



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escuela Técnica Superior de Ingeniería Informática
Universidad Politécnica de Valencia

Aplicación de Gestión de Incidencias Municipales

Trabajo Fin de Grado

Grado en Ingeniería Informática

Autor: Carlos Frias Ruiz

Tutores: Juan Carlos Ruiz García

David de Andrés Martínez

2015-2016

Resumen

El Ayuntamiento de Chiva tiene la necesidad de gestionar las órdenes de trabajo con la brigada municipal, para el control de los trabajos realizados en todo el término municipal. Se pretende analizar y desarrollar un sistema de gestión de órdenes de trabajo para la brigada municipal, cuya gestión se realice de forma centralizada. Se pretende además una asignación de órdenes de trabajo en tiempo real a través de tecnología móvil, de tal forma que los trabajadores reciban y tramiten las órdenes de trabajo que tengan asignadas a través de dispositivos móviles. Del mismo modo, toda la actividad quedará registrada de forma centralizada de modo que se pueda consultar en cualquier momento las Órdenes tramitadas y la resolución de las mismas. Otro de los aspectos que gestionará la solución ofrecida es la gestión de recursos materiales, al quedar registrados en dichas Órdenes la cantidad de estos que han sido utilizados. Esta última parte dependerá del acceso que se tenga al inventario de dichos materiales.

Abstract

The Chiva's Council needs to manage the Task Orders with its local squad, to have control over work done on the whole municipality. It is intended to analyze and develop a task order management system for the local squad, whose management will be performed centrally. It is also intended to have a real time task assignment via mobile technology, in such a way that workers will be able to get and manage their assigned task orders via their mobile devices. In the same way, any activity will be centrally registered in a such way that allows task orders to be consulted at any time as well as the resolution these tasks. Another issue that will offer this software solution is the management of the material resources due to the fact that will be recorded the amount of them used. That last feature depends on the access to these material inventories.

AGIM – TFG.

Índice de contenido

1· Introducción	6
2· Aplicaciones similares	11
2.1·IMK - App Partes de Trabajo (Ilustración:2.1)	11
2.2·Seaforms: Partes de Trabajo desde el Móvil (Ilustración:2.3)	12
2.3·Ovelan - Gestión de partes de trabajo On-Line (Ilustración:2.4)	12
2.4·Conclusiones	12
3·Especificación de requisitos de Software	13
3.1·Introducción	13
3.1.1·Propósito	13
3.1.2·Alcance del sistema	13
3.1.3·Definiciones	13
3.1.4·Visión global	18
3.2·Descripción general	19
3.2.1·Perspectiva del producto	19
3.2.1.1·Servicio Web	19
3.2.1.2·Aplicación Móvil	19
3.2.1.3·Aplicación de Escritorio	19
3.2.2·Funciones del producto	20
3.2.2.1·Funciones del Servicio Web	20
3.2.2.2·Funciones de la Aplicación Móvil	21
3.2.2.3·Funciones de la Aplicación de Escritorio	21
3.2.3·Características del Usuario	21
3.2.4·Restricciones generales	22
3.2.5·Supuestos y dependencias	22
3.3·Requisitos específicos	22
3.3.1·Requisitos interfaces externas	22
3.3.1.1·Requisitos Interfaz Operario	22
3.3.1.2·Requisitos interfaz Administrador	23
3.4·Requisitos Funcionales	23
3.4.1·Operaciones permitidas para el Administrador	23
3.4.2·Operaciones permitidas para el Operario	26
3.4.3·Operaciones permitidas para el Encargado	28
3.5·Rendimiento y Atributos	30
3.5.1·Rendimiento	30
3.5.2·Seguridad	30
4·Análisis:	31
5·Diseño	46
5.1·Servicio Web	47
5.2·Cliente de Escritorio (Ilustraciones 5.2 y 5.3)	48
5.3·Cliente Móvil (Ilustraciones 5.4, 5.5 y 5.6)	48
5.4·Base de Datos	49
6·Implementación	50
6.1·Servicio Web	50
6.2·Aplicación de Escritorio	52

6.3·Aplicación Móvil	53
7·Resultados	55
7.1·Aplicación de escritorio	55
7.2·Aplicación Móvil	55
7.3·Pruebas	56
8·Conclusiones y trabajo futuro	57
8.1·Conclusiones	57
8.2·Trabajo Futuro	57
9·Referencias y documentos consultados	59
9.1·Referencias	59
9.2·Paginas consultadas	59
9.3·Apuntes Asignaturas	59

Índice de ilustraciones

Ilustración 1.1: Distribución de las versiones de Android.....	7
Ilustración 2.1: IMK - App Movil.....	11
Ilustración 2.2: IMK - Arquitectura.....	11
Ilustración 2.3: Seaforms: Partes de Trabajo desde el Móvil.....	12
Ilustración 2.4: Ovelan - Gestión de partes de trabajo On-Line.....	12
Ilustración 4.1: Diagrama de Casos de Uso.....	31
Ilustración 5.1: Arquitectura del sistema.....	46
Ilustración 5.2: Boceto Aplicacion de Escritorio - Listado.....	48
Ilustración 5.3: Boceto Aplicación de escritorio – Edicion/Detalle.....	48
Ilustración 5.4: Boceto Android Login.....	48
Ilustración 5.5: Boceto Android Lista.....	48
Ilustración 5.6: Boceto Android Lista.....	49
Ilustración 5.7: Estructura de la Base de Datos.....	49
Ilustración 6.1: DataContract.....	50
Ilustración 6.2: Objeto Orden de Trabajo.....	50
Ilustración 6.3: AGIMComboboxItem.....	50
Ilustración 6.4: AGIMImagenItem.....	51
Ilustración 6.5: ServiceContract.....	51
Ilustración 6.6: Fragmento del archivo Web.config.....	51
Ilustración 6.7: Ventana de edición del formulario AXML.....	52
Ilustración 7.1: Formulario de creación de una Orden de Trabajo en la Aplicación de Escritorio.....	55
Ilustración 7.2: Pantalla de login.....	56
Ilustración 7.3: Listado de Ordenes.....	56
Ilustración 7.4:Detalle de Orden.....	56

Índice de tablas

Tabla 3.1: Listar Órdenes de Trabajo.....	23
Tabla 3.2: Filtrar Órdenes de Trabajo.....	23
Tabla 3.3: Eliminar Órdenes de Trabajo.....	24
Tabla 3.4: Crear una Orden de Trabajo.....	24
Tabla 3.5: Modificar una Orden de Trabajo.....	24
Tabla 3.6: Añadir/Eliminar/Re-ordenar Imágenes en una Ordenen de Trabajo.....	25
Tabla 3.7: Añadir Operarios a una Orden de Trabajo.....	25

Tabla 3.8: Eliminar Operarios de una Orden de Trabajo.....	26
Tabla 3.9: Identificarse.....	26
Tabla 3.10: Visualizar el listado de Órdenes Asignadas.....	27
Tabla 3.11: Acceder a Orden de Trabajo.....	27
Tabla 3.12: Acceder a Geolocalización.....	27
Tabla 3.13: Acceder a Imagen.....	28
Tabla 3.14: Modificar Orden.....	28
Tabla 3.15: Añadir Imágenes desde Archivos.....	28
Tabla 3.16: Añadir Imagen desde la Cámara.....	29
Tabla 3.17: Modificar Observaciones.....	29
Tabla 3.18: Modificar Ubicación.....	29
Tabla 4.1: Caso de uso: Listar Órdenes de Trabajo.....	32
Tabla 4.2: Caso de uso: Filtrar Órdenes de Trabajo.....	32
Tabla 4.3: Caso de uso: Eliminar Órdenes de Trabajo.....	33
Tabla 4.4: Caso de uso: Crear una Orden de Trabajo.....	34
Tabla 4.5: Caso de uso: Modificar una Orden de Trabajo.....	35
Tabla 4.6: Caso de uso: Añadir Imágenes en una Ordenen de Trabajo.....	36
Tabla 4.7: Caso de uso: Eliminar Imágenes en una Ordenen de Trabajo.....	36
Tabla 4.8: Caso de uso: Re-ordenar Imágenes en una Ordenen de Trabajo.....	37
Tabla 4.9: Caso de uso: Añadir Operarios a una Orden de Trabajo.....	37
Tabla 4.10: Caso de uso: Eliminar Operarios de una Orden de Trabajo.....	38
Tabla 4.11: Caso de uso: Identificarse.....	39
Tabla 4.12: Caso de uso: Visualizar el listado de Órdenes Asignadas.....	39
Tabla 4.13: Caso de uso: Acceder a Orden de Trabajo.....	40
Tabla 4.14: Caso de uso: Acceder a Geolocalización.....	41
Tabla 4.15: Caso de uso: Acceder a Imagen.....	41
Tabla 4.16: Caso de uso: Modificar Orden.....	42
Tabla 4.17: Caso de uso: Añadir Imágenes desde Archivos.....	43
Tabla 4.18: Caso de uso: Añadir Imagen desde la Cámara.....	43
Tabla 4.19: Caso de uso: Modificar Observaciones.....	44
Tabla 4.20: Caso de uso: Modificar Ubicación.....	44

1 Introducción

Dentro de las actividades que todo ayuntamiento realiza diariamente, se encuentran las labores de mantenimiento de los inmuebles y espacios municipales. Para ello cada consistorio tiene a su disposición un grupo humano más o menos extenso encargado de dicha labor.

Para municipios de cierta entidad, se suele organizar estos recursos en brigadas municipales que se pueden encargar de actividades tan variopintas como podar árboles o reparar desperfectos en las aceras. La organización que cada ayuntamiento realiza sobre estas actividades es diversa, pero como norma general suele haber un encargado dependiente de una concejalía, ya sea urbanismo o cualquier otra que asuma esta función.

El encargado suele recibir incidencias, ya sea de la policía, vecinos, etc. como del propio consistorio si se trata de realizar, por ejemplo mejoras o ampliaciones de las instalaciones actuales. Estas incidencias se procesan y a continuación se asigna su realización a uno o varios miembros de la brigada, que una vez realizada la intervención, informan de su finalización o, en caso de que no se haya podido realizar, del estado de la misma y los motivos que no han permitido llevarla a cabo.

Todo este proceso en el mejor de los casos queda reflejado en una base de datos donde se puede hacer un seguimiento de cada una de las órdenes y su estado actual, en el peor, se realizan a través de órdenes de trabajo, que si no son verbales, quedan archivadas en el propio ayuntamiento.

El caso concreto que este proyecto va a tratar es el del Ayuntamiento de Chiva, en el que podríamos decir que si bien las órdenes se almacenan digitalmente, la gestión de las mismas deja bastante que desear. En concreto la gestión se realiza a través de una aplicación de "Microsoft Access" que aparte de necesitar para su consulta de un sistema propietario, no permite, por la estructura de la propia base de datos, una gestión lo suficientemente completa de los datos que aloja.

Por otro lado, excepto la gestión de estos datos, el resto del proceso se realiza de manera manual. Se imprimen las órdenes de trabajo, las cuales son recogidas por un miembro de la brigada y a continuación distribuidas entre el resto de los integrantes dependiendo de las

habilidades y disposición de los mismos.

En este contexto se pretende un cambio completo del sistema, de modo que se simplifiquen las interacciones y la gestión de la información. Para ello, se pretende hacer uso de la tecnología que los dispositivos móviles ponen a nuestra disposición.

De este modo el proceso, en líneas generales, se simplificará al dar de alta en el sistema las órdenes de trabajo y enviarlas directamente al dispositivo de la persona o personas encargadas de realizarlas. A su vez, una vez resuelta, estas personas podrán directamente validar la finalización en el sistema.

Por otro lado la introducción de los dispositivos móviles añade ciertos aspectos de relevado interés como pueden ser, la geolocalización, la cual puede ayudar tanto en la asignación de tareas por proximidad a las mismas, como en su localización por parte de las brigada. O la realización de fotografías y su envío para identificar los objetivos o mostrar el estado de los mismos.

Para la realización del mismo se pretende implementar un sistema basado en tres capas. datos, lógica y presentación. Implementado a través de SOAP e interconectando diferentes sistemas como son Microsoft Windows y Android (por el momento, puesto que la flexibilidad que esta tecnología aporta hace relativamente sencillo extrapolarla a otros sistemas).

Dado que el ayuntamiento de Chiva ha realizado recientemente una inversión en equipos, incorporando un gran número de terminales con Microsoft Windows 8.1 (que probablemente se actualice en breve a Windows 10) y unos cuantos sistemas de virtualización de servidores con Windows Server 12, se pretende realizar la implementación de la capa lógica y de la de datos en este entorno.

