguiendo el modelo se crean las siguientes entidades y asociaciones.

Ilustración 55: Mapeo E-E ZGW\_CALENDARIOPLAN

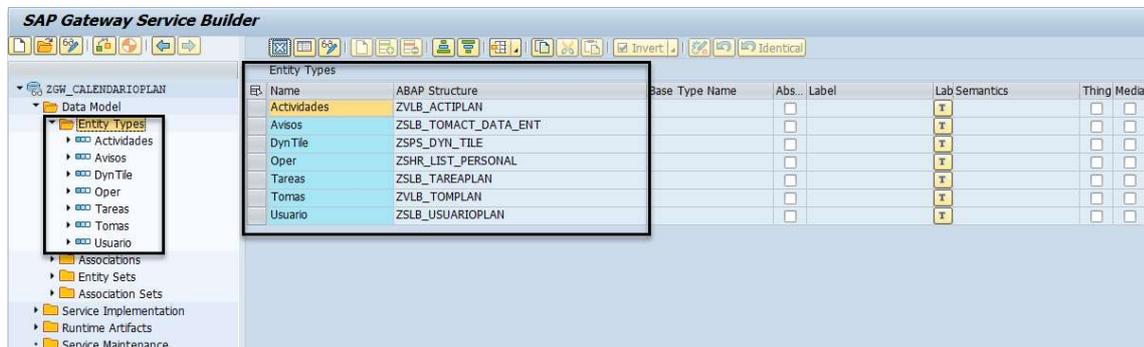
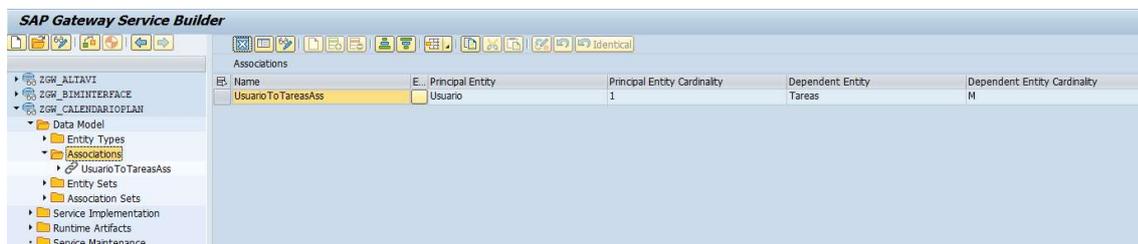
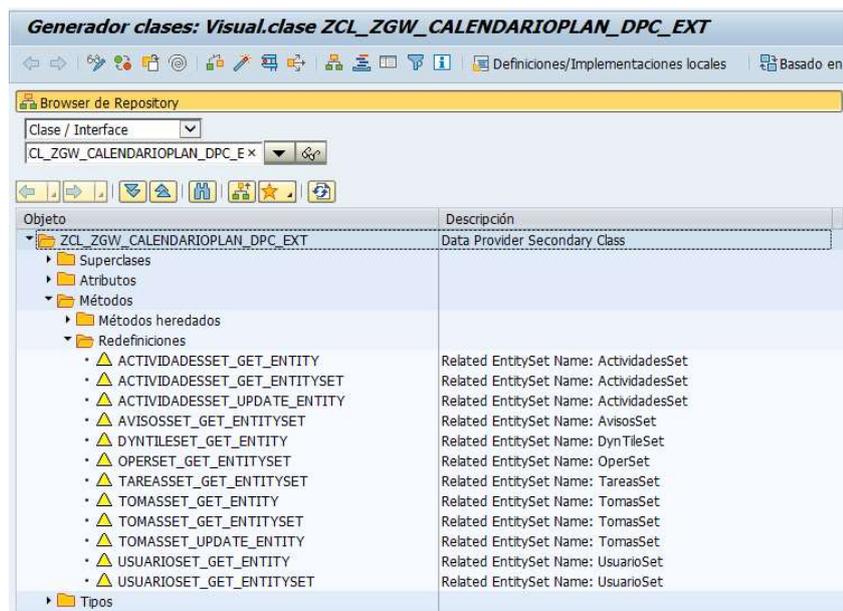


Ilustración 56: Asociaciones ZGW\_CALENDARIOPLAN



Los métodos que se han redefinido en este servicio son los siguientes.

Ilustración 57: Metodos redefinidos en ZGW\_CALENDARIOPLAN



Para la aplicación del listado de avisos no se ha definido un servicio específico se han reutilizado los servicios ZGW\_CALENDARIOPLAN

### 3.3.3 Frontend

El desarrollo de toda la parte de frontend se realiza usando el entorno que proporciona SAP, SAP WebIDE. Para implementar las aplicaciones SAPUI5 que conforman el sistema de gestión de avisos. Siguiendo la regla 1-1-3 se han creado cuatro proyectos distintos para cuatro aplicaciones distintas.

La regla 1-1-3 nos dice que debemos simplificar la funcionalidad de cada aplicación a un perfil de usuario, la aplicación debe implementar un caso de uso y tener 3 pantallas. De esta manera creamos una aplicación para el alta, otra para la edición y la visualización, una tercera para el listado y por último una para la gestión de la carga de trabajo.

#### 3.3.3.1 Implementación de una aplicación SAPUI5

Existe tres tipos de archivos básicos en una aplicación sapui5:

##### **Descriptor (manifest.json)**

Se recomienda usar el archivo para configurar la configuración de la aplicación y poner toda la información importante necesaria para ejecutar la aplicación allí. El uso de este enfoque significa que necesita escribir menos código de aplicación y ya puede tener acceso a la información antes de que se cree una instancia de la aplicación.manifest.json

##### **Vista raíz (App.view.xml)**

El archivo define la vista raíz de la aplicación. En la mayoría de los casos, contiene un control o un control como un control raíz App.view.xmlAppSplitApp

SAPUI5 admite varios tipos de vista (XML, HTML, JavaScript, JSON). Se recomienda utilizar vistas XML, ya que para estas debe separar la lógica del controlador de la definición de vista en un archivo de controlador (por ejemplo). App.controller.js

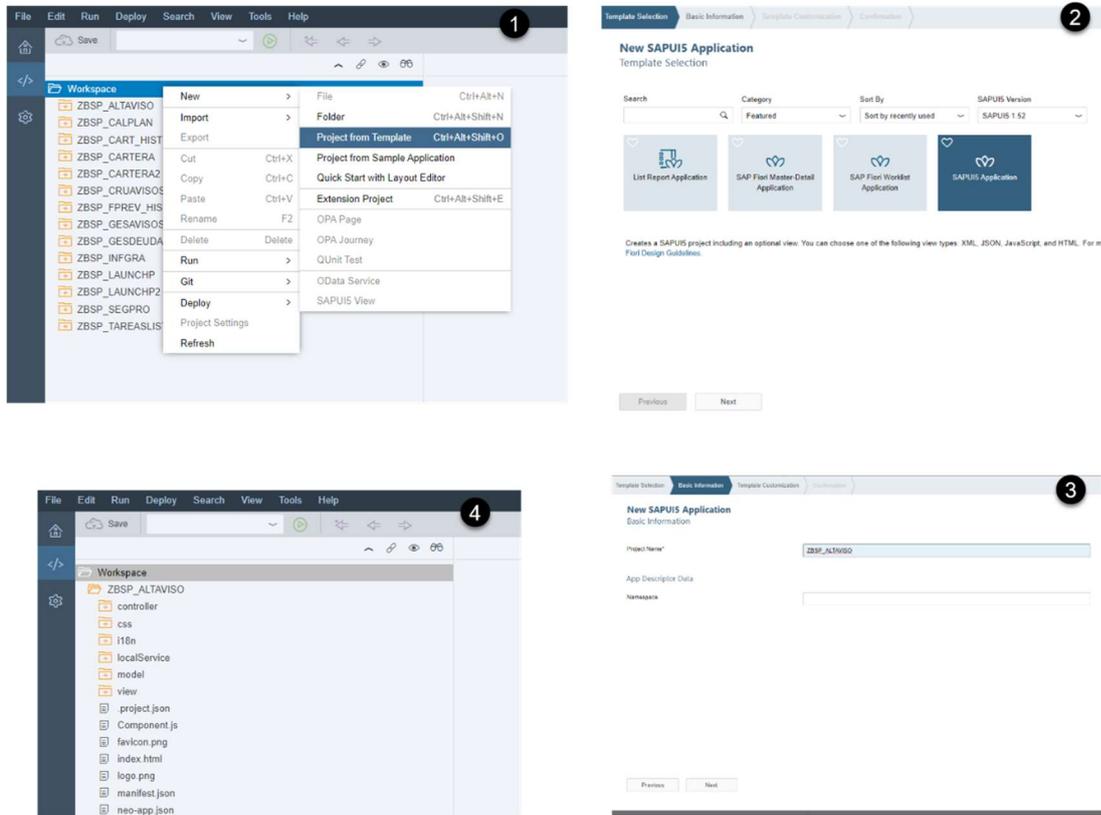
##### **Componente (Component.js)**

El archivo contiene la configuración de la aplicación. SAPUI5 inicia automáticamente la función del componente cuando se crea una instancia del componente.Component.js.

Como hicimos para describir el proceso de implementación de servicios Odata en este apartado también mediante la implementación de la aplicación Altaviso describiremos el procedimiento detallado que luego se extenderá al resto ya que todas las aplicaciones de este TFG se desarrollan siguiendo ese procedimiento.

Dentro de la herramienta WebId vamos a crear un nuevo proyecto para nuestra aplicación al que nombraremos ZBSP\_ALTAVISO.

Ilustración 58: WebId Crear proyecto



Al seleccionar la plantilla vemos que el proyecto se nos ha generado con la estructura de carpetas típica de una aplicación SAPUI5. Los siguientes pasos son configurar la aplicación desde el archivo manifest.json. Empezamos por la asignación de servicios y modelos.

Ilustración 59: Asignación de Servicio Odata

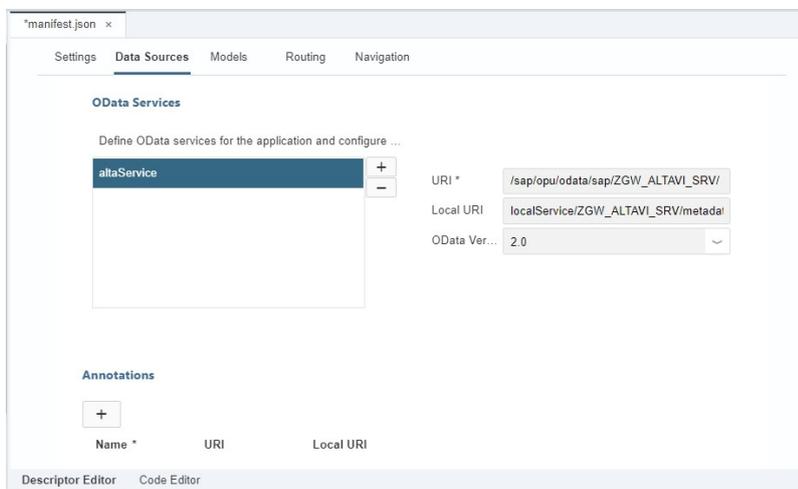
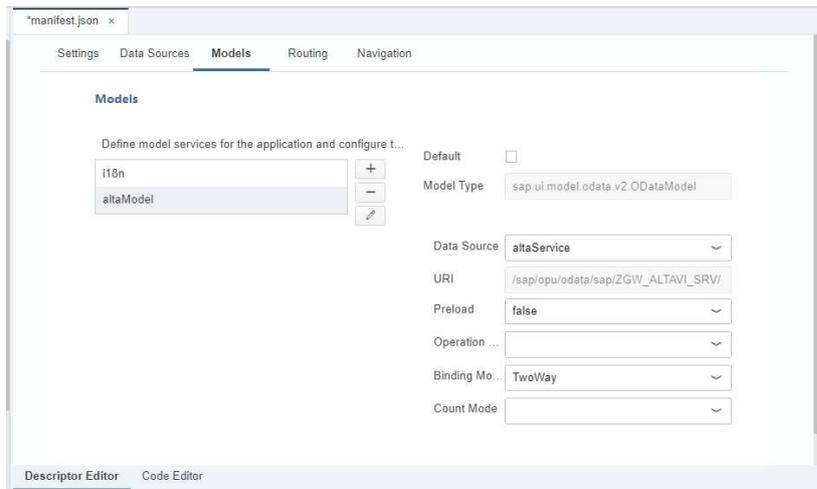
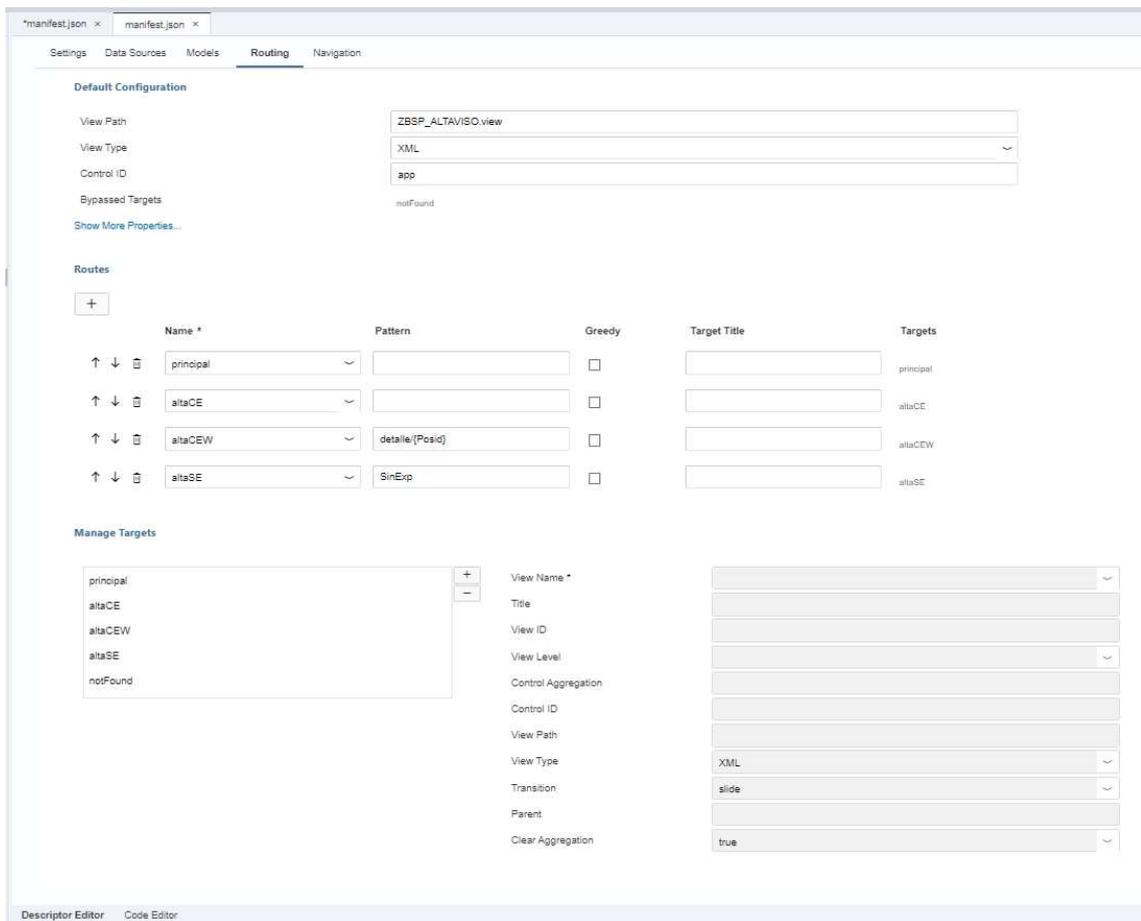


Ilustración 60:Asignación de modelo



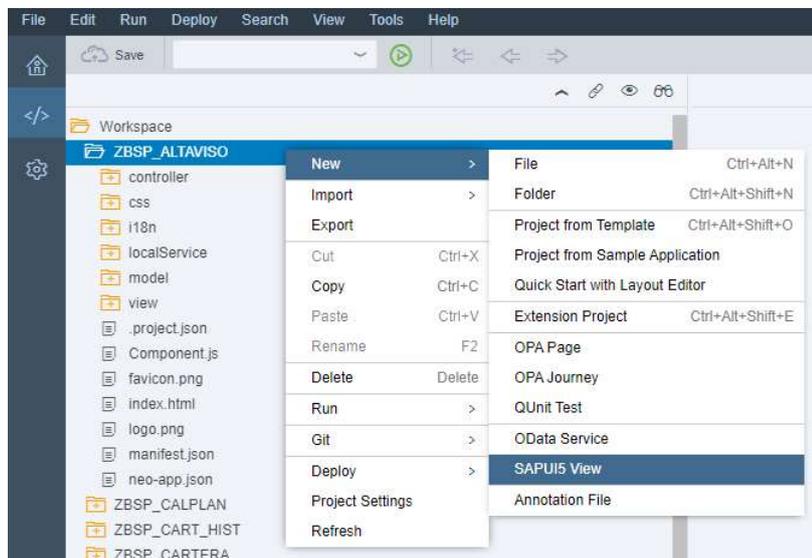
Se debe configurar también el Routing para habilitar la navegación entre vistas en la aplicación.

Ilustración 61:Conf de rutas



El siguiente paso es crear las vistas para definir la interfaz de usuario de nuestra aplicación.

Ilustración 62: Crear vistas



En SPAUI5 una vista puede ser de distintos tipos:

- **XML view** (file or string in XML format); the XMLView type supports a mix of XML and plain HTML.
- **JSON view** (file or string in JSON format)
- **JS view**, constructed in a traditional manner
- **HTML view** (file or string in HTML format)

Todas las vistas que implementemos serán del tipo XML view. Este tipo de vista se define en un archivo XML con la etiqueta `<mvc:View>` `</mvc:View>` ya que todas son instancias de la clase base `sap.ui.core.mvc.View`. Una vista XML debe incluir el nombre del controlador asociado a la misma y espacio de nombres que representan el mapeo de las librerías de control SAPUI5 donde se encuentran los elementos que contendrá la vista.

Ilustración 63: Estructura vista XML

```

1 <mvc:View xmlns:core="sap.ui.core" xmlns:mvc="sap.ui.core.mvc" xmlns="sap.m" controllerName="ZBSP_ALTAVISO.controller.AltaCEW"
2   xmlns:html="http://www.w3.org/1999/xhtml" xmlns:layout="sap.ui.layout" xmlns:form="sap.ui.layout.form" xmlns:table="sap.ui.table">
3   <Page title="Alta de aviso Con Expediente" showNavButton="true" navButtonPress="onNavBack">
4     <content>
5       <ObjectHeader id="cabExp" visible="false"></ObjectHeader>
6       <Wizard id="AltaCEWizard" enableBranching="true" complete="wizardCompletedHandler" finishButtonText="Guardar"></Wizard>
272     </content>
273     <footer>
274       <Toolbar >
275         <content>
276           <ToolbarSpacer/>
277           <Button press="addObsvDialogPress" id="addObsv" icon="sap-icon://post"/>
278           <ToolbarSpacer id="space1" width="1em"/>
279         </content>
280       </Toolbar>
281     </footer>
282   </Page>
283 </mvc:View>
  
```

Además de las vistas XML asociadas a ellas tenemos los fragmentos XML. Los fragmentos son partes de la interfaz de usuario (subárboles de la interfaz de usuario) que se pueden reutilizar, pero no tienen ningún controlador. Esto significa que, siempre que desee definir una parte determinada de la interfaz de usuario para que sea reutilizable en varias vistas, o cuando desee intercambiar algunas partes de una vista entre sí en determinadas circunstancias (diferentes roles de usuario, modo de edición frente al modo de solo lectura), un fragmento es un buen candidato, especialmente cuando no se requiere ninguna lógica de controlador adicional. Los fragmentos nos permiten también implementar cuadros de diálogo y pop ups.

Ilustración 64: estructura fragmento XML

```

1 <core:FragmentDefinition
2   xmlns="sap.m"
3   xmlns:core="sap.ui.core">
4
5   <SelectDialog noDataText="No se han encontrado operadores" title="seleccion de operador" search="operDialogStaticSearch"
6     liveChange="operDialogLiveSearch" confirm="operDialogHandleConfirm" close="operDialogHandleClose" items="{Operadores}/results">
7     <StandardListItem title="{Operadores}>Ename"
8       type="Active" />
9   </SelectDialog>
10
11 </core:FragmentDefinition>
  
```

Cuando creamos una vista se crea automáticamente un controlador en el que deberemos implementar el manejo de eventos en la vista y la comunicación con el servidor backend asociado a esta. Un controlador contiene los métodos que definen cómo interactúan el modelo y la vista tiene una relación de herencia con la clase base sap.ui.core.mvc.Controller todos los controladores extienden de ella. SAPUI5 presenta cuatro métodos predefinidos para implementar el ciclo de vida de la vista y el controlador.

- **onInit():** se llama cuando se crea una instancia de una vista y ya se han creado sus controles (si están disponibles); para modificar la vista antes de que se muestre para enlazar controladores de eventos y hacer otra inicialización de una sola vez
- **onExit():** se llama cuando se destruye la vista; para liberar recursos y finalizar actividades
- **onAfterRendering():** se llama cuando se ha representado la vista y, por lo tanto, su HTML forma parte del documento; utilizado para hacer manipulaciones posteriores a la representación del HTML. Los controles SAPUI5 obtienen este enlace después de ser renderizados.
- **onBeforeRendering():** se invoca antes de que se vuelva a representar la vista del controlador y **no** antes de la primera representación; uso para invocar el gancho antes de la primera representación onInit()

Ilustración 65: Estructura controlador

```

*AltaCEW.controller.js x
1  sap.ui.define([
2    "./BaseController",
3    "sap/m/MessageBox",
4    "sap/ui/core/Fragment",
5    "sap/m/Dialog",
6    "sap/m/Button",
7    "sap/m/MessageToast",
8    "sap/ui/model/Filter",
9    "sap/ui/model/FilterOperator",
10   "sap/ui/model/json/JSONModel"
11  ], function(BaseController, MessageBox, Fragment, Dialog, Button, MessageToast, Filter, FilterOperator, JSONModel) {
12    "use strict";
13
14    return BaseController.extend("ZBSP_ALTAVISO.controller.AltaCEW", {
15
16      onInit: function() {},
21      _onObjectMatched: function(oEvent) {},
117     varInit: function() {},
194     onExit: function() {},
199     setAvisoTypeFromSegmented: function(evt) {},
205     valSelMatStep: function(oEvent) {},
224     valNextSelMatStep: function() {},
241     valStepTomaCE: function(oEvent) {},
265     valNextStepTomaCE: function() {},
286     tomaStepComplete: function() {},
297     valStepActividadCE: function(oEvent) {},
300     valNextStepActividadCE: function() {},
322     actStepComplete: function() {},
325     wizardCompletedHandler: function() {},
363     changeDateFormat: function(fecha) {},
369     changeAvisoStruct: function() {},
499     discardProgress: function() {},
524     avisoTypeComplete: function() {},
541     matCESelectPress: function(oEvent) {},
633     muesSelectPress: function(oEvent) {},
678     muesDialogLiveSearch: function(oEvent) {},
684     muesSelectConfirm: function(oEvent) {},
784     servSelectPress: function(oEvent) {},
778     servDialogLiveSearch: function(oEvent) {},
776     servSelectConfirm: function(oEvent) {},
798     hormSelectPress: function() {},
811     hormDialogLiveSearch: function() {},
814     hormSelectConfirm: function(oEvent) {},
833     operTomSelectPress: function(oEvent) {},
862     operDialogLiveSearch: function(oEvent) {},
868     operDialogStaticSearch: function(oEvent) {},
876     operDialogHandleConfirm: function(oEvent) {}
909
910   });
911
912 });

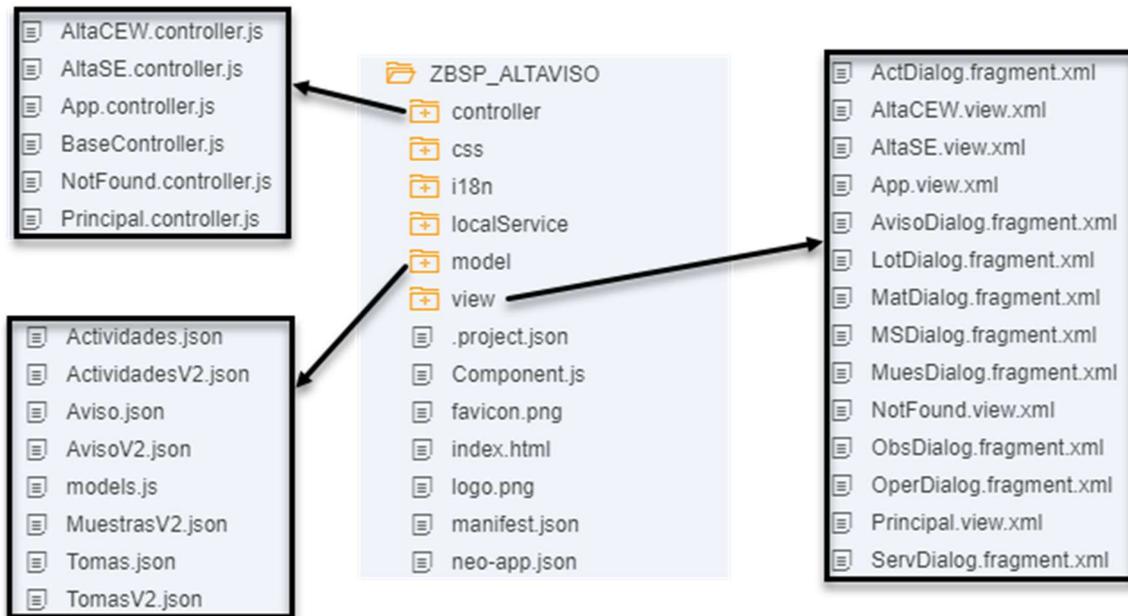
```

### 3.3.3.2 Aplicaciones SAPUI5 implementadas

#### ZBSP\_\_ALTAVISO

En el proyecto asociado a la aplicación alta de aviso se han definido las vistas controladores y fragmentos siguientes

Ilustración 66:ZBSP\_ALTAVISO



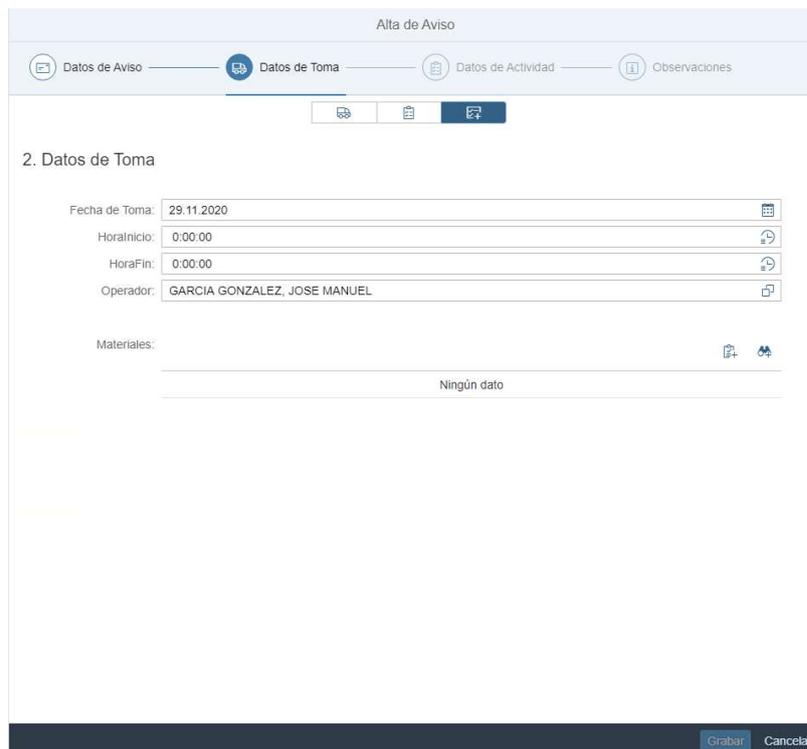
En esta aplicación destaca el papel de un elemento en particular dentro de la vista AltaCEW el elemento wizard que mediante las definición de pasos y control de eventos nos permite definir una serie de pasos secuenciales para realizar el proceso de alta de un aviso.

