Universitat Oberta de Catalunya

Memoria Final

Proyecto Fin de Carrera

Alumna: Patricia Riera Quetglas
Consultor: Angel Acha Lizama
07-01-2013

Índice

Introducción 6
Justificación del proyecto 7
¿Por qué el proyecto? 7
Estudio de mercado 7
Descripción del proyecto 9
Objetivos 10
Objetivos generales 10
Objetivos específicos 10
Resultados esperados 10
Funcionalidades a implementar 11
Productos obtenidos 12
Planificación inicial vs planificación final 13
Análisis y diseño 17
Requerimientos funcionales/ no funcionales 17
Requerimientos funcionales 17
Requerimientos no funcionales 22
Diagrama de casos de uso 23
Sistema de Administración y gestión de incidencias 23
Sistemas de consulta de cliente 26
Diagrama de clases 26
Diagrama de estados 27
Diagrama de flujo 27
Diseño de la base de datos: modelo E-R 29
Arquitectura de la aplicación 29
Hardware 30
Software 31
Diseño de la interfaz de usuario 31
Acceso al panel de administración 31
Sistema de Administración y gestión de incidencias 32
Sistema de consulta de cliente 38

Implementación 46
Software usado 46
Capas de la aplicación 46
Capas de Datos 47
Capa de Entidades 47
Capa de Negocio 48
Capa de Servicios 48
Capa de Presentación 49
Evaluación de costes 50
Trabajo futuro 32
Conclusiones 53
Bibliografía 54

Índice de Figuras

Figura 1 14
Figura 2 15
Figura 3 16
Figura 4 16
Figura 5 23
Figura 6 23
Figura 7 24
Figura 8 24
Figura 9 25
Figura 10 25
Figura 11 26
Figura 12 26
Figura 13 27
Figura 14 28
Figura 15 28
Figura 16 29
Figura 17 30
Figura 18 31
Figura 19 32
Figura 20 33
Figura 21 33
Figura 22 34
Figura 23 34
Figura 24 35
Figura 25 35
Figura 26 36
Figura 27 36
Figura 28 37
Figura 29 37
Figura 30 38

Figura 31 38
Figura 32 39
Figura 33 39
Figura 34 40
Figura 35 40
Figura 36 41
Figura 37 42
Figura 38 42
Figura 39 43
Figura 40 43
Figura 41 44
Figura 42 44
Figura 43 45
Figura 44 48
Figura 45 49
Figura 46 50
Figura 47 51

Introducción

El objetivo de este Proyecto Final de Carrera es la implementación de un sistema de gestión de incidencias.

Primero vamos a definir qué entendemos por gestor de incidencias. Para ello aclaramos antes algunos términos:

"*La Gestión de Incidentes tiene como objetivo resolver cualquier incidente que cause una interrupción en el servicio de la manera más rápida y eficaz posible.*"(Osiatis, ITIL – Gestión de Servicio TI, Visión General)

A su vez, "*la* ***Gestión de Incidentes*** *no debe confundirse con la* ***Gestión de Problemas****, pues a diferencia de esta última, no se preocupa de encontrar y analizar las causas subyacentes a un determinado incidente sino exclusivamente a restaurar el servicio.*"(Osiatis, ITIL – Gestión de Servicio TI, Visión General)

Por otro lado, tenemos que una incidencia se puede definir como "Cualquier evento que no forma parte de la operación estándar de un servicio y que causa, o puede causar, una interrupción o una reducción de calidad del mismo."(Osiatis, ITIL – Gestión de Servicio TI, Visión General)

En resumen, un gestor de incidencias sería aquel encargado de administrar todas las correcciones, errores o caídas de un determinado sistema.

La aplicación desarrollada ofrece las funcionalidades necesarias para llevar a cabo esa gestión de fallos reportados para los diferentes programas que tengamos a cargo.

Para la elaboración de esta aplicación se han seleccionado tecnologías .NET ya que uno de los objetivos del proyecto era el desarrollo de una aplicación donde se empleara *Entity Framework*, *ASP.NET* y *WCF*.

El presente documento es la Memoria del proyecto donde exponemos la descripción del sistema, el análisis y diseño del gestor.

Justificación del proyecto

¿Por qué el proyecto?

Hoy en día los sistemas informáticos son cada vez más complejos y cada vez las empresas dependen más de servicios tecnológicos por lo que un mal funcionamiento o la interrupción de servicios de éstos pueden llegar a tener importantes consecuencias en la consecución de los objetivos de esas empresas.

Es crucial, entonces, que las compañías que brindan estos servicios a terceros sean capaces de solucionar esos errores o fallos en sus servicios y de hacerlo en el menor tiempo posible. Para ello deben tener controlados en todo momento cuáles son los problemas de sus productos y qué personal tienen dedicados a ellos en todo momento.

La calidad que brinden a la hora de solucionar incidencias puede ser un hecho determinante para captar o mantener a sus clientes. De ahí también la importancia de que conozcan la valoración de sus clientes en cuánto a la rapidez y eficacia en la solución de los problemas reportados.

Con todo lo dicho anteriormente se extrae la necesidad de que cualquier negocio que provee de un servicio tecnológico a otros negocios o que desarrolle servicios para uso propio posea este tipo de aplicación.

Además, es muy interesante que estas aplicaciones puedan disponer de una asignación automática de los problemas reportados a los técnicos disponibles en ese momento, valorando la carga de trabajo de cada uno de ellos, con el objetivo de reducir el tiempo de dedicado a la gestión y planificación de las incidencias según recursos.

Teniendo en cuenta todo lo anterior, se pretende implementar un sistema gestor de incidencias para llevar el seguimiento de todos los errores/fallos, capaz de poderse adaptar a los distintos flujo de trabajo de las empresas y que permita a los clientes opinar sobre el servicio ofrecido.

Estudio de mercado

Actualmente, en el mercado tenemos multitud de gestores de incidencias. De entre todos ellos destacaremos los que hemos encontrado más importantes o destacables por algún motivo, ya sea el coste o las particularidades que ofrece.

Entre todas las aplicaciones dedicadas a la gestión de incidencias tenemos tanto soluciones gratuitas como de pago. Las gratuitas pueden parecer en un primer momento las más atractivas, revisemos unas cuantas de estas soluciones y sus especificaciones:

- **BugTracker.NET**: se presenta como una aplicación web donde datos y web están alojados en sus propios servidores. Algunas de sus características más interesantes son: la capacidad de crear informes propios, permitir definir el flujo de trabajo a seguir e importación de emails a "tickets" entre otros. No se ha referencia a que la asignación de incidencias se pueda establecer de forma automática.

- **Mantis Bug Tracker**: aplicación en PHP. También puede ser usada como herramienta para gestión de proyectos. De esta solución se puede destacar que permite configurar la transición de estados de las incidencias y el soporte en varios idiomas. Pero como en el caso anterior parece que la asignación de incidencias es de forma manual o estática dependiendo del proyecto al que se refiera la incidencia.

- **Request Tracker:** de esta aplicación cabe señalar que permite a los usuarios configurar los campos de sus incidencias así como crear incidencias a partir de emails y generar respuestas automáticas.

Entre las soluciones de pago destacamos:

- **JIRA**: al igual que **Matis Bug Tracker** también posee características para ser usado como gestor de proyectos. Es altamente configurable, permite integración con otros sistemas y ofrece estadísticas e informes bastante completos. El principal inconveniente que quizás a primera vista se le puede atribuir al producto es que si lo que pretende tener es simplemente un gestor de incidencias las tareas y opciones que la hacen versátil también como gestor de proyectos pueden dificultar su usabilidad.

- **ServiceDesk Plus**: la funcionalidad de esta solución es muy completa. Posee reglas de negocio para la asignación automática de solicitudes, base de conocimientos, gestión de tareas programadas, integración con correo electrónico,… No es sólo un gestor de incidencias sino también posee características de un gestor de cambios y problemas.

- **I-Solver**: muy completa. Al igual que el anterior programa permite la asignación automática, creación de incidencias por email y creación de incidencias programadas. Además de todo lo anterior, añade la posibilidad de que el cliente valore el grado de satisfacción.

Como se puede ver de los ejemplos expuestos (solo una pequeña parte de todos los existentes en el mercado actual) hay variadas y múltiples soluciones.

Se espera que la aplicación web sea suficientemente completa pero a la vez simple para ser atractiva y que contemple las necesidades mínimas que debe cumplir un gestor de incidencias con asignación automática a los distintos recursos.

Descripción del proyecto

El proyecto consiste en una aplicación web la cual permitirá por parte de un administrador del sistema, más concretamente de un usuario de atención al público (tal como dicta el enunciado de la propuesta del Proyecto Fin de Carrera), la introducción de incidencias. Es decir, el único punto de entrada de incidencias es un usuario de atención al público.

En el sistema nos encontramos con 3 tipos de distintos de usuario: usuarios de atención al público (ya mencionados en el párrafo anterior), usuarios de tipo supervisor y usuarios de tipo técnico.

Una vez introducida la incidencia, con los datos mínimos necesarios, en el sistema ésta será reasignará a un usuario encargado de solucionarla. Esta asignación se podrá hacer de forma automática según unas reglas establecidas y que sólo un administrador, de tipo supervisor, habrá configurado previamente. Se pretende con ello minimizar los tiempos de asignación de forma manual por parte del usuario que introduce la incidencia y que puede desconocer el estado del usuario al que la asigna (por ejemplo, puede estar de vacaciones o su carga de trabajo es muy elevada respecto a lo demás). De no ser asignada automáticamente por el sistema, un supervisor podrá llevar a cabo la asignación de forma manual.

Las incidencias sólo podrán ser asignadas a usuarios de tipo técnico.

El técnico encargado de solucionar la incidencia podrá asignar el tiempo dedicado a su resolución con el fin de llevar un mejor control del coste en tiempo de las incidencias resueltas.

Por otro lado, el cliente podrá consultar el estado de su incidencia en cualquier momento, incluso podrá recibir comunicaciones en su correo o aportar más información a la incidencia si así se le permite en la configuración del sistema. Además, también podrán opinar sobre el servicio recibido.

Por último, los administradores del sistema podrán revisar una serie de informes para conocer que carga de trabajo se tiene; ya sea por cliente, por recurso o por servicio para poder conocer y llevar un mayor control sobre el sistema y así tomar decisiones tácticas adecuadas para mejorar el servicio ofrecido.

Objetivos

Objetivos generales

- Diseñar una aplicación empleando la arquitectura N-Capas, en este caso, en 4 capas: capa de presentación, capa de servicios, capa de negocio y capa de datos.
- Desarrollar una aplicación web haciendo uso de la tecnología .NET, concretamente ASP.NET, WCF y Entity Framework.

Objetivos específicos

- Poder registrar todas las incidencias de los diferentes sistemas ofrecidos a nuestros clientes así como el tiempo invertido en la resolución de las mismas.
- Facilitar el trabajo de la asignación de las incidencias creadas mediante un algoritmo de asignación automática dadas unas reglas configurables.
- Permitir al cliente conocer el estado de su incidencia en cualquier momento y la fecha estimada de resolución así como, opcionalmente, opinar sobre el servicio dado una vez resuelta.
- Poder monitorizar el estado de todas las incidencias del sistema, los tiempos de respuesta, incidencias por cliente, incidencias por recurso…. gracias a una serie de informes ideados con tal propósito

Resultados esperados

Al final del proceso de implementación lo que se pretende ofrecer es una aplicación web que sea lo suficientemente flexible para adaptarse a los diferentes flujos de trabajos de las distintas empresas así como ofrecer una interfaz sencilla y simple donde se pueda revisar en todo momento el estado de los servicios ofrecidos, la satisfacción del cliente y los tiempos empleados para ello y donde el cliente pueda consultar e incluso aportar nueva información de las incidencias ya creadas.

Funcionalidades a implementar

De forma general se expone más abajo la funcionalidad a desarrollar:

- Gestión de usuarios del sistema. Dar de alta, modificar y eliminar los diferentes usuarios del sistema así como sus perfiles.
- Configuración de reglas del sistema. Poder dar de alta diferentes reglas para la asignación automática de las incidencias, los diferentes tipos de incidencias y las distintas opciones ofrecidas a los clientes para interactuar con el sistema: consultar el estado de una incidencia, añadir comentarios, opinar sobre el servicio recibido y recibir en el email una alerta cuando su incidencia ha sido resuelta.
- Gestión de incidencias: Poder dar de alta incidencias, consultarlas, cambiar su estado, insertar las horas dedicadas, añadir comentarios visibles para el cliente,…
- Proceso de reasignación automática de incidencias.
- Gestión de clientes. Poder llevar a cabo una sencilla gestión de clientes dando de alta sus datos básicos así como los programas/servicios que tiene contratados.
- Listado de informes. Se tendrían los siguientes informes: puntuación media obtenida cliente, servicio y/o técnico, número de incidencias por cliente, número de incidencias por técnico, número de incidencias por tipo y/o programa, incidencias fuera de planificación e informe entre tiempo estimado y tiempo real en la resolución.

Productos obtenidos

Los productos finales que se han obtenido son:

- Script SQL para la creación de la base de datos.
- Aplicación web para la administración y gestión de incidencias.
- Aplicación web para la consulta de una incidencia por parte de un cliente.
- Webservice WCF para proveer la funcionalidad de los dos puntos anteriores.
- Memoria final del proyecto.
- Manual de usuario.
- Manual de instalación.
- Presentación virtual.

Planificación inicial vs Planificación final

Los hitos principales a realizar correspondían con las distintas etapas mostradas en el Plan Docente:

- Plan de Trabajo
- Análisis y diseño
- Implementación
- Presentación final

La planificación inicial era la siguiente:

Nombre de tarea	Duración	Comienzo	Fin
Plan de Trabajo	**1,63 días**	**vie 21/09/12**	**lun 01/10/12**
Elección y revisión del proyecto	3 horas	vie 21/09/12	vie 21/09/12
Reunion virtual	1 hora	sáb 29/09/12	sáb 29/09/12
Búsqueda de información y redacción del documento	12 horas	sáb 29/09/12	lun 01/10/12
Entrega PEC1	0 horas	lun 01/10/12	lun 01/10/12
Análisis y diseño	**8,25 días**	**lun 01/10/12**	**dom 28/10/12**
Análisis de los requisitos funcionales	12 horas	lun 01/10/12	dom 07/10/12
Definición conceptual de datos	12 horas	dom 07/10/12	lun 15/10/12
Diseño modelo relacional	6 horas	lun 15/10/12	mar 16/10/12
Casos de Uso	12 horas	mar 16/10/12	jue 18/10/12
Diseño interfaz gráfica	16 horas	vie 19/10/12	jue 25/10/12
Redacción documentación	8 horas	jue 25/10/12	dom 28/10/12
Entrega PEC2	0 horas	dom 28/10/12	dom 28/10/12
Implementación	**14,38 días**	**dom 28/10/12**	**lun 10/12/12**
Creación BDD	4 horas	dom 28/10/12	jue 01/11/12
Implementación capa de datos	8 horas	jue 01/11/12	sáb 03/11/12
Implementación capa de negocio	30 horas	sáb 03/11/12	vie 16/11/12
Implementación capa de servicio	8 horas	vie 16/11/12	dom 18/11/12
Implementación web	**7,5 días**	**dom 18/11/12**	**dom 09/12/12**
Plantillas, estilo y menú	8 horas	dom 18/11/12	vie 23/11/12
Gestión usuarios	12 horas	vie 23/11/12	mar 27/11/12
Gestión reglas y configuración	8 horas	mar 27/11/12	sáb 01/12/12

Gestión de incidencias	8 horas	sáb 01/12/12	mar 04/12/12
Gestión clientes	4 horas	jue 06/12/12	vie 07/12/12
Informes	12 horas	vie 07/12/12	dom 09/12/12
Proceso asignación automática	16 horas	sáb 01/12/12	jue 06/12/12
Documentación	5 horas	dom 09/12/12	lun 10/12/12
Entrega PEC3	0 días	dom 09/12/12	dom 09/12/12
Presentación final	**9,13 días**	**dom 09/12/12**	**lun 07/01/13**
Correcciones y mejoras	30 horas	dom 09/12/12	vie 21/12/12
Realización de la memoria	12 horas	vie 21/12/12	mar 25/12/12
Presentación virtual	30 horas	jue 27/12/12	dom 06/01/13
Entrega final	0 días	lun 07/01/13	lun 07/01/13

Se estimó que se dedicarían por semana unas 20 horas de media, excepto para la tarea de implementación que para poder cumplir el calendario se debían incrementar a 24 horas por semana.

