

UNIVERSIDAD CARLOS III DE MADRID

ESCUELA POLITECNICA SUPERIOR



INGENIERÍA TÉCNICA DE INFORMÁTICA DE GESTIÓN

PROYECTO FIN DE CARRERA

**SISTEMA DE GESTIÓN Y NOTIFICACIÓN DE
INCIDENCIAS**

Autor: Laura Díaz Mendiola

Tutor: Jorge Morato Lara

Leganés, Julio 2015



AGRADECIMIENTOS

A todos mis familiares y amigos que han estado ahí y me han animado durante todos estos años a continuar hasta conseguir esta meta, en especial a mis padres y a mi hermana.



ÍNDICE

| | |
|---|----|
| AGRADECIMIENTOS..... | 2 |
| ÍNDICE..... | 3 |
| ÍNDICE DE ILUSTRACIONES..... | 5 |
| ÍNDICE DE TABLAS..... | 8 |
| 1 INTRODUCCIÓN..... | 10 |
| 1.1 Motivación..... | 10 |
| 1.2 Objetivos..... | 11 |
| 1.3 Descripción del resto de la memoria..... | 12 |
| 2 ESTADO DEL ARTE..... | 14 |
| 2.1 Comparación con herramientas similares..... | 14 |
| 2.2 Comparación con el sistema utilizado previamente..... | 20 |
| 3 SOFTWARE UTILIZADO..... | 22 |
| 4 ANÁLISIS..... | 28 |
| 4.1 Roles..... | 28 |
| 4.2 Requisitos..... | 29 |
| 4.2.1 Requisitos funcionales..... | 30 |
| 4.2.2 Requisitos no funcionales..... | 40 |
| 4.3 Diagrama y especificación de Casos de Uso..... | 41 |
| 5 DISEÑO..... | 46 |
| 5.1 Arquitectura..... | 46 |
| 5.2 Diagrama de Clases..... | 48 |
| 5.3 Diagrama de Secuencia..... | 50 |
| 5.4 Diseño de la Base de Datos..... | 54 |
| 6 IMPLEMENTACIÓN..... | 55 |
| 6.1 Tablas de base de datos..... | 55 |



| | |
|--|-----|
| 6.2 Procedimientos de base de datos..... | 60 |
| 6.3 Métodos de los Web Services..... | 68 |
| 6.4 Funcionalidades | 79 |
| 7 EVALUACIÓN Y PRUEBAS | 92 |
| 7.1 Problemas encontrados | 93 |
| 7.2 Pruebas | 95 |
| 8 ANÁLISIS ECONÓMICO..... | 107 |
| 8.1 Planificación | 107 |
| 8.2 Desglose de tareas | 110 |
| 8.2.1 TAREA 1. Análisis del proyecto..... | 111 |
| 8.2.2 TAREA 2. Desarrollo y pruebas..... | 111 |
| 8.2.3 TAREA 3. Documentación y memoria del proyecto..... | 112 |
| 8.2.4 Resumen de tareas..... | 112 |
| 8.3 Costes..... | 113 |
| 8.2.1 Costes materiales | 113 |
| 8.2.2 Costes de personal | 114 |
| 8.2.3 Costes indirectos | 114 |
| 8.2.4 Costes totales | 115 |
| 9 CONCLUSIONES Y TRABAJOS FUTUROS..... | 116 |
| 9.1 Conclusiones | 116 |
| 9.2 Trabajos futuros..... | 117 |
| 10 REFERENCIAS | 118 |
| ANEXO A: Manual de usuario del Sistema de notificación de incidencias..... | 121 |
| ANEXO B: Manual de usuario del Sistema de gestión de incidencias..... | 129 |
| ANEXO C: Instalación y configuración del IIS..... | 142 |
| ANEXO D: Publicación de las aplicaciones en el IIS | 144 |
| ANEXO E: Configuración de las aplicaciones en el IIS..... | 149 |

ÍNDICE DE ILUSTRACIONES

| | |
|---|----|
| Ilustración 1: Flujo de trabajo anterior..... | 20 |
| Ilustración 2: Flujo de trabajo con el Sistema de gestión y notificación de incidencias | 21 |
| Ilustración 3: Logotipo de Visual Studio 2010 | 22 |
| Ilustración 4: Logotipo de .NET Framework | 23 |
| Ilustración 5: Logotipo de Microsoft Silverlight | 24 |
| Ilustración 6: Logotipo de Internet Explorer | 25 |
| Ilustración 7: Logotipo de SQL Server 2008 R2 | 25 |
| Ilustración 8: Logotipo de IIS | 26 |
| Ilustración 9: Logotipo de Microsoft Office 2013..... | 27 |
| Ilustración 10: Diagrama Casos de uso – Usuario | 42 |
| Ilustración 11: Diagrama Casos de uso – Desarrollador..... | 44 |
| Ilustración 12: Arquitectura..... | 47 |
| Ilustración 13: Diagrama de Clases..... | 49 |
| Ilustración 14: Diagrama de Secuencia – Alta de incidencia..... | 50 |
| Ilustración 15: Diagrama de Secuencia – Modificar datos de contacto..... | 51 |
| Ilustración 16: Diagrama de Secuencia – Consultar incidencias anteriores..... | 51 |
| Ilustración 17: Diagrama de Secuencia – Consultar incidencias | 52 |
| Ilustración 18: Diagrama de Secuencia – Asignar incidencia | 52 |
| Ilustración 19: Diagrama de Secuencia – Cerrar incidencia | 53 |
| Ilustración 20: Diagrama de Secuencia – Generar informes..... | 53 |
| Ilustración 21: Diseño de base de datos..... | 54 |
| Ilustración 22: Botón para acceder al Sistema de notificación de incidencias | 79 |
| Ilustración 23: Pantalla del Sistema de notificación de incidencias..... | 79 |
| Ilustración 24: Notificar incidencia | 80 |
| Ilustración 25: Botón para ver incidencias del usuario | 82 |
| Ilustración 26: Consulta Histórico de incidencias del usuario..... | 82 |
| Ilustración 27: Botón para modificar datos de contacto del usuario..... | 83 |
| Ilustración 28: Pantalla de modificación de datos del usuario | 83 |
| Ilustración 29: Configuración de botones para ver la consulta de incidencias en desarrollo por el usuario conectado | 84 |



| | |
|--|-----|
| Ilustración 30: Configuración de botones para ver la consulta de todas las incidencias sin asignar o en desarrollo..... | 85 |
| Ilustración 31: Configuración de botones para ver la consulta de incidencias solucionadas por el usuario conectado | 85 |
| Ilustración 32: Configuración de botones para ver la consulta de todas las incidencias solucionadas | 86 |
| Ilustración 33: Botón que indica las incidencias que tiene el usuario sin asignar y pulsándolo se irá a dicha consulta..... | 86 |
| Ilustración 34: Consulta incidencias sin asignar | 87 |
| Ilustración 35: Botón para cerrar incidencia | 87 |
| Ilustración 36: Botón para refrescar la consulta de incidencias | 88 |
| Ilustración 37: Tipos de filtros | 88 |
| Ilustración 38: Consulta incidencias con un dato agrupado | 89 |
| Ilustración 39: Consulta incidencias con varios datos agrupados..... | 89 |
| Ilustración 40: Botón para exportar datos de la consulta..... | 90 |
| Ilustración 41: Problema solucionado al enviar correos | 93 |
| Ilustración 42: Problema solucionado al instalar y configurar el IIS | 94 |
| Ilustración 43: Carga masiva de datos..... | 105 |
| Ilustración 44: Diagrama de Gantt – Planificación inicial..... | 108 |
| Ilustración 45: Diagrama de Gantt – Planificación real | 109 |
| Ilustración 46: Botón para acceder al Sistema de notificación de incidencias | 121 |
| Ilustración 47: Sistema de notificación de incidencias..... | 122 |
| Ilustración 48: Sistema de notificación de incidencias - Modificar datos del usuario . | 123 |
| Ilustración 49: Sistema de notificación de incidencias..... | 124 |
| Ilustración 50: Sistema de notificación de incidencias – Datos de la incidencia | 125 |
| Ilustración 51: Sistema de notificación de incidencias – Histórico de mis incidencias | 126 |
| Ilustración 52: Sistema de notificación de incidencias – Notificar incidencia..... | 127 |
| Ilustración 53: Sistema de notificación de incidencias – Envío de correo al usuario... | 127 |
| Ilustración 54: Sistema de notificación de incidencias – Envío de correo al desarrollador/es de mantenimiento | 128 |
| Ilustración 55: Sistema de gestión de incidencias - Consulta Incidencias en desarrollo por el usuario conectado (consulta sin datos registrados) | 131 |
| Ilustración 56: Sistema de gestión de incidencias - Consulta Incidencias en desarrollo por el usuario conectado..... | 132 |
| Ilustración 57: Sistema de gestión de incidencias – Cerrar incidencia..... | 133 |



| | |
|---|-----|
| Ilustración 58: Sistema de gestión de incidencias - Envío de correo con incidencia solucionada al usuario | 133 |
| Ilustración 59: Sistema de gestión de incidencias - Consulta Incidencias solucionadas por el usuario conectado..... | 134 |
| Ilustración 60: Sistema de gestión de incidencias - Consulta Incidencias no solucionadas por todos los desarrolladores..... | 135 |
| Ilustración 61: Sistema de gestión de incidencias - Consulta Incidencias solucionadas por todos los desarrolladores..... | 136 |
| Ilustración 62: Sistema de gestión de incidencias - Búsqueda rápida | 137 |
| Ilustración 63: Sistema de gestión de incidencias - Filtrar por columnas | 138 |
| Ilustración 64: Sistema de gestión de incidencias - Exportar datos a Excel..... | 139 |
| Ilustración 65: Sistema de gestión de incidencias – Datos exportados | 139 |
| Ilustración 66: Sistema de gestión de incidencias - Incidencias sin asignar..... | 140 |
| Ilustración 67: Sistema de gestión de incidencias – Asignar incidencia..... | 141 |
| Ilustración 68: Instalación y configuración del IIS | 142 |
| Ilustración 69: Instalación y configuración del IIS | 143 |
| Ilustración 70: Publicación de las aplicaciones en el IIS | 145 |
| Ilustración 71: Publicación de las aplicaciones en el IIS | 146 |
| Ilustración 72: Publicación de las aplicaciones en el IIS | 147 |
| Ilustración 73: Publicación de las aplicaciones en el IIS..... | 148 |
| Ilustración 74: Configuración de las aplicaciones en el IIS | 149 |
| Ilustración 75: Configuración de las aplicaciones en el IIS | 150 |
| Ilustración 76: Configuración de las aplicaciones en el IIS | 151 |
| Ilustración 77: Configuración de las aplicaciones en el IIS | 151 |
| Ilustración 78: Configuración de las aplicaciones en el IIS..... | 152 |
| Ilustración 79: Configuración de las aplicaciones en el IIS | 153 |
| Ilustración 80: Configuración de las aplicaciones en el IIS..... | 154 |
| Ilustración 81: Configuración de las aplicaciones en el IIS | 155 |
| Ilustración 82: Configuración de las aplicaciones en el IIS | 156 |
| Ilustración 83: Configuración de las aplicaciones en el IIS..... | 157 |
| Ilustración 84: Configuración de las aplicaciones en el IIS | 158 |
| Ilustración 85: Configuración de las aplicaciones en el IIS | 158 |
| Ilustración 86: Configuración de las aplicaciones en el IIS | 159 |
| Ilustración 87: Configuración de las aplicaciones en el IIS | 159 |
| Ilustración 88: Configuración de las aplicaciones en el IIS..... | 160 |

ÍNDICE DE TABLAS

| | |
|---|----|
| Tabla 1: Comparación con herramientas similares | 14 |
| Tabla 2: Esquema general de los requisitos | 29 |
| Tabla 3: Requisitos – RF01: Usuarios registrados en base de datos | 30 |
| Tabla 4: Requisitos – RF02: Desarrolladores registrados en base de datos..... | 30 |
| Tabla 5: Requisitos – RF03: Acceder al Sistema de notificación de incidencias..... | 31 |
| Tabla 6: Requisitos – RF04: Acceder al Sistema de gestión de incidencias..... | 31 |
| Tabla 7: Requisitos – RF05: Modificar datos de contacto del usuario | 32 |
| Tabla 8: Requisitos – RF06: Consultar incidencias anteriores | 32 |
| Tabla 9: Requisitos – RF07: El usuario da de alta una incidencia | 33 |
| Tabla 10: Requisitos – RF08: Se registra una incidencia por fallo de las aplicaciones.. | 34 |
| Tabla 11: Requisitos – RF09: Recibir correo al registrar una incidencia el usuario..... | 34 |
| Tabla 12: Requisitos – RF10: Recibir correo al registrar una incidencia por fallo de la aplicación..... | 35 |
| Tabla 13: Requisitos – RF11: Recibir justificante al registrar una incidencia | 35 |
| Tabla 14: Requisitos – RF12: Recibir correo de incidencia solucionada..... | 36 |
| Tabla 15: Requisitos – RF13: Consultar incidencias en desarrollo por el usuario conectado | 36 |
| Tabla 16: Requisitos – RF14: Consultar incidencias solucionadas por el usuario conectado | 37 |
| Tabla 17: Requisitos – RF15: Consultar incidencias sin asignar y en desarrollo por todos los desarrolladores | 37 |
| Tabla 18: Requisitos – RF16: Consultar incidencias solucionadas por todos los desarrolladores..... | 38 |
| Tabla 19: Requisitos – RF17: Asignar incidencia manualmente | 38 |
| Tabla 20: Requisitos – RF18: Cerrar incidencia | 39 |
| Tabla 21: Requisitos – RF20: Generar informes | 39 |
| Tabla 22: Requisitos – RNF01: Trabajar con Windows..... | 40 |
| Tabla 23: Requisitos – RNF02: Trabajar con Internet Explorer | 40 |
| Tabla 24: Esquema general de la especificación de los casos de uso | 41 |
| Tabla 25: Especificación de Casos de Uso – CU01: Alta de incidencia | 42 |
| Tabla 26: Especificación de Casos de Uso – CU02: Modificar datos de contacto..... | 42 |
| Tabla 27: Especificación de Casos de Uso – CU03: Consultar incidencias anteriores.... | 43 |



| | |
|---|-----|
| Tabla 28: Especificación de Casos de Uso – CU04: Consultar incidencias | 44 |
| Tabla 29: Especificación de Casos de Uso – CU05: Asignar incidencia | 44 |
| Tabla 30: Especificación de Casos de Uso – CU06: Cerrar incidencia | 45 |
| Tabla 31: Especificación de Casos de Uso – CU07: Generar informes | 45 |
| Tabla 32: Esquema general de las pruebas | 95 |
| Tabla 33: Casos de prueba – CP-RF01: Usuarios registrados en base de datos..... | 96 |
| Tabla 34: Casos de prueba – CP-RF02: Desarrolladores registrados en base de datos . | 96 |
| Tabla 35: Casos de prueba – CP-RF03: Acceder al Sistema de notificación de incidencias | 97 |
| Tabla 36: Casos de prueba – CP-RF04: Acceder al Sistema de gestión de incidencias .. | 97 |
| Tabla 37: Casos de prueba – CP-RF05: Modificar datos de contacto del usuario..... | 98 |
| Tabla 38: Casos de prueba – CP-RF06: Consultar incidencias anteriores | 98 |
| Tabla 39: Casos de prueba – CP-RF07: El usuario da de alta una incidencia | 99 |
| Tabla 40: Casos de prueba – CP-RF08: Se registra una incidencia por fallo de la aplicación..... | 100 |
| Tabla 41: Casos de prueba – CP-RF13: Consultar incidencias en desarrollo por el usuario conectado | 100 |
| Tabla 42: Casos de prueba – CP-RF14: Consultar incidencias solucionadas por el usuario conectado | 101 |
| Tabla 43: Casos de prueba – CP-RF15: Consultar incidencias sin asignar y en desarrollo por todos los desarrolladores..... | 101 |
| Tabla 44: Casos de prueba – CP-RF16: Consultar incidencias solucionadas por todos los desarrolladores..... | 102 |
| Tabla 45: Casos de prueba – CP-RF17: Asignar incidencia manualmente | 102 |
| Tabla 46: Casos de prueba – CP-RF18: Cerrar incidencia | 103 |
| Tabla 47: Casos de prueba – CP-RF19: Generar informes..... | 104 |
| Tabla 48: Desglose de tareas | 110 |
| Tabla 49: Tarea 1 – Análisis del proyecto | 111 |
| Tabla 50: Tarea 2 – Desarrollo y pruebas | 111 |
| Tabla 51: Tarea 3 – Documentación y memoria del proyecto | 112 |
| Tabla 52: Resumen de tareas | 112 |
| Tabla 53: Costes materiales..... | 113 |
| Tabla 54: Costes de personal..... | 114 |
| Tabla 55: Costes indirectos..... | 114 |
| Tabla 56: Costes totales..... | 115 |

1 INTRODUCCIÓN

En este Proyecto de Fin de Carrera realizado por Laura Díaz Mendiola se desarrollan dos aplicaciones: una para que el equipo de desarrollo lleve la gestión de incidencias de un proyecto perteneciente a una intranet y otra para que los usuarios puedan notificar cuando vean que la aplicación no funciona correctamente.

Estas aplicaciones se han desarrollado con tecnologías de Microsoft: Microsoft Visual Studio 2010, Microsoft Silverlight 5, Microsoft SQL Server 2008 R2 y el lenguaje utilizado ha sido C#.

1.1 Motivación

Llevo unos años formando parte de un equipo de trabajo que se dedica a gestionar procesos aeronáuticos.

Este proyecto se inició hace 10 años y constantemente se están añadiendo nuevas funcionalidades que solicita el cliente, además de dar un soporte de mantenimiento. También se desarrollan nuevas herramientas para hacer más fácil el trabajo del día a día del equipo de desarrollo.

Todas estas aplicaciones se encuentran en una intranet por lo que los usuarios que las usan están previamente registrados, dados de alta en el sistema y no son un número muy elevado.

Como este PFC trata sobre la notificación de incidencias (entre otras cosas), a continuación explicaré un poco como se hace hasta ahora este proceso.

Cuando al usuario le sale un aviso de error o ve que algo no está funcionando correctamente se pone en contacto con el CAU (Centro de Atención al Usuario) a través de una llamada telefónica o correo electrónico y explica lo que ha sucedido. El CAU notifica el error al equipo de desarrollo ya sea también por teléfono o correo electrónico y el equipo de desarrollo lo soluciona según orden de entrada o urgencia.

De esta manera no se tiene un control de las incidencias que hay solucionadas o si ya se ha avisado al CAU de su solución para que a su vez avise al usuario.

Para el equipo de desarrollo también es bastante complicado saber en qué estado se encuentran las incidencias, si ya se ha solucionado, si alguien se está encargando de ello o quien es el que lo está solucionando.

Por ello se planteó hacer este PFC, para facilitar la visualización de los datos al equipo de desarrollo y para que el usuario en lugar de ponerse en contacto con el CAU para informarle de los errores encontrados, este se ponga en contacto directamente con el equipo de desarrollo a través de una aplicación para la inserción de incidencias. De esta manera el usuario sólo utilizará el servicio del CAU para preguntar dudas.

Las aplicaciones que se van a desarrollar se realizarán con tecnologías de Microsoft ya que son las que el cliente tiene validadas y son las que se están usando.

1.2 Objetivos

Con este PFC lo que se pretende es mejorar la manera de gestionar los errores que tienen lugar en las aplicaciones pertenecientes al proyecto.

El principal objetivo es crear una consulta para facilitar la visualización de las incidencias registradas.

También se va a cambiar y mejorar la manera en la que los usuarios introducen las incidencias. En lugar de hacer ese trámite por teléfono o correo, se va a hacer una aplicación para que introduzcan las incidencias a través de ella.

A raíz de estas dos ideas principales se fueron añadiendo nuevas funcionalidades, como enviar correos a los desarrolladores en el momento de tener lugar un error o avisar al usuario a través de un correo que se generará automáticamente en el momento en el que el desarrollador de por cerrada la incidencia.

1.3 Descripción del resto de la memoria

A continuación se exponen los apartados de la memoria.

1. **Introducción:** Se pone en situación al lector explicando que es lo que hay ahora mismo, cuál fue el motivo que llevó a desarrollar este proyecto y cuáles son los objetivos que se quieren alcanzar. Al final de este punto hay una descripción de las partes de las que consta la memoria.

2. **Estado del arte:** Se analizan herramientas similares ya existentes y se explica por qué se ha elegido desarrollar este sistema en lugar de utilizar una de esas.

3. **Software utilizado:** En este punto se hace una breve descripción de las herramientas software más destacadas que se han utilizado para la elaboración del proyecto, tanto para el Sistema de gestión y notificación de incidencias como para la memoria.

4. **Análisis:** Se hace un análisis de las aplicaciones desarrolladas, Sistema de notificación de incidencias y Sistema de gestión de incidencias. Primero se describen los roles o tipos de usuarios que nos podemos encontrar. Y a continuación se detallan los requisitos de las aplicaciones y los casos de uso.

5. **Diseño:** En este punto se muestran la arquitectura del sistema y los diagramas de las aplicaciones. El primero es el diagrama de clases, a continuación el de secuencia y finalmente el diseño de base de datos.

6. **Implementación:** En este punto se explican las tablas y procedimientos que se han creado de base de datos, los métodos de los web services, las principales funcionalidades de las aplicaciones y qué relación hay entre todos estos elementos y cómo interactúan entre ellos.

7. **Evaluación y pruebas:** En este punto se explica en qué momento de la implantación se encuentra este proyecto en el cliente. Se explican los problemas que se han encontrado durante la realización del PFC y cómo se han solventado. Y por último muestra el resultado de las pruebas realizadas con la aplicación que validarán los requisitos establecidos.

8. **Análisis económico:** Aquí se hace un análisis económico del proyecto. Se hace un desglose en tareas y se analizan en función de la duración y los recursos utilizados.



9. Conclusiones y trabajos futuros: En este punto se analiza lo realizado a lo largo del proyecto para comprobar si se han cumplido los objetivos planteados al principio [\[1.2 Objetivos\]](#).

También se explica, analiza y se da una solución a los problemas que se han encontrado durante la realización del mismo. Finalmente se habla del proyecto en el futuro, qué se pretende hacer con él, que uso se le va a dar en el día a día y posibles mejoras que añadir.

10. Referencias: Se muestra la bibliografía consultada, en su mayoría recursos de internet, durante la realización del proyecto.

Para finalizar se han realizado dos anexos con los manuales de usuario de las aplicaciones y otros tres para la instalación del IIS y para la publicación de las aplicaciones en el mismo.

2 ESTADO DEL ARTE

2.1 Comparación con herramientas similares

Antes de comenzar a desarrollar las aplicaciones de este proyecto se han buscado herramientas similares ya existentes, a poder ser de licencia gratuita ya que es algo que va a implantar el equipo de desarrollo para facilitar su trabajo del día a día y para que el cliente no tenga que intervenir económicamente.

A continuación se muestra una serie de herramientas con sus características principales y así poder decidir si podrían interesarnos:

Tabla 1: Comparación con herramientas similares

| Nombre | Tipo de licencia | Servidor | Lenguaje(s) de implementación | Base de datos | Manda email | Exportar a Excel |
|--------------------|------------------|-----------------------------------|-------------------------------|--|-------------|------------------|
| RT | GNU | Apache (Unix, Windows, Macintosh) | Perl | MySql, Postgre Sql, Oracle, SQLite | Si | Si |
| OTRS | GPL | Apache (Unix, Windows, Macintosh) | Perl | MySql, Postgre Sql, DB2, Oracle, MS Sql Server | Si | Si |
| GLPI | GPL | Apache (Unix, Windows, Macintosh) | PHP | MySql | Si | No |
| Spiceworks Desktop | Propietario | | Ruby on Rails | SQLite | Si | Si |
| Redmine | GPL | Ubuntu | Ruby on Rails | MySql, Postgre Sql, SQLite | Si | Si |
| MantisBT | GPLv2 | | PHP | MySql, Postgre Sql, MS Sql Server | Si | Si |
| Bugzilla | MPL | Apache (Unix, Windows, Macintosh) | Perl | MySql, Oracle, Postgre Sql | Si | Si |



La mayoría de las herramientas se descartan por el tipo de servidor en el que necesitan ser instaladas o por la base de datos necesaria. En nuestro caso los servidores que tiene el cliente son Windows Server 2008 R2 y la base de datos SQL Server 2008 R2.

Las herramientas marcadas en verde son las que se podrían usar por el tipo de requerimientos que se necesitan para su instalación.

A continuación se muestra una pequeña descripción y las principales características de las herramientas elegidas.

2.1.1 Otrs (Open Ticket Request System)

Otrs es una aplicación para la gestión y respuesta de solicitudes de los clientes de una empresa. Es un sistema de solicitud de ticket basado en código abierto y distribuido bajo la Licencia Publica General (GPL), con ciertas características para gestionar llamadas, e-mails o faxes de los clientes.

Puesto que no utiliza contenidos Web, como las applets de Java y Flash, este sistema puede ser usado desde cualquier navegador Web, tanto de teléfonos móviles como ordenadores personales, además de que no requiere de un sistema operativo en especial, sino únicamente un navegador Web. Otrs está basado en varios componentes, el componente básico lo constituye el framework Otrs, el cual contiene todas las funciones centrales para la aplicación y el sistema de ticket y desde el cual se pueden instalar aplicaciones adicionales como una red de correo electrónico, un gestor de contenidos y uno de archivos, y una herramienta para supervisar el estado del sistema de información. Las características principales del framework Otrs son:

1-Interfaz Web:

- Fácil manejo con un navegador Web.
- Soporte para varios lenguajes.
- Una interfaz Web para manejar las peticiones del cliente por parte de los empleados / agentes a través de la Web se integra.

2-Interfaz de correo:

- Soporte para archivos adjuntos de correo.
- Autorespuestas de los clientes, configurable por colas.
- Correos de notificaciones para los agentes sobre nuevos tickets, así como el seguimiento de los tickets abiertos.



3-Sistema:

- Otrs se ejecuta sobre cualquier sistema operativo (Linux, Solaris, FreeBSD, OpenBSD, Mac OS 10.x, Windows).
- Soporte ASP (active service providing).
- Autenticación de clientes por medio de la base de datos.
- Soporte para diferentes base de datos, por ejemplo MySQL, PostgreSQL, Oracle, MS Sql Server.

Este sistema está concebido para permitirles a las empresas responder rápidamente a los requerimientos y solicitudes que reciben de sus clientes, reduciendo costos y aumentando la satisfacción de los clientes.

En Otrs, todas las incidencias, solicitudes de información, reclamos, etc.; es decir, todos aquellos requerimientos de los clientes de la empresa, son tickets, independientemente si son recibidos como e-mails o llamadas telefónicas. Todos los tickets son gestionados como e-mails normales y se almacenan en forma de texto en el disco duro, las cabeceras se almacenan como base de datos para poder clasificar los tickets y darles respuesta.

Otra particularidad de esta aplicación es que posee un sistema de cola, que es en donde se almacenan los tickets, y desde donde sólo un agente los puede mover de una cola a otra, dependiendo del tema y/o según los criterios establecidos.

2.1.2 Mantis Bug Tracker

Es un software que constituye una solución completa para gestionar tareas en un equipo de trabajo. Es una aplicación OpenSource realizada con php y mysql que destaca por su facilidad y flexibilidad de instalar y configurar. Esta aplicación se utiliza para testear soluciones, hacer un registro histórico de alteraciones y gestionar equipos remotamente.

Esta aplicación permite la creación de diversas cuentas de usuario desde las cuales se puede informar de los bugs detectados. Con Mantis se puede dividir un proyecto en varias categorías, lo cual permite hacer un seguimiento más exacto de éste. El flujo de trabajo también se puede configurar desde la propia herramienta, de forma que puede definirse quién puede causar problemas, quién puede analizarlos y quién puede atenderlos.

Una de las características de Mantis es el gran abanico de posibilidades que ofrece para su configuración, nombradas a continuación:

- Permite configurar la transición de estados (abierto, encaminado, testeado, devuelto, cerrado, reabierto...).
- Se puede especificar un número indeterminado de estados para cada tarea (abierta, encaminada, testeada, devuelta, cerrada, reabierta...).
- Permite introducir diferentes perfiles (programador, tester, coordinador, visualizador...).
- Se permite la carga de plugins programados para esta plataforma, los que añaden ciertas funcionalidades extra.

Mantis incluye filtros, un sistema de búsqueda, tiene soporte para varios idiomas y también informa por e-mail de la resolución de los errores de los que se ha informado.

Las finalidades de esta aplicación son las siguientes:

- Agilizar el proceso de atención al cliente en las áreas de servicios informáticos, redes y cableado telefónico.
- Eliminar el uso de papel en cuanto a la recepción de solicitudes de servicio correspondiente.
- Permitir tener un mayor control sobre el tiempo que los técnicos abarcan en un orden de servicio.
- Optimizar el tiempo de los técnicos, permitiéndoles observar sus tareas desde cualquier lugar.
- Proporcionar al usuario la facilidad de observar en qué etapa del proceso se encuentra el servicio solicitado.
- Disminuir las entradas de solicitudes por línea telefónica.
- Obtener de manera rápida información de donde se emplea el tiempo por parte del personal de soporte
- Obtener estadísticas de empleo de tiempo y de estado del servicio

2.1.3 Motivos por los que se ha elegido desarrollar la aplicación

Aunque las dos herramientas descritas anteriormente se podrían adaptar a nuestras necesidades al final se acaba optando por desarrollarla por los siguientes motivos:

- Para mantener la apariencia que hay en el resto de la intranet y no suponga un gran cambio para el cliente.
- En caso de querer añadir nuevas funcionalidades no habría ningún problema en modificar un código desarrollado por el propio equipo.
- En el caso del Sistema de gestión de incidencias como son consultas con opciones muy personales hubiera resultado difícil encontrar una herramienta que se adaptara exactamente a las necesidades.
- Algo necesario es que puedan exportarse los datos de las consultas desde la aplicación a un documento Excel.

2.2 Comparación con el sistema utilizado previamente

Antes de la implantación del Sistema de gestión y notificación de incidencias el procedimiento a seguir era el siguiente:

El usuario se pone en contacto con el CAU por teléfono o correo electrónico tanto para informar de que algo no funciona correctamente como para preguntar dudas referentes al funcionamiento de las aplicaciones de la intranet. A continuación el CAU notifica el error al equipo de desarrollo ya sea también por teléfono o correo electrónico, se toma nota de la incidencia y se van resolviendo por orden de entrada o de urgencia.

Cuando se resuelve se avisa al CAU y este avisa de nuevo al usuario.

Esta forma de resolver las incidencias aunque lleva tiempo haciéndose así no es la más adecuada ya que es difícil mantener un control de las incidencias ya solucionadas o si ya se ha avisado al CAU de su solución para que a su vez avise al usuario.

Con este procedimiento el CAU solía tener muchos picos de trabajo ya que se le acumulaban las dudas que tenían los usuarios con las incidencias que entraban. Con este PFC las incidencias de los usuarios van a llegar directamente al equipo de desarrollo ahorrándose el paso intermedio del CAU.

A continuación se muestran unos diagramas con el flujo de trabajo que se seguía al principio y el que se va a seguir con el nuevo sistema.

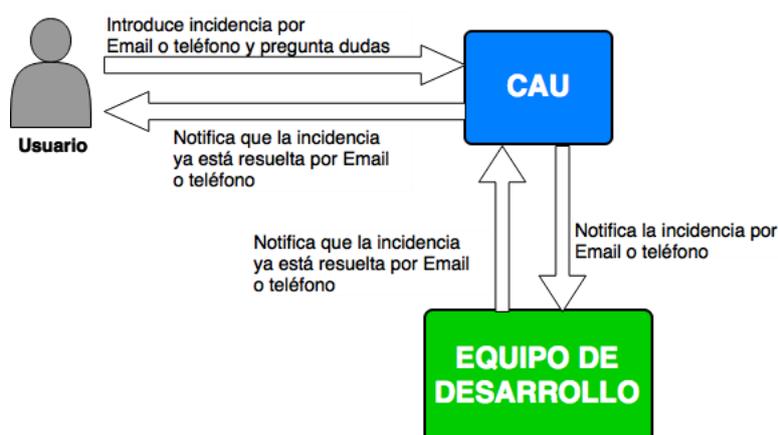


Ilustración 1: Flujo de trabajo anterior

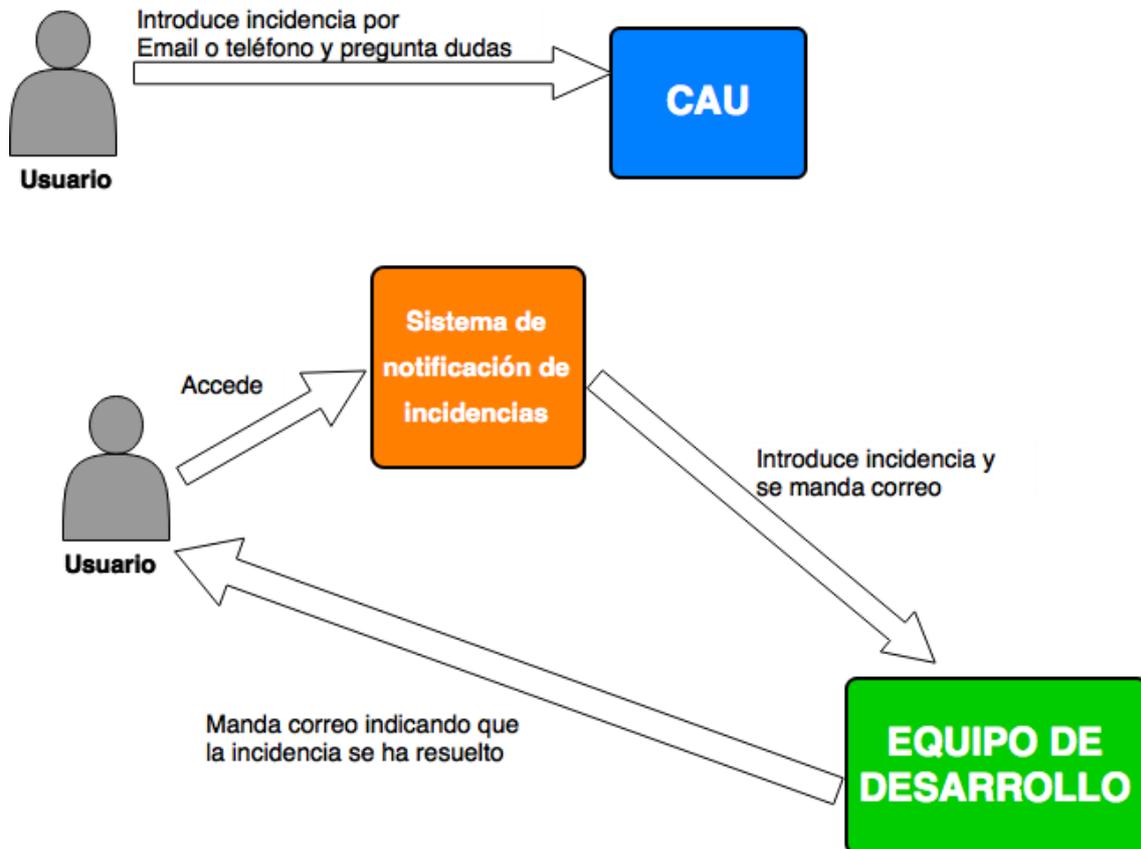


Ilustración 2: Flujo de trabajo con el Sistema de gestión y notificación de incidencias

Como se puede ver en el procedimiento que se ha seguido hasta ahora había mucho intercambio de mensajes y el CAU tenía un papel importante en todo el proceso.

Con el nuevo sistema el CAU sólo se va a dedicar a resolver dudas y en cuanto a las incidencias, el usuario y el equipo de desarrollo tendrán un trato más directo sin intermediarios.

3 SOFTWARE UTILIZADO

En este punto se describen las herramientas y programas que se han empleado durante el desarrollo del proyecto.

Se ha elegido este software porque el cliente es el que ha validado y paga las licencias. Además de que es el que ya está implantado.

- **Microsoft Visual Studio 2010:** es un conjunto completo de herramientas de desarrollo para sistemas operativos Windows. Se pueden generar aplicaciones web ASP.NET, Servicios Web XML, aplicaciones de escritorio y aplicaciones móviles.

Soporta varios lenguajes de programación tales como [Visual Basic](#), [Visual C#](#) y [Visual C++](#). Asimismo, dichos lenguajes utilizan las funciones de .NET Framework, las cuales ofrecen acceso a tecnologías clave para simplificar el desarrollo de aplicaciones web ASP y Servicios Web XML.

Para este proyecto se ha usado ASP .NET y Visual C# a través de los cuales se han creado dos aplicaciones web, utilizando como servidor Internet Information Services (IIS).

Para poder desarrollar con Silverlight (este complemento se explica en el punto 3) ha sido necesario instalar Microsoft Visual Studio 2010 SP1.



Ilustración 3: Logotipo de Visual Studio 2010

- **.NET Framework 3.5:** es un componente integral de Windows que admite la creación y la ejecución de la siguiente generación de aplicaciones y servicios Web XML.

El diseño de .NET Framework está enfocado a cumplir los objetivos siguientes:

- Proporcionar un entorno coherente de programación orientada a objetos, en el que el código de los objetos se pueda almacenar y ejecutar de forma local, ejecutar de forma local pero distribuida en Internet o ejecutar de forma remota.
- Proporcionar un entorno de ejecución de código que elimine los problemas de rendimiento de los entornos en los que se utilizan secuencias de comandos o intérpretes de comandos.
- Proporcionar un entorno de ejecución de código que reduzca lo máximo posible la implementación de software y los conflictos de versiones.
- Basar toda la comunicación en estándares del sector para asegurar que el código de .NET Framework se puede integrar con otros tipos de código.
- Ofrecer un entorno de ejecución de código que fomente la ejecución segura del mismo, incluso del creado por terceras personas desconocidas o que no son de plena confianza.
- Ofrecer al programador una experiencia coherente entre tipos de aplicaciones muy diferentes, como las basadas en Windows o en el Web.

.NET Framework contiene dos componentes principales: Common Language Runtime y la biblioteca de clases de .NET Framework.

- Common Language Runtime es el fundamento de .NET Framework.
- La biblioteca de clases es una completa colección orientada a objetos de tipos reutilizables que se pueden emplear para desarrollar aplicaciones que abarcan desde las tradicionales herramientas de interfaz gráfica de usuario (GUI) o de línea de comandos hasta las aplicaciones basadas en las innovaciones más recientes proporcionadas por ASP.NET, como los formularios Web Forms y los servicios Web XML.



Ilustración 4: Logotipo de .NET Framework

- **Microsoft Silverlight 5:** Microsoft Silverlight es un complemento de Microsoft que nos permite desarrollar aplicaciones enriquecidas para la web. Agrega nuevas funciones multimedia como la reproducción de vídeos, gráficos vectoriales, animaciones e interactividad, en forma similar a lo que hace Adobe Flash.