Ilustración 67: AltaCEW.view.xml

```

*AltaCEW.view.xml x
1 <mvc:View xmlns:core="sap.ui.core" xmlns:mvc="sap.ui.core.mvc" xmlns="sap.m" controllerName="ZBSP_ALTAVISO.controller.AltaCEW"
2   xmlns:html="http://www.w3.org/1999/xhtml" xmlns:layout="sap.ui.layout" xmlns:form="sap.ui.layout.Form" xmlns:table="sap.ui.table">
3   <Page title="Alta de aviso Con Expediente" showNavButton="true" navButtonPress="onNavBack">
4     <content>
5       <ObjectHeader id="cabExp" visible="false"></ObjectHeader>
6       <Wizard id="AltaCEWWizard" enableBranching="true" complete="wizardCompletedHandler" finishButtonText="Guardar">
7         <WizardStep id="SelMatStep" title="Selección de Materiales" complete="avisoTypeComplete" subsequentSteps="TomaCEStep ActiCEStep"
8           validated="false">
9           <form:SimpleForm editable="true" layout="ResponsiveGridLayout"></form:SimpleForm>
10          <HBox alignItems="Center" justifyContent="Center" width="100%">
11            <SegmentedButton id="segmentButtonCE" width="230px" select="setAvisoTypeFromSegmented"></SegmentedButton>
12          </HBox>
13        </WizardStep>
14        <WizardStep id="TomaCEStep" title="Datos de Toma" validated="false" complete="tomaStepComplete">
15          <form:SimpleForm editable="true" layout="ResponsiveGridLayout"></form:SimpleForm>
16          <form:SimpleForm id="datosHormigon" editable="true" layout="ResponsiveGridLayout" visible="false"></form:SimpleForm>
17          <Table id="tableMuesPC" items="{MuesPL}" growing="true" growingThreshold="50" visible="false"></Table>
18          <Table id="tableLot" items="{Tlotif}" growing="true" growingThreshold="50" visible="false"></Table>
19          <Toolbar visible="false"></Toolbar>
20        </WizardStep>
21        <WizardStep id="ActiCEStep" title="Datos Actividad" complete="actStepComplete" validated="false">
22          <form:SimpleForm editable="true" layout="ResponsiveGridLayout"></form:SimpleForm>
23          <Table id="tableServPC" items="{ServPC}" growing="true" growingThreshold="50" updateFinished="valNextStepActividadCE"></Table>
24          <Toolbar visible="false"></Toolbar>
25        </WizardStep>
26      </Wizard>
27    </content>
28    <footer>
29      <Toolbar >
30        <content>
31          <ToolbarSpacer/>
32          <Button press="addObsvDialogPress" id="addObsv" icon="sap-icon://post"/>
33          <ToolbarSpacer id="space1" width="1em"/>
34        </content>
35      </Toolbar>
36    </footer>
37  </Page>
38 </mvc:View>
  
```

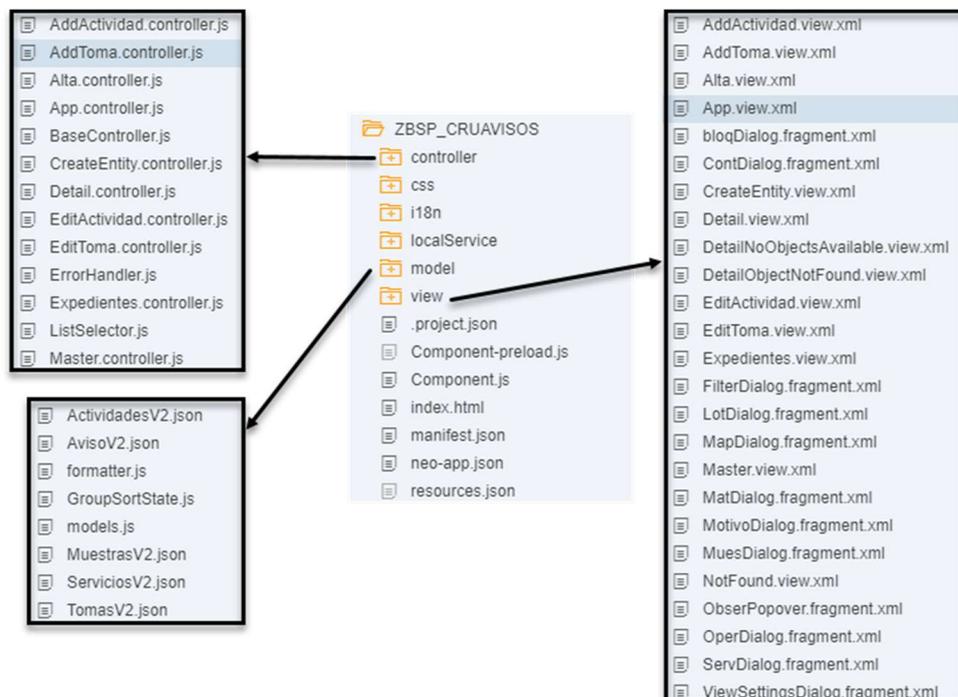
Ilustración 68: Interfaz gráfica Wizard



## ZBSP\_AVISOS

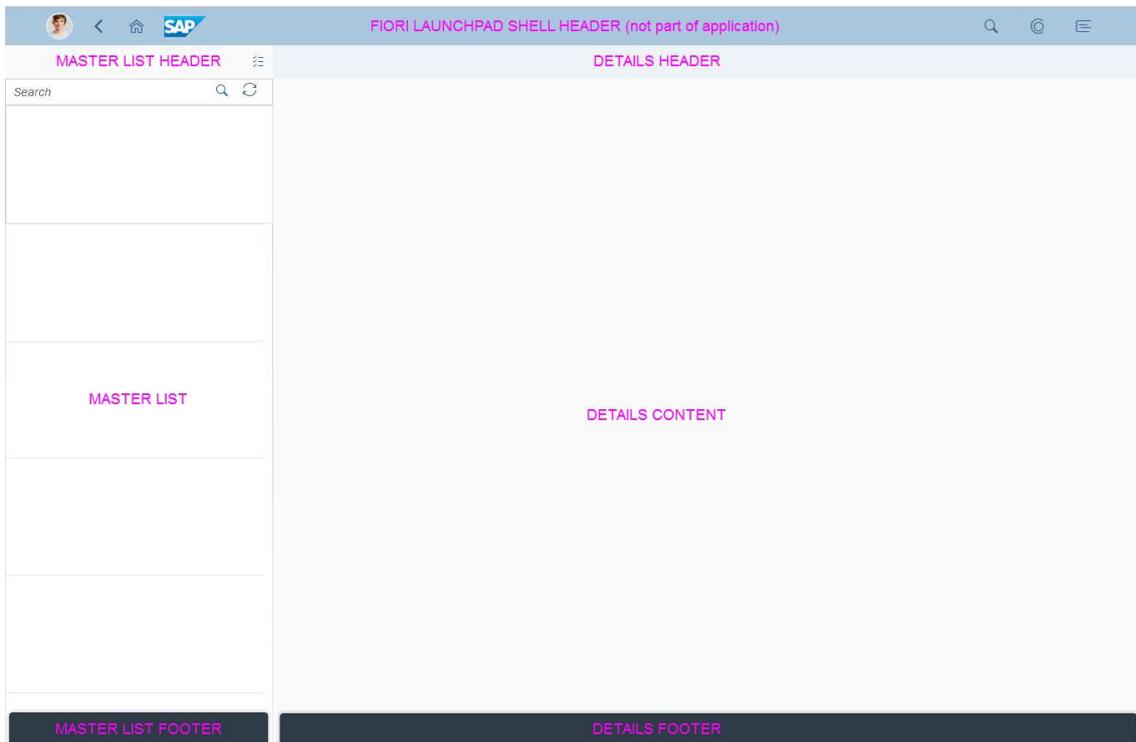
Los componentes creados para esta aplicación son los siguientes.

Ilustración 69: ZBSP\_AVISOS



La aplicación avisos se ha implementado siguiendo un modelo de pantalla dividida, un modelo recomendado para implementar procesos de negocio en los cuales se debe revisar y procesar diferentes elementos rápidamente con una navegación mínima.

Ilustración 70: Estructura Fuente: split-screen layout



Fuente: <https://experience.sap.com/fiori-design-web/v1-44/split-screen/>

Esto se define en la vista principal de la aplicación y a partir de ello podemos desarrollar una vista independiente para cada área.

Ilustración 71: App.view.xml

```
*AltaCEW.view.xml x App.view.xml x
1 <mvc:View controllerName="avisos.controller.App" xmlns:mvc="sap.ui.core.mvc" displayBlock="true" busy="{appView>/busy}"
2   busyIndicatorDelay="{appView>/delay}" xmlns="sap.m">
3   <SplitApp id="idAppControl"/>
4 </mvc:View>
```

## ZBSP\_CALPLAN

La estructura del proyecto y sus elementos son los siguientes

Ilustración 72:ZBSP\_CALPLAN



Toda la aplicación se basa en la inclusión de un elemento de control de tipo Planning Calendar el cual permite a los usuarios ver diferentes tareas al mismo tiempo y modificarlas. Permite al usuario mostrar citas para varios objetos, como un calendario de equipo, y compararlas entre sí.

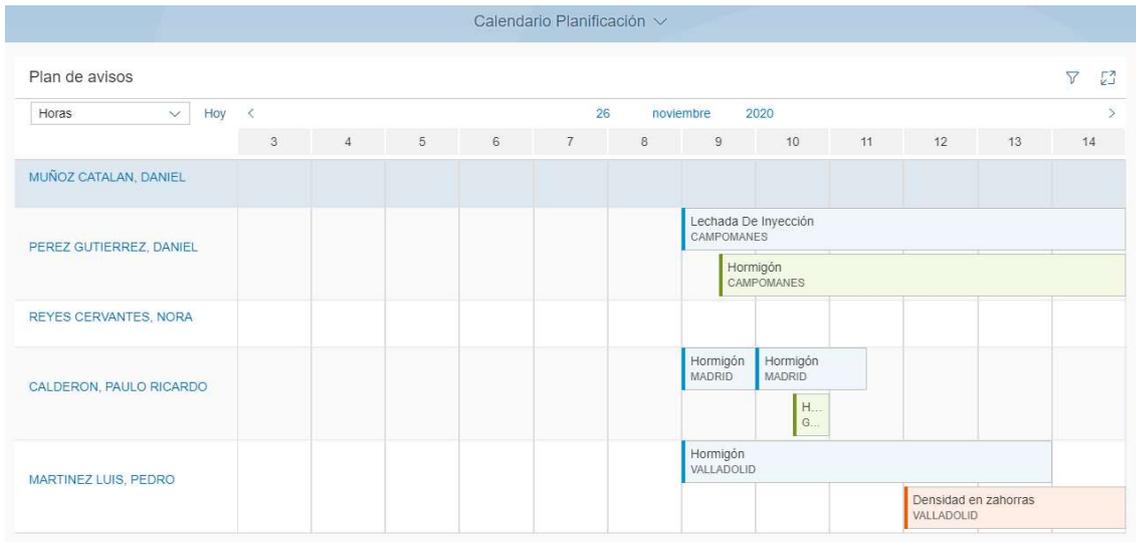
Ilustración 73: Planificación.view.xml

```

*Planificacion.view.xml <
1 <mvc:View xmlns:core="sap.ui.core" xmlns:mvc="sap.ui.core.mvc" xmlns="sap.m" controllerName="calendario.controller.Planificacion"
2 xmlns:html="http://www.w3.org/1999/xhtml" xmlns:unified="sap.ui.unified" xmlns:l="sap.ui.layout">
3 <Page showHeader="false">
4 <content>
5 <VBox class="sapUiSmallMargin">
6 <PlanningCalendar class="ClassPlanCal" id="PC1" rows="{path: 'planModel/UsuarioToTareasNav', appointmentsVisualization='Filled' viewKey='Hour' viewChange='handleViewChange'
7 appointmentSelect='handleAppointmentSelect' showEmptyIntervalHeaders='false' intervalSelect='handleIntervalChange' startDateChange='handleStartDate' }>
8 <toolbarContent>
9 <Title text="{i18n>plTitle}" titleStyle="H4"/>
10 <ToolbarSpacer/><ToolbarSpacer/>
11 <Button tooltip="Aplicar filtro" icon="sap-icon://filter" press="handleFl10peButtpress"/>
12 <Button id="idFullScreen" icon="sap-icon://full-screen" press="onFullScreenPress" visible="{= !${device}/system/phone }"/></Button>
13 </toolbarContent>
14 <rows >
15 <PlanningCalendarRow title="{planModel>Ename}"
16 appointments="{path : 'planModel>UsuarioToTareasNav', templateShareable: 'true'}" >
17 <customData>
18 </customData>
19 <appointments>
20 <unified:CalendarAppointment key="{path: 'planModel>Idaviso'"
21 startDate="{parts:[ {path: 'planModel>Fecha'}, {path: 'planModel>HoraIni' } ], formatter: '.formatter.formatDate'}"
22 endDate="{parts:[ {path: 'planModel>Fecha'}, { path: 'planModel>HoraFin'} ], formatter: '.formatter.formatDate'}"
23 title="{planModel>Texto}" text="{planModel>Poblacion}" type="{planModel>Tipo}"/></unified:CalendarAppointment>
24 </appointments>
25 </PlanningCalendarRow>
26 </rows>
27 </PlanningCalendar>
28 </VBox>
29 </content>
30 </Page>
31 </mvc:View>

```

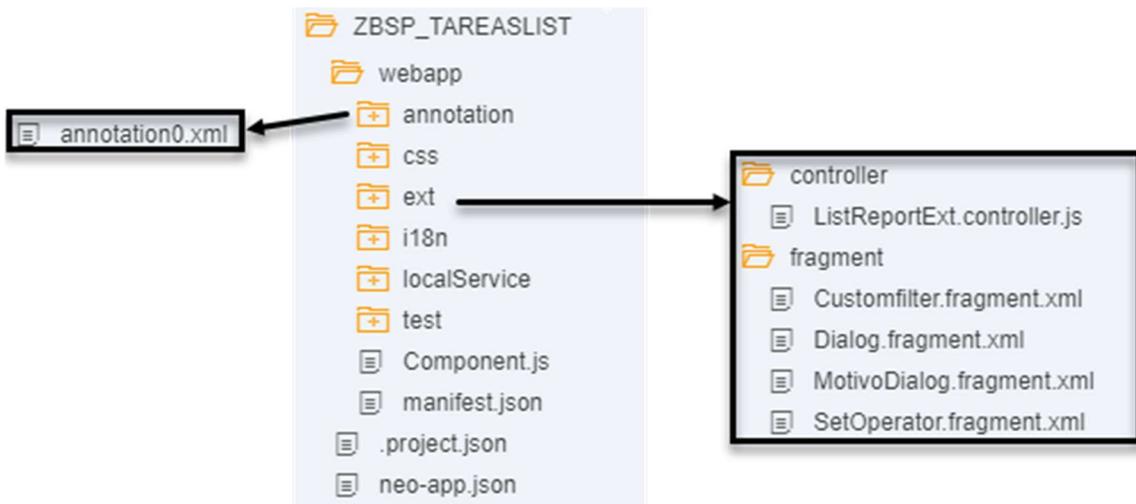
Ilustración 74: Interfaz Planning Calendar



## ZBSP\_TAREASLIST

Esta aplicación difiere del resto en estructura ya que se ha implementado usando un Fiori – Element “List report”. Los elementos de SAP Fiori (anteriormente conocidos como plantillas inteligentes) proporcionan un marco para los patrones de aplicación más comunes.

Ilustración 75:ZBSP\_TAREASLIST



El uso de elementos permite tener una aplicación disponible en un entorno productivo de manera rápida ya que simplemente tenemos que construir el servicio web e instanciarlo a la plantilla inteligente. Si los requerimientos de una aplicación son exactamente los que proporciona el elemento no es necesario escribir ni una línea de código. En el caso de que no cumpla exactamente con los requerimientos, pero se parezca mucho a lo que queremos implementar este elemento proporciona una api para extender tanto su interfaz de usuario como su

controlador. Este último es nuestro escenario por ello se han creado fragmentos XML para extender la funcionalidad de la vista y se ha extendido el controlador estándar de la aplicación.

Ilustración 76: ListReportExt.controller.js

```

*ListReportExt.controller.js x
1  sap.ui.controller("avisoslistzbsp_tareaslist.ext.controller.ListReportExt", {
2
3  ▶  onInit: function() {
36 ▶  _handleRouteMatched: function(oEvent) {
133 ▶  onBeforeRebindTableExtension: function(oEvent) {
158 ▶  onClickActionAvisosSet1: function(oEvent) {
204 ▶  onClickActionAvisosSet2: function(oEvent) {
300 ▶  handleModifButton: function(oEvent) {
436 ▶  handelModcontext: function(oEvent) {
440 ▶  handleAnulButton: function(oEvent) {
454 ▶  rejectNotif: function(oEvent) {
459 ▶  confirmNotif: function(oEvent) {
467 ▶  handleCancelButton: function(oEvent) {
480 ▶  afterClose: function(oEvent) {
484 ▶  handleAfterClose: function(oEvent) {
498 ▶  navToAvisos: function(oEvent) {
502
503  });

```

Estrechamente relacionado al conceto de elementos Fiori están las anotaciones. Son ficheros XML que nos permiten ampliar los servicios de OData mediante la adición de información sobre cómo interpretar el servicio OData y sus datos.

Ilustración 77: Interfaz de implementación de anotaciones

The screenshot shows the 'Annotation Structure' window in SAP. At the top, it says 'OData Data Source: mainService'. Below that is a 'Select Targets' button and a search bar. The main part is a table with the following columns: Node, Edit Qualifier, Key Information, Expression Type, Value, and Actions. The table is expanded to show a tree structure under 'Local Annotations' and 'UI.LineItem'. Under 'UI.LineItem', there are several 'UI.DataField' entries with various 'Value' properties like '(Fecha)', '(HoraInI)', '(Tipo)', '(Descripcion)', '(Posid)', '(Descob)', '(Ktext)', '(Operador)', and '(Namecd)'. There is also a 'UI.SelectionFields' entry with a value of '{(Descob), (Posid), (Name1), (Descripcion), (Operador), ...}'. At the bottom, there is a 'Properties' section with a 'Local Annotations' entry.

Node	Edit Qualifier	Key Information	Expression Type	Value	Actions
Local Annotations		Source: annotation0			+
UI.LineItem					+ 🗑
UI.DataField		Value: (Fecha)			+ 🗑
UI.DataField		Value: (HoraInI)			+ 🗑
UI.DataField		Value: (Tipo)			+ 🗑
UI.DataField		Value: (Descripcion)			+ 🗑
UI.DataField		Value: (Posid)			+ 🗑
UI.DataField		Value: (Descob)			+ 🗑
UI.DataField		Value: (Ktext)			+ 🗑
UI.DataField		Value: (Operador)			+ 🗑
UI.DataField		Label: Cliente; Value: (Namecd)			+ 🗑
UI.DataField		Value: (Observaciones)			+ 🗑
UI.SelectionFields		{(Descob), (Posid), (Name1), (Descripcion), (Operador), ...}			+ 🗑
Properties					
Idaviso					
Local Annotations		Source: annotation0			+

# 4

## Aplicación de evaluación de deuda

A lo largo de este capítulo se tratará el desarrollo de una aplicación SAPUI5 analítica que nos sirva como herramienta para determinar el perfil de crédito de cada cliente.

### **4.1 Análisis**

#### **4.1.1 Descripción Funcional**

La funcionalidad de esta aplicación es la de servir de herramienta analítica para evaluar distintos aspectos económicos de los clientes y delegaciones que forman la empresa, cuyos resultados sirven de ayuda en la toma de decisiones en los procesos de gestión de cobro, clasificación de clientes y gestión de crédito dentro de SAP-ERP.

Partiendo de un listado con todos los clientes/delegaciones con importes adeudados. La funcionalidad específica de este es presentar junto con cada registro de cliente una gráfica de balance trimestral, con el objetivo de evaluar rápidamente si el cliente ha subido su deuda, la ha bajado o la ha mantenido y en qué porcentaje.

Este listado implementará funcionalidades de búsqueda por nombre o delegación para acelerar el proceso de evaluación de un cliente/delegación específico/a. Al seleccionar un cliente/delegación específico podemos ver en detalle los datos de evolución temporal de importes en cuanto a deuda cobros, ventas y contratación en formato gráfico o tabla.

La principal ventaja de la presentación de los datos en formato de gráficas es que se puede ver rápidamente como ha ido evolucionando el nivel de deuda o contratación de un cliente de manera casi inmediata. Además, se puede ver un KPI de contratación sobre deuda la que permite hacer un análisis coherente del ratio de valoración de un cliente. El hecho de que un cliente tenga importes altos de deuda no quiere siempre decir que sea un cliente malo.

Sobre esta aplicación existen distintas variantes según el periodo de endeudamiento se pueden ver los datos de deuda vencida a >-90, -60, -30 y deuda a vencer a 30 60 y >90. Todas ellas cuentan con una funcionalidad adicional, las gráficas se pueden visualizar en pantalla completa.

#### 4.1.2 Catálogo de requisitos

##### 4.1.2.1 Funcionales

<b>RFED001</b>	<i>Listado de clientes</i>
<b>Descripción</b>	<i>La aplicación debe recuperar y mostrar un listado con los todos los clientes que tengan importes de deuda en el sistema.</i>
<b>RFED002</b>	<i>Búsqueda de cliente</i>
<b>Descripción</b>	<i>Se debe permitir la búsqueda de un cliente específico.</i>
<b>RFED003</b>	<i>Evolución contratación</i>
<b>Descripción</b>	<i>El sistema debe recuperar y mostrar en formato de gráfica lineal o tabla los datos de evolución temporal los importes de contratación asociado a un cliente.</i>
<b>RFED004</b>	<i>Evolución ventas</i>
<b>Descripción</b>	<i>El sistema debe recuperar y mostrar en formato de gráfica lineal o tabla los datos de evolución temporal los importes de ventas asociado a un cliente.</i>

<b>RFED005</b>	<i>Evolución deuda</i>
<b>Descripción</b>	<i>El sistema debe recuperar y mostrar en formato de gráfica lineal o tabla los datos de evolución temporal los importes de deuda asociado a un cliente.</i>

<b>RFED006</b>	<i>Evolución Cobro</i>
<b>Descripción</b>	<i>El sistema debe recuperar y mostrar en formato de gráfica lineal o tabla los datos de evolución temporal los importes de cobro asociado a un cliente.</i>

<b>RFED007</b>	<i>Cambio formato</i>
<b>Descripción</b>	<i>El usuario debe poder elegir en que formato ver los datos para ello debe poder cambiar entre formato gráfico y tabla.</i>

<b>RFED008</b>	<i>Pantalla completa</i>
<b>Descripción</b>	<i>Cada gráfica o tabla de manera independiente debe poder ampliarse a pantalla completa.</i>

<b>RFED009</b>	<i>Datos por delegación y total empresa</i>
<b>Descripción</b>	<i>Al listado de clientes con deuda debe añadirse como registros cada delegación la empresa para poder ver los datos totalizados en función de estas.</i>

<b>RFED010</b>	<i>Variantes</i>
<b>Descripción</b>	<i>Se deben crear varios mosaicos con distintas variantes de los datos de deuda diferenciando entre total y solo vencida, así como rangos de tiempo de 30 a 60 de 60 a 90 y más de 90 días.</i>

<b>RFED011</b>	<i>Exportar datos de deuda</i>
<b>Descripción</b>	<i>El sistema debe implementar la funcionalidad necesaria para poder exportar los datos de deuda a una hoja de cálculo.</i>

#### 4.1.2.2 No Funcionales

<b>RNFCC001</b>	<i>Eficiencia</i>
-----------------	-------------------

<b>Descripción</b>	<i>Debido al volumen de los datos dentro de este desarrollo es necesario tener en cuenta que el optimizar la obtención de estos es prioritario.</i>
--------------------	---

#### 4.1.3 Casos de Uso

##### 4.1.3.1 Especificación

<b>CUED001</b>	<i>Evaluar cliente</i>
<b>Actores</b>	<i>Personal del dep. de dirección.</i>
<b>Descripción</b>	<i>Este caso de uso permite al actor involucrado evaluar a un cliente usando para ello los datos del perfil financiero de este.</i>
<b>Requisito</b>	<i>RFED003, RFED004, RFED005, RFED006</i>

<b>CUED002</b>	<i>Ver datos de cobros</i>
<b>Actores</b>	<i>Personal del dep. de dirección.</i>
<b>Descripción</b>	<i>El caso de uso permite al actor obtener la información sobre los datos de cobros de un cliente presentados en forma de una gráfica evolutiva de importes cobrados a lo largo de un periodo concreto.</i>
<b>Requisito</b>	<i>RFED006</i>

<b>CUED003</b>	<i>Ver datos de ventas</i>
<b>Actores</b>	<i>Personal del dep. de dirección.</i>
<b>Descripción</b>	<i>El caso de uso permite al actor obtener la información sobre los datos de cobros de un cliente presentados en forma de una gráfica evolutiva de importes cobrados a lo largo de un periodo concreto.</i>
<b>Requisito</b>	<i>RFED004</i>

<b>CUED004</b>	<i>Ver datos de deuda</i>
<b>Actores</b>	<i>Personal del dep. de dirección.</i>
<b>Descripción</b>	<i>El caso de uso permite al actor obtener la información sobre los datos de deuda de un cliente presentados en forma de una gráfica evolutiva de importes adeudados a lo largo de un periodo concreto.</i>

<b>Requisito</b>	<i>RFED005</i>
------------------	----------------

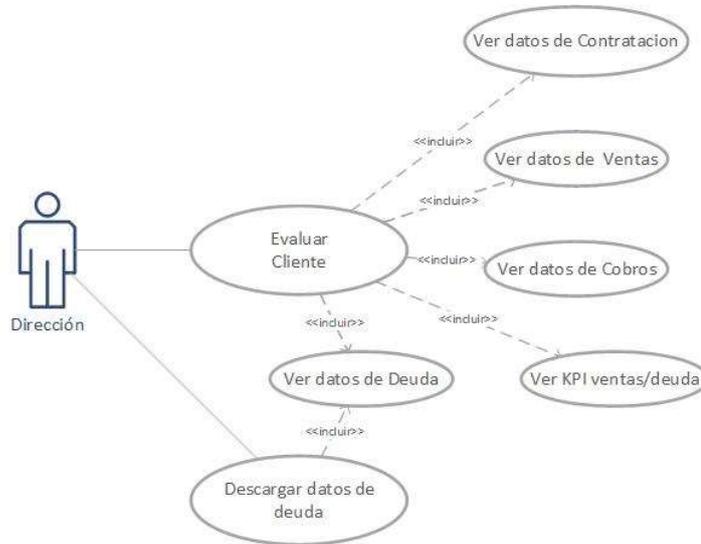
<b>CUED005</b>	<i>Ver datos de contratación</i>
<b>Actores</b>	<i>Personal del dep. de dirección.</i>
<b>Descripción</b>	<i>El caso de uso permite al actor obtener la información sobre los datos de contratación de un cliente presentados en forma de una gráfica evolutiva de importes de contratación a lo largo de un periodo concreto.</i>
<b>Requisito</b>	<i>RFED003</i>

<b>CUED006</b>	<i>Ver Kpi ventas/deuda</i>
<b>Actores</b>	<i>Personal del dep. de dirección.</i>
<b>Descripción</b>	<i>El caso de uso permite al actor obtener la información sobre el ratio de endeudamiento de un cliente presentado los resultados en forma de una gráfica evolutiva de ratios calculados a lo largo de un periodo concreto.</i>
<b>Requisito</b>	<i>RFED004, RFED005</i>

<b>CUED007</b>	<i>Descargar datos de deuda</i>
<b>Actores</b>	<i>Personal del dep. de dirección.</i>
<b>Descripción</b>	<i>Este caso de uso permite obtener por medio de la aplicación un listado mensual de los importes adeudados en un archivo Excel para su tratamiento fuera del sistema.</i>
<b>Requisito</b>	<i>RFGA011</i>

### 4.1.3.2 Diagramas

Ilustración 78: Casos de uso de deuda



### 4.1.4 Actores del Sistema

**Personal de dirección.** - Personas que componen el departamento de dirección de la empresa, así como los responsables de cada delegación.

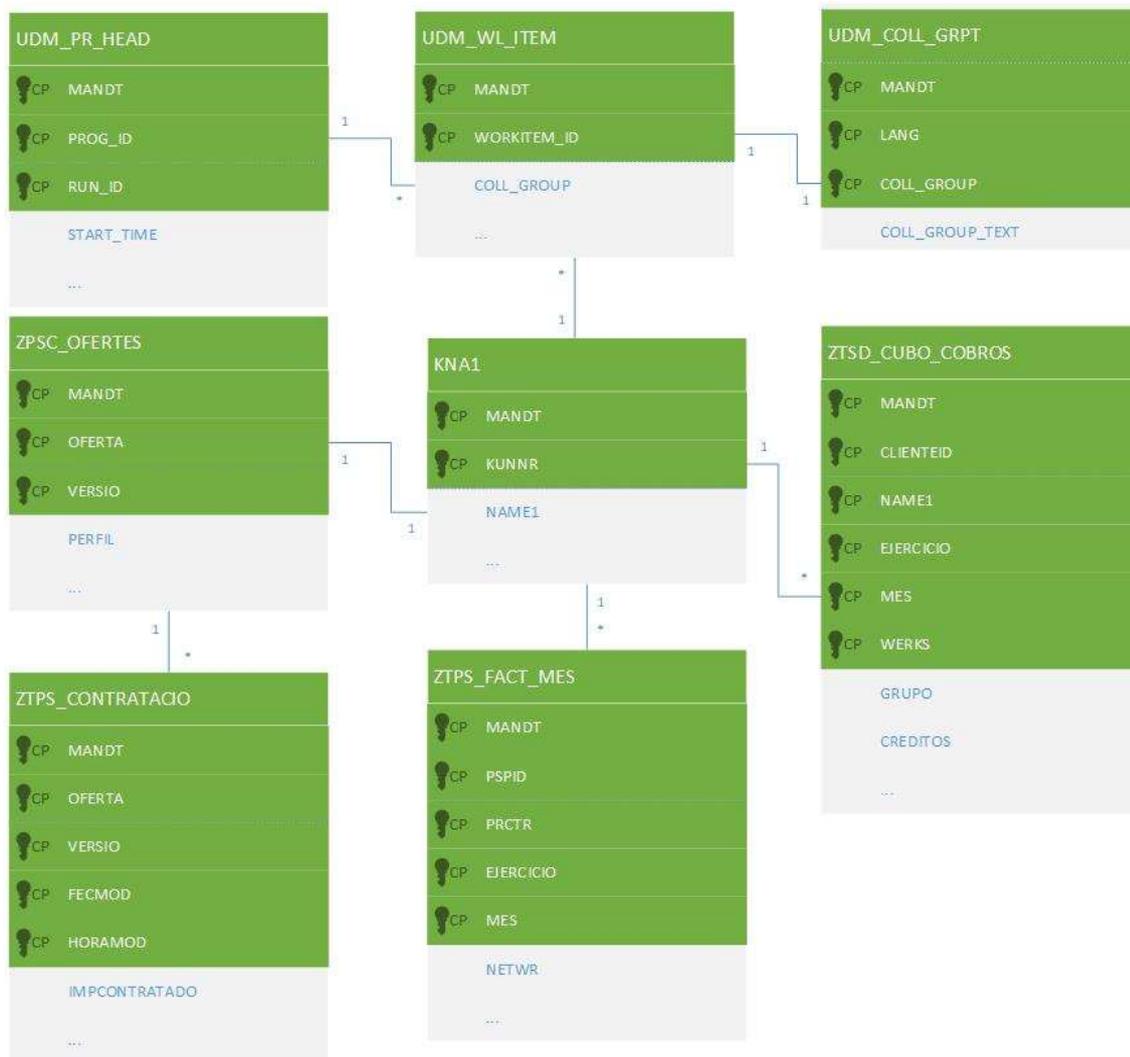
## 4.2 Diseño

### 4.2.1 Modelos entidad-relación BBDD

Ya que las funciones principales de esta aplicación solo están relacionadas con la explotación de datos no será necesario definir tablas para almacenar ningún tipo de dato. Lo que si es necesario es establecer son las tablas donde se almacenan los datos que necesita la aplicación.

