



**Universitat Autònoma  
de Barcelona**

**HERRAMIENTA WEB  
PARA LA GESTIÓN DE PROYECTOS  
DE INVESTIGACIÓN**

Memoria del proyecto  
de Ingeniería Técnica en  
Informática de Gestión

Realizado por  
Marta Rubia Jiménez  
y dirigido por  
Gonzalo Vera Rodríguez

Escuela de Ingeniería  
Sabadell, Septiembre de 2011

[El/La] abajo firmante, **Gonzalo Vera Rodríguez**,  
profesor[/a] de la Escuela de Ingeniería de la UAB,

**CERTIFICA:**

Que el trabajo al que corresponde la presente  
memoria ha sido realizada bajo su dirección  
por

**Marta Rubia Jiménez**

Y para que conste firma la presente.  
Sabadell, **Septiembre** de **2011**

-----  
Firmado: **Gonzalo Vera Rodríguez**  
**director/a**

## RESUMEN

<b>Título del proyecto:</b> HERRAMIENTA WEB PARA LA GESTIÓN DE PROYECTOS DE INVESTIGACIÓN	
<b>Autora:</b> Marta Rubia Jiménez	<b>Fecha:</b> Septiembre de 2011
<b>Tutor:</b> Gonzalo Vera Rodríguez	
<b>Titulación:</b> Ingeniería técnica informática de gestión.	
<b>Palabras clave</b> (mínimo 3) <ul style="list-style-type: none"><li>• Catalán: Gestió, administració, web, projectes de investigació, gestió de projectes.</li><li>• Castellano: Gestión, administración, web, proyectos de investigación, gestión de proyectos.</li><li>• Inglés: Management, government, web, research projects, research project management</li></ul>	
<b>Resumen del proyecto</b> <ul style="list-style-type: none"><li>• Catalán:<p>El projecte té com a objectiu el desenvolupament d'una aplicació web per a la gestió de projectes de investigació que es realitzen el Servei de Nutrició i Benestar Animal (SNiBA) de la Universitat Autònoma de Barcelona. L'aplicació ha de ser una eina per a facilitar l'organització i l'administració, al personal del centre, dels diferents projectes de I+d que són realitzats per aquesta entitat. SNiBA ja disposa d'una aplicació per a aquest fi però l'aplicació té algunes mancances.</p><p>Amb aquesta aplicació es pretèn facilitar i millorar el procès de gestió dels serveis que ofereix SNiBA de forma que el personal pugui oferir un millor servei als seus clients.</p></li><li>• Castellano:<p>El proyecto tiene como objetivo el desarrollo de una aplicación web para la gestión de proyectos de investigación que se llevan a cabo por el Servei de Nutrició i Benetar</p></li></ul>	

Animal (SNiBA) de la Universitat Autònoma de Barcelona. La aplicación ha de ser una herramienta para facilitar la organización y administración, al personal del centro, de los diferentes proyectos de I+D que son realizados por esta entidad. SNiBA ya dispone de una aplicación para este fin pero con ciertas carencias.

Con esta aplicación se pretende facilitar y mejorar el proceso de gestión de los servicios que ofrece SNiBA de forma que el personal pueda ofrecer un mejor servicio a sus clientes.

- Inglés:

The goal of the project is to develop a web application to manage the research projects which are made by Servei de Nutrició i Benestar Animal (SNiBA) of the Universitat Autònoma de Barcelona. The application is intended to improve and help with the current research project management of SNiBA. SNiBA, currently has an application for similar purposes, but it has deficiencies.

With this project we will try to improve the management of the services that SNiBA offers to its customers.



## ÍNDICE

<b>1. INTRODUCCIÓN</b>	<b>1</b>
1.1 MOTIVACIÓN	1
1.2 OBJETIVOS	2
1.3 ESTADO DEL ARTE	3
1.4 ESTRUCTURA DE LA MEMORIA	5
<b>2. ESTUDIO DE VIABILIDAD</b>	<b>6</b>
2.1 INTRODUCCIÓN	6
2.2 ESTUDIO DE LA SITUACIÓN ACTUAL	6
2.3 OBJETIVOS DEL PROYECTO	7
2.4 REQUERIMIENTOS GENERALES DEL PROYECTO	7
2.5. PROPUESTA DE SOLUCIÓN Y ALTERNATIVAS	10
2.6. PLANIFICACIÓN DEL PROYECTO	11
2.7. EVALUACIÓN DE LOS RIESGOS	14
2.8. EVALUACIÓN DE COSTES Y BENEFICIOS	16
2.9. CONCLUSIONES	18
<b>3. ANÁLISIS</b>	<b>19</b>
3.1 INTRODUCCIÓN	19
3.2 DOMINIO DEL PROBLEMA	19
3.3 REQUERIMIENTOS FUNCIONALES	20
3.4 REQUERIMIENTOS NO FUNCIONALES	27
3.5 MARCO TECNOLÓGICO	27
<b>4.DISEÑO</b>	<b>30</b>
4.1 INTRODUCCIÓN	30
4.2 ARQUITECTURA DE LA APLICACIÓN	30
4.3 TECNOLOGÍAS Y HERRAMIENTAS A UTILIZAR	31
4.4 PERFILES DE USUARIOS	31
4.5 DIAGRAMA ENTIDAD RELACIÓN	32
4.6 MODELO DE DATOS	34
4.7 INTERFAZ	42
<b>5.IMPLEMENTACIÓN</b>	<b>44</b>
5.1 INTRODUCCIÓN	44

<b>5.2 INTERFAZ DE USUARIO</b>	<b>44</b>
<b>5.3 INTERFAZ DE LOS PERFILES DE USUARIOS</b>	<b>49</b>
<b>5.4 BASE DE DATOS</b>	<b>51</b>
<b>5.5 MULTILENGUAGE</b>	<b>52</b>
<b>6. PRUEBAS</b>	<b>55</b>
<b>6.1 INTRODUCCIÓN</b>	<b>55</b>
<b>6.2 PRUEBAS DE INTERFACES</b>	<b>55</b>
<b>6.3 PRUEBAS DE SEGURIDAD</b>	<b>56</b>
<b>6.4 PRUEBAS</b>	<b>57</b>
<b>7.CONCLUSIONES</b>	<b>62</b>
<b>7.1 SEGUIMIENTO DEL PROYECTO</b>	<b>62</b>
<b>7.2 DESVIACIONES</b>	<b>65</b>
<b>7.3 FUTURO DEL PROYECTO Y POSIBLES MEJORAS</b>	<b>66</b>
<b>7.4 VALORACIÓN PERSONAL</b>	<b>67</b>
<b>8.BIBLIOGRAFIA Y REFERENCIAS</b>	<b>68</b>

## ÍNDICE DE FIGURAS

FIGURA 1 ACCESIBILIDAD .....	11
FIGURA 2 DIAGRAMA DE GANTT .....	13
FIGURA 3 GENERAR OFERTA .....	20
FIGURA 4 ACEPTACIÓN OFERTA .....	20
FIGURA 5 FICHA OFERTA .....	22
FIGURA 6 FICHA PROYECTO .....	23
FIGURA 7 FICHA CLIENTE .....	24
FIGURA 8 DIAGRAMA CREAR OFERTA .....	25
FIGURA 9 CASOS DE USO .....	26
FIGURA 10 ARQUITECTURA APLICACIÓN .....	30
FIGURA 11 DIAGRAMA ENTIDAD RELACIÓN .....	33
FIGURA 12 GAMA DE COLORES .....	43
FIGURA 13 PÁGINA DE INICIO .....	44
FIGURA 14 PRIMERA PÁGINA .....	45
FIGURA 15 MENÚ COMÚN .....	45
FIGURA 16 FORMULARIO OFERTA .....	46
FIGURA 17 MENÚ OFERTAS .....	47
FIGURA 18 MENÚ PROYECTOS .....	47

FIGURA 19 BUSCAR PROYECTO.....	48
FIGURA 20 VISUALIZACIÓN PERFIL ADMINISTRADOR .....	49
FIGURA 21 VISUALIZACIÓN PERFIL LECTOR.....	49
FIGURA 22 BÚSQUEDA OFERTAS.....	50
FIGURA 23 BÚSQUEDA PERFIL LECTOR .....	50
FIGURA 24 BASE DE DATOS .....	51
FIGURA 25 FORMULARIO IDIOMA .....	52
FIGURA 26 LLAMADA A FUNCIONES GETLANG(CONEXIÓN, USUARIO) Y CREATEDICTIONARY(LANG) .....	53
FIGURA 27 CONSULTA BASE DE DATOS .....	53
FIGURA 28 UBICACIÓN DE LAS VARIABLES DE SESIÓN .....	54
FIGURA 29 FUNCIÓN VALIDAR NUMÉRICOS.....	54
FIGURA 30 LLAMADA FUNCIÓN VALNUMERO()).....	54
FIGURA 31 ERROR NOMBRE USUARIO.....	56
FIGURA 32 MENSAJE ERROR.....	57
FIGURA 33 DIAGRAMA DE GANTT, PLANIFICACIÓN REAL .....	63
FIGURA 34 DIAGRAMA DE GANTT, PLANIFICACIÓN REAL .....	64
FIGURA 35 DIAGRAMA DE GANTT, PLANIFICACIÓN REAL .....	64

## 1. INTRODUCCIÓN

### 1.1 MOTIVACIÓN

El proyecto nace de la necesidad de desarrollar una aplicación para poder gestionar los proyectos que realiza el centro Servei de Nutrició i Benestar Animal (SNiBA) de la Universitat Autònoma de Barcelona.

El centro SNiBA es un servicio científico-técnico de la Universitat Autònoma de Barcelona que promueve y lidera investigación en el sector agroalimentario.

El objetivo de sus servicios es el de fomentar la transferencia de conocimiento entre la universidad pública y la empresa privada mediante el asesoramiento, gestión, desarrollo y resolución integral de proyectos I+D+i sobre nutrición, manejo, producción y bienestar animal.

Los servicios que ofrecen son los siguientes:

- ❖ Gestión y desarrollo de proyectos de I+D+i con empresas.
- ❖ Estudios experimentales y técnicas analíticas.
- ❖ Asesoramiento al sector agroalimentario.
- ❖ Actividades de formación (masters y jornadas técnicas para empresas).

La razón por la que se ha escogido esta propuesta se fundamenta en que para llevarlo a cabo es necesario poner en práctica el conjunto de habilidades y conocimientos que se han adquirido a lo largo de la carrera. Por otro lado, SNiBA ya dispone de una aplicación para la gestión de proyectos, pero esta herramienta tiene algunas carencias.

Con este proyecto se pretende mejorar la aplicación de gestión de proyectos que dispone actualmente SNiBA, y poder eliminar esas carencias y añadir nuevas funcionalidades útiles para la gestión de proyectos.

## 1.2 OBJETIVOS

El objetivo principal de este proyecto es mejorar la aplicación que actualmente utiliza el centro SNI BA para la gestión de sus proyectos. Se pretende dotar a la nueva aplicación de características y nuevas funcionalidades para que toda la información que poseen de los diferentes proyectos sea mucho más fácil de gestionar. De esta manera, los empleados e investigadores del centro podrán acceder a la información de una forma más rápida y organizada.

Gracias a la nueva aplicación, los empleados del centro, podrán ahorrar tiempo en consultar datos referentes a los proyectos, hacer una búsqueda de proyectos teniendo en cuenta diferentes filtros, podrán almacenar los datos necesarios de los clientes con los que trabajan o acceder a los datos de dichos clientes.

Poder acceder rápidamente a la información que se requiere en cada momento y poder gestionar esta información de la mejor forma posible, permite un ahorro considerable de tiempo, y el ahorro en tiempo se traduce en dinero.

El desarrollo del proyecto consta de las siguientes partes bien diferenciadas:

1. **La gestión de los datos** necesarios para llevar a cabo los proyectos de investigación. Antes de realizar un proyecto se genera una solicitud de oferta, ésta ha de ser validada, si se valida se crea la ficha de proyecto. Todos estos datos se han de poder gestionar de la mejor manera posible para facilitar el trabajo a los empleados del centro que son los que realizarán dichos proyectos.
2. **La gestión de usuarios.** Los usuarios que interactúen con la aplicación tendrán diferentes roles, cada uno de ellos podrá acceder a cierta información relacionada con los proyectos. Esta gestión de usuario es importante para garantizar un buen funcionamiento en el desarrollo de los proyectos que se realizarán, así como asegurar la integración de los datos.
3. **La accesibilidad a la aplicación.** Conseguir que la aplicación sea accesible desde cualquier ordenador, sin necesidad de estar en el mismo centro de trabajo, agiliza y facilita el trabajo a los usuarios de la aplicación.

4. **Múltiples idiomas.** Actualmente, la herramienta de gestión de proyectos que utiliza SNiBA, sólo se muestra en un idioma, en catalán, por lo tanto otro objetivo es hacer que la aplicación se muestre en varios idiomas, además de que el administrador de la aplicación, si lo desea pueda añadir otro idioma o modificar y eliminar los existentes.

### 1.3 ESTADO DEL ARTE

Actualmente SNiBA dispone de una herramienta para poder gestionar proyectos de investigación en el sector agroalimentario. La herramienta está desarrollada con una base de datos en *Microsoft Acces* a la cual pueden acceder tres perfiles de usuarios diferentes. El principal inconveniente de la herramienta es la carencia de control de estos usuarios, actualmente cualquier usuario puede tener acceso a la aplicación sin necesidad de identificarse con un nombre de usuario y una contraseña.

Por otro lado, sólo se puede acceder a la aplicación desde el mismo centro, esto limita mucho a los usuarios, ya que se han de desplazar hasta el centro para poder trabajar con la aplicación. Por este motivo, se decide crear la aplicación en entorno web, para hacer más fácil el acceso.

El gestor de proyectos con el que trabaja SNiBA actualmente sólo se muestra en un idioma, en catalán, éste es otro de los motivos por los que se pretende mejorar la aplicación.

Se ha buscado información sobre aplicaciones similares, las cuales nos puedan ser útiles y cubran las necesidades para la gestión de los proyectos de SNiBA.

Una de las aplicaciones que quizás se ajuste a nuestras necesidades sería dotProject. Es una aplicación de código abierto y programada en PHP utilizando una base de datos MySQL. Está basada en web, es multiusuario y soporta varios lenguajes. La aplicación consta de un conjunto de entidades, las más importantes son:

- Compañías: Son las entidades que agrupan proyectos, actividades y usuarios.
- Departamentos: Son áreas dentro de las compañías, que permiten agrupar usuarios en dicho nivel.
- Usuarios/Contactos: dotProject tiene usuarios los cuales son capaces de loguearse a dotProject y trabajar dentro del esquema de permisos que posea el rol de dicho usuario. Los contactos son usuarios especiales que asignados a un determinado proyecto pueden recibir por ejemplo: correo, actualizaciones y noticias pero no necesariamente deben tener acceso al sistema dotProject. Los usuarios y contactos pertenecen a una compañía.
- Proyectos: Es la entidad que contiene el grupo de tareas necesarias para desarrollar un determinado producto o servicio.
- Actividades: son las tareas asignadas dentro de un proyecto.

Aunque dotProject se aproxima a las necesidades de SNIiBA para la gestión de sus proyectos, no es suficiente ya que necesitamos más funcionalidades las cuales, dotProject no nos las ofrece.

Se ha buscado más información al respecto pero no se ha encontrado ningún software que cumpla con los requerimientos que necesitamos. Por lo que, podemos decir que SNIiBA necesita un software a medida.

Para que SNIiBA disponga de un software que cumpla con sus necesidades se propone desarrollar una aplicación en entorno web, para que pueda ser accesible no solo desde el centro y multiusuario.

Para crear un software entorno web, se requiere la utilización de varias tecnologías y componentes. Para alojar la aplicación y que sea accesible se necesita un servidor web y un servidor de base de datos para poder almacenar toda la información que se vaya generando. La aplicación se puede desarrollar en varios lenguajes de programación enfocados a web como por ejemplo, PHP, ASP.Net, Java, ASP o python.

Más adelante, en el punto 3.5 MARCO TECNOLÓGICO, se explican cada una de las tecnologías de las que disponemos para desarrollar el software, se toma una decisión y se explica porque se ha tomado esa decisión.

## 1.4 ESTRUCTURA DE LA MEMORIA

La memoria se estructura en diferentes capítulos.

En el segundo capítulo, encontramos el estudio de viabilidad, donde se analizan cuales son las necesidades que hay que cubrir y de qué forma se han de desarrollar las soluciones para determinar si es viable desde el punto de vista técnico y de recursos.

En el tercer capítulo, se analiza en profundidad cuales son las funcionalidades esperadas de la aplicación, que funcionalidades hay que mejorar de la antigua aplicación y bajo qué condiciones se ha de desarrollar una solución a los objetivos del proyecto. En esta parte también se analiza la tecnología que hay disponible y las alternativas que se pueden hacer servir. Una vez analizado el problema se plantea un diseño de la solución.