Por otro lado la capa de presentación se realizará en equipos con el sistema operativo de Google, Android en versiones iguales o superiores a la 4.4

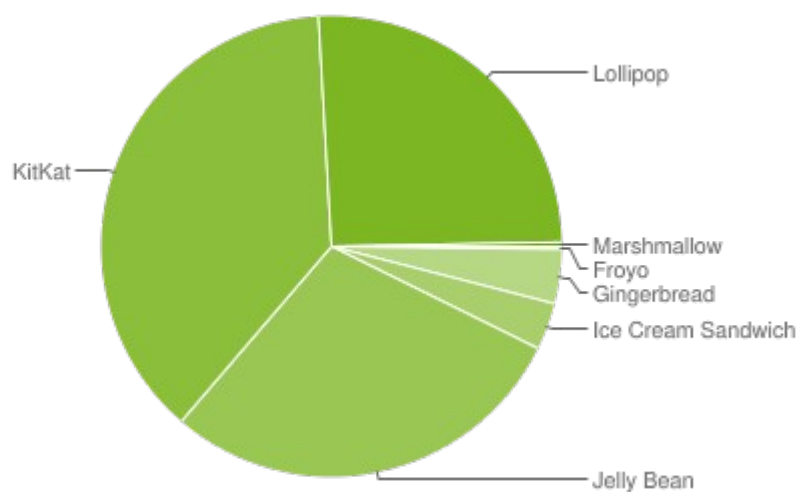


Ilustración 1.1: Distribución de las versiones de Android

(Kitkat), puesto que engloba a más de la mitad de los teléfonos en activo *(ver Ilustración 1.1)* y dado que actualmente los operarios no disponen de un smartphone de empresa y habría que adquirirlos, es el sistema que montan la gran mayoría de los terminales seleccionados.

Así pues el objetivo principal es crear la estructura que alojará los datos del sistema, crear un servicio web que se encargue de acceder a estos y comunicarse con los clientes que necesiten acceder a esta información, un cliente para Windows y por último el cliente Android.

Los datos, siguiendo la lógica de utilización de las infraestructuras existentes, se alojarán en una base de datos Microsoft SQL Server que actualmente corre en el mismo servidor que el servicio web, dado que únicamente se ha dado acceso a uno de los servidores, aunque no necesita encontrarse en la misma ubicación, si requiriría de la implementación de ciertas medidas de seguridad suplementarias para mantener el acceso externo restringido.

El servicio web, se implementará también en la plataforma de Microsoft a través del servicio WCF (Windows Communication Foundation) que facilita la creación de interfaces de comunicación para los servicios web de una forma rápida y sencilla. De este modo, será necesario crear interfaces tanto para consultar las incidencias existentes como para darlas de alta o modificarlas, según su estado vaya progresando. Así mismo también será necesario crear tipos de datos que engloben ciertas estructuras para facilitar su transporte por medio de estas interfaces, de modo que en una única transacción se pueda enviar, por ejemplo, una Orden de Trabajo que puede contener tipos muy diversos, como fechas, cadenas de texto o enteros.

La aplicación de Windows, se empleará básicamente para dar de alta Órdenes de trabajo así como para su consulta y modificación. En este caso se utilizará la estructura de aplicación orientada a la tienda de aplicaciones de Windows (Windows Store) cuyos formularios utilizan la misma estructura que los WPF (Windows Presentation Foundation) y que difieren sensiblemente de los tradicionales formularios de Windows, teniendo los primeros un aspecto más limpio y moderno. La elección de todos modos se basa, más que en el aspecto estético, en la posibilidad que esta plataforma ofrece para trasladar estas aplicaciones al sistema Windows Phone, lo que permitiría en un futuro ampliar el número de sistemas compatibles.

La aplicación Android ofrecerá la posibilidad de evitar pasos intermedios entre el alta de una

nueva incidencia y su entrega al operario asignado. Para ello la aplicación contará con un sistema de Login (Identificación) más para poder filtrar las órdenes de trabajo que por una verdadera necesidad de privacidad, ya que se trata de incidencias, que son públicas y de las que, en último término, cualquiera debería poder conocer su estado. Aunque no podemos obviar que pese a que las modificaciones que se podrán realizar son mínimas, debería controlarse su acceso.

Una vez identificado el operario tendrá acceso al listado de las órdenes activas en las que figure como personal asignado, en el que podrá ver un pequeño resumen de cada una y podrá seleccionar cualquiera de ellas para consultarla con más profundidad. Esto le llevará a una segunda pantalla en la que se observarán ciertos detalles como el encargado asignado a la misma, un listado con imágenes de la incidencia o la ubicación, a través de la cual podrá acceder a un mapa que permitirá su localización.

A su vez el encargado asignado a cada orden, podrá actualizar el estado de la misma conforme progresa la intervención, así como finalizarla, momento en el que se fijaría la fecha de finalización en el sistema

Dado que por la naturaleza de la propia actividad a la que está enfocada no se puede asegurar siempre la cobertura de la señal de datos, se deberá hacer un respaldo de las órdenes descargadas en el propio terminal, así como mantener esa copia al día cada vez que se vuelva a conectar, de manera que no solo las Órdenes de trabajo estén actualizadas sino que se asegure el borrado de las finalizadas, y por supuesto la inclusión de las nuevas.

En esta memoria, se tratará de presentar el proceso de realización de la solución y de las decisiones que se han tomado a la hora de acometer el proyecto. Para ello se desarrollará la descripción de cada uno de los pasos seguidos, estructurándolos en varias secciones que se corresponden con los capítulos que se exponen a continuación:

Aplicaciones similares: Se presentaran unas cuantas opciones existentes en el mercado para evaluar los enfoques que se ha seguido en su desarrollo. El principal punto de esta sección es, atendiendo a lo que se ha hecho hasta ahora, dotar al desarrollo de nuestra solución de una base sobre la que ir adaptando nuestros requerimientos.

Especificación de requisitos de Software: Una vez que sabemos lo que se ha hecho, debemos

averiguar qué necesitamos nosotros, de este modo se analizarán los requerimientos del cliente (en este caso el Ayuntamiento) así como las herramientas a las que tenemos acceso y el entorno en el que se desarrollará la actividad. También se introducirán ciertos conceptos que nos acompañarán durante el resto de la memoria.

Análisis: Partiendo de las necesidades extraídas en el apartado anterior, se pasará a describir como se integrarán estas en el sistema y se desglosará el procedimiento que, para cada una de ellas, seguirá este a la hora de procesarlas.

Diseño: Una vez que sabemos que necesitamos que haga nuestro sistema y cómo pretendemos que lo haga, deberemos estructurar la solución de manera que sea lo mas eficiente posible. Aquí describiremos como se distribuirá, que partes lo compondrán y como se comunicaran entre ellas.

Implementación: Esta es la parte mas práctica de todo el proceso. En ella se presentaran las soluciones concretas que se han llevado a cabo, por medio de pequeños fragmentos representativos del código utilizado.

Resultados: Finalmente se presentará el resultado final mostrando sus prestaciones y el aspecto final de los módulos que componen la solución.

Conclusiones y trabajo futuro: Con el trabajo completado es hora de pasar a analizar en conjunto el esfuerzo realizado y obtener una reflexión sobre lo que se ha sacado de esta experiencia. También se comentaran ciertas mejoras y ampliaciones que durante el proceso han ido surgiendo y que pese a que no podían ser incluidas por motivos de tiempo o complejidad, se consideran interesantes y dignas de futuras ampliaciones.

Referencias: Para acabar se incluyen referencias utilizadas en la realización de esta memoria, así como el material consultado.

2 Aplicaciones similares

El mercado de las aplicaciones de gestión, podría ser uno de los más copados en el mundo del desarrollo software. Tenemos desde aplicaciones para gestionar peluquerías, hasta soluciones de gestión logística para grandes empresas internacionales. El objetivo es obtener un producto que solucione una problemática concreta que aparentemente no se ajusta a las soluciones existentes.

En el sector de la gestión de tareas y grupos de trabajo existen en la actualidad varias aplicaciones que comparten rasgos de interés con la que se quiere generar. Es por ello que, con el fin de sacar ideas y enfocar el desarrollo de nuestra solución, se han analizado una serie de soluciones, de las que se muestran a continuación unos ejemplos.

2.1 IMK - App Partes de Trabajo *(Ilustración.2.1)*

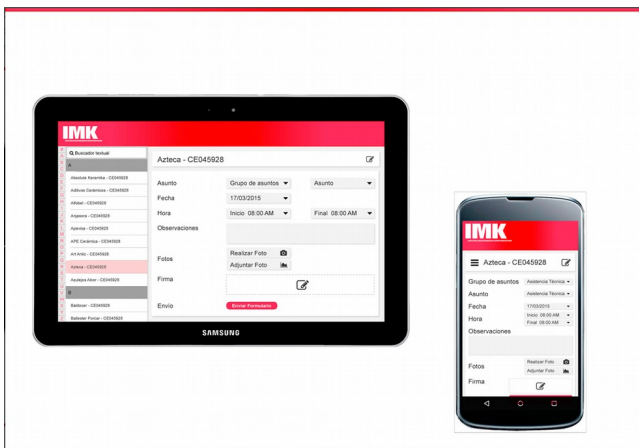


Ilustración 2.1: IMK - App Movil

Se trata de una aplicación móvil que gestiona partes de trabajo y asistencia técnica. Tiene un cometido bastante similar al que se pretende aspirar con este proyecto.

La arquitectura también es similar al separar la capa de presentación, que se ejecutaría en el móvil, de la lógica que se ejecuta en el equipo de las oficinas y de la base de datos que parece ser independiente.

Por otro lado está demasiado enfocada al ámbito de la asistencia técnica unipersonal, no parece que puedan manejarse equipos de trabajo, y tampoco se hace mención a la geolocalización. Tampoco parece que exista un cliente de escritorio, sino que la aplicación central se ejecute completamente en el equipo del administrador.

<http://www.imk.es/2015/10/app-partes-de-trabajo/>

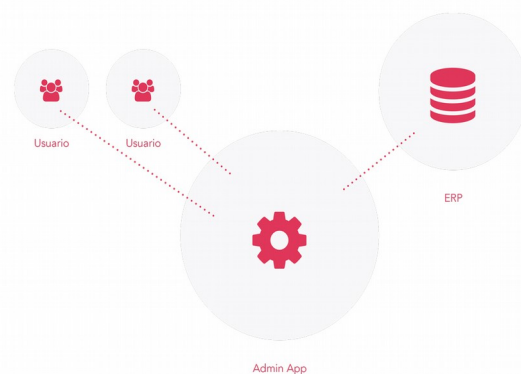
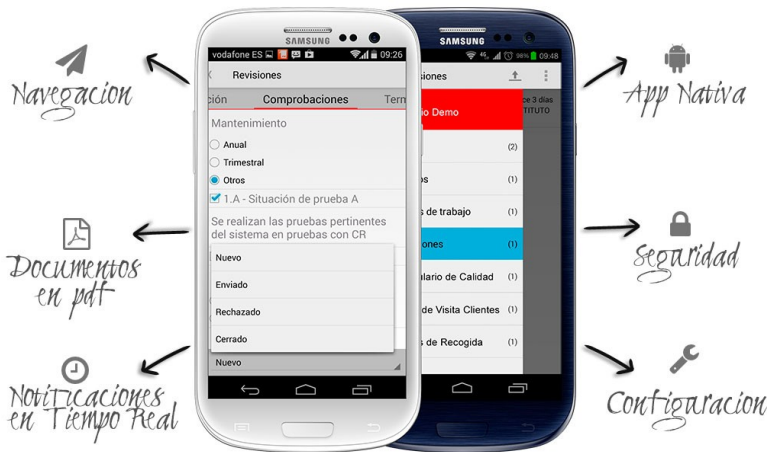


Ilustración 2.2: IMK - Arquitectura

2.2 Seaforms: Partes de Trabajo desde el Móvil (Ilustración:2.3)



Esta aplicación está enfocada básicamente a manejar el papeleo que supone acarrear y rellenar los partes de trabajo en desplazamientos. Permite gestionar carpetas de documentos y rellenarlos, pudiéndolos firmar el cliente. También incluye notificaciones, lo cual es un punto a tener en cuenta.

Ilustración 2.3: Seaforms: Partes de Trabajo desde el Móvil

<http://eseaforms.com/partes-de-trabajo>

2.3 Ovelan - Gestión de partes de trabajo On-Line (Ilustración:2.4)

Esta se ha escogido por la geolocalización de los empleados. Estos pueden ser localizados a través del móvil mediante una aplicación web. El mayor problema es que la aplicación principal no está en la propia empresa, si no que es propiedad de los desarrolladores, los cuales proporcionan el acceso a la misma.



Ilustración 2.4: Ovelan - Gestión de partes de trabajo On-Line

<http://www.ovelan.es/control-localizacion-empleados.php>

2.4 Conclusiones

Hay varias aplicaciones en el mercado, pero ninguna se ajusta a las especificaciones que el Ayuntamiento ha marcado, ni a las que parecen lógicas en una aplicación como la que se quiere crear, como por ejemplo la gestión de grupos de trabajo.

Sí que se han observado puntos a tener en cuenta como la arquitectura, la geolocalización o la posibilidad de adjuntar imágenes.