- **UDM\_PR\_HEAD.** - Tabla de referencia de ejecuciones de programas en paralelo.
- **UDM\_WL\_ITEM.** - Tabla de registro de entrada en el pool de trabajo
- **UDM\_COLL\_GRPT.** - Tabla de denominación de grupo de colección.
- **ZPSC\_OFERTES.** - Tabla de datos de cabecera de oferta.
- **KNA1.**- Tabla maestro de clientes.
- **ZTSD\_CUBO\_COBROS.** - Tabla de datos de cobros por cliente centro ejercicio y mes.
- **ZTPS\_CONTRATACIO.** - Tabla de registro semanal de contratación.
- **ZTPS\_FACT\_MES.** - Tabla cubo de datos facturación por mes.

Ilustración 79:ER App Deuda



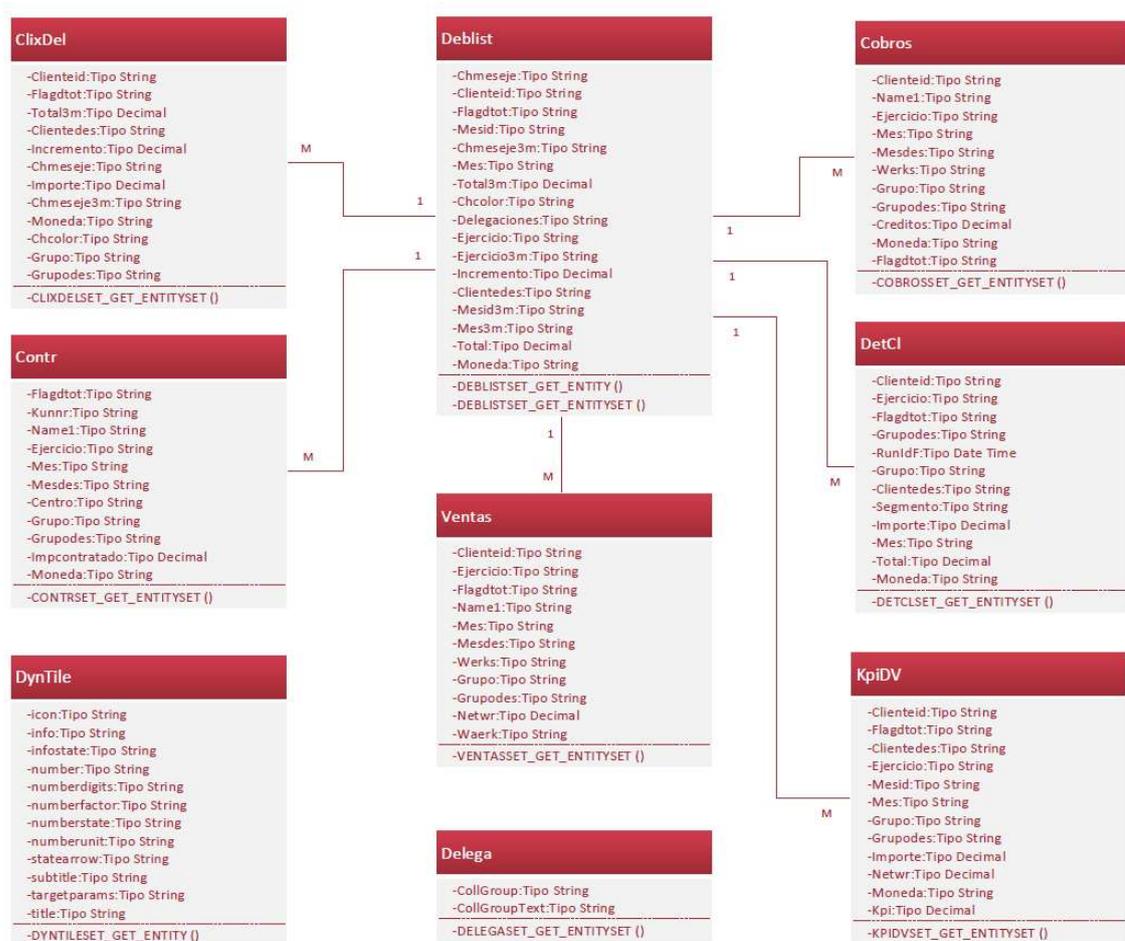
Como se aprecia en la Ilustración 79 que recoge el modelo ER de las tablas que son nuestra fuente de datos. En el modelo el eje central es la tabla estándar KNA1. Los datos de deuda se recuperan de las tablas del módulo de FCSM Gestión de cobro mientras que los datos de facturación ventas cobros y contratación son tablas Z que pertenecen al módulo PS y LB. En el caso de la tabla de cobros (ZTSD\_CUBO\_COBROS) es una tabla cubo, un contenedor de datos con antigüedad que nos permite tener disponible datos que generalmente lleva tiempo calcularlos directamente. Como desventaja al usar este tipo de tablas tenemos que el dato seleccionado tendrá un periodo de antigüedad.

#### 4.2.2 Modelado de Servicios Odata

La Ilustración 80 representa el modelo para la construcción del servicio Odata necesario para la aplicación que estamos desarrollando. La entidad principal denominada Deblast nos servirá para representar tanto a clientes como delegaciones. A esta entidad están ligadas el resto de las entidades referentes a

importes: Cobros, Ventas, DetCl (Deuda), Contr (Contratación) y el KpiDV conceptualmente todas ellas representan un punto en el tiempo que tiene un importe asociado. Al tener un conjunto de cada una de estas entidades podemos construir la evolución temporal de un tipo concreto de importe. Junto con estas existe otra entidad que solo toma sentido cuando la entidad principal representa una delegación ya que la finalidad de la entidad ClixDel es la de representar los clientes de una delegación.

Ilustración 80:MC Evolución de deuda



Como ya se ha ido viendo en los distintos modelos anteriores este también incluye la entidad DynTile para gestionar los datos dinámicos del tile de la aplicación. También incluye una entidad independiente que representará a un grupo de colección este. El concepto de grupo de colección se refiere a un parámetro de clasificación de clientes dentro del sistema.

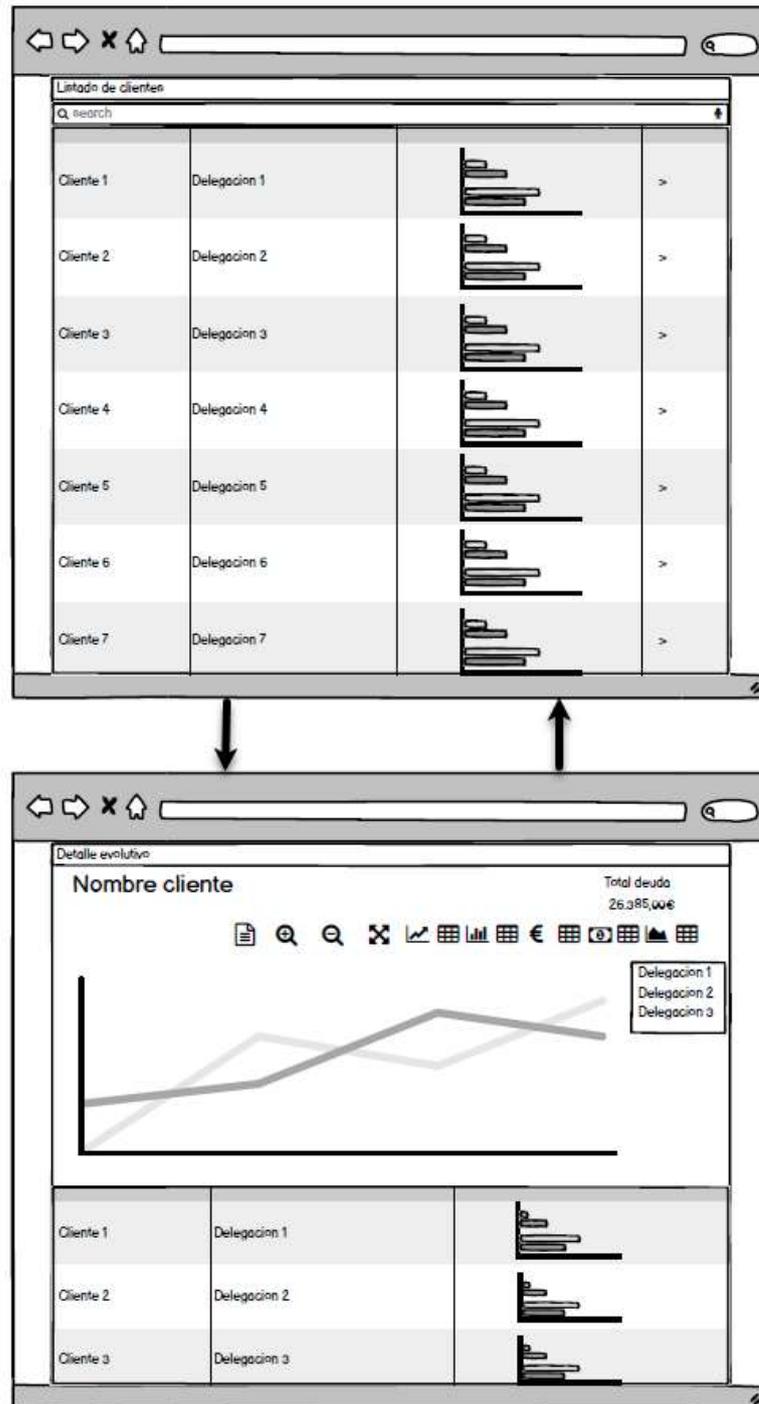
#### 4.2.3 Modelado aplicaciones SAPUI5

#### **4.2.3.1 Wireframes**

La interfaz de usuario de esta aplicación como representa la Ilustración 81 estará compuesta por dos vistas una principal y otra de detalle. La vista principal está compuesta por un listado de clientes/delegaciones desde el cual al pulsar un registro podemos navegar a la vista de detalle del elemento seleccionado.

La vista de detalle está estructurada en: una cabecera con datos del elemento, un contenedor gráfico y un listado de clientes que solo estará disponible si el elemento tratado es una delegación. El contenedor grafico merece especial mención ya que su función es la de presentar los datos que el usuario desee visualizar así este puede ir seleccionando que grafica quiere analizar.

Ilustración 81:WF App evolución de deuda



## 4.3 Implementación

### 4.3.1 Servicios implementados

ZGW\_GESDEUDACL

Se ha seguido el procedimiento de implementación descrito en el punto. EL resultado del modelado se puede ver en el mapeo de las entidades y la creación de asociaciones véase figura.

Ilustración 82: Mapeo E-E ZGW\_GESDEUDACL

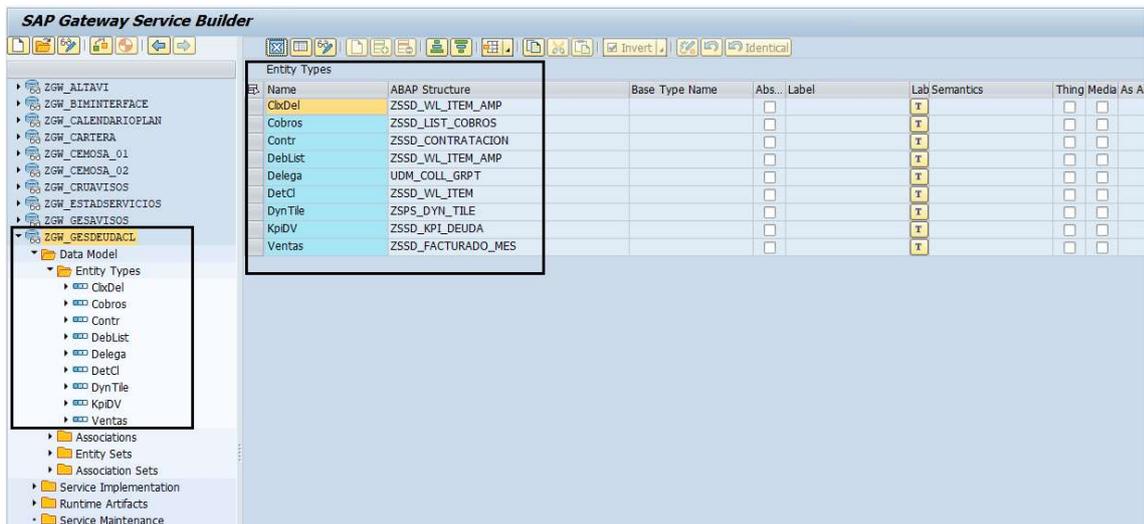
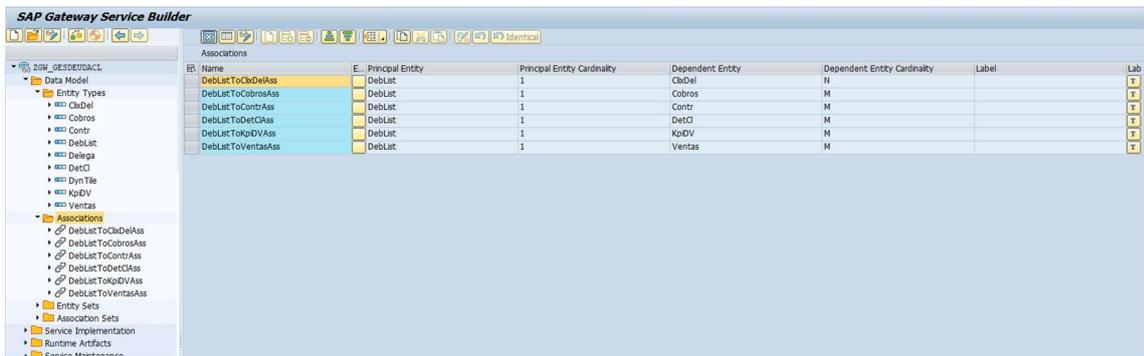
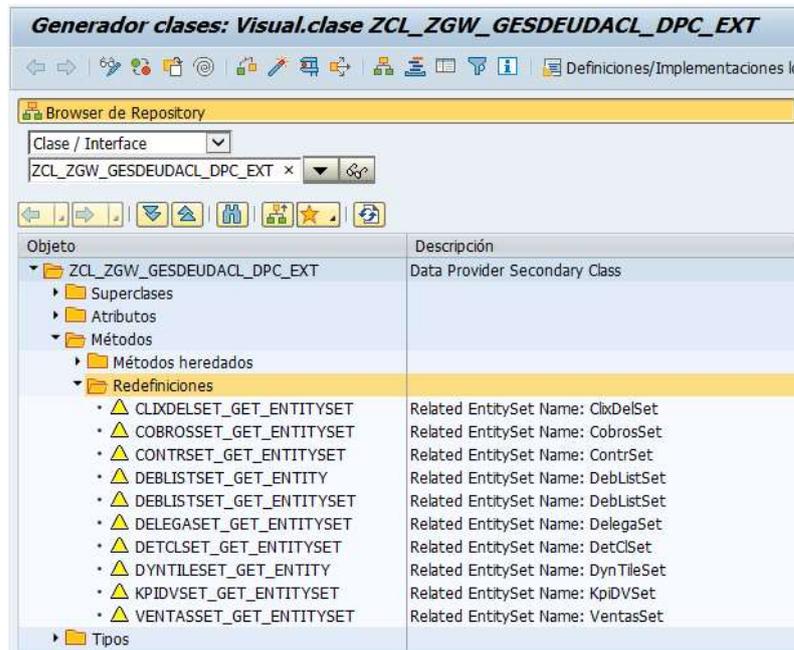


Ilustración 83: Asociaciones ZGW\_GESDEUDACL



En la fase de implementación se han redefinido los siguientes métodos.

Ilustración 84: Métodos redefinidos ZGW\_GESDEUDACL



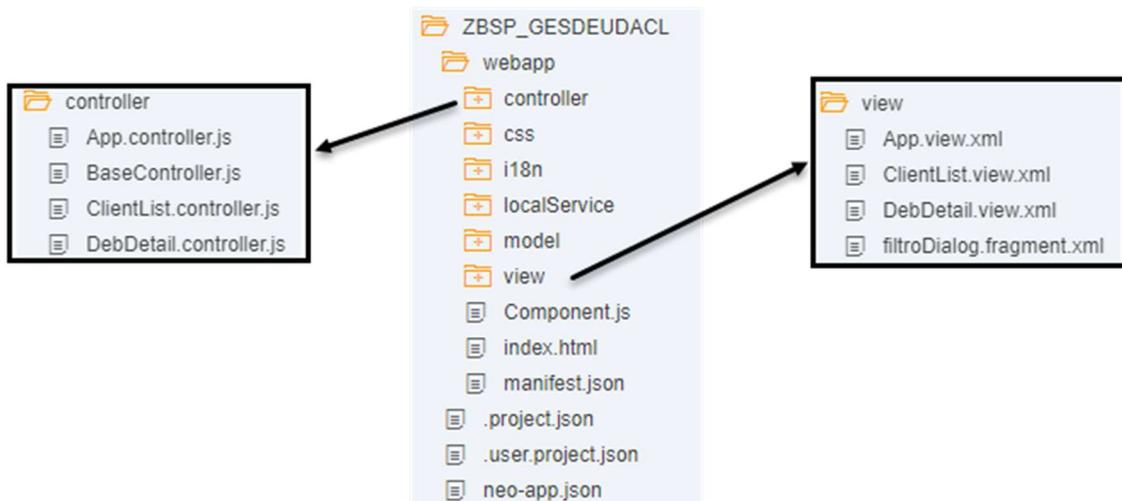
### 4.3.2 Frontend

#### 4.3.2.1 Aplicaciones SAPUI5 implementadas

##### ZBSP\_GESDEUDACL

Se han implementado las siguientes vistas y controladores para la aplicación de deuda.

Ilustración 85: ZBSP\_GESDEUDACL



Debido a que en la aplicación para presentar los datos se usan gráficas describiremos los controles con los que cuenta SAPUI5 para ese fin VizFrames.

Ilustración 86: Debdetail.view.xml

```

1 <mvc:View xmlns:core="sap.ui.core" xmlns:mvc="sap.ui.core.mvc" xmlns="sap.m" xmlns:html="http://www.w3.org/1999/xhtml"
2   xmlns:com="sap.suite.ui.commons" xmlns:viz="sap.viz.ui5.controls" xmlns:vizData="sap.viz.ui5.data"
3   xmlns:vizFeeds="sap.viz.ui5.controls.common.feeds" controllerName="ZBSP_GESDEUACL.controller.DebDetail"
4   xmlns:micro="sap.suite.ui.microchart">
5   <Page id="debDet" class="sapUiContentPadding" title="Detalle Evolutivo Cliente/Delegación" showNavButton="true" navButtonPress="onNavBack">
6     <ObjectHeader id="detClient" title="{cabCliente}/Clienteid" - {cabCliente}/Clientedes"></ObjectHeader>
7     <content>
8       <com:ChartContainer id="deboetChartContainer" showFullscreen="true" showPersonalization="false" autoAdjustHeight="true"
9         personalizationPress="attachPersonalizationPress" contentChange="attachContentChange">
10        <com:content>
11          <com:ChartContainerContent id="contKpiDV" icon="sap-icon://line-chart-time-axis" title="KpiV/D">
12            <com:content>
13              <viz:VizFrame id="vizFrameKpiDV" busy="false" busyIndicatorDelay="1000" visible="true" uiConfig="{ 'applicationSet': 'fiori' }"
14                vizType="line" vizProperties="{ title: {text: 'Kpi Ventas / Deuda', visible: true}}">
15                <viz:dataset>
16                  <vizData:FlattenedDataset data="{FrameKpiDV}/">
17                    <vizData:dimensions>
18                      <vizData:DimensionDefinition name="Mes" value="{FrameKpiDV}/Mes"/>
19                      <vizData:DimensionDefinition name="Delegación" value="{FrameKpiDV}/Grupodes"/>
20                    </vizData:dimensions>
21                    <vizData:measures>
22                      <vizData:MeasureDefinition name="Ratio" value="{FrameKpiDV}/Kpi"/>
23                    </vizData:measures>
24                    </vizData:FlattenedDataset>
25                  </viz:dataset>
26                  <viz:feeds>
27                    <vizFeeds:FeedItem uid="categoryAxis" type="Dimension" values="Mes"/>
28                    <vizFeeds:FeedItem uid="color" type="Dimension" values="Delegación"/>
29                    <vizFeeds:FeedItem uid="valueAxis" type="Measure" values="Ratio"/>
30                  </viz:feeds>
31                </viz:VizFrame>
32              </com:content>
33            </com:ChartContainerContent>
34            <com:ChartContainerContent id="contContra" icon="sap-icon://capital-projects" title="Contratación"></com:ChartContainerContent>
35            <com:ChartContainerContent icon="sap-icon://table-chart" title="Tabla Contratación"></com:ChartContainerContent>
36            <com:ChartContainerContent id="contVentas" icon="sap-icon://loan" title="Ventas"></com:ChartContainerContent>
37            <com:ChartContainerContent icon="sap-icon://table-chart" title="Tabla Ventas"></com:ChartContainerContent>
38            <com:ChartContainerContent icon="sap-icon://monitor-payments" title="Deuda"></com:ChartContainerContent>
39            <com:ChartContainerContent icon="sap-icon://table-chart" title="Tabla Deuda"></com:ChartContainerContent>
40            <com:ChartContainerContent id="contCobros" icon="sap-icon://money-bills" title="Cobros"></com:ChartContainerContent>
41            <com:ChartContainerContent icon="sap-icon://table-chart" title="Tabla Cobros"></com:ChartContainerContent>
42          </com:content>
43        </com:ChartContainer>
44      </content>
45      <Panel id="clixDelPanel" class="panelStyle" expandable="true" expanded="false" headerText="Listado Deudores" visible="false"></Panel>
46    </footer>
47  </Page>
48 </mvc:View>

```

# 5

## Aplicación Seguimiento de Conservación de carreteras

Este capítulo está destinado a describir las características, funcionalidades e implementación de una aplicación SAPUI5 destinada a un cliente específico, la cual debe ser capaz de ofrecerle un informe gráfico en tiempo real del estado de trabajos realizados.

### **5.1 Análisis**

#### **5.1.1 Descripción Funcional**

La funcionalidad principal de esta aplicación es la de servir de informe de seguimiento a tiempo real de los trabajos que se realizan sobre la conservación de carreteras para la diputación de Granada.

El informe se presenta en formato gráfico agrupando los datos por ejercicio/mes, zona y tipo de actuación o visita. Para ver en detalle las actuaciones está

disponible la funcionalidad de realizar filtros por los parámetros de agrupación: ejercicio/mes, zona y tipo de actuación.

Cumple también con la funcionalidad de presentar los resultados totalizados por zona junto con un KPI de actuaciones sobre ventas.

Como funcionalidad adicional para la visualización de los datos evolutivos de actuaciones y visitas se permite la presentación de las gráficas en pantalla completa.

Una funcionalidad específica para la gráfica de actuaciones es la de cambiar el formato de visualización a tabla y exportar está en un archivo Excel.

## 5.1.2 Catálogo de requisitos

### 5.1.2.1 Funcionales

<b>RFCC001</b>	<i>Visualizar Actuaciones</i>
<b>Descripción</b>	<p><i>Representación gráfica (gráfica de barras) de la evolución de actuaciones:</i></p> <ul style="list-style-type: none"> <li>○ <i>Señalización</i></li> <li>○ <i>Limpieza/Cunetas</i></li> <li>○ <i>Bacheos</i></li> <li>○ <i>Poda y desbroce</i></li> <li>○ <i>Fitosanitarios</i></li> <li>○ <i>Obras de fábrica y cunetas hormigón</i></li> <li>○ <i>Refuerzo de firme</i></li> <li>○ <i>Estabilización</i></li> <li>○ <i>Otros</i></li> </ul> <p><i>por zona y mes.</i></p>
<b>RFCC002</b>	<i>Visualizar Visitas</i>
<b>Descripción</b>	<p><i>Representación gráfica (gráfica de barras) de la evolución de Visitas por zona y mes.</i></p>
<b>RFCC003</b>	<i>Kpi Actuaciones / Visitas</i>
<b>Descripción</b>	<p><i>Representación gráfica (gráfica lineal) del porcentaje de actuaciones sobre visitas por zona.</i></p>
<b>RFCC004</b>	<i>Visualizar Totales x Zona</i>

<b>Descripción</b>	<i>Representación gráfica de las actuaciones y las visitas totales por zona.</i>
--------------------	--

<b>RFCC005</b>	<i>Filtrar resultados actuaciones</i>
<b>Descripción</b>	<p><i>El sistema debe implementar la posibilidad de permitir filtros por:</i></p> <ul style="list-style-type: none"> <li><i>○ Mes/año</i></li> <li><i>○ Zona</i></li> <li><i>○ Tipo de actuación</i></li> </ul> <p><i>para los datos de evolución de actuaciones.</i></p>

<b>RFCC006</b>	<i>Cambio de formato</i>
<b>Descripción</b>	<i>Es necesario que la aplicación permita presentar los datos en formato de tabla además de en formato de grafica de barras.</i>

<b>RFCC007</b>	<i>Pantalla completa</i>
<b>Descripción</b>	<i>Cada grafica o tabla de manera independiente debe poder ampliarse a pantalla completa.</i>

<b>RFCC008</b>	<i>Exportar a Excel</i>
<b>Descripción</b>	<i>Si los datos son presentados en formato de tabla se debe implementar la posibilidad de descargar esos datos en un fichero Excel.</i>

### 5.1.2.2 No Funcionales

<b>RNFCC001</b>	<i>Código de colores</i>
<b>Descripción</b>	<i>Se debe usar un código de colores para identificar las actuaciones o visitas.</i>

### 5.1.3 Casos de Uso

#### 5.1.3.1 Especificación

<b>CUED001</b>	<i>Ver informe de conservación de carreteras</i>
<b>Actores</b>	<i>Dip. De Granada, Delegación de Granada</i>
<b>Descripción</b>	<i>Este caso de uso permite visualizar el informe gráfico completo sobre las actuaciones y visitas realizadas y</i>

	<i>registradas en el sistema para el proyecto de conservación de carreteras de la Dip. de Granada.</i>
<b>Requisito</b>	<i>RFCC001, RFCC002, RFCC003, RFCC004</i>

<b>CUED002</b>	<i>Descargar datos de actuaciones</i>
<b>Actores</b>	<i>Dip. De Granada, Delegación de Granada</i>
<b>Descripción</b>	<i>Este caso de uso permite obtener por medio de la aplicación un listado de las actuaciones registradas en el sistema en un archivo Excel para su tratamiento fuera del sistema.</i>
<b>Requisito</b>	<i>RFCC008</i>

### 5.1.3.2 Diagramas



### 5.1.4 Actores del Sistema

**Personal de dirección de la delegación de Granada.** - Es el personal que forma parte de la dirección de la delegación de CEMOSA en granada como el delegado y los directores de obra.

**Diputación de Granada.** - Personal de la diputación de granada que se encarga de realizar el seguimiento a los trabajos de conservación de carreteras.

## 5.2 Diseño

### 5.2.1 Modelos entidad-relación BBDD

Antes de plantear el modelo físico es importante definir que son y como se representan en BBDD los dos elementos que pretendemos recuperar actividades y visitas.

- Una actividad es un trabajo para un cliente que suele durar poco es de un tipo específico y pertenece a un proyecto. En el sistema se crea como una obra.

- Una visita es un tipo específico de operación dentro de un proyecto.

Con estas definiciones podemos especificar que las tablas necesarias para nuestra aplicación son las siguientes:

- **KNA1.** - Tabla maestro de clientes.
- **PROJ.** - Tabla maestro de proyectos.
- **PRPS.** - Tabla maestro de elementos pep.
- **AFVC.** - Tabla de operaciones.
- **ZTLB\_OBRAS.** - Tabla maestro de obras.