En el capítulo referente al diseño, podemos encontrar cómo se ha diseñado la base de datos, el diagrama de entidad-relación con sus respectivos significados y los motivos por los que se ha escogido una interfaz de usuario determinada.

Después del diseño, encontramos la implementación. En el capítulo de la implementación se explica el resultado final de la aplicación y como se ha puesto en funcionamiento. Para que las explicaciones sean más comprensibles, se presentan algunos ejemplos de programación.

En la memoria se explican también las pruebas que se han realizado en la aplicación para detectar errores, tanto errores en el código como errores de seguridad.

Seguidamente se detallan las conclusiones a las que se han llegado después de finalizar el proyecto, que diferencias y desviaciones han surgido desde el principio hasta el final del desarrollo, la valoración personal y que mejoras puede tener la aplicación en un futuro.

Por último, se anexa la bibliografía y referencias de donde se ha buscado información.



## **2. ESTUDIO DE VIABILIDAD**

### **2.1 INTRODUCCIÓN**

En este apartado analizaremos un conjunto concreto de necesidades para poder proponer una solución a corto plazo para el proyecto.

Se han de tener en cuenta una serie de restricciones tales como restricciones técnicas, económicas y operativas para poder decidir si el proyecto es viable y se puede llevar a cabo.

Primero se analiza la situación actual y si existen soluciones al problema. A partir de la solución propuesta se hace una estimación de la planificación, un análisis de los riesgos y una evaluación de los costes y los beneficios.

### **2.2 ESTUDIO DE LA SITUACIÓN ACTUAL**

Actualmente SNI BA dispone de una herramienta para poder gestionar proyectos de investigación en el sector agroalimentario. La herramienta está desarrollada con una base de datos Microsoft Acces a la cual pueden acceder tres perfiles de usuarios diferentes. El principal inconveniente de la herramienta existente es la carencia de control de estos usuarios. Otro problema a solventar es el poder identificar cada proyecto con un identificador único así como poder tener la aplicación en varios idiomas y poder acceder a ella desde cualquier ubicación.

Con el desarrollo de la nueva herramienta se mejorará el sistema de gestión de proyectos ya existente solventando las mancan en la aplicación, reduciendo el tiempo de trabajo de los usuarios resultando más fácil y cómodo interactuar con esta.

## 2.3 OBJETIVOS DEL PROYECTO

El objetivo de este proyecto es mejorar el sistema de gestión de proyectos actual ampliando sus funciones de forma que los usuarios finales, los cuales serán el equipo que integra el centro SNI BA, adquieran mayor rapidez y mayor comodidad con la gestión de las investigaciones que realizan. Los objetivos para llevar a cabo el proyecto son:

- Realizar un estudio sobre la herramienta existente.
- Estudiar la metodología utilizada a la hora de gestionar los proyectos.
- Creación de la interfaz de usuario.
- Identificar los diferentes perfiles.
- Hacer más accesible la aplicación desde cualquier ubicación.
- Convertirla en multilinguaje.
- Reducir el tiempo de trabajo de los usuarios.

## 2.4 REQUERIMIENTOS GENERALES DEL PROYECTO

Para definir los requerimientos del proyecto hace falta conocer las funcionalidades que necesitará tener la aplicación.

En función de las características del entorno y las restricciones que tendrá la aplicación en su uso definiremos los requerimientos no funcionales y unas restricciones.

## Requerimientos funcionales

Los requerimientos funcionales son el conjunto de funciones que deberá cumplir el sistema para cumplir con los objetivos. Ese conjunto de requerimientos se describen a continuación:

### ❖ Usuarios

- Cada usuario se deberá identificar con un nombre de usuario y contraseña, el sistema debe autenticar esas credenciales.
- Dar acceso a usuarios.
- Asignar un perfil a cada usuario.
- Poder modificar el perfil de un usuario, pudiendo adjudicarle otro.
- Eliminación, creación y modificación de usuarios.
- Permitir acceso concurrente.

### ❖ Ofertas

- Crear una nueva oferta.
- Asignarle un estado y un cliente a cada oferta.
- Asignar a la oferta los empleados que trabajarán en ella.
- Búsqueda de ofertas teniendo en cuenta varios filtros.
- Modificar los datos de una oferta.
- Eliminar una oferta.
- Crear un nuevo proyecto al modificar el estado de la oferta a “cerrada”.

### ❖ Proyectos

- Crear un nuevo proyecto, sólo en el caso que la oferta que esté en estado cerrada.
- Asignarle un cliente.
- Asignarle los empleados que trabajarán en el proyecto.
- Modificar los datos del proyecto.
- Búsqueda de proyectos teniendo en cuenta varios filtros.
- Eliminar un proyecto.

### ❖ Gestión de la aplicación

- Introducir en el sistema nuevos idiomas.
- Modificar idiomas.
- Eliminar idiomas.
- Crear copias de seguridad y permitir la restauración de esas copias.

### Requerimientos no funcionales

Los requerimientos no funcionales hacen relación a las características, condiciones o restricciones que pone el cliente a la solución del sistema. Estos requerimientos no están relacionados con los requerimientos funcionales pero son adicionales a ellos y son igual de importantes ya que son el conjunto de características que corresponden a aspectos tales como la disponibilidad, escalabilidad, seguridad o facilidad de uso.

Los requerimientos no funcionales son los siguientes:

- **Facilidad de uso:** El sistema debe ser de fácil uso para los usuarios de la aplicación, además la navegación ha de ser rápida y entendible con una interfaz amigable.
- **Disponibilidad:** La aplicación debe estar disponible al 100%, sobre todo durante el horario laboral de SNIIBA, se ha de poder acceder a ella desde cualquier ordenador, además de poder funcionar independientemente del navegador que se utilice.
- **Escalabilidad:** El sistema debe ser construido de manera que las nuevas funcionalidades y requerimientos relacionados puedan implementarse afectando al código existente de la menor manera posible. Además debe permitir poder eliminar funcionalidades existentes de la misma forma.
- **Validación de información:** El sistema debe validar automáticamente la información contenida en los formularios de ingreso. En el proceso de validación de la información, se deben tener en cuenta aspectos tales como obligatoriedad de campos o longitud de caracteres permitida por campo.

### **Restricciones del sistema**

Las restricciones del sistema son el conjunto de limitaciones que hay que tener en cuenta a la hora de escoger una solución ya que estas restricciones impedirán la elección de una u otra solución. El conjunto de restricciones son:

- La aplicación ha de ser una aplicación web a través de internet para facilitar el acceso.
- La aplicación se ha de desarrollar en un lenguaje para desarrollo web.

## **2.5. PROPUESTA DE SOLUCIÓN Y ALTERNATIVAS**

Se propone diseñar y desarrollar una aplicación web que facilite la gestión de proyectos de investigación llevados a cabo por el centro SNIBA, agilizando la organización y administración de los proyectos así como la introducción, modificación y creación de la información necesaria asociada a cada proyecto.

Con el objetivo de mejorar el sistema actual hay que identificar a los diferentes usuarios los cuales tendrán acceso a la información y podrán modificar o no los datos dependiendo del rol que tengan, facilitándoles de esta manera el trabajo. Se debe diseñar una base de datos relacional que permita la integridad de los datos.

Existen un conjunto de alternativas tecnológicas para llevar a cabo la aplicación web, estas alternativas se deben estudiar para determinar qué es lo que mejor se ajusta para implementar la solución propuesta.

La solución propuesta es implementar una aplicación web con una base de datos para almacenar la información. De esta manera la aplicación podrá ser accesible desde cualquier ordenador y con ello se solucionaría el problema de acceso que actualmente existe en la aplicación que se utiliza en SNIBA. La Figura 1 representa cómo varios usuarios pueden acceder a la aplicación desde diferentes ordenadores sin importar su ubicación.

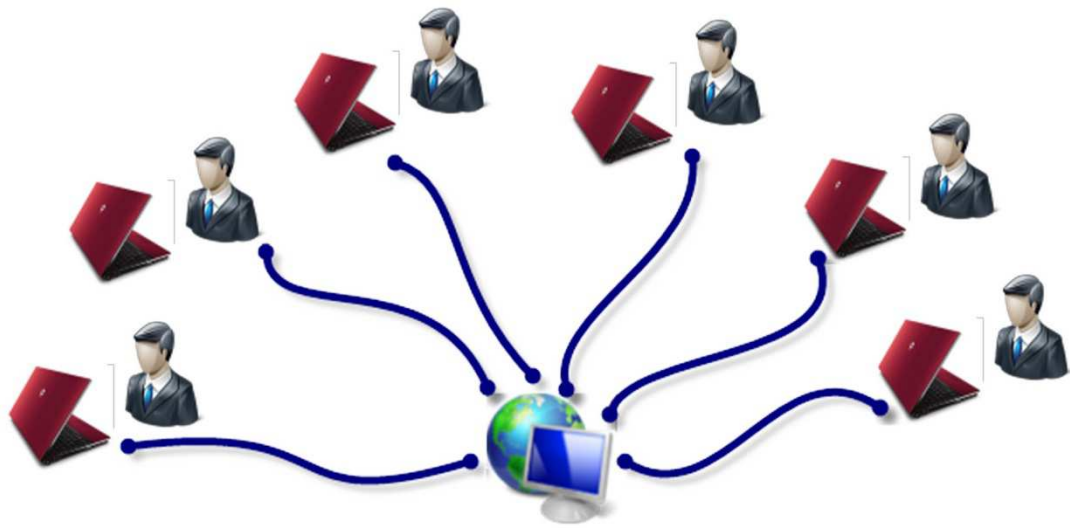


Figura 1 Accesibilidad

### Modelo de desarrollo

El sistema a realizar seguirá un modelo de desarrollo evolutivo, puesto que hay partes del proyecto como el diseño de la interfaz de usuario que se puede diseñar paralelamente con el desarrollo de las funciones de la aplicación, así como también el diseño de la base de datos.

## 2.6. PLANIFICACIÓN DEL PROYECTO

En la planificación del proyecto se estima el tiempo requerido en las diferentes tareas que se van a desarrollar. La tabla muestra en la primera columna la tarea a realizar y en la segunda el tiempo estimado en horas. Más adelante, en la Figura 2 se muestra el diagrama de Gantt.

A la hora de hacer la planificación, se ha tenido en cuenta que sólo se puede dedicar al proyecto 4 horas diarias de lunes a viernes, esto se ve reflejado en diagrama de Gantt.

### Tareas del proyecto

<b>Tareas</b>	
Estudio de viabilidad	14 h
Análisis de los datos	6 h
Casos de uso	4 h
Diseño base de datos	5 h
Diseño de la aplicación	14 h
Diseño de la interfaz	7 h
<b>Preparación del entorno de programación</b>	
Instalación y configuración del servidor web	3 h
Instalación y configuración del servidor de base de datos	4 h
<b>Implementación</b>	
Codificación	90 h
Test de pruebas	8 h
Corrección de errores	10 h
<b>Documentación</b>	
Elaboración de la memoria	20 h
Corrección de la memoria	5 h
Preparación de la presentación	10 h
<b>Total</b>	<b>200 h</b>

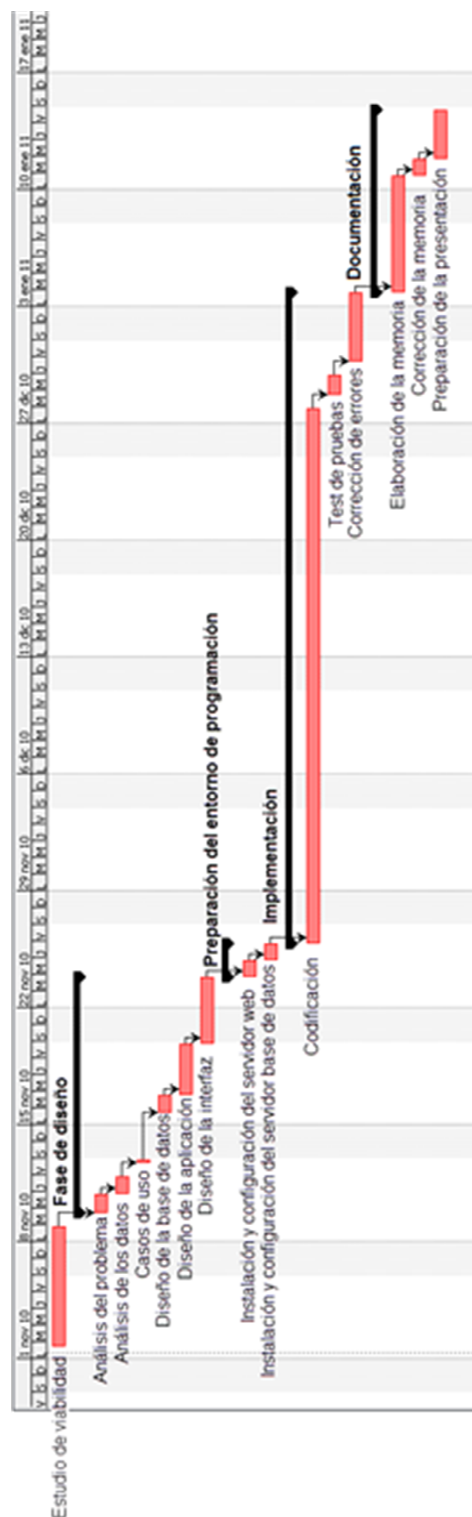


Figura 2 Diagrama de Gantt



## 2.7. EVALUACIÓN DE LOS RIESGOS

Existen varios riesgos asociados al proyecto. Uno de los riesgos que nos podríamos encontrar una vez desarrollada la aplicación y después de su puesta en marcha podría ser la incompatibilidad con diferentes navegadores, en ocasiones dependiendo del navegador que se utilice la interfaz de usuario podría variar, dando errores de diseño.

El problema radica en que no todos los navegadores interpretan en código HTML y CSS de la misma manera, entre ellos existen pequeñas variaciones que son las que hacen que el resultado no sea el mismo de unos a otros.

Algunas de esas diferencias son tan importantes que pueden hacer que partes de la aplicación no funcionen o no se vean, así que nos interesa que la web funcione en el mayor número de navegadores posibles.

Para solucionar la incompatibilidad, previamente se debería visualizar la web en diferentes navegadores como Internet Explorer, Opera o Firefox, que son los más utilizados. Para evitar tener que instalar varios navegadores en nuestro PC, existen sitios web que realizan capturas de pantalla de nuestra página web en diferentes navegadores. Algunos de estos sitios web son:

- ❖ Browsershot (todos los SO + todos los navegadores)
- ❖ Browsercam (de prueba- todos los SO + múltiples navegadores)
- ❖ IEcapture (IE7)
- ❖ Icapture (Mac + Safari)
- ❖ Scapture (Linux + Firefox)

Otra solución para mejorar la compatibilidad de navegadores sería validar el código en base a los estándares del W3C. W3C es una organización que se encarga de desarrollar especificaciones técnicas y directivas, así como normas a seguir en el desarrollo de una aplicación web para asegurar su calidad.

A continuación se detallan otros riesgos y sus respectivos planes de contingencia.

Riesgo	Plan de contingencia
Planificación temporal optimista: El proyecto no se finaliza en el tiempo previsto.	Acortar algunas de las tareas y afrontar las posibles pérdidas.
Cambio de requisitos: Retraso en el desarrollo y el resultado.	Renegociar con el cliente para recortar funcionalidades y/o modificar la planificación.
Diseño de la solución inadecuado.	Mejorar la formación del equipo y prever diseños alternativos.
Problema relacionado con la red interna del cliente, pérdida de conexión a Internet, caída del servidor o similar.	Realizar copias de seguridad periódicamente.
La fase de test no se hace correctamente por lo que la aplicación no tendrá la calidad deseada.	Diseñar el test con tiempo, probar todos los casos de uso y realizar el test exhaustivamente.
Falta de controles de seguridad en la aplicación: Podría provocar manipulación no deseada de la información y pérdidas de datos.	Revisar y testear la seguridad.

## 2.8. EVALUACIÓN DE COSTES Y BENEFICIOS

A continuación se evalúan los costes del proyecto.

En cuanto a los recursos humanos, el coste vendrá dado por el precio / hora de cada uno de los cargos que intervienen en el proyecto.

Cargo	Precio €/hora	Horas	Total
Jefe Proyecto	50 €	49 h	2.450 €
Analista/programador	40 €	36 h	1.440 €
Programador	30 €	107 h	3.210 €
Técnico de test	25 €	8 h	200 €
<b>TOTAL</b>	<b>145 €</b>	<b>200 h</b>	<b>7.300 €</b>

En este proyecto, no se consideran costes de software, ya que estos recursos nos han sido facilitados por la escuela, en caso contrario nos hubieran supuesto un coste aproximado de:

Recursos	Características	Precio
Servidor base de datos	SQL Server 2008	4.271 €
Servidor web	Windows server 2008 Enterprise	2.178 €
Licencia visual studio 2008		371,84 €
<b>TOTAL</b>		<b>6.820,84 €</b>

El alojamiento nos lo facilita la UAB por ser estudiantes de la escuela, nos presta el servicio gratuitamente, por lo que tampoco supone ningún gasto.

