



Universitat Autònoma
de Barcelona

Módulo de gestión de proyectos para Microsoft Dynamics CRM

Memòria del projecte
d'Enginyeria Tècnica en
Informàtica de Gestió
realitzat per

Francisco José Beltrán Álvarez

i dirigit per

Xavier Verge Mestre

Escola d'Enginyeria

Sabadell, febrer de 2010

El sotasignat, Xavier Verge Mestre,
professor de l'Escola d'Enginyeria de la UAB,

CERTIFICA:

Que el treball al que correspon la present memòria
ha estat realitzat sota la seva direcció
per en Francisco José Beltrán Álvarez.

I per a què consti firma la present.
Sabadell, febrer de 2010

Signat: Xavier Verge Mestre

Resumen

CRM (Customer Relationship Management) es la administración de todas las relaciones que pueda tener un negocio con sus clientes. Se centra en optimizar el ciclo de vida del cliente en su totalidad, desde la primera llamada realizada por el departamento de telemarketing hasta el último contacto. Además, CRM reúne metodologías, software y las capacidades de Internet para administrar de una manera eficiente y rentable las relaciones de un negocio con sus clientes.

Los módulos que forman Microsoft CRM 3.0 son marketing, ventas y servicio al cliente. Ofrece una solución rápida, flexible y asequible que ayuda a impulsar mejoras cuantificables en todos los procesos empresariales que afectan a sus clientes.

La gestión de proyectos es la disciplina de organizar y administrar recursos de manera tal que se pueda terminar el trabajo requerido en el proyecto dentro del alcance, el tiempo, y coste definidos. Un proyecto es un esfuerzo temporal, único y progresivo, emprendido para crear un producto o un servicio también único.

El proyecto consistirá en instalar Microsoft CRM 3.0 en un servidor con Windows Server 2003, configurar la aplicación (establecer permisos, introducir la información de la empresa...), parametrizarlo (modificaciones que se realizan dentro del propio CRM para adaptarlo al negocio, como crear campos, tablas, relaciones, vistas...) y desarrollar nueva funcionalidad (callout). Además Microsoft CRM estará integrado con Microsoft Office Outlook mediante el conector que proporciona Microsoft de manera que desde nuestro Outlook podremos acceder y utilizar CRM y así se registrarán en CRM los registros de nuestro Outlook que nos interesen.

El desarrollo consistirá en la parte de gestión de proyectos para la solución de CRM de Microsoft, Microsoft CRM 3.0. Esto consistirá en introducir toda la información referente a los proyectos en el CRM para poder evaluar el progreso de los mismos. No hablamos de un conector, sino de desarrollarlo en la propia aplicación.

Por lo tanto Microsoft CRM deberá ser adaptado para poder albergar información de los proyectos y poder realizar un análisis de los mismos introduciendo costes y duraciones reales frente a las previstas.

Los objetivos a cumplir son varios: comprender el funcionamiento interno de la aplicación así como su funcionalidad y comprender mejor en qué consiste la gestión de proyectos y como se controla la evolución de estos.

Por el momento no he encontrado ningún horizontal para esta solución que cubra el área de proyectos, sí que hay conectores para otros productos de Microsoft, por ejemplo el Project y múltiples verticales aunque la mayoría de desarrollos son en inglés.

Índice

1.-Introducción	6
1.1.-Presentación	6
1.2.-Estado del arte	6
2.-Estudio viabilidad	7
2.1.-Introducción.....	7
2.2.-Ventajas	7
2.3.-Inconvenientes.....	8
2.4.-Objetivos	8
2.5.-Recursos.....	9
2.6.-Entorno de programación.....	10
2.7.-Inversión necesaria.....	13
2.8.-Detalle presupuesto	15
2.9.-Evaluación de riesgos.....	15
2.10.-Organización del proyecto	16
2.11.-Características importantes a tener en cuenta	16
2.12.-Planificación del proyecto	17
2.13.-Conclusiones	19
3.-Análisis de funcionalidad	20
3.1.-¿Qué es análisis de funcionalidad?	20
3.2.-Implantar un sistema estándar.....	20
3.3.-Diagrama de contexto	21
3.4.-La seguridad	22
3.5.-Requerimientos funcionales.....	23
3.6.-Otros requerimientos funcionales.....	28
3.7.-Requerimientos no funcionales	29
3.8.-Requisitos software.....	30
4.-Diseño técnico	32
4.1.-Metodología de desarrollo	32

4.2.-Entorno de desarrollo	33
4.3.-Tecnologías utilizadas.....	36
4.4.-Permisos y control de acceso a datos	36
4.5.-Modelo Entidad Relación	37
4.6.-Modelo de datos	37
4.7.-Lógica de negocio	42
4.8.-Interfaz Gráfica	44
4.9.-Vistas.....	51
5.-Plan de pruebas	55
5.1.-En qué consiste	55
5.2.-Tipos de pruebas	56
5.3.-Test 1	57
6.-Conclusiones	63
6.1.-Cumplimiento de objetivos.....	63
6.2.-Retrasos en la planificación	64
6.3.-Líneas de ampliación	64
6.4.-Valoración personal.....	64
7.-Bibliografía.....	66
7.1.-Bibliografía	66
7.2.-Fuentes electrónicas:	66

1.-Introducción

1.1.-Presentación

El proyecto presentado consiste en implantar Microsoft CRM, configurarlo, parametrizarlo y desarrollar la parte de gestión de proyectos para la solución de CRM de Microsoft, Microsoft CRM 3.0. Además esta solución estará integrada en Microsoft Office Outlook. Esto consistirá en realizar una serie de adaptaciones y desarrollos que permitan introducir toda la información referente a los proyectos en el CRM para poder evaluar el estado y progreso de los mismos. No hablamos de un conector, sino de desarrollarlo en la propia aplicación.

1.2.-Estado del arte

En este tipo de soluciones existen muchos verticales pero las soluciones más habituales son conectores con otras aplicaciones aprovechando las posibilidades que tiene Microsoft CRM para integrarse. Así pues pienso que este apartado sería al algo que por el momento no incorpora CRM y puede resultar muy útil para empresas que trabajen con proyectos y que quieran tener en el sistema toda la información que necesario para poder consultar la información y el estado de los proyectos.

2.-Estudio viabilidad

2.1.-Introducción

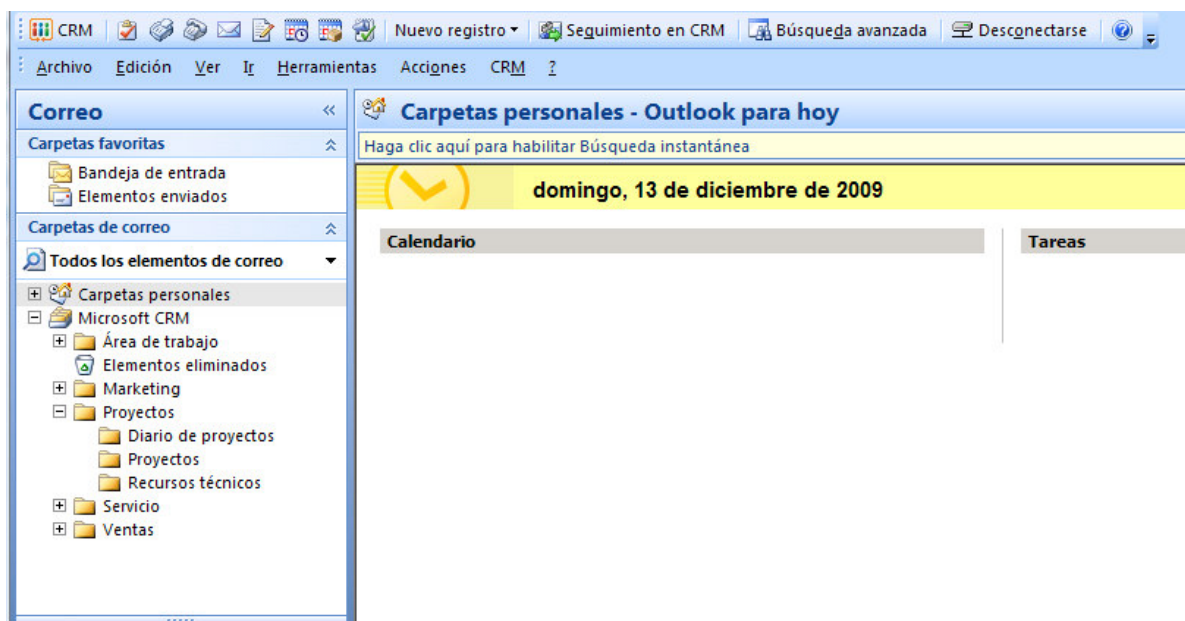
El proyecto que se presenta consiste, por un lado, en instalar y configurar (parametrizar) Microsoft CRM y, por otro lado, en desarrollar la parte de gestión de proyectos para la solución de CRM de Microsoft, Microsoft CRM en la versión 3.0. Además de realizar las acciones necesarias para que nuestro CRM quede completamente integrado en Outlook y de esta forma para el usuario será un cambio menos intrusivo y más amigable. Consistirá en un desarrollo que permitirá introducir toda la información referente a los proyectos para poder evaluar el estado y progreso de los mismos. No hablamos de un conector, sino de desarrollos en la propia aplicación que nos permitirán llevar la gestión de proyectos, funcionalidad que la versión estándar no contempla.

2.2.-Ventajas

Si disponemos de una solución de CRM tendremos mayor control y conocimiento sobre las necesidades y deseos de nuestros clientes de manera que disminuirémos los gastos (por ejemplo en las campañas, se podrán hacer más directas porque tendremos mucha información para segmentar en lugar de envíos más masivos y más costosos) y además aumentará el número de oportunidades cerradas y se aumentará el grado de satisfacción del cliente.

Con el desarrollo de este módulo de gestión de proyectos en Microsoft CRM conseguiremos una mejor gestión de éstos, proporcionando mejoras para aumentar el control, aportar información real y actualizada en todo momento sobre el estado de cada uno de los proyectos. Con la ampliación de la funcionalidad conseguiremos también poder conocer en qué proyecto gasta las horas nuestro personal mediante la imputación de horas a proyectos.

Con la utilización del conector de Microsoft CRM para Microsoft Outlook que proporciona Microsoft, podremos utilizar CRM sin salir de nuestro Outlook de manera que la interfaz será completamente conocida por el cliente y así hacer más fácil la adaptación a esta solución.



2.3.-Inconvenientes

Los usuarios deben formarse para utilizar correctamente la funcionalidad añadida. Ésta quedará reducida a los desarrollos realizados si ya están familiarizados al uso de la aplicación, en caso contrario la formación incluirá funcionalidad estándar de toda la aplicación.

En caso que Microsoft CRM esté personalizado de manera que alguna de las entidades implicadas en el desarrollo ya esté modificada (esto se dará en casos, incluir la gestión de proyectos implicaría realizar tareas adicionales para integrar los nuevos desarrollos).

2.4.-Objetivos

Con este proyecto se pretende conseguir implantar con éxito (instalar, configurar, parametrizar y programar) la solución de Microsoft en el mundo de los CRM, Microsoft CRM 3.0.

Los objetivos a conseguir son centralizar la toda la información que tengamos con el cliente para ofrecerle una mejor atención y más personalizada, mejorar el control que tenemos sobre los proyectos de manera que de una manera fácil y rápida tengamos toda la información que necesitemos sobre el proyecto para, por ejemplo, dar una respuesta más rápida ante una posible llamada del cliente o que permita conocer de una forma sencilla al director del proyecto toda la información que necesite. Por último todo esto se debe conseguir de manera que para el usuario el impacto sea el mínimo posible.

A nivel personal también hay unas metas a alcanzar. Detrás de este proyecto hay muchas ganas e inquietud por ampliar conocimientos en el mundo del CRM y conocer cómo funciona una empresa en cuanto a la gestión de proyectos se refiere, es por esto que he elegido este proyecto. Por un lado, al finalizar el proyecto tengo que conocer más a fondo la funcionalidad de esta herramienta y también a nivel de conceptos y de negocio sobre qué es, cómo funciona y de qué manera puede ayudar a una empresa un software de este tipo. Otro de los objetivos a cumplir es aprender a parametrizar Microsoft CRM (son "desarrollos" realizados dentro de la propia aplicación con herramientas que ya incluye Microsoft CRM), aprender a configurarlo (asignación de permisos, administración de unidades de negocio, unidades de medida, contadores...) y

por último y el más desconocido por mi hasta el momento, ampliación de la funcionalidad mediante callouts que son ficheros con extensión .dll donde se alberga el código que desarrollemos, programación en .NET que nos permitirá crear código para adaptar o crear lógica de negocio que se adapte de la mejor manera posible.

A modo de resumen, a continuación hay un listado con las acciones a llevar a cabo que nos permitirán conseguir los objetivos fijados:

1	Instalar con éxito Microsoft CRM 3.0
2	Desarrollar el módulo de proyectos mediante callouts
3	Permitir la imputación de horas
4	Interfaz de gran usabilidad y sencillez
5	Mantener histórico de proyectos
6	Ampliar conocimientos sobre Microsoft CRM
7	Conocer a nivel de conceptos y de negocio el mundo de los CRM
8	Conocer cómo se gestionan los proyectos internamente en una empresa
9	Realizar la integración con Outlook

2.5.-Recursos

Los recursos necesarios de tipo **software** se detallan a continuación:

Pc Cliente:

Software requerido para los usuarios que quieran ejecutar CRM, que se conecten al servidor de CRM:

Windows 98, 98 Second Edition, Millenium Edition, 2000 SP4, XP o Vista

Navegador Internet Explorer 6.0 SP1 o 7.0

Microsoft Office 2003 o 2007 (requerido para la integración Con Outlook)

Microsoft Data Access Components (MDAC) 2.8 SP1 o superior

Servidor

Software requerido para la correcta ejecución de Microsoft CRM:

Windows Server 2000 o 2003

Microsoft Data Access Components (MDAC) 2.8

Internet Information Services (IIS) 6.0 o superior

SQL Server 2000 SP4 o 2005

Microsoft Exchange Server 2000 o 2003

2.6.-Entorno de programación

Para la realización de los desarrollos, como lenguajes y herramientas de programación se utilizará programación en lenguaje .NET y las herramientas de personalización que incluye el propio producto el cuál soporta código en el lenguaje JavaScript para desarrollar scripts.