3 Especificación de requisitos de Software

3.1 Introducción

La especificación de requisitos se realizó a partir de reuniones con el encargado de la introducción de los partes de incidencias, así como con el responsable del departamento de informática del Ayuntamiento.

3.1.1 Propósito

Marcar las necesidades del sistema y el alcance del mismo, a fin de delimitar los objetivos que se deberían de completar.

3.1.2 Alcance del sistema

El proyecto pretende solucionar el problema de gestión que el actual sistema genera, así como aportar ciertas herramientas adicionales, de manera que el ritmo de trabajo sea mas fluido y se pueda conseguir una comunicación mas eficiente

3.1.3 Definiciones

AGIM

Aplicación de Gestión de Incidencias Municipales, es el sistema que se pretende crear para simplificar las interacciones del **Administrador** del registro de **Órdenes de Trabajo** a la hora de dar de alta estas y asignarlas al **Encargado** de un grupo de **Operarios**.

Incidencia

Se considerará incidencia cualquier contingencia que requiera la actuación de los operarios de la brigada municipal, ya sea para reparar, eliminar, construir... Cada incidencia irá asociada a una ubicación y una fecha y se verá reflejada en la **Orden de Trabajo** que se generara para darle solución. No tiene entidad de manera independiente.

Orden de Trabajo

Cada incidencia dará lugar a una Orden de Trabajo que contendrá, tanto información sobre la incidencia, como sobre los materiales necesarios para solventarla, el ámbito de actuación o el grupo de trabajo asignado a la misma. Sera dada de alta por el **Administrador**.

Administrador

Es la persona encargada de introducir las incidencias en el sistema y generar las

Órdenes de trabajo, así mismo se encarga de seleccionar el personal a asignar a las mismas: **Operarios** y responsable del equipo, **Encargado**.

Encargado

Es la persona responsable del grupo de trabajo asignado a una incidencia, tiene cierta capacidad de decisión sobre la forma de actuar en la misma, por lo que deberá poder editar algunos aspectos de las Órdenes de trabajo, pese que no tendrá acceso a la totalidad de su contenido.

Operarios

Son los integrantes del grupo de trabajo, pudiendo el Encargado ser uno de ellos o no. No pueden modificar las Órdenes de trabajo, y únicamente podrán acceder a parte de su contenido.

Aplicación Móvil

Una aplicación es un tipo de programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos trabajos. En este caso la aplicación se ejecutará en un dispositivo móvil (*smartphone*, tableta...), teniendo que adaptarse a las características de estos, como pueden ser los controles táctiles o las reducidas dimensiones de sus pantallas, pero también aprovechar sus ventajas, geolocalización, disponibilidad, cámara... Para este proyecto, se utilizara **Android**.

Android

Se trata de un sistema operativo basado en Linux y desarrollado por la empresa Android Inc. (inicialmente apoyada por Android para mas tarde pasar a comprarla) y diseñado principalmente para ejecutarse en dispositivos móviles con pantallas táctiles.

Aplicación de Escritorio

Al igual que lo comentado anteriormente, en este caso la aplicación se ejecutaría en un ordenador personal, concretamente sobre el sistema operativo **Microsoft Windows**.

Servicio Web (WebService)

El término describe una forma estandarizada de integrar aplicaciones WEB mediante el uso de **XML**, **SOAP**, **WSDL** y **UDDI** sobre los protocolos de la Internet. XML es usado para describir los datos, SOAP se ocupa para la transferencia de los datos, WSDL se emplea para describir los servicios disponibles y UDDI se ocupa de conocer cuales son los servicios disponibles. Uno de los usos principales es permitir la comunicación entre

las empresas y entre estas y sus clientes. Los Web Services permiten a las organizaciones intercambiar datos sin necesidad de conocer los detalles de sus respectivos Sistemas de Información.

A diferencia de los modelos Cliente/Servidor, tales como un servidor de páginas Web, los Web Services no proveen al usuario una interfaz gráfica (GUI). En vez de ello, los Web Services comparten la lógica del negocio, los datos y los procesos, por medio de una interfaz de programas a través de la red. Es decir conectan programas, por tanto son programas que no interactúan directamente con los usuarios. Los desarrolladores pueden por consiguiente agregar a los Web Services la interfaz para usuarios, por ejemplo mediante una página Web o un programa ejecutable, con el fin de entregarle a los usuarios un funcionalidad específica que provee un determinado Web Service.

XML [1]

EXtensible Markup Language, Es un formato de texto, simple y muy versátil, derivado del SGML (*Standard Generalized Markup Language*, ISO 8879). Diseñado originalmente para enfrentarse a los retos que ofrecía la publicación electrónica a gran escala, XML esta jugando un papel de creciente importancia en el intercambio de una amplia variedad de datos tanto en la web como en otros lugares.

WSDL [2]

Se trata de un formato XML que describe el Servicio Web como un conjunto de *endpoints* operando sobre *mensajes* que contienen información bien orientada a datos o bien orientada a procedimientos. Las operaciones y mensajes son descritos de manera abstracta para, posteriormente, ligarlas a un protocolo de red concreto y a un formato de mensaje para definir un *endpoint*. Los *endpoints* concretos relacionados son combinados in *endpoints* abstractos (servicios). WSDL se puede ampliar para permitir describir endpoints y mensajes independientemente de qué formato de mensaje o protocolo de red sean usados para comunicarse.

Endpoint

Es un extremo del sistema y es el punto de entrada y/o salida del servicio. El *endpoint* de un servicio puede ser, tanto parte de un sistema constantemente disponible hospedado en un IIS (*Internet Information Service*), como tratarse de un servicio alojado en una aplicación. Un endpoint puede ser un cliente de un servicio al que solicita datos

a través de su endpoint.

SOAP [3]

Se trata de un protocolo ligero para el intercambio de información en un entorno distribuido y descentralizado. Es un protocolo basado en XML que consta de tres partes

- Un Envoltorio (*envelope*) que define una infraestructura (*framework*) que describe qué contiene un mensaje y como debe ser procesado.
- Un conjunto de reglas para expresar las instancias de los tipos de datos (*datatypes*) definidos por la aplicación.
- Y una convención de cómo deben ser representadas las llamadas y respuestas remotas a procedimientos.

SOAP puede, potencialmente, ser usado en combinación con una gran variedad de protocolos.

Ksoap [4]

Dado que Android no integra nativamente el protocolo SOAP, necesitaremos añadir esta librería para poder hacer uso de la comunicación con servicios webs basados en esta tecnología.

Mensajes

Los mensajes transmitidos por el servicio pueden ser tan simples como un sólo carácter o palabra enviado como un XML, o tan complejos como un flujo de datos binarios.

WCF [5]

Windows Communication Foundation, Es una plataforma para construir aplicaciones orientadas a servicios. Usando WCF, se puede enviar datos como mensajes asíncronos desde el extremo de un servicio (*endpoint*) al de otro.

SQL [6]

Lenguaje de Consultas Estructurado (**Structured Query Language**), hace referencia a un lenguaje de programación de propósito específico diseñado para manipular información contenida en un sistema de manipulación de bases de datos (RDBMS, *Relational Database Management System*) o para el proceso de flujos, en un sistema de manipulación de flujos de datos relacionales (RDSMS, *Relational Data Stream Management System*). Permite especificar diversos tipos de operaciones en ellas. Una

de sus características es el manejo del álgebra y el cálculo relacional que permiten efectuar consultas con el fin de recuperar, de forma sencilla, información de bases de datos, así como hacer cambios en ellas.

Microsoft SQL Server

Se trata de un sistema de manipulación de bases de datos relacionales desarrollado por Microsoft. Como servidor de bases de datos, es un *software*, cuya función principal es almacenar y distribuir datos a petición de otras aplicaciones de *software*, lo cual puede ser realizado tanto de forma local como a través de la red, esto incluye Internet.

SQLite

Es un sistema de manipulación de bases de datos contenido en una librería de programación C. A diferencia de otros muchos sistemas de manipulación de bases de datos, SQLite no es un motor de bases de datos cliente-servidor, sino que está contenido en el programa final.

Google Maps (API)

Google Maps es un servidor de aplicaciones de mapas en la web que pertenece a Google. Ofrece imágenes de mapas desplazables, así como fotografías por satélite del mundo e incluso la ruta entre diferentes ubicaciones, o imágenes a pie de calle con Google Street View. La interfaz del servicio utiliza JavaScript, XML y Ajax. Google ofrece una API que permite alojar sus mapas en páginas de terceros.

API

Application Programming Interface (Interfaz de programación de aplicaciones) es un conjunto de rutinas, protocolos y herramientas para la creación de aplicaciones software. Una API describe un componente de software en términos de sus operaciones, entradas, salidas y tipos de datos subyacentes.

3.1.4 Visión global

El sistema que se pretende crear es, básicamente, una plataforma para la introducción y gestión de Incidencias. Para ello el proyecto se construirá partiendo de tres piezas diferenciadas. Por un lado una aplicación por medio de la que se puedan introducir, listar y modificar Órdenes de Trabajo. Cada Incidencia generará una Orden de Trabajo y esta se

asignará a un grupo de Operarios que serán los encargados de darle solución. Por otro una aplicación móvil que permita a estos Operarios visualizar las Órdenes a las que han sido asignados y, en el caso del Encargado, interactuar con las mismas. Por último, un servicio web que se encargue de gestionar estos procesos de forma centralizada. A esto habría que añadir la base de datos, donde se almacenará toda la información.

La idea general es dejar toda la lógica de negocio fuera de las aplicaciones que hacen de interfaz del usuario, para ello estará implementada en el servidor web con el que se comunicarán estas, cuya lógica se limitará a la propia del funcionamiento de la interfaz y la necesaria para realizar la conexión con el servicio, empaquetar los mensajes y enviarlos o recibirlos. Por motivo de las características propias de los dispositivos móviles, estos tendrán que mantener una pequeña base de datos puesto que no se puede asegurar el acceso constante a la red en esos dispositivos, más si tenemos en cuenta la tipología del sector en el que se trabaja, ya que las brigadas se están continuamente desplazándose y pueden, en muchos casos encontrarse en zonas con poca o nula cobertura.

Así pues, tenemos que el servidor alojara la lógica de negocio, encargándose de tramitar todas las peticiones y realizar las consultas o la introducción o modificación de información en la base de datos.

El producto va dirigido a personal que no tiene porqué poseer conocimientos mínimos de informática o incluso ofimática, a excepción del administrador que no deja de ser personal administrativo del Ayuntamiento. Por lo tanto el manejo de las interfaces debe ser lo mas sencillo e intuitivo posible.

3.2 Descripción general

3.2.1 Perspectiva del producto

Como ya se ha explicado anteriormente el sistema AGIM se dividirá en tres partes:

3.2.1.1 Servicio Web

El servicio web, se alojará en los servidores del Ayuntamiento, se trata de varios servidores virtualizados sobre los que corre Microsoft Windows Server 2012 R2. Es por ello que se ha elegido utilizar el protocolo WCF que, desarrollado a través de Microsoft Visual Studio, permite de manera sencilla desplegar un servicio SOAP simplemente

declarando los métodos y los datos que se van a utilizar. Esto nos permite generar la interfaz y desplegar los servicios, aunque solamente se trata de una interfaz y aun deberemos implementar la lógica.

Otra de las ventajas de utilizar este protocolo es que está completamente integrado con la *suite* de desarrollo con la que también programaremos la aplicación de escritorio. No es tan sencillo integrarla en dispositivos Android puesto que este sistema no tiene soporte nativo para SOAP, pero por suerte hay bibliotecas que nos permiten hacerlo en nuestro proyecto e incluso herramientas que, a partir de un archivo WSDL, nos permiten generar los métodos para realizar las comunicaciones.

3.2.1.2 Aplicación Móvil

La aplicación móvil se desarrollará en Android para terminales que tengan instalada como mínimo la versión 4.4(Kitkat), puesto que no merece la pena lidiar con el problema que supone la retro-compatibilidad debida a la fragmentación del mercado. Esto se debe a que los equipos serían de nueva adquisición y la oferta en este tipo de terminales está enfocada principalmente a versiones iguales o superiores a la escogida. Se hará uso de las tecnologías que estos dispositivos aportan como puedan ser las redes móviles, la geolocalización o la toma de fotografías. Dadas las limitaciones que esta tecnología presenta en materia de interacción con la interfaz gráfica, por ser dispositivos primordialmente táctiles, la información deberá ser presentada de manera sencilla y clara, facilitando el control de los elementos que se muestren en pantalla.

3.2.1.3 Aplicación de Escritorio

Para esta aplicación se ha escogido el nuevo formato de formularios de Microsoft, conocidos como WPF (*Windows Presentation Foundation*), para crear aplicaciones con interfaz **Modern UI**. Están basados en C# y una variante de XML bautizada por Microsoft como AXML (*EXtensible Application Markup Language*). La elección se ha realizado, no tanto por razones estéticas, como porque se ha decidido utilizar el desarrollo de Aplicaciones Universales de Microsoft, puesto que ofrece la posibilidad de, mas adelante, portarlo a la plataforma Windows Phone. Esto por otro lado supone el inconveniente de migrar a la nueva API Windows Runtime que, aunque ofrece una mayor integración con lenguajes como JavaScript o estilos CSS, introduce ciertos

cambios en las librerías soportadas por la plataforma .Net anterior.