Silverlight funciona sobre varias plataformas y múltiples exploradores y proporciona una nueva generación de experiencias de usuario basadas en .NET. Tras una descarga que se instala en segundos, Silverlight posibilita una nueva plataforma rica, segura y escalable.

Silverlight ofrece un modelo de programación flexible y coherente compatible con lenguajes .NET como Visual Basic y C#. Silverlight supone una nueva forma de aprovechar los elementos multimedia en los principales navegadores entre los que se incluyen Firefox, Safari e Internet Explorer tanto en MacOS como en Windows.

La base de su programación es XAML y el acceso a los objetos está dado por C# y Visual Basic.

Para su instalación es necesario tener instalado también Microsoft Visual Studio 2010 SP1.



Ilustración 5: Logotipo de Microsoft Silverlight

- **Internet Explorer (de la versión 9 en adelante):** Navegador web desarrollado por Microsoft para el sistema operativo Microsoft Windows. La herramienta desarrollada en este proyecto está optimizada para su uso en Internet Explorer.



Ilustración 6: Logotipo de Internet Explorer

- **Microsoft SQL Server 2008 R2:** es un sistema de administración y análisis de bases de datos relacionales desarrollado por la empresa Microsoft para soluciones de comercio electrónico, línea de negocio y almacenamiento de datos. Le permite almacenar datos de documentos estructurados, semiestructurados o no estructurados como son las imágenes, música y archivos directamente dentro de la base de datos. SQL Server 2008 le ayuda a obtener más rendimiento de los datos, poniendo a su disposición una amplia gama de servicios integrados como son consultas, búsquedas, sincronizaciones, informes y análisis. Sus datos pueden almacenarse y recuperarse desde sus servidores más potentes del Data Center hasta los desktops y dispositivos móviles, permitiéndole tener un mayor control sobre la información sin importar dónde se almacena físicamente. SQL Server 2008 le permite utilizar sus datos en aplicaciones a medida desarrolladas con Microsoft® .NET y Visual Studio y también desde su propia Arquitectura Orientada a Servicio (SOA).



Ilustración 7: Logotipo de SQL Server 2008 R2

- **Internet Information Services o IIS:** es un servidor web y un conjunto de servicios para el sistema operativo Microsoft Windows. Originalmente era parte del *Option Pack* para Windows NT. Luego fue integrado en otros sistemas operativos de Microsoft destinados a ofrecer servicios, como Windows 2000 o Windows Server 2003. Windows XP Profesional incluye una versión limitada de IIS. Los servicios que ofrece son: FTP, SMTP, NNTP y HTTP/HTTPS. Este servicio convierte un ordenador en un servidor web para Internet o una intranet, es decir que en las computadoras que tienen este servicio instalado se pueden publicar páginas web tanto local como remotamente. Se basa en varios módulos que le dan capacidad para procesar distintos tipos de páginas. Por ejemplo, Microsoft incluye los de Active Server Pages (ASP) y ASP.NET.



Ilustración 8: Logotipo de IIS

- **Microsoft Office 2013:** Es un conjunto de programas que abarca e interrelaciona aplicaciones de escritorio, servidores y servicios para los sistemas operativos Microsoft Windows y Mac OS X. Es compatible con [Windows 7](#), [Windows 8.1](#), [Windows Server 2012](#) y [Windows Server 2012 R2](#). Es una versión sucesora de Microsoft Office 2010.

Office 2013 está disponible como parte de Windows RT para procesadores ARM y por separado para las versiones de Windows para plataformas de 32 y 64 bits (x86 y x64). Esta edición cuenta con la ya tradicional interfaz Ribbon UI al igual que sus predecesoras, y se destaca por presentar la imagen simple de la interfaz Modern UI (antes Metro UI).

Esta versión de Office se destaca por ser la primera pensada especialmente para dispositivos táctiles.

Entre los distintos programas que se encuentran en Office, se han utilizado los siguientes para la realización del proyecto

- **Microsoft Office Word:** Procesador de texto.
- **Microsoft Office Excel:** Hoja de cálculo.
- **Microsoft Office PowerPoint:** Sirve para desarrollar y desplegar presentaciones visuales. Es usado para crear diapositivas multimedia, es decir, compuestas por texto, imágenes, sonido y vídeos.



Ilustración 9: Logotipo de Microsoft Office 2013



4 ANÁLISIS

4.1 Roles

En las dos aplicaciones de este proyecto podemos encontrar dos tipos de roles:

- **Usuario:** es cualquier persona que tiene acceso a la intranet y además está dada de alta en la tabla Usuarios de la base de datos del proyecto para poder acceder.
- **Desarrollador:** es la persona que además de cumplir con los requisitos del rol de Usuario también está dado de alta en la tabla Desarrolladores de la base de datos del proyecto para poder entrar a ciertas aplicaciones como es el caso del Sistema de gestión de incidencias.

4.2 Requisitos

En este punto se muestran de forma detallada los requisitos de las aplicaciones.

Un requisito es una condición o capacidad que un usuario necesita para poder resolver un problema o lograr un objetivo.

Existen dos tipos de requisitos.

- Requisito funcional: puede ser una descripción de lo que un sistema debe hacer.
- Requisito no funcional: especifica una capacidad sobre el propio sistema y cómo debe realizar sus funciones.

Para su descripción se usará la siguiente plantilla:

Tabla 2: Esquema general de los requisitos

| | | | |
|--|--|------------------|--|
| Código | | | |
| Nombre | | | |
| Necesidad | | Prioridad | |
| Verificabilidad | | Fuente | |
| Descripción | | | |
| Requisitos relacionados | | | |
| Pre-requisitos y pos-requisitos | | | |

A continuación se explican los campos de la plantilla:

- **Código:** Sirve para la identificación de cada requisito. Se asigna de forma única y su formato es el siguiente “RF#” o “RNF#”, donde “RF” indica si es un requisito funcional, “RNF” indica si es un requisito no funcional y “#” es el número asignado a este requisito.
- **Nombre:** Breve resumen del contenido del requisito.
- **Necesidad:** Indica el nivel de necesidad que tiene el requisito: **Obligatorio, Deseable y Opcional.**
- **Verificabilidad:** Indica si se puede comprobar que el requisito ha sido añadido al diseño.
- **Prioridad:** Indica la dependencia que se tiene por el desarrollo del requisito, se divide en tres niveles: **Alta, Media y Baja.**
- **Fuente:** Indica si el requisito lo ha propuesto el **cliente** o la **empresa.**

- **Descripción:** Explicación detallada y clara del requisito.
- **Requisitos relacionados:** Lista de requisitos que se puedan relacionar de alguna forma o sean importantes para el entendimiento del requisito en cuestión.
- **Pre-requisitos y pos-requisitos:** requisitos que son necesarios a parte de los indicados.

4.2.1 Requisitos funcionales

4.2.1.1 Validación de usuarios y acceso a las aplicaciones

Tabla 3: Requisitos – RF01: Usuarios registrados en base de datos

| | | | |
|--|--|------------------|---------|
| Código | RF01 | | |
| Nombre | Usuarios registrados en base de datos | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | Los usuarios que vayan a utilizar la intranet deberán estar dados de alta en la tabla Usuarios en la base de datos del proyecto. | | |
| Requisitos relacionados | | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

Tabla 4: Requisitos – RF02: Desarrolladores registrados en base de datos

| | | | |
|--|---|------------------|---------|
| Código | RF02 | | |
| Nombre | Desarrolladores registrados en base de datos | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | Los usuarios que vayan a tener el rol de Desarrollador, además de cumplir con los requisitos de Usuario deberán estar dados de alta en la tabla Desarrolladores en la base de datos del proyecto. | | |
| Requisitos relacionados | RF01 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

Tabla 5: Requisitos – RF03: Acceder al Sistema de notificación de incidencias

| | | | |
|--|--|------------------|---------|
| Código | RF03 | | |
| Nombre | Acceder al Sistema de notificación de incidencias | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | El usuario debe pulsar el botón de Sistema de notificación de incidencias que se encontrará en cualquier página de la intranet. Al acceder se cargarán algunos datos del usuario conectado. Se cargarán también las aplicaciones que haya dadas de alta en la base de datos del proyecto para poder registrar una incidencia. | | |
| Requisitos relacionados | RF01 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

Tabla 6: Requisitos – RF04: Acceder al Sistema de gestión de incidencias

| | | | |
|--|---|------------------|---------|
| Código | RF04 | | |
| Nombre | Acceder al Sistema de gestión de incidencias | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | El usuario debe entrar en el Sistema de gestión de incidencias al cual se accederá desde la intranet. Al acceder se cargarán algunos datos del usuario conectado. | | |
| Requisitos relacionados | RF01, RF02 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

4.2.1.2 Modificación de datos

Tabla 7: Requisitos – RF05: Modificar datos de contacto del usuario

| | | | |
|--|---|------------------|---------|
| Código | RF05 | | |
| Nombre | Modificar datos de contacto del usuario | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | <p>El usuario se encuentra en el Sistema de notificación de incidencias y debe pulsar en el botón “Modificar datos de contacto del usuario”. Una vez en esta ventana podrá hacer las siguientes acciones:</p> <ul style="list-style-type: none"> - Añadir o modificar el “Email alternativo”. - Añadir o modificar el “Teléfono”. - Añadir o modificar el “Móvil”. <p>Por ultimo pulsará el botón “Actualizar” y los datos se modificarán en base de datos del proyecto.</p> | | |
| Requisitos relacionados | RF01, RF03 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

4.2.1.3 Consulta de histórico de incidencias del usuario

Tabla 8: Requisitos – RF06: Consultar incidencias anteriores

| | | | |
|--|---|------------------|---------|
| Código | RF06 | | |
| Nombre | Consultar incidencias anteriores | | |
| Necesidad | Obligatorio | Prioridad | Media |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | <p>El usuario se encuentra en el Sistema de notificación de incidencias y debe pulsar en el botón “Ver mis incidencias”. Una vez en esta ventana podrá ver las incidencias que ha introducido a lo largo del tiempo, el estado en el que se encuentran y más información acerca de la incidencia.</p> | | |
| Requisitos relacionados | RF01, RF03 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

4.2.1.4 Registro de incidencias en base de datos

Tabla 9: Requisitos – RF07: El usuario da de alta una incidencia

| | | | |
|--|--|------------------|---------|
| Código | RF07 | | |
| Nombre | El usuario da de alta una incidencia | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | El usuario se encuentra en el Sistema de notificación de incidencias. Primero debe rellenar una serie de datos y elegir a que correo quiere que se le notifique la resolución de la incidencia. Pulsar en el botón “Notificar”. -La incidencia se registrará en base de datos con criticidad alta. -Si la aplicación del error tiene un solo desarrollador se asigna a él directamente y se pone en estado “En desarrollo”. - Si la aplicación del error tiene varios desarrolladores no se asigna a nadie y se pone en estado “Sin asignar”. | | |
| Requisitos relacionados | RF01, RF03 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

Tabla 10: Requisitos – RF08: Se registra una incidencia por fallo de las aplicaciones

| | | | |
|--|---|------------------|---------|
| Código | RF08 | | |
| Nombre | Se registra una incidencia por fallo de las aplicaciones | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | El código de las aplicaciones falla por no haber contemplado algún caso, fallo de conexión, etc. -La incidencia se registrará en base de datos con criticidad baja. -Si la aplicación del error tiene un solo desarrollador se asigna a él directamente y se pone en estado “En desarrollo”. - Si la aplicación del error tiene varios desarrolladores no se asigna a nadie y se pone en estado “Sin asignar”. | | |
| Requisitos relacionados | RF01, RF02 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

4.2.1.5 Recepción de correos

Tabla 11: Requisitos – RF09: Recibir correo al registrar una incidencia el usuario

| | | | |
|--|---|------------------|---------|
| Código | RF09 | | |
| Nombre | Recibir correo al registrar una incidencia el usuario | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | Cuando un usuario registre una incidencia se mandará un correo al desarrollador de esa parte de la aplicación. - Si la aplicación del error tiene un solo desarrollador se enviará un correo a él directamente. - Si la aplicación del error tiene varios desarrolladores se enviará un correo a todos ellos. | | |
| Requisitos relacionados | RF01, RF02, RF07 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

Tabla 12: Requisitos – RF10: Recibir correo al registrar una incidencia por fallo de la aplicación

| | | | |
|--|--|------------------|---------|
| Código | RF10 | | |
| Nombre | Recibir correo al registrar una incidencia por fallo de la aplicación | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | Cuando la aplicación falle se mandará un correo al desarrollador de esa parte de la aplicación. - Si la aplicación del error tiene un solo desarrollador se enviará un correo a él directamente. - Si la aplicación del error tiene varios desarrolladores se enviará un correo a todos ellos. | | |
| Requisitos relacionados | RF01, RF02, RF08 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

Tabla 13: Requisitos – RF11: Recibir justificante al registrar una incidencia

| | | | |
|--|--|------------------|---------|
| Código | RF11 | | |
| Nombre | Recibir justificante al registrar una incidencia | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | El usuario recibe un correo con información de la incidencia que acaba de registrar. | | |
| Requisitos relacionados | RF01, RF03, RF07 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

Tabla 14: Requisitos – RF12: Recibir correo de incidencia solucionada

| | | | |
|--|--|------------------|---------|
| Código | RF12 | | |
| Nombre | Recibir correo de incidencia solucionada | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | El usuario recibe un correo indicando que la incidencia ha sido solucionada. | | |
| Requisitos relacionados | RF01, RF03, RF07, RF11 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

4.2.1.6 Consulta de incidencias registradas

Tabla 15: Requisitos – RF13: Consultar incidencias en desarrollo por el usuario conectado

| | | | |
|--|---|------------------|---------|
| Código | RF13 | | |
| Nombre | Consultar incidencias en desarrollo por el usuario conectado | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | El usuario se encuentra en el Sistema de gestión de incidencias y debe seleccionar la siguiente combinación: - Pulsar en “Mostrar incidencias del desarrollador conectado”. - No seleccionar el check de “Incidencias solucionadas”. Mostrará las incidencias registradas en base de datos cuya cuenta del Desarrollador del error coincida con la cuenta del Usuario conectado y se encuentren en estado “En desarrollo” Si el usuario está registrado en la tabla Desarrolladores de la base de datos y tiene incidencias en desarrollo podrá ver los datos en la consulta. | | |
| Requisitos relacionados | RF01, RF02, RF04 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

Tabla 16: Requisitos – RF14: Consultar incidencias solucionadas por el usuario conectado

| | | | |
|--|---|------------------|---------|
| Código | RF14 | | |
| Nombre | Consultar incidencias solucionadas por el usuario conectado | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | El usuario se encuentra en el Sistema de gestión de incidencias y debe seleccionar la siguiente combinación: - Pulsar en “Mostrar incidencias del desarrollador conectado”. - Seleccionar el check de “Incidencias solucionadas”. Mostrará las incidencias registradas en base de datos cuya cuenta del Desarrollador del error coincida con la cuenta del usuario conectado y se encuentren en estado “Solucionada” Si el usuario está registrado en la tabla Desarrolladores de la base de datos y tiene incidencias solucionadas podrá ver los datos en la consulta. | | |
| Requisitos relacionados | RF01, RF02, RF04 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

Tabla 17: Requisitos – RF15: Consultar incidencias sin asignar y en desarrollo por todos los desarrolladores

| | | | |
|--|---|------------------|---------|
| Código | RF15 | | |
| Nombre | Consultar incidencias sin asignar y en desarrollo por todos los desarrolladores | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | El usuario se encuentra en el Sistema de gestión de incidencias y debe seleccionar la siguiente combinación: - Pulsar en “Mostrar incidencias de todos desarrolladores”. - No seleccionar el check de “Incidencias solucionadas”. | | |
| Requisitos relacionados | RF01, RF02, RF04 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

Tabla 18: Requisitos – RF16: Consultar incidencias solucionadas por todos los desarrolladores

| | | | |
|--|--|------------------|---------|
| Código | RF16 | | |
| Nombre | Consultar incidencias solucionadas por todos los desarrolladores | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | <p>El usuario se encuentra en el Sistema de gestión de incidencias y debe seleccionar la siguiente combinación:</p> <ul style="list-style-type: none"> - Pulsar en “Mostrar incidencias de todos desarrolladores”. - Seleccionar el check de “Incidencias solucionadas”. | | |
| Requisitos relacionados | RF01, RF02, RF04 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

4.2.1.7 Asignación de incidencias

Tabla 19: Requisitos – RF17: Asignar incidencia manualmente

| | | | |
|--|---|------------------|---------|
| Código | RF17 | | |
| Nombre | Asignar incidencia manualmente | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | <p>El usuario se encuentra en el Sistema de gestión de incidencias y pulsa el botón “Incidencias sin asignar” Mostrará las incidencias registradas en base de datos cuya cuenta del Desarrollador del error coincida con la cuenta del Usuario conectado y se encuentren en estado “Sin asignar”.</p> <p>Al pulsar en el botón “Asignarme incidencia” se modificará el desarrollador de dicha incidencia poniendo al usuario conectado y se modificará también el estado que pasará a estar “En desarrollo”</p> | | |
| Requisitos relacionados | RF01, RF02, RF04 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

4.2.1.8 Solución de incidencias

Tabla 20: Requisitos – RF18: Cerrar incidencia

| | | | |
|--|---|------------------|---------|
| Código | RF18 | | |
| Nombre | Cerrar incidencia | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | El usuario se encuentra en el Sistema de gestión de incidencias y configura la consulta para poder ver las incidencias que se encuentren en desarrollo. Si tiene alguna incidencia “En desarrollo” que le corresponda pulsará el botón “Cerrar incidencia”. Se abrirá otra ventana, rellenará los datos necesarios y pulsará en “Aceptar”. Se modificará la incidencia en base de datos. | | |
| Requisitos relacionados | RF01, RF02, RF04 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

4.2.1.9 Generación de informes

Tabla 21: Requisitos – RF20: Generar informes

| | | | |
|--|--|------------------|---------|
| Código | RF20 | | |
| Nombre | Generar informes | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | El usuario se encuentra en el Sistema de gestión de incidencias. Configura la consulta como el usuario desee. Pulsa en el botón “Exportar datos”. | | |
| Requisitos relacionados | RF01, RF02, RF04 | | |
| Pre-requisitos y pos-requisitos | Para poder entrar en la intranet hay que identificarse en el equipo del puesto de trabajo y se usará esa cuenta para validar al usuario. | | |

4.2.2 Requisitos no funcionales

Tabla 22: Requisitos – RNF01: Trabajar con Windows

| | | | |
|--|---|------------------|---------|
| Código | RNF01 | | |
| Nombre | Trabajar con Windows | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | El usuario debe tener instalado en su equipo de trabajo el sistema operativo Windows. | | |
| Requisitos relacionados | | | |
| Pre-requisitos y pos-requisitos | | | |

Tabla 23: Requisitos – RNF02: Trabajar con Internet Explorer

| | | | |
|--|--|------------------|---------|
| Código | RNF02 | | |
| Nombre | Trabajar con Internet Explorer | | |
| Necesidad | Obligatorio | Prioridad | Alta |
| Verificabilidad | Verificable | Fuente | Cliente |
| Descripción | El usuario debe tener instalado en su equipo de trabajo el explorador Internet Explorer. | | |
| Requisitos relacionados | | | |
| Pre-requisitos y pos-requisitos | | | |

4.3 Diagrama y especificación de Casos de Uso

En este punto se muestran de forma detallada los casos de uso de las aplicaciones.

Un caso de uso es una descripción de las actividades que deberán realizarse para llevar a cabo algún proceso. Los personajes o entidades que participarán en un caso de uso se denominan actores.

Para su descripción se usará la siguiente plantilla:

Tabla 24: Esquema general de la especificación de los casos de uso

| | |
|---------------------|--|
| Caso de Uso | |
| Actor | |
| Descripción | |
| Flujo básico | |

A continuación se explican los campos de la plantilla:

- **Caso de Uso:** es el nombre del Caso de uso. El CU debe comenzar por un verbo y ser lo más corto posible, pero que a su vez, describa lo que el CU hace. El nombre del CU comienza por su identificación CU#, donde # es el número asignado a este CU.
- **Actor:** es el actor/es que interactúan directamente con el sistema.
- **Descripción:** es un párrafo que resume el objetivo del caso de uso. Sin dar detalles del cómo, la descripción del caso de uso resume todo lo que el caso de uso hace, que es el valor que da al actor (o actores) primario, así como la necesidad de la existencia de actores secundarios, para los casos que aplique.
- **Flujo básico:** El flujo básico describe los pasos que se sucederían en un escenario simple, sin ramificaciones.

Cada paso del flujo básico contiene:

- 1) un número de paso o flujo
- 2) título del paso. Representa un resumen de lo que el paso realiza, suele ser una oración que inicia con un verbo en estado activo
- 3) la descripción del paso. La descripción del paso contiene el detalle de lo que se espera que ocurra en el paso.

4.3.1 Caso de Uso: Usuario

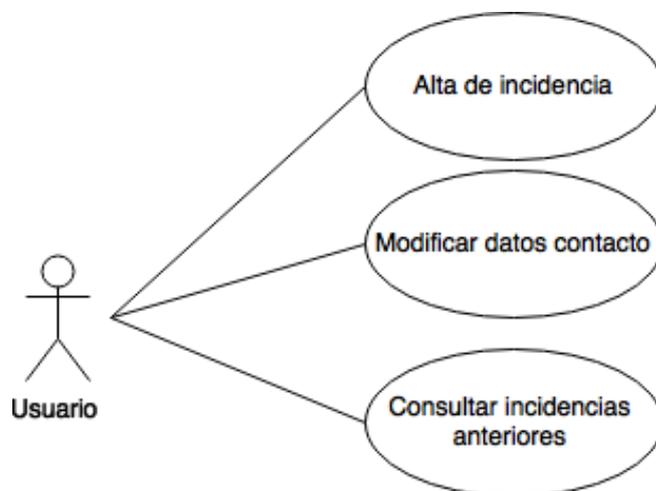


Ilustración 10: Diagrama Casos de uso – Usuario

Tabla 25: Especificación de Casos de Uso – CU01: Alta de incidencia

| | |
|---------------------|---|
| Caso de Uso | CU01: Alta de incidencia |
| Actor | Usuario. |
| Descripción | El usuario debe rellenar los datos necesarios para poder introducir una incidencia en el sistema. |
| Flujo básico | <ol style="list-style-type: none"> 1. Seleccionar un origen del error. 2. Explicar brevemente en el campo “Descripción” que es lo que ha sucedido. <ol style="list-style-type: none"> 2.1. Si tiene registrado el “Email alternativo” puede elegir entre seleccionarlo o no para que le notifiquen en ese email. 3. Pulsar el botón “Notificar”. |

Tabla 26: Especificación de Casos de Uso – CU02: Modificar datos de contacto

| | |
|---------------------|---|
| Caso de Uso | CU02: Modificar datos de contacto |
| Actor | Usuario. |
| Descripción | El usuario modifica sus datos de contacto. |
| Flujo básico | <ol style="list-style-type: none"> 1. Pulsar el botón “Modificar datos de contacto del usuario”. 2. Añadir o modificar el “Email alternativo”. 3. Añadir o modificar el “Teléfono”. 4. Añadir o modificar el “Móvil”. 5. Pulsar el botón “Actualizar”. |



Tabla 27: Especificación de Casos de Uso – CU03: Consultar incidencias anteriores

| | |
|---------------------|--|
| Caso de Uso | CU03: Consultar incidencias anteriores |
| Actor | Usuario. |
| Descripción | El usuario consulta su histórico de incidencias registradas. |
| Flujo básico | 1. Pulsar el botón “Ver mis incidencias”. |

4.3.2 Caso de Uso: Desarrollador

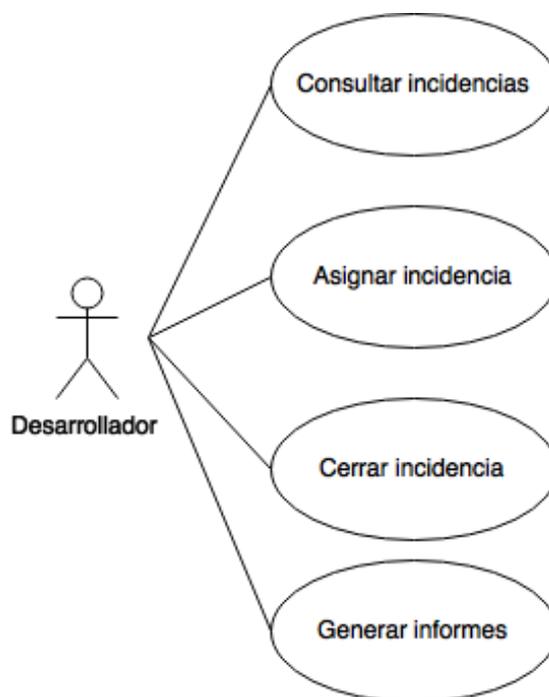


Ilustración 11: Diagrama Casos de uso – Desarrollador

Tabla 28: Especificación de Casos de Uso – CU04: Consultar incidencias

| | |
|---------------------|--|
| Caso de Uso | CU04: Consultar incidencias |
| Actor | Desarrollador. |
| Descripción | El desarrollador consulta las incidencias que se encuentran registradas en el sistema. |
| Flujo básico | 1. Configurar la consulta como desee el desarrollador. |

Tabla 29: Especificación de Casos de Uso – CU05: Asignar incidencia

| | |
|---------------------|---|
| Caso de Uso | CU05: Asignar incidencia |
| Actor | Desarrollador. |
| Descripción | El desarrollador se asigna una incidencia de la consulta de incidencias sin asignar. |
| Flujo básico | <ol style="list-style-type: none"> 1. Pulsar en “Incidencias sin asignar”. 2. Pulsar en “Asignarme incidencia”. <ol style="list-style-type: none"> 2.1. Pulsar “Aceptar” si quiere confirmar la acción. 2.2. Pulsar “Cancelar” si quiere cancelar la acción. |

Tabla 30: Especificación de Casos de Uso – CU06: Cerrar incidencia

| | |
|---------------------|---|
| Caso de Uso | CU06: Cerrar incidencia |
| Actor | Desarrollador. |
| Descripción | El desarrollador pone una fecha de solución a una incidencia y esta se da por finalizada. |
| Flujo básico | <ol style="list-style-type: none">1. Pulsar en “Cerrar incidencia”.2. Escribir una “Fecha solución”.3. Escribir una “Acción correctora” y “Comentarios” si lo ve conveniente.4. Pulsar en “Guardar”. |

Tabla 31: Especificación de Casos de Uso – CU07: Generar informes

| | |
|---------------------|--|
| Caso de Uso | CU07: Generar informes |
| Actor | Desarrollador. |
| Descripción | El desarrollador exporta los datos de las consultas a un Excel. |
| Flujo básico | <ol style="list-style-type: none">1. Pulsar en “Exportar datos”.<ol style="list-style-type: none">1.1. Seleccionar “Selección actual” si se quiere exportar una serie de datos filtrados.1.2. Seleccionar “Todos los datos” si se quieren exportar todos los datos. |

5 DISEÑO

5.1 Arquitectura

A continuación se explica la arquitectura interna de Silverlight, cómo funciona y cómo se comunica el servidor (capa de negocio), con el cliente (capa de presentación) y la capa de datos (base de datos).

Capa de presentación: la que ve el usuario (también se la denomina "capa de usuario"), presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso. También es conocida como interfaz gráfica y debe tener la característica de ser "amigable" (entendible y fácil de usar) para el usuario. Esta capa se comunica únicamente con la capa de negocio.

Capa de negocio: es la capa donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos, almacenar o recuperar datos de él. También se consideran aquí los programas de aplicación.

Capa de datos: es la capa donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

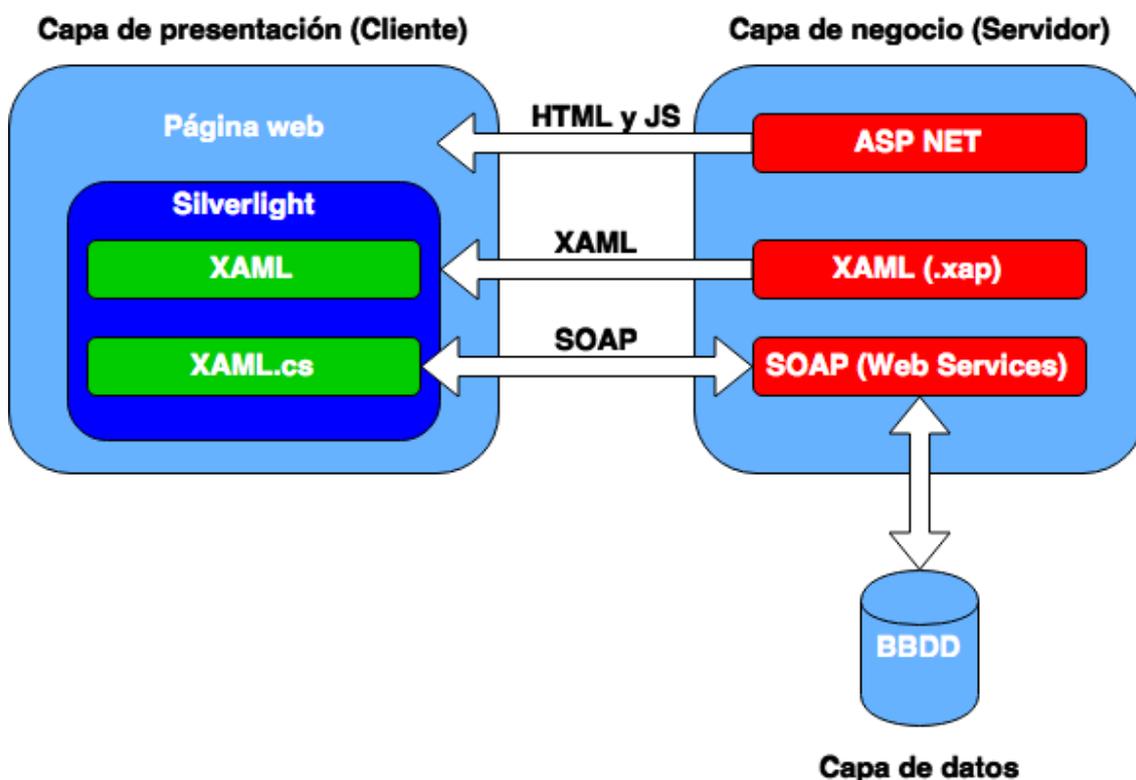


Ilustración 12: Arquitectura

Se puede ver que el servidor se conecta a través de ASP NET por medio de HTML y JS para llegar a la página web. Una vez se dispone de la página de Silverlight se envía el XAML desde el servidor hasta el cliente. Por último se encuentra la transferencia de datos por SOAP. En este paso pueden suceder dos cosas: que este recupere los datos de base de datos y posteriormente termine en los archivos de código XAML.cs para así completar el traspaso de información, o que el código XAML.cs sea el que envíe una inserción o modificación de datos y a través de SOAP se actualicen en base de datos.



5.2 Diagrama de Clases

Un diagrama de clases es un tipo de diagrama de estructura estática que describe la estructura de un sistema mostrando las clases del sistema, atributos, operaciones (o métodos) y las relaciones entre los objetos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargarán del funcionamiento y la relación entre uno y otro.

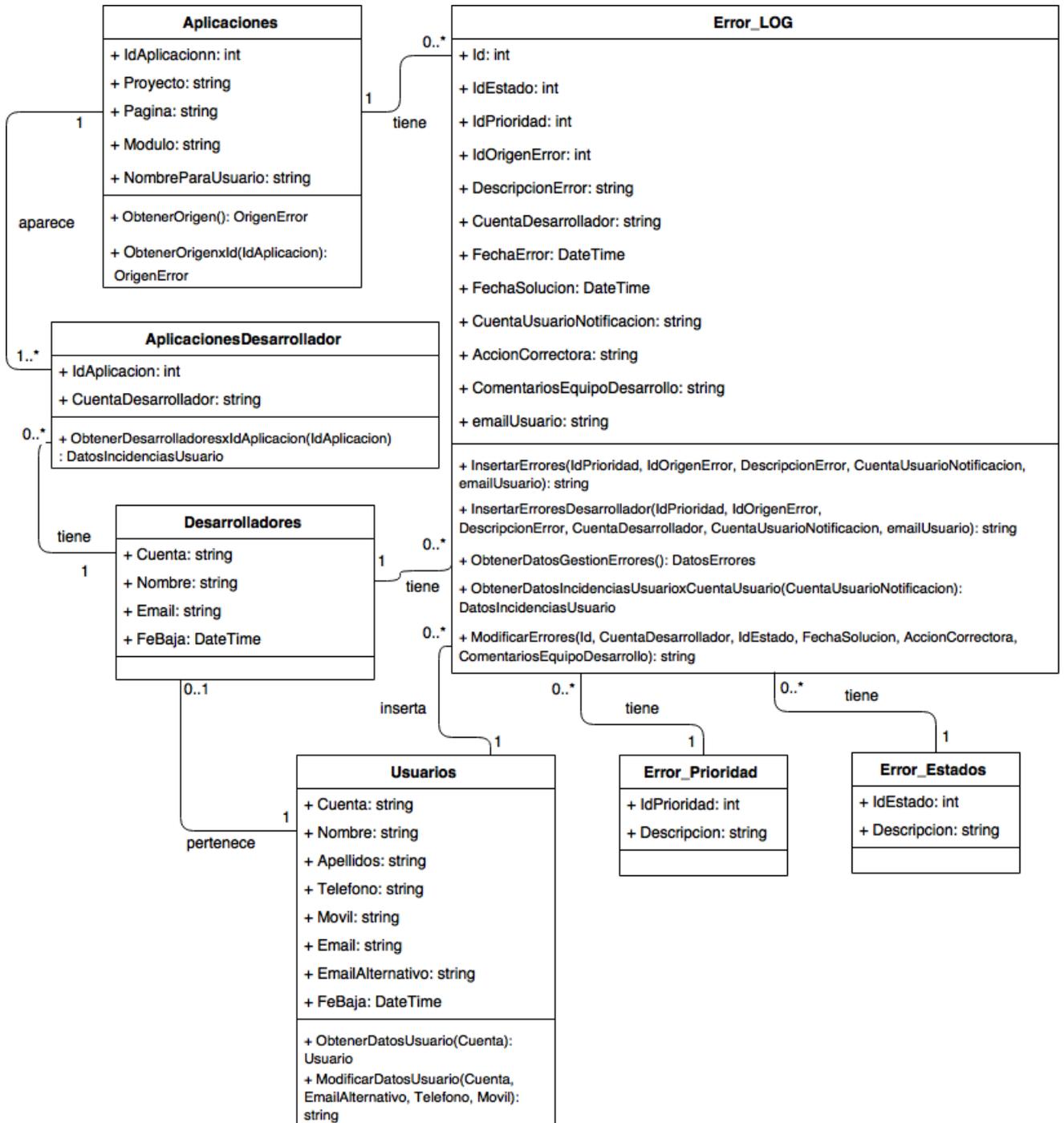


Ilustración 13: Diagrama de Clases

5.3 Diagrama de Secuencia

Un diagrama de secuencia en UML muestra la forma en la que los objetos se comunican entre sí al transcurrir el tiempo.

El diagrama contiene:

- Objetos con sus líneas de vida.
- Mensajes intercambiados entre objetos en una secuencia ordenada.

5.3.1 Diagrama de Secuencia: Alta de incidencia

En este diagrama aparece un Usuario a la hora de insertar una incidencia.

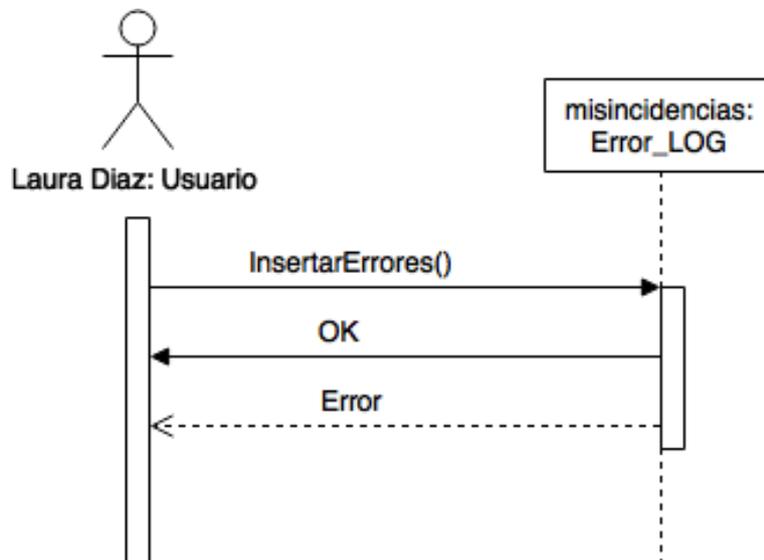


Ilustración 14: Diagrama de Secuencia – Alta de incidencia

5.3.2 Diagrama de Secuencia: Modificar datos de contacto

En este diagrama aparece un Usuario a la hora de modificar sus datos de contacto.

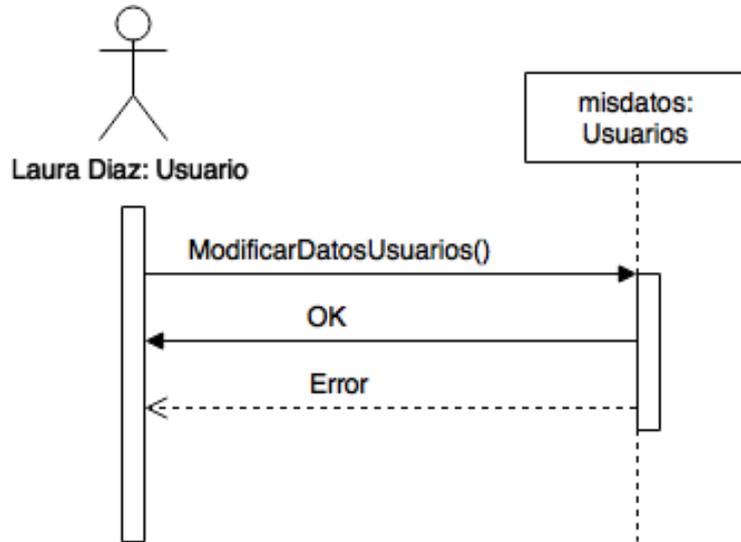


Ilustración 15: Diagrama de Secuencia – Modificar datos de contacto

5.3.3 Diagrama de Secuencia: Consultar incidencias anteriores

En este diagrama aparece un Usuario a la hora consultar su historial de incidencias.