Los recursos necesarios tipo hardware son los siguientes:

Recursos recomendados cliente

Procesador: Intel Pentium IV

Memoria RAM: 512

Tarjeta de red: 10/100 Mb

Disco duro: 20Gb

DVD-Rom

Monitor

Teclado y ratón

Recursos recomendados servidor CRM de 1 a 25 usuarios

Para una instalación de hasta 25 usuarios se requiere un único servidor donde alojar la Aplicación, el servidor Web y la base de datos con una configuración similar a la siguiente:

Procesador: Dual 1.5 Ghz Intel Pentium Xeon

Memoria RAM: 1Gb

Controlador: SCSI con RAID 5

Tarjeta de red: 10/100 Mb

Disco duro: 60Gb

DVD-Rom

Monitor

Teclado y ratón

Recursos recomendados servidor CRM de 25 a 100 usuarios

En este caso se requieren 2 servidores, uno para la aplicación junto con el servidor web y otro para la base de datos que serán suficientes para manejar con agilidad hasta 100 usuarios.

Procesador: Dual 2 GHz Intel Pentium Xeon

Memoria RAM: 2 Gb

Controlador: SCSI con RAID 5

Tarjeta de red: 10/100 Mb

Disco duro: 150 Gb

DVD-Rom

Monitor

Teclado i ratón

Recursos recomendados servidor CRM de 100 a 250 usuarios

Para poder soportar hasta 250 usuarios con agilidad requeriremos 3 servidores, uno para la aplicación, otro para el servidor web y un tercero para la base de datos.

Procesador: Dual 3.0 Ghz Intel Pentium Xeon

Memoria RAM: 4 Gb

Controlador: SCSI con RAID 5

Tarjeta de red: 10/100 Mb

Disco duro: 250 Gb

DVD-Rom

Monitor

Teclado y ratón

Recursos recomendados servidor CRM más de 250 usuarios

En este caso serán necesarios 5 servidores, dos para la aplicación, en alta disponibilidad, 1 para el servidor web y otros 2 para la base de datos.

Procesador: Dual 3,5 Ghz Intel Pentium Xeon

Memoria RAM: 6Gb

Controlador: SCSI con RAID 5

Tarjeta de red: 10/100 Mb

Disco duro: 300 Gb

DVD-Rom

Monitor

Teclado y ratón

Los recursos humanos necesarios para realizar el proyecto son los siguientes:

Técnico programador

Técnico de sistemas

Equipo de pruebas

Técnico de mantenimiento

2.7.-Inversión necesaria

Los costes que se deben asumir en el caso que no se disponga de infraestructura ni de software son los siguientes. En el caso que se disponga de algún recurso software o hardware se podrán utilizar y no será necesario comprar otro.

Costes de material

Recurso	Coste total	Coste mensual
*Servidor (hasta 25 usuarios)	2200€	
*Servidor (hasta 100 usuarios)	2500€	
*Servidor (hasta 250 usuarios)	2800€	
*Servidor (más de 250 usuarios)	3200€	
Conexión internet		50€
Windows Server 2003	450€	
SQL Server 2005	966€	
Amortización PC programador*	24€	
Total (con servidor hasta 25u.)	3616€	

**Nota: las amortizaciones es de tipo lineal y con una duración de 4 años*

Costes de personal

nº	Descripción de la actividad	Duración (horas)	Coste (€)
1	Estudio de viabilidad	20	400
2	Análisis y estudio de la gestión de proyectos	30	600
3	Planificar la instalación y configuración de Microsoft CRM 3.0	10	200
4	Realizar la instalación y la configuración de Microsoft CRM 3.0	20	400
5	Analizar Microsoft CRM	26	520
6	Estudiar las modificaciones a realizar Microsoft CRM	20	400
7	Realizar el desarrollo	50	1000
8	Realizar pruebas	20	400
9	Implantación	16	320
#	Total	216	4040

Nota: los costes se basan sobre una tarifa de 20€/hora ya que utilizaríamos recursos junior dada la baja complejidad de los desarrollos.

2.8.-Detalle presupuesto

Servidor

El servidor presupuestado corresponde con las características del servidor recomendado hasta 100 usuarios.

Concepto	Importe
Procesador: Dual 2.00 Ghz Intel Pentium Xeon Memoria RAM: 2Gb Controlador: SCSI con RAID 5 Tarjeta de red: 10/100 Mb Disco duro: 150Gb DVD-Rom Monitor Teclado y ratón	2200,00 €
Desarrollo y puesta en marcha aplicación	3600,00 €
Total	5800,00 €

2.9.-Evaluación de riesgos

Hay que tener en cuenta que se trata de desarrollos no oficiales y que, por lo tanto, podría haber problemas con futuras versiones y también pueden haber conflictos si al incorporar los desarrollos se hace en una instalación de CRM con personalizaciones realizadas anteriormente.

Dado que adaptamos una aplicación desarrollada por terceros (en este caso Microsoft) nos podemos encontrar con limitaciones a la hora de hacer las personalizaciones y los diseños de los formularios.

La disponibilidad de información sobre la gestión de proyectos deberá ser suficiente como para poder llevar a cabo el proyecto con un mínimo de seguridad.

2.10.-Organización del proyecto

Para la realización de este proyecto se seguirá el esquema siguiente:

- 1. Planificar la instalación de Microsoft CRM 3.0.**
- 2. Documentar la configuración inicial.**
- 3. Realizar la instalación y la configuración**
- 4. Obtener una lista de la información que habría que almacenar.**
- 5. Establecer requisitos funcionales del sistema.**
- 6. Establecer un conjunto de requerimientos no funcionales del sistema.**
- 7. Estudiar alternativas.**
- 8. Planificación de las etapas de desarrollo del proyecto.**
- 9. Buscar recursos software y hardware para la realización de las tareas.**
- 10. Realización de los desarrollos y las parametrizaciones.**

2.11.-Características importantes a tener en cuenta

Para no tener problemas de mantenimiento en un futuro con los desarrollos realizados en este proyecto es importante tener en cuenta algunos aspectos como actualizaciones o futuras ampliaciones para facilitar futuras actualizaciones de la versión del software. Otras características importantes son la portabilidad a nuevas versiones y la documentación relacionada con los cambios en la funcionalidad.

Modelo de desarrollo:

El modelo de desarrollo que se seguirá en este proyecto será un modelo lineal el cual tiene como características principales:

1. Estructura dividida en módulos.

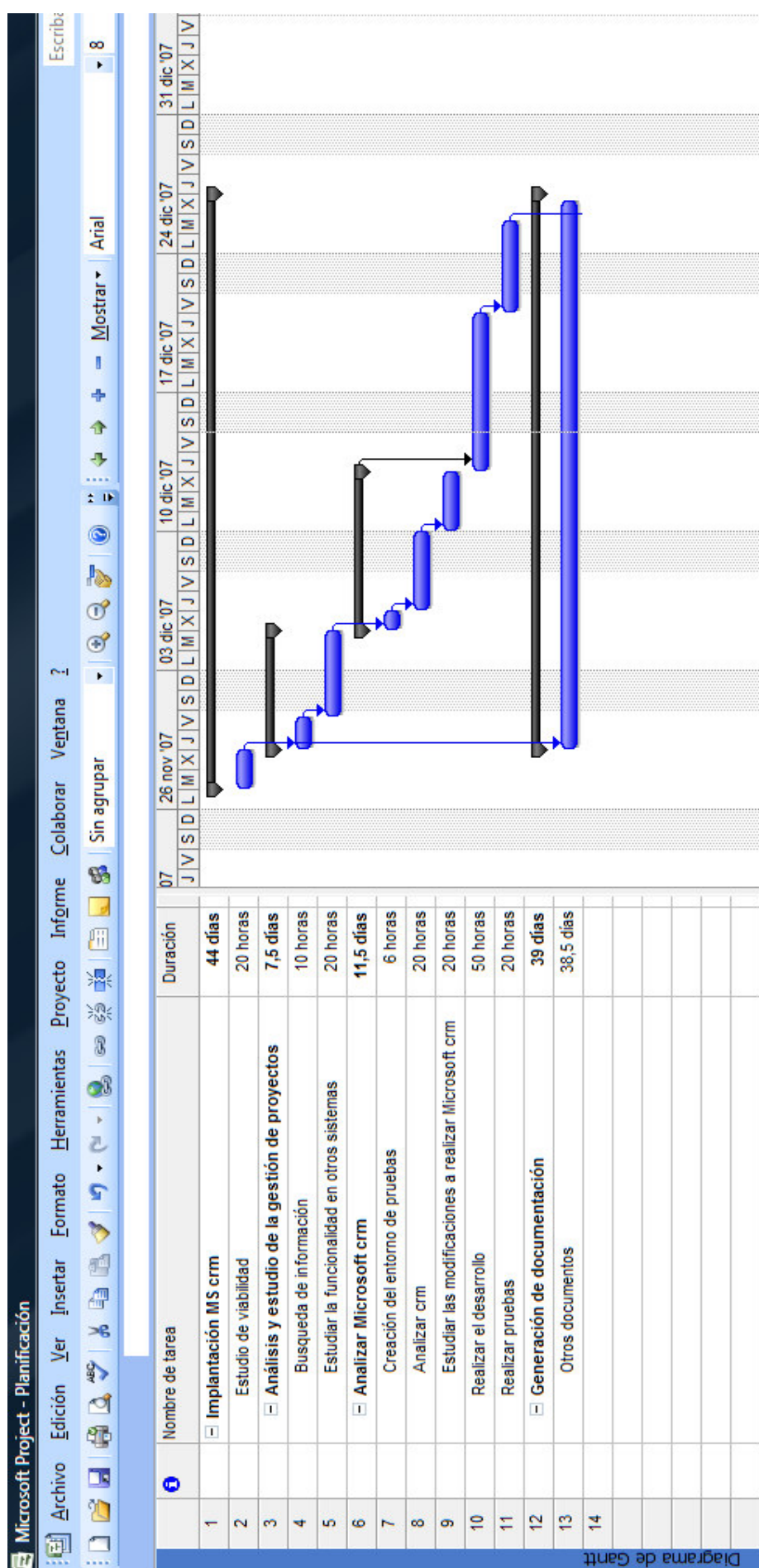
2. El software se ha de diseñar por etapas.

3. Las etapas se pueden realizar por separado sin afectar el desarrollo de las otras.

Se ha elegido este modelo por la fácil estructuración y por la sencillez en llevarlo a cabo.

2.12.-Planificación del proyecto

El proyecto se distribuirá en diferentes etapas detalladas a continuación:



Y a continuación el detalle de la planificación:

Como ocurre en todo proyecto lo primero es realizar el estudio de viabilidad para evaluar las ventajas y los inconvenientes de llevarlo a cabo y decidir si resulta rentable realizarlo. Después llega la fase donde hay que documentarse para conocer las posibilidades. Del mismo modo, habrá que estudiar la aplicación donde se realizarán los desarrollos. Llegados a este punto, es el momento de ver y concretar qué y cómo hay que modificar CRM para conseguir nuestros objetivos. Una vez tenemos claro qué hacer y cómo hay que hacer los desarrollos. Cuando han finalizado hay que realizar las pruebas para comprobar que el sistema se comporta como esperamos. Por último, pasar todos los desarrollos y modificaciones del sistema de pruebas al sistema real.

nº	Descripción de la actividad	Duración (horas)
1	Estudio de viabilidad	20
2	Análisis y estudio de la gestión de proyectos	30
3	Analizar Microsoft CRM	26
4	Estudiar las modificaciones a realizar Microsoft CRM	20
5	Realizar el desarrollo	50
6	Realizar pruebas	20
7	Implantación	16

2.13.-Conclusiones

Con la incorporación del módulo de gestión de proyectos a Microsoft CRM conseguiremos tener mejor control y supervisión de los proyectos ya que los podremos tener en el sistema. Además, al imputar los costes podremos obtener informes para ver las posibles desviaciones que puedan tener los proyectos. También nos permitirá tener una base de datos actualizada y contar con un histórico para consultar proyectos antiguos con rapidez y sencillez. Y todo esto lo podremos hacer desde nuestro Outlook, de manera que para el usuario resulta más sencillo utilizar esta solución.

Por todo lo comentado en este apartado, a raíz de los estudios y las investigaciones realizadas, teniendo en cuenta los costes, los recursos necesarios, el hardware y software necesario y la estimación de horas necesarias. Por todo esto veo este proyecto como un proyecto viable y que será realidad una vez finalice el proyecto.

3.-Análisis de funcionalidad

3.1.-¿Qué es análisis de funcionalidad?

El análisis de funcionalidad consiste en un conjunto de técnicas y procedimientos que permiten definir un proyecto de software. Se entiende las necesidades de nuestros clientes requeridas para el diseño del sistema. En esta etapa se logra claridad sobre lo que desea el usuario y la forma en la cual se le va a presentar la solución que está buscando. Las necesidades se dividen en dos grandes grupos, requerimientos funcionales (que son condiciones o capacidades de un sistema requeridas por el usuario para resolver un problema o alcanzar un objetivo) y requerimientos no funcionales (condiciones o capacidades que debe poseer un sistema para satisfacer un contrato, un estándar, una especificación u otro documento formalmente impuesto). En resumen, los requerimientos son el punto de acuerdo entre el cliente y el proyecto de desarrollo de software, este entendimiento es necesario para poder construir software que satisfaga las necesidades de nuestro cliente.

