



**Universitat Autònoma  
de Barcelona**

**Módulo de Modelado Hardware  
GISCMO**

Memoria del proyecto  
de Ingeniería Técnica en  
Informática de Sistemas  
realizada por  
Javier Enrique Ruiz Salvador  
y dirigido por  
Rafael Cortés Fité.

Escola Universitària d'Informàtica  
Sabadell, junio de 2010.

El abajo firmante, **Rafael Cortés Fité**,  
de l'Escola d'Enginyeria de la UAB,

**CERTIFICA:**

Que el trabajo al que corresponde la presente memoria ha estado realizado bajo su dirección por **Javier Enrique Ruiz Salvador**

Y para a que conste firma la presente.  
Sabadell, **junio de 2010**

-----  
Firmado: **Rafael Cortés Fité**.

## Resumen

Este proyecto tiene como objetivo desarrollar un aplicativo en formato web capaz de dar soporte al modelado de equipos hardware para su posterior publicación en librerías. El proyecto se incluye en un entorno de trabajo constituido por una serie de herramientas de las que nuestro proyecto es una pequeña, pero impórtate parte.

Nuestro objetivo comprenderá la creación y posterior gestión de modelos de equipos hardware para ser utilizados por los proyectos de instalaciones de control industrial, que generen sus trabajos mediante el uso del entorno de desarrollo GISCMO. (Gestor integrado de sistemas de control y modelado de objetos).

# 1. Contenido.

1.	Contenido .....	4
2.	Introducción.....	6
2.1	Resumen de contenidos. ....	6
2.2	Estado del Arte.....	7
2.3	Objetivos y alcance previstos.....	11
2.4	Aproximación a la tecnología de proyecto. ....	12
3.	Estudio de viabilidad.....	15
3.1	Introducción .....	15
3.1.1	Tipología y palabras clave. ....	15
3.1.2	Descripción. ....	15
3.2	Objetivos del proyecto.....	15
3.2.1	Objetivos del proyecto.....	15
3.2.2	Priorización de los objetivos del proyecto.....	15
3.3	Definiciones, acrónimos y abreviaturas.....	16
3.4	Partes interesadas. ....	16
3.4.1	Stakeholders. ....	16
3.4.2	Perfiles de usuario. ....	16
3.4.3	Project Team. ....	16
3.5	Referencias.....	17
3.6	Estudio de la situación actual.....	17
3.6.1	Contexto.....	17
3.6.2	Lógica del sistema. ....	17
3.6.3	Descripción física. ....	18
3.6.4	Usuarios y/o personal del sistema. ....	18
3.6.5	Diagnóstico del sistema. ....	18
3.6.6	Normativas y legislación. ....	18
3.7	Requisitos del proyecto.....	19
3.7.1	Requisitos funcionales. ....	19
3.7.2	Requisitos no funcionales. ....	19
3.7.3	Restricciones del sistema. ....	19
3.8	Alternativas y selección de la solución.....	19
3.8.1	Alternativa 1.....	19
3.8.2	Alternativa 2.....	20
3.8.3	Solución propuesta. ....	20
3.9	Planificación del proyecto.....	20
3.9.1	Recursos del proyecto. ....	20
3.9.2	Tareas del proyecto. ....	21
3.9.3	Planificación temporal. ....	22
3.10	Evaluación de riesgos.....	23
3.10.1	Lista de riesgos.....	23
3.10.2	Catalogación de riesgos. ....	23
3.10.3	Plan de contingencia. ....	24
3.11	Presupuesto.....	24
3.11.1	Estimación costes de personal. ....	24
3.11.2	Estimación coste de los recursos. ....	25
3.11.3	Resumen y análisis de costes de beneficios. ....	25
3.12	Conclusiones.....	25
4.	Ánalysis del proyecto.....	26

4.1	Requerimientos funcionales.....	26
4.1.1	Visión general .....	26
4.1.2	Requisitos funcionales de usuario.....	26
4.1.3	Diagramas de casos de uso.....	27
4.1.4	Diagramas de secuencias.....	30
4.1.5	Menús de navegación.....	32
4.1.6	Diseño de pantallas.....	33
4.2	Restricciones de diseño.....	33
5.	Diseño .....	35
5.1	Introducción .....	35
5.1.1	Selección de bases de datos .....	35
5.2	Estructura de la base de datos .....	35
5.2.1	Tabla eqDisp .....	36
5.2.2	Tabla adData.....	37
5.2.3	Tabla bornasEq.....	38
5.2.4	Tabla adDataBornas .....	38
5.2.5	Tabla de filiaciones .....	39
5.2.6	Tabla de usuarios.....	39
6.	Implementación .....	40
6.1	Introducción .....	40
6.2	Codificación de las diferentes capas de la aplicación.....	40
6.2.1	Interfaces.....	40
6.2.2	Entorno de usuarios .....	41
6.2.3	Motor de aplicación.....	41
6.2.4	Capa de datos .....	43
7.	Pruebas .....	45
7.1	Introducción .....	45
7.2	Pruebas de compatibilidad.....	45
7.3	Pruebas de seguridad .....	45
7.4	Pruebas de integración de servicio .....	45
8.	Conclusiones y resultados .....	46
8.1	Consecución de objetivos .....	46
8.2	Desviaciones observadas .....	46
8.3	Líneas de ampliación .....	46
8.4	Valoración personal de la experiencia .....	46
9.	Bibliografía.....	47

## 2. Introducción.

### 2.1 Resumen de contenidos.

Este proyecto tiene como objetivo desarrollar un aplicativo en formato web capaz de dar soporte al modelado de equipos hardware para su posterior publicación en librerías. El proyecto se incluye en un entorno de trabajo constituido por una serie de herramientas de las que nuestro proyecto es una pequeña, pero importante parte.

Nuestro objetivo comprenderá la creación y posterior gestión de modelos de equipos hardware para ser utilizados por los proyectistas de instalaciones de control industrial, que generen sus trabajos mediante el uso del entorno de desarrollo GISCMO. (Gestor integrado de sistemas de control y modelado de objetos).

Este módulo de modelado Hardware para aplicativos de gestión de proyectos de control industrial, forma parte de la plataforma de desarrollo GISCMO. Una fracción de lo que será GISCMO actualmente está implementada bajo el programa EPS2008 y una base de datos con *Microsoft Access®*.

GISCMO se compone de la unión de cinco proyectos, los cuales todos tienen vínculos entre sí, pero a la vez son independientes.

Lo forman:

- Módulo de modelado Hardware
- Módulo de modelado de objetos
- Módulo de diccionarios de señalización
- Módulo de gestión de proyectos
  - Model View
  - Net View
- Módulo de compilación para SCADA/PLC

El módulo de modelado Hardware está diseñado para contener la información de los elementos hardware que se encuentren en un sistema determinado mediante bases de datos como: PLCS, buses, routers, etc. Dentro de cada componente se deberá guardar diversa información como una matrícula única que identificará el objeto, fabricante, software utilizado, conexiones, etc.

El módulo de modelado de objetos es muy similar al módulo de modelado hardware, donde la mayor diferencia es el tipo de contenido. Este módulo es el encargado de almacenar, también con bases de datos, la información de los objetos que no son componentes hardware del sistema, como podrían serlo detectores de incendio, detectores de puertas, etc.

El módulo de diccionarios será útil para, una vez definidos los objetos y elementos en las librerías de trabajo saber mediante códigos las diferentes características de cada equipo y su significado.

El objetivo del compilador es generar los códigos fuente y listas de variables para cargar en los elementos hardware como por ejemplo en los PLCs. Dichos códigos se crean automáticamente mediante la información de las fichas existentes en la base de datos.

Nosotros diseñaremos e implementaremos en formato web el proyecto módulo de modelado hardware. A grandes rasgos consistirá en la creación, modificación y supresión (si fuera necesario), de tablas de información para los componentes hardware del sistema, así mismo como introducir la información necesaria y complementaria de utilidad.

Además existirán diferentes niveles de usuarios, con su propia área de trabajo y en los que cada uno desempeñará una función específica. Existirá al menos un administrador, un desarrollador, un validador y un visitante. Sus funciones serán:

- Administrador: Administración de permisos del sistema, gestión de usuarios.
- Desarrollador: Desarrolla las fichas de componentes y su contenido.
- Validador: Observa el estado del trabajo efectuado por el desarrollador y gestión de librerías publicadas de modelos.
- Visitante: cliente que observa el estado del trabajo efectuado por el desarrollador.

## 2.2 Estado del Arte

Actualmente en el mercado existen una gran cantidad de software orientado a la programación de PLC's y SCADAS. Por otro lado existe muchos programas de gestión de proyectos y control de versiones, orientados a plazos de ejecución, control de tiempos y trazabilidad del software. La mayoría de este software está diseñado para cubrir áreas muy concretas de las diferentes etapas de un proyecto, pero al tratarse herramientas generalistas no cubren las necesidades específicas del mundo del control industrial.

Los proyectos de control industrial, ha sufrido a lo largo del tiempo de una lacra muy grave derivada de la interdisciplinariedad. La gran cantidad de personal que intervine con perfiles técnicos completamente diferentes como ingenieros industriales, químicos, informáticos, electricistas, etc..., crea una serie de problemática difícilmente controlable. Estos proyectos están muy segmentados en función de cada fase del proceso de evolución y por ello cada usuario ha optado por utilizar las herramientas que mejor conoce. En muchos casos estas herramientas software se han desarrollado enfocadas sobre una necesidad en concreto o incluso para dar soporte a un hardware determinado.

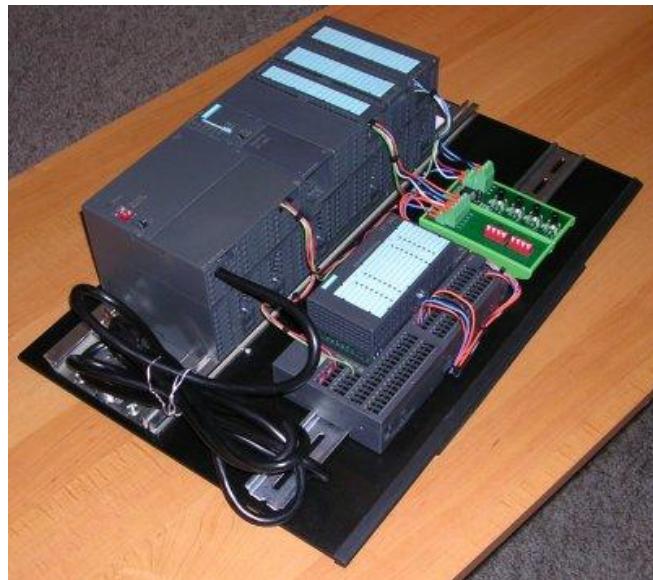
Esta estratificación de los trabajos crea verdaderos abismos, tanto de conocimiento como de colaboración, entre los diferentes participantes en el proyecto.

Los diferentes aspectos involucrados en un proyecto de control industrial, han conducido a herramientas muy especializadas, sin vocación de compartir la información entre ellas. En la mayoría de los casos las empresas que han invertido su esfuerzo y trabajo en estos desarrollos, han establecidos altos precios y peajes en forma de estándares propietarios con la intención de mantener cautivo al mercado.

Como consecuencia de estas actitudes de bloqueo y opacidad de los diferentes fabricantes, se han producido una cantidad inimaginable de herramientas, protocolos, productos, etc... sin ningún tipo posible de

integración mutua, y buscando imponerse en el mercado creando estándares de facto.

En esta línea cabe mencionar algunos de los fabricantes más importantes como SIEMENS AUTOMATION. Este mediante un titánico esfuerzo en I+D+I, ha establecido sus productos como estándar de facto en la mayor parte de la industria química. SIEMENS AUTOMATION dispone en la actualidad de un gran número de productos, capaces de dar servicio a cualquier necesidad del mercado pero al precio de ser un cliente cautivo. En la vanguardia de los productos de SIEMENS AUTOMATION para los entornos de control industrial, están los autómatas de la familia S7 en sus diversas gamas S7-200, S7-300 y S7-400. Estos productos han demostrado su eficiencia en casi todos los procesos industriales desde la química fina hasta el mundo de la automoción.



Autómata SIEMENS AUTOMATION S7-300 con periferia descentralizada ET200

Otro gran fabricante de productos de automatización es Schneider electric, que es quizás el más cercano competidor de SIEMENS en el mundo de la automatización industrial.