Estas tablas y sus relaciones a nivel de base de datos nos permiten construir el modelo que nos servirá como fuente de datos y se puede ver en la Ilustración 87.

Ilustración 87:ER App informe



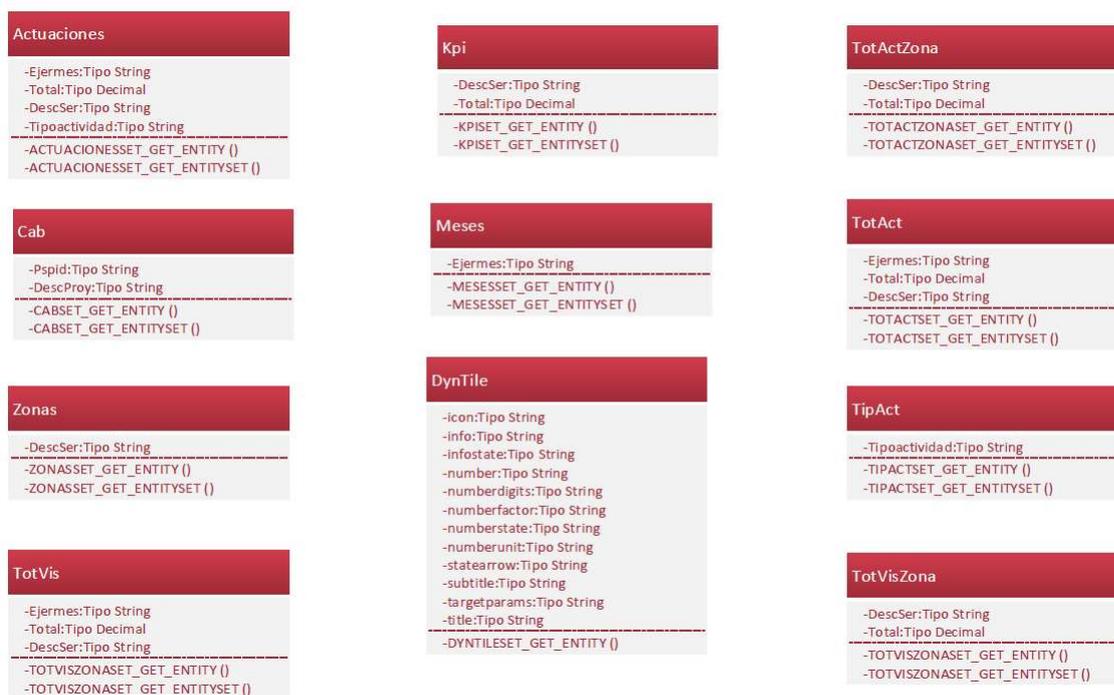
### 5.2.2 Modelado de Servicios Odata

La Ilustración 88 presenta el modelo de servicio Odata necesario para implementar la aplicación de informe. Llama la atención en el modelo que no existen relaciones entre las entidades y esto se debe a tres razones.

La primera es que la aplicación estará compuesta por una sola vista y en ella no existirá ninguna navegación, generalmente se implementan asociaciones por ejemplo para navegar al detalle de un objeto. La segunda es que por las practicas

que usamos en los desarrollos de aplicaciones SAPUI5 en el departamento como norma toda la lógica de negocio y las transformaciones de datos de cara a entregarlos a la interfaz de usuario se realizan en el servidor backend que es el proveedor de servicios. La tercera tiene que ver con las características específicas de la aplicación por ejemplo el cliente será único no necesito modelarlo será un valor fijo a la hora de recuperar los datos al igual que el proyecto.

Ilustración 88:MC informe conservación de carreteras

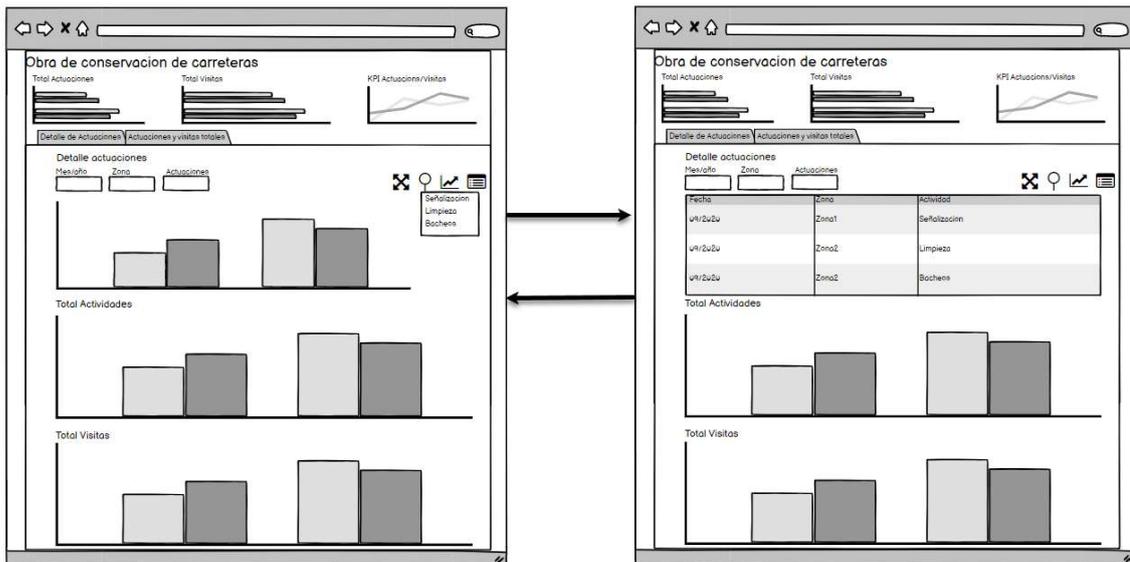


### 5.2.3 Modelado aplicaciones SAPUI5

### 5.2.4 Wireframes

La interfaz gráfica la definirá una solo vista compuesta por una página que contendrá distintas gráficas. En la Ilustración 89 se puede ver la estructura de la vista y las flechas no representan una navegación es simplemente una variación en la vista al seleccionar que los datos de actuaciones se presenten en formato de tabla.

Ilustración 89:WF App Informe conservación de carreteras



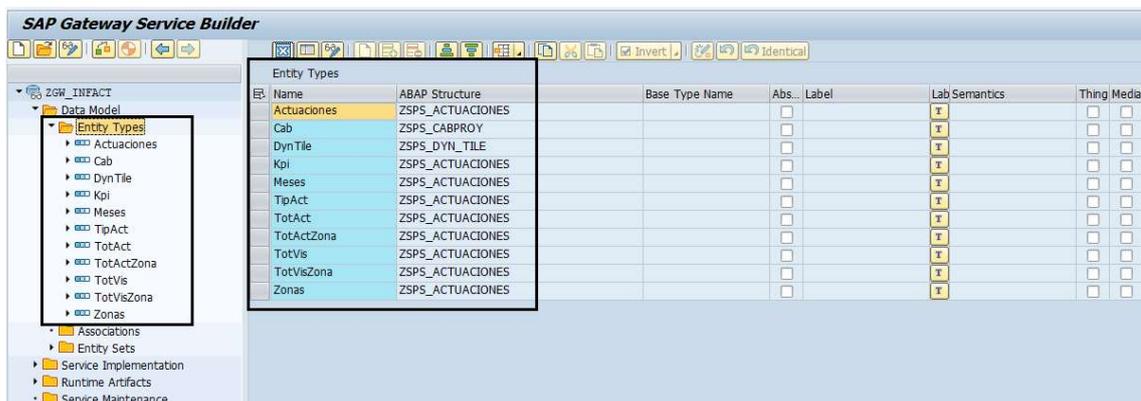
## 5.3 Implementación

### 5.3.1 Servicios implementados

#### ZGW\_INFAC T

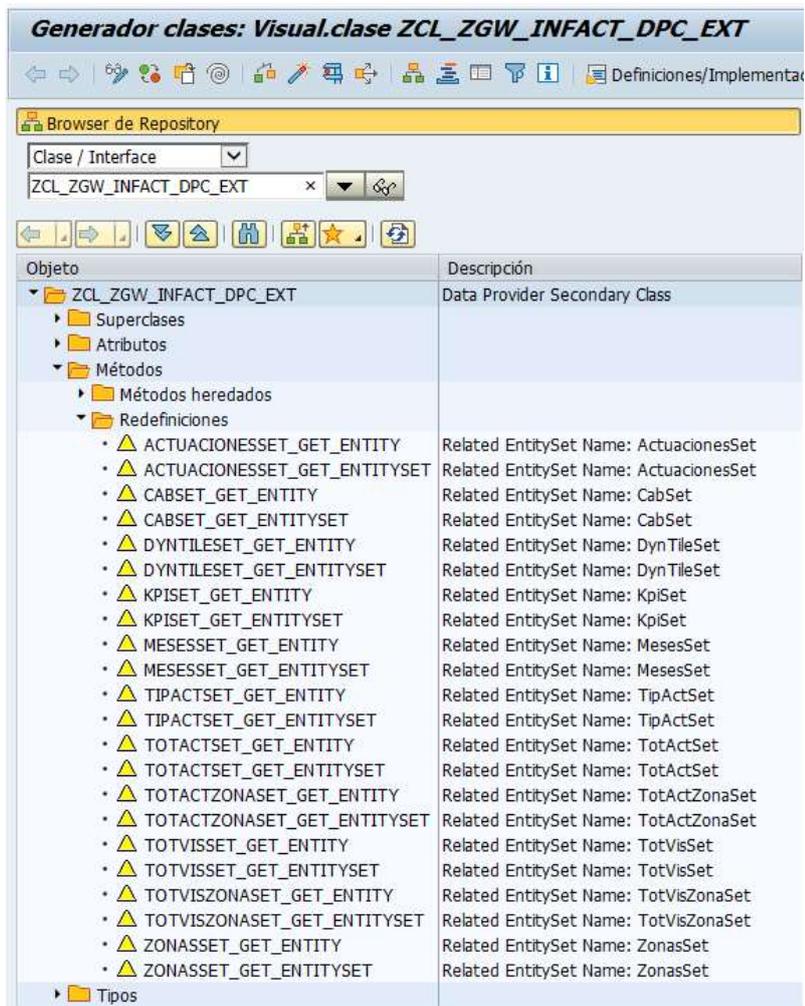
Ya que en la definición del modelo no existían asociaciones la fase de modelado se completa simplemente con el mapeo de las entidades.

Ilustración 90:Mapeo E-E ZGW\_INFAC T



En la fase de implementación se han redefinido los siguientes métodos.

Ilustración 91: Métodos redefinidos ZGW\_INFACT



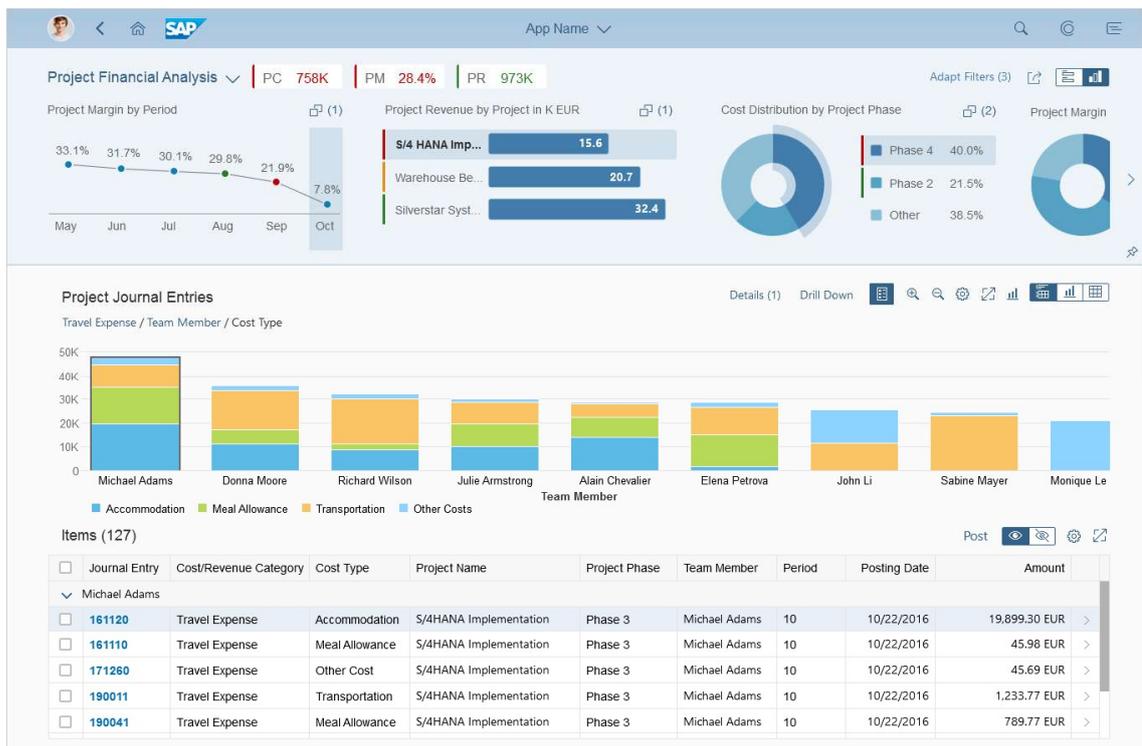
### 5.3.2 Frontend

#### 5.3.2.1 Aplicaciones SAPUI5 implementadas

##### ZBSP\_INFGRA

En cuanto al desarrollo de esta aplicación en un principio estaba planteada con el uso de un elemento Fiori denominado Analytical List Page que cumplía exactamente con los requerimientos para implementar la aplicación de informe gráfico. En la figura se puede ver una el elemento.

Ilustración 92: Analytical List Page



Cuando empezamos con la implementación nos dimos cuenta de que teníamos una restricción en las librerías para el uso de este elemento con lo que se tuvo que reconducir la implementación dando como resultado una aplicación sapui5 para la cual se implementaron las vistas y controladores que se ven en la imagen.

Ilustración 93: ZBSP\_INFGRA



La vista principal se basa en una página de secciones ObjectPageLayout ya que este control es bastante parecido en diseño a la Analytical List

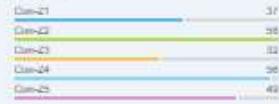
```
Principal.view.xml x
1 <mvc:View xmlns:core="sap.ui.core" xmlns:mvc="sap.ui.core.mvc" xmlns="sap.m" controllerName="Infgrn.controller.Principal"
2   xmlns:html="http://www.w3.org/1999/xhtml" xmlns:uxap="sap.uxap" xmlns:viz="sap.viz.ui5.controls" xmlns:commons="sap.suite.ui.commons"
3   xmlns:viz.feeds="sap.viz.ui5.controls.common.feeds" xmlns:viz.data="sap.viz.ui5.data" xmlns:layout="sap.ui.layout"
4   xmlns:micro="sap.suite.ui.microchart">
5   <uxap:ObjectPageLayout id="ObjectPageLayout">
6     <uxap:headerTitle>
7       <uxap:ObjectPageHeader id="objHead" isObjectTitleAlwaysVisible="true"></uxap:ObjectPageHeader>
8     </uxap:headerTitle>
9     <uxap:headerContent></uxap:headerContent>
55 <uxap:sections>
56   <uxap:ObjectPageSection title="Payroll">
57     <uxap:subSections>
58       <uxap:ObjectPageSubSection title="Detalle Actuaciones"></uxap:ObjectPageSubSection>
141     </uxap:subSections>
142   </uxap:ObjectPageSection>
143   <uxap:ObjectPageSection title="Payroll">
144     <uxap:subSections>
145       <uxap:ObjectPageSubSection title="Actuaciones y Visitas Totales"></uxap:ObjectPageSubSection>
217     </uxap:subSections>
218   </uxap:ObjectPageSection>
219 </uxap:sections>
220 </uxap:ObjectPageLayout>
221 </mvc:View>
```

SE 19/2017 SERVICIO EN MATERIA DE SEGURIDAD Y SALUD EN LAS ACTIVIDADES DE CONSERVACIÓN DE LA RED DE CARRETERAS DE LA DIPUTACIÓN PROVINCIAL DE GRANADA

Total Actuaciones: 2330



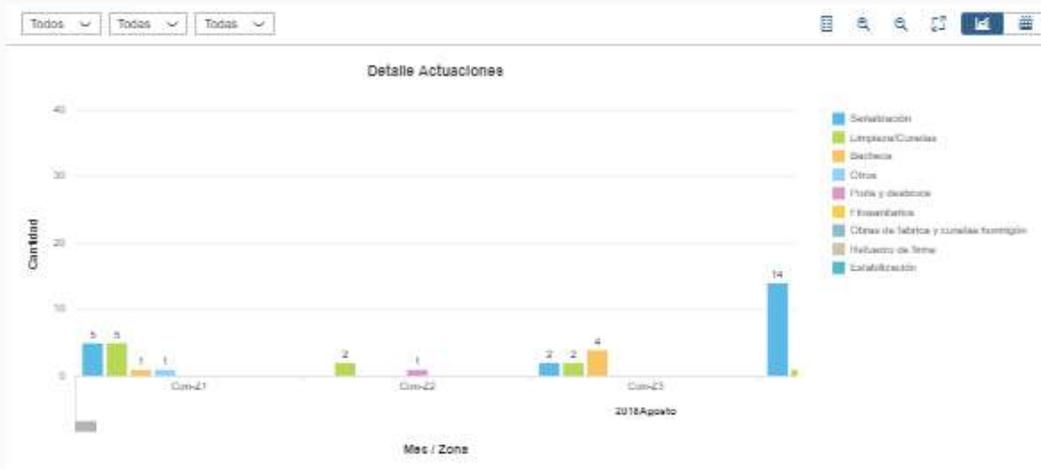
Total Visitas: 232



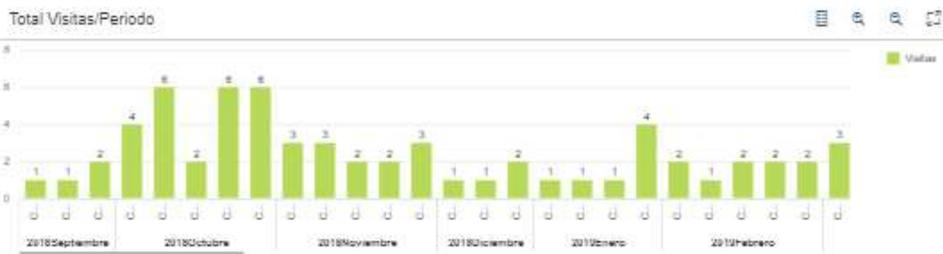
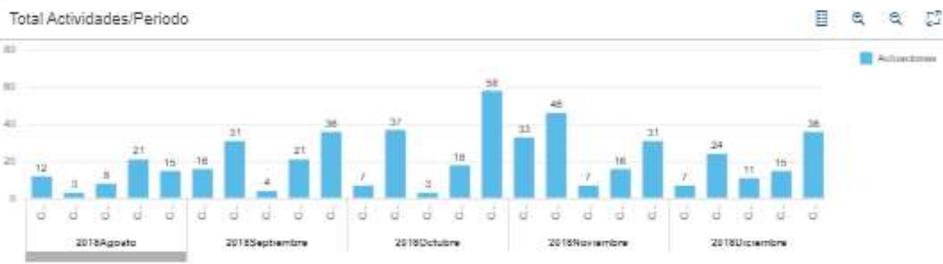
RpI-Actuaciones/Visitas



DETALLE ACTUACIONES    ACTUACIONES Y VISITAS TOTALES



ACTUACIONES Y VISITAS TOTALES



# 6

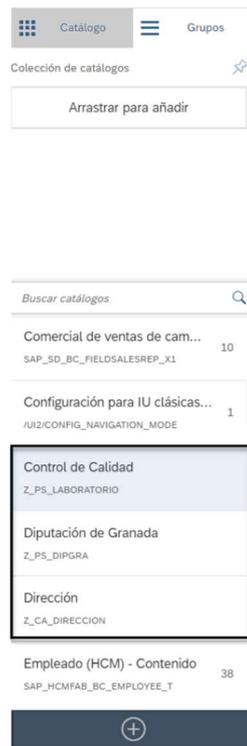
## Despliegue en SAP- Fiori Launchpad

En este capítulo vamos a describir las acciones necesarias para la configuración de la plataforma de lanzamiento " SAP-Fiori Launchpad" y la publicación de las aplicaciones SAPUI5 desarrolladas en la misma ya esta es el punto de acceso a todas las aplicaciones del sistema.

### **6.1 Catálogos**

Desde la herramienta Launchpad Designer se han creado tres catálogos distintos para para almacenar las aplicaciones sapui5 desarrolladas, se han creado tres ya que los usuarios a los que están destinadas tienen perfiles distintos.

Ilustración 94:SFLP Grupos/Catálogos

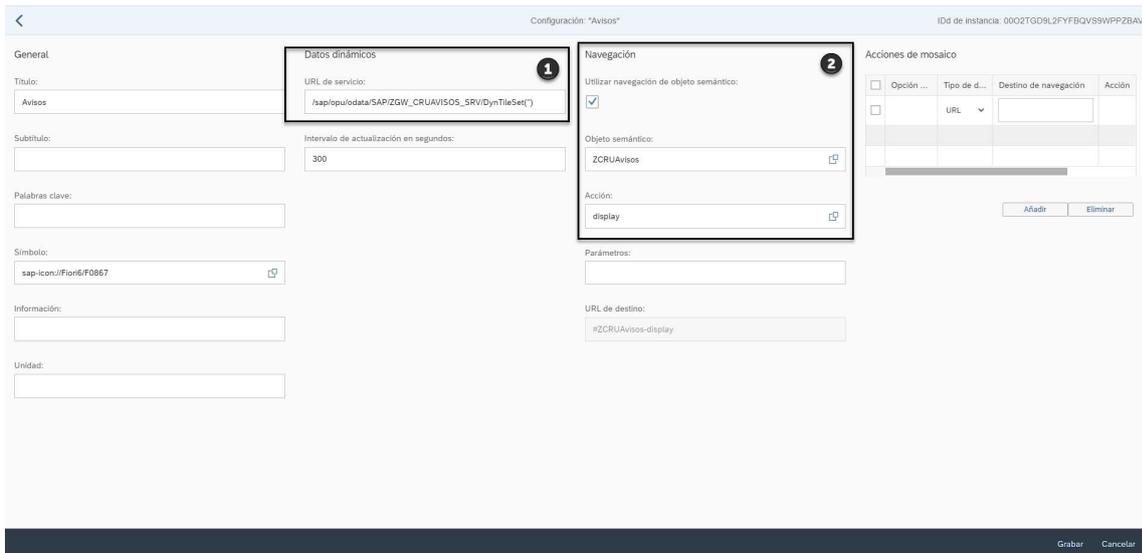


### 6.1.1 Mosaicos (Tiles)

Para incluir una aplicación en un catálogo es necesario crearle un mosaico el cual sirve como una especie de acceso directo a la aplicación además de permitir definir información sobre la aplicación en él. A la hora de crear un mosaico se debe escoger entre estático o dinámico la diferencia es que los datos que muestre el mosaico serán variables o fijos. En el caso de los dinámicos es necesario implementar un servicio web encargado de recuperar periódicamente los datos dinámicos y especificarlo en el mosaico.

Es necesario también dentro del mosaico definir el objeto semántico asociado a nuestra aplicación ya que este es el que definirá el destino del intento de navegación al presionar el mosaico.

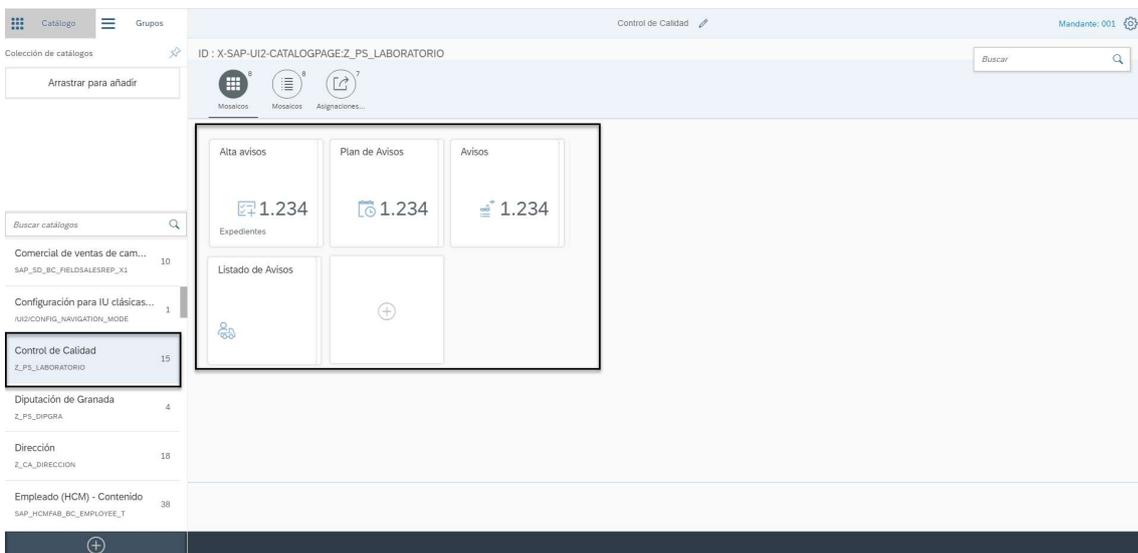
Ilustración 95:SFLP Configuración de mosaico



Se han creado y configurado los siguientes mosaicos para cada catálogo creado anteriormente:

- Catálogo Z\_PS\_LABORATORIO
  - Mosaico Alta avisos
  - Mosaico Plan de avisos
  - Mosaico Avisos
  - Mosaico Listado avisos

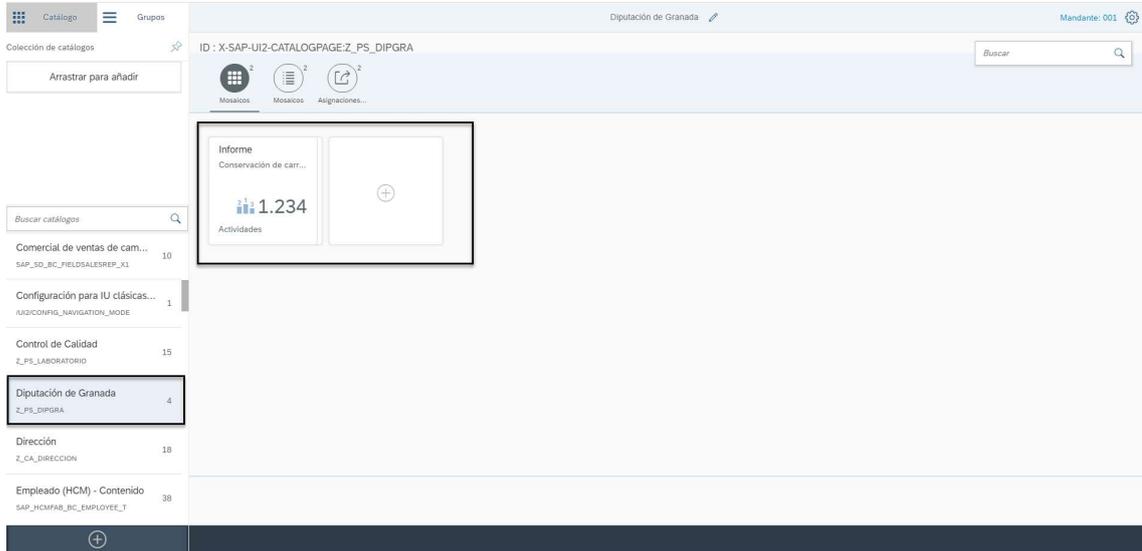
Ilustración 96:SFLP Catálogo Z\_PS\_LABORATORIO



- Catalogo Z\_PS\_DIPGRA

- Mosaico Informe

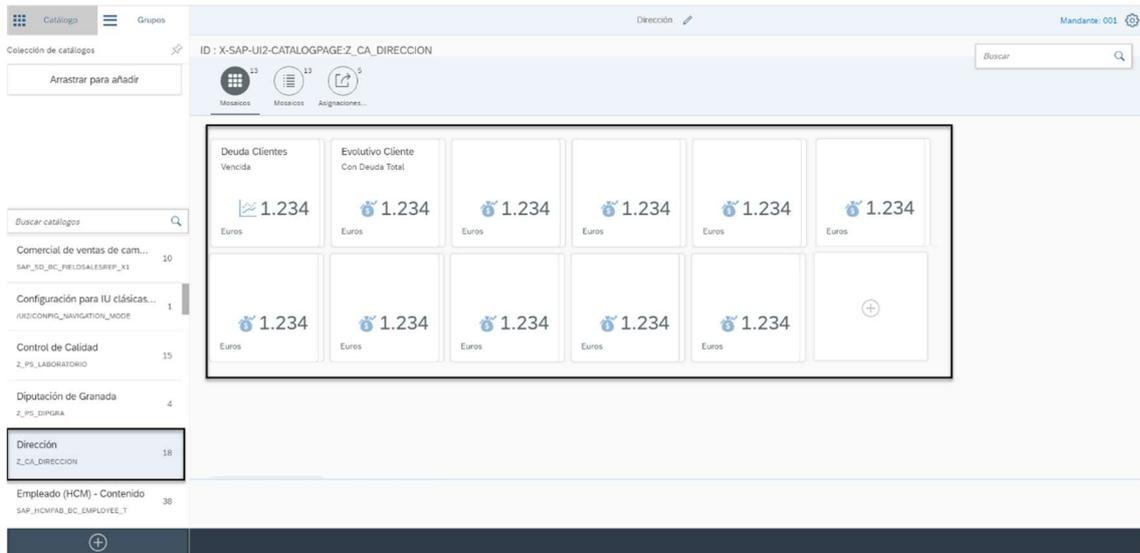
Ilustración 97: Catálogo Z\_PS\_DIPGRA



- Catalogo Z\_CA\_DIRECCION

- Mosaico Evolutivo Cliente Deuda a vencer >90 días
  - Mosaico Evolutivo Cliente Deuda a vencer de 60 a 90 días
  - Mosaico Evolutivo Cliente Deuda a vencer de 31 a 60 días
  - Mosaico Evolutivo Cliente Deuda a vencer de 1 a 30 días
  - Mosaico Evolutivo Cliente Deuda vencida >90 días
  - Mosaico Evolutivo Cliente Deuda vencida de 60 a 90 días
  - Mosaico Evolutivo Cliente Deuda vencida de 31 a 60 días
  - Mosaico Evolutivo Cliente Deuda vencida de 1 a 30 días