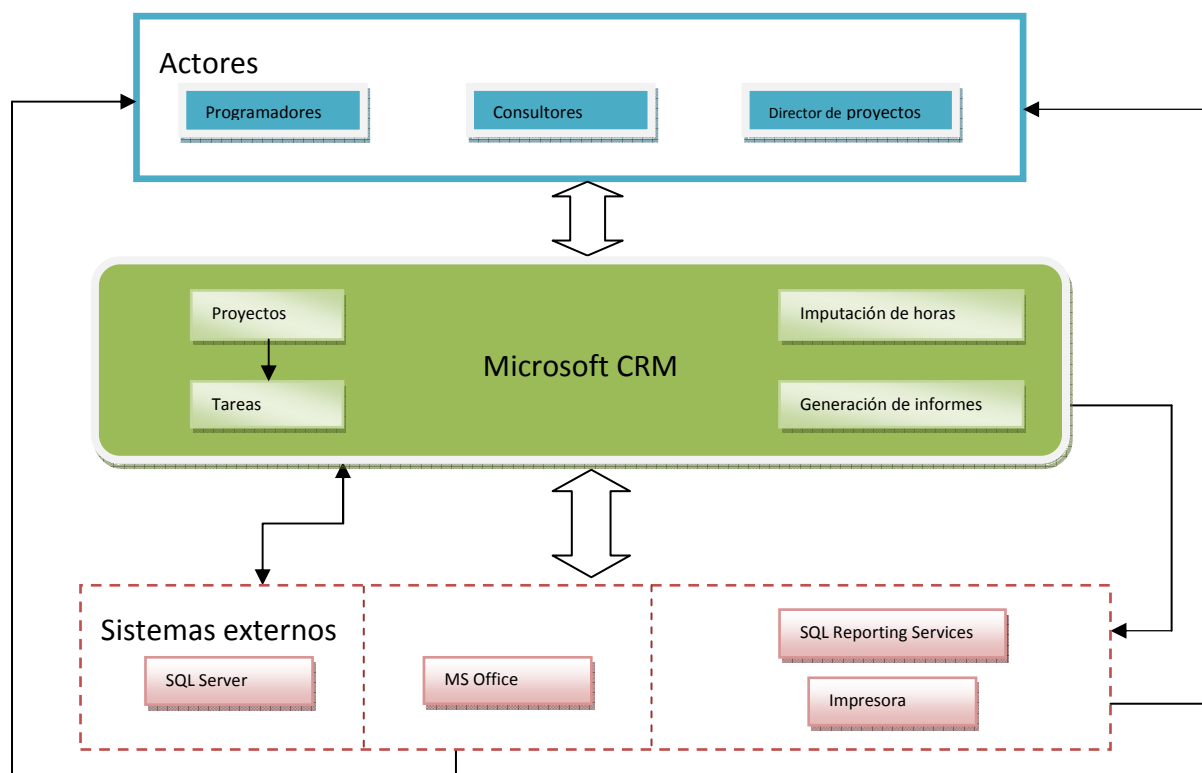
Este análisis consiste en obtener información acerca de lo que los usuarios desean, esto suele ser en visitas al cliente, recabando información sobre cómo el cliente desea que funcione el software. Después clasificarlos estableciendo jerarquías para poder estructurar estos "deseos" de los usuarios, como requerimientos funcionales, no funcionales, de entorno... entre otros. Por otro lado es importante que se utilice un lenguaje que nos permita especificar los requerimientos precisamente y sin ambigüedad. Después de todo esto, debemos numerar todos estos requerimientos de forma única para que queden claramente definidos.

3.2.-Implantar un sistema estándar

Al implantar un software estándar estamos reduciendo el coste del proyecto ya que incorporará métodos y herramientas para parametrizar la aplicación con lo que reducimos los desarrollos. Disminuirá el riesgo del proyecto al utilizar una base sólida. También dispondremos de actualizaciones de versiones. Al finalizar su vida útil podríamos obtener algún tipo de bono en función del nuevo software que se adquiera. Además podremos consultar con otros clientes qué tal funciona y así partir con cierta seguridad.

Para los requerimientos funcionales se ha utilizado una codificación, que es CRMCUxxx (para el primero CRMCU001), donde CRM es porque el proyecto es entorno a Microsoft CRM, CU porque estamos numerando casos de uso y xxx es la numeración, que irá variando, incrementándose positivamente para cada caso de uso.

3.3.-Diagrama de contexto

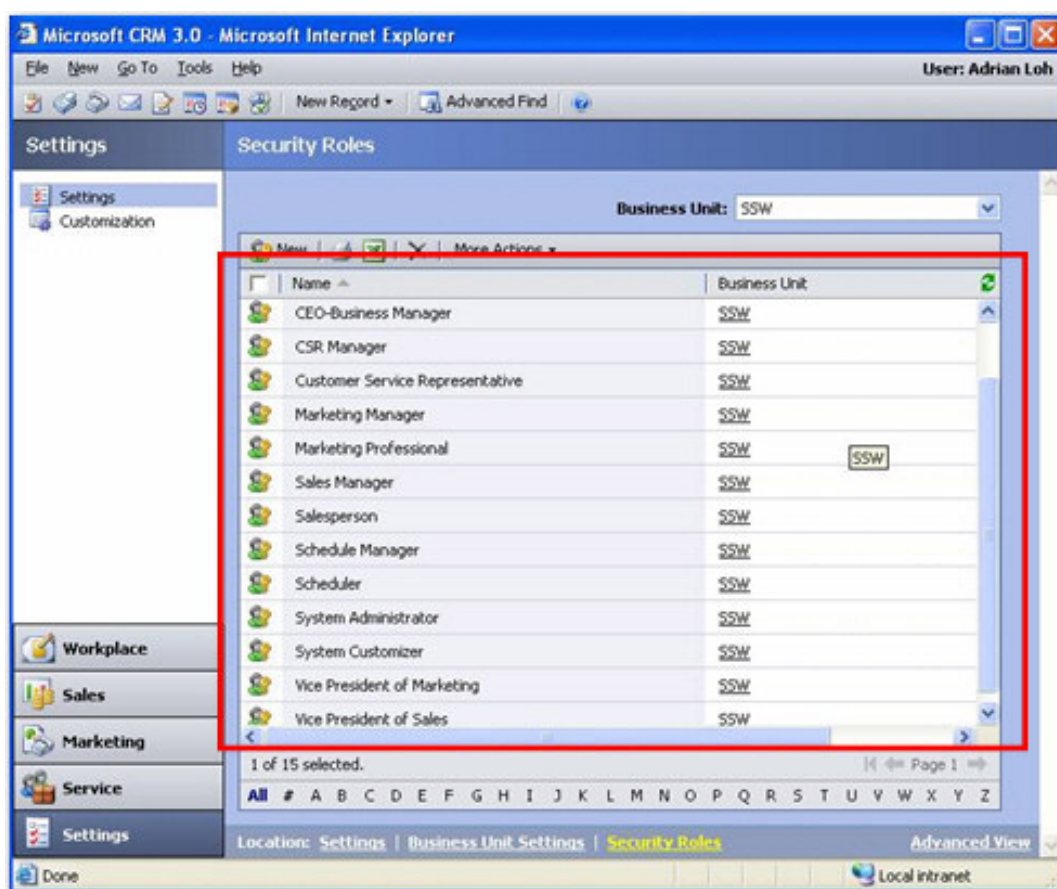


Los usuarios del sistema (en cuanto a gestión de proyectos se refiere) que en este caso es personal técnico, programadores, consultores y directores de proyecto, interactúan directamente con el sistema (Microsoft CRM) tanto para obtener información como introduciendo datos nuevos referente a los proyectos, los cuales están divididos en tareas. Para los directores de proyectos estarán disponibles diversos informes para ver, por ejemplo, desviaciones. Los técnicos, consultores y programadores dispondrán de diversos formularios para introducir nuevos proyectos, nuevas tareas, introducir las horas gastadas en cada tarea o para la generación de informes. El sistema requiere tener instalado Microsoft SQL Server para albergar y gestionar la base de datos del sistema CRM y Microsoft SQL Reporting Services para la generación de los informes. Por otra parte, para poder imprimir listados o informes por ejemplo, utiliza una impresora. Por último, Microsoft CRM es compatible con Microsoft Office para permitir que se puedan importar datos desde CRM a Microsoft Office, por ejemplo para tener listados en Excel o informes en Word, o importar datos desde Microsoft Office a Microsoft CRM para poder realizar importaciones masivas de clientes, contactos o compañías.

3.4.-La seguridad

La seguridad y el acceso en Microsoft CRM se gestionan mediante dos sistemas. Por un lado aprovecha la seguridad que proporciona Active Directory para la autenticación, el control de acceso a la aplicación. Hasta aquí tenemos controlado el acceso pero falta por ver cómo se encarga Microsoft CRM de asegurar que cada usuario únicamente vea lo que tiene que ver. Se basa en privilegios (llamados roles de seguridad) que a su vez son basados en funciones de seguridad (es la unidad mínima para definir permisos, se decide si se puede acceder a una tabla y con qué permisos: lectura, escritura, edición y borrado). Podemos crear roles, utilizar o modificar los que vienen definidos. Es decir, al crear un rol lo que hacemos es definir si en ventas un usuario puede crear ofertas, o modificarlas (¿las suyas, las de su unidad de negocio o todas?), modificar o eliminar. De igual manera lo haremos para cada una de las áreas de Microsoft CRM.

En la siguiente imagen vemos un listado de roles definidos en Microsoft CRM:



Desde el cliente de Outlook también disponemos de estas medidas de seguridad ya que el cliente logado en CRM será el usuario que inicie sesión en Windows, igual que cuando accedemos desde el cliente web.

3.5.-Requerimientos funcionales

El primer gran grupo de los requerimientos, los requerimientos funcionales, que son aquellos relacionados con la funcionalidad, establecen los comportamientos del sistema, definen el comportamiento interno del software: cálculos, manipulación de datos y otras funcionalidades específicas. Un requisito funcional típico, como mínimo, tiene un nombre, un código único y una descripción. Se detallan a continuación:

Nombre: Crear proyecto

Código: CRM CU001

Descripción: Este caso de uso describe como los usuarios crean proyectos.

Actores: Responsable de proyectos

Flujo principal:

-Si tiene permisos para crear proyectos se le mostrara el formulario de creación.

Flujos alternativos:

-Si no dispone de permisos suficientes se le presentara mensaje de error.

Nombre: Modificar proyecto existente

Código: CRM CU002

Descripción: Este caso de uso describe como los usuarios pueden modificar proyectos.

Actores: Responsable de proyectos, Consultores

Flujo principal:

-Si tiene permisos se le permitirá abrir el proyecto para modificarlo. Una vez abierto puede completar la información que necesite.

Flujos alternativos:

-Si no dispone de permisos no se le abrirá el formulario del proyecto y se le informara por pantalla del error.

-Si en lugar de completar información va a actualizar, deberá tener permisos, no solo de lectura y escritura sino también permisos para poder actualizar.

-Si al completar/modificar información se introduce alguna que

no sea válida, no se guardará y se le mostrará el error al usuario para que lo solucione.

Nombre: Borrar proyecto

Código: CRM CU003

Descripción: Este caso de uso describe como los usuarios pueden borrar proyectos

Actores: Responsable de proyectos

Flujo principal:

-Si dispone de permisos se le permitirá borrarlo. Se pedirá confirmación.

Flujos alternativos:

-Si no dispone de los permisos necesarios, se le informará al usuario.

Nombre: Consultar estado

Código: CRM CU004

Descripción: Este caso de uso describe como los usuarios pueden consultar el estado del proyecto.

Actores: Responsable de proyectos, Consultores, Programadores, Analistas

Flujo principal:

-Si el usuario tiene permisos de lectura podrá consultar la información que requiera. Se le mostrará listado de proyectos y la posibilidad de abrirlos.

Flujos alternativos:

-En lugar de información el usuario consultar informes (reports). Para este caso también se requiere que disponga de permisos suficientes.

Nombre: Realizar búsquedas sobre proyectos

Código: CRM CU005

Descripción: Este caso de uso describe como los usuarios pueden buscar por determinada información de los proyectos.

Actores: Responsable de proyectos, Consultor, Programador, Analista,...

Flujo principal:

-Si dispone de suficientes permisos se le mostrará el listado y los campos de búsqueda para que la pueda llevar a cabo.

Flujos alternativos:

-La búsqueda se puede realizar, también, mediante búsqueda avanzada (filtros).

Nombre: Creación de actividades

Código: CRM CU006

Descripción: Este caso de uso describe como los usuarios pueden crear actividades o notas asociadas a un proyecto.

Actores: Responsable de proyectos, Consultores, Programadores, Analistas

Flujo principal:

-Desde el formulario del proyecto se pueden crear actividades y/o notas asociadas a ese proyecto, esto será posible si el usuario dispone de los permisos necesarios.

Flujos alternativos:

-A parte de crear las actividades, si es necesario, se pueden asignar a otros usuarios.
-Completar actividades de los proyectos.

Nombre: Imputación de horas

Código: CRM CU007

Descripción: Este caso de uso describe como los usuarios pueden imputar horas en los proyectos.

Actores: Director de proyectos, Consultores, Programadores, Analistas

Flujo principal:

-Si tiene permisos, podrá abrir el proyecto y acceder al formulario para imputar horas.

Flujos alternativos:

-Si no tiene permisos se le notificará del error.

Nombre: Completar un proyecto

Código: CRM CU008

Descripción: Este caso de uso describe como los usuarios pueden dar por finalizado un proyecto

Actores: Director de proyectos

Flujo principal:

-Si dispone de los permisos necesarios, podrá dar por completado el proyecto con éxito (ya no se permitirá añadir más información al mismo).

Flujos alternativos:

-Si no dispone de permisos no se le permitirá completar proyectos.

-El proyecto puede completarse con fracaso (si se da el caso que los costes reales son superiores a los presupuestados).

Nombre: Modificar actividad

Código: CRM CU009

Descripción: Este caso de uso describe como los usuarios pueden modificar una actividad relacionada con un proyecto.

Actores: Director de proyectos

Flujo principal:

-Si dispone de los permisos necesarios, podrá modificar la actividad que se precise para actualizar la información que sea necesaria.

Flujos alternativos:

-Si no dispone de permisos no se le permitirá guardar el formulario de actividad y se notificará al usuario del error.

Nombre: Borrar actividad

Código: CRM CU010

Descripción: Este caso de uso describe como los usuarios pueden

eliminar actividades.

Actores: Director de proyectos

Flujo principal:

-Si dispone de los permisos necesarios, podrá realizar la acción de borrar actividad sin que se le muestre error.

Flujos alternativos:

-Si no dispone de permisos no se le permitirá eliminar la actividad.