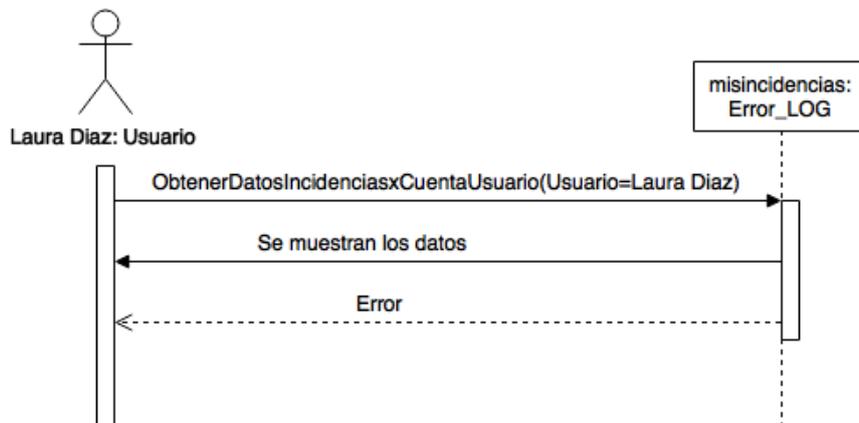


Ilustración 16: Diagrama de Secuencia – Consultar incidencias anteriores

5.3.4 Diagrama de Secuencia: Consultar incidencias

En este diagrama aparece un Desarrollador a la hora de consultar las incidencias registradas en el sistema.

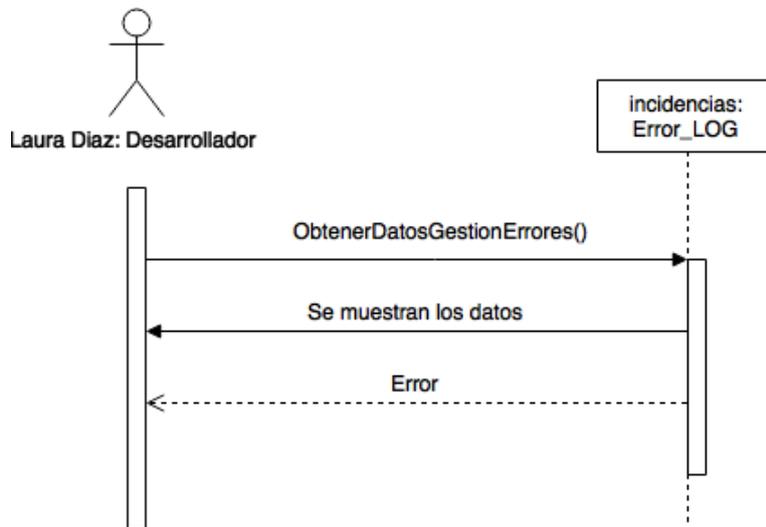


Ilustración 17: Diagrama de Secuencia – Consultar incidencias

5.3.5 Diagrama de Secuencia: Asignar incidencia

En este diagrama aparece un Desarrollador a la hora de asignarse una incidencia.

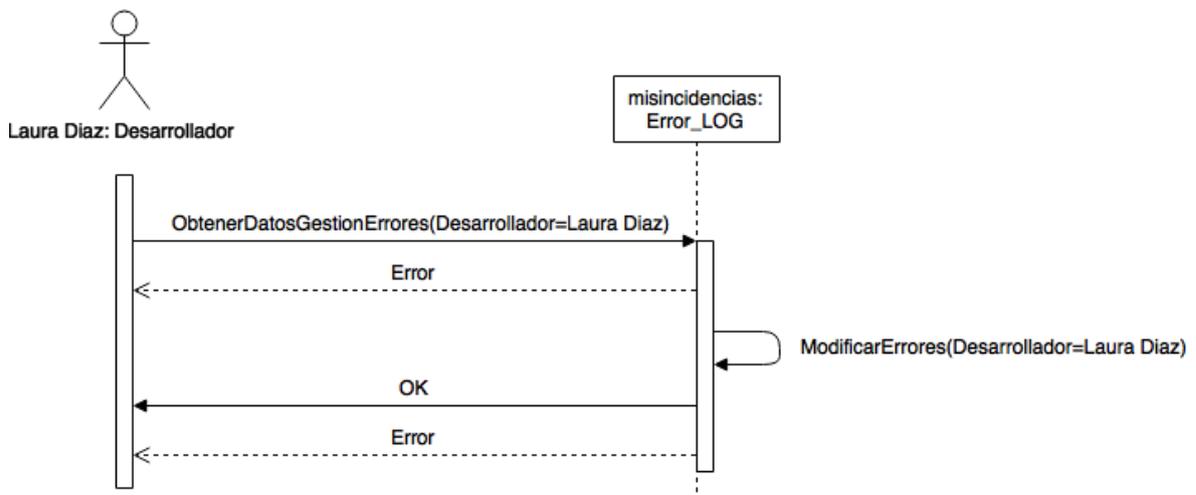


Ilustración 18: Diagrama de Secuencia – Asignar incidencia

5.3.6 Diagrama de Secuencia: Cerrar incidencia

En este diagrama aparece un Desarrollador a la hora de dar por finalizada una incidencia.

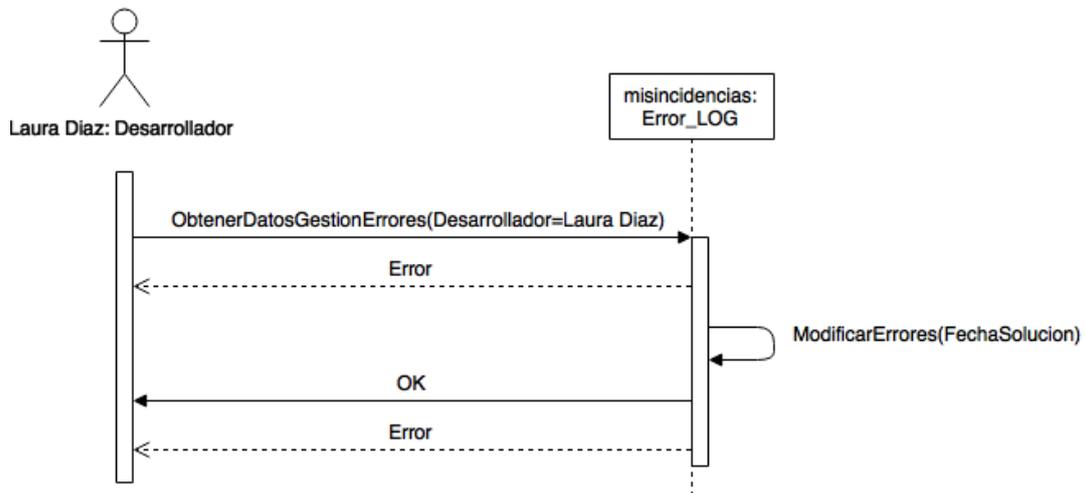


Ilustración 19: Diagrama de Secuencia – Cerrar incidencia

5.3.7 Diagrama de Secuencia: Generar informes

En este diagrama aparece un Desarrollador a la hora de generar informes con los datos de la consulta que se deseen.

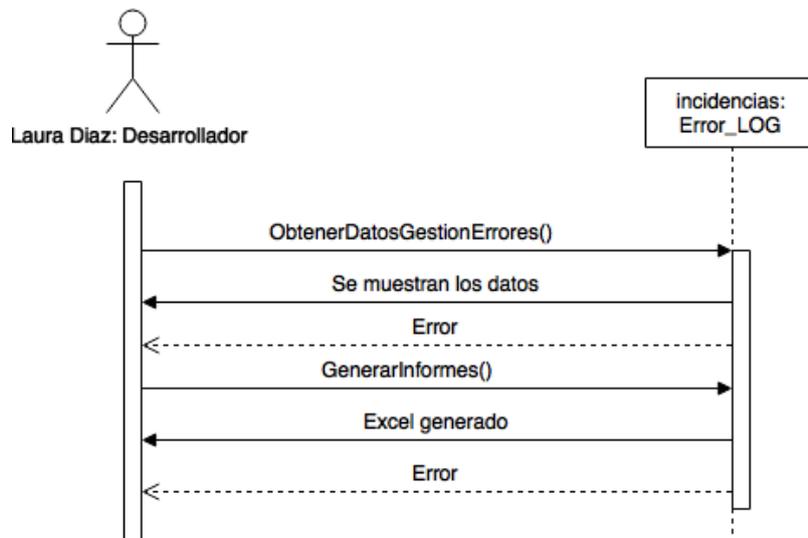


Ilustración 20: Diagrama de Secuencia – Generar informes

5.4 Diseño de la Base de Datos

Una base de datos correctamente diseñada permite obtener acceso a información exacta y actualizada. Un diseño correcto es esencial para lograr los objetivos fijados para la base de datos.

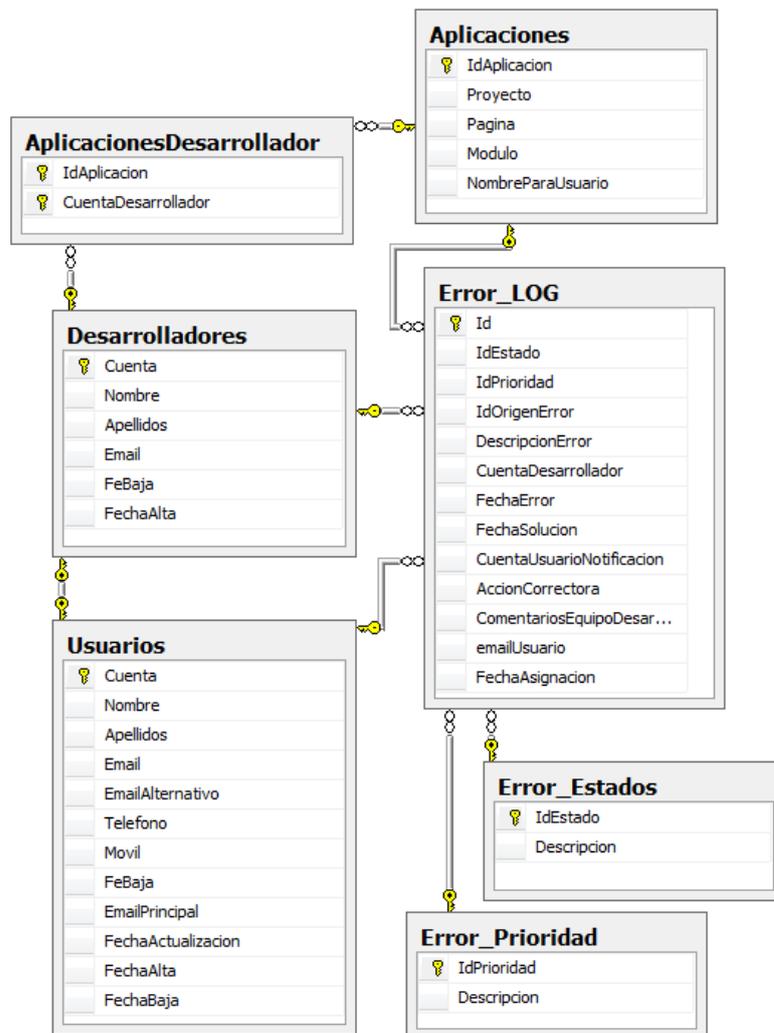


Ilustración 21: Diseño de base de datos

6 IMPLEMENTACIÓN

En este punto se explican las tablas y procedimientos de base de datos y los métodos del web services. También se explican las principales funcionalidades de las aplicaciones y qué relación tienen estas con todos estos elementos anteriores y cómo interactúan entre ellos.

6.1 Tablas de base de datos

A continuación se explica brevemente que información contiene cada una de las tablas y como se relacionan con las demás tablas.

6.1.1 Usuarios

La tabla *Usuarios* contiene información de los usuarios que se encuentran en el sistema.

Columnas:

- *Cuenta*(PK, *nvarchar*(100), No NULL): es un identificador único.
- *Nombre*(*nvarchar*(100), No NULL): nombre del usuario.
- *Apellidos*(*nvarchar*(200), No NULL): apellidos del usuario.
- *Email*(*nvarchar*(150), No NULL): email del usuario.
- *EmailAlternativo*(*nvarchar*(150), NULL): email alternativo del usuario.
- *Telefono*(*varchar*(15), No NULL): teléfono del usuario.
- *Movil*(*varchar*(15), No NULL): móvil del usuario.
- *FechaAlta*(*datetime*, No NULL): fecha de alta del usuario en el sistema.
- *FechaActualizacion*(*datetime*, NULL): última fecha de actualización de los datos del usuario en el sistema.
- *FeBaja*(*datetime*, NULL): fecha de baja del usuario en el sistema.

Claves:

- *PK_Usuarios*

6.1.2 Aplicaciones

La tabla *Aplicaciones* contiene información de las aplicaciones que se encuentran en el sistema.

Columnas:

- *IdAplicacion(PK, int, No NULL)*: es un identificador único.
- *Proyecto(nvarchar(100), No NULL)*: es el nombre del proyecto.
- *Pagina(nvarchar(100), No NULL)*: es cada una de las partes de las que consta el proyecto, por ejemplo la página *MainPage* o *ModificarDatosUsuario*.
- *Modulo(nvarchar(100), No NULL)*: es cada función o procedimiento de la página, por ejemplo *ModificarDatosUsuario_Loaded*.
- *NombreParaUsuario(nvarchar(100), No NULL)*: son las tres columnas anteriores juntas (*Proyecto*, *Pagina* y *Modulo*) pero poniendo una descripción más clara para que el usuario pueda entenderla a la hora de seleccionarlo para poner un error.

Claves:

- *PK_Aplicacion*

6.1.3 Desarrolladores

La tabla *Desarrolladores* contiene información de los desarrolladores que se encuentran en el sistema. Para poder insertar un *Desarrollador* tiene que estar insertado previamente en la tabla *Usuarios*.

Columnas:

- *Cuenta(PK, FK, nvarchar(100), No NULL)*: es un identificador único.
- *Nombre(nvarchar(100), No NULL)*: nombre del desarrollador.
- *Apellidos(nvarchar(200), No NULL)*: apellidos del desarrollador.
- *Email(nvarchar(150), No NULL)*: email del desarrollador.
- *FechaAlta(datetime, No NULL)*: fecha de alta del desarrollador en el sistema.
- *FeBaja(datetime, NULL)*: fecha de baja del desarrollador en el sistema.

Claves:

- *PK_Desarrolladores*
- *FK_Desarrolladores_Usuarios*

6.1.4 AplicacionesDesarrollador

La tabla *AplicacionesDesarrollador* contiene todas las aplicaciones de la tabla *Aplicaciones* junto con el desarrollador de la tabla *Desarrolladores* encargado de la aplicación. Cada aplicación puede aparecer varias veces ya que puede haber varios desarrolladores para cada aplicación.

Columnas:

- *IdAplicacion*(PK, FK, int, No NULL): es el identificador de la tabla *Aplicaciones*.
- *CuentaDesarrollador*(PK, FK, nvarchar(100), No NULL): es el identificador de la tabla *Desarrolladores*.

Claves:

- *PK_AplicacionesDesarrollador*
- *FK_AplicacionesDesarrollador_Aplicaciones*
- *FK_AplicacionesDesarrollador_Desarrolladores*

6.1.5 Error_Estados

La tabla *Error_Estados* contiene información de los estados que pueden tener los errores que se registren en base de datos.

Columnas:

- *IdEstado*(PK, int, No NULL): es un identificador único.
- *Descripcion*(nvarchar(100), No NULL): descripción del estado.

Claves:

- *PK_Error_Estados*

6.1.6 Error_Prioridad

La tabla *Error_Prioridad* contiene información de las prioridades que pueden tener los errores que se registren en base de datos.

Columnas:

- *IdPrioridad*(PK, int, No NULL): es un identificador único.
- *Descripcion*(nvarchar(100), No NULL): descripción de la prioridad.

Claves:

- *PK_Error_Prioridad*

6.1.7 Error_LOG

La tabla *Error_LOG* contiene información de los errores insertados en el sistema. Hay algunas columnas que tienen relación con otras tablas, como son el caso de *IdEstado* (Tabla *Error_Estados*), *IdPrioridad* (Tabla *Error_Prioridad*), *IdOrigenError* (Tabla *Aplicaciones*), *CuentaDesarrollador* (Tabla *Desarrolladores*), *CuentaUsuarioNotificacion* (Tabla *Usuarios*).

Columnas:

- *Id*(PK, int), No NULL): es un identificador único.
- *IdEstado*(FK, int, No NULL): id del estado que tiene el error.
- *IdPrioridad*(FK, int, No NULL): id de la prioridad que tiene el error.
- *IdOrigenError*(FK, int, No NULL): id de la aplicación de origen que tiene el error.
- *DescripcionError*(nvarchar(max), No NULL): descripción que introduce el usuario para explicar el error que se ha encontrado.
- *CuentaDesarrollador*(FK, nvarchar(100), NULL): cuenta del desarrollador que se está encargando de resolver la incidencia o que ya la ha solucionado. Si la incidencia aún no se ha asignado a nadie esta columna aparecerá vacía.
- *FechaError*(datetime, No NULL): fecha en la que tiene lugar el error.
- *FechaAsignacion*(datetime, NULL): fecha en la que se asigna la incidencia a un desarrollador.
- *FechaSolucion*(datetime, NULL): fecha en la que el desarrollador da por solucionada la incidencia.
- *CuentaUsuarioNotificacion*(FK, nvarchar (100), NULL): cuenta del usuario que ha

registrado la incidencia.

- *emailUsuario(nvarchar(100), NULL)*: email del usuario que ha registrado la incidencia para poder notificarle cuando se resuelva la incidencia.
- *AccionCorrectora(nvarchar(max), NULL)*: breve descripción que escribirá el desarrollador explicando lo que se ha realizado para resolver la incidencia.
- *ComentariosEquipoDesarrollo(nvarchar(max), NULL)*: comentarios adicionales que puede introducir el desarrollador si lo ve oportuno.

Claves:

- *PK_Error_LOG*
- *FK_Error_LOG_Aplicaciones*
- *FK_Error_LOG_Desarrolladores*
- *FK_Error_LOG_Error_Estados*
- *FK_Error_LOG_Error_Prioridad*
- *FK_Error_LOG_Usuarios*

6.2 Procedimientos de base de datos

A continuación se explican los procedimientos almacenados que se encuentran en la base de datos.

6.2.1 ObtenerDatosUsuarioxCuenta

Descripción: Procedimiento para obtener los datos de un registro de la tabla *Usuarios* pasando como parámetro *@pCuenta*.

Parámetros de entrada:

- *@pCuenta*: Cuenta del usuario conectado

Parámetros de salida: Devuelve los siguientes datos:

- *Cuenta*
- *Nombre*
- *Apellidos*
- *Email*
- *EmailAlternativo*
- *Telefono*
- *Movil*
- *FeBaja*

Tablas de base de datos:

- *Usuarios*

6.2.2 ModificarDatosUsuario

Descripción: Procedimiento para modificar algunos de los datos de la tabla *Usuarios*.

Parámetros de entrada:

- *@pCuenta*: Cuenta del usuario conectado.
- *@pEmailAlternativo*: Email alternativo del usuario conectado en caso de querer modificarlo o añadirlo.
- *@pTelefono*: Teléfono del usuario conectado en caso de querer modificarlo o añadirlo.
- *@pMovil*: Móvil del usuario conectado en caso de querer modificarlo o añadirlo.
- *@pFechaActualizacion*: Fecha en la que tiene lugar la modificación.

Parámetros de salida: Devuelve 0 en caso de ir bien y -1 en caso de ir mal.

Tablas de base de datos:

- *Usuarios*

6.2.3 ObtenerDatosDesarrolladorxCuenta

Descripción: Procedimiento para obtener los datos de un registro de la tabla *Desarrolladores* pasando como parámetro *@pCuenta*.

Parámetros de entrada:

- *@pCuenta*: Cuenta del desarrollador conectado.

Parámetros de salida: Devuelve los siguientes datos:

- *Cuenta*
- *Nombre*
- *Apellidos*
- *Email*
- *FeBaja*

Tablas de base de datos:

- *Desarrolladores*

6.2.4 ObtenerDatosAplicaciones

Descripción: Procedimiento para obtener los datos de la tabla *Aplicaciones*.

Parámetros de entrada: no tiene parámetros de entrada.

Parámetros de salida: Devuelve los siguientes datos:

- *IdOrigen*
- *OrigenError*
- *OrigenErrorParaUsuario*

Tablas de base de datos:

- *Aplicaciones*

6.2.5 ObtenerDatosAplicacionesxIdAplicacion

Descripción: Procedimiento para obtener los datos de un registro de la tabla *Aplicaciones* pasando como parámetro *@pIdAplicacion*.

Parámetros de entrada:

- *@pIdAplicacion*: Id de la aplicación de la que queremos obtener los datos.

Parámetros de salida: Devuelve los siguientes datos:

- *IdOrigen*
- *OrigenError*
- *OrigenErrorParaUsuario*

Tablas de base de datos:

- *Aplicaciones*

6.2.6 ObtenerDesarrolladoresxIdAplicacion

Descripción: Procedimiento para obtener los desarrolladores que están vinculados con una aplicación pasando como parámetro *@pIdAplicacion*.

Parámetros de entrada:

- *@pIdAplicacion*: Id de la aplicación de la que queremos obtener los datos.

Parámetros de salida: Devuelve los siguientes datos:

- *IdAplicacion*
- *CuentaDesarrollador*
- *EmailDesarrollador*
- *FeBajaDesarrollador*

Tablas de base de datos:

- *Desarrolladores*
- *AplicacionesDesarrollador*

6.2.7 Error_LOG_Insertar

Descripción: Procedimiento para insertar un registro de error en la tabla *Error_LOG*.

Parámetros de entrada:

- *@pIdEstado*: Id del estado en el que se encuentra el error que se inserta.
- *@pIdPrioridad*: Id de la prioridad del error que se inserta.
- *@pIdOrigenError*: Id del origen del error que se inserta
- *@pDescripcion*: Texto con el mensaje de error o una descripción que haya introducido el usuario para que sirva como aclaración a la hora de realizar el seguimiento del error.
- *@pFechaError*: Fecha en la que tiene lugar el error.
- *@pCuentaUsuarioNotificacion*: Cuenta del usuario al que le ha fallado la aplicación.
- *@pEmailUsuario*: Email del usuario al que le ha fallado la aplicación.

Parámetros de salida: Devuelve el número del registro insertado en caso de ir bien y -1 en caso de ir mal.

Tablas de base de datos:

- *Error_LOG*

6.2.8 Error_LOG_InsertarDesarrollador

Descripción: Procedimiento para insertar un registro de error en la tabla *Error_LOG* indicando el *Desarrollador*.

Parámetros de entrada:

- *@pldEstado*: Id del estado en el que se encuentra el error que se inserta.
- *@pldPrioridad*: Id de la prioridad del error que se inserta.
- *@pldOrigenError*: Id del origen del error que se inserta
- *@pDescripcion*: Texto con el mensaje de error o una descripción que haya introducido el usuario para que sirva como aclaración a la hora de realizar el seguimiento del error.
- *@pFechaError*: Fecha en la que tiene lugar el error.
- *@pFechaAsignacion*: Fecha en la que tiene lugar el error.
- *@pCuentaDesarrollador*: Cuenta del desarrollador al que se le asigna la incidencia.
- *@pCuentaUsuarioNotificacion*: Cuenta del usuario al que le ha fallado la aplicación.
- *@pEmailUsuario*: Email del usuario al que le ha fallado la aplicación.

Parámetros de salida: Devuelve el número del registro insertado en caso de ir bien y -1 en caso de ir mal.

Tablas de base de datos:

- *Error_LOG*



6.2.9 Error_LOG_Modificar

Descripción: Procedimiento para modificar algunos de los datos de un registro de la tabla *Error_LOG*.

Parámetros de entrada:

- *@pIdError*: Id del error que se quiere modificar.
- *@pCuentaDesarrollador*: Cuenta del desarrollador al que se le asigna el error.
- *@pIdEstado*: Id del estado al que se modifica el error.
- *@pFechaAsignacion*: Fecha en la que se asigna el error a un desarrollador.
- *@pFechaSolucion*: Fecha en la que el desarrollador da por solucionada la incidencia.
- *@pAccionCorrectora*: Descripción de la solución al error.
- *@pComentariosEquipoDesarrollo*: Comentarios acerca del error.

Parámetros de salida: Devuelve 0 en caso de ir bien y -1 en caso de ir mal.

Tablas de base de datos:

- *Error_LOG*

6.2.10 Error_LOG_ObtenerDatos

Descripción: Procedimiento para obtener los datos de la tabla *Error_LOG*.

Parámetros de entrada: No tiene parámetros de entrada.

Parámetros de salida: Devuelve los siguientes datos:

- *ReferenciaError*
- *IdPrioridad*
- *DescripcionPrioridad*
- *IdEstado*
- *DescripcionEstado*
- *IdOrigenError*
- *OrigenError*
- *OrigenErrorParaUsuario*
- *DescripcionError*
- *FechaError*
- *FechaAsignacion*
- *FechaSolucion*
- *CuentaDesarrollador*
- *Desarrollador*
- *EmailDesarrollador*
- *CuentaUsuarioNotificacion*
- *UsuarioNotificacion*
- *EmailUsuarioNotificacion*
- *AccionCorrectora*
- *ComentariosEquipoDesarrollo*

Tablas de base de datos:

- *Error_LOG*
- *Error_Prioridad*
- *Error_Estados*
- *Aplicaciones*
- *Desarrolladores*
- *Usuarios*

6.2.11 Error_LOG_ObtenerDatosxCuentaUsuarioNotificacion

Descripción: Procedimiento para obtener los datos de la tabla *Error_LOG* pasando el parámetro *@pCuentaUsuarioNotificacion*.

Parámetros de entrada:

- *@pCuentaUsuarioNotificacion*: Cuenta del usuario conectado.

Parámetros de salida: Devuelve los siguientes datos:

- *ReferenciaError*
- *IdPrioridad*
- *DescripcionPrioridad*
- *IdEstado*
- *DescripcionEstado*
- *IdOrigenError*
- *OrigenError*
- *DescripcionError*
- *FechaError*
- *FechaSolucion*
- *CuentaDesarrollador*
- *Desarrollador*
- *EmailDesarrollador*
- *CuentaUsuarioNotificacion*
- *UsuarioNotificacion*
- *EmailUsuarioNotificacion*
- *AccionCorrectora*
- *ComentariosEquipoDesarrollo*

Tablas de base de datos:

- *Error_LOG*
- *Error_Prioridad*
- *Error_Estados*
- *Aplicaciones*
- *Desarrolladores*
- *Usuarios*

6.3 Métodos de los Web Services

Uno de los usos principales de los Web Services es permitir la comunicación entre las empresas y sus clientes. Los Web Services permiten a las organizaciones intercambiar datos sin necesidad de conocer los detalles de sus respectivos sistemas de información.

En las aplicaciones desarrolladas en este PFC se han desarrollado dos Web Services, los cuales se han creado dentro de los proyectos:

- WS_NotificacionIncidencias: para crear métodos referentes al proyecto NotificacionIncidencias.
- WS_GestionErrores: para crear métodos referentes al proyecto GestionErrores.

A continuación se explican los métodos que se utilizan en estos Web Services.

6.3.1 ObtenerDatosUsuario (WS_NotificacionIncidencias)

```
public Usuario ObtenerDatosUsuario(string strCuenta)
```

Descripción: Método para obtener de base de datos los datos del usuario conectado.

Parámetros de entrada:

- *strCuenta*: Cuenta del usuario conectado

Parámetros de salida: Devuelve un objeto Usuario con los siguientes datos:

- *Nombre*
- *Apellidos*
- *Telefono*
- *Movil*
- *Email*
- *EmailAlternativo*

Procedimiento de base de datos: Usa el procedimiento *ObtenerDatosUsuarioxCuenta* y se le pasa el parámetro *@pCuenta* al cual se le asigna el valor de *strCuenta*.

6.3.2 ModificarDatosUsuario (WS_NotificacionIncidencias)

`public string ModificarDatosUsuario(string strCuenta, string strEmailAlternativo, string strTelefono, string strMovil)`

Descripción: Método para modificar en base de datos los datos del usuario conectado.

Parámetros de entrada:

- *strCuenta*: Cuenta del usuario conectado
- *strEmailAlternativo*: [Opcional] Email alternativo del usuario conectado en caso de querer modificarlo o añadirlo.
- *strTelefono*: [Opcional] Teléfono del usuario conectado en caso de querer modificarlo o añadirlo.
- *strMovil*: [Opcional] Móvil del usuario conectado en caso de querer modificarlo o añadirlo.

Parámetros de salida: Devuelve un string con "OK" en caso de ir bien. En caso de ir mal devuelve un string con la excepción.

Procedimiento de base de datos: Usa el procedimiento *ModificarDatosUsuario* y se le pasan los siguientes parámetros:

- *@pCuenta*: se le asigna el valor de *strCuenta*.
- *@pEmailAlternativo*: se le asigna el valor de *strEmailAlternativo*.
- *@pTelefono*: se le asigna el valor de *strTelefono*.
- *@pMovil*: se le asigna el valor de *strMovil*.
- *@pFechaActualizacion*: se le asigna el valor *DateTime.Now*.

6.3.3 InsertarErrores (WS_NotificacionIncidencias y WS_GestionErrores)

```
public string InsertarErrores(int idPrioridad, int idOrigen, string strDescripcion,  
string strUsuario, string strEmailUsuario = null)
```

Descripción: Método para insertar errores en base de datos.

Parámetros de entrada:

- *idPrioridad*: Id de la prioridad del error que se inserta.
- *idOrigen*: Id del origen del error que se inserta.
- *strDescripcion*: Texto con el mensaje de error o una descripción que haya introducido el usuario para que sirva como aclaración a la hora de realizar el seguimiento del error.
- *strUsuario*: Cuenta del usuario al que le ha fallado la aplicación.
- *strEmailUsuario*: Email del usuario al que le ha fallado la aplicación.

Parámetros de salida: Devuelve el número del error insertado. En caso de ir mal devuelve -1.

Procedimiento de base de datos: Usa el procedimiento *Error_LOG_Insertar* y se le pasan los siguientes parámetros:

- *@pIdEstado*: se le asigna 1 ya que de primeras se inserta la incidencia en estado "Sin asignar".
- *@pIdPrioridad*: se le asigna *idPrioridad*.
- *@pIdOrigenError*: se le asigna *idOrigen*.
- *@pDescripcion*: se le asigna *strDescripcion*.
- *@pFechaError*: se le asigna *DateTime.Now*.
- *@pCuentaUsuarioNotificacion*: se le asigna *strUsuario*.
- *@pEmailUsuario*: se le asigna *strEmailUsuario*.

6.3.4 InsertarErroresDesarrollador (WS_NotificacionIncidencias y WS_GestionErrores)

```
public string InsertarErroresDesarrollador(int idPrioridad, int idOrigen, string  
strDescripcion, string strCuentaDesarrollador, string strUsuario, string  
strEmailUsuario = null)
```

Descripción: Método para insertar errores en base de datos pasándole también el desarrollador.

Parámetros de entrada:

- *idPrioridad*: Id de la prioridad del error que se inserta.
- *idOrigen*: Id del origen del error que se inserta.
- *strDescripcion*: Texto con el mensaje de error o una descripción que haya introducido el usuario para que sirva como aclaración a la hora de realizar el seguimiento del error.
- *strCuentaDesarrollador*: Cuenta del desarrollador de la aplicación.
- *strUsuario*: Cuenta del usuario al que le ha fallado la aplicación.
- *strEmailUsuario*: Email del usuario al que le ha fallado la aplicación.

Parámetros de salida: Devuelve el número del error insertado. En caso de ir mal devuelve -1.

Procedimiento de base de datos: Usa el procedimiento

Error_LOG_InsertarDesarrollador y se le pasan los siguientes parámetros:

- *@pIdEstado*: se le asigna 2 ya que estamos asignando directamente la incidencia a un desarrollador, por lo que se le pone el estado "En desarrollo".
- *@pIdPrioridad*: se le asigna *idPrioridad*.
- *@pIdOrigenError*: se le asigna *idOrigen*.
- *@pDescripcion*: se le asigna *strDescripcion*.
- *@pFechaError*: se le asigna *DateTime.Now*.
- *@pFechaAsignacion*: se le asigna *DateTime.Now*.
- *@pCuentaDesarrollador*: se le asigna *strCuentaDesarrollador*.
- *@pCuentaUsuarioNotificacion*: se le asigna *strUsuario*.
- *@pEmailUsuario*: se le asigna *strEmailUsuario*.

6.3.5 ObtenerDatosIncidenciasUsuarioxCuentaUsuario (WS_NotificacionIncidencias)

`public List<DatosIncidenciasUsuario>`

`ObtenerDatosIncidenciasUsuarioxCuentaUsuario(string strCuenta)`

Descripción: Método para obtener los datos de las incidencias del usuario conectado que se encuentran registradas en base de datos.

Parámetros de entrada:

- *strCuenta*: Cuenta del usuario conectado.

Parámetros de salida: Devuelve una lista de objetos DatosIncidenciasUsuario con los siguientes datos:

- *ReferenciaError*
- *idEstado*
- *Estad*
- *idPrioridad*
- *Prioridad*
- *idOrigenError*
- *OrigenError*
- *FechaError_sinHora*
- *FechaSolucion_sinHora*
- *DescripcionError*
- *CuentaDesarrollador*
- *Desarrollador*
- *EmailDesarrollador*
- *UsuarioNotificacion*
- *CuentaUsuarioNotificacion*
- *EmailUsuarioNotificacion*
- *AccionCorrectora*
- *ComentariosEquipoDesarrollo*

Procedimiento de base de datos: Usa el procedimiento

Error_LOG_ObtenerDatosxCuentaUsuarioNotificacion y se le pasa el siguiente parámetro:

- *@pCuentaUsuarioNotificacion*: se le asigna strCuenta.

6.3.6 ObtenerOrigen (WS_NotificacionIncidencias)

```
public List<OrigenError> ObtenerOrigen()
```

Descripción: Método para obtener los datos de las aplicaciones que haya registradas en la base de datos.

Parámetros de entrada: No tiene.

Parámetros de salida: Devuelve una lista de objetos OrigenError con los siguientes datos:

- *idOrigen*
- *Origen*
- *OrigenParaUsuario*

Procedimiento de base de datos: Usa el procedimiento *ObtenerDatosAplicaciones* y no se le pasa ningún parámetro.

6.3.7 ObtenerOrigenxId (WS_NotificacionIncidencias y WS_GestionErrores)

```
public OrigenError ObtenerOrigenxId(int idOrigen)
```

Descripción: Método para obtener de base de datos los datos del idOrigen indicado.

Parámetros de entrada:

- *idOrigen*: Id del origen del que queremos obtener los datos.

Parámetros de salida: Devuelve un objeto OrigenError con los siguientes datos:

- *idOrigen*
- *Origen*
- *OrigenParaUsuario*

Procedimiento de base de datos: Usa el procedimiento

ObtenerDatosAplicacionesxIdAplicacion y se le pasa el siguiente parámetro:

- *@pldAplicacion*: se le asigna idOrigen.

6.3.8 ObtenerDesarrolladoresxIdAplicacion (WS_NotificacionIncidencias y WS_GestionErrores)

```
public List<DatosIncidenciasUsuario> ObtenerDesarrolladoresxIdAplicacion(int idAplicacion)
```

Descripción: Método para obtener de base de datos los desarrolladores del idAplicacion indicado.

Parámetros de entrada:

- *idAplicacion*: Id de la aplicación del que queremos obtener los datos.

Parámetros de salida: Devuelve una lista de objetos DatosIncidenciasUsuario con los siguientes datos:

- *idOrigen*
- *CuentaDesarrollador*
- *EmailDesarrollador*
- *FechaBajaDesarrollador*

Procedimiento de base de datos: Usa el procedimiento

ObtenerDesarrolladoresxIdAplicacion y se le pasa el siguiente parámetro:

- *@pIdAplicacion*: se le asigna idAplicacion.

6.3.9 EnvioCorreoUsuario (WS_NotificacionIncidencias)

```
public string EnvioCorreoUsuario(string strDestinatario, string strOrigen, string strDescripcion, int numReferencia)
```

Descripción: Método para realizar el envío de correos informativos sobre errores a los usuarios.

Parámetros de entrada:

- *strDestinatario*: Usuario al que se va a mandar el correo.
- *strOrigen*: Nombre de la aplicación donde ha tenido lugar el error.
- *numReferencia*: Número de la incidencia registrada.

Parámetros de salida: Devuelve un string con "OK" en caso de ir bien. En caso de ir mal devuelve un string con la excepción.

Procedimiento de base de datos: No usa procedimiento de base de datos. Usa la función MandarCorreo que se encuentra también en el web services.

6.3.10 EnvioCorreoDesarrollador (WS_NotificacionIncidencias y WS_GestionErrores)

```
public string EnvioCorreoDesarrollador(string strDestinatario, string strOrigen,  
string strDescripcion, int numReferencia, string strCuentaUsuarioConectado)
```

Descripción: Método para realizar el envío de correos informativos sobre errores a los desarrolladores.

Parámetros de entrada:

- *strDestinatario*: Desarrollador al que se va a mandar el correo.
- *strOrigen*: Nombre de la aplicación donde ha tenido lugar el error.
- *numReferencia*: Número de la incidencia registrada.
- *strCuentaUsuarioConectado*: Cuenta del usuario que se encuentra conectado y le ha sucedido el error.

Parámetros de salida: Devuelve un string con "OK" en caso de ir bien. En caso de ir mal devuelve un string con la excepción.

Procedimiento de base de datos: No usa procedimiento de base de datos. Usa la función MandarCorreo que se encuentra también en el web services.

6.3.11 EnvioCorreoSolucionUsuario (WS_GestionErrores)

```
public string EnvioCorreoSolucionUsuario(string strDestinatario, string strOrigen,  
string strDescripcion, string strComentarios, int numReferencia)
```

Descripción: Método para realizar el envío de correos informativos sobre errores a los usuarios.

Parámetros de entrada:

- *strDestinatario*: Usuario al que se va a mandar el correo.
- *strOrigen*: Nombre de la aplicación donde ha tenido lugar el error.
- *strComentarios*: Comentarios sobre la solución por parte del equipo de desarrollo.
- *numReferencia*: Número de la incidencia registrada.

Parámetros de salida: Devuelve un string con "OK" en caso de ir bien. En caso de ir mal devuelve un string con la excepción.

Procedimiento de base de datos: No usa procedimiento de base de datos. Usa la función MandarCorreo que se encuentra también en el web services.

6.3.12 MandarCorreo (WS_NotificacionIncidencias y WS_GestionErrores)

```
public string MandarCorreo(MailMessage mensaje)
```

Descripción: Función para realizar el envío de correos.

Parámetros de entrada:

- *mensaje*: Datos necesarios para el envío del correo como son el cuerpo del mensaje, destinatario, etc.

Parámetros de salida: Devuelve un string con "OK MandarCorreo" en caso de ir bien. En caso de ir mal devuelve un string con la excepción.

6.3.13 ObtenerDatosUsuario (WS_GestionErrores)

```
public Usuario ObtenerDatosUsuario(string strCuenta)
```

Descripción: Método para obtener de base de datos los datos del usuario conectado.