En las siguientes figuras se muestra el Diagrama de Gantt correspondiente a las tareas y actividades comentadas anteriormente:

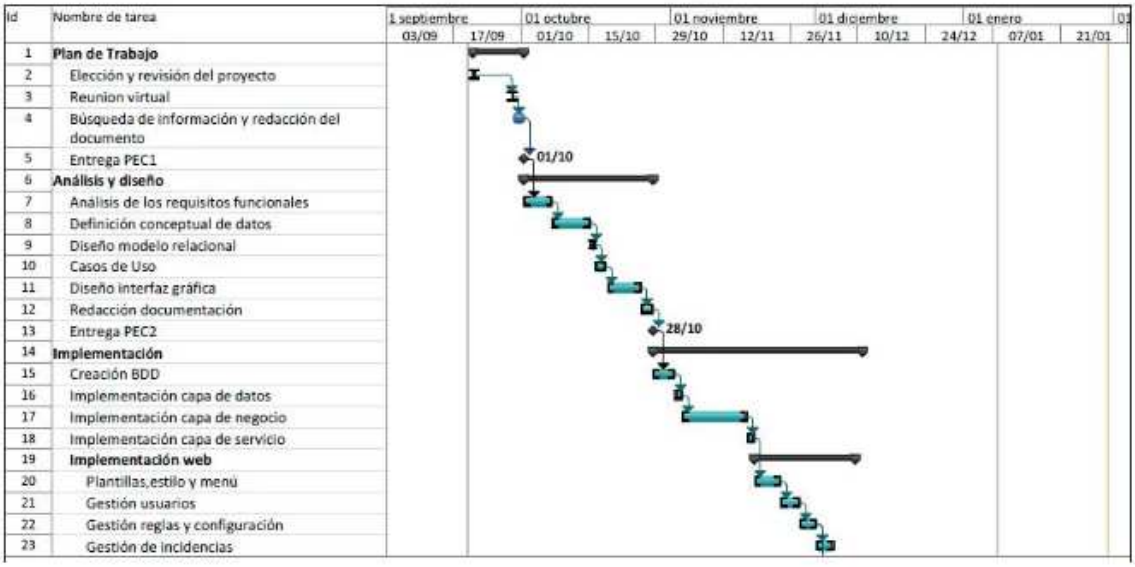

Figura 1

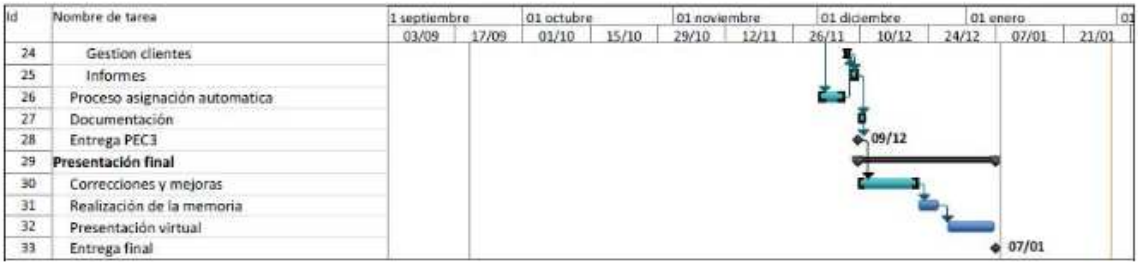

Figura 2

El tiempo estimado para los hitos **Entrega PEC1** y **Entrega PEC2** fueron correctos: los entregables se pudieron liberar en las fechas establecidas, el 01/10/2012 y el 28/10/2012 respectivamente.

Sin embargo, el tiempo estimado para la fase de **Implentación** no fue adecuado: se fue muy optimista a la hora de valorar el coste algunas de las pantallas a implementar como ocurrió en los casos de las pantallas del módulo de Gestión de incidencias y de las pantallas del módulo de Gestión de clientes.

Además, en la planificación inicial tampoco se tuvieron en cuenta algunas tareas: no se tuvo en cuenta ni el tiempo dedicado a investigar sobre el uso de *Entity Framework* sobre el cual no se tenía apenas conocimiento ni la implementación de la aplicación web para la consulta de incidencias por parte del cliente.

Por todo ello, para poder cumplir el hito **Entrega PEC3** la última semana se tuvieron que dedicar más horas de las estimadas.

El tiempo dedicado a la actividad de **Correcciones y mejoras** también se ha visto incrementado. El tiempo finalmente dedicado ha sido de 45 horas, esto ha provocado que se dedicaran menos horas en las actividades que la sucedían.

En la siguiente imagen se muestra la planificación final con todos los cambios anteriormente comentados respecto a la planificación inicial:

Id	Nombre de tarea	Duración	Comienzo	Fin
1	**Plan de Trabajo**	**1,63 días**	**vie 21/09/12**	**lun 01/10/12**
2	Elección y revisión del proyecto	3 horas	vie 21/09/12	vie 21/09/12
3	Reunion virtual	1 hora	sáb 29/09/12	sáb 29/09/12
4	Búsqueda de información y redacción del documento	12 horas	sáb 29/09/12	lun 01/10/12
5	Entrega PEC1	0 horas	lun 01/10/12	lun 01/10/12
6	**Análisis y diseño**	**8,25 días**	**lun 01/10/12**	**dom 28/10/12**
7	Análisis de los requisitos funcionales	12 horas	lun 01/10/12	dom 07/10/12
8	Definición conceptual de datos	12 horas	dom 07/10/12	lun 15/10/12
9	Diseño modelo relacional	6 horas	lun 15/10/12	mar 16/10/12
10	Casos de Uso	12 horas	mar 16/10/12	jue 18/10/12
11	Diseño interfaz gráfica	16 horas	vie 19/10/12	jue 25/10/12
12	Redacción documentación	8 horas	jue 25/10/12	dom 28/10/12
13	Entrega PEC2	0 horas	dom 28/10/12	dom 28/10/12
14	**Implementación**	**14,75 días**	**dom 28/10/12**	**lun 10/12/12**
15	Creación BDD	4 horas	dom 28/10/12	jue 01/11/12
16	**Implementación capa de datos**	**1,38 días**	**jue 01/11/12**	**dom 04/11/12**
17	Investigación EF y creación capa Entidades independiente	8 horas	jue 01/11/12	sáb 03/11/12
18	Creación capa de datos	3 horas	sáb 03/11/12	dom 04/11/12
19	Implementación capa de negocio	30 horas	dom 04/11/12	sáb 17/11/12
20	Implementación capa de servicio	8 horas	sáb 17/11/12	mar 20/11/12
21	**Implementación web**	**7,5 días**	**mar 20/11/12**	**dom 09/12/12**
22	Plantillas,estilo y menú	8 horas	mar 20/11/12	sáb 24/11/12

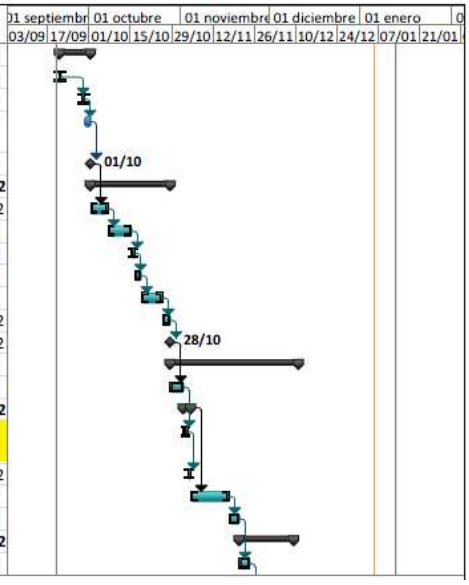

Figura 3

Id	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
23	Gestión usuarios	12 horas	sáb 24/11/12	jue 29/11/12	22
24	Gestión reglas y configuración	8 horas	jue 29/11/12	sáb 01/12/12	23
25	Gestión de incidencias	16 horas	dom 02/12/12	jue 06/12/12	24
26	Gestion clientes	8 horas	jue 06/12/12	vie 07/12/12	28
27	Informes	12 horas	vie 07/12/12	dom 09/12/12	26
28	Proceso asignación automatica	12 horas	dom 02/12/12	jue 06/12/12	24
29	**Implementación web cliente**	**0,63 días**	**dom 09/12/12**	**dom 09/12/12**	
30	Página consulta	3 horas	dom 09/12/12	dom 09/12/12	27
31	Página opinión	2 horas	dom 09/12/12	dom 09/12/12	30
32	Documentación	5 horas	dom 09/12/12	lun 10/12/12	29
33	Entrega PEC3	0 días	lun 10/12/12	lun 10/12/12	32
34	**Presentación final**	**8,88 días**	**mar 11/12/12**	**lun 07/01/13**	
35	Correcciones y mejoras	45 horas	mar 11/12/12	sáb 29/12/12	33
36	Realización de la memoria	8 horas	sáb 29/12/12	mar 01/01/13	35
37	Presentación virtual	18 horas	mar 01/01/13	lun 07/01/13	36
38	Entrega final	0 días	lun 07/01/13	lun 07/01/13	37

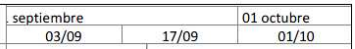

Figura 4

Se ha marcado en amarillo las etapas que no se consideraron en la planificación inicial y en gris las tareas que cuya duración estimada no fue correcta y provocó una desviación.

Análisis y diseño

Requerimientos funcionales / no funcionales

Requerimientos funcionales

Los requerimientos generales del sistema son los siguientes:

1. El sistema se compondrá de dos subsistemas: un sistema de administración y gestión de incidencias al que podrán acceder sólo los usuarios propios de éste; y otro sistema de consulta de incidencias por parte de los clientes.

2. El sistema diferenciará cuatro tipos de usuarios: usuario de atención al público, técnico, supervisor y cliente.

3. A la primera parte del sistema, expuesta en el punto 1, sólo podrán acceder usuarios activos de atención al público, técnicos y supervisores previa identificación por medio de un usuario y contraseña.

4. Este sistema se compondrá de 5 módulos diferentes: gestión de usuarios del sistema, configuración y reglas del sistema, gestión de clientes y servicios, gestión de incidencias e informes.

5. A la segunda parte del sistema podrá acceder el cliente por medio de su email y el identificador de la incidencia a revisar.

<u>Sistema gestor de incidencias</u>

En los siguientes puntos se listan los diferentes requerimientos por módulo:

<u>Gestión de Usuarios del Sistema</u>

2.1 El sistema debe permitir dar de alta y modificar usuarios del sistema. Esta acción sólo puede ser llevada a cabo por un usuario de tipo supervisor.

2.2 El sistema debe permitir listar y consultar los diferentes usuarios del sistema. Esta acción puede ser llevada a cabo por supervisores, técnicos y usuarios de atención al público.

2.3 La ficha de un usuario incluirá los siguientes campos: login, contraseña, nombre, email, tipo de usuario, activo, horas de trabajo diario, listado de rangos de vacaciones y permitir añadir comentarios.

2.4 El campo tipo de usuario podrá tener tres posibles valores: usuario de atención al público, supervisor y técnico

2.5 El sistema debe permitir a un usuario cambiar su propia contraseña.

Configuración del sistema y reglas

3.1 El sistema debe permitir a los supervisores poder configurar si los clientes pueden consultar las incidencias dado un identificador de cliente e identificador de incidencia.

3.2 El sistema debe permitir configurar si al dar de alta una incidencia será asignada automáticamente a un técnico en base a unas reglas.

3.3 El criterio para la asignación de incidencias sólo tendrá en cuenta una regla de entre las 3 posibles. Estas 3 posibilidades son:

- Asignar al técnico que antes pueda iniciar la resolución de una incidencia en base a la última fecha de resolución estimada de sus incidencias sin resolución.
- Asignar al técnico con menor número de horas estimadas total en sus incidencias sin resolución.
- Asignar al técnico con menor número de incidencias sin resolución.

3.4 El sistema debe permitir seleccionar únicamente una de estas reglas como criterio aplicación en la asignación automática.

3.5 El sistema debe permitir configurar si el cliente recibirá un email cuando una incidencia sea resuelta o se actualice sus comentarios.

3.6 El sistema debe permitir configurar si a los clientes se les ofrecerá un encuesta para pedir su opinión sobre el servicio prestado una vez se haya resuelto una incidencia.

3.7 El sistema deberá permitir configurar si el cálculo de fecha de resolución estimada de una incidencia se calculará automáticamente en base al tiempo

estimado en su resolución, horas dedicadas por parte del usuario asignado y fecha estimada de su incidencia más tardía.

3.8 El sistema debe permitir configurar al supervisor el tiempo por defecto que se le asignará a una incidencia nada más abrirse si el usuario de atención al cliente no establece ninguno siempre que este valor configurado esté informado y sea mayor a 0.

Gestión de incidencias

4.1 Una incidencia tendrá los siguientes campos: asunto, descripción, comentarios, cliente que notificó la incidencia, programa al que se refiere, fecha de creación, usuario que creo la incidencia, fecha de resolución, estado, técnico asignado, tiempo estimado de resolución, tiempo real de resolución, fecha de resolución estimada y fecha final de resolución.

4.2 Los posibles estados de las incidencias son los siguientes: Pendiente de asignación, Pendientes de resolución, Iniciada y Resuelta.

4.3 El sistema debe permitir a un usuario de atención al público abrir una incidencia.

4.4 Al abrir una incidencia se deberán rellenar siempre los campos de: cliente, programa asociado o servicio al que se refiere, asunto y una descripción del problema

4.5 El campo fecha de creación lo establecerá automáticamente el sistema en el momento de abrirse la incidencia

4.6 Opcionalmente al abrirse la incidencia se podrán rellenar los campos de tiempo estimado y fecha de resolución estimada.

4.7. El sistema al asignar una incidencia a un técnico establecerá el estado de la misma como Pendiente de resolución.

4.8. El sistema al asignar una incidencia a la cola de pendientes de asignación establecerá el estado de la misma como Pendiente de asignación.

4.9. El sistema debe permitir a un supervisor asignar manualmente una incidencia a un técnico.

4.10. El sistema debe permitir supervisores y usuarios de atención al público consultar cualquier incidencia del sistema, esté resuelta o no.

4.11 El sistema debe permitir a técnicos consultar cualquier incidencia asignada a su usuario, esté resuelta o no.

4.12. El sistema debe permitir a supervisores o usuarios de atención al público cambiar el estado de una incidencia de Resuelta a Pendiente de resolución.

4.13 El sistema debe permitir a un técnico cambiar el estado de una incidencia asignada a su usuario de Pendiente de resolución a Iniciada y de Iniciada a Resuelta.

4.14 El sistema debe permitir a supervisores cambiar el estado de una incidencia de Iniciada a Resuelta.

4.15 El sistema debe permitir a los usuarios añadir comentarios a las incidencias.

4.16 El sistema debe permitir a los supervisores establecer o cambiar el tiempo estimado de resolución y establecer o cambiar la fecha estimada de resolución.

4.17 El sistema debe permitir a los técnicos establecer o cambiar el tiempo dedicado a la incidencia.

4.18 El sistema establecerá automáticamente la fecha final de resolución en el momento de cambiar el estado de una incidencia a Resuelta.

4.19 El sistema debe permitir a usuarios de atención y supervisores al público filtrar y ordenar las incidencias del sistema por técnico asignado, estado y fecha estimada de resolución incluyendo las asignadas a la cola de pendientes de asignar.

4.20 El sistema de no tener configurada la asignación automática de incidencias asignará la incidencia a una cola de incidencias pendientes de asignación.

