

UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMÁTICA

SISTEMAS INFORMÁTICOS

CURSO 2007 - 2008

---



CRM INTELIGENTE

Director: Miguel A. Blanco Rodríguez

Javier Cuadra Aranzubía | Iván Romero Sobrino | Carlos Terrado Bruna

Los abajo firmantes autorizan a la Universidad Complutense de Madrid a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a sus autores, tanto la propia memoria, como el código, la documentación y/o el prototipo desarrollado.

Javier Cuadra Aranzubía

Iván Romero Sobrino

Carlos Terrado Bruna

## Prefacio

La documentación que figura en las siguientes páginas corresponde a la memoria del proyecto de fin de carrera, consistente en el diseño y especificación de un CRM Inteligente o un Gestor de Clientes Inteligente.

El proyecto ha sido desarrollado por Javier Cuadra Aranzubía, Iván Romero Sobrino y Carlos Terrado Bruna, y dirigido y tutorizado por D. Miguel Ángel Blanco Rodríguez.

Como se explica en las siguientes páginas, un CRM es una utilidad software que resulta de una gran utilidad en un entorno empresarial, en cuanto que forma parte de una estrategia de negocio, utilizando las tecnologías para el tratamiento y el proceso automático de datos, con el objetivo de obtener mejores clientes, más rentables, más fieles y más satisfechos.

Nuestro CRM guarda en la base de datos información de clientes y productos, relacionando estos conceptos. Además, también se guardan las incidencias que se generan en el servicio de atención al cliente, que pueden ser de tres tipos: Averías, Reclamaciones y Consultas.

El sistema consta de inteligencia, que decide qué acciones aplicables son favorables o beneficiosas al aplicarlas a un cliente, según sean sus características y/o atributos. Mediante un módulo estadístico y con la información disponible (reacciones de clientes similares) se obtienen los atributos discriminatorios para la elección de la acción a aplicar.

En cuanto al desarrollo del proyecto, hemos estado reuniéndonos periódicamente (cada semana) todos los miembros, con el tutor, para que nos corrigiera las incorrecciones a la hora de especificar, implementar, etc. Las conclusiones de las reuniones las transcribíamos y las guardábamos en un repositorio wiki.

En cuanto al resto de tecnologías empleadas para la implementación del prototipo, JSP y Servlets (desarrollando desde el Entorno de Desarrollo eclipse), JDBC, XHTML, CSS y CVS. Como servidores hemos usado el Apache-Tomcat. La base de datos que hemos usado es MySQL 5 que utiliza el motor MyISAM.

## Summary

The documentation featured in the following pages corresponds to the memory of the end of career project which consists of the design and specification of an Intelligent CRM or an Intelligent agent of customers.

The project has been developed by Javier Cuadra Aranzubía, Iván Romero Sobrino and Carlos Terrado Bruna, and managed and supervised by D. Ángel Blanco Rodríguez.

As explained in the following pages, a CRM is a utility software that results from a big utility in a business environment, as soon as that is part of a business strategy, using the technologies for the treatment and the automatic process of information, with the objective to obtain better clients, more profitable ones, more faithful and more satisfied.

Relating these concepts, our CRM saves the customers' and products' information in the database. Moreover, the incidents generated in customer service are also saved. There can be three types of incidents: damages, claims, and consultations.

The system consists of intelligence that decides which applicable actions are favorable and beneficial upon applying them to a customer, according to whether they are their characteristics and/or attributes. By means of a statistical module and the available information (simulated customer reactions), the discriminated attributes are obtained for the selection of the action to apply.

Regarding the project's development, we have gathered periodically (every week) with the supervisor so that he could correct the incorrectness for us at the time of specifying, helping, etc. We transcribed and saved the conclusions from the meetings in a Wiki repository.

Regarding the rest of the employed technologies for the implementation of the prototype, JSP and Servlets (developed using the integrated development environment Eclipse), JDBC, XHTML, CSS, CVS, we have used the Apache-Tomcat as a server. The database we used is the MySQL 5 with the engine MyISAM.

# Índice general

<b>1. Antecedentes</b>	<b>1</b>
1.1. Objetivos de un CRM . . . . .	1
1.2. Problemas de un CRM . . . . .	3
<b>2. Objetivo del proyecto</b>	<b>5</b>
2.1. Nuestro proyecto . . . . .	8
<b>3. Metodología de desarrollo</b>	<b>11</b>
3.1. Ciclo de vida del sistema . . . . .	11
3.2. Cronograma del proyecto . . . . .	13
3.3. Tecnologías aplicadas en el desarrollo . . . . .	15
3.3.1. Wiki . . . . .	15
3.3.2. JSP y Servlets . . . . .	15
3.3.3. MYSQL . . . . .	16
3.3.4. JDBC . . . . .	17
3.3.5. XHTML . . . . .	18
3.3.6. CSS . . . . .	19
3.3.7. CVS . . . . .	21
<b>4. Análisis</b>	<b>22</b>
4.1. Toma de requerimientos . . . . .	22
4.1.1. Requisitos funcionales . . . . .	23
4.1.2. Requisitos no funcionales . . . . .	24

---

4.2. Análisis predictivo . . . . .	24
4.2.1. Tipos de análisis predictivos . . . . .	25
4.3. Información de los clientes . . . . .	26
4.3.1. Datos descriptivos . . . . .	28
4.3.2. Datos de promociones . . . . .	28
4.3.3. 1.3. Datos Transaccionales . . . . .	29
4.4. Orígenes de los Datos de Cliente . . . . .	30
4.4.1. Internas a la empresa . . . . .	30
4.4.2. Internet . . . . .	30
4.5. Normalización de información de clientes de diferentes sistemas	31
4.5.1. Almacenes de Datos . . . . .	31
4.5.2. Conectores de Datos . . . . .	32
4.6. Árboles de decisión . . . . .	32
4.7. Árboles de decisión alternativos . . . . .	33
<b>5. Diseño</b>	<b>37</b>
5.1. Arquitectura . . . . .	37
5.2. Casos de uso . . . . .	37
5.2.1. UC - 1. Iniciar sesión . . . . .	39
5.2.2. UC - 2. Registrar un nuevo cliente . . . . .	40
5.2.3. UC - 3. Crear una nueva incidencia . . . . .	41
5.2.4. UC - 4. Consultar una incidencia . . . . .	44
5.2.5. UC - 5 Proponer acciones . . . . .	47
5.2.6. UC - 6 Cerrar sesión . . . . .	48
5.3. Diseño de la base de datos . . . . .	49
5.4. Diseño conjunto del sistema . . . . .	52
<b>6. Implementación</b>	<b>54</b>
6.1. Decisiones sobre el diseño de la Base de Datos . . . . .	54
6.2. Capturas de pantalla del prototipo . . . . .	56

<b>7. Ejemplos de funcionamiento</b>	<b>62</b>
7.1. Ejemplo 1 (Averías) . . . . .	62
7.2. Ejemplo 2 (Reclamaciones) . . . . .	65
7.3. Ejemplo 3 (Consultas) . . . . .	67
<b>8. Conclusiones</b>	<b>69</b>
<b>9. Extensiones</b>	<b>71</b>
<b>A. Glosario</b>	<b>73</b>
<b>B. Listado de las actas de las reuniones</b>	<b>75</b>
B.1. Acta 1 (24/10/2007) . . . . .	75
B.2. Acta 2 (31/10/2007) . . . . .	77
B.3. Acta 3 (07/11/2007) . . . . .	80
B.4. Acta 4 (14/11/2007) . . . . .	82
B.5. Acta 5 (21/11/2007) . . . . .	83
B.6. Acta 6 (28/11/2007) . . . . .	84
B.7. Acta 7 (05/12/2007) . . . . .	85
B.8. Acta 8 (12/12/2007) . . . . .	86
B.9. Acta 9 (19/12/2007) . . . . .	87
B.10. Acta 10 (09/01/2008) . . . . .	88
B.11. Acta 11 (16/01/2008) . . . . .	90
B.12. Acta 12 (23/01/2008) . . . . .	91
B.13. Acta 13 (04/03/2008) . . . . .	92
B.14. Acta 14 (11/03/2008) . . . . .	93
B.15. Acta 15 (2/04/2008) . . . . .	94
B.16. Acta 16 (16/04/2008) . . . . .	96
B.17. Acta 17 (29/04/2008) . . . . .	98
B.18. Acta 18 (21/05/2008) . . . . .	99
B.19. Acta 19 (04/06/2008) . . . . .	100
B.20. Acta 20 (10/06/2008) . . . . .	101

# Índice de figuras

1.1. Ciclo de explotación de un CRM . . . . .	4
2.1. Flujo del proceso <i>Acción - Reacción</i> . . . . .	9
3.1. Modelo en cascada . . . . .	12
3.2. Ciclo de funcionamiento de las reuniones . . . . .	14
4.1. Tres tipos de datos sobre clientes . . . . .	27
4.2. Información suficiente para realizar Data Mining . . . . .	28
4.3. Ejemplo de <i>árbol de decisión</i> que permite decidir si se oferta un determinado producto a un determinado cliente. . . . .	33
4.4. Ejemplo de un <i>árbol de decisión alternativo</i> que permite obtener una puntuación, evaluando así cómo de apto es el cliente para una determinada acción. . . . .	35
4.5. Explotación CRMI . . . . .	36
5.1. Diagrama entidad - relación . . . . .	50
5.2. Interrelación de los módulos del sistema . . . . .	53
6.1. Inicio de sesión en el sistema CRMI . . . . .	57
6.2. Formulario de alta de un nuevo cliente . . . . .	57
6.3. Modo de recuperación de la contraseña de usuario olvidada . . . . .	58
6.4. Página inicial al entrar al sistema . . . . .	59
6.5. Averías activas . . . . .	60
6.6. Reclamaciones activas . . . . .	60

---

6.7. Consultas activas . . . . .	61
6.8. Formulario de edición de una incidencia . . . . .	61
7.1. Ejemplo de un <i>árbol de decisión alternativo</i> para un hipotético caso de averías. . . . .	63
7.2. Ejemplo de un <i>árbol de decisión alternativo</i> para un hipotético caso de reclamaciones. . . . .	66
7.3. Ejemplo de un <i>árbol de decisión alternativo</i> para un hipotético caso de consultas. . . . .	68
B.1. Diagrama de flujo . . . . .	95
B.2. Diagrama de flujo - Versión 2 . . . . .	97

# Capítulo 1

## Antecedentes

Un Customer Relationship Management, CRM, es parte de una estrategia de negocio que utiliza tecnologías de la información centrándose en crear relaciones con clientes, de tal forma, que se consiga un conocimiento preciso de sus necesidades, intereses y patrones de compra. Todo esto es posible gracias a sistemas software que permitan gestionar la información de los clientes y las operaciones comerciales relacionadas con ellos. Es por este motivo, que un CRM no es sólo una aplicación informática sino que va más allá y supone idear una estrategia de negocio al cliente. Este negocio depende de la capacidad de renovación de la propia empresa.

No sería posible pensar en un CRM sin un sistema capaz de trabajar con grandes cantidades de datos, así como proporcionar técnicas de tratamiento de datos para su posterior análisis. Por este motivo, es necesario el uso de tecnologías informáticas que lo posibiliten.

### 1.1. Objetivos de un CRM

La finalidad que persigue un CRM es maximizar los beneficios, y de ahí radica su interés para la incorporación en el mundo de los negocios. Para conseguir esta meta es preciso tener en cuenta las siguientes consideraciones:

- **Mayor conocimiento del cliente y personalización del trato**  
Incorporando un sistema CRM se permite identificar y conocer a los clientes, y por tanto personalizar con un mayor nivel de detalle las ofertas y el trato recibido. El CRM dispone de una gran cantidad de datos sobre los clientes que podrán ser utilizados para categorizar al

cliente, conocer su rentabilidad actual y futura, su grado de fidelización y las posibles acciones a realizar.

Un sistema CRM mantiene toda la información de un cliente centralizada, evitando así posibles incoherencias o datos no actualizados. De esta manera, posibilita acceder uniformemente a la información de un cliente por parte de cualquier usuario autorizado de la empresa, atendiendo a los distintos roles. Los clientes podrán conectarse al sistema desde distintos medios (teléfono, internet, puntos de venta, ...).

Un cliente responde en cada momento a un perfil concreto a lo largo de su estancia en la base de datos. Además dicho perfil puede experimentar cambios (mutaciones), es decir, el cliente dependiendo de su actuación podrá adoptar un perfil distinto al actual. Este es un comportamiento muy normal dentro de un CRM. La empresa deberá encaminar a los clientes hacia aquellos perfiles que proporcionen un mayor beneficio. Por ejemplo los clientes de la empresa pasan de ser jóvenes a tener familia y posteriormente a estar jubilados, es por este motivo que los perfiles de cada cliente pueden cambiar (y cambiarán) a lo largo del tiempo.

Debido a su facilidad de implantación y flexibilidad, la mayor parte de empresas, sobretodo del sector servicios, han decidido adoptar un sistema CRM que les ayude a conocer de un modo más completo a sus clientes.

Por otra parte, un sistema CRM tiene como propiedad la escalabilidad. Es por ello que el sistema posee gran habilidad para, o bien manejar el crecimiento continuo de trabajo de manera fluida, o bien para estar preparado para hacerse más grande sin perder calidad en los servicios ofrecidos.

- **Aumento de la satisfacción y lealtad de los clientes**

Los datos contenidos en la base de datos también deberán ser usados para mantener al cliente con la mayor satisfacción posible. Esto mejorará su fidelidad hacia la empresa y será más difícil que abandone sus servicios. Mantener al cliente satisfecho puede contribuir a captar nuevos clientes ya que el "boca a boca" es uno de los mejores métodos publicitarios.

- **Aumento de las ventas**

Un mayor conocimiento de los clientes permite conocer mejor cuáles son las preferencias de cada uno, y de esta manera, personalizar las

propuestas y ofrecer a los consumidores los servicios que más se adecuan a sus necesidades.

La fidelidad de un cliente puede marcar la frecuencia de compras y el precio que esté dispuesto a pagar. Cuando la fidelidad es alta, en general entre los clientes, la empresa puede plantearse una subida en los precios.

- **Rápida obtención de resultados**

Este es uno de los principales objetivos de un CRM. La rapidez de obtención de datos viene determinada por el índice de ROI. Este valor permite saber cual es la relación que existe entre la ganancia neta obtenida y la inversión realizada.

Mediante este índice se permite a las empresas disminuir la incertidumbre, e incrementar la seguridad, en la toma de decisiones en materia de sus proyectos de inversión.

- **Reducción de los costes de servicio**

Un cliente con fidelidad alta suele tener un menor coste de atención al cliente porque conoce los servicios y productos de la empresa. Además, la empresa conoce mejor al cliente y sabe cómo debe actuar.

Una empresa acostumbrada a rotaciones de empleados puede tener problemas para manejar información. Este riesgo disminuye si la empresa dispone de un sistema CRM ya que los contactos quedan registrados en la base de datos y es muy difícil perder información. También evita los problemas provenientes de aquellos trabajadores que abandonan la empresa, siempre que hayan usado la base de datos para almacenar la información acerca del cliente.

## 1.2. Problemas de un CRM

Para que un sistema CRM funcione sin problemas es necesario realizar un diseño previo de la estrategia de relación con los clientes. Este es uno de los principales motivos por los que el CRM no funcione correctamente. Por ello, antes de implantar el sistema debemos tener claro la homogeneización en cuanto se creen familias de productos/servicios de comportamiento similar por sus características de valor, funcionalidad y precio.

Es importante establecer los distintos perfiles de clientes que puedan existir y se establezcan unos objetivos de ventas y satisfacción razonables. Los beneficios del CRM son tan atractivos que muchas empresas se olvidan de la

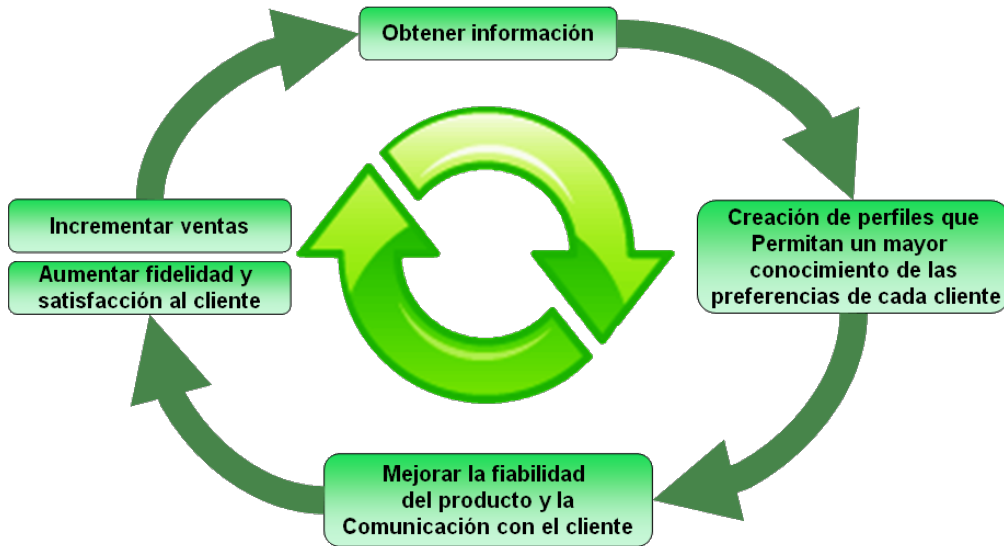


Figura 1.1: Ciclo de explotación de un CRM

importancia de conocer con qué clientes se va intentar establecer una relación y qué se pretende conseguir con ella.

En algunas ocasiones las empresas tienen la mentalidad de establecer una larga relación con el cliente, mientras que el cliente no lo desea, lo que genera una pérdida en la eficiencia del uso del recurso.

Existe la idea de que cuanto más tecnología mejor. La realidad es que en algunos casos no es necesario el uso de tecnología avanzada. Esto supone un gasto importante para la empresa que no se amortiza.

Es recomendable no bombardear al cliente con demasiada información ya que puede sentirse agobiado. La información a enviar al cliente debe ser elegida previamente con el objetivo de informar sobre temas que son de su interés.

## Capítulo 2

# Objetivo del proyecto

El objetivo principal del proyecto es diseñar y especificar un Sistema de Gestión de Clientes Inteligente (CRM-I). El sistema podrá ser utilizado por cualquier empresa que se dedique a la venta y asistencia a clientes. Las capacidades del producto funcionarán mucho mejor con grandes volúmenes de clientes y de transacciones, debido al tratamiento estadístico de la información, lo que requiere que las muestras sean significativas.

Un CRM es un sistema software que utiliza una empresa para *guardar toda la información de las interacciones con cada cliente* (llamadas, consultas, compras, etc). El gran *volumen de datos* generado por esta actividad muestra la necesidad de implementar una aplicación que los procese de la manera más automática posible, para obtener unos datos que permitan actuar de forma más eficiente.

El sistema también puede administrar la relación con los clientes, incorporando funcionalidades de toma de decisiones, y siendo capaz de proponer las acciones más adecuadas que se aplicarán a cada perfil de cliente, así como proporcionar herramientas para la gestión de los mismos. Un CRMI es una potente herramienta de marketing. Más aún cuando incorpora capacidades de toma de decisiones inteligentes, en un dominio con exceso de información sin correlación aparente. Para su uso debe establecerse ciertas premisas y procedimientos.

Este proyecto adoptará un comportamiento similar. El sistema será capaz de interpretar toda clase de incidencias, y reaccionará lanzando una serie de acciones en base a la experiencia adoptada y recogida, para ser tratada mediante algoritmos de IA (Inteligencia Artificial).

Una empresa tendrá varios departamentos, entre ellos, nos interesan los siguientes cuatro:

- Ventas
- Marketing
- Help-desk (consultas)
- Servicio (reparación, mantenimiento, SAT)

El sistema CRM se debe de ocupar de cada uno de estos temas, haciendo la gestión o acción necesaria cada vez que la empresa hace una oferta, publicita, u ofrece soporte.

En una empresa, es muy importante conocer al cliente. Por ello, el trato y las opciones deben estar adaptadas lo más posible a cada uno. Las estrategias de marketing, cada vez, incluyen más el acercamiento del cliente. En un mundo en el que todo está a la distancia de un clic de ratón, la competencia es máxima. Los objetivos de la mayoría de las empresas están bastante relacionados con la captación de clientes y su fidelización. Para ello, hace falta poner en marcha *mecanismos inteligentes de negocio*.

Para cada cliente tenemos información detallada de sus datos personales y un histórico de cliente donde registramos todas las incidencias y cómo se han subsanado. El histórico de cada cliente se guarda con mucho detalle. De esa forma cada acción realizada sobre el cliente y cada reacción del cliente será almacenada para su posterior análisis y se usarán cuando llegue una nueva incidencia. Teniendo en cuenta la experiencia adquirida, el sistema será capaz de proponer la acción / acciones correspondientes. Para ello el sistema tendrá un módulo de inteligencia artificial que contendrá un conjunto de reglas y mecanismos de inferencia que propondrá las acciones requeridas.

Como se explicó anteriormente en la sección Antecedentes, aunque cada empresa deberá adaptar el modelo según su actividad, los objetivos de las empresas en general se pueden resumir en lo siguiente:

- Incrementar las Ventas
  - Captación de clientes
  - Ofrecer a los clientes antiguos productos que son susceptibles de compra
  - Satisfacer a los clientes
- Disminuir los Costos

- Mejorar la productividad
- Reducir el número de consultas a los clientes
- Reducir errores en facturación y despacho de productos

En lo referente a la aplicación que estamos diseñando, la captación de clientes implicaría que la empresa tuviera una buena imagen y popularidad, y eso suele proceder de las opiniones de los que ya son clientes.

A los clientes les gusta que se muestre lo que busca, sin mucho esfuerzo. Por lo general, interesa “resolver los problemas” de la forma más rápida y sencilla posible. Si un cliente queda descontento de manera reiterada, es posible que se vaya a la competencia, con lo cual la empresa dejaría de percibir ingresos.