Teniendo en cuenta todos los costes, el coste final del proyecto de 14.120,84 €, tal y como se detalla en la siguiente tabla.

Total costes	Total en €
Recursos humanos	7.300 €
Recursos de software y hardware	6.820,84 €
<b>TOTAL</b>	<b>14.120,84 €</b>

A continuación se hace una estimación de cual podría ser el beneficio.

En la tabla siguiente, se estima el presupuesto de un proyecto realizado por SNIIBA el cual consta de 190 horas. Se tiene en cuenta el personal que trabaja en un proyecto, éstos son el Director, el responsable y el técnico. La segunda columna indica el precio que cobra por hora cada uno de los empleados, la tercera columna el tiempo en horas que cada uno de los empleados le dedica al proyecto utilizando la aplicación actual de la que disponen.

#### Actual aplicación.

Personal	Precio €/h	Horas / proyecto	Total
Director	40 €	50 h	2.000 €
Responsable	25 €	60 h	1.500 €
Técnico	20 €	80 h	1.600 €
<b>TOTAL</b>		<b>190 h</b>	<b>5.100 €</b>

Con la nueva aplicación se estima que el tiempo de trabajo de cada uno de los empleados disminuirá, quedando el coste de un proyecto como indica la siguiente tabla.

#### Nueva aplicación.

Personal	Precio €/h	Horas / proyecto	Total
Director	40 €	35 h	1.400 €
Responsable	25 €	55 h	1.375 €
Técnico	20 €	75 h	1.500 €
<b>TOTAL</b>		<b>165 h</b>	<b>4.275 €</b>

El ahorro de tiempo de trabajo utilizando la nueva aplicación es de 25 horas, esto supone un ahorro de 825 € por proyecto.

	Actual aplicación	Nueva aplicación	Total
Tiempo invertido	190 h	165 h	<b>25 h</b>
Coste	5.100 €	4.275 €	<b>825 €</b>

Teniendo en cuenta que el centro no realiza más de 100 proyectos al año, estimamos que cada año se generan 15 proyectos.

	Nº proyectos /año	Coste año
Actual aplicación	15	76.500 €
Nueva aplicación	15	64.125 €
Diferencia		12.375 €

Con las estimaciones que se han realizado anteriormente podemos ver que el ahorro que supondrá la nueva aplicación es de **12.375 €**. Los costes totales de la nueva aplicación son de 14.120,84 €. Por lo que antes de terminar el segundo año se habrán amortizado los costes totales de la nueva aplicación.

## 2.9. CONCLUSIONES

Finalmente, debemos determinar si el proyecto es viable o no.

Teniendo en cuenta todo lo que se ha expuesto anteriormente, podemos decir que tenemos los medios necesarios para poder desarrollar la aplicación. Podemos hacer uso de un servidor web para alojar la aplicación, podemos acceder a un servidor de base de datos para almacenar toda la información que se vaya generando a través de la aplicación, tenemos acceso a información y conocimiento de los lenguajes de programación en los cuales se desarrollará la herramienta web.

Por otro lado, el proyecto se puede desarrollar dentro del tiempo establecido.

Teniendo en cuenta que gracias a la aplicación mejoraremos el acceso a los datos y a la organización y administración de los proyectos de investigación, obtendremos beneficios a nivel de usuario y a nivel técnico. Reduciremos el tiempo en consultas y acceso a la información, además de tener una mejor organización de los datos, por lo tanto, al reducir tiempo de trabajo de los empleados, gracias a la aplicación, éste tiempo se traduce en dinero por lo que económicamente resulta un beneficio a largo plazo.

Como conclusión, podemos decir que el proyecto es viable, ya que cumplimos con los requisitos deseados y poseemos la tecnología necesaria.

### **3. ANÁLISIS**

#### **3.1 INTRODUCCIÓN**

En el análisis se examina el problema para llegar a una solución. Se observan las carencias del actual sistema, se estudia cómo poder mejorar esas carencias. Gracias al análisis, se podrá diseñar y posteriormente construir una solución.

Con un estudio exhaustivo del problema podremos presentar un conjunto de especificaciones funcionales y no funcionales que la aplicación deberá tener. También se analizarán aquel conjunto de declaraciones que identifican atributos, capacidades, características y/o cualidades que necesitará cumplir el sistema para que tenga valor y utilidad para el usuario, es decir, los requerimientos.

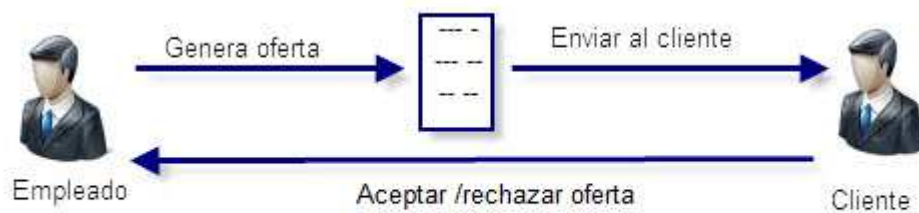
Por otro lado, también se estudian las opciones tecnológicas de las que disponemos y que se nos ofrecen para poder cumplir con los requerimientos. Algo importante a tener en cuenta para poder desarrollar el proyecto con éxito.

#### **3.2 DOMINIO DEL PROBLEMA**

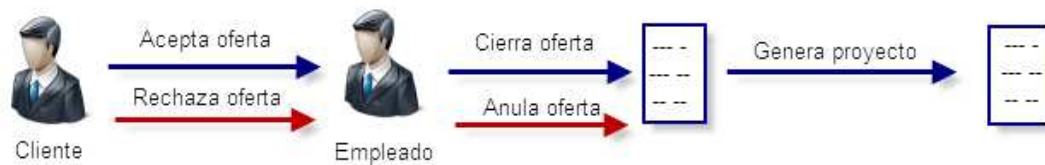
En este punto identificaremos los conceptos más importantes que necesitamos entender para poder llegar a la solución.

A continuación se explica la forma en la que se gestionan los proyectos en el centro SNiBA.

Primeramente, un empleado del centro genera una oferta la cual es enviada al cliente, el cliente ha de aceptar esa oferta o rechazar esa oferta. La siguiente imagen ilustra cómo se genera una oferta.

**Figura 3 Generar oferta**

Si la oferta es aceptada por el cliente, el empleado cambia el estado de la oferta a cerrada y se genera un proyecto, por el contrario si el cliente la rechaza, el empleado anula la oferta por lo que no se generará un nuevo proyecto. La siguiente imagen ilustra el funcionamiento de este procedimiento.

**Figura 4 Aceptación oferta**

En este procedimiento nos encontramos con algunas de las entidades más importantes, la entidad usuario que será el empleado que utilice la aplicación, la entidad cliente, la entidad oferta y la entidad proyecto.

### 3.3 REQUERIMIENTOS FUNCIONALES

#### Administración y gestión de la aplicación

Para el buen funcionamiento de la aplicación y que ésta cumpla con los objetivos deseados, se necesitan algunas funcionalidades para poder gestionar la aplicación, funcionalidades a las que no todo el mundo puede tener acceso y que dependerá del rol que se tenga asignado en el sistema.

Esas funcionalidades son las que se enumeran a continuación:

- Poder dar acceso a la aplicación a otros usuarios y asignarles un perfil.
- Poder editar, añadir o eliminar aquellos datos que se muestran por defecto pero que un futuro pueden variar, como por ejemplo los estados en los que se encuentran los proyectos o los tipos de proyectos que existen.
- Poder añadir un idioma más a la aplicación para que ésta se muestre en ese nuevo idioma.
- Poder eliminar o modificar alguna o algunas palabras de algún idioma en el que se muestre la página, o incluso eliminar un o más idiomas.
- Crear copias de seguridad de la base de datos y poder restaurar esas copias cuando se deseé.

### **Uso de la aplicación**

Existen funcionalidades que serán comunes para todos los usuarios, es decir, no importará el rol o el perfil que se tenga asignado para poder tener acceso a ellas, a diferencia de las funcionalidades de gestión de la aplicación, ya mencionadas anteriormente. Esas funcionalidades son las siguientes:

- Acceder a la aplicación.
- Visualizar los datos de las fichas de ofertas o proyectos en los que el usuario esté trabajando.
- Visualizar datos de clientes.
- Visualizar datos de contactos.

Por otro lado, existen usuarios que podrán crear, modificar o eliminar datos de las fichas de ofertas o proyectos en los que aparezcan como responsables.

Para poder determinar de qué forma se le ha de dar uso a la aplicación, primero hay que especificar cuál es la forma en que el centro SNiBA gestiona sus proyectos. Para entender mejor la manera en que trabajan, el centro nos ha facilitado información de la actual herramienta de trabajo que utilizan. Las siguientes imágenes, pertenecen a la actual aplicación.



La Figura 5 es una captura de pantalla del actual formulario que utilizan en SNIiBA para ingresar los datos de las ofertas.

Figura 5 Ficha oferta

La siguiente imagen es el formulario de proyectos.

**GQ094: FORMULARI**

DEPARTAMENT DE CIÈNCIA ANIMAL I DELS ALIMENTS

**CAA 104: CONTROL DE PROYECTOS (CONVENIS)**

CODI OFERTA: 000000 CODI INTERN: 2010-38-MAR

CODI conveni: DATA DE SIGNATURA OFERTA:

CENTRE DE COST: IMPORT:

TÍTOL PROJECTE: Desarrollar métodos de cuantificación y mejora del bienestar de los animales del Zoológico de Barcelona

CLIENT: PLANIFICACIÓ:

RESPONSABLE: SUBSTITUT: TÈCNIC:

DATA D'INICI: DATA DE FI (prevista):

COMENTARIS:

TIPUS DE PROJECTE: Projecte Especific

DOCUMENTACIÓ:

**ENTREGADA** Dates:

**REBUDA** Dates:

Dates informes parcials: Data informe final:

PNT ASSOCIAT:

REUNIONS:

Buttons: Editar (codi intern), Editar (codi oferta), Vista previa (CAA104)

Figura 6 Ficha proyecto

La Figura 7 es el formulario de clientes, mediante este formulario SNIIBA agrega nuevos clientes al sistema.



Microsoft Access - [ENTRADA CLIENTES]

Archivo Edición Ver Insertar Formato Registros Herramientas Ventana ? PDF de Adobe

MS Sans Serif 8

**ENTRADA DADES CLIENTS**

CLIENT

CLIENT: adreça

CLIENT: persona de contacte

CLIENT: tel\_fax\_altres

Figura 7 Ficha cliente

El centro realiza proyectos de investigación sobre nutrición, manejo y bienestar animal ofreciendo a sus clientes asesoramiento, gestión y desarrollo en estos proyectos. Por lo tanto, primeramente se genera una ficha de oferta al cliente, en la cual se especifica, entre otras cosas, el tipo de proyecto, a qué cliente va dirigida esa oferta y quiénes son los responsables y técnicos que trabajarán en ese proyecto. Más tarde esa oferta deberá ser validada, en ese caso, se generará el proyecto. En la Figura 8 se explica el funcionamiento a seguir en caso de ser anulada o aceptada la oferta.

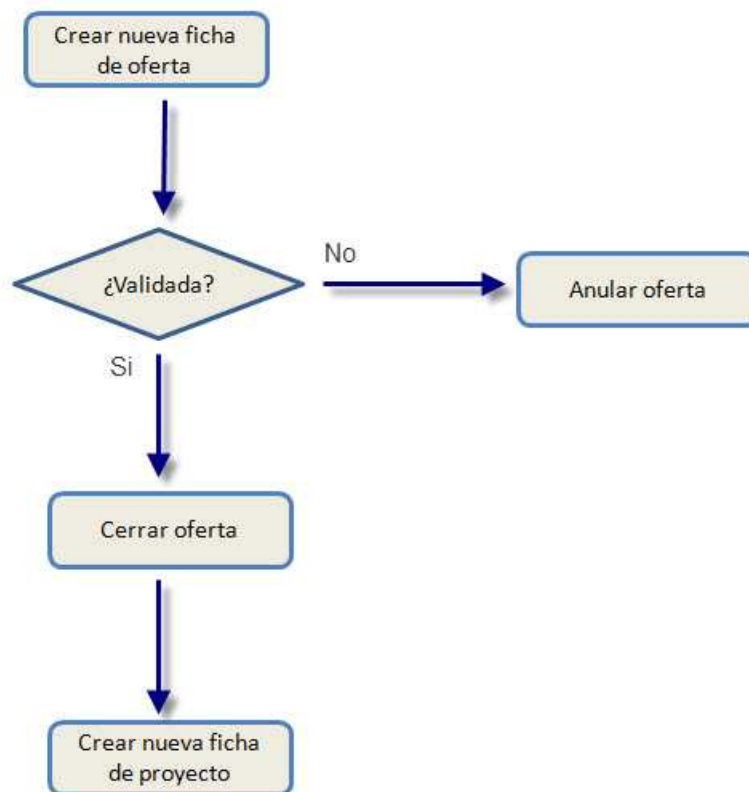


Figura 8 Diagrama crear oferta

La aplicación permite crear esa ficha de oferta a la cual se le asigna un código y un estado, que puede ser cerrada, pendiente de respuesta o anulada. El estado podrá ser modificado en un futuro y en el momento en que la oferta tenga el estado de cerrada, entonces se genera el proyecto.

Una vez se ha generado el proyecto, en él se especifica, entre otras cosas el código de la oferta, la fecha en que se firma el proyecto, el centro de coste, importe, fecha de inicio y fin del proyecto, el cliente y quienes trabajarán en el proyecto.

Por otro lado, durante el periodo en que se realiza el proyecto, el centro deberá hacer un seguimiento y tener una serie de reuniones con el cliente. La aplicación permite registrar esas reuniones pudiendo guardar, la fecha y hora de la reunión, el lugar y los asistentes. Además, puede que sea necesario adjuntar documentos relacionados con ese proyecto por lo que el sistema permite agregar más de un documento a un proyecto.

A continuación se muestra el diagrama de casos de uso.

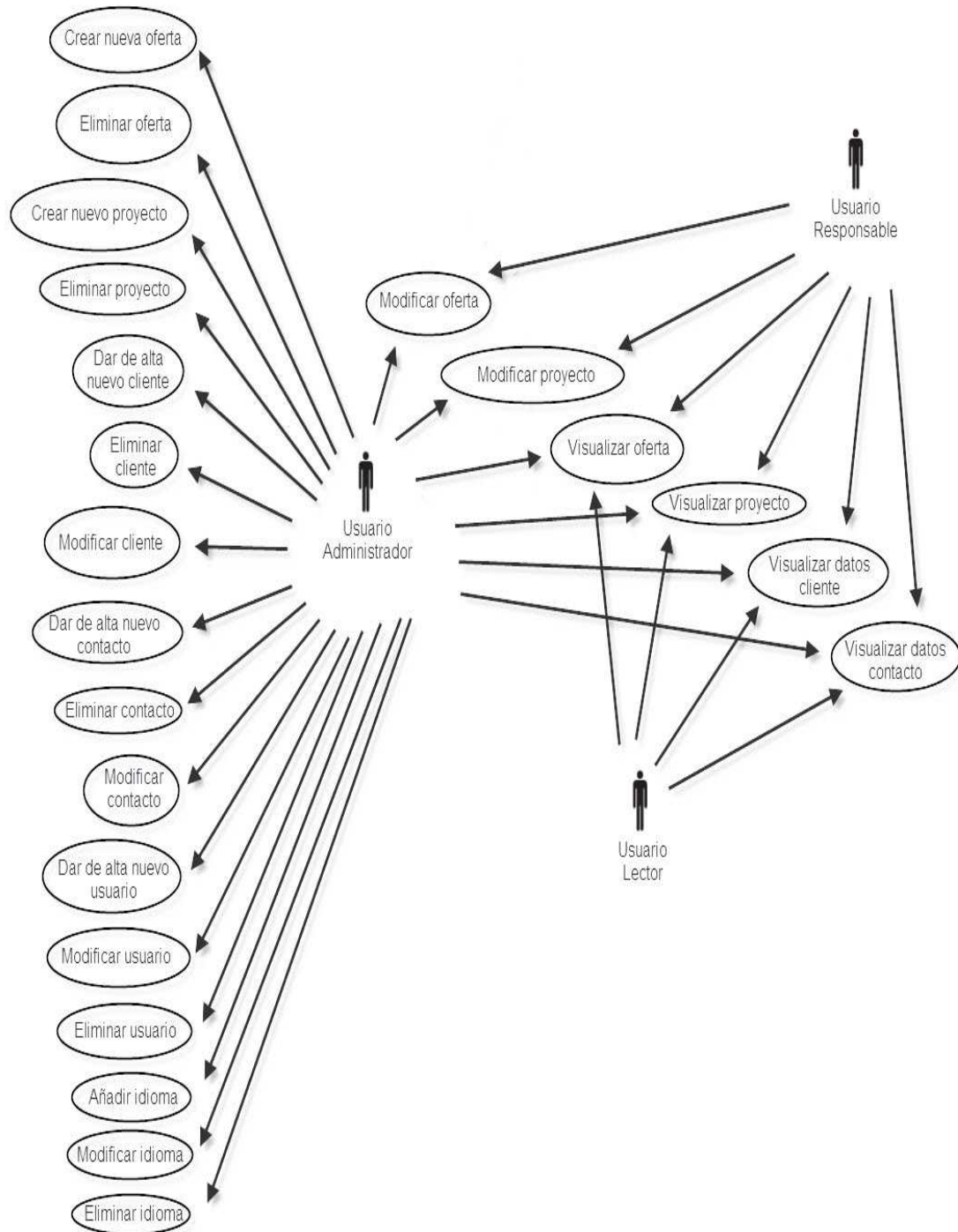


Figura 9 Casos de uso

### 3.4 REQUERIMIENTOS NO FUNCIONALES

También existen un conjunto de requerimientos relacionados con el entorno en el que ha de funcionar la aplicación y las características necesarias para cumplir los objetivos. Estos requerimientos son los que ayudarán a hacer un mejor uso de la aplicación:

Los requerimientos no funcionales son los siguientes:

- **Facilidad de uso:** El sistema debe ser de fácil uso para los usuarios de la aplicación, además la navegación ha de ser rápida y entendible con una interfaz amigable.
- **Disponibilidad:** La aplicación debe estar disponible al 100%, sobre todo durante el horario laboral de SNiBA, se ha de poder acceder a ella desde cualquier ordenador, además de poder funcionar independientemente del navegador que se utilice.
- **Escalabilidad:** El sistema debe ser construido de manera que las nuevas funcionalidades y requerimientos relacionados puedan implementarse afectando al código existente de la menor manera posible. Además debe permitir poder eliminar funcionalidades existentes de la misma forma.
- **Validación de información:** El sistema debe validar automáticamente la información contenida en los formularios de ingreso. En el proceso de validación de la información, se deben tener en cuenta aspectos tales como obligatoriedad de campos o longitud de caracteres permitida por campo.