4.21 El sistema mostrará por defecto a los supervisores el listado de incidencias pendientes de asignar.

4.22 El sistema debe permitir a los técnicos filtrar y ordenar sus incidencias asignadas por estado y fecha final de resolución.

4.23 El sistema dejará constancia el establecimiento o cambios de fecha estimada de resolución u horas estimadas, así como el cambio de una incidencia de Resuelto a Pendiente de resolución. Esta constancia puede ser llevada a cabo por medio de un comentario automático en la incidencia.

4.24 El sistema debe ofrecer a los supervisores la posibilidad volver a ejecutar la asignación automática de incidencias sobre una en concreto siempre que el sistema esté configurado para ello.

4.25 Los supervisores y usuarios de atención al cliente podrán revisar la opinión dada por el cliente en referencia al servicio ofrecido en la resolución de una incidencia.

4.26 Los técnicos podrán revisar la opinión dada por los clientes de sus incidencias asignadas.

Gestión de clientes y servicios

5.1 El sistema debe permitir a los supervisores dar de alta y modificar los datos de clientes.

5.2 El sistema debe permitir indicar si un cliente puede añadir comentarios extras a una incidencia.

5.3 El sistema debe permitir a un supervisor dar de alta diferentes servicios y/o programas y asociarlos a un cliente.

5.4 El sistema debe permitir listar todos los clientes dados de alta así como consultar su ficha a supervisores, técnicos y usuarios de atención al público y los diferentes servicios/programas asociados a esos clientes.

Informes

6.1 El sistema debe facilitar diferentes informes para conocer el estado de las incidencias así como de los tiempos estimados, dedicados y la opinión de sus clientes del servicio ofrecido. Estos informes solo serán visibles por supervisores.

6.2 El primer informe ofrecerá la posibilidad de filtrar y/o agrupar las incidencias por los siguientes criterios: cliente, servicio y técnico, mostrando el número de

incidencias en el sistema dado ese criterio y pudiendo incluir solo las incidencias resueltas.

6.2 El segundo informe mostrará el tiempo dedicado a cada incidencia contra el tiempo estimado en ella, pudiendo filtrar las incidencias por cliente, servicio y técnico y por aquellas incidencias cuyo tiempo real sea mayor al estimado. También debe mostrar el tiempo real total y el tiempo dedicado total de las incidencias filtradas.

6.3 El tercer informe facilitará la puntuación media de todas las incidencias de un cliente, técnico o programa.

6.4 El último informe mostrará las incidencias fuera de planificación, lo que se entiende por todas aquellas incidencias que ya debieran estar resueltas en el momento de revisar el informe.

Sistema de consulta de los clientes

7.1 El cliente podrá acceder a una incidencia por medio del número de incidencia que desea consultar y el email del cliente y consultar su estado, los comentarios hechos y la fecha estimada de resolución.

7.2. El cliente podrá añadir comentarios a la incidencia siempre y cuando el sistema esté configurado para ello.

7.3 El cliente podrá dar opinión del servicio ofrecido una vez resuelta la incidencia siempre que el sistema esté configurado para ello.

Requerimientos no funcionales

1. Emplear la plataforma .NET, en concreto la tecnologías de ASP.NET, WCF y Entity Framework

2. Aplicación web que resulte atractiva así como fácil de usar.

3. La aplicación debe ser compatible con las últimas versiones de los siguientes navegadores: Internet Explorer, Firefox y Chrome.

4. Proporcionar un sistema flexible que se puede adaptar a los distintos flujos de trabajo adoptados por diferentes tipos de empresas.

Diagrama de casos de uso

El sistema presenta los siguientes actores:

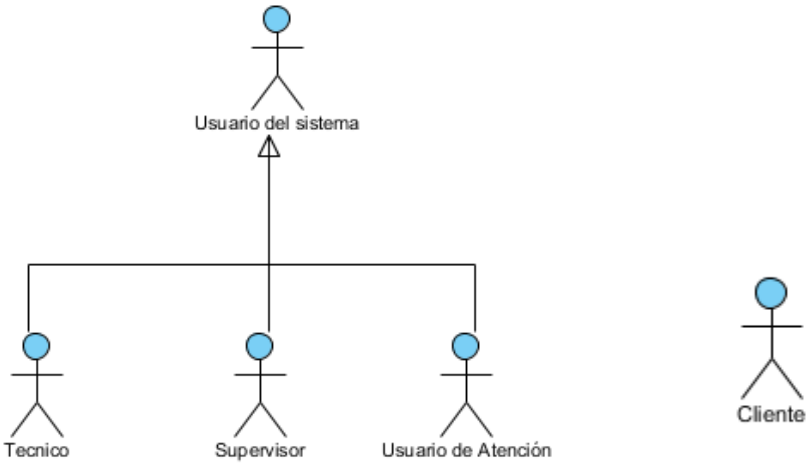

Figura 5

Por una lado tenemos los usuarios propios del sistema, entre los que podemos distinguir tres perfiles: técnicos, supervisores y usuarios de atención al cliente, y por otro los clientes que accederán su sistema de consulta de incidencias.

Dividiremos el diagrama entre estos dos subsistemas mencionados en el apartado anterior.

Sistema de Administración y gestión de incidencias

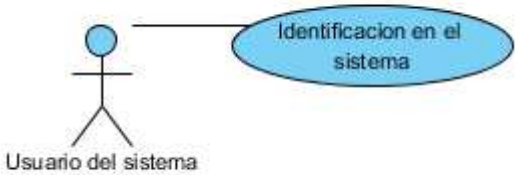

Figura 6

A continuación mostramos los digramas de los casos de uso de los diferentes módulos:

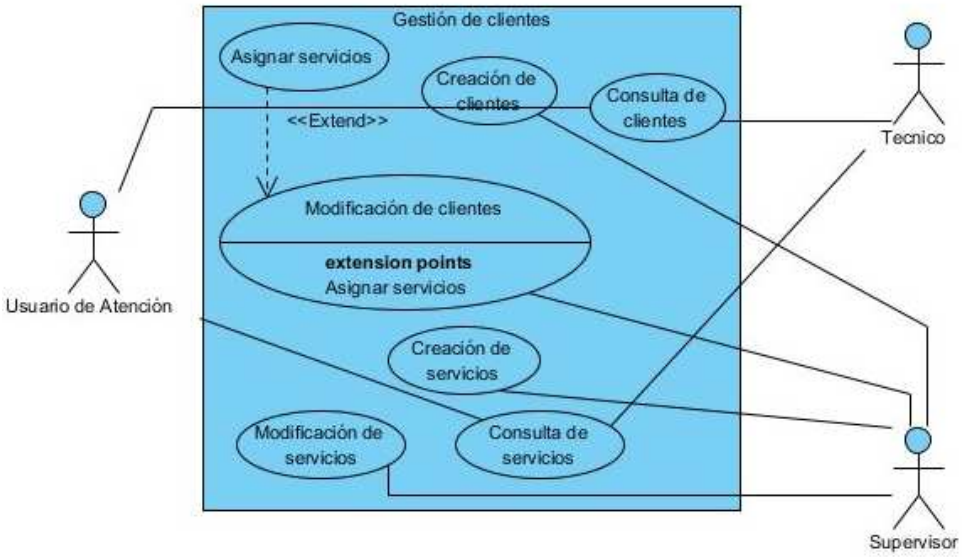

Figura 7

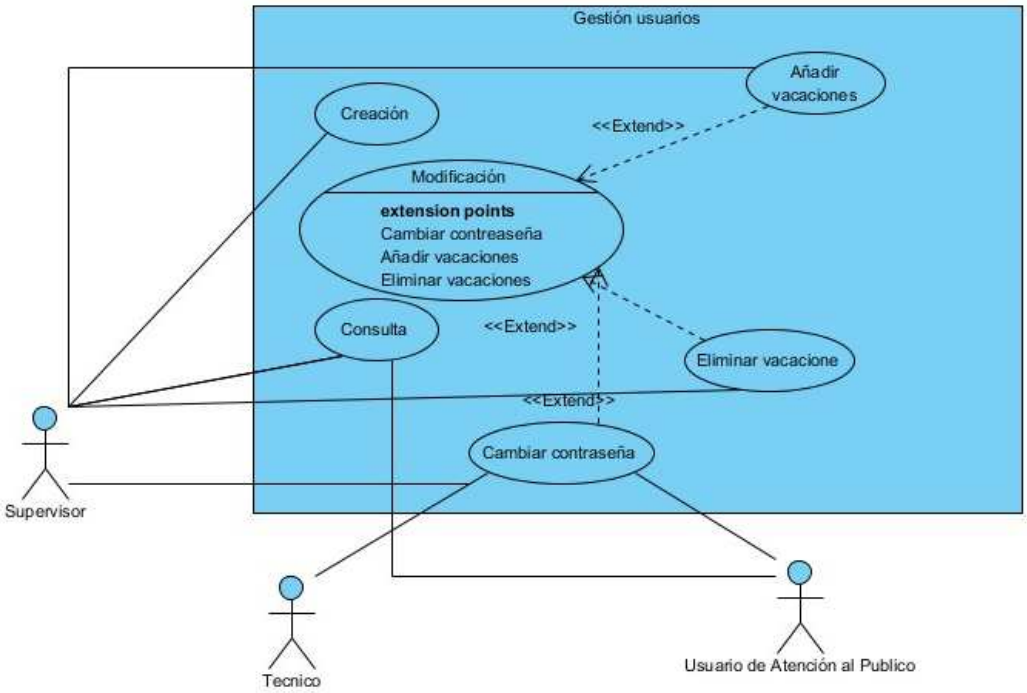

Figura 8

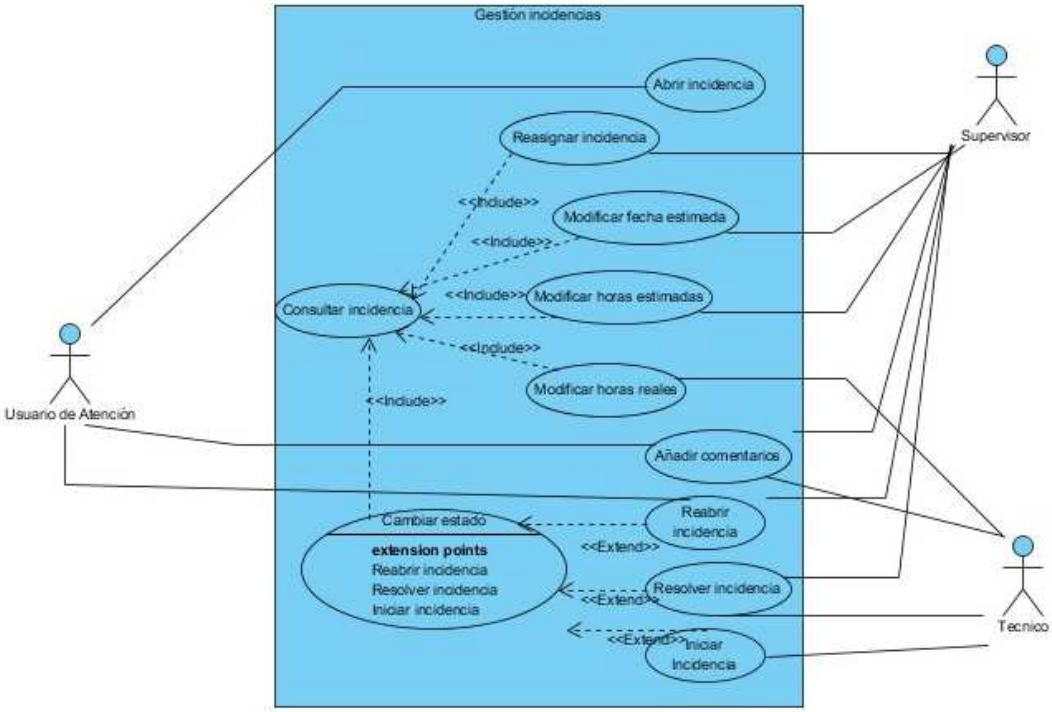

Figura 9

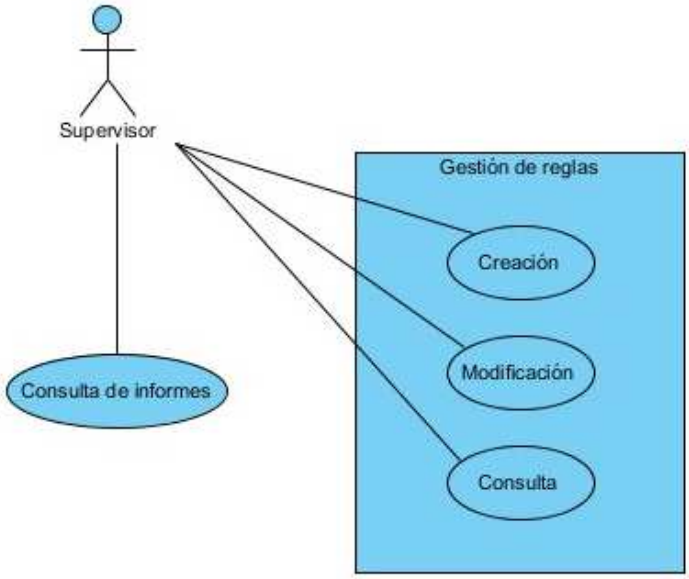

Figura 10

Sistema de consulta del cliente

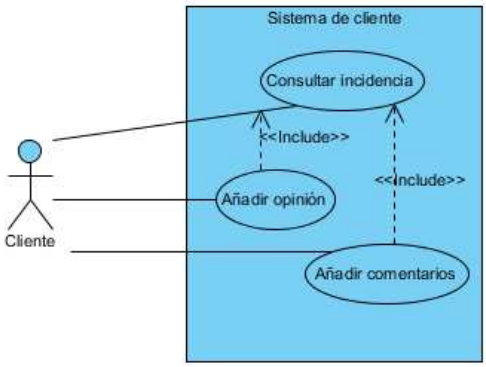

Figura 11

Diagrama de clases

A continuación se muestra el diagrama de las clases de negocio del sistema:

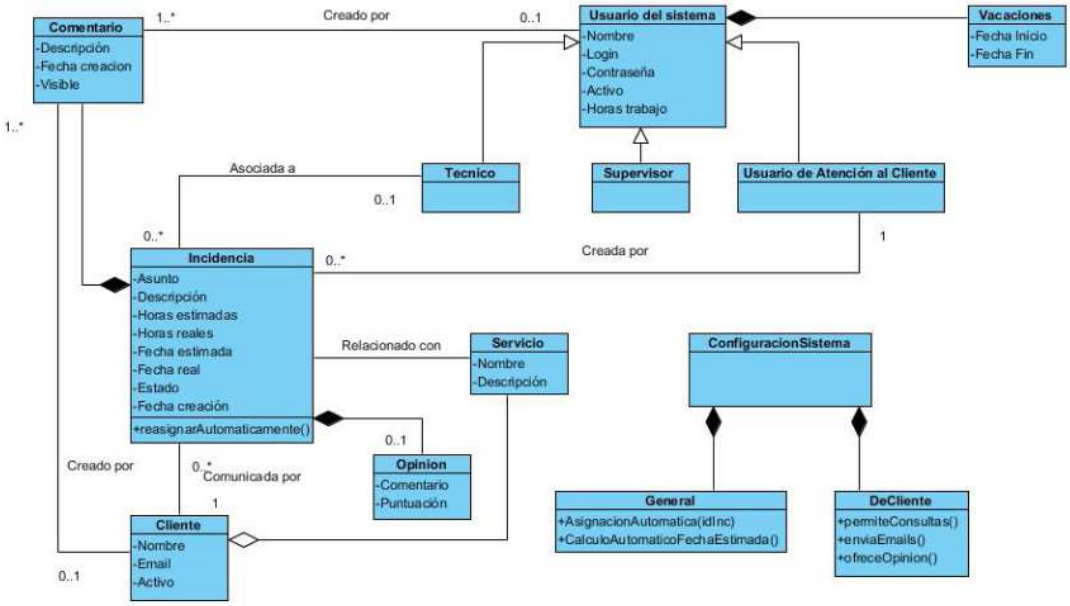

Figura 12

Diagrama de estados

Como ya se ha comentado en apartados anteriores, una incidencia siempre tendrá un estado y ese estado variará durante el ciclo de vida de la incidencia. El siguiente diagrama pretende describir los diferentes estados por los que pasará así como las acciones que provocan la transición de un estado a otro:

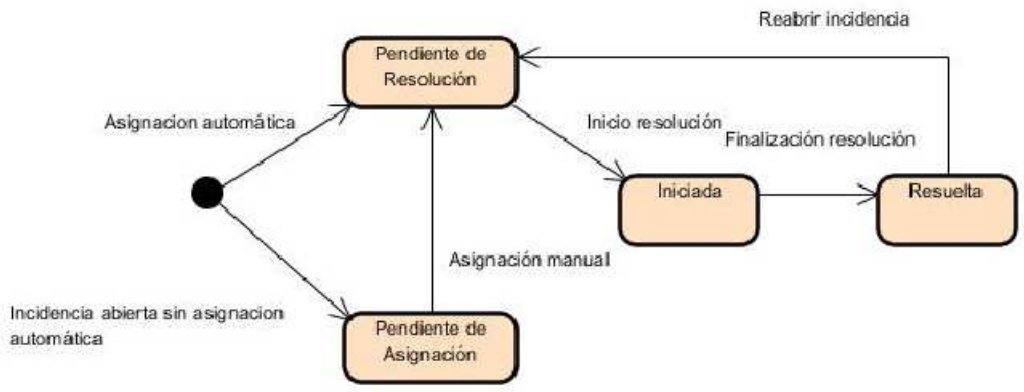

Figura 13

Una incidencia recién abierta podrá tener dos posibles estados: Pendiente de Asignación o Pendiente de Resolución dependiendo de si la incidencia está pendiente de asignar a un técnico o no, respectivamente.

