



Universidad  
Complutense  
Madrid



Facultad  
de  
Informática

**PROYECTO DE SISTEMAS INFORMÁTICOS**

**CURSO 2009/2010**

**ASISTENCIA GUIADA A LA CONDUCCIÓN EN  
FUNCIÓN DE ERRORES HISTÓRICOS PREVIOS**

**Autor:** Juan Pedro Franco Abia

**Director:** Miguel Ángel Blanco Rodríguez



El abajo firmante autoriza a la Universidad Complutense 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.

Fdo: Juan Pedro Franco Abia

*A mi padre*

## **Agradecimientos**

Al final de este viaje, la carrera, me gustaría dar las gracias a todas las personas que han formado parte de él.

Empezaré con mi familia que con cariño ha sabido mostrarme su apoyo. Tía Asun, por los sándwiches mixtos y los “caramelos de viajar”; Jaime y Luis, ¿qué sería de un hermano pequeño sin sus hermanos mayores?; y Mamá, por todo.

A los compañeros y amigos de la facultad porque con las conversaciones en la cafetería (o en clase) han hecho que recuerde estos 6 años el resto de mi vida. Bryan, Charlie, Javi, Julio, Luis, Nacho... gracias.

A los amigos de Saldaña (Isa, Manu, Mario, Rafa y toda la Whyzerby) porque una vez al mes (aproximadamente) puedo olvidarme de lo que hago el resto del mes y disfrutar de la vida como solo allí sabemos.

Al resto de amigos hechos por el camino (Abi, Ana, Lukasz, Marina...) porque me han demostrado que cualquier lugar y situación es buena para crear una amistad.

A los profesores porque con su trabajo han conseguido que sea el ingeniero que soy hoy.

Y, por último, a usted, lector, por dedicar su tiempo a revisar este proyecto, espero que le sea útil y le sirva de ayuda.

## **Resumen**

El objetivo del proyecto es crear un sistema experto basado en casos que indique a un usuario que accidentes puede sufrir en la carretera. Nuestra aplicación consiste, básicamente, en una base de datos donde se recoge toda la información de los usuarios. A partir de esta información se obtienen los casos. Cuando un usuario quiere conocer que accidentes puede sufrir en la carretera se valoran los casos.

### **Palabras claves:**

Base de datos, motor de inferencia, sistema experto, razonamiento basado en casos.

## **Abstract**

The goal of this project is to create a case-based expert system which shows an user the traffic accidents he/she may have. Our application has a database where the users' information is kept. Using this information we obtain the cases. When a user wants to know the traffic accidents he/she may have, the cases are studied.

### **Key words:**

Database, inference engine, expert system, case-based reasoning.

## Índice general

	<b>Página</b>
<b>Autorización</b>	<b>ii</b>
<b>Agradecimientos</b>	<b>iv</b>
<b>Resumen</b>	<b>v</b>
<b>Abstract</b>	<b>vi</b>
<b>Introducción</b>	<b>1</b>
<b>CAPÍTULO 1. Desarrollo del proyecto</b>	<b>2</b>
<b>CAPÍTULO 2. Análisis funcional</b>	<b>3</b>
2.1. Funcionamiento interno del sistema .....	3
2.2. Interacción con el usuario .....	4
2.3. Información guardada de los usuarios .....	4
2.4. Historial de accidentes .....	7
2.5. Casos .....	12
<b>CAPÍTULO 3. Diseño de la base de datos</b>	<b>14</b>
3.1. Base de datos del usuario .....	14
3.2. Base de inferencia .....	18
<b>CAPÍTULO 4. Creación de un prototipo</b>	<b>21</b>
4.1. Aplicación para inicializar la base de datos .....	21
4.2. Prototipo .....	28

<b>CAPÍTULO 5. Creación de la memoria</b>	<b>31</b>
5.1. Contenido del CD .....	31
<b>Conclusiones</b>	<b>32</b>
<b>APÉNDICE A. Sentencias SQL</b>	<b>33</b>
<b>APÉNDICE B. Instrucciones de uso de los ejecutables</b>	<b>37</b>
B.1. Crear la base de datos.....	37
B.2. Ejecutar la aplicación de inicialización.....	38
B.3. Ejecutar la aplicación .....	38
<b>APÉNDICE C. Actas</b>	<b>39</b>
16 de noviembre de 2009.....	40
23 de noviembre de 2009.....	41
30 de noviembre de 2009.....	42
14 de diciembre de 2009.....	50
21 de diciembre de 2009.....	51
12 de enero de 2010.....	53
18 de enero de 2010.....	54
25 de enero de 2010.....	55
22 de febrero de 2010.....	56
<b>Bibliografía</b>	<b>57</b>

## Índice de figuras

	<b>Página</b>
2.1. Características de un usuario.....	7
2.2. Clasificación de los errores .....	8
2.3. Clasificación de los accidentes.....	9
2.4. Clasificación de las consecuencias.....	10
2.5. Ejemplo de un suceso.....	11
2.6. Ejemplo de un caso I.....	13
2.7. Ejemplo de un caso II.....	13
3.1. Diagrama E-R de la base de datos del usuario .....	16
3.2. Esquema E-R de la base de inferencia I.....	19
3.3. Esquema E-R de la base de inferencia II.....	20
4.1. Formulario de inserción de un coche .....	22
4.2. Ventana para consultar los coches introducidos.....	22
4.3. Pestaña Relaciones – Ver Usuario.....	23
4.4. Relaciones de un usuario.....	24
4.5. Inserción de un suceso en el historial de un usuario .....	24
4.6. Como rellenar los campos Valor inicial y Valor final.....	25

<b>4.7.</b> Un caso con 3 premisas y 2 conclusiones .....	28
<b>4.8.</b> Inserción de un usuario.....	29
<b>4.9.</b> Pestaña Gestión usuario.....	29
<b>4.10.</b> Posibles accidentes de un usuario .....	30

## INTRODUCCIÓN

El objetivo del proyecto es crear un sistema experto basado en casos que indique a un usuario que accidentes puede sufrir en la carretera.

La aplicación consiste, básicamente, en una base de datos donde se recoge toda la información de los usuarios. A partir de esta información se obtienen los casos. Cada caso tiene una serie de premisas (características del usuario) y una serie de conclusiones (accidentes). Cuando un usuario quiere conocer que accidentes puede sufrir en la carretera se valoran los casos, si el usuario cumple todas las premisas entonces podrá sufrir los accidentes detallados en las conclusiones del caso.

La información que se guarda sobre cada usuario es toda aquella que puede tener relación con las causas de una accidente. Entre esta información se incluyen datos personales (edad, nivel de estudios...), hábitos de vida (kilómetros recorridos al año, aficiones...) y un historial con accidentes sufridos anteriormente.

## CAPÍTULO 1

### DESARROLLO DEL PROYECTO

El proyecto se ha desarrollado siguiendo las instrucciones y explicaciones que el director proporcionaba en las reuniones semanales. En cada una de estas reuniones se presentaba el trabajo realizado durante la semana y se discutía sobre él, valorando que debía mantenerse y que debía cambiarse. Tras la valoración del trabajo realizado, el director indicaba el siguiente paso a realizar. Cada reunión ha quedado registrada en un acta<sup>1</sup> donde consta la fecha de la reunión, los asistentes, los temas tratados, los acuerdos alcanzados y el trabajo a realizar durante la siguiente semana.

En la primera reunión mantenida se decidió dividir el proyecto en cuatro fases para que su seguimiento y desarrollo fueran más sencillos. Dichas fases son:

Fase 1 - Análisis funcional: En esta fase se definió el proyecto y todos los elementos que intervienen en él.

Fase 2 - Diseño de la base de datos: Durante esta fase diseñamos la base de datos basándonos en los elementos analizados en la fase anterior.

Fase 3 - Creación de un prototipo: En esta fase del proyecto se implementaron dos aplicaciones. La primera aplicación sirve para insertar en la base de datos todos los datos de una forma sencilla sin tener que recurrir a sentencias SQL. La segunda aplicación es el objetivo de este proyecto.

Fase 4 - Creación de la memoria: En la última fase del proyecto se redactó esta memoria.

El trabajo realizado en las diferentes fases se desarrolla en los sucesivos puntos de la memoria.

---

<sup>1</sup> Las actas se pueden consultar en el apéndice C.

## CAPÍTULO 2

# ANÁLISIS FUNCIONAL

Durante esta fase definimos el funcionamiento interno de nuestro sistema, la interacción con el usuario, la información guardada de los usuarios, el historial de accidentes de un usuario y la forma de los casos.

### **2.1. Funcionamiento interno del sistema**

Nuestra aplicación es un sistema experto basado en casos.

La aplicación consiste, básicamente, en una base de datos donde se recoge toda la información de los usuarios. A partir de esta información se obtienen los casos. Cada caso tiene una serie de premisas (características del usuario) y una serie de conclusiones (accidentes).

En un sistema experto los casos se crean usando un motor de inferencia. El motor de inferencia procesa la información. Para cada accidente estudia los usuarios que han sufrido dicho accidente y obtiene las características que tienen en común todos los usuarios o la mayor parte de ellos. Con estas características y el accidente construye un caso.

En nuestra aplicación hemos creado nosotros mismos los casos ya que la implementación de un motor de inferencia queda fuera del alcance del proyecto.

Cuando un usuario quiere conocer que accidentes puede sufrir en la carretera se valoran los casos, si el usuario cumple todas las premisas entonces podrá sufrir los accidentes detallados en las conclusiones del caso.

## 2.2. Interacción con el usuario

El usuario debe, en primer lugar, introducir sus datos en la base de datos usando la aplicación. Cuántos más datos introduzca el usuario más correctos serán los posibles accidentes obtenidos.

Una vez que ha introducido sus datos, el usuario puede consultar los accidentes con más probabilidades de sucederle. También puede añadir nueva información, por ejemplo, puede añadir un nuevo accidente a su historial de accidentes. La nueva información introducida modificará los accidentes obtenidos.

## 2.3. Información guardada de los usuarios

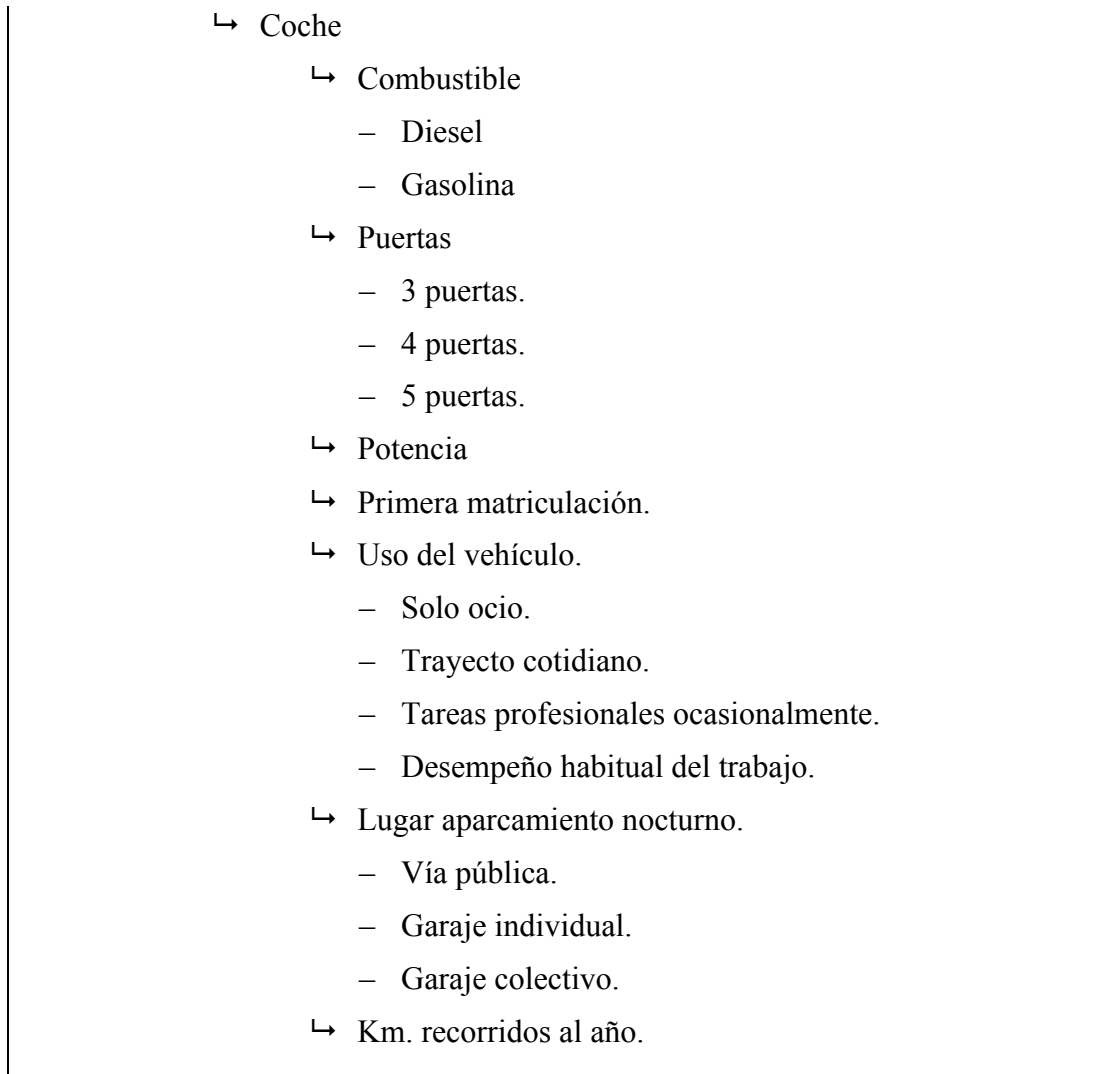
La información almacenada de cada usuario es usada por el motor de inferencia para calcular los casos. Por tanto, debemos guardar todas las características del usuario que puedan tener relación con las causas de un accidente y un historial de accidentes sufridos para cada usuario, así el motor de inferencia podrá relacionar los accidentes con los usuarios.

La características del usuario las hemos dividido en cuatro grupos: personales (edad, sexo...), sociales (estado civil, hijos...), profesionales (empresa, horario de trabajo...) y de conducción (fecha del carnet, kilómetros recorrido al año...). Las características guardadas de un usuario son:

- Personales
  - ↳ NIF
  - ↳ Nombre
  - ↳ Apellidos
  - ↳ Edad
  - ↳ Sexo
- Sociales
  - ↳ Estado civil.
    - Soltero
    - Pareja de hecho.

- Casado
  - Separado
  - Divorciado
  - Viudo
- ↳ Número de hijos.
  - ↳ Menores de edad.
  - ↳ 18 – 25
  - ↳ > 25
- ↳ Zona de vivienda.
  - Zona residencial.
  - Centro ciudad.
  - Resto ciudad.
  - Pueblo
- ↳ Código postal.
- ↳ Hobbies
  - ↳ Públicos
    - Conciertos
    - Eventos deportivos.
    - Turismo fin de semana.
    - Senderismo
    - Cine
    - Teatro
  - ↳ Privados
    - Leer
    - Cocina
    - Pintura
    - Videojuegos
- Profesionales
  - ↳ Empresa
    - Multinacional / Grande.
    - Mediana
    - Pequeña

- Autónomo
  - Estado
  - Estudiante
  - Parado
  - Jubilado
- ↳ Ganancias anuales.
  - < 20.000
  - 20.000 – 50.000
  - > 50.000
- ↳ Horario trabajo.
  - Mañana
  - Tarde
  - Noche
  - Variable
- ↳ Puesto
  - Ejecutivo
  - Empleado
  - Conductor profesional.
  - Mecánico de automóviles.
  - Viajante
  - Reportero
  - Ama de casa.
- Conducción
  - ↳ Carnet de conducir.
    - ↳ Fecha
    - ↳ Categoría
  - ↳ Estilo conducción en carretera.
    - Despacio
    - Rápido
  - ↳ Estilo conducción en ciudad.
    - Despacio
    - Rápido



**Fig. 2.1:** *Características de un usuario*

Al realizar la lista de características de un usuario se discutió si se debía incluir o no el número de la matrícula del coche. Dado que la única función de la matrícula es de identificación y que no influye en los accidentes de un usuario se ha decidido no incluir esta característica.

#### **2.4. Historial de accidentes**

El historial de cada usuario recoge todos los sucesos, relacionados con la conducción, que le hayan ocurrido al usuario. Cada entrada del historial contiene el error que provocó el accidente, el accidente, las consecuencias del accidente, el lugar en que ocurrió (carretera, autopista o ciudad) y la fecha.

Tras presentar los componentes principales del historial (error, accidente y consecuencia), mostraremos un ejemplo de entrada en el historial.

**2.4.1. Error:** El error indica los motivos por los que se produce el accidente. Los accidentes están clasificados para que sea más fácil obtener información a partir de ellos. La clasificación usada es la siguiente:

- Cometidos por el propio conductor.
  - ↳ Velocidad
    - ↳ Frenada
    - ↳ Lenta
    - ↳ Alta
  - ↳ Accesorio
    - ↳ Móvil
    - ↳ Navegador
    - ↳ Cinturón
    - ↳ Cadenas
    - ↳ Luces
  - ↳ Estilo/Riesgo
    - ↳ Rotonda
    - ↳ Cruce
    - ↳ Carril de aceleración
    - ↳ Adelantamiento
    - ↳ Distancia
    - ↳ Carril
  - ↳ Otros
- Cometidos por otro conductor.
  - ↳ Velocidad
    - ↳ [...]
  - ↳ [...]
  - ↳ Otros
    - ↳ [...]

**Fig. 2.2:** Clasificación de los errores

La categoría *Estilo/Riesgo* contiene aquellos errores producidos por el usuario al realizar una maniobra de forma incorrecta, ya sea porque desconoce la forma correcta de hacerla o porque considera que no hay peligro al realizarla. Ejemplos de esta categoría son: “Saltarse un *STOP* en un cruce” o “No guardar la distancia de seguridad”.

Cada error es clasificado en una de esas clases. Por ejemplo, si el usuario se sale de la calzada porque iba escribiendo un SMS con el móvil, se guardará el error “El usuario escribe un SMS” que pertenece a la clase “Cometido por el usuario -> Accesorios -> Móvil”.

La lista de errores se halla en el acta del 30 de noviembre de 2009 en el apéndice C.

**2.4.2. Accidente:** Un accidente es el hecho que produce el daño a los ocupantes del coche o al coche. Por ejemplo: salirse de la calzada o recibir un golpe lateral. Al igual que los errores, los accidentes se encuentran clasificados. La lista de accidentes y su clasificación es la siguiente:

- Recibir golpe.
  - ↳ Frontal
  - ↳ Lateral
  - ↳ Por detrás.
- Dar golpe.
  - ↳ Frontal
  - ↳ Lateral
  - ↳ Por detrás.
- Salirse de la calzada.
  - ↳ Chocar con un elemento.
  - ↳ No chocar con ningún elemento.
- Atropellar
  - ↳ Una persona.
  - ↳ Un ciclista.
  - ↳ Un motorista.
  - ↳ Un animal.

**Fig. 2.3:** Clasificación de los accidentes

**2.4.3. Consecuencia:** Una consecuencia es el estado en el que queda un ocupante o un coche tras un accidente. La muerte del conductor, la necesidad de llevar el coche a un taller y la incapacidad del copiloto son ejemplos de consecuencias. Al igual que los errores y los accidentes, las consecuencias también se han clasificado. La lista de consecuencias es:

- Coche propio.
  - ↳ Conductor
    - Mortal
    - Incapacitado
    - Grave
    - Leve
    - Intacto
  - ↳ Copiloto
    - Mortal
    - [...]
    - Intacto
  - ↳ Pasajero
    - Mortal
    - [...]
    - Intacto
  - ↳ Coche
    - Siniestro total.
    - Taller
    - Chapa y pintura.
- Coche de un tercero.
  - ↳ Conductor
    - [...]
  - ↳ Copiloto
    - [...]
  - ↳ Pasajero
    - [...]
  - ↳ Coche
    - [...]

**Fig. 2.4:** *Clasificación de las consecuencias*

Inicialmente la consecuencia se indicaba implícitamente en el accidente pero posteriormente decidimos separar accidente y consecuencia<sup>2</sup>. Por ejemplo: en un primer momento teníamos “*Muerte del conductor por golpe frontal*” como un accidente, en la versión final tenemos “*Golpe frontal*” como accidente y “*Muerte del conductor*” como consecuencia. Hemos decidido separar accidente y consecuencia porque no hay una relación directa entre accidentes y consecuencias, un accidente puede producir cualquier consecuencia y una consecuencia determinada puede ser debida a cualquier accidente.

Ahora ya podemos presentar un ejemplo de entrada en el historial.

<p><b>Suceso</b></p> <p><b>Fecha:</b> 02 - Junio - 2010</p> <p><b>Lugar:</b> Carretera convencional</p> <p><b>Error:</b> Cometido por el usuario : Estilo/Riesgo : Adelantamiento : <i>Arriesgado</i>.</p> <p><b>Accidente:</b> Dar golpe : <i>Frontal</i>.</p> <p><b>Consecuencias:</b></p> <p>Coche propio : Conductor : Mortal</p> <p>Coche propio : Copiloto : Incapacitado</p> <p>Coche de un tercero : Conductor : Grave</p> <p>Coche propio : Coche : Siniestro total.</p> <p>Coche de un tercero : Coche : Siniestro total.</p> <p><b>Descripción:</b> <i>María va conduciendo y su esposo viaja como copiloto. María decide adelantar al coche que los precede ya que cree tener espacio y tiempo suficiente antes de cruzarse con el coche que viene en la otra dirección, pero no le da tiempo y se chocan los dos coches. Ambos coches quedan irreparables. María pierda la vida y su marido un brazo. El conductor del otro coche es internado en el hospital durante una semana, pero no le queda ninguna consecuencia.</i></p>
---

**Fig. 2.5:** Ejemplo de un suceso

<sup>2</sup> Ver actas del 30 de noviembre de 2009 y del 21 de diciembre de 2009 en el apéndice C.

## 2.5. Casos

Como ya hemos indicado en el apartado “*Funcionamiento interno del sistema*”, nuestra aplicación evalúa una serie de casos para obtener los posibles accidentes del usuario.

Cada caso tiene una serie de premisas y una serie de conclusiones. Cuando un usuario quiere conocer que accidentes puede sufrir en la carretera valoramos los casos, si el usuario cumple todas las premisas entonces podrá sufrir los accidentes detallados en las conclusiones del caso.

Al analizar el problema observamos que entran en juego tres conjuntos de datos:

- A. Características del usuario.
- B. Errores cometidos en la conducción.
- C. Accidentes y consecuencias.

Estos conjuntos pueden relacionarse de las dos siguientes formas:

1. A partir de A inferir B, y a partir del B inferido inferir C.

$$A \rightarrow B$$

$$B \rightarrow C$$

2. A partir de A y B inferir C.

$$A,B \rightarrow C$$

En el primer caso, a partir de las características de un usuario obtenemos los errores que dicho usuario comete. Posteriormente obtenemos los accidentes y las consecuencias producidos por dicho error.

En el segundo caso, a partir de las características de un usuario y su historial de errores, obtenemos los accidentes y las consecuencias que puede sufrir el usuario.

Hemos decidido trabajar con la segunda opción porque conlleva una relación más fuerte que la primera, ya que solo se realiza una inferencia. En el primer caso los accidentes y las consecuencias solo se determinan por los errores, pero en una situación real los accidentes y las consecuencias no solo dependen de los errores sino también de las características del usuario. Ante un error (por ejemplo, un coche que se salta un stop) un usuario joven reacciona de forma diferente a como lo hace un usuario mayor debido a la falta de reflejos del último. También influye en las consecuencias del accidente, un usuario joven se recupera más fácilmente de un accidente.

Por tanto, las premisas de nuestros casos son una serie de condiciones sobre las características del usuario y sobre errores cometidos por el usuario. Las conclusiones son una serie de accidentes con las consecuencias producidas, la probabilidad de que estos ocurran y el error por el que se producen. Hemos añadido el error y la probabilidad para dar más información al usuario sobre los accidentes sufridos.

En los dos siguientes casos podemos ver la diferencia en la gravedad del accidente debido a las diferentes características del usuario.

<b>Caso I</b>	
Premisas:	Tiene más de 60 años. Nunca ha tenido un accidente en la ciudad.
Conclusiones:	60% de posibilidad de recibir golpe lateral por entrar mal en una rotonda y con consecuencias leves para el conductor.

**Fig. 2.6:** *Ejemplo de un caso I*

<b>Caso II</b>	
Premisas:	Tiene menos de 30 años. Nunca ha tenido un accidente en la ciudad. Tiene el carnet BTP.
Conclusiones:	60% de posibilidad de recibir golpe lateral por entrar mal en una rotonda y no sufrir ninguna consecuencia el conductor.

**Fig. 2.7:** *Ejemplo de un caso II*

## CAPÍTULO 3

# DISEÑO DE LA BASE DE DATOS

En la fase anterior definimos todos los elementos que forman parte de la aplicación. En esta nueva fase hemos diseñado la base de datos.

El estudio de la base de datos lo vamos a dividir en dos partes. Por un lado vamos a estudiar la parte de la base de datos donde guardamos los datos de los usuarios y por otro lado la base de inferencia donde guardamos los casos.

### 3.1. Base de datos del usuario

En esta parte de la base de datos guardamos toda la información que tenemos sobre los usuarios, es decir, sus características personales y su historial de accidentes. Para ello hemos definido las siguientes entidades:

- Usuario: Esta entidad guarda los datos básicos de un usuario como son el NIF, el nombre o el sexo. Sus atributos son: *NIF*, *Nombre*, *Apellidos*, *Sexo* y *FechaNacimiento*.
- Profesional: Los datos profesionales son guardados en esta entidad. Los atributos son los siguientes: *Empresa*, *Ganancias*, *Puesto* y *HorarioTrabajo*. Es una entidad débil relacionada con la entidad *Usuario* a través de la relación *DatosProfesionales*.
- Conduccion: Esta entidad contiene los datos sobre los hábitos de conducción del usuario. Sus atributos son: *FechaCarnet*, *CategoriaCarnet*, *EstiloConduccionCiudad* y *EstiloConduccionCarretera*. *Conduccion* es una entidad débil relacionada con la entidad *Usuario* a través de la relación *DatosConduccion*. Los

datos sobre el coche (o coches) del usuario están guardados en otra entidad.

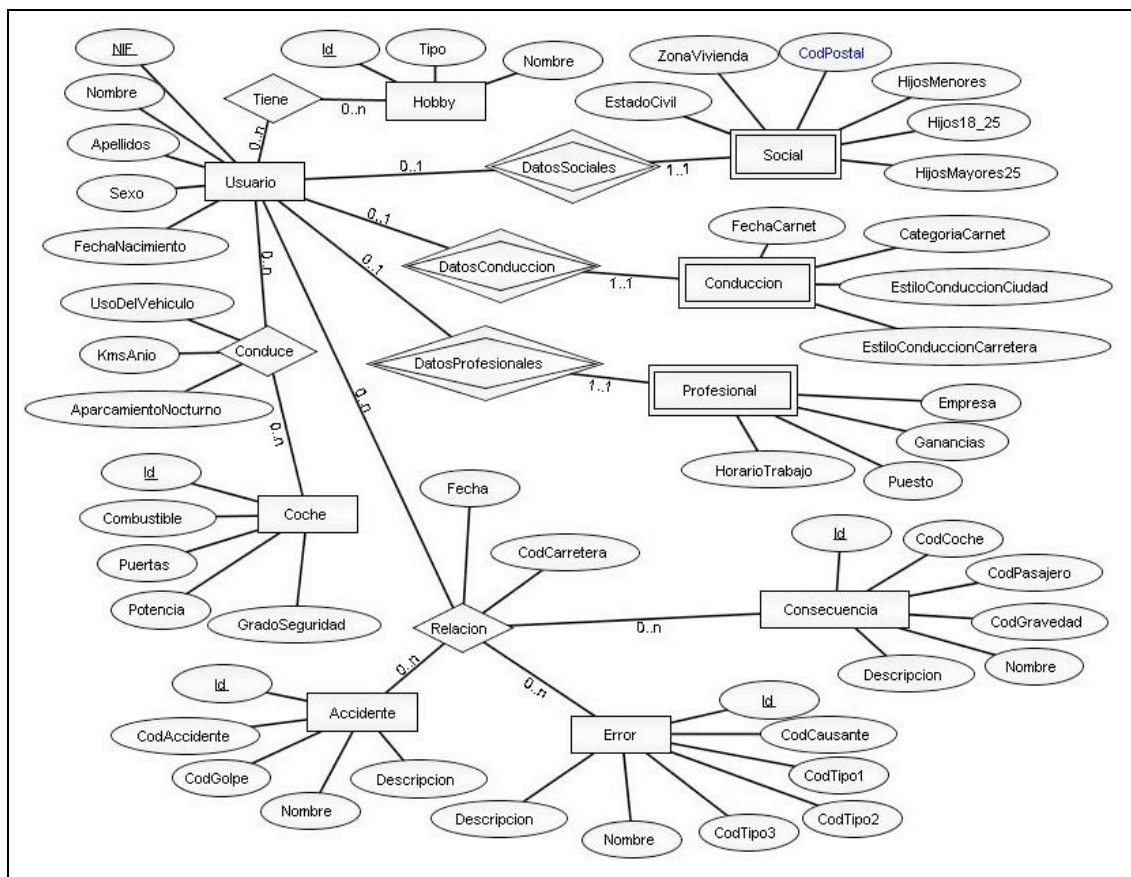
- Coche: Información de cada coche. Esta entidad se relaciona con los usuarios a través de la relación *Conduce*. Los atributos de la entidad *Coche* son *Id*, *Combustible*, *Puertas*, *Potencia* y *GradoSeguridad*.
- Social: En esta entidad guardamos los datos sociales de cada usuario, estos datos son: *EstadoCivil*, *ZonaVivienda*, *CodPostal*, *HijosMenores*, *Hijos18\_25* e *HijosMayores25*. Es una entidad débil relacionada con la entidad *Usuario* a través de la relación *DatosSociales*. Los datos sobre los hobbies del usuario que pertenecen a las características sociales están guardados en otra entidad.
- Hobby: Contiene la información sobre un hobby. La relación *Tiene* relaciona esta entidad con los usuarios. Sus atributos son: *Id*, *Tipo* y *Nombre*.
- Accidente: En esta entidad se guardan los diferentes accidentes que puede sufrir un usuario. La clasificación que realizamos de los accidentes se guarda a través de códigos. Sus atributos son: *Id*, *CodAccidente*, *CodGolpe*, *Nombre* y *Descripcion*.
- Error: Esta entidad contiene los errores. La clasificación de un error se guarda a través de códigos. Los atributos de esta entidad son: *Id*, *CodCausante*, *CodTipo1*, *CodTipo2*, *CodTipo3*, *Nombre* y *Descripcion*.
- Consecuencia: Las consecuencias que tienen los accidentes se guardan en esta entidad. La clasificación de cada consecuencia se guarda a través de códigos. Los atributos son los siguientes: *Id*, *CodCoche*, *CodPasajero*, *CodGravedad*, *Nombre* y *Descripcion*.

Las relaciones de la base de datos son las siguientes:

- DatosProfesionales: Esta relación une la entidad *Usuario* con la entidad débil *Profesional*.
- DatosConduccion: Esta relación une la entidad *Usuario* con la entidad débil *Conduccion*.

- DatosSociales: Esta relación une la entidad *Usuario* con la entidad débil *Social*.
- Conduce: Es una relación “muchos a muchos” entre las entidades *Usuario* y *Coches*. Además contiene tres atributos: *KmsAnio*, *UsoDelVehiculo* y *AparcamientoNocturno*.
- Tiene: Esta relación “muchos a muchos” relaciona a la entidad *Usuario* con la entidad *Hobby*.
- Suceso: Esta relación se corresponde con el historial de accidentes del usuario. Por tanto, *Suceso* relaciona las entidades *Usuario*, *Error*, *Accidente* y *Consecuencia*. Todas las entidades participan con cardinalidad 0..n. También tiene 2 atributos: *Fecha* y *CodCarretera*.

El diagrama E-R resultante es el siguiente:



**Fig. 3.1:** Diagrama E-R de la base de datos del usuario

A nuestro diseño hemos tenido que eliminarle las entidades débiles. Para ello simplemente hemos incluido la clave primaria de la entidad *Usuario* (es decir, el atributo *NIF*) en las entidades débiles (*Conduccion*, *Profesional* y *Social*).

Por tanto, las tablas que forman parte de nuestra base de datos son:

- Usuario(NIF, Nombre, Apellidos, Sexo, FechaNacimiento)
- Profesional(NIF, Empresa, Ganancias, Puesto, HorarioTrabajo)
- Social(NIF, NivelEstudios, EstadoCivil, ZonaVivienda, CodPostal, HijosMenores, Hijos18\_25, HijosMayores25)
- Conduccion(NIF, FechaCarnet, CategoriaCarnet, EstiloConduccionCiudad, EstiloConduccionCarretera)
- Coche(Id, Combustible, Puertas, Potencia, GradoSeguridad)
- Conduce(NIF, Id, KmsAnio, UsoDelVehiculo, AparcamientoNocturno)
- Hobby(Id, Tipo, Nombre)
- Tiene( NIF, Id)
- Error(Id, CodCausante, CodTipo1, CodTipo2, CodTipo3, Nombre, Descripcion)
- Accidente(Id, CodAccidente, CodGolpe, Nombre, Descripcion)
- Consecuencia(Id, CodCoche, CodPasajero, CodGravedad, Nombre, Descripcion)
- Suceso(NIF, Id\_Error, Id\_Accidente, Id\_Consecuencia, Fecha, CodCarretera)

A partir de las tablas obtenidas hemos creado las sentencias SQL necesarias para la creación de la base de datos. Las sentencias se pueden consultar en el apéndice A.

### 3.2. Base de inferencia

Al diseñar la base de datos se barajaron tres opciones<sup>3</sup> para representar los casos:

1. Bloques if-then-else.
2. Una tabla donde cada fila representara las premisas y la conclusión.
3. Dos tablas, una para premisas y otra para conclusiones.

La primera opción aunque tiene una gran versatilidad, ya que cada caso puede tener un número diferente de premisas y conclusiones, es rechazada porque no se considera una solución elegante.

La segunda opción tiene un inconveniente importante, el número de premisas y de conclusiones es el mismo para todos los casos. También se rechaza la segunda opción.

La tercera opción es la elegida, ya que proporciona libertad en el número de premisas y conclusiones y facilidad para añadir, modificar o eliminar casos.

El siguiente paso fue definir las entidades *Premisa*, *Conclusión* y *Caso*:

- Premisa: Esta entidad contiene las premisas de los casos y está formada por los siguientes atributos:
  - Id: La clave primaria, su objetivo es diferenciar las premisas.
  - TipoPremisa: Nos dice si la premisa es positiva o negativa, es decir, si se tiene que cumplir o no.
  - Tabla: La tabla a la que se refiere la premisa.
  - Campo: El campo de la tabla al que se refiere la premisa.
  - ValorInicial: Si el atributo *ValorFinal* es NULL, *ValorInicial* contiene el valor que debe cumplir el campo. Si el atributo *ValorFinal* no es NULL, *ValorInicial-ValorFinal* es el intervalo cerrado en el que se tiene que hallar el valor del campo.

---

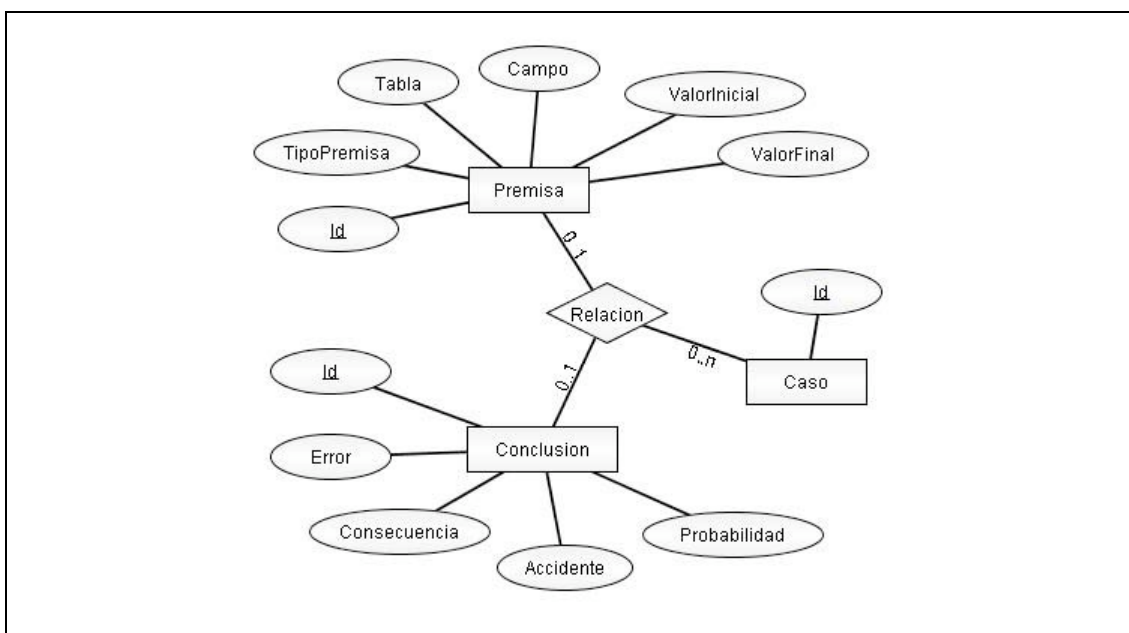
<sup>3</sup> Ver acta del 25 de enero de 2010 en el apéndice C.

- Conclusion: En esta entidad guardamos las conclusiones de los casos y está formada por los siguientes atributos:
  - Id: La clave primaria.
  - Error: Id del error que produce el accidente. Es una clave externa al atributo *Id* de la entidad *Error*.
  - Accidente: Id del accidente. Es una clave externa al atributo *Id* de la entidad *Accidente*.
  - Consecuencia: Id de la consecuencia que produce el accidente. Es una clave externa al atributo *Id* de la entidad *Consecuencia*.
  - Probabilidad: La probabilidad de que ocurra este accidente.
- Caso: Entidad que representa un caso. Su único atributo es el *Id*.

También tenemos una relación que une las tres entidades:

- Relacion: Esta relación une las entidades *Premisa*, *Conclusion* y *Caso*. *Caso* participa con cardinalidad 0..n (un caso tiene muchas premisas y muchas conclusiones), *Premisa* participa con cardinalidad 1 (una premisa solo pertenece a un caso) y *Conclusion* participa con cardinalidad 1 (una conclusión solo pertenece a un caso).

El esquema E-R resultante es el siguiente:

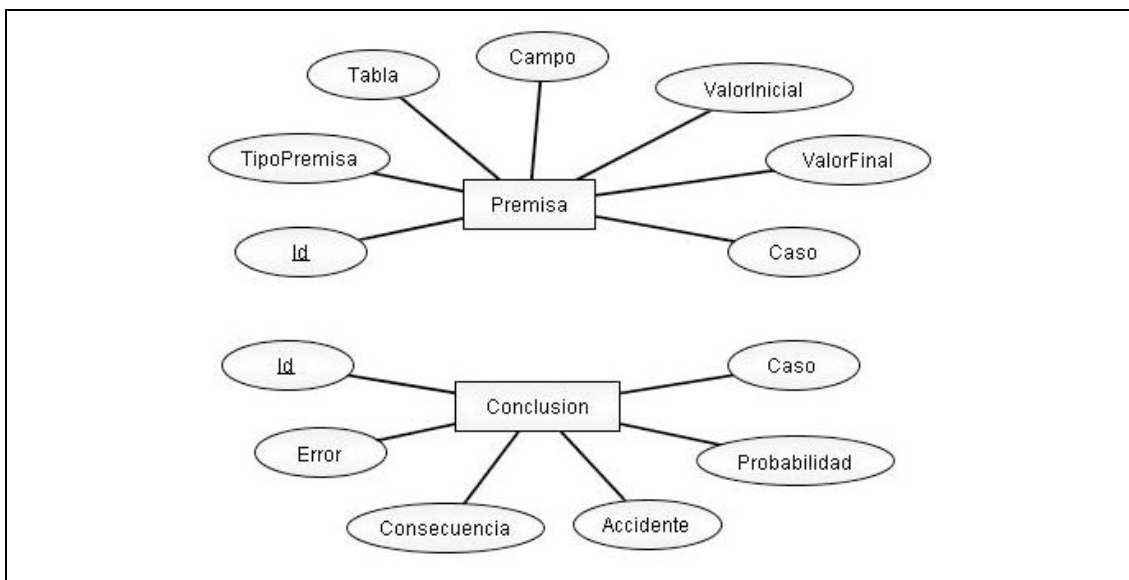


**Fig. 3.2:** Esquema E-R de la base de inferencia I

Como las entidades *Premisa* y *Conclusion* participan con cardinalidad 1 en la relación *Relacion* podemos eliminar dicha relación si incluimos la clave primaria de *Caso* en ambas entidades. Una vez hemos eliminado la relación y como la entidad *Caso* solo tiene el atributo *Id* podemos eliminar dicha entidad. Las tablas resultantes son:

- Premisa(Id, TipoPremisa, Tabla, Campo, ValorInicial, ValorFinal, Caso)
- Conclusion(Id, Error, Accidente, Consecuencia, Probabilidad, Caso)

Y el diagrama entidad-relación es:



**Fig. 3.3:** Esquema E-R de la base de inferencia II

A partir de estas tablas creamos las sentencias SQL necesarias para la creación de la base de datos. Las sentencias se pueden consultar en el apéndice A.

## CAPÍTULO 4

# CREACIÓN DE UN PROTOTIPO

El siguiente paso fue la implementación de dos aplicaciones. La primera aplicación sirve para insertar en la base de datos todos los datos de una forma sencilla sin tener que recurrir a sentencias SQL. La segunda aplicación es el objetivo de este proyecto.

La aplicación de inicialización y del prototipo se ha realizado en Java y se ha utilizado MySQL para la base de datos.

### 4.1. Aplicación para inicializar la base de datos

Para probar el funcionamiento de la base de datos y para facilitar la introducción de datos en ella, hemos creado una aplicación que nos ayude. Esta aplicación consiste en una serie de formularios para insertar datos y una serie de ventanas para consultar los datos introducidos. Por tanto por cada entidad y relación en la base de datos existe un formulario de inserción y una ventana de consulta.

**4.1.1. Inserción de un usuario, coche, hobby, accidente, error o consecuencia:** Los formularios y ventanas para las entidades usuarios, coches, hobbies, accidentes, errores o consecuencias son muy parecidas. Por ello solo vamos a explicar como insertar o consultar un coche, ya que los pasos para insertar cualquiera de las otras entidades son los mismos.

En primer lugar, elegimos la pestaña *Coche* de la barra de pestañas situada a la izquierda. Una vez seleccionada, en la barra superior nos salen dos pestañas con las opciones posibles: *Insertar coche* o *Ver coches*.

La pestaña *Ver coches* muestra una tabla con todos los coches disponibles en la base de datos. La pestaña *Insertar coche* muestra un formulario con todos los atributos de un coche. Una vez que hayamos introducido los datos que nos interesen pulsamos el botón *INSERTAR* y el coche será añadido a la base de datos. Podemos comprobar que ha sido insertado en la pestaña *Ver coches*.

**Fig. 4.1:** Formulario de inserción de un coche

Id	Combustible	Puertas	Potencia	GradoSeguridad
109	Gasolina	3	320	4
110	Gasolina	3	320	5
111	Gasolina	4	320	1
112	Gasolina	4	320	2
113	Gasolina	4	320	3
114	Gasolina	4	320	4
115	Gasolina	4	320	5
116	Gasolina	5	320	1
117	Gasolina	5	320	2
118	Gasolina	5	320	3
119	Gasolina	5	320	4
120	Gasolina	5	320	5
121	Diesel	2	75	1
122	Diesel	2	75	2
123	Diesel	2	75	3
124	Diesel	2	75	4
125	Diesel	2	75	5
126	Diesel	3	75	1
127	Diesel	3	75	2
128	Diesel	3	75	3
129	Diesel	3	75	4
130	Diesel	3	75	5
131	Diesel	4	75	1
132	Diesel	4	75	2
133	Diesel	4	75	3
134	Diesel	4	75	4
135	Diesel	4	75	5
136	Diesel	5	75	1
137	Diesel	5	75	2
138	Diesel	5	75	3
139	Diesel	5	75	4
140	Diesel	5	75	5
141	Diesel	2	160	1
142	Diesel	2	160	2
143	Diesel	2	160	3
144	Diesel	2	160	4

**Fig. 4.2:** Ventana para consultar los coches introducidos

**4.1.2 Inserción de las relaciones Conduce, Tiene y Suceso:** Ya hemos explicado los formularios para insertar usuarios, coches, hobbies, errores, accidentes o consecuencias. Los siguientes formularios que vamos a explicar son los que usamos para insertar las relaciones *Conduce* (entre un usuario y un coche), *Tiene* (entre un usuario y un hobby) y *Suceso* (entre un usuario, un error, un accidente y una consecuencia).

Para ir a los formulario debemos seleccionar la pestaña *Relaciones* de la barra lateral izquierda.

The screenshot shows a web application window titled "Asistencia guiada a la conducción en función de errores históricos previos - Prueba". The main content area is under the "Relaciones" tab. At the top, there are input fields for "NIF:" (22222222B), "Nombre:" (María), "Apellidos:" (del Valle), "Sexo:" (F), and "Fecha de nacimiento:" (1989-07-12). Below these are navigation buttons: "Ver usuario", "Ver relaciones", "Insertar coche", "Insertar hobby", "Insertar suceso", and "Ver casos".

There are three data tables displayed:

NIF	Empresa	Ganancias	Puesto	HorarioTrabajo
22222222B	Estudiante	-	-	null

NIF	EstadoCivil	ZonaVivienda	CodPostal	HijosMenores	Hijos18_25	HijosMayores25
22222222B	Soltero	Zona residencial	28040	-	-	-

NIF	FechaCarnet	CategoriaCarnet	EstiloConduccionCiudad	EstiloConduccionCarretera
22222222B	2009-09-12	B	Despacio	Despacio

**Fig. 4.3:** Pestaña *Relaciones* – *Ver Usuario*

Los formularios están pensados para insertar todas las relaciones de un usuario de la forma más sencilla posible. Por ello, solo tendremos que seleccionar el usuario una vez en todo el proceso. El usuario se selecciona en el menú desplegable situado en la parte superior de la pantalla. Una vez que hemos elegido a un usuario podemos ver sus datos, las relaciones que ya existen sobre él o insertar relaciones nuevas según la pestaña que elijamos.

En la pestaña *Ver usuario* (Fig. 4.3) se muestran todos los datos del usuario y en la pestaña *Ver relaciones* (Fig. 4.4) todas sus relaciones.

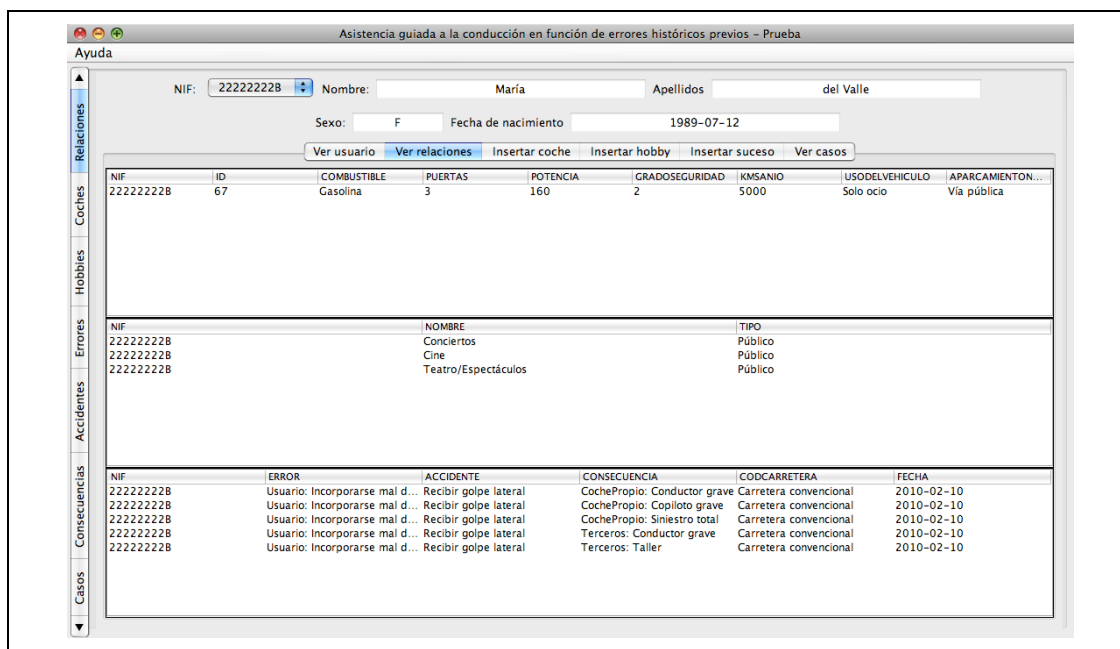


Fig. 4.4: Relaciones de un usuario

Para insertar las relaciones disponemos de las pestañas *Insertar coche*, *Insertar hobby* e *Insertar suceso*. Cada una de estas pestañas contiene un formulario donde se pueden elegir los otros miembros de la relación e insertar los atributos de la relación. Una vez que hayamos rellenado el formulario solo tenemos que pulsar el botón *INSERTAR* y comprobar en la pestaña *Ver relaciones* que se ha insertado correctamente.

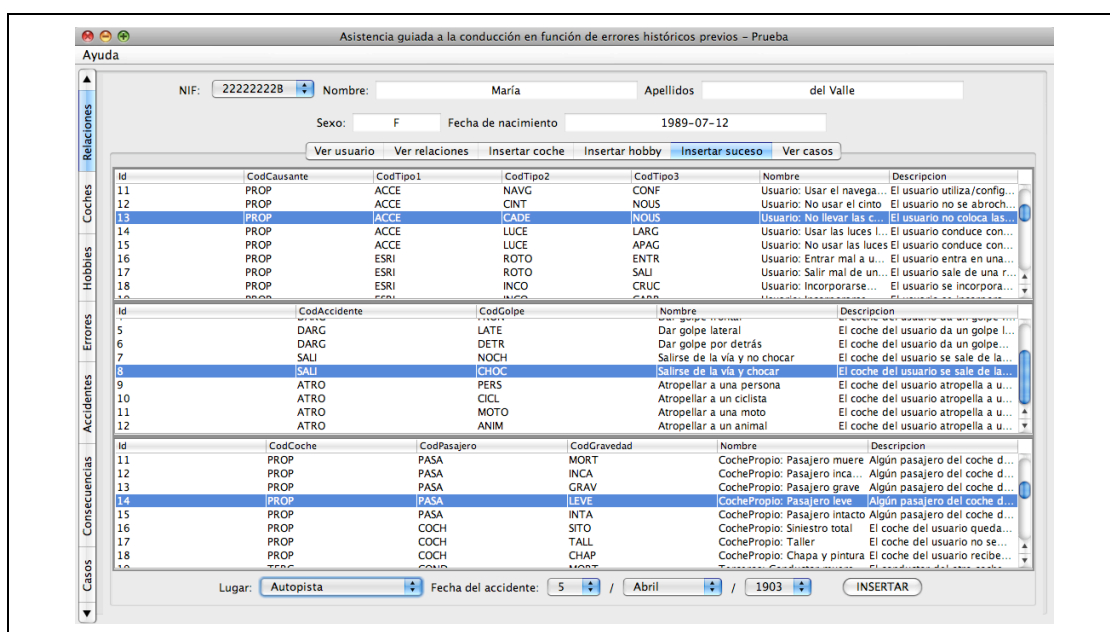


Fig. 4.5: Inserción de un suceso en el historial de un usuario

**4.1.3. Inserción de un caso:** Al igual que anteriormente, el primer paso es elegir en el panel de la izquierda una pestaña, en este caso, la pestaña *Casos*. En la pantalla obtendremos dos nuevas pestañas *Insertar caso* y *Ver casos*.

Un caso está compuesto por premisas y conclusiones, por tanto, en el formulario de la pestaña *Insertar caso* tenemos dos partes, una para añadir premisas y otra para añadir conclusiones. Cuando hayamos añadido todas las premisas y conclusiones del caso lo insertaremos pulsando el botón *INSERTAR CASO*.

Las premisas se añaden en la parte superior del formulario. Los valores tipo premisa, tabla y campo se seleccionan en los menús desplegables. Si queremos que el campo elegido solo tenga un valor determinado, rellenamos el campo *Valor inicial* y dejamos *Valor final* vacío. Si queremos que el campo elegido se encuentre entre dos valores, rellenamos los campos *Valor inicial* y *Valor final*.

The image shows two examples of the 'Insertar caso' form. Each example has a header with three dropdown menus: 'Tipo premisa' (set to 'PP'), 'Tabla' (set to 'Coche'), and 'Campo' (set to 'Grado seguridad'). Below the header are two input fields: 'Valor inicial' and 'Valor final', followed by an 'Añadir premisa' button.

**Example 1:** 'Valor inicial' is set to '4' and 'Valor final' is empty. Below the form, the text reads: *El grado de seguridad del coche tiene que ser 4.*

**Example 2:** 'Valor inicial' is set to '2' and 'Valor final' is set to '4'. Below the form, the text reads: *El grado de seguridad del coche se encuentra entre los valores 2 y 4.*

**Fig. 4.6:** Como rellenar los campos *Valor inicial* y *Valor final*

Según el campo elegido (Usuario-Sexo, Coche-Puertas...) las premisas deben cumplir unas características u otras. A continuación se detalla como rellenar los campos *Valor inicial* y *Valor final*:

- Tabla USUARIO
  - Campo SEXO
    - Solo se tiene en cuenta el valor inicial.
  - Campo FECHA NACIMIENTO
    - Se tienen en cuenta el valor inicial y el final.
    - Formato de fechas YYYY-MM-DD donde - puede ser cualquier separador (-,/, \$...).
    - Ejemplo: 1986-09-24

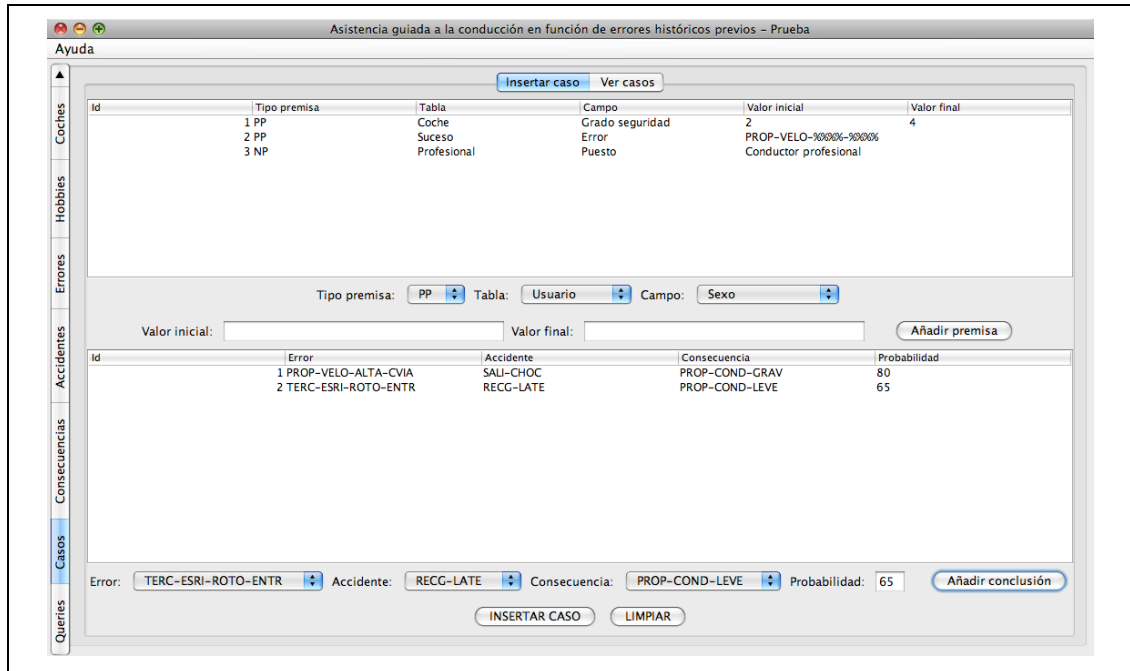
- Tabla CONDUCCION
  - Campo FECHA CARNET
    - Se tienen en cuenta el valor inicial y el final.
    - Formato de fechas YYYY-MM-DD donde - puede ser cualquier separador (-,/, \$...).
  - Campo CATEGORÍA CARNET
  - Campo ESTILO CONDUCCIÓN CIUDAD
  - Campo ESTILO CONDUCCIÓN CARRETERA
    - Solo se tiene en cuenta el valor inicial.
- Tabla PROFESIONAL
  - Campo EMPRESA
  - Campo PUESTO
  - Campo HORARIO TRABAJO
    - Solo se tiene en cuenta el valor inicial.
  - Campo GANANCIAS
    - Se tienen en cuenta el valor inicial y el final.
- Tabla SOCIALES
  - Campo ESTADO CIVIL
  - Campo ZONA VIVIENDA
    - Solo se tiene en cuenta el valor inicial.
  - Campo HIJOS MENORES
  - Campo HIJOS 18\_25
  - Campo HIJOS MAYORES 25
    - Se tienen en cuenta el valor inicial y el final.
  - Campo CODIGO POSTAL
    - Se tiene en cuenta el valor inicial y el final.
    - El código postal, el valor inicial y el valor final tienen que tener la misma longitud.
    - Los % en el valor inicial o el final, no se comparan:  
Ejemplo: valorInicial = 34%%%, devuelve *true* para todos

los códigos postales que empiecen por 34 y tengan 5 caracteres (34100, 34SDF, 34-DF...).

- Tabla COCHE
  - Campo COMBUSTIBLE
  - Campo APARCAMIENTO NOCTURNO
  - Campo USO DEL VEHICULO
    - Solo se tiene en cuenta el valor inicial.
  - Campo PUERTAS
  - Campo POTENCIA
  - Campo KMS AÑO
  - Campo GRADO SEGURIDAD
    - Se tienen en cuenta el valor inicial y el valor final.
- Tabla HOBBY
  - Campo NOMBRE
  - Campo TIPO
    - Solo se tiene en cuenta el valor inicial.
- Tabla HISTORIAL
  - Campo CARRETERA
    - Solo se tiene en cuenta el valor inicial.
  - Campo FECHA
    - Se tienen en cuenta el valor inicial y el final.
    - Formato de fechas YYYY-MM-DD donde - puede ser cualquier separador (-,/, \$...).
  - Campo ERROR
  - Campo ACCIDENTE
  - Campo CONSECUENCIA
    - Se tiene en cuenta el valor inicial y el final.
    - El valor, el valor inicial y el valor final tienen que tener la misma longitud.
    - Los % en el valor inicial o el final, no se comparan:
    - Ejemplo: Consecuencia = %%%%-%%%-MORT, representa todas las consecuencias mortales,

independientemente del usuario que las hayas sufrido (PROP-COND-MORT, TERC-PASA-MORT...)

Para añadir una conclusión simplemente seleccionamos en los menús desplegables el error, el accidente y la consecuencia, indicamos la probabilidad con la que puede ocurrir y pulsamos el botón *Añadir conclusión*.



**Fig. 4.7:** Un caso con 3 premisas y 2 conclusiones

En la pestaña *Ver casos* podemos comprobar que se ha insertado el caso correctamente, así como consultar el resto de casos disponibles en la base de datos.

## 4.2. Prototipo.

Tras probar la base de datos e inicializarla con la aplicación de inicialización, hemos desarrollado el prototipo.

El prototipo permite insertar usuarios (al igual que la aplicación de inicialización) y consultar los posibles accidentes de un usuario.

**4.2.1. Inserción de un usuario:** Para desarrollar el prototipo nos hemos servido de la aplicación de inicialización, por tanto, esta función se realiza de forma similar en ambos. Por tanto, seguiremos las instrucciones detalladas en el apartado *4.1.1. Inserción de un usuario, coche, hobby, accidente, error o consecuencia.*

**Fig. 4.8:** *Inserción de un usuario*

Para insertar las relaciones de un usuario (coches que conduce, hobbies que tiene o accidentes que ha sufrido) seguiremos las instrucciones detalladas en el apartado *4.1.2. Inserción de las relaciones Conduce, Tiene y Suceso.* El único cambio es que ahora la pestaña se llama *Gestión usuario* en lugar de *Relaciones*.

NIF	Empresa	Ganancias	Puesto	HorarioTrabajo
222222228	Estudiante	-	-	null

NIF	EstadoCivil	ZonaVivienda	CodPostal	HijosMenores	Hijos18_25	HijosMayores25
222222228	Soltero	Zona residencial	28040	-	-	-

NIF	FechaCarnet	CategoriaCarnet	EstiloConduccionCiudad	EstiloConduccionCarretera
222222228	2009-09-12	B	Despacio	Despacio

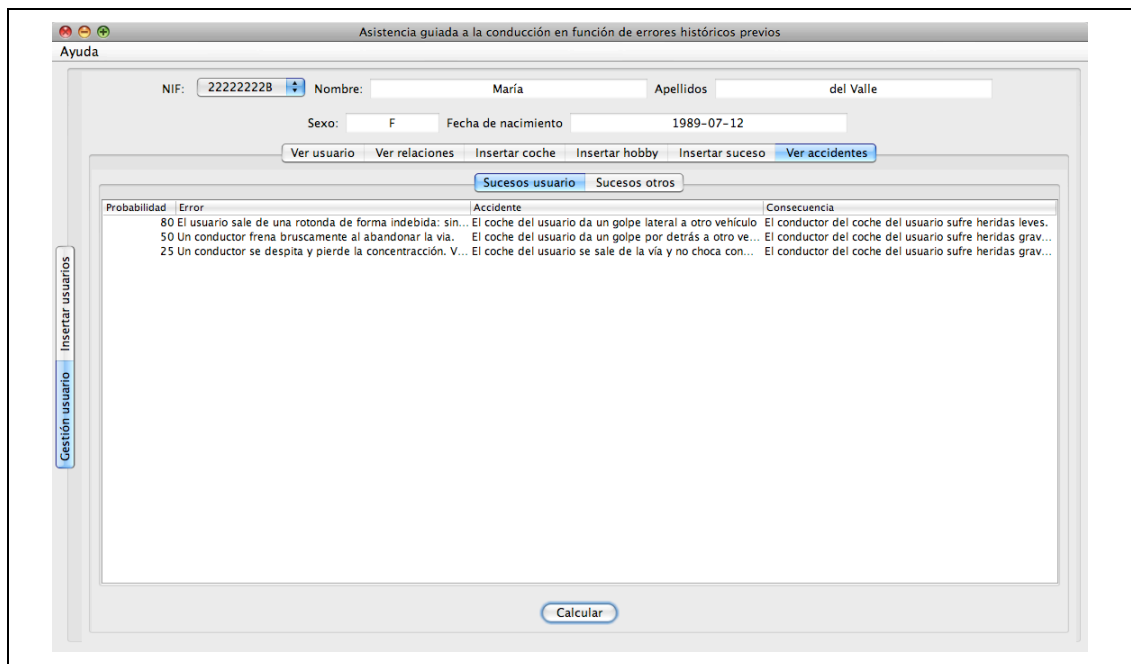
**Fig. 4.9:** *Pestaña Gestión usuario*

**4.2.2 Consultar los posibles accidentes:** Para consultar los posibles accidentes que puede sufrir un usuario acudiríamos a la pestaña *Gestión usuario*. En el menú desplegable de la parte superior podemos elegir el usuario a través del NIF, para comprobar que el NIF elegido corresponde al usuario que queremos se nos muestran algunos de sus datos personales: nombre, apellidos, sexo y fecha de nacimiento.

De las pestañas disponibles en esta vista, elegiremos la pestaña *Ver accidentes*. Cuando hayamos pulsado el botón *Calcular* se mostrarán los posibles accidentes que puede sufrir el usuario. Los accidentes aparecen divididos en dos grupos, según el causante del accidente sea el propio usuario u otro conductor (pestañas *Accidentes usuario* y *Accidentes otro* respectivamente).

En cada grupo, los accidentes aparecen ordenados por la probabilidad con la que estos pueden ocurrir. Además de la consecuencia del accidente, también se muestran el error que produjo el accidente y el accidente.

Para consultar los posibles accidentes de otro usuario basta con cambiar el NIF en el menú desplegable de la parte superior.



**Fig. 4.10:** Posibles accidentes de un usuario

## CAPÍTULO 5

# CREACIÓN DE LA MEMORIA

Una vez que habíamos terminado el proyecto, empezamos a redactar esta memoria. En primer lugar juntamos toda la información sobre el proyecto de la que disponíamos: actas realizadas a lo largo del curso, javadoc de las aplicaciones, libros consultados durante la realización del proyecto, notas tomadas a mano...

Para la creación de la memoria decidimos seguir las fases usadas durante el curso porque nos permitían mostrar de una forma sencilla el trabajo realizado y las decisiones tomadas.

Además de la memoria en papel, se ha creado un CD que contiene la memoria en formato digital, el código de las aplicaciones y los ejecutables.

### 5.1. Contenido del CD

El CD contiene una copia de la memoria en formato electrónico, el código de las dos aplicaciones realizadas y los ejecutables.

El contenido del CD esta dividido en tres carpetas:

- Memoria: Contiene una copia de la memoria en formato pdf.
- Código: Contiene el código de las aplicaciones y el javadoc.
- Ejecutables: Contiene los ejecutables y un documento<sup>4</sup> con instrucciones sobre como llevar a cabo la ejecución.

---

<sup>4</sup> El documento se ha incluido en el apéndice B.

## CONCLUSIONES

Durante el desarrollo del proyecto hemos conseguido cumplir los objetivos que nos habíamos propuesto al comenzar este proyecto: diseñar un prototipo de una aplicación que permita a un usuario conocer que accidentes puede sufrir en la carretera.

Para el diseño e implementación del prototipo hemos utilizado los conocimientos, tanto teóricos como prácticos, adquiridos a lo largo de nuestra formación académica.

Nuestro prototipo acepta diversas ampliaciones. Entre ellas proponemos la creación de un algoritmo que genere los casos a partir de los datos disponibles en la base de datos.

## APÉNDICE A

# SENTENCIAS SQL

Sentencias SQL necesarias para crear la base de datos tal y como está explicada en el punto 4 de la memoria.

```
DROP TABLE IF EXISTS Error;
```

```
CREATE TABLE Error( Id INTEGER NOT NULL AUTO_INCREMENT, CodCausante CHAR(4) NOT NULL, CodTipo1 CHAR(4) NOT NULL, CodTipo2 CHAR(4) NOT NULL, CodTipo3 CHAR(4) NOT NULL, Nombre VARCHAR(50), Descripcion TEXT, CONSTRAINT error_pk PRIMARY KEY (id));
```

```
DROP TABLE IF EXISTS Consecuencia;
```

```
CREATE TABLE Consecuencia( Id INTEGER NOT NULL AUTO_INCREMENT, CodCoche CHAR(4) NOT NULL, CodPasajero CHAR(4) NOT NULL, CodGravedad CHAR(4) NOT NULL, Nombre VARCHAR(50), Descripcion TEXT, CONSTRAINT consecuencia_pk PRIMARY KEY (id));
```

```
DROP TABLE IF EXISTS Accidente;
```

```
CREATE TABLE Accidente( Id INTEGER NOT NULL AUTO_INCREMENT, CodAccidente CHAR(4) NOT NULL, CodGolpe CHAR(4) NOT NULL, Nombre VARCHAR(50), Descripcion TEXT, CONSTRAINT accidente_pk PRIMARY KEY (id));
```

```
DROP TABLE IF EXISTS Suceso;
```

```
CREATE TABLE Suceso( NIF CHAR(9) NOT NULL, Id_Error INTEGER NOT NULL, Id_Accidente INTEGER NOT NULL, Id_Consecuencia INTEGER NOT NULL, Fecha DATE NOT
```

```
NULL, CodCarretera VARCHAR(25) NOT NULL, CONSTRAINT suceso_pk PRIMARY KEY
(NIF, Id_Error, Id_Accidente, Id_Consecuencia, Fecha, CodCarretera));
```

```
DROP TABLE IF EXISTS Hobby;
```

```
CREATE TABLE Hobby( Id INTEGER NOT NULL AUTO_INCREMENT, Tipo VARCHAR(25),
Nombre VARCHAR(25), CONSTRAINT hobby_pk PRIMARY KEY (id));
```

```
DROP TABLE IF EXISTS Coche;
```

```
CREATE TABLE Coche( Id INTEGER NOT NULL AUTO_INCREMENT, Combustible
VARCHAR(25), Puertas INTEGER, Potencia INTEGER, GradoSeguridad INTEGER,
CONSTRAINT coche_pk PRIMARY KEY (id));
```

```
ALTER TABLE Coche ADD CONSTRAINT coche_seguridad CHECK (GradoSeguridad >= 0 AND
GradoSeguridad <= 5);
```

```
DROP TABLE IF EXISTS Social;
```

```
CREATE TABLE Social( NIF CHAR(9) NOT NULL, EstadoCivil VARCHAR(25), ZonaVivienda
VARCHAR(25), CodPostal CHAR(5), HijosMenores INTEGER, Hijos18_25 INTEGER,
HijosMayores25 INTEGER, CONSTRAINT social_pk PRIMARY KEY (NIF));
```

```
DROP TABLE IF EXISTS Profesional;
```

```
CREATE TABLE Profesional( NIF CHAR(9) NOT NULL, Empresa VARCHAR(25), Ganancias
INTEGER, Puesto VARCHAR(25), HorarioTrabajo VARCHAR(25), CONSTRAINT profesional_pk
PRIMARY KEY (NIF));
```

```
DROP TABLE IF EXISTS Conduccion;
```

```
CREATE TABLE Conduccion( NIF CHAR(9) NOT NULL, FechaCarnet DATE, CategoriaCarnet
VARCHAR(3), EstiloConduccionCiudad VARCHAR(25), EstiloConduccionCarretera
VARCHAR(25), CONSTRAINT conduccion_pk PRIMARY KEY (NIF));
```

DROP TABLE IF EXISTS Usuario;

CREATE TABLE Usuario( NIF CHAR(9) NOT NULL, Nombre VARCHAR(25), Apellidos VARCHAR(25), Sexo CHAR(1), FechaNacimiento DATE, CONSTRAINT usuario\_pk PRIMARY KEY (NIF));

ALTER TABLE Usuario ADD CONSTRAINT usuario\_sex CHECK (Sexo='V' OR Sexo='F');

DROP TABLE IF EXISTS Tiene;

CREATE TABLE Tiene( NIF CHAR(9) NOT NULL, Id INTEGER NOT NULL, CONSTRAINT tiene\_pk PRIMARY KEY (NIF,Id));

DROP TABLE IF EXISTS Conduce;

CREATE TABLE Conduce( NIF CHAR(9) NOT NULL, Id INTEGER NOT NULL, KmsAnio INTEGER, UsoDelVehiculo VARCHAR(35), AparcamientoNocturno VARCHAR(25), CONSTRAINT conduce\_pk PRIMARY KEY (NIF,Id));

DROP TABLE IF EXISTS Premisa;

CREATE TABLE Premisa( Id INTEGER NOT NULL AUTO\_INCREMENT, TipoPremisa CHAR(2) NOT NULL, Tabla VARCHAR(15) NOT NULL, Campo VARCHAR(30) NOT NULL, ValorInicial VARCHAR(25) NOT NULL, ValorFinal VARCHAR(25), Caso INTEGER NOT NULL, CONSTRAINT premisa\_pk PRIMARY KEY (Id));

DROP TABLE IF EXISTS Conclusion;

CREATE TABLE Conclusion( Id INTEGER NOT NULL AUTO\_INCREMENT, Error INTEGER, Accidente INTEGER, Consecuencia INTEGER NOT NULL, Probabilidad INTEGER, Caso INTEGER NOT NULL, CONSTRAINT conclusion\_pk PRIMARY KEY (Id));

ALTER TABLE Suceso ADD FOREIGN KEY (NIF) REFERENCES Usuario (NIF) ON DELETE CASCADE;

ALTER TABLE Suceso ADD FOREIGN KEY (Id\_Error) REFERENCES Error (Id) ON DELETE CASCADE;

ALTER TABLE Suceso ADD FOREIGN KEY (Id\_Accidente) REFERENCES Accidente (Id) ON DELETE CASCADE;

ALTER TABLE Suceso ADD FOREIGN KEY (Id\_Consecuencia) REFERENCES Consecuencia (Id) ON DELETE CASCADE;

ALTER TABLE Social ADD FOREIGN KEY (NIF) REFERENCES Usuario (NIF) ON DELETE CASCADE;

ALTER TABLE Profesional ADD FOREIGN KEY (NIF) REFERENCES Usuario (NIF) ON DELETE CASCADE;

ALTER TABLE Conduccion ADD FOREIGN KEY (NIF) REFERENCES Usuario (NIF) ON DELETE CASCADE;

ALTER TABLE Tiene ADD FOREIGN KEY (NIF) REFERENCES Usuario (NIF) ON DELETE CASCADE;

ALTER TABLE Tiene ADD FOREIGN KEY (Id) REFERENCES Hobby (Id) ON DELETE CASCADE;

ALTER TABLE Conduce ADD FOREIGN KEY (NIF) REFERENCES Usuario (NIF) ON DELETE CASCADE;

ALTER TABLE Conduce ADD FOREIGN KEY (Id) REFERENCES Coche (Id) ON DELETE CASCADE;

ALTER TABLE Conclusion ADD FOREIGN KEY (Accidente) REFERENCES Accidente (Id) ON DELETE CASCADE;

ALTER TABLE Conclusion ADD FOREIGN KEY (Consecuencia) REFERENCES Consecuencia (Id) ON DELETE CASCADE;

ALTER TABLE Conclusion ADD FOREIGN KEY (Error) REFERENCES Error (Id) ON DELETE CASCADE;

## APÉNDICE B

### INSTRUCCIONES DE USO DE LOS EJECUTABLES

Las aplicaciones están diseñadas para correr usando MySQL como gestor de base de datos. Si no se quiere usar MySQL se deberán modificar las clases *ControlBBDD* y *AccesoBD* del código fuente. El javadoc y los ficheros fuente contienen información sobre como modificar las clases.

#### B.1. Crear la base de datos

Antes de ejecutar las aplicaciones debemos crear la base de datos que las aplicaciones necesitan. Para ello se ha creado una pequeña aplicación que se encarga de crear las tablas y rellenarlas con algunos datos. Los pasos que se deben seguir son;

1. Instalar MySQL (si no está ya instalado).
2. Crear una nueva base de datos en MySQL.
3. Abrir una consola.
4. Desplazarnos al directorio donde se encuentre el archivo *inicializadorBD.jar*.
5. Ejecutar el siguiente comando:

```
> java -jar inicializadorBD.jar serverName userName password bd
```

Donde *serverName* es el nombre del servidor donde se encuentra la base de datos, *userName* el nombre del usuario en la base de datos, *password* la contraseña asociada al usuario y *bd* el nombre de la base de datos creada.

Por ejemplo:

```
> java -jar inicializadorBD.jar localhost root 012345 si
```

Con este proceso habremos creado una nueva base de datos con las tablas que se necesitan para las aplicaciones y algunos datos para facilitar el uso de las aplicaciones.

## **B.2. Ejecutar la aplicación de inicialización**

Para ejecutar la aplicación de inicialización y añadir más datos a la base de datos, el proceso es el siguiente:

1. Arrancar MySQL.
2. Abrir una consola.
3. Desplazarnos al directorio donde se encuentre el archivo *interfazBD.jar*.
4. Ejecutar el siguiente comando.

```
> java -jar interfazBD.jar serverName userName password bd
```

donde los parámetros *serverName*, *userName*, *password* y *bd* tienen los significados ya explicados.

## **B.3. Ejecutar la aplicación**

Para ejecutar la aplicación y comprobar que accidentes puede sufrir un usuario:

1. Arrancar MySQL.
2. Abrir una consola.
3. Desplazarnos al directorio donde se encuentre el archivo *aplicacion.jar*.
4. Ejecutar el siguiente comando.

```
> java -jar aplicacion.jar serverName userName password bd
```

donde los parámetros *serverName*, *userName*, *password* y *bd* tienen los significados explicados anteriormente.

## APÉNDICE C

### **ACTAS**

A continuación incluimos las actas redactadas durante el desarrollo del proyecto.

**Fecha:** 16 de noviembre de 2009

**Asistentes:** Miguel Ángel Blanco Rodríguez y Juan Pedro Franco Abia

El proyecto consiste en la creación de una aplicación que informará de los accidentes que un usuario sufrirá conduciendo un coche teniendo en cuenta errores históricos previos. La aplicación tendrá en cuenta las características personales (edad, sexo...), sociales (hobbies, estilo de vida...), técnicas (antigüedad del carnet, velocidad en carretera...) y profesionales (profesión, puesto en la empresa...) del usuario para inferir los accidentes (adelantamiento indebido y choque con un coche del otro carril, usar el móvil y salirse de la calzada...) que sufrirá.

Se ha decidido titular el proyecto como “Asistencia guiada en la conducción en función de errores históricos previos”.

Durante la realización del proyecto se harán reuniones semanales, las cuales quedarán registradas en actas que servirán para la fase de análisis funcional. En las reuniones se discutirá el trabajo realizado durante la semana y se decidirán los puntos a trabajar durante la siguiente semana.

También se ha decidido dividir el proyecto en cuatro fases:

1. Análisis funcional.
2. Diseño técnico.
3. Creación de un prototipo.
4. Redacción de la memoria.

El trabajo durante la semana consistirá en definir de una forma precisa las características del usuario y los errores que puede cometer.

**Fecha:** 23 de noviembre de 2009

**Asistentes:** Miguel Ángel Blanco Rodríguez y Juan Pedro Franco Abia

El alumno presenta una lista de características del usuario, una de errores que puede cometer y una de accidentes que puede sufrir. Se comenta que los elementos de la lista son adecuados pero que deberían presentarse categorizados y jerarquizados. Se propone que los errores y los accidentes se dividan teniendo en cuenta el lugar en el que pueden producirse: autopista, carretera convencional o ciudad.

El trabajo durante la semana consistirá en categorizar las listas de errores, accidentes y características del usuario.

**Fecha:** 30 de noviembre de 2009

**Asistentes:** Miguel Ángel Blanco Rodríguez y Juan Pedro Franco Abia

El alumno presenta las listas categorizadas y jerarquizadas de características del usuario, de errores que puede cometer y de accidentes que puede sufrir. Las listas se toman como definitivas aunque en el futuro puedan sufrir variaciones.

Las listas son las siguientes:

### **Características del usuario**

- Personales
  - ↳ NIF
  - ↳ Nombre
  - ↳ Apellidos
  - ↳ Edad
  - ↳ Sexo
- Sociales
  - ↳ Estado civil.
    - Soltero
    - Pareja de hecho.
    - Casado
    - Separado
    - Divorciado
    - Viudo
  - ↳ Número de hijos.
    - ↳ Menores de edad.
    - ↳ 18 – 25
    - ↳ > 25
  - ↳ Zona de vivienda.
    - Zona residencial.

- Centro ciudad.
  - Resto ciudad.
  - Pueblo
- ↳ Código postal.
- ↳ Hobbies
  - ↳ Públicos
    - Conciertos
    - Eventos deportivos.
    - Turismo fin de semana.
    - Senderismo
    - Cine
    - Teatro
  - ↳ Privados
    - Leer
    - Cocina
    - Pintura
    - Videojuegos
- Profesionales
  - ↳ Empresa
    - Multinacional / Grande.
    - Mediana
    - Pequeña
    - Autónomo
    - Estado
    - Estudiante
    - Parado
    - Jubilado
  - ↳ Ganancias anuales.
    - < 20.000
    - 20.000 – 50.000
    - > 50.000

- ↳ Horario trabajo.
  - Mañana
  - Tarde
  - Noche
  - Variable
- ↳ Puesto
  - Ejecutivo
  - Empleado
  - Conductor profesional.
  - Mecánico de automóviles.
  - Viajante
  - Reportero
  - Ama de casa.
- Conducción
  - ↳ Carnet de conducir.
    - ↳ Fecha
    - ↳ Categoría
  - ↳ Estilo conducción en carretera.
    - Despacio
    - Rápido
  - ↳ Estilo conducción en ciudad.
    - Despacio
    - Rápido
  - ↳ Coche
    - ↳ Combustible
      - Diesel
      - Gasolina
    - ↳ Puertas
      - 3 puertas.
      - 4 puertas.
      - 5 puertas.
    - ↳ Potencia

- ↳ Primera matriculación.
- ↳ Uso del vehículo.
  - Solo ocio.
  - Trayecto cotidiano.
  - Tareas profesionales ocasionalmente.
  - Desempeño habitual del trabajo.
- ↳ Lugar aparcamiento nocturno.
  - Vía pública.
  - Garaje individual.
  - Garaje colectivo.
- ↳ Km. recorridos al año.

### **Errores en la conducción**

- Cometidos por el usuario.
  - ↳ Velocidad
    - ↳ Frenada
      - ↳ Salir de la vía.
      - ↳ Aparcar
    - ↳ Lenta
      - ↳ Incorporación lenta.
      - ↳ Velocidad lenta para el tráfico.
    - ↳ Alta
      - ↳ Salir de la vía.
      - ↳ Velocidad alta para el tráfico.
      - ↳ Velocidad alta para las características de la vía.
  - ↳ Accesorio
    - ↳ Móvil
      - ↳ Llamar
      - ↳ Escribir SMS.
      - ↳ Usar el manos libres.

- ↳ Navegador
  - ↳ Configurar el navegador.
- ↳ Cinturón
  - ↳ No usar el cinturón.
- ↳ Cadenas
  - ↳ No usar cadenas cuando lo aconsejan las condiciones de la vía.
- ↳ Luces
  - ↳ Usar las luces largas constantemente.
  - ↳ Llevarlas apagadas.
- ↳ Estilo/Riesgo
  - ↳ Rotonda
    - ↳ Entrar
    - ↳ Salir
  - ↳ Cruce
    - ↳ Incorporación
    - ↳ Saltarse un stop.
  - ↳ Carril de aceleración.
    - ↳ Incorporación
  - ↳ Adelantamiento
    - ↳ Arriesgado
    - ↳ Zona prohibida.
  - ↳ Distancia
    - ↳ No guardar la distancia de seguridad.
  - ↳ Carril
    - ↳ Cambiar bruscamente de carril.
- ↳ Otros
  - Pisar la línea continua.
  - Conducir en sentido contrario.
  - Despistarse.

- Cometidos por otro conductor.
  - ↳ Velocidad
    - ↳ ...
  - ↳ Accesorio
    - ↳ ...
  - ↳ Estilo/Riesgo.
    - ↳ ...
  - ↳ Otros
    - ↳ ...

### **Accidentes**

- ↳ Recibir golpe.
  - ↳ Frontal
    - Mortal conductor.
    - Mortal resto.
    - Heridos graves.
    - Heridos leves.
    - Chapa y pintura.
  - ↳ Lateral
    - Mortal conductor.
    - ...
    - Chapa y pintura.
  - ↳ Detrás
    - Mortal conductor.
    - ...
    - Chapa y pintura.
  - ↳ Aparcado
    - Mortal conductor.
    - ...
    - Chapa y pintura.

- Dar golpe.
  - ↳ Frontal
    - ...
  - ↳ Lateral
    - ...
  - ↳ Detrás
    - ...
- Salirse de la calzada.
  - Mortal conductor.
  - ...
  - Chapa y pintura.
- Atropellar
  - ↳ Persona
    - Mortal
    - Herido grave.
    - Herido leve.
  - ↳ Ciclista
    - Mortal
    - Herido grave.
    - Herido leve.
  - ↳ Ciclomotor
    - Mortal
    - Herido grave.
    - Herido leve.
- Chocar con.
  - ↳ Animal
    - Mortal conductor.
    - ...
    - Chapa y pintura.
  - ↳ Señalización
    - Mortal conductor.
    - ...
    - Chapa y pintura.

El próximo lunes 7 de diciembre la facultad permanecerá cerrada, por tanto, se acuerda que la próxima reunión tendrá lugar el lunes 14 de diciembre. Para dicho día el alumno presentará 3 casos.

**Fecha:** 14 de diciembre de 2009

**Asistentes:** Miguel Ángel Blanco Rodríguez y Juan Pedro Franco Abia

El alumno presenta 3 casos tal y como se acordó en la reunión anterior. Al analizar los 3 casos se observa que en la inferencia intervienen 3 conjuntos de datos:

- A. Datos del usuario.
- B. Errores cometidos en la conducción.
- C. Accidentes.

que pueden relacionarse de las dos siguientes formas:

1. A partir de A inferir B, y partir del B inferido inferir C.

$A \rightarrow B$

$B \rightarrow C$

2. A partir de A y B inferir C.

$A, B \rightarrow C$

En los casos presentados se utilizó la primera opción. Sin embargo, se decide trabajar con la segunda opción porque conlleva una relación más fuerte que la primera opción, ya que solo se realiza una inferencia.

Para la siguiente semana el alumno presentará un diseño para la base de datos.

**Fecha:** 21 de diciembre de 2009

**Asistentes:** Miguel Ángel Blanco Rodríguez y Juan Pedro Franco Abia

El alumno presenta un diseño para la base de datos. Tras analizar el diseño se decide que se debe modificar para que refleje las categorías y jerarquías en que dividimos los diferentes datos.

También se discute la diferencia entre un accidente (por ejemplo, salirse de la calzada) y su consecuencia (por ejemplo, la muerte).

Los accidentes y consecuencias quedan jerarquizados de la siguiente forma:

#### **Accidentes**

- Recibir golpe.
  - Frontal
  - Lateral
  - Detrás
- Dar golpe.
  - Frontal
  - Lateral
  - Detrás
- Salirse de la calzada.
  - No chocar.
  - Chocar
- Atropellar
  - Persona
  - Ciclista
  - Moto
  - Animal

## Consecuencia

- Coche propio.
  - ↳ Conductor
    - Mortal
    - Incapacitado
    - Grave
    - Leve
    - Intacto
  - ↳ Copiloto
    - Mortal
    - ...
    - Intacto
  - ↳ Pasajero
    - Mortal
    - ...
    - Intacto
  - ↳ Coche
    - Siniestro total.
    - Taller
    - Chapa y pintura.
- Coche de un tercero.
  - ↳ Conductor
    - ...
  - ↳ Copiloto
    - ...
  - ↳ Pasajero
    - ...
  - ↳ Coche
    - ...

Para la siguiente reunión, el trabajo del alumno será rediseñar la base de datos teniendo en cuenta las observaciones hechas.

**Fecha:** 12 de enero de 2010

**Asistentes:** Miguel Ángel Blanco Rodríguez y Juan Pedro Franco Abia

El alumno presenta un nuevo diseño para la base de datos. Se decide continuar con este diseño.

Durante esta semana el trabajo del alumno será diseñar las sentencias SQL necesarias para crear la base de datos.

**Fecha:** 18 de enero de 2010

**Asistentes:** Miguel Ángel Blanco Rodríguez y Juan Pedro Franco Abia

Se presentan las sentencias SQL necesarias para la creación de la base de datos propuesta anteriormente.

Se discute qué campos de la base de datos pueden contener datos nulos, se decide que todos salvo los que formen partes de claves primarias (por razones obvias) y los códigos de error, accidente o consecuencia (para mantenerlos jerarquizados y categorizados).

Para la semana siguiente, el alumno creará una aplicación para probar el correcto funcionamiento de la base de datos y facilitar su inicialización.

**Fecha:** 25 de enero de 2010

**Asistentes:** Miguel Ángel Blanco Rodríguez y Juan Pedro Franco Abia

El alumno presenta la aplicación que funciona perfectamente. De esta forma ha conseguido inicializar la base de datos y comprobar que esta funciona correctamente y que presenta todas las características necesarias.

El siguiente paso es crear la base de inferencia. Se presentan 3 opciones:

1. Bloques de if-then-else.
2. Una tabla donde cada fila represente las premisas y la conclusión.
3. Dos tablas una para premisas y otra para conclusiones.

Durante la semana, el trabajo del alumno será valorar las opciones y elegir la que considera más adecuada.

**Fecha:** 22 de febrero de 2010

**Asistentes:** Miguel Ángel Blanco Rodríguez y Juan Pedro Franco Abia

En la reunión anterior se presentaron 3 opciones para crear la base de inferencia. Las opciones eran:

1. Bloques de if-then-else.
2. Una tabla donde cada fila represente las premisas y la conclusión.
3. Dos tablas una para premisas y otra para conclusiones.

El alumno elige la tercera opción para implementar la base de inferencia. Por tanto, presenta un diseño de la base de datos acorde a la opción elegida. También presenta un prototipo con la implementación de la opción elegida que funciona.

A partir de ahora, el trabajo del alumno será crear la memoria del proyecto.

## **Bibliografía**

- Korth, Henry F. y Silberschatx, Abraham. *Fundamentos de bases de datos*. Traducción: Vaquero García, Antonio y Vaquero Martín, M<sup>a</sup> Ángeles. McGraw-Hill, 2006.
- Oracle. *MySQL 5.0 Reference Manual*. 2010