### 3.5 MARCO TECNOLÓGICO

Para determinar cómo se ha de hacer la implementación y el diseño se ha hecho un análisis tecnológico. Se analizan las diferentes herramientas con las que podemos trabajar y de las que disponemos teniendo en cuenta que ha de ser una aplicación accesible remotamente y que permita el acceso concurrente de diferentes usuarios. Por lo tanto crear la aplicación en entorno web parece ser la mejor opción para cubrir estas necesidades.

Existen diferentes lenguajes de programación para el desarrollo web. ASP.Net, ASP, PHP, Python, entre otros.

**ASP.NET** Este es un lenguaje comercializado por Microsoft, y usado por programadores para desarrollar entre otras funciones, sitios web. ASP.NET es el sucesor de la tecnología ASP. Para su funcionamiento de las páginas se necesita tener instalado IIS con el Framework .Net.

**PHP** es un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas web dinámicas, embebidas en páginas HTML y ejecutadas en el servidor. PHP no necesita ser compilado para ejecutarse. Para su funcionamiento necesita tener instalado Apache o IIS con las librerías de PHP. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas.

**Python**, es un lenguaje de programación multiparadigma, lo cual fuerza a que los programadores adopten por un estilo de programación particular:

- Programación orientada a objetos.
- Programación estructurada.
- Programación funcional.

**ASP** no necesita ser compilado para ejecutarse. Es una tecnología dinámica funcionando del lado del servidor, lo que significa que cuando el usuario solicita un documento ASP, las instrucciones de programación dentro del script son ejecutadas para enviar al navegador únicamente el código HTML resultante. Esto se traduce en seguridad ya que el usuario solo tiene acceso a la página resultante.

Existen varios lenguajes que se pueden utilizar para crear páginas ASP. El más utilizado es VBScript, nativo de Microsoft. El código ASP puede ser insertado junto con el código HTML. Algunas de las ventajas de asp es que es un lenguaje fácil de aprender y con instrucciones sencillas.

## **Sistema gestor de base de datos**

A parte de decidir que lenguaje utilizaremos para implementar la aplicación, se necesita una base de datos para almacenar toda la información y posteriormente poder acceder a ella para la modificación o eliminación de los datos.

Podemos elegir entre varios gestores de bases de datos. Se han evaluado dos posibles opciones, MySQL y SQL Server. Existen algunas diferencias entre los dos pero la decisión de escoger uno u otro dependerá de factores como por ejemplo el uso que le vayamos a dar.

MySQL es uno de los gestores de base de datos más usado en el mundo, es excelente para su uso web y tiene la ventaja de ser gratuito, por lo que es ideal para proyectos de gran envergadura en los que se busca un bajo coste, además se puede conectar a diversos sistemas como Windows, Linux o Mac OS. Por otro lado, se puede recibir apoyo y soporte por parte de los propios profesionales que desarrollan MySQL, además de recibir automáticamente actualizaciones de software.

SQL Server es otro gestor de base de datos propiedad de Microsoft, a diferencia de MySQL no es de código abierto pero es útil para desarrollos de gran envergadura y soporta accesos simultáneos sin problemas. Una de las ventajas que aporta SQL Server frente a MySQL es que permite el uso de Triggers y la principal desventaja es que su uso es bajo licencia, aunque existe la versión SQL Server Express gratuita.



## 4.DISEÑO

### 4.1 INTRODUCCIÓN

Según los requerimientos funcionales anteriormente descritos, se definirán los módulos de los que está compuesta la aplicación. A partir del análisis de los requerimientos se ha de diferenciar las funcionalidades que han de estar disponibles dependiendo del usuario que acceda a la aplicación.

### 4.2 ARQUITECTURA DE LA APLICACIÓN

La arquitectura de la aplicación, estará dividida en capas que normalmente serán tres: una capa que servirá para guardar los datos (base de datos), una capa para centralizar la lógica de negocio (modelo) y por último una interfaz gráfica que facilite al usuario el uso del sistema.

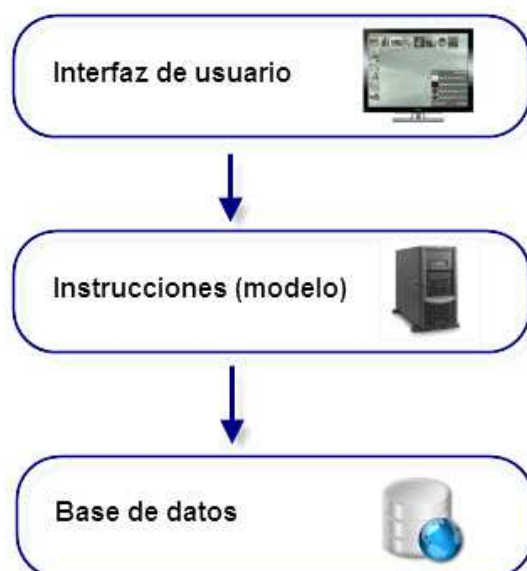


Figura 10 Arquitectura aplicación

### 4.3 TECNOLOGÍAS Y HERRAMIENTAS A UTILIZAR

Ya que se dispone de varias tecnologías para desarrollar la aplicación, tal y como se ha descrito en el punto 3.5 MARCO TECNOLÓGICO, de las cuales todas parecen óptimas, finalmente se decide utilizar ASP ya que se dispone de más experiencia y conocimiento en este lenguaje, lo que hace disminuir el tiempo de aprendizaje y tiempo de desarrollo, lo cual se traduce en reducción de costes.

Entre las posibles opciones de gestores de base de datos que existen, se escoge SQL Server que parece ser la mejor opción ya que la aplicación será implementada en ASP y a la hora de escoger este lenguaje de programación se recomienda utilizar SQL Server ya que es el sistema de base de datos recomendado para las aplicaciones ASP sobre servidores Windows.

### 4.4 PERFILES DE USUARIOS

Existen diferentes perfiles de usuario que se les podrá asignar a cada usuario dependiendo del cargo que tengan en el equipo de proyecto del centro SNiBA.

El centro divide los cargos de los empleados en diferentes niveles, los niveles son los siguientes:

	<b>Cargo en SNiBA</b>	<b>Perfil usuario asignado en la aplicación</b>
<b>Nivel A</b>	Dirección de SNiBA Comisión ejecutiva Directores de proyectos.	Administrador
<b>Nivel B</b>	Responsables de proyectos Responsables de calidad Promotor.	Responsable
<b>Nivel C</b>	Técnicos de proyectos Investigadores en formación.	Lector

Cada cargo que aparece en la tabla anterior, tiene un perfil de usuario asignado en la aplicación, a continuación, en la siguiente tabla se describen los perfiles de usuario y las funciones de cada uno.

Perfiles usuario	Funciones
<b>Administrador</b>	<ul style="list-style-type: none"><li>• Funciones de administración y gestión.<ul style="list-style-type: none"><li>○ Crear y dar acceso a nuevos usuarios.</li><li>○ Asignarles un perfil a los usuarios.</li><li>○ Crear, modificar y eliminar los idiomas en los que se muestra la web.</li></ul></li></ul>
<b>Responsable</b>	<ul style="list-style-type: none"><li>• Estos usuarios podrán modificar y visualizar únicamente las fichas de oferta y de proyectos donde aparezca su nombre como RESPONSABLE DE PROYECTO.</li></ul>
<b>Lector</b>	<ul style="list-style-type: none"><li>• Estos usuarios podrán visualizar únicamente las fichas de ofertas y proyectos donde aparezca su nombre como:<ul style="list-style-type: none"><li>○ TÉCNICOS DE PROYECTO o como</li><li>○ DIRECTOR DE PROYECTO.</li></ul>(pero no se les permitirá modificar)</li></ul>

## 4.5 DIAGRAMA ENTIDAD RELACIÓN

Una vez planteada la aplicación y todas las funcionalidades que esta tendrá, se analizará que información se tendrá que guardar y se ha de estudiar la mejor manera de hacerlo. Para ello se desarrolla el diagrama de entidad relación y en él se distinguen cuatro entidades importantes:

- Oferta
- Proyecto
- Cliente
- Contacto
- Usuario

A continuación se muestra el diagrama entidad-relación.

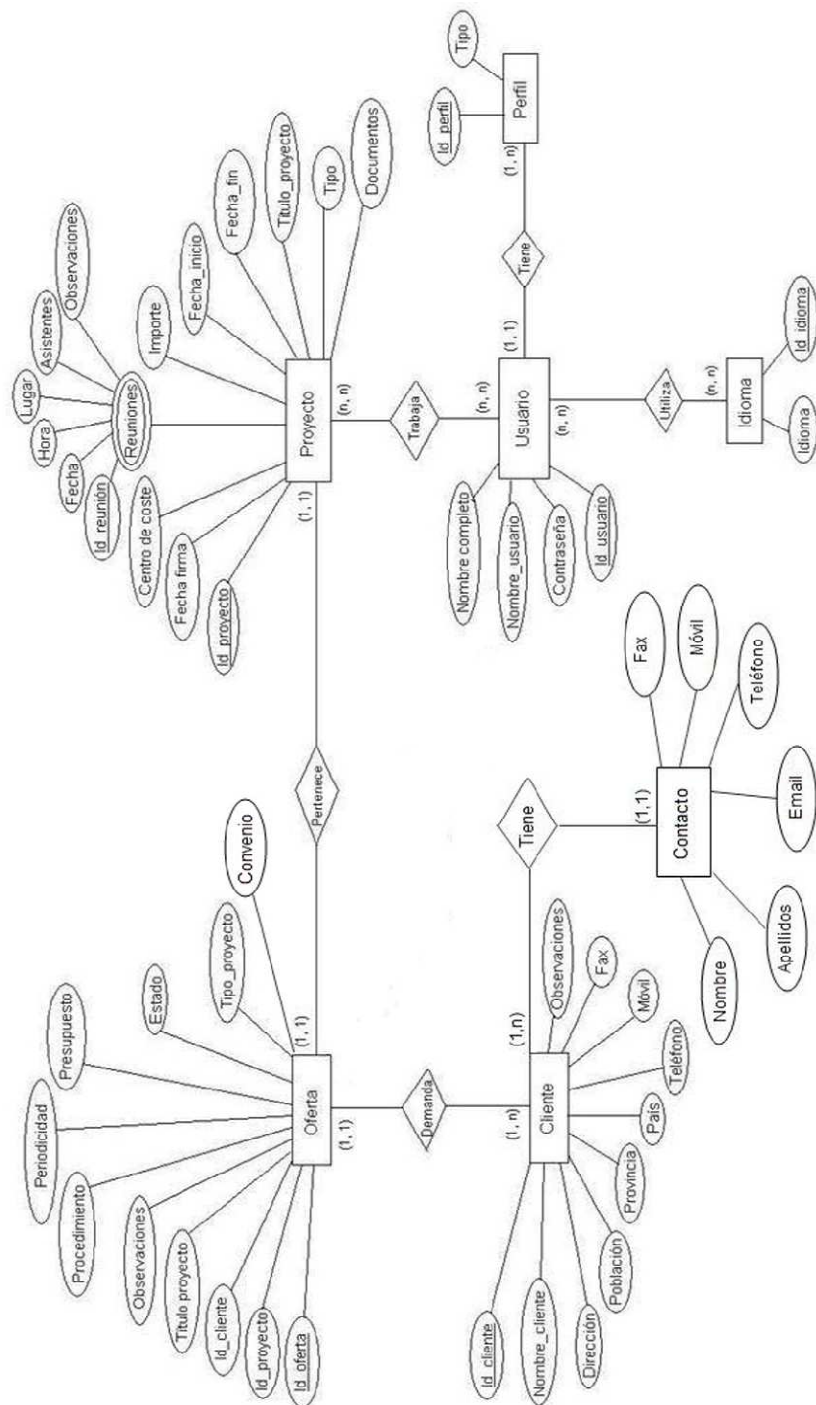


Figura 11 Diagrama entidad relación

A continuación se explica que representa cada entidad.

**La entidad oferta** representa la ficha de oferta que el centro previamente envía al cliente, representado por la entidad cliente en la Figura 11 Diagrama entidad relación. Cuando el cliente recibe la oferta, éste decide si la acepta o no. Si la oferta es aceptada por el cliente, entonces, se genera una ficha de proyecto la cual es representada por la entidad proyecto. Cada proyecto tiene asociada una única oferta, ya que si el cliente no la acepta, ésta es anulada y se genera otra nueva.

**La entidad convenio** representa un contrato que se firma entre el cliente y la rectora de la UAB cuando el proyecto es de larga duración, por lo que no todos los proyectos tienen convenio.

**La entidad usuario** simboliza a las personas que tendrán acceso a la aplicación y la entidad perfil será el rol que esas personas tendrán dentro de la aplicación.

Por último nos encontramos la **entidad idioma**, la cual representa el lenguaje en que cada usuario escogerá para visualizar la web.

Una vez explicado el significado de cada entidad, se detalla la relación que hay entre ellas.

Una oferta pertenece sólo a un cliente, pero un cliente puede tener varias ofertas. Las ofertas generan la ficha de proyecto, por lo que un proyecto tendrá asociada una única oferta, ya que, como se ha explicado anteriormente, el cliente ha de aceptar esa oferta, si no es aceptada, se anula y se crea una nueva. Por otro lado, los usuarios de la aplicación serán los que trabajen en los proyectos, así que a un usuario se le pueden asignar varios proyectos y cada proyecto tendrá más de un usuario trabajando en él, por lo que es una relación de muchos a muchos.

## 4.6 MODELO DE DATOS

A la hora de diseñar la base de datos se ha decidido que el primer campo de cada tabla sea un campo autoincremental único. De esta manera se podrá identificar cada registro por este campo, el tipo de datos que se le asigna siempre a este campo, es del tipo

Numeric(18,0), este tipo de datos indica que es un campo numérico, donde el “18” indica el número de dígitos que tendrá la parte entera y el “0” indica el número de dígitos que tendrá la parte decimal. Los campos que no necesitan ser numéricos se les ha asignado el tipo de datos “Nvarchar()”, este tipo de datos se utiliza para almacenar caracteres de tipo Unicode, ya que la aplicación será implementada en varios idiomas, se recomienda utilizar este tipo de datos para almacenar datos escritos en diferentes lenguajes como por ejemplo japonés.

A continuación se detallan todas las tablas de la base de datos y la información que se guarda en cada una de ellas, así como el tipo de datos asignado a cada campo.

### Tablas de base de datos

**TBL\_CLIENTES:** En esta tabla se guarda toda la información referente a los clientes y corresponde a la entidad **cliente**.