Una incidencia cuyo estado sea Pendiente de Asignación cambiará a Pendiente de Resolución cuando un supervisor reasigne la incidencia manualmente a un técnico.

El técnico será el encargado de pasar la incidencia a Iniciada y luego a Resuelta una vez solucionado el fallo.

Diagrama de flujo

La siguiente figura muestra el flujo que seguirá el proceso de asignación automática de incidencias:

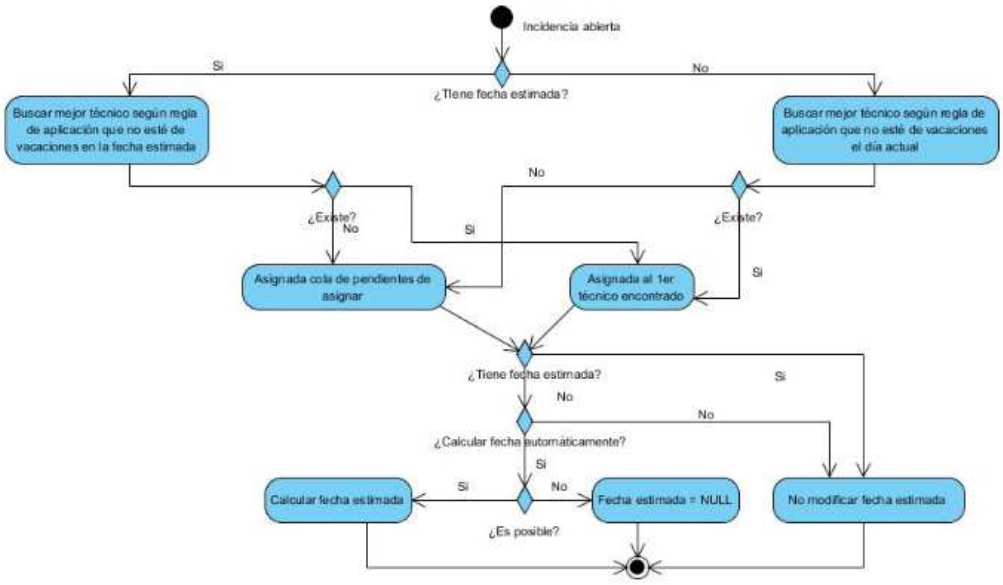

Figura 14

Más abajo, se muestra el diagrama de flujo que seguirá el proceso de Calcular fecha estimada que vemos en la figura anterior:

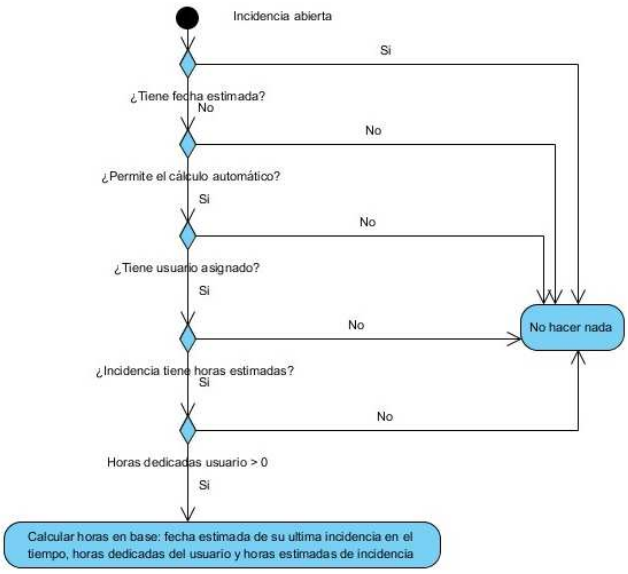

Figura 15

Diseño de la base de datos: modelo E-R

El modelo Entidad-Relación de la base de datos es el siguiente:

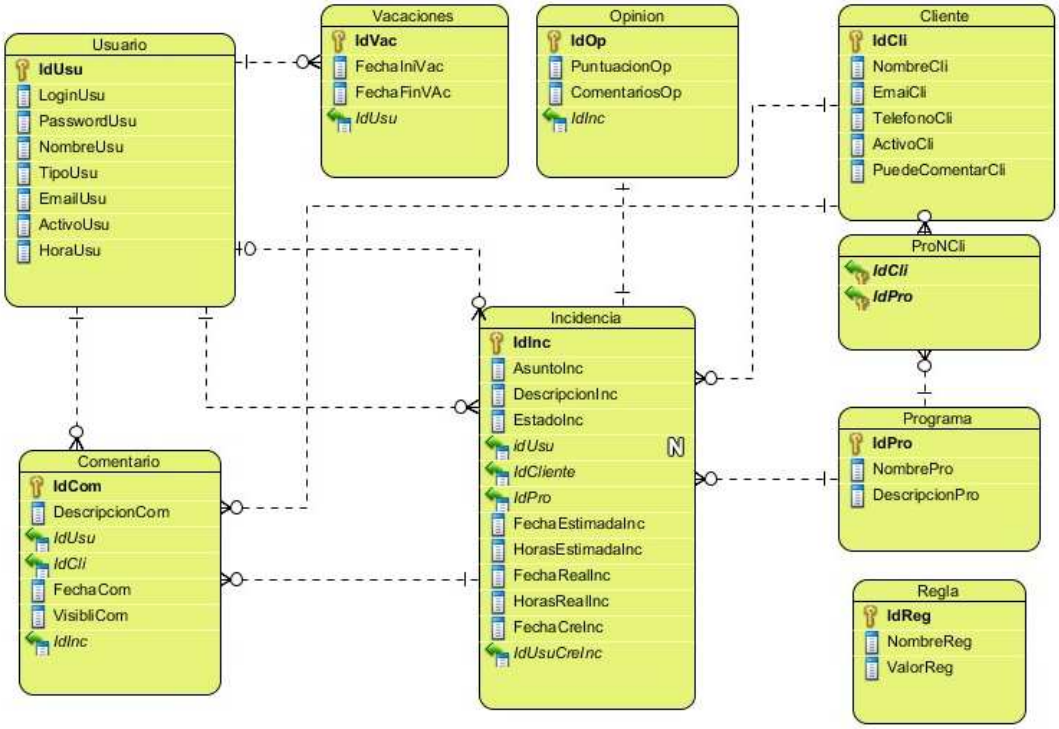

Figura 16

Arquitectura de la aplicación

Hardware

A continuación se ha adjuntado el diagrama de la arquitectura hardware a seguir para poder implantar el sistema.

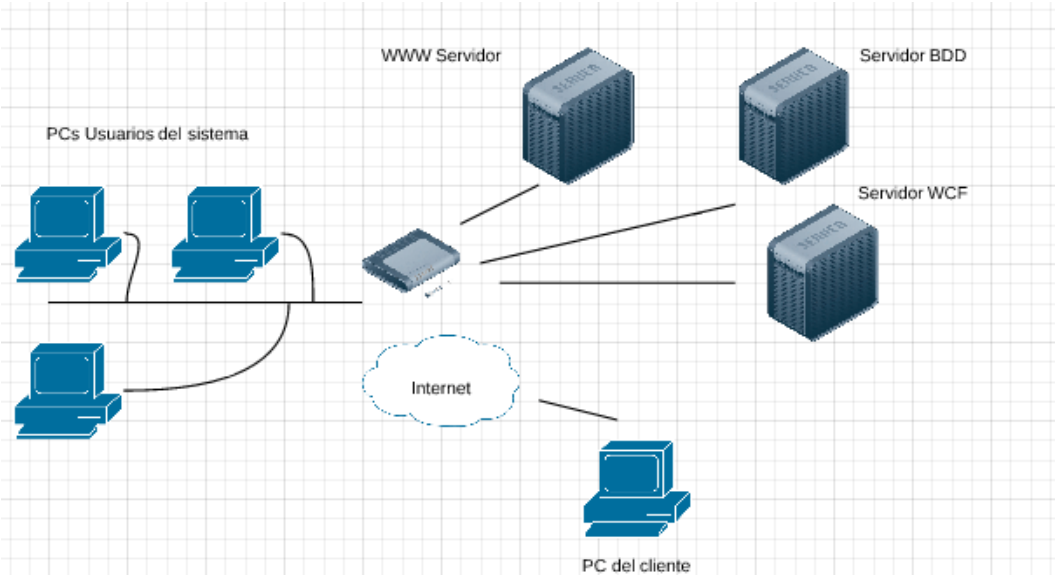

Figura 17

En este esquema, se ha separado los servicios en un servidor diferente al servidor web, al igual que el servidor de BDD que podemos alojar en un servidor diferente a los servicios WCF y al de las páginas ASP.NET. Aunque en este esquema se han dispuestos servidores diferentes para alojar la BDD, el servicio WCF y las aplicaciones webs, si la carga de operaciones es baja, se pueden alojar todos en un mismo servidor.

Estos servidores, a su vez, deben tener acceso a la red LAN donde estarán conectados los PCs de los usuarios del sistema y a Internet para que puedan tener acceso al sistema de consultas de cliente los usuarios externos a la LAN.

Software

Se pretende desarrollar un sistema software dividido en varias capas donde en la capa que se ocupa de la visualización es llevada a cabo por una serie de formularios ASP.NET más código HTML, CSS y Javascript mientras que toda la lógica de negocio es implementada en otra capa independiente y el acceso a los datos, en otra capa, por medio de ADO.Net Entity Framework.

Entre las capas de servicio y presentación se situa la capa de servicios, donde servicios WCF exponen la funcionalidad públicamente de la capa de negocio.

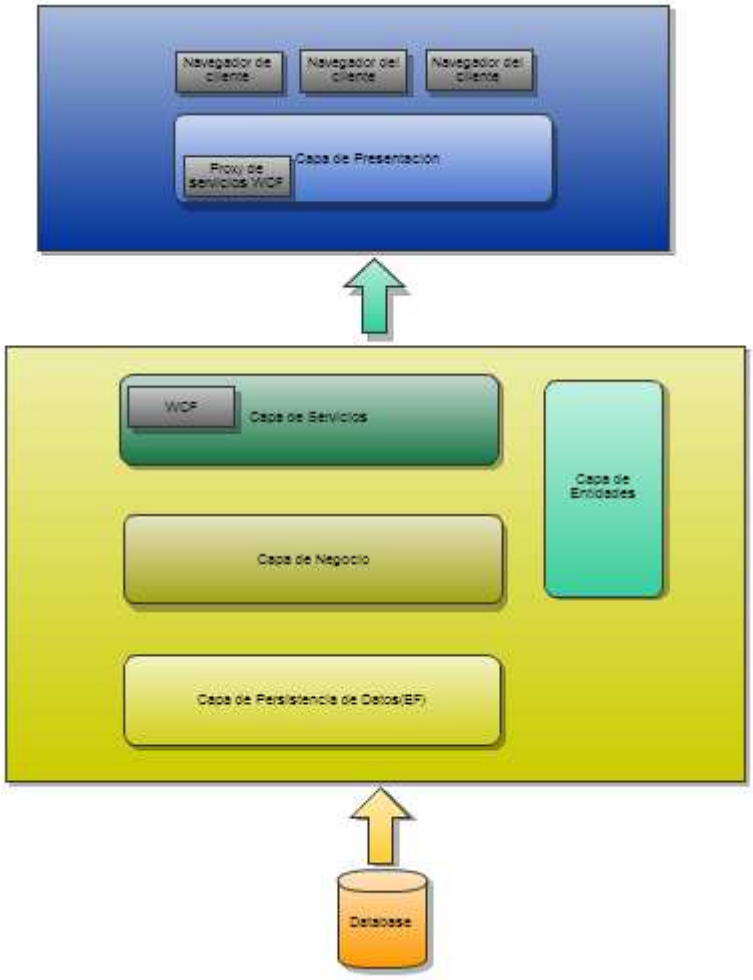

Figura 18

Diseño de la interfaz de usuario

En las siguientes pantallas se muestran los *wireframes* confeccionados con la herramienta *Pencil Project*. No se ha entrado en detalle en aspectos de estilos, sólo en la navegación del sistema, la estructura de las pantallas y en la información presentada.

Acceso al panel de administración

Lo primero que debe hacer un administrador del sistema para acceder a éste es identificarse mediante su usuario y contraseña:

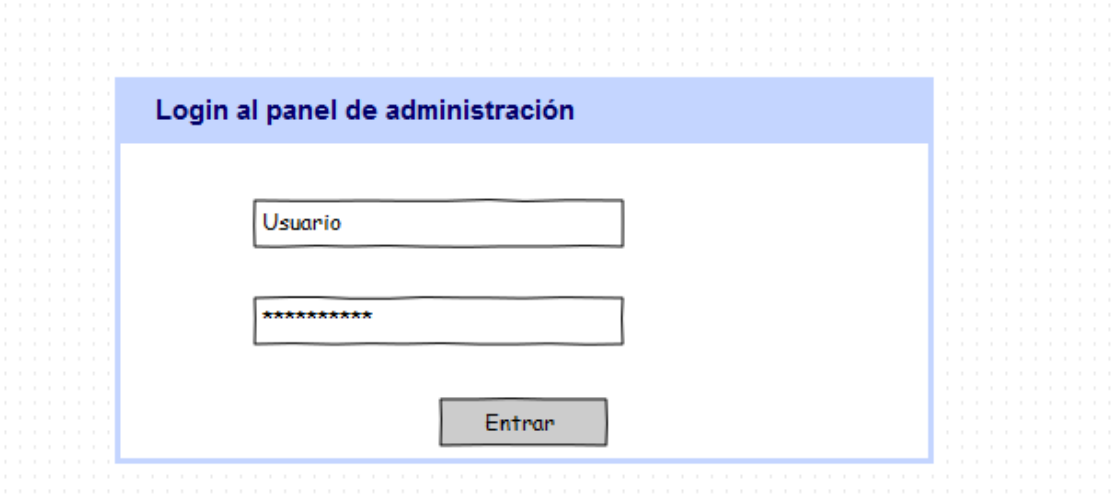

Figura 19

Una vez identificado el sistema redireccionará al usuario a la home, que por defecto será el listado de incidencias.

En los siguientes apartados se mostrarán las diferentes pantallas del sistema divididas en los 5 módulos distintos que lo componen.

Sistema de Administración y gestión de incidencias

Gestión de incidencias

Se presentan las diferentes pantallas que conforman este módulo dependiendo del usuario que este identificado.