Nombre: Generar informes

Código: CRMCU011

Descripción: Este caso de uso describe como los usuarios pueden generar informes.

Actores: Director de proyectos

Flujo principal:

-Si dispone de los permisos necesarios, podrá generar informes de listados sobre el proyecto.

Flujos alternativos:

-Si no dispone de permisos no se le permitirá generar informes.

Nombre: Asignar tareas

Código: CRMCU012

Descripción: Este caso de uso describe como los usuarios pueden asignar tareas de proyectos a recursos.

Actores: Director de proyectos

Flujo principal:

-Si dispone de los permisos necesarios, podrá asignar tareas a los recursos que se precise.

Flujos alternativos:

-Si no dispone de permisos no se le permitirá asignar tareas.

Nombre: Exportar a Microsoft Excel

Código: CRMCU013

Descripción: Este caso de uso describe como los usuarios pueden exportar datos a Excel.

Actores:

Flujo principal:

-Si dispone de los permisos necesarios, podrá exportar a Excel listados e información detallada sobre los proyectos.

Flujos alternativos:

-Si no dispone de permisos no se le permitirá realizar ninguna exportación.

-El proyecto puede completarse con fracaso (si se da el caso que los costes reales

Nombre: Flujos de trabajo

Código: CRMCU014

Descripción: Este caso de uso describe como los usuarios pueden ejecutar flujos de trabajo

Actores: Director de proyectos, Consultores, Programadores, Analistas

Flujo principal:

-Si dispone de los permisos necesarios podrá ejecutar los flujos de trabajo asignados a la entidad del registro o registros seleccionados.

Flujos alternativos:

-Si no dispone de los permisos necesarios no le aparecerá la opción de ejecutar los flujos.

3.6.-Otros requerimientos funcionales

Al crear líneas de diario aumentan las cantidades consumidas de la tarea

Al eliminar líneas de diario disminuirán las cantidades consumidas de la tarea

Al crear líneas de diario aumentan las cantidades consumidas del proyecto

Al eliminar líneas de diario disminuirán las cantidades consumidas del proyecto

Los costes se actualizarán en base a un precio hora de 40€ que se puede parametrizar

3.7.-Requerimientos no funcionales

El segundo gran grupo de los requerimientos: Los requerimientos no funcionales, que hacen referencia a las especificaciones del sistema como pueden ser sus propiedades y restricciones.

Éstos se dividen en diferentes agrupaciones dependiendo del origen del mismo.

Requerimientos corporativos

Estos requerimientos son una consecuencia de las políticas y procedimientos de la organización cómo, por ejemplo, estándares usados en los procesos, los requerimientos de implementación, etc.

El software a utilizar para el desarrollo de este proyecto debe ser de Microsoft y, por lo tanto, el software de CRM elegido para realizar los desarrollos será Microsoft CRM 3.0

Para realizar las personalizaciones se utilizará el entorno de desarrollo que proporciona Microsoft CRM.

Para albergar las bases de datos se precisa Microsoft SQL Server 2000 o 2005 y SQL Server Reporting Services 2000 o 2005.

Para realizar exportaciones se debe disponer de Microsoft Office 2003 o 2007.

Para realizar la integración con Outlook debemos disponer de Microsoft Office Outlook.

Requerimientos *técnicos*

El segundo subgrupo de los requerimientos de tipo no funcional son los técnicos, los cuáles especifican el comportamiento del producto cómo, por ejemplo, rapidez de ejecución o fiabilidad.

Para simular el entorno de producción del sistema CRM se utilizará una máquina virtual con Windows Server 2003 creada con Microsoft Virtual PC 2007.

Al utilizar una máquina virtual, el equipo huésped debe disponer de suficientes recursos hardware para el correcto funcionamiento del servidor. Entiéndase como correcto funcionamiento unos tiempos de espera hasta un 130% de los tiempos normales.

Realizar los desarrollos de manera que se asemejen a la interface estándar de Microsoft CRM.

Para facilitar el mantenimiento se utilizará una *Guide-Lines* (son las pautas seguidas para la realización de las personalizaciones) con el fin de facilitar a terceros el mantenimiento o nuevos desarrollos.

3.8.-Requisitos *software*

En un escenario básico se describe una implementación con un servidor de CRM (Microsoft CRM), un servidor de bases de datos (Microsoft SQL Server) un servidor de correo (Microsoft Exchange) y un controlador de dominio (Active Directory).

Service Pack 3 de Microsoft Windows 2000 Server (o Advanced Server) o versiones posteriores

Microsoft Message Queue Server (también conocido como MSMQ)

Servicio de Index Server

Microsoft Internet Information Services

Microsoft Data Access Components (MDAC) 2.7a

Microsoft XML Core Services 4.0 Service Pack 1

Microsoft .NET framework Service Pack 2

Active Directory (AD) de Microsoft

Microsoft Office 2003 o 2007 para la integración con Microsoft CRM

Además de los requisitos especificados en los apartados anteriores, si utilizamos el conector para Exchange (Microsoft CRM Exchange E-mail Router), tendremos que cumplir adicionalmente otros requisitos:

Service Pack 3 de Microsoft Windows 2000 Server (o Advanced Server) o versiones posteriores

Microsoft Exchange 2000 Server Service Pack 3

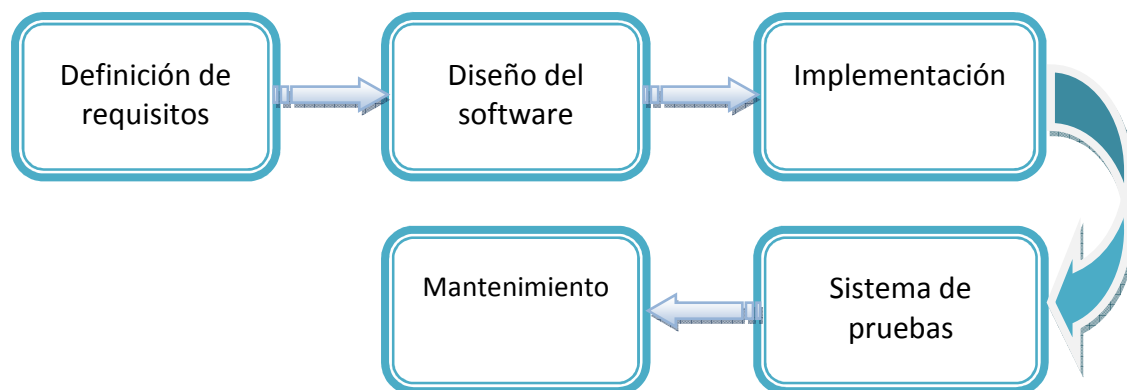
Microsoft Internet Explorer 5.5 Service Pack 2 o versiones posteriores

4.-Diseño técnico

4.1.-Metodología de desarrollo

Para la parametrización realizada se elegirá un prefijo "proy" igual para todos los campos y tablas creadas para poder reconocer las modificaciones realizadas en este proyecto. Por ejemplo el nombre del campo *cantidad* sería *proy_cantidad*, de la misma manera que el nombre de la tabla tarea sería *proy_tarea*.

Los pasos para hacer el desarrollo deberán seguir el esquema siguiente:



En la primera fase tenemos la definición de requisitos que nos sirve para identificar y documentar qué modificaciones hay que llevar a cabo para conseguir nuestro objetivo. Una vez tenemos claro qué hay que hacer el siguiente paso que es hacerlo, diseñar el software. Ahora hay que poner en funcionamiento los desarrollos, es decir, implementarlos. La siguiente fase es hacer el sistema de pruebas para comprobar que los desarrollos cumplen los requerimientos, son estables, no dan errores... Por último, el mantenimiento hay que tenerlo en cuenta por si hay que hacer cambios o adaptaciones, bien por actualizaciones de versión o bien por mejoras del área.

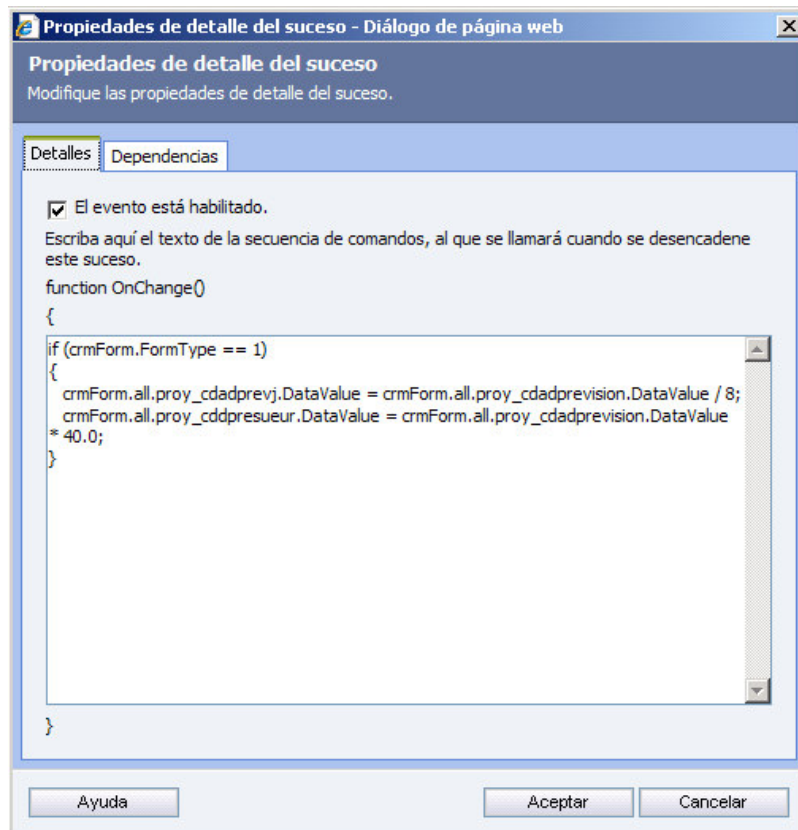
En este proyecto el desarrollo del software seguirá un esquema lineal, desarrollando por partes cada una de las funciones hasta completarla con éxito para después empezar con la siguiente función, continuando con la misma dinámica hasta completar todo el desarrollo. Además se intentará simplificar el proceso siguiendo un esquema en la nomenclatura de las variables y funciones de modo que sea simple para otro hipotético desarrollador retomar el proyecto y poder conocer la utilidad de las funciones y variables ya creadas.

4.2.-Entorno de desarrollo

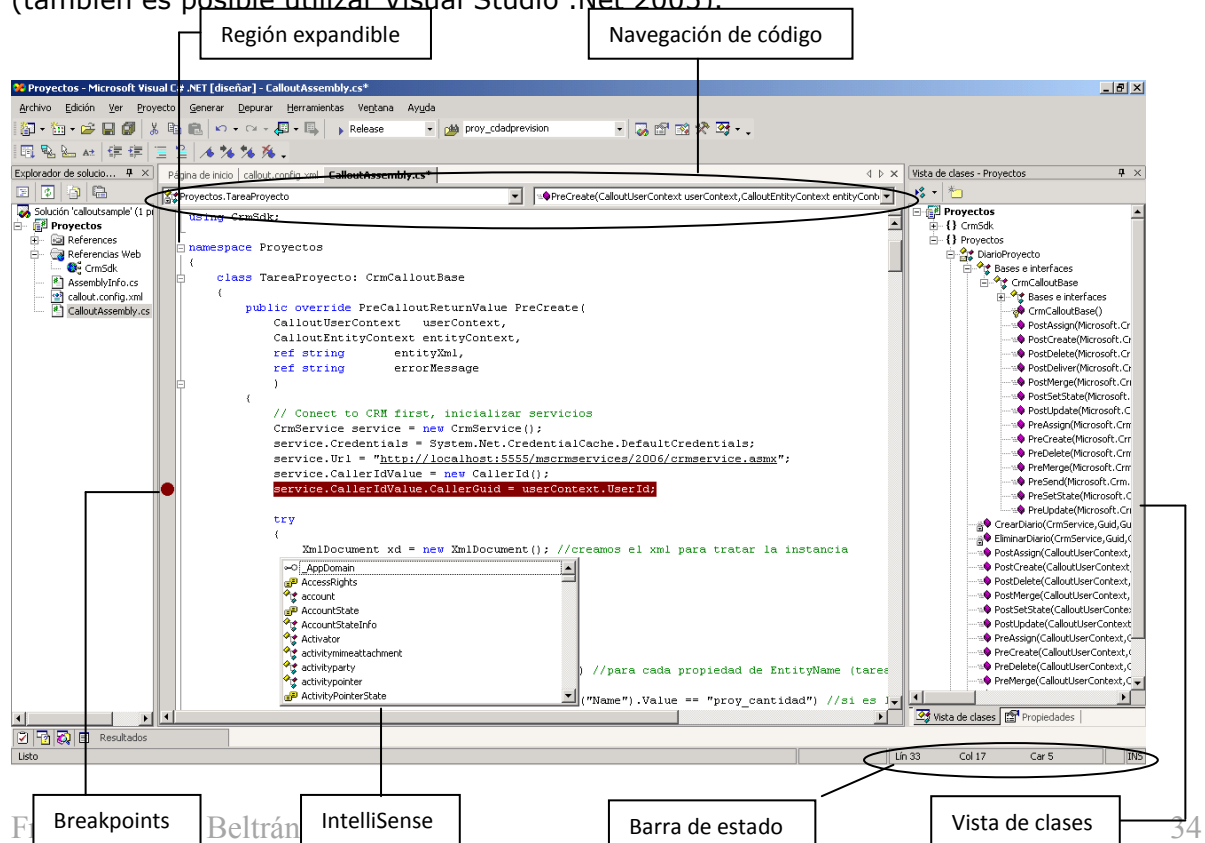
Los desarrollos a realizar en este proyecto se podrían dividir en dos grandes grupos i, por tanto tenemos 2 entornos de desarrollo diferenciados.