Parámetros de entrada:

- *strCuenta*: Cuenta del usuario conectado

Parámetros de salida: Devuelve un objeto Usuario con los siguientes datos:

- *Cuenta*
- *Nombre*
- *Apellidos*
- *Email*

Procedimiento de base de datos: Usa el procedimiento ObtenerDatosDesarrolladorxCuenta y se le pasa el parámetro *@pCuenta* al cual se le asigna el valor de *strCuenta*.

6.3.14 ObtenerDatosGestionErrores (WS_GestionErrores)

```
public List<DatosErrores> ObtenerDatosGestionErrores()
```

Descripción: Método para obtener de base de datos las incidencias que se encuentran registradas.

Parámetros de entrada: No tiene.

Parámetros de salida: Devuelve una lista de objetos DatosErrores con los siguientes datos:

- ReferenciaError
- idEstado
- Estado
- idPrioridad
- Prioridad
- idOrigenError
- OrigenError
- OrigenErrorParaUsuario
- FechaError_sinHora
- FechaAsignacion_sinHora
- FechaSolucion_sinHora
- imgSolucionar
- imgAsignar
- DescripcionError
- CuentaDesarrollador
- Desarrollador
- EmailDesarrollador
- CuentaUsuario
- UsuarioNotificacion
- EmailUsuarioNotificacion
- AccionCorrectora
- ComentariosEquipoDesarrollo

Procedimiento de base de datos: Usa el procedimiento *Error_LOG_ObtenerDatos* y no se le pasa ningún parámetro.

6.3.15 ModificarErrores (WS_GestionErrores)

```
public string ModificarErrores(int idError, string strDesarrollador, int idEstado,  
string strFechaAsignacion, string strFechaSolucion, string strAccionCorrectora,  
string strComentariosEquipoDesarrollo)
```

Descripción: Método para modificar en base de datos los datos de una incidencia.

Parámetros de entrada:

- *idError*: Número del error que se va a modificar.
- *strDesarrollador*: Cuenta del desarrollador encargado del error indicado.
- *idEstado*: Id del estado al que se va a cambiar el error.
- *strFechaAsignacion*: Fecha en la que se asigna un desarrollador a la incidencia.
- *strFechaSolucion*: Fecha en la que se da por solucionada la incidencia.
- *strAccionCorrectora*: Descripción de la solución al error.
- *strComentariosEquipoDesarrollo*: Comentarios acerca del error.

Parámetros de salida: Devuelve un string con "OK" en caso de ir bien. En caso de ir mal devuelve un string con la excepción.

Procedimiento de base de datos: Usa el procedimiento *Error_LOG_Modificar* y se le pasan los siguientes parámetros:

- *@pIdError*: se le asigna el valor de idError.
- *@pCuentaDesarrollador*: se le asigna el valor de strDesarrollador.
- *@pIdEstado*: se le asigna el valor de idEstado.
- *@pFechaAsignacion*: se le asigna el valor de strFechaAsignacion.
- *@pFechaSolucion*: se le asigna el valor strFechaSolucion.
- *@pAccionCorrectora*: se le asigna el valor strAccionCorrectora.
- *@pComentariosEquipoDesarrollo*: se le asigna el valor
strComentariosEquipoDesarrollo.

6.4 Funcionalidades

En este punto se explicarán las funcionalidades de las aplicaciones de las que consta este PFC, qué pasos hay que seguir y cómo actúa internamente el código.

6.4.1 Sistema de notificación de incidencias

El usuario puede insertar en cualquier momento una incidencia pulsando un botón que se encontrará en todas las páginas de la intranet.



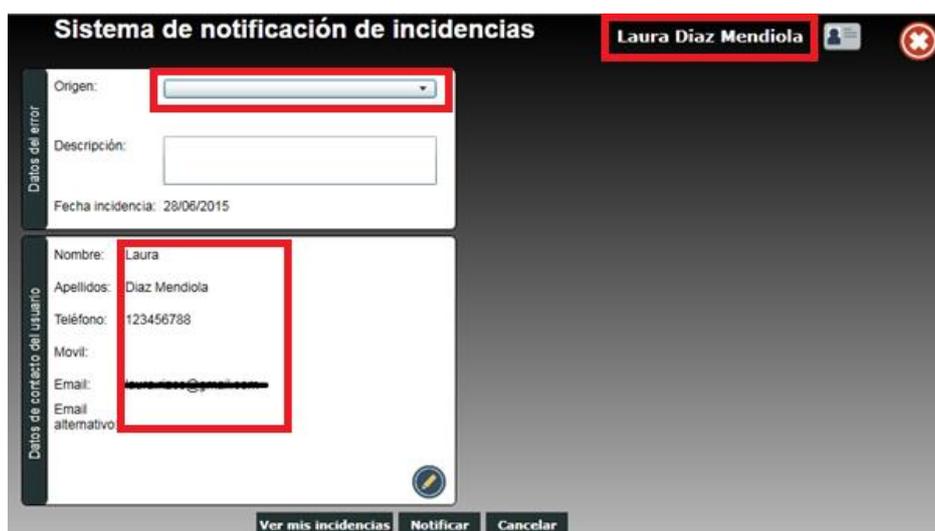
Sistema de notificación de incidencias

Ilustración 22: Botón para acceder al Sistema de notificación de incidencias

Al pulsar dicho botón se navegará al Sistema de notificación de incidencias.

Lo primero que hace el programa al entrar en esta aplicación es recoger la cuenta del usuario que se encuentra conectado en el equipo para obtener algunos datos de él como son el nombre, apellidos, email, teléfono. [[ObtenerDatosUsuario](#)]

Una vez que se han recuperado estos datos, el programa obtendrá todas las aplicaciones que estén registradas en base de datos para que el usuario pueda seleccionar donde ha tenido lugar el error. [[ObtenerOrigen](#)]



La imagen muestra la interfaz de usuario del 'Sistema de notificación de incidencias'. En la parte superior derecha, se muestra el nombre de usuario 'Laura Diaz Mendiola' y un icono de perfil. El formulario principal está dividido en dos secciones: 'Datos del error' y 'Datos de contacto del usuario'. En la sección 'Datos del error', hay un campo 'Origen' con un menú desplegable, un campo 'Descripción' y la fecha 'Fecha incidencia: 28/06/2015'. En la sección 'Datos de contacto del usuario', se muestran los datos: 'Nombre: Laura', 'Apellidos: Diaz Mendiola', 'Teléfono: 123456788', 'Movil:', 'Email:' y 'Email alternativo'. En la parte inferior del formulario, hay tres botones: 'Ver mis incidencias', 'Notificar' y 'Cancelar'.

Ilustración 23: Pantalla del Sistema de notificación de incidencias

A continuación se explicarán las funciones que se pueden realizar desde el Sistema de notificación de incidencias.

6.4.1.1 Insertar incidencia a través de la aplicación

Cuando el usuario pulse sobre el botón “Notifica” se comprobará que se haya seleccionado una aplicación de origen del error y que se haya escrito una pequeña descripción explicando lo sucedido.

Datos del error

Origen: Sistema de notificación de incidencias; M

Descripción: No puedo modificar los datos de contacto

Fecha incidencia: 06/07/2015

Datos de contacto del usuario

Nombre: Laura

Apellidos: Diaz Mendiola

Teléfono: 123456788

Movil:

Email: lauradiaz06@gmail.com

Email alternativo: laura@ccc.es

Notificar a este email

Ver mis incidencias Notificar Cancelar

Ilustración 24: Notificar incidencia

El usuario además puede elegir a que correo quiere que se le notifique todo lo relacionado con la incidencia introducida (en caso de haber introducido dos correos).

Si está todo correctamente se procederá a registrar la incidencia en base de datos.



Para ello lo primero que se comprobará es el número de desarrolladores que hay encargados para la aplicación seleccionada. [[ObtenerDesarrolladoresxIdAplicacion](#)]
Si hay varios desarrolladores encargados de la aplicación se insertará el error sin especificar quien se encargará de resolver la incidencia. [[InsertarErrores](#)]

Por el contrario si sólo hay uno encargado se insertará en base de datos indicando ya quien será el responsable. [[InsertarErroresDesarrollador](#)]

Una vez insertado el error en base de datos, ya que sólo disponemos del id de la aplicación, se procederá a obtener un nombre más descriptivo a partir del id [[ObtenerOrigenxId](#)] para tener esta información cuando se vaya a enviar el correo de notificación.

Por último se enviará un correo al usuario [[EnvioCorreoUsuario](#)] para que tenga el número de referencia de la incidencia y otro al desarrollador/es [[EnvioCorreoDesarrollador](#)] encargados de esa aplicación.

Cuando el usuario inserta la incidencia se le da una prioridad 1 (Crítica) porque se considera que si el usuario está registrando una incidencia será porque el fallo impide realizar correctamente su trabajo y así saber que hay que solucionarla lo antes posible.

6.4.1.2 Consultar incidencias registradas por el usuario conectado (a través del Sistema de Notificación de incidencias)

Pulsando sobre el botón “Ver mis incidencias” el usuario puede consultar su historial de incidencias [[ObtenerDatosIncidenciasUsuarioxCuentaUsuario](#)]. En esta consulta podrá ver algunos datos acerca de sus incidencias registradas.



Ilustración 25: Botón para ver incidencias del usuario

| Histórico de mis incidencias | | | | | | |
|--|---------------|---|-------------|----------------|---------------------------|-----------------------------|
| Arrastre una columna aquí para agrupar los datos por ella. | | | | | | |
| Num. referencia | Estado | Origen | Fecha error | Fecha solución | Desarrollador | Email desarrollador |
| 332 | En desarrollo | Sistema de notificación de incidencias; Modificar datos del usuario | 28/06/2015 | | Laura Diaz Mendiola | laura.diaz.mendiola@uc3m.es |
| Descripción error: No puedo modificar el correo | | | | | | |
| 414 | En desarrollo | Sistema de notificación de incidencias; Modificar datos del usuario | 28/06/2015 | | Laura Diaz Mendiola | laura.diaz.mendiola@uc3m.es |
| 448 | En desarrollo | Sistema de notificación de incidencias; Modificar datos del usuario | 28/06/2015 | | Laura Diaz Mendiola | laura.diaz.mendiola@uc3m.es |
| 476 | En desarrollo | Sistema de notificación de incidencias; Modificar datos del usuario | 29/06/2015 | | Laura Diaz Mendiola | laura.diaz.mendiola@uc3m.es |
| 44 | Solucionada | Sistema de gestión de incidencias; Consulta sin asignar | 30/05/2015 | 31/05/2015 | Laura Diaz Mendiola | laura.diaz.mendiola@uc3m.es |
| 45 | Solucionada | Sistema de notificación de incidencias; Ventana principal | 30/05/2015 | 31/05/2015 | desarrollador2 apellidos2 | desarrollador2@uc3m.es |
| 477 | Solucionada | Sistema de notificación de incidencias; Modificar datos del usuario | 29/06/2015 | 29/06/2015 | Laura Diaz Mendiola | laura.diaz.mendiola@uc3m.es |
| 478 | Solucionada | Sistema de gestión de incidencias; Consulta sin asignar | 05/07/2015 | 05/07/2015 | Laura Diaz Mendiola | laura.diaz.mendiola@uc3m.es |
| Total: 8 | | | | | | |

Ilustración 26: Consulta Histórico de incidencias del usuario

6.4.1.3 Modificar datos de contacto

Pulsando el botón inferior de “Modificar datos de contacto del usuario” el usuario podrá añadir o modificar alguno de sus datos de contacto.

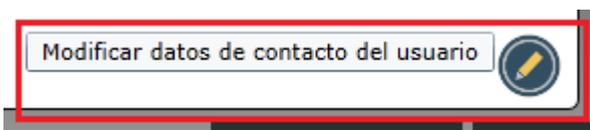


Ilustración 27: Botón para modificar datos de contacto del usuario

Se abrirá otra ventana en la que se cargarán algunos datos del usuario. [[ObtenerDatosUsuario](#)]

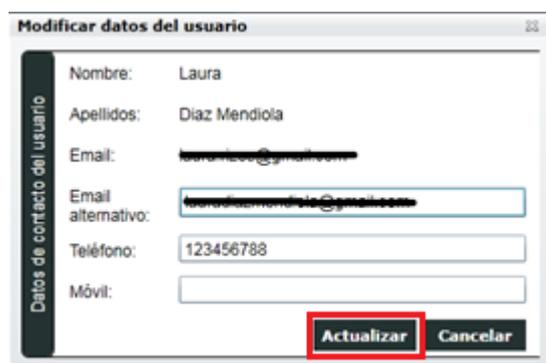
A screenshot of a web application window titled "Modificar datos del usuario". The window has a vertical sidebar on the left with the text "Datos de contacto del usuario". The main area contains a form with the following fields: "Nombre:" with the value "Laura"; "Apellidos:" with the value "Diaz Mendiola"; "Email:" with a redacted value; "Email alternativo:" with a text input field containing "laura.diazmendiola@gmail.com"; "Teléfono:" with the value "123456788"; and "Móvil:" with an empty text input field. At the bottom right of the form are two buttons: "Actualizar" (highlighted with a red border) and "Cancelar".

Ilustración 28: Pantalla de modificación de datos del usuario

El email principal no lo podrá modificar pero puede añadir otro para que los correos referentes a la incidencia se manden a este otro si lo prefiere.

Una vez haya modificado la información y pulse el botón “Actualizar” se modificarán los datos en base de datos. [[ModificarDatosUsuario](#)]

6.4.2 Sistema de gestión de incidencias

En el Sistema de gestión de incidencias podrá entrar cualquier usuario que tenga permisos para ello, por lo general los usuarios que puedan entrar serán los que estén dados de alta como desarrolladores.

Lo primero que hace el programa al entrar en esta aplicación es recoger la cuenta del usuario que se encuentra conectado en el equipo para obtener algunos datos de él como son el nombre, apellidos, email. [[ObtenerDatosUsuario](#)]

Una vez que se han recuperado estos datos el programa obtendrá todos los errores que estén registrados en base de datos [[ObtenerDatosGestionErrores](#)] y los mostrará en una consulta.

A continuación se explicarán las funciones que se pueden realizar desde el Sistema de gestión de incidencias

6.4.2.1 Consultar incidencias (a través del Sistema de Gestión de incidencias)

Al entrar en el Sistema de gestión de incidencias se obtendrán todos los errores que estén registrados en base de datos [[ObtenerDatosGestionErrores](#)] y los mostrará en una consulta. Esta consulta se puede configurar de las siguientes maneras:

➤ *En desarrollo por el usuario conectado*

Para conseguir que en la consulta se muestren las incidencias en desarrollo por el usuario conectado se tendrá que pulsar sobre el icono “Mostrar incidencias del desarrollador conectado” y no se seleccionará el check de “Incidencias solucionadas”. La lista de datos obtenida al principio se filtrará de manera que muestre sólo los datos en los que coincida la cuenta del usuario conectado con la cuenta del desarrollador encargado de las incidencias y no tengan fecha de solución.

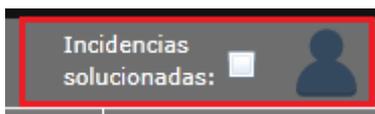


Ilustración 29: Configuración de botones para ver la consulta de incidencias en desarrollo por el usuario conectado

➤ *Todas las que se encuentren sin asignar o en desarrollo*

Para conseguir que en la consulta se muestren todas las incidencias en desarrollo o sin asignar se tendrá que pulsar sobre el icono “Mostrar incidencias de todos los desarrolladores” y no se seleccionará el check de “Incidencias solucionadas”. La lista de datos obtenida al principio se filtrará de manera que muestre sólo los datos que no tengan fecha de solución.



Ilustración 30: Configuración de botones para ver la consulta de todas las incidencias sin asignar o en desarrollo

➤ *Solucionadas por el usuario conectado*

Para conseguir que en la consulta se muestren las incidencias solucionadas por el usuario conectado se tendrá que pulsar sobre el icono “Mostrar incidencias del desarrollador conectado” y se seleccionará el check de “Incidencias solucionadas”. La lista de datos obtenida al principio se filtrará de manera que muestre sólo los datos en los que coincida la cuenta del usuario conectado con la cuenta del desarrollador encargado de las incidencias y tengan fecha de solución.

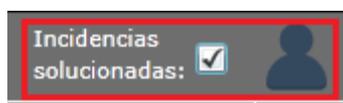


Ilustración 31: Configuración de botones para ver la consulta de incidencias solucionadas por el usuario conectado

➤ *Todas las solucionadas*

Para conseguir que en la consulta se muestren todas las incidencias solucionadas se tendrá que pulsar sobre el icono “Mostrar incidencias de todos los desarrolladores” y se seleccionará el check de “Incidencias solucionadas”. La lista de datos obtenida al principio se filtrará de manera que muestre sólo los datos que tengan fecha de solución.

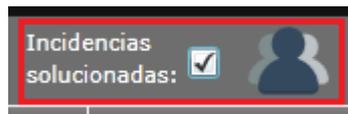


Ilustración 32: Configuración de botones para ver la consulta de todas las incidencias solucionadas

6.4.2.2 *Asignarse incidencia*

Al entrar en la consulta de incidencias [[ObtenerDatosGestionErrores](#)] se creará otra lista en la que se filtrarán los datos obtenidos de manera que sólo se tenga las incidencias sin desarrollador asignado y a la vez alguno de los encargados de la aplicación sea el usuario conectado. Este número de incidencias sin asignar aparecerá sobre el botón “Incidencias sin asignar”.



Ilustración 33: Botón que indica las incidencias que tiene el usuario sin asignar y pulsándolo se irá a dicha consulta

Una vez se pulse el botón “Incidencias sin asignar” llevará a otra ventana en la que aparecerá la lista filtrada explicada anteriormente y el usuario conectado podrá asignarse la incidencia pulsando sobre el botón “Asignarme incidencia”. Si confirma la acción se modificará el dato indicado en base de datos indicando quien se encargará de la incidencia, la fecha de asignación (es decir la fecha de ese momento) y se pondrá en estado 2 (En desarrollo). [[ModificarErrores](#)]

| Prioridad | Num. referencia | Origen | Fecha error | |
|-----------|-----------------|--|-------------|--|
| Baja | 475 | NotificacionIncidencias, MainPage.xaml.cs, MainPage_Loaded | 29/06/2015 | |

Descripción error:
ERROR: Error al recuperar rutaWSNotificacionIncidencias

Ilustración 34: Consulta incidencias sin asignar

6.4.2.3 Cerrar incidencia

Cuando un desarrollador termine de hacer las modificaciones necesarias y considere que ya está resuelta la incidencia procederá a cerrar la incidencia a través del Sistema de gestión de incidencias. Para ello pulsará en el botón “Cerrar incidencia” que aparecerá en la consulta, se abrirá otra ventana con datos del usuario conectado [[ObtenerDatosUsuario](#)] y datos de la incidencia seleccionada.

El desarrollador tendrá que rellenar algunos datos y al dar a “Guardar” se modificará ese registro en base de datos indicando que el estado de la incidencia pasa a ser 3 (Solucionada) e indicando la fecha de la solución entre otros datos. [[ModificarErrores](#)]

Cuando se haya modificado en base de datos se enviará al usuario que introdujo la incidencia un correo notificando que ya se ha solucionado el error. [[EnvioCorreoSolucionUsuario](#)]

| Origen | Fecha error | Desarrollador |
|--|-------------|---------------------|
| NotificacionIncidencias, ModificarDatosUsuario.xaml.cs, ModificarDatosUsuario_Loaded | 28/06/2015 | Laura Diaz Mendiola |

Ilustración 35: Botón para cerrar incidencia

6.4.2.4 Refrescar consulta

En cualquier momento el usuario podrá dar al botón “Refrescar” para actualizar la lista de incidencias, por si se han registrado nuevas incidencias o se han actualizado. [[ObtenerDatosGestionErrores](#)]



Ilustración 36: Botón para refrescar la consulta de incidencias

6.4.2.5 Filtrar datos

Además de todas las combinaciones explicadas anteriormente que se pueden hacer en la consulta, se puede filtrar escribiendo en la caja de texto de “Búsqueda rápida” o eligiendo algunas de las opciones de los combos. Se pueden filtrar todos los combos que se quieran y combinar tantas opciones como se deseen.

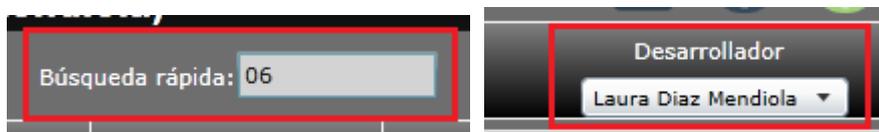


Ilustración 37: Tipos de filtros

6.4.2.6 Agrupar datos

El usuario podrá seleccionar cualquiera de las columnas y arrastrarla a la parte de arriba de la consulta. De esta manera los datos aparecerán agrupados por la columna seleccionada. Se pueden agrupar todas las columnas que se deseen y además es combinable con el resto de los filtros.

| Prioridad | Estado | Num. referencia | Fecha error | Desarrollador |
|-----------|---------------|-----------------|-------------|---------------------|
| Crítica | En desarrollo | 332 | 28/06/2015 | Laura Diaz Mendiola |
| Crítica | En desarrollo | 414 | 28/06/2015 | Laura Diaz Mendiola |
| Crítica | En desarrollo | 448 | 28/06/2015 | Laura Diaz Mendiola |
| Baja | En desarrollo | 437 | 28/06/2015 | Laura Diaz Mendiola |

Ilustración 38: Consulta incidencias con un dato agrupado

| Prioridad | Estado | Num. referencia | Fecha error | Desarrollador |
|-----------|---------------|-----------------|-------------|---------------------|
| Baja | En desarrollo | 437 | 28/06/2015 | Laura Diaz Mendiola |

Ilustración 39: Consulta incidencias con varios datos agrupados

6.4.2.7 Exportar datos

El usuario podrá exportar a un Excel los datos de cualquier consulta. Además podrá elegir entre exportar todos los datos o sólo los que se encuentren visibles en ese momento.

Al poder configurar la consulta estos informes pueden llegar a ser de gran utilidad. Por ejemplo se podría consultar cual es el desarrollador con mayor número de incidencias resueltas.

Al exportar los datos como se tienen las fechas de alta de la incidencia, cierre de la incidencia y la fecha de asignación a un desarrollador de la misma se podrían hacer informes con estadísticas como por ejemplo cual es el tiempo medio en resolver una incidencia o cual es el desarrollador que las resuelve más rápido.



Ilustración 40: Botón para exportar datos de la consulta

6.4.3 Insertar incidencia automáticamente por fallo interno de las aplicaciones

Cuando el código falla internamente, ya sea por algún caso que no se ha contemplado correctamente, fallo de datos, error de conexión, etc., se registran los errores y se insertan en base de datos de manera similar a cuando se insertan a través de la aplicación de notificación de incidencias.

Primero se comprobará el número de desarrolladores que hay encargados para la aplicación que ha fallado. [[ObtenerDesarrolladoresxIdAplicacion](#)]
Si hay varios desarrolladores encargados de la aplicación se insertará el error sin especificar quien se encargará de resolver la incidencia. [[InsertarErrores](#)]

Por el contrario si sólo hay uno encargado se insertará en base de datos indicando ya quien será el responsable. [[InsertarErroresDesarrollador](#)]

Una vez insertado el error en base de datos, ya que sólo disponemos del id de la aplicación, se procederá a obtener un nombre más descriptivo a partir del id [[ObtenerOrigenxId](#)] para tener esta información cuando se vaya a enviar el correo de notificación.

Por último se enviará un correo al desarrollador/es [[EnvioCorreoDesarrollador](#)] encargados de esa aplicación. En este caso no se enviará al usuario. Si el usuario ve que es un error que le influye a la hora de realizar su trabajo será el quien decida si quiere introducir la incidencia por el sistema.

En este caso a la incidencia se le da una prioridad 3 (Baja) porque puede que sean errores sin mayor importancia. Si el usuario lo ve oportuno que la registre a través del sistema.

7 EVALUACIÓN Y PRUEBAS

Como la parte del Sistema de gestión de incidencias va a ser una aplicación para uso exclusivo del equipo de desarrollo se ha comenzado con la implantación de esta parte.

Para ello es necesario modificar todos los proyectos existentes para modificar la función RegistrarError() que aunque ya existía de antes, ahora se le pasan otros parámetros y añadir nuevos métodos a los web services de cada proyecto. Este cambio se está realizando poco a poco y aun no se están registrando todos los errores generados por las excepciones correctamente. Los proyectos que ya se han modificado si funcionan bien e incluso el envío de correos. Lo que también está funcionando a pleno rendimiento es la consulta de las incidencias registradas en el sistema.

En cuanto al Sistema de notificación de incidencias también se ha desplegado la aplicación pero falta por añadir el botón que lleva a este sistema en todas las aplicaciones.

Para poder realizar algunas pruebas lo que se realizó fue añadir este botón a una de las páginas principales primero en el entorno de desarrollo, para que el equipo de desarrollo fuera el encargado de realizar estas pruebas. Cuando se comprobó que era estable se fue añadiendo el botón al resto de entornos, pero en principio sólo en la página principal.

Para finales de julio/primer quincena de agosto se tiene previsto haber añadido el botón que lleva al Sistema de notificación de incidencias en todas las aplicaciones para que cualquier usuario pueda insertar las incidencias desde cualquier sitio y también terminar de hacer las modificaciones necesarias para que todas las aplicaciones registren sus errores generados por las excepciones.

7.1 Problemas encontrados

7.1.1 Envío de correos.

Al principio se planteó mandar los correos a través de la base de datos. Cuando la aplicación tuviera que mandar un correo llamaría a un procedimiento de base de datos pasándole algunos parámetros con la información necesaria como el email del destinatario, descripción del error, etc. Ya se había implementado el procedimiento y a la hora de probarlo no se enviaban los correos. Tras buscar información se pudo comprobar que el problema era por estar usando SQL Server 2008 R2 Express en lugar de SQL Server 2008 R2 por lo que se optó por buscar alternativas. [\[4\]](#)[\[5\]](#)[\[6\]](#)

La solución fue enviar los correos directamente desde .NET pero también hubo algunos problemas a la hora de enviarlos. Por un lado hubo que desactivar el antivirus y el firewall como ponía en algunos foros y deshabilitar la verificación de dos pasos de la cuenta que se creó para que los correos se enviaran a través de esta. [\[7\]](#) a [\[17\]](#)

Finalmente todos estos obstáculos encontrados en cuanto al envío de correos fueron solventados.

A continuación se muestra un ejemplo de un correo enviado a través del Sistema de notificación de incidencias.

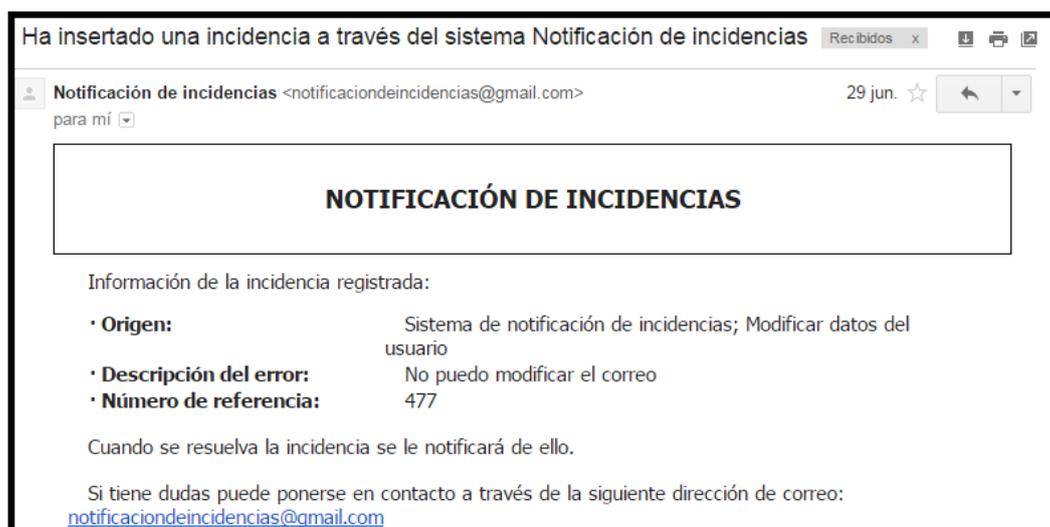


Ilustración 41: Problema solucionado al enviar correos

7.1.2 Instalación y configuración del IIS [\[Anexo C\]](#)

Surgieron muchos problemas en cuanto a la instalación y configuración del IIS al no tener muchos conocimientos previos sobre este servidor web y al tener que instalarlo y configurarlo desde cero en el equipo donde se ha realizado este PFC. Poco a poco se fueron solventando y la prueba está en que las aplicaciones desarrolladas en el PFC se pueden ejecutar desde un servidor local.

A continuación se muestran las aplicaciones desplegadas en el IIS y si a una de ellas le damos a Examinar se puede ver cómo se está ejecutando la aplicación en un servidor local.

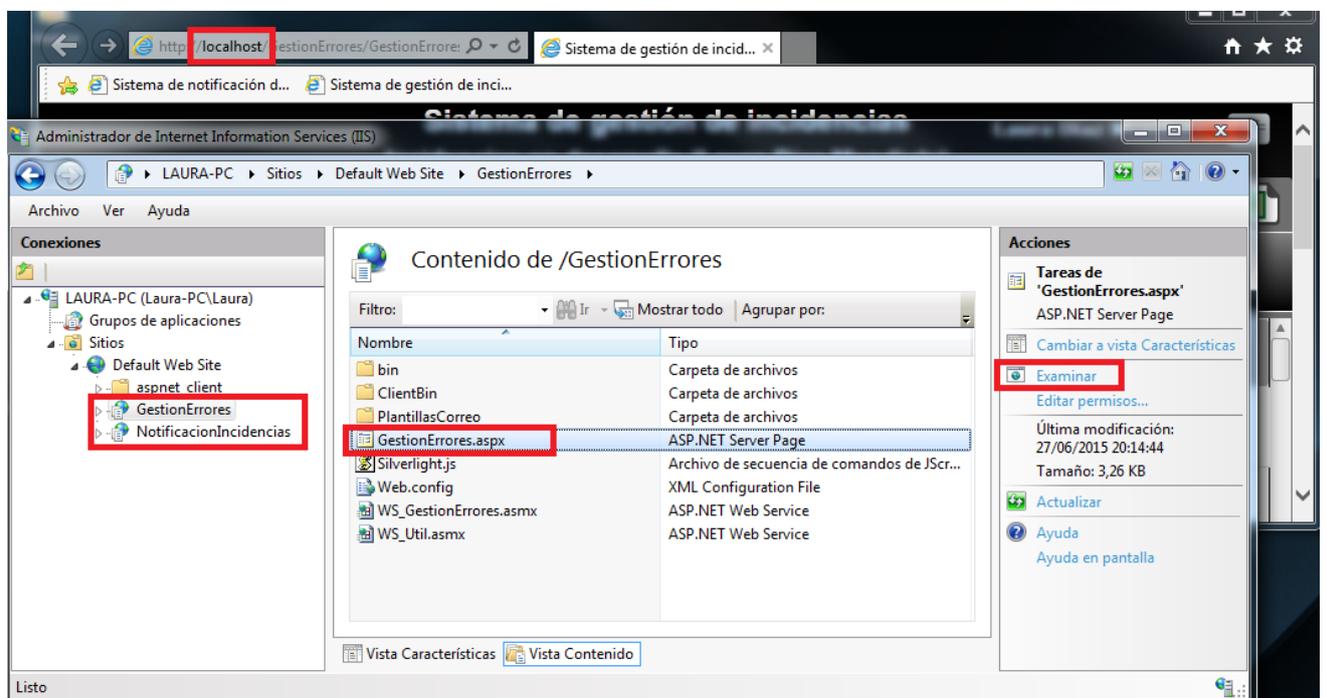


Ilustración 42: Problema solucionado al instalar y configurar el IIS

7.2 Pruebas

A continuación se muestra el resultado de las pruebas realizadas con la aplicación que validarán los requisitos determinados en el apartado de [Requisitos](#).

Se usará la siguiente plantilla para mostrar las pruebas realizadas:

Tabla 32: Esquema general de las pruebas

| | |
|--|---------------------|
| Código | |
| Nombre | |
| Identificador de requerimientos | |
| Escenario | |
| Datos de entrada | |
| Resultados | |
| Resultados esperados | |
| Resultados obtenidos | |
| Resultado | Correcto/Incorrecto |

A continuación se explican los campos de la plantilla:

- **Código:** Es un identificador único para cada caso de prueba.
- **Nombre:** Es el nombre del caso de prueba. Debe ser un título entendible para la fácil comprensión del propósito del caso de prueba y su campo de aplicación.
- **Identificador de requerimientos:** Es el identificador al que hace referencia el caso de prueba. Puede referirse a requisitos o casos de uso.
- **Escenario:** Es una breve explicación de donde tenemos que encontrarnos (en que página, si es necesario que sea a alguna hora en concreto, etc..).
- **Datos de entrada:** Es una serie de datos que habrá que introducir para comprobar si el resultado obtenido es el que se esperaba.
- **Resultados esperados:** Son los resultados que se deberían obtener tras introducir los datos que se indican en Datos de entrada.
- **Resultados obtenidos:** Son los resultados que se obtienen tras haber introducido los Datos de entrada.
- **Resultado:** Podrá ser **Correcto** o **Incorrecto**. Si los resultados esperados y los obtenidos coinciden se pondrá que ha sido Correcto. En caso contrario se pondrá Incorrecto.