- ***Usuario de Atención al cliente***

El listado se mostrará vacío hasta que no se aplique algún filtro:

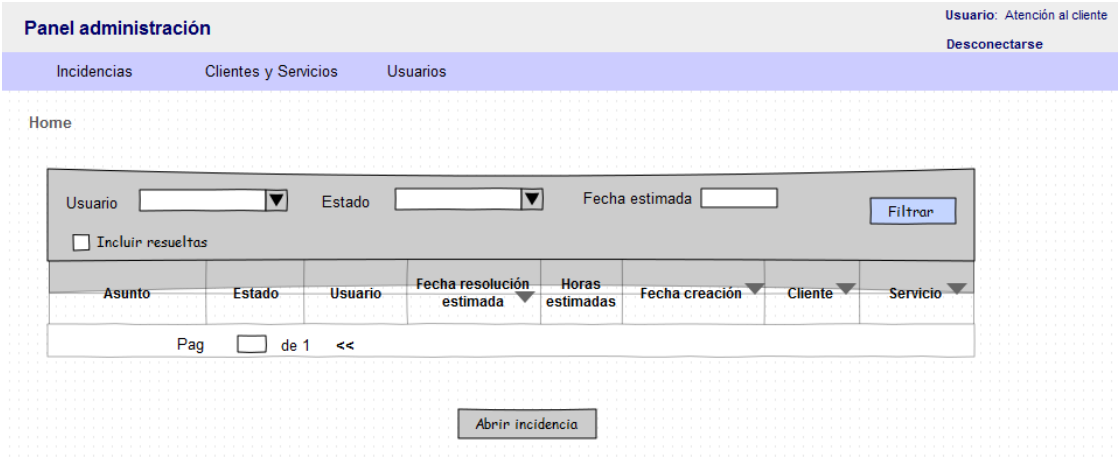

Figura 20

Desde esta pantalla se visualizará el botón de "*Abrir incidencias*" desde el que se podrán abrir nuevas incidencias:

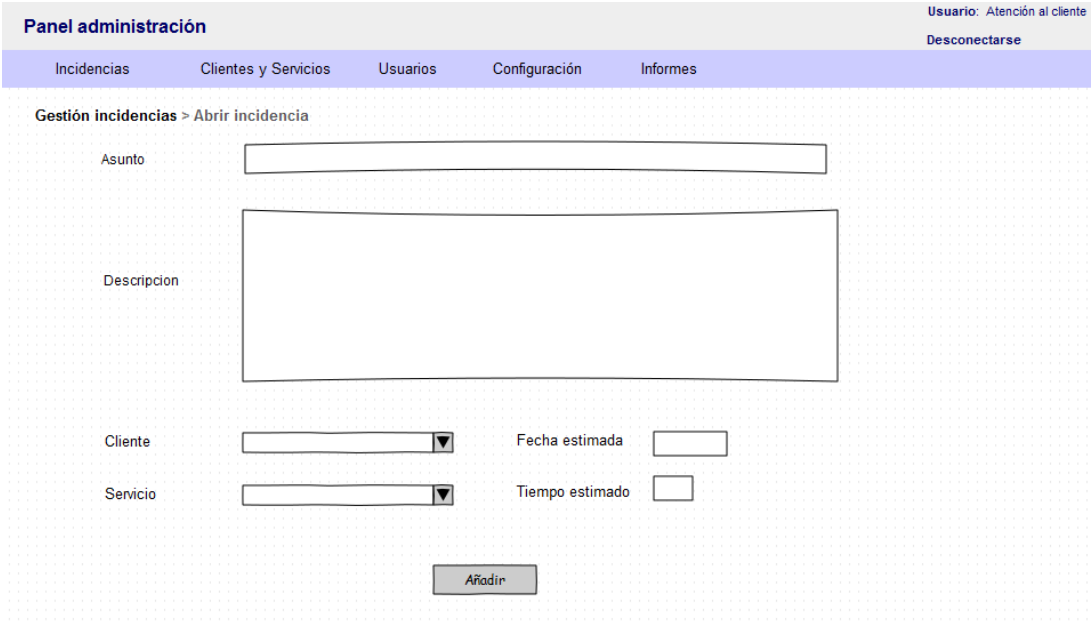

Figura 21

Desde el listado anterior, al hacer doble click sobre una incidencia, se podrán consultar los detalles de ésta:

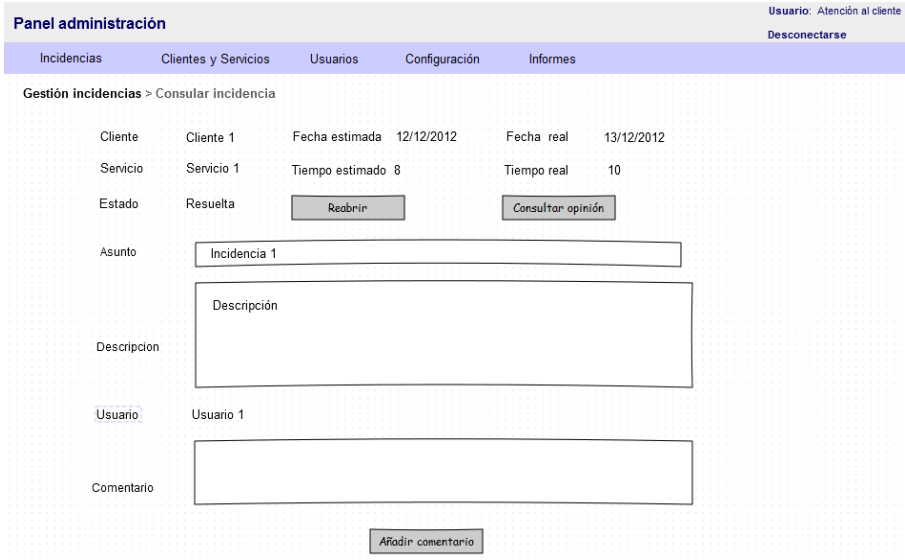

Figura 22

Desde esta pantalla se podrán añadir comentarios y reabrir las incidencias ya resueltas.

- ***Supervisor***

Por defecto, el listado mostrará las incidencias de la cola de pendientes de asginación:

Figura 23

Como en el caso anterior, se podrán consultar las incidencias dadas de alta en el sistema.

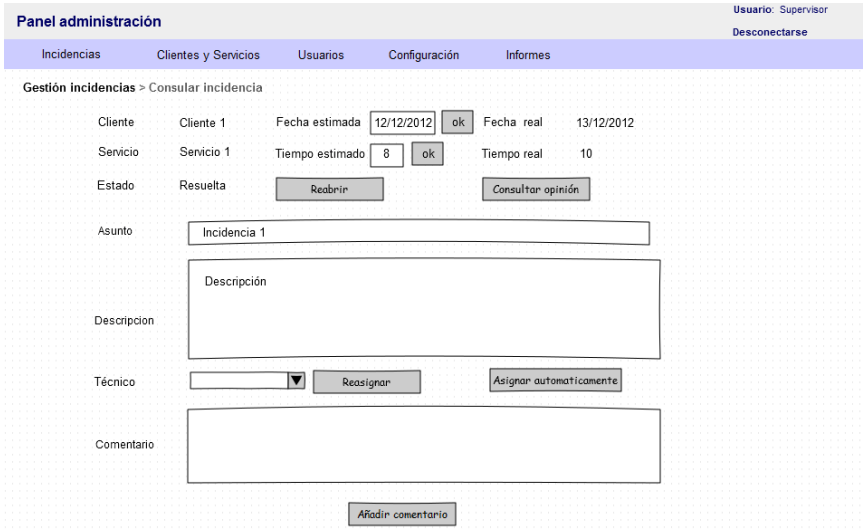

Figura 24

- ***Técnicos***

Su listado de incidencias sólo mostrará aquellas asignadas a su usuario y no podrán buscar ni acceder a incidencias de otros usuarios:

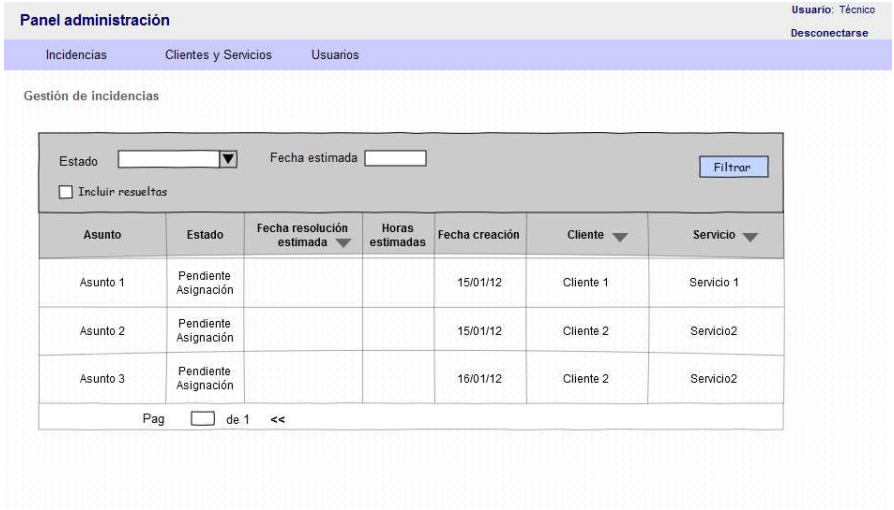

Figura 25

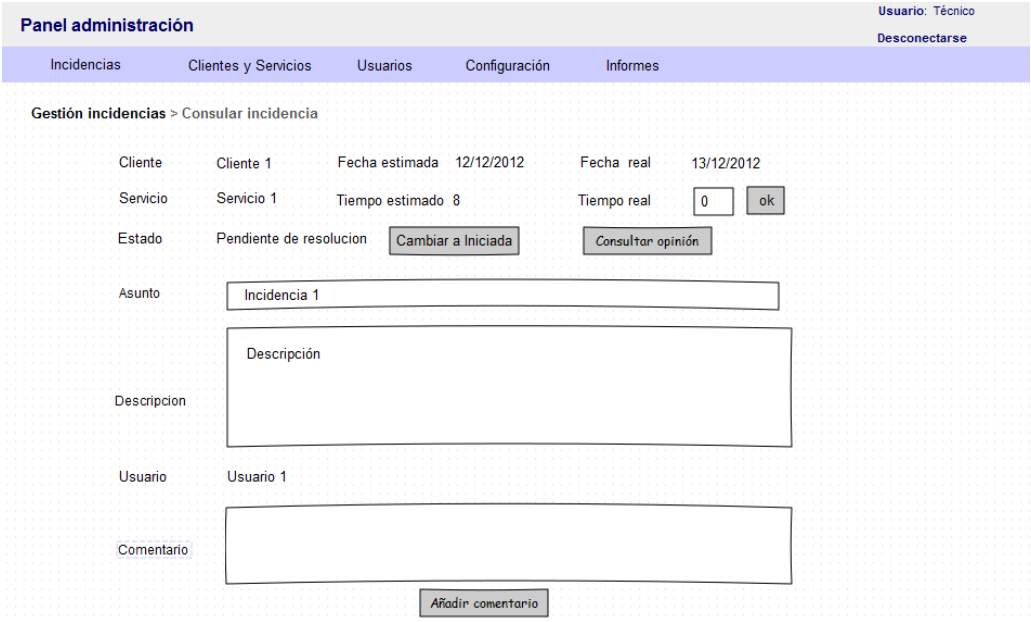

Figura 26

Todos los usuarios podrán consultar la opinión de las incidencias a las que pueden acceder desde el botón "*Ver opinión*".

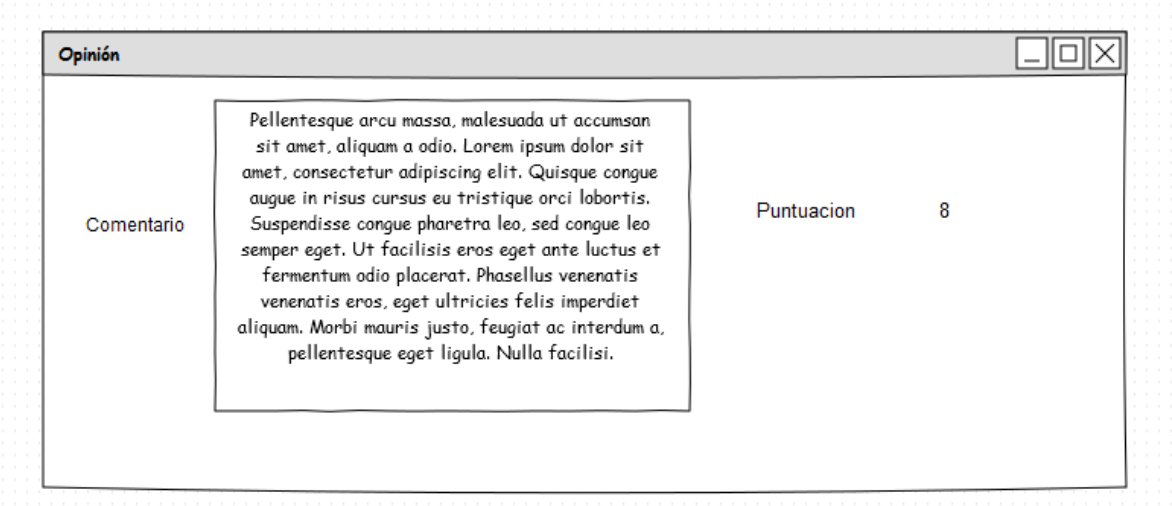

Figura 27

Gestión de clientes y servicios

A las siguientes pantallas de este módulo podrán acceder todos los usuarios:

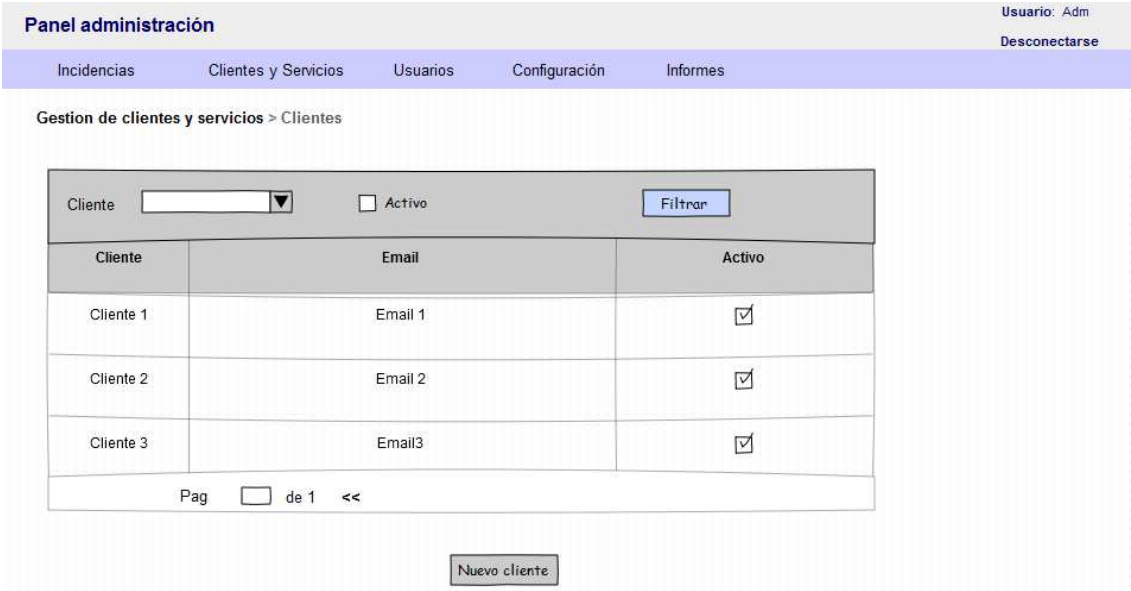

Figura 28

El botón "*Nuevo cliente*" solo será visible para los usuarios de tipo Supervisor al igual que la pantalla de creación de clientes:

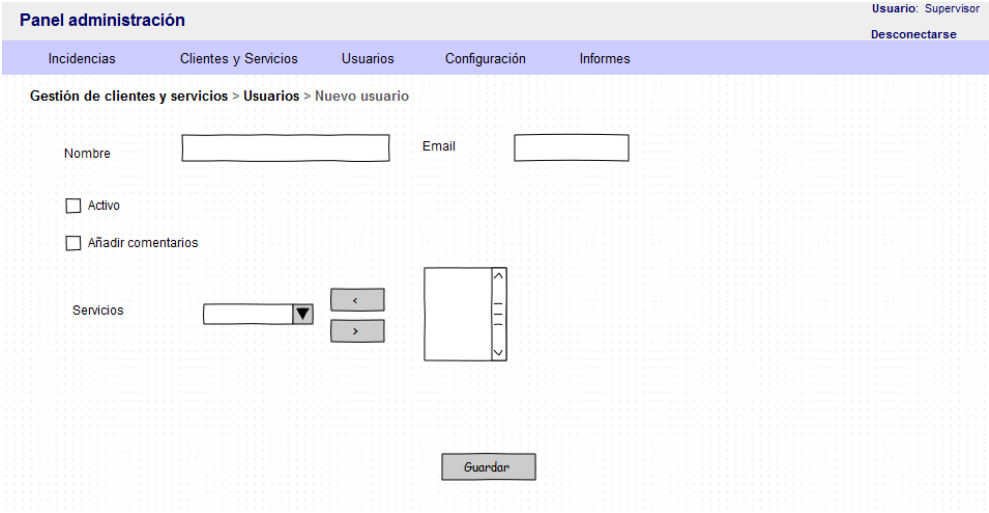

Figura 29

A la pantalla que permite consultar los datos de un cliente podrán acceder todos los usuarios aunque sólo los supervisores podrán editar los valores de los campos y ver el botón de "*Guardar*".