3.2.2 Funciones del producto

3.2.2.1 Funciones del Servicio Web

Identificar al Usuario

Creación y Gestión de Órdenes

Obtener una Orden de Trabajo concreta

Obtener la lista de Órdenes de Trabajo para un usuario concreto

Insertar una Orden de Trabajo

Modificar una Orden de Trabajo

Asignar uno o varios Operarios a una Orden

Inserción y listado de Imágenes

Añadir una o varias Fotografías a una Orden

Re-ordenar la lista de Fotografías de una Orden

Obtención de información para la generación de menús

Obtener la lista de Áreas de Operación

Obtener la lista de los posibles Estados de una Incidencia

Obtener la lista de Operarios con el rol de Encargados

Obtener la lista de Tipos de Incidencia

Obtener la lista de clasificación por Prioridades

Obtener la lista de Tipos de Incidencia

3.2.2.2 Funciones de la Aplicación Móvil

Identificar al usuario

Listar Órdenes de Trabajo asignadas

Mostrar una Orden concreta con información detallada

Modificar campos de la Orden (**Encargado**)

Capturar y enviar imágenes (**Encargado**)

Geolocalizar y mostrar en mapa la Incidencia (**Encargado**)

3.2.2.3 Funciones de la Aplicación de Escritorio

Listar Órdenes de Trabajo

Filtrar Órdenes de Trabajo

Crear Orden de Trabajo

Modificar/Eliminar Orden de Trabajo

Geolocalizar incidencia

Añadir Imágenes.

3.2.3 Características del Usuario

Administrador

Tiene acceso a la aplicación de escritorio y puede crear, modificar y eliminar Órdenes de trabajo. Tiene también acceso al listado total de Órdenes de trabajo.

Encargado

Tiene acceso tanto a las Órdenes a las que ha sido asignado como a aquellas en las que figura como Encargado. Tiene acceso únicamente a las Órdenes que no estén finalizadas. Puede modificar ciertos aspectos de la Orden como el estado de la misma (añadiendo la fecha de finalización si se asigna el estado finalizado), modificar la geolocalización (para ajustarla sobre el terreno) o añadir imágenes (para informar sobre la evolución, problemas...).

Operario

Tiene acceso a las Órdenes de Trabajo a las que a sido asignado y que se encuentren actualmente en curso. No puede realizar ninguna modificación sobre estas.

3.2.4 Restricciones generales

Por el momento no se ha contemplado ninguna restricción en la aplicación de Escritorio, pues la conexión se hará de modo local. En la aplicación móvil se restringirá el acceso a funciones según el rol del usuario y su asignación a cada Orden de Trabajo.

3.2.5 Supuestos y dependencias

Supondremos que sólo tienen acceso a las aplicaciones los equipos en los que hayan sido instaladas directamente por el departamento de informática del ayuntamiento, de manera que no pueda haber accesos indeseados al sistema.

Los Dispositivos móviles habrán de tener conexión de datos y tecnología de geolocalización GPS. Además deberán ejecutar una versión de Android igual o superior a la 4.4 (Kitkat).

3.3 Requisitos específicos

3.3.1 Requisitos interfaces externas

3.3.1.1 Requisitos Interfaz Operario

Por ser un dispositivo móvil, la interfaz debe presentar un aspecto simple y claro ya que se visualizará en pantallas de reducidas dimensiones. En este caso además se va a utilizar el estilo *Material Design*, basado en bloques de colores planos y aspecto minimalista. También se debe tener en cuenta que en estos dispositivos la interacción se realiza a través de pantallas táctiles por lo que los controles deben ser sencillos e intuitivos para facilitar el acceso con simples gestos de los dedos.

3.3.1.2 Requisitos interfaz Administrador

La interfaz de Administrador no tiene ningún requerimiento especial a priori. A pesar de ello se ha decidido utilizar el estilo *Modern UI* y el desarrollo a partir de *Windows Runtime* con el fin de avanzar una posible implementación posterior en la plataforma *Windows Phone*.

3.4 Requisitos Funcionales

3.4.1 Operaciones permitidas para el Administrador

Tabla 3.1: Listar Órdenes de Trabajo

Listar Órdenes de Trabajo	
Introducción	Listar todas las Órdenes registradas en el sistema. Esta sera la opción por defecto si no se fija ningún filtro, ordenándose descendentemente por fecha de introducción.
Entradas	No hay entradas.
Proceso	Abrir la aplicación, Actualizar, quitar filtros,o retornar al formulario de lista
Salida	Lista de todas las Órdenes en formato <i>Grid</i> .

Tabla 3.2: Filtrar Órdenes de Trabajo

Filtrar Órdenes de Trabajo	
Introducción	Muestra un listado de las Órdenes que cumplen un determinado requisito
Entradas	Dependiendo del filtro: Encargado, Operario, Área...
Proceso	Pulsar el botón de añadir filtro, escoger uno de los que se ofrece y rellenar con el valor a filtrar.
Salida	Lista de todas las Órdenes que cumplen los criterios en formato <i>Grid</i>

Tabla 3.3: Eliminar Órdenes de Trabajo

Eliminar Órdenes de Trabajo	
Introducción	Permite seleccionar una o mas Órdenes del listado, para posteriormente eliminarlas del sistema.
Entradas	Lista de Filas
Proceso	Seleccionar las filas a eliminar, pulsar el botón de eliminar, pulsar el botón de aceptar en el cuadro de dialogo de confirmación.
Salida	Ventana de confirmación del éxito de la eliminación.

Tabla 3.4: Crear una Orden de Trabajo

Crear una Orden de Trabajo	
Introducción	Genera una nueva Orden de Trabajo. También puede adjuntar imágenes a esta y geolocalizarla en un mapa.
Entradas	Cada uno de los campos de La orden de Trabajo, Descripción Fechas, Encargado...
Proceso	Desde el listado de Órdenes de Trabajo, pulsar el botón de creación rellenar los campos obligatorios, y los opcionales que se desee, seleccionar, en caso de que haya, las imágenes a adjuntar y la geolocalización si se conoce. Pulsar el botón de guardado.
Salida	Ventana de confirmación del éxito de la operación

Tabla 3.5: Modificar una Orden de Trabajo

Modificar una Orden de Trabajo	
Introducción	Modifica uno o varios campos de la orden de trabajo. También puede añadir, borrar o re-ordenar las imágenes adjuntas a esta o modificar la geolocalización.
Entradas	Las mismas que en el formulario de creación.

Proceso	Desde el listado de Órdenes de Trabajo, seleccionar la orden a modificar. Pulsar el botón de modificar, alterar los campos a modificar, el mapa o la selección de imágenes.
Salida	Ventana de confirmación del éxito de la operación

Tabla 3.6: Añadir/Eliminar/Re-ordenar Imágenes en una Ordenen de Trabajo

Añadir/Eliminar/Re-ordenar Imágenes en una Ordenen de Trabajo	
Introducción	En los formulario de creación o modificación se podrá seleccionar imágenes para adjuntar a la Orden de Trabajo.
Entradas	Lista de Objetos AGIMImageltem que contendrán, índice, nombre e imagen.
Proceso	Desde el formulario de Creación/Modificación, pulsar el botón de selección de imágenes, o Seleccionarlas de la lista para ordenarlas o eliminarlas
Salida	Grid con las imágenes actualizadas. En caso de fallo ventana de error.

Tabla 3.7: Añadir Operarios a una Orden de Trabajo

Añadir Operarios a una Orden de Trabajo	
Introducción	En los formularios de Creación o Modificación, podremos añadir un listado de operarios asignados a una determinada Orden de Trabajo.
Entradas	Índice del Operario seleccionado.
Proceso	Desde el formulario de Creación/Modificación, seleccionar un operario del desplegable, pulsar el botón de añadir.
Salida	El listado se actualizará para reflejar los cambios.

Tabla 3.8: Eliminar Operarios de una Orden de Trabajo

Eliminar Operarios de una Orden de Trabajo	
Introducción	En los formularios de Creación o Modificación, podremos eliminar uno o varios Operarios asignados a una determinada Orden de Trabajo.
Entradas	Lista con los índices correspondientes a los Operarios seleccionados.
Proceso	Desde el formulario de Creación/Modificación, seleccionar un Operario del listado, pulsar el botón de eliminar.
Salida	El listado se actualizará para reflejar los cambios.

3.4.2 Operaciones permitidas para el Operario

Tabla 3.9: Identificarse

Identificarse	
Introducción	El Encargado deberá, al menos la primera vez, identificarse en el dispositivo con su nombre de usuario y contraseña. Cuando el sistema lo reconozca como un usuario válido quedará identificado en dicho dispositivo pudiendo acceder a la información y las operaciones a las que este autorizado
Entradas	usuario y contraseña
Proceso	Introducir el nombre de usuario y la contraseña en sus campos correspondientes y pulsar el botón de envió.
Salida	Acceso a el listado o, en caso de fallo, ventana de error.

Tabla 3.10: Visualizar el listado de Órdenes Asignadas

Visualizar el listado de Órdenes Asignadas	
Introducción	Todo usuario Identificado, accede directamente a este listado, donde se mostrarán las Órdenes de Trabajo que tiene asignadas
Entradas	Identificador del Encargado
Proceso	Acceder a la aplicación, identificarse o pulsar el botón de refresco
Salida	Lista Actualizada o, en caso de fallo, ventana de error y la última lista válida obtenida.

Tabla 3.11: Acceder a Orden de Trabajo

Acceder a Orden de Trabajo	
Introducción	Desde la lista de Órdenes se puede navegar hasta encontrar la Orden deseada y seleccionarla para su edición.
Entradas	Identificador de la Orden de Trabajo
Proceso	Desde la lista de Órdenes, pulsar la orden seleccionada.
Salida	Ventana de detalle de Orden o notificación de error.

Tabla 3.12: Acceder a Geolocalización

Acceder a Geolocalización	
Introducción	Desde la página de detalle de Orden, se podrá acceder a un mapa que reflejará la ubicación de la incidencia así como la del propio usuario.
Entradas	Coordenadas de la incidencia y del usuario
Proceso	Desde la página de detalle de Orden, pulsar el botón Mapa situado al lado de la dirección.
Salida	Pantalla de Mapa o, en caso de fallo, notificación de error

Tabla 3.13: Acceder a Imagen

Acceder a Imagen	
Introducción	Desde la página de detalle de Orden se podrá acceder a un listado de las imágenes vinculadas a la orden seleccionada.
Entradas	Identificador de la Orden
Proceso	Desde la página de detalle de Orden, Pulsar el botón de imágenes.
Salida	Página con el listado de imágenes o, en caso de fallo, notificación de error.

3.4.3 Operaciones permitidas para el Encargado

Posee las propias del Operario añadiéndosele las siguientes:

Tabla 3.14: Modificar Orden

Modificar Orden	
Introducción	Modificar ciertos campos y propiedades de la Orden.
Entradas	Objeto del tipo OrdenTrabajo. Lista de Objetos del tipo AGIMImageltem (opcional)
Proceso	Pulsar el botón con el icono de editar (lápiz)
Salida	Se habilitarán los campos de edición.

Tabla 3.15: Añadir Imágenes desde Archivos

Añadir Imágenes desde Archivos	
Introducción	Durante el proceso de modificación de la Orden, se podrá añadir imágenes a esta desde la memoria del dispositivo.
Entradas	Lista de archivos de imagen.
Proceso	Pulsar el botón "Escoger Imágenes" y seleccionar las imágenes pertinentes desde la memoria del dispositivo. Tras realizar las modificaciones pulsar el botón de enviar.

Salida	Lista de las imágenes a incluir actualizada o, en caso de fallo, notificación de error.
--------	---

Tabla 3.16: Añadir Imagen desde la Cámara

Añadir Imagen desde la Cámara	
Introducción	Durante el proceso de modificación de la Orden, se podrá tomar imágenes con la cámara del dispositivo y añadirlas .
Entradas	Archivo de imagen.
Proceso	Pulsar el botón "Capturar Imagen" y proceder a realizar la fotografía que se desee incluir.
Salida	Lista de las imágenes a incluir actualizada o, en caso de fallo, notificación de error.

Tabla 3.17: Modificar Observaciones

Modificar Observaciones	
Introducción	Durante el proceso de modificación de la Orden, se podrá ampliar el contenido del campo observaciones a fin de ampliar información.
Entradas	Texto.
Proceso	Pulsar sobre el campo observaciones e introducir el texto deseado.
Salida	Campo de texto actualizado

Tabla 3.18: Modificar Ubicación

Modificar Ubicación	
Introducción	Durante el proceso de modificación de la Orden, se podrá modificar la ubicación de la incidencia.
Entradas	Coordenadas actuales, Lat, Long de tipo Double.
Proceso	Pulsar el botón mapa y en la página de la ubicación desplazar el marcador para ajustar su posición, pulsar el botón Hecho

Salida

Notificación confirmando el éxito o fracaso de la operación.