## Ilustración 98:SFLP Catálogo Z\_PS\_DIRECCION



### 6.1.2 Asignación de destino

Una vez se ha creado el mosaico y asignado el objeto semántico para el intento de navegación es necesario realizar el mapeado de dicho elemento a la aplicación en cuestión que se encuentra alojada en el servidor frontend. Para realizar el mapeo es necesario especificar el intento con el objeto semántico y la acción asociada y el destino con los parámetros de la aplicación:

Tipo. - El tipo de aplicación que se va a mapear ya que además de aplicaciones sapui5 se pueden mapear:

- Aplicaciones Fiori SAPUI5
- Aplicaciones Fiori configuradas en la LPD\_CUST
- Transacciones
- URL
- Web Dynpros

Título. - El título de la aplicación

URL. - La url relativa que especifica la ruta en la que se encuentra la aplicación en el servidor frontend

ID. -El id asignado a la aplicación en los parámetros de configuración (Archivo Manifest.js)

### Ilustración 99:SFLP Conf. O.S. ZCRUAvisos

Configuración: "App CRU Avisos" ID de instancia: 0002TG9L2FYFBQW06HF8A2UV

**Intención**

Objeto semántico: ZCRUAvisos

Acción: display

**Destino**

Tipo de aplicación: Aplicación Fiori SAPIUI5

Título: App CRU Avisos

URL: /sap/lob/ui5\_ui5/sap/zbsp\_cruavisos

ID: avisos

**General**

Información:

Tipos de aparato:  Escritorio  Tablet  Teléfono

Nombre	Obligatorio	Valor	Es una expresión regular	Valor estándar	Nombre ...
indicar nomt	<input type="checkbox"/>		<input type="checkbox"/>		

Permitir parámetros adicionales

Añadir Borrar

Grabar Cancelar

Se han configurado las siguientes asignaciones de destino:

### Ilustración 100:SFLP Asig. de destinos Z\_PS\_LABORATORIO

Control de Calidad Mandante: 001

ID : X-SAP-UIZ-CATALOGPAGE-Z\_PS\_LABORATORIO

Objeto semántico	Acción	Tipo de navegación	Información	Escritorio	Tablet	Teléfono	Obsoleto	Referencia...
ZPlanTemp	display	Aplicación Fiori SAPIUI5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
ZCRUAvisos	display	Aplicación Fiori SAPIUI5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
ZMUEXP	display	Aplicación Fiori SAPIUI5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		

Visualizar asignaciones de destino similares | Crear asignación de destino | Configurar | Crear referencia | Borrar | Referencia de utilización | Original

Ilustración 101:SFLP Asig. de destinos Z\_PS\_DIPGRA

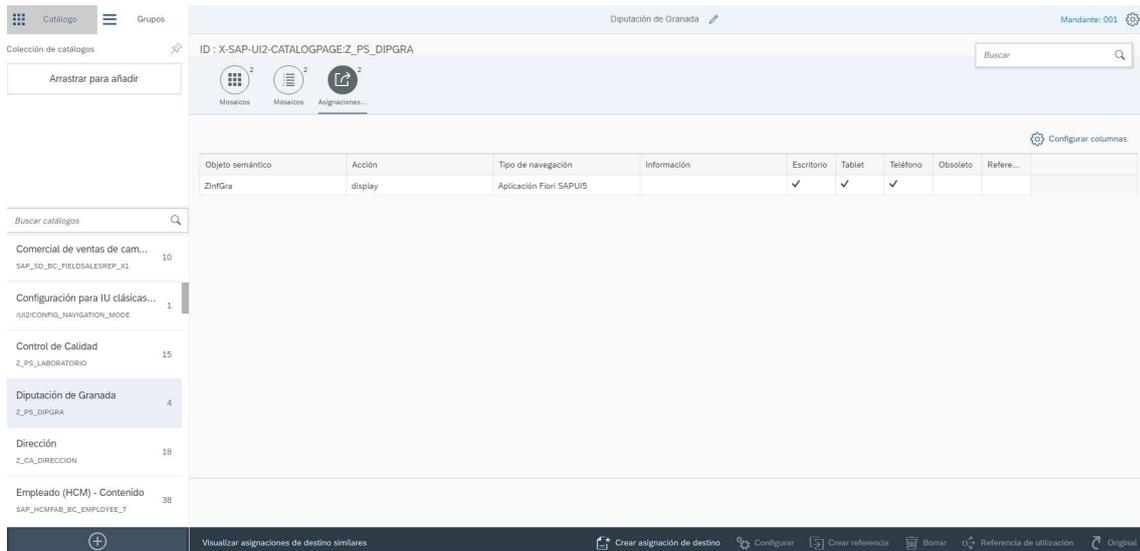
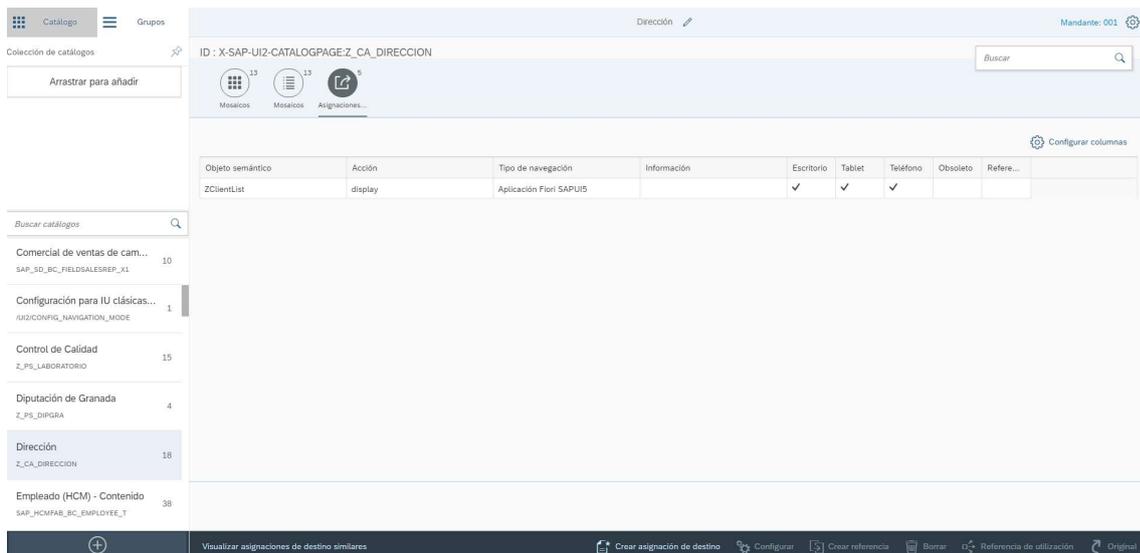


Ilustración 102:SFLP Asig. de destinos Z\_PS\_DIRECCION



## 6.2 Grupos

Desde el Launchpad Designer también es posible crear grupos. Los grupos representan un modo adicional de organizar los Mosaicos, a diferencia de los catálogos esta forma de organización está disponible para el usuario final y lo dota de la capacidad de personalizar su SAP Fiori Launchpad.

Los grupos también se diferencian de los catálogos en que un grupo permite agrupar aplicaciones procedentes de diferentes catálogos. Se han creado y configurado los siguientes grupos:

### Ilustración 103:SFLP Grp.ZLAB

**Control de Calidad**  
ZLAB

Visualizar como mosaicos

Control de Calidad Expedientes 1.234	Plan de Avisos 1.234	Avisos 1.234	Listado de Avisos	Gestión de muestras 1.234	+
--	-------------------------	-----------------	-------------------	------------------------------	---

Visualizar como enlaces

+
---

**Grupos:**

- Control de Calidad. ZLAB (5)
- Data Aging (SAP\_BASIS\_BCG\_ILM\_DAG) (4)
- Deuda por periodos (ZDEUDA) (9)
- Dictionary ABAP (SAP\_BASIS\_BCG\_DIC) (1)
- Diputación de Granada (ZDIPGRA) (1)
- Diputación de Granada Direcci... (ZDIPGRADIR) (1)
- Diputación de Jaén (ZDIPJAE) (1)
- Dirección (ZDIR) (4)
- Editor de KPIs (IUI2SAP\_KPIMOD\_TCG\_S) (8)
- Empleado (HCM) (6)

### Ilustración 104:SFLP Grp. ZDEUDA

**Deuda por periodos**  
ZDEUDA

Visualizar como mosaicos

1.234 Euros	1.234 Euros	1.234 Euros	1.234 Euros	1.234 Euros	1.234 Euros
1.234 Euros	1.234 Euros	Evolutivo Cliente Con Deuda Total 1.234 Euros	+		

Visualizar como enlaces

+
---

**Grupos:**

- Deuda por periodos (ZDEUDA) (9)
- Dictionary ABAP (SAP\_BASIS\_BCG\_DIC) (1)
- Diputación de Granada (ZDIPGRA) (1)
- Diputación de Granada Direcci... (ZDIPGRADIR) (1)
- Diputación de Jaén (ZDIPJAE) (1)
- Dirección (ZDIR) (4)
- Editor de KPIs (IUI2SAP\_KPIMOD\_TCG\_S) (8)
- Empleado (HCM) (SAP\_HCM\_BCG\_Employee\_X1) (4)
- Empleado (HCM) (SAP\_HCMFAB\_BCG\_EMPLOYEE\_T) (19)
- Empleado (viaje) (1)

# Ilustración 105:SFLP Grp. ZDIPGRADIR

The screenshot displays the SAP Fiori 'Grupos' application interface. On the left, a navigation pane lists various groups with their counts:

- Diputación de Granada Direcci... (1)
- Diputación de Jaén (1)
- Dirección (4)
- Editor de KPIs (8)
- Empleado (HCM) (4)
- Empleado (HCM) (19)
- Empleado (Viaje) (1)
- Encargado de transporte (1)
- Extensibilidad (7)
- Gestión de procesos empres... (15)

The main content area is titled 'ID : ZDIPGRADIR' and offers two visualization options:

- Visualizar como mosaicos:** This view displays a dashboard with an 'Informe' card for 'Conservación de carr...' showing a value of 1.234, and a '+' button to add more widgets.
- Visualizar como enlaces:** This view displays a single '+' button to add links.

The top navigation bar includes 'Catálogo', 'Grupos', and 'Mandante: 001'.

# 7

## Gestión de Roles

Una parte importante en el desarrollo e implantación de aplicaciones SAP-Fiori es la elaboración de perfiles de usuario para gestionar los accesos a las distintas aplicaciones. Por la característica intrínseca (Orientada a roles) del desarrollo con esta tecnología es necesario crear roles tanto en el backend para los servicios Odata implementados como en frontend para el acceso a los catálogos y grupos de aplicaciones.

Por la arquitectura sobre la que estamos trabajando al tener dos servidores distintos para backend y frontend también son necesarios roles que permitan la comunicación entre estos dos entornos para garantizar la seguridad sobre las comunicaciones. La gestión de roles se realiza mediante la herramienta (Transacción) PFCG incluida en SAP-ERP para este fin.

### 7.1 Roles Backend

- **Z\_ACCESO\_RFC\_PRODUCTIVO:** Rol que permite la combinación segura entre los sistemas backend y frontend. Todos los perfiles deben tener este rol ya que es transversal.

### 7.2 Roles Frontend

- **ZUI5\_CATALOGO\_DIPGRA:** Rol que da acceso a las aplicaciones del catálogo de dirección de la delegación de Granada.
- **ZUI5\_CATALOGO\_DIRECCION:** Rol que da acceso a las aplicaciones del catálogo de Dirección.

- **ZUI5\_CATALOGO\_LAB:** Rol que da acceso a las aplicaciones del catálogo de Laboratorio.
- **ZUI5\_LAUNCHPAD\_ZDEUDA:** Rol que da acceso al grupo de aplicaciones de deuda.
- **ZUI5\_LAUNCHPAD\_ZDIPGRADIR:** Rol que da acceso al grupo de aplicaciones de la dirección de Granada.
- **ZUI5\_LAUNCHPAD\_ZLAB:** Rol que da acceso al grupo de aplicaciones de la dirección de Laboratorio.
- **ZUI5\_CRUAVISO\_SRV:** Rol permite consumir datos del servicio implementado para visualizar avisos.
- **ZUI5\_CALPLAN\_SRV:** Rol permite consumir datos del servicio para el calendario de tareas.
- **ZUI5\_ALTAVISO\_SRV:** Rol permite consumir datos del servicio de alta de aviso.
- **ZUI5\_GESAVISOS\_SRV:** Rol permite consumir datos del servicio gestión de avisos.
- **ZUI5\_INFACCT\_SRV:** Rol permite consumir datos del servicio de informe de conservación de carreteras.
- **ZUI5\_GESDEUDACL\_SRV:** Rol permite consumir datos del servicio de datos de deuda de cliente.

### 7.3 Perfiles

Ilustración 106: Perfil Laboratorio

Status	Rol	Ti.	Fecha de inicio	Fe.fin modifc.	Descripción breve del rol	Indir.
<input checked="" type="checkbox"/>	ZUI5_ALTAVISO_SRV		29.07.2019	31.12.9999	Acceso al servicio de la app ALTAVISO	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ZUI5_CALPLAN_SRV		29.07.2019	31.12.9999	Acceso al servicio de la app plan avisos	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ZUI5_CATALOGO_LAB		29.07.2019	31.12.9999	Muestra el catálogo de laboratorio	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ZUI5_CRUAVISO_SRV		29.07.2019	31.12.9999	Acceso al servicio de la app gestión de avisos	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ZUI5_GESAVISO_SRV		29.07.2019	31.12.9999	Acceso al servicio de la app GESAVISOS	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ZUI5_LAUNCHPAD_ZLAB		29.07.2019	31.12.9999	Muestra el grupo de Apps Fiori para Laboratorio	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Z_SAP_UI2_USER_700		29.07.2019	31.12.9999	Rol de usuario para servicios web simples de UI2 7...	<input type="checkbox"/>

Ilustración 107:Perfil Dirección

**Actualizar usuarios**

Usuario

Modificado por  02.06.2020 17:11:18 Status

Documentación Direc. Datos logon SNC Valores fijos Parám. Roles Perfiles Grupos Personalización Dat.licenc.

Usuario de referencia

Asignación de funciones

Status	Rol	Ti...	Fecha de inicio	Fe.fin modif.	Descripción breve del rol	Indir.
<input checked="" type="checkbox"/>	Z_SAP_UI2_USER_700		29.07.2019	31.12.9999	Rol de usuario para servicios web simples de UI2 7...	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ZUI5_CATALOGO_DIRECCION		21.09.2020	31.12.9999	Muestra el catálogo de laboratorio	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ZUI5_LAUNCHPAD_ZDIR		21.09.2020	31.12.9999	Muestra el grupo de Apps Fiori para Dirección	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ZUI5_GESDEUDAACL_SRV		21.09.2020	31.12.9999	Acceso al servicio de la app DEUDA CLIENTES	<input type="checkbox"/>

Ilustración 108:Perfil Dirección de Granada

**Actualizar usuarios**

Usuario

Modificado por  02.06.2020 17:11:18 Status

Documentación Direc. Datos logon SNC Valores fijos Parám. Roles Perfiles Grupos Personalización Dat.licenc.

Usuario de referencia

Asignación de funciones

Status	Rol	Ti...	Fecha de inicio	Fe.fin modif.	Descripción breve del rol	Indir.
<input checked="" type="checkbox"/>	Z_SAP_UI2_USER_700		29.07.2019	31.12.9999	Rol de usuario para servicios web simples de UI2 7...	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ZUI5_CATALOGO_DIPGRA		21.09.2020	31.12.9999	Muestra el catálogo de Apps Fiori para Dip.Granad...	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ZUI5_LAUNCHPAD_ZDIPGRADIR		21.09.2020	31.12.9999	Grupo dip. granada dirección	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ZUI5_INFACNT_SRV		21.09.2020	31.12.9999	Acceso al servicio de la app Informe gráfico Granada	<input type="checkbox"/>

# 8

## Pruebas

En este capítulo se especificarán las distintas pruebas que se han realizado a lo largo del desarrollo e implementación de las aplicaciones SAPUI5. Las pruebas que se han realizado se pueden dividir en tres tipos.

- **Unitarias.** - Estas pruebas tienen el objetivo de comprobar el funcionamiento correcto de los módulos de función, performs, métodos e incluso fragmentos de código aislado que se fueron implementando. Además de verificar lo anterior, como efecto colateral estas pruebas nos sirven en gran medida para detectar y corregir errores en fases tempranas del desarrollo.
- **Funcionales.** - El objetivo de estas pruebas es el de ver si las aplicaciones desarrolladas cumplen a nivel funcional con los requisitos establecidos al inicio del desarrollo. Este tipo de pruebas las solemos realizar con un grupo de prueba de usuarios finales no solamente para detectar algún fallo sino para recibir el feedback que muchas veces se transforman en solicitudes de mejoras de las aplicaciones.
- **Casos de prueba.** - Tienen el mismo objetivo que las pruebas funcionales, es decir, validar el nivel de cumplimiento de los requisitos establecidos. Aunque éstos son más específicos ya que describen unas condiciones y variables específicas bajo las cuales se tiene que llevar a cabo una prueba.

## 8.1 Pruebas unitarias

### 8.1.1 Servicios Odata

Se han realizado pruebas unitarias de cada método redefinido en los servicios implementado en el desarrollo de este TFG. Estas pruebas unitarias nos ayudan a detectar errores antes de pasar a la implementación de las aplicaciones SAPUI5, consumidoras de estos servicios.

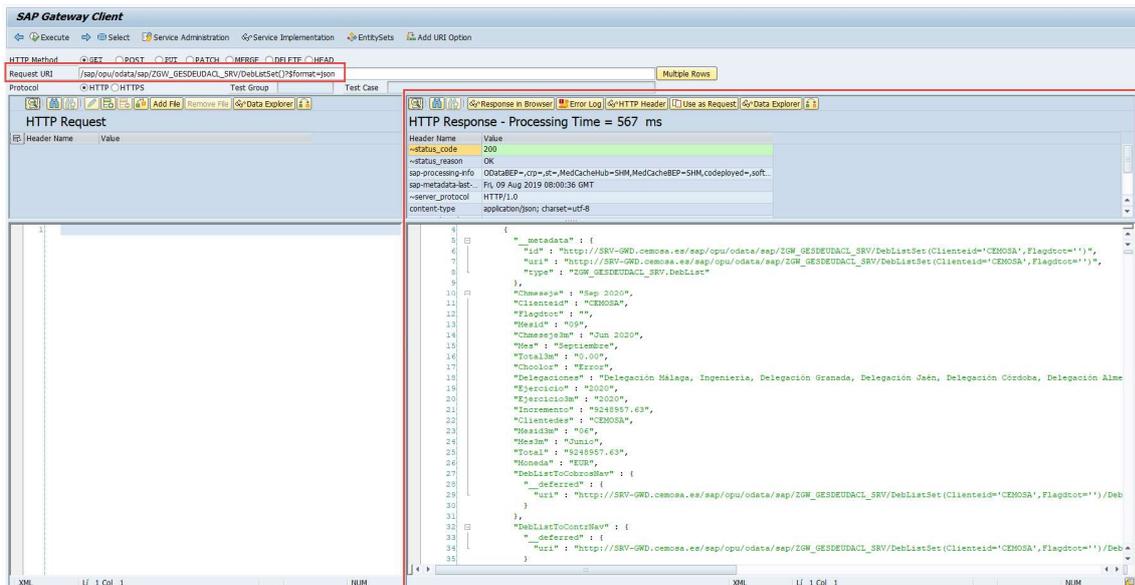
Para realizar esta prueba tenemos disponible una herramienta dentro de SAP-ERP mediante la transacción /IWFND/MAINT\_SERVICE.

Ilustración 109:TCODE /IWFND/MAINT\_SERVICE

Tipo	Nombre serv.zénico	Descripción del servicio	Nombre del servicio externo	Ár.nom.	Ako.	Est.simple	Modo tratamiento
BEP	REP1_SALESORDERDETAILSO_CDS	1	REP1_SALESORDERDETAILSO_CDS			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	REP1_SALESORDERPRODUCTQUERY_CDS	1	REP1_SALESORDERPRODUCTQUERY_CDS			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	SADL_V_EXP_QUERY_CDS	1	SADL_V_EXP_QUERY_CDS			<input type="checkbox"/>	Basado en enluta.
BEP	SADL_V_SALESORDER_BO_CDS	1	SADL_V_SALESORDER_BO_CDS			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	SEPM_HANA_EXT_PAL_ODATA_SRV	1	SEPM_HANA_EXT_PAL_ODATA_SRV			<input checked="" type="checkbox"/>	No soportados Basado en enluta.
BEP	/IWFND/SUBSCRIPTIONMANAGEMENT	2	SUBSCRIPTIONMANAGEMENT	/IWFND/		<input type="checkbox"/>	Basado en enluta.
BEP	TRANSPORT	1	TRANSPORT	/IWFND/		<input type="checkbox"/>	No soportados Basado en enluta.
/IWFND/USAGEEXTRACTOR	1	Meterng Usage Extractor	USAGEEXTRACTOR	/IWFND/		<input type="checkbox"/>	Basado en enluta.
/IWFND/US_USER_SERVICE	1	Information Worker - User Service	USERSERVICE	/IWFND/		<input type="checkbox"/>	Basado en enluta.
BEP	UIOR_ADAPT_UII_SRV	1	UIOR_ADAPT_UII_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_ALTAVI_SRV	1	ZGW_ALTAVI_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_BIMINTERFACE_SRV	1	ZGW_BIMINTERFACE_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_CALENDAROPLAN_SRV	1	ZGW_CALENDAROPLAN_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_CARTERA_SRV	1	ZGW_CARTERA_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_CEMOSA_01_SRV	1	ZGW_CEMOSA_01_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_CEMOSA_02_SRV	1	ZGW_CEMOSA_02_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_CENAVISO_SRV	1	ZGW_CENAVISO_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_CENAVISOS_SRV	1	ZGW_CENAVISOS_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_CENAVISOS_SRV	1	ZGW_CENAVISOS_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_GESAVISOS_SRV	1	ZGW_GESAVISOS_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_GESAVISOS_SRV	1	ZGW_GESAVISOS_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_GESAVISOS_SRV	1	ZGW_GESAVISOS_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_GESTODC_SRV	1	ZGW_GESTODC_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_GESTODC_SRV	1	ZGW_GESTODC_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_GESTORA_SRV	1	ZGW_GESTORA_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_INFACI_SRV	1	ZGW_INFACI_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_MODELOSBM_SRV	1	ZGW_MODELOSBM_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_MUEXP_SRV	1	ZGW_MUEXP_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_PUNERA_SRV	1	ZGW_PUNERA_SRV			<input type="checkbox"/>	No soportados Basado en enluta.
BEP	ZGW_TOMMUESTRAS_SRV	1	ZGW_TOMMUESTRAS_SRV			<input type="checkbox"/>	No soportados Basado en enluta.

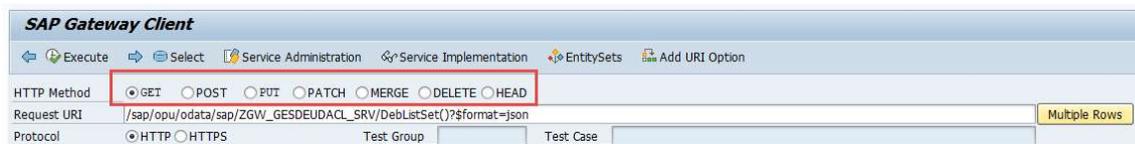
Donde podemos simular las peticiones que van a realizar las aplicaciones introduciendo la URI y ver los resultados de dicha petición.

Ilustración 110: Prueba peticiones ZGW\_GESDEUDACL



Podemos realizar las pruebas de todas las peticiones que tenga implementado el servicio POST, PUT, PATCH, MERGE, DELETE y HEAD.

Ilustración 111: Tipo de petición



### 8.1.2 Aplicaciones SAPUI5

Para realizar pruebas funcionales la herramienta de desarrollo SAP WebId permite configurar ejecuciones de las aplicaciones con datos ficticios "mock".

Ilustración 112: Conf. mock data

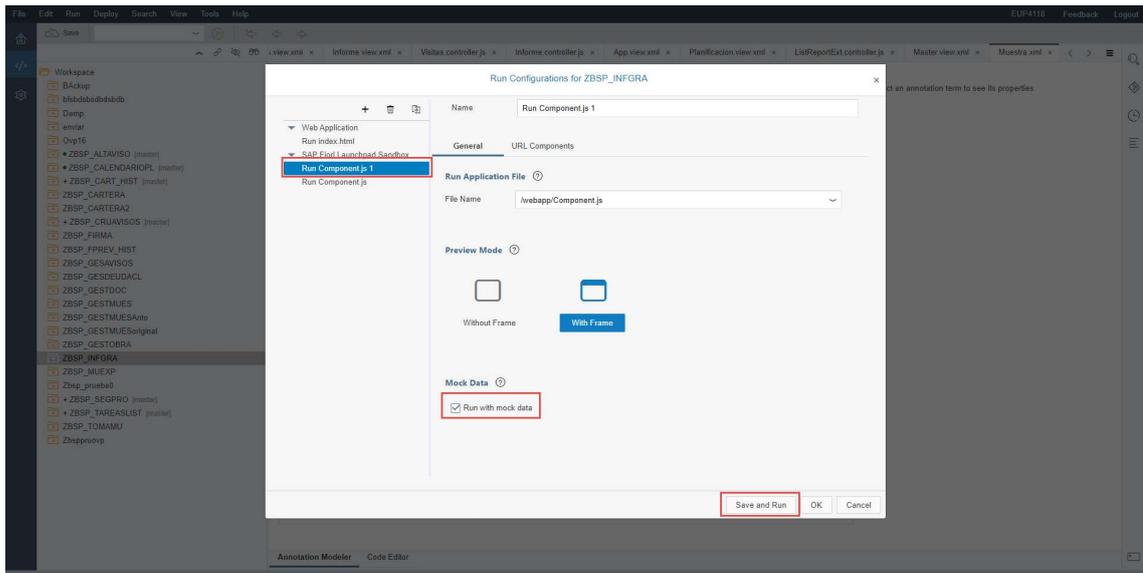
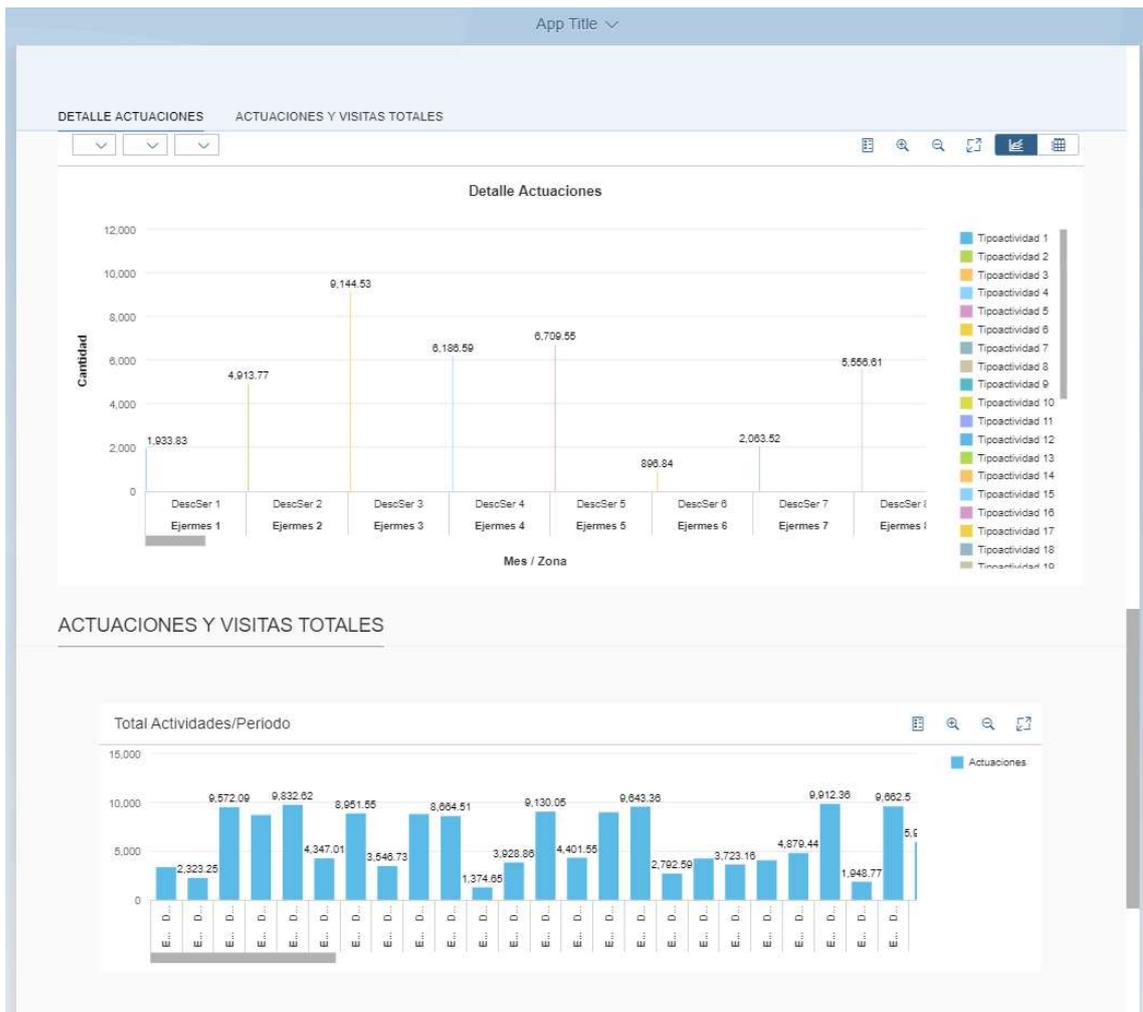


Ilustración 113: Prueba mock App Inf. conservación de carreteras



## **8.2 Pruebas funcionales.**

Estas pruebas se han realizado llevando a cabo la implantación de las aplicaciones en el entorno intermedio de integración y con un grupo de usuarios de prueba especializado.