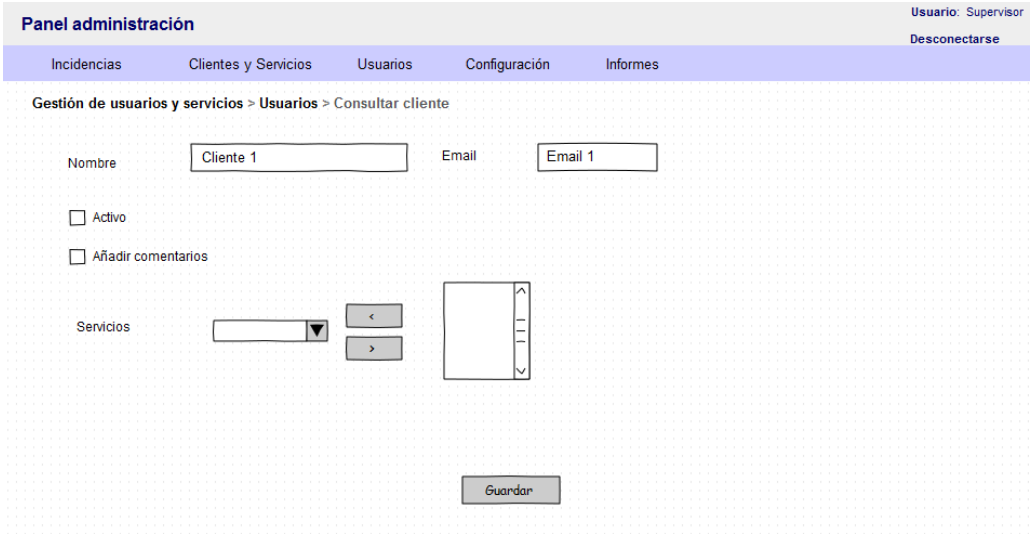

Figura 30

Todos los usuarios podrán consultar los diferentes servicios dados de alta, pero sólo podrán ser creados, editados o eliminados por supervisores

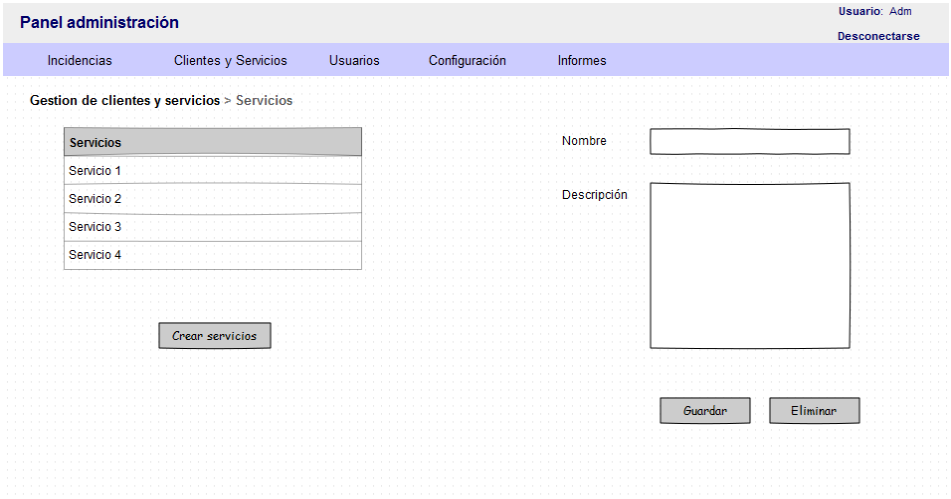

Figura 31

Gestión de usuarios

Las siguientes dos pantallas serán accesibles por todos los usuarios aunque solo a los de tipo supervisor se les mostrará el botón de "*Crear usuario*":

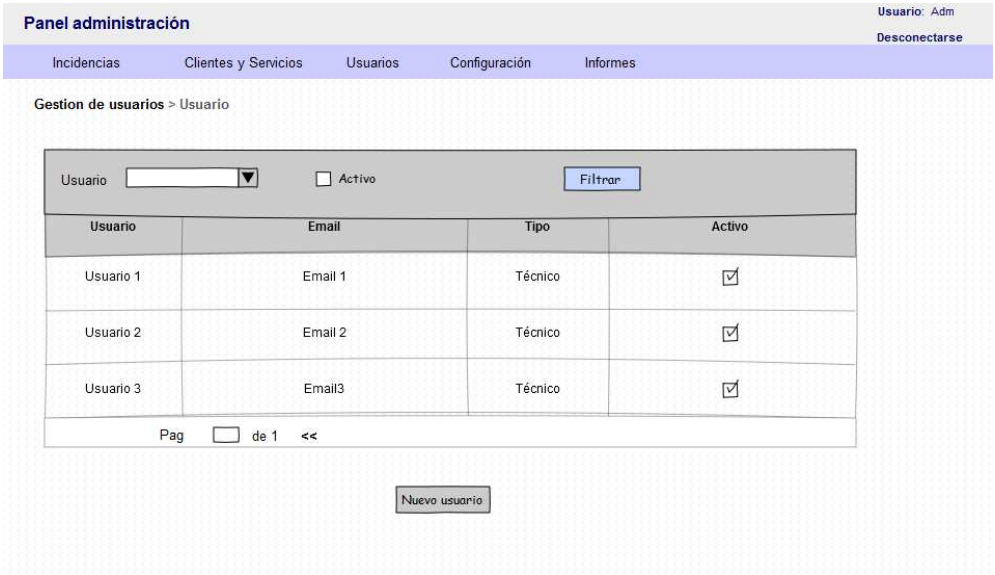

Figura 32

Accediendo a la pantalla de más abajo:

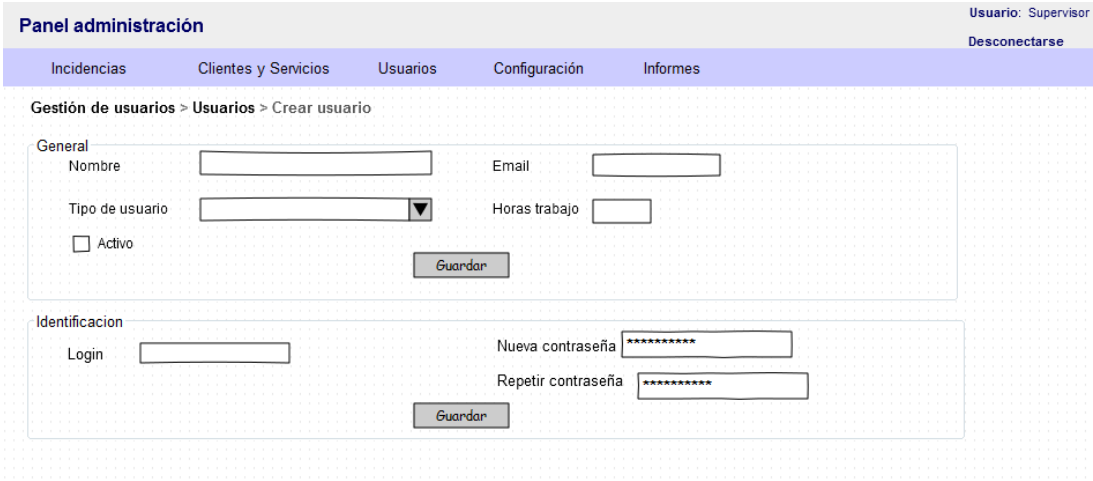

Figura 33

Al hacer doble click sobre cualquier usuario del listado se podrá consultar los datos de éste y únicamente podrán cambiar su la contraseña de su propio usuario, a excepción de supervisores que pueden modificar cualquier dato así como añadir y eliminar vacaciones.

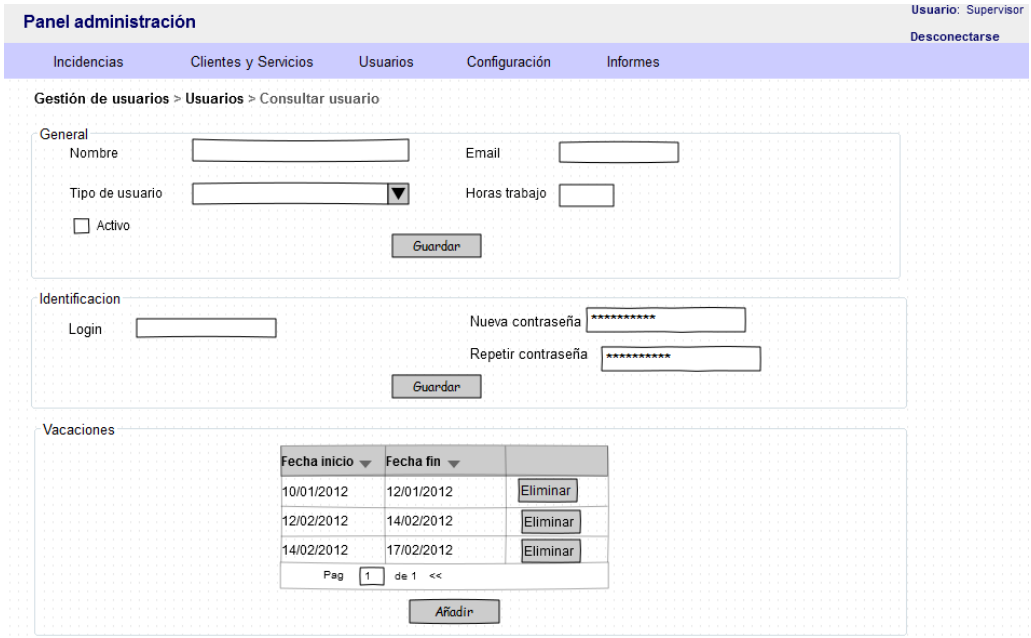

Figura 34

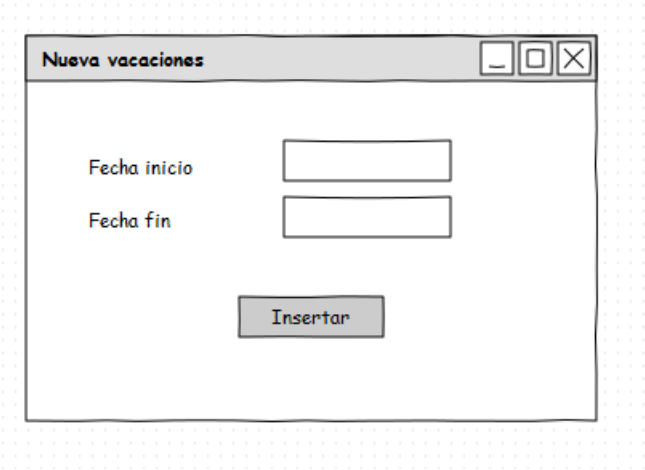

Figura 35

Configuración y reglas del sistema

Este modulo sólo será visible por supervisores. Se compone de una única pantalla:

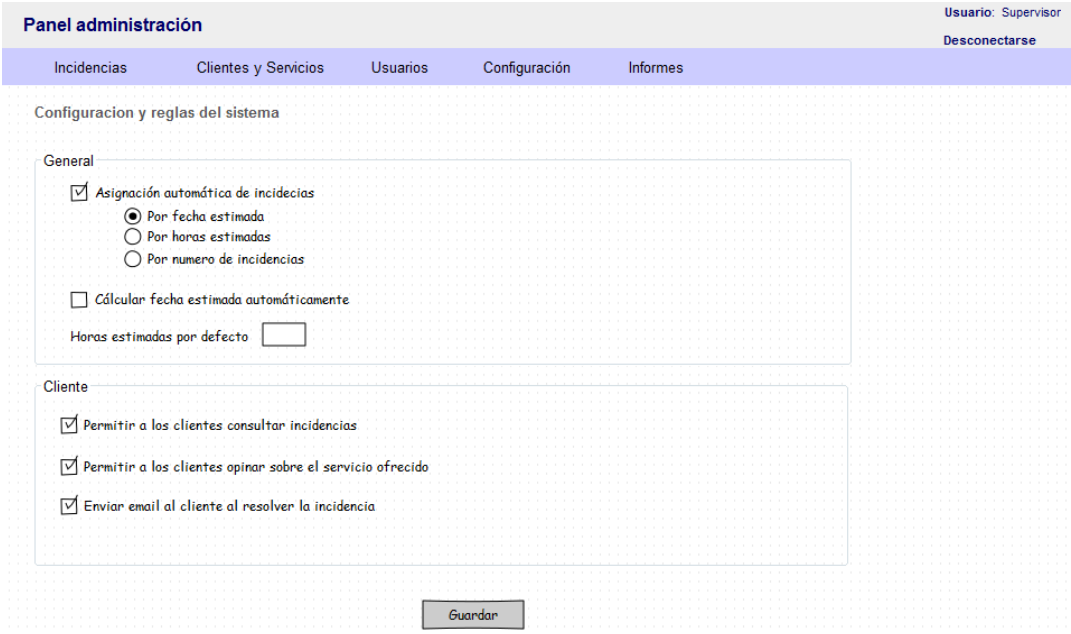

Figura 36

Informes

Este módulo sólo es accesible por supervisores. En total el sistema ofrecerá cuatro tipos de informes:

- Informe de numero de incidencias, ya sea por usuario, servicio y/o técnico:

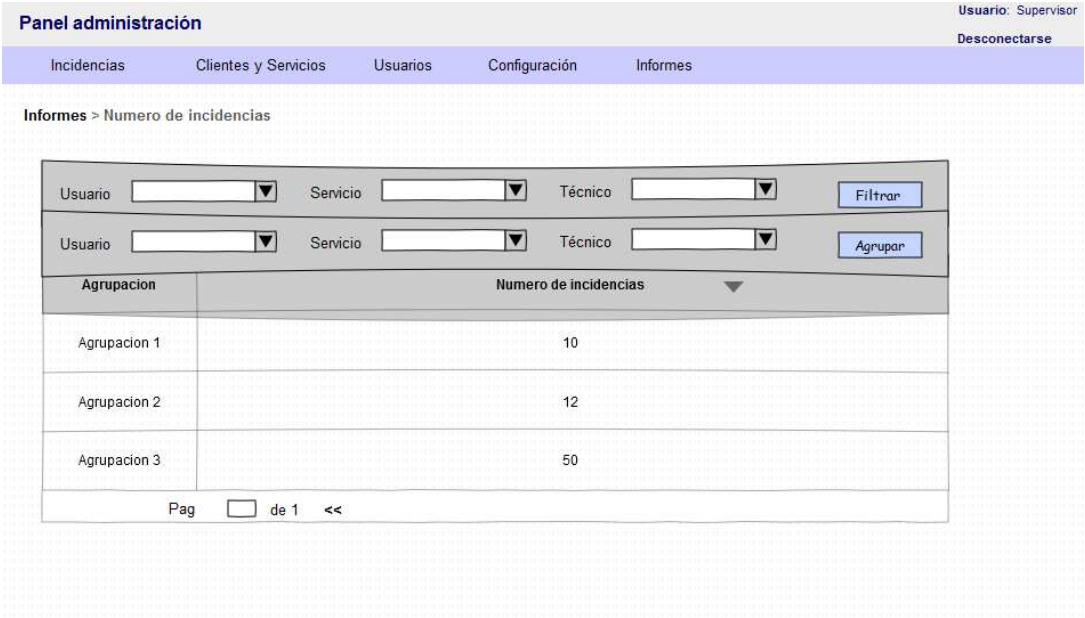

Figura 37

- Informe de incidencias fuera de planificación: mostrará el listado de aquellas incidencias cuya fecha estimada de resolución es menor al momento de solicitud del informe y cuyo estado aún no sea Resuelta:

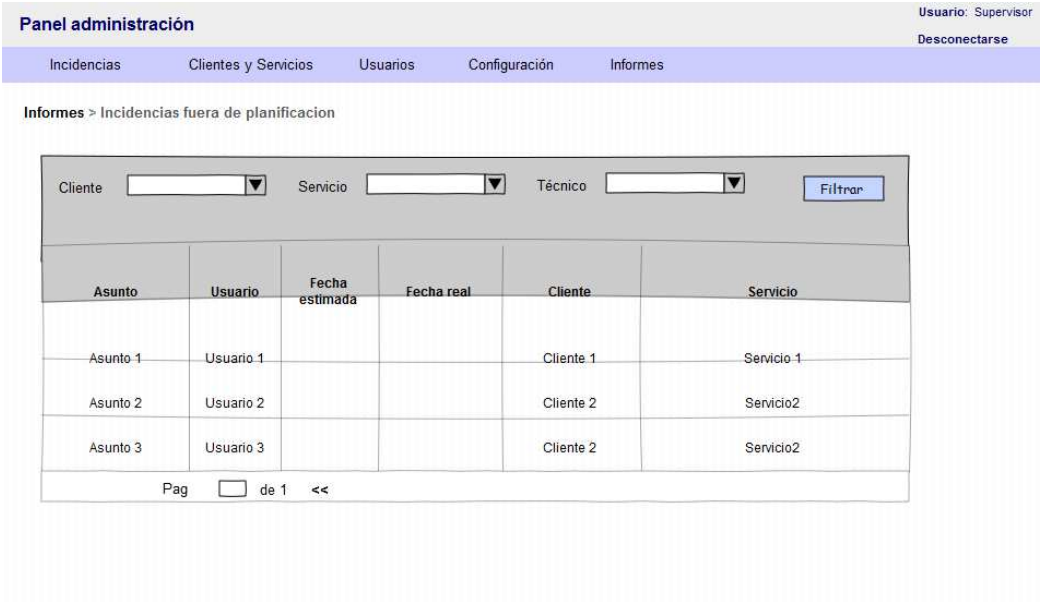

Figura 38

- Informe de incidencias tiempo real de resolución vs tiempo estimado:

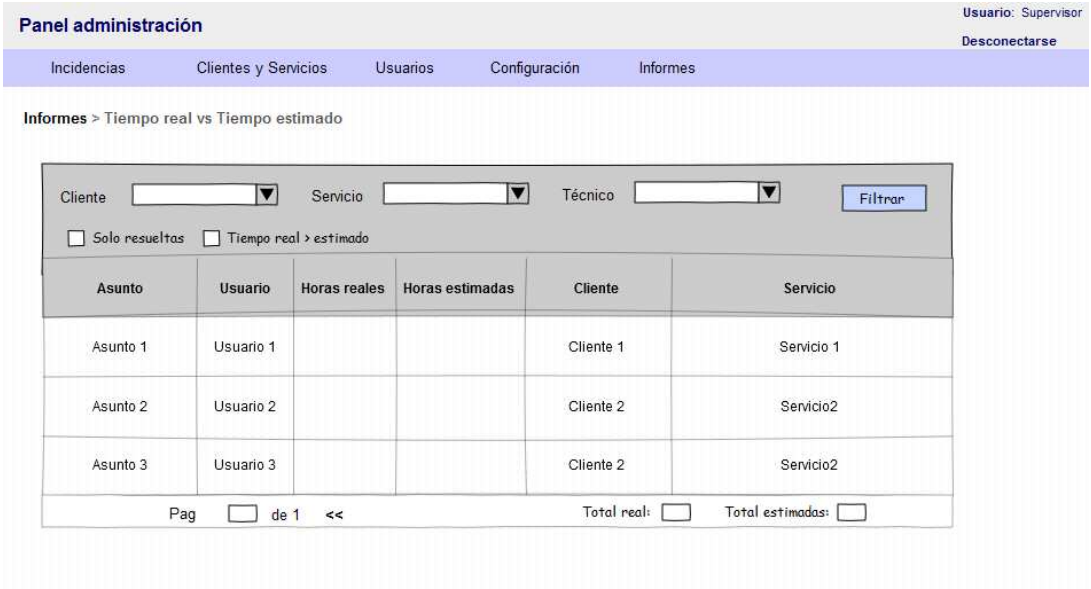

Figura 39

- Informe de puntuación media por cliente, servicio y/o técnico.

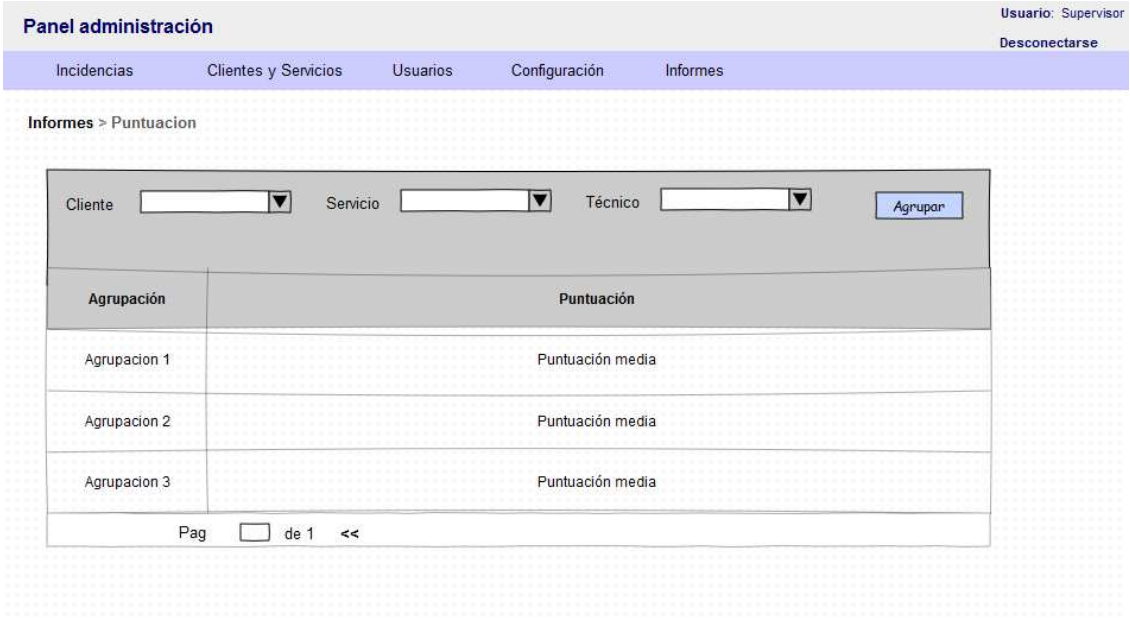

Figura 40

Sistema de consulta del cliente

Estas pantallas se han dispuesto para el acceso de los clientes a sus incidencias. Para poder consultar una incidencia, el cliente debe conocer el número de incidencia a la que desea acceder junto a su email:

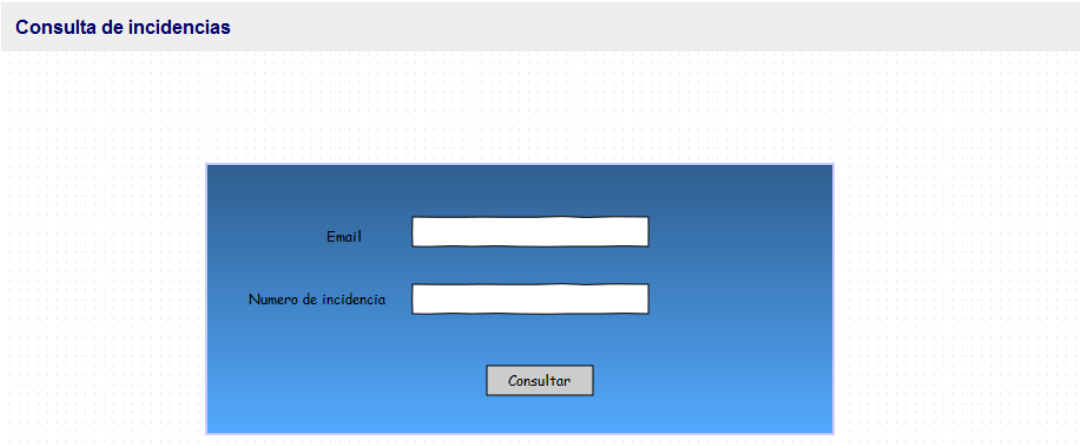

Figura 41

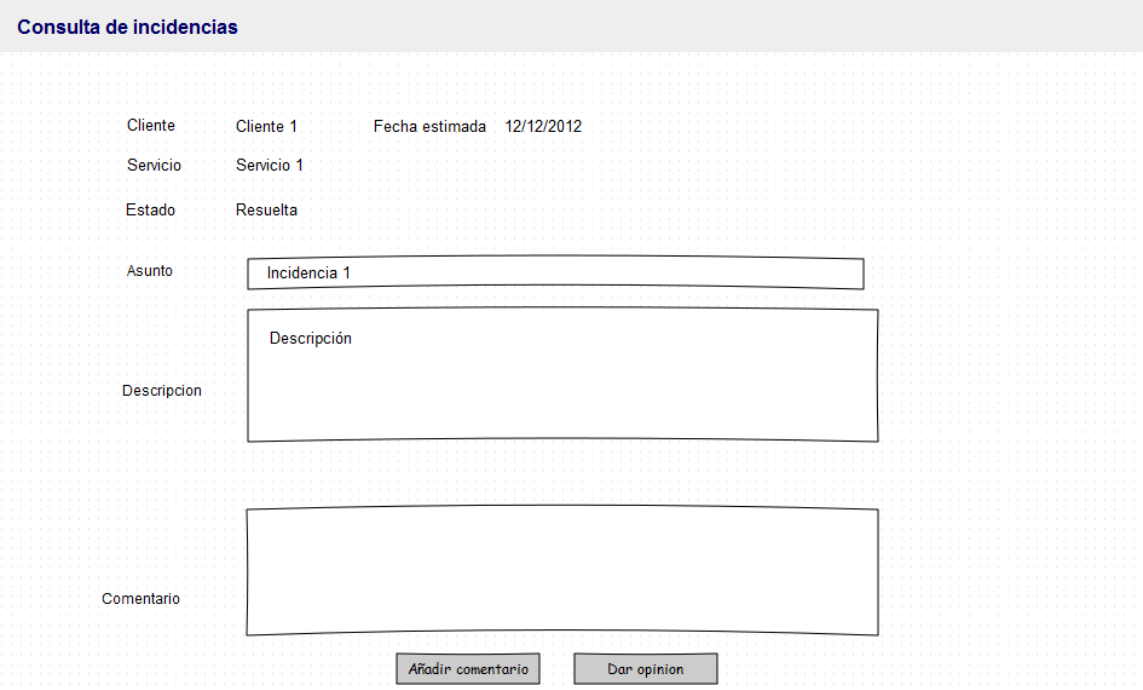

Figura 42

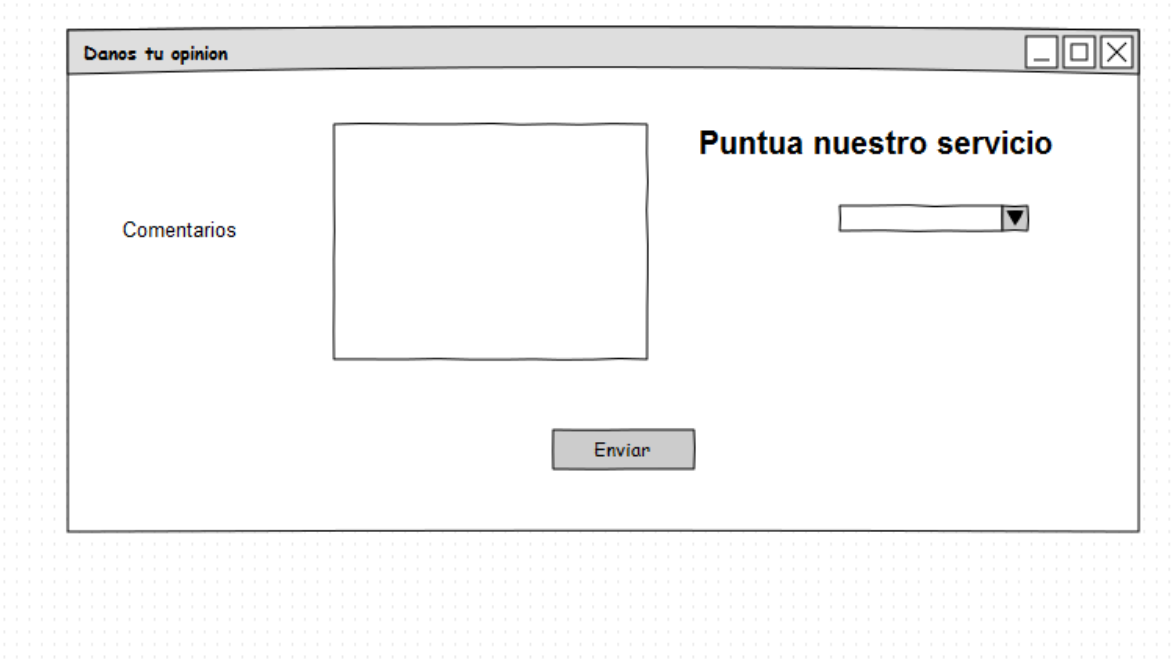
Danos tu opinion
Puntua nuestro servicio
Comentarios
Enviar

Figura 43

Implementación

Software usado

Para poder llevar a cabo implementación del producto y la realización de las distintas pruebas se ha utilizado el siguiente software:

- Entornos de desarrollo:
  - Microsoft Visual Studio 2010.
- Frameworks y librerías:
  - Entity Framework: conjunto de APIs para acceso a datos.
  - ASP.NET
  - NET Framework 4.0
  - Windows Communication Foundation
  - JQuery: librería de Javascript que permite de forma más sencilla la implementación de tareas en ese lenguaje.
- Sistema administrador de base de datos:
  - Microsoft SQL Server 2008 R2.
- Servidor web:
  - Internet Information Services (IIS).
- Navegadores empleados para las pruebas:
  - Chrome v23
  - Firefox v17
  - Internet Explorer v9

Capas de la aplicación

Se ha divido la aplicación en 4 capas diferentes:

- Capa de Datos: para el acceso a los datos y generación del modelo se ha empleado *Entity Framework*.