Por otra parte, es un buen método de reducir costos. Sabiendo lo que un cliente necesita para sentirse satisfecho, se pueden reducir el número y la duración de las consultas, con lo que, a la vez que el cliente no se cansa ni se aburre, al estar en contacto con la operadora al teléfono, se pueden reducir el número de personas atendiendo en las centralitas, atendiendo más consultas en el mismo tiempo. Además, al guardar todos los datos en la base de datos, es más fácil acceder a ellos, haciendo un archivo de los mismos mucho más fiable y eficiente. Con ello se evitan los errores y se contribuye a la satisfacción del cliente.

Por último, y dado el carácter “inteligente” del sistema diseñado, se procesarían los datos recogidos, los cuales, tras un estudio por similitudes, generarían un conjunto de reglas (aprendizaje) que servirán para, anticipándose a lo que un cliente es susceptible de necesitar, atenderle de la mejor manera posible. Dicho estudio se realizaría periódicamente desde un sistema estadístico, que aportaría una serie de normas a seguir para realizar una gestión lo más eficiente posible, es decir, conseguir maximizar el beneficio, manteniendo satisfechos a los clientes.

En cuanto a la implementación, dado que los objetivos eran mayoritariamente el diseño y la especificación de un sistema CRM, no se ha tratado a fondo, si bien se ha creado un prototipo, creando una base de datos utilizando los comandos del lenguaje SQL con las herramientas de bases de datos proporcionadas en la facultad. Nos hemos servido para ello de los privilegios de administrador, para crear tablas e introducir datos, aunque también se ha diseñado un formulario de entrada de datos para acceder de forma rápida y sencilla a la funcionalidad de la aplicación. Asimismo se utilizaría una interfaz de integración con un programa del estilo de SPSS (Statistical Package for the Social Sciences) para hacer un discriminante, ver qué variables tiene, etc.

## 2.1. Nuestro proyecto

El Sistema CRM sobre el que hemos estado trabajando consta de tres funcionalidades principales, relacionadas con los tres tipos de incidencias que distinguimos: Antes de nada, es necesario definir lo que es “incidencia”: todo tipo de sucesos que el cliente comunica a una empresa para resolver un problema con un producto.

Las incidencias las clasificamos en tres tipos:

- **Averías:** Son aquellas incidencias que llegan debido a un mal funcionamiento de un producto.
- **Reclamaciones:** Llamaremos a las incidencias que nos llegan por un servicio contratado por el cliente que no realiza la función deseada.
- **Consultas:** Son incidencias en las que se pide información a la empresa y ésta responde en el menor plazo posible.

Para su gestión se dispone de un modelo de comportamiento que se explicará a continuación.

**Los modelos de Reclamaciones y Averías** funcionan de manera muy similar.

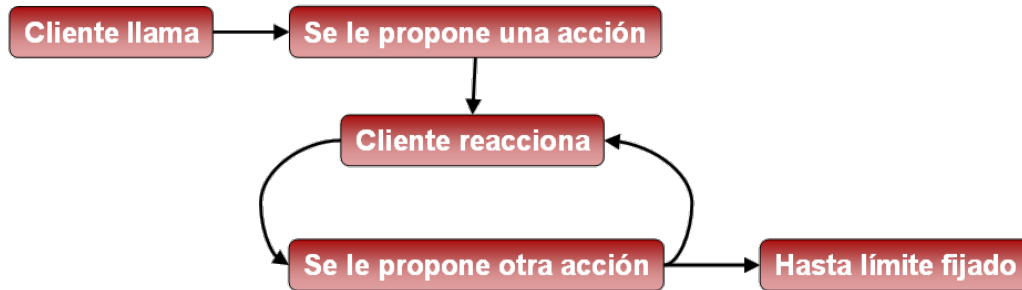
En ellos, el cliente llama, informando de una avería o enunciando una queja hacia un producto o servicio, de tal manera que recibe una respuesta de acuerdo a la situación.

El flujo de proceso se basa en el concepto de **ACCIÓN - REACCIÓN:**

Cuando un cliente llama, se le propone una opción, a lo que reacciona de una manera determinada. Si la conversación no termina en eso, se le vuelve a ofrecer otra opción, o bien una subopción, de forma que los pares acción-reacción van conformando sucesivas iteraciones, hasta su finalización, solucionando el problema o desistiendo de ello.

Cliente llama → Se le propone una opción → Cliente reacciona → Se le propone otra opción → ... → Hasta que se solucione la incidencia (comprando o desistiendo) o un límite fijado.

Dicho límite se establecería indicando la cantidad máxima de pares acción-reacción que se puede aplicar al cliente, de tal manera que éste no reciba la sensación de que se encuentre en un camino sin salida o en una situación irresoluble.

Figura 2.1: Flujo del proceso *Acción - Reacción*

También hay que tener en cuenta lo ocurrido en las anteriores llamadas (histórico de llamadas) de tal manera que se pueda establecer una relación entre cada acción con su correspondiente reacción, además de realizar un seguimiento del conjunto de acciones realizadas al cliente.

Para que el sistema proponga acciones, necesitamos conocer en qué situación de partida nos encontramos. Esta situación viene dada por los siguientes datos que debemos disponer:

- Conocimiento (datos) del cliente
- Motivo de la queja/avería
- Estado de ánimo del cliente
- Resultado de proponer una acción en una situación (Histórico)
- Listado de acciones posibles
- Comportamiento (reacción) del cliente anteriormente tras las acciones realizadas (acciones propuestas) (Histórico Cliente)

Por ejemplo, al explicar como funciona un aparato electrónico, no se trataría de igual manera a un ingeniero que a un abogado, si, por ejemplo, se ha comprobado que el perfil de la gente que estudia derecho es menos propenso a consultar aspectos técnicos de los productos, interesándose simplemente en que funcione correctamente.

Teniendo en cuenta todos estos datos, el sistema seleccionará las acciones óptimas a realizar, de acuerdo también al conocimiento obtenido.

El conocimiento de este sistema consiste en saber cuáles de las posibles acciones a realizar por parte del servicio de asistencia al cliente son las mejores para que el cliente quede satisfecho, y, a ser posible, siga generando ingresos a la empresa.

El conocimiento se adquiere o bien mediante reglas fijas o bien mediante un proceso de aprendizaje, procesando los datos estadísticamente desde los resultados de toda la información que ha recogido el propio sistema CRM en el pasado.

**El flujo de proceso para el modelo de funcionamiento de las consultas** se comporta de manera diferente.

Las "consultas" se refieren a la llamada que hace un cliente solicitando información de un producto, o una recomendación a la hora de tomar una decisión relativa a un producto o un servicio.

Un cliente llama con el motivo de una consulta. Dependiendo de su perfil y de las reacciones de los clientes a anteriores consultas parecidas, se le dará una respuesta u otra. También puede ocurrir que sea la primera vez que se demanda esa información, por lo que se generaría un ticket para que un experto generara una o varias explicaciones a la consulta.

En el caso de que lo que busque un cliente sea adquirir un producto o contratar un servicio, se le dará una lista de los productos que más han satisfecho a otros clientes con perfiles similares.

Nuevamente, en esta funcionalidad se va a aplicar el proceso Acción-Reacción, pero de una manera diferente. Por ejemplo, si un cliente no reacciona como se esperaba, puede que se trate de que está dentro de un perfil equivocado y haya que corregirlo.

Para cada caso, se llega a una Solución. Dependiendo de si la solución es satisfactoria o no, el sistema estadístico generará unas reglas, que, aplicando ingeniería del Conocimiento, se traducirá en el conocimiento de la aplicación de la que se ha hablado anteriormente.

Los datos pueden ser en parte compartidos con los otros modelos de funcionamiento, sobre todo en lo relativo a los datos de los clientes y sus perfiles, pero evidentemente, habrá datos diferentes propios, como datos del funcionamiento de los productos (características, funcionalidades. . .)

# Capítulo 3

## Metodología de desarrollo

### 3.1. Ciclo de vida del sistema

Definimos el ciclo de vida de un sistema como el proceso que se sigue para construir, entregar y hacer evolucionar el software, desde la concepción de la idea hasta la entrega y el mantenimiento del sistema. Antes de escoger el tipo de ciclo de vida debemos especificar cuáles son las etapas por las que va a seguir la puesta en marcha del sistema. Las etapas por las que transcurre nuestro proyecto son las siguientes:

1. **Análisis:** Construye un modelo de los requisitos
2. **Diseño:** A partir del de análisis se deducen las estructuras de datos, la estructura en la que descompone el sistema y la interfaz de usuario.
3. **Implementación:** Construye el sistema. Se genera el código a ejecutar.
4. **Pruebas:** Se comprueba que se cumplen criterios de corrección y calidad.
5. **Mantenimiento:** En esta fase, que tiene lugar después de la entrega se asegura que el sistema siga funcionando y adaptándose a nuevos requisitos.

Una vez descritas cada una de las etapas elegimos el estilo de ciclo de vida que usaremos para el desarrollo de nuestro proyecto. Todos los ciclos de vida tienen características positivas y negativas, los creadores del sistema deben elegirlo acorde con sus pretensiones. Debido a que un CRM-I es un

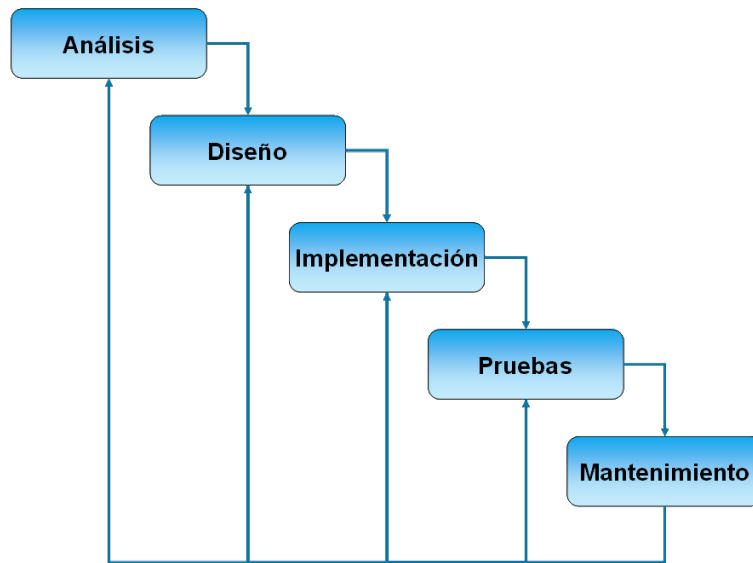


Figura 3.1: Modelo en cascada

sistema muy propenso a cambios en el diseño durante toda fase del ciclo de vida, elegimos un ciclo de vida en cascada. El diagrama que describe el funcionamiento de este modelo puede verse en la Figura 3.1

Esta metodología elegida, de modelo en cascada, facilita una planificación sencilla. Podemos pasar por alto detalles concretos y después, en una nueva planificación, llevarlos a cabo. En nuestro caso, en una primera instancia nos ocupamos del funcionamiento básico de un CRM y más tarde, abordamos los aspectos más avanzados, como el módulo de inteligencia.

La calidad de este tipo de proyectos es muy alta dado que una vez terminada una versión puede mejorarse e incluso rediseñarse para que su funcionamiento sea más eficiente, fundamentalmente en las fases de interacción con el usuario del sistema y en la estructura del árbol de decisión.

Además, esta metodología estuvo bastante acertada, pues los requisitos no los tuvimos completados hasta casi la fase final. Además, dado el carácter del proyecto, principalmente descriptivo, no nos resultó muy traumático en el momento de hacer los cambios en la implementación.

## 3.2. Cronograma del proyecto

Tras acordar el proyecto con el profesor a finales de septiembre, coincidiendo con el comienzo del curso, nos pusimos manos a la obra. Inicialmente la planificación del proyecto tal como pensó el profesor sería:

- 1er trimestre: Diseño del sistema: Requisitos, especificaciones, casos de uso, etc.
  - En esta primera etapa son muy importantes las reuniones con el profesor para contrastar opiniones y poder realizar las especificaciones de acuerdo a la idea del sistema del profesor.
- 2º trimestre: Diseño de la base de datos.
  - El diseño de la base de datos se realizó de acuerdo a las especificaciones establecidas en la etapa anterior.
- 3er trimestre: Implementación del prototipo.
  - Se realizó una pequeña implementación del sistema a modo de prototipo. La implementación se realizó en Java utilizando JSP para la interfaz y servlets para la lógica de la aplicación así como librerías de etiquetas. Se utilizó la base de datos diseñada en la etapa anterior implementándola en MySQL.

Al principio del desarrollo discutimos cual sería la metodología de desarrollo para llevar a cabo el proyecto en nuestro grupo. Coincidimos con el profesor en la realización de reuniones semanales para la revisión del trabajo realizado durante la semana. En las reuniones exponíamos los documentos y trabajos pedidos por el profesor para su valoración y corrección. En algunas ocasiones los documentos precisaban ser corregidos y en otros ampliados. Además en cada reunión se proponían nuevos contenidos a añadir.

Las reuniones tenían la siguiente estructura:

1. Exposición del trabajo realizado.
2. Comentarios sobre lo anteriormente expuesto.
3. Punto de vista por parte del profesor. Estableciendo un nuevo objetivo.

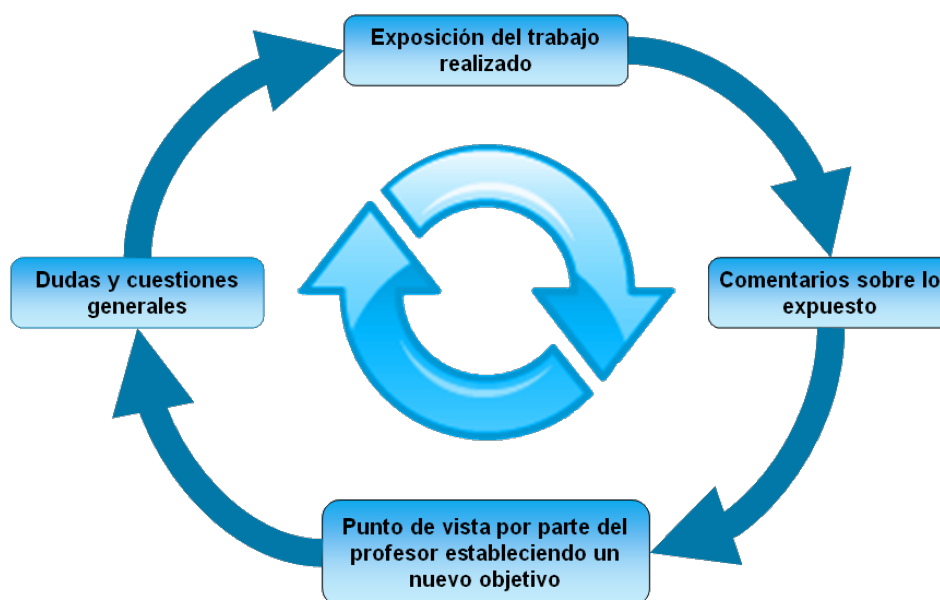


Figura 3.2: Ciclo de funcionamiento de las reuniones

#### 4. Dudas y cuestiones generales.

Además se han establecido actas para cada reunión realizada, en cada una de las cuales constan los temas tratados, juntos con los contenidos presentados o cualquier cuestión relacionada con la orden día. Pueden consultarse en la sección de anexos. La figura 3.2 ilustra la dinámica de funcionamiento de las reuniones se trata de un comportamiento cíclico hasta la obtención de un nuevo objetivo para la siguiente reunión.

Después de cada reunión nos dividíamos el trabajo para cumplir el objetivo. En numerosas ocasiones, y debido a las características del proyecto realizábamos trabajos equivalentes. Puede parecer redundante pero en ningún caso lo es ya que aporta diferentes puntos de vista a las soluciones posibles para abordar el objetivo. El trabajo realizado necesitaba modificaciones a fin de enriquecerse con las aportaciones de cada miembro.

Para organizar la documentación generada en las reuniones utilizamos una Wiki (<http://si2008.wikispaces.com/>). Nuestra Wiki está dividida en varias secciones, correspondientes a las actas generadas tras las reuniones, así como otra sección de la memoria. Cada página dispone de imágenes, diagramas y otros recursos que complementan su contenido. Además el uso de

la Wiki nos proporciona grandes facilidades para realizar la documentación, especificaciones del proyecto, realización de la memoria, etc.

### 3.3. Tecnologías aplicadas en el desarrollo

#### 3.3.1. Wiki

Una wiki es un sitio web cuyas páginas pueden ser editadas por distintos usuarios a través del navegador web. Los usuarios pueden crear, modificar o eliminar una página que comparten. Las páginas de la wiki tienen títulos únicos. Si se escribe el título de una "página-wiki" en algún lugar de la wiki, esta palabra se convierte en un hipervínculo a la página web.

#### Características principales:

- Permite la creación colectiva de documentos en un lenguaje simple de marcas utilizando un navegador web.
- Es mas sencillo de usar que una base de datos.
- La mayoría de Wikis están abiertas al publico. Algunas de ellas no obligan a registrar una cuenta al usuario.
- Facilita su edición, a diferencia de los sistemas tradicionales donde resulta más difícil que los usuarios del sitio contribuyan a mejorarlo.
- Dispone de un sistema de control de versiones en donde las versiones antiguas nunca se eliminan y pueden restaurarse.
- Es muy fácil de combinar con el lenguaje HTML

#### 3.3.2. JSP y Servlets

JavaServer Pages (JSP) es una tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.

JSP permite crear aplicaciones web que se ejecutan en diversos servidores web, de múltiples plataformas, ya que Java es un lenguaje multiplataforma. Las páginas JSP están compuestas de código HTML/XML con etiquetas

especiales para programar scripts de servidor en sintaxis Java. Por tanto, las JSP pueden desarrollarse con un editor HTML/XML habitual.

El motor de las páginas JSP está basado en los servlets de Java, programas en Java que se ejecutan en el servidor, aunque el número de desarrolladores que pueden afrontar la programación de JSP es mucho mayor, dado que resulta mucho más sencillo aprender que los servlets.

En JSP creamos páginas de manera parecida a como se crean en ASP o PHP -otras dos tecnologías de servidor-. Generamos archivos con extensión .jsp que incluyen, dentro de la estructura de etiquetas HTML, las sentencias Java a ejecutar en el servidor. Antes de que sean funcionales los archivos, el motor JSP lleva a cabo una fase de traducción de esa página en un servlet, implementado en un archivo class (Byte codes de Java). Esta fase de traducción se lleva a cabo habitualmente cuando se recibe la primera solicitud de la página .jsp, aunque existe la opción de compilar en código para evitar ese tiempo de espera la primera vez que un cliente solicita la página.

Para trabajar con JSP es necesario conocer HTML y JAVA como lenguaje de programación Orientado a Objetos por completo. También es recomendable tener una ligera idea sobre Servlets, lo que nos dará una mejor idea del funcionamiento interno del motor JSP.

Para que JSP pueda ser utilizado es necesario un servidor web. Tomcat, es el contenedor de servlets usado en la referencia oficial de implementación de JSP.

### 3.3.3. MYSQL

Es un gestor de bases de datos relacionales multiusuario y multihilo. Debido a su gran sencillez de instalación y uso optamos por la utilización de este sistema para realizar las pruebas. Su gran aceptación se debe en parte, a la existencia de numerosas librerías y herramientas que permiten su uso. Además dispone de soporte para gran cantidad de lenguajes de programación, y, en concreto para Java. Es fácil su instalación y configuración.

El servidor está disponible como un programa separado para usar en un entorno de red cliente/servidor. También está disponible como biblioteca y puede ser incrustado (linkado) en aplicaciones autónomas. Dichas aplicaciones pueden usarse por sí mismas o en entornos donde no hay red disponible.

### Características

1. Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
2. Soporta gran cantidad de tipos de datos para las columnas.
3. Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc).
4. MySQL server tiene soporte para comandos SQL para chequear, optimizar, y reparar tablas. Estos comandos están disponibles a través de la línea de comandos y el cliente mysqlcheck. MySQL también incluye myisamchk, una utilidad de línea de comandos muy rápida para efectuar estas operaciones en tablas MyISAM.
5. Soporte completo para operadores y funciones en las cláusulas de consultas SELECT y WHERE.
6. Gran portabilidad entre sistemas.
7. Soporta hasta 32 índices por tabla.
8. Gestión de usuarios y contraseñas, manteniendo un muy buen nivel de seguridad en los datos.

#### 3.3.4. JDBC

JDBC es un API (Application programming interface) que describe o define una librería estándar para acceso a fuentes de datos, principalmente orientado a Bases de Datos relacionales que usan SQL (Structured Query Language). JDBC es un interfaz para acceso a motores de bases de datos y también define una arquitectura estándar, para que los fabricantes puedan crear los drivers que permitan a las aplicaciones java el acceso a los datos.

Las características más importantes de JDBC son:

- JDBC está orientado a permitir ejecutar comandos SQL directamente, y procesar los resultados obtenidos. Esto supone que será tarea del programador crear APIs de más alto nivel apoyándose directamente sobre JDBC.

- Es compatible con el lenguaje SQL. Cada motor de Base de Datos implementa una amplia variedad de comandos SQL, y muchos de ellos no tienen porque ser compatibles con el resto de motores de Base de Datos. JDBC, para solventar este problema de incompatibilidad, realiza lo siguiente:
  - Una aplicación Java puede hacer uso de toda la funcionalidad que provea el motor de Base de Datos, con el riesgo de que esto pueda producir errores o no en función del motor de Base de Datos.
  - Con el objetivo de conseguir que un driver sea compatible con SQL, se obliga a que al menos, el driver cumpla el Estándar ANSI SQL 92.
- Es reutilizable sobre cualquier otro API de acceso a Bases de Datos, o más concretamente con ODBC (Open Database Connectivity)
- JDBC debe ser un API simple, para después ampliarse de una manera más fácil.
- Es fuertemente tipado, y siempre que sea posible de manera estática, es decir, en tiempo de compilación, para evitar errores en tiempo de ejecución.
- JDBC debe mantener los casos comunes de acceso a Base de Datos lo más sencillo posible:
  - Mantener la sencillez en los casos más comunes (SELECT, INSERT, DELETE y UPDATE)
  - Hacer realizables los casos menos comunes: Invocación de procedimientos almacenados...
- No es un problema crear múltiples métodos para múltiple funcionalidad. Es preferible incluir gran cantidad de métodos, en lugar de hacer métodos grandes y complejos con gran cantidad de parámetros.

### 3.3.5. XHTML

XHTML (Lenguaje de Marcado de Hipertexto Extensible) es una versión más estricta y limpia de HTML. Empieza precisamente con el objetivo de remplazar a HTML ante su limitación de uso con las cada vez más abundantes herramientas basadas en XML. XHTML extiende HTML 4.0 combinando

la sintaxis de HTML, diseñado para mostrar datos, con la de XML, diseñado para describir los datos. Ante la llegada al mercado de un gran número de dispositivos, XHTML surge como el lenguaje cuyo etiquetado, más estricto que HTML, va a permitir una correcta interpretación de la información independientemente del dispositivo desde el que se accede a ella. XHTML puede incluir otros lenguajes como MathML, SMIL o SVG, al contrario que HTML.

XHTML exige una serie de requisitos básicos a cumplir en lo que a código se refiere. Entre estos requisitos básicos se puede mencionar una estructuración coherente dentro del documento donde se incluirían elementos correctamente anidados, etiquetas en minúsculas, elementos cerrados correctamente, atributos de valores entrecomillados, etc.

Las ventajas más evidentes que ofrece el migrar a XHTML son:

- Los documentos XHTML se establecen en base a las reglas XML. Por tanto, pueden ser visualizados, editados y validados por cualquier herramienta XML.
- Los documentos XHTML pueden escribirse para que funcionen igual o mejor que lo hacían antes tanto en los agentes de usuarios conformes a HTML 4.0 como en los nuevos agentes conformes a XHTML 1.0.
- Los documentos XHTML pueden contener aplicaciones (por ejemplo applets o scripts) que se basen en DOM y que modifiquen la propia estructura del documento XHTML.
- Permite insertar en el documento XHTML nuestras propias marcas que no tienen por qué estar definidas en el estándar general. Esto es lo que se llama modularización XHTML.
- Hacer las páginas legibles por personas discapacitadas. Al no tener marcas que indiquen forma de representación entremezcladas con el propio contenido, es mucho más fácil construir agentes de usuario que lean ese contenido a personas invidentes o lo pasen a otros formatos como Braille.

### 3.3.6. CSS

El lenguaje HTML está limitado a la hora de aplicarle forma a un documento. En sus orígenes no se contempló estas opciones ya que estaba orientado a otros temas.

Para solucionar estos problemas los diseñadores han utilizado técnicas tales como la utilización de tablas imágenes transparentes para ajustarlas, utilización de etiquetas que no son estándares del HTML y otras. Estas "trampas" han causado a menudo problemas en las páginas a la hora de su visualización en distintas plataformas.

Finalmente, otro antecedente que ha hecho necesario el desarrollo de esta tecnología consiste en que las páginas web tienen mezclado en su código HTML el contenido del documento con las etiquetas necesarias para darle forma. Esto tiene sus inconvenientes ya que la lectura del código HTML se hace pesada y difícil a la hora de buscar errores o depurar las páginas. Aunque, desde el punto de vista de la riqueza de la información y la utilidad de las páginas a la hora de almacenar su contenido, es un gran problema que estos textos estén mezclados con etiquetas incrustadas para dar forma a estos: se degrada su utilidad.

CSS u Hojas de Estilo en Cascada (Cascading Style Sheets), es una tecnología simple que describe cómo será mostrado un documento (normalmente páginas web) en los distintos medios soportados (pantalla, impresora, etc.) De esta forma se consigue el control total sobre estilo y formato de sus documentos.

CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los Estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento.

CSS funciona a base de reglas, es decir, declaraciones sobre el estilo de uno o más elementos. Las hojas de estilo están compuestas por una o más de esas reglas aplicadas a un documento HTML o XML. La regla tiene dos partes: un selector y la declaración. A su vez la declaración está compuesta por una propiedad y el valor que se le asigne.

```
h1 {color: red;}
```

`h1` es el selector

`{color: red;}` es la declaración

De esta forma todos los elementos de los documentos que incluyan dicha regla establecerán la propiedad `color` el valor `red` (rojo), de tal forma que se mostrarán los encabezados de nivel 1 en color rojo.

### 3.3.7. CVS

CVS es un sistema de control de versiones usado para mantenimiento de código fuente (grupos de ficheros en general) extraordinariamente útil para grupos de desarrolladores que trabajan cooperativamente usando alguna clase de red.

CVS permite a un grupo de desarrolladores trabajar y modificar concurrentemente ficheros organizados en proyectos. Por ello, dos o más personas pueden modificar un mismo fichero sin que se pierdan los trabajos de ninguna.

Como añadido a lo anterior, CVS guarda todas las versiones antiguas de los ficheros. Esto permite recuperar en cualquier momento versiones anteriores a la actual.

Dado que trabaja con ficheros ASCII es igual de útil para trabajar con código fuente de programas o con toda clase de documentos siempre que su formato sea completamente de texto, como pueden ser ficheros sgml/html/xml.

Conceptos:

**Repositorio:** Jerarquía de directorios alojada en el servidor CVS que contiene diferentes módulos a disposición de los usuarios.

**Módulo:** Árbol de directorios que forma parte del repositorio. Cuenta con un nombre identificador gracias al cual podremos trabajar con él de forma selectiva.

Se pueden visualizar los archivos del repositorio junto con sus correspondientes versiones en la siguiente URL <http://cvs.berlios.de/cgi-bin/viewcvs.cgi/banderas/CRMI/>

# Capítulo 4

## Análisis

Como fase de comienzo en el proyecto es necesario realizar el análisis de requisitos junto la organización del conocimiento necesario para su implantación. En la fase de análisis se determinarán los plazos que se requieren para su puesta en marcha. Dicha información se contrastará con cada Jefe de Área utilizando documentación existente y reuniones en las que se expondrán el alcance.

### 4.1. Toma de requerimientos

La toma de requerimientos o Análisis de requerimientos es necesaria para determinar las necesidades del sistema y posteriormente extraer los casos de uso.

En esta fase extraeremos los requisitos del sistema. Divididos en requisitos funcionales y no funcionales tal como se detalla a continuación.

Se diferenciará entre requisitos funcionales y requisitos no funcionales. Los requisitos funcionales son las declaraciones de los servicios que debe proporcionar el sistema. Es necesario especificar tanto lo que el programa debe hacer como las limitaciones funcionales que le impidan lo que el programa no debe hacer. Los requisitos no funcionales son aquellos que describen las facilidades que debe proporcionar el sistema en cuanto a rendimiento. Estos últimos definen propiedades y restricciones del sistema: tiempos de respuesta, requisitos de almacenamiento, etc.

### 4.1.1. Requisitos funcionales

- **RF - 1:** El sistema guardará información acerca de los productos concretos tales como: nombre, garantía, modelo, además será necesario identificar cada producto de forma unívoca a través del S/N (Serial Number). Como es natural, pueden existir varios modelos para un mismo producto. Por ello será necesario poder recabar información de cada modelo, como puede ser la versión o el año de creación.
- **RF - 2:** Para los clientes el sistema ha de manejar sus datos personales: nombre, apellidos, fecha de nacimiento, . . . así como un atributo adicional para identificarlo y distinguirlo del resto de clientes.
- **RF - 3:** El sistema recogerá información acerca de las acciones de los clientes
- **RF - 4:** El aprendizaje del sistema es importante para determinar de qué forma suele actuar cada cliente. Esto nos permite averiguar cuáles son sus preferencias y en general los patrones de comportamiento. Los datos se extraerán bien de la forma de actuar de cada cliente o mediante cuestionarios. A partir de estos datos podemos crear perfiles de clientes que actúen de la misma forma.
- **RF - 5:** El sistema registrará cada incidencia que el cliente plantee. Cada incidencia recogerá datos como modelo afectado, cliente, así como alguna información extra relacionada con la forma de proceder o solventar. Además se extraerán datos relevantes relacionados con el funcionamiento de cada producto y sobre el uso que cada usuario hace. Dependiendo del tipo de incidencia puede llevarse a cabo distintas alternativas para la solución de la misma como por ejemplo sustituir el producto por otro de características similares. Incluso dependiendo del tipo de cliente (por ejemplo un VIP) que tiene la incidencia se puede actuar de distinta forma como por ejemplo ofrecer un modelo superior a un VIP.
- **RF - 6:** Otro aspecto importante a destacar es el funcionamiento de cada modelo. Muchas veces los clientes no entienden el funcionamiento de un producto y existen dudas bastantes comunes. Por ello se crean manuales de usuario que ayuden a comprender su funcionamiento y contienen una lista de preguntas más frecuentes (FAQ). Además si de manera usual una duda sobre el uso del producto lleva a otra duda el sistema indicará la respuesta a la 2<sup>a</sup> duda relacionada con la 1<sup>a</sup>.

### 4.1.2. Requisitos no funcionales

#### Requisitos de sistema

- **RSi - 1:** El sistema deberá ser capaz de acceder a los datos de la BBDD en un tiempo razonable. No es lógico que tarde varios minutos cuando se produzca un consulta simple a la BBDD.

#### Requisitos de seguridad

- **RSe - 1:** En los casos que se registre una incidencia, deberá ser solucionada en el menor tiempo posible, haciendo uso de los recursos de la empresa que sean oportunos, siempre que la importancia de la incidencia y el cliente lo permita.
- **RSe - 2:** Cada semana se deberá realizar una copia de seguridad de todos los datos con el fin de poder recuperarlos en el caso de que se pierdan por un motivo inesperado.

#### Requisitos de eficiencia

- **REf - 1:** El sistema debe ser capaz de manejar un gran volumen de datos de manera escalable y eficiente. Primando en la complejidad de los algoritmos utilizados ya que el tamaño de los datos es elevado.

## 4.2. Análisis predictivo

El Análisis predictivo utiliza estadística junto con algoritmos de minería de datos. Se basan en el análisis de los datos actuales e históricos para hacer predicciones sobre futuros eventos. Dichas predicciones raramente suelen ser afirmaciones absolutas, pareciéndose más a eventos y su probabilidad de que suceda en el futuro.

En el mundo de los negocios los modelos predictivos explotan los patrones de comportamiento encontrados en el pasado para poder identificar riesgos y oportunidades. Los modelos capturan las relaciones entre muchos factores permitiendo capturar riesgos potenciales asociados a un conjunto de condiciones, guiando así a la toma de decisiones.

En la banca, típicamente, antes de conceder un crédito, préstamo o hipoteca, evalúan el perfil de riesgo de la persona usando un modelo de puntuación.

Los modelos de puntuación tienen en cuenta el comportamiento histórico del cliente, como puede ser el saldo de su cuenta bancaria a lo largo del tiempo, descubiertos, impagados, así como datos “estáticos” del cliente.

El análisis predictivo se utiliza en multitud de campos, aseguradoras, telecomunicaciones, agencias de viaje, farmacéuticas, médicas, etc.

### 4.2.1. Tipos de análisis predictivos

Para llevar a cabo el análisis predictivo existen 3 modelos que pasaremos a describir:

**Modelos predictivos:** Los modelos predictivos analizan los resultados anteriores para evaluar qué probabilidad tiene un cliente para mostrar un comportamiento específico en el futuro con el fin de mejorar la eficacia de marketing. Esta categoría también incluye modelos que buscan patrones discriminatorios de datos para responder a las preguntas sobre el comportamiento del cliente, tales como la detección de tipos de fraudes. Los modelos de predicción a menudo realizan cálculos en tiempo real, durante las operaciones, por ejemplo, para evaluar el riesgo o la oportunidad de un determinado cliente o transacción, a fin de orientar una decisión.

**Modelos descriptivos:** Los modelos descriptivos describen las relaciones en los datos para poder clasificar a los clientes en grupos. A diferencia de modelos de predicción que se centran en predecir el comportamiento de un único cliente (como el riesgo de crédito), los modelos descriptivos identifican diferentes relaciones entre clientes o productos. Pero los modelos descriptivos no clasifican a los clientes según su probabilidad de tomar una acción en particular. Los modelos descriptivos se utilizan a menudo “offline” por ejemplo, clasificar a los clientes por las preferencias de los productos según la etapa de la vida. Las herramientas de modelado descriptivo pueden ser utilizadas para desarrollar modelos basados en agentes simulando una gran cantidad de agentes individuales pudiendo predecir acciones futuras

**Modelos de decisión:** Los modelos de decisión describen la relación entre todos los elementos de una decisión – los datos conocidos (incluidos los resultados de los modelos de predicción), la decisión y el plan de variables y valores que determinan la decisión – con el fin de predecir los resultados de las decisiones de muchas variables. Estos modelos pueden ser utilizados en optimización.

Para obtener unos buenos resultados en las acciones propuestas a los usuarios es necesario llevar un seguimiento exhaustivo de todos los movimientos realizados por el cliente. Además resultará de gran importancia datos adicionales que describan el perfil del cliente con el objetivo de poder proponerle acciones más adecuadas, no solo atendiendo a su interacción con el CRM si no también atendiendo a su edad, profesión, condición social, etc.

Llevar esto a cabo es una tarea compleja y requiere de gran cantidad de datos para que los resultados sean fiables. El sistema hará uso de distintas herramientas para obtener todo el conocimiento que necesita. Las distintas herramientas serán:

- Herramientas estadísticas
- Data Mining
- Análisis de archivos de traza

### 4.3. Información de los clientes

El sistema realiza un seguimiento continuo de todas las interacciones de los clientes con el sistema. De forma conceptual los datos sobre los clientes pueden provenir de distintas fuentes. Por ello el módulo de análisis (mediante técnicas de Data Mining) recuperará distintos datos para describir al cliente:

- Quién es el cliente (Datos personales, etc)
- Qué promociones se ofrecieron al cliente (¿Qué acciones se le han hecho al cliente?)
- Cómo reaccionó el cliente a estas promociones (¿Qué reacciones ha tenido el cliente?)

Conociendo estos 3 tipos de datos sobre un cliente, sería suficiente para que el sistema a través de técnicas de Data Mining pueda buscar patrones de comportamiento y conocer posibles comportamientos futuros. Sin saber quien es el cliente, qué se hizo con él y cómo reaccionó es imposible que el sistema pueda aprender e inferir nuevos comportamientos.

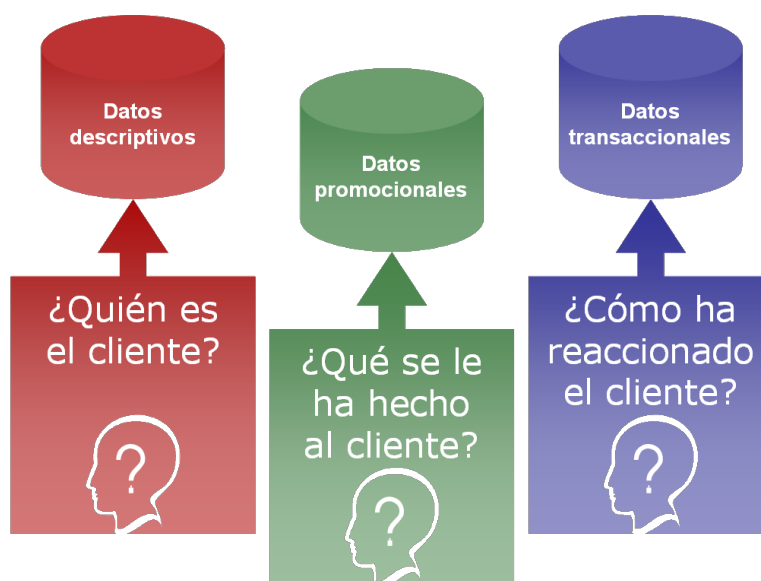


Figura 4.1: Tres tipos de datos sobre clientes

### 1. Quién es el cliente

Tanto para optimizar y rentabilizar las interacciones con los clientes como para optimizar el rendimiento del sistema CRMI, es necesario poder distinguir entre clientes buenos y malos, rentables y no rentables. Para ello, es imprescindible conocer quiénes son y cómo se diferencian.

### 2. Qué promociones se ofrecieron al cliente (acciones realizadas sobre el cliente)

Para saber si las inversiones en promociones son rentables, hay que tener presente qué se hizo para cada cliente. El departamento de marketing suele llevar a cabo muchas pequeñas promociones y necesita poder diferenciarlas, para poder valorar cuáles funcionan y cuáles no.

### 3. Cómo reaccionó el cliente a estas promociones (reacciones del cliente)

Para juzgar el valor real del sistema hay que saber evaluar los resultados. Para ello, es imprescindible saber si el resultado de la promoción fue bueno o malo, información que puede utilizarse para mejorar el sistema en el futuro.

Esta manera de agrupar los datos no es casual. Reflejará las diferencias entre distintos tipos de datos reales almacenados la base de datos. Además, este

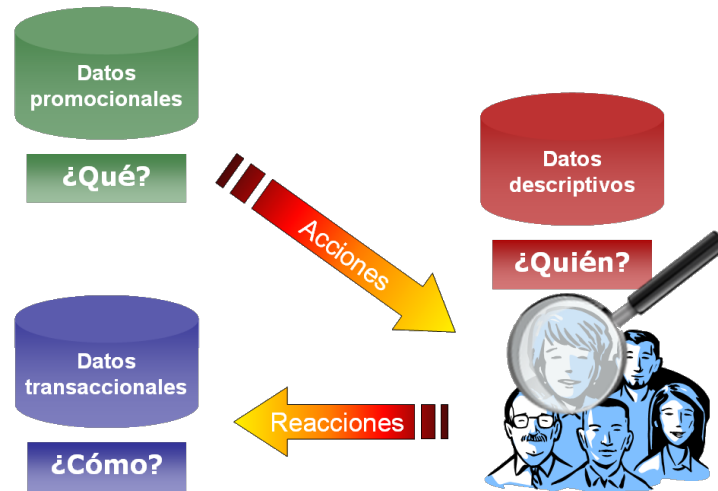


Figura 4.2: Información suficiente para realizar Data Mining

enfoque permite asegurarse de que el sistema está siendo abastecido con los tipos de datos necesarios para la herramienta de DM, cuyas conclusiones, a su vez, permiten llevar a cabo la optimización del sistema de CRM.

### 4.3.1. Datos descriptivos

Los datos descriptivos proporcionan *información* sobre el cliente. Suele ser algún tipo de datos resumidos. Normalmente se almacenan en la base de datos en una sola tabla. La descripción del cliente no es un tipo de datos que cambie muy a menudo, dado que recoge parámetros como edad, sexo, domicilio, número de hijos, e-mail, beneficios familiares e individuales, etc. Esta información se debe contrastar periódicamente con fuentes externas. Además sería deseable pero no siempre posible que la información sea revisada de manera periódica (p.e. una vez al año) para comprobar la validez de los mismos. Debido a las dimensiones de la BBDD dicho proceso podría ser irrealizable.

### 4.3.2. Datos de promociones

Los datos promocionales incluyen información sobre las *acciones* emprendidas para cada cliente. La riqueza de este tipo de datos depende de la

sofisticación del sistema de CRM. Puede ser una simple lista de promociones (acciones) realizadas para el cliente, por ejemplo, envío de catálogos, muestras gratuitas, descuentos, regalos, etc. Además puede ser una información muy precisa e individualizada, por ejemplo, e-mails enviados y las visitas de clientes a las páginas web sugeridas en estos.

Resumiendo, se pueden obtener los siguientes tipos de información:

- **Tipo de promoción**  
Ventas, marketing, publicidad impresa, en radio, en Internet, descuentos, regalos.
- **Descripción de la promoción**  
Color de tarjeta postal, contenido del anuncio radiofónico.
- **Medio**  
Mercados por los que circula la publicidad, portales en Internet que muestran los banners.
- **Tiempo**  
Fecha y quizá hora de la promoción.
- **Descripción del intento**  
Una breve descripción del cliente para el que se concibió la promoción y las razones por las que se eligió la música de fondo utilizada.
- **Financiero**  
Coste fijo y variable de la promoción.

### 4.3.3. 1.3. Datos Transaccionales

Los datos transaccionales engloban los datos referentes a una interacción con el cliente. Pueden incluirlo, desde las llamadas telefónicas realizadas, hasta una el estado de ánimo en cada una de las llamadas, pasando por la lista de productos/servicios adquiridos por el cliente. Estos datos, al igual que los datos promocionales, cambian muy rápidamente. Por ello, se suelen almacenar en estructuras que permiten actualizar y cambiarlos con mucha facilidad. Es un tipo de información muy diferente de la descriptiva, en la que el tipo de datos almacenados no cambia casi nunca. La estructura de los datos transaccionales puede variar drásticamente en un corto período de tiempo, por ejemplo, la introducción de nuevos productos para la venta y la baja de productos más viejos que ya no se venden. Normalmente responden

a procesos que estimulan la conducta del cliente, aletargando (sin generar respuesta) o activando su comportamiento.

## 4.4. Orígenes de los Datos de Cliente

Los datos del cliente se pueden recoger de diferentes fuentes. La partición de los datos, como se comentó anteriormente, tiene su interés ya que el agrupamiento indica de cierta forma de dónde se pueden obtener los datos. Los datos descriptivos provienen de lo que los propios clientes proporcionan o si no de bases de datos adquiridas a proveedores. La información contenida en las bases de datos puede contener tanto información geográfica como preferencias personales informando de la afinidad hacia ciertos sectores o artículos.

### 4.4.1. Internas a la empresa

Un CRM es utilizado frecuentemente por el departamento de marketing, ellos son quienes deciden las políticas de la empresa, campañas a realizar, etc. Por este motivo será fácil obtener datos promocionales, tan solo hace falta que el departamento de marketing lleve control de las acciones realizadas a los clientes.

### 4.4.2. Internet

El volumen de datos que se puede obtener de internet es cada vez mayor. Esto es debido a que cada día mas promociones y transacciones ocurren a través de internet. Además internet es perfecto como medio difusor y observador. Por poner un ejemplo, cualquier acción que realice el cliente con el sistema queda registrada, las páginas que visita, los productos que compra, el tiempo que se detiene en cada página, en cada producto, etc. Toda esta información se guarda en un fichero log para posteriormente ser analizada.

Existen diseños de páginas web que su navegación responde a perfiles definidos de cliente, de forma que conduzca en un mínimo número de accesos a la oferta que realmente está interesado el potencial cliente.

De este modo se genera un rico sistema de información en el cual se pueden medir distintas variables sobre el cliente ( sectores de interés, capacidad de compra, etc.) Aún así hay cuestiones que la herramienta de Data Mining

no es capaz de inferir con seguridad, por ejemplo: ¿Se pueden obtener las verdaderas motivaciones que hicieron a un cliente elegir un determinado artículo?

Actualmente existen en el mercado herramientas capaces de analizar ficheros log. Son capaces de separar los acontecimientos significativos de los que no lo son.

## 4.5. Normalización de información de clientes de diferentes sistemas

### 4.5.1. Almacenes de Datos

El almacén de datos (*Data Warehouse*, DW) es un repositorio que se encarga de conectar y aunar distintas fuentes de información sobre los clientes. El DW suele estar implementado con una única base de datos. En nuestro caso utilizamos una base de datos para el DW para almacenar todos los datos de origen heterogéneo. Además por simplicidad en la misma BBDD se han incorporado algunas tablas que contienen información de gestión para el propio CRMI así como los datos promocionales y transaccionales. El DW solo alberga datos de los clientes que son interesantes para la toma de decisiones. Por rendimiento y prestaciones suele estar alojado en una gran servidor de base de datos que soporte transacciones. El DW servirá de soporte para la toma de decisiones de negocios conforme con los datos almacenados. De ahí radica la importancia de la completitud y exhaustividad de los datos recogidos y almacenados de los clientes.

Un DW necesita identificar al cliente, por lo que se podrá hacer fácilmente utilizando el atributo (Clave primaria ) asignada de cada cliente. Dicha clave primaria coincidirá con su número de DNI.

El DW será una de tantas fuentes que utilice la herramienta de Data Mining para el sistema CRMI. Esto se debe a que el DW almacena gran cantidad de datos haciendo que su capacidad para cambiar se vea mermada. Por otro lado los datos transaccionales y promocionales proporcionan más dinamismo, pudiéndose adaptar más rápidamente a los cambios de mercado y de la competencia. Por este motivo los datos del DW no estarán tan actualizados como debieran, pero debido a su carácter monolítico sería muy costoso actualizarlo constantemente. Algunas partes del DW serán estáticas y otras dinámicas, de esta forma se facilita su actualización.

### 4.5.2. Conectores de Datos

Los conectores son pequeñas piezas software que permiten interconectar fuentes de datos incompatibles entre si. De tal forma que pueda usarse la herramienta de DM y el CRMI. Estas aplicaciones proporcionan capas de abstracción entre la aplicación y los datos. De esta forma se permite:

- Incorporar nuevas y cambiantes fuentes de datos.
- Cambiar el formato de alguna de las fuentes de datos de forma transparente para la aplicación.
- Permitir el traslado de los datos de manera consistente y reproducible de manera que haya posibilidad de validación.

En nuestro proyecto sólo utilizamos un conector de datos: Connector/J proporcionado por MySQL para conectar la BBDD en MySQL desde Java. En caso de cambiar el diseño del sistema, toda la información referente a los datos se encuentra debidamente separada. Esto ha sido posible mediante la utilización de una librería de etiquetas para recuperar información de la BBDD.

Una gran ventaja de utilizar conectores de datos es la separación del diseño y los datos. De tal forma que se facilite el acceso a los mismos por parte del CRMI y DM. Estos conectores puede ser simple código SQL o enteras aplicaciones software proporcionadas por terceros.

## 4.6. Árboles de decisión

Un *árbol de decisión* (decision tree) es un método de análisis que se usa para discriminar información en función de atributos (variables) con un grado de significancia determinada, lo que permitirá identificar la estrategia más apropiada y lograr la eficiencia en el trato que se dará al cliente en curso. Para ellos se genera un grafo (árbol) y sus posibles consecuencias asociando a cada una de ellas la probabilidad que ocurra.

El árbol de decisión usa conjuntamente técnicas de minería de datos (Data Mining, DM) y aprendizaje automático para crear un modelo predictivo, esto es, relacionar las observaciones con las conclusiones. Dentro de los árboles de decisión hay varios tipos (árboles de clasificación, árboles de regresión, etc.), cada uno de ellos sirve para un determinado objetivo. En todos estos tipos de árboles las hojas representan la clasificación y las ramas representan conjunciones de características.

Características

- La decisión sigue un único camino hasta tomar la decisión.
- No es posible cuánto de adecuada es una decisión tomada.
- Necesita muchos criterios para tomar una decisión.

### ¿Ofertar producto al cliente?

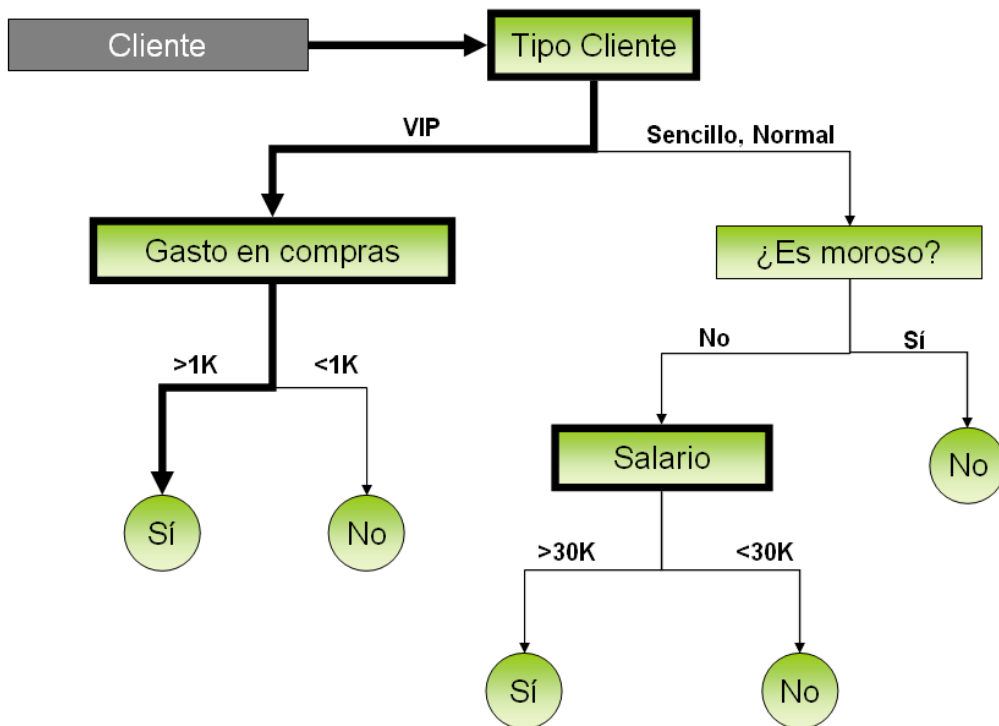


Figura 4.3: Ejemplo de *árbol de decisión* que permite decidir si se oferta un determinado producto a un determinado cliente.

### 4.7. Árboles de decisión alternativos

Un *árbol de decisión alternativo* (ADTree) es un método de clasificación. Se comportan como una generalización de los árboles de decisión anteriormente explicados.

Su funcionamiento es similar al de un árbol de decisión, pero contiene algunas modificaciones que aportan más flexibilidad y mejores características que un árbol de decisión. De tal forma que en cada momento se evalúan alguna de las características de la muestra pero no siempre en el mismo orden ya que no siempre tendrá sentido evaluar todas las características de la muestra dependiendo de sus valores. Además tras cada evaluación se le asigna un valor heurístico. La suma de todos los valores heurísticos o cualquier otra combinación lineal proporciona una medida cuantitativa de la bondad de la muestra. Se establecen intervalos (tranchas) para clasificar los valores de atributos continuos (edad, salario, saldo de cuenta corriente) De este modo se establecen intervalos como valores que permitan preguntar por si los valores corresponden a un intervalo u otro. También es significativo poder definir comportamientos o pesos no absolutos (1 ó 0), sino valores que corresponden a una ecuación lineal, ya que el poder dar un valor al extremo de un intervalo en cierto modo debe mostrar su cercanía al valor del intervalo contiguo.

Además y lo más importante para el proyecto es que a cada clase, en nuestro caso sería <tipo cliente> se le podría asignar a un conjunto de acciones que podrán ser aplicadas de acuerdo a otros criterios del cliente concreto.

### Características

- La decisión puede seguir más de un camino hasta tomar la decisión.
- Proporciona una heurística de la bondad de la muestra suministrada.
- El número de criterios para tomar una decisión depende de las características de la muestra.

Uno de los objetivos del CRMI es aumentar los beneficios de la empresa en la que se implanta. Esto es posible por la clasificación de los clientes en grupos. Y a cada uno de los grupos tomar acciones concretas para sacar el máximo beneficio. Como ejemplo clásico e ilustrativo se propone la siguiente clasificación de los clientes, en 4 grupos. En un ejemplo práctico esta aproximación grosso modo podría valer, aunque a posteriori, cada subgrupo y dependiendo del campo de negocio de la empresa podrían aparecer nuevas clasificaciones y políticas. Centrándonos en el ejemplo, el conjunto de clientes se particiona en 4 posibles grupos, aunque el objetivo ideal es que los clientes acaben solo en 2 grupos, el grupo que maximiza la fidelidad y rentabilidad o aquel que la minimiza en cuyo caso dichos clientes la empresa los catalogaría como no deseables y tratará de que dejen de formar parte de los clientes de

la empresa ya que no traerán mas que problemas y malos resultados. Entre el blanco y el negro existe una gran variedad de grises correspondiente a los otros 2 grupos aquel en que los clientes son fieles pero no rentables, por lo que habría que proponer acciones para estimular su rentabilidad y el otro grupo compuesto por clientes rentables pero no fieles cuyo objetivo sea aumentar su fidelidad mediante las acciones correspondientes.

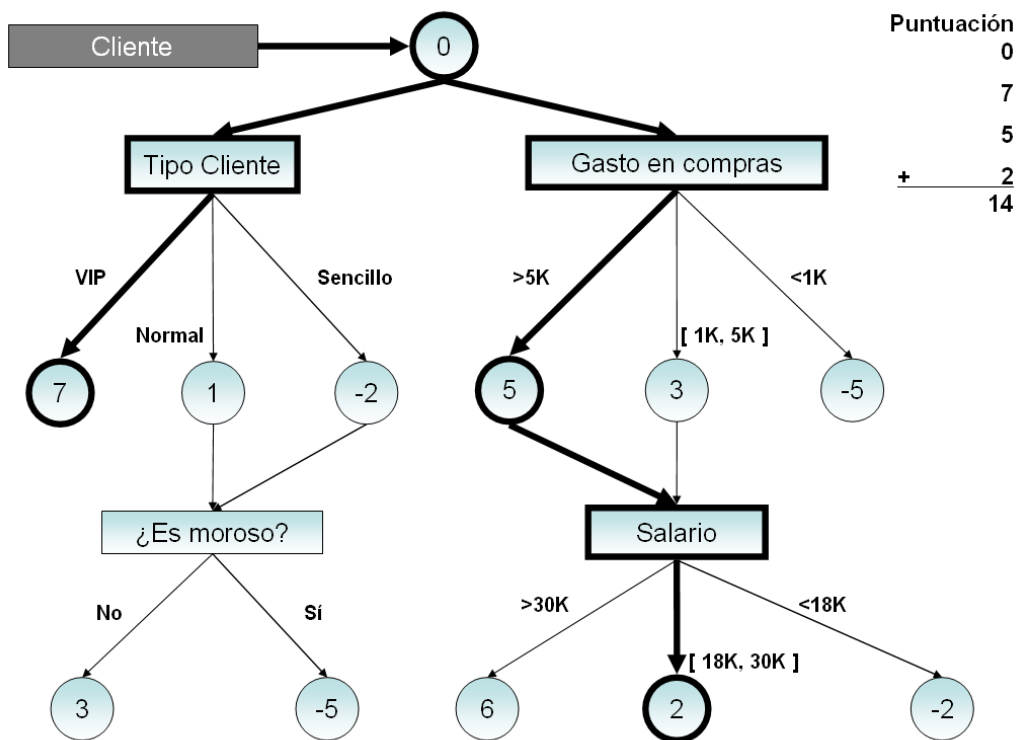


Figura 4.4: Ejemplo de un árbol de decisión alternativo que permite obtener una puntuación, evaluando así cómo de apto es el cliente para una determinada acción.

En la figura 4.5 se muestra un diagrama de los 4 grupos con unas pequeñas leyendas de las estrategias a seguir para cada clase. Los círculos de color negro representan clientes concretos. Además se incorporan flechas que indican el movimiento de los clientes a través del espacio bidimensional en función de las variables fidelidad y rentabilidad. Como se puede notar las flechas conducen a los clientes a 2 grupos como se comentó anteriormente.

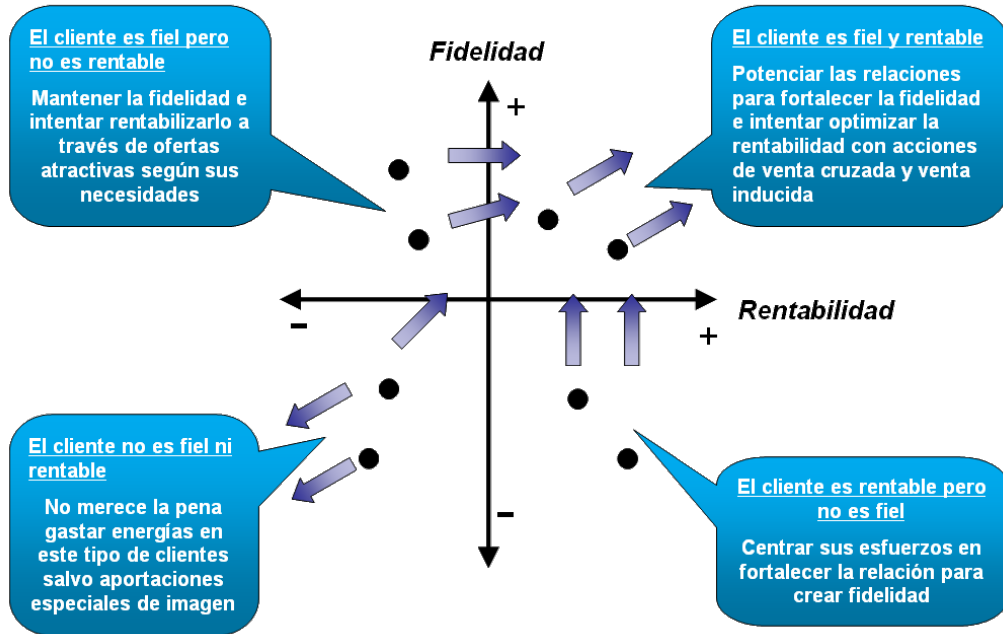


Figura 4.5: Explotación CRM

# Capítulo 5

## Diseño

En esta fase se explicarán cómo se llevarán a cabo lo definido en la fase de análisis. De esta forma se detallarán las distintas decisiones que hemos tomado para el desarrollo del sistema. Además se incluyen los casos de uso extraídos a partir de la toma de requerimientos.

### 5.1. Arquitectura

El sistema esta compuesto por 4 grandes componentes:

- **Base de datos.** almacena toda la información sobre los clientes y su interacción con el sistema. En un CRM es de vital importancia recoger la información de usuario acerca de cuando interactúa y de que modo.
- **Análisis de resultados.** Este módulo se encarga de examinar las acciones de los clientes y obtener tendencias, generar árboles de decisión, etc.
- **Capa de presentación.** Se encarga de mostrar una interfaz al usuario.
- **Lógica de negocio.** Se encarga de controlar las operaciones y funcionamiento de la aplicación.

### 5.2. Casos de uso

Son una forma de especificar los requisitos de un sistema. Tienen como misión describir la interacción entre un usuario ajeno al sistema y el sistema

con texto en lenguaje natural. Los casos de uso representan los requisitos funcionales desde el punto de vista del usuario y cada uno de ellos realiza una funcionalidad concreta (iniciar sesión, registrar incidencia, ...). Para ello será necesario identificar cuáles son los usuarios que interactúan con el sistema y cuáles son sus operaciones dentro de él.

### Objetivos de los casos de uso

Los objetivos de los casos de uso son:

- Comprender qué es lo que debe realizar el sistema
- Discutir con el usuario nuestra visión de esa funcionalidad
- Identificar conceptos del sistema, clases, atributos y operaciones
- Validar el análisis y el diseño del modelo
- Proporcionar información para las pruebas de validación y aceptación del sistema

### Elementos que intervienen

Existen dos tipos de elementos que intervienen en los casos de uso:

**Actores:** Son todas aquellas personas que tienen un rol en la interacción con el sistema. Todos los actores pueden desempeñar varios roles y un rol puede ser desempeñado por varios actores. Un usuario adopta la función de un actor cuando éste interactúa con el sistema.

**Caso de uso:** Es la interacción que se quiere simular.

## 5.2.1. UC - 1. Iniciar sesión

Caso de uso 1	Iniciar sesión
Objetivos	Identificar y validar a un cliente en el sistema
Actores	Usuario, Sistema
Entradas	Identificador de usuario, contraseña
Condiciones previas	La aplicación debe estar activa, no debe haber sesión iniciada
Salidas	Mensaje informativo de inicio correcto o fallido
Post-condición si éxito	Sesión iniciada, mostrar la pantalla de bienvenida
Post-condición si fallo	No se inicia sesión, y se muestra un mensaje de error por pantalla
Activador	Botón "Iniciar Sesión" en la pantalla de inicio de la aplicación
Secuencia	Introducir los datos, pulsar el activador
Excepciones	Los datos no son validados correctamente
Frecuencia de uso	Siempre. Previamente a cualquier acción en el sistema es preciso haber iniciado la sesión.
Asuntos pendientes	

### 5.2.2. UC - 2. Registrar un nuevo cliente

<b>Caso de uso 2</b>	<b>Registrar un nuevo cliente</b>
Objetivos	Dar de alta en la aplicación a un nuevo cliente e incorporar sus datos al sistema
Actores	Usuario, sistema
Entradas	Datos personales de usuario
Condiciones previas	Estar en la pantalla de registro de nuevos clientes
Salidas	Confirmación de registro correcto o excepciones E1, E2
Post-condición si éxito	El usuario queda registrado como cliente en el sistema
Post-condición si fallo	El usuario queda registrado como cliente en el sistema
Activador	Botón "Registrar usuario" en la pantalla de registro
Secuencia	Introducir los datos de usuario, pulsar el botón activador
Excepciones	E1: Ya hay un cliente registrado con el mismo identificador E2: Faltan datos obligatorios o son incorrectos (formato, etc)
Frecuencia de uso	Baja. Una vez por cliente.
Asuntos pendientes	

### 5.2.3. UC - 3. Crear una nueva incidencia

#### UC - 3.1 Crear una nueva avería

Caso de uso 3.1	Crear una nueva avería
Objetivos	Registrar los datos correspondientes a una avería
Actores	Usuario, sistema.
Entradas	Descripción de la avería, tipo de avería, fecha, urgencia, efectos y repercusión de la avería
Condiciones previas	Que se haya detectado una avería
Salidas	Notificación de operación correctamente finalizada, Identificador de avería, Acciones a tomar
Post-condición si éxito	Registro correcto de la avería; Notificación a los técnicos (si procede) u otras acciones para resolver el problema; Notificación de confirmación al cliente
Post-condición si fallo	Mensaje informativo explicando el motivo del fallo
Activador	Botón "Introducir Avería"
Secuencia	Rellenar los datos de la avería y pulsar el activador Si error, E1
Excepciones	E1: No haber rellenado todos los campos obligatorios
Frecuencia de uso	Alta. Cada vez que se detecta una avería
Asuntos pendientes	

## UC - 3.2 Crear una nueva reclamación

<b>Caso de uso 3.2</b>	<b>Crear una nueva reclamación</b>
Objetivos	Registrar los datos correspondientes a una reclamación
Actores	Usuario, sistema.
Entradas	Descripción de la reclamación, tipo de reclamación, fecha, causas de la reclamación
Condiciones previas	Cliente descontento
Salidas	Notificación de operación correctamente finalizada, Identificador de reclamación, Acciones a tomar
Post-condición si éxito	Registro correcto de la avería Notificación a los técnicos (si procede) u otras acciones para resolver el problema Notificación de confirmación al cliente
Post-condición si fallo	Mensaje informativo explicando el motivo de la reclamación
Activador	Botón "Introducir Reclamación"
Secuencia	Rellenar los datos de la reclamación y pulsar el activador Si error, E1
Excepciones	E1: No haber rellenado todos los campos obligatorios
Frecuencia de uso	Alta. Cada vez que se enuncia una reclamación
Asuntos pendientes	

## UC - 3.3 Crear una nueva consulta

<b>Caso de uso 3.3</b>	<b>Crear una nueva consulta</b>
Objetivos	Obtener información
Actores	Usuario, sistema.
Entradas	Descripción de la consulta, tipo de consulta, fecha
Condiciones previas	Sesión iniciada. Cliente pide información.
Salidas	Respuesta a la información que se pide, si se conoce, o un aviso de que la consulta va a ser enviada a un experto
Post-condición si éxito	Registro correcto de la consulta Si se conoce la respuesta: notificación de la respuesta a la consulta Si no: notificar a un experto para que encuentre y envíe al cliente una respuesta a la consulta
Post-condición si fallo	Mostrar el mensaje de error informando del fallo
Activador	Botón "Consultar"
Secuencia	Rellenar los datos de la reclamación y pulsar el activador Si error, E1
Excepciones	
Frecuencia de uso	Alta. Cada vez que se enuncia una reclamación
Asuntos pendientes	

### 5.2.4. UC - 4. Consultar una incidencia

#### UC - 4.1 Consultar una avería

Caso de uso 4.1	Consultar una avería
Objetivos	Obtener información de una avería existente
Actores	Usuario, Sistema
Entradas	Identificador de la avería
Condiciones previas	Que el usuario haya iniciado sesión, que la avería exista
Salidas	Se muestran los datos de la avería solicitada y su estado actual Acciones a tomar
Post-condición si éxito	Se guarda la actividad del cliente en el histórico del cliente
Post-condición si fallo	Se muestra un mensaje de error indicando el motivo del fallo
Activador	Botón "Consultar" en la pantalla de averías
Secuencia	Introducir el identificador del cliente. Pulsar el activador. Si fallo E1
Excepciones	E1 que el identificador no sea válido
Frecuencia de uso	Alta
Asuntos pendientes	

## UC - 4.2 Consultar una reclamación

<b>Caso de uso 4.2</b>	<b>Consultar una reclamación</b>
Objetivos	Obtener información de una reclamación existente
Actores	Usuario, Sistema
Entradas	Identificador de la reclamación
Condiciones previas	Que el usuario haya iniciado sesión, que la reclamación exista
Salidas	Se muestran los datos de la reclamación solicitada y su estado actual Acciones a tomar
Post-condición si éxito	Se guarda la actividad del cliente en el histórico del cliente
Post-condición si fallo	Se muestra un mensaje de error indicando el motivo del fallo
Activador	Botón "Consultar" en la pantalla de reclamaciones
Secuencia	Introducir el identificador del cliente. Pulsar el activador. Si fallo E1
Excepciones	E1 que el identificador no sea válido
Frecuencia de uso	Alta
Asuntos pendientes	

## UC - 4.3 Recuperar una consulta

<b>Caso de uso 4.3</b>	<b>Recuperar una consulta</b>
Objetivos	Obtener información de una consulta existente
Actores	Usuario, Sistema
Entradas	Identificador de la consulta
Condiciones previas	Que el usuario haya iniciado sesión, que la consulta exista
Salidas	Se muestran los datos de la consulta solicitada y su estado actual Respuesta a la consulta si se ha encontrado
Post-condición si éxito	Se guarda la actividad del cliente en el histórico del cliente
Post-condición si fallo	Se muestra un mensaje de error indicando el motivo del fallo
Activador	Botón "Recuperar" en la pantalla de consultas
Secuencia	Introducir el identificador del cliente. Pulsar el activador. Si fallo E1
Excepciones	E1 que el identificador no sea válido
Frecuencia de uso	Alta
Asuntos pendientes	




## 5.2.5. UC - 5 Proponer acciones




<b>Caso de uso 5</b>	<b>Proponer acciones</b>
Objetivos	Obtener una o varias acciones aplicables al contexto de la incidencia
Actores	Sistema
Entradas	Identificador del cliente, identificador de la incidencia
Condiciones previas	Que el usuario haya iniciado sesión, que la incidencia exista
Salidas	Se muestran las acciones adecuadas a la incidencia correspondiente
Post-condición si éxito	
Post-condición si fallo	Se muestra un mensaje de error indicando el motivo del fallo
Activador	Automático
Secuencia	
Excepciones	
Frecuencia de uso	Muy Alta. Cada vez que se crea una incidencia
Asuntos pendientes	

**5.2.6. UC - 6 Cerrar sesión**



<b>Caso de uso 6</b>	<b>Cerrar sesión</b>
Objetivos	Finalizar la sesión del cliente
Actores	Usuario, Sistema
Entradas	
Condiciones previas	Que el usuario haya iniciado sesión
Salidas	Se muestra un mensaje informativo notificando la desconexión del sistema
Post-condición si éxito	Sesión cerrada
Post-condición si fallo	No se cierra la sesión
Activador	Botón "Cerrar sesión"
Secuencia	Pulsar el activador.
Excepciones	
Frecuencia de uso	Siempre
Asuntos pendientes	

### 5.3. Diseño de la base de datos

Una vez realizado el análisis y establecido los requisitos, es necesario decidir diseñar la solución al problema. En el diseño de la base de datos se utilizó un diagrama entidad - relación. Como suele ser habitual en este tipo de diagramas cada tabla se representa junto con los atributos que la componen. Además se especifica el tipo de cada atributo. Junto a cada atributo se muestran unos pequeños iconos  ó  ó  que aportan información sobre cada atributo. Tienen el siguiente significado:

-  : Indica que el atributo forma parte de la clave primaria.
-  : Indica que el atributo forma parte de una clave ajena.
-  : Icono por defecto.

Cada tabla según el modelo *entidad - relación* puede estar relacionada con una o varias tablas del modelo. La nomenclatura utilizada para las relaciones es la siguiente:

-  : Relación uno a uno
-  : Relación uno a muchos

Además para clarificar la lectura del diagrama se agrupan las tablas en regiones.

Dependiendo de su funcionalidad cada una de las tablas pertenecen a un grupo de los siguientes:

**Datos descriptivos (Perfil del cliente):** Los datos descriptivos proporcionan información sobre el cliente. Dentro de este grupo distinguiremos 2 tipos de datos: los datos estáticos y los datos dinámicos. Los datos estáticos corresponden a aquellos datos que no cambian o lo hacen muy poco. Deben actualizarse periódicamente en plazos no menores a un año aunque existen excepciones como cuando un cliente cambia de domicilio. Los datos dinámicos de un cliente son aquellos que son más propensos a cambios. Un cliente puede adoptar varios perfiles dentro de la empresa durante su vida y existen factores que determinan su perfil. Estos factores pueden ser: deudas establecidas con la empresa, beneficios aportados, gastos, ...

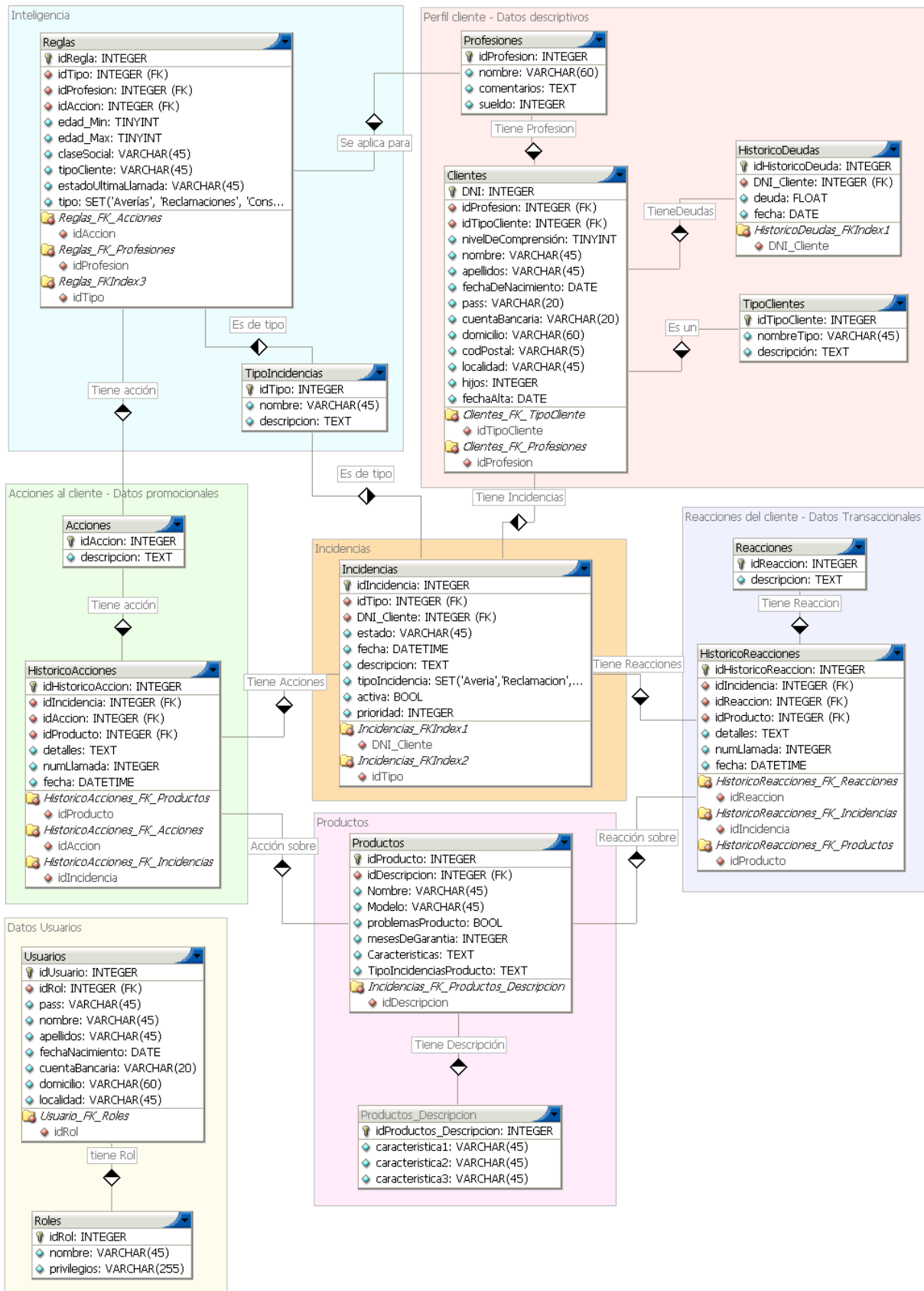


Figura 5.1: Diagrama entidad - relación

**Datos promocionales (Acciones al cliente):** Cuando se produce una incidencia la empresa proporciona en el menor tiempo posible una solución al cliente. Esa solución se denomina 'acción' y para cada incidencia y cada producto se guardarán todas las acciones propuestas en la tabla 'HistóricoAcciones'. Todas las acciones se almacenarán en la tabla 'Acciones' y en el caso de que se proporcione un nuevo tipo de acción se guardará en la tabla 'Acciones'.

**Datos transaccionales (Reacciones del cliente):** Dependiendo del grado de aceptación por parte del cliente a la acción propuesta por la empresa ante una incidencia, se generará una reacción (en algunos casos no se produce ninguna). Esta reacción se guardará en la tabla 'HistóricoReacciones.' Si el tipo de reacción no se ha producido nunca se creará una nueva entrada en la tabla 'Reacciones'

**Inteligencia:** Esta es la parte más compleja del diseño. A través de estas tablas se quiere proporcionar al sistema de inteligencia. El módulo de inteligencia está compuesto de reglas que mediante unos procesos estadísticos determinan cuáles son las acciones que deben realizarse ante una determinada incidencia. Los factores que intervienen para determinar que acción se va a tomar en los procesos estadísticos son muy diversos, pero se basa en que reacciones se han obtenido a las diversas acciones que se han planteado ante incidencias del mismo tipo.

**Datos usuarios:** Almacena los datos de los usuarios (trabajadores) y sus roles. Esta información no será procesada por las herramientas de Data Mining. Su tamaño dependerá del número de trabajadores de la empresa. Además de almacenar datos personales, también almacena el rol de cada usuario, de tal forma que cada rol tiene asociados unos privilegios. De esta forma como suele ser habitual cada trabajador, dependiendo de su rango podrá ver más o menos datos o incluso algunos datos podrán dejar de ser interesantes para determinados tipos de usuarios por lo que no se mostrarían a los usuarios con ese rol.

**Incidencias:** En esta parte del diseño aparentemente simple reside la tabla de incidencias que se interconecta a través de claves ajenas con los módulos de inteligencia, datos promocionales, datos transaccionales, y datos del cliente. Cada vez que haya una nueva incidencia se almacenará en este módulo. Además para el análisis y clasificación de los datos este módulo junto con algún otro resultara fundamental ya que permite averiguar qué acciones llevaron al cliente a reaccionar.

**Productos:** En este módulo se distinguen 2 tablas. Ambas almacenan información de productos, los productos y servicios con los que comercializa la empresa. La razón de que haya 2 tablas es para separar la información, meramente descriptiva del producto, esto es, sus características, desde el punto de vista del cliente. Dependiendo de los productos y servicios que venda la empresa dicha tabla podría ser necesario ampliarla o duplicarla de tal manera que una tabla almacena productos o servicios de un tipo y otros productos o servicios de otro tipo. En cambio la tabla Productos almacena información estadística de primera mano relevante y alguna característica necesaria para inferir comportamientos en torno a ese producto.

## 5.4. Diseño conjunto del sistema

El sistema dispondrá de varios módulos para alcanzar su objetivo:

- Módulo de inteligencia / Estadístico / Inferencia
- Base de datos
- Interfaz con el usuario

El módulo de inteligencia recopilará la información de la base de datos, en general del DW y otros almacenes de datos dinámicos para elaborar los árboles de decisión necesarios para el proceso productivo y toma de decisiones.

Este proceso se realizará “offline” dada la gran cantidad de datos almacenados. En producción el flujo de datos será el siguiente:

- El usuario necesita una acción para proponer al cliente.
- El sistema puede acceder a los datos del cliente tras su identificación
- Además el sistema podría solicitar datos referentes al estado actual del cliente.
- Con toda esa información disponible y haciendo uso de los árboles y estructuras generadas por el módulo de inteligencia es capaz de recuperar la acción o conjunto de ellas más apropiada para el cliente, asegurando que el cliente no se marcha.

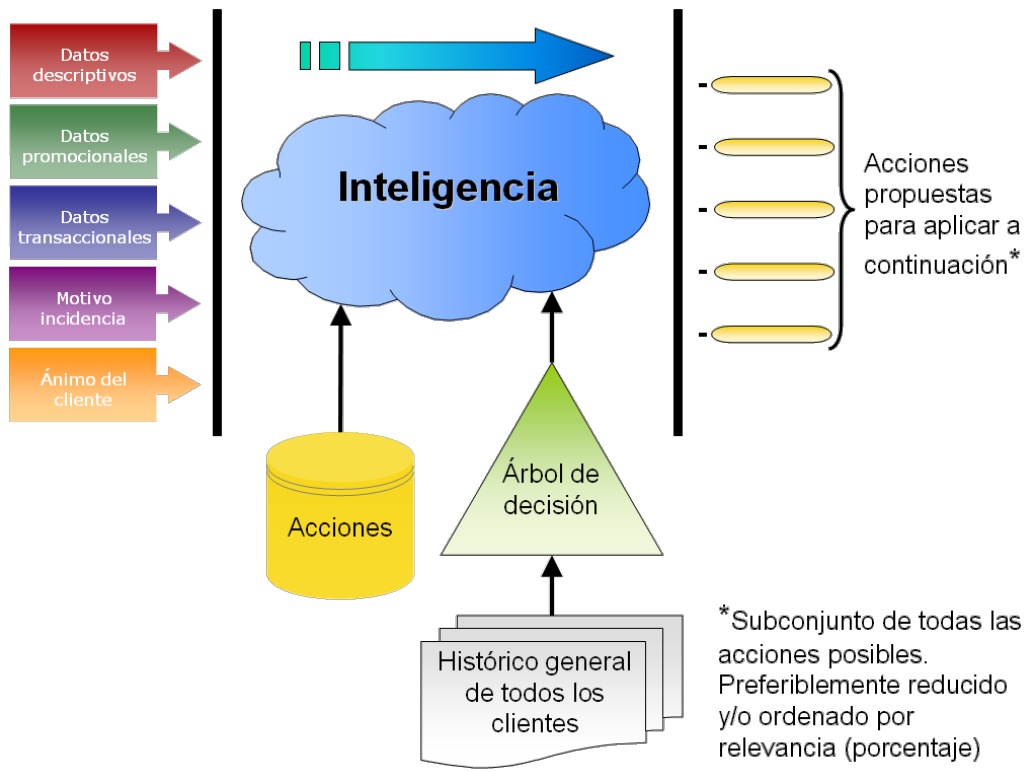


Figura 5.2: Interrelación de los módulos del sistema

# Capítulo 6

## Implementación

Debido a que el objetivo del proyecto es diseñar un sistema CRM Inteligente, la implementación se ha reducido a la realización de un prototipo.

El prototipo se ha implementado como una aplicación web usando JSP. Se ha utilizado una base de datos relacional creada en MySQL versión 5.0.51. El motor usado es MyISAM. Se ha optado por su utilización debido a la sencillez de uso, además se trata del motor por defecto con el SGBD MySQL.

El servidor web utilizado ha sido Apache Tomcat 6 (<http://tomcat.apache.org/>). Como entorno de desarrollo, optamos por Eclipse EE 3.3 (<http://www.eclipse.org/>)

La implementación para el sistema en producción se llevaría a cabo de manera similar. Probablemente habría que cambiar de motor de almacenamiento en la BBDD sustituyendo MyISAM por InnoDB. La versión de SQL utilizada en la BBDD es SQL:2003.

La aplicación, aunque no usa ningún framework para el prototipo, para la implementación del sistema Modelo-Vista-Controlador sería recomendable utilizar uno del estilo de Struts, incorporado en el proyecto Apache (<http://struts.apache.org/>)

### 6.1. Decisiones sobre el diseño de la Base de Datos

La base de datos creada para nuestro sistema tuvo algunos cambios desde el inicio de su diseño. Para ello y para su creación e implementación nos

valimos de la herramienta DB Designer 4, disponible gratuitamente desde [fabforce.net](http://fabforce.net).

Al principio creamos tres bases de datos independientes, relacionadas con las tres funcionales a las que íbamos a proveer al sistema, esto son, los siguientes tipos de incidencias por las que un cliente podía ponerse en contacto con la empresa: Averías, Reclamaciones y Consultas de Información.

En un principio, tal y como lo veíamos, cada subsistema tenía elementos diferentes. Al haber tantas funcionalidades como componentes del grupo de proyecto, nos las repartimos, de manera que cada uno hiciera una parte, de modo que al hacerlas individualmente, nos sirviera como una especie de “tormenta de ideas”, y cada uno aportara lo que pensaba que fuera conveniente. Por tanto, aunque los conceptos eran mayoritariamente parecidos, había incompatibilidades entre los tres diseños, empezando por el nombre de atributos que representaban lo mismo pero tenían nombres diferentes.

Para juntar las tres partes creadas, nos fijamos primero en las similitudes de los tres modelos. En todos los casos habíamos creado una entidad llamada cliente, que, intuitivamente, guardaba todos los datos personales necesarios. Sin embargo, algunas veces parecía necesario guardar la información de cómo había ido reaccionando, para lo cual en un diseño se incluyó como una tabla adicional relacionada con la tabla clientes. Al juntar los tres diseños, esta tabla se suprimió, integrándose esta información dentro de otra tabla que contenía el histórico.

Otra de las similitudes que disponían los tres diseños era un sistema de Acciones y Reacciones. La filosofía de la aplicación, de guardar todos los datos relacionados con lo que se le hace al cliente, obligaba a guardar un registro de Históricos de Acciones y Reacciones, así como de otra tabla, diseñada para ser más leída que modificada, en la que se almacenaban los distintos tipos de acciones que se pueden aplicar a un cliente. Esta tabla de Acciones guarda un identificador de acciones, junto con una descripción en un campo de texto.

En un principio, por simplicidad a la hora de juntar el trabajo que habíamos hecho por separado, pensamos que lo mejor sería crear tres históricos de Acciones y de Reacciones distintos para cada funcionalidad. Esto aseguraba que no se pudieran aplicar erróneamente. Sin embargo, la similitud entre las funcionalidades de Averías y de Reclamaciones (ya que muchas veces el cliente detecta un fallo en el producto o una deficiencia en el servicio, pero no sabe distinguirlo de un error de configuración) hizo pensar que, posiblemente, el conjunto de las Acciones de ambas funcionalidades pudieran no ser disjuntos. Como uno de los objetivos de una base de datos es, en la medida de lo posible, evitar información redundante, para simplificar las modificaciones,

decidimos crear históricos conjuntos para las Averías y Reclamaciones.

Con las Consultas tuvimos serias dudas para integrarlas con las mismas tablas, pero al final, resultaba más sencillo de mantener. Para ello, pusimos un campo en las tablas que identificaba a cada registro con el tipo al que pertenecía

Con estas y otras modificaciones, terminamos agrupando todo en un único modelo en el cual encontramos, vertebrando, las tablas Incidencias, Reclamaciones y Consultas. Estas tres tablas, que no se relacionan entre ellas (son subsistemas diferentes) en definitiva, terminan comportándose de manera similar entre sí. Se relacionan con la tabla Clientes, y con las tablas Acciones y Reacciones, estas últimas, a su vez, referencian a un producto.

El resultado final es el mostrado en el apartado [diseño](#) de la memoria.

## 6.2. Capturas de pantalla del prototipo

En las siguientes capturas de pantalla se podrá ver el aspecto del prototipo de la aplicación. Se trata de una aplicación web lo que facilita su implantación así como la incorporación de nuevos cambios sin realizar instalaciones expresas en cada uno de los terminales. Se muestran una serie de pantallas con algunas de las funcionalidades que incorporaría la aplicación funcional. Quedan pendientes incluir nuevas funcionalidades a la interfaz. Como ejemplo se han incorporado datos inventados sobre clientes así como mensajes de advertencia, información o alerta al usuario sobre el perfil del cliente. El usuario deberá utilizar la información de los mensajes para tratar al cliente así como seguir el protocolo de la empresa en sintonía con las políticas de negocio incorporadas al *CRMI*. El sistema proporcionará acciones que se aplicarán a cada cliente en función de lo que se explicó a lo largo del presente documento.

Es posible que en alguna captura de pantalla la cantidad de información contenida sea excesiva, en este caso, el sistema ocultará la información no relevante. Es necesario que los usuarios estén familiarizados con el uso de la aplicación. Alguna de la información que se muestra puede no ser necesaria para el usuario encargado de responder las llamadas de los clientes, pero por el contrario si puede resultar de gran utilidad a otros usuarios interesados en la optimización y funcionamiento del sistema. Un sistema de estas características necesita de un proceso de funcionamiento previo para adaptarse a los clientes y satisfacer sus necesidades asegurando un porcentaje mínimo de bajas de los clientes en la empresa.

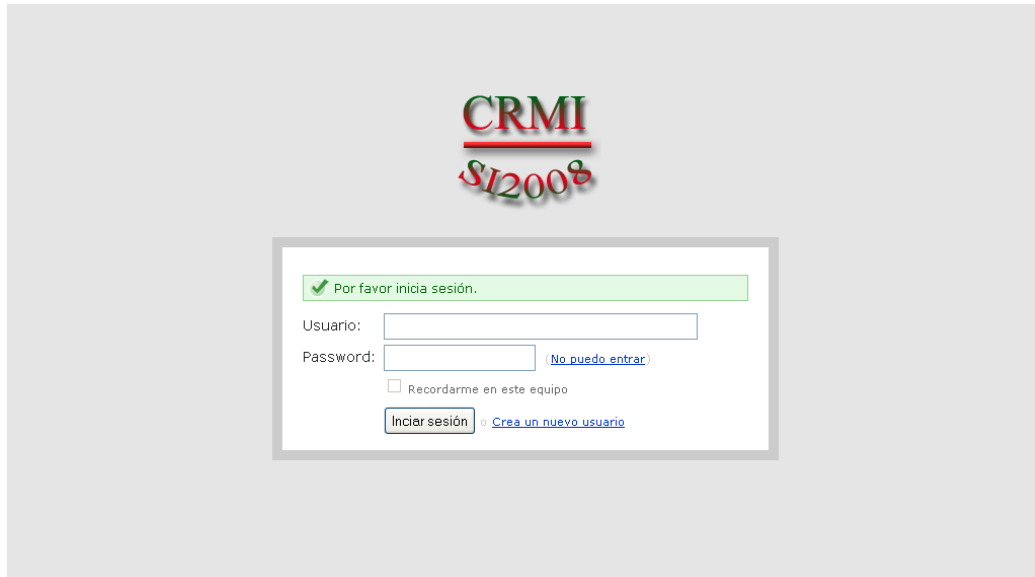


Figura 6.1: Inicio de sesión en el sistema CRMI



Figura 6.2: Formulario de alta de un nuevo cliente



Figura 6.3: Modo de recuperación de la contraseña de usuario olvidada

## CRMI

Porque gestionar es vital.

Bienvenido **Juan Perez**

[Cerrar sesión](#)

Resumen
Averías
Reclamaciones
Consultas
Opciones
Estadísticas

Productos interesantes para el cliente:

- [Producto #1](#)
- [Producto #2](#)
- [Producto #3](#)
- [Producto #4](#)
- [Producto #5](#)
- [Producto #6](#)

[Buscar](#)

El cliente presenta un adecuado nivel de fidelización pero un escaso nivel de beneficios. El sistema dispone de suficiente información para la toma de decisiones. [Informe detallado de la previsión del cliente](#)

### Resumen de movimientos

Actualizado 12 junio 2008

- 12 Averías. Satisfacción media (6,4)
- 7 Reclamaciones. Satisfacción media (7,1)
- 2 Consultas. Satisfacción media (6,9)

En los últimos 3 meses ha abierto un gran número de incidencias. Debe primar reducir el número de incidencias abiertas y aumentar la valoración de cada una de ellas.

**Plan de fidelización**

Lea el protocolo corporativo de actuación para mantener la fidelidad del cliente y como sacar el máximo rendimiento.

Incluye planes de actuación es distintos escenarios y para perfiles de clientes distintos. Más información consulte con el supervisor de su sección.

**Plan de mejora del beneficio**

Lea el protocolo corporativo de actuación para mejorar el rendimiento del cliente. En un cliente se priman fundamentalmente 2 factores: Fidelidad y Beneficios

Incluye planes de actuación es distintos escenarios y para perfiles de clientes distintos. Más información consulte con el supervisor de su sección.

**Plan de reducción de incidencias**

Lea el protocolo corporativo de actuación para reducir el número de incidencias y en general aumentar su satisfacción.

Incluye planes de actuación es distintos escenarios y para perfiles de clientes distintos. Más información consulte con el supervisor de su sección.

**Información bancaria confirmada**

Se ha verificado exitosamente los datos bancarios disponibles en el perfil. Además se ha comprobado la solvencia de dichas fuentes. Vea en el [informe](#) los detalles de la comprobación de solvencia.

**Perfil completo a 90%**

El sistema dispone de datos suficientes para la toma de decisiones. Es importante que dichos datos se mantengan actualizados. La siguiente actualización de los mismos se realizará dentro de 178 días. El [perfil](#) se encuentra disponible para su consulta y modificación..

**Información fiable al 90%**

Los datos presentados tienen una fiabilidad igual o superior al 90%. Conforme la antigüedad del cliente aumente, también a través de campañas y otros medios se aumentará la información relevante para la toma de decisiones.

[RSS Feed](#) · [Contacto](#) · [Accesibilidad](#) · [Productos](#) · [Disclaimer](#) · [CSS](#) and [XHTML](#)

© 2008 CRMI, Diseño: CSS technologies - [CSS inc](#)

Figura 6.4: Página inicial al entrar al sistema

**CRMI**

Porque gestionar es vital.

Bienvenido **Juan Perez**

[Cerrar sesión](#)

**Resumen** | **Averías** | Reclamaciones | Consultas | Opciones | Estadísticas

Productos interesantes para el cliente:

- [Producto #1](#)
- [Producto #2](#)
- [Producto #3](#)
- [Producto #4](#)
- [Producto #5](#)
- [Producto #6](#)

**Buscar**

**Averías activas**

Última avería: jueves 29 de mayo de 2008 [Mostrar todas](#)

idTipo	Estado	Descripción	Fecha	
0	cerrada	No tiene línea en su terminal móvil. Proviene de un alta reciente. Necesita sc	7/04/08	
0	cerrada	Fallo en el terminal. Se apaga constantemente sin previo aviso. Posible defe	10/04/08	
0	en proceso	Puede recibir llamadas pero no puede emitir las	16/04/08	
0	en proceso	Problemas con el servicio de video-llamada. Comprobada configuración del b	21/04/08	
0	en proceso	Caida y rotura del terminal. El terminal puede no estar en garantía.	29/05/08	

Se muestran las averías en proceso además de las existentes en los 3 últimos meses

**! Numerosas averías pendientes**

El cliente tiene numerosas averías pendientes de resolverse. El sistema tendrá en cuenta este escenario para conseguir resolverlas en la mayor brevedad intentando causar las menores molestias al cliente. Se dispone de un [informe](#) resumen con los plazos estimados de resolución.

**✖ Cliente moroso**

El cliente acumula deudas con la empresa por un valor total de 385 €

Para consultar las el detalle de las deudas, vea el siguiente [informe](#). El sistema tomará decisiones antendiendo al nuevo escenario.

[RSS Feed](#) | [Contacto](#) | [Accesibilidad](#) | [Productos](#) | [Disclaimer](#) | [CSS and XHTML](#)

© 2008 CRMI, Diseño: CSS technologies - [CSS inc](#)

Figura 6.5: Averías activas

**CRMI**

Porque gestionar es vital.

Bienvenido **Juan Perez**

[Cerrar sesión](#)

**Resumen** | **Averías** | **Reclamaciones** | Consultas | Opciones | Estadísticas

Productos interesantes para el cliente:

- [Producto #1](#)
- [Producto #2](#)
- [Producto #3](#)
- [Producto #4](#)
- [Producto #5](#)
- [Producto #6](#)

**Buscar**

**Reclamaciones activas**

Última reclamación: lunes 21 de abril de 2008 [Mostrar todas](#)

idTipo	Estado	Descripción	Fecha	
0	cerrada	Precisa devolución por el cobro de los servicios prestados. El usuario no pud	21/04/08	
0	cerrada	Reclama una compensación económica por el malestar producido durante es	21/04/08	

Se muestran las reclamaciones en proceso además de las existentes en los 3 últimos meses

**✖ Cliente moroso**

El cliente acumula deudas con la empresa por un valor total de 385 €

Para consultar las el detalle de las deudas, vea el siguiente [informe](#). El sistema tomará decisiones antendiendo al nuevo escenario.

**✓ Cliente satisfecho**

El cliente está satisfecho con el resultado después de las últimas reclamaciones. Se dispone de un [informe](#) de tendencias de satisfacción con la reclamaciones.

[RSS Feed](#) | [Contacto](#) | [Accesibilidad](#) | [Productos](#) | [Disclaimer](#) | [CSS and XHTML](#)

© 2008 CRMI, Diseño: CSS technologies - [CSS inc](#)

Figura 6.6: Reclamaciones activas

**CRM**  
Porque gestionar es vital.

Bienvenido Juan Perez [Cerrar sesión](#)

Resumen **Averías** Reclamaciones **Consultas** Opciones Estadísticas

Productos interesantes para el cliente:

- Producto #1
- Producto #2
- Producto #3
- Producto #4
- Producto #5
- Producto #6

### Consultas activas

Última consulta: viernes 20 de junio de 2008 [Mostrar todas](#)

idTipo	Estado	Descripción	Fecha
0	resuelta	¿Cómo se utiliza el servicio de video-llamada?	24/04/08
0	pendiente	¿Cómo se utiliza el modo nocturno de la camara de fotos?	2/05/08
0	resuelta	¿Cuanto cuesta el servicio de video conferencia?	7/05/08
0	resuelta	¿Puedo utilizar mi terminal en EULU?	20/06/08

Se muestran las consultas en proceso además de las existentes en los 3 últimos meses

**?** **Se desconoce el nivel de compresión**  
No se conoce con precisión el nivel de compresión del cliente. El sistema ha generado un apetición al departamento correspondiente para poder realizar la encuesta correspondiente. Puede ver el detalle de la [detalle](#) para obtener más información.

**i** **El cliente ha adquirido un nuevo artículo recientemente**  
El cliente ha adquirido un nuevo teléfono móvil de altas prestaciones recientemente. Es probable que tenga alguna consulta sobre este dispositivo, de cualquier modo se mostrará interés por averiguar la satisfacción con el producto. Consulte el detalle de la [factura](#) o las [tendencias](#) asociadas al producto.

[RSS Feed](#) [Contacto](#) [Accesibilidad](#) [Productos](#) [Disclaimer](#) [CSS and XHTML](#) © 2008 CRM, Diseño: CSS technologies - [CSS Inc](#)

Figura 6.7: Consultas activas

**CRM**  
Porque gestionar es vital.

Bienvenido Juan Perez [Cerrar sesión](#)

Resumen **Averías** Reclamaciones **Consultas** Opciones Estadísticas

Productos interesantes para el cliente:

- Producto #1
- Producto #2
- Producto #3
- Producto #4
- Producto #5
- Producto #6

### Editar incidencia

# Incidencia: 10

Tipo:

Estado:

Fecha:

Activa:

Descripción:

No edite la incidencia a no ser que haya algún error en su tipificación. Es preferible abrir una incidencia nueva con la nueva información.

**!** **Numerosas averías pendientes**  
El cliente tiene numerosas averías pendientes de resolverse. El sistema tendrá en cuenta este escenario para conseguir resolverlas en la mayor brevedad intentando causar las menores molestias al cliente. Se dispone de un [informe](#) resumen con los plazo estimados de resolución.

**x** **Cliente moroso**  
El cliente acumula deudas con la empresa por un valor total de 385 €  
Para consultar las el detalle de las deudas, vea el siguiente [informe](#). El sistema tomará decisiones antendiendo al nuevo escenario.

[RSS Feed](#) [Contacto](#) [Accesibilidad](#) [Productos](#) [Disclaimer](#) [CSS and XHTML](#) © 2008 CRM, Diseño: CSS technologies - [CSS Inc](#)

Figura 6.8: Formulario de edición de una incidencia

# Capítulo 7

## Ejemplos de funcionamiento

En esta sección trataremos unos pequeños ejemplos para ilustrar el funcionamiento del sistema. El sistema utiliza árboles de decisión para averiguar los atributos discriminatorios y obtener una acción adecuada al tipo de cliente. Con la suficiente información en la BBDD y con procedimientos estadísticos podemos asegurar que el cliente no abandona la empresa. De esta forma veremos como se comporta el módulo de inteligencia.

### 7.1. Ejemplo 1 (Averías)

Un cliente tuvo un accidente la semana pasada con su moto. Abrió una incidencia (avería) para solventar su problema. En la actualidad la avería sigue abierta y el cliente llama al call center exigiendo una solución a su situación. El cliente es atendido por el operador (usuario del sistema). Tras proporcionar su DNI para identificarse en el sistema e identificar la avería. El sistema obtiene los siguientes datos del cliente:

- DNI: 42695439-F
- Profesión: Director bancario
- Tipo de cliente: VIP
- Nombre: Juan
- Apellidos: Martínez Ponte
- Fecha de nacimiento: 25/04/1958

- Cuenta bancaria: 1568-4762-62-8468631379
- Domicilio: Av. Pez Volador, 34
- Código postal: 28032
- Localidad: Madrid
- Fecha de alta: 03/05/1999
- Salario: 50.000 €
- Moroso: No
- Nº hijos: 2

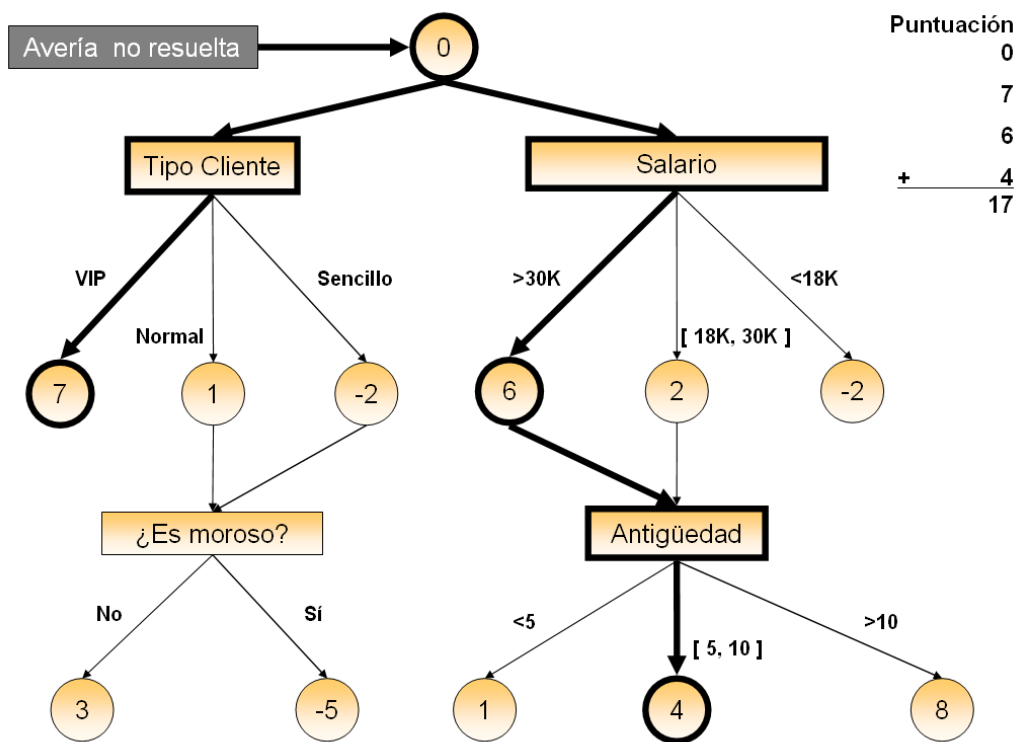


Figura 7.1: Ejemplo de un árbol de decisión alternativo para un hipotético caso de averías.

El módulo de inteligencia utiliza el conocimiento extraído de anteriores situaciones similares para crear el árbol de decisión. El propio módulo de

inteligencia es capaz de inferir qué atributos son relevantes para tomar la decisión y elegir la acción más adecuada al perfil de ese cliente.

El módulo de inteligencia genera un árbol que permite elegir a través de una heurística la acción a aplicar más apropiada para el perfil del cliente.

Las posibles acciones para el perfil de ese cliente en esta situación son:

- **Acción 1:** Proporcionar una moto de características similares a la suya en reparación sin coste adicional para el cliente. [15;25)
- **Acción 2:** Proporcionar una moto de características inferiores a la suya en reparación sin coste adicional para el cliente.[10;15)
- **Acción 3:** Proporcionar una moto de características similares a la suya en reparación con cargo adicional para el cliente.[5;10)
- **Acción 4:** Comprometerse con el cliente a tener su moto reparada en los próximos días.[0;5)
- **Acción 5:** Mantener la moto a la espera de la reparación hasta que se mejore la situación empresa - cliente.[-10; 0)

El módulo de inteligencia obtiene una heurística utilizando el árbol de decisión generado para el perfil del cliente. La heurística resultante es 17. Con esta puntuación le corresponde la acción n<sup>o</sup> 1

## 7.2. Ejemplo 2 (Reclamaciones)

Un cliente ha recibido una factura que considera excesiva en cuanto a precio. Abrió una incidencia (reclamación) para estudiar su problema. En la actualidad la reclamación sigue abierta y el cliente llama al call center exigiendo una solución a su situación. El cliente es atendido por el operador (usuario del sistema). Tras proporcionar su DNI para identificarse en el sistema e identificar la reclamación. El sistema obtiene los siguientes datos del cliente:

- DNI: 52657882-F
- Profesión: Estudiante
- Tipo de cliente: Normal
- Nombre: Pedro
- Apellidos: Jiménez Lacruz
- Fecha de nacimiento: 25/04/1985
- Cuenta bancaria: 1521-2578-98-1548756988
- Domicilio: C/ Olimpo 16, 2º
- Código postal: 28039
- Localidad: Madrid
- Salario: n.d.
- Moroso: Sí; - 2300 €
- Nº hijos: 0

El módulo de inteligencia utiliza el conocimiento extraído de anteriores situaciones similares para crear el árbol de decisión. El propio módulo de inteligencia es capaz de inferir qué atributos son relevantes para tomar la decisión y elegir la acción más adecuada al perfil de ese cliente.

El módulo de inteligencia genera un árbol que permite elegir a través de una heurística la acción a aplicar más apropiada para el perfil del cliente.

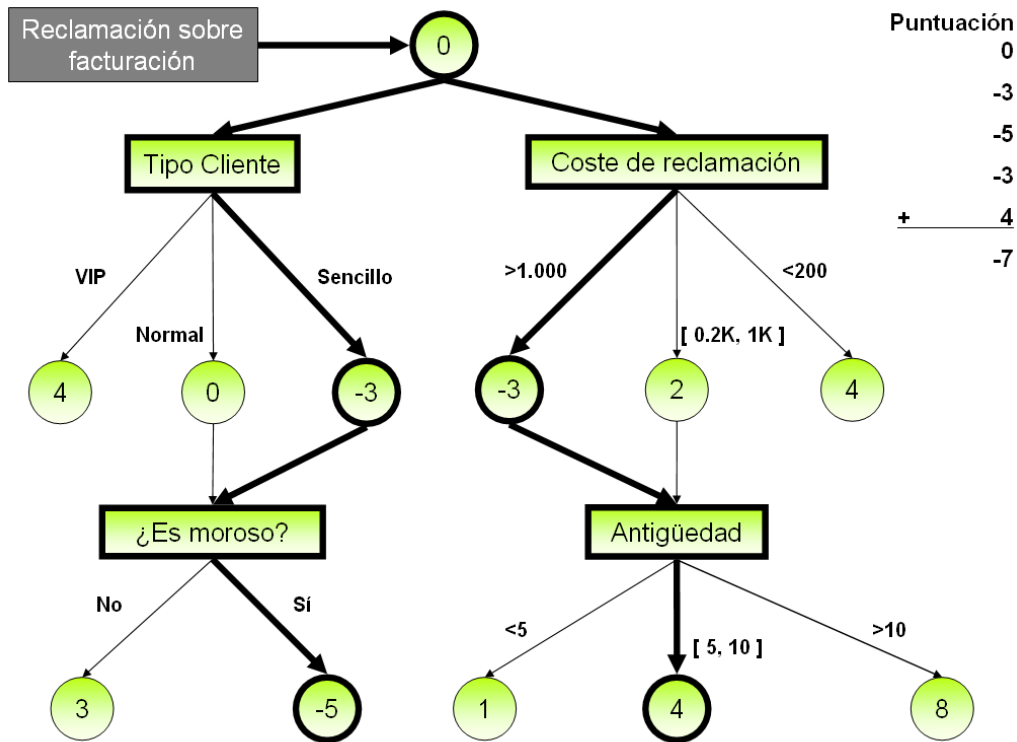


Figura 7.2: Ejemplo de un *árbol de decisión alternativo* para un hipotético caso de reclamaciones.

Las posibles acciones para el perfil de ese cliente en esta situación son:

- **Acción 1:** Consultar al departamento de facturación la posibilidad de resolver la incidencia. Respuesta inmediata. [15;20)
- **Acción 2:** Comprometerse con el cliente a resolver la reclamación en los próximos 3 días.[7;15)
- **Acción 3:** Comprometerse con el cliente a resolver la reclamación en los próximos 10 días.[0;7)
- **Acción 4:** Mantener la reclamación sin solventar hasta que se mejore la situación empresa - cliente.[-10; 0)

El módulo de inteligencia obtiene una heurística utilizando el árbol de decisión generado para el perfil del cliente. La heurística resultante es -7. Con esta puntuación le corresponde la acción nº4

### 7.3. Ejemplo 3 (Consultas)

Un cliente llama por primera vez preguntando acerca de una funcionalidad concreta sobre un GPS. Abre una incidencia (consulta) para registrar su duda. El cliente es atendido por el operador (usuario del sistema). Tras proporcionar su DNI para identificarse en el sistema e identificar la consulta en el HelpDesk. El sistema obtiene los siguientes datos del cliente:

- DNI: 26589741-F
- Profesión: Carpintero
- Tipo de cliente: Sencillo
- Nombre: Luis
- Apellidos: Ríos López
- Fecha de nacimiento: 25/04/1949
- Cuenta bancaria: 5159-5414-12-2153698547
- Domicilio: C/ La Teja 10, 1<sup>ª</sup>A
- Código postal: 26300
- Localidad: Logroño
- Salario: 20.000 €
- Moroso: No
- N<sup>º</sup> hijos: 3

El módulo de inteligencia utiliza el conocimiento extraído de anteriores situaciones similares para crear el árbol de decisión. El propio módulo de inteligencia es capaz de inferir qué atributos son relevantes para tomar la decisión y elegir la acción más adecuada al perfil de ese cliente.

El módulo de inteligencia genera un árbol que permite elegir a través de una heurística la acción a aplicar más apropiada para el perfil del cliente.

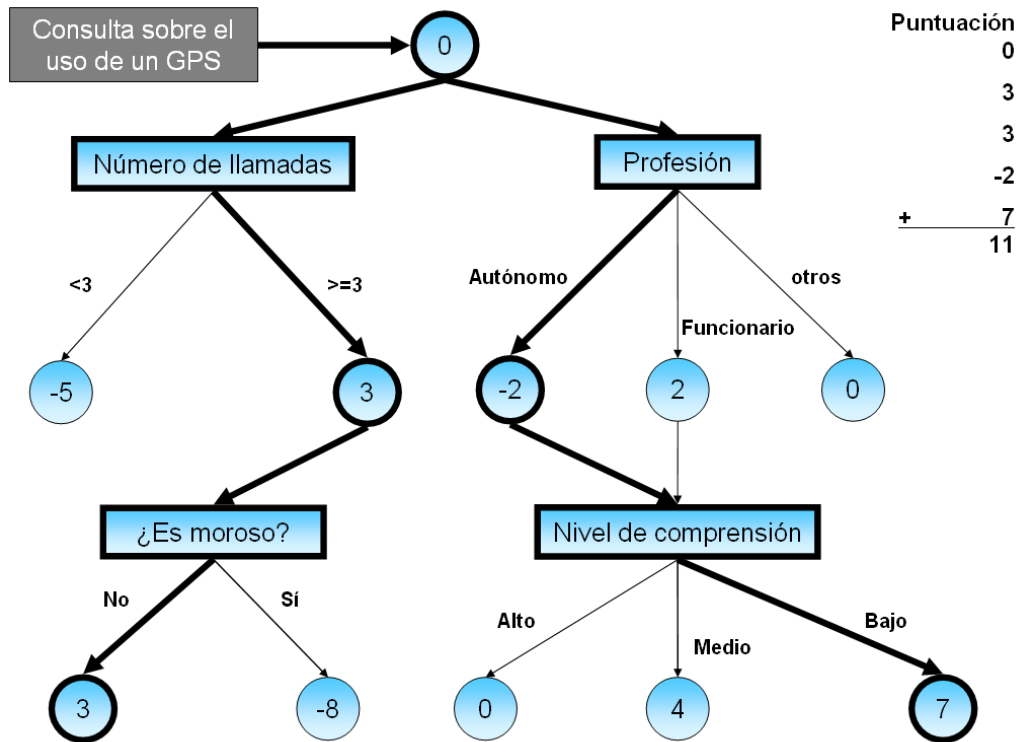


Figura 7.3: Ejemplo de un *árbol de decisión alternativo* para un hipotético caso de consultas.

Las posibles acciones para el perfil de ese cliente en esta situación son:

- **Acción 1:** Proporcionar ayuda detallada en el momento de la llamada. [7;15)
- **Acción 2:** Proporcionar ayuda poco concisa en el momento de la llamada. [2;7)
- **Acción 3:** Proporcionar la ayuda en los próximos días. [-7;2)
- **Acción 4:** Mantener la consulta sin resolver hasta que se mejore la situación empresa - cliente. [-15; -7)

El módulo de inteligencia obtiene una heurística utilizando el árbol de decisión generado para el perfil del cliente. La heurística resultante es 11. Con esta puntuación le corresponde la acción nº 1

# Capítulo 8

## Conclusiones

La intención de este proyecto es diseñar un CRM inteligente para aumentar los beneficios de la empresa, tanto por volumen de ventas como por fidelización de clientes. Para obtener buenos resultados es importante que el sistema este dotado con la mayor inteligencia posible.

A medida que el sistema iba creciendo nos dimos cuenta de la importancia que tiene disponer de información en abundancia y bien relacionada. Cuanta más información dispone, más adecuada es la acción propuesta. Además de esto, es importante seleccionar los perfiles de clientes adecuados para encontrar la mejor acción posible.

Al diseñar la inteligencia del sistema es necesario incluir los métodos estadísticos, que, a partir de un análisis estadístico de los datos recogidos anteriormente, mejor convengan a cada situación. También se sintonizará el sistema para hallar los mejores umbrales que determinen las distintas acciones. Todo ello, tras un proceso de comparación del cliente en curso con los resultados más similares, intenta proponer la opción más favorable, tanto para el cliente (se pretende que se quede satisfecho), como para la empresa (que le interesa fidelizarlo). Por ello, en el fondo, nuestro sistema ha sido un Sistema Basado en Casos (CBR).

Lo ideal es conseguir una solución que proporcione el máximo beneficio a la empresa pero sin llegar a tener problemas con el cliente. No interesa que la incidencia se prolongue ó que se abran nuevas reclamaciones.

En cuanto al desarrollo del proyecto, los principales problemas que hemos tenido que afrontar surgieron de no dar la importancia necesaria a la fase de análisis. En principio, la especificación de requisitos no fue detallada y esto nos produjo cierta ambigüedad durante la fase de diseño y codificación. Por

ello, tuvimos que retocar el diseño de la base de datos en varias ocasiones para que contemplara todas las funcionalidades deseadas. Por otra parte, no tuvimos en cuenta adaptar el diseño a futuras extensiones y cuando quisimos agregar una nueva funcionalidad fue necesario reorganizarlo en exceso.

Otro de los errores que cometimos fue empezar a codificar el sistema cuando todavía no habíamos terminado con el diseño. Esto hizo que gran parte de la implementación tuviera que ser desechada. Más tarde nos dimos cuenta de nuestro error y no comenzamos la codificación hasta haber terminado con el diseño.

Este proyecto incorpora mucha carga de investigación derivada de la parte de análisis, a la que podíamos haber dedicado más esfuerzo en lugar de haberlo invertido en otras partes del proyecto.

Hemos aprendido a trabajar en un proyecto de grupo con distintas fases, afrontando los problemas relacionados con la captura de requisitos, como si el profesor fuera el cliente, con los problemas que conllevaba el ponerse de acuerdo y entender lo que se pedía.

Se consiguen mejorar el ratio de número de clientes que se marchan, así como la rentabilidad de los clientes que permanecen, que compran más, y más productos.

# Capítulo 9

## Extensiones

Alguna de las extensiones propuestas para el proyecto son las siguientes.

**Explicación pertinente:** el sistema explica los motivos por los cuales propone una acción a un determinado cliente. De esta forma se permite averiguar qué atributos del cliente son los que llevan al sistema a proponer dicha acción

**Productos Tabú:** Con el objetivo de mejorar los ingresos para la empresa el sistema dispondrá de listas negras o listas tabú que contendrán las acciones, en particular productos/servicios, que no se aplicarán a los clientes. Es posible que en algunos casos dicha acción genere beneficios, pero en general la aplicación de dicha acción traerá resultados negativos.

**CBR JColibrí** Implementación de la inteligencia usando los módulos del framework de razonamiento CBR jColibrí del grupo de investigación gaia (<http://gaia.fdi.ucm.es/projects/jcolibri/>).

# Bibliografía

- *R. Pressman; Ingeniería del Software. Un enfoque práctico; 6<sup>ª</sup> edición, Ed. McGraw-Hill, 2005;*
- *SILBERSCHATZ, A., KORTH, H.F., SUDARSHAN, S. ; Fundamentos de bases de datos; 5<sup>ª</sup> edición, McGraw-Hill, 2006;*
- *Gonzalo Pajares y Matilde Santos; Inteligencia Artificial e ingeniería del Conocimiento; RA-MA, 2005 (Primera Edición en español)*
- *Russell, S. y Norvig, P. ; Artificial Intelligence: A Modern Approach ; Prentice Hall, New Jersey, Edición del 2004 en español*
- *Página de wikipedia: [www.wikipedia.org](http://www.wikipedia.org)*

# Apéndice A

## Glosario

**Acción:** Lo que se le realiza al cliente.

**Acción- Reacción, modelo de,:** Es un modelo de actuación que se basa en que cada individuo reacciona en función de las acciones aplicadas sobre él. De esta manera se pueden prever comportamientos muy interesantes.

**Árbol de decisión:** es un modelo de predicción utilizado en el ámbito de la inteligencia artificial. Un primer algoritmo para construir estos árboles es el algoritmo ID3.

**Árbol de decisión alternativo:** es una variante de los *árboles de decisión* que permite evaluar cuantitativamente el individuo.

**Avería:** Daño que impide el funcionamiento correcto de un producto adquirido por el cliente.

**Clientes:** Persona física o jurídica que mantiene relación con la empresa por haber realizado compras o percibido servicios en la misma.

**Contactos:** Persona física o jurídica que sin haber aún realizado ninguna compra o recibido servicios, se prevee que en el futuro realice alguna compra o reciba un servicio.

**Consulta:** Dictamen que se pide a la empresa acerca de un servicio ofrecido.

**Data Mining (DM):** Combinación de tecnologías y técnicas que permiten la extracción de información de grandes bases de datos, para convertirla en conocimiento que será utilizado para tomar decisiones empresariales.

**Data Warehouse (DW):** Es un almacén de datos, que actúa como repositorio para conectar las distintas fuentes de información sobre los clientes.

**Incidencia:** Evento que afectando al cliente se corresponde con uno de estos 3 tipos, Avería, Reclamación o Consulta.

**jCOLIBRI:** Framework desarrollado por GAIA que constituye un marco para el desarrollo de sistemas CBR.

**Reacción:** Lo que el cliente realiza sobre la empresa y su uso con los productos y servicios proporcionados por la misma.

**Reclamación:** Exigencia del cliente a que le proporcionen un servicio al que tiene derecho

**ROI:** Es una medida de rendimiento usada para evaluar la eficiencia de una inversión o para comparar la eficiencia de diferentes inversiones. Se calcula:

$$ROI_{Arit} = \frac{V_f - V_i}{V_i} \text{ donde:}$$

- $V_i$  es el valor inicial de la inversión
- $V_f$  es el valor final de la inversión

Características del resultado:

- $ROI_{Arit} = +1,00 = 100\%$ : cuando el valor final es el doble que el inicial
- $ROI_{Arit} > 0$ : cuando la inversión tiene beneficios. (Invertir)
- $ROI_{Arit} < 0$ : cuando la inversión traerá pérdidas. (No invertir)
- $ROI_{Arit} = -1,00 = -100\%$ : cuando lo invertido no podrá ser recuperado

**Usuario:** Persona física que interactuará de intermediario entre el cliente o contacto, usualmente por vía telefónica y el sistema a través de su interfaz.

# Apéndice B

## Listado de las actas de las reuniones

Durante el desarrollo del proyecto tal como se explico en el apartado de metodología de desarrollo hicimos reuniones periódicas. El seguimiento de dichas reuniones se encuentra en este listado de actas.

### B.1. Acta 1 (24/10/2007)

El proyecto trata de documentar diseñar e implementar un Sistema de Atención al Cliente o CRM que tiene como objetivo guardar y gestionar toda la información posible acerca de los productos, los clientes, ventas, etc así como las relaciones que puedan existir entre ellos. Existen elementos fundamentales tales como:

- Productos
- Clientes
- ...

Nuestro sistema tendrá información acerca de los productos concretos tal como: nombre, garantía, modelo, además será necesario identificar cada producto de forma unívoca a través del S/N (Serial Number).

Como es natural, pueden existir varios modelos para un mismo producto. Por ello será necesario poder recabar información de cada modelo, como puede ser la versión o el año de creación.

Para los clientes el sistema ha de manejar sus datos personales: nombre, apellidos, fecha de nacimiento, . . . así como atributo adicional para identificarlo y distinguirlo del resto de clientes.

El sistema recogerá información acerca de las acciones de los clientes.

En la operación compra se relaciona un producto con el cliente que lo adquirió. Almacenando algún dato identificativo del cliente así como un identificador para compra, siendo así posible recuperar posteriormente esta información para cualquier otro fin. Además cada compra tendrá el importe y la fecha en la que fue hecha.

El aprendizaje del sistema es importante para determinar de qué forma suele actuar cada cliente. Esto nos permite averiguar cuáles son sus preferencias y en general patrones de comportamiento. Los datos se extraerán bien de la forma de actuar de cada cliente o mediante cuestionarios. A partir de estos datos podemos crear perfiles de clientes que actúen de la misma forma.

El sistema registrará cada incidencia que el cliente plantee. Cada incidencia recogerá datos como modelo afectado, cliente, así como alguna información extra relacionada con la forma de proceder o solventar. Además se extraerán datos relevantes relacionados con el funcionamiento de cada producto y sobre el uso que cada usuario hace. Dependiendo del tipo de incidencia puede llevarse a cabo distintas alternativas para la solución de la misma como por ejemplo sustituir el producto por otro de características similares. Incluso dependiendo del tipo de cliente (por ejemplo un VIP) que tiene la incidencia se puede actuar de distinta forma como por ejemplo ofrecer un modelo superior a un VIP.

Otro aspecto importante a destacar es el funcionamiento de cada modelo. Muchas veces los clientes no entienden el funcionamiento de un producto y existen dudas bastantes comunes. Por ello se crean manuales de usuario que ayuden a comprender su funcionamiento y contienen una lista de preguntas más frecuentes (FAQ). Además si de manera usual una duda sobre el uso del producto lleva a otra duda el sistema indicará la respuesta a la 2<sup>a</sup> duda relacionada con la 1<sup>a</sup>.

## B.2. Acta 2 (31/10/2007)

El proyecto trata de documentar diseñar e implementar un *Sistema de Atención al Cliente o CRM* que tiene como objetivo **guardar y gestionar** toda la información posible acerca de productos, clientes, ventas, incidencias, quejas, soluciones, así como las **relaciones** que puedan existir entre ellos, y **dar** respuestas a las consultas de los clientes. Además, una funcionalidad muy importante que debería implementar el sistema que se está especificando es el **aprendizaje**, basado en casos, por lo cual podremos obtener soluciones satisfactorias.

El sistema registrará cada consulta que el cliente plantee. El aprendizaje del sistema es muy importante para determinar de qué forma suele actuar cada cliente. Esto nos permite averiguar cuáles son sus preferencias y en general patrones de comportamiento. Los datos se extraerán bien de la forma de actuar de cada cliente o mediante cuestionarios. A partir de estos datos podemos crear perfiles de clientes que actúen de la misma forma. Pasando a la descripción técnica de nuestro sistema, existen elementos fundamentales tales como:

- Productos o Equipos
- Clientes
- Consultas
- Incidencias o Averías
- Quejas o reclamaciones
- Recomendaciones
- Soluciones

Los productos son aquellos elementos de consumo, que han sido vendidos o están en stock o incluso puede que aun estén en desarrollo experimental (prototipos). Los productos se pueden identificar unívocamente por su Serial Number (S/N), además de disponer una serie de atributos, que aun no interesa detallar. Como es natural, pueden existir varios modelos para un mismo producto.

Los clientes son todos aquellos usuarios, pasados, presentes y futuros, de los que tengamos constancia y datos en nuestro sistema, que podrán ser utilizados para la generación de estadísticas, y, por consiguiente, para el aprendizaje. Se les podrá clasificar según perfiles, para darle un servicio de asistencia

diferente dependiendo del perfil (económico, social, etc). Por ejemplo: Pepito. Varón. 45 años. Dos hijos. Médico = Torpe. Explicarle las cosas de manera sencilla. Juan. Varón. 30 años. Sin hijos. Ingeniero de Telecomunicaciones = Listo. Hablarle con términos técnicos. El sistema recogerá información acerca de las acciones de los clientes. Las compras, las reclamaciones, las consultas, los gustos, los problemas que tienen con los productos, debido a su uso, a su manejo o a la facilidad que tienen de entenderlos.

Las consultas son las llamadas que hacen los clientes al servicio técnico. Pueden ser por varias causas: averías, reparaciones, quejas, recomendaciones o información adicional.

Las Incidencias o Averías ocurren cuando un producto deja de funcionar correctamente. Entonces, el cliente usará el servicio de soporte técnico para intentar solucionarlo. Puede haber muchas clases de averías, desde, por ejemplo, rotura por caída, o que se moje un producto, hasta que el cliente haya desconfigurado el producto y no sepa como restaurar el estado original. Evidentemente habrá más consultas por averías mientras el producto esté en garantía, ya que a partir de esa fecha, es muy probable que compense vender un producto nuevo que arreglar el viejo, pero puede que no. Cada incidencia recogerá datos como modelo afectado y tipo de cliente, y a partir de ella se generará información extra relacionada con la forma de proceder o solventar. Además se extraerán datos relevantes relacionados con el funcionamiento de cada producto y sobre el uso que cada usuario hace.

Las Reparaciones se refieren al arreglo de un producto. Como consultas, las reparaciones pueden proporcionar al cliente información sobre fechas, plazos y costes de la reparación antes de que se produzca. Todos estos datos pueden surgir a partir de la experiencia con otros productos similares o en ocasiones distintas. También se podría proporcionar información, una vez que el producto se ha mandado a reparar, del tiempo que quedará (por ejemplo, se detecta que es un fallo de un componente y que va a tardar un tiempo en llegar uno nuevo). Una idea extra sería hacer un sistema de alarmas, que cuando se modificara un dato de reparaciones se avise al cliente.

Las quejas son las reclamaciones de los clientes, que no quedan satisfechos con el producto o con el servicio, por ejemplo con un retraso en una entrega de un equipo en reparación o venta.

Las consultas de Información se refieren a las características que tiene un producto, funcionalidad (con vistas a adquirirlo o no), propuestas de equipos, dependiendo del perfil socio-económico del cliente, o sobre su manejo. Muchas veces los clientes no entienden el funcionamiento de un producto y existen dudas bastantes comunes. Cada modelo de producto debería disponer de

un manual y de una lista de preguntas más frecuentes (FAQ) que pueden ser diferentes dependiendo del perfil. Además si de manera usual una duda sobre el uso del producto lleva a otra duda el sistema indicará la respuesta a la segunda duda relacionada con la primera.

Las recomendaciones son las propuestas que se le da a un cliente que pide un producto con unas características determinadas. Se le intentaría proponer algo que contentara lo máximo posible al cliente, dependiendo de su perfil socioeconómico y de lo que requería.

Las soluciones son aquellas medidas que el equipo del Servicio Técnico toma para cada situación, dependen del tipo de cliente, del caso o problema que se presenta, y del grado de satisfacción conseguido en ocasiones anteriores. Según el tipo de cliente (por ejemplo un VIP) que tiene la incidencia se puede actuar de distinta forma como por ejemplo ofrecerle un modelo superior. Por ejemplo, si a un representante de una empresa se le ha dado un terminal telefónico cutre y ha quedado decepcionado porque la imagen que han tenido los clientes de su compañía ha sido mala, el sistema aprende y la próxima vez no toma esa decisión. Dependiendo de la consulta puede llevarse a cabo distintas alternativas para la solución de la misma como por ejemplo sustituir el producto por otro de características similares.

### B.3. Acta 3 (07/11/2007)

Se le presenta el documento de presentación del proyecto con las modificaciones de la versión anterior, completo y adecuado a las expectativas salvo un par de matices

Ver [documento modificado](#).

- Los CLIENTES que todavía no han comprado, no son CLIENTES sino CONTACTOS
- Los USUARIOS son los que van a usar la utilidad que estamos desarrollando.
- Las CONSULTAS van a ser todas las llamadas que realizan los CLIENTES y los CONTACTOS, y pueden ser de varios tipos dependiendo del motivo:
  1. Averías o INCIDENCIAS
  2. Quejas o RECLAMACIONES
  3. CONSULTAS DE INFORMACIÓN, que a su vez, puede ser por conocer la funcionalidad de un determinado producto, alguna duda de su uso, o también para pedir recomendación para adquirir uno nuevo, con unas características determinadas y en base a su perfil socioeconómico.
  4. Las SOLUCIONES comprenderán, para cada consulta, el desenlace de la misma, así como el grado de satisfacción de cada cliente. Por tanto, podría ser, o bien la reparación de una avería, la tramitación de una queja, o la recomendación de producto que se le ha dado a un cliente.

Además, no interesa que un producto vaya a tener más reparaciones mientras esté en garantía, ya que eso, como mucho, sería un dato estadístico más.

Una vez que ya tenemos una idea de lo que trata el producto, se nos pide definirlo mediante [diagramas de secuencia](#) .

Un diagrama de secuencia, (formalmente diagrama de traza de eventos) muestra las interacciones entre objetos ordenadas en secuencia temporal. Muestra los objetos que se encuentran en el escenario y la secuencia de mensajes intercambiados entre los objetos para llevar a cabo la funcionalidad descrita por el escenario.

---

Por ejemplo, qué sucede exactamente cuando un cliente llama por una avería, si se le toman los datos, si se consulta en la Base de Datos de qué tipo de avería se puede tratar, o si se trata únicamente de una desconfiguración del producto, ayudando al cliente a configurarlo correctamente.

Otro ejemplo sería el de un cliente que pide información porque se quiera comprar un nuevo modelo de un producto que ya tiene. Haría falta, por ejemplo, consultar qué producto está usando y cual es la nueva versión, y se le enumerarían las características del nuevo producto, por ejemplo.

## B.4. Acta 4 (14/11/2007)

Revisión de los diagramas de secuencia, resultando demasiado triviales en algunas partes como demasiado orientados al código en otras. Es por eso que nos dividimos el trabajo para hacer cada uno de nosotros un tipo de *consulta*. De manera individual realizaremos un trabajo de abstracción e imaginación de la lógica de negocio. Generaremos un diagrama en el que quede reflejado cuáles son las características o atributos o propiedades o circunstancias del cliente y su consulta, por lo que se ha de priorizar para que el sistema proporcione alternativas. Como aún estamos en una fase preliminar, los tipos de respuesta serán de carácter general, (p.e. acción dura, acción media, acción blanda, etc)

[Diagrama de flujo de las incidencias](#)

## B.5. Acta 5 (21/11/2007)

Exponemos los diagramas generados, en los que se representa el árbol de decisiones que el sistema tendría que tomar para ofrecer una respuesta adecuada dependiendo de las condiciones del cliente. Estas condiciones serían por las que el sistema tendría que priorizar. Entre las condiciones del cliente propuestas tenemos:

- Tipo de cliente
- Número de vez que llama
- Otras relacionadas con la naturaleza del caso en cuestión ( reclamaciones, incidencias, soluciones)

Tras revisar los diagramas encontramos algunos fallos:

- Necesidad de aumentar el número de condiciones.
- Necesidad de aprendizaje para que el sistema sea capaz de seguir las políticas de la empresa.

El documento de las incidencias se puede encontrar en [\[PNG \]](#) o [\[PDF\]](#)

## B.6. Acta 6 (28/11/2007)

Revisamos los diagramas de manera conjunta con el profesor. En la revisión descubrimos algunos cambios y mejoras.

- Debemos prescindir por el momento de la lógica de negocio interna, de las acciones que se llevarán a cabo así como de la manera que las *propiedades* del cliente interactúan entre sí. Todos estos factores será impuestos por la empresa que use la aplicación para sus intereses comerciales.
- El sistema será capaz de proponer un conjunto de acciones que dependerán de manera dinámica del histórico del cliente así como de otros valores sociales.
- Revisando el histórico del cliente se podrán extraer información acerca de las acciones realizadas sobre el cliente. Esto es, se podrán sacar datos estadísticos que permitan cambiar las acciones disponibles.
- Las acciones que serán propuestas al cliente serán determinadas por un sistema inteligente que tendrá en cuenta los puntos anteriores

Por ellos generamos los diagramas correspondientes atendiendo a estos cambios. En formato [[PNG](#)] y [[PDF](#)]

## B.7. Acta 7 (05/12/2007)

Realizamos una revisión de los diagramas con las modificaciones de la reunión anterior. Limitándonos a mostrar la estructura que tendría el sistema así como los “atributos” requeridos para que la inteligencia proponga una o varias respuestas. Lo que llamaremos *acciones*.

Empezamos una nueva fase: el diseño de la base de datos. Primeramente capturamos los atributos que hacen falta para modelar el comportamiento deseado. En esta primera fase nos centraremos en poner el nombre de la tabla junto con los campos que contendría, sin centrarnos en el tipo de datos de los campos, ni en los datos que contendría cada tabla, ni optimizaciones sobre la bdd.

El sistema debe tener un comportamiento de acción - reacción. Por ellos es importante almacenar las acciones que hemos hecho sobre cada cliente, así como las reacciones que los clientes tienen al realizar dicha acción. De todos estos datos se podrán extraer comportamientos estadísticos con los que se sustituirán las acciones según los intereses de la empresa que use *CRMI* (por ejemplo: aquellas acciones que provoquen bajas de los clientes deberán ser sustituidas por otras si así lo aconsejan)

Como primera aproximación generamos una vista preliminar de las tablas y los campos que contendrían para gestionar las *incidencias*. Disponible en versión [[PDF](#)] y [[PNG](#)]

## B.8. Acta 8 (12/12/2007)

En la reunión revisamos el esquema de tablas y atributos generados. Sin encontrar mayor problema, generaremos un ejemplo a modo ilustrativo para hacer más palpable que las tablas propuestas y sus atributos cumplen las especificaciones del proyecto. Aun no estamos en condiciones de asegurar de que esas serán todas las tablas de nuestra BBDD en nuestro sistema (seguramente harán falta más) es por ello por lo que es necesario ver la completitud del sistema con el soporte de las tablas actual. Al hacer la revisión de las tablas nos dimos cuenta de que hacía falta unificar los nombres así como la disposición de las mismas.

El documento generado para la orden del día es un ejemplo que muestra para un caso genérico *incidencias*, aunque se han usado algunos datos concretos para aportar mas claridad al ejemplo, de cuales serían los atributos que se han de recuperar de la bases de datos así como la relación de algunas atributos con algunas tuplas en otras tablas. El documento generado se encuentra en formato [\[PNG\]](#) y [\[PDF\]](#)

Para ir entrando en materia se propone una implementación de la base de datos para ellos se genera el diagrama E/R relativo a las *incidencias*. El diagrama E/R se encuentra en [\[PNG\]](#) y [\[PDF\]](#)

### **B.9. Acta 9 (19/12/2007)**

No hubo reunión. Debido a la cercanía del comienzo de las vacaciones de navidad, el profesor no tuvo clase la hora anterior a la reunión, por lo tanto se suspendió la reunión posponiéndola para el siguiente miércoles de enero.

## B.10. Acta 10 (09/01/2008)

En la primera parte el profesor revisa los documentos generados en la [Reunión 8](#) y muestra su conformidad con lo que se propone en ellos. Desde hace algunas reuniones, nuestro sistema tiene un *modulo* que se incorpora como una *caja negra*. Se trata del módulo de la inteligencia. Como es natural ese módulo dispondrá de una serie de reglas, las cuales serán actualizadas y coherentes con el conocimiento estadístico extraído de la propia interacción de los clientes con el sistema. Además puede estar asentado sobre un conocimiento mas general, pero siempre estará adaptado a las reacciones que tienen los clientes. Existen diversas maneras de implementar el módulo de inteligencia:

- Usando un sistema basado en reglas, haciendo uso de la ingeniería del conocimiento, así como el uso de las herramientas que proporciona esta metodología.
- Usando algún componente que almacene las reglas y sea mantenido de forma manual, como puede ser una base de datos que contenga las condiciones para que una determinada acción se pueda realizar.

Por el momento el profesor sugiere que usemos el 2<sup>o</sup> modo de implementación (usando una BBDD)

En ambos sistemas las reglas deben tener la siguiente forma:

Condición 1	Condición 1	...	Condición i	...	Condición N	Acción
Valor a	Valor b	...	Valor c	...	Valor d	Acción A
Valor b	Valor f	...	NULL	...	Valor g	Acción B

De tal forma que si se cumplen todas las condiciones de una fila se activara la acción correspondiente a dichos valores.

Como condiciones aparecerán todas las condiciones usadas por alguna de las reglas almacenadas en el sistema. De esta forma será frecuente la aparición de valores NULL para alguna de las condiciones de una regla. Esto será así siempre y cuando ese valor no sea relevante para aplicar dicha acción.

Para esta reunión se trata de esbozar el diseño del módulo de inteligencia. Centrándonos en el diseño de la tabla que almacene las reglas.

Se muestra un ejemplo de lo que contendría la tabla de reglas.

id	idAccion	edad_Min	edad_Max	profesión	tipoCliente	estUltLlamada
1	1	0	17	NULL	Normal	NULL
2	2	18	26	Ingeniería	Normal	conforme
3	1	18	26	NULL	moroso	disconforme
4	1	35	45	Funcionario	NULL	conforme

La tabla reglas forma parte de la BBDD que se muestra en formato

[\[PDF\]](#) y [\[PNG\]](#)

## B.11. Acta 11 (16/01/2008)

Revisamos los diagramas propuestos por el profesor. Dado el visto bueno, nos pide que implementemos la base de datos. Para ello usaremos el diseño generado con el programa fabFORCE generando así las correspondientes instrucciones SQL de creación de tablas.

El código SQL para la generación de la base de datos se muestra a continuación: Notar que el código que se genera crea las tablas necesarias y algunas filas para su uso con las reclamaciones. La incorporación de la parte de soluciones y reclamaciones solo conllevaría algunas modificaciones en el actual diseño. Siendo necesario la inclusión de alguna nueva tabla así como la posible inclusión de atributos adicionales en las ya existentes.

Descarga el archivo [1.sql](#)

## B.12. Acta 12 (23/01/2008)

Presentamos los diseños preliminares de la base de datos al profesor. Sin entrar en los detalles propios del diseño, para las siguientes reuniones empezaremos a implementar la BBDD. Como comentamos en reuniones pasadas, utilizaremos MySQL para gestionar la base de datos del proyecto. Para poder realizar pruebas en los laboratorios se solicitó acceso a MySQL.

Comprobado que se carga la base de datos en los laboratorios. Se ha modificado el archivo generado para la pasada reunión que se adjunta a continuación. [2.sql](#)

### **B.13. Acta 13 (04/03/2008)**

Realizamos una integración de las 3 partes constituyentes el CRMI, a saber:  
Reclamaciones, Incidencias y Consultas de Información

El diseño de la BBDD realizado es el siguiente [Diseño completo](#)

Las instrucciones SQL para crear dicha BBDD son

## **B.14. Acta 14 (11/03/2008)**

Como se pedía para esta reunión se genera un pequeño ejemplo de los pasos que se llevarían a cabo vistos desde el punto de vista de la BBDD. El ejemplo trata de manera esquemática el proceso desde que un nuevo cliente llega al sistema, crea una nueva incidencia, pasado un tiempo consulta el estado de la misma. Además de quedar registrado en el según el modelo acción - reacción.

Al final se muestra un pequeño ejemplo de una selección para obtener las acciones que se le aplicarían al cliente. Tener en cuenta que el ejemplo es una simplificación. Para la obtención de las acciones sería necesario restringir mas por mayor numero de campos. Además la información de las reglas es meramente ilustrativa no contiene significado adecuado en el dominio.

## B.15. Acta 15 (2/04/2008)

Esta reunión (Reunión 15) estaba prevista para el día 25/03/2008, por motivos de incompatibilidad de horarios se ha pospuesto a una reunión posterior. Corresponde al Acta 16.

Se han realizado algunos cambios en la BBDD. De esta forma se clarifica la estructura de la BBDD en aras de facilitar su genericidad y facilitar el desarrollo.

- Cambio de nomenclatura, para que el siguiente punto tenga sentido semántico. Ahora las Incidencias pasan a llamarse Averías. Y del mismo modo el conjunto de Averías, Reclamaciones y Consultas pasan a llamarse Incidencias.
- Ahora sólo hay una tabla en la que se almacena la información referente a Averías, Reclamaciones y Consultas. Por ello se combinan las 3 tablas anteriores para crear una nueva que se llame Incidencias. Además como es natural es necesario añadir un campo extra para poder diferenciar el tipo de incidencia. Del mismo modo las tablas de acciones y reacciones se simplifican reduciendo el número de campos sin uso.
- Para poder elegir de manera correcta la acción es necesario identificar las acciones con el tipo de Incidencia, en sentido global, ya que en cualquier caso, la acción no sólo depende del cliente, sino que depende en mas medida de la incidencia asociada.

### Diagrama de flujo

El diagrama de flujo muestra el comportamiento del CRMI para el caso de un cliente existente o no quiera interactuar con el sistema. El diagrama se centra sobre todo en las reglas y la selección de las acciones que se realizaran al cliente.

Nueva versión del documento disponible en el Acta 16

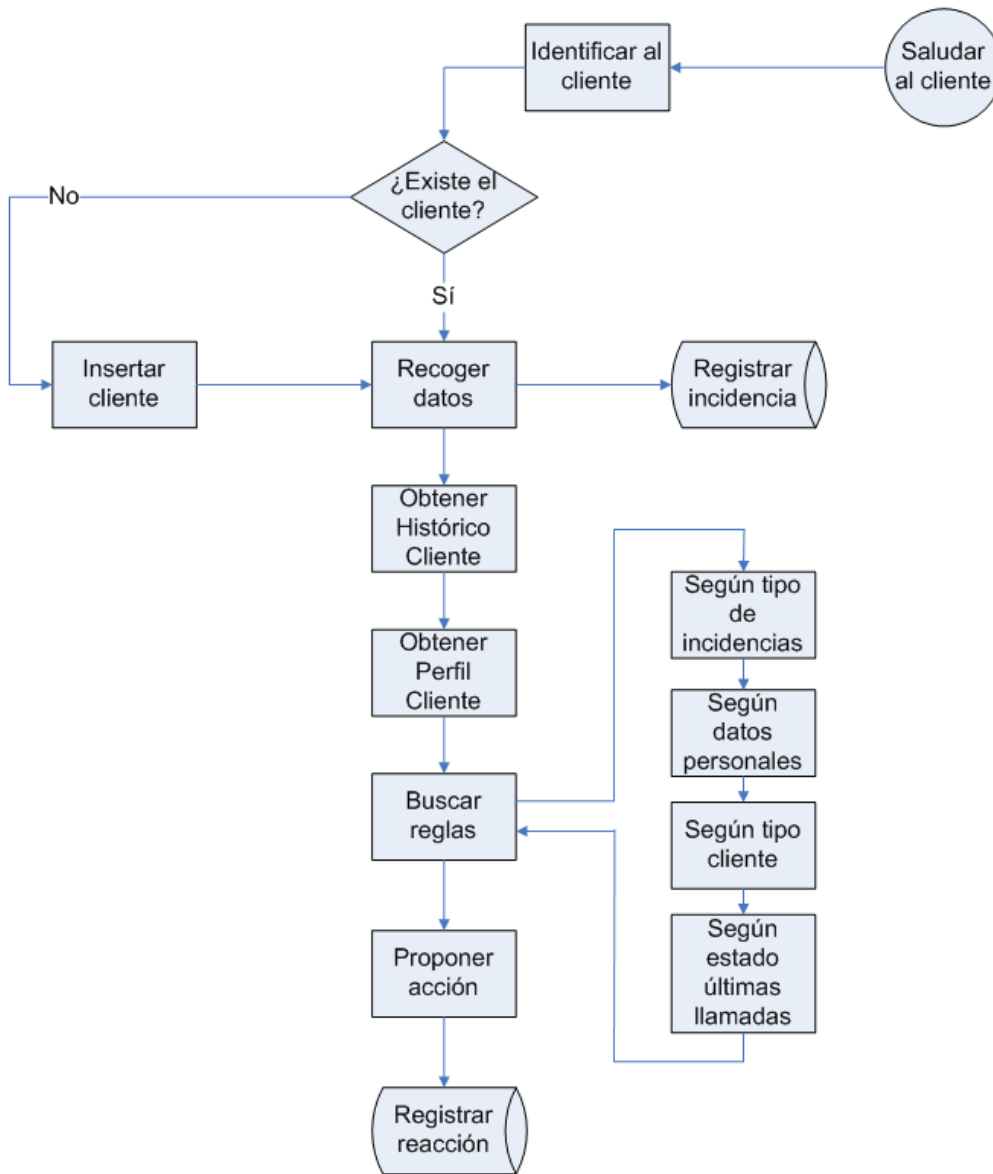


Figura B.1: Diagrama de flujo

## B.16. Acta 16 (16/04/2008)

En esta reunión se enseña al profesor los casos de uso, la documentación referente a la memoria y el diagrama de flujo generado en el acta 15. Respecto al dicho diagrama se sugiere la modificación y explicación de las variables que intervienen en el módulo buscar reglas correspondientes a la etapa de inteligencia. Por ellos se genera un diagrama modificado según lo propuesto.

En una segunda parte de la reunión se concretan las [partes que debe contener la memoria](#).

Se incluye una nueva versión del diagrama de flujo generado en el acta 15 incorporando el árbol discriminatorio, en lugar de un filtro como aparecía en la versión anterior. Las diferencias entre filtro y árbol discriminatorio son:

- Forma de realizar las preguntas
  - En el filtro se pregunta por todas las variables al la vez.
  - En el árbol se pregunta cada vez por una variable distinta. De tal forma que se pregunte primera por las variables mas discriminatorias. Además no siempre preguntará por todas las variables en todos los casos.
- Resultados obtenidos
  - En el filtro los resultados obtenidos son siempre iguales sin importar el orden relativo de las variables.
  - En el árbol discriminatorio los resultados podrían ser distintos dependiendo del orden de las variables.

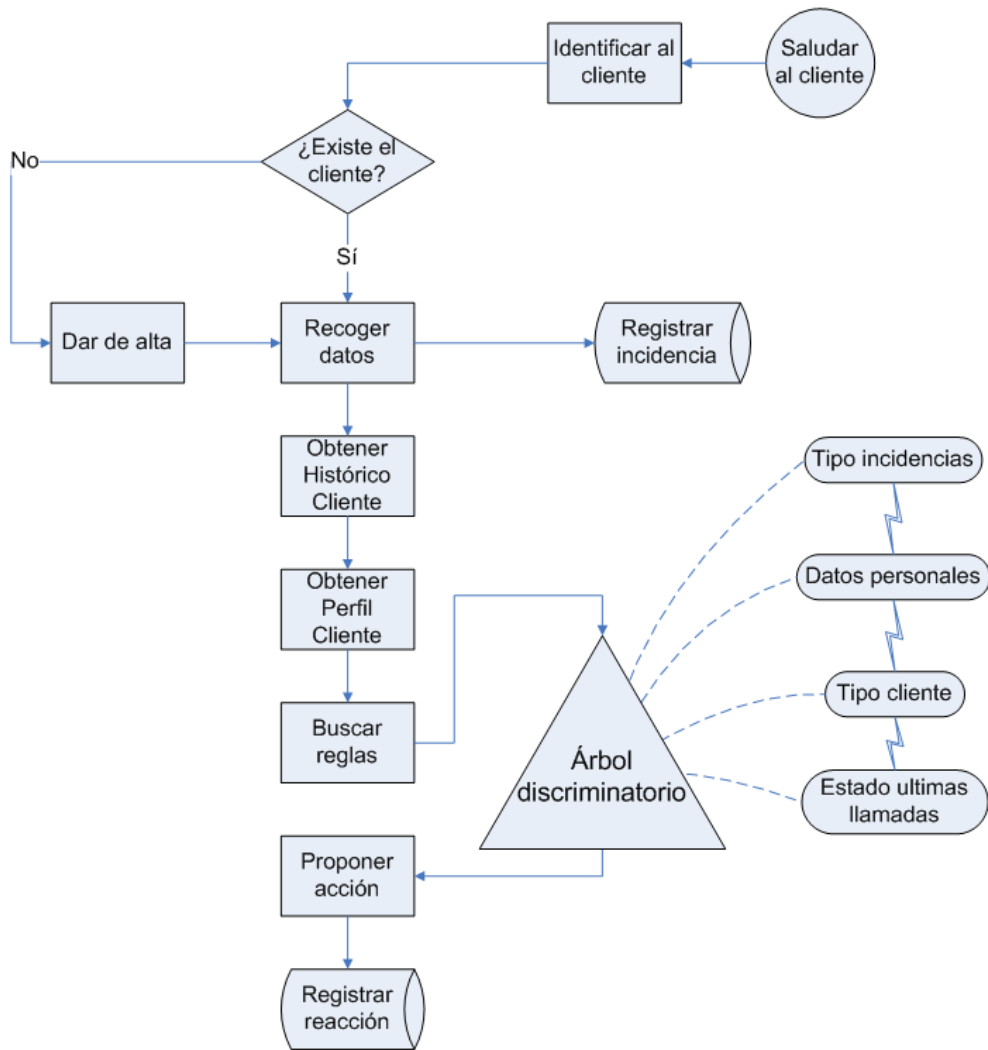


Figura B.2: Diagrama de flujo - Versión 2

## B.17. Acta 17 (29/04/2008)

En esta reunión hemos estado hablando de como debe estar estructurada la memoria final del proyecto. Los distintos apartados que la componen son los siguientes:

**Antecedentes:** Aquí definimos que es un CRM-I mediante una definición completa

**Objetivos del proyecto:** Dejaremos claro que es lo que queremos mostrar con el proyecto

**Metodología de desarrollo:** Explicaremos cómo nos hemos organizado en el proyecto y cómo se han llevado a cabo las reuniones.

**Análisis:** Se describen las distintas técnicas estudiadas para llevar a cabo nuestro objetivo

**Diseño:** En este apartado mostraremos los diagramas y la estructura final de la base de datos. Se incluyen los casos de usos ya implementados

**Implementación:** Describimos las tecnologías de desarrollo utilizadas y mencionamos parte del interfaz con algún pantallazo incluido

**Resultados:** Se mencionan los casos más significativos.

**Conclusiones:**

**Futuras extensiones:**

**Bibliografía:**

En cuanto a las extensiones de cada apartado no está muy claro.

**B.18. Acta 18 (21/05/2008)**

Reunión de control para revisar la memoria. El objetivo de esta reunión es presentar al profesor la memoria para que nos de su opinión y corrija posibles fallos, tanto como cambios de contenidos, punto de vista, temas que se debieran tratar, etc. A fecha de la reunión la memoria se encuentra en pleno desarrollo, aún no esta terminada, pero si bastante avanzada, por eso requerimos el punto de vista del profesor para que establezca nuevos objetivos de la memoria. Por falta de tiempo del profesor, en la reunión solo pudimos darle la memoria en formato editable.

**B.19. Acta 19 (04/06/2008)**

En la reunión se revisa la memoria y se establecen nuevos objetivos. El contenido de la memoria es adecuado, pero quedan limar detalles relacionados con la expresión y el estilo. Aunque el contenido es correcto quedan temas pendientes de desarrollar. Es una reunión bastante breve aunque nos sirve de punto de partida para seguir ampliando la memoria.

## B.20. Acta 20 (10/06/2008)

Tras la revisión de la memoria del acta anterior (Acta 19) y habiendo ampliado la memoria en los puntos comentados, se procede a una revisión mas exhaustiva con el profesor revisando más en detalle los contenidos de la memoria. Se establecen nuevos objetivos para la memoria: añadir un apartado de ejemplos de funcionamiento que refleje el funcionamiento del sistema. Además comentamos cuestiones relacionadas con la exposición del proyecto:

- Transparencia con la metodología desarrollada
- Transparencia con la estructura del sistema (arquitectura componente... en alto nivel)
  - Diagrama de bloques (BBDD mecanismos inferencia, paquete estadístico)
- Transparencia con los resultados (ejemplos de funcionamiento)
- Transparencia con el diseño de la BBDD (reducido, no mostrar atributos ni relaciones como por ejemplo es un o es de tipo. quitar la tabla tipo que relaciona reglas con incidencias)