Tabla 33: Casos de prueba – CP-RF01: Usuarios registrados en base de datos

| | |
|--|--|
| Código | CP-RF01 |
| Nombre | Usuarios registrados en base de datos |
| Identificador de requerimientos | RF01 |
| Escenario | Acceder a la intranet |
| Datos de entrada | - Introducir usuario y contraseña al conectarse en el equipo de trabajo. - Entrar en la intranet. |
| Resultados | |
| Resultados esperados | El usuario puede entrar correctamente en la intranet |
| Resultados obtenidos | El usuario entra correctamente en la intranet |
| Resultado | Correcto |

Tabla 34: Casos de prueba – CP-RF02: Desarrolladores registrados en base de datos

| | |
|--|--|
| Código | CP-RF02 |
| Nombre | Desarrolladores registrados en base de datos |
| Identificador de requerimientos | RF02 |
| Escenario | Acceder a la intranet |
| Datos de entrada | - Introducir usuario y contraseña al conectarse en el equipo de trabajo. - Entrar en la intranet. |
| Resultados | |
| Resultados esperados | El desarrollador puede entrar correctamente en la intranet |
| Resultados obtenidos | El desarrollador entra correctamente en la intranet |
| Resultado | Correcto |

Tabla 35: Casos de prueba – CP-RF03: Acceder al Sistema de notificación de incidencias

| | |
|--|--|
| Código | CP-RF03 |
| Nombre | Acceder al Sistema de notificación de incidencias |
| Identificador de requerimientos | RF03 |
| Escenario | Cualquier página de la intranet que contenga el botón para acceder al Sistema de notificación de incidencias |
| Datos de entrada | - Pulsar el botón de Sistema de notificación de incidencias |
| Resultados | |
| Resultados esperados | El usuario puede entrar en el Sistema de notificación de incidencias |
| Resultados obtenidos | El usuario entra correctamente en el Sistema de notificación de incidencias |
| Resultado | Correcto |

Tabla 36: Casos de prueba – CP-RF04: Acceder al Sistema de gestión de incidencias

| | |
|--|---|
| Código | CP-RF04 |
| Nombre | Acceder al Sistema de gestión de incidencias |
| Identificador de requerimientos | RF04 |
| Escenario | Intranet |
| Datos de entrada | - Pulsar el botón de Sistema de gestión de incidencias que se encontrará en la intranet |
| Resultados | |
| Resultados esperados | El usuario puede entrar en el Sistema de gestión de incidencias |
| Resultados obtenidos | El usuario entra correctamente en el Sistema de gestión de incidencias |
| Resultado | Correcto |

Tabla 37: Casos de prueba – CP-RF05: Modificar datos de contacto del usuario

| | |
|--|--|
| Código | CP-RF05 |
| Nombre | Modificar datos de contacto del usuario |
| Identificador de requerimientos | RF05 |
| Escenario | Sistema de notificación de incidencias |
| Datos de entrada | - Pulsar el botón “Modificar datos de contacto del usuario” - Email alternativo = “ email@example.es ” - Teléfono = “123456789” - Móvil = “987654321” - Pulsar el botón “Actualizar” |
| Resultados | |
| Resultados esperados | El usuario puede modificar sus datos de contacto |
| Resultados obtenidos | El usuario modifica correctamente sus datos de contacto |
| Resultado | Correcto |

Tabla 38: Casos de prueba – CP-RF06: Consultar incidencias anteriores

| | |
|--|--|
| Código | CP-RF06 |
| Nombre | Consultar incidencias anteriores |
| Identificador de requerimientos | RF06 |
| Escenario | Sistema de notificación de incidencias |
| Datos de entrada | - Pulsar el botón “Ver mis incidencias” |
| Resultados | |
| Resultados esperados | El usuario puede consultar su histórico de incidencias registradas |
| Resultados obtenidos | El usuario visualiza correctamente su histórico de incidencias registradas |
| Resultado | Correcto |

Tabla 39: Casos de prueba – CP-RF07: El usuario da de alta una incidencia

| | |
|--|---|
| Código | CP-RF07 |
| Nombre | El usuario da de alta una incidencia |
| Identificador de requerimientos | RF07 |
| Escenario | Sistema de notificación de incidencias |
| Datos de entrada | <ul style="list-style-type: none">- Origen del error = “Sistema de notificación de incidencias; Modificar datos del usuario”- Descripción = “Al modificar los datos estos no se guardan”- Marcar el check de Email alternativo- Pulsar el botón “Notificar” |
| Resultados | |
| Resultados esperados | <ul style="list-style-type: none">- El usuario puede dar de alta una incidencia a través del Sistema de notificación de incidencias- El usuario recibe un correo con el número de la incidencia registrada- Los desarrolladores responsables de esa parte de la aplicación reciben un correo para que sepan que se ha insertado una incidencia |
| Resultados obtenidos | <ul style="list-style-type: none">- El usuario da de alta una incidencia correctamente a través del Sistema de notificación de incidencias- El usuario recibe un correo correctamente con el número de la incidencia registrada- Los desarrolladores responsables de esa parte de la aplicación reciben un correo correctamente para que sepan que se ha insertado una incidencia |
| Resultado | Correcto |

Tabla 40: Casos de prueba – CP-RF08: Se registra una incidencia por fallo de la aplicación

| | |
|--|--|
| Código | CP-RF08 |
| Nombre | Se registra una incidencia por fallo de la aplicación |
| Identificador de requerimientos | RF08 |
| Escenario | Sistema de notificación de incidencias |
| Datos de entrada | - El servicio web de la aplicación está caído |
| Resultados | |
| Resultados esperados | <ul style="list-style-type: none">- La aplicación no puede conectarse al servicio web por lo que se generará una excepción- Se registrará el error en base de datos- Saldrá un aviso de error al usuario indicando que no se han recuperado los datos correctamente- Los desarrolladores responsables de esa parte de la aplicación reciben un correo para que sepan que se ha insertado una incidencia |
| Resultados obtenidos | <ul style="list-style-type: none">- Se inserta el error en base de datos correctamente- Los desarrolladores responsables de esa parte de la aplicación reciben un correo correctamente para que sepan que se ha insertado una incidencia |
| Resultado | Correcto |

Tabla 41: Casos de prueba – CP-RF13: Consultar incidencias en desarrollo por el usuario conectado

| | |
|--|---|
| Código | CP-RF13 |
| Nombre | Consultar incidencias en desarrollo por el usuario conectado |
| Identificador de requerimientos | RF13 |
| Escenario | Sistema de gestión de incidencias |
| Datos de entrada | <ul style="list-style-type: none">- Pulsar en “Mostrar incidencias del desarrollador conectado”- No seleccionar el check de “Incidencias solucionadas” |
| Resultados | |
| Resultados esperados | El usuario (que en este caso además estará dado de alta como desarrollador) puede ver en la consulta las incidencias que le corresponden y se encuentran en desarrollo |
| Resultados obtenidos | El usuario visualiza correctamente la consulta con las incidencias que le corresponden y se encuentran en desarrollo |
| Resultado | Correcto |

Tabla 42: Casos de prueba – CP-RF14: Consultar incidencias solucionadas por el usuario conectado

| | |
|--|---|
| Código | CP-RF14 |
| Nombre | Consultar incidencias solucionadas por el usuario conectado |
| Identificador de requerimientos | RF14 |
| Escenario | Sistema de gestión de incidencias |
| Datos de entrada | - Pulsar en “Mostrar incidencias del desarrollador conectado” - Seleccionar el check de “Incidencias solucionadas” |
| Resultados | |
| Resultados esperados | El usuario (que en este caso además estará dado de alta como desarrollador) puede ver en la consulta las incidencias que le corresponden y se encuentran solucionadas |
| Resultados obtenidos | El usuario visualiza correctamente la consulta con las incidencias que le corresponden y se encuentran solucionadas |
| Resultado | Correcto |

Tabla 43: Casos de prueba – CP-RF15: Consultar incidencias sin asignar y en desarrollo por todos los desarrolladores

| | |
|--|--|
| Código | CP-RF15 |
| Nombre | Consultar incidencias sin asignar y en desarrollo por todos los desarrolladores |
| Identificador de requerimientos | RF15 |
| Escenario | Sistema de gestión de incidencias |
| Datos de entrada | - Pulsar en “Mostrar incidencias de todos los desarrolladores” - No seleccionar el check de “Incidencias solucionadas” |
| Resultados | |
| Resultados esperados | El usuario (que en este caso además estará dado de alta como desarrollador) puede ver en la consulta todas las incidencias que se encuentran sin asignar o en desarrollo |
| Resultados obtenidos | El usuario visualiza correctamente la consulta con todas las incidencias que se encuentran sin asignar o en desarrollo |
| Resultado | Correcto |

Tabla 44: Casos de prueba – CP-RF16: Consultar incidencias solucionadas por todos los desarrolladores

| | |
|--|---|
| Código | CP-RF16 |
| Nombre | Consultar incidencias solucionadas por todos los desarrolladores |
| Identificador de requerimientos | RF16 |
| Escenario | Sistema de gestión de incidencias |
| Datos de entrada | - Pulsar en “Mostrar incidencias de todos los desarrolladores” - Seleccionar el check de “Incidencias solucionadas” |
| Resultados | |
| Resultados esperados | El usuario (que en este caso además estará dado de alta como desarrollador) puede ver en la consulta todas las incidencias que se encuentran solucionadas |
| Resultados obtenidos | El usuario visualiza correctamente la consulta con todas las incidencias que se encuentran solucionadas |
| Resultado | Correcto |

Tabla 45: Casos de prueba – CP-RF17: Asignar incidencia manualmente

| | |
|--|---|
| Código | CP-RF17 |
| Nombre | Asignar incidencia manualmente |
| Identificador de requerimientos | RF17 |
| Escenario | Sistema de gestión de incidencias |
| Datos de entrada | - Pulsar en “Incidencias sin asignar” - Pulsar en “Asignarme incidencia” |
| Resultados | |
| Resultados esperados | - La incidencia se asigna al usuario conectado - La incidencias asignada ya no aparece en esta consulta |
| Resultados obtenidos | - La incidencia se ha asignado correctamente al usuario conectado - La incidencias asignada ya no aparece en esta consulta |
| Resultado | Correcto |

Tabla 46: Casos de prueba – CP-RF18: Cerrar incidencia

| | |
|--|--|
| Código | CP-RF18 |
| Nombre | Cerrar incidencia |
| Identificador de requerimientos | RF18 |
| Escenario | Sistema de gestión de incidencias |
| Datos de entrada | <ul style="list-style-type: none">- Seleccionar la consulta incidencias en desarrollo por el usuario conectado- Pulsar el botón “Cerrar incidencia”- Fecha solución = “01/07/2015”- Pulsar en “Aceptar” |
| Resultados | |
| Resultados esperados | <ul style="list-style-type: none">- La incidencia cambia su estado de “En desarrollo” a “Solucionada”- La incidencia aparece en la consulta de incidencias solucionadas- Se envía un correo al usuario que introdujo la incidencia indicando que ya se ha solucionado |
| Resultados obtenidos | <ul style="list-style-type: none">- La incidencia ha cambiado su estado de “En desarrollo” a “Solucionada”- La incidencia aparece en la consulta de incidencias solucionadas- Se envía un correo correctamente al usuario que introdujo la incidencia indicando que ya se ha solucionado |
| Resultado | Correcto |

Tabla 47: Casos de prueba – CP-RF19: Generar informes

| | |
|--|--|
| Código | CP-RF19 |
| Nombre | Generar informes |
| Identificador de requerimientos | RF19 |
| Escenario | Sistema de gestión de incidencias |
| Datos de entrada | - Pulsar el botón “Exportar datos” - Elegir la opción “Selección actual” |
| Resultados | |
| Resultados esperados | - Se exportan los datos que aparecen en la consulta en el momento de exportar los datos |
| Resultados obtenidos | - Se han exportado los datos que aparecen en la consulta en el momento de exportar los datos correctamente |
| Resultado | Correcto |

7.2.1 Otras pruebas

Además de las pruebas realizadas referidas a los requisitos, se ha realizado una carga masiva de datos para comprobar qué sucedería en un caso de estas características.

En la siguiente imagen se puede ver que se han insertado unos 60 errores a la vez, este error ha sido provocado a propósito para poder comprobar cómo actuaría el sistema en una situación similar.

| Id | Descripci... | CuentaDe... | FechaError |
|-----|--------------|-------------|-------------------------|
| 485 | Acceso de... | Laura | 2015-07-04 20:14:46.243 |
| 486 | Acceso de... | Laura | 2015-07-04 20:14:46.250 |
| 487 | Acceso de... | Laura | 2015-07-04 20:14:46.307 |
| 488 | Acceso de... | Laura | 2015-07-04 20:14:46.307 |
| 489 | Acceso de... | Laura | 2015-07-04 20:14:46.847 |
| 490 | Acceso de... | Laura | 2015-07-04 20:14:46.853 |
| 491 | Acceso de... | Laura | 2015-07-04 20:14:46.847 |
| 492 | Acceso de... | Laura | 2015-07-04 20:14:46.993 |
| 493 | Acceso de... | Laura | 2015-07-04 20:14:46.997 |
| 494 | Acceso de... | Laura | 2015-07-04 20:14:46.997 |
| 495 | Acceso de... | Laura | 2015-07-04 20:14:47.173 |
| 496 | Acceso de... | Laura | 2015-07-04 20:14:47.173 |
| 497 | Acceso de... | Laura | 2015-07-04 20:14:47.983 |
| 498 | Acceso de... | Laura | 2015-07-04 20:14:48.133 |
| 499 | Acceso de... | Laura | 2015-07-04 20:14:48.147 |
| 500 | Acceso de... | Laura | 2015-07-04 20:14:48.150 |
| 501 | Acceso de... | Laura | 2015-07-04 20:14:48.257 |
| 502 | Acceso de... | Laura | 2015-07-04 20:14:48.457 |
| 503 | Acceso de... | Laura | 2015-07-04 20:14:48.460 |
| 504 | Acceso de... | Laura | 2015-07-04 20:14:48.560 |
| 505 | Acceso de... | Laura | 2015-07-04 20:14:48.560 |
| 506 | Acceso de... | Laura | 2015-07-04 20:14:48.570 |
| 507 | Acceso de... | Laura | 2015-07-04 20:14:48.813 |
| 508 | Acceso de... | Laura | 2015-07-04 20:14:48.863 |
| 509 | Acceso de... | Laura | 2015-07-04 20:14:48.933 |
| 510 | Acceso de... | Laura | 2015-07-04 20:14:49.117 |
| 511 | Acceso de... | Laura | 2015-07-04 20:14:49.113 |
| 512 | Acceso de... | Laura | 2015-07-04 20:14:49.143 |
| 513 | Acceso de... | Laura | 2015-07-04 20:14:50.070 |
| 514 | Acceso de... | Laura | 2015-07-04 20:14:50.070 |
| 515 | Acceso de... | Laura | 2015-07-04 20:14:51.070 |
| 516 | Acceso de... | Laura | 2015-07-04 20:14:51.060 |
| 517 | Acceso de... | Laura | 2015-07-04 20:14:51.223 |
| 518 | Acceso de... | Laura | 2015-07-04 20:14:51.230 |
| 519 | Acceso de... | Laura | 2015-07-04 20:14:51.233 |
| 520 | Acceso de... | Laura | 2015-07-04 20:14:51.433 |
| 521 | Acceso de... | Laura | 2015-07-04 20:14:51.443 |
| 522 | Acceso de... | Laura | 2015-07-04 20:14:51.450 |
| 523 | Acceso de... | Laura | 2015-07-04 20:14:51.483 |
| 524 | Acceso de... | Laura | 2015-07-04 20:14:51.500 |
| 525 | Acceso de... | Laura | 2015-07-04 20:14:51.520 |
| 526 | Acceso de... | Laura | 2015-07-04 20:14:51.530 |
| 527 | Acceso de... | Laura | 2015-07-04 20:14:51.533 |
| 528 | Acceso de... | Laura | 2015-07-04 20:14:51.533 |
| 529 | Acceso de... | Laura | 2015-07-04 20:14:51.563 |
| 530 | Acceso de... | Laura | 2015-07-04 20:14:51.563 |
| 531 | Acceso de... | Laura | 2015-07-04 20:14:51.627 |
| 532 | Acceso de... | Laura | 2015-07-04 20:14:51.670 |
| 533 | Acceso de... | Laura | 2015-07-04 20:14:51.803 |
| 534 | Acceso de... | Laura | 2015-07-04 20:14:51.810 |
| 535 | Acceso de... | Laura | 2015-07-04 20:14:51.803 |
| 536 | Acceso de... | Laura | 2015-07-04 20:14:51.857 |
| 537 | Acceso de... | Laura | 2015-07-04 20:14:51.860 |
| 538 | Acceso de... | Laura | 2015-07-04 20:14:51.913 |
| 539 | Acceso de... | Laura | 2015-07-04 20:14:51.930 |
| 540 | Acceso de... | Laura | 2015-07-04 20:14:51.940 |
| 541 | Acceso de... | Laura | 2015-07-04 20:14:51.940 |
| 542 | Acceso de... | Laura | 2015-07-04 20:14:51.957 |
| 543 | Acceso de... | Laura | 2015-07-04 20:14:51.980 |
| 544 | Acceso de... | Laura | 2015-07-04 20:14:51.990 |
| * | NULL | NULL | NULL |

Ilustración 43: Carga masiva de datos



De todas maneras al ser una intranet el número de usuarios está controlado. Ahora mismo en el sistema hay datos de alta 308 usuarios que están repartidos en 19 aplicaciones y la aplicación con mayor número de usuarios tiene 122. Por lo que si todos los usuarios estuvieran usando las aplicaciones a la vez y tuviera lugar cualquier error a nivel general, por ejemplo una caída de la red, el sistema no tendría problema en soportar una carga de este rango de datos e incluso superior.

8 ANÁLISIS ECONÓMICO

En este punto se proporcionará un presupuesto aproximado sobre lo que hubiera costado económicamente este proyecto.

Para ello se ha hecho una planificación inicial y una planificación real del proyecto, se han desglosado las tareas y se explica detalladamente el tiempo empleado en cada una de ellas y finalmente cada una será cuantificada en función de su duración y los recursos consumidos, tanto personales como materiales.

8.1 Planificación

Finalizadas las reuniones con el Jefe de proyecto para la toma de requisitos y tras marcar una fecha aproximada de entrega se realizó una planificación inicial.

La fecha de inicio fue el 10 de marzo de 2015 y ya que el cliente no fijó una fecha límite de entrega, la fecha que se marcó como fin fue el 10 de julio de 2015.

Teniendo un periodo de cuatro meses para realizar el proyecto y conociendo la experiencia de los trabajadores se distribuyen esos cuatro meses de manera que se trabaje 3h de lunes a viernes y 6h los sábados y domingos. Lo que supondría aproximadamente 430horas de trabajo.

Hay tareas que se podrían haber hecho en paralelo pero como en este caso ha sido una única persona la que ha desempeñado todos los roles se ha puesto que hasta que no se terminaba de hacer una tarea no se comenzaba a realizar la siguiente.

A continuación se puede ver el diagrama de Gantt con la planificación inicial con los hitos de inicio y fin de proyecto.

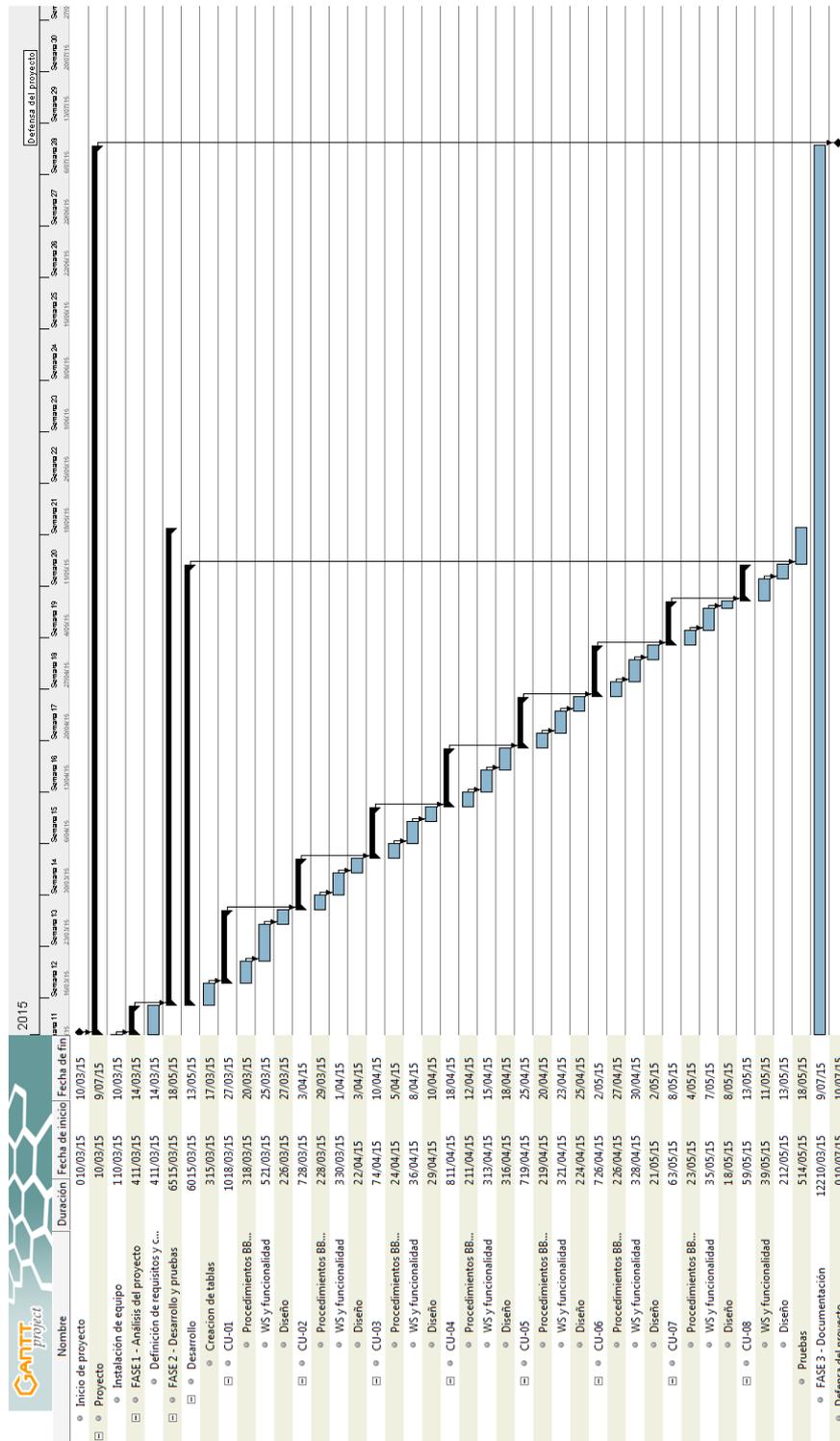


Ilustración 44: Diagrama de Gantt – Planificación inicial

Finalizado el proyecto se vuelve a hacer otro diagrama de Gantt con los valores reales.

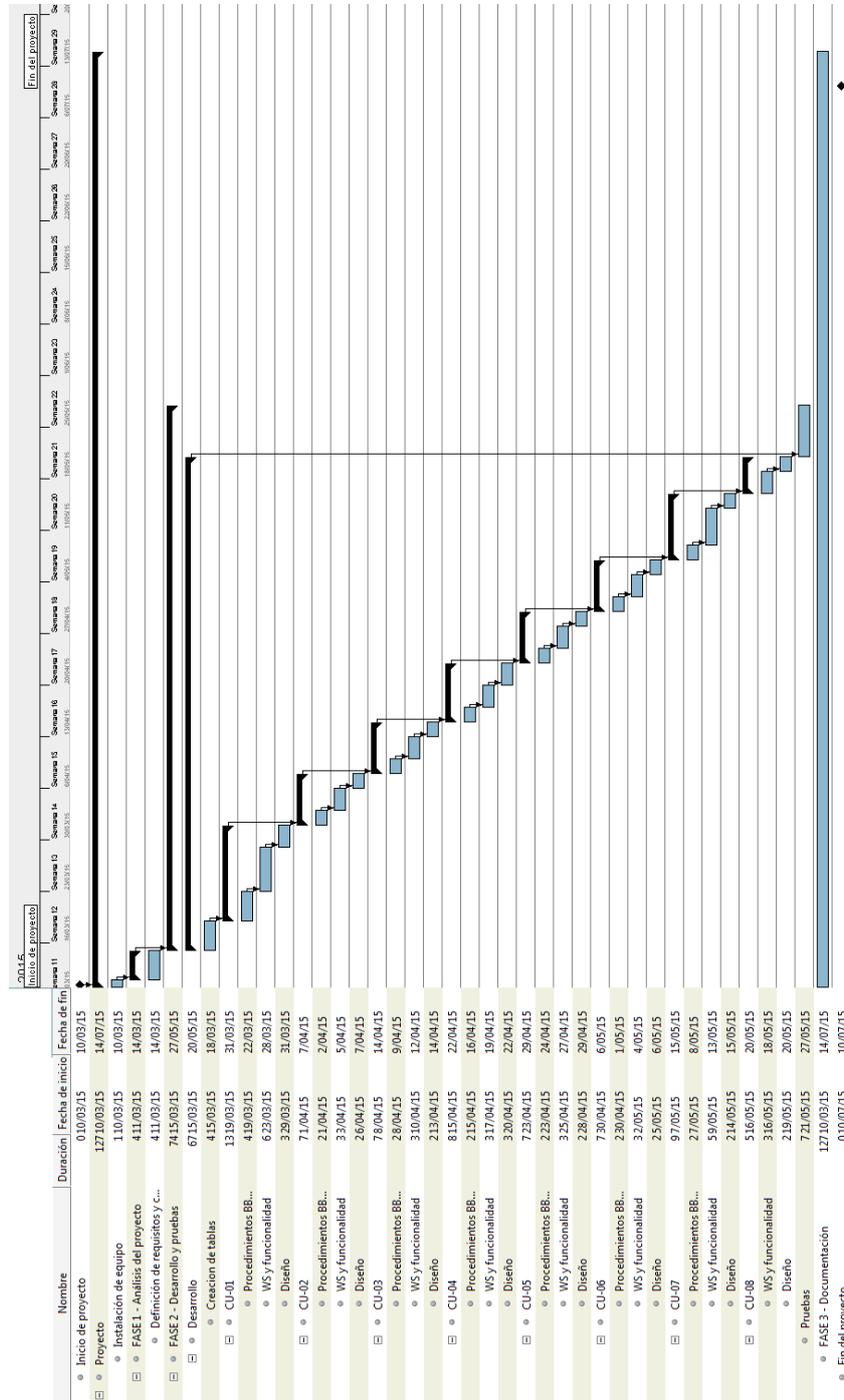


Ilustración 45: Diagrama de Gantt – Planificación real

Se puede observar que en lugar de terminar el día 10 de julio de 2015 como estaba previsto se termina el 14 de julio de 2015. Esto puede haberse debido a que no se tuvo en cuenta en la planificación inicial imprevistos como problemas encontrados durante el desarrollo o ausencia de los trabajadores por motivos personales.

Por lo que la **duración total del proyecto han sido 440h** aproximadamente.

8.2 Desglose de tareas

A continuación se detallan las tareas que han compuesto el proyecto.

El proyecto se ha dividido en tres tareas:

Tabla 48: Desglose de tareas

| TAREA | DESCRIPCIÓN |
|-------|---------------------------------------|
| 1 | Análisis del proyecto. |
| 2 | Desarrollo y pruebas. |
| 3 | Documentación y memoria del proyecto. |

Cada una de las tareas se desglosa en subtareas más descriptivas indicando el tiempo empleado en cada una de ellas.

Aunque a continuación se podrá ver que han intervenido en la realización del proyecto una serie de roles, todos ellos los va a realizar una única persona, excepto el de Jefe de proyecto.

8.2.1 TAREA 1. Análisis del proyecto.

En la tarea 1 intervienen el Jefe de proyecto del proyecto donde se van a implantar estas aplicaciones, un Analista funcional, un Analista programador de bases de datos y un Analista programador de .NET.

Tabla 49: Tarea 1 – Análisis del proyecto

| SUBTAREA | DESCRIPCIÓN | PERSONAS IMPLICADAS | DURACIÓN |
|----------|---|---|----------|
| 1.1 | Análisis y definición de requisitos y casos de uso. | Analista funcional y Jefe de proyecto del proyecto en el cliente. | 16 horas |
| 1.2 | Definición de base de datos | Analista programador de base de datos | 8 horas |
| 1.3 | Definición de la aplicación y diseño | Analista programador de .NET | 8 horas |

8.2.2 TAREA 2. Desarrollo y pruebas.

La tarea 2 es trabajo a realizar por un programador de base de datos y un programador de .NET.

Tabla 50: Tarea 2 – Desarrollo y pruebas

| SUBTAREA | DESCRIPCIÓN | PERSONAS IMPLICADAS | DURACIÓN |
|----------|--|--|-----------|
| 2.1 | Desarrollo de tablas y procedimientos de base de datos | Programador de base de datos | 50 horas |
| 2.2 | Desarrollo de interfaz gráfica | Programador de .NET | 30 horas |
| 2.3 | Desarrollo de la lógica del programa | Programador de .NET | 130 horas |
| 2.4 | Pruebas y resolución de errores | Programador de base de datos y Programador de .NET | 30 horas |

8.2.3 TAREA 3. Documentación y memoria del proyecto.

Por último la tarea 3 es trabajo a realizar por un documentalista.

Tabla 51: Tarea 3 – Documentación y memoria del proyecto

| SUBTAREA | DESCRIPCIÓN | PERSONAS IMPLICADAS | DURACIÓN |
|----------|---------------------------------------|---------------------|-----------|
| 3.1 | Redacción de la memoria del proyecto. | Documentalista | 170 horas |

8.8.4 Resumen de tareas

En la siguiente tabla se puede ver un resumen de las tareas y subtareas descritas anteriormente y la suma final de la duración total del proyecto.

Tabla 52: Resumen de tareas

| TAREA | SUBTAREA | DURACIÓN |
|--|----------|------------------|
| 1. Análisis del proyecto. | 1.1 | 16 horas |
| | 1.2 | 8 horas |
| | 1.3 | 8 horas |
| 2. Desarrollo y pruebas. | 2.1 | 50 horas |
| | 2.2 | 30 horas |
| | 2.3 | 130 horas |
| | 2.4 | 30 horas |
| 3. Documentación y memoria del proyecto. | 3.1 | 170 horas |
| TOTAL | | 442 horas |

Según se indica en la anterior tabla se puede ver que la duración del proyecto han sido **442 horas**. Como sólo ha habido una persona dedicada al proyecto, esas 442 horas será la dedicación total del proyecto.

Si esas horas las traducimos a una jornada laboral de 40 horas semanales se puede comprobar que el proyecto hubiera tardado en realizarse aproximadamente **1 mes y 3 semanas**.

8.3 Costes

En este punto se calcularán los costes del proyecto. Por un lado se calcularán los costes materiales de los recursos que hayan sido necesarios durante la realización del proyecto y por otro los costes de personal teniendo en cuenta el tiempo que se ha detallado en el punto anterior y los sueldos de los roles de las personas que lo han realizado. Al final de este punto se podrá ver el coste total del proyecto.

8.2.1 Costes materiales

La siguiente tabla muestra los costes de los recursos necesarios para la realización del proyecto.

Tabla 53: Costes materiales

| CONCEPTO | COSTE |
|---|---------------|
| Ordenador Portátil con Windows 7 64bits | 400€ |
| Visual Studio 2010 | 550€ |
| Microsoft Office 2013 | 539€ |
| TOTAL | 1.489€ |

8.2.2 Costes de personal

La siguiente tabla muestra los costes de los sueldos de los roles que han realizado el proyecto.

A continuación se ven los roles que han intervenido en la realización del proyecto, las horas trabajadas, su sueldo y el coste total del personal.

Tabla 54: Costes de personal

| OCUPACIÓN | HORAS | PRECIO/HORA | IMPORTE |
|---------------------------------------|------------------|-------------|----------------|
| Jefe de proyecto | 8 | 70€/hora | 560€ |
| Analista funcional | 8 | 50€/hora | 400€ |
| Analista programador de base de datos | 8 | 35€/hora | 280€ |
| Analista programador de .NET | 8 | 35€/hora | 280€ |
| Programador de base de datos | 65 | 30€/hora | 1.950€ |
| Programador de .NET | 175 | 30€/hora | 5.250€ |
| Documentalista | 170 | 20€/hora | 3.400€ |
| TOTAL | 442 HORAS | | 12.120€ |

8.2.3 Costes indirectos

En la siguiente tabla se muestran los costes indirectos que se aplican sobre el 10% de la suma del coste material y personal.

Tabla 55: Costes indirectos

| CONCEPTO | COSTE |
|---------------------------------------|----------------------------|
| Costes materiales + Costes personales | 1.489€ + 12.120€ = 13.609€ |
| Porcentaje Costes indirectos | 10% |
| TOTAL | 1.360,9€ |

8.2.4 Costes totales

El coste total del proyecto se desglosa en la siguiente tabla:

Tabla 56: Costes totales

| CONCEPTO | COSTE |
|-------------------|-------------------|
| Costes materiales | 1.489€ |
| Coste de personal | 12.120€ |
| Costes indirectos | 1.360,9€ |
| Subtotal | 14.969,9€ |
| I.V.A (21%) | 3.143,68€ |
| TOTAL | 18.113,58€ |

Por tanto el coste total del proyecto serian **18.113,58€**

9 CONCLUSIONES Y TRABAJOS FUTUROS

9.1 Conclusiones

Todos los objetivos que se plantearon al principio se han alcanzado e incluso sobre la marcha se fueron añadiendo nuevas funcionalidades que proporcionaban un valor añadido a la solución.

Hubo algunos casos en los que tras mucho pensar y encontrar obstáculos para poder realizar algunas tareas se acabó optando por buscar alternativas. Algunos de estos casos se han explicado previamente en el apartado de [Evaluación y pruebas](#) y se consiguieron solventar ya fuera investigando o buscando alternativas.

En cuanto al aprendizaje de nuevas tecnologías puedo decir que tras la realización del PFC he adquirido bastante conocimiento sobre el servidor web IIS ya que previamente no había instalado ninguno desde cero ni investigado tan profundamente.

Respecto a base de datos, antes de comenzar con el PFC tenía conocimientos más bien básicos ya que no suelo programar bases de datos y había olvidado muchos conocimientos aprendidos durante la carrera. Aunque en el desarrollo del PFC no se han realizado procedimientos muy complicados me ha servido para recordar y que me hayan entrado ganas de volver a retomar este tema.

Como conclusión final tras la realización del PFC tengo que decir que he acabado muy satisfecha ya que todo lo he realizado sola desde cero y me ha servido para aprender a planificar las tareas, tiempos y darme cuenta de la importancia que tiene hacer una buena toma de requisitos.

9.2 Trabajos futuros

A continuación se exponen algunas opciones para futuros trabajos:

- Aunque algunos usuarios ya han podido probar el nuevo sistema de notificación de incidencias, a finales del mes de Julio aprovechando que hay menos usuarios se terminará de implantar en el cliente y todo tendrá que seguir funcionando correctamente.
- Cuando haya mayor número de incidencias añadir más filtros como por ejemplo filtrar por años.
- Desarrollar otro módulo para que el usuario además de incidencias pueda introducir posibles mejoras para el sistema y de esta manera queden registradas para su posterior evaluación.



10 REFERENCIAS

[1] MSDN, Microsoft Developer Network:

<https://msdn.microsoft.com>

[2] Deepu MI's Blog - WordPress.com:

<https://deepumi.wordpress.com/2010/02/21/silverlight-change-button-background-color/>

[3] Microsoft - página oficial:

<http://windows.microsoft.com/es-es/windows-vista/install-internet-information-services-iis-7-0>

[4] MSDN, Microsoft Developer Network:

<https://msdn.microsoft.com/es-es/library/ms191189.aspx>

[5] TechNet - Microsoft:

[https://technet.microsoft.com/es-es/library/ms187540\(v=sql.105\).aspx](https://technet.microsoft.com/es-es/library/ms187540(v=sql.105).aspx)

[6] Hernan Vivani's Blog | linux, electronics, bigdata, astronomy:

<http://hvivani.com.ar/2013/09/20/enviar-correo-desde-sql-server-utilizando-cuenta-de-gmail-send-email-from-sql-server-using-gmail-account/>

[7] Google Help:

<https://support.google.com/mail/answer/78775?hl=es>

<https://support.google.com/accounts/answer/6010255?hl=es>



[8] Msdn forums - Microsoft:

<https://social.msdn.microsoft.com/Forums/es-ES/986bc157-059c-4f72-9a4c-d8fa0076616d/enviar-email-desde-aspnet?forum=netfxwebes>

<https://social.msdn.microsoft.com/Forums/es-ES/11572512-f4a5-41e6-b5ac-cc7dcc74e364/mandar-un-mail-con-c-tengo-cdigo?forum=vcses>

<https://social.msdn.microsoft.com/Forums/es-ES/c56e84f6-fd53-4a96-89b4-21a15e8da02a/demora-al-enviar-email-por-smtp?forum=vcses>

[9] Stack Overflow:

<http://stackoverflow.com/questions/32260/sending-email-in-net-through-gmail>

[10] Sergio Inzunza - WordPress.com:

https://inxunxa.wordpress.com/2010/01/19/mail_csharp/

[11] Leandro Tuttini Blog:

<http://ltuttini.blogspot.com.ar/2012/07/mailmessage-envia-mail-con-gmail-13.html>

[12] codigo fuente | Aprendiz del NET - WordPress.com:

<https://codigofuente.net.wordpress.com/2012/06/20/mandar-correos-con-visual-estudio-2010-c/>

[13] Foros del Web, comunidad para aprender web:

<http://www.forosdelweb.com/f29/configurar-smtp-para-mandar-maels-298450/>

<http://www.forosdelweb.com/f59/como-configurar-smtp-para-envio-emails-288828/>

[14] Blogs - Geeks•ms:

<http://geeks.ms/blogs/jalarcon/archive/2007/06/23/c-243-mo-enviar-correo-electr-243-nico-por-c-243-digo-usando-cuentas-de-gmail.aspx>

[15] Mis Vivencias Web:

<http://vivenciasdiariaspropias.blogspot.com.es/2014/11/el-servidor-smtp-requiere-una-conexion.html>



[16] 7Sabores:

<http://7sabores.com/blog/enviar-correo-electronico-mediante-smtp-csharp>

[17] Iteramos.com: Preguntas de Stack Overflow en Español:

<http://www.iteramos.com/pregunta/5249/enviar-correo-electronico-a-traves-del-servidor-smtp-de-gmail-con-c>

ANEXO A: Manual de usuario del Sistema de notificación de incidencias

En cualquier página de la intranet aparecerá un botón para ir al Sistema de notificación de incidencias y poder insertar incidencias. El usuario que vea alguna anomalía en el funcionamiento o le haya salido un mensaje de error podrá pulsarlo para introducir una incidencia.



Sistema de notificación de incidencias

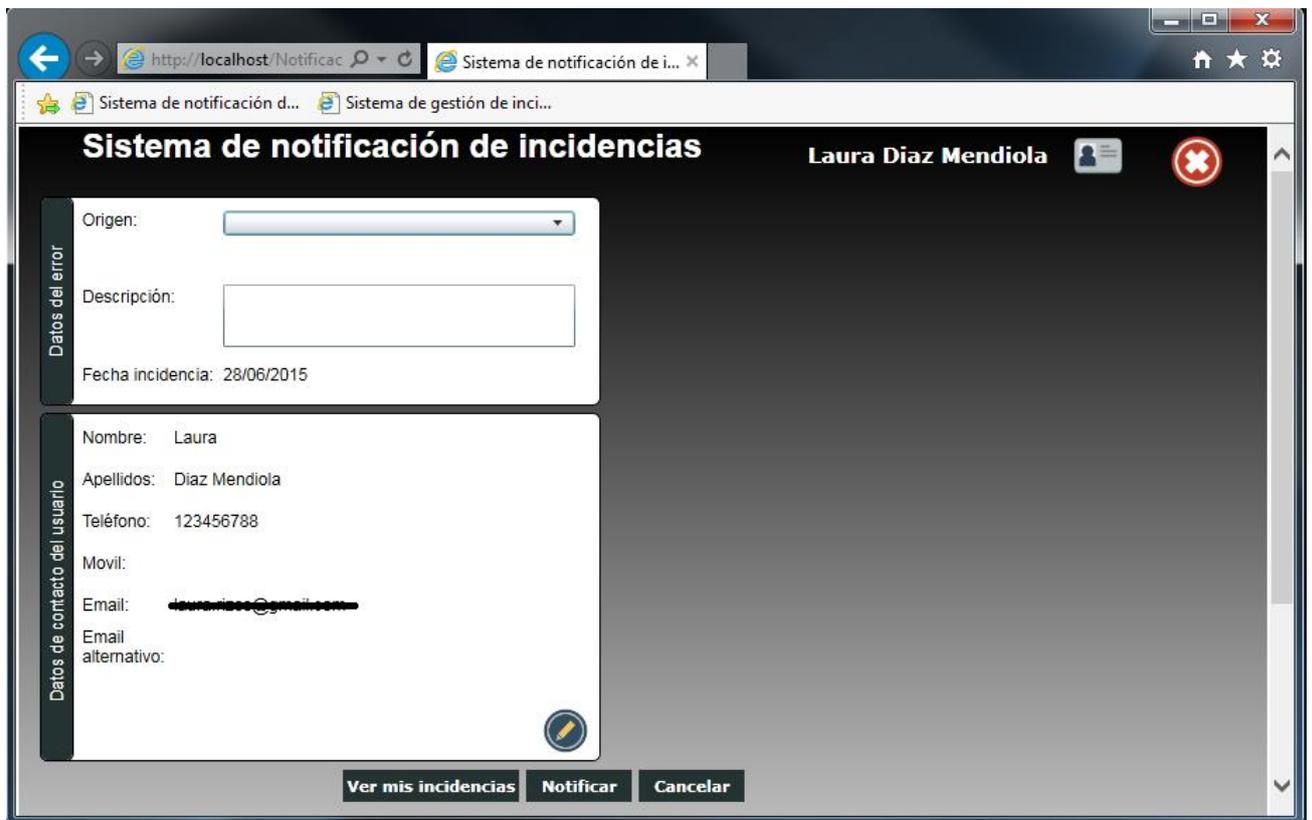
Ilustración 46: Botón para acceder al Sistema de notificación de incidencias

Al pulsar el botón de “Sistema de notificación de incidencias” el usuario irá a una ventana para poder introducir la información que crea conveniente para que los desarrolladores la solucionen. [[Ilustración 47](#)]

Esta ventana está dividida en dos:

En la parte superior se introducirán datos relacionados con la incidencia.

Y en la parte inferior se verán los datos referentes al usuario conectado.



Sistema de notificación de incidencias Laura Diaz Mendiola

Datos del error

Origen:

Descripción:

Fecha incidencia: 28/06/2015

Datos de contacto del usuario

Nombre: Laura

Apellidos: Diaz Mendiola

Teléfono: 123456788

Movil:

Email: [redacted]

Email alternativo:

Ilustración 47: Sistema de notificación de incidencias

En esta página se cargarán algunos datos referentes al usuario que se encuentre conectado.

Este podrá añadir o modificar alguno de los datos pulsando el botón inferior de “Modificar datos de contacto del usuario”.

Se abrirá otra ventana en la que podrá modificar o añadir los datos.