## **8.3 Casos de prueba**

Si bien en el anterior apartado se describen las pruebas que se van realizando durante la implementación, al finalizar esta se han verificado todos los siguientes casos de prueba y se han obtenido los siguientes resultados. Son pruebas funcionales del sistema con datos y usuarios reales. Es el paso previo al despliegue en producción.

Ilustración 114: CP Sis. Gestión Avisos

CP.ID	Objetivo	Contexto	Prerequisito	Pasos	Datos de Entrada	Salida esperada	Salida	Estado	Ref. RF
CPGA002	Dar de alta un aviso	App alta de aviso		1. Seleccionar expediente 2. Introducir datos de aviso 3. Pulsar guardar	Datos de aviso introducidos	El aviso se guarda en BBDD	El aviso se guarda en BBDD	OK	RFGA002 RFGA013
CPGA003	Visualizar tomas / actividades aviso	App Avisos	Aviso existente	1. Seleccionar aviso 2. Seleccionar toma/Actividad		Ver datos de tomas/actividades	Ver datos de tomas/actividades	OK	RFGA003 RFGA016
CPGA004	Modificar tomas / actividades aviso	App Listado avisos / Plan avisos	Aviso existente	1. Seleccionar aviso 2. Seleccionar toma/Actividad 3. Modificar datos 4. Pulsar guardar	Datos modificados	Toma/Actividad modificada	Toma/Actividad modificada	OK	RFGA004
CPGA005	Anular tomas / actividades aviso	App Listado avisos / Plan avisos	Aviso existente	1. Seleccionar aviso 2. Seleccionar toma/Actividad 3. Pulsar anular 4. introducir motivo 5. Pulsar ok	Motivo	Toma/Actividad Anulada	Toma/Actividad anulada	OK	RFGA005
CPGA007	Notificar	App avisos	Aviso existente ,Deben existir interlocutores	1. Seleccionar aviso 2. Pulsar icono interlocutores 3. Seleccionar interlocutores 4. Presionar botón notificar	Interlocutores	Interlocutores notificados	Interlocutores notificados	OK	RFGA006 RFGA007
CPGA008	Ver agenda	App pan avisos		1. Navegar en el calendario		Ver carga de trabajo	ver carga de trabajo	OK	RFGA008
CPGA009	Buscar avisos	App Listado de avisos / App Avisos		1. Introducir filtros 2. Pulsar ir / enter		Ver avisos / No existen	Ver avisos / No existen	OK	RFGA009 RFGA010
CPGA011	Modificar varias tomas / actividades a la vez	App Listado avisos		1. Seleccionar tomas/Actividades 3. Modificar datos 4. Pulsar guardar	Datos modificados	Tomas/Actividades modificada	Tomas/Actividades modificadas	OK	RFGA011
CPGA012	Exportar listado de avisos	App Listado avisos		1. Pulsar boton exportar	Listado	Archivo excel con el listado	Archivo excel con el listado	OK	RFGA012
CPGA015	Cambiar disposicion listado	App Listado avisos		1. Modificar disposicion de las columnas del listado	Listado	Disposicion modificada	Disposicion modificada	OK	RFGA015
CPGA016	Navegar entre aplicaciones	App Listado avisos/App plan aviso / App Avisos		1. Seleccionar tomas/Actividades 3. Pulsar navegar a detalle	Toma/Actividad	Mostrar detalle de aviso en App avisos	Mostrar detalle de aviso en App avisos	OK	RFGA016

Ilustración 115:CP App Evolutivo cliente

CP.ID	Objetivo	Contexto	Prerequisito	Pasos	Datos de Ent	Salida esper	Salida	Estado	Ref. RF
CPED001	Listar clientes	App Evolutivo cliente	El cliente debe tener deuda	1.Entrar en App		Listado de Clientes con deuda	Listado de Clientes con deuda	Ok	RFDE001
CPED002	Buscar cliente	App Evolutivo cliente	El cliente debe existir y tener deuda	1.usar barra de búsqueda 2.presionar enter	Token de búsqueda	Cliente/s/ no existe	Cliente/s/ no existe	OK	RFDE002
CPED003	Visualizar evolucion de contratación	App Evolutivo cliente		1.Seleccionar cliente 2.seleccionar grafica/tabla contratación		Ver evolutivo de contratación del cliente	Ver evolutivo de contratación del cliente	OK	RFDE003 RFED007
CPDE004	Visualizar evolucion de ventas	App Evolutivo cliente		1.Seleccionar cliente 2.seleccionar grafica/tabla ventas		Ver evolutivo de ventas del cliente	Ver evolutivo de ventas del cliente	OK	RFDE004 RFED007
CPDE005	Visualizar evolucion de deuda	App Evolutivo cliente		1.Seleccionar cliente 2.seleccionar grafica/tabla deuda		Ver evolutivo de deuda del cliente	Ver evolutivo de deuda del cliente	OK	RFDE005 RFED007
CPDE006	Visualizar evolucion de cobro	App Evolutivo cliente		1.Seleccionar cliente 2.seleccionar grafica/tabla cobro		Ver evolutivo de cobro del cliente	Ver evolutivo de cobro del cliente	OK	RFDE006 RFED007
CPED010	Mosaicos variantes por periodo	Variantes App Evolutivo cliente		1.Entrar en cada variante de la aplicación		Ver evolutivo de deuda del cliente por periodos especificos	Ver evolutivo de deuda del cliente por periodos especificos	OK	RFED010

Ilustración 116:CP App Informe

CP.ID	Objetivo	Contexto	Prerequisito	Pasos	Datos de Ent	Salida esper	Salida	Estado	Ref. RF
CPCC001	Visualizar informe	App Informe (Conservación de carreteras)		1. Entrar en App		Ver todas las graficas del informe	Ver todas las graficas del informe	Ok	RFCC001 RFCC002 RFCC003 RFCC004
CPCC005	Filtrar actuaciones por periodo / tipo / zona	App Informe (Conservación de carreteras)		1. Introducir filtros en contenedor de gráfica /tabla de actuaciones		Filtros aplicados en grafica/tabla	Filtros aplicados en grafica/tabla	Ok	RFCC005 RFCC006
CPCC007	Ver en pantalla completa las graficas de actuaciones y visitas y las de totales	App Informe (Conservación de carreteras)		1. Pulsar sobre el boton pantalla completa de la gráfica que se desea ver		Visualizar la gráfica en pantalla completa	Visualizar la gráfica en pantalla completa	Ok	RFCC007
CPCC008	Exportar listado de actuaciones	App Informe (Conservación de carreteras)		1. Pulsar boton exportar	Listado	Archivo excel con el listado	Archivo excel con el listado	OK	RFCC008

Estas las pruebas de estos casos se han realizado en el sistema de integración y desarrollo respectivamente.

# 9

## Conclusiones

Una vez finalizado el proceso de desarrollo e implantación de las aplicaciones SAPUI5, la formación a personal y el seguimiento de incidencias se puede dar por conseguido el objetivo principal de este TFG, desarrollar e implantar un conjunto de aplicaciones SAP-Fiori en el entorno productivo de la empresa CEMOSA. Por la naturaleza de este TFG y la definición misma de SAP-Fiori con el desarrollo se ha aportado a la empresa valor añadido aumentando la productividad automatizando y homogenizando procesos no solo con una interfaz de usuario, sino con una experiencia de usuario que es un concepto más amplio cuyas características se pueden resumir en:

- Una experiencia adecuada al rol laboral del usuario, y personalizable por él mismo.
- Una experiencia simple, una aplicación para cada tarea, no una gran aplicación que permite ejecutar todas las tareas posibles.
- Una experiencia coherente, la misma para todas las aplicaciones Fiori.
- Una experiencia agradable, con un diseño estudiado y optimizado para los usuarios.
- Una experiencia adaptativa, donde la misma aplicación se adapta al dispositivo, a su tamaño de pantalla, en forma inteligente.

Todo esto sumado a los beneficios del contexto pues el desarrollo al contar con un sistema SAP-ERP no ha supuesto una inversión económica alta. Por las propias características y las recomendaciones seguidas durante el desarrollo contamos con un sistema dinámico y escalable en el cual el tiempo de desarrollo de una aplicación productiva con características multidispositivo y hecha a medida tampoco implican un coste temporal alto.

## **9.1 Mantenimiento y Evolución**

Luego de la implantación y los primeros meses de incidencias se seguirá dando soporte y atendiendo a las solicitudes de cambios o inclusión de nuevas funcionalidades por parte de los usuarios.

Continuando con el flujo de procesos del área de control de materiales pendientes de transformar los trabajos futuros sobre esta tecnología se enfocará en la toma de muestras y emisión de albaranes.

## **9.2 Valoración**

El proyecto me ha supuesto en cuanto a aprendizaje y desarrollo profesional una oportunidad que valoró muy positivamente. Aunque ha requerido un esfuerzo extra ya que en un periodo corto de tiempo han sido muchísimos conceptos que entender y asimilar, y muchas herramientas que se ha tenido que aprender a usar.

Me parece que las tecnologías usadas en este TFG a pesar de no tener mucho tiempo de vida, pues es en el 2015 cuando se lanzan, han tenido una evolución meteórica y son una apuesta a futuro solida de la compañía SAP. Actualmente SAP – Fiori ya está en su versión tres y el protocolo Odata en su versión 4, el catálogo de aplicaciones SAP Fiori estándar ha crecido bastante contando en la actualidad con más de 300 aplicaciones para cubrir procesos de distintos módulos de SAP.

# 10

## Referencias

Miroslav Antolovic; Getting Started with SAPUI5, 1ª edición, 2014

Anil Bavaraju; SAP Fiori Implementation and Development, 2ª edición, 2017

Carsten Bönnen, Volker Drees, André Fischer, Ludwig Heinz, Karsten Strothmann; SAP Gateway and OData, 3ª edición, 2019

Christiane Goebels, Denise Nepraunig, Thilo Seidel; SAPUI5, 1ª edición, 2016

Steve Guo; SAP Fiori Launchpad, 1ª edición, 2019

Stefan Haas, Bince Mathew; ABAP Programming Model for SAP Fiori, 1ª edición, 2019

SAP SE, <https://sapui5.hana.ondemand.com/>

SAP Fiori Design Guidelines, <https://experience.sap.com/fiori-design-web/>

SAPUI5 SDK (ondemand.com), <https://sapui5.hana.ondemand.com/>

# Apéndice A

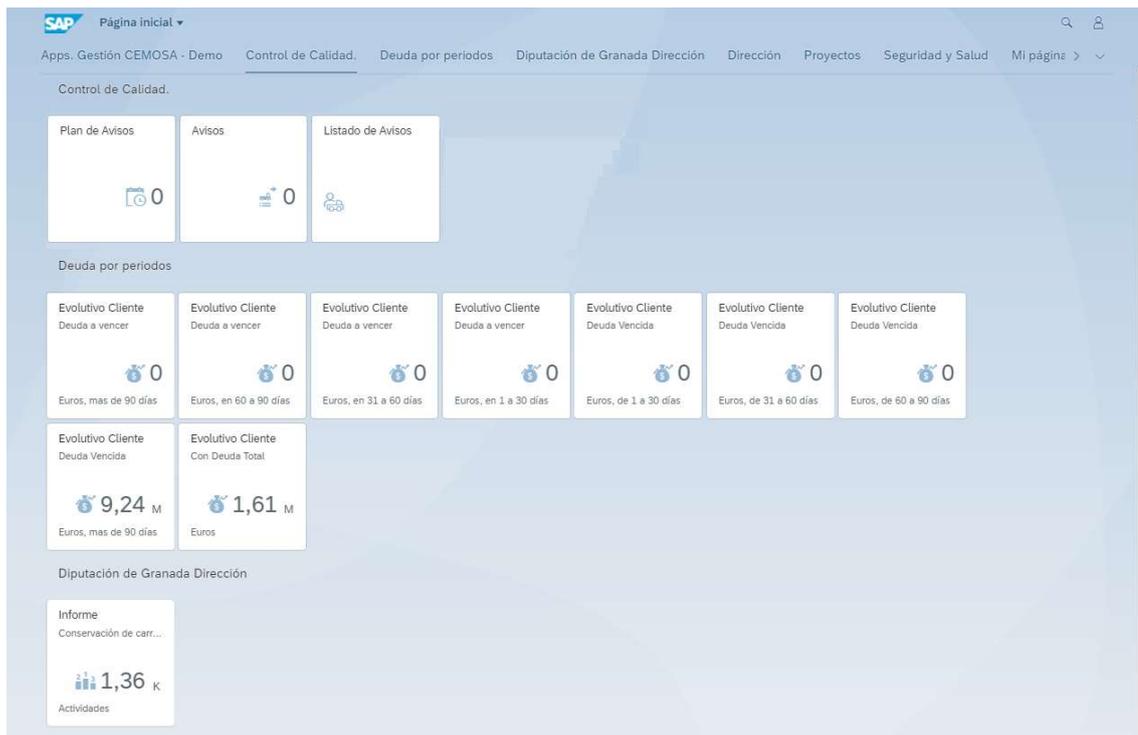
## Manual de Usuario

### *Acceso a SAP-Fiori Launchpad:*

1. Existen dos medios de acceso dependiendo del dispositivo si está en un ordenador acceda mediante la URL corporativa y si está en un dispositivo móvil acceda mediante la aplicación SAP-Fiori.
2. Introducir las credenciales proporcionadas por el Dep. de Informática.

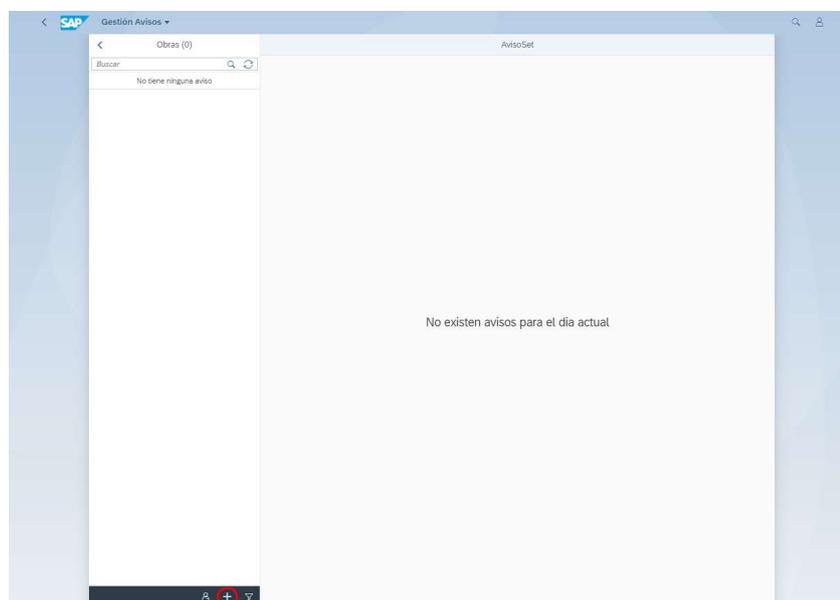


3. Una vez realizado el login podrá visualizar el portal de aplicaciones SAP-Fiori Launchpad.

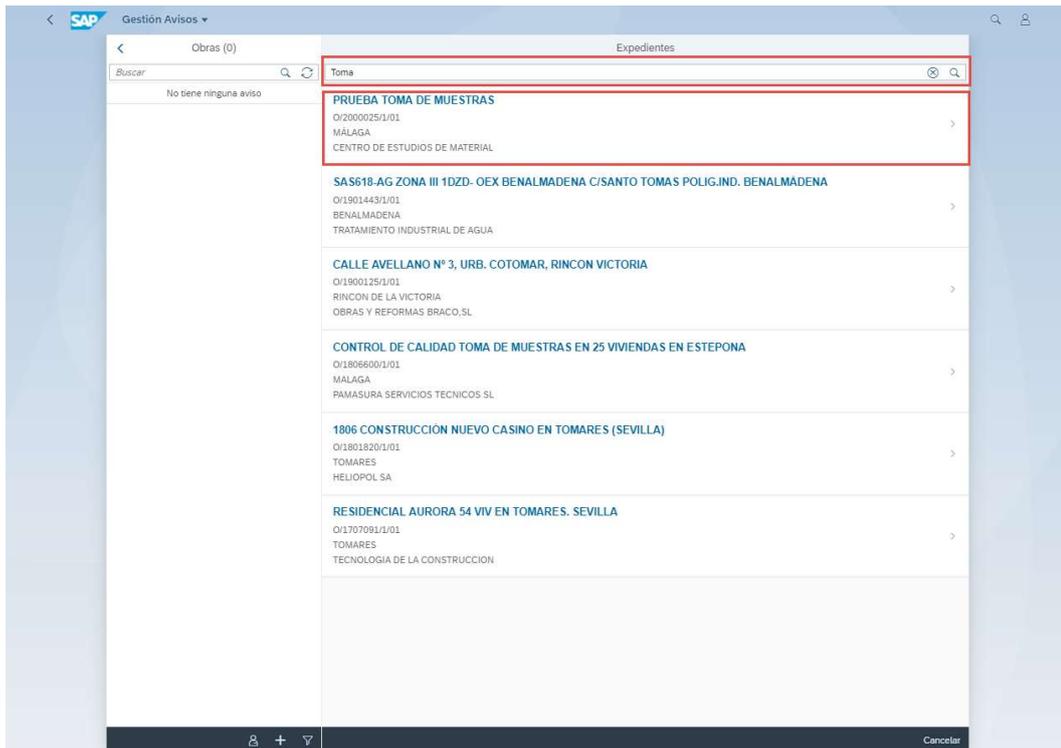


## *App Alta Aviso:*

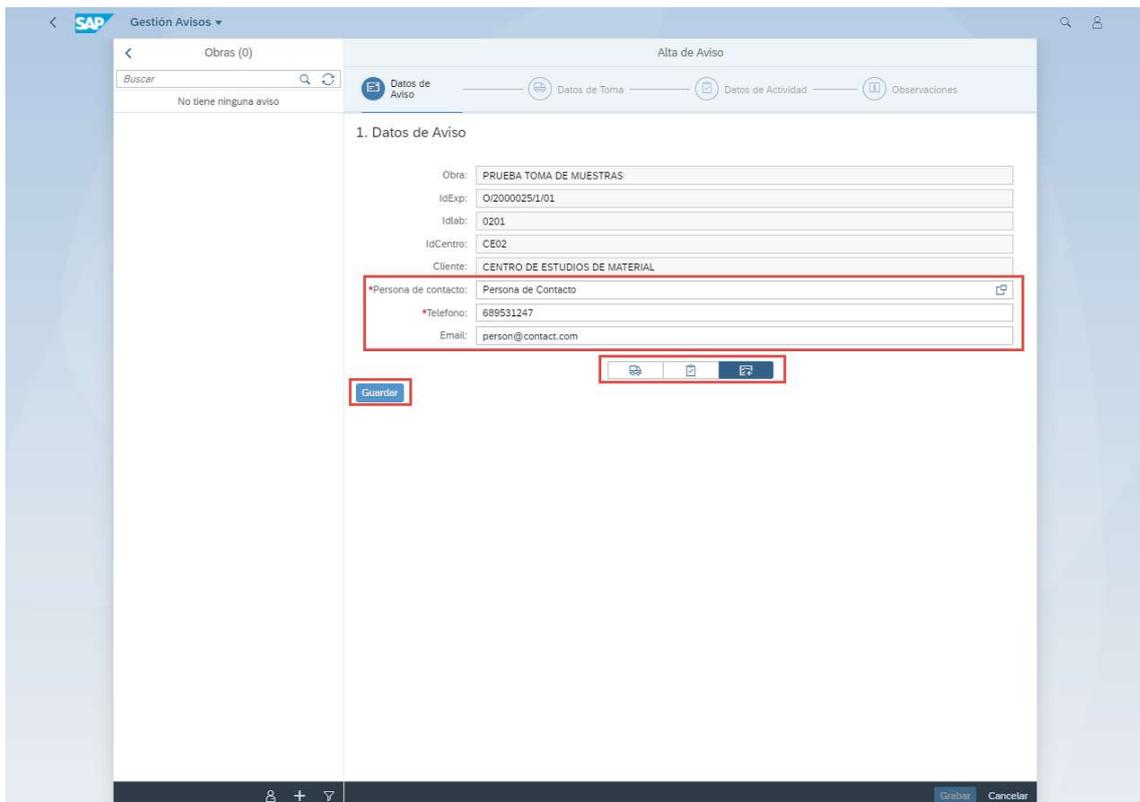
1. Entramos en la aplicación Avisos y le damos al botón + para registrar un aviso.



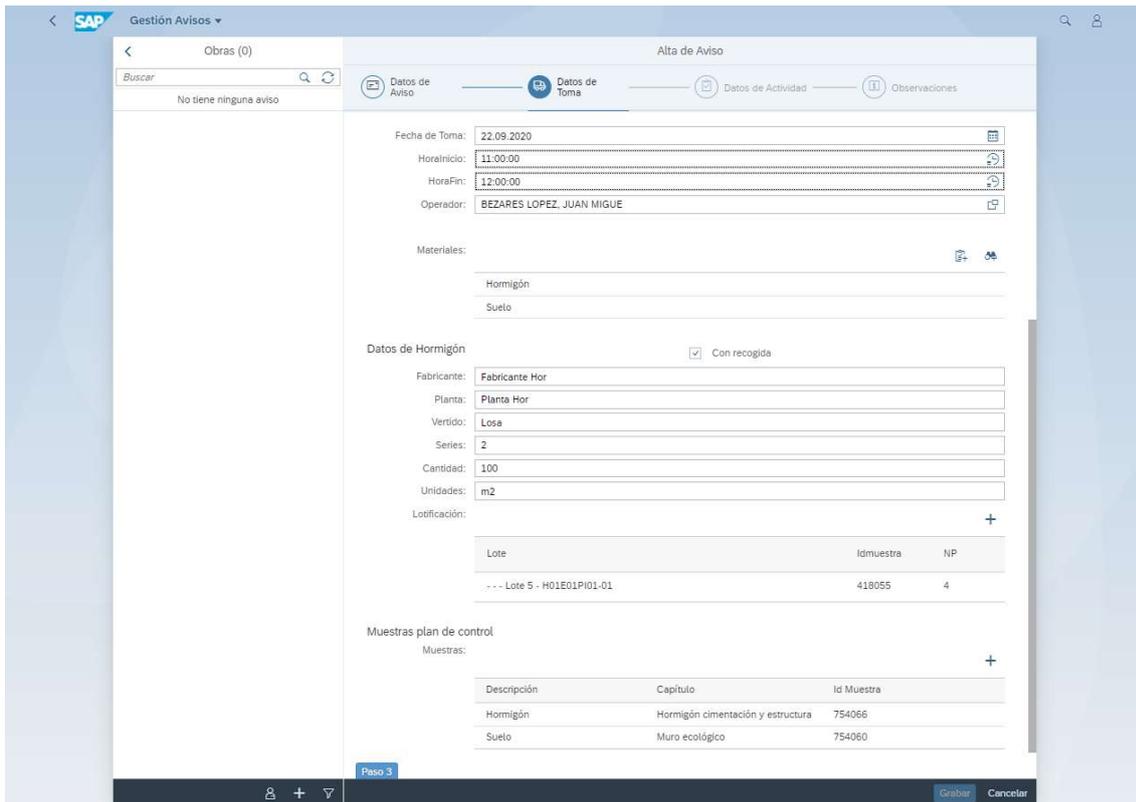
2. Buscamos y seleccionamos el expediente al que se asociara el aviso.



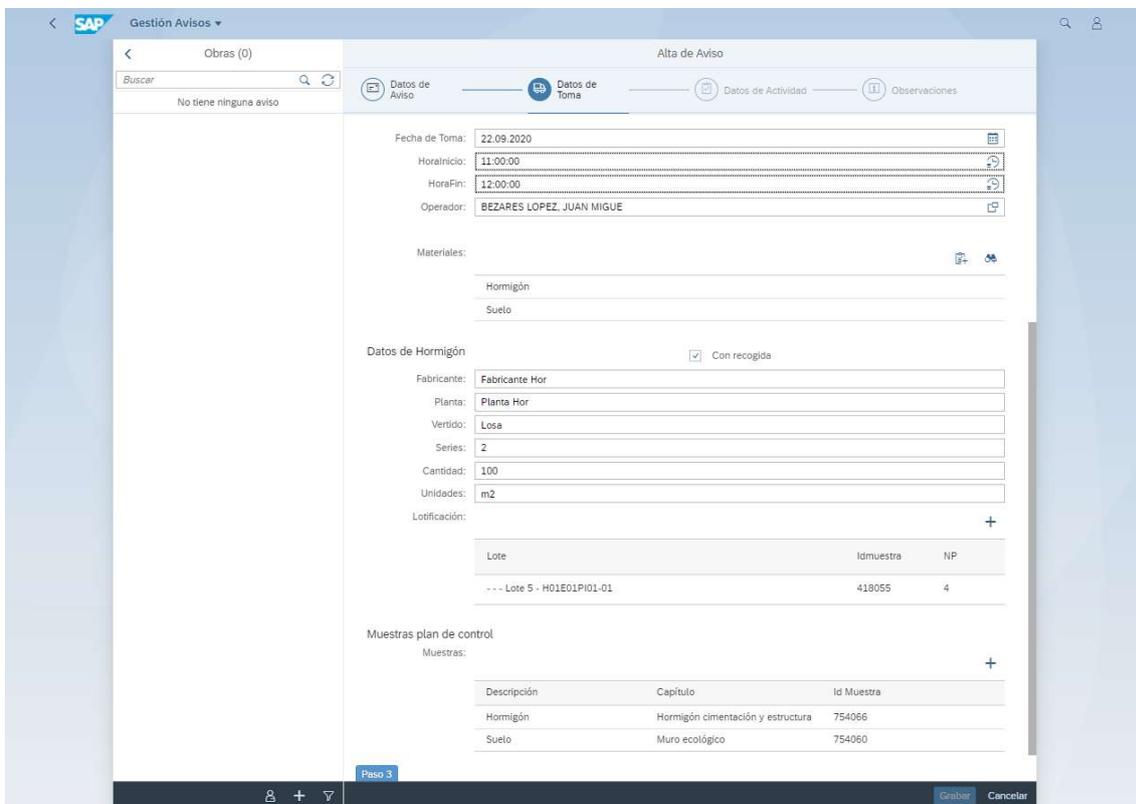
3. Introducimos la persona y sus datos de contacto, seleccionamos el tipo de aviso y pulsamos el botón de siguiente paso.