PK / FK	Campo	Tipo de datos	Descripción del campo
<b>PK</b>	Cmp_cliente_id	Numeric(18,0)	Identificador del cliente.
	Cmp_cliente_nombre	Nvarchar(100)	Nombre del cliente.
	Cmp_cliente_dirección	Nvarchar(100)	Dirección del cliente.
	Cmp_cliente_población	Nvarchar(100)	Población del cliente.
	Cmp_cliente_provincia	Nvarchar(100)	Provincia del cliente.
	Cmp_cliente_pais	Nvarchar(100)	País del cliente.
	Cmp_cliente_telf	Nvarchar(50)	Teléfono del cliente.
	Cmp_cliente_movil	Nvarchar(50)	Movil del cliente.
	Cmp_cliente_fax	Nvarchar(50)	Fax del cliente.
	Cmp_cliente_email	Nvarchar(100)	Email del cliente.
	Cmp_cliente_observaciones	Nvarchar(2000)	Campo donde se puede añadir información extra sobre cliente.

**TBL\_CONTACTOS:** Tabla donde se guardan los datos de los contactos y corresponde a la entidad **contacto**.

PK / FK	Campo	Tipo de datos	Descripción del campo
<b>PK</b>	Cmp_contacto_id	Numeric(18,0)	Identificador del contacto.
<b>FK</b>	Cmp_contacto_idcliente	Numeric(18,0)	Identificador del cliente al cual pertenece el contacto.
	Cmp_contacto_nombre	Nvarchar(50)	Nombre del contacto.
	Cmp_contacto_apellidos	Nvarchar(100)	Apellidos del contacto.
	Cmp_contacto_téfono	Nvarchar(50)	Teléfono del contacto.
	Cmp_contacto_movil	Nvarchar(50)	Móvil del contacto.
	Cmp_contacto_mail	Nvarchar(100)	Email del contacto.
	Cmp_contacto_fax	Nvarchar(50)	Fax del contacto.

**TBL\_ESTADOS:** Tabla donde se guardan los diferentes estados en los que se puede encontrar una oferta.

PK / FK	Campo	Tipo de datos	Descripción del campo
<b>PK</b>	Cmp_estado_id	Numeric(18,0)	Identificador del estado.
	Cmp_estado	Nvarchar(500)	Nombre del estado.

**TBL\_TIPOS\_PROYECTOS:** Tabla donde se guardan los datos referentes las diferentes tipologías de proyectos.

PK / FK	Campo	Tipo de datos	Descripción del campo
<b>PK</b>	Cmp_tipo_proy_id	Numeric(18,0)	Identificador del tipo de proyecto.
	Cmp_tipo_proy_nombre	Nvarchar(500)	Nombre del tipo de proyecto.

**TBL\_OFERTAS:** En la siguiente tabla se guardará toda la información que corresponde a la ficha de oferta de un proyecto. Esta tabla corresponde a la entidad **oferta**.

PK / FK	Campo	Tipo de datos	Descripción del campo
<b>PK</b>	Cmp_oferta_id	Numeric(18,0)	Identificador de la oferta.
	Cmp_oferta_codigo	Numeric(18,0)	Código de la oferta.
<b>FK</b>	Cmp_oferta_idcliente	Numeric(18,0)	Id del cliente al cual pertenece la oferta.

	Cmp_oferta_titulo_proy	Nvarchar(500)	Título del proyecto.
	Cmp_oferta_procedimiento	Nvarchar(max)	Procedimiento a seguir.
	Cmp_oferta_fecha_inicio	Nvarchar(50)	Fecha en la que se prevé que se iniciará el proyecto.
	Cmp_oferta_fecha_fin	Nvarchar(50)	Fecha en la que se prevé que finalizará el proyecto.
	Cmp_oferta_presupuesto	Decimal(18,0)	Precio estimado que costará el proyecto.
<b>FK</b>	Cmp_oferta_responsable_id	Numeric(18,0)	Id del empleado que será el responsable del proyecto.
<b>FK</b>	Cmp_oferta_sustituto_id	Numeric(18,0)	Id del empleado que sustituirá al responsable en el caso que sea necesario.
<b>FK</b>	Cmp_oferta_estado	Numeric(18,0)	Estado en el que se encuentra la oferta.
	Cmp_oferta_genera_convenio	Nvarchar(50)	Este campo indica si la oferta ha generado un convenio.
<b>FK</b>	Cmp_oferta_tipo_proyecto	Numeric(18,0)	Tipo del proyecto
	Cmp_oferta_observaciones	Nvarchar(2000)	Información extra sobre la oferta.

**TBL\_PROYECTOS:** Tabla en la que se guardará toda la información referente a la ficha de proyecto. Corresponde a la entidad **proyecto**.

PK / FK	Campo	Tipo de datos	Descripción del campo
<b>PK</b>	Id proyecto	Numeric(18,0)	Identificador del proyecto
<b>FK</b>	Cmp_proj_idoferta	Numeric(18,0)	Identificador de la oferta a la que pertenece el proyecto.
	Cmp_proj_codigo_proy	Nvarchar(500)	Código del proyecto compuesto por los dos últimos días del año y el identificador del proyecto.
	Cmp_proj_data_signatura	Nvarchar(50)	Fecha en la que se firma el proyecto.
	Cmp_proj_centre_cost	Nvarchar(50)	Centro de coste al cual pertenece el proyecto
	Cmp_proj_import	Nvarchar(50)	Precio en € del coste del proyecto.
	Cmp_proj_data_inici	Nvarchar(50)	Fecha de inicio del proyecto
	Cmp_proj_data-fi	Nvarchar(50)	Fecha de fin del proyecto.



	Cmp_proj_título	Nvarchar(500)	Título del proyecto.
<b>FK</b>	Cmp_proj_tipo	Numeric(18,0)	Tipo de proyecto.
<b>FK</b>	Cmp_proj_client	Numeric(18,0)	Identificador del cliente del proyecto.
<b>FK</b>	Cmp_proj_responsable	Numeric(18,0)	Id del empleado que será el responsable del proyecto.
<b>FK</b>	Cmp_proj_sustituto	Numeric(18,0)	Id del empleado que sustituirá al responsable en el caso que sea necesario.
<b>FK</b>	Cmp_proj_tecnico	Numeric(18,0)	Id del empleado que trabajará en el proyecto como técnico.
	Cmp_proj_observaciones	Nvarchar(2000)	Información extra sobre el proyecto.

**TBL\_REUNIONES:** En esta tabla se guarda la información necesaria a las reuniones que están asociadas a un proyecto.

PK / FK	Campo	Tipo de datos	Descripción del campo
<b>PK</b>	Cmp_reunion_id	Numeric(18,0)	Identificados de la reunión.
	Cmp_reunion_fecha	Nvarchar(50)	Fecha en la que se realiza la reunión.
	Cmp_reunion_hora	Nvarchar(50)	Hora en la que se realiza la reunión.
	Cmp_reunion_lugar	Nvarchar(500)	Lugar donde se realizará la reunión.
	Cmp_reunion_comentarios	Nvarchar(2000)	Información extra sobre la reunión.
<b>FK</b>	Cmp_reunion_idproj	Numeric(18,0)	Identificador del proyecto al que pertenece la reunión.
	Cmp_reunion_asistentes	Nvarchar(500)	Asistentes a la reunión.

**TBL\_EMPLEADOS:** Se guardan los datos de los empleados. Corresponde a la entidad **usuario**.

PK / FK	Campo	Tipo de datos	Descripción del campo
<b>PK</b>	Cmp_emp_id	Numeric(18,0)	Identificador del empleado.
	Cmp_emp_nombre	Nvarchar(500)	Nombre del empleado.
	Cmp_emp_apellidos	Nvarchar(500)	Apellidos del empleado.
	Cmp_emp_dirección	Nvarchar(500)	Dirección del empleado.
	Cmp_emp_población	Nvarchar(500)	Población de residencia del empleado.

Cmp_emp_provincia	Nvarchar(500)	Provincia de residencia del empleado.
Cmp_emp_CP	Nvarchar(50)	Código postal del empleado.
Cmp_emp_telefono	Nvarchar(50)	Teléfono del empleado.
Cmp_emp_email	Nvarchar(500)	Email del empleado.
Cmp_emp_movil	Nvarchar(50)	Número del móvil del empleado.

**TBL\_PERFILES:** En la tabla perfiles se guarda el listado de los diferentes perfiles que se pueden encontrar en la aplicación, cada perfil irá asociado a un empleado. Esta tabla corresponde a la entidad **perfil**.

PK / FK	Campo	Tipo de datos	Descripción del campo
<b>PK</b>	Id_perfil	Numeric(18,0)	Identificador del perfil.
	Cmp_perfil_tipo	Nvarchar(50)	Tipo de perfil.

**TBL\_USERS:** En la tabla users se guardará toda la información necesaria de cada usuario que tenga acceso a la aplicación, nombre de usuario, contraseña, perfil o asignado.

PK / FK	Campo	Tipo de datos	Descripción del campo
<b>PK</b>	Id_user	Numeric(18,0)	Identificador del usuario
	Cmp_user_name	Nvarchar(50)	Alias con el que se validará en la aplicación.
	Cmp_user_pwd	Nvarchar(100)	En este campo se guarda la contraseña codificada, con la que el usuario accederá a la aplicación
<b>FK</b>	Cmp_idperfil	Numeric(18,0)	Identificador del perfil que tiene este usuario
<b>FK</b>	Cmp_idemp	Numeric(18,0)	Identificador del empleado

**TBL\_IDIOMAS\_US:** Cuando un usuario accede a la aplicación, el usuario escogerá el idioma deseado para visualizar la web. En esta tabla se guarda el identificador del usuario que accedió y el idioma que escogió la última vez que entró a la aplicación. De esta forma, la próxima vez que el usuario acceda la web se le mostrará en esa lengua.

PK / FK	Campo	Tipo de datos	Descripción del campo
<b>PK</b>	Cmp_user_id	Numeric(18,0)	Identificador del usuario
<b>FK</b>	Cmp_leng_id	Numeric(18,0)	Identificador del idioma

Cmp_leng	Nvarchar(100)	Nombre del idioma
----------	---------------	-------------------

**TBL\_IDIOMAS:** En la tabla idiomas se guarda cada palabra o frase que aparece en la web para poder hacer la traducción a diferentes lenguas. Corresponde a la entidad **idioma**.

PK / FK	Campo	Tipo de datos	Descripción del campo
<b>PK</b>	Cmp_idioma_id	Numeric(18,0)	Identificador del idioma.
	Cmp_idioma	Nvarchar(100)	Nombre del idioma.
	Cmp_i_idioma	Nvarchar(100)	Palabra "idioma".
	CMP_I_FORMULARIO	Nvarchar(100)	Palabra "formulario".
	CMP_I_PROYECTO	Nvarchar(100)	Palabra "proyecto".
	CMP_I_BUSCAR_PROJ	Nvarchar (500)	Frase "buscar proyecto".
	CMP_I_NUEVO_PROJ	Nvarchar (500)	Frase "nuevo proyecto".
	CMP_I_CONTACTOS	Nvarchar(100)	Palabra "contactos".
	CMP_I_CONTACTO	Nvarchar(100)	Palabra "contacto".
	CMP_I_NUEVO_CONTACTO	Nvarchar (500)	Frase "nuevo contacto".
	CMP_I_BUSCAR_CONTACTO	Nvarchar (500)	Frase "buscar contacto".
	CMP_I_BUSCAR	Nvarchar(100)	Palabra "buscar".
	CMP_I_NOMBRE	Nvarchar(100)	Palabra "nombre".
	CMP_I_APELLIDOS	Nvarchar(100)	Palabra "apellidos".
	CMP_I_TELEFONO	Nvarchar(100)	Palabra "telefono".
	CMP_I_MOVIL	Nvarchar(100)	Palabra "movil".
	CMP_I_FAX	Nvarchar(100)	Palabra "fax".
	CMP_I_DIRECCION	Nvarchar(100)	Palabra "dirección".
	CMP_I_POBLACION	Nvarchar(100)	Palabra "población".
	CMP_I_PROVINCIA	Nvarchar(100)	Palabra "provincia".
	CMP_I_OBSERSERVACIONES	Nvarchar(100)	Palabra "observaciones".
	CMP_I_PAIS	Nvarchar(100)	Palabra "país".
	CMP_I_CODIGO_OFERTA	Nvarchar (500)	Frase "código oferta".
	CMP_I_CODIGO_PROYECTO	Nvarchar (500)	Frase "código proyecto".
	CMP_I_CENTRO_COSTE	Nvarchar (500)	Frase "centro coste".
	CMP_I_FECHA_FIRMA	Nvarchar (500)	Frase "fecha firma".
	CMP_I PRESUPUESTO	Nvarchar(100)	Palabra "presupuesto".
	CMP_I IMPORTE	Nvarchar(100)	Palabra "importe".
	CMP_I_FECHA_INICIO	Nvarchar (500)	Palabra "fecha inicio".
	CMP_I_FECHA_FIN	Nvarchar(100)	Palabra "fecha fin".
	CMP_I_TITULO_PROJ	Nvarchar (500)	Frase "título proyecto".
	CMP_I_TIPO_PROJ	Nvarchar (500)	Frase "tipo proyecto".
	CMP_I_CLIENTE	Nvarchar(100)	Palabra "cliente".
	CMP_I_CLIENTES	Nvarchar(100)	Palabra "clientes".
	CMP_I_NUEVO_CLIENTE	Nvarchar (500)	Frase "nuevo cliente".

CMP_I_BUSCAR_CLIENTES	Nvarchar (500)	Frase “buscar clientes”.
CMP_I_RESPONSABLE	Nvarchar(100)	Palabra “responsable”.
CMP_I_SUSTITUTO	Nvarchar(100)	Palabra “sustituto”.
CMP_I_TECNICO	Nvarchar(100)	Palabra “técnico”.
CMP_I_COMENTARIOS	Nvarchar(100)	Palabra “comentarios”.
CMP_I_DOCUMENTACION	Nvarchar(100)	Palabra “documentación”.
CMP_I_DOCUMENTO	Nvarchar (500)	Palabra “documento”.
CMP_I_FECHA_DOC	Nvarchar (500)	Frase “fecha documentación”.
CMP_I_REUNIONES	Nvarchar(100)	Palabra “reuniones”.
CMP_I_FECHA_REUNION	Nvarchar (500)	Frase “fecha reunión”.
CMP_I_HORA_REUNION	Nvarchar (500)	Frase “hora reunión”.
CMP_I_LUGAR_REUNION	Nvarchar (500)	Frase “lugar reunión”.
CMP_I_ASISTENTES	Nvarchar(100)	Palabra “asistentes”.
CMP_I_ESTADO	Nvarchar(100)	Palabra “estado”.
CMP_I_GENERA_CONVENIO	Nvarchar (500)	Frase “genera convenio”.
CMP_I_EDITAR	Nvarchar(100)	Palabra “editar”.
CMP_I_VISUALIZAR	Nvarchar(100)	Palabra “visualizar”.
CMP_I_MODIFICAR	Nvarchar(100)	Palabra “modificar”.
CMP_I_ELIMINAR	Nvarchar(100)	Palabra “eliminar”.
CMP_I_GUARDAR	Nvarchar(100)	Palabra “guardar”.
CMP_I_VISTA_PREVIA	Nvarchar (500)	Frase “vista previa”.
CMP_I_PNT_ASOCIADO	Nvarchar (500)	Frase “PNT asociado”.
CMP_I_CAMBIAR_IDIOMA	Nvarchar (500)	Frase “cambiar idioma”.
CMP_I_BORRAR_FORM	Nvarchar (500)	Frase “borrar formulario”.
CMP_I_ESCRIBIR_AST	Nvarchar (500)	Frase “Escribe * para visualizar todo”.
CMP_I_ADMINISTRAR	Nvarchar(100)	Palabra “administrar”.
CMP_I_CREAR_USUARIO	Nvarchar (500)	Frase “crear usuario”.
CMP_I_ANADIR_IDIOMA	Nvarchar (500)	Frase “añadir idioma”.
CMP_I_ANADIR	Nvarchar(100)	Palabra “añadir”.
CMP_I_ESCTRADUCCION	Nvarchar (500)	Frase “Escribe la traducción de cada palabra en el nuevo idioma que quieres añadir”.
CMP_I_NUEVO	Nvarchar(100)	Palabra “nuevo”.
CMP_I_ELEGIR	Nvarchar(100)	Palabra “elegir”.
CMP_I_SELECT_DOC	Nvarchar(100)	Palabra “Selecciona un documento”.
CMP_I_CAMPO_VACIO	Nvarchar (500)	Frase “Revisa los campos en rojo”.
CMP_I_OFERTA	Nvarchar(100)	Palabra “oferta”.
CMP_I_OFERTAS	Nvarchar(100)	Palabra “ofertas”.
CMP_I_PROYECTOS	Nvarchar(100)	Palabra “proyectos”.
CMP_I_NUEVA	Nvarchar(100)	Palabra “nueva”.
CMP_I_GENERA	Nvarchar(100)	Palabra “genera”.