- Capa de Entidades: se han abstraído en esta capa las entidades, de modo que sean clases serializables para que puedan ser empleadas por los servicios WCF. Además esta capa también incluye otras clases empleadas para la comunicación entre la capa de presentación y la capa de negocio.
- Capa de Negocio: aquí es donde tenemos todas las clases que llevan a cabo la lógica de negocio.
- Capa de Servicio: servicios implementados a través de WCF.
- Capa de Presentación: esta capa la comprenden varios proyectos dentro de la solución: **WebTicketReport**, aplicación web para las administración y gestión de incidencias, y **WebClientSupport**, aplicación web para las consulta de incidencias por parte del cliente.

A continuación vamos a entrar un poco más en detalle de la implementación de las diferentes capas.

Capa de Datos

La capa de datos se ha llevado a cabo mediante el uso de Entity Framework. La herramienta de generación de modelo a partir de una base de datos ya existente se ha encargado de realizar todo el mapeo de las distintas tablas y campos así como sus relaciones a las diferentes clases dejándonos una serie de funciones desde las que podemos realizar la inserción y borrado de datos sin preocuparnos de las características internas de esta obteniendo un mayor nivel abstracción.

Capa de Entidades

Con el fin de la máxima segregación de las capas se ha creado una capa de Entidades independiente a la capa de datos. De este modo, la capa de servicios no interactuará nunca directamente con la de datos, sólo la capa de negocio será la encargada de acceder a los métodos de la capa de datos. Para ello se ha hecho uso de una herramienta automática del **Entity Framework**, "*Plantilla generador de entidades de seguimiento propio*", que nos ha generado clases **STE** a partir del modelo. Estas pueden ser serializadas para su empleo a través de un cliente **WCF**. Las clases STE no tienen dependencia con la capa de datos pero a su vez incluyen mecanismos que gestionan los cambios en sus propiedades.

Además dentro de esta capa tenemos otras clases implementadas manualmente usadas para conocer las propiedades por las que se filtrarán los listados de resultados de los objetos solicitados desde la capa de servicios.

Capa de negocio

Como ya se ha mencionado, es en esta capa donde se lleva a cabo toda la lógica del negocio. Toda la funcionalidad, ajena a la presentación de los contenidos, se lleva a cabo en esta capa. Aquí es donde se hace uso de la funciones ofrecidas por el **Entity Framework** para la perseverancia de los datos y **de Linq to Entites** para llevar a cabo las consultas.

El mayor peso de la implementación ha recaído sobre la clase Incidencia. Es aquí, por ejemplo, donde se ha desarrollado la funcionalidad que permite el envío de correos electrónicos a los clientes. Para ello se ha empleado la clase `System.Net.Mail.MailMessage` a la cuál hemos de establecer una serie de parámetros:

```
Dim correo As New System.Net.Mail.MailMessage
correo.From = New System.Net.Mail.MailAddress(System.Configuration.ConfigurationManager.AppSettings("FromEmail"))
correo.To.Add(mail)
correo.Subject = "Su incidencia " & _incidencia.IdInc & " ha sido resuelta"
correo.Body = texto
correo.Priority = System.Net.Mail.MailPriority.Normal
correo.IsBodyHtml = True

Dim smtp As New System.Net.Mail.SmtpClient
smtp.Host = System.Configuration.ConfigurationManager.AppSettings("HostEmail")
smtp.Port = System.Configuration.ConfigurationManager.AppSettings("PortEmail")
smtp.EnableSsl = System.Configuration.ConfigurationManager.AppSettings("SSLEmail")
smtp.Credentials = New System.Net.NetworkCredential(correo.From.Address, System.Configuration.ConfigurationManager.AppSettings("PassEmail"))

smtp.Send(correo)
```

Figura 44

Se ha de configurar el correo email desde el que se va a realizar el envío, la contraseña de esa cuenta, y el servidor smpt desde el que se va a realizar junto al puerto e indicando es necesario habilitar el protocolo ssl. Todas estas características se informan en el *web.config* de la siguiente capa.

Capa de servicios

Esta capa está integrada dentro del proyecto **Servicios**.

Los servicios aquí implementados serán los que proveerán a la capa de presentación de los distintos métodos de las clases de negocio.

Como se ha comentado en el punto anterior, es en el **web.config** de este proyecto, en la sección *appSettings*, donde se especifican los datos de la cuenta de correo electrónico y del servidor desde el que se realizará el envío de correos a los clientes.

```
<appSettings>
  <add key="FromEmail" value="eathyl@gmail.com"/>
  <add key="HostEmail" value="smtp.gmail.com"/>
  <add key="PassEmail" value="xxxxxx"/>
  <add key="PortEmail" value="587"/>
  <add key="SSLEmail" value="1"/>
</appSettings>
```

Figura 45

Capa de presentación

En esta capa tenemos toda la lógica de visualización y control de las distintas webs. Aunque la mayor parte de la implementación se ha llevado a cabo dentro de la capa de negocio, en la capa de presentación se realiza el control sobre la visualización de las acciones a realizar y control de permisos.

Está compuesta por dos aplicaciones web diferentes: **WebTicketReport** y **WebSupportClient**.

Comentar que la validación de los campos de los formularios así como la mayor parte de funciones realizadas en *javascript* se ha implementado mediante la librería *jquery*, por su potencia y sencillez en muchas de las operaciones y controles que habitualmente se llevan a cabo mediante *scripts* del navegador. Para la realización de la *grids* se ha empleado un *plug-in* ya desarrollado en esta misma librería, llamada *JqGrid*, sobre la cual se han tenido que efectuar una serie modificaciones para poder otorgar de la funcionalidad necesaria mostrada en la fase de diseño a las *grids* .

Evaluación de costes

Para evaluar los costes se han diferenciado 4 perfiles diferentes:

- Director de proyecto: 60 € /hora
- Analista: 50 €/hora
- Diseñador: 40 €/hora
- Programador: 40 €/hora

De esta manera si revisamos la planificación final del proyecto mostradas en el apartado de Planificación inicial vs planificación final, tenemos que:

Id	Nombre de tarea	Duración	Comienzo	Fin
1	**Plan de Trabajo**	**1,63 días**	**vie 21/09/12**	**lun 01/10/12**
2	Elección y revisión del proyecto	3 horas	vie 21/09/12	vie 21/09/12
3	Reunion virtual	1 hora	sáb 29/09/12	sáb 29/09/12
4	Búsqueda de información y redacción del documento	12 horas	sáb 29/09/12	lun 01/10/12
5	Entrega PEC1	0 horas	lun 01/10/12	lun 01/10/12
6	**Análisis y diseño**	**8,25 días**	**lun 01/10/12**	**dom 28/10/12**
7	Análisis de los requisitos funcionales	12 horas	lun 01/10/12	dom 07/10/12
8	Definición conceptual de datos	12 horas	dom 07/10/12	lun 15/10/12
9	Diseño modelo relacional	6 horas	lun 15/10/12	mar 16/10/12
10	Casos de Uso	12 horas	mar 16/10/12	jue 18/10/12
11	Diseño interfaz gráfica	16 horas	vie 19/10/12	jue 25/10/12
12	Redacción documentación	8 horas	jue 25/10/12	dom 28/10/12
13	Entrega PEC2	0 horas	dom 28/10/12	dom 28/10/12
14	**Implementación**	**14,75 días**	**dom 28/10/12**	**lun 10/12/12**
15	Creación BDD	4 horas	dom 28/10/12	jue 01/11/12
16	**Implementación capa de datos**	**1,38 días**	**jue 01/11/12**	**dom 04/11/12**
17	Investigación EF y creación capa Entidades independiente	8 horas	jue 01/11/12	sáb 03/11/12
18	Creación capa de datos	3 horas	sáb 03/11/12	dom 04/11/12
19	Implementación capa de negocio	30 horas	dom 04/11/12	sáb 17/11/12
20	Implementación capa de servicio	8 horas	sáb 17/11/12	mar 20/11/12
21	**Implementación web**	**7,5 días**	**mar 20/11/12**	**dom 09/12/12**
22	Plantillas,estilo y menú	8 horas	mar 20/11/12	sáb 24/11/12

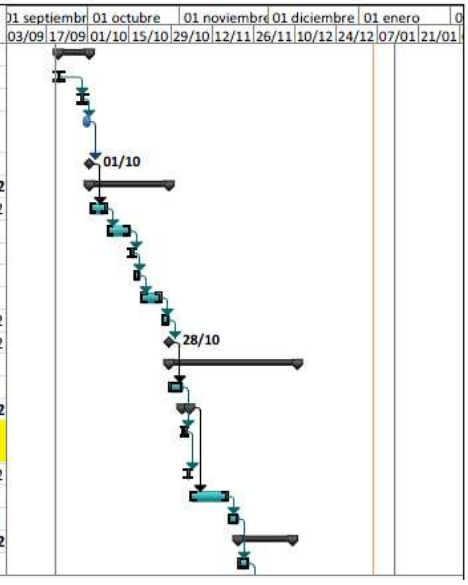

Figura 46

Id	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
23	Gestión usuarios	12 horas	sáb 24/11/12	jue 29/11/12	22
24	Gestión reglas y configuración	8 horas	jue 29/11/12	sáb 01/12/12	23
25	Gestión de incidencias	16 horas	dom 02/12/12	jue 06/12/12	24
26	Gestion clientes	8 horas	jue 06/12/12	vie 07/12/12	28
27	Informes	12 horas	vie 07/12/12	dom 09/12/12	26
28	Proceso asignación automatica	12 horas	dom 02/12/12	jue 06/12/12	24
29	**Implementación web cliente**	**0,63 días**	**dom 09/12/12**	**dom 09/12/12**	
30	Página consulta	3 horas	dom 09/12/12	dom 09/12/12	27
31	Página opinión	2 horas	dom 09/12/12	dom 09/12/12	30
32	Documentación	5 horas	dom 09/12/12	lun 10/12/12	29
33	Entrega PEC3	0 días	lun 10/12/12	lun 10/12/12	32
34	**Presentación final**	**8,88 días**	**mar 11/12/12**	**lun 07/01/13**	
35	Correcciones y mejoras	45 horas	mar 11/12/12	sáb 29/12/12	33
36	Realización de la memoria	8 horas	sáb 29/12/12	mar 01/01/13	35
37	Presentación virtual	18 horas	mar 01/01/13	lun 07/01/13	36
38	Entrega final	0 días	lun 07/01/13	lun 07/01/13	37

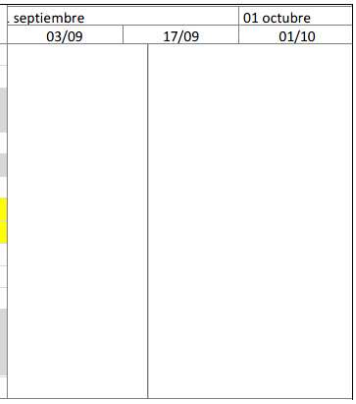

Figura 47

- Director de proyecto: 42 horas dedicadas
- Analista: 50 horas dedicadas
- Diseñador: 16 horas dedicadas
- Programador: 184 horas dedicadas

El coste total serían: 13020 euros.

Trabajo futuro

El proyecto contempla los objetivos trazados, uno de ellos, era la asignación y cálculo automático de usuarios y fechas y este sentido donde más mejoras podemos hacer:

- Poder configurar técnicos por servicios.. Sería muy interesante poder tener para un servicio o programa un listado de técnicos destinados a resolver las incidencias de ese servicio.
- Añadir a la configuración del usuario sus días laborables. Ahora mismo el sistema toma por defecto que los usuarios pueden resolver incidencias de lunes a viernes. Si tuviéramos un servicio de incidencias las 24 horas del día los 7 días de la semana, esta asignación no sería precisa.
- Poder aplicar más de un regla para escoger al mejor de los técnicos a resolver la incidencia, en caso de empates.
- Tipificar los servicios y poder configurar usuarios en base a esa tipificación.. Por ejemplo, diferencia entre errores hardware o software.

Otros puntos a mejorar serían:

- Añadir más filtros en los listados. Sería muy interesante por no decir casi necesarios que en el listado de incidencias se pudieran filtro por parte de un asunto, entre otros.
- Al escoger un cliente al dar de alta una incidencia, solo poder ver los servicios relacionados con ese cliente.
- Poder configurar los perfiles de usuario. Ahora mismo los permisos de los tres diferentes tipos de usuarios que nos encontramos en el sistema vienen predefinidos.
- Poder adjuntar ficheros a la incidencia. Siempre aportan mayor información, a veces una imagen puede aclarar mucho más que veinte líneas de explicación.

Por ultimo comentar lo atractivo que sería poder ofrecer la posibilidad de poder generar informes y formularios de opinión manualmente. Aunque esto ofrecería una mayor complejidad al sistema y su desarrollo no es sencillo, esta mejora ya se podría considerar como una fase 2 del proyecto.

Conclusiones

El producto obtenido ha sido un sistema básico de administración y gestión de incidencias. Cumple con la funcionalidad mínima necesaria que se requiere de este tipo de sistemas añadiendo una serie de características interesantes como son la asignación automática de incidencias, el cálculo automático de la fecha estimada de reolución y el *feedback* con el cliente permitiéndole introducir sus opiniones respecto a la eficiencia en el servicio ofrecido.

Con este proyecto como base, y como se ha comentado en el apartado anterior, se puede llevar a cabo una serie mejoras de ampliación en su funcionalidad haciéndola mucho más completa y flexible con el objetivo de que pueda cubrir las necesidades de casi cualquier empresa; repartiendo las distintas mejoras en diferentes fases de implementación para conseguir un producto mucho más completo y que permita una mayor distribución.

En este sentido, gracias a la arquitectura software en capas empleada se espera también que se pueda aumentar las plataformas sobre las que se pueda entregar el producto en caso de que así se requiera. Por ejemplo, poder realizar una aplicación móvil si uno de los nuevos requerimientos fuera la movilidad por parte del técnico. En este caso, solo tendríamos que desarrollar la aplicación móvil puesto que los servicios web nos proveerán de toda la lógica del negocio.

Una gran ayuda en este aspecto es el uso de *Entity Framework* que genera automáticamente el modelo de datos y el acceso a estos no teniéndonos que preocupar del motor de la base de datos.

En resumen, el marco de tecnologías .NET nos ofrece todas las herramientas necesarias para poder crear sistemas cuya arquitectura sea n-capas con el objetivo de crear aplicaciones escalables y flexibles, con mayor robustez y donde cada capa lleva a cabo unas tareas diferentes.

Bibliografía

- **Ecured:** "Patrón Modelo Vista Controlador" [artículo en línea]

  http://www.ecured.cu/index.php/Patr%C3%B3n_Modelo_Vista_Controlador

- **MSDN Magazine:** "Creación de aplicaciones web sin formularios Web Forms" [artículo en línea]

  http://msdn.microsoft.com/es-es/magazine/cc337884.aspx

- **Learn-a-holic:** "The Clash of MVC Frameworks" [artículo en línea]

  http://www.learnaholic.me/tag/asp-net-mvc

- **Wikipedia** *Comparison of issue-tracking systems*
  http://en.wikipedia.org/wiki/Comparison_of_ticket-tracking_systems

- **Bug Tracker.NET - Free Bug Tracking** *About Bug Tracker.NET*
  http://ifdefined.com/bugtrackernet.html

- **Mantis Bug Tracker Wiki** *Mantis Feature List*
  http://www.mantisbt.org/wiki/doku.php/mantisbt:features

- **Wikipedia** *Request* Tracker
  http://en.wikipedia.org/wiki/Request_Tracker

- **Covetel** *Request Tracker*
  http://www.covetel.com.ve/productos/requestracker.html

- **Software i-Solver** *Características del software de gestión de incidencias y tareas*
  http://itil.osiatis.es/Curso_ITIL/Gestion_Servicios_TI/gestion_de_incidentes/vision_general_gestion_de_incidentes/vision_general_gestion_de_incidentes.php

- **Microsoft Spain** *Domain Oriented N-Layered .NET 4.0 Sample App*
  http://microsoftnlayerapp.codeplex.com/

- **Osiatis** *Gestión de Incidentes*

  http://itil.osiatis.es/Curso_ITIL/Gestion_Servicios_TI/gestion_de_incidentes/vision_general_gestion_de_incidentes/vision_general_gestion_de_incidentes.php