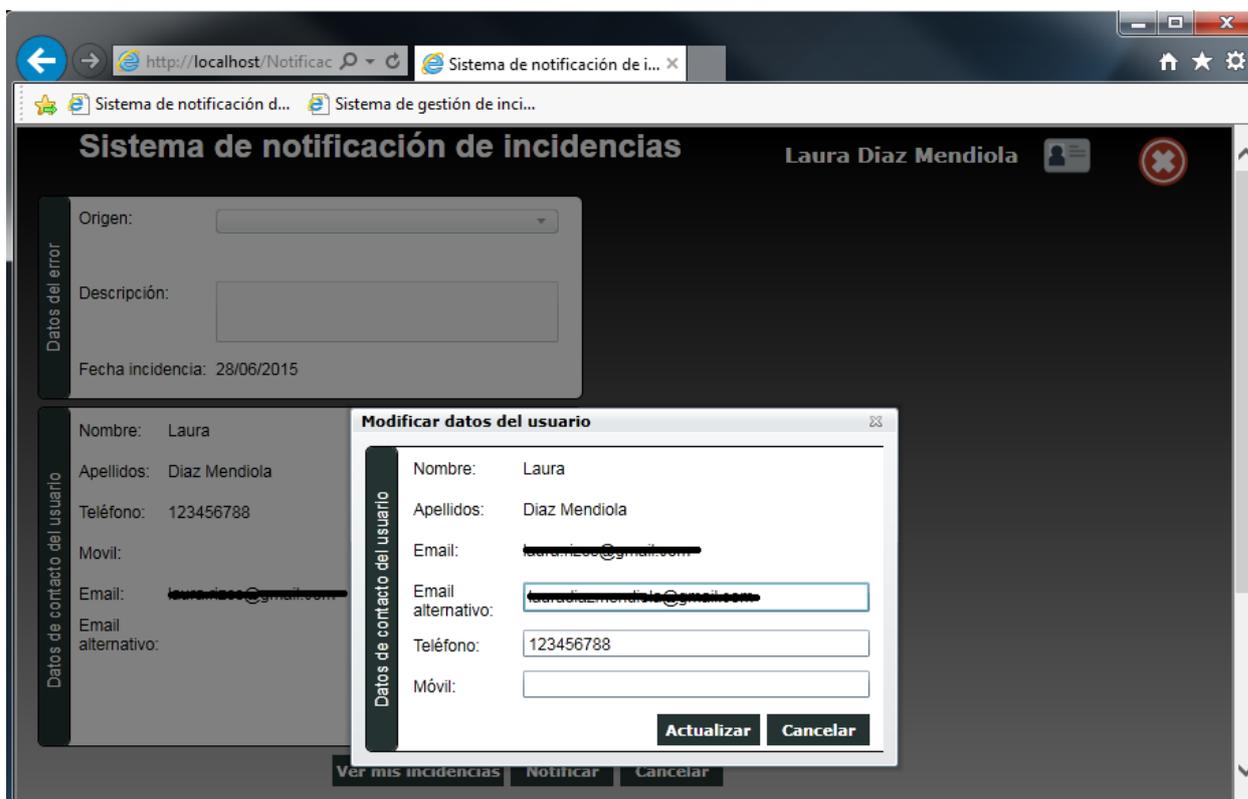
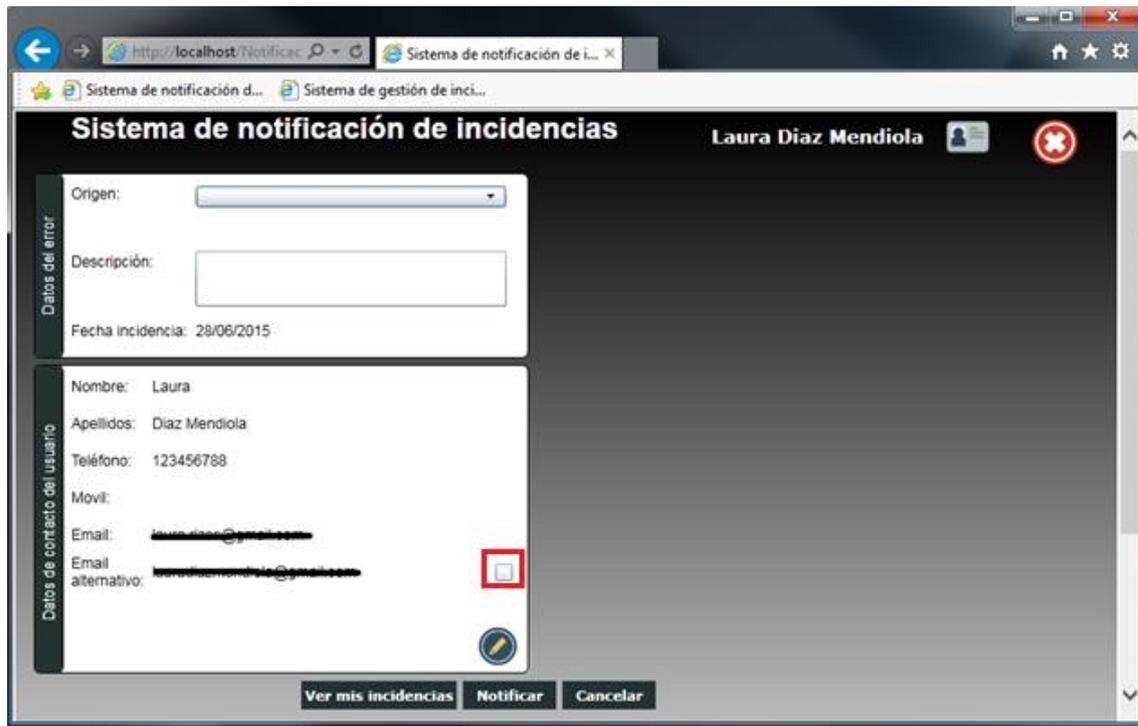


Ilustración 48: Sistema de notificación de incidencias - Modificar datos del usuario

El email principal no lo podrá modificar pero puede añadir otro para que los correos referentes a la incidencia se manden a este otro si lo prefiere.

Si se vuelve a la ventana principal se puede ver que cuando se introduce un segundo email sale un check para poder decir si se quiere que los correos se manden a este email o al principal.



Sistema de notificación de incidencias Laura Díaz Mendiola

Datos del error

Origen:

Descripción:

Fecha incidencia: 28/06/2015

Datos de contacto del usuario

Nombre: Laura

Apellidos: Diaz Mendiola

Teléfono: 123456788

Movil:

Email:

Email alternativo:

Ilustración 49: Sistema de notificación de incidencias

En la parte superior se ve información acerca de la incidencia como ya se ha dicho anteriormente.

El usuario podrá seleccionar la página y módulo donde ha tenido lugar el error o anomalía y escribir una descripción con lo que ha sucedido.

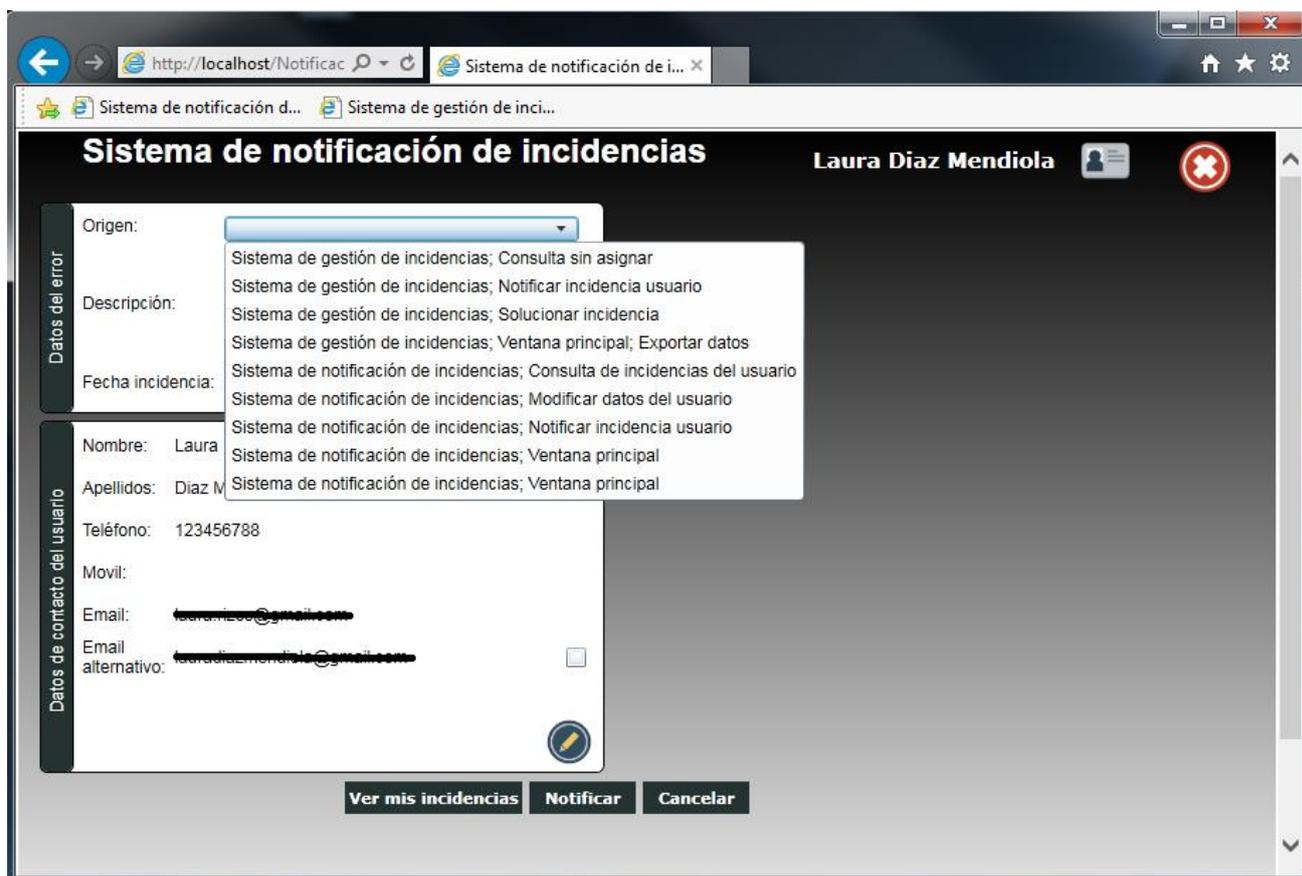


Ilustración 50: Sistema de notificación de incidencias – Datos de la incidencia

El usuario también podrá ver un histórico de incidencias pulsando en el botón inferior “Ver mis incidencias”.

En esta ventana se ven las incidencias que ha introducido el usuario conectado. Se puede ver la siguiente información:

- Número de referencia.
- Estado: es el estado en el que se encuentra la incidencia: Sin asignar, En desarrollo o Solucionada.
- Origen: es la página/aplicación donde tuvo lugar la incidencia.
- Fecha error: fecha en la que se introdujo la incidencia.
- Fecha solución: esta fecha aparecerá en caso de haberse solucionado la incidencia.

- Desarrollador: es la cuenta del desarrollador que resolvió la incidencia o se encuentra aún en ello por si hiciera falta ponerse en contacto con este desarrollador.
- Email desarrollador: es el email del desarrollador que resolvió la incidencia o se encuentra aún en ello por si hiciera falta ponerse en contacto con este desarrollador.

Arrastre una columna aquí para agrupar los datos por ella.

| Num. referencia | Estado | Origen | Fecha error | Fecha solución | Desarrollador | Email desarrollador |
|---|---------------|---|-------------|----------------|---------------------------|-----------------------|
| 332 | En desarrollo | Sistema de notificación de incidencias; Modificar datos del usuario | 28/06/2015 | | Laura Diaz Mendiola | XXXXXXXXXX@XXXXXXXXXX |
| <i>Descripción error:</i> No puedo modificar el correo | | | | | | |
| 414 | En desarrollo | Sistema de notificación de incidencias; Modificar datos del usuario | 28/06/2015 | | Laura Diaz Mendiola | XXXXXXXXXX@XXXXXXXXXX |
| 448 | En desarrollo | Sistema de notificación de incidencias; Modificar datos del usuario | 28/06/2015 | | Laura Diaz Mendiola | XXXXXXXXXX@XXXXXXXXXX |
| 476 | En desarrollo | Sistema de notificación de incidencias; Modificar datos del usuario | 29/06/2015 | | Laura Diaz Mendiola | XXXXXXXXXX@XXXXXXXXXX |
| 44 | Solucionada | Sistema de gestión de incidencias; Consulta sin asignar | 30/05/2015 | 31/05/2015 | Laura Diaz Mendiola | XXXXXXXXXX@XXXXXXXXXX |
| 45 | Solucionada | Sistema de notificación de incidencias; Ventana principal | 30/05/2015 | 31/05/2015 | desarrollador2 apellidos2 | XXXXXXXXXX@XXXXXXXXXX |
| 477 | Solucionada | Sistema de notificación de incidencias; Modificar datos del usuario | 29/06/2015 | 29/06/2015 | Laura Diaz Mendiola | XXXXXXXXXX@XXXXXXXXXX |
| 478 | Solucionada | Sistema de gestión de incidencias; Consulta sin asignar | 05/07/2015 | 05/07/2015 | Laura Diaz Mendiola | XXXXXXXXXX@XXXXXXXXXX |

Total: 8

Ilustración 51: Sistema de notificación de incidencias – Histórico de mis incidencias

Y ya por ultimo sólo queda Insertar la incidencia. Le daremos al botón inferior “Notificar” y si todo está relleno correctamente saldrá un aviso indicando que todo ha ido bien.

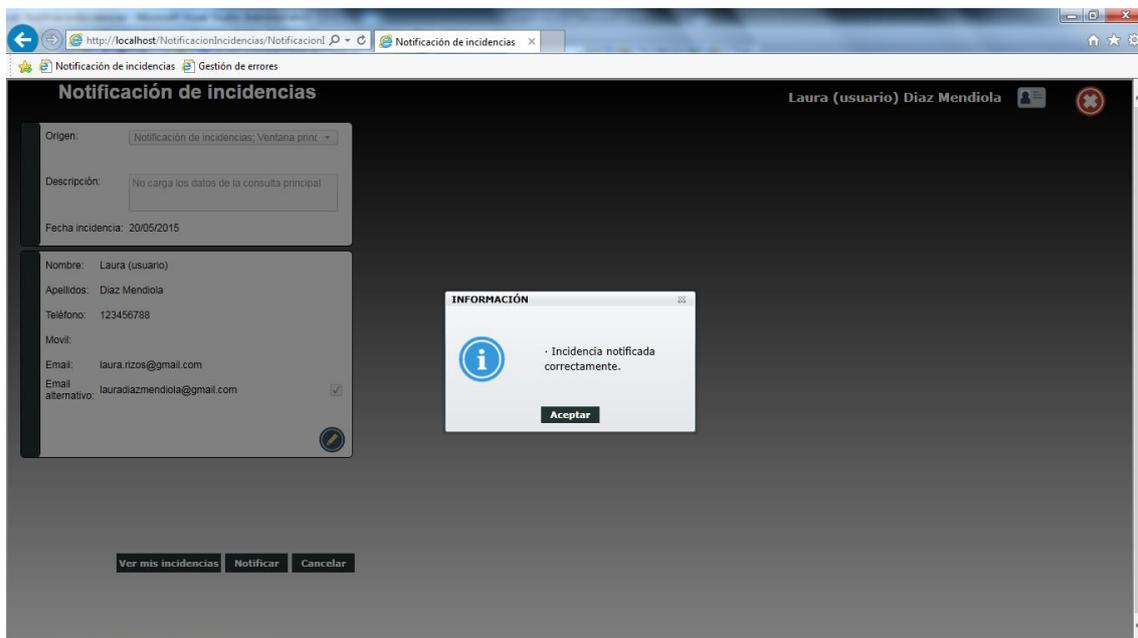


Ilustración 52: Sistema de notificación de incidencias – Notificar incidencia

Si vamos a la bandeja de entrada del correo que hemos indicado para que nos notifiquen podremos ver que hemos recibido un correo en el que se indican algunos datos referentes a la incidencia.

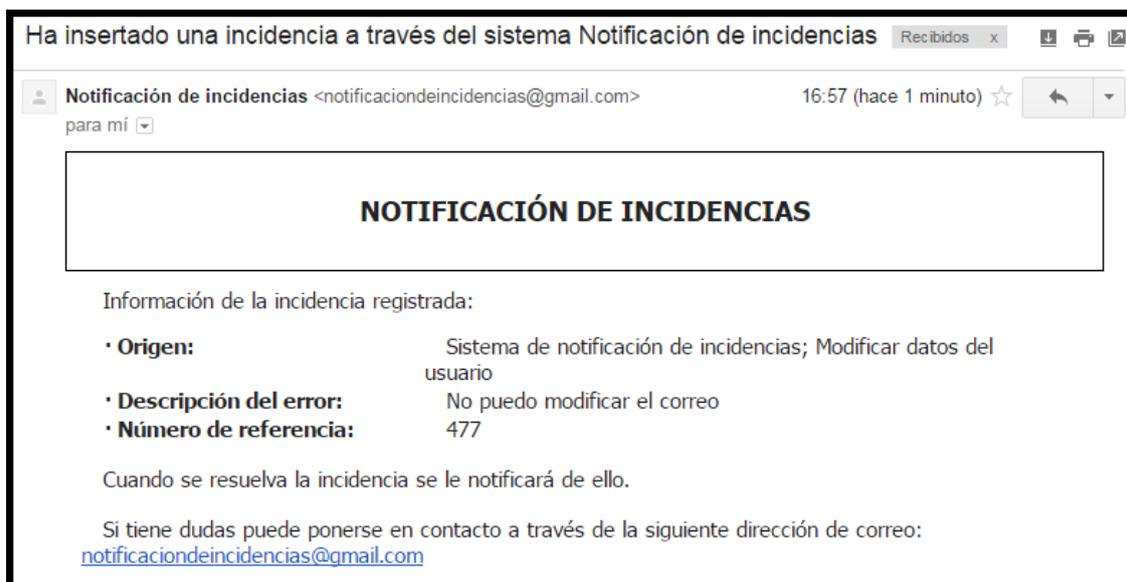


Ilustración 53: Sistema de notificación de incidencias – Envío de correo al usuario

Los desarrolladores de esa parte del código también recibirán un correo con algunos detalles de la incidencia.

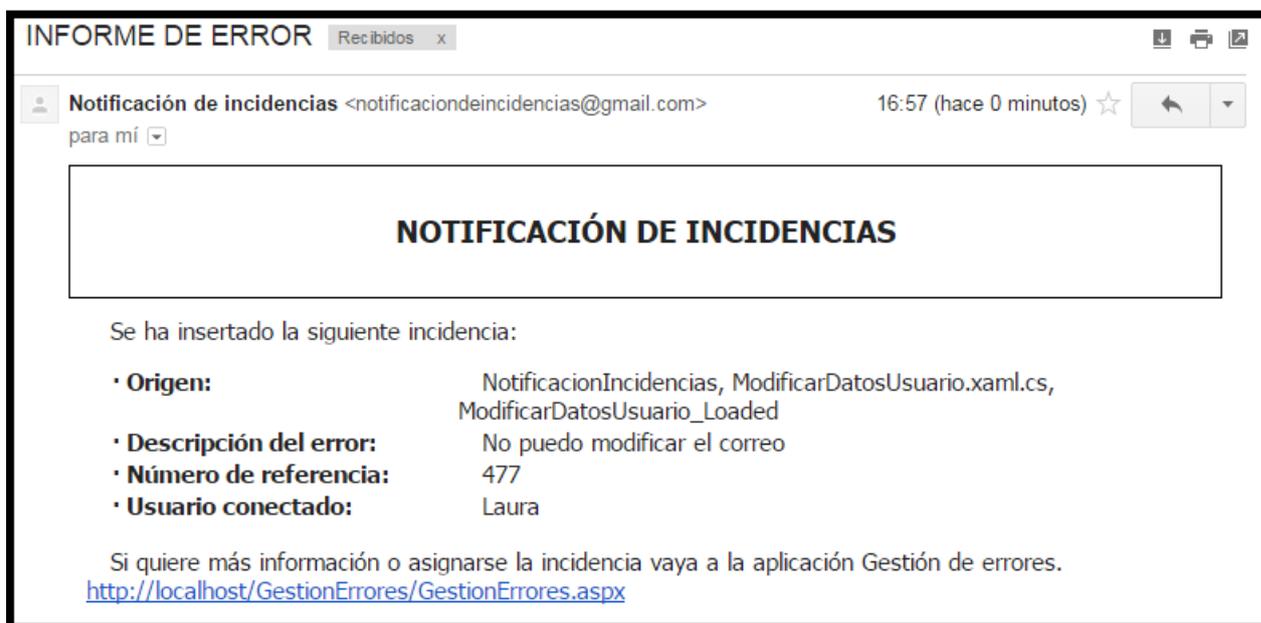


Ilustración 54: Sistema de notificación de incidencias – Envío de correo al desarrollador/es de mantenimiento

Cuando el equipo de mantenimiento la haya solucionado, el usuario volverá a recibir un correo indicando que ya se ha resuelto.

ANEXO B: Manual de usuario del Sistema de gestión de incidencias

A través del Sistema de gestión de incidencias podremos hacer un seguimiento de los errores que tienen lugar a través del código o las incidencias que introducen los usuarios.

Tendremos una consulta que mostrará estos datos y se podrá ir configurando al gusto del usuario. A continuación se irán explicando las distintas opciones.

Todas las consultas de primeras aparecerán ordenadas por Prioridad y Estado, primero las críticas y las que estén sin asignar.

En todas las consultas habrá una serie de columnas que siempre aparecerán que son:

- **Prioridad:** indica la prioridad de la incidencia. *Alta, Media o Baja*
- **Estado:** es el estado en el que se encuentra la incidencia. *Sin asignar, En desarrollo o Solucionada.*
- **Número de referencia.**
- **Origen de la incidencia.**
- **Fecha error:** es la fecha en la que ha tenido lugar el error.
- **Descripción del error:** si pulsamos sobre alguna de las incidencias veremos que se despliega y aparece la descripción del error.

Nada más entrar nos encontramos con una consulta que tendrá las incidencias que se encuentran en desarrollo por el desarrollador conectado. Si no tiene ninguna saldrá un aviso indicándolo. [[Ilustración 55](#)]

Consulta Incidencias en desarrollo por el usuario conectado: [[Ilustración 55](#) e [Ilustración 56](#)]

En esta consulta además de lo indicado anteriormente podremos ver las siguientes columnas:

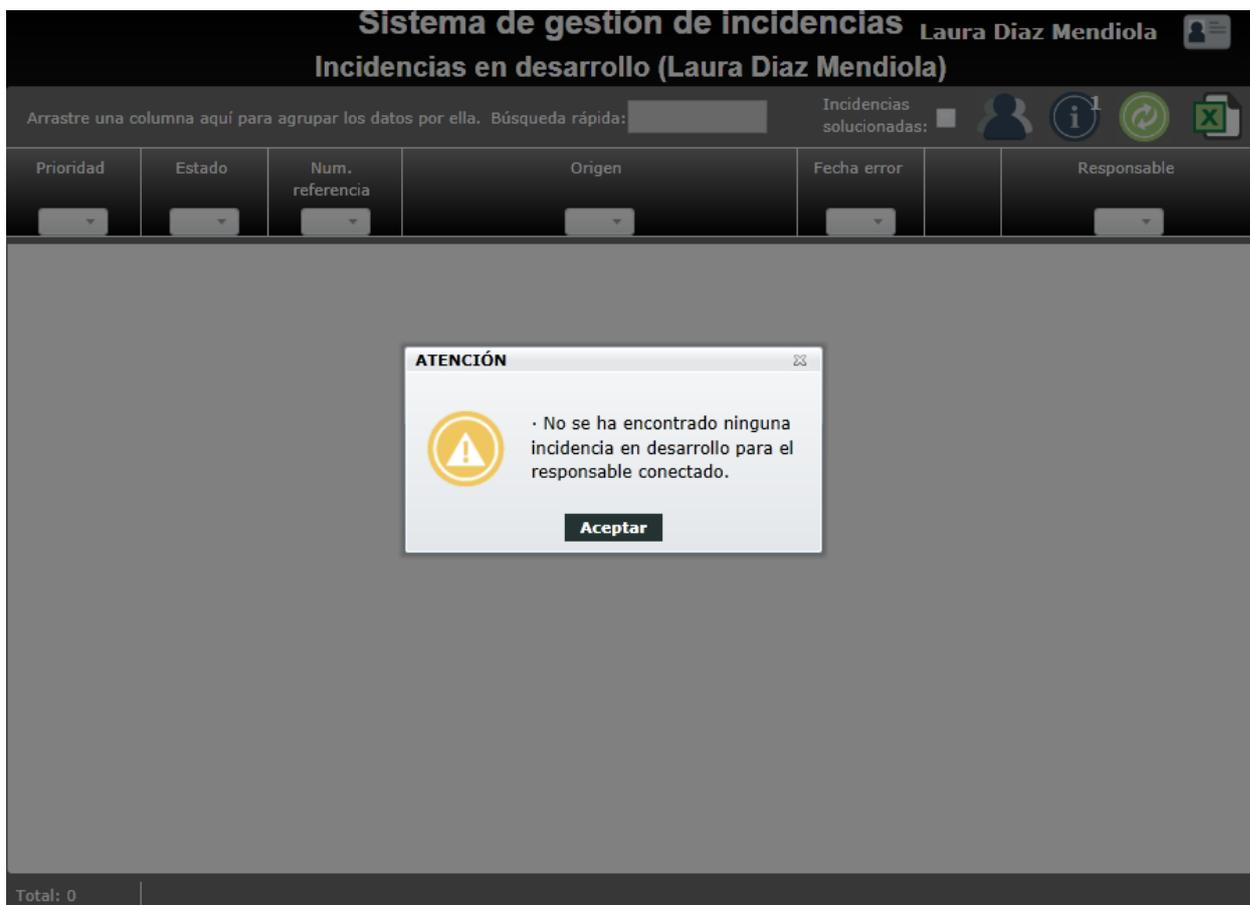
- **Desarrollador:** es el nombre del desarrollador que se está encargando de solucionar la incidencia. En este caso en esta columna sólo aparecerá el nombre del usuario conectado.
- **Botón para cerrar incidencia:** pulsando este botón el usuario conectado podrá dar por solucionadas sus incidencias.

Si pulsamos sobre el botón de cerrar incidencia aparecerá otra ventana. [[Ilustración 57](#)]

En esta ventana se pueden ver algunos datos de la incidencia. El desarrollador de esa incidencia antes de darla por solucionada tendrá que rellenar algunos datos, como son la fecha de solución (que no podrá ser mayor a la fecha de hoy), una breve descripción de lo que ha hecho para solucionar el problema y también podrá escribir unos comentarios adicionales.

Cuando le dé a “Guardar” se mandará un correo al usuario que aparezca en la parte superior de la ventana, notificándole que ya se ha corregido el error. [[Ilustración 58](#)]

Al cerrarse esta ventana la consulta general se actualizará y esta incidencia no aparecerá ya en esta consulta.



The screenshot displays the 'Sistema de gestión de incidencias' interface for user 'Laura Diaz Mendiola'. The main heading is 'Incidencias en desarrollo (Laura Diaz Mendiola)'. Below this, there is a search bar and a table with columns: 'Prioridad', 'Estado', 'Num. referencia', 'Origen', 'Fecha error', and 'Responsable'. A modal window titled 'ATENCIÓN' is centered on the screen, containing a warning icon and the text: '· No se ha encontrado ninguna incidencia en desarrollo para el responsable conectado.' with an 'Aceptar' button. At the bottom left, it shows 'Total: 0'.

Ilustración 55: Sistema de gestión de incidencias - Consulta Incidencias en desarrollo por el usuario conectado (consulta sin datos registrados)

| Sistema de gestión de incidencias | | | | | | | Laura Díaz Mendiola |
|--|---------------|-----------------|---|-------------|---------------------------|---------------------|---------------------|
| Incidencias en desarrollo (Laura Díaz Mendiola) | | | | | | | |
| Arrastre una columna aquí para agrupar los datos por ella. | | | Búsqueda rápida: | | Incidencias solucionadas: | | |
| Prioridad | Estado | Num. referencia | Origen | Fecha error | | Responsable | |
| Crítica | En desarrollo | 232 | NotificacionIncidencias, ModificarDatosUsuario.xaml.cs, ModificarDatosUsuario_Loaded | 28/06/2015 | + | Laura Díaz Mendiola | |
| Descripción error: No puedo modificar el correo | | | | | | | |
| Crítica | En desarrollo | 332 | NotificacionIncidencias, ModificarDatosUsuario.xaml.cs, ModificarDatosUsuario_Loaded | 28/06/2015 | + | Laura Díaz Mendiola | |
| Crítica | En desarrollo | 414 | NotificacionIncidencias, ModificarDatosUsuario.xaml.cs, ModificarDatosUsuario_Loaded | 28/06/2015 | + | Laura Díaz Mendiola | |
| Crítica | En desarrollo | 448 | NotificacionIncidencias, ModificarDatosUsuario.xaml.cs, ModificarDatosUsuario_Loaded | 28/06/2015 | + | Laura Díaz Mendiola | |
| Baja | En desarrollo | 70 | GestionIncidencias, Utiles.xaml.cs, WS_EnvioCorreoDesarrolladorCompleted | 31/05/2015 | + | Laura Díaz Mendiola | |
| Baja | En desarrollo | 72 | GestionIncidencias, SolucionarIncidencia.xaml.cs, WS_EnvioCorreoSolucionUsuarioCompleted | 31/05/2015 | + | Laura Díaz Mendiola | |
| Baja | En desarrollo | 437 | NotificacionIncidencias, ConsultaIncidenciasUsuario.xaml.cs, WS_ObtenerDatosIncidenciasUsuarioCompleted | 28/06/2015 | + | Laura Díaz Mendiola | |
| Baja | En desarrollo | 472 | NotificacionIncidencias, MainPage.xaml.cs, MainPage_Loaded | 29/06/2015 | + | Laura Díaz Mendiola | |
| Total: 10 | | | | | | | |

Ilustración 56: Sistema de gestión de incidencias - Consulta Incidencias en desarrollo por el usuario conectado

Sistema de gestión de incidencias Laura Diaz Mendiola

Incidentes en desarrollo (Laura Diaz Mendiola)

Solucionar incidencia

Núm. referencia: 232 Usuario de la incidencia: usuario1 apellidos1

Fecha error: 28/06/2015

Fecha solución: <dd/MM/yyyy> 15

Origen: NotificacionIncidencias, ModificarDatosUsuario.xaml.cs, ModificarDatosUsuario_Loaded

Descripción: No puedo modificar el correo

Responsable: Laura Diaz Mendiola

Acción correctora:

Comentarios:

Guardar **Cancelar**

Ilustración 57: Sistema de gestión de incidencias – Cerrar incidencia

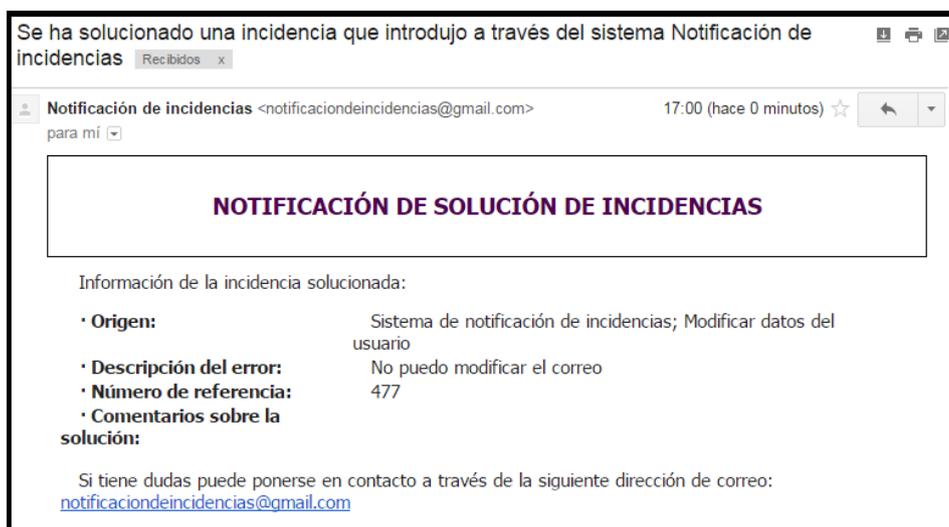


Ilustración 58: Sistema de gestión de incidencias - Envío de correo con incidencia solucionada al usuario

Consulta Incidencias solucionadas por el usuario conectado: [\[Ilustración 59\]](#)

Si en la consulta explicada anteriormente [\[Ilustración 56\]](#) seleccionamos el check de “Incidencias solucionadas” podremos ver las incidencias que haya resuelto el usuario conectado.

En esta consulta además de lo indicado al principio se pueden ver las siguientes columnas:

- **Desarrollador:** es el nombre del desarrollador que se ha encargado de solucionar la incidencia. En este caso en esta columna sólo aparecerá el nombre del usuario conectado.
- **Fecha solución:** es la fecha en la que se solucionó la incidencia.
- **Acción correctora y comentarios equipo desarrollo:** si seleccionamos una incidencia veremos que se desplegará y aparecerá además de la descripción del error, la acción correctora que ha realizado el desarrollador de la incidencia para solucionarla y unos comentarios si el desarrollador lo ve oportuno.

| Prioridad | Estado | Num. referencia | Origen | Fecha error | Fecha solución | Responsable |
|---|-------------|-----------------|--|-------------|----------------|---------------------|
| Crítica | Solucionada | 44 | GestionIncidencias, ConsultaSinAsignar.xaml.cs, Show | 30/05/2015 | 31/05/2015 | Laura Diaz Mendiola |
| <i>Descripción error:</i> No carga los datos al abrir la consulta de errores | | | | | | |
| <i>Acción correctora:</i> | | | | | | |
| <i>Comentarios equipo desarrollo:</i> | | | | | | |
| Baja | Solucionada | 67 | GestionIncidencias, SolucionarIncidencia.xaml.cs, WS_EnvioCorreoSolucionUsuarioCompleted | 31/05/2015 | 31/05/2015 | Laura Diaz Mendiola |
| Baja | Solucionada | 68 | GestionIncidencias, Utiles.xaml.cs, WS_EnvioCorreoDesarrolladorCompleted | 31/05/2015 | 31/05/2015 | Laura Diaz Mendiola |
| Baja | Solucionada | 69 | GestionIncidencias, Utiles.xaml.cs, WS_EnvioCorreoDesarrolladorCompleted | 31/05/2015 | 31/05/2015 | Laura Diaz Mendiola |
| Crítica | Solucionada | 134 | NotificacionIncidencias, ModificarDatosUsuario.xaml.cs, ModificarDatosUsuario_Loaded | 28/06/2015 | 29/06/2015 | Laura Diaz Mendiola |
| Crítica | Solucionada | 232 | NotificacionIncidencias, ModificarDatosUsuario.xaml.cs, ModificarDatosUsuario_Loaded | 28/06/2015 | 29/06/2015 | Laura Diaz Mendiola |

Total: 6

Ilustración 59: Sistema de gestión de incidencias - Consulta Incidencias solucionadas por el usuario conectado

Consulta Incidencias no solucionadas por todos los desarrolladores: [[Ilustración 60](#)]

Partiendo de nuevo de la consulta que se ha explicado al principio [[Ilustración 56](#)] ahora pulsaremos el icono de “Todos los desarrolladores de mantenimiento” y podremos ver las incidencias que hay sin solucionar en todo el sistema, es decir, las que se encuentran en desarrollo o están sin asignar.

En esta consulta se verán además de las columnas indicadas al principio las siguientes columnas:

- **Desarrollador:** aparecerá el nombre del desarrollador que se está encargando de solucionar la incidencia en caso de que la incidencia ya esté asignada a alguien y se encuentre en desarrollo.
- **Botón para cerrar incidencia:** si alguna de esas incidencias que se encuentran en desarrollo son del usuario conectado verá el botón de cerrar incidencia. En caso de pulsarlo se abrirá la ventana para cerrar la incidencia.



| Prioridad | Estado | Num. referencia | Origen | Fecha error | Responsable | |
|---|---------------|-----------------|---|-------------|---------------------------|---|
| Baja | Sin asignar | 475 | NotificacionIncidencias, MainPage.xaml.cs, MainPage_Loaded | 29/06/2015 | | |
| Crítica | En desarrollo | 332 | NotificacionIncidencias, ModificarDatosUsuario.xaml.cs, ModificarDatosUsuario_Loaded | 28/06/2015 | Laura Diaz Mendiola | + |
| Crítica | En desarrollo | 414 | NotificacionIncidencias, ModificarDatosUsuario.xaml.cs, ModificarDatosUsuario_Loaded | 28/06/2015 | Laura Diaz Mendiola | + |
| Crítica | En desarrollo | 448 | NotificacionIncidencias, ModificarDatosUsuario.xaml.cs, ModificarDatosUsuario_Loaded | 28/06/2015 | Laura Diaz Mendiola | + |
| <i>Descripción error:</i> No puedo modificar el correo | | | | | | |
| Baja | En desarrollo | 47 | NotificacionIncidencias, ConsultaIncidenciasUsuario.xaml.cs, Show | 30/05/2015 | desarrollador2 apellidos2 | |
| Baja | En desarrollo | 70 | GestionIncidencias, Utiles.xaml.cs, WS_EnvioCorreoDesarrolladorCompleted | 31/05/2015 | Laura Diaz Mendiola | + |
| Baja | En desarrollo | 72 | GestionIncidencias, SolucionarIncidencia.xaml.cs, WS_EnvioCorreoSolucionUsuarioCompleted | 31/05/2015 | Laura Diaz Mendiola | + |
| Baja | En desarrollo | 437 | NotificacionIncidencias, ConsultaIncidenciasUsuario.xaml.cs, WS_ObtenerDatosIncidenciasUsuarioCompleted | 28/06/2015 | Laura Diaz Mendiola | + |
| Total: 11 | | | | | | |

Ilustración 60: Sistema de gestión de incidencias - Consulta Incidencias no solucionadas por todos los desarrolladores

Consulta Incidencias solucionadas por todos los desarrolladores: [\[Ilustración 61\]](#)

Por último tendremos la consulta que muestra las incidencias solucionadas por todos los desarrolladores.