Por un lado tenemos las personalizaciones realizadas desde el propio CRM también llamadas client-side porque se ejecutan en el navegador del usuario. Éstas se realizan desde el área de configuración del menú principal de CRM, que son la creación y edición de tablas campos y formularios y una zona específica para escribir código Java Script. En este entorno se incluye la creación de entidades, que son las tablas de la base de datos. También incluimos los campos para estas nuevas entidades y los creados en las entidades ya existentes. La creación y/o modificación de vistas, que son una forma concreta de visualizar los registros de una entidad en concreto con filtros y con los campos que se deseen visualizar.

Y por ultimo en este grupo incluimos los desarrollos con código JavaScript que en este caso lo utilizaremos para hacer cálculos sobre campos que estén en la misma entidad aunque puede tener muchas más utilidades. Todas estas modificaciones comentadas se realizan sobre el propio CRM en el apartado de Personalización de entidades destinado a tal fin.



En el segundo grupo tenemos los desarrollos externos o *server-side*. Se denominan así porque se ejecutan en el servidor. Se trata de uno o varios archivos con extensión .dll que contienen funciones para crear, modificar, asignar, combinar y eliminar registros de las entidades existentes en la base de datos de CRM. Estas .dll's se llaman *callout* y el entorno para realizar estos desarrollos es Visual Studio .Net en la versión 2003 (también es posible utilizar Visual Studio .Net 2005).



En el entorno de desarrollo de Visual Studio .Net 2003, diferenciamos la zona superior o de menús y la zona inferior a ésta que es donde veremos todo lo relativo a nuestro proyecto de desarrollo. Por tanto dejando de lado la zona de menús, de izquierda a derecha, tenemos primero una línea vertical para los puntos de ruptura (o breakpoints) que nos sirve para pausar la aplicación mientras se está ejecutando código y ver nuestro código paso a paso. Justo al lado tenemos un área con los números de línea, que nos proporciona una manera fácil de navegar por nuestro código. En el recuadro central tenemos el espacio destinado a escribir el código con algunas ayudas como los menús inteligentes o “*IntelliSense*” que nos proporciona ayuda para autocompletar los nombres mientras escribimos código.

En Visual Studio también podemos expandir y contraer partes de código para que sólo se muestre aquél código que es necesario en ese momento. En esta área también podemos escribir lo que serán nuestros comentarios XML para la documentación de usuario. La zona que queda inmediatamente por encima del área del código nos es útil para navegar rápidamente permitiendo seleccionar “espacio de nombres.clase” y las funciones. En la parte inferior de la pantalla tendremos la barra de estado, común en la mayoría de software que muestra información como el número de línea, columna y carácter en que nos encontramos o bien si tenemos el bloque numérico activado... El área derecha de Visual Studio la utilizamos para mostrar en la parte superior la vista de las clases de nuestro proyecto o solución. Y en la parte nos muestra un detalle de éstas. El área de la derecha es parametrizable en cuanto a la zona donde se puede mostrar y la información que mostrar.

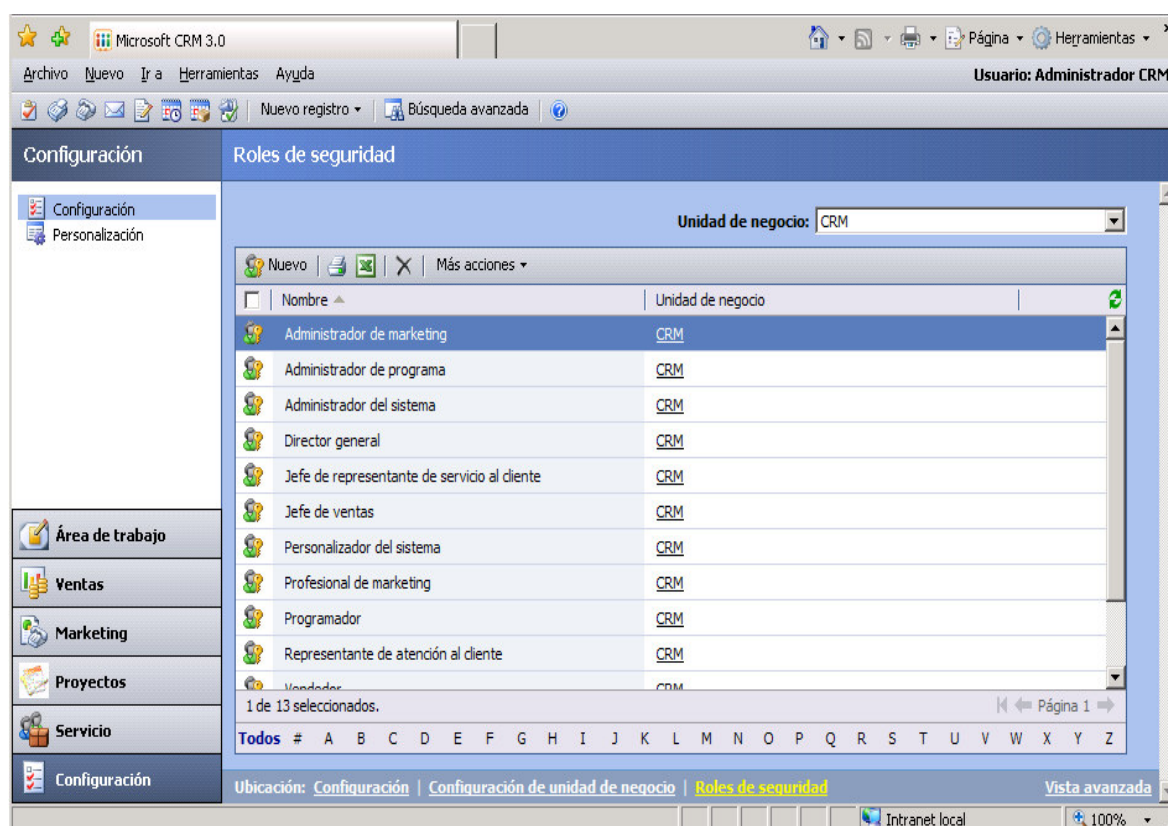
Por último tenemos que modificar un fichero llamado *callout.xml* el cual nos permitirá decirle a nuestro CRM que hemos hecho modificaciones para unas tablas en concreto y dónde encontrarlas (ruta y fichero). Éste archivo lo modificaremos desde Visual Studio pero se podría utilizar perfectamente el bloc de notas, por ejemplo.

4.3.-Tecnologías utilizadas

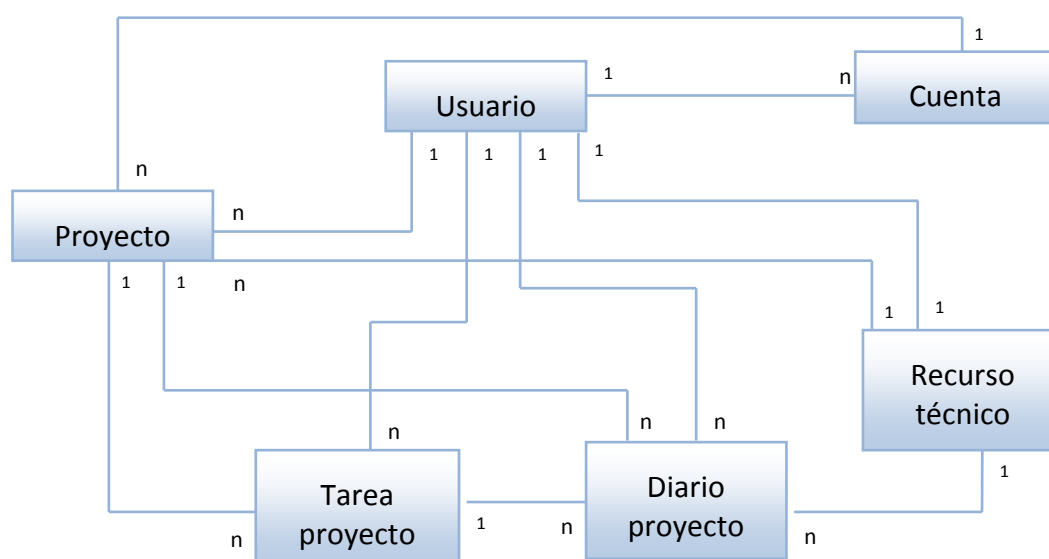
Para el desarrollo de este proyecto se utilizará, en lo que a software se refiere, Visual Studio .NET 2003 (licencia procedente del MSDNAA) y en este caso correrá bajo un Windows Server 2003. Y también utilizaremos el propio CRM ya que la parte de formularios y Java Script se realizan desde el CRM. Emplearemos, en Visual Studio, el lenguaje C# (llamado C Sharp) a partir del cual generaremos un fichero .dll que habrá que copiar dentro la carpeta de CRM, en una ubicación concreta. También tenemos que modificar un fichero que está en lenguaje XML. Mientras que para los desarrollos que realizan en CRM utilizaremos el lenguaje JavaScript para actualizar valores de campos y para la creación y modificación de formularios y campos no es necesario escribir código, nuestro CRM se encargará de hacerlo por nosotros. En cuanto al hardware utilizado será un portátil con procesador Centrino Duo utilizado para hacer correr la máquina virtual donde estará alojado el servidor de CRM y Visual Studio.

4.4.-Permisos y control de acceso a datos

Para gestionar la seguridad en CRM utilizaremos los roles que incorpora el propio CRM. Estos roles son un conjunto de permisos agrupados para facilitar la gestión de la seguridad y así poder definir el acceso a los datos para cada usuario. Para las tablas y campos que creemos también podremos utilizar este método. Mediante los roles definimos, por usuario, a qué entidades tiene acceso de: lectura, escritura, creación, edición, eliminación, asignación... Por otra parte los usuarios tienen que ser usuarios del dominio, es decir que tienen que estar dados de alta en AD y éste también tiene seguridad.



4.5.-Modelo Entidad Relación



En el gráfico entidad-relación anterior vemos las entidades (tablas de la base de datos) y las relaciones entre las mismas, que se representan mediante líneas y en sus extremos tienen indicado el tipo de relación. Por ejemplo, entre la entidad proyecto y la entidad tarea proyecto tenemos una relación 1 a n porque 1 mismo proyecto puede aparecer en muchas tareas, dicho de otra forma, un proyecto puede tener muchas tareas mientras que una tarea no puede pertenecer a muchos proyectos. También aparecen relaciones 1 a 1 como es el caso de la relación existente entre Usuario y Recurso técnico. Es así porque 1 usuario de CRM no puede ser más de un técnico y de la misma forma un recurso técnico es 1 persona por lo tanto solo puede ser 1 usuario de CRM.

4.6.-Modelo de datos

En Microsoft CRM, todas las tablas tienen un campo con el nombre igual al nombre de la tabla de tipo primarykey, este campo es la clave primaria de la tabla. También podemos relacionar tablas entre si y por cada relación se creará un campo de tipo look up que se encargará de mantener la relación. En definitiva, en las siguientes tablas donde se describe cada uno de los campos, utilizaremos la columna tipo para identificar los campos que son clave primaria y los campos que son claves foráneas. En el caso de las claves primarias sólo tenemos que tener en cuenta que ese campo es clave primaria de la tabla que estamos detallando los campos. En el caso de las claves foráneas hay que saber que la tabla a la que hacen referencia es el nombre que indica la columna nombre y siempre se relaciona por la clave única de esta tabla (que es el id de ese registro).

Tabla proyecto

Nombre	Tipo	Descripción
Nombre	Nvarchar	Nombre del proyecto

Código	Nvarchar	Código del proyecto
Cuenta	Lookup	Empresa (cliente) relacionada con el proyecto
Estado	Picklist	Estado del proyecto
Descripción	Nvarchar	Descripción del proyecto
Provincia	Nvarchar	Provincia de la cuenta asociada al proyecto
Tipo	Picklist	Tipo de proyecto
Fecha firma DRP	Date	Fecha de firma del documento de requerimientos de proyecto
Fecha finalización	Date	Fecha de finalización del proyecto
Fecha inicio	Date	Fecha de inicio del proyecto
Previsión (h)	Float	Estimación (en horas) de la duración del proyecto
Previsión (j)	Float	Estimación (en jornadas) de la duración del proyecto. Calculado a partir de la previsión en horas a razón de 8 horas por jornada.
Cant. presupuestada (€)	Float	Estimación (en €) del presupuesto destinado al proyecto. Calculado a partir de la previsión en horas con un precio hora establecido de 40€.
Consumo (h)	Float	Cantidad (en horas) real consumida en este proyecto. Calculado a partir de los consumos realizados en el diario de proyectos.
Consumo (j)	Float	Cantidad (en jornadas) real consumida en este proyecto. Calculado a partir del consumo en horas.
Cant. Consumida (€)	Float	Cantidad (en €) real consumida en este proyecto. Calculado a partir del consumo en horas.
Responsable	Lookup	Recurso Técnico responsable del proyecto
Fecha creación	Date	Fecha de creación del proyecto
Modificado por	Lookup	Usuario que ha realizado el último cambio sobre

		algún dato del proyecto
Ultima modificación	Date	Fecha de la última modificación realizada sobre el proyecto
Otros costes (€)	Float	Costes (en €) del proyecto no derivados de los técnicos implicados en el proyecto
Total costes (€)	Float	Total de costes (en €) del proyecto. Calculado a partir de la previsión de costes en € más otros costes.
Proyecto	Primarykey	Identificador único de proyectos
Id. De la organización	Lookup	Identificador único de la organización
Estado	State	Estado del proyecto
Razón para el estado	Status	Razón para el estado del proyecto

Tabla Tarea Proyecto

Nombre	Tipo	Descripción
Nombre	Nvarchar	Nombre de la tarea
Descripción	Nvarchar	Descripción de la tarea
Fecha inicio	Date	Fecha de inicio de la tarea
Fecha finalización	Date	Fecha de finalización de la tarea
Previsión (h)	Float	Estimación (en horas) de la duración la tarea
Previsión (j)	Float	Estimación (en jornadas) de la duración la tarea. Calculado a partir de la previsión en horas.