3.5 Rendimiento y Atributos

3.5.1 Rendimiento

El rendimiento no es un factor prioritario, puesto que el cometido no reviste ninguna urgencia ni el tipo de cometido puede consumir excesivos recursos. Sí que es cierto que, como toda aplicación que deba tener un uso cotidiano, debe presentar cierta eficiencia y velocidad, lo cual en la aplicación de escritorio no supone un problema y en la de la plataforma móvil se solventará con copias internas para eventuales pérdidas de conexión o ralentización del servicio. Otro factor a tener en cuenta es la memoria interna de los dispositivos móviles por lo que habrá que gestionar el almacenamiento de imágenes para que sean borradas una vez la Incidencia deje de estar activa.

3.5.2 Seguridad

La seguridad se ha dejado por ahora de lado, puesto que por una parte el Ayuntamiento no pretende poner en marcha el servicio en un futuro cercano, y aún no se ha planteado como quieren afrontar este aspecto. Y por otro, no menos importante, porque las librerías que permitirán a la aplicación ejecutar SOAP en Android no tienen muy avanzada la implementación de los protocolos de SOAP 1.2 el cual permite implementar *WsHttpBinding* que habilita el encriptado de mensajes [7].

4 Análisis:

Diagrama de Casos de Uso (Ilustración 4.1)



Ilustración 4.1: Diagrama de Casos de Uso

Se ha considerado tres tipos de usuarios: Administrador, el que poseerá mayores privilegios y únicamente actuará en el ámbito de la aplicación de escritorio y los usuarios de la aplicación móvil; el Operario, que no podrá realizar ninguna modificación y únicamente accederá a los datos que le correspondan; y el Encargado, que hereda los atributos del operario (no deja de ser uno de ellos) y se le añaden algunos privilegios que le permiten realizar ciertas modificaciones a las Órdenes.

Tabla 4.1: Caso de uso: Listar Órdenes de Trabajo

Caso de uso: Listar Órdenes de Trabajo	
Resumen	Mostrar el listado de todas las Órdenes del sistema (En realidad una primera página con las 'x' ultimas) sin filtrar.
Actor	Administrador
Precondición	Haber accedido a la aplicación desde una ubicación con acceso local al servidor y que este se encuentre activo
Descripción	Flujo Básico: 1) Acceder a la aplicación 2) Accede al listado de Órdenes Flujo Alternativo: 2) En caso de fallo, Mostrar ventana de Error
Postcondición	Se mostrarán, además del listado, las operaciones disponibles

Tabla 4.2: Caso de uso: Filtrar Órdenes de Trabajo

Caso de uso: Filtrar Órdenes de Trabajo	
Resumen	Mostrar un listado de todas las Órdenes que cumplen un determinado requisito
Actor	Administrador
Precondición	Acceder al listado e introducir un criterio válido de filtrado desde una ubicación con acceso local al servidor y que este se encuentre activo

Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Seleccionar un criterio 2) Seleccionar añadir 3) Accede al listado de Órdenes filtrado <p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 3) En caso de fallo el listado se mostrará vacío (El caso de error en esta operación, sería un fallo de implementación)
Postcondición	Las Órdenes listadas cumplirán con los criterios seleccionados

Tabla 4.3: Caso de uso: Eliminar Órdenes de Trabajo

Caso de uso: Eliminar Órdenes de Trabajo	
Resumen	Eliminar las Órdenes de Trabajo seleccionadas
Actor	Administrador
Precondición	Haber seleccionado al menos una Orden de Trabajo desde una ubicación con acceso local al servidor y que este se encuentre activo
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Seleccionar una Orden 2) Pulsar el botón de Eliminar 3) Confirmar en la ventana emergente 4) El listado se actualizará para dejar de mostrar las Órdenes eliminadas <p>Flujo Alternativo:</p> <p>1.1) Seleccionar otra manteniendo pulsada la tecla 'Control' y volviendo al paso 1</p> <p>Flujo Alternativo 2:</p> <p>1.1) Seleccionar varias contiguas pulsando la tecla 'Mayúscula' mientras se ejecuta el punto 1 sobre la última orden a seleccionar</p> <p>Flujo Alternativo 3:</p> <p>4) En caso de fallo el listado se mantendrá inalterado y se mostrará una ventana de error</p>

Postcondición	El listado no contendrá las Órdenes eliminadas y por lo tanto no se podrá operar sobre ellas. La base de datos se alterará para eliminar la Orden de Trabajo
---------------	--

Tabla 4.4: Caso de uso: Crear una Orden de Trabajo

Caso de uso: Crear una Orden de Trabajo	
Resumen	Crear una nueva Orden de Trabajo y guardarla en la base de datos, así como las relaciones de Operarios asignados o de las imágenes adjuntas
Actor	Administrador
Precondición	Acceder a la aplicación desde una ubicación con acceso local al servidor y que este se encuentre activo
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Pulsar el botón de 'Nueva Orden' 2) Rellenar los campos de Descripción y Fecha de alta 3) Pulsar Guardar 4) El listado se actualizará para mostrar la nueva Orden <p>Flujo Alternativo:</p> <p>2.1) Rellenar los campos restantes que se desee</p> <p>Flujo Alternativo 2:</p> <p>2.1) Añadir Operarios</p> <p>Flujo Alternativo 3:</p> <p>2.1) Añadir Imágenes</p> <p>Flujo Alternativo 4:</p> <p>2.1) Cualquier combinación de los anteriores flujos alternos</p> <p>Flujo Alternativo 5</p> <p>4) <i>En caso de fallo se mostrará una ventana de error y se permanecerá en la ventana de creación de Orden</i></p>
Postcondición	El Listado mostrará a partir de ese momento la nueva Orden de Trabajo pudiéndose operar sobre ella. La Orden se incluirá en la base de datos

Tabla 4.5: Caso de uso: Modificar una Orden de Trabajo

Caso de uso: Modificar una Orden de Trabajo	
Resumen	Modificar la Orden de trabajo seleccionada.
Actor	Administrador
Precondición	Seleccionar una orden del listado desde una ubicación con acceso local al servidor y que este se encuentre activo
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Seleccionar una Orden de trabajo 2) Pulsar Editar. 3) Rellenar o modificar un campo 4) Pulsar Guardar. 5) El listado se actualizará para mostrar la Orden Modificada (si los campos modificados son visibles en las columnas de la lista) <p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 2.1) Rellenar los campos restantes que se deseen <p>Flujo Alternativo 2:</p> <ol style="list-style-type: none"> 2.1) Añadir Operarios <p>Flujo Alternativo 3:</p> <ol style="list-style-type: none"> 2.1) Eliminar Operarios <p>Flujo Alternativo 4:</p> <ol style="list-style-type: none"> 2.1) Añadir Imágenes <p>Flujo Alternativo 5:</p> <ol style="list-style-type: none"> 2.1) Eliminar Imágenes <p>Flujo Alternativo 6:</p> <ol style="list-style-type: none"> 2.1) Cualquier combinación de los anteriores flujos alternos <p>Flujo Alternativo 7</p> <ol style="list-style-type: none"> 5) <i>En caso de fallo se mostrará una ventana de error y la ventana de creación de Orden permanecerá activa.</i>

Postcondición	El listado mostrará la Orden modificada. La Orden se modificará en la base de datos
---------------	--

Tabla 4.6: Caso de uso: Añadir Imágenes en una Ordenen de Trabajo

Caso de uso: Añadir Imágenes en una Ordenen de Trabajo	
Resumen	Adjuntar imágenes a una Orden de Trabajo
Actor	Administrador
Precondición	Estar en la ventana de creación o en la de modificación de Órdenes de Trabajo
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Pulsar Buscar en el campo Imagen 2) Seleccionar las imágenes que se deseen 3) Pulsar guardar 4) Las imágenes aparecerán en la lista situada debajo <p>Flujo Alternativo:</p> <p>4) <i>En caso de fallo se mostrará una ventana de error y el control donde se muestran las imágenes permanecerá inalterado.</i></p>
Postcondición	Las imágenes estarán disponibles en la lista correspondiente para operar con ellas. La base de datos se alterará para crear los registros de relación de cada Imagen con la Orden

Tabla 4.7: Caso de uso: Eliminar Imágenes en una Ordenen de Trabajo

Caso de uso: Eliminar Imágenes en una Ordenen de Trabajo	
Resumen	Elimina imágenes de una Orden de Trabajo
Actor	Administrador
Precondición	Estar en la ventana de creación o en la de modificación de Órdenes de Trabajo y tener imágenes adjuntas.

Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Seleccionar las Imágenes que se deseen de la lista 2) Pulsar eliminar 3) Las imágenes desaparecerán de la lista <p>Flujo Alternativo:</p> <p>3) <i>En caso de fallo se mostrará una ventana de error y la lista donde se muestran las imágenes permanecerá inalterado.</i></p>
Postcondición	Las imágenes dejarán de estar disponibles en la lista correspondiente para operar con ellas. La base de datos se alterará para eliminar los registros de relación Imagen/Orden

Tabla 4.8: Caso de uso: Re-ordenar Imágenes en una Ordenen de Trabajo

Caso de uso: Re-ordenar Imágenes en una Ordenen de Trabajo	
Resumen	Re-ordenar las imágenes de una Orden de Trabajo
Actor	Administrador
Precondición	Estar en la ventana de creación o en la de modificación de Órdenes de Trabajo
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Seleccionar una Imagen de la lista 2) Pulsar los botones de desplazamiento hasta que se encuentre en la posición deseada 3) Las imágenes cambiarán su orden en la lista para adaptarse a la nueva distribución <p>Flujo Alternativo:</p> <p>3) <i>En caso de fallo se mostrará una ventana de error y la lista donde se muestran las imágenes permanecerá inalterado.</i></p>
Postcondición	Las imágenes se mostrarán en el nuevo orden en la lista. La base de datos se alterará para modificar los índices de las relaciones Imagen/Orden

Tabla 4.9: Caso de uso: Añadir Operarios a una Orden de Trabajo

Caso de uso: Añadir Operario a una Orden de Trabajo	
Resumen	Asignar un Operario a una orden de Trabajo
Actor	Administrador
Precondición	Estar en la ventana de creación o en la de modificación de Órdenes de Trabajo y contar, al menos, con un Operario en la base de Datos
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Seleccionar un Operario en el desplegable 2) Pulsar el botón 'Añadir' en el campo Operarios 3) El Operario aparecerá en la lista situada debajo <p>Flujo Alternativo:</p> <p>3) <i>En caso de fallo se mostrará una ventana de error y la lista donde se muestran los Operarios permanecerá inalterado.</i></p>
Postcondición	El Operario se mostrará en la lista. La base de datos se alterará para crear los registros de relación del Operario con la Orden

Tabla 4.10: Caso de uso: Eliminar Operarios de una Orden de Trabajo

Caso de uso: Eliminar Operarios de una Orden de Trabajo	
Resumen	Eliminar los Operarios seleccionados de la lista de Operarios asignados.
Actor	Administrador
Precondición	Estar en la ventana de creación o en la de modificación de Órdenes de Trabajo y tener Operarios asignados a la Orden
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Seleccionar los Operarios que se deseen eliminar

	<p>de la lista</p> <ol style="list-style-type: none"> 2) Pulsar el botón 'Eliminar' en el campo Operarios 3) El Operario desaparecerá de la lista situada debajo <p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 3) <i>En caso de fallo se mostrará una ventana de error y la lista donde se muestran los Operarios permanecerá inalterado.</i>
Postcondición	El Operario no se mostrará en la lista. La base de datos se alterará para eliminar los registros de relación de dichos Operarios con la Orden

Tabla 4.11: Caso de uso: Identificarse

Caso de uso: Identificarse	
Resumen	Identificar al usuario de la aplicación
Actor	Operario
Precondición	El usuario y la contraseña deben coincidir con algún registro de la tabla Operarios de la base de datos.
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Introducir 'Nombre' y 'Contraseña' en los campos correspondientes 2) Pulsar el botón 'Conectar' 3) Cambiará de página para ofrecernos la lista de Órdenes de trabajo asignadas <p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 3) <i>En caso de fallo se mostrará una ventana de error. Permaneceremos en la pantalla de Login</i>
Postcondición	Los campos 'Nombre' y 'Código' del archivo de 'Settings' cambiarán a los correspondientes del usuario identificado

Tabla 4.12: Caso de uso: Visualizar el listado de Órdenes Asignadas

Caso de uso: Visualizar el listado de Órdenes Asignadas	
Resumen	Muestra un listado de las Órdenes asignadas al Operario identificado en la aplicación
Actor	Operario
Precondición	El campo 'Código' del archivo de 'Settings' debe corresponderse con el de un Operario de la base de datos y debemos tener conexión con el servidor. O haber cargado con éxito la base de datos en el dispositivo al menos una vez.
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Acceder a la pantalla de listado 2) Listar el resumen de las Órdenes asignadas <p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 1) Pulsar el botón de Refresco 2) Listar el resumen de las Órdenes asignadas <p>Flujo Alternativo2:</p> <ol style="list-style-type: none"> 2) En caso de fallo se mostrará una notificación de error y se cargará la última lista almacenada <p>Flujo Alternativo3:</p> <ol style="list-style-type: none"> 2) En caso de que se compruebe que no existe el usuario, se mostrará una notificación de error y se le enviará a la pantalla de Login
Postcondición	Se actualizará la base de datos interna con los datos recibidos.