CMP_I_CONVENIO	Nvarchar(100)	Palabra “convenio”.
CMP_I_CP	Nvarchar (500)	Frase “código postal”.
CMP_I_EMAIL	Nvarchar(100)	Palabra “email”.
CMP_I_USUARIO	Nvarchar(100)	Palabra “usuario”.
CMP_I_PERFIL	Nvarchar(100)	Palabra “perfil”.
CMP_I_SELECCIONAR	Nvarchar(100)	Palabra “seleccionar”.
CMP_I_CONTRASENA	Nvarchar(100)	Palabra “contraseña”.
CMP_I_HOLA	Nvarchar(100)	Palabra “hola”.
CMP_I_BIENVENIDO	Nvarchar(100)	Palabra “bienvenido”.
CMP_I_INICIO	Nvarchar(100)	Palabra “inicio”.
CMP_I_SIGUIENTE	Nvarchar(100)	Palabra “siguiente”.
CMP_I_ANTERIOR	Nvarchar(100)	Palabra “anterior”.
CMP_I_PAGINAS	Nvarchar(100)	Palabra “páginas”.
CMP_I_ALIAS	Nvarchar(100)	Palabra “alias”.
CMP_I_DATOS_ACCESO	Nvarchar(500)	Frase “Datos de acceso”.
CMP_I_PREG_ELIMINAR	Nvarchar(500)	Frase “¿Desea eliminar?”.
CMP_I_ENTRAR	Nvarchar(100)	Palabra “entrar”.
CMP_I_VOLVER	Nvarchar(100)	Palabra “volver”.
CMP_I_ALERTA	Nvarchar(100)	Palabra “alerta”.
CMP_I_NO_TIENES_ACCESO	Nvarchar(500)	Mensaje de alerta.
CMP_I_ALERTA_SESION	Nvarchar(500)	Mensaje de alerta “No ha iniciado sesión”.
CMP_I_INICIAR_SESION	Nvarchar(500)	Frase “Iniciar sesión”.
CMP_I_DATOS_ELIMINADOS	Nvarchar(500)	Frase “Datos eliminados correctamente”.
CMP_I_DATOS_GUARDADOS	Nvarchar(500)	Frase “Datos guardados correctamente”.

## 4.7 INTERFAZ

A la hora de diseñar la interfaz de usuario, se ha tenido en cuenta la usabilidad, es decir que la navegación por la aplicación sea fácil, rápida e intuitiva.

Por otro lado, ya que la aplicación se utilizará durante muchas horas seguidas, se han escogido colores suaves para diseñar la interfaz. Las tonalidades escogidas no sobrecargan la vista. El tipo de letra utilizado tampoco tiene muchas variaciones importantes de fuente, todo esto hace que se conserve la homogeneidad visual de las

pantallas. La Figura 12 muestra la gama de colores escogida para la visualización de la web.



**Figura 12 Gama de colores**

Además, para que la aplicación sea de fácil uso, y entendible para el usuario. Se han añadido iconos gráficos, los cuales tienen funcionalidad. Por ejemplo, si se desea editar una ficha de oferta, sólo con clicar en el icono que representa un lápiz, nos redirecciona a la página de edición de la oferta. En el siguiente capítulo se detallan los iconos que aparecen en la aplicación y su significado.

## 5.IMPLEMENTACIÓN

### 5.1 INTRODUCCIÓN

En la implementación se explica el resultado final de la aplicación.

Una vez realizado el análisis de la aplicación, se diseñó la estructura que debería tener el sistema para conseguir los objetivos definidos, así como una coherencia a la hora de trabajar con los datos. Es importante que estén estructurados los ficheros de manera coherente en el servidor para facilitar el estilo de codificación y que también estén clasificados según su estilo de datos. Ahora explicaremos la organización de dichos ficheros.

### 5.2 INTERFAZ DE USUARIO

Tal y como se explicó en el punto 4.7 INTERFAZ, para el diseño de la aplicación se han escogido colores suaves y poco cargados, colores adecuados para que el usuario pueda trabajar con la aplicación durante mucho tiempo seguido. Las imágenes que son utilizadas en la web se guardan en un fichero llamado “images” y el estilo a las diferentes páginas de la aplicación se añade utilizando páginas de estilos CSS. Un ejemplo de estilos que se ha utilizado lo podemos ver con la Figura 13 la cual muestra la página donde el usuario ha de identificarse para poder acceder a la aplicación.



Figura 13 Página de inicio

La siguiente Figura 14 pertenece a la primera página que se nos muestra después de entrar a la aplicación.



Figura 14 Primera página

Si nos fijamos en la imagen anterior, en la parte superior izquierda aparece la opción del idioma, podemos ver qué idioma está seleccionado y si lo deseamos podemos cambiar el idioma en que queremos que se nos muestre la aplicación.

Por otro lado aparece el menú, Figura 15, que será común en todas las páginas. Desde este menú podemos acceder a cualquier parte de la aplicación. Según el perfil de usuario podrá acceder y visualizar unas u otras opciones.



Figura 15 Menú común



En la siguiente imagen se muestra el formulario de la oferta con los campos a rellenar.

The image shows a web application interface for managing research projects. The main menu at the top includes 'Inicio', 'Ofertas', 'Proyectos', 'Clientes', 'Contactos', and 'Administrar web'. The 'Ofertas' section is active, showing a 'Nueva oferta' (New offer) form. The form has the following fields: 'Código oferta' (text), 'Título proyecto' (text), 'Tipo proyecto' (dropdown), 'Cliente' (text with add/delete icons), 'Procedimiento' (text), 'Periodicidad' (text), 'Presupuesto' (text with '€' symbol), 'Responsable' (text with add/delete icons), 'Sustituto' (text with add/delete icons), 'Estado oferta' (dropdown), 'Genera convenio' (checkbox), 'Código convenio' (text), and 'Comentarios' (large text area). A 'Guardar' (Save) button is at the bottom right. On the right side, there is a sidebar with 'Ofertas' and options for 'Nueva oferta' and 'Buscar ofertas'.

Figura 16 Formulario oferta

Si nos fijamos en la parte de la izquierda, aparece otro menú donde podremos acceder al formulario para crear una nueva oferta o acceder al listado de ofertas registradas en la base de datos. Éste menú aparecerá también en todas las páginas pero irá variando dependiendo en que parte de la aplicación estemos, por ejemplo si estamos en la opción del menú ofertas, aparecerá como muestra la siguiente imagen.

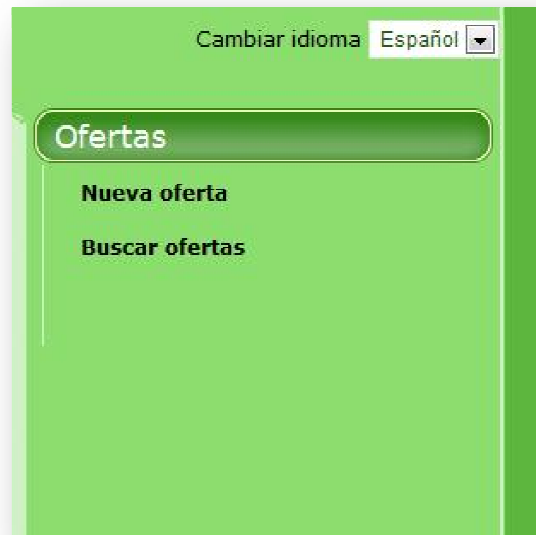


Figura 17 Menú ofertas

Por otro lado, si estamos en la opción de menú proyectos, el menú de la derecha aparecerá de esta forma, Figura 18.

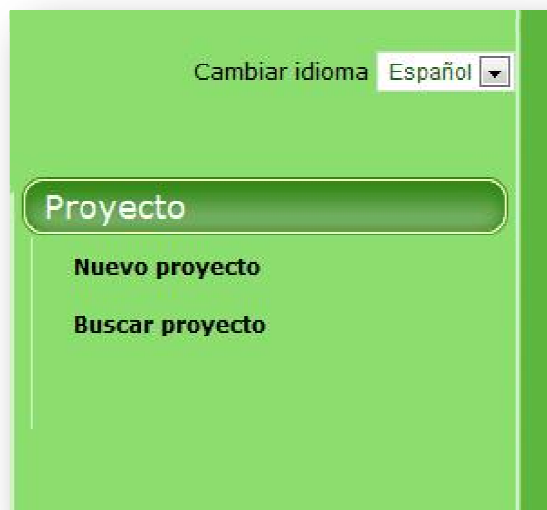


Figura 18 Menú proyectos

Si clicamos en buscar proyecto, la aplicación nos lleva al listado de proyectos que hay disponibles, en la Figura 19 podemos ver un ejemplo.



**Figura 19** Buscar proyecto

En el listado anterior podremos hacer una búsqueda de proyectos por cualquier campo, además a la derecha de cada proyecto nos aparecen unos iconos. Clicando en estos iconos podremos visualizar, editar o eliminar ese proyecto de la base de datos.

A continuación se detalla el significado de algunos de los iconos que aparecen en la aplicación

.n	Significado
	Clicando en esta imagen, podremos visualizar la información
	Clicando en este icono podremos editar la información.
	Si clicamos sobre esta imagen, se eliminará la información a la que hace referencia.
	Este icono aparece en las listas de búsqueda de ofertas. En la tabla donde aparecen todas las ofertas, algunas ofertas tendrán visibles éste icono, lo que significará que la oferta ha generado un proyecto y clicando en el icono accederemos a la pantalla donde se visualizarán todos los datos del proyecto.
	Este otro icono, también aparece sólo en las listas de búsqueda de ofertas. Será visible sólo en algunas ofertas, lo que significará que la oferta ha sido cerrada y se puede generar la ficha de proyecto. Si clicamos en este icono nos llevará al formulario de ficha de proyecto.
	Para seleccionar un usuario.
	En los lugares donde
	Aparece en las tablas de búsqueda, clicando sobre el icono ordena las tablas ascendentemente.



Aparece en las tablas de búsqueda, clicando sobre el icono ordena las tablas descendientemente.

No todos los usuarios tendrán visibles todos los iconos en todos los proyectos o en todas las ofertas, eso dependerá del tipo de perfil que tenga asignado el usuario.

### 5.3 INTERFAZ DE LOS PERFILES DE USUARIOS

Tal y como se ha descrito en el punto 4.2 *Perfiles de usuario*, existen tres perfiles diferentes en la aplicación, y cada uno de ellos tendrá acceso a diferentes partes de la aplicación, además, dependiendo del perfil que el usuario tenga asignado podrá visualizar sólo aquella información a la que esté autorizado.

Para cada perfil de usuario se conserva el mismo estilo, no hay variaciones en el diseño de la página. Las únicas diferencias se encuentran en el acceso a la información y en las opciones de menú. Por ejemplo, un usuario que tenga el perfil de administrador, verá el menú principal de esta manera, Figura 20.



Figura 20 Visualización perfil administrador

En cambio, un usuario con perfil de Lector, verá el menú principal de esta otra forma.



Figura 21 Visualización perfil lector

Si nos fijamos en las dos imágenes anteriores, la única diferencia que podemos encontrar es la opción *Administrar web*, donde aparece en la vista de administrador pero no en la de lector, en cambio el diseño es el mismo y los estilos son los mismos.

Otro ejemplo, lo podríamos ver en las pantallas de búsqueda. La Figura 22 muestra la pantalla de búsqueda de ofertas. Podremos visualizar de esta manera la pantalla si accedemos a la aplicación con un perfil de administrador. Donde el administrador, tiene permisos para poder visualizar, editar y eliminar cualquier oferta. Esto se indica con los iconos que aparecen a la derecha de la oferta.



Figura 22 Búsqueda ofertas

En cambio, si accedemos a la aplicación con perfil de lector la misma pantalla se nos mostrará de la siguiente forma, Figura 23. Donde algunos de los iconos desaparecen ya que el perfil de lector no tiene acceso a todas las opciones, y en este caso sólo podrá visualizar la oferta.

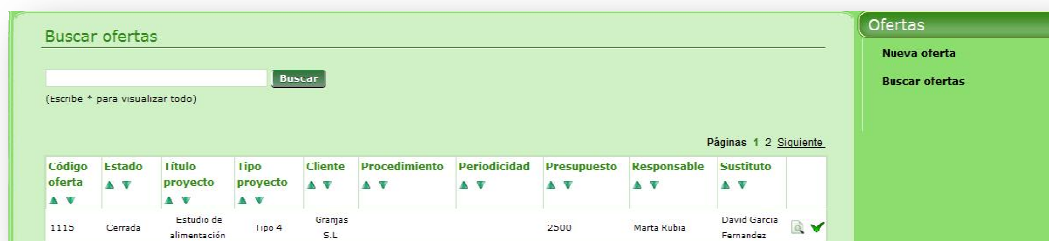


Figura 23 Búsqueda perfil lector

## 5.4 BASE DE DATOS

La siguiente imagen representa las tablas en la base de datos y la relación entre ellas.

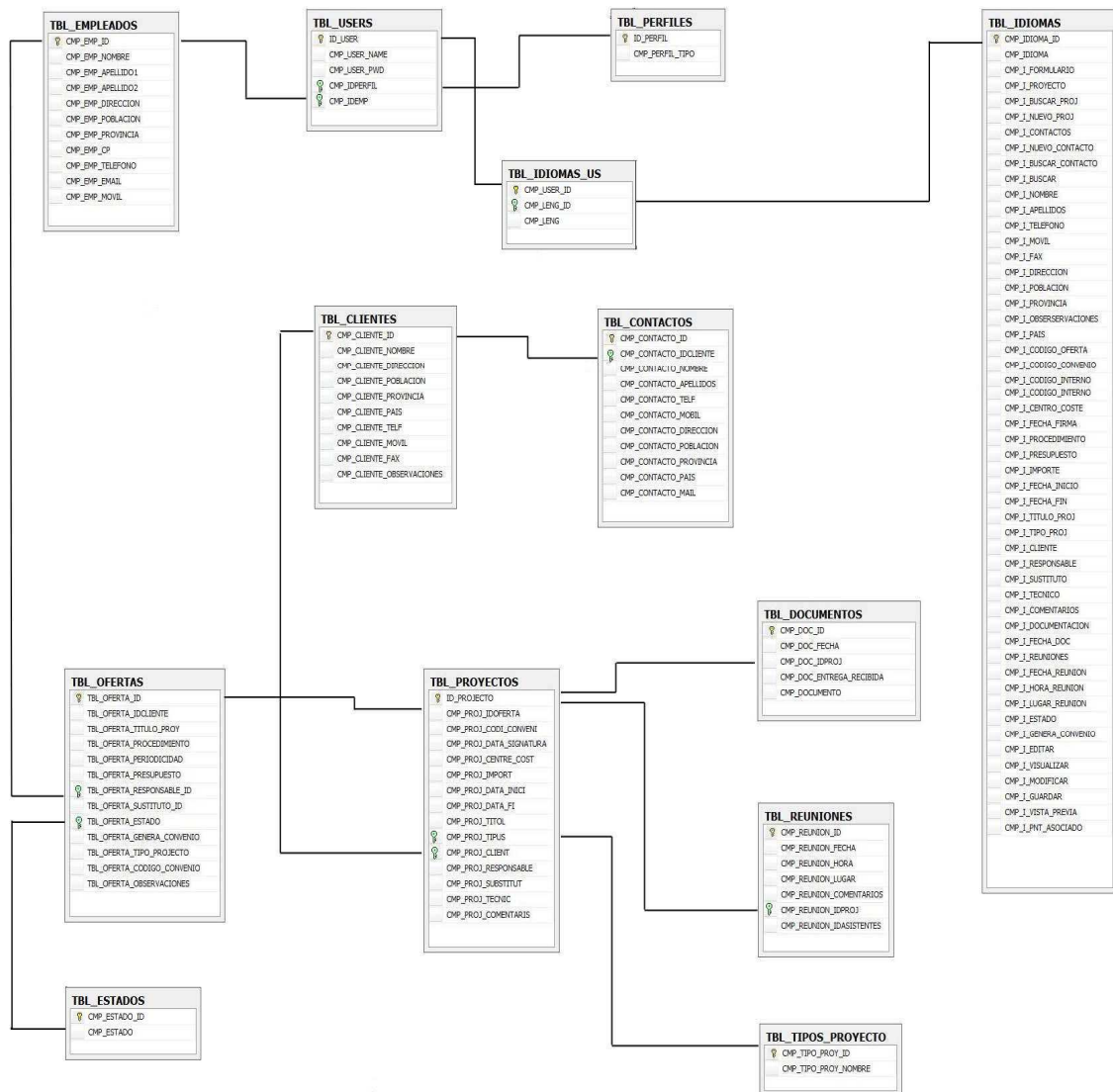
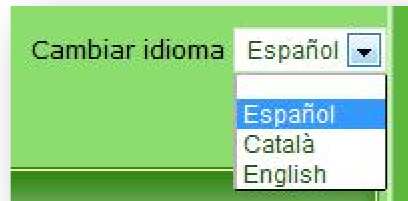


Figura 24 Base de datos

## 5.5 MULTILENGUAGE

La aplicación se puede visualizar en varios idiomas. En cada página aparece un selector de idioma donde podemos seleccionar el idioma el cual queremos que se nos muestre la página.



Selector idioma 1

Además se puede añadir cualquier idioma, donde el administrador de la aplicación puede insertar un nuevo idioma mediante un formulario. En este formulario se muestran todas y cada una de las palabras que aparecen en la web, la Figura 25 muestra el formulario.

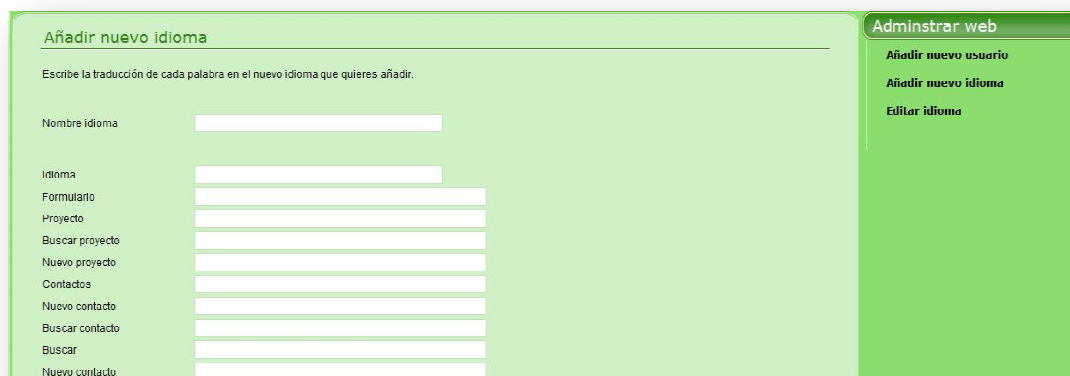
A screenshot of a web application interface for adding a new language. The main section is titled 'Añadir nuevo idioma' (Add new language) and contains the instruction 'Escribe la traducción de cada palabra en el nuevo idioma que quieres añadir.' (Write the translation of each word in the new language you want to add). Below this, there is a list of words to be translated, each with a corresponding input field: 'Nombre idioma', 'Idioma', 'Formulario', 'Proyecto', 'Buscar proyecto', 'Nuevo proyecto', 'Contactos', 'Nuevo contacto', 'Buscar contacto', 'Buscar', and 'Nuevo contacto'. On the right side, there is a sidebar titled 'Administrar web' (Manage website) with three links: 'Añadir nuevo usuario' (Add new user), 'Añadir nuevo idioma' (Add new language), and 'Editar idioma' (Edit language).

Figura 25 Formulario idioma

El usuario debe escribir la traducción de cada palabra al nuevo idioma, una vez se ha guardado el idioma en la base de datos, si se detecta algún error en la traducción existe la posibilidad de poder modificar el idioma o eliminarlo. Se ha decidido hacer la introducción de un nuevo idioma de esta forma porque en la aplicación aparecen palabras técnicas las cuales sería difícil poder traducir de otra forma.

La implementación del idioma se ha hecho de la siguiente forma. Primeramente se llama a dos funciones *getLang(conexión, usuario)* donde le pasamos la conexión a la base de datos y el identificador del usuario que está validado en la web, con esta función recogemos el identificador del idioma que el usuario seleccionó la última vez que se conectó. Después llamamos a la función *createDictionary(lang)* donde le pasamos el identificador del idioma. Figura 26 es un ejemplo de código donde podemos ver la llamada a estas funciones.

```
lang = getLang(con, usuario)
createDictionary(lang)
```

Figura 26 Llamada a funciones *getLang(conexión, usuario)* y *createDictionary(lang)*

La función *createDictionary(lang)* realiza una consulta a la base de datos, accede a la tabla donde se guardan las palabras traducidas y extrae las palabras en el idioma seleccionado. Cada palabra se guarda en variables de sesión.

```
if id <> "" then
    rs.Open "SELECT * FROM TBL_IDIOMAS WHERE CMP_IDIOMA_ID = " & id, con, 0, 1
    if rs.EOF = false then
        session("DICC_IDIOMA_ID") = rs.Fields("CMP_IDIOMA_ID")
        session("DICC_IDIOMA") = rs.Fields("CMP_IDIOMA")
        session("DICC_I_IDIOMA") = rs.Fields("CMP_I_IDIOMA")
        session("DICC_FORMULARIO") = rs.Fields("CMP_I_FORMULARIO")
        session("DICC_PROYECTO") = rs.Fields("CMP_I_PROYECTO")
        session("DICC_BUSCAR_PROJ") = rs.Fields("CMP_I_BUSCAR_PROJ")
        session("DICC_NUEVO_PROJ") = rs.Fields("CMP_I_NUEVO_PROJ")
        session("DICC_CONTACTOS") = rs.Fields("CMP_I_CONTACTOS")
```

Figura 27 Consulta base de datos



Cada variable representa una palabra y se ubican en el código HTML. La siguiente imagen corresponde a una captura de pantalla del código donde se puede ver la ubicación de las variables de sesión.

```
<div id="buttons">
  <a href="home.asp" class="but" title=""><%=session("DICC_INICIO")%></a>
  <a href="buscarOfertas.asp" class="but" title=""><%=session("DICC_OFERTAS")%></a>
  <a href="buscarProyectos.asp" class="but" title=""><%=session("DICC_PROYECTOS")%></a>
  <a href="buscarClientes.asp" class="but" title=""><%=session("DICC_CLIENTES")%></a>
  <a href="buscarContactos.asp" class="but" title=""><%=session("DICC_CONTACTOS")%></a>
  <a href="adminweb.asp" class="but" title=""><%=session("DICC_ADMINISTRAR")%></a>
</div>
```

Figura 28 Ubicación de las variables de sesión

Por otro lado, los datos que el usuario introduce en los formularios, son validados antes de ser enviados al servidor y se guarden en la base datos, de esta manera nos aseguramos que no ocurra un error en la aplicación. Para validar los formularios se han desarrollado funciones en javascript. Un ejemplo de función lo podemos ver en la Figura 29 Esta función comprueba que los datos que se estén ingresando en el formulario sean numéricos.

```
function Solo_Numerico(variable) {
  Numer=parseInt(variable);
  if (isNaN(Numer)) {
    return "";
  }
  return Numer;
}
function ValNumero(Control) {
  Control.value=Solo_Numerico(Control.value);
}
```

Figura 29 Función validar numéricos

Para llamar a dicha función sólo basta con poner el siguiente control en el input que se quiera validar.

```
<input type='text' id='client_telf' maxlength='11' value='' & CMP_CLIENT_TELF & " onKeyUp="return ValNumero(this)"/>
```

Figura 30 Llamada función ValNumero()

## 6. PRUEBAS

### 6.1 INTRODUCCIÓN

La fase de pruebas es una parte muy importante del desarrollo de la aplicación, ya que es un requisito importante para finalizar el proyecto es conseguir los objetivos fijados, es decir, la aplicación ha de funcionar correctamente y de la forma esperada.

Normalmente en el desarrollo de un proyecto, la parte de pruebas toma una parte importante del tiempo. En este proyecto, debido a la falta de tiempo, no se ha invertido el tiempo necesario para conseguir un 100% de fiabilidad de la aplicación, pero siguiendo una metodología apropiada se ha conseguido reducir el mayor número de errores durante el tiempo invertido.

A continuación se explican las pruebas y métodos realizados en la aplicación

### 6.2 PRUEBAS DE INTERFACES

Ya que existen varios navegadores y el uso de uno u otro depende del usuario, se ha tenido en cuenta que un posible error en la interfaz de usuario podría darse dependiendo del software que éste utilice. Para probar la interfaz de usuario y que ésta se visualice correctamente en cualquier navegador, se ha accedido a la aplicación desde varios navegadores para poder probar las hojas de estilo aplicadas en la web y el código HTML.

Navegador	
Opera 11.5	Parcialmente compatible
Internet Explorer 7	Compatible
Internet Explorer 8	Compatible
Firefox 5.0	Compatible

A parte de comprobar la forma en que se comportan las interfaces en cada navegador se ha comprobado el código javascript y el comportamiento de los archivos css utilizando las herramientas de detección de problemas de los diferentes navegadores, “Consola de Errores” de Firefox, “Herramientas de desarrollo” de Internet Explorer y “Dragonfly” de Opera.

A la hora de escribir el código HTML se ha ido con cuidado para que no hayan errores en su estructura, los errores de HTML no impide que la web ejecute erróneamente sus funciones pero sí que causa desorden a la hora de mostrar la información.

### 6.3 PRUEBAS DE SEGURIDAD

Comprueba los mecanismos de seguridad como por ejemplo que un usuario no pueda acceder a la aplicación sin que previamente se haya identificado con un nombre de usuario y una contraseña que esté dada de alta en la base de datos o si el usuario y contraseña no son los correctos. En este caso la aplicación mostraría un mensaje de error, ver Figura 31.

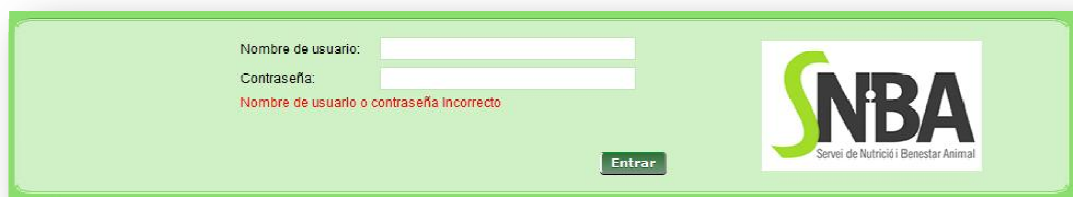
The image shows a login interface with a light green background. On the left, there are two input fields: "Nombre de usuario:" and "Contraseña:". Below these fields, a red error message reads "Nombre de usuario o contraseña incorrecto". To the right of the input fields is a dark green button labeled "Entrar". On the far right, there is a logo for "SNBA" with the text "Servei de Nutrició i Benestar Animal" underneath it.

Figura 31 Error nombre usuario

También se comprueba si un usuario intenta acceder a un parte de la aplicación a la cual no tiene acceso. Por ejemplo que un usuario quiera modificar un proyecto el cual sólo puede visualizar. En ese caso se mostraría el siguiente error.

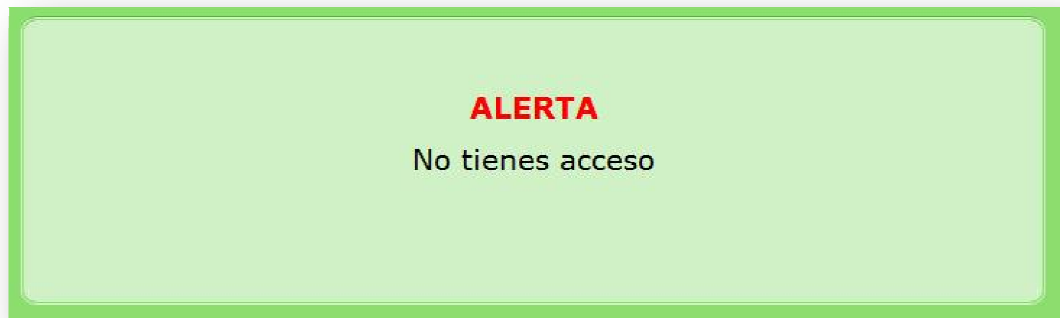


Figura 32 Mensaje error

## 6.4 PRUEBAS

A continuación se detalla la batería de pruebas que se ha realizado en la aplicación. En estas pruebas, se han probado tanto requisitos funcionales como no funcionales.

En la tabla primeramente se describe la entrada de datos, a continuación que es lo que se espera que haga la aplicación y por último si el resultado ha sido satisfactorio o por lo contrario la aplicación ha fallado.

### Pruebas en los formularios

Pruebas en el formulario de ficha de oferta y proyecto.

Entrada	Salida esperada	Formulario oferta	Formulario proyecto
Guardar una ficha de oferta sin antes rellenar ningún campo.	Que la aplicación no deje guardar y muestre un mensaje de error "Los campos con (*) son obligatorios"	KO	OK

<b>Introducir manualmente los datos en los campos de “fecha”.</b>	Que no deje introducir los datos manualmente, pero sí con el calendario emergente.	OK	OK
<b>Introducir caracteres en lugar de números en el campo “Presupuesto”/“Importe”.</b>	Que no deje introducir caracteres.	OK	KO
<b>Introducir manualmente datos en los campos, cliente, responsable y sustituto</b>	Que no deje introducir manualmente datos, pero sí a través del listado emergente que aparece al clicar en el icono “seleccionar usuario”.	OK	OK
<b>Rellenar todos los campos y clicar en guardar.</b>	Que muestre un mensaje de alerta si el código de la oferta ha sido ya utilizado y muestre el nuevo código.	OK	OK
<b>Rellenar con el máximo de caracteres posibles el campo “comentarios.”</b>	Que se muestre el número de caracteres restantes por entrar y al finalizar muestre un mensaje de “completado.”	KO	OK

Pruebas en el formulario clientes, contactos y usuario.

Entrada	Salida esperada	Formulario cliente	Formulario contacto	Formulario usuario
<b>Clicar en guardar sin antes rellenar ningún campo.</b>	Que la aplicación no deje guardar y muestre un mensaje de error “Los campos con (*) son obligatorios” o “Revisa los campos en rojo”	OK	KO	OK
<b>Rellenar todos los campos del formulario y clicar en “borrar formulario”</b>	Que todos los campos del formulario se queden vacíos.	OK	OK	OK

<b>Introducir caracteres en lugar de números en los campos “teléfono, móvil y fax”.</b>	Que no deje introducir caracteres.	OK	OK	--
<b>Rellenar con el máximo de caracteres posibles el campo “comentarios.”</b>	Que se muestre el número de caracteres restantes por entrar y al finalizar muestre un mensaje de “completado.”	OK	OK	--
<b>Rellenar todos los campos con el máximo de caracteres posibles y clicar en guardar.</b>	Que guarde correctamente.	OK	OK	OK
<b>Introducir manualmente datos en el campo, cliente.</b>	Que sólo deje introducir el cliente mediante la ventana emergente.	-	OK	--

.Pruebas en formulario añadir nuevo idioma.

Entrada	Salida esperada	OK o KO
<b>Clicar en guardar sin antes rellenar ningún campo.</b>	Que la aplicación no deje guardar sin rellenar los campos obligatorios y muestre un mensaje de error “Los campos con (*) son obligatorios” o “Revisa los campos en rojo”.	OK
<b>Comprobar que todos los datos se hayan guardado correctamente en la base de datos</b>	Que guarde correctamente.	OK

Pruebas en la página de búsqueda de ofertas, búsqueda de proyectos y búsqueda de clientes.

Entrada	Salida esperada	Ofertas	Proyectos	Clientes
<b>Buscar por cualquier campo</b>	Que en el listado muestre las oferta/proyecto/cliente según el criterio de búsqueda.	OK	OK	OK
<b>Clicar en el icono de visualizar.</b>	Que la página redireccione a visualizar oferta/proyecto/cliente.	OK	OK	OK
<b>Clicar en el icono de editar.</b>	Que la página redireccione a editar oferta/proyecto/cliente.	OK	OK	OK
<b>Clicar en el icono de eliminar.</b>	Que muestre un mensaje de alerta "Seguro que quieres eliminar?" y al clicar en aceptar se elimine la oferta/proyecto/cliente seleccionada.	OK	OK	OK
<b>Entrar en el sistema una nueva oferta con estado "cerrada."</b>	Que en la página de búsqueda aparezca un icono informando que la oferta está cerrada pero no tiene ficha de proyecto.	OK	--	--
<b>Crear una ficha de proyecto a partir de una oferta con estado "cerrada".</b>	Que en la página de búsqueda de ofertas aparezca un icono informando que la oferta está cerrada y existe ficha de proyecto.	OK	--	--

Pruebas en la página de búsqueda de contactos y usuarios.

Entrada	Salida esperada	Contacto	Usuario
<b>Buscar por cualquier campo</b>	Que en el listado muestre los contacto/usuario según el criterio de búsqueda.	OK	OK
<b>Clicar en el icono de visualizar.</b>	Que la página redirija a visualizar contacto/usuario.	OK	OK
<b>Clicar en el icono de editar.</b>	Que la página redirija a editar contacto.	OK	--
<b>Clicar en el icono de eliminar.</b>	Que muestre un mensaje de alerta "Seguro que quieres eliminar?" y al clicar en aceptar se elimine el cliente seleccionado.	OK	OK

El resultado de algunas de las pruebas que se han realizado no ha sido el esperado, esto se indica en las tablas anteriores con un "KO", por ejemplo en el formulario contactos se ha clicado en el botón guardar sin antes rellenar ningún campo, esta prueba ha sido fallida debido a que no se controlaba la obligatoriedad de los campos, finalmente el problema se soluciona llamando a una función javascript que valide todos los campos obligatorios antes de guardar los datos. Las pruebas que han sido satisfactorias, se indica con un "OK". Por el contrario, si aparece "--" significa que esa prueba no se ha llevado a cabo en esa página porque no tiene implementada esa funcionalidad, como por ejemplo, en el formulario clientes no se ha podido entrar una nueva oferta con estado cerrada, ya que esa funcionalidad solo existe en el formulario ofertas.

A parte de este listado de pruebas que se han realizado en los formularios y en las páginas de búsqueda, se ha comprobado que los datos introducidos sean guardados correctamente en la base de datos sin dar lugar a errores. También se ha comprobado que todas las páginas tengan codificación UTF-8, y que los caracteres con acentos u otros símbolos sean correctamente interpretados por el navegador.



## 7.CONCLUSIONES

Después de haber finalizado la implementación y testado la aplicación, se hace un balance de cuál ha sido la planificación real del proyecto. Comparando la planificación inicial con la real se observan algunas desviaciones, las cuales se son comentadas más adelante dando una explicación de porqué han surgido y problemas que han ido apareciendo a lo largo del proyecto. Por último se comentan posibles mejoras que podría tener la aplicación en un futuro.

### 7.1 SEGUIMIENTO DEL PROYECTO

El seguimiento del proyecto permite controlar en qué momento se encuentra el desarrollo de la aplicación. También es importante a la hora de valorar la planificación hecha al final de la fase de diseño y mostrar el tiempo real que ha durado cada tarea. De esta manera se puede identificar en qué momento se han producido desviaciones. En la siguiente tabla se detalla el tiempo previsto para cada tarea, cual ha sido realmente el tiempo de duración y la desviación que ha habido entre la planificación prevista y la real.

Tareas	Tiempo previsto	Tiempo real	Desviación
Estudio de viabilidad	14 h	28 h	14 h
Casos de uso	4 h	4 h	0 h
Análisis de los datos	6 h	6 h	0 h
Diseño base de datos	5 h	6 h	1 h
Diseño de la aplicación	14 h	14 h	0 h
Diseño de la interfaz	7 h	9 h	2 h
<b>Preparación del entorno de programación</b>			
Instalación y configuración del servidor web	3 h	3 h	0 h
Instalación y configuración del servidor de base de datos	4 h	12 h	8 h
<b>Implementación</b>			
Codificación	90 h	105 h	15 h
Test de pruebas	8 h	8 h	0 h

Corrección de errores	10 h	12 h	2 h
<b>Documentación</b>			
Elaboración de la memoria	20 h	24 h	4 h
Corrección de la memoria	5 h	8 h	3 h
Preparación de la presentación	10 h	14 h	4 h
<b>Total</b>	<b>200 h</b>	<b>251 h</b>	<b>49 h</b>

A continuación se muestra el diagrama de Gantt con los tiempos reales de cada tarea. El diagrama se ha dividido en tres imágenes, cada una de las cuales precede a la anterior.

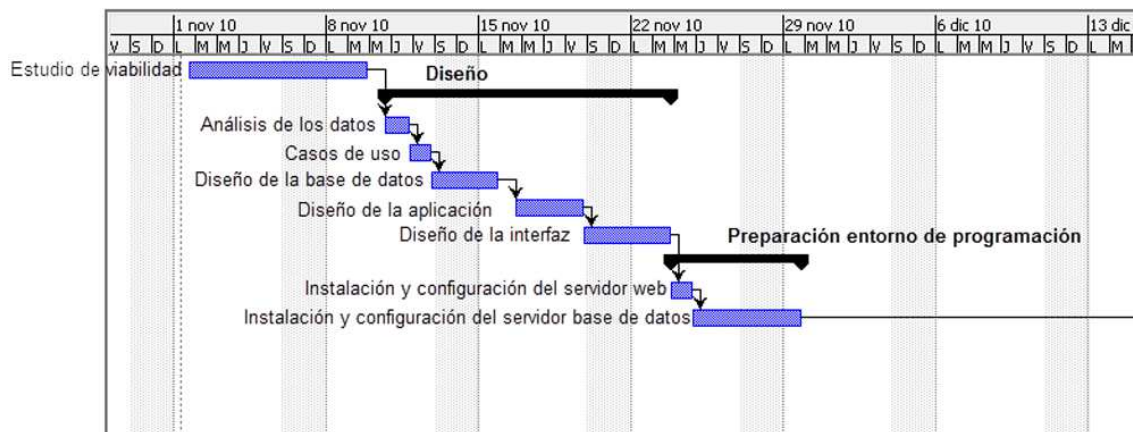


Figura 33 Diagrama de Gantt, planificación real

La Figura 34, es la continuación de la **¡Error! No se encuentra el origen de la referencia.**, en ella se puede observar el tiempo que se ha necesitado para la codificación.

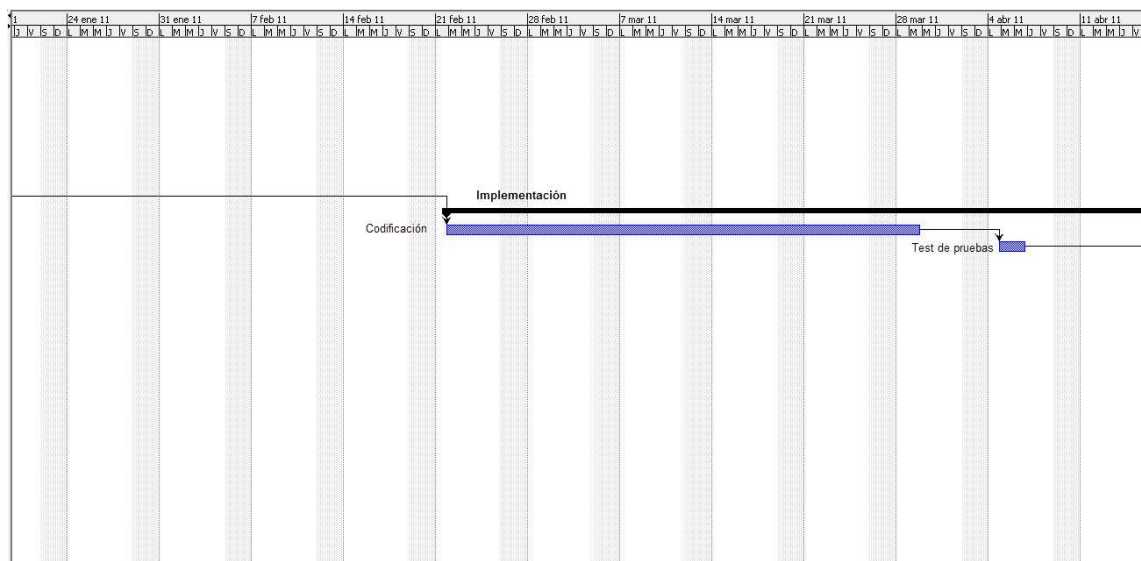


Figura 34 Diagrama de Gantt, planificación real

La siguiente imagen, es la parte final del diagrama de Gantt. Como se puede observar hubo un paro durante el desarrollo del proyecto desde Abril hasta Junio. En el siguiente punto se explica más detalladamente porqué no se pudo trabajar en el proyecto.

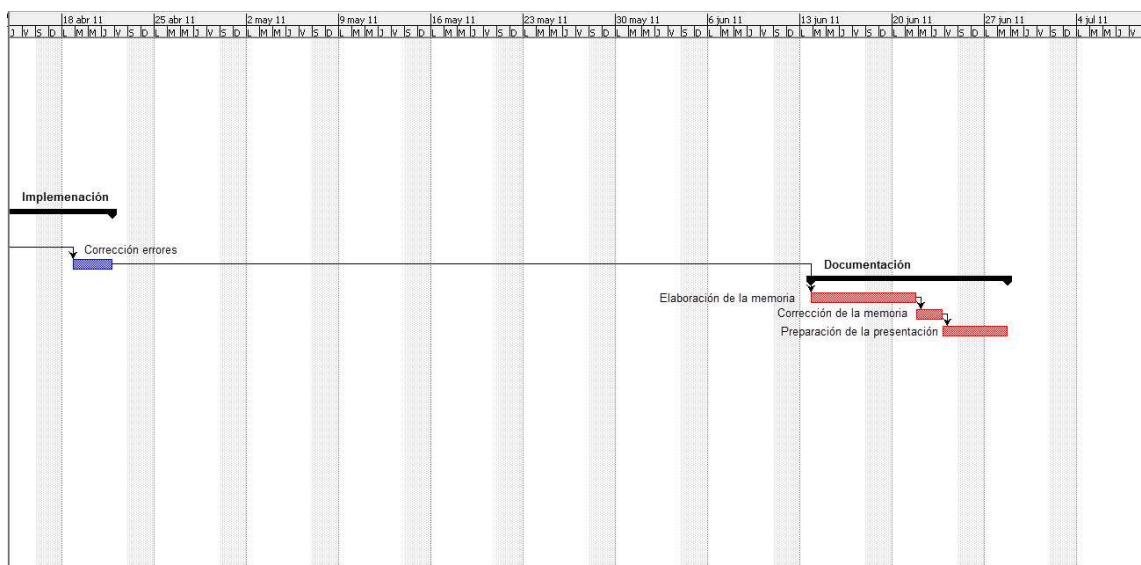


Figura 35 Diagrama de Gantt, planificación real

## 7.2 DESVIACIONES

Tal y como muestra la tabla de tareas que se ha visto en el punto 7.1 SEGUIMIENTO DEL PROYECTO se pueden observar las desviaciones que han surgido en la planificación del proyecto.

En las imágenes anteriores se muestra el diagrama de Gantt con la planificación real. Se puede observar que ha habido pausas en las actividades relacionadas con el proyecto, la más notable es la que va desde el mes de Diciembre al mes de Febrero. Esto se debe a la preparación de los exámenes de Febrero. Otra pausa importante es la que va desde Abril hasta Junio, esta pausa se debe al exceso de trabajo laboral que se tuvo en esos meses.

Por otro lado la tarea relacionada con la programación de la aplicación también se ha retrasado bastante, dando lugar a retrasos en las siguientes tareas. En el estudio de viabilidad y se estimaba que la codificación se podría terminar mucho antes de la finalización real. Esto ha se ha debido a que durante el desarrollo no se ha podido dedicarle el tiempo necesario que se había estimado al principio.

A principio de curso, se solicitó a la escuela un servidor web y un servidor de base de datos para poder alojar la aplicación. En el transcurso del desarrollo se ha ido trabajando en un servidor web y de base de datos ajeno a la escuela, pero al querer traspasar la aplicación al servidor de la escuela, han surgido varios problemas. Por un lado, el servidor web no era accesible por escritorio remoto, por lo que no era posible alojar la aplicación y hacer pruebas remotamente. Después que el servicio de informática distribuida solucionase el problema, surgieron más inconvenientes. Por un lado, el usuario que nos habían facilitado para acceder a la base de datos no tenía los permisos necesarios para crear y modificar tablas. El poder acceder a un servidor pero no tener permisos para realizar configuraciones necesarias para el buen funcionamiento de la aplicación ha hecho que la planificación se retrase.

## 7.3 FUTURO DEL PROYECTO Y POSIBLES MEJORAS

Este proyecto nace de la necesidad de mejorar una aplicación ya existente, que tiene como fin gestionar proyectos del centro SNI BA. Gracias a la nueva aplicación el personal del centro podrá acceder a la información de forma más organizada, ahorrando tiempo de trabajo. Se ha conseguido mejorar la aplicación pero aún así se considera que la nueva herramienta se podría mejorar.

### Mejoras de diseño

Con la nueva aplicación se ha mejorado el diseño de la anterior, haciendo que sea fácil de usar y entendible para el usuario. Pero con la experiencia adquirida, se ha detectado que quizás sería mejor cambiar la parte del menú y crear menús jerárquicos, esto facilitaría la navegación haciéndola más rápida.

### Mejoras de interfaz

Una de las funcionalidades que por falta de tiempo no se ha conseguido, es poder crear gráficos de barras a partir de las páginas de búsqueda, los gráficos podrían mostrarse según los filtros establecidos, esto ayudaría a hacer diferentes análisis de los datos.

Por otro lado, en los formularios donde aparecen campos en los que se debe seleccionar un usuario, cliente o contacto, la forma de selección es mediante una ventana emergente. Estas ventanas emergentes se podrían eliminar e implementar una forma de selección autocompletable, es decir, donde el sistema fuera sugiriendo los diferentes resultados que se encuentran almacenados en la base de datos en concordancia con los datos que se fueran entrando. Esta mejora, se intentó implementar durante el desarrollo, pero surgieron problemas de compatibilidad en diferentes navegadores, donde en algunos funcionaba correctamente el código pero no en otros. Así que, por la falta de tiempo se decidió dejar esta parte como una posible mejora.

## 7.4 VALORACIÓN PERSONAL

La experiencia ha sido positiva. La principal razón es porque los conocimientos adquiridos durante la carrera se han visto reforzados durante el desarrollo del proyecto. Los conceptos aprendidos han sido útiles para resolver los problemas que ha planteado el proyecto, además de adquirir nuevos conocimientos durante el desarrollo. A pesar de las dificultades que han ido surgiendo y las pausas que se han visto reflejadas en la planificación real finalmente se han podido cumplir la mayoría de los objetivos, hecho que ha causado motivación.

Gracias a esta motivación se considera que todos los conocimientos que se han adquirido serán de gran utilidad en un futuro laboral.

## 8.BIBLIOGRAFIA Y REFERENCIAS

**Viguer J.M** *Fundamentos de programación en ASP 3.0*,  
McGRAWHILL/INTERAMERICANA DE ESPAÑA, S.A.U, 2001

### Referencias internet

- <http://sourceforge.net/openproj> (Noviembre 2010)
- <http://es.wikipedia.org/wiki/DotProject> (Enero 2011)
- [http://es.wikipedia.org/wiki/Anexo:Software de gesti%C3%B3n de proyectos](http://es.wikipedia.org/wiki/Anexo:Software_de_gesti%C3%B3n_de_proyectos) (Enero 2011)
- <http://blogvecindad.com/lista-completa-de-menus-con-css-para-diseandores-web/> (Enero 2011)
- [https://developer.mozilla.org/es/Migrar aplicaciones desde Internet Explorer a Mozilla](https://developer.mozilla.org/es/Migrar_aplicaciones_desde_Internet_Explorer_a_Mozilla) (Enero 2011)
- [http://es.wikipedia.org/wiki/Anexo:Comparaci%C3%B3n de sistemas administradores de bases de datos relacionales](http://es.wikipedia.org/wiki/Anexo:Comparaci%C3%B3n_de_sistemas_administradores_de_bases_de_datos_relacionales) (Enero 2011)
- <http://msdn.microsoft.com> (Febrero 2011)
  - <http://msdn.microsoft.com/es-es/library/ms186939.aspx> (Febrero 2011)
- <http://www.devjoker.com/contenidos/Articulos/82/Como-encryptar-o-cifrar-cadenas-y-campos-en-SQL-server-2005-ENCRYPTBYPASSPHRASE-y-DECRYPTBYPASSPHRASE.aspx> (Febrero 2011)
- [www.w3schools.com](http://www.w3schools.com)
  - [http://www.w3schools.com/asp/ev\\_app\\_onend\\_onstart.asp](http://www.w3schools.com/asp/ev_app_onend_onstart.asp) (Mayo 2011)
  - [http://www.w3schools.com/asp/met\\_getbasename.asp](http://www.w3schools.com/asp/met_getbasename.asp) (Septiembre 2011)
  - [http://www.w3schools.com/ado/ado\\_connect.asp](http://www.w3schools.com/ado/ado_connect.asp) (Enero 2011)
  - [http://www.w3schools.com/VBScript/func\\_left.asp](http://www.w3schools.com/VBScript/func_left.asp) (Julio 2011)
  - [http://www.w3schools.com/aspnet/aspnet\\_vsasp.asp](http://www.w3schools.com/aspnet/aspnet_vsasp.asp) (Enero 2011)

- [http://www.w3schools.com/ajax/ajax\\_xmlhttprequest\\_create.asp](http://www.w3schools.com/ajax/ajax_xmlhttprequest_create.asp)  
(Marzo 2011)
- 
- [www.desarrolloweb.com](http://www.desarrolloweb.com) (Julio 2011)
- [www.w3c.es/estandares/](http://www.w3c.es/estandares/) (Septiembre 2011)
- [www.webestilo.com/javascript](http://www.webestilo.com/javascript) (Septiembre 2011)
- <http://asp.astalaweb.com> (Septiembre 2011)
- [http://www.asptutor.com/asp/vart.asp?id=93#asptips\\_tip2](http://www.asptutor.com/asp/vart.asp?id=93#asptips_tip2) (Septiembre 2011)



Marta Rubia Jiménez