Cant. presupuestada (€)	Float	Estimación (en €) del presupuesto destinado a la tarea. Calculado a partir de la previsión en horas.
Consumo (h)	Float	Cantidad (en horas) real consumida en esta tarea. Calculado a partir de las imputaciones realizadas en el diario de proyectos.
Consumo (j)	Float	Cantidad (en jornadas) real consumida en esta tarea. Calculado a partir del consumo en horas a razón de 8 horas por jornada.
Cant. Consumida (€)	Float	Cantidad (en €) real consumida en este proyecto. Calculado a partir del consumo de horas a un precio de 40€ por hora.
Fecha creación	Date	Fecha de creación de la tarea
Modificado por	Lookup	Usuario que ha realizado el último cambio sobre algún dato del proyecto
Ultima modificación	Date	Fecha de la última modificación realizada sobre el proyecto
Proyecto	Lookup	Proyecto relacionado con la tarea
Tarea proyecto	Primarykey	Identificador único de la tarea
Id. De la organización	Lookup	Identificador único de la organización

Diario proyecto

Nombre	Tipo	Descripción
Descripción	Nvarchar	Concepto de la imputación
Fecha creación	Date	Fecha de registro de la imputación
Cantidad (h)	Date	Cantidad de horas que se imputan
Autor	Lookup	Identificador único del usuario que creó el registro

Modificado por	Lookup	Usuario que ha realizado el último cambio sobre algún dato del proyecto
Ultima modificación	Date	Fecha de la última modificación realizada sobre el proyecto
Unidad de negocio	Lookup	Identificador único de la unidad de negocio
Diarioproyecto	Primarikey	Identificador único de cada registro de diario proyecto
Proyecto		Proyecto donde se imputarán las horas
Tarea proyecto		Tarea donde se imputarán las horas
Estado	State	Estado del diario de proyecto
Razón para el estado	Status	Razón para el estado del diario de proyecto

Recurso técnico

Nombre	Tipo	Descripción
Nombre recurso	Nvarchar	Nombre del recurso
Descripción	Nvarchar	Descripción referente al recurso
Unidad de negocio	Lookup	Identificador único de la unidad de negocio
Tipo	Picklist	Tipo de recurso
Precio venta	Float	Precio de venta (en €) del recurso por hora
Recurso técnico	Primarykey	Identificador único del recurso
Estado	State	Estado del recurso
Razón para el estado	Status	Razón para el estado del recurso

Fecha creación	Date	Fecha de creación del recurso
Modificado por	Lookup	Último usuario que modificó la ficha del recurso
Ultima modificación	Date	Fecha de la última modificación de la ficha del recurso
Creado por	Lookup	Usuario que creó el recurso

4.7.-Lógica de negocio

En Microsoft CRM contamos con 2 potentes herramientas para personalizar y adaptar la lógica de negocio a nuestra empresa. Por un lado contamos con los procesos de venta y por otro lado con los flujos de trabajo que nos permitirán automatizar tareas repetitivas, por ejemplo que al crear una oportunidad se genere una tarea de llamada al cliente. También nos permitirá dirigir usuarios, indicando los pasos a seguir, por ejemplo, al dar de alta un nuevo cliente en CRM asignándole una actividad de "envío de mail de bienvenida" al comercial responsable. Enviar e-mails de forma automática, por ejemplo en el caso de dar de baja un cliente que se envíe un mail al director comercial para que esté al corriente. De esta manera también podemos establecer alertas en función de unos requerimientos específicos. En definitiva, con estas herramientas podemos crear procesos para cualquier tabla de la base de datos de CRM y, concretar en qué eventos activarse, si al crear un registro, al borrarlo, al modificarlo o también tenemos la opción de activarlo manualmente.

Formulario de regla

Seleccionar comportamiento

Utilice los menús de Tareas comunes para definir el comportamiento de la regla, el proceso de ventas o la plantilla.

Nombre de regla: Asignación oportunidad

Descripción de la regla:

Tareas comunes:

- Insertar condición
- Insertar acción
- Editar selección
- Eliminar selección

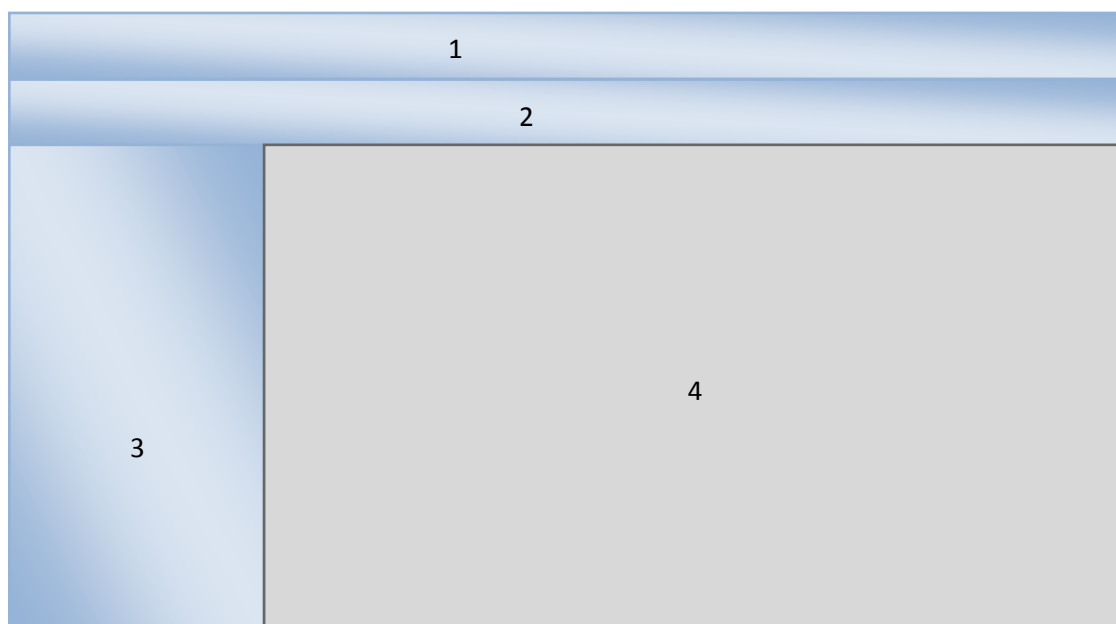
Cuando se cree actividad de la campaña

Ayuda Guardar Cancelar

En Microsoft CRM los informes se pueden crear y se modifican en Visual Studio .NET con proyectos de Business Intelligent o bien utilizando el conjunto de herramientas que nos proporciona Microsoft SQL Server Reporting Services. De forma que en CRM solo tenemos que cargar el informe, configurar permisos de ejecución y la visibilidad para poder empezar a utilizarlo.

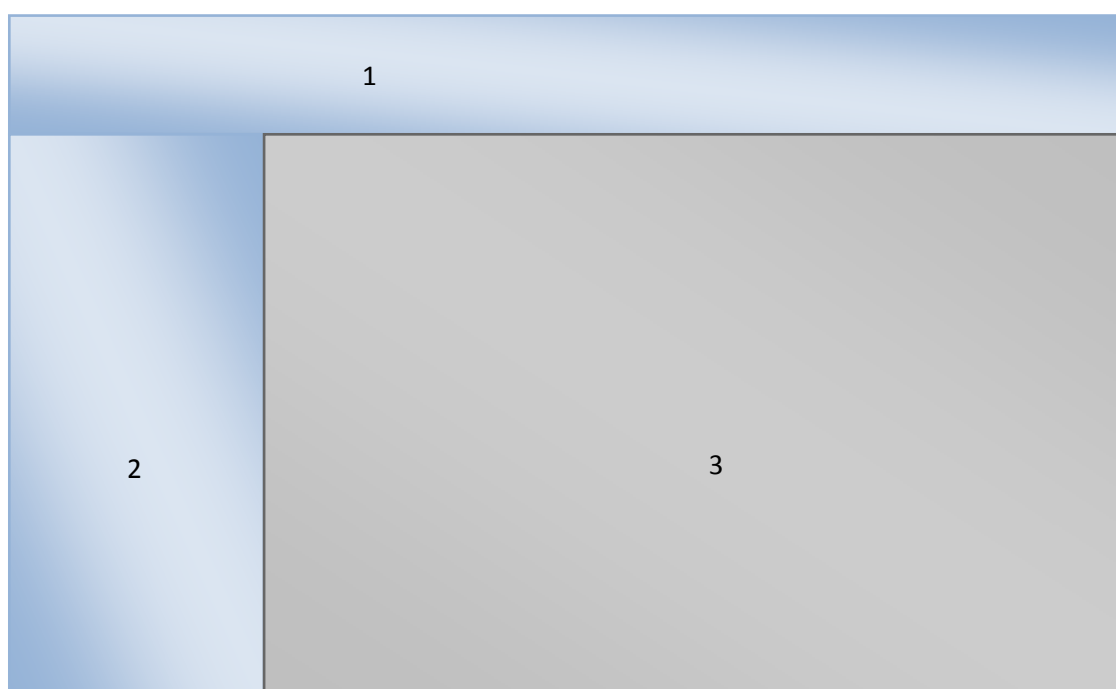
4.8.-Interfaz Gráfica

Formulario general



Este formulario es el principal, nos aparece al iniciar Microsoft CRM y está dividido en grandes partes. En el área 1 y 2 tendremos la zona de menús estándar de Microsoft CRM desde donde podremos crear nuevos registros y acceder a opciones. Además dispone de iconos para acceder rápidamente a distintas partes del CRM. En la zona 3, en la parte inferior tenemos los accesos a cada módulo de CRM mientras que en la parte superior tendremos un detalle del módulo seleccionado. Por último, en el área más grande tenemos el espacio para mostrar el contenido del formulario seleccionado mostrándose según la vista seleccionada en cada ocasión.

Formulario entidad



Este formulario es estándar en CRM y nos lo encontraremos cada vez que se abra el formulario de cualquier registro. Tenemos otra vez todo el espacio dividido en grandes partes y está estructurado de la misma manera. En la parte superior se dedica el espacio a los menús. En la zona 2 encontraremos un acceso rápido a los registros de las entidades relacionadas con el registro abierto. Y, como antes, en la zona más extensa está el espacio reservado a los campos del formulario.

Detalle del formulario de proyecto

General	Duración	Costes	Administració	Notas
Nombre				
Código		Dirección		
Cuenta		Provincia		
Estado		Tipo		
Descripción				

General	Duración	Costes	Administració	Notas
Fecha firma			Fecha inicio	
Fecha final más desfavorable			Fecha fin	
Fecha final más probable			Fecha inicial pactada	

General	Duración	Costes	Administració	Notas
Previsión (h)			Consumo (h)	
Previsión (j)			Consumo (j)	
Cant. Presup.			Cant. Cons.	
Otros costes			Total costes	

General	Duración	Costes	Administración	Notas
Responsable	<input type="text"/>			
Creado por	<input type="text"/>		Fecha creación	<input type="text"/>
Modificado por	<input type="text"/>		Ultima modif.	<input type="text"/>

General	Duración	Costes	Administración	Notas

Detalle del formulario de Tarea proyecto

General	Duración	Costes	Notas
Número			
Descripción			
Proyecto			

General	Duración	Costes	Notas
Fecha inicio		Fecha creación	
Fecha fin		Fecha modific.	

General	Duración	Costes	Notas
Previsión (h)	<input type="text"/>	Consumo (h)	<input type="text"/>
Previsión (j)	<input type="text"/>	Consumo (j)	<input type="text"/>
Cant. Presup.	<input type="text"/>	Cant. Cons.	<input type="text"/>

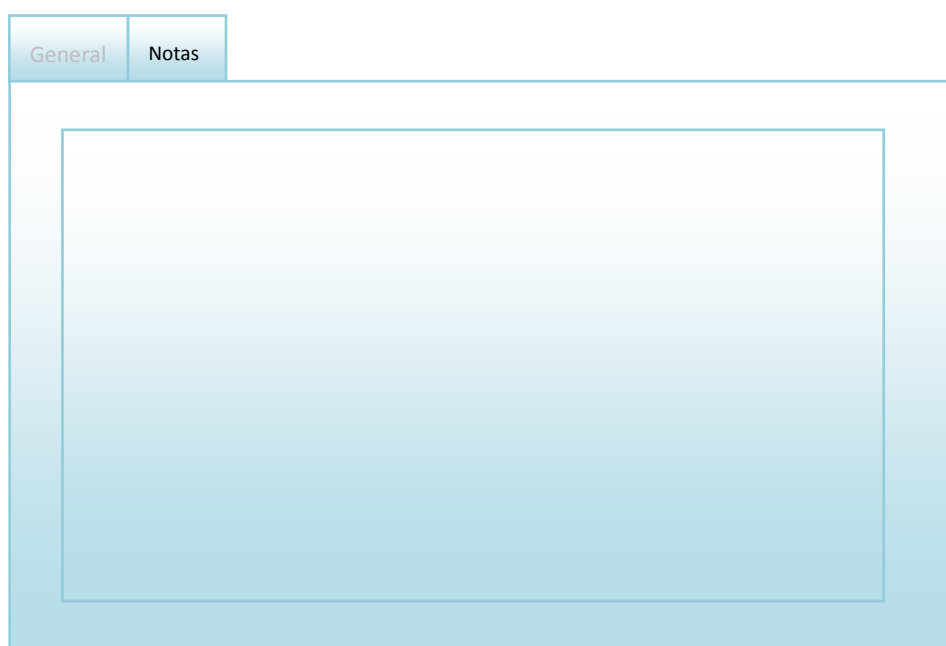
General	Duración	Costes	Notas

Detalle del formulario Diario Proyecto

General			
Proyecto	<input type="text"/>	Tarea Project	<input type="text"/>
Propietario	<input type="text"/>	Cantidad	<input type="text"/>
Descripción	<input type="text"/>		

Detalle del formulario de Recurso técnico

General	Notas
Nombre	<input type="text"/>
Autor	<input type="text"/>
Tipo	<input type="text"/>
Descripción	<input type="text"/>



4.9.-Vistas

Hasta el momento hemos visto de qué manera se introducirá la información en nuestro sistema pero no hemos comentado nada sobre la manera de visualizar esta información. Microsoft CRM utiliza lo que se llaman vistas para mostrarnos la información introducida anteriormente.