Tabla 4.13: Caso de uso: Acceder a Orden de Trabajo

Caso de uso: Acceder a Orden de Trabajo	
Resumen	Muestra Los campos pertinentes para el operario en un nuevo formulario
Actor	Operario

Precondición	Pulsar sobre una orden existente en la base de datos interna
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Pulsar sobre una Orden 2) Mostrar el formulario con los datos de la Orden seleccionada <p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 2) En caso de que se compruebe que no existe el usuario, se mostrará una notificación de error y se le enviará a la pantalla de Login.
Postcondición	La variable 'Código' del Formulario contendrá el valor del identificador de la Orden

Tabla 4.14: Caso de uso: Acceder a Geolocalización

Caso de uso: Acceder a Geolocalización	
Resumen	Muestra un mapa con la ubicación de la incidencia
Actor	Operario
Precondición	La orden tiene que tener valores en los campos Lat Long que hacen referencia a las coordenadas
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Pulsar sobre el botón 'Mapa' del campo Ubicación 2) Mostrar la ubicación de la incidencia en una nueva ventana de mapa <p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 2) En caso de que los campos de coordenadas se encuentren vacíos, se notificará este hecho <p>Flujo Alternativo 2:</p> <ol style="list-style-type: none"> 2) En caso de no haber conexión, se notificará
Postcondición	Se cambiará de ventana para pasar a la de mapa. Este formulario almacenará el valor de las coordenadas

Tabla 4.15: Caso de uso: Acceder a Imagen

Caso de uso: Acceder a Imagen	
Resumen	Muestra una de las imágenes, adjuntas a la Orden, ampliada
Actor	Operario
Precondición	La Orden debe tener imágenes y haber sido descargadas a la memoria del dispositivo
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Pulsar sobre el botón 'Imágenes' 2) en la ventana de selección pulsar sobre una de ellas 3) Mostrar la imagen seleccionada <p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 3) En caso de fallo se notificará del error
Postcondición	Se cambiará dos veces de ventana para pasar a la de listado de imágenes y luego a la visualización de imagen

Tabla 4.16: Caso de uso: Modificar Orden

Caso de uso: Modificar Orden	
Resumen	Permite modificar determinados campos de la orden así como añadir fotografías a la misma
Actor	Encargado
Precondición	Figurar como encargado en la Orden seleccionada
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Pulsar sobre el botón de edición (icono del lápiz) 2) Modificar los campos deseados de entre los que se activen para la edición 3) Pulsar 'Enviar' 4) Se enviará la información de la orden al servidor 5) Se modificará la base de datos interna

	<p>Flujo Alternativo:</p> <p>2.1) Añadir Fotografía (Repetible)</p> <p>3) Pulsar 'Enviar'</p> <p>4) Se enviará la información al servidor</p> <p>5) Se modificará la base de datos interna</p> <p>6) Se enviará el listado de objetos AGIMImageltem al servidor junto con el identificador de la Orden (el cual las añadirá junto con las referencias Imagen/Orden)</p> <p>7) Se actualizará la lista de Imágenes en el dispositivo así como las referencias a la Orden.</p> <p>Flujo Alternativo 2:</p> <p>5) En caso de fallo se notificará del error, no se modificará la información de la memoria interna ni del servidor</p> <p>Flujo Alternativo 3:</p> <p>7) En caso de fallo se notificará del error, no se modificará la información de la memoria interna ni del servidor</p>
Postcondición	La información tanto de la base de datos interna como de la del servidor se actualizarán

Tabla 4.17: Caso de uso: Añadir Imágenes desde Archivos

Caso de uso: Añadir Imágenes desde Archivos	
Resumen	Permite seleccionar archivos de imagen para adjuntar a la Orden
Actor	Encargado
Precondición	Estar en la ventana de modificación
Descripción	<p>Flujo Básico:</p> <p>1) Pulsar sobre el botón de 'Añadir Imagen'</p> <p>2) Seleccionar las Imágenes que se desee incorporar</p> <p>3) Pulsar 'Guardar'</p> <p>4) Los iconos de las imágenes aparecerán en el listado inferior</p>

	Flujo Alternativo: 4) En caso de fallo se notificará del error
Postcondición	La información de las imágenes será añadida a la variable 'listaimagenes'

Tabla 4.18: Caso de uso: Añadir Imagen desde la Cámara

Caso de uso: Añadir Imagen desde la Cámara	
Resumen	Permite tomar una fotografía para adjuntar a la Orden
Actor	Encargado
Precondición	Estar en la ventana de modificación
Descripción	Flujo Básico: 1) Pulsar sobre el botón de 'Capturar Imagen' 2) realizar la fotografía 3) Aceptar la imagen 4) El icono de las imagen aparecerán en el listado inferior Flujo Alternativo: 4) En caso de fallo se notificará del error
Postcondición	La información de la imagen será añadida a la variable 'listaimagenes'

Tabla 4.19: Caso de uso: Modificar Observaciones

Caso de uso: Modificar Observaciones	
Resumen	Añadirá texto al campo observaciones
Actor	Encargado
Precondición	Estar en la ventana de modificación
Descripción	Flujo Básico: 1) Seleccionar el campo Observaciones 2) Escribir el texto deseado 3) El texto aparecerá listo para enviar justo debajo del ya existente

Postcondición	El campo texto del campo de texto editable 'Observaciones' contendrá el nuevo texto sin modificar el ya existente (listo para ser añadido en el momento del envío)
---------------	--

Tabla 4.20: Caso de uso: Modificar Ubicación

Caso de uso: Modificar Ubicación	
Resumen	Modifica la ubicación de la incidencia
Actor	Operario
Precondición	Estar en la ventana de modificación, tener activada la geolocalización GPS
Descripción	<p>Flujo Básico:</p> <ol style="list-style-type: none"> 1) Pulsar sobre el botón 'Mapa' del campo Ubicación 2) Mover el marcador hasta la ubicación deseada 3) Pulsar el botón de regresar 4) El campo ubicación mostrará la dirección seleccionada. Y las coordenadas se actualizarán <p>Flujo Alternativo 1:</p> <ol style="list-style-type: none"> 2) En caso de no haber conexión, se notificará <p>Flujo Alternativo 2:</p> <ol style="list-style-type: none"> 4) En caso fallo, se notificará <p>Flujo Alternativo 3:</p> <ol style="list-style-type: none"> 5) Modificar el texto de la ubicación.
Postcondición	El campo texto 'Ubicación' se actualizará así como las coordenadas.

5 Diseño

Arquitectura del sistema

Se ha escogido una arquitectura de tres capas (*Ilustración 5.1*), por ser la que mejor se adapta al caso a tratar. Dado que se pretende utilizar dos interfaces bien diferenciadas, una de ellas móvil y había que contar con la centralización de los datos en el servidor del propio Ayuntamiento, esta arquitectura es idónea.

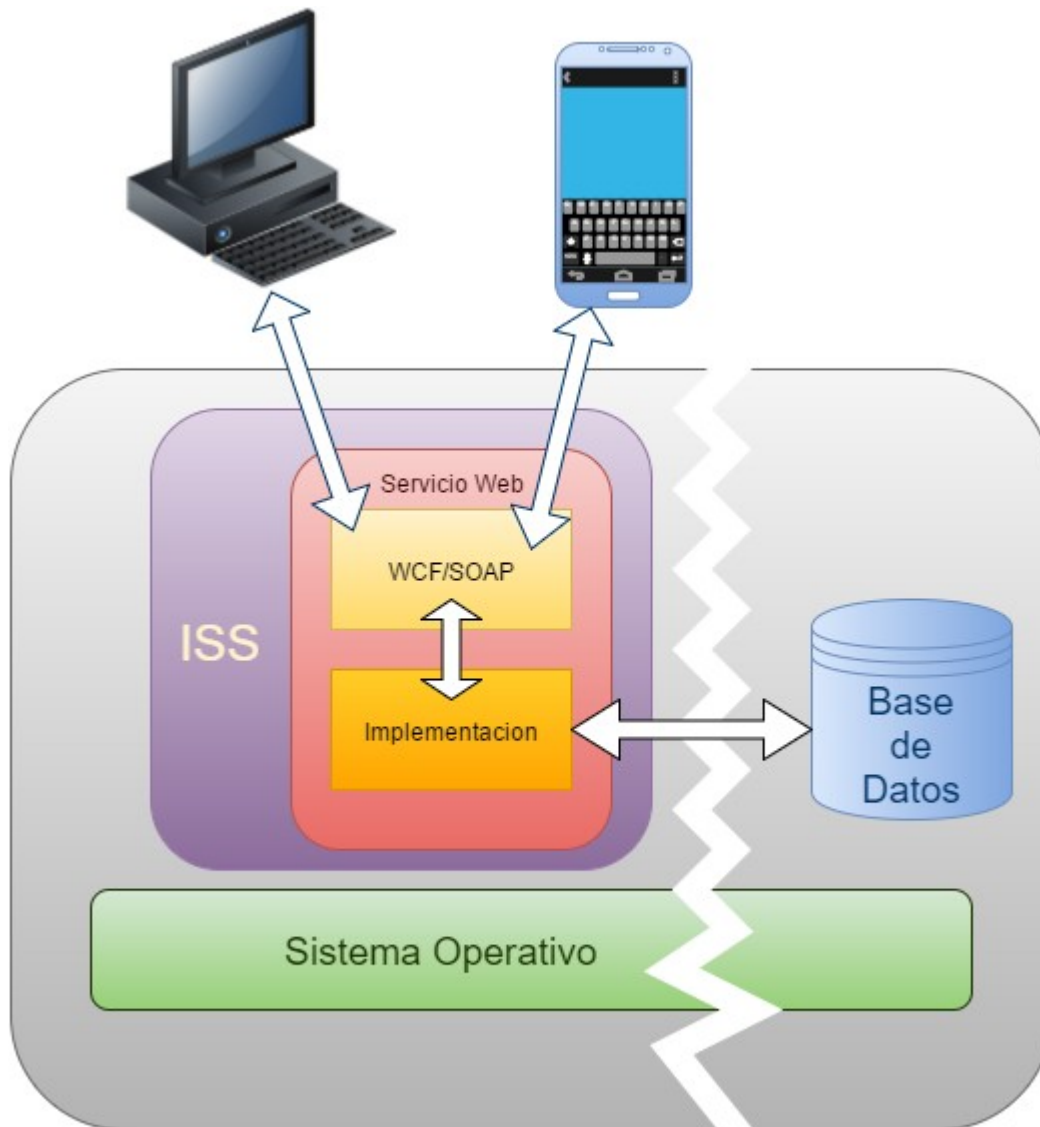


Ilustración 5.1: Arquitectura del sistema

Así pues por un lado tenemos la capa lógica que contendrá al servidor web, las interfaces de datos y métodos para la conexión con la capa de presentación, así como su implementación para la conexión con la base de datos para extraer la información de las

peticiones o la inclusión de la información enviada por la capa de presentación.

La de presentación, que estará formada por la aplicación de escritorio y la aplicación móvil. A priori estas capas no deberían tener una parte lógica muy compleja, aparte de la propia para manejar las comunicaciones con el servicio, pero dado que se ha incluido el envío de imágenes se debe implementar métodos para la conversión de estas en flujos de datos. Aparte la aplicación móvil deberá implementar una pequeña base de datos y los métodos para su gestión, para garantizar la persistencia de los datos cuando el dispositivo no tenga conexión.

Por último la Base de datos está alojada en el mismo servidor que el servicio web pero no se descarta trasladarla a otro, en el que se encuentra la base de datos principal, cuando el proyecto se encuentre en una fase más avanzada.

5.1 Servicio Web

Por un lado se han generado las interfaces (o contratos, contracts) tanto de los métodos como de los tipos de datos. Las interfaces de los métodos declaran las entradas y las salidas de cada uno de ellos de manera completamente independiente de la implementación concreta. De este modo, si cambiásemos de base de datos, por ejemplo, únicamente deberemos modificar la implementación, con lo que la comunicación con los clientes no se vería afectada.

Lo mismo sucede con los contratos de datos, independientemente de los tipos que maneje la implementación, solo hay que asegurarse que la respuesta se empaquete en uno de los tipos que estamos manejando (esto no afecta a los tipos simples que no requieren contrato alguno) pudiéndose enviar agrupados en listas o colecciones que también están soportadas por el protocolo.

Claro está que si añadimos algún método o cambiamos las entradas o salidas de alguno de los existentes habrá que modificar tanto el servidor como los clientes. Pero esto no quita para que sea la opción más cómoda a la hora de mantener el servicio.