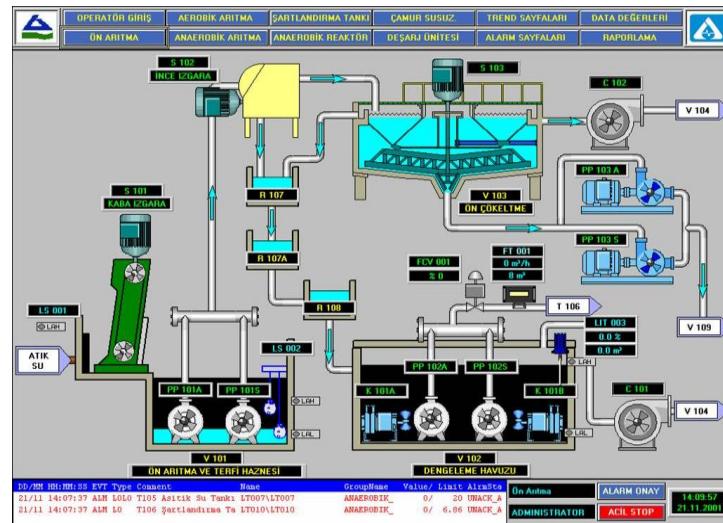
Aunque algunos de sus productos superan en potencia y capacidad de almacenaje a los de SIEMENS, esta marca sigue siendo un referente. Algunos de los productos estrella de Schneider electric están superando a los SIEMENS debido a la estrategia de mercado que ha impulsado a Schneider a introducir un elevado grado de conectividad estándar, como TCPIP, y lenguajes de programación tipo C en el entorno de PLC's industriales. Esta decisión de los PLC's de Schneider está revolucionado el mundo del control industrial desde hace unos años.

Uno de los PLC's que ha marcado el despegue de las tecnologías de comunicación en el mundo del control industrial es el famoso Momentum M1 de Schneider, que dotado de puerto Ethernet, RS-232, RS-485 y un lenguaje de programación parecido al Pascal rompió los esquemas de la mayoría de programadote de PLC acostumbrados a trabajar con programación ladder logic.

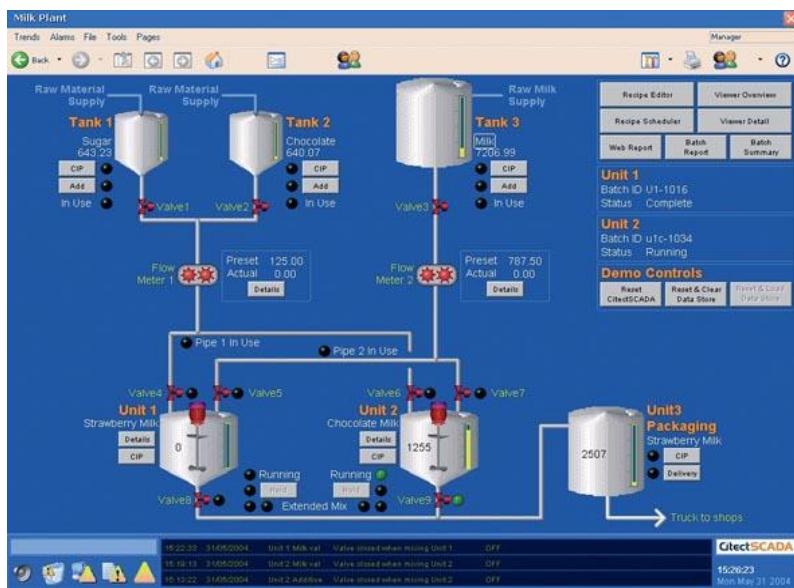


PLC Mmometum M1

Poco a poco los ordenadores han ido tomando mayor presencia en los procesos de producción y la posibilidad de disponer de programas capaces de presentar la información de los procesos que ejecutan los PLC's en tiempo real, abrió un nuevo mercado al que las grandes corporaciones se han lanzado de lleno con programas potentísimos capaces de mostrar, gobernar y historizar procesos como el PCS7 de Siemens o el Citec de Schneider.



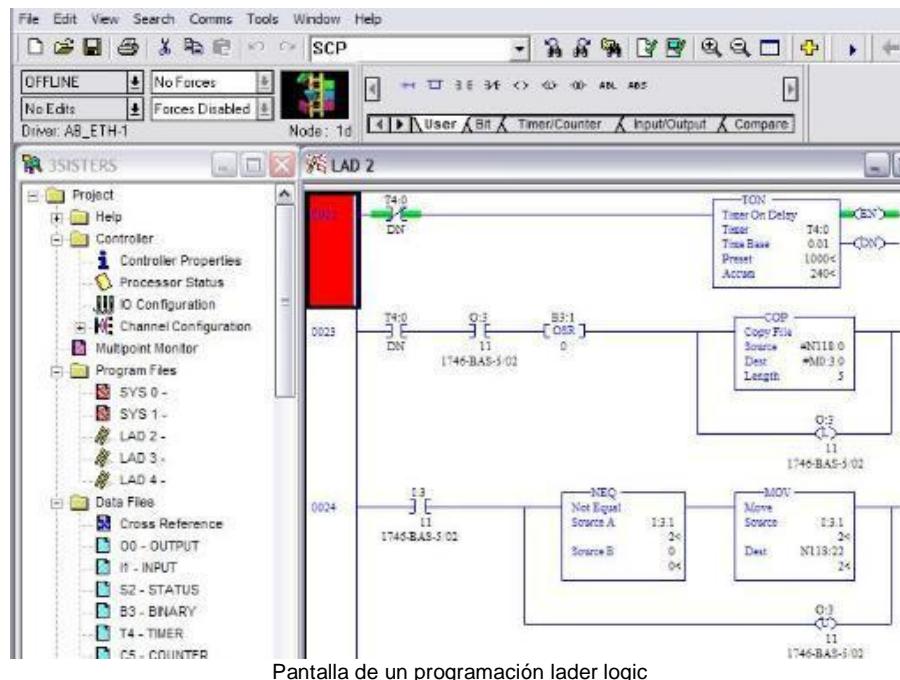
Pantalla de un proceso industrial con PCS7



Pantalla de un proceso industrial con Citec

Estos cambios están acercando a los profesionales de la informática al mundo de la automatización industrial de forma cada vez más evidente. Los PLC's son servidores WEB, se programan en C y pueden gestionar bases de datos.

El paradigma es que las empresas están incorporando las nuevas tecnologías a sus productos para ofrecer mejores prestaciones pero aun siguen con la política de cierre absoluto de sus protocolos y códigos.



A raíz de esta aparente inconsistencia, están apareciendo tímidamente algunas plataformas basadas en Linux para control industrial que utilizan las capacidades de gestión de tiempo real que ofrece Linux para crear kernels emuladores de PLC.

El gran problema de estas nuevas plataformas es la fiabilidad del hardware y los desarrollos, motivo por el cual su implantación es muy pequeña y solo se utilizan en situaciones muy particulares.

Todo este escenario de fabricantes de PLC's, SCADAS, Gestores documentales, CAD's, etc.. y la cantidad de usuarios con perfiles diferentes conduce a que la coordinación del trabajo y la capacidad de reutilizar los desarrollos se consideren imposibles y queden ceñidos a gamas de productos, empresas instaladoras e incluso a grupos de trabajo dentro de las mismas empresas.

Actualmente Internet a forzado a una mayor y mejor comunicación entre los usuarios que reclaman un esfuerzo para estandarizar formas de programación, visualización, gestión documental y colaboración.

Empresas como SIEMENS ha optado por crear gamas de productos verticales que cubren todas las necesidades desde planificación, programación de PLC's/SCADA y reutilización y documentación. El problema de estos paquetes es que son prohibitivos y complejos, aparte de propietarios.

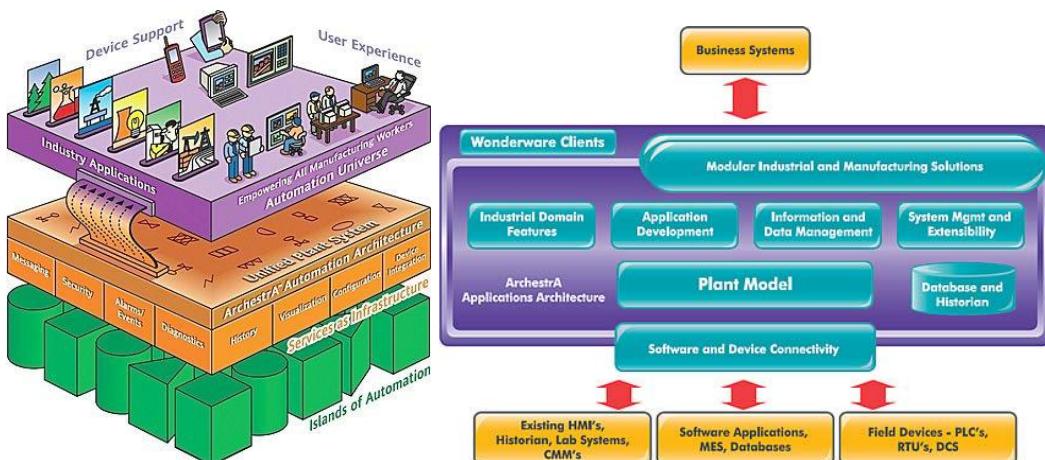
Aun con la presencia en el mercado de estos paquetes, no existe una herramienta de gestión de proyecto unificada para entornos de control industrial. Estas herramientas comúnmente conocidas como PLM (Project Live Manager) comportan la interrelación de las diferentes partes de un proyecto como documentación, arquitectura de red, modelos de instalación,

estandarización de programas, etc.. Una de las mejores en este momento es el entorno EPlan-Platform, que pretende establecer vínculos técnico-comerciales con las diferentes plataformas de mercado e integrar capacidad de comunicación con estas.



Uno de los grandes problemas de EPlan Platform es que nació como un CAD-Eléctrico y no tiene la filosofía de PLM de control de procesos, sino de un “engineering center” para los proyectos de diseño de materiales no de software.

Una de las mejores aproximaciones a una solución, es la propuesta por la empresa americana de SCADAS Wonderware y su nuevo producto “Industrial Application Server”, el cual a efectuado algunos progresos en esa dirección, pero aun está muy lejos de la solución PLM completa.



En definitiva la preceptiva de disponer de un PLM orientado a proyectos de control industrial en su vertiente de programación, con métodos de comparición de conocimiento y capacidad de trabajar con diferentes plataformas tanto hardware (PLC), software (SCADA), como documental, es actualmente un sueño.

## 2.3 Objetivos y alcance previstos.

Uno de los objetivos principales de un entorno de trabajo unificado en un portal Web es la accesibilidad de los desarrollos que se están llevando a cabo. En el caso que nos ocupa, un sistema basado en un portal permite tanto al desarrollador como al cliente interactuar y mantenerse informados del estado del proyecto y los flujos de trabajo en curso. Así mismo poder disfrutar

de un entorno de trabajo privado ligado a cuentas de usuario. Este entorno web tiene como objetivo, además de sus funcionalidades, tener un entorno de los usuarios fácil de usar, cómodo y agradable visualmente.

Se pueden conseguir varias ventajas de este sistema, de las cuales se pueden destacar la coordinación y ahorro de trabajo, además de la reutilización de dicho trabajo.

En cuanto a la información, se estima un almacenamiento detallado y adicional de cada componente hardware, además de unificarlo en única base de datos.

## 2.4 Aproximación a la tecnología de proyecto.

A lo largo de la historia, el hombre ha deseado que las máquinas realicen el trabajo más pesado y repetitivo de forma automática. Esta necesidad surge del afán de dedicar más tiempo a tareas más complejas y por lo tanto más gratificantes.

Aunque en los orígenes de la historia ya se diseñaron sistemas automáticos para tareas simples como mantener el nivel de líquido de un depósito mediante una válvula regulada por una boya de flotación, no se considera que empieza la automatización industrial hasta 1750 con el inicio de la revolución industrial.

1745: Máquinas de tejido controladas por tarjetas perforadas Jackard.

1817-1870: Máquinas especiales para corte de metal.

1863: Primer piano automático, inventado por M. Fourneaux.

1856-1890: Sir Joseph Whitworth enfatiza la necesidad de piezas intercambiables.

1870: Primer torno automático, inventado por Christopher Spencer.

1940: Surgen los controles hidráulicos, neumáticos y electrónicos para máquinas de corte automáticas.

1945-1948: John Parsons comienza investigación sobre control numérico.

1960-1972: Se desarrollan técnicas de control numérico directo y manufactura computadorizada.

Aunque inicialmente la “inteligencia” de estos dispositivos se basaba en combinaciones de relés usados como puertas lógicas, rápidamente se observó que reprogramar los circuitos de relés era una tarea ardua y frecuentemente se producían errores.