En esta consulta además de los datos que hay en las demás se pueden ver las siguientes columnas:

- **Desarrollador:** es el nombre del desarrollador que se ha encargado de cerrar la incidencia. En este caso en esta columna sólo aparecerá el nombre del usuario conectado.
- **Fecha solución:** es la fecha en la que se solucionó la incidencia.
- **Acción correctora y comentarios equipo desarrollo:** si seleccionamos una incidencia veremos que se desplegará y aparecerá además de la descripción del error, la acción correctora que ha realizado el desarrollador de la incidencia para solucionarla y unos comentarios si el desarrollador lo ve oportuno.

| Sistema de gestión de incidencias | | | | | | | Laura Díaz Mendiola |
|---|-------------|-----------------|--|---|----------------|---------------------------|---|
| Incidencias solucionadas (todos los responsables de mantenimiento) | | | | | | | |
| Arrastre una columna aquí para agrupar los datos por ella. | | | Búsqueda rápida: | Incidencias solucionadas: <input checked="" type="checkbox"/> | | |     |
| Prioridad | Estado | Num. referencia | Origen | Fecha error | Fecha solución | Responsable | |
| Crítica | Solucionada | 44 | GestionIncidencias, ConsultaSinAsignar.xaml.cs, Show | 30/05/2015 | 31/05/2015 | Laura Díaz Mendiola | |
| <p><i>Descripción error:</i> No carga los datos al abrir la consulta de errores</p> <p><i>Acción correctora:</i></p> <p><i>Comentarios equipo desarrollo:</i></p> | | | | | | | |
| Crítica | Solucionada | 45 | NotificacionIncidencias, MainPage.xaml.cs, MainPage_Loaded | 30/05/2015 | 31/05/2015 | desarrollador2 apellidos2 | |
| Baja | Solucionada | 48 | NotificacionIncidencias, ConsultaIncidenciasUsuario.xaml.cs, cmbCabecera_MouseEnter | 30/05/2015 | 31/05/2015 | desarrollador1 apellidos1 | |
| Baja | Solucionada | 49 | NotificacionIncidencias, MainPage.xaml.cs, MainPage_Loaded | 30/05/2015 | 31/05/2015 | desarrollador2 apellidos2 | |
| Baja | Solucionada | 50 | NotificacionIncidencias, ConsultaIncidenciasUsuario.xaml.cs, cmbCabecera_MouseEnter | 30/05/2015 | 31/05/2015 | desarrollador1 apellidos1 | |
| Baja | Solucionada | 67 | GestionIncidencias, SolucionarIncidencia.xaml.cs, WS_EnvioCorreoSolucionUsuarioCompleted | 31/05/2015 | 31/05/2015 | Laura Díaz Mendiola | |
| Baja | Solucionada | 68 | GestionIncidencias, Utiles.xaml.cs, WS_EnvioCorreoDesarrolladorCompleted | 31/05/2015 | 31/05/2015 | Laura Díaz Mendiola | |
| Baja | Solucionada | 69 | GestionIncidencias, Utiles.xaml.cs, WS_EnvioCorreoDesarrolladorCompleted | 31/05/2015 | 31/05/2015 | Laura Díaz Mendiola | |
| Total: 10 | | | | | | | |

Ilustración 61: Sistema de gestión de incidencias - Consulta Incidencias solucionadas por todos los desarrolladores

Otras formas de configurar la consulta:

- **Búsqueda rápida:** [\[Ilustración 62\]](#)

Si escribimos en la caja de búsqueda, la consulta que estuviera visible en ese momento se filtrará.

| Prioridad | Estado | Num. referencia | Origen | Fecha error | Fecha solución | Responsable |
|---|-------------|-----------------|--|-------------|----------------|---------------------------|
| Crítica | Solucionada | 44 | GestionIncidencias, ConsultaSinAsignar.xaml.cs, Show | 30/05/2015 | 31/05/2015 | Laura Diaz Mendiola |
| <i>Descripción error:</i> No carga los datos al abrir la consulta de errores <i>Acción correctora:</i> <i>Comentarios equipo desarrollo:</i> | | | | | | |
| Crítica | Solucionada | 45 | NotificacionIncidencias, MainPage.xaml.cs, MainPage_Loaded | 30/05/2015 | 31/05/2015 | desarrollador2 apellidos2 |
| Baja | Solucionada | 48 | NotificacionIncidencias, ConsultaIncidenciasUsuario.xaml.cs, cmbCabecera_MouseEnter | 30/05/2015 | 31/05/2015 | desarrollador1 apellidos1 |
| Baja | Solucionada | 49 | NotificacionIncidencias, MainPage.xaml.cs, MainPage_Loaded | 30/05/2015 | 31/05/2015 | desarrollador2 apellidos2 |
| Baja | Solucionada | 50 | NotificacionIncidencias, ConsultaIncidenciasUsuario.xaml.cs, cmbCabecera_MouseEnter | 30/05/2015 | 31/05/2015 | desarrollador1 apellidos1 |
| Baja | Solucionada | 67 | GestionIncidencias, SolucionarIncidencia.xaml.cs, WS_EnvioCorreoSolucionUsuarioCompleted | 31/05/2015 | 31/05/2015 | Laura Diaz Mendiola |
| Baja | Solucionada | 68 | GestionIncidencias, Utiles.xaml.cs, WS_EnvioCorreoDesarrolladorCompleted | 31/05/2015 | 31/05/2015 | Laura Diaz Mendiola |
| Baja | Solucionada | 69 | GestionIncidencias, Utiles.xaml.cs, WS_EnvioCorreoDesarrolladorCompleted | 31/05/2015 | 31/05/2015 | Laura Diaz Mendiola |
| Total: | | 8 | | | | |

Ilustración 62: Sistema de gestión de incidencias - Búsqueda rápida

- Filtrar por columnas: [Ilustración 63]

Seleccionando cualquier columna a la altura del nombre de la cabecera y arrastrándola hasta la parte gris se podrán agrupar las columnas y filtrar los datos.

The screenshot shows the 'Sistema de gestión de incidencias' interface. At the top, it displays the user name 'Laura Díaz Mendiola' and the title 'Incidencias solucionadas (todos los responsables de mantenimiento)'. Below the title, there is a search bar and a filter for 'Incidencias solucionadas' which is checked. The main table has columns for 'Prioridad', 'Estado', 'Num. referencia', 'Origen', 'Fecha error', 'Fecha solución', and 'Responsable'. The 'Prioridad' column is currently filtered to show 'Crítica' incidents. The table contains four rows of data for critical incidents, all of which are 'Solucionada'.

| Prioridad | Estado | Num. referencia | Origen | Fecha error | Fecha solución | Responsable |
|-----------|-------------|-----------------|--|-------------|----------------|---------------------------|
| Crítica | Solucionada | 44 | GestionIncidencias, ConsultaSinAsignar.xaml.cs, Show | 30/05/2015 | 31/05/2015 | Laura Diaz Mendiola |
| Crítica | Solucionada | 45 | NotificacionIncidencias, MainPage.xaml.cs, MainPage_Loaded | 30/05/2015 | 31/05/2015 | desarrollador2 apellidos2 |
| Crítica | Solucionada | 134 | NotificacionIncidencias, ModificarDatosUsuario.xaml.cs, ModificarDatosUsuario_Loaded | 28/06/2015 | 29/06/2015 | Laura Diaz Mendiola |
| Crítica | Solucionada | 232 | NotificacionIncidencias, ModificarDatosUsuario.xaml.cs, ModificarDatosUsuario_Loaded | 28/06/2015 | 29/06/2015 | Laura Diaz Mendiola |

Total: 10

Ilustración 63: Sistema de gestión de incidencias - Filtrar por columnas

Exportar datos a Excel: [\[Ilustración 64\]](#)

Desde cualquiera de las combinaciones que podemos realizar podremos pulsar al icono de “Exportar datos”. Se puede seleccionar entre exportar los datos que están visibles en ese momento en la consulta o todos.

Los datos se exportarán en un fichero Excel. [\[Ilustración 65\]](#)

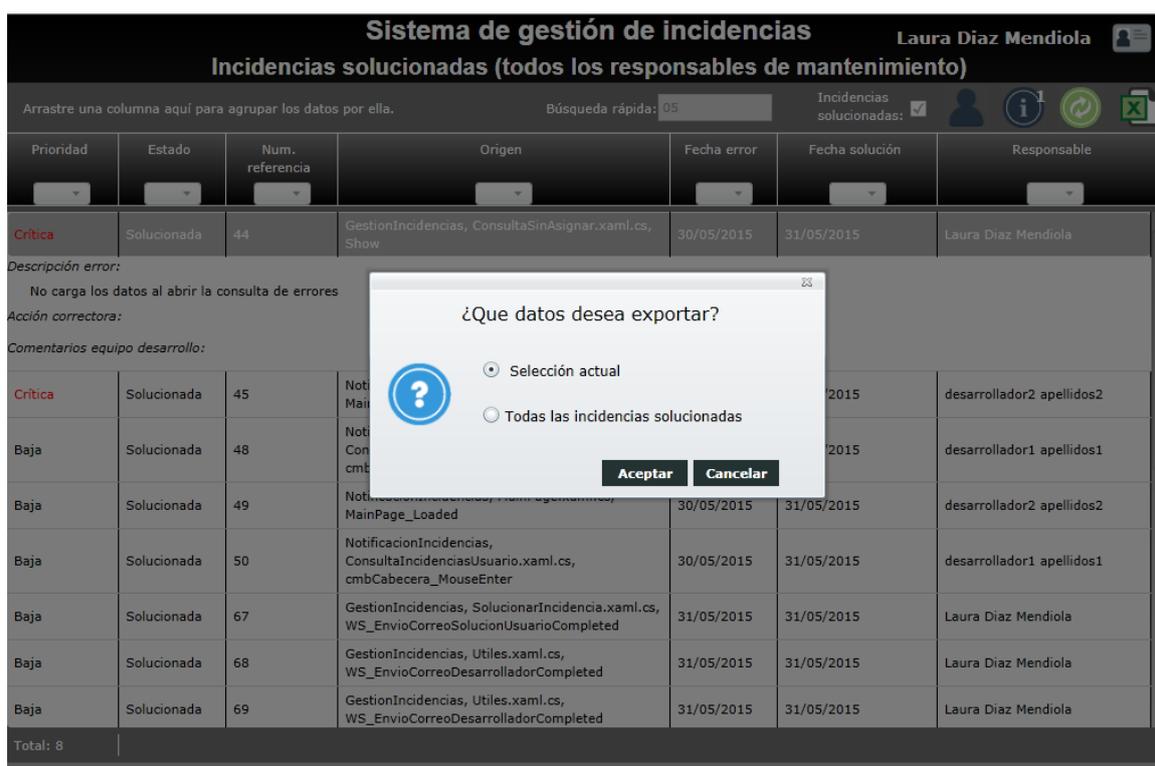


Ilustración 64: Sistema de gestión de incidencias - Exportar datos a Excel

| | A | B | C | D | E | F | G |
|----|-----------|-------------|-----------------|---|-------------|----------------|---------------------------|
| 1 | Prioridad | Estado | Num. referencia | Origen | Fecha error | Fecha solución | Responsable |
| 2 | Crítica | Solucionada | 44 | GestionIncidencias, ConsultaSinAsignar.xa | 30/05/2015 | 31/05/2015 | Laura Diaz Mendiola |
| 3 | Crítica | Solucionada | 45 | NotificacionIncidencias, MainPage.xaml.cs, | 30/05/2015 | 31/05/2015 | desarrollador2 apellidos2 |
| 4 | Baja | Solucionada | 48 | NotificacionIncidencias, ConsultaIncidencia | 30/05/2015 | 31/05/2015 | desarrollador1 apellidos1 |
| 5 | Baja | Solucionada | 49 | NotificacionIncidencias, MainPage.xaml.cs, | 30/05/2015 | 31/05/2015 | desarrollador2 apellidos2 |
| 6 | Baja | Solucionada | 50 | NotificacionIncidencias, ConsultaIncidencia.x | 30/05/2015 | 31/05/2015 | desarrollador1 apellidos1 |
| 7 | Baja | Solucionada | 67 | GestionIncidencias, SolucionarIncidencia.x | 31/05/2015 | 31/05/2015 | Laura Diaz Mendiola |
| 8 | Baja | Solucionada | 68 | GestionIncidencias, Utiles.xaml.cs, WS_Env | 31/05/2015 | 31/05/2015 | Laura Diaz Mendiola |
| 9 | Baja | Solucionada | 69 | GestionIncidencias, Utiles.xaml.cs, WS_Env | 31/05/2015 | 31/05/2015 | Laura Diaz Mendiola |
| 10 | | | | | | | |
| 11 | | | | | | | |

Ilustración 65: Sistema de gestión de incidencias – Datos exportados

Incidencias sin asignar: [\[Ilustración 66\]](#)

Y por último tenemos el botón “Incidencias sin asignar” que puede que aparezca con un número arriba. Este número indicará las incidencias que le corresponden al usuario conectado y aun no las ha seleccionado.

Si lo pulsamos iremos a una consulta similar a la general pero con algo menos de información.

Podemos ver el **número de referencia**, la **prioridad**, el **origen** del error, la **fecha del error**, la **descripción del error** y una columna con un botón para asignar la incidencia.

Si pulsamos en el icono verde significa que queremos hacernos cargo de esa incidencia, nos saldrá un mensaje de confirmación y cuando lo aceptemos esa incidencia se quitará de esta lista y se añadirá a la consulta general de incidencias en desarrollo y aparecerá con el desarrollador ya asignado. [\[Ilustración 56\]](#)

| Prioridad | Estado | Num. referencia | Origen | Fecha error | Responsable |
|-----------|--------|-----------------|-------------------------|-------------|-------------|
| Critica | En d | | NotificacionIncidencias | | |
| Critica | En d | | | | ndiola |
| Critica | En d | | | | ndiola |
| Critica | En d | | | | ndiola |
| Critica | En d | | | | ndiola |
| Baja | En d | | | | ndiola |
| Baja | En d | | | | ndiola |
| Baja | En d | | | | ndiola |
| Baja | En d | | | | ndiola |
| Baja | En d | | | | ndiola |
| Baja | En d | | | | ndiola |
| Baja | En d | | | | ndiola |
| Total: | | 10 | | | |

| Prioridad | Num. referencia | Origen | Fecha error | Asignar |
|-----------|-----------------|--|-------------|---------|
| Baja | 475 | NotificacionIncidencias, MainPage.xaml.cs, MainPage_Loaded | 29/06/2015 | ✓ |

Descripción error:
ERROR: Error al recuperar rutaWSNotificacionIncidencias

Total: 1

Ilustración 66: Sistema de gestión de incidencias - Incidencias sin asignar

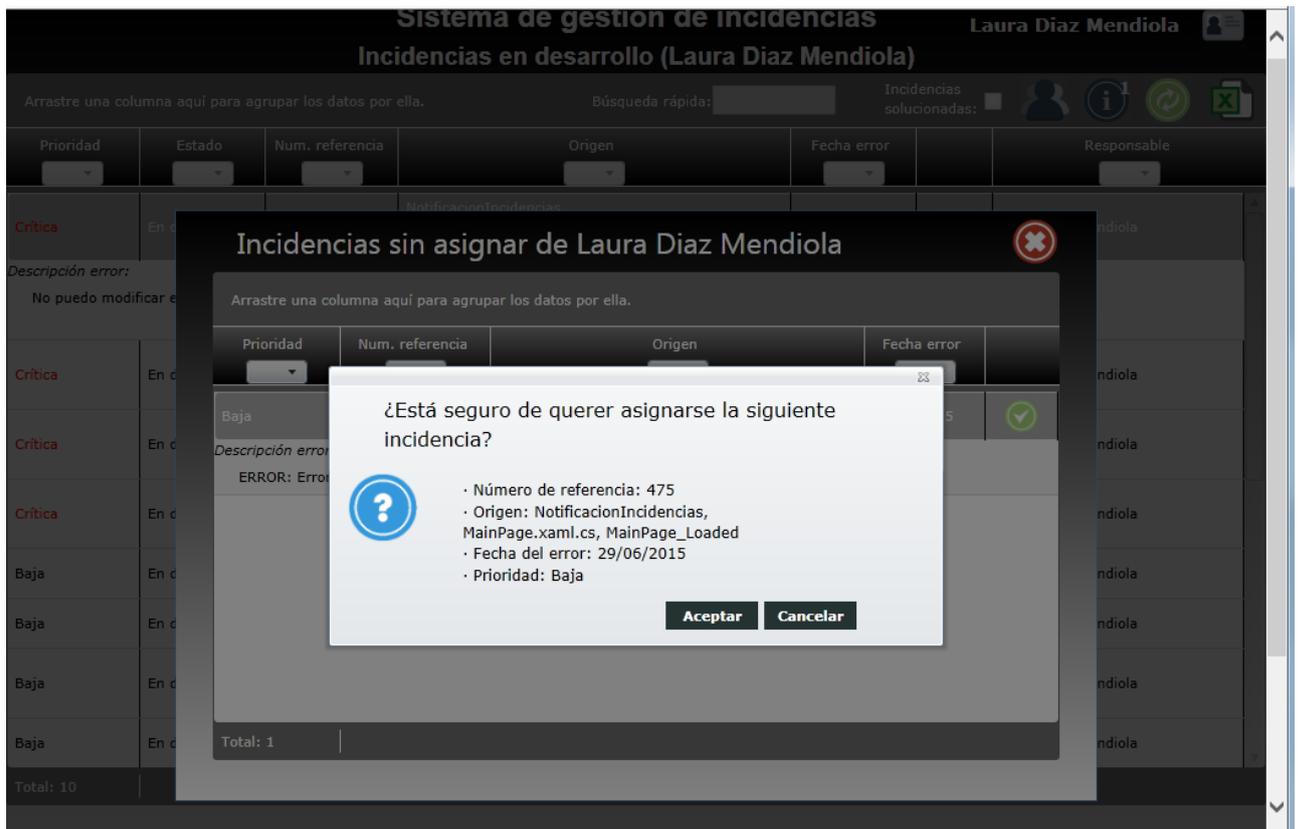


Ilustración 67: Sistema de gestión de incidencias – Asignar incidencia

ANEXO C: Instalación y configuración del IIS

Para poder ejecutar las aplicaciones en el equipo en un servidor local se ha tenido que instalar el servidor web IIS. A continuación se explica el proceso a seguir tanto para su instalación y posterior configuración.

Para la instalación del IIS tendremos que ir al “Panel de Control”, “Desinstalar un programa” y seleccionaremos “Activar o desactivar las características de Windows” como se indica en la siguiente imagen. [\[Ilustración 68\]](#) [\[3\]](#)

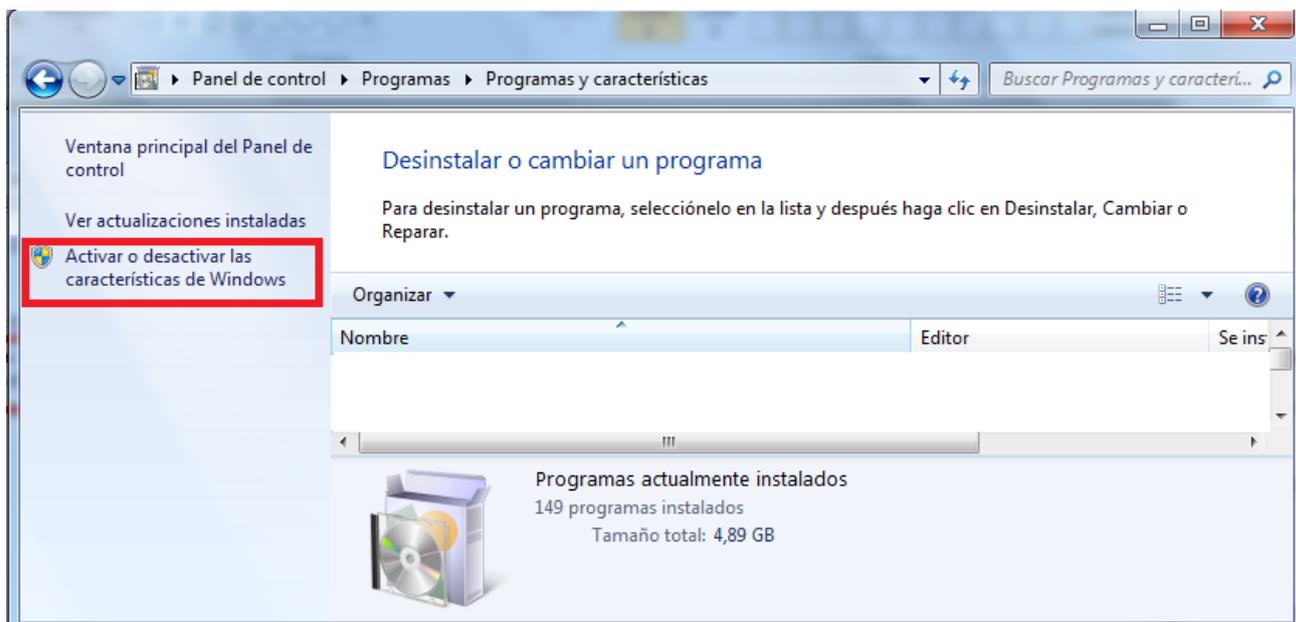


Ilustración 68: Instalación y configuración del IIS

Se abrirá una ventana y seleccionaremos todas las opciones que están dentro de la carpeta Internet Information Services. [\[Ilustración 69\]](#)

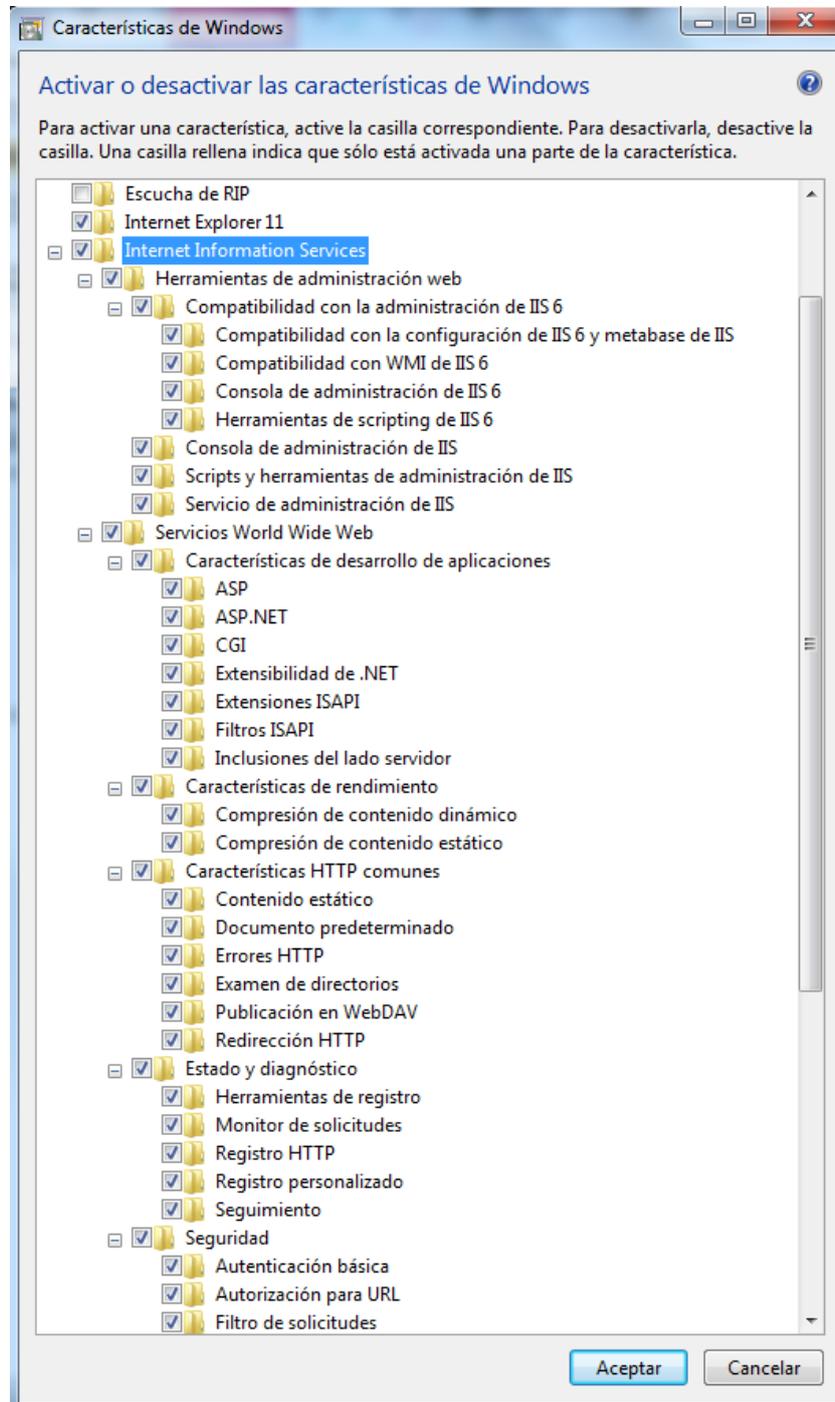


Ilustración 69: Instalación y configuración del IIS

ANEXO D: Publicación de las aplicaciones en el IIS

Una vez instalado y configurado el IIS [[Anexo C](#)] se procederá a publicar las aplicaciones. [[3](#)]

Para publicar las aplicaciones en el IIS tenemos que tener abierto el proyecto que queremos publicar. Una vez que estamos dentro del proyecto pulsaremos en la parte web de este con el botón derecho del ratón y seleccionaremos “Publish” [[Ilustración 70](#)], a continuación saldrá una ventana en la que pondremos lo que se indica en la ilustración adjunta [[Ilustración 71](#)] y daremos a “Publish”. La ruta de “Target Location” será una carpeta de nuestro equipo donde se publiquen las aplicaciones.

Para poder publicar sin problemas se ha tenido que ejecutar Visual Studio como administrador [[Ilustración 72](#)], ya que si no al publicar me salía el error indicado en la siguiente ilustración. [[Ilustración 73](#)]

Por tanto se ejecuta Visual Studio como administrador, se abre el proyecto y se publica como se ha indicado. [[Ilustración 70](#) e [Ilustración 71](#)]

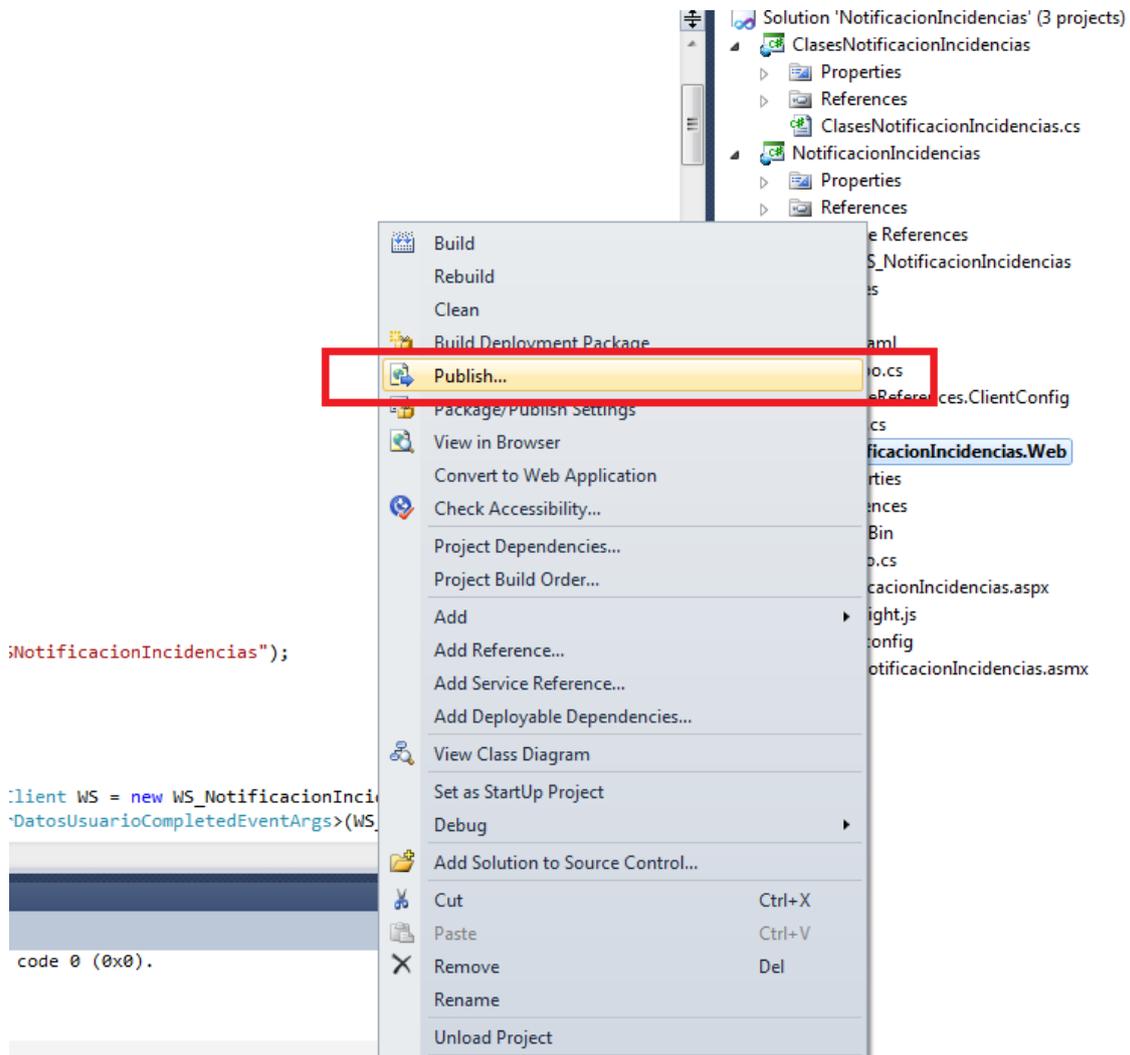


Ilustración 70: Publicación de las aplicaciones en el IIS

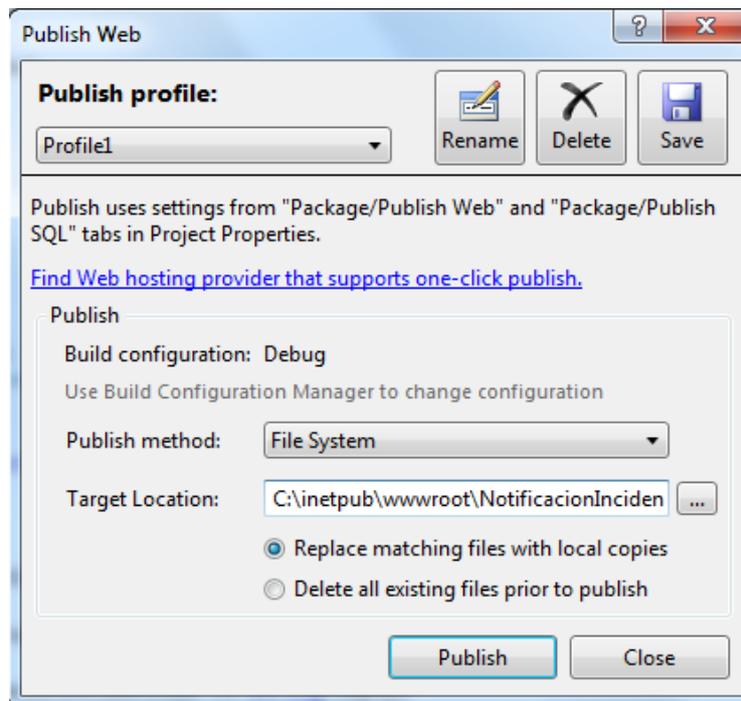


Ilustración 71: Publicación de las aplicaciones en el IIS

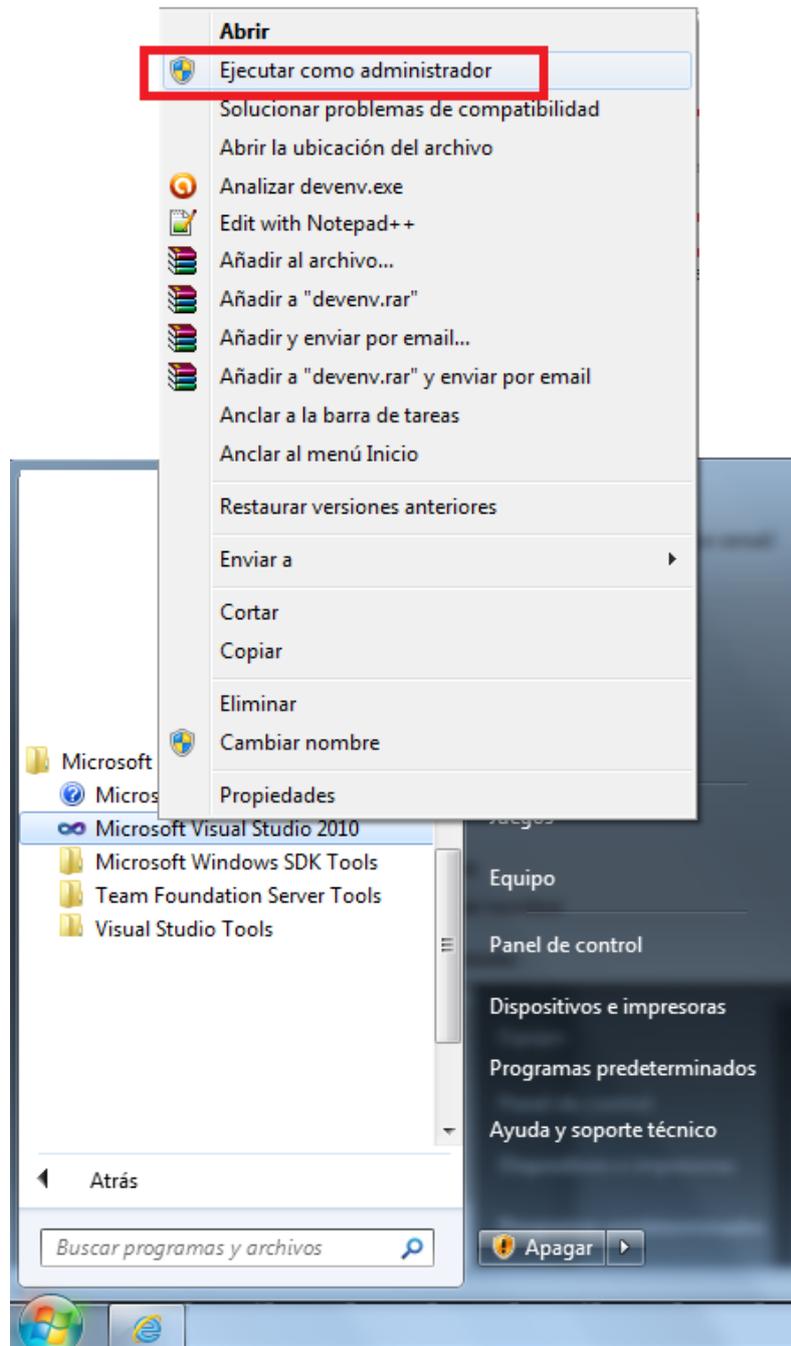


Ilustración 72: Publicación de las aplicaciones en el IIS

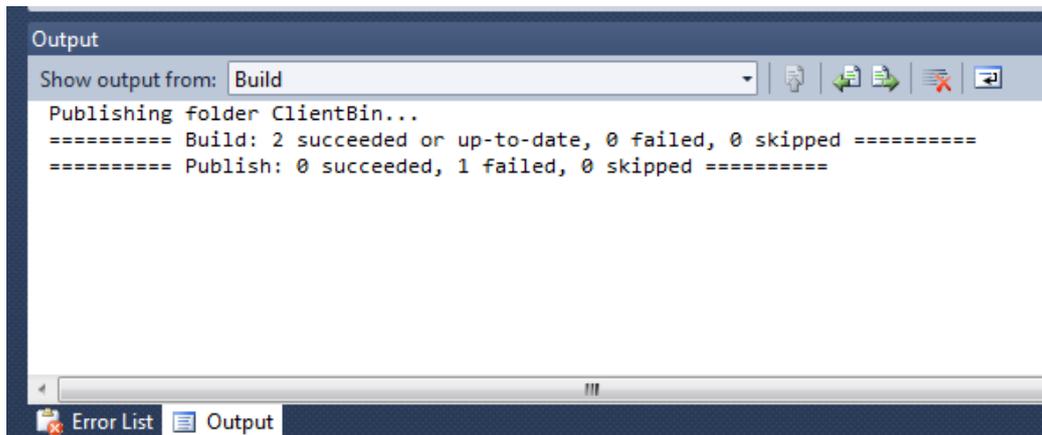


Ilustración 73: Publicación de las aplicaciones en el IIS

ANEXO E: Configuración de las aplicaciones en el IIS

Si se ha instalado correctamente el IIS en los pasos anteriores [[Anexo C](#)], procederemos a abrirlo. [[Ilustración 74](#)]

Para ello iremos al menú Inicio de Windows, pondremos en el buscador “iis” y abrimos “Administrador de Internet Information Services (IIS)”.

Se puede comprobar que ya aparecen los dos proyectos. [[Ilustración 75](#)]

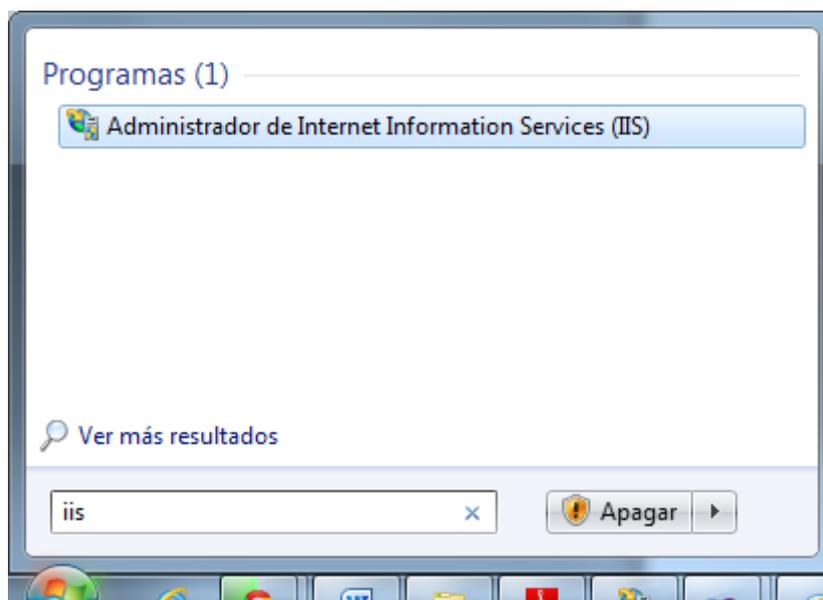


Ilustración 74: Configuración de las aplicaciones en el IIS

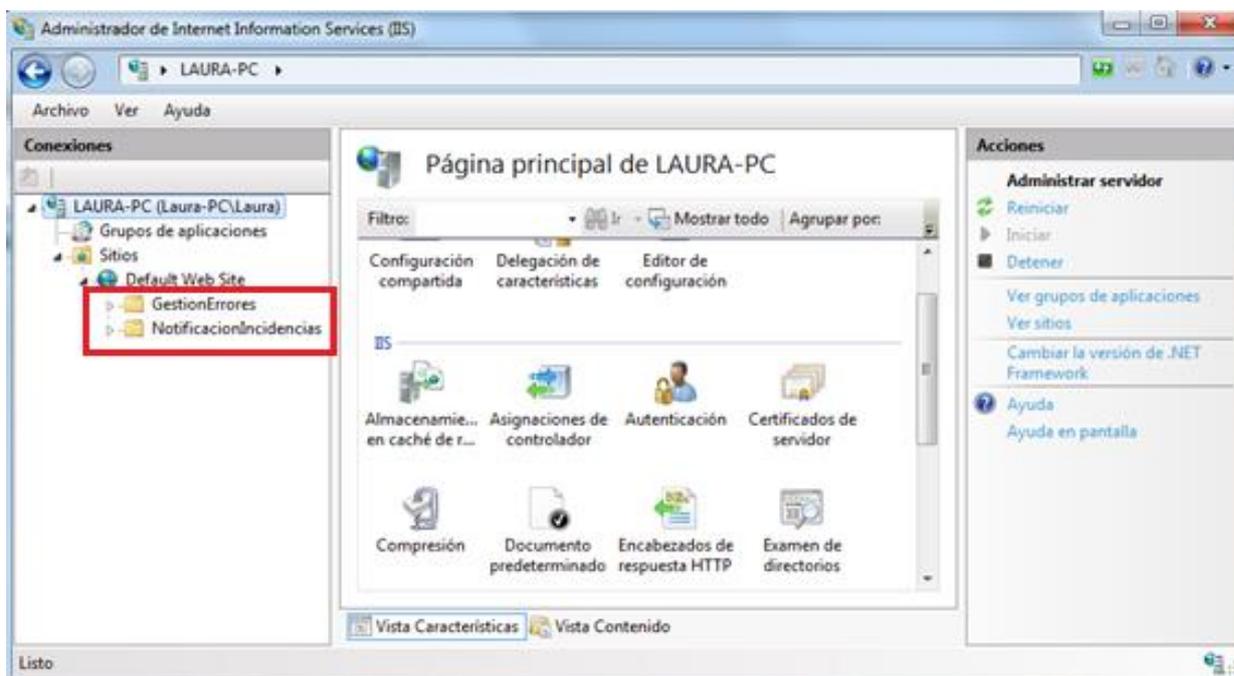


Ilustración 75: Configuración de las aplicaciones en el IIS

Una vez que hemos comprobado que los proyectos están publicados en el IIS, convertiremos cada uno en aplicación. [[Ilustración 76](#)]

Para ello seleccionamos la carpeta del proyecto, botón derecho del ratón y “Convertir en aplicación”.