Por otro lado tenemos la implementación que es la que se encargará de realizar las comunicaciones con la base de datos y realizar las operaciones necesarias con los datos

recibidos por el servicio web o por los adquiridos de la base de datos.

Por último tenemos la configuración de las comunicaciones, donde declararemos protocolos de transporte, de seguridad, los meta-datos, la dirección del servicio, etc. Todo esto junto con los contratos de Métodos y datos estarán incluidos en el archivo WSDL que es a partir del cual podremos generar los métodos de comunicación en las aplicaciones Cliente

5.2 Cliente de Escritorio (Ilustraciones 5.2 y 5.3)

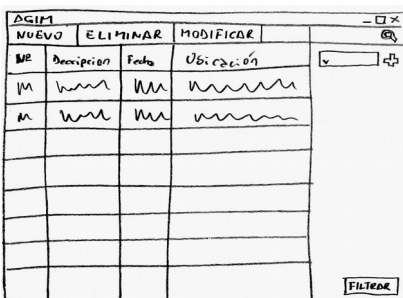


Ilustración 5.2: Boceto Aplicación de Escritorio - Listado

El cliente de escritorio, básicamente no es más que una interfaz donde introducir los datos (o eliminarlos) que queremos que se envíen a la base de datos. Para ello a través de las interfaces generadas en el servicio web se efectúan las diversas peticiones de los métodos del servidor, realizándose las operaciones en este y devolviendo al cliente los datos solicitados, o la confirmación (o no) del éxito de la operación.

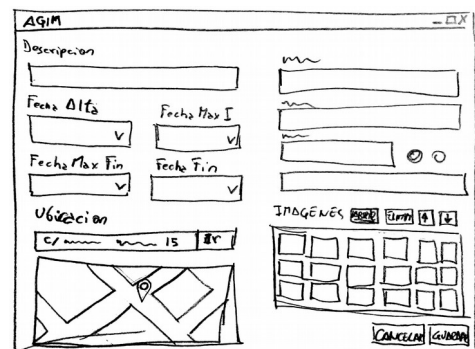


Ilustración 5.3: Boceto Aplicación de escritorio - Edición/Detalle

5.3 Cliente Móvil (Ilustraciones 5.4, 5.5 y 5.6)



Ilustración 5.4: Boceto Android Login

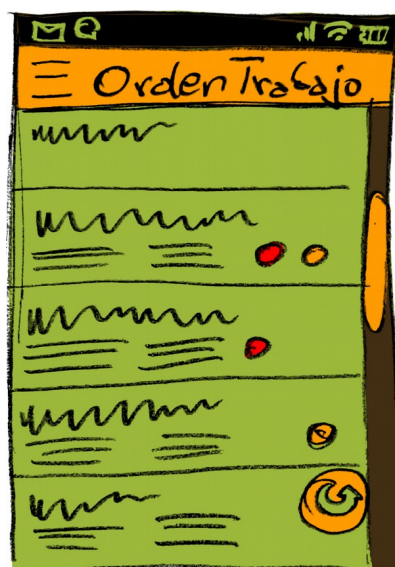


Ilustración 5.5: Boceto Android Lista

El funcionamiento es similar al del cliente de escritorio con el añadido de tener que mantener una pequeña base de datos con el subconjunto de las Órdenes de trabajo para consolidar la persistencia de la información y que durante los periodos en los que este no tenga acceso a la red pueda seguir dando servicio.

5.4 Base de Datos

La base de datos aloja toda la información del sistema en tablas de entidades, Órdenes de Trabajo, Operarios, Imágenes, etc, características, roles, áreas, prioridades, etc, así como relaciones entre unas y otras. Al estar diseñadas de esta manera cada tabla no tiene toda la información (por ejemplo la tabla Órdenes no tiene los nombres de los operarios sino una referencia a la tabla que los contiene) pero hace mas sencillo y ágil gestionar la información contenida. La distribución sería la que podemos ver en la siguiente imagen (ilustración 5.7).

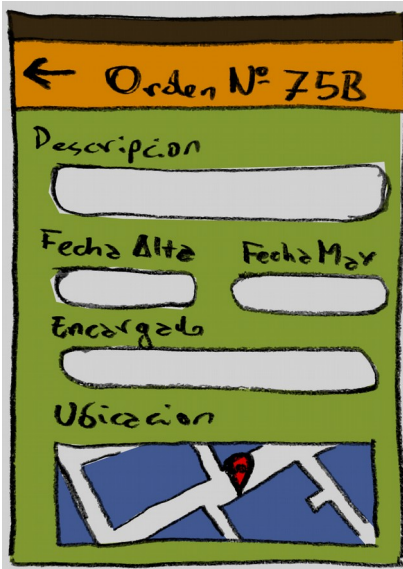


Ilustración 5.6: Boceto Android Lista

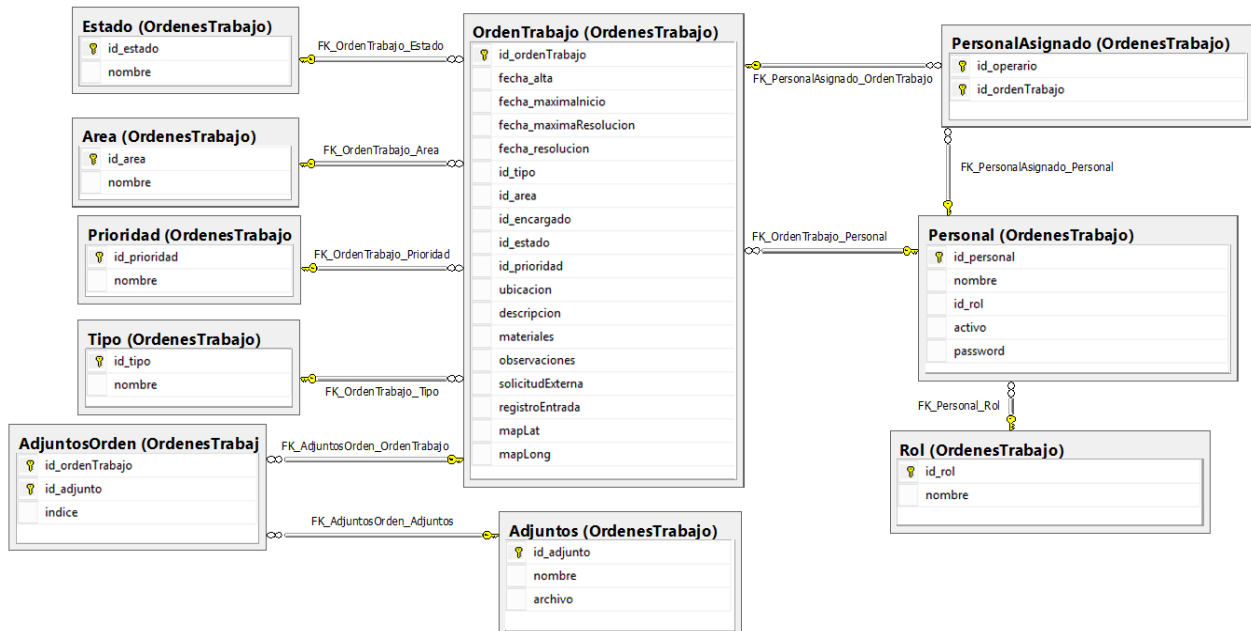


Ilustración 5.7:

6 Implementación

6.1 Servicio Web

Para la implementación del servicio web, aparte de crear las interfaces de métodos se decidió crear Interfaces de datos (*Ilustración 6.1*). Por un lado el objeto 'OrdenTrabajo' (*Ilustración 6.2*) permite reducir el envío del conjunto de datos que conforman una Orden de Trabajo como un único objeto sin tener que preocuparnos de los diferentes

```
[DataContract]
19 references
public class OrdenTrabajo[...]

[DataContract]
15 references
public class AGIMComboBoxItem[...]

[DataContract]
9 references
public class AGIMImageItem[...]
```

Ilustración 6.1: DataContract

```
[DataContract]
19 references | Carlos Frias Ruiz | 8 changes
public class OrdenTrabajo
{
    [DataMember]
    2 references | Carlos Frias Ruiz | 5 changes
    public int Id_ordenTrabajo { get; set; }

    [DataMember]
    3 references | Carlos Frias Ruiz | 4 changes
    public DateTime Fecha_alta { get; set; }

    [DataMember]
    3 references | Carlos Frias Ruiz | 4 changes
    public DateTime Fecha_maximaInicio { get; set; }
}
```

Ilustración 6.2: Objeto Orden de Trabajo

de ellos en un único mensaje.

Por otro lado el objeto 'AGIMComboboxItem' (*Ilustración 6.3*) se utiliza para poder enviar un listado de nombres e índices que puedan ser usados de manera inmediata para rellenar los controles tipo combobox de un formulario lo cual es más sencillo que realizar una petición para los nombres y otra para los índices (aparte de evitar errores).

```
[DataContract]
15 references | Carlos Frias Ruiz | 1 change
public class AGIMComboBoxItem
{
    [DataMember]
    1 reference | Carlos Frias Ruiz | 1 change
    public int Index { get; set; }

    [DataMember]
    1 reference | Carlos Frias Ruiz | 1 change
    public string Name { get; set; }
}
```

Ilustración 6.3: AGIMComboboxItem

tipos básicos que los conforman. En principio se implementó como un 'array' de 'strings', pero había que realizar conversiones a y hacia 'Strings' siendo que es mucho más sencillo utilizar los tipos originales que nos provee la base de datos. Grupos de estos objetos se pueden, además agrupar en listas o colecciones para enviar un conjunto

```
[DataContract]
9 references | Carlos Frias Ruiz | 1 change
public class AGIMImageItem
{
    [DataMember]
    2 references | Carlos Frias Ruiz | 1 change
    public int Index { get; set; }

    [DataMember]
    2 references | Carlos Frias Ruiz | 1 change
    public string Name { get; set; }

    [DataMember]
    2 references | Carlos Frias Ruiz | 1 change
    public string Image { get; set; }
}
```

Ilustración 6.4: AGIMImageItem

Las interfaces de los métodos, 'ServiceContract' (Ilustración 6.5) no tienen mayor misterio, se declara el método con sus entradas y salidas para realizar la implementación aparte. Todos los métodos devuelven un valor o un booleano para evidenciar el

```
[ServiceContract]
1 reference | Carlos Frias Ruiz | 8 changes
public interface IAGIMService
{
    [OperationContract]
    1 reference | Carlos Frias Ruiz | 2 changes
    List<OrdenTrabajo> GetListaOrdenes(int usuario);

    [OperationContract]
    1 reference | Carlos Frias Ruiz | 1 change
    Boolean AuthEncargado(int usuario, String Password);

    [OperationContract]
    1 reference | Carlos Frias Ruiz | 1 change
    OrdenTrabajo GetOrdenTrabajo(int orden);
}
```

Ilustración 6.5: ServiceContract

éxito o fracaso de la operación. Para ello también se podría hacer uso de metadatos, pero se ha decidido simplificar la implementación de esta manera.

Por ultimo se ha decidido utilizar el protocolo de SOAP 1.1 'BasicHttpBinding' que junto a otras opciones de configuración del protocolo de comunicación se definen en el archivo 'Web.config'

```
<system.serviceModel>
  <services>
    <service name="AGIM_WcfService.Service1">
      <endpoint address="" binding="basicHttpBinding" bindingConfiguration="" contract="AGIM_WcfService.IAGIMService"/>
      <endpoint address="mex" binding="mexHttpBinding" contract="IMetadataExchange"/>
    </service>
  </services>
  <host>
    <baseAddresses>
      <add baseAddress="http://localhost:8000/AGIM_WcfService/AGIMService"/>
    </baseAddresses>
  </host>
</system.serviceModel>
```

Ilustración 6.6: Fragmento del archivo Web.config

El caso del objeto AGIMImagen (Ilustración 6.4) es similar, con la particularidad de que cuando enviamos una imagen esta siempre tiene que ir acompañada de su nombre e identificador, ya que la imagen se envía como un flujo de bytes en un 'String' en formato 'Base 64' y no como un archivo con su propio nombre. El identificador es necesario para poder identificar la imagen en la base de datos y encontrar a qué orden esta asignada.

6.2 Aplicación de Escritorio

Ilustración 6.7: Ventana de edición del formulario AXML

La aplicación de escritorio se ha realizado, como ya se ha comentado anteriormente en el formato de aplicaciones de la Windows Store, de modo que la interfaz esta generada a partir de un documento XAML (Ilustración 6.7). en el que los diversos elementos se estructuran a través de una serie de etiquetas. Los elementos mas remarcables serian el control 'WebView', que permite incrustar un elemento navegador web que nos permitirá situar el mapa a través de un archivo HTML que cargaremos al comienzo, y el código que nos permite interactuar con las funciones JavaScript que controlan los eventos de la API de Google Maps.

```
<Grid x:Name="Campo_7_1" Grid.Column="1" Margin="5,5,5,5" Grid.Row="7"
  Grid.RowSpan="3">
  <WebView x:Name="wv_mapa" Grid.Row="1" ScriptNotify="AGIM_ScriptNotify"
    NavigationCompleted="AGIM_NavigationCompleted" Source="ms-appx-web:///map.html"/>
</Grid>
```

Con la expresion '*ScriptNotify="AGIM_ScriptNotify"*' se genera el manejador que, tras arrastrar el marcador disponible en el mapa, nos notificara cuando nuestro código JavaScript embebido en la

página html termine de obtener la localización de esta manera obtendremos tanto las coordenadas (que se guardaran para ser enviadas en el momento de generar la Orden) como la dirección, que utilizaremos para rellenar el respectivo campo de texto que se podrá corregir antes del envío.

```
function geocodePosition(pos) {
    function geocodePosition(pos) {
        latLng: pos
    }, function (responses) {
        if (responses && responses.length > 0) {
            marker.formatted_address = responses[0].formatted_address;
        } else {
            marker.formatted_address =
                'Cannot determine address at this location.';
        }
        infowindow.setContent(marker.formatted_address);
        infowindow.open(map, marker);
        Alert();
    });
}
```

6.3 Aplicación Móvil

La aplicación móvil, aparte de la propia interfaz, tiene cierta complejidad al tener que desarrollar los métodos de conexión y la base de datos interna. Para la creación de los métodos se ha hecho uso de la utilidad que la empresa 'Easy WSDL' ofrece para la generación automática de las conexiones y que en la plataforma de Microsoft hace directamente el entorno de desarrollo.

Esta herramienta, también disponible como complemento para instalar en entornos de desarrollo Android como 'Android Studio' permite a partir del archivo WSDL generar todos los métodos de conexión así como las clases que albergan los tipos de datos utilizados por el servicio.

De este modo ya solo nos queda crear el objeto que nos dará el servicio :

```
private static BasicHttpBinding_IAGIMService service = new BasicHttpBinding_IAGIMService();
```

Y a continuación utilizarlo para llamar a los métodos asignados a las diversas interfaces del servicio web. Hay que destacar que todas estas peticiones se deben hacer a través de llamadas asíncronas puesto que, aparte de ser una buena práctica, cuando se trabaja con servicios que pueden ralentizar el hilo de ejecución principal, desde ya hace unas cuantas versiones, el propio sistema no nos permite generar este tipo de tareas en el hilo principal.

Es por ello que se deberán ejecutar por medio de una clase que extienda a `AsyncTask`. Implementando para ello los métodos '`onPreExecute()`', '`doInBackground()`', '`onProgressUpdate()`' y '`onPostExecute()`'. Esto nos permite generar un segundo hilo que a medida que vaya progresando (y dependiendo de como se haya implementado) lo notificará a través del método '`onProgressUpdate()`' y que al finalizar procesará los comandos que contenga el método '`onPostExecute()`'. Todo ello sin bloquear en ningún momento el hilo principal.

```
protected static class RellenarBD extends AsyncTask <Integer, Float, Integer>
{
    private ArrayOfOrdenTrabajo a;
    private SQLiteOpenHelper oh;
    private SQLiteDatabase agimDB;
    protected void onPreExecute() {
        ...
        oh = new AgimDBHelper(context);
        agimDB = oh.getWritableDatabase();
    }
    protected Integer doInBackground(Integer... id) {
        boolean con= false;
        try{
            con = InetAddress.getByName
                (context.getString(R.string.url)).isReachable(5000);
        } catch (IOException e){...}
        if (con){
            service.setUrl(context.getString(R.string.url));
            try {
                a = service.GetListaÓrdenes(id[0]);
            } catch (Exception e) {...}
            ...
        }
    }
    protected void onProgressUpdate (Float... valores) {
        int p = Math.round(100*valores[0]);
        pBar.setProgress(p);
        pBar2.setProgress(p);
    }
    protected void onPostExecute(Integer bytes) {
        l.SetLista();
        linlaProgressBar.setVisibility(View.GONE);
        pBar2.setVisibility(View.GONE);
    }
}
```

El otro método de interés es el que genera la base de datos interna, para ello se ha creado una clase '`AgimDBHelper`' que extiende de '`SQLiteOpenHelper`' que nos facilitará la tarea de generar y

actualizar la base de datos SQLite.

```
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
public class AgimDBHelper extends SQLiteOpenHelper {
    public AgimDBHelper(Context context)
    {
        super(context, "AgimDB.db", null, 1);
    }
    @Override
    public void onCreate(SQLiteDatabase arg0)
    {
        arg0.execSQL("CREATE TABLE ÓrdenesTrabajo (...)")...}...}
```

7 Resultados

A continuación, se muestran algunas de las pantallas de las aplicaciones realizadas en las que se puede ver el aspecto que presentan.

7.1 Aplicación de escritorio

The screenshot shows a web-based form for creating a work order. The form is titled "Orden de Trabajo Nº" and is set against a dark background. It contains several sections:

- Descripción:** A text input field containing "Fuga en canalización de agua".
- Fecha de inicio:** A date picker set to 30 de noviembre de 2015.
- Fecha máxima de inicio:** A date picker set to 2 de diciembre de 2015.
- Fecha máxima de finalización:** A date picker set to 18 de diciembre de 2015.
- Fecha de finalización:** A date picker set to 2 de diciembre de 2015.
- Ubicación:** A text input field containing "Calle Pilar, 6, 46370 Chiva, Valencia, España" and a button "Ir al mapa". Below this is a Google Maps view showing the location.
- Materiales:** A text input field containing "Estanto, Tubos de cobre, Estopa, Cemento".
- Tipo:** A dropdown menu set to "Reparacion".
- Area:** A dropdown menu set to "Fontaneria".
- Prioridad:** A dropdown menu set to "Alta".
- Estado:** A dropdown menu set to "Asignada".
- Encargado:** A dropdown menu set to "Jacinto Garcia Machancoses".
- Registro de Entrada:** Radio buttons for "Si" (selected) and "No".
- Solicitud Externa:** A text input field containing "EXG-723546/FGTY-69876234-X".
- Observaciones:** A text area containing "Atender a las manchas de humedad que se obserban en el edificio aledaño y valorar la gravedad de la incidencia sobre este."
- Imágenes:** A gallery of image thumbnails with a "Buscar Imagen" button.
- Buttons:** "Cancelar" and "Guardar" buttons at the bottom right.

Ilustración 7.1: Formulario de creación de una Orden de Trabajo en la Aplicación de Escritorio

La aplicación de escritorio, como se ha dicho anteriormente, no es más que una interfaz para acceder a la información contenida en la base de datos. En la [Ilustración 7.1](#) podemos observar la ventana de creación de nuevas Órdenes, se ha elegido esta porque muestra tanto el control de mapa como el de añadir imágenes. Al pulsar sobre el botón guardar se envía la información introducida en los diversos campos al servicio web y este la inserta en la base de datos.

7.2 Aplicación Móvil

Es más que probable que se le cambien los colores e incluso el logotipo a la aplicación, pero provisionalmente el aspecto es el que se puede observar en la tabla de la página siguiente, donde podemos ver la ventana de identificación ([Ilustración 7.2](#)), el listado de Órdenes asignadas ([Ilustración 7.3](#)) y el detalle de la Orden seleccionada ([Ilustración 7.4](#)). Además de estas, el Encargado debería poder acceder a la pantalla de edición de la orden, que es básicamente la misma, pero con algún campo editable.



Ilustración 7.2: Pantalla de login



Ilustración 7.3: Listado de Ordenes



Ilustración 7.4:Detalle de Orden

7.3 Pruebas

A pesar de que no se han realizado test unitarios, ambas aplicaciones han sido probadas con la introducción de distintos tipos de datos para probar su efectividad. Bien es cierto que dado que la mayor parte de los campos no pueden ser rellenados de manera manual y el resto son campos de texto que no requieren de valores específicos, no existen demasiadas posibilidades de producirse errores en ellas.

En cuanto a la usabilidad, han sido utilizadas en entorno de desarrollo por personal del Ayuntamiento presentando en una buena aceptación. Existe, eso si, cierta tendencia a considerar la aplicación Android mejorable en lo que a navegación y estética se refiere. Estos aspectos y alguno más están pendientes de revisión para futuras iteraciones.

8 Conclusiones y trabajo futuro

8.1 Conclusiones

Se puede pues concluir que el desarrollo a partir de servicios web es un método muy a tener en cuenta en esta era de proliferación de todo tipo de dispositivos tanto móviles como domésticos, con diversas tipologías y sistemas completamente diferentes.

En este panorama, los servicios web nos proporcionan gran dinamismo y libertad a la hora de idear soluciones informáticas a cualquier problema, pues solamente debemos preocuparnos de generar las interfaces correctas sin pensar demasiado en como se efectuará la implementación final. Puesto que con que funcione correctamente en el servidor donde este alojado el servicio, los clientes, siempre que realicen las peticiones a través de las interfaces que hemos habilitado, recibirán la información de la misma manera, no importa que hardware o que software posean, el resultado obtenido siempre será el mismo.

Esto no es solo útil a la hora de implementar la solución, si no que también nos facilita la labor de mantenimiento puesto que reduce la carga de código en las aplicaciones cliente y cualquier modificación en la implementación se centrara básicamente en el código del servidor.

Por lo demás, se han cumplido en gran medida los objetivos que se perseguían. Es cierto que, por ejemplo, el modulo de la gestión de recursos materiales no se ha podido implementar, puesto que a día de hoy el Ayuntamiento no realiza seguimiento alguno de los materiales almacenados. En lo personal la experiencia ha sido buena tanto en el trato con los empleados del Ayuntamiento como en el apoyo prestado por estos y por la Entidad así como por la experiencia conseguida.

Por parte del Ayuntamiento el proyecto, pese que en la actualidad no puede ponerse aun en funcionamiento, les resulta muy prometedor y se seguirá trabajando en ello para, dependiendo de las disponibilidades presupuestarias, llevarlo a término en algún momento.

8.2 Trabajo Futuro

La solución que se ha presentado en este proyecto la considero una simple iteración de

una solución mucho mas ambiciosa. Creo que hay mucho margen de mejora que a día de hoy con los conocimientos que he ido adquiriendo durante el desarrollo de la misma me gustaría desarrollar en algún momento.

Una de las mejoras que vengo analizando desde hace tiempo es implementar una serie de métodos que gestionen ciertos aspectos de la logística. Uno de ellos sería tener una base de datos de los materiales almacenados, de manera que se pueda hacer previsión de los mismos conforme se asignan a las diferentes Órdenes de trabajo.

Otro es la gestión del personal, de manera que, a la hora de asignar a un Operario, el sistema nos ofrezca primero los que están libres o los que tienen menos horas computadas. O incluso que haga una preselección teniendo en cuenta no solo la disposición sino su afinidad con el área de actuación (fontanería, Pintura...)

Otra, quizás mas ambiciosa seria un método que permita priorizar los Operarios a incluir en una orden, dependiendo de la proximidad tanto, física como temporal, de las Órdenes que ya tienen asignada para optimizar los desplazamientos.

9 Referencias y documentos consultados

9.1 Referencias

[1] Extensible Markup Language (XML)

<http://www.w3.org/XML>

[2] Web Services Description Language (WSDL) 1.1

<http://www.w3.org/TR/wsdl>

[3] Simple Object Access Protocol (SOAP) 1.1

<http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

[4] Consuming Web Services with kSOAP

<http://code.tutsplus.com/tutorials/consuming-web-services-with-ksoap--mobile-21242>

[5] ¿Qué es Windows Communication Foundation?

[https://msdn.microsoft.com/es-es/library/ms731082\(v=vs.110\).aspx](https://msdn.microsoft.com/es-es/library/ms731082(v=vs.110).aspx)

[6] SQL

<https://es.wikipedia.org/wiki/SQL>

[7] Difference between BasicHttpBinding and WsHttpBinding

<http://www.codeproject.com/Articles/36396/Difference-between-BasicHttpBinding-and-WsHttpBind>

[8] ntext, text, and image (Transact-SQL)

<https://msdn.microsoft.com/en-us/library/ms187993.aspx>

9.2 Paginas consultadas

Hosting and Consuming WCF Services

<https://msdn.microsoft.com/en-us/library/bb332338.aspx>

Windows Communication Foundation FAQ quick starter Part 1

<http://www.codeproject.com/Articles/29475/Windows-Communication-Foundation-FAQ-quick-starter>

Windows Communication Foundation FAQ quick starter Part 2

<http://www.codeproject.com/Articles/29480/Windows-Communication-Foundation-FAQ-quick-start>

9.3 Apuntes Asignaturas

Análisis y especificación de requisitos (11570)

Integración e interoperabilidad (11572)

Proyecto de ingeniería de software (11574)

Ingeniería del software (11564)

Sistemas multimedia interactivos multicanal (11629)

Soluciones informáticas para dispositivos móviles (11634)