A principios de los años 70 aparecieron los primeros calculadores lógicos, precursores de los PLC's, en el sector de la automoción en Estados Unidos. Estos equipos tenían la ventaja que no hacía falta recablear relés para cambiar el programa de ejecución ya que disponían de consolas para entrar las condiciones de activación de las salidas en función de las entradas.

En 1969 la División Hydramatic de la General Motors instaló el primer PLC para reemplazar los sistemas inflexibles alambrados usados entonces en sus líneas de producción.

Ya en 1971, los PLCs se extendían a otras industrias y, en los ochentas, ya los componentes electrónicos permitieron un conjunto de operaciones en 16 bits,-comparados con los 4 de los 70s -, en un pequeño volumen, lo que los popularizó en todo el mundo.

Aunque durante muchos años estos PLC's (Programable logic Controller)

fueron patrimonio de los electricistas, con el paso del tiempo estos elementos han evolucionado hasta convertirse en pequeños ordenadores capaces de efectuar gran numero de cálculos.

Uno de los pasos más importantes en el control por PLC's fue asumir la tareas de los controladores PID (Proporcional integrador Derivador) para regulación de procesos continuos. Esto fue posible gracias a las grandes capacidades de calculo entregadas por los microprocesadores modernos.

Actualmetne los PLC's se están lanzando al mundo de las programación orientada a objetos, complejos buses de comunicación, bases de datos , servicios WEB, etc.. Es decir los PLC's ya no son patrimonio de los electricistas.

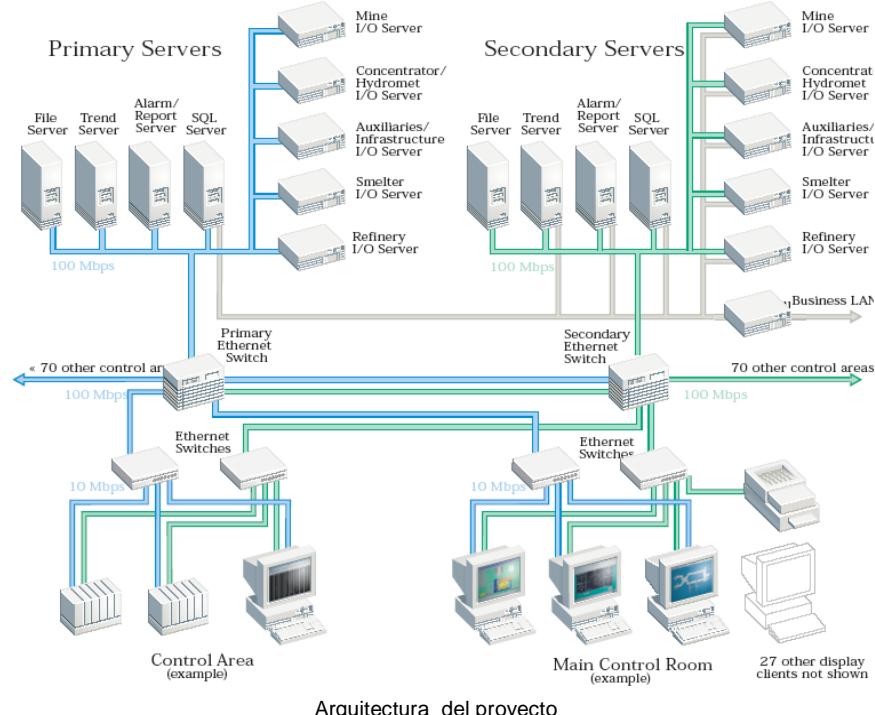
Con la aparición de los ordenadores a bajo coste y las posibilidades graficas avanzadas vieron la luz los programas SCADA (Supervisory Control and Data Acquisition), estos programas permitían comunicar los PLC's y disponer de información en tiempo real de los procesos y estado de las variables internas del equipo. Esto supuso una revolución en todos los entornos de procesos industriales, ya que permitió efectuar estudios muy precisos del comportamiento de las líneas de producción mientras estas se producían. Por primera vez la información de los procesos de producción se podía almacenar, visualizar y analizar de forma precisa.

Actualmente existen gran número de aplicaciones de estas características en el mercado capaces de monitorizar y gobernar instalaciones de cientos de miles de señales en tiempo real.

La aplicación más grande de SCADA en la actualidad es el sistema de control de la mina Olympic Dam de Australia basado en el sistema Citec. Este sistema control más de 400.000 variables en tiempo real. El equipo que desarrolló el proyecto contaba de más de 40 personas y trabajo durante 18 meses.



Mina Olympic Dam, Australia



Arquitectura del proyecto

Si ir tan lejos el sistema de control de la nueva terminal T1 del aeropuerto de Barcelona controla 198.000 variables en tiempo real en una plataforma de más de 30 servidores y 1000 PLC's compartiendo información.



Terminal T1 Aeropuerto de Barcelona

En proyectos de esta envergadura es cuando la organización, coordinación y capacidad de gestión de un entorno unificado por un PLM orientado a sistemas de control, se hace sencillamente indispensable.

### 3. Estudio de viabilidad.

#### 3.1 Introducción

##### 3.1.1 Tipología y palabras clave.

Automatización, Bases de datos, topología, Web, HTML, PHP, MySQL, Java.

##### 3.1.2 Descripción.

El módulo de modelado Hardware es la parte de GISCMO, donde se almacena la información de los elementos hardware de un sistema determinado. Este módulo almacena dicha información mediante bases de datos, además de ser accesible desde un portal web.

### 3.2 Objetivos del proyecto.

#### 3.2.1 Objetivos del proyecto.

- O.1 Seguimiento del proyecto en todo momento vía web.
- O.2 Gestión y control de permisos de los usuarios para acceder a la aplicación.
- O.3 Espacio propio de trabajo ligado a las cuentas de usuario.
- O.4 Estandarización en la creación de componentes.
- O.5 Reutilización de trabajo.
- O.6 Gestión y aprovechamiento del trabajo.
- O.7 Almacenamiento de la información de manera eficiente.
- O.8 Centralizar la información en una única base de datos.
- O.9 Crear un portal web agradable visualmente e intuitivo de usar.
- O.10 Acceso a la aplicación mediante web, desde cualquier ubicación.
- O.11 Facilitar la comunicación entre los usuarios mediante un foro.

#### 3.2.2 Priorización de los objetivos del proyecto.

Objetivo	Crítico	Prioritario	Secundario
O.1	X		
O.2	X		
O.3	X		
O.4	X		
O.5	X		
O.6	X		
O.7	X		
O.8	X		
O.9			X
O.10	X		
O.11		X	

### 3.3 Definiciones, acrónimos y abreviaturas.

GISCMO (Gestor integrado de sistemas de control y modelado de objetos)  
 PLC (Programmable Logic Controller), PHP (Personal Home Page), SQL (Structured Query Language), HTML (HyperText Markup Language).

### 3.4 Partes interesadas.

#### 3.4.1 Stakeholders.

nombre	Descripción	Responsabilidad
<b>A</b>	Responsable de entidad	Participa en la definición del proyecto y efectúa un seguimiento
<b>B</b>	Responsable contable	Define los requisitos y funcionalidades. Seguimiento del proyecto.
<b>C</b>	Director de proyecto	Supervisa el trabajo del alumno y evalúa el proyecto

#### 3.4.2 Perfiles de usuario.

nombre	Perfil	Responsabilidad
<b>U1</b>	Administrador	Gestiona y controla el sistema, gestión de usuarios, responsable contable, relaciones con las entidades financieras.
<b>U2</b>	Desarrollador	Desarrolla las fichas de componentes y su contenido.
<b>U3</b>	Visitante	Observa el estado del trabajo efectuado por el desarrollador.
<b>U4</b>	Validador	Supervisa las plantillas y contenido finalizado por el desarrollador y lo publica.

#### 3.4.3 Project Team.

nombre	Descripción	Responsabilidad
<b>A</b>	Director de proyecto	Define, gestiona, planifica y controla el proyecto.
<b>B</b>	Analista	Colabora con el jefe de proyecto en el estudio de viabilidad y planificación. Analiza la aplicación: arquitectura, metodología, especificaciones, estándares,... Participa en el diseño e implementación.
<b>C</b>	Programador	Diseña y desarrolla la aplicación de acuerdo con el análisis y planificación prevista. Participa en el proceso de validación e implementación.
<b>D</b>	Técnico de pruebas	Participa en el diseño de las pruebas internas y externas. Participa en el proceso de control de calidad.

### 3.5 Referencias.

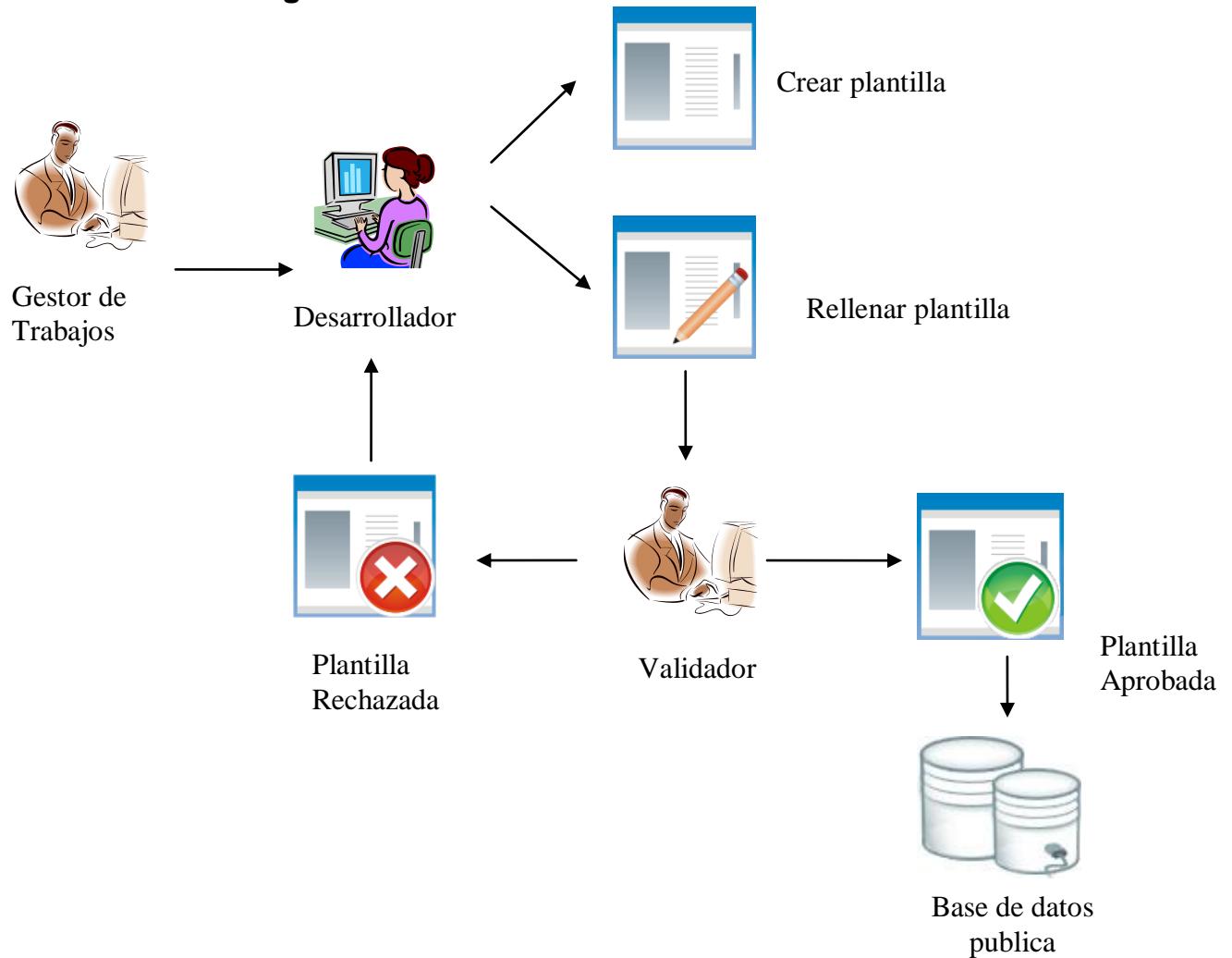
Normativa de proyectos de ingeniería técnica:  
<http://www.uab.cat/Document/330/254/projectes-normativa20090630.pdf>  
 LOPD:  
<https://www.agpd.es/portalweb/canaldocumentacion/legislacion/estatal/indice-ides-idphp.php>  
[www.w3schools.com/](http://www.w3schools.com/)

### 3.6 Estudio de la situación actual

#### 3.6.1 Contexto.

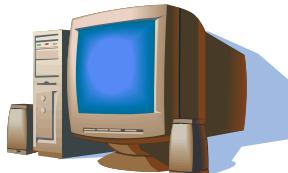
Se dispone de un módulo de modelado hardware desarrollado con el software EPS2008. La gestión de las fichas así como su información es implementada con dicho programa y gestionada por varias bases de datos implementadas en Microsoft Access®.

#### 3.6.2 Lógica del sistema.



### 3.6.3 Descripción física.

Estructura informática básica.



Ordenadores



Base de datos

### 3.6.4 Usuarios y/o personal del sistema.

nombre	Perfil	Responsabilidad
<b>U1</b>	Administrador	Gestiona y controla el sistema, gestión de usuarios, responsable contable, relaciones con las entidades financieras.
<b>U2</b>	Desarrollador	Desarrolla las fichas de componentes y su contenido.

### 3.6.5 Diagnóstico del sistema.

#### Deficiencias:

- Acceso limitado por la necesidad de utilizar un software instalable.
- Bases de datos anticuadas, lentas y limitadas.
- Difícil utilización para el usuario.
- Falta de estándares.
- Posibilidad de redundancia.
- Falta de comunicación.

#### Mejoras:

- Acceso ilimitado desde cualquier lugar, ya que se usa vía web. Solo se necesita un identificador.
- Gestión de usuarios con su espacio de trabajo.
- Utilización de MySQL en las bases de datos (mejora de velocidad, amplitud, interconexiones, etc).
- Gestión intuitiva del entorno para los usuarios.
- Utilización de plantillas y estándares.
- Coordinación y comunicación de usuarios.
- Supervisión de fichas mediante un validador.

### 3.6.6 Normativas y legislación.

- LOPD: Ley orgánica de protección de datos.
- Normativa de proyectos de fin de carrera de la EI.
- Ley de protección intelectual.

## 3.7 Requisitos del proyecto.

### 3.7.1 Requisitos funcionales.

Estos requisitos recogen las necesidades expresadas por las diferentes partes interesadas en el proyecto y una serie de directrices y objetivos que deben ser cumplidos para dar las funcionalidades esperadas a la plataforma de desarrollo.

- Altas, bajas y login de los usuarios para acceder a la aplicación.
- Cuentas de usuarios independientes.
- Altas, bajas y modificaciones de las plantillas de los componentes.
- Altas, bajas y modificaciones de las fichas de componentes
- Centralizar la información en una única base de datos.
- Crear un portal web agradable visualmente e intuitivo de usar.
- Acceso a la aplicación mediante web, desde cualquier ubicación.
- Facilitar la comunicación entre los usuarios mediante un foro.

### 3.7.2 Requisitos no funcionales.

Estos requisitos recogen necesidades no explícitamente vinculadas al diseño de las funcionalidades pero que deben ser cumplidas para el correcto desarrollo del proyecto.

- Estandarización en la creación de componentes.
- Reutilización de trabajo.
- Gestión y aprovechamiento del trabajo.
- Cumplimiento de la LOPD.
- Cumplimiento de las leyes de propiedad intelectual.
- Cumplimiento de la normativa de proyectos de fin de carrera de la EI.
- Normalización de la base de datos según el estándar SQL 99.
- Utilización de los estándares web.

### 3.7.3 Restricciones del sistema.

Estas restricciones son necesarias para el correcto funcionamiento del software en diferentes navegadores web, además de la fecha límite que debe ser cumplida.

- Ha de soportar los navegadores Opera, Firefox, I.Explorer, Chrome.
- Resolución óptima para estos navegadores.
- El proyecto tiene que estar finalizado antes del 30 de junio de 2010.

## 3.8 Alternativas y selección de la solución.

### 3.8.1 Alternativa 1.

Utilización del un lenguaje de programación estándar orientado a la generación de aplicaciones ejecutables locales, como Visual Basic para desarrollar un software estático y bases de datos en Access.

### **3.8.2 Alternativa 2.**

Utilización del MySQL i un portal web mediante java, php...

### **3.8.3 Solución propuesta.**

Comparando las dos alternativas, un lenguaje de programación estándar como Visual Basic, no ofrece la libertad de movimiento que este tipo de aplicación requiere. Al tener que utilizar un software ligado a su instalación en una plataforma determinada, se dificulta la comunicación entre usuarios y la compartición de datos. En consecuencia el control de usuarios y sus espacios de trabajo, resulta del todo imposible.

Un problema adicional es que algunas versiones de Visual Basic no son gratuitas.

Por otro lado, la diferencia entre MySQL y Access, es que MySQL es mucho más rápido y puede manejar un volumen de datos mayores y con más facilidad.

Por lo tanto la solución escogida es la alternativa 2, utilizando MySQL y portal Web.

## **3.9 Planificación del proyecto.**

El proyecto se desarrollará desde Noviembre del 2009 al 28 de Abril 2010.

Con una dedicación de 10 horas semanales, haciendo un total de 121 días.

Fecha inicio prevista: 1 de Noviembre de 2009.

Fecha finalización prevista: 28 de Abril 2010.

Herramientas de planificación y control: Microsoft Project.

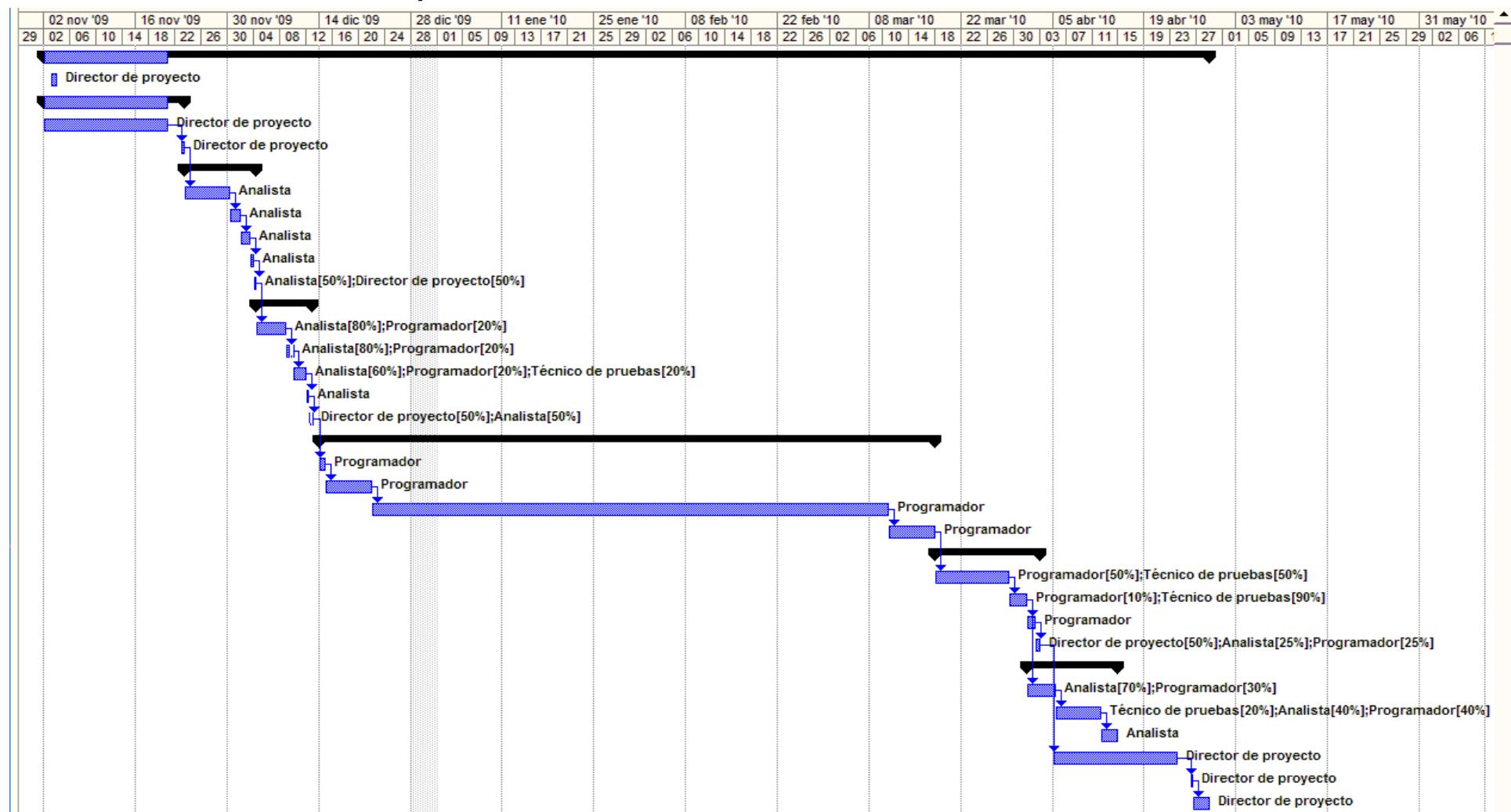
### **3.9.1 Recursos del proyecto.**

<b>Recursos humanos</b>		<b>Valoración</b>
Director de proyecto		100€/h
Analista		50€/h
Programador		30€/h
Técnico de pruebas		20€/h

### 3.9.2 Tareas del proyecto.

<b>nº</b>	<b>Descripción</b>	<b>Durada</b>	<b>Recursos</b>	<b>Pred.</b>
1	Inicio del proyecto. Asignación y matriculación	2h	DP	
2	<b>Planificación</b>	31h		
3	Estudio de viabilidad	30h	DP	
4	Aprobación Estudio viabilidad	1h	DP	3
5	<b>Análisis de aplicación</b>	6dias		
6	Análisis de requisitos	10h	A	4
7	Análisis de datos	5h	A	6
8	Análisis de seguridad	5h	A	7
9	Documentación de análisis	3h	A	8
10	Aprobación de análisis	1h	A,DP	9
11	<b>Diseño de aplicación</b>	4,25dias		
12	Diseño base de datos	5h	A80%, P20%	10
13	Diseño aplicación	3h	A80%,P20%	12
14	Diseño de pruebas	5h	A60%,P60%,T20%	13
15	Documentación	3h	A	14
16	Aprobación diseño	1h	DP50%,A50%	15
17	<b>Desarrollo de aplicación</b>	30,5días		
18	Preparación entorno desarrollo	2h	P	16
19	Configuración base de datos	10h	P	18
20	Funcionalidades	100h	P	19
21	Desarrollo web	10h	P	20
22	<b>Test y pruebas</b>	6días		
23	Pruebas funcionalidades	15h	P50%,T50%	21
24	Pruebas riesgos	5h	P10%,T90%	23
25	Documentación test	3h	P	24
26	Aprobación desarrollo	1h	DP50%P25%A25%	25
27	<b>Implantación</b>	5días		
28	Instalación	5h	A70%,P30%	25
29	Pruebas reales	10h	T20%,P40%,A40%	28
30	Formación de usuarios	5h	A	29
31	Memoria de proyecto	30h	DP	26
32	Cierre de proyecto	1h	DP	31
33	Defensa de proyecto	5h	DP	32

### 3.9.3 Planificación temporal.



Total horas: 256h aprox.

## 3.10 Evaluación de riesgos.

### 3.10.1 Lista de riesgos.

A continuación se detallan los diferentes riesgos que ponen en peligro el correcto funcionamiento y desarrollo del proyecto.

- R1: Planificación temporal optimista:** Estudio de viabilidad. No se acaba en la fecha prevista, aumentan los recursos.
- R2: Falta de alguna tarea necesaria:** Estudio de viabilidad. No se cumplen los objetivos del proyecto.
- R3: Presupuesto poco ajustado:** Estudio de viabilidad. Menos calidad y pérdidas económicas.
- R4: Cambio de requisitos:** Estudio de viabilidad, análisis. Retraso en el desarrollo y los resultados.
- R5: Equipo del proyecto demasiado reducido:** Estudio de viabilidad. Retraso en la finalización del proyecto, no se cumplen los objetivos del proyecto.
- R6: Dificultad para acceder a los stakeholders:** Estudio de viabilidad, análisis, pruebas, formación. Faltan requisitos o son inadecuados, retrasos, insatisfacción de los usuarios.
- R7: No se hace correctamente la fase de test:** Desarrollo, implantación. Falta de calidad, deficiencias en la operativa, insatisfacción de usuarios, perdida económica.
- R8: Incumplimiento de alguna norma, reglamento o legislación:** En cualquier fase. No se cumplen los objetivos, repercusiones legales.
- R9: Falta de implantación de medidas de seguridad:** Estudio de viabilidad, análisis, desarrollo. Perdida de información, incumplimiento legal, pérdidas económicas.
- R10: Abandonamiento del proyecto antes de la finalización:** En cualquier fase. Pérdidas económicas, frustración.

### 3.10.2 Catalogación de riesgos.

	<i>Probabilidad</i>	<i>Impacto</i>
<b>R1</b>	Alta	Crítico
<b>R2</b>	Alta	Crítico
<b>R3</b>	Alta	Crítico
<b>R4</b>	Alta	Marginal
<b>R5</b>	Alta	Crítico
<b>R6</b>	Baja	Crítico
<b>R7</b>	Alta	Crítico
<b>R8</b>	Mediana	Crítico
<b>R9</b>	Alta	Crítico
<b>R10</b>	Mediana	Catastrófico

### 3.10.3 Plan de contingencia.

	<b><i>Solución que se necesita adoptar</i></b>
<b>R1</b>	Aplazar alguna funcionalidad, afrontar posibles pérdidas, hacer un seguro.
<b>R2</b>	Revisar el estudio de viabilidad, modificar la planificación.
<b>R3</b>	Renegociar con el cliente, afrontar posibles pérdidas, hacer un seguro.
<b>R4</b>	Renegociar con el cliente. Aplazar la funcionalidad, modificar la planificación y presupuesto.
<b>R5</b>	Pedir un aplazamiento, negociar con el cliente, afrontar pérdidas.
<b>R6</b>	Usar un calendario de reuniones, mejorar el contacto con el cliente
<b>R7</b>	Diseñar los test con antelación, realizar test automáticos, negociar contratos de mantenimiento, dar garantías, afrontar pérdidas económicas.
<b>R8</b>	Revisar las normas y legislación, consultar a un experto, afrontar posibles repercusiones penales.
<b>R9</b>	Revisar la seguridad en cada fase, aplicar políticas de seguridad activas.
<b>R10</b>	No tiene solución.

## 3.11 Presupuesto.

### 3.11.1 Estimación costes de personal.

<b><i>Recursos humanos</i></b>	<b><i>Horas trabajo</i></b>	<b><i>Coste</i></b>
Director de proyecto	70h	7.000€
Analista	49h	2.450€
Programador	141h	4.230€
Técnico de pruebas	15h	300€

Total: 13.980€

### 3.11.2 Estimación coste de los recursos.

	<b>Coste de amortización</b>	<b>Coste unitario</b>	<b>Periodo amortización</b>	<b>Periodo utilización</b>
Amortización PC	100€	999€	30m	3m
Amortización MSOffice	20,8€	250€	36m	3m
Amortización MSPProject	30€	360€	36m	3m

### 3.11.3 Resumen y análisis de costes de beneficios.

<b>Tipo de Coste</b>	<b>Precio</b>
Coste de desarrollo del proyecto	13.980€
Coste amortización del material	150,8€
<b>Total</b>	<b>14.130,8€</b>

## 3.12 Conclusiones.

Al ser un software que no existe prácticamente en el mercado, cuya utilidad es además importante y sumando la novedad de acceso en web, se puede observar que los beneficios sobrepasarán las perdidas en un tiempo relativamente corto.

La estimación del tiempo exacto en que sea beneficioso es complicada, por lo que es un proyecto con bastantes riesgos inherentes a la naturaleza del I+D y tantear un terreno no muy explorado.

A pesar de ello, para el tipo de software que se desarrollará no es un proyecto excesivamente caro, por lo que se puede deducir que ES VIABLE.

## 4. Análisis del proyecto.

### 4.1 Requerimientos funcionales.

#### 4.1.1 Visión general.

Como dijimos este proyecto tiene como objetivo desarrollar un aplicativo en formato web capaz de dar soporte al modelado de equipos hardware para su posterior publicación en librerías.

Nuestro objetivo comprende la creación y posterior gestión de modelos de equipos hardware para ser utilizados por los proyectistas de instalaciones de control industrial, que generen sus trabajos mediante el uso del entorno de desarrollo GISCMO.

Como módulo de modelado Hardware está diseñado para contener la información de los elementos hardware que se encuentren en un sistema determinado mediante bases de datos como: PLCS, buses, routers, etc. Dentro de cada componente se deberá guardar diversa información como una matrícula única que identificará el objeto, fabricante, software utilizado, conexiones, etc.

Además existirán diferentes niveles de usuarios, con su propia área de trabajo y en los que cada uno desempeñará una función específica. Existirá al menos un administrador, un desarrollador, un validador y un visitante. Sus funciones serán:

- Administrador: Administración de permisos del sistema, gestión de usuarios.
- Desarrollador: Desarrolla las fichas de componentes y su contenido.
- Validador: Observa el estado del trabajo efectuado por el desarrollador y gestión de librerías publicadas de modelos.
- Visitante: cliente que observa el estado del trabajo efectuado por el desarrollador.

#### 4.1.2 Requisitos funcionales de usuario.

A continuación se presentan los requisitos funcionales de cada tipo de usuario, con fin de detallar los roles o capacidades de cada uno de ellos en el proyecto.

##### 4.1.2.1 Usuario administrador.

- Login
- Comentar, borrar y editar mensajes del foro.
- Crear, borrar y editar categorías del foro.
- Establecer jerarquía de permisos en el foro para distintos usuarios.
- Altas y bajas de usuarios para acceder a la aplicación.
- Altas y bajas de usuarios para acceder al foro.
- Gestión de cuentas de usuario.
- Gestión de la base de datos.
- Añadir, eliminar y editar elementos de la base de datos.
- Ver plantillas de componentes.
- Ver fichas de componentes.

#### **4.1.2.2 Usuario validador.**

- Login.
- Comentar en el foro.
- Publicar plantillas de componentes en la base de datos.
- Publicar ficha de componentes en la base de datos.

#### **4.1.2.3 Usuario desarrollador.**

- Login.
- Comentar en el foro.
- Ver plantillas de componentes.
- Altas, bajas y modificaciones de sus plantillas de los componentes.
- Ver fichas de componentes.
- Altas, bajas y modificaciones de sus fichas de componentes.

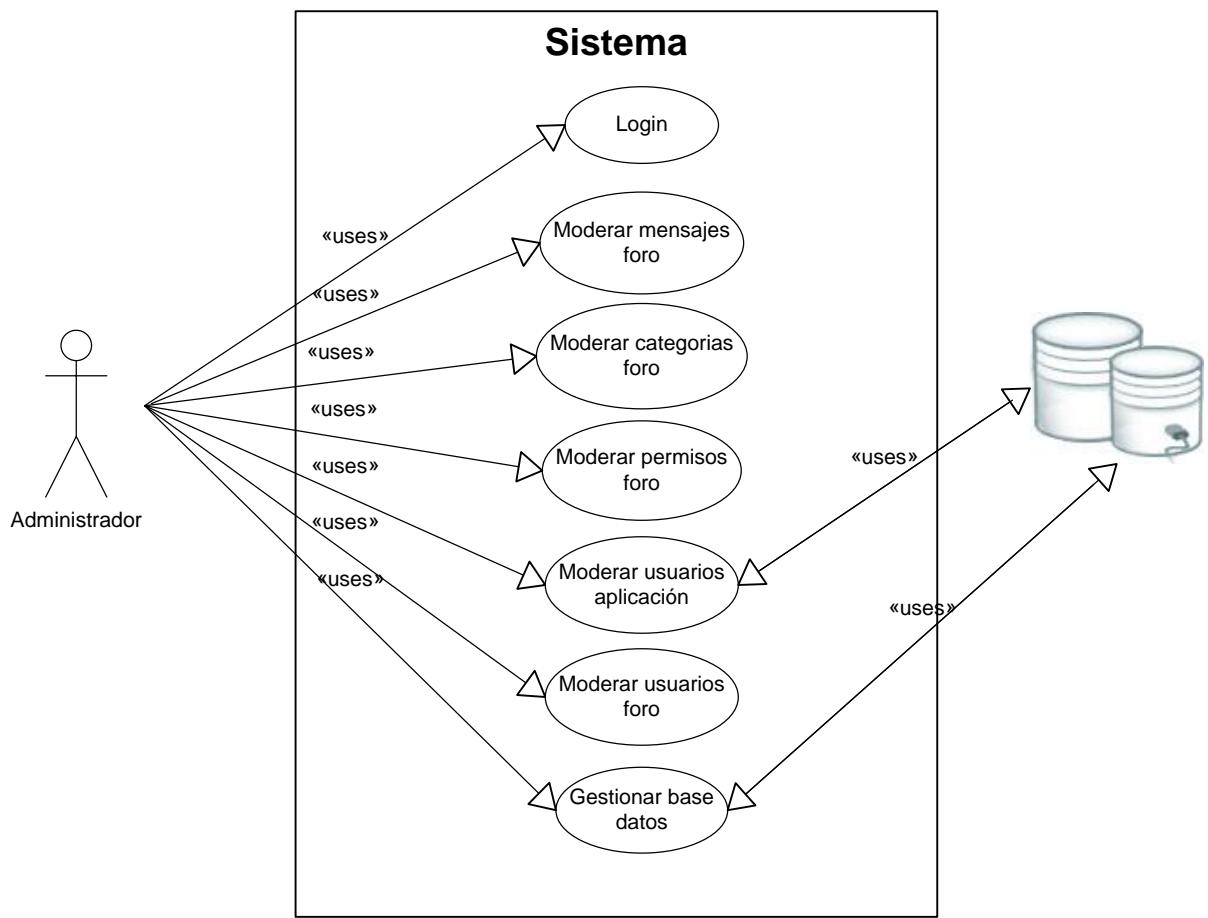
#### **4.1.2.4 Usuario visitante.**

- Login.
- Comentar en el foro.
- Ver las plantillas de los componentes.
- Ver de las fichas de componentes.

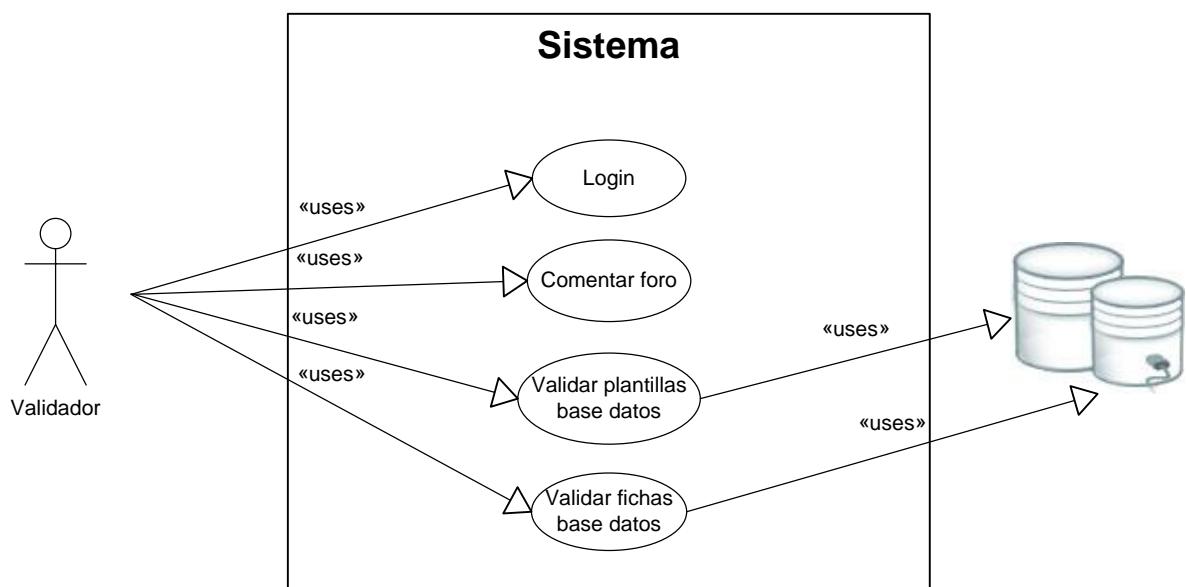
### **4.1.3 Diagramas de casos de uso.**

A continuación, se presentan los diagramas de casos de uso para cada tipo de usuario, con fin de detallar mas visualmente, los roles o capacidades de cada uno de ellos en el proyecto.

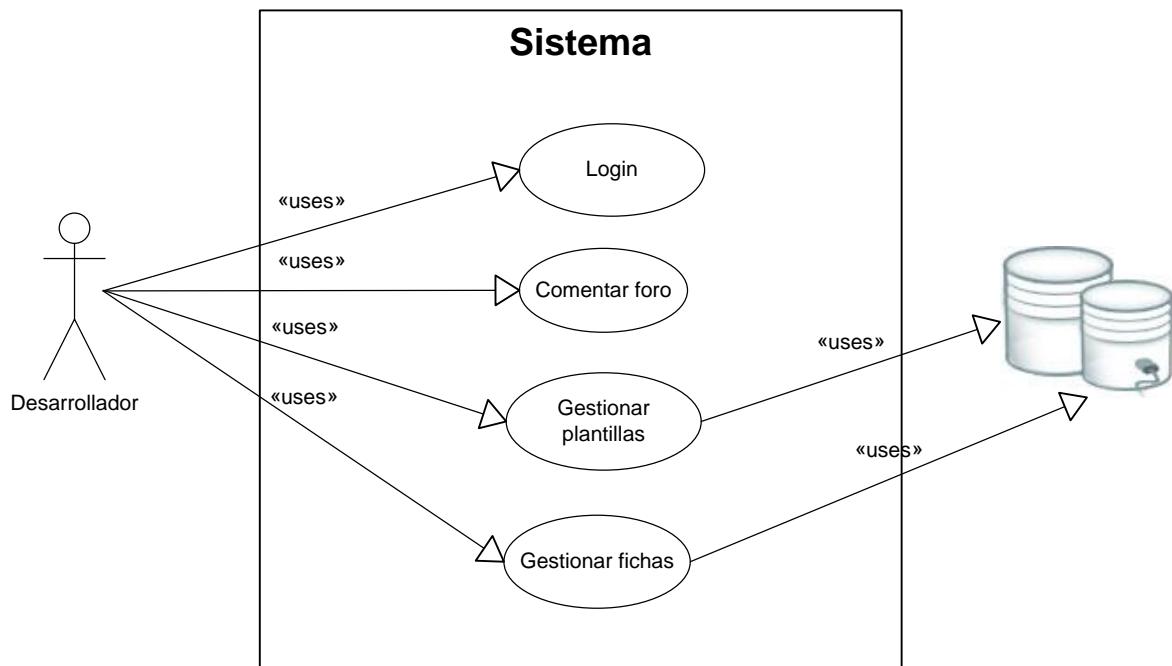
#### 4.1.3.1 Diagrama de casos de uso para el usuario administrador.



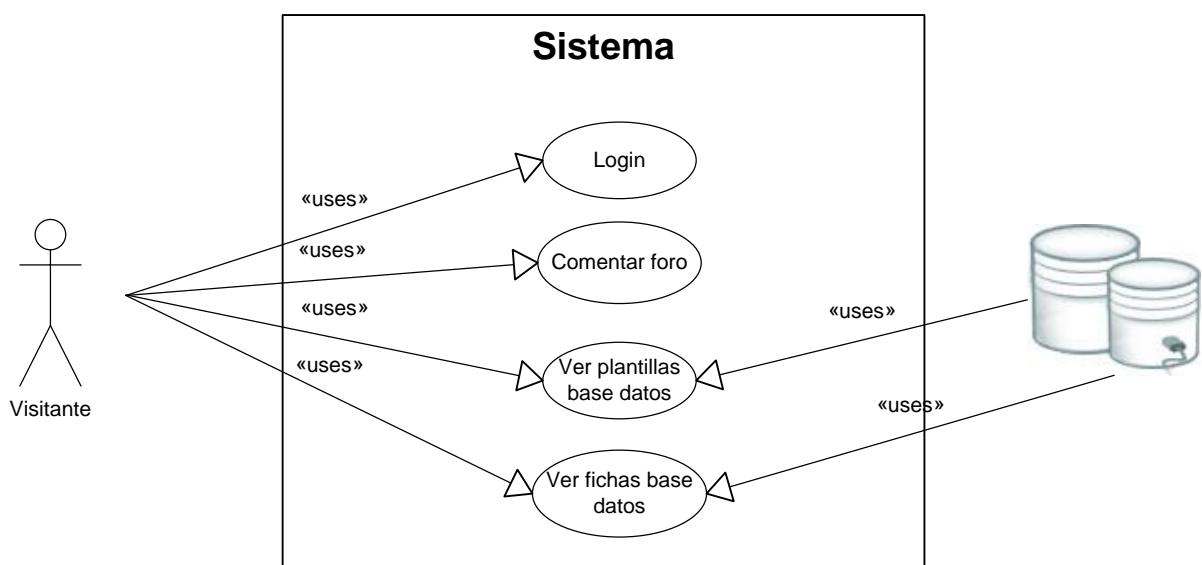
#### 4.1.3.2 Diagrama de casos de uso para el usuario validador.



#### 4.1.3.3 Diagrama de casos de uso para el usuario desarrollador.



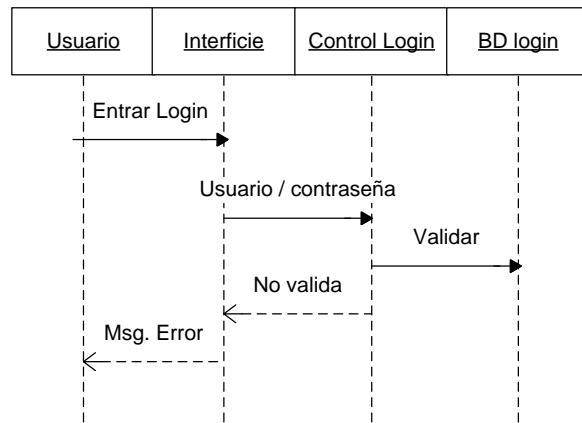
#### 4.1.3.4 Diagrama de casos de uso para el usuario visitante.



#### 4.1.4 Diagramas de secuencias.

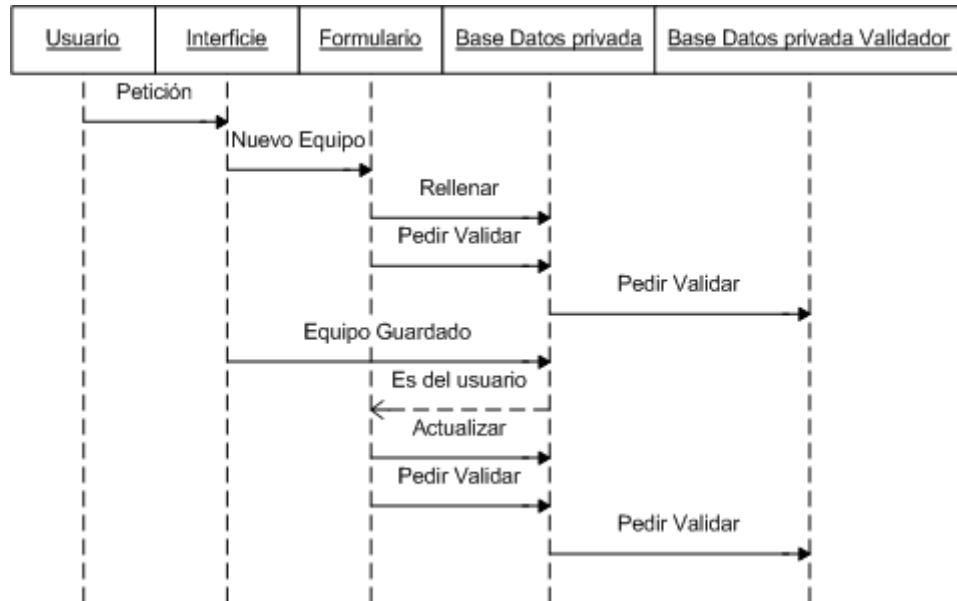
Seguidamente se muestran los distintos diagramas de secuencia para los estados más destacables. Se han omitido los más evidentes, los que son idénticos a otros y los diagramas referentes al foro.

##### 4.1.4.1 Login.



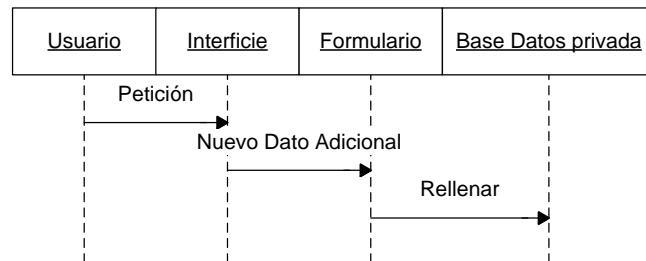
##### 4.1.4.2 Gestión de equipos.

Este diagrama de secuencia también es aplicable al diagrama que correspondería a gestión de plantillas de conexiones.



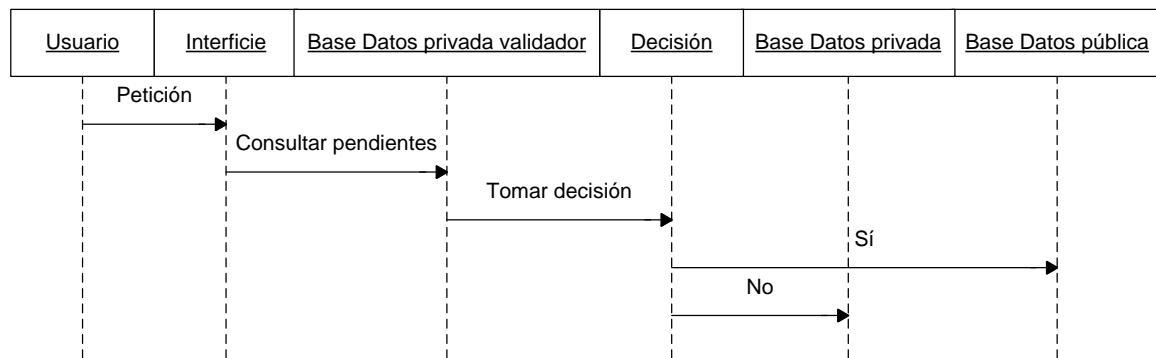
#### 4.1.4.3 Datos adicionales.

Este diagrama de secuencia también es aplicable al diagrama que correspondería a crear filiación.



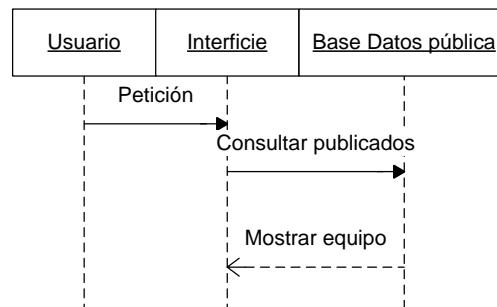
#### 4.1.4.4 Validar equipo.

Este diagrama de secuencia también es aplicable al diagrama que correspondería a validar plantilla de conexiones.



#### 4.1.4.5 Ver Equipo.

Este diagrama de secuencia también es aplicable al diagrama que correspondería a ver plantilla de conexiones.



#### **4.1.5 Menús de navegación.**

Seguidamente se listan los diferentes menús que aparecerán en la página web que permitirán navegar por las diferentes categorías de la aplicación. Cada tipo de usuario poseerá su tipo de menú acorde con su labor.

##### **4.1.5.1 Administrador.**

- Nuevo usuario.
- Ver Equipo.
- Ver Plantilla.
- Foro.
- Logout.

##### **4.1.5.2 Validador.**

- Equipos pendientes.
- Plantillas pendientes.
- Ver Equipo.
- Ver Plantilla.
- Foro.
- Logout.

##### **4.1.5.3 Desarrollador.**

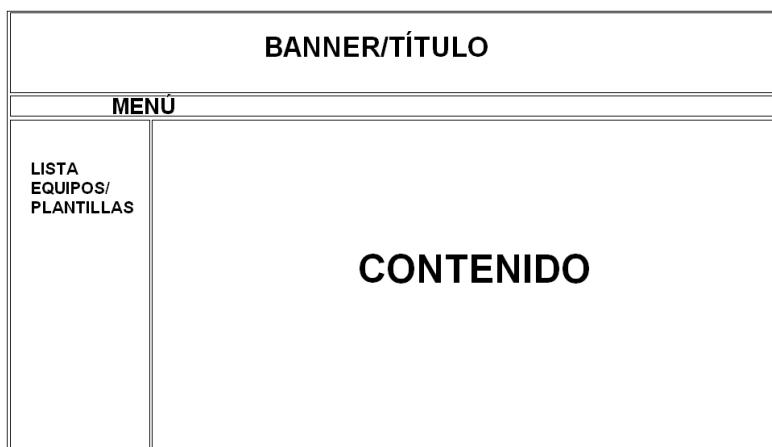
- Gestionar equipo.
- Gestionar plantilla.
- Crear filiación.
- Ver Equipo.
- Ver Plantilla.
- Foro.
- Logout.

##### **4.1.5.4 Visitante.**

- Ver Equipo.
- Ver Plantilla.
- Foro.
- Logout.

#### 4.1.6 Diseño de pantallas.

Seguidamente se muestra el diseño de pantallas. Dado que todos los usuarios tienen el mismo diseño de pantallas, se ha unificado y solo se muestra un diseño general para todos los casos.



#### 4.2 Restricciones de diseño.

Para este aplicativo se presentan diversas restricciones de diseño. Habrá únicamente cuatro tipos de usuario, que serán como se ha mencionado en capítulos anteriores, el administrado, validador, desarrollador y visitante.

Cada usuario solo podrá gestionar sus propios trabajo, los cuales habrá creado previamente, ya sean fichas de equipos, plantillas o filiaciones.

Para la gestión de fichas de equipos se impondrán las siguientes restricciones de distintos campos

En el campo de Bornas Equipo, solo se podrán seleccionar las plantillas que hayan sido validadas previamente.

En el campo de Tipo Equipo, solo se mostrarán 5 tipos de equipo:

- BUS
- CPU
- GW (Gate Way)
- HMI (interface hombre-máquina)
- SLOT

En el campo de código de dibujo, solo se podrán seleccionar las imágenes previamente definidas.

La última restricción para la gestión de equipos el campo de Requerimiento de CPU, solo podrá ser Sí o No.

Para la gestión de plantillas se establece una restricción que será en el campo de Tipo de Señal, el cual solo podrán ser

- AI, Analog input (entrada analógica)
- AO, Analog output (salida analógica)

- DI, Digital input (entrada digital)
- DO, Digital output (salida digital)

Finalmente, el usuario validador solo podrá validar o rechazar los elementos en estado pendiente de validar, sin poder modificar ningún campo, excepto el campo Notas en gestión de equipos.

## 5. Diseño.

### 5.1 Introducción.

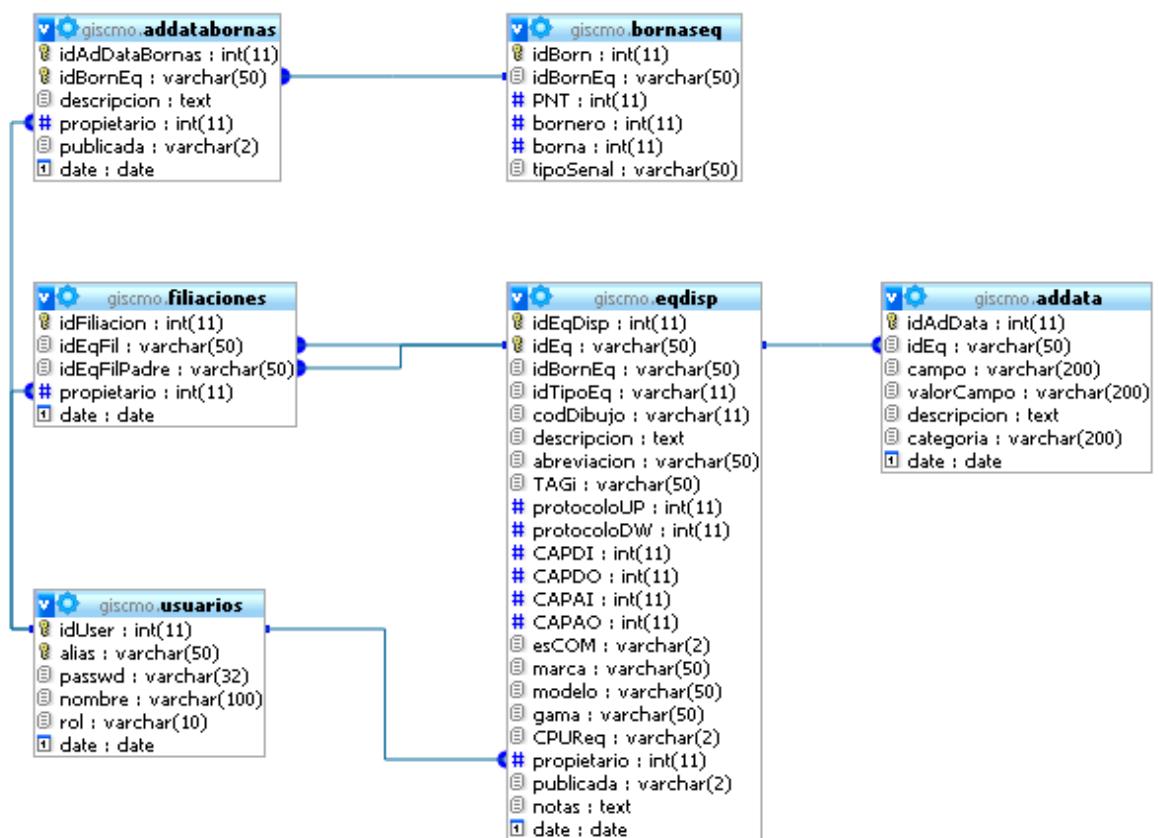
#### 5.1.1 Selección de bases de datos.

Se ha decidido crear las bases de datos en MySQL por la rapidez en consultas que ofrece, por la fácil implementación y gestión, y por la gran compatibilidad que ofrece respecto a otros sistemas.

### 5.2 Estructura de la base de datos.

Para el diseño de la base de datos, se ha decidido crear 6 tablas interrelacionadas. Estas tablas contendrán toda la información de los equipos, plantillas, usuarios, datos adicionales y filiaciones.

Se pasa a detallar el contenido principal de cada una de ellas, el motivo de las relaciones y los tipos de datos más significativos. En la figura se muestran las tablas gráficamente.



### 5.2.1 Tabla eqDisp.

*Eqdisp* es el acrónimo seleccionado para la tabla de equipos disponibles, la cual se encarga de almacenar la información de los equipos. Destacar que posee un identificador de “equipo único” de 32 bits, que es generado mediante unos cálculos matemáticos basados en la fecha, hora y microsegundos. Este campo esta diseñado para la futura integración de varias orígenes de datos generadas y gestionadas en diversas ubicaciones de trabajo independientes.

Detalle de los campos de eqDisp a continuación:

- **idEqDisp**: Identificador único de la tabla auto-incremental. Es un entero.
- **IdEq.**: Identificador de equipo único y universal, generado por operación matemática basado en fecha, hora y microsegundos. Es un varchar.
- **IdBornasEq**: Identificador de Bornas de Equipo. Guarda el identificador de la plantilla de conexiones que usa. Este campo tiene relación con el campo IdBornasEq de la tabla BornasEq. Es un varchar.
- **IdTipoEq**: Tipo de equipo. Solo puede ser CPU, BUS, GW, HMI y SLOT. Es un varchar.
- **CodDibujo**: Código del dibujo que será utilizado para mostrarlo en la web. Es un varchar.
- **Descripción**: Descripción del equipo breve. Es el *nombre* del equipo en la web. Es un varchar.
- **Abreviación**: Abreviación del nombre o descripción. Es un varchar.
- **TAGi**: Tag o etiqueta de lenguaje de marcado. Es un varchar.
- **ProtocoloUP**: Protocolo de subida. Es un entero.
- **ProtocoloDW**: Protocolo down o de bajada. Es un entero.
- **CAPDI**: Captación de digital input o de entrada digital. Es un entero.
- **CAPDO**: Captación de digital output o de salida digital. Es un entero.
- **CAPAI**: Captación de analog input o entrada analógica. Es un entero.
- **CAPAO**: Captación de analog output o salida analógica. Es un entero.

- **EsCOM:** Indica si usa señales calculadas. Solo puede ser sí o no. Es un varchar.
- **Marca:** Marca del equipo. Es un varchar.
- **Modelo:** Modelo del equipo. Es un varchar.
- **Gama:** Gama del equipo. Es un varchar.
- **CPUReq:** Requerimiento de CPU. Solo puede ser sí o no. Es un varchar.
- **Propietario:** Id del propietario que lo esté gestionando. Tiene relación a idUser de la tabla usuarios. Es un entero.
- **Publicada:** Estado del equipo. Solo puede ser sí, no o pv (pendiente de validar). Es un varchar.
- **Notas:** Notas adicionales que se pueden enviar desarrollador como validador. Es texto.
- **Date:** Fecha de la última modificación. Es una fecha.

### 5.2.2 Tabla adData.

Addata es el acrónimo de adicional data (datos adicionales). Está destinada a contener cualquier información adicional que pueda necesitar cualquier equipo. El objetivo de esta información adicional es aumentar la información de los equipos en campos específicos o de características importantes que la tabla de equipos disponibles no ofrece. Por lo tanto se busca una mayor eficiencia y eficacia en la información de equipos.

Detalle de los campos de AdData a continuación:

- **IdAdData:** Identificador único de la tabla auto-incremental. Es un entero.
- **IdEq:** Hace referencia a IdEq de la tabla eqDisp. Identifica al equipo a quien pertenece la información adicional. Es un varchar.
- **Campo:** Nombre del campo. Es un varchar.
- **ValorCampo:** Valor del campo. Es un varchar.
- **Descripción:** Descripción adicional del dato adicional. Es un varchar.
- **Categoría:** Categoría del dato adicional. Es un varchar.
- **Date:** Fecha de la última modificación. Es una fecha.

### 5.2.3 Tabla bornasEq.

BornasEq es el acrónimo de bornas de quipo. El objetivo de esta tabla es almacenar las plantillas de conexiones que serán útiles a los equipos disponibles.

Detalle de los campos de bornasEq a continuación:

- **IdBorn:** Identificador único de la tabla auto-incremental. Es un entero.
- **IdBornEq:** Identificador único que identifica la plantilla de conexiones. Es un entero.
- **PNT:** Punto de conexión. Es un entero.
- **Bornero:** Número de bornero. Es un entero.
- **Borna:** Número de la borna. Es un entero.
- **TipoSenal:** Tipo de señal. Solo puede ser AI (analog input), AO (analog output), DI (digital input) o DO (digital output). Es un varchar.

### 5.2.4 Tabla adDataBornas.

AdDataBornas es el acrónimo de Adicional Data Bornas. Esta tabla almacena la información adicional de la tabla bornas equipo. El objetivo de esta tabla es guardar distintos campos sin que se repitan numerosas veces haciendo así más eficiente la tabla de bornas equipo.

Detalle de los campos de adDataBornas a continuación:

- **IdAdDataBornas:** Identificador único de la tabla auto-incremental. Es un entero.
- **IdBornasEq:** Hace referencia a IdBornasEq de la tabla bornasEq. Identifica a la plantilla de conexiones a quien pertenece la información adicional. Es un entero.
- **Descripción:** Descripción de la plantilla. Se usa además para mostrar el nombre en la web. Es un varchar.
- **Propietario:** Id del propietario que la esté gestionando. Tiene relación a idUser de la tabla usuarios. Es un entero.
- **Publicada:** Estado de la plantilla de conexiones. Solo puede ser sí, no o pv (pendiente de validar). Es un varchar.
- **Date:** Fecha de la última modificación. Es una fecha.

### 5.2.5 Tabla de filiaciones.

Filiaciones es la tabla encargada de guardar las relaciones que hay entre equipos, guardando así los tipos de conexiones entre equipos que pueden tener.

Detalle de los campos de filiaciones a continuación:

- **idFiliacion:** Identificador único de la tabla auto-incremental. Es un entero.
- **idEqFil:** Guarda la idEq de eqDisp del *hijo* en la relación. Es un varchar.
- **idEqFilPadre:** Guarda la idEq de eqDisp del *padre* en la relación. Es un varchar.
- **Propietario:** Id del propietario que la esté gestionando. Tiene relación a idUser de la tabla usuarios. Es un entero.
- **Date:** Fecha de la última modificación. Es una fecha.

### 5.2.6 Tabla de usuarios.

Usuarios es la tabla que guarda la información de los usuarios que pueden acceder al aplicativo.

Detalle de los campos de usuarios a continuación:

- **idUser:** Identificador único de la tabla auto-incremental. Es un entero.
- **Alias:** Alias o Nick del usuario para hacer login en el aplicativo. Es un varchar.
- **Passwd:** Contraseña en codificación md5. Es un varchar.
- **Nombre:** Nombre completo del usuario. Es un varchar.
- **Rol:** Nivel de usuario o tipo de usuario. Solo puede ser administrador, validador, desarrollador o visitante. Es un varchar.
- **Date:** Fecha de la última modificación. Es una fecha.

## 6. Implementación.

### 6.1 Introducción.

Se han implementado todas las aplicaciones que se habían propuesto anteriormente. Las más características según los roles de los usuarios son:

- **Administrador:** Gestión de usuarios y foro.
- **Validador:** Validar equipos o plantillas.
- **Desarrollador:** Gestión de equipos, plantillas, datos adicionales y filiaciones.
- **Visitante:** Ver equipos y plantillas validadas.

### 6.2 Codificación de las diferentes capas de la aplicación.

#### 6.2.1 Interfaces.

A pesar de que cada tipo de usuario desarrolla un papel distinto, como ya adelantábamos en el capítulo anterior (4.1.6) la interface para todos los usuarios es la misma. Por lo que también se ha unificado en una sola captura del resultado de la aplicación.

En este caso específico se muestra *ver equipo*, que es una opción del menú, al que tienen acceso todos los usuarios.

# GISCMO

MÓDULO DE MODELADO HARDWARE

[Gestionar Equipo](#) [Gestionar Plantilla](#) [Crear Filiación](#) [Ver Equipos](#) [Ver Plantillas](#) [Foro](#) [Logout](#)

**Lista de equipos**

 <b>CPU300</b>	<input style="width: 100%;" type="button" value="Nuevo"/>
 <b>RouterDLink</b>	<input style="width: 100%;" type="button" value="Nuevo"/>
 <b>Slot10</b>	<input style="width: 100%;" type="button" value="Nuevo"/>

Plantilla:  Tipo:  Marca:

Bornera de Equipo:  CAPAO:

Tipo de Equipo:  Calcula Señales:

Código de Dibujo:  Marca:

Abreviación:  Modelo:

TAGI:  Gama:

ProtocoloUP:  CPUREq:

ProtocoloDW:  Propietario:

CAPDI:  Fecha:

CAPDO:  Descripción:

CAPAI:  Slot10:

**Datos Adicionales:**

Categoría	Campo	valorCampo	Descripción
huecos	derecha	pequeño	hueco petit
huecos	izq	grande	hueco grande
huecos	medio	mediano	hueco mediano

**Filiaciones:**

Img	Padre
	<b>RouterDLink</b>
	<b>CPU300</b>
	<b>Slot10</b>

W3C XHTML 1.0 | W3C CSS

### 6.2.2 Entorno de usuarios.

Como decíamos se han conseguido implementar todos los objetivos para los distintos usuarios. Se pasa a detallarlos más concretamente a continuación.

- **Administrador:**
  - Gestión de usuarios: Altas, bajas y modificación de usuarios en la aplicación.
  - Gestión del foro: Altas, bajas de usuarios en el foro. Moderar las secciones, categorías e hilos de discusiones.
  - Ver Equipos y plantillas: Ver equipos y plantillas validadas.
  - Comentar en foro: Comentar en los hilos de discusión.
  - Logout: Salir de la aplicación.
- **Desarrollador:**
  - Gestión de equipos: Altas, bajas y modificación de equipos en desarrollo.
  - Gestión de plantillas: Altas, bajas y modificación de plantillas en desarrollo.
  - Datos adicionales: Datos adicionales para los equipos en desarrollo.
  - Filiaciones: Filiaciones entre los equipos en desarrollo (hijos) y validados (padres).
  - Ver Equipos y plantillas: Ver equipos y plantillas validadas.
  - Comentar en foro: Comentar en los hilos de discusión.
  - Logout: Salir de la aplicación.
- **Validador:**
  - Validar Equipos: Dar el visto bueno a los equipos pendientes de validar.
  - Validar Plantillas: Dar el visto bueno a las plantillas pendientes de validar.
  - Ver Equipos y plantillas: Ver equipos y plantillas validadas.
  - Comentar en foro: Comentar en los hilos de discusión.
  - Logout: Salir de la aplicación.
- **Visitante:**
  - Ver Equipos y plantillas: Ver equipos y plantillas validadas.
  - Comentar en foro: Comentar en los hilos de discusión.
  - Logout: Salir de la aplicación.

### 6.2.3 Motor de aplicación.

La aplicación se ha desarrollado sobre distintos lenguajes de programación que otorgan la capacidad y flexibilidad oportunas.

En mayor parte se ha desarrollado en *PHP*, donde se encuentra el motor principal del aplicativo. En *PHP* se han desarrollado los formularios para gestionar equipos, plantillas, etc. El motor principal, que se encuentra en el fichero *insertar\_elem.php*, es el encargado de interactuar con la base de datos, controlar que no se dupliquen datos y en general, que las bases de datos sean consistentes.

Por otra parte se ha implementado en *JavaScript* el control de los campos

de los formularios para evitar campos vacíos en los casos que no puedan darse, así como tipo de contenido adecuado para prevenir ataques de *SQL injection*. Dentro del *JavaScript* encontramos también *Ajax* que da un aspecto dinámico a la web.

Para finalizar se ha utilizado CSS (Cascading Style Sheets) para dar el diseño físico de la aplicación.

Como parte interesante a destacar del código encontraríamos el encabezado de cada *página.php*. Este encabezado se encarga de controlar los tiempos de sesión además del tipo de usuario, con tal de tener la seguridad de que no se accede a zonas desautorizadas.

---

---

```
<?php session_start();
include("globals.php");

if($_SESSION['caduca']>time() && $_SESSION['rol']=="desa")
{
    $_SESSION['caduca']=time()+$expireTime; //renovar sesion

}else //si ha caducado
{?>
    <script type="text/javascript">
        alert("Tiempo de sesión agotado o no tiene permiso
              para acceder a esta página.");
        window.location=("../index.html";
    </script>
<?php
}
?>
```

---

---

Como se puede observar, se utilizan variables para controlar los tiempos de sesión. Estas variables vienen incluidas en el fichero *globals.php* donde se pueden modificar, para que afecten a todos las *paginas.php*.

También encontramos interesante destacar una función que es la encargada de cargar las *páginas.php* en los diferentes *divs* con tal de obtener un resultado dinámico.

---

---

```
function llamadasin(url, div)
{
    var pagina_requerida = false;

    if (window.XMLHttpRequest) //Comprobar si navegador nuevo
    {

        pagina_requerida = new XMLHttpRequest();
    }
}
```

---

---

```



---




---


else if (window.ActiveXObject) //Comprobar si navegador IE
{
    try
    {
        pagina_requerida=new
        ActiveXObject("Msxml2.XMLHTTP");
    }
    catch (e)
    {

        try // Version Obsoleta IE
        {
            pagina_requerida = new
            ActiveXObject("Microsoft.XMLHTTP");
        }
        catch (e)
        {
        }
    }
}
else
{
    return false;
}

pagina_requerida.onreadystatechange=function()
{
    pintapagina(pagina_requerida, div);
}

pagina_requerida.open('GET', url, true);
pagina_requerida.send(null);
}

function pintapagina(pagina_requerida,div)
{
    if (pagina_requerida.readyState == 4 &&
pagina_requerida.status==200 || window.location.href.indexOf("http") ===
1))
    {
        document.getElementById(div).innerHTML=pagina_requerida.responseText;
    }
}

```

---



---

#### 6.2.4 Capa de datos.

La aplicación se ha desarrollado con unas bases de datos en MySQL, según el diseño original previsto. Así pues toda la información queda almacenada en sus respectivas tablas.

Se hace destacable el uso de cookies para almacenar las páginas que

estamos visitando, con tal de que a la hora de insertar elementos en las bases de datos, o se utilice la opción del navegador de refrescar, sigamos donde estábamos.

Otra forma de manejar datos entre *divs* de las páginas es a través de la url. Para finalizar, se cree conveniente destacar que, se gestionan los usuarios y tiempos de sesión, con variables de sesión. Ya que esto otorga una seguridad extra que no proporcionan las cookies.

Por otra parte, no se cree destacable ninguna parte del código, pues son meras consultas a las bases de datos.

## 7. Pruebas.

### 7.1 Introducción.

Se han efectuado varias pruebas para comprobar la solidez y robustez de la aplicación. Como se puede ver a continuación se han efectuado pruebas de compatibilidad, de seguridad y de integración de servicio o funcionamiento.

### 7.2 Pruebas de compatibilidad.

Se ha probado el funcionamiento de la aplicación en los navegadores, I.Explorer, Mozilla Firefox y Google Chrome.

Se han probado para todos ellos todas las opciones de la aplicación, tales como el ciclo de vida de un equipo a manos del desarrollador y del validador. Se ha conseguido un resultado satisfactorio, por lo que el motor de la aplicación funciona correctamente.

Por el contrario se ha detectado un problema de compatibilidades con I.Explorer en el entorno gráfico, donde algunos marcos no obtienen el tamaño correcto.

Se le ha restado importancia al ser un problema de estética y que no afecta al correcto funcionamiento del aplicativo.

Cabe destacar que la aplicación valida los estándares de *World Wide Web Consortium (W3C, <http://validator.w3.org>)* tanto en HTML como en CSS.

### 7.3 Pruebas de seguridad.

Como se decía en capítulos anteriores, cada usuario tendrá un rol definido así como una interfaz propia. Para esto se ha tratado de acceder logeado o sin logear en categorías que no son propias. El resultado ha sido la expulsión de la aplicación.

Además se ha intentado acceder a la aplicación con usuarios o contraseñas incorrectas, con el mismo resultado, obviamente. Finalmente se ha probado a introducir una inyección de SQL en el login, que también ha sido bloqueada.

### 7.4 Pruebas de integración de servicio.

En el capítulo anterior 7.2, Pruebas de compatibilidad, se probó el funcionamiento de la aplicación, para todas sus opciones y en diferentes navegadores. Por esto se puede concluir que, el funcionamiento de la aplicación, ha sido correcta soportando todos los tests de pruebas realizados.

## 8. Conclusiones y resultados.

### 8.1 Consecución de objetivos.

Se habían propuesto los 11 objetivos siguientes:

- O.1 Seguimiento del proyecto en todo momento vía web.
- O.2 Gestión y control de permisos de los usuarios para acceder a la aplicación.
- O.3 Espacio propio de trabajo ligado a las cuentas de usuario.
- O.4 Estandarización en la creación de componentes.
- O.5 Reutilización de trabajo.
- O.6 Gestión y aprovechamiento del trabajo.
- O.7 Almacenamiento de la información de manera eficiente.
- O.8 Centralizar la información en una única base de datos.
- O.9 Crear un portal web agradable visualmente e intuitivo de usar.
- O.10 Acceso a la aplicación mediante web, desde cualquier ubicación.
- O.11 Facilitar la comunicación entre los usuarios mediante un foro.

De estos objetivos se puede afirmar que se han concluido exitosamente todos ellos.

### 8.2 Desviaciones observadas.

Se ha producido una desviación de casi dos meses respecto a la fecha finalización prevista: 28 de Abril 2010.

Se había pensado en trabajar 10 horas semanales en lo que no siempre ha sido posible. Además se ha visto una prolongación en la duración de algunas tareas, respecto al tiempo que se tenía previsto.

### 8.3 Líneas de ampliación.

El aplicativo es ampliable en diferentes aspectos según necesidades posteriores al uso. Se citan algunas posibles líneas de ampliación:

- Mejorar el aspecto estético de la web.
- Aplicar efectos visuales a la web.
- Mostrar en tablas o forma visual más agradable de los equipos validados.
- Gestionar los dibujos de equipos dinámicamente, según los elementos de la base de datos.
- Mejorar el gestor de usuarios.

### 8.4 Valoración personal de la experiencia.

Como la mayoría de las experiencias nuevas, he obtenido muchos puntos positivos de este proyecto. He aprendido a desarrollar un proyecto desde cero, sobretodo la memoria. He mejorado mis conocimientos de programación web. He aprendido a tener que rehacer o mejorar parte del trabajo pues no era exactamente lo que se precisaba.

En general he ganado experiencia en muchos aspectos psicológicos y ganado virtudes de todo en general. Por lo tanto solo me queda concluir que, pese a haberme encontrado con problemas puntuales, ha sido una experiencia muy positiva y gratificante.

## 9. Bibliografía.

Seguidamente se muestra la bibliografía, que ha servido de utilidad para concluir el proyecto. En particular, toda la información visitada ha sido online.

- [1] World wide web consortium (W3C) [Online]. <http://www.w3.org> (25 mayo, 2010).
- [2] PHP: Hypertext Preprocessor [Online]. <http://php.net/index.php> (25 mayo, 2010).
- [3] W3 Schools [Online]. <http://www.w3schools.com> (25 mayo 2010).
- [4] Libros Web [Online]. <http://www.librosweb.es/css/index.html> (25 mayo 2010).
- [5] Material de la asignatura de la EUI Sabadell: metodología y gestión de proyectos. Curso 2009-2010.
- [6] Foros del web [Online]. <http://www.forosdelweb.com> (25 mayo 2010).
- [7] PHP BB Hispano [Online]. <http://www.phpbb-es.com> (25 mayo 2010).

---

Firmado: ***Javier Enrique Ruiz  
Salvador.***