Saldrá una ventana [[Ilustración 77](#)] en la que las opciones se tienen que quedar como las de la imagen y damos a “Aceptar”.

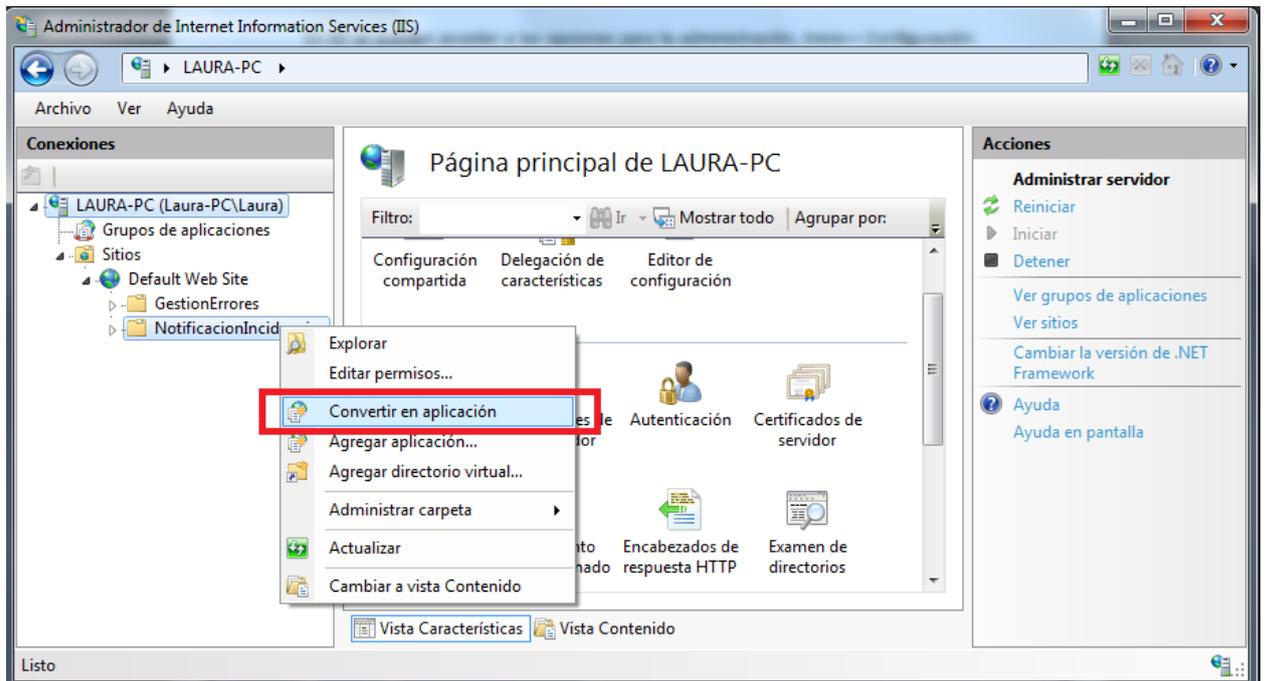


Ilustración 76: Configuración de las aplicaciones en el IIS

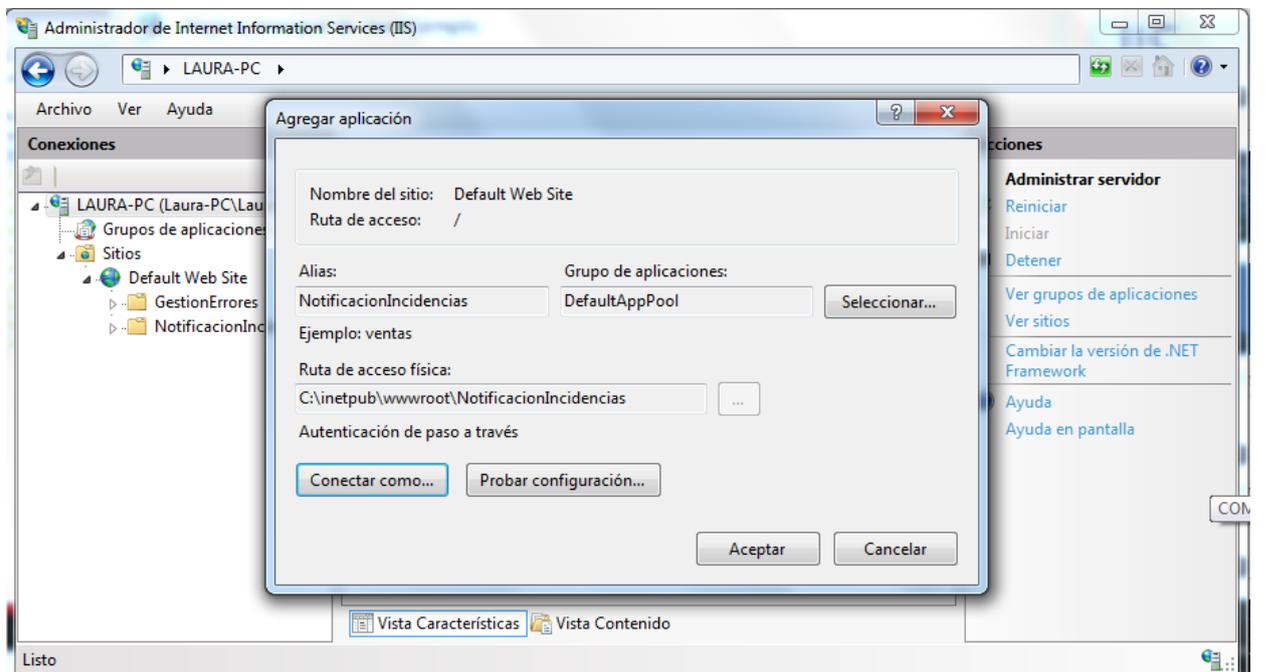


Ilustración 77: Configuración de las aplicaciones en el IIS

A continuación procederemos a cambiar el Framework. [Ilustración 78]
Teniendo seleccionado el nodo raíz del IIS pulsaremos la opción “Cambiar la versión de .NET Framework” y en la ventana que salga seleccionaremos el Framework 4.

Ahora seleccionamos “Grupos de aplicaciones”, “Configuración avanzada” y aquí también seleccionaremos el Framework 4. [Ilustración 79]

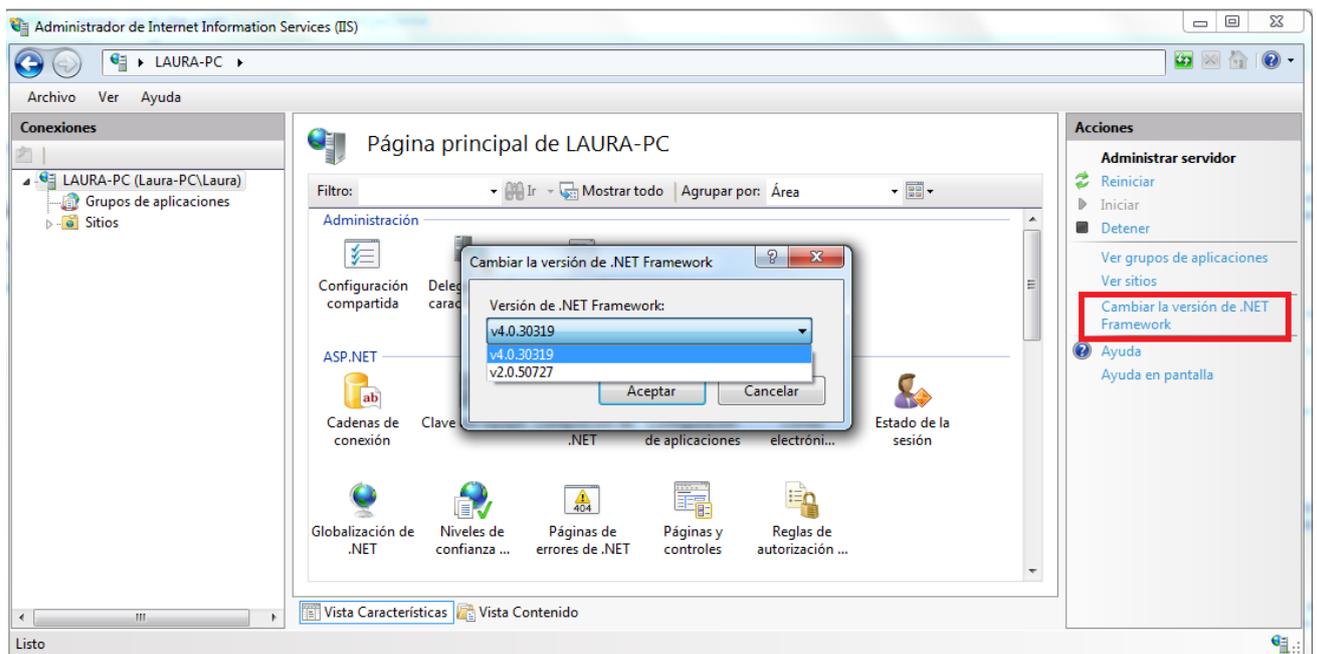


Ilustración 78: Configuración de las aplicaciones en el IIS

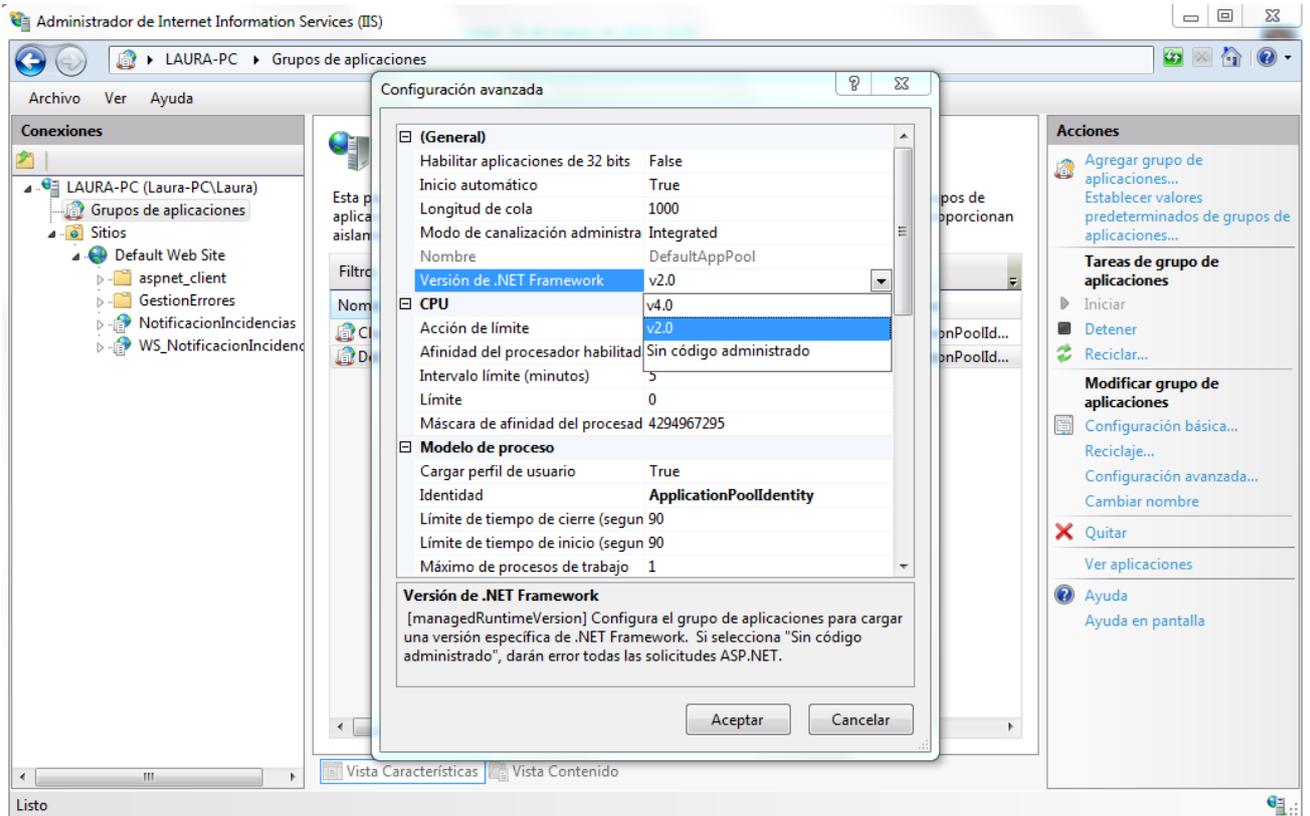


Ilustración 79: Configuración de las aplicaciones en el IIS

En esta misma ventana de configuración avanzada cambiaremos también la propiedad que tiene en Identidad, la cambiaremos por “NetworkService” como se indica en la imagen. [Ilustración 80]

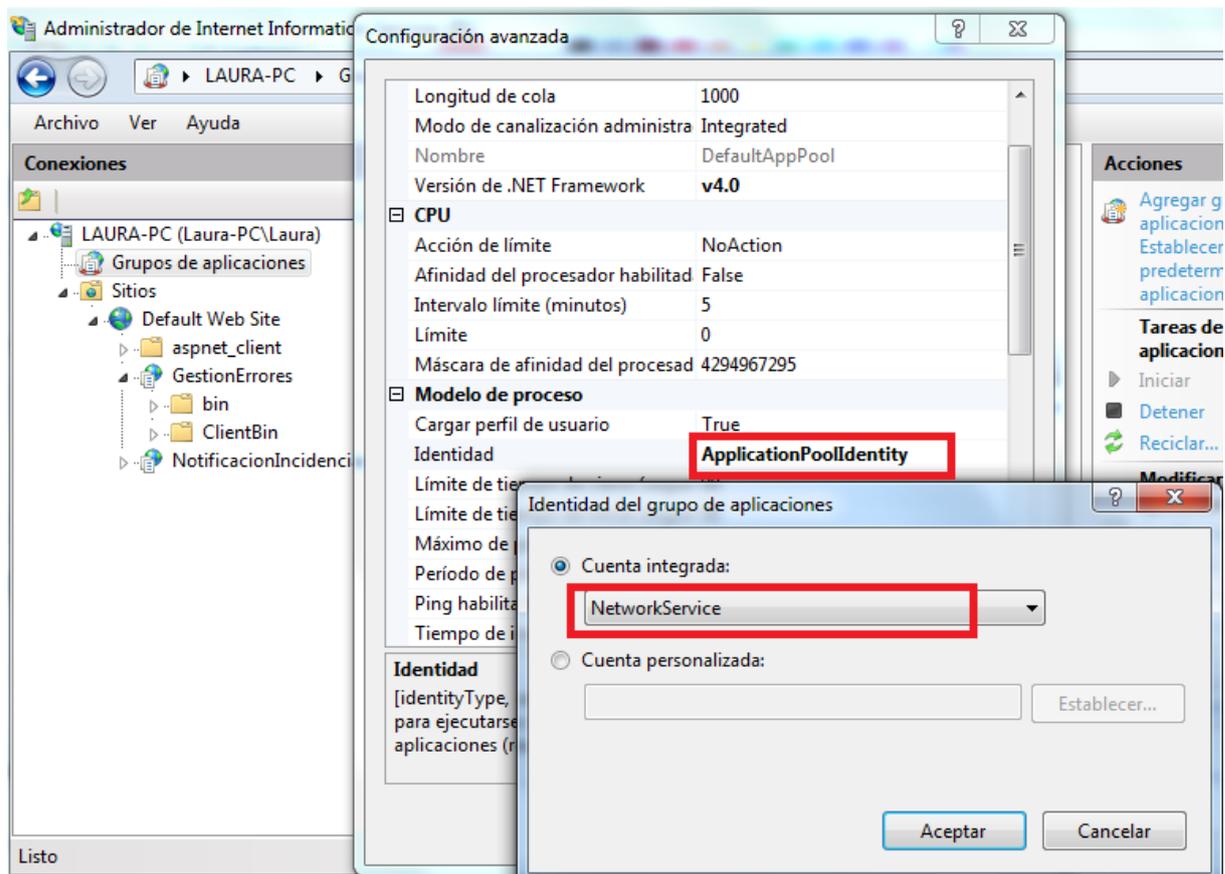


Ilustración 80: Configuración de las aplicaciones en el IIS

Si según están configuradas las aplicaciones en el IIS se le da a examinar saldrá un error. [Ilustración 82]

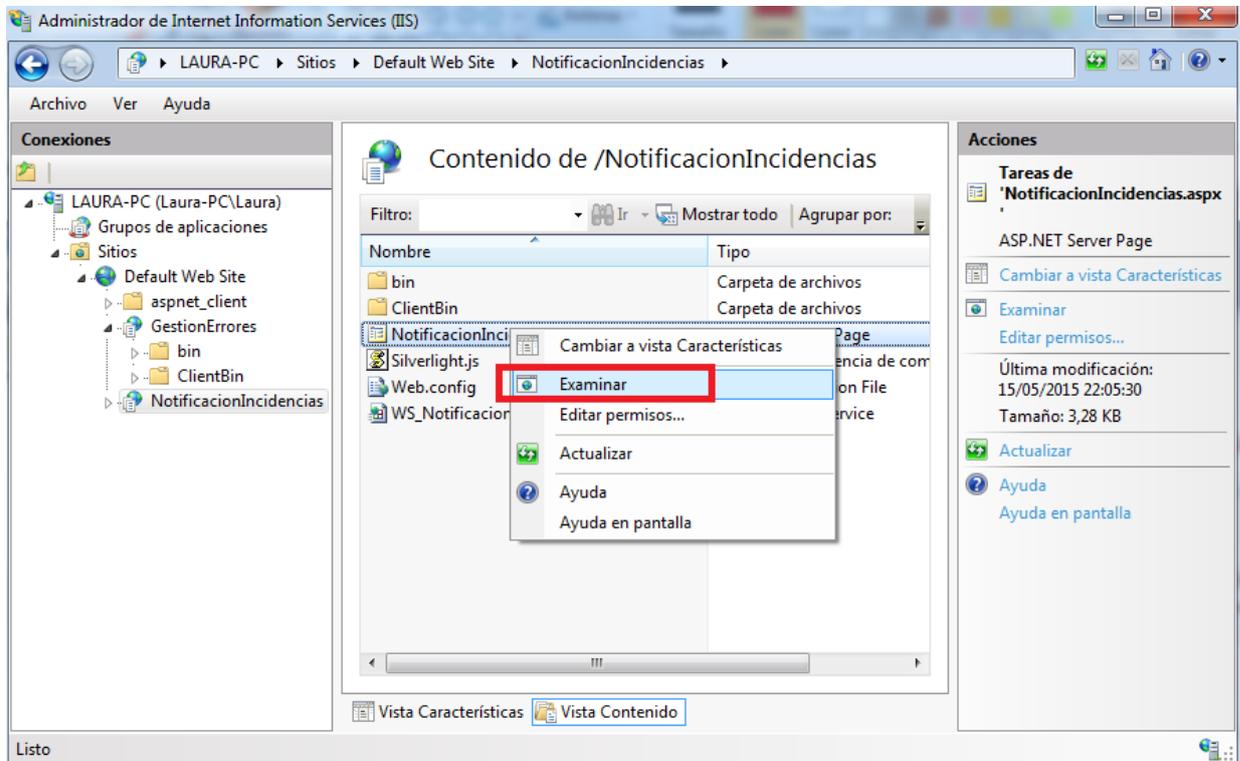


Ilustración 81: Configuración de las aplicaciones en el IIS

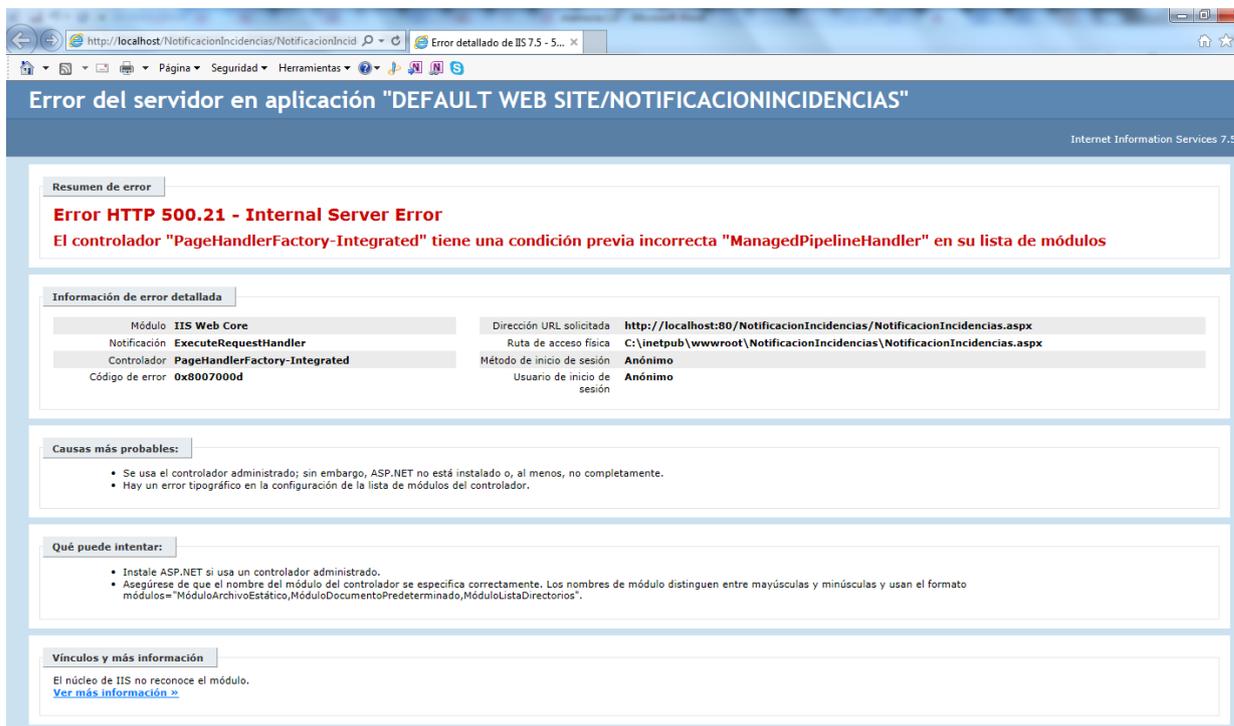


Ilustración 82: Configuración de las aplicaciones en el IIS

Para solucionarlo se ejecuta la consola de comandos como administrador y se escribe lo que aparece en la [Ilustración 84](#)

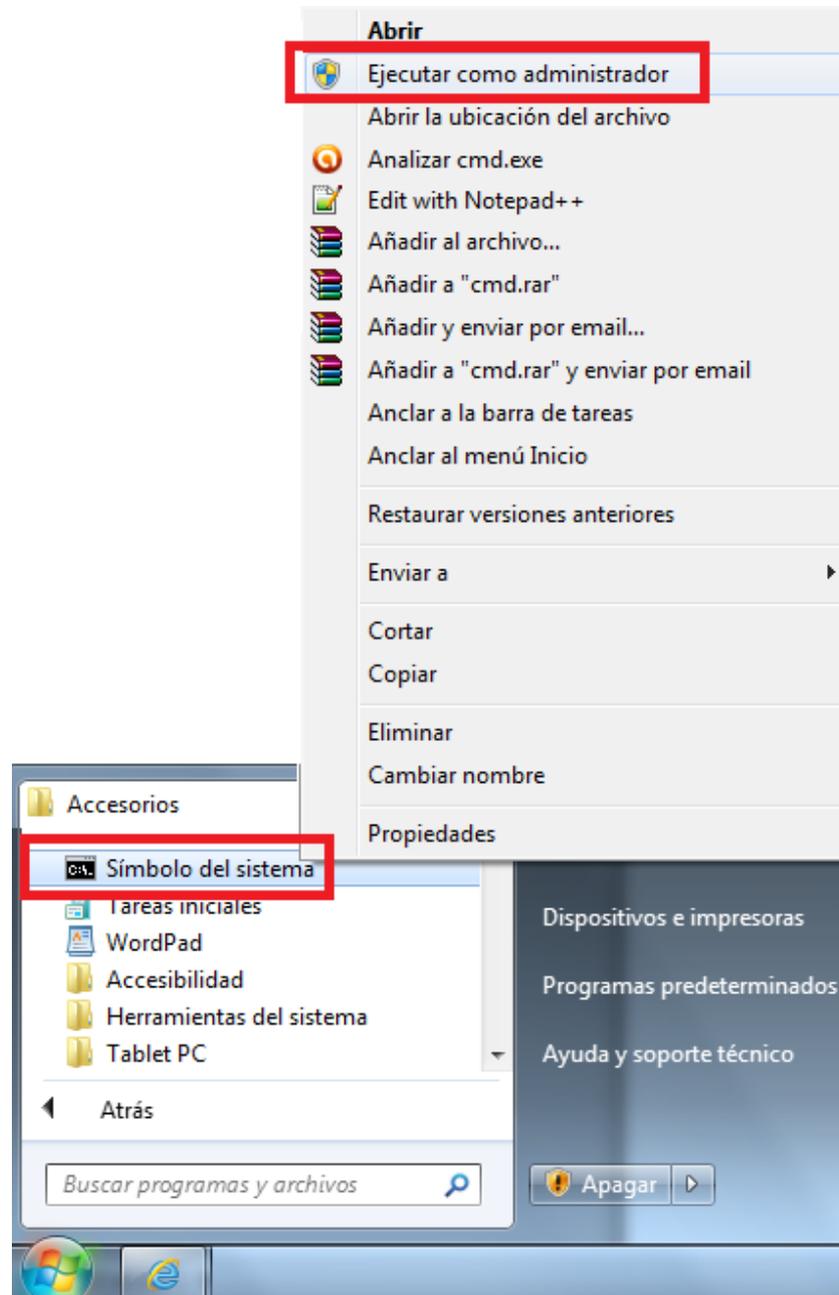


Ilustración 83: Configuración de las aplicaciones en el IIS

```
Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Windows\system32>cd C:\Windows\Microsoft.NET\Framework64\v4.0.30319

C:\Windows\Microsoft.NET\Framework64\v4.0.30319>aspnet_regiis.exe -i
Inicie la instalación de ASP.NET (4.0.30319).
Finalizó la instalación de ASP.NET (4.0.30319).

C:\Windows\Microsoft.NET\Framework64\v4.0.30319>
```

Ilustración 84: Configuración de las aplicaciones en el IIS

Al dar a examinar la aplicación aparecerá un nuevo error que para solucionarlo modificaremos lo siguiente: [[Ilustración 86](#) e [Ilustración 87](#)]

Nos situamos sobre nuestro sitio web y dentro de “Vista Características” seleccionaremos “Examen de directorios” y daremos a “Habilitar”.

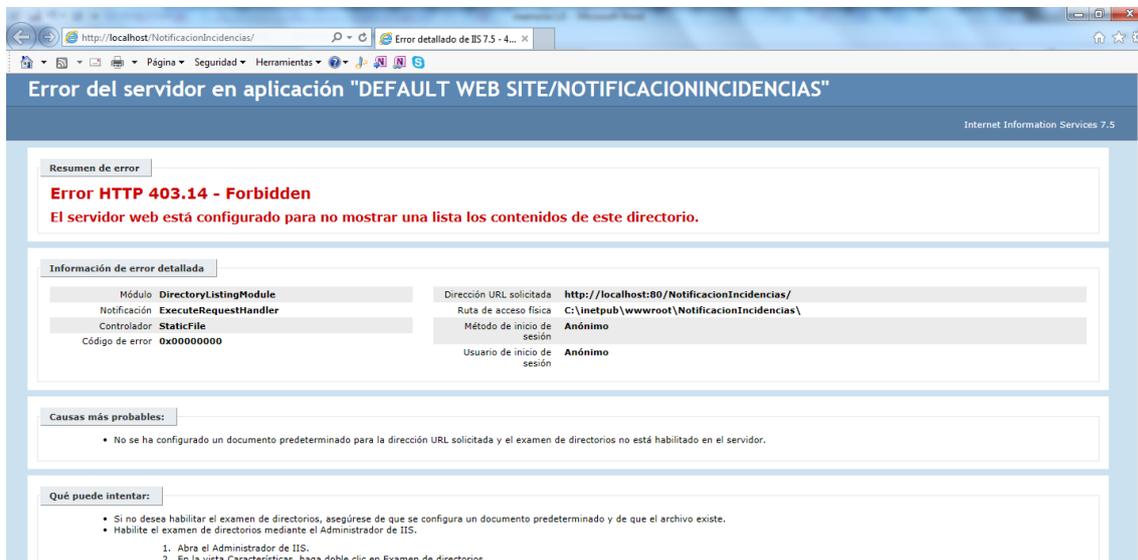


Ilustración 85: Configuración de las aplicaciones en el IIS

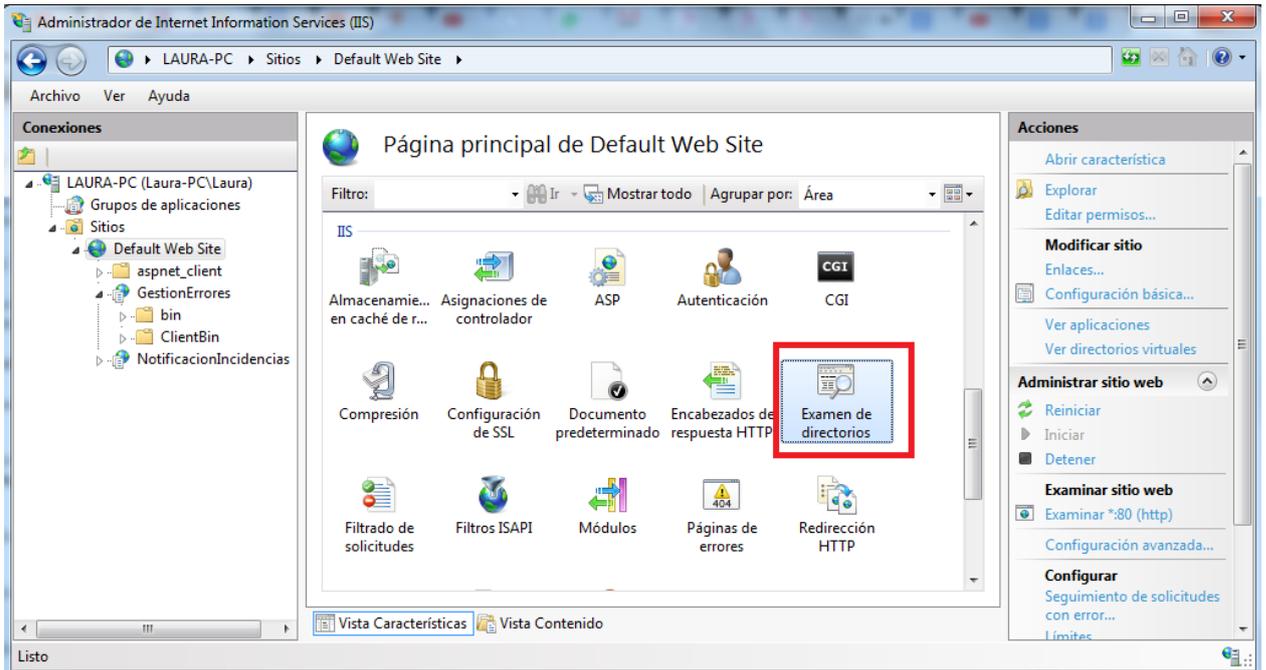


Ilustración 86: Configuración de las aplicaciones en el IIS

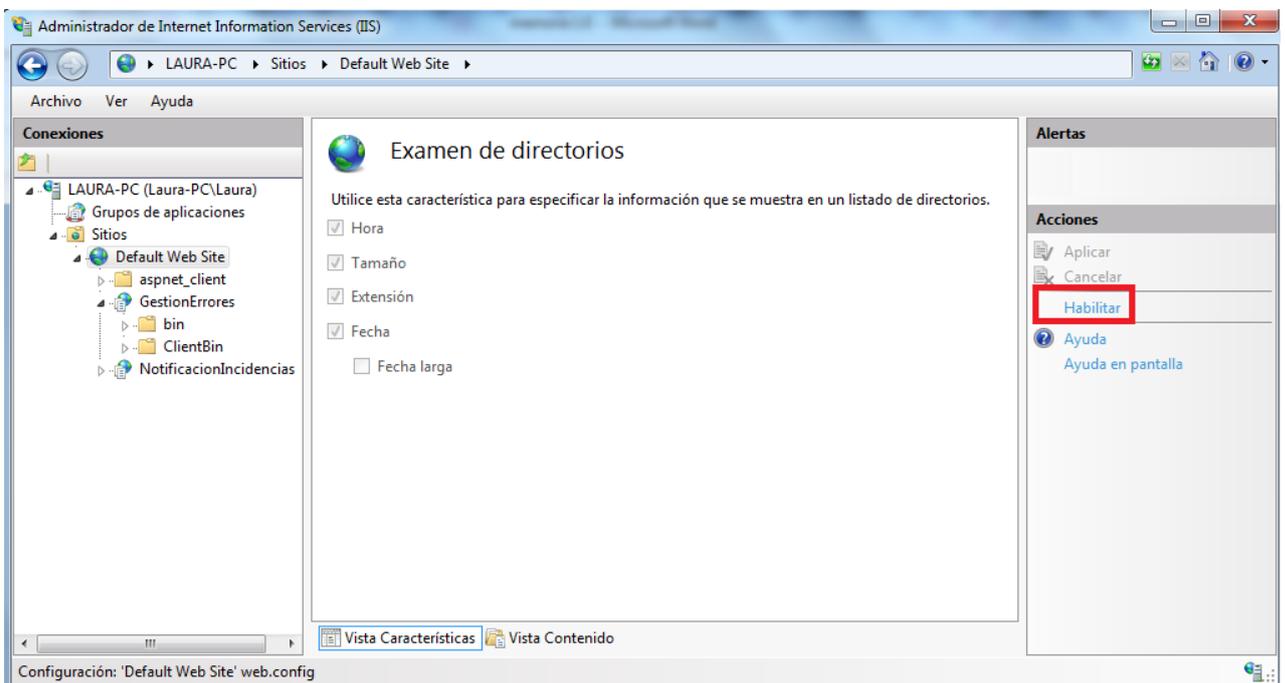


Ilustración 87: Configuración de las aplicaciones en el IIS

Después de todos estos pasos si volvemos a dar a Examinar podremos ver la aplicación correctamente.

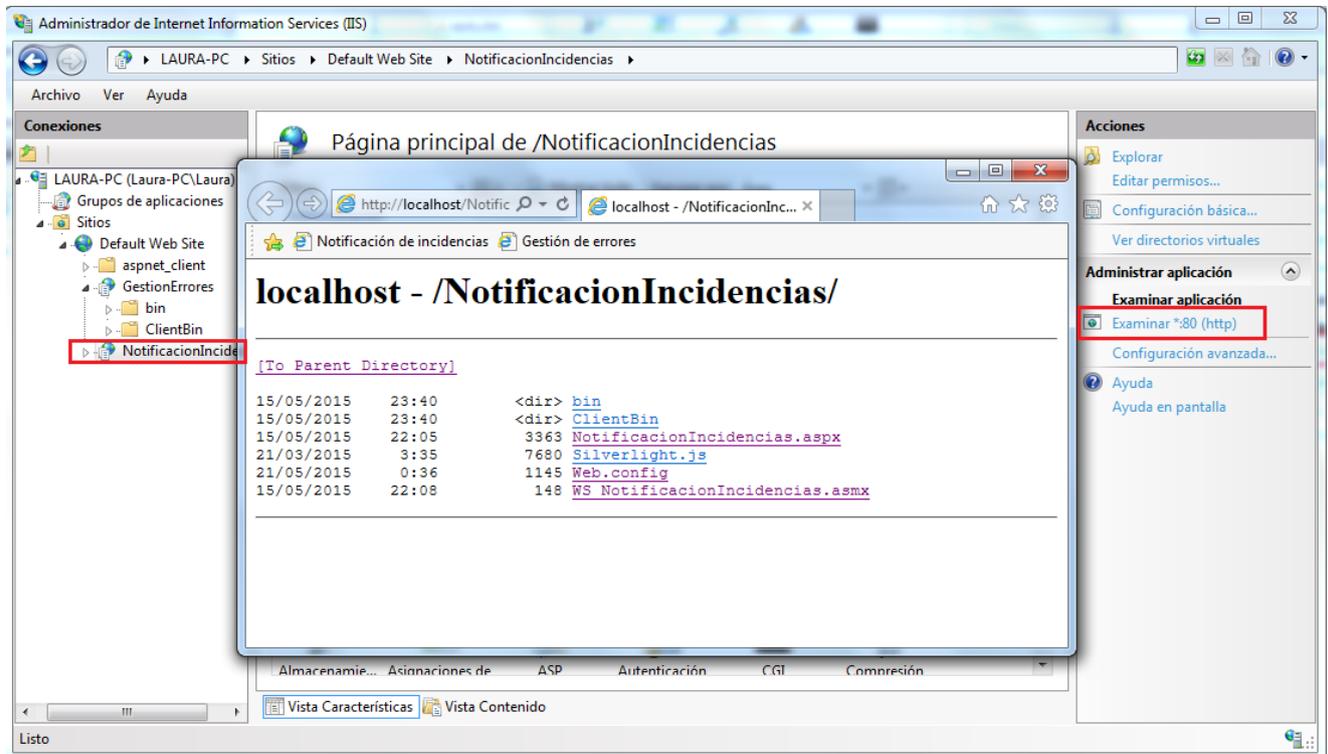


Ilustración 88: Configuración de las aplicaciones en el IIS