Una vista es un conjunto de atributos (de campos a nivel de base de datos) que se mostraran al acceder a una entidad desde el menú de CRM. Cada entidad tiene 1 vista predeterminada y otras vistas alternativas. Se pueden crear vistas desde el área de personalización (entonces la vista seria pública, en el sentido que la verían todos los usuario con permisos de lectura para acceder a esa entidad. También es posible que cada usuario se cree sus propias vistas (estas vistas son propias de cada usuario por lo tanto, solo las ve él) desde búsqueda avanzada, se crean los criterios de filtrado y se pueden guardar.

En las vistas podemos decidir qué campos se mostrarán y qué tamaño ocupara del ancho de la ventana. La vista predeterminada es la que se muestra automáticamente al seleccionar la entidad desde el menú del CRM mientras que las vistas alternativas se pueden elegir mediante un desplegable situado en la parte superior izquierda.

Vista predeterminada de Proyecto

Archivo Nuevo Ir a Herramientas Ayuda Usuario: Administrador CR

Nuevo registro Búsqueda avanzada

Proyectos

- Proyectos
- Diario de proyectos
- Recursos técnicos
- Tareas proyecto

Proyectos

Buscar: Buscar Vista: Proyectos activo

Nuevo Más acciones

Nombre	Consumo(horas)	Previsión(horas)	Fecha inicio	Fecha finalización
▶ Instalación CRM en Obos	0,00	50,00	27/07/2009	10/08/2009
▶ Soporte ERP	0,00	50,00	01/01/2009	31/12/2009
▶ Soporte Generico	4,00	95,00	01/01/2009	31/12/2009

1 de 3 seleccionados. Página 1

Todos # A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Área de trabajo
 Ventas
 Marketing
 Proyectos
 Servicio
 Configuración

Vista predeterminada de Tarea Proyecto

Archivo Nuevo Ir a Herramientas Ayuda Usuario: Administrador CRM

Nuevo registro Búsqueda avanzada

Proyectos

- Proyectos
- Diario de proyectos
- Recursos técnicos
- Tareas proyecto**

Tareas proyecto

Buscar: Buscar Vista: Tareas proyecto activo

Nuevo Más acciones

<input type="checkbox"/>	Nombre	Descripción	Consumo (horas)	Previsión (horas)	Fecha creación	
<input type="checkbox"/>	1000	Soporte crm y erp para SoftPel	4,00	15,00	24/07/2009	
<input type="checkbox"/>	1001	Soporte telefonico cliente Tlis	0,00	20,00	24/07/2009	
<input type="checkbox"/>	1002	Soporte telefonico cliente Claus	0,00	20,00	24/07/2009	
<input type="checkbox"/>	1003	Soporte de crm y erp para Ise-Lang	0,00	20,00	24/07/2009	

1 de 4 seleccionados. Página 1

Todos # A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Área de trabajo

- Ventas
- Marketing
- Proyectos**
- Servicio
- Configuración

Vista predeterminada de Diario de Proyecto

Archivo Nuevo Ir a Herramientas Ayuda Usuario: Administrador CRM

Nuevo registro Búsqueda avanzada

Proyectos

- Proyectos
- Diario de proyectos**
- Recursos técnicos
- Tareas proyecto

Diario de proyectos

Buscar: Buscar Vista: Diario de proyectos activo

Nuevo Más acciones

<input checked="" type="checkbox"/>	Fecha creación	Descripción	Cantidad(horas)	Propietario	Proyecto	Tarea proyecto	
<input checked="" type="checkbox"/>	24/07/2009	error al instalar sp1	4,0	Administrador CRM	Soporte Generico	1000	

1 de 1 seleccionados. Página 1

Todos # A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Área de trabajo

- Ventas
- Marketing
- Proyectos**
- Servicio
- Configuración

Vista predeterminada de Recurso

Usuario: Administrador CR

Nuevo registro | Búsqueda avanzada

Proyectos

- Proyectos
- Diario de proyectos
- Recursos técnicos**
- Tareas proyecto

Recursos técnicos

Buscar: Buscar Vista: Recursos técnicos activo

Nuevo | | | Más acciones

Nombre	Fecha creación	Tipo
Francisco Jose Beltran	17/07/2009	CONS3 (>1 año experiencia)
Luis Lopez	24/07/2009	CONS1 (<1 año experiencia)
Toni Galón	24/07/2009	CONS2 (1 año experiencia)

1 de 3 seleccionados. |< <=> Página 1 >=> >

Todos # A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Vistas alternativas

Desde el desplegable que encontramos en la parte superior derecha podemos seleccionar la vista que se necesitemos (a elegir entre las del sistema y las creadas por el propio usuario) en cada momento para que nos presente unos determinados campos u otros en función de las necesidades.

Vista: Proyectos activo

Vistas del sistema

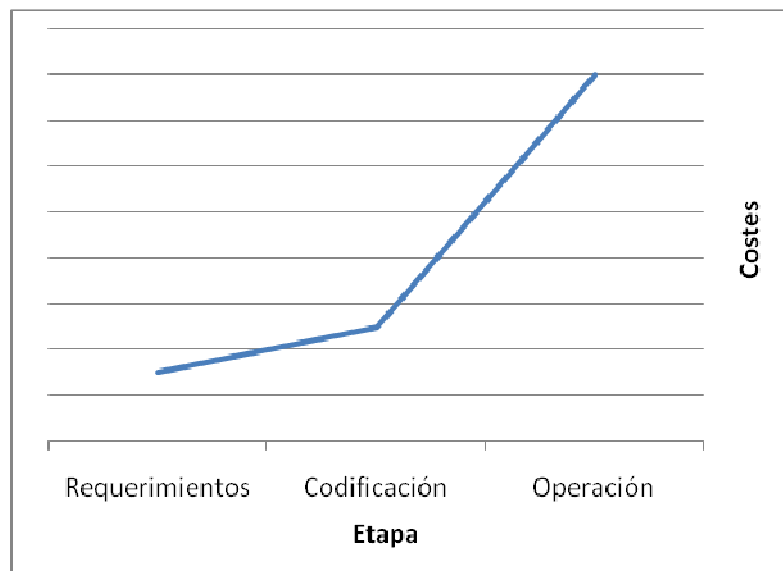
- Proyectos activo
- Proyectos inactivo
- Proyectos que terminan este mes

5.-Plan de pruebas

5.1.-En qué consiste

La prueba es un proceso de ejecución de un programa con la intención de descubrir un error. Una buena prueba es aquella que tiene alta probabilidad de encontrar un error no descubierto hasta entonces.

Al realizar un plan de pruebas el objetivo que se pretende alcanzar es descubrir en el producto las desviaciones de las especificaciones, es decir, si el sistema final cumple con los requisitos para lo que se construyó, permite responder a la pregunta: ¿Es correcto el producto?. Como podemos ver en el gráfico de más abajo, los costes de encontrar errores son muy diferentes en función de la etapa en la que se encuentre nuestro proyecto por lo que es importante detectarlos justo antes de dar por acabada la fase de codificación.



Las pruebas no pueden asegurar la ausencia de defectos pero sí puede demostrar que existen errores en el software.

Una manera muy extendida de llevar a cabo las pruebas funcionales es mediante casos de uso. La principal ventaja de utilizar casos de uso es que ocultan los detalles internos, son fáciles de entender por los clientes, de construir y de modificar. Además pueden utilizarse en otros sistemas o proyectos.

5.2.-Tipos de pruebas

Pruebas de caja blanca

Las pruebas de caja blanca, también llamadas pruebas de caja transparente o pruebas estructurales, se centran en el código, intentan comprobar el código y aproximarse al máximo, al 100%, de código testado aunque en desarrollos complejos pueda resultar inviable ya que el número de pruebas sería, en muchos casos, infinito. Estas pruebas se utilizan para comprobar que el código hace correctamente lo que el diseño de bajo nivel indica. Se realizan pruebas de segmentos para comprobar bloques de código. Para segmentos sin puntos de decisión es suficiente ejecutar 1 vez para comprobar el 100% del código, como esto es poco habitual (nos encontraremos con puntos de decisión) hay otro tipo de pruebas llamadas pruebas de ramas que consiste en recorrer todas las posibles salidas de los puntos de decisión.

En resumen, las pruebas de caja blanca nos convencen que hace bien lo que hace pero no comprueba que hace lo que necesitamos.

Por ejemplo, un listado de clientes muestra los clientes (lo que hace lo hace bien) pero en lugar de hacerlo en orden ascendente por código de cliente, que lo haga orden descendente (no hace lo que necesitamos). Por lo tanto una prueba de caja blanca lo detectaría.

Pruebas de caja negra

Las pruebas de caja negra, también llamadas pruebas funcionales, intentan encontrar los casos en que un módulo no atiende a su especificación y tiene que ser, por tanto, reportado y corregido. El probador se limita a suministrar datos de entrada, ejecutar la prueba comprobar el resultado sin preocuparse por el funcionamiento interno. En este tipo de pruebas la complejidad no radica en el número de funciones sino en los datos que se les pasa a estas funciones. Por ejemplo si tenemos valores comprendidos en un rango no podemos probar todas las combinaciones, así que definimos 3 casos, si el valor es inferior al rango, si se encuentra dentro del rango o si es superior al rango. Otro ejemplo sería si una entrada espera un valor dentro de un conjunto, no probaríamos todos los valores sino que diferenciamos 2 grupos, si el valor se encuentra dentro del conjunto o bien se encuentra fuera.

Pruebas de sistema

Las pruebas de sistema se encargan de validar que se alcanza la funcionalidad y el rendimiento global del sistema por lo tanto se verifica tanto el hardware como el software. Algunas de estas pruebas son:

- Pruebas de validación: son básicamente pruebas de caja negra. En este punto comprobamos que se cumplen los requerimientos funcionales y de rendimiento.
- Pruebas de recuperación: estas pruebas provocan fallos de software para comprobar si la recuperación es correcta.
- Pruebas de seguridad: sirven para verificar que se cumple con las especificaciones sobre protección de datos.

-Pruebas de rendimiento: se encargan de comprobar, en tiempo de ejecución, el rendimiento del software.

-Pruebas de instalación: se trata de probar el software en distintas instalaciones software y hardware y bajo condiciones especiales como sería el caso de espacio de disco disponible insuficiente.

Pruebas de regresión

Con este tipo de pruebas, lo que se pretende es descubrir nuevos errores generados a partir de las modificaciones realizadas para solucionar otros errores detectados en pruebas anteriores para reportarlos al responsable correspondiente y que de esta manera pueda ser corregido. Estos errores se producen, por lo tanto, por consecuencias inesperadas de las últimas modificaciones.

Es una estrategia que consiste en ejecutar de nuevo las pruebas realizadas anteriormente sobre el sistema modificado, después de corregir los errores detectados en la fase anterior.

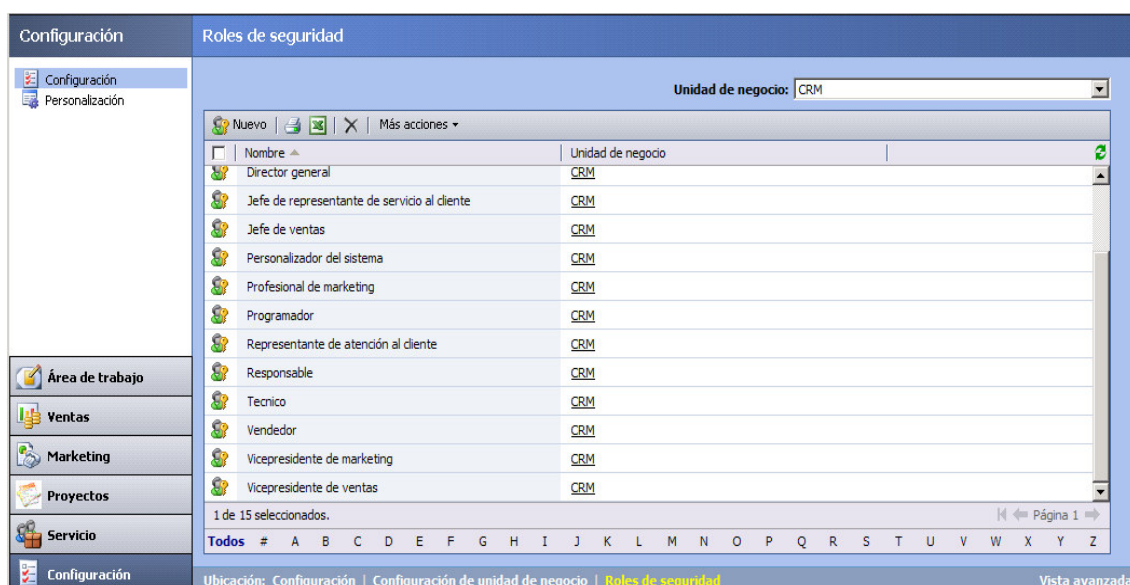
Una buena práctica sería que cuando se detecta y corrige un error, éste sea anotado y que posteriormente se ejecuten de nuevo las pruebas para comprobar que ha sido corregido y que no ha generado nuevos.

Pruebas de aceptación

Este tipo de pruebas las realiza el cliente, son básicamente pruebas funcionales y sirven para validar que se cumple la especificación de requisitos y el manual de usuario. Las podríamos clasificar en dos grupos, las que se realizan en el entorno de desarrollo, que las realiza el cliente antes de que se terminen los desarrollos y cuenta con el apoyo de un técnico para ayudarlo. El segundo grupo son las pruebas que realiza el cliente una vez terminado el producto, en su casa y sin contar con un técnico al lado.

5.3.-Test 1

Para la realización de las pruebas se han creado dos nuevos roles que agrupan una serie de permisos. Por un lado tenemos el rol llamado *Técnico*, con este rol asignado, un usuario dispondrá de permisos para hacer y deshacer únicamente con los registros propios. Por otro lado hemos creado el rol de *Responsable* que tiene permisos para administrar sus registros y los de su unidad organizativa (elemento parametrizable mediante el cual establecemos jerarquías) para así poder supervisar y corregir si es necesario, además de poder asignar registros a otros usuarios.



Todas las pruebas siguientes se realizan con un usuario con permisos suficientes para permitirnos realizarlas.

Nº: CRM CU001

Escenario: Un usuario con permisos debe poder crear un proyecto.

Resultado esperado: Después de rellenar el formulario y guardar, el proceso debe realizar sin problemas.