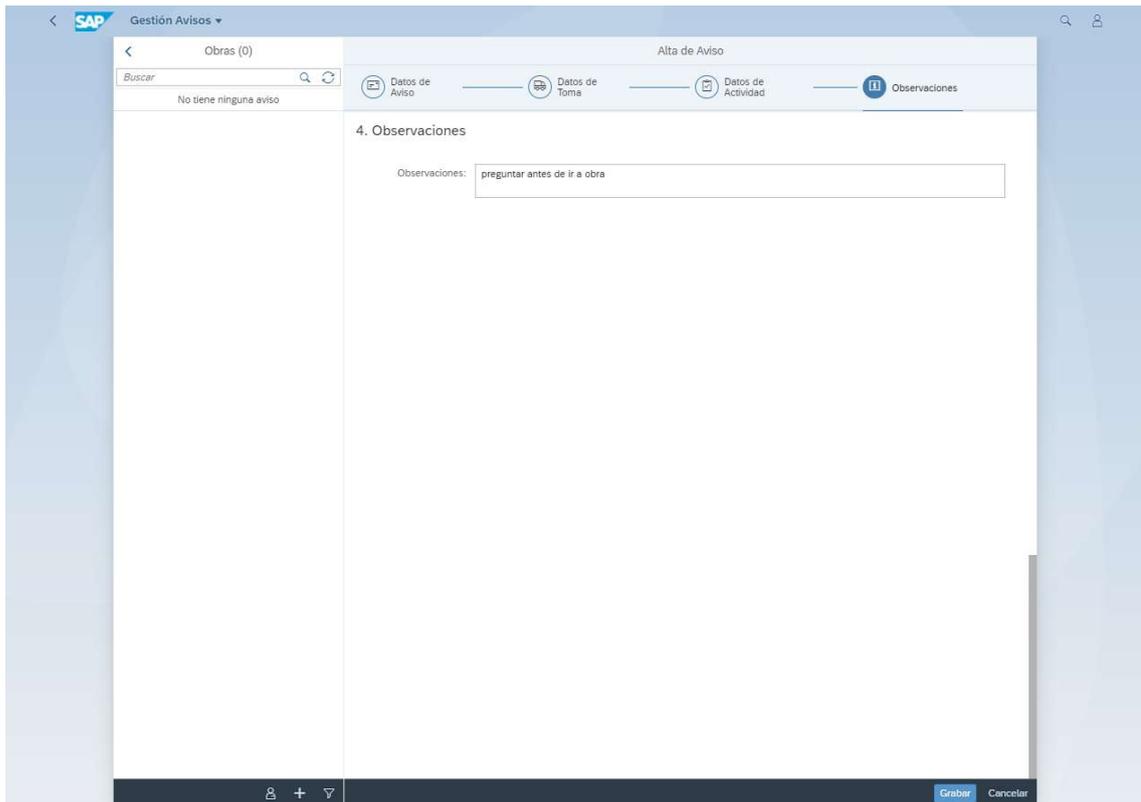
4. Introducimos los datos de toma Fecha, Horas, Operador, materiales. Si el tipo de material hormigón se halla entre los seleccionados se introducen los datos de este. Por último, en este paso se seleccionan las muestras asociadas para los materiales si estas están ya programadas y pulamos el botón de siguiente paso.



5. Introducimos los datos de Actividad Fecha, Horas, Operador, Materiales, servicios a realizar y pulsamos el botón de siguiente paso.

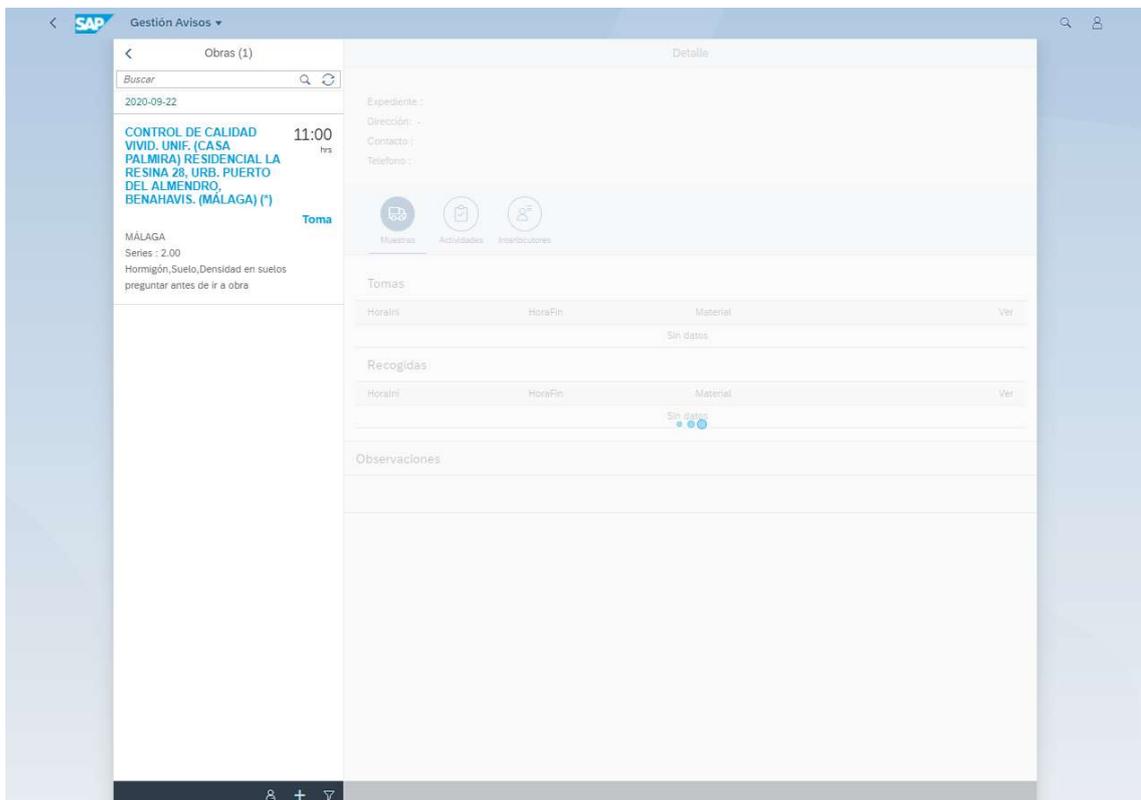


6. Introducimos las observaciones para el/los operadores y guardamos.

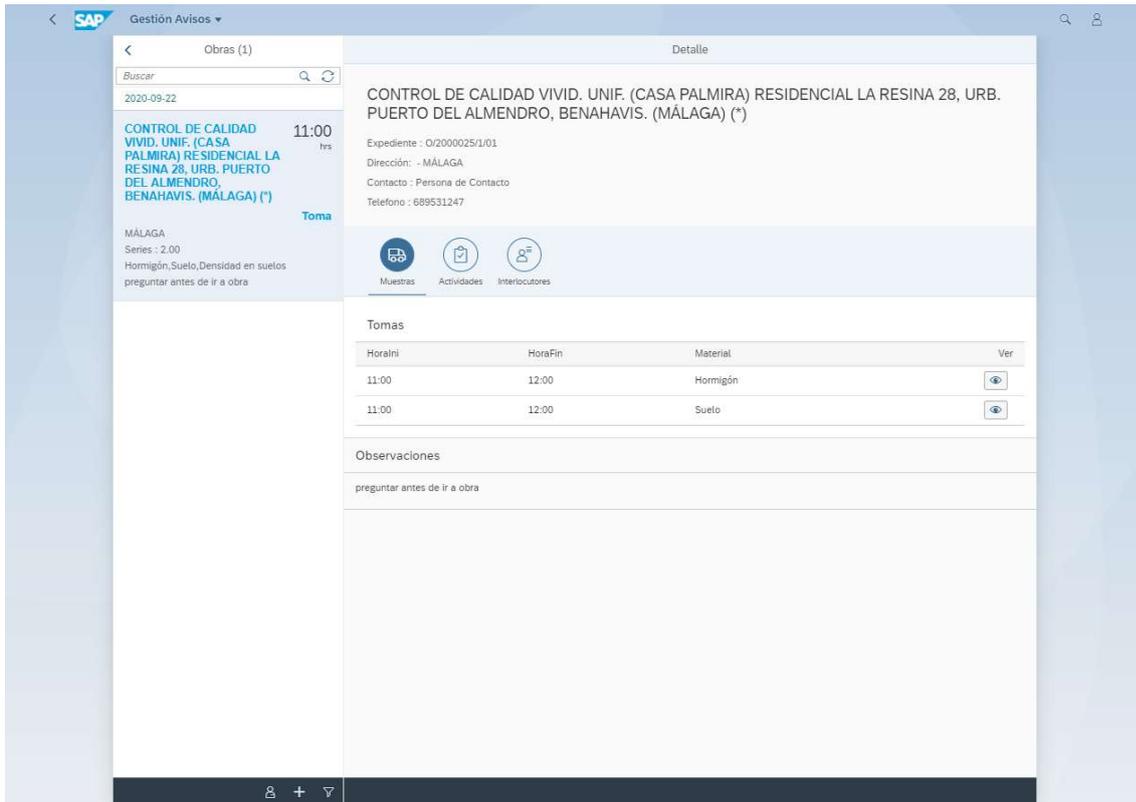


## *App Avisos:*

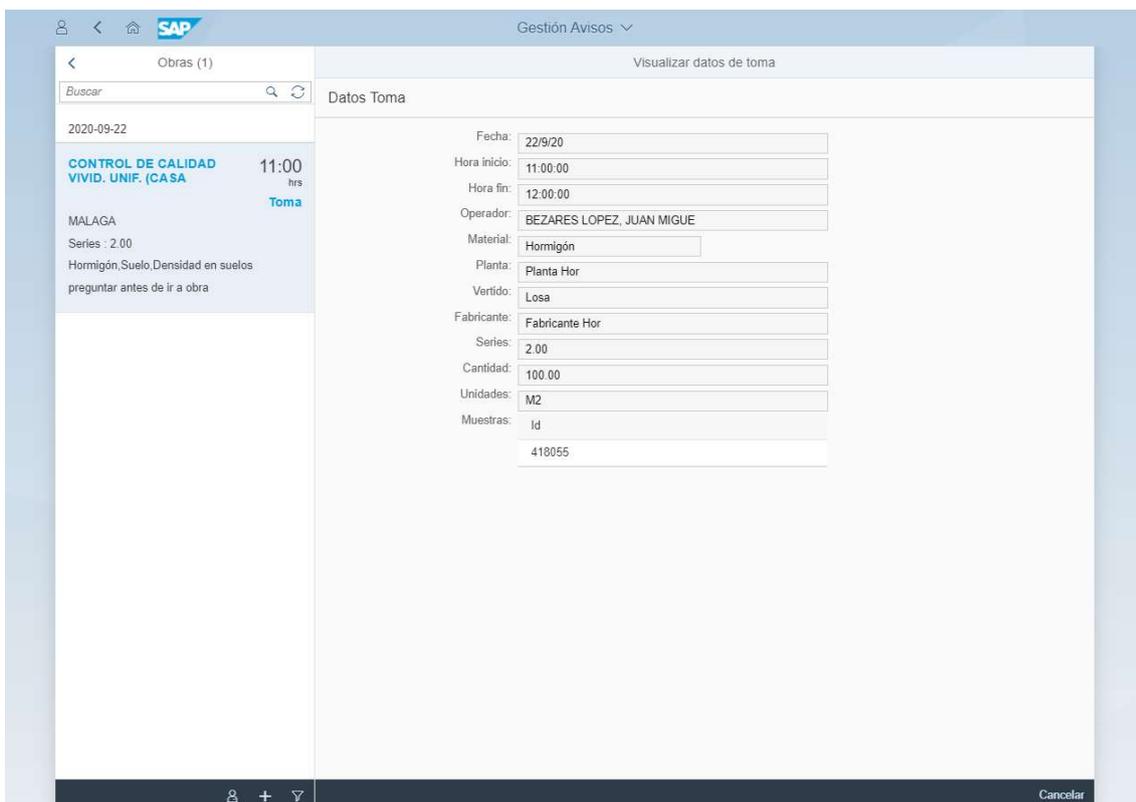
1. Entramos en la aplicación Avisos y seleccionamos el aviso de la lista de avisos diarios asignados.



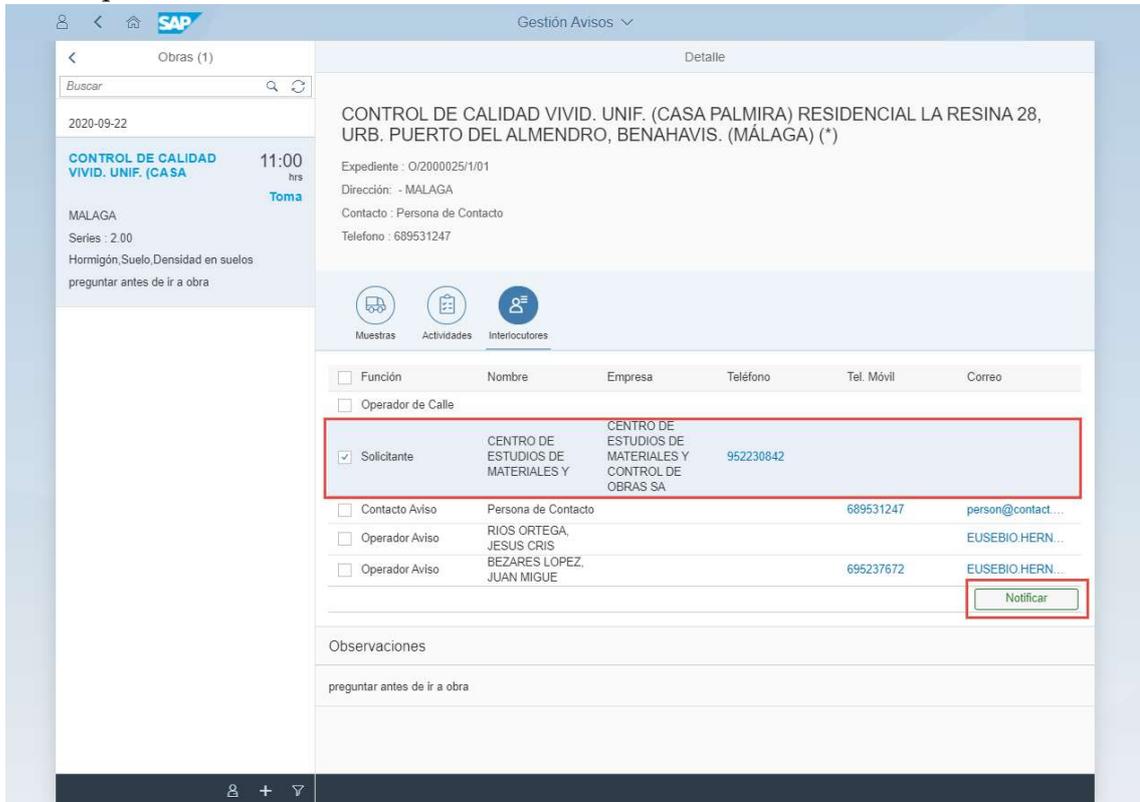
2. En la siguiente pantalla (Detalle del aviso) podemos ver las tomas, actividades, observaciones e interlocutores asociados al aviso



3. Pulsando el botón de ver en cada registro de toma o actividad podemos en detalle todos los datos de la tarea seleccionada.

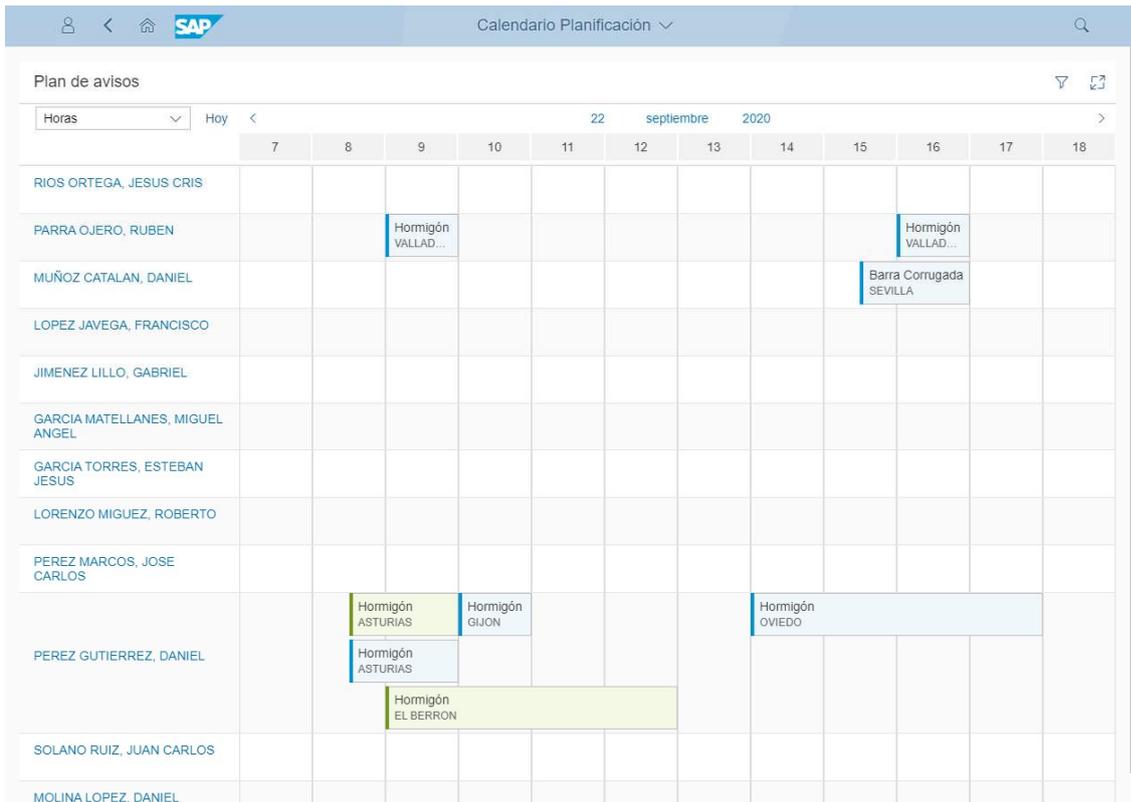


4. La notificación se realiza desde la pestaña de interlocutores para lo cual debemos seleccionar los interlocutores a los que queremos notificar y presionar el botón notificar.

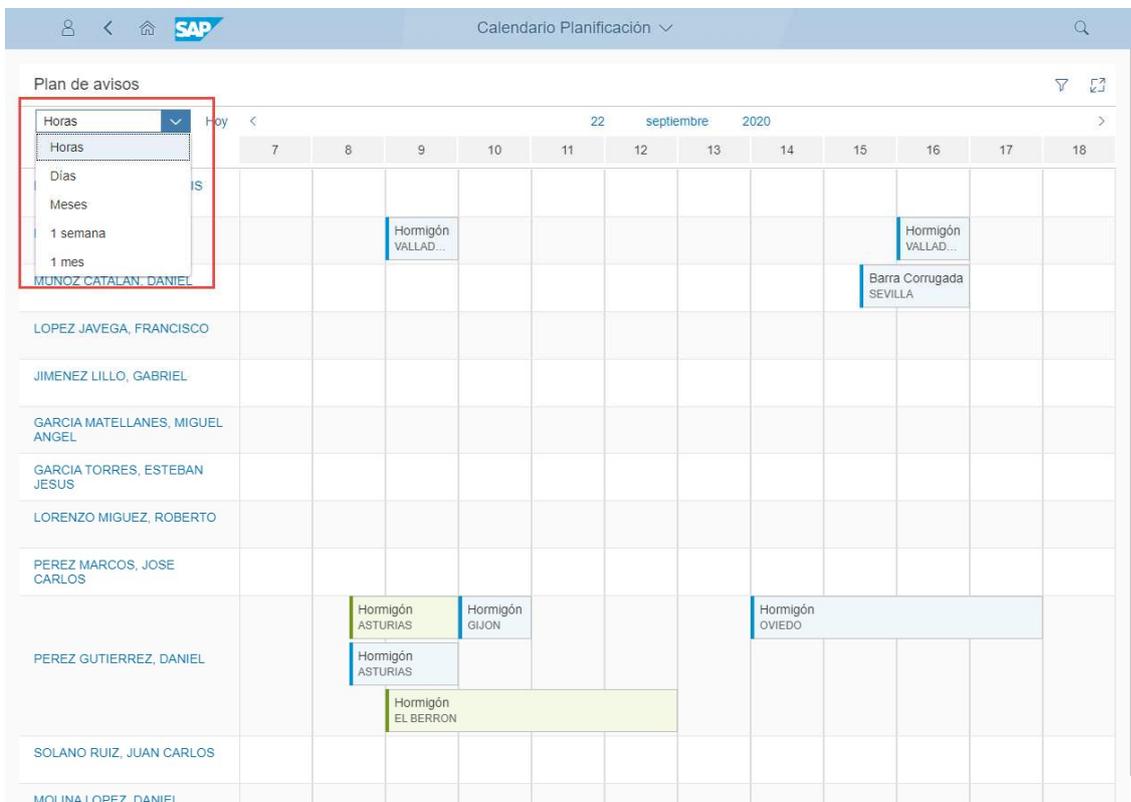


### ***App Plan de Avisos:***

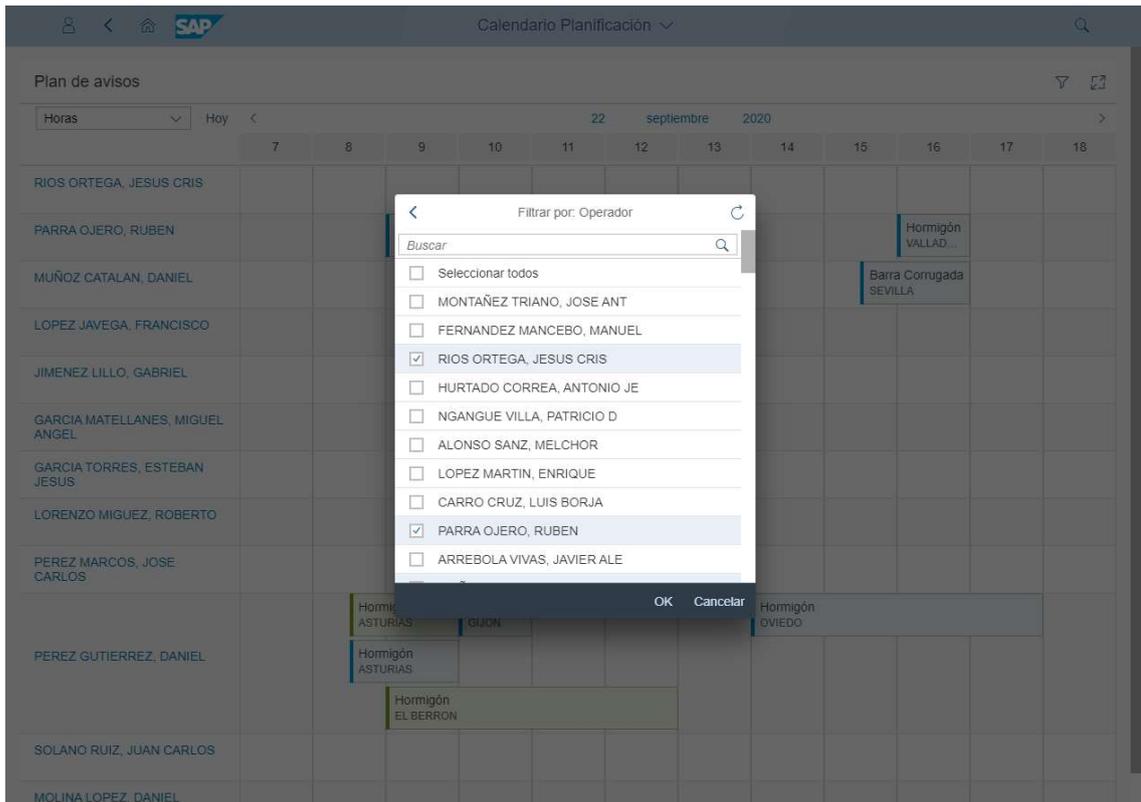
1. Entramos en la aplicación Plan de Avisos dentro podemos visualizar la agenda de trabajos de los operadores por horas donde cada tarea se representa por un elemento fijado dentro del su horario correspondiente.



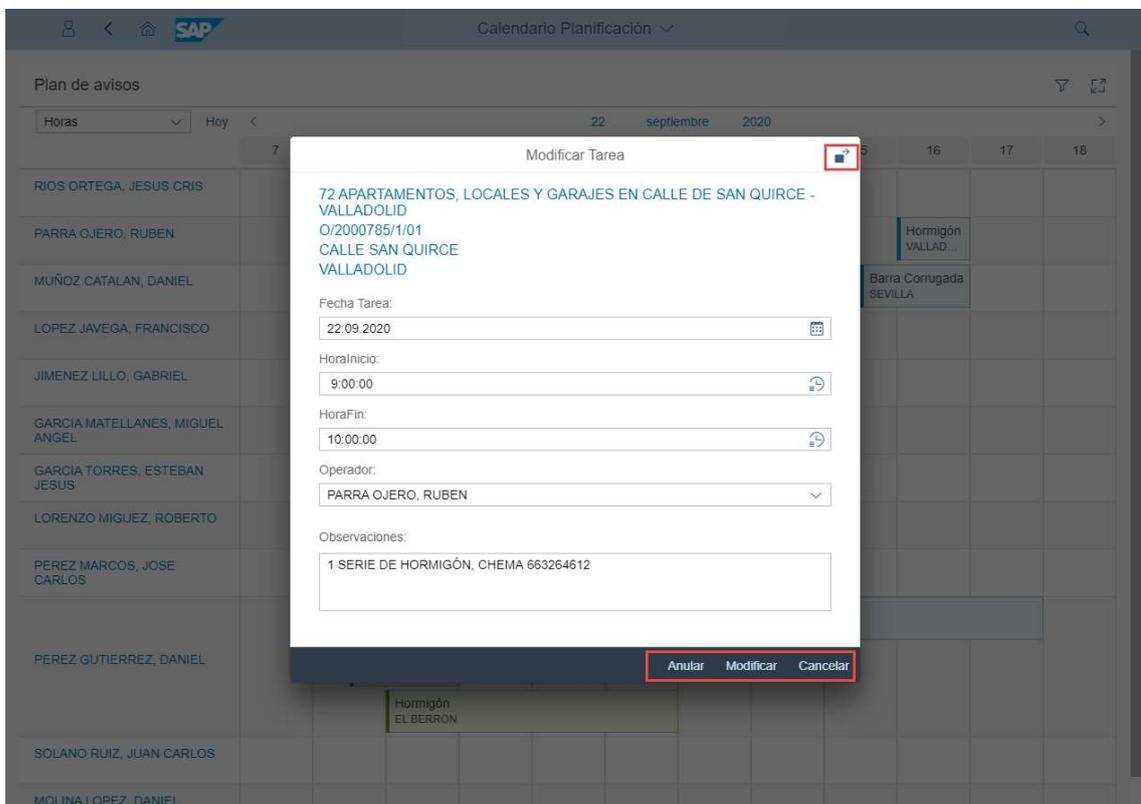
2. Existe la posibilidad de cambiar el periodo de horas por días semanas y meses.



3. Mediante el filtro por operador se puede visualizar uno o varios operadores.



- Para ver el detalle de la tarea se debe hacer clic sobre la misma esto abre un pop up donde podemos visualizar los principales datos de la tarea y desde este se nos da la posibilidad de modificar los mismos, anular la tarea o navegar a la aplicación Avisos.



## App Listado de Avisos:

1. Entramos en la aplicación Listado de avisos, al entrar el listado sale vacío para recuperar los datos podemos usar los distintos filtros que se ofrecen para restringir los resultados y pulsamos el botón "ir".

Estándar \* ☺ Suprimir filtros

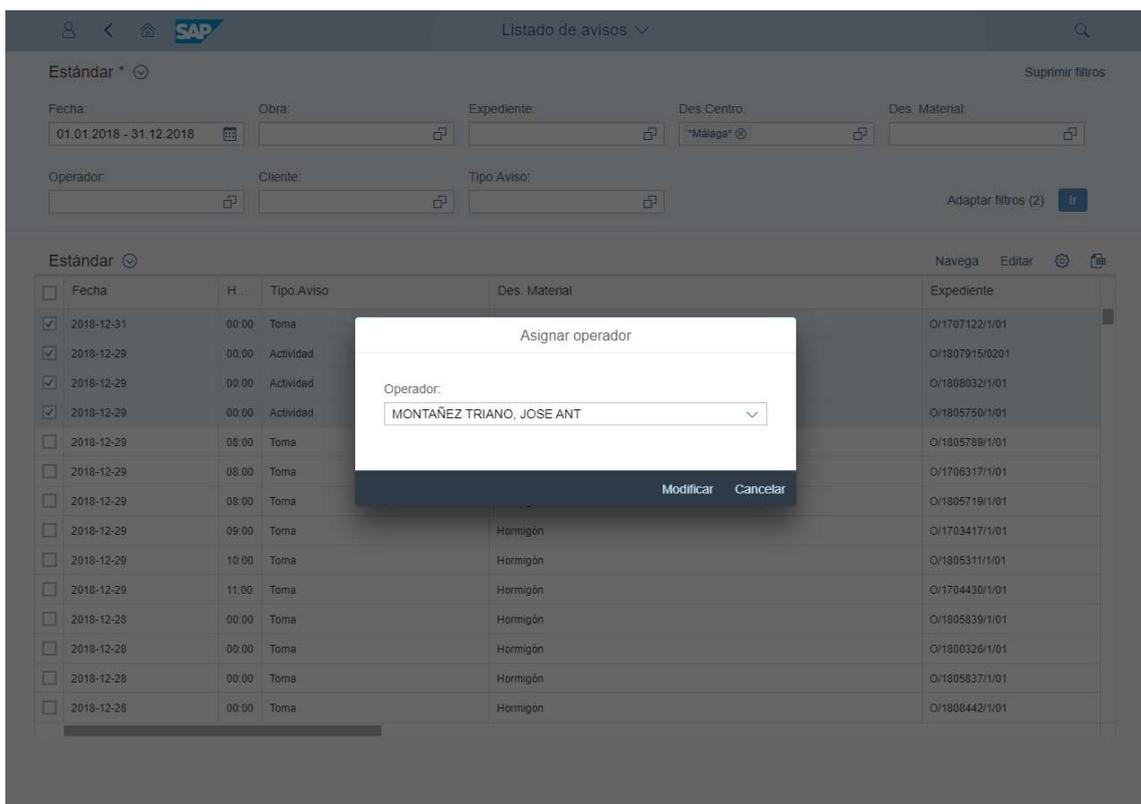
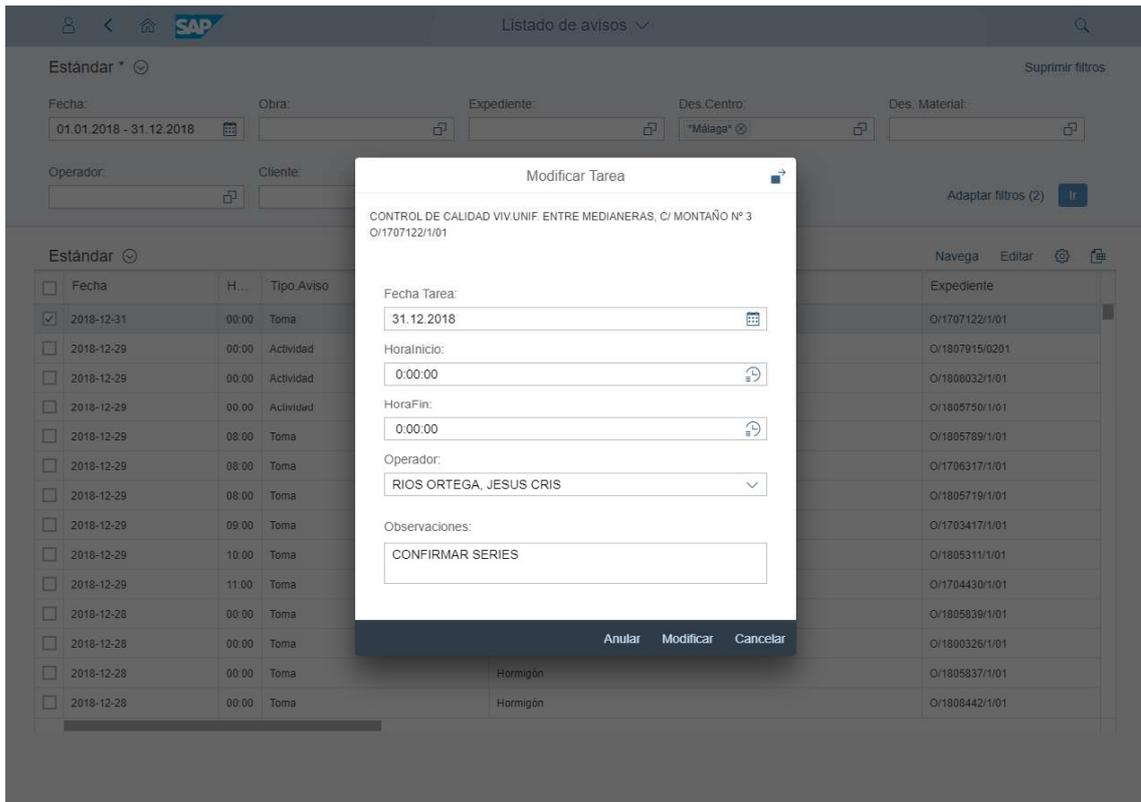
Fecha: 01.01.2018 - 31.12.2018   Expediente:  Des. Centro: \*Málaga\*  Des. Material:

Operador:  Cliente:  Tipo.Aviso:  Adaptar filtros (2) **Ir**

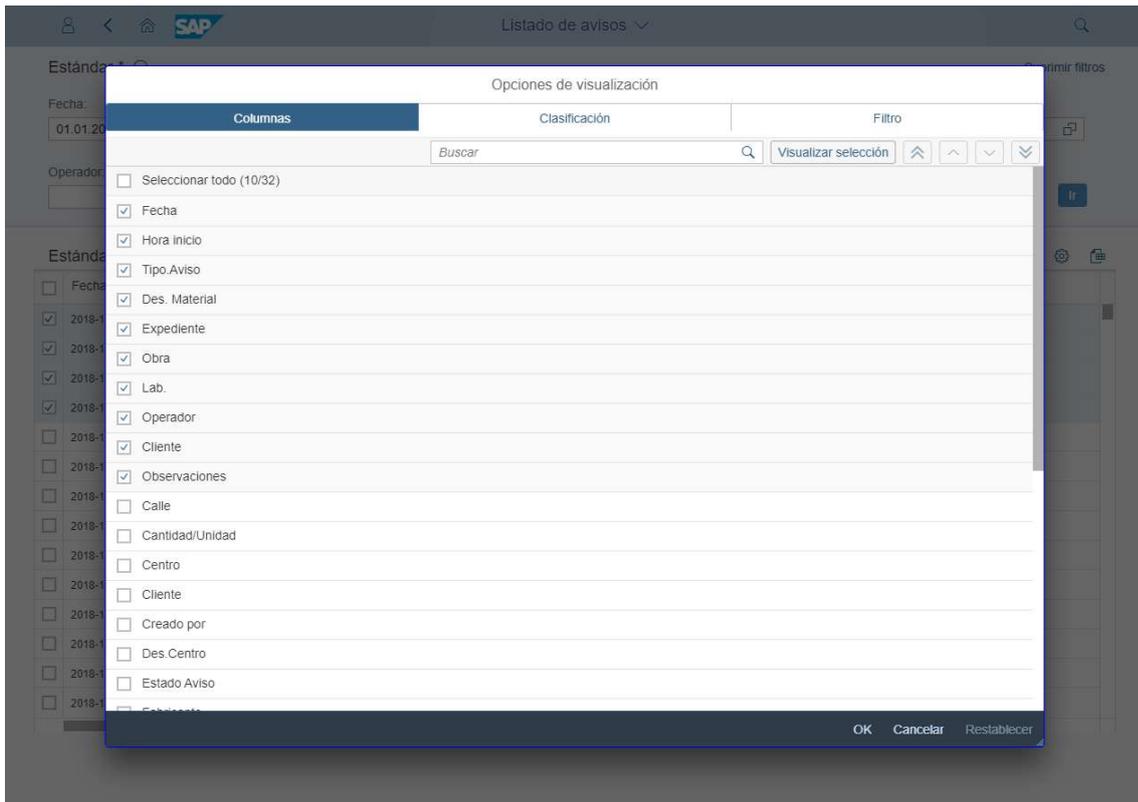
Estándar ☺ Navega Editar

<input type="checkbox"/>	Fecha	H...	Tipo.Aviso	Des. Material	Expediente
<input type="checkbox"/>	2018-12-31	00:00	Toma	Hormigón	O/1707122/1/01
<input type="checkbox"/>	2018-12-29	00:00	Actividad	Densidad en suelos	O/1807915/0201
<input type="checkbox"/>	2018-12-29	00:00	Actividad	Densidad en suelos	O/1808032/1/01
<input type="checkbox"/>	2018-12-29	00:00	Actividad	Densidad en suelos	O/1805750/1/01
<input type="checkbox"/>	2018-12-29	08:00	Toma	Hormigón	O/1805789/1/01
<input type="checkbox"/>	2018-12-29	08:00	Toma	Hormigón	O/1706317/1/01
<input type="checkbox"/>	2018-12-29	08:00	Toma	Hormigón	O/1805719/1/01
<input type="checkbox"/>	2018-12-29	09:00	Toma	Hormigón	O/1703417/1/01
<input type="checkbox"/>	2018-12-29	10:00	Toma	Hormigón	O/1805311/1/01
<input type="checkbox"/>	2018-12-29	11:00	Toma	Hormigón	O/1704430/1/01
<input type="checkbox"/>	2018-12-28	00:00	Toma	Hormigón	O/1805839/1/01
<input type="checkbox"/>	2018-12-28	00:00	Toma	Hormigón	O/1800326/1/01
<input type="checkbox"/>	2018-12-28	00:00	Toma	Hormigón	O/1805837/1/01
<input type="checkbox"/>	2018-12-28	00:00	Toma	Hormigón	O/1808442/1/01

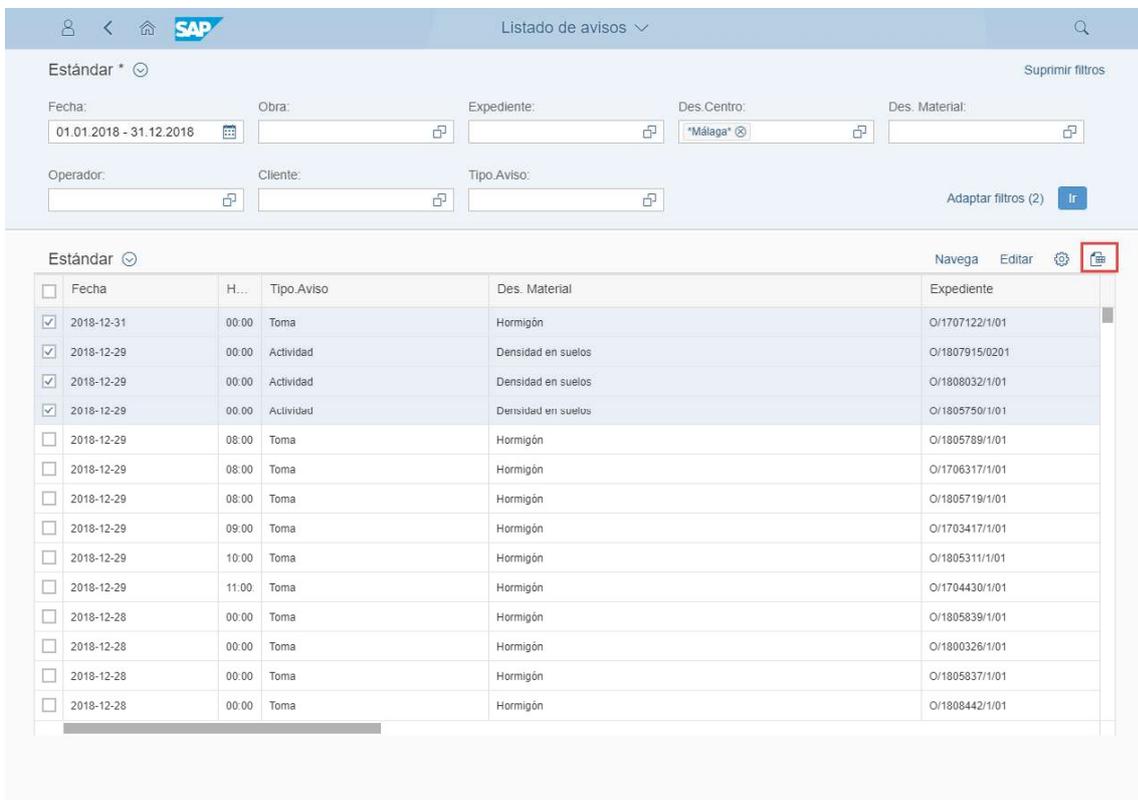
2. Una vez tenemos el listado de tomas y actividades podemos modificar uno o varios registros seleccionándolos y presionando el botón editar si seleccionamos un registro individual salta un popup con los datos de la toma/actividad y las opciones de modificar anular o navegar a la aplicación Avisos en cambio si se seleccionan varios registros el pop up restringe los datos que se pueden modificar a solo el operador asignado.



3. También es posible navegar directamente seleccionando un registro y presionando el botón navegar.
4. El botón opciones nos permite personalizar la disposición de la tabla.



- Una vez tenemos seleccionado un grupo de registros y configurada una disposición podemos exportar los datos a un fichero Excel presionando el botón Exportar a hoja de cálculo.



## App Informe (Conservación carreteras):

1. Entramos en la aplicación de Informe de conservación de carreteras. Al entrar podemos visualizar todas las graficas que forman parte del informe.



2. Para la grafica de actuaciones se disponen de filtros para acotar los datos.



3. Si presionamos sobre el botón de cambio de perspectiva de la gráfica de detalle de evolución de actuaciones podremos visualizar los datos en formato de tabla.

SE 19/2017 SERVICIO EN MATERIA DE SEGURIDAD Y SALUD EN LAS ACTIVIDADES DE CONSERVACIÓN DE LA RED DE CARRETERAS DE LA DIPUTACIÓN PROVINCIAL DE GRANADA

DETALLE ACTUACIONES    ACTUACIONES Y VISITAS TOTALES

Todos    Todas    Todas

Actuaciones

Año/Mes	Zona	Actividad	Total
2018Agosto	Con-Z1	Señalización	5
2018Agosto	Con-Z1	Limpieza/Cunetas	5
2018Agosto	Con-Z1	Bacheos	1
2018Agosto	Con-Z1	Otros	1
2018Agosto	Con-Z2	Poda y desbroce	1
2018Agosto	Con-Z2	Limpieza/Cunetas	2
2018Agosto	Con-Z3	Señalización	2
2018Agosto	Con-Z3	Limpieza/Cunetas	2
2018Agosto	Con-Z3	Bacheos	4
2018Agosto	Con-Z4	Señalización	14
2018Agosto	Con-Z4	Poda y desbroce	5
2018Agosto	Con-Z4	Limpieza/Cunetas	1
2018Agosto	Con-Z4	Otros	1
2018Agosto	Con-Z5	Señalización	2
2018Agosto	Con-Z5	Limpieza/Cunetas	2
2018Agosto	Con-Z5	Bacheos	1
2018Agosto	Con-Z5	Fitosanitarios	3
2018Agosto	Con-Z5	Obras de fabrica y cunetas hormigón	2
2018Agosto	Con-Z5	Otros	5
2018Septiembre	Con-Z1	Poda y desbroce	2
2018Septiembre	Con-Z1	Limpieza/Cunetas	8
2018Septiembre	Con-Z1	Obras de fabrica y cunetas hormigón	8
2018Septiembre	Con-Z2	Señalización	5
2018Septiembre	Con-Z2	Limpieza/Cunetas	10
2018Septiembre	Con-Z2	Obras de fabrica y cunetas hormigón	15
2018Septiembre	Con-Z2	Otros	1
2018Septiembre	Con-Z3	Bacheos	4
2018Septiembre	Con-Z4	Señalización	2
2018Septiembre	Con-Z4	Poda y desbroce	1
2018Septiembre	Con-Z4	Limpieza/Cunetas	12
2018Septiembre	Con-Z4	Obras de fabrica y cunetas hormigón	5
2018Septiembre	Con-Z4	Otros	1
2018Septiembre	Con-Z5	Señalización	11
2018Septiembre	Con-Z5	Limpieza/Cunetas	7

4. En este formato de visualización es posible exportar los datos a un fichero Excel presionando el botón Exportar a hoja de cálculo.

SE 19/2017 SERVICIO EN MATERIA DE SEGURIDAD Y SALUD EN LAS ACTIVIDADES DE CONSERVACIÓN DE LA RED DE CARRETERAS DE LA DIPUTACIÓN PROVINCIAL DE GRANADA

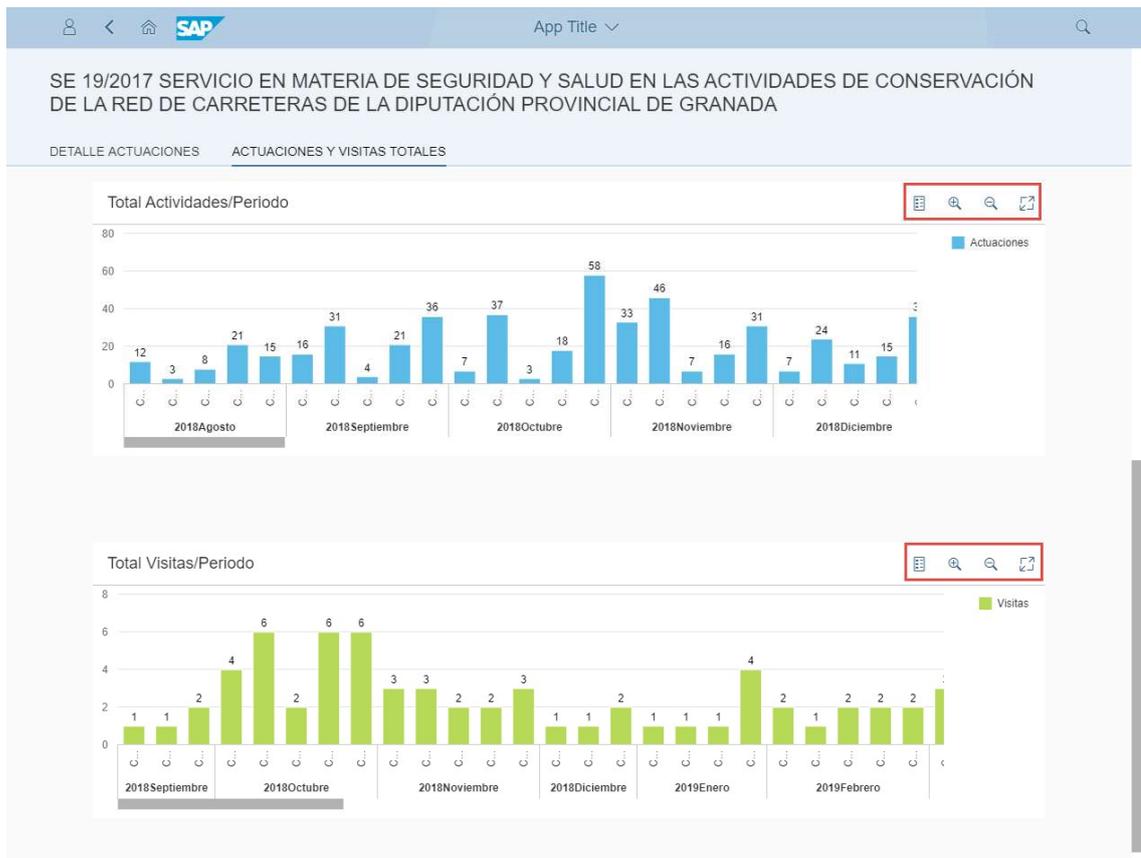
DETALLE ACTUACIONES    ACTUACIONES Y VISITAS TOTALES

Todos    Todas    Todas

Actuaciones

Año/Mes	Zona	Actividad	Total
2018Agosto	Con-Z1	Señalización	5
2018Agosto	Con-Z1	Limpieza/Cunetas	5
2018Agosto	Con-Z1	Bacheos	1
2018Agosto	Con-Z1	Otros	1
2018Agosto	Con-Z2	Poda y desbroce	1
2018Agosto	Con-Z2	Limpieza/Cunetas	2
2018Agosto	Con-Z3	Señalización	2
2018Agosto	Con-Z3	Limpieza/Cunetas	2
2018Agosto	Con-Z3	Bacheos	4
2018Agosto	Con-Z4	Señalización	14
2018Agosto	Con-Z4	Poda y desbroce	5
2018Agosto	Con-Z4	Limpieza/Cunetas	1
2018Agosto	Con-Z4	Otros	1
2018Agosto	Con-Z5	Señalización	2
2018Agosto	Con-Z5	Limpieza/Cunetas	2
2018Agosto	Con-Z5	Bacheos	1
2018Agosto	Con-Z5	Fitosanitarios	3
2018Agosto	Con-Z5	Obras de fabrica y cunetas hormigón	2
2018Agosto	Con-Z5	Otros	5

5. Cada grafica tiene varios botones de funcionalidades adicionales como leyenda, lupa y pantalla completa.



### ***App Evolutivo Cliente:***

Nota\*. -La funcionalidad descrita es aplicable a todos los tiles de esta aplicación que representan distintos periodos y tipos de importes adeudados.

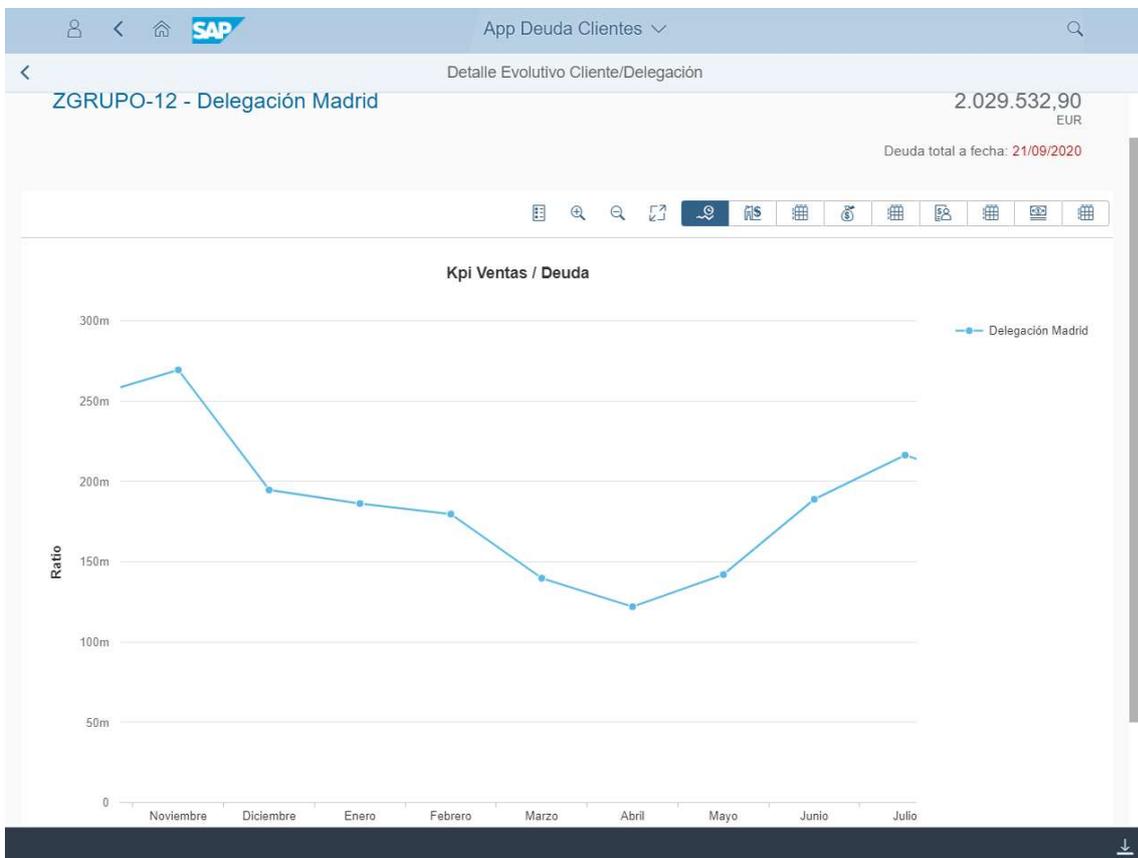
1. Entramos en la aplicación de Evolutivo de cliente donde se nos presenta un listado con las delegaciones/clientes que cumplen la condición de importe adeudado de la variante del tile escogido.

App Deuda Clientes

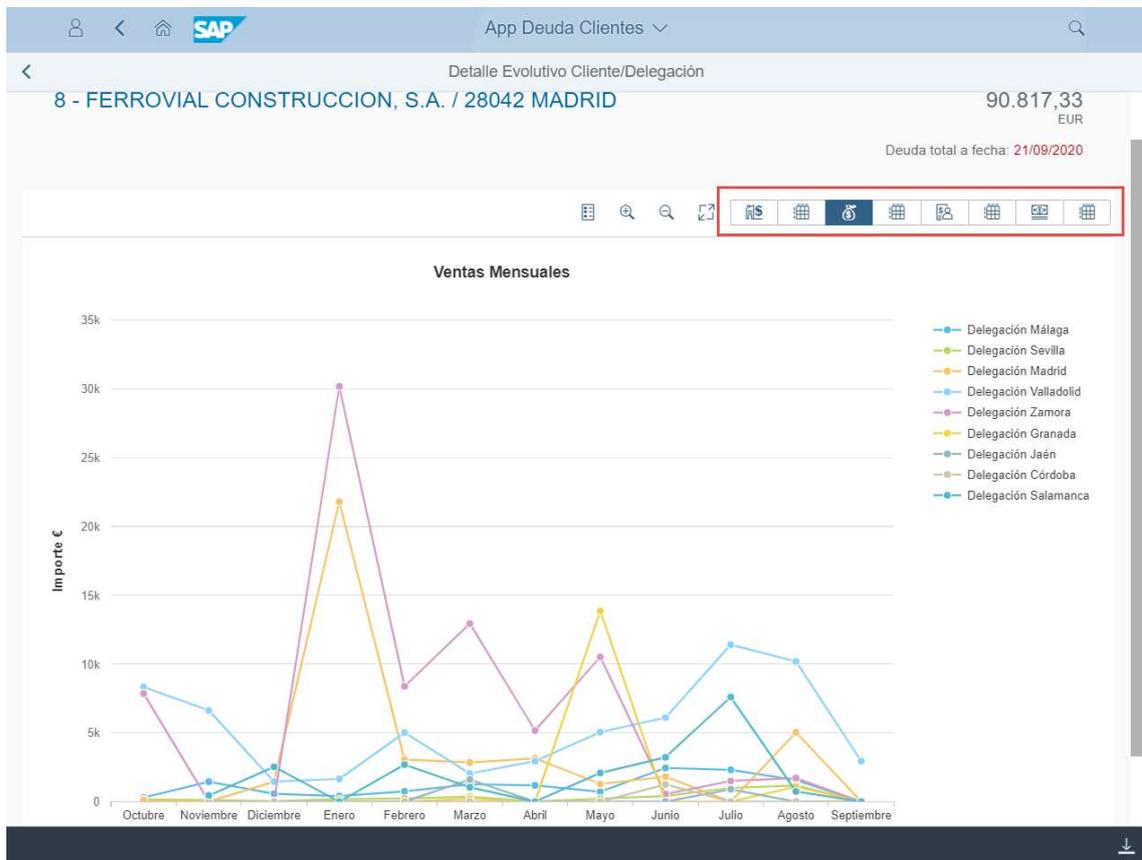
Evolutivo Clientes/Delegación (Contratación - Ventas - Deuda Total - Cobros)

Cliente	Delegaciones	Inci/Dec Deuda
Delegación Madrid	Delegación Madrid	Jun 2020 1,81 M 221,77 K Sep 2020 2,03 M
AYUNTAMIENTO DE MADRID / 28004 MADRID	Delegación Madrid	Jun 2020 176,56 K 155,85 K Sep 2020 332,41 K
AENA S.M.E., S.A. / 28042 MADRID	Ingeniería, Delegación Sevilla	Jun 2020 592,78 92,03 K Sep 2020 92,62 K
FERROVIAL CONSTRUCCION, S.A. / 28042 MADRID	Delegación Palma de Mallorca, Delegación Málaga, Delegación Jaén, Delegación Sevilla, Delegación Valladolid, Delegación Alicante, Delegación Salamanca, Delegación Zamora, Delegación Madrid	Jun 2020 95,11 K -4,3 K Sep 2020 90,82 K
INGENIERIA Y ECONOMIA TRANSPORTE / 28036 MADRID	Delegación Sevilla, Delegación Madrid, Delegación Málaga	Jun 2020 29,73 K 45,7 K Sep 2020 75,43 K
SERVICIO MADRILEÑO DE SALUD / 28020 MADRID	Delegación Madrid	Jun 2020 73,53 K 0 Sep 2020 73,53 K
NEINOR SUR S.A.U / 28046 MADRID	Delegación Málaga, Delegación Sevilla, Delegación Madrid, Delegación Córdoba	Jun 2020 78,95 K -6,4 K Sep 2020 72,55 K
UTE EPTISA-CEMOSA / 28037 MADRID	Delegación Almería	Jun 2020 60,85 K 10,01 K Sep 2020 70,86 K
UTE AUTOVIA A-12 / 28051 MADRID	Delegación Valladolid	Jun 2020 0 67,97 K Sep 2020 67,97 K
IBERENOVA PROMOCIONES S.A. / 28033 MADRID	Delegación Madrid	Jun 2020 2,83 K 60,81 K Sep 2020 63,64 K
SOCIEDAD DE INFRAESTRUCTURAS Y / 28001 MADRID	Delegación Jerez, Delegación Sevilla, Delegación Madrid	Jun 2020 45,51 K 17,97 K Sep 2020 63,48 K
	Delegación Madrid, Delegación Sevilla	Jun 2020 67,2 K

2. Seleccionando un Cliente/Delegación navegamos a una pantalla donde se nos muestra la gráfica de KPI de Deuda/Contratación.



3. Para visualizar los datos de cobros ventas deuda y contratación es necesario pulsar en el botón correspondiente dentro del contenedor de graficas.



4. Junto con cada grafica existe un botón para visualizar los datos en formato tabla.

App Deuda Clientes			
Detalle Evolutivo Cliente/Delegación			
8 - FERROVIAL CONSTRUCCION, S.A. / 28042 MADRID			90.817,33 EUR
Deuda total a fecha: 21/09/2020			
<b>Ventas Mensuales</b>			
Ejercicio	Mes	Delegación	Importe
2019	Octubre	Delegación Málaga	300,00 EUR
2019	Octubre	Delegación Sevilla	158,00 EUR
2019	Octubre	Delegación Madrid	92,00 EUR
2019	Octubre	Delegación Valladolid	8.332,10 EUR
2019	Octubre	Delegación Zamora	7.854,00 EUR
2019	Noviembre	Delegación Málaga	1.450,00 EUR
2019	Noviembre	Delegación Granada	0,00 EUR
2019	Noviembre	Delegación Jaén	0,00 EUR
2019	Noviembre	Delegación Córdoba	0,00 EUR
2019	Noviembre	Delegación Sevilla	120,00 EUR
2019	Noviembre	Delegación Madrid	0,00 EUR
2019	Noviembre	Delegación Valladolid	6.629,70 EUR
2019	Noviembre	Delegación Zamora	0,00 EUR
2019	Noviembre	Delegación Salamanca	452,12 EUR
2019	Diciembre	Delegación Málaga	570,00 EUR