Resultado: El usuario con permisos ha creado un proyecto exitosamente.

Observaciones:

Nº: CRM CU002

Escenario: Un usuario con permisos debe poder modificar datos de los proyectos creados.

Resultado esperado: Tiene que poder modificar datos de un proyecto y guardar los cambios.

Resultado: Se ha podido modificar el proyecto sin mensajes de error.

Observaciones:

Nº: CRM CU003

Escenario: El usuario con permisos puede borrar un proyecto.

Resultado esperado: El proyecto se debe poder borrar.

Resultado: El usuario con los correspondientes permisos puede borrar el proyecto.

Observaciones:

<p>Nº: CRM CU004</p> <p>Escenario: El usuario puede abrir un proyecto creado para consultar información sobre el mismo.</p> <p>Resultado esperado: Se deben mostrar los registros y se deben poder abrir los registros de proyectos a un usuario con permisos de lectura.</p> <p>Resultado: El usuario con permisos puede acceder al formulario de proyecto.</p> <p>Observaciones:</p>
<p>Nº: CRM CU005</p> <p>Escenario: Se debe poder realizar búsquedas sobre determinados campos de los que se muestran en el formulario de proyecto.</p> <p>Resultado esperado: Búsqueda sobre:</p> <p>Resultado:</p> <p>Observaciones:</p>
<p>Nº: CRM CU006</p> <p>Escenario: El usuario tiene que poder crear actividades relacionadas con los proyectos.</p> <p>Resultado esperado: Después de rellenar correctamente el formulario de proyectos y hacer clic sobre el botón guardar, éste debe crearse.</p> <p>Resultado: Se puede crear el proyecto sin errores.</p> <p>Observaciones:</p>
<p>Nº: CRM CU007</p> <p>Escenario: Registrar horas en el diario de proyectos.</p> <p>Resultado esperado: El usuario debe poder guardar en el diario de proyectos líneas de imputación de horas en tareas y proyectos.</p> <p>Resultado:</p> <p>Observaciones:</p>
<p>Nº: CRM CU008</p> <p>Escenario: Un proyecto puede grabarse como completado.</p> <p>Resultado esperado: Al dar por finalizado un proyecto, el formulario se muestra de solo lectura.</p>

<p>Resultado: Al guardarlo como completado el proyecto se puede modificar.</p> <p>Observaciones: Se modifica el código JavaScript y se repite la prueba obteniendo el resultado esperado.</p>
<p>Nº: CRM CU009</p> <p>Escenario: Se debe poder modificar las tareas ya creadas.</p> <p>Resultado esperado: El usuario, en función de los permisos que tenga, podrá modificar tareas, las suyas o cualquiera de las creadas.</p> <p>Resultado: Con el rol de técnico se pueden modificar las tareas propias, con el responsable se pueden modificar las tareas propias y las creadas por otros usuarios.</p> <p>Observaciones:</p>
<p>Nº: CRM CU010</p> <p>Escenario: El usuario tiene que poder borrar una tarea sin completar.</p> <p>Resultado esperado: Dependiendo de los roles que tenga el asignados el usuario debe poder eliminar una o varias tareas.</p> <p>Resultado: Si el usuario tiene asignado el rol de técnico solo se borran sus propias tareas. Si tiene el rol de responsable no puede borrar también las tareas que sean de otros usuarios aunque sí las propias.</p> <p>Observaciones: Con el rol de responsable tiene que poder eliminar también tareas de otros usuarios, se revisan los permisos y se corrige el problema. Se repite la prueba y se completa con éxito.</p>
<p>Nº: CRM CU011</p> <p>Escenario: Un usuario debe poder ejecutar informes estándares de CRM (y personalizados si los hubiera).</p> <p>Resultado esperado: El usuario debe poder ejecutar los informes disponibles en función de la entidad a la que pertenezcan los registros seleccionados en cada ocasión.</p> <p>Resultado: Se ejecutan los informes correctamente.</p> <p>Observaciones:</p>
<p>Nº: CRM CU012</p> <p>Escenario: El usuario debe poder asignar tareas a otros usuarios.</p> <p>Resultado esperado: Después de crear una tarea o editando una tarea ya creada el usuario tiene que poder asignar tareas a otros usuarios (también se puede parametrizar, mediante la unidad organizativa, los usuarios a los que tiene permiso para asignar registros).</p>

<p>Resultado: El usuario puede completar el proceso de asignación de tareas correctamente.</p> <p>Observaciones:</p>
<p>Nº: CRM CU013</p> <p>Escenario: Exportar la información que muestran las vistas a Microsoft Excel.</p> <p>Resultado esperado: El usuario selecciona una vista y después de clicar y seleccionar si quiere exportar los registros seleccionados o todos los registros que muestra la vista, se debe abrir Microsoft Excel mostrando esta información.</p> <p>Resultado: Se puede exportar a Microsoft Excel y después trabajar esa información utilizando toda la potencia que nos proporciona Excel.</p> <p>Observaciones:</p>
<p>Nº: CRM CU14</p> <p>Escenario: Flujos de trabajo, ponerlos en análisis de funcionalidad</p> <p>Resultado esperado: El usuario tiene que poder ejecutar los flujos asociados a cada entidad.</p> <p>Resultado: Se pueden ejecutar los flujos correctamente.</p> <p>Observaciones:</p>
<p>Nº: CRM1</p> <p>Escenario: Comprobar permisos del rol <i>técnico</i>.</p> <p>Resultado esperado: Un usuario con el rol de técnico no podrá administrar registros de otros usuarios (como administrar entendemos: crear en nombre de, modificar, eliminar y asignar).</p> <p>Resultado: El rol de técnico asignado no podemos administrar registros de otros usuarios.</p> <p>Observaciones:</p>
<p>Nº: CRM2</p> <p>Escenario: Comprobar permisos del rol <i>responsable</i></p> <p>Resultado esperado: Un usuario que tenga asignado el rol de responsable podrá administrar registros suyos y de otros usuarios.</p> <p>Resultado: Puede administrar los registros de otros usuarios pero al eliminar da error de permisos.</p> <p>Observaciones: Se revisa los permisos y se modifican para obtener el resultado esperado. Se repite la prueba y termina con éxito.</p>

Nº: CRM3

Escenario: En el formulario de proyectos aparecen una serie de campos que son de cálculo automático, se identifican porque aparecen deshabilitados para edición manual.

Resultado esperado: Al modificar costes, previsiones y/o consumos reales, se tienen que actualizar los campos correspondientes.

Resultado: Se modifican correctamente los campos que tocan en función del campo modificado.

Observaciones:

Nº: CRM4

Escenario: En el formulario de tarea de proyectos aparecen una serie de campos que son de cálculo automático, se identifican porque aparecen deshabilitados para edición manual.

Resultado esperado: Al modificar costes, previsiones y/o consumos reales, se tienen que actualizar los campos correspondientes.

Resultado: Se modifican correctamente los campos que tocan en función del campo modificado.

Observaciones:

6.-Conclusiones

6.1.-Cumplimiento de objetivos

Los objetivos iniciales han sido cubiertos por completo todo y que ha sido requerido un gran esfuerzo para poder cumplirlos y dar por completado el proyecto. Si los agrupamos en grandes grupos, tenemos por un lado la comprensión en otros sistemas de cómo funciona y cómo se gestionan los proyectos en otros sistemas para después poder aplicar estos conocimientos adquiridos al CRM y por otro lado comprender el funcionamiento del CRM de Microsoft y aprender a configurar y desarrollar el CRM para realizar los cambios necesarios.

Ahora voy a comentar los objetivos uno a uno, son los objetivos especificados en el estudio de viabilidad.

El primer objetivo era tener la información centralizada y se ha conseguido ya que solo hay una base de datos.

El segundo objetivo (contener información sobre proyectos) también ha sido completado al 100% aunque este es el objetivo que más me ha costado cumplir.

El tercer objetivo (interfaz sencilla) también se ha cubierto, ya que la interfaz es simple y todavía mucho más sencilla para quién haya trabajado con Outlook.

Realizar los desarrollos manteniendo la usabilidad y la sencillez de Microsoft CRM ha supuesto realizar los desarrollos siguiendo el estilo que proporciona este software.

El quinto objetivo es mantener el histórico de proyectos, esto es posible y se consigue cambiando el estado del proyecto (estado finalizado) para que pase a formar parte del histórico.

Los siguientes objetivos tienen carácter personal pero no por ello son menos importantes:

El primero de ellos es ampliar los conocimientos sobre Microsoft CRM. Al empezar este proyecto sí que conocía algo de esta herramienta pero muy por encima, de haberla visto en alguna ocasión. Después de estar las horas que le he dedicado al proyecto, delante del CRM, ahora puedo manejarlo con mucha más fluidez, conozco las posibilidades de esta solución y sé donde hay que ir a buscar cada cosa. Por lo tanto este objetivo lo considero cubierto también.

El segundo objetivo (el 7º en la lista) también ha sido alcanzado ya que he investigado sobre los pros y contras de un sistema CRM, sus puntos fuertes, cómo utilizarlos para extraer todo el potencial...

De la misma manera, para desarrollar un módulo de proyectos primero tenía que investigar y formarme sobre esta área para poder documentar y ponerme manos a la obra. Creo que si este objetivo no se hubiera trabajado, no habría sido posible realizar los desarrollos correctamente (no técnicamente sino con la lógica de negocio).

6.2.-Retrasos en la planificación

La realización de este proyecto final de carrera ha supuesto un gran esfuerzo siendo el factor que más ha hecho aumentar, de manera drástica, la duración, el gran desconocimiento sobre el área de desarrollo en este tipo de programación de servicios web y que por lo tanto ha originado un retraso sobre la planificación inicial ya que la tarea de desarrollo (propriadamente la programación en .NET para generar el fichero dll, que hemos comentado en otros apartados como callouts) se ha prolongado muy por encima de lo previsto inicialmente pero después de todo se ha completado de forma exitosa.

Ha habido otro retraso producido por un riesgo de seguridad no contemplado, las copias de seguridad y por ello ha habido pérdida de información y el consiguiente trabajo perdido.

6.3.-Líneas de ampliación

- Crear más campos en las tablas de proyecto, tarea de proyecto y diario de proyecto que permitan mejorar el análisis y el estado de los proyectos y sus tareas.
- Además de poder ver los datos como números, en listados o en los propios formularios, podemos verlos con gráficos, desarrollándolos en Microsoft SQL Server Reporting Services. Así podremos tener una visión distinta del estado de los proyectos que resultará muy útil para dirección o responsables de proyectos o de área.
- Predefinir flujos de trabajo "estándares" para tareas periódicas y así definir procesos de negocio estandarizados que puedan resultar útiles para los clientes y que ellos mismos decidan cuáles utilizar o modificar.
- La utilización del CRM en una pda con Windows Mobile permitiría a los usuarios poder llevarse parte de la información contenida en el CRM para así disponer de esta información siempre que se encuentre fuera de la oficina y que, al volver, se actualice con el servidor para estar siempre actualizado.

6.4.-Valoración personal

Llegados este punto, con el proyecto finalizado y después de todo el tiempo dedicado, han quedado cosas que se podrían ampliar y otras que se me han ocurrido durante la realización del mismo y no todas han sido aplicadas. Así pues hay una serie de mejoras que se podrían incluir en futuras versiones o posteriores proyectos de ampliación de éste. Por un lado, el apartado de informes. En este proyecto están instalados y funcionando los informes estándares de CRM pero sería útil disponer de informes específicos desarrollados a medida para estos nuevos desarrollos que nos permitirían disponer de la información extrapolada (o la que necesitemos) de forma más rápida y sencilla ya que no habría que exportar la información (todo y que el mismo CRM se encarga de hacer la exportación, el usuario sólo le tiene que indicar qué registros quiere llevar a Microsoft Excel). Por otro lado tenemos los formularios específicos que podrían incluir más campos (ya sean de información introducida manualmente o de cálculo automático) para disponer de mayor información y control sobre los proyectos.

Mis sensaciones sobre este proyecto han sido muy buenas aunque durante la fase del desarrollo del fichero dll, en ocasiones estuve desanimado ya que no conseguía avanzar

en la programación pero después de muchas horas y esfuerzo he conseguido finalizar los desarrollos. El mayor problema que me he encontrado ha sido la inexistencia de una buena base en programación .NET ya que información por internet hay, y mucha, pero en muchas ocasiones no comprendía el porqué o el cómo y a base de muchas pruebas y ejemplos he ido mejorando dicha base. Con estos desarrollos he intentado realizar unas funciones "modelo" para que en futuras mejoras sirvan como base y con pequeñas modificaciones podamos aprovecharlas. Dejando de lado los desarrollos, con este proyecto he ampliado mis conocimientos sobre cómo una aplicación empresarial (y en el fondo, una empresa) gestiona los proyectos para ver su rentabilidad, entre otros factores. Por último también me ha permitido ampliar mis conocimientos sobre Microsoft CRM ya que, si bien la conocía, no controlaba la parametrización ni la configuración y esto ahora no es así. Por lo tanto creo que con este proyecto he aprendido en distintos caminos y como dije desde el primer día que defendí este proyecto es un área donde me gustaría mejorar e ir ampliando mis conocimientos y con este proyecto he conseguido darle un buen empujón.

7.-Bibliografía

7.1.-Bibliografia

Los libros consultados como soporte para hacer el proyecto han sido:

- Mike Snyder y Jim Steger, "Working with Microsoft Dynamics CRM 3.0".
- Mike Snyder, Jim Steger, Kara O'Brien y Brendan Landert, "Microsoft Dynamics CRM 3.0 Step by step".

Estos dos libros han sido de gran ayuda, el primero para conocer mejor la funcionalidad del CRM y el segundo me ha servido como soporte en la iniciación de los desarrollos realizados.

7.2.-Fuentes electrónicas:

- Microsoft Dynamics CRM Developer Centre.
<http://msdn.microsoft.com/en-us/dynamics/CRM/default.aspx>
- El del CRM
<http://geeks.ms/blogs/marco/>
- Buscador
<http://www.google.es>

De forma generalizada también he utilizado Google para la búsqueda de información necesaria para el desarrollo del proyecto.

Fecha y firma: