

# Aplicación Web para la ejecución y gestión de auditorías de calidad internas

## ANEXOS

Autor: Juan Carrey Labarta

Director: D. Ignacio Vilalta

Ponente: José Javier Merseguer Hernáiz

Empresa: Comex Grupo Ibérica



Centro Politécnico Superior

Universidad de Zaragoza



Departamento de Informática e Ingeniería de Sistemas  
Ingeniería Informática  
Curso 2009-2010, Mayo 2010

# ANEXO 1 - Planificación

## Índice

ANEXO 1 - PLANIFICACION .....	2
ÁMBITO DEL DOCUMENTO .....	3
HERRAMIENTAS Y TECNOLOGIAS .....	4
<i>Herramientas</i> .....	4
<i>Tecnologías</i> .....	4
FASES DEL PROYECTO .....	6
PLANIFICACION DEL PROYECTO .....	7
<i>Hitos</i> .....	8
REALIZACION REAL .....	8
<i>Calendario</i> .....	8
<i>Leyenda</i> .....	15

## Ámbito del documento

Este documento centra su contenido en la fase de planificación del proyecto posterior al análisis del proyecto. Una vez conocidos los requerimientos del sistema escogemos las herramientas y tecnologías adecuadas o sugeridas por la empresa en la que se enmarca el proyecto.

Al final de éste documento se recoge también el calendario final a fin de comparar con la planificación real las diferencias, que servirán como experiencia para futuros proyectos.

## **Herramientas y tecnologías**

### **Herramientas**

#### **Apache**

Apache Tomcat se ha utilizado como Servidor Web de uso extendido.

#### **OpenSVN y Tortoise**

Debido a que en la empresa en la que se enmarca el proyecto trabajan con un sistema de control de versiones se ha optado por aprender a utilizarlo y usarlo para este proyecto. Se utilizan OpenSVN como servicio gratuito para almacenar las versiones y Tortoise para la gestión local de los ficheros.

#### **Navegadores Web**

Para la visualización del resultado y realización de pruebas se han utilizado Internet Explorer y Mozilla Firefox.

#### **Microsoft Office 2003 y Acrobat Reader**

Para la generación de documentos se han utilizado estas herramientas de edición de texto.

#### **Eclipse**

Como entorno de desarrollo se ha utilizado Eclipse debido a las grandes ventajas que nos ofrece para el desarrollo de aplicaciones java.

#### **Edge Diagramer y GranttProject**

Hemos escogido estas dos aplicaciones para la generación de diagramas y calendarios debido a que ya las conocíamos anteriormente y nos parecieron unas herramientas sencillas de manejar, completas y muy útiles.

#### **Testlink**

Para la planificación y ejecución de pruebas hemos utilizado Testlink, se trata de una aplicación Web de gestión donde se introducen los planes de prueba con los pasos para que cualquier persona con acceso a la aplicación a la que se le hayan asignado la ejecución de diversos casos de prueba pueda ejecutarlos.

Los resultados se obtienen en unos informes generados por la aplicación, se desconocía esta herramienta que se está empezando a usar en la empresa.

#### **Microsoft SQL Server Studio**

Para la gestión de la base de datos en general hemos usado este software que ya estaba previamente instalado en el sistema y nos pareció bastante bueno para manejar SQL Server.

## **Tecnologías**

#### **Persistencia: JPA Object Relational Mapping**

Se ha decidido utilizar JPA como ORM debido a su gran potencia y sencillez, nos hemos intentado abstraer lo más posible del proveedor de persistencia (implementación), dado que de esta forma nos es muy como cambiar.

#### **Proveedor de persistencia: Hibernate**

Por su madurez y porque otros proveedores nos dieron problemas anteriormente (para la misma implementación), además será la tecnología usada en la empresa en un futuro.

#### **Base de datos: SQL Server 2005**

Como gestor de base de datos, la empresa decidió almacenar la información para las auditorías en su base de datos SQL Server 2005 para mantener en ésta todos los datos internos.

#### **Servicios Web: WSS4J y Axis 1.4**

Para el acceso a los Servicios Web existentes, hemos utilizado WSS4J por su fácil configuración a través de un fichero de despliegue, mientras que en otros la configuración iba metida en el código, además buscando información sobre como compatibilizar con WSE3.0 era de las pocas tecnologías para las que se dejaba clara la solución a uno de los problemas que hay en la interoperabilidad entre estos servicios Web.

#### **Spring Framework:**

Como tecnología Web para la creación de sistemas Web complejos que facilitan la estructuración en tres capas (Modelo – Vista – Controlador, MVC) así como las grandes ventajas que ofrece sobre otros frameworks similares como Struts, el cual se planteó inicialmente como opción debido a que iba a ser lo que se iba a utilizar en la empresa en proyectos futuros, pero se cambió a Spring porque ya que también cambió de opción la empresa, lo vimos como un proyecto más completo, sencillo y que mejoraba sustancialmente parte de los problemas que tiene Struts.

#### **Spring Security:**

Dentro de Spring, existe una herramienta que sirve para configurar permisos y cuentas con el fin de restringir el acceso a diversas vistas de la Web a determinados usuarios, es por ello que decidimos escoger esta opción como seguridad en la capa de vista, porque nos venía con el propio Framework , ya estaba implementado y además era bastante bueno.

#### **Junit**

Herramienta para la realización de tests de pruebas tanto unitarias como de integración del software.

#### **Java, HTML, JavaScript, Ajax, JSP, CSS**

Se han utilizado estos lenguajes de programación, en principio estipulados por la empresa para la realización del proyecto dado que es con lo que habitualmente se trabaja.

Java con lenguaje de programación para todo el control y acceso a datos.

JSP para las vistas, que mezclan código HTML con Java

CSS para el diseño

Ajax para añadirle usabilidad al sistema

JavaScript para añadirle usabilidad al sistema (Para hacer llamadas Ajax y mostrar calendarios)

HTML como código final generado por el servidor para el navegador Web.

## **Fases del proyecto**

Las diversas fases que se han establecido en la planificación del proyecto son las fases comunes a la mayoría de proyectos software.

### **Planificación**

En esta etapa se planifican los calendarios y se establece la gestión de configuraciones.

### **Análisis**

En esta fase se intentan captar todas las necesidades del cliente, generando un análisis de requisitos, se escogen las herramientas a utilizar y se establecen las tecnologías que se utilizarán

### **Diseño**

En el diseño se generan todos los documentos necesarios para plasmar la solución diseñada, lo más claramente posible, de forma que se generarán diagramas de clases, esquemas conceptuales, casos de uso, etc.

Se escogerán los métodos para realizar la implementación (Arquitectura del sistema, patrones de diseño, etc.), y se definirán cada uno de los nombres en el diccionario de datos.

Se generará además el modelo físico de la base de datos donde se almacenará la información

### **Implementación**

En esta fase se desarrollara todo el código fuente siguiendo las pautas del diseño.

### **Pruebas**

Este apartado consiste en encontrar todos los posibles errores del sistema que hayan podido quedar carentes, por ello se generará un plan de pruebas posterior al análisis o al menos antes de realizar las pruebas.

Así mismo se probara la usabilidad del sistema de forma que resulte amigable la interacción con el mismo.

### **Instalación**

En este proceso se tiene en cuenta la fase de instalación del aplicativo en el servidor, configuración del servidor, etc.

### **Documentación**

En esta fase se tiene en cuenta la documentación del proyecto, redacción de documentos y revisión de los mismos, esta fase del proyecto viene un poco mezclada con las demás ya que en muchas de las fases se generan documentos, en esta fase se revisan y se redactan correctamente todos ellos, a parte de generar la Memoria final.

### **Mantenimiento**

Esta fase del proyecto no viene contemplada en la planificación, dado que su extensión puede llegar a ser "indefinida", pero forma parte del proyecto y hay que tenerla en cuenta.

## Planificación del proyecto

El proyecto se enmarca en el desarrollo de un proyecto fin de carrera de unas 400 horas aproximadamente para una única persona, así pues se ha desarrollado un calendario planificando las distintas etapas del proyecto y marcando además los hitos que se esperan superar en cada etapa.

En negro se marcan los días festivos, en amarillo se marcan los distintos hitos del proyecto (H1, H2, H3, H4).

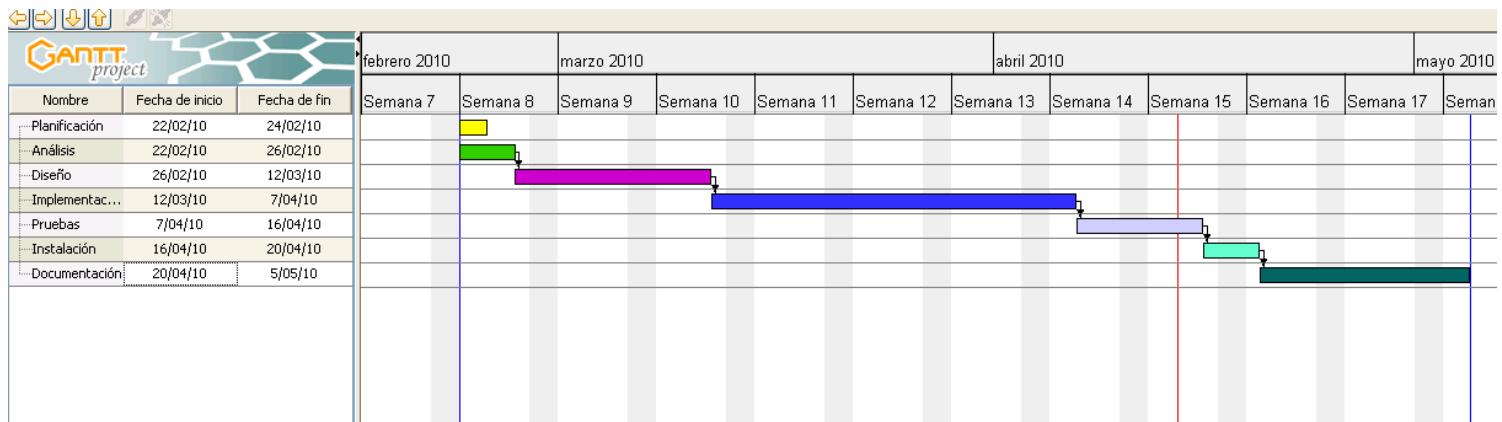


Figura 1 – Planificación inicial

54 días planificados (4 días festivos. 1 en diseño, 2 en implementación, 1 en documentación).

De lunes a jueves: 8 horas 30 minutos. Viernes 6 horas.

Horas totales con margen: 399 horas.

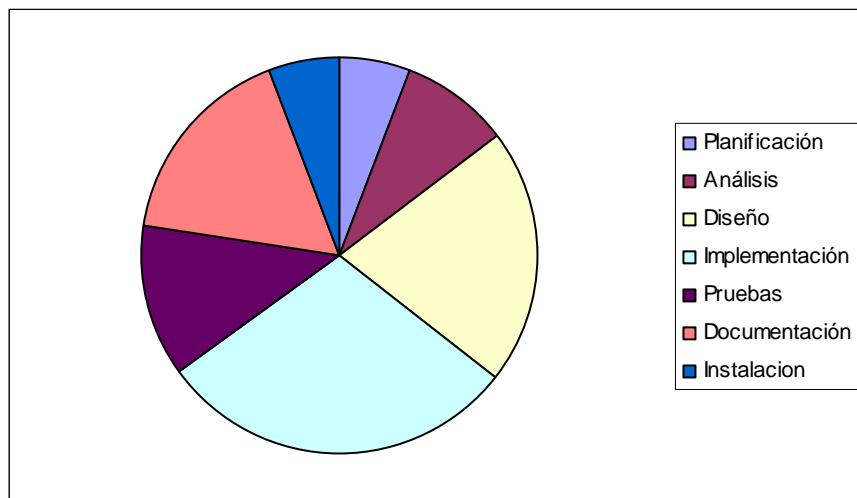


Figura 2 – Planificación inicial

No se ha planificado el periodo de aprendizaje de las tecnologías a usar, dado que en algunas de ellas el conocimiento previo es nulo y se desconocen en principio las horas necesarias, se intentará en la medida de lo posible compaginar el aprendizaje aplicando los ejemplos básicos a alguna parte de la implementación donde el ejemplo modificado sirva a su vez para el sistema final.

## Hitos

Los hitos planificados son los siguientes.

### Hito 1: 2 Marzo

Análisis completado, con los requisitos y la planificación realizada, gestión de configuraciones, etc.

### Hito 2: 8 Marzo

Diseño de la Base de datos, esquema conceptual, lógico y físico.

### Hito 3: 12 Marzo

Diseño general del sistema, tecnologías, esquemas del sistema etc.

### Hito 4: 19 Marzo

Diseño detallado del sistema, casos de uso, etc.

### Hito 5: 9 Abril

Sistema implementado con pruebas genéricas.

### Hito 6: 5 Mayo

Pruebas, documentación básica y documentación del PFC y el sistema implantado.

## Realización real

Se ha ido apuntando cada día del proyecto cual era la tarea realizada durante el día a modo de resumen, viéndose en este nuevo calendario los días perdidos por aprendizaje (amarillo), cambios de tecnologías(X5) y diversas problemáticas que han ido surgiendo a lo largo del proyecto (\*).

Axial pues el siguiente calendario muestra día a día cuales han sido las tareas realizadas, durante la implementación los problemas relacionados con la inexperiencia en el uso de las tecnologías no ha sido descrito dado que se pierde un tiempo cada cierto tiempo lo cual hace imposible llevar la contabilidad y a la vez buscar solución eficientemente a estos pequeños problemas que surgen.

## Calendario

Hemos desarrollado un calendario real en unas hojas Excel para poder actualizarlo con facilidad día a día, una aproximación resumida del calendario en un diagrama grantt sería el siguiente.

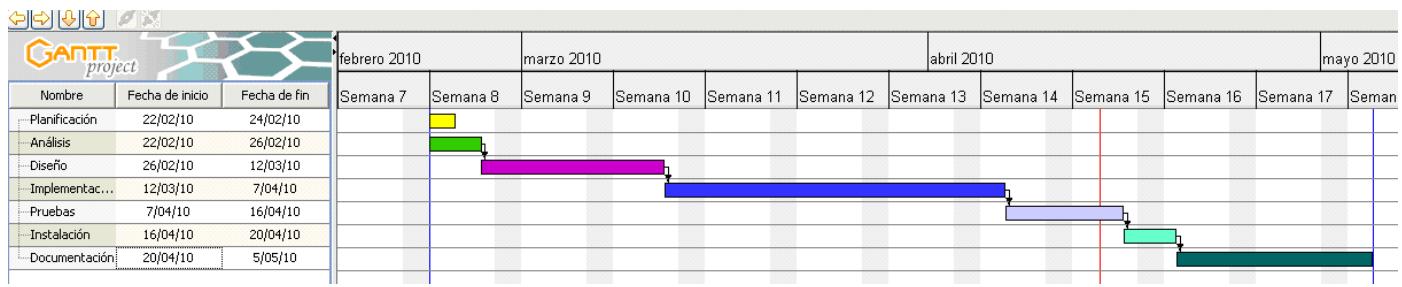


Figura 3 – Planificación inicial

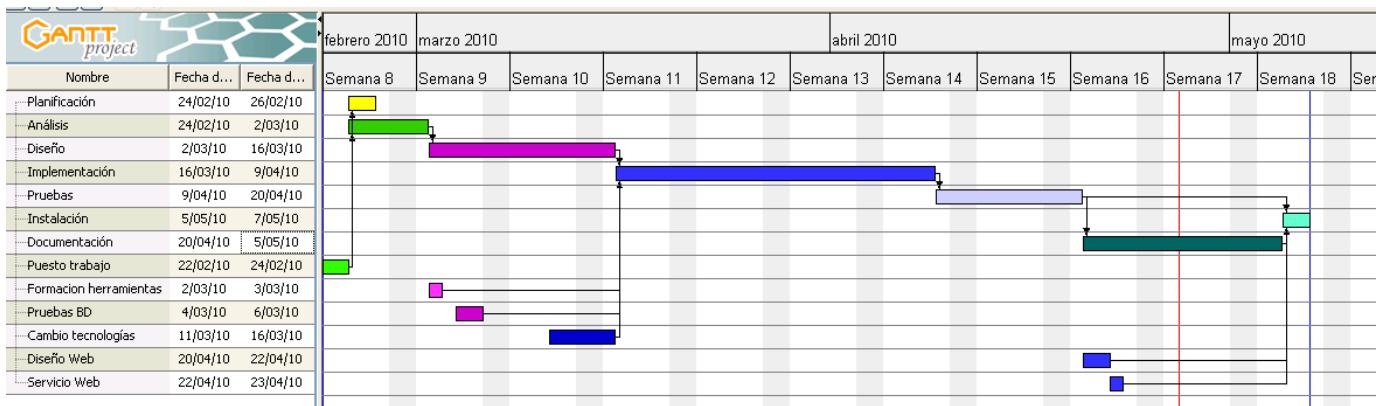


Figura 4 - Re-planificación

Arriba observamos la planificación inicial, y abajo la re-planificación debido a las tareas pequeñas que corresponden a tareas no planificadas que interfieren con la planificación.

Las más destacadas son las dos últimas incidencias, correspondientes a las esperas de trabajo ajeno, que tienen que ver con la implementación y tienen lugar en el apartado posterior a las pruebas e instalación del sistema, por lo que retrasa completamente la instalación del software y la ejecución de algunas pruebas.

La re-planificación consiste en redactar la documentación, justo después de las pruebas y retrasar la instalación del sistema a la semana 18.

Podemos observar también que el inicio del análisis y planificación se ve retrasado por la formación en el puesto de trabajo.

Observamos también que el cambio de tecnologías y la formación interfiere directamente con el diseño (solapándose), dejando menos tiempo para el diseño. Lo mismo ocurre con la implementación de aquellas partes del sistema afectados por las esperas, que se solapa con la documentación.

En la siguiente figura podemos comparar en horas la planificación real de lo realmente realizado, podemos observar que la formación se ha llevado gran parte del periodo planificado para el análisis, diseño y planificación, mientras que la implementación ha visto aumentado su trozo del pastel debido a las esperas, las pruebas se planificaron para más tiempo, sin embargo la implementación lleva consigo las pruebas unitarias y de integración.

En la siguiente imagen tenemos una comparación en horas de lo planificado con lo realmente realizado.

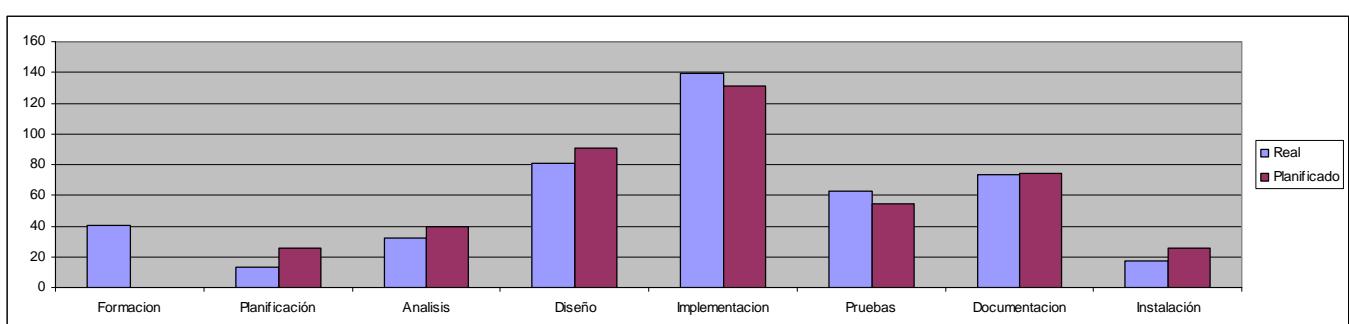


Figura 5 – Horas planificadas / reales

## Realización Real

Durante la realización del proyecto se fue apuntando cada día la tarea realizada y las horas ejecutadas en cada una de ellas, para mantener un control del tiempo requerido en cada tarea.

El total de horas realizadas es de 459.5 divididas de la siguiente forma:

<b>Formacion</b>	<b>40,5</b>
<b>Planificación</b>	<b>13</b>
<b>Analisis</b>	<b>32</b>
<b>Diseño</b>	<b>81</b>
<b>Implementacion</b>	<b>139,5</b>
<b>Pruebas</b>	<b>63</b>
<b>Documentacion</b>	<b>73,5</b>
<b>Instalación</b>	<b>17</b>

**Tabla 1 – Horas realizadas**

Cabe destacar las 40 horas y media de formación que no fueron planificados, este número es tan elevado debido al cambio repentino de tecnologías por las que se perdieron cerca de 25 horas.

Aproximadamente se ha llevado a cabo una jornada laboral de 40 horas semanales, que se han visto afectadas por días festivos y salidas a la universidad.

A continuación se detalla el calendario final:

## Primeras cuatro semanas

	ANÁLISIS							DISEÑO																			
	Febrero							Marzo																			
	22	23	24	25	26	27	28	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
X0, IO	A0	P0, P1	A1	A1				A1																			
X1								D0	D1	D1,D2	X2				J0	J2						IMPL					
										I1					I2	X3	X4	D3	D4			X5	D4	D2	D2	D2	
Formacion	4	1								0,5	8,5				1,5	1,5	2,5					4					
Planificación	4,5		8,5																								
Analisis		7,5		8,5	6			7,5	2,5																		
Diseño								1	6	8					7	7	6	8,5	6				8,5	8,5	8,5	6	
Implementacion																						2					
Pruebas																											
Documentacion																											
Instalación																											

Quinta, sexta y séptima semana.

	IMPLEMENTACION																					
Marzo														Abril								
	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	
			X5	X5	IMPL	IMPL	IMPL			IMPL	IMPL	IMPL					IMPL	IMPL	IMPL	IMPL	IMPL	
Formación			8,5	8,5																		
Planificación																						
análisis																						
Diseño																						
Implementación					7,5	8,5	6			7,5	8,5	8,5						8,5	8,5	8,5	7,5	6
Pruebas						1				1											1	
Documentación																						
Instalación																						
	0	0	8,5	8,5	8,5	8,5	6	0	0	8,5	8,5	8,5	0	0	0	0	8,5	8,5	8,5	8,5	6	

Tabla 2– Calendario real parte 2

Octava, novena y décima semana

	PRUEBAS				INSTALACIÓN				DOCUMENTACIÓN												
<b>Abril</b>																					
	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
		IMPL	IMPL	IMPL	IMPL					PR	PR		PR	PR	PR	PR	PR				
																		DOC	IMPL	IMPL	IMPL
			E*2	E*2	E*2 E*3	E*2 E*3				E*2 E*3			Acomodación diseño web		E*3						
Formación																					
Planificación																					
análisis																					
Diseño																					
Implementación		7,5	8,5	8,5	2													8,5	8,5	8,5	
Pruebas		1			6,5	6				8,5	8,5	8,5	8,5	8,5	6		6,5				
Documentación																	2				6
Instalación																					
	0	0	8,5	8,5	8,5	8,5	6	0	0	8,5	8,5	8,5	8,5	8,5	6	0	0	8,5	8,5	8,5	6

Tabla 3 – Calendario real parte 3

Onceaba y duodécima semana

DOCUMENTACIÓN			SIN PLANIFICACIÓN													
Mayo			1	2	3	4	5	6	7	8	9	10	11	12	13	14
				DOC E*3	DOC E*3	DOC E*3	DOC E*3	DOC E*3			DOC E*3	DOC	DOC	INST	INST	
Formación																
Planificación																
análisis																
Diseño																
Implementación																
Pruebas			8,5	8,5	8,5	8,5	8,5	6		8,5	8,5	8,5				
Documentación													8,5	8,5		
	0	0	8,5	8,5	8,5	8,5	8,5	6	0	0	8,5	8,5	8,5			

## Leyenda

Cada celda corresponde a las horas, salvo las cabeceras que corresponden a la planificación la de arriba del todo, mes actual la segunda fila y el día del mes la tercera.

La cuarta fila que contiene diferentes siglas X0,X1...,A0,A1... etc. corresponden a símbolos definidos en la leyenda y que corresponden a distintas tareas X=Formación, I=Instalación, P=Planificación, A=Análisis, D=Diseño, J=Documentación, IMP=Implementación, P=Pruebas, etc.

Aquí se definen las tareas realizadas, a que sección corresponden y una breve descripción de la misma

Tarea	Nombre	Descripción
X0	<b>Formación</b>	Formación, estudio de tecnologías
X1	<b>Formación</b>	Formación en el puesto de trabajo, orientación, estudio de tecnologías.
X2	<b>Formación</b>	Pruebas con las distintas herramientas
X3	<b>Formación</b>	Pruebas de acceso a BD, pruebas de consultas, pruebas del esquema
X4	<b>Formación</b>	Pruebas de struts y Ajax
X5	<b>Formación</b>	Estudio de Springs, cambio de tecnologías
A0	<b>Análisis</b>	Recepción de requisitos, análisis
A1	<b>Análisis</b>	Análisis de los requisitos.
P0	<b>Planificación</b>	Planificación del proyecto
P1	<b>Planificación</b>	Gestión de configuraciones.
D0	<b>Diseño</b>	Refinamiento de requisitos, Diseño Conceptual, diseño de Base de datos.
D1	<b>Diseño</b>	Casos de uso, métodos
D2	<b>Diseño</b>	Diseño detallado casos de uso y métodos
D3	<b>Diseño</b>	Diseño de interfaz
D4	<b>Diseño</b>	Diagrama de navegación
I0	<b>Instalación</b>	Instalación del entorno de desarrollo, librerías, etc.
I1	<b>Instalación</b>	Instalación de Sqlserver
I2	<b>Instalación</b>	Implantación provisional de BD en servidor local
J0	<b>Informes</b>	Documentación de informes de análisis y requisitos
E*1	<b>Salidas</b>	Salidas a la universidad.
E*2	<b>Esperas</b>	Esperas al diseño del apartado Web para su incorporación al sistema.
E*3	<b>Esperas</b>	Esperas a la implementación del servicio Web.

Tabla 4 – Leyenda del calendario

# ANEXO 2 – Gestión de configuraciones

## Índice

ANEXO 2 – GESTION DE CONFIGURACIONES .....	16
ALCANCE DE ESTE DOCUMENTO .....	17
DESCRIPCION GENERAL .....	17
DISTRIBUCION .....	18
<i>Ficheros fuente</i> .....	18
<i>Contenido web</i> .....	19
<i>Test</i> .....	20
<i>Software utilizado</i> .....	20
<i>Backups</i> .....	20
<i>Documentación</i> .....	20

## **Alcance de este documento**

El objetivo de este documento trata de establecer una forma estándar de localizar los diversos documentos y ficheros fuente del proyecto, estableciendo una jerarquía de documentos.

Para ello se describirán cada uno de los módulos que se han separado indicando su localización.

## **Descripción general**

La estructura que se ha seguido a la hora de distribuir los archivos es una muy similar a la estructuración general de los proyectos de la empresa, esta gestión de configuraciones abarca tanto los archivos fuente, como el software que se utilice e instale, como los diversos documentos y archivos de la documentación.

## Distribución

### Ficheros fuente

Los ficheros fuente son todos aquellos ficheros y archivos que se requieren para la generación de la aplicación, que contengan código o sean necesarios para la compilación del producto.

Estructuración general:

La carpeta **raíz** del proyecto tendrá el nombre de la aplicación (gauditint), dentro de la raíz diferenciaremos las subcarpetas según el tipo de código fuente almacenado.

Esta carpeta raíz será la que se utilice para el control de versiones e ira contenida en:  
**./gauditint/proyecto/** por lo que el directorio **raíz** equivale a:

**./gauditint/proyecto/gauditint**

#### Componentes (Clases java)

Se tratan de todas aquellas clases que sirven de apoyo o utilidad general para la aplicación, como pueden ser las excepciones, los formateadores (de emails, de fechas), configuraciones, etc.

Se encontraran en el directorio: **raíz/components/gauditint/src/java/**

#### DAO (Clases java)

Se tratan de todas aquellas clases que sirven para el acceso a datos, ya sean los transfer object (beans) como los dao propiamente dichos, así mismo estarán aquí todas aquellas clases que interfieran de alguna forma con la base de datos directamente.

Se encontraran en el directorio: **raíz/apps/gauditint/gauditint-dao/src/java**

#### Web User (Clases java)

Se tratan de todos aquellos componentes que tienen que ver con la capa de lógica de control a nivel de **usuario**, son aquellas clases necesarias para Spring, como los controladores, formularios, servlets, validadores, etc.

Se encontraran en el directorio: **raíz/apps/gauditint/gauditint-webuser/src/java**

#### Web Admin (Clases java)

Se tratan de todos aquellos componentes que tienen que ver con la capa de lógica de control a nivel de **administración**, son aquellas clases necesarias para Spring, como los controladores, formularios, servlets, validadores, etc.

Se encontraran en el directorio: **raíz/apps/gauditint/gauditint-admin/src/java**

## Contenido Web

El contenido Web es una carpeta por defecto en un proyecto Web, es aquella carpeta donde se emplazan todos los ficheros (que después equivaldría al directorio raíz de la aplicación en el servidor). Mientras que el código fuente se compila y se deposita en un directorio localizado en el servidor, el resto de ficheros se localizan en el sitio donde nosotros deseemos.

Si que hay una estructura medio definida, donde encontramos las carpetas META-INF y WEB-INF que quedarán ocultas al usuario, por ello disponemos en esta carpeta todo lo relacionado con la configuración y páginas jsp (que incorporan código fuente) y dejamos fuera de estas carpetas todo lo que debe ser accesible por los usuarios.

Para el proyecto en local, se almacenará en la carpeta /raíz/apps/gauditint/, de forma que el directorio raíz del servidor será el equivalente a **/raíz/apps/gauditint/**

En esta carpeta encontraremos al menos el index.jsp, que simplemente redirecciona a una dirección lógica conocida, que será el inicio.

### Web utilities

Se trata de la carpeta donde se encontraran ficheros públicos necesarios para la generación de las páginas web. Se llamará **web**

**CSS** : Ficheros con hojas de estilo (.css) importados por las distintas páginas, se encontraran dentro de la carpeta: **/raíz/apps/gauditint/web/css**

**JavaScript**: Ficheros con archivos javascript (.js), que deben ser públicos dado que el código se ejecuta en el cliente. Se almacenaran en la carpeta: **/raíz/apps/gauditint/web/js**

**Imágenes**: Ficheros de imágenes (.jpg, .gif, .png, etc.), que también deben ser públicos. Se almacenaran en la carpeta: **/raíz/apps/gauditint/web/img**

### Meta información

Meta información de la página web, en esta carpeta por defecto se localiza el manifest.mf, y además como nosotros utilizamos JPA en el proyecto, el archivo persistence.xml debe estar localizado aquí, en el viene definido el proveedor de persistencia y las clases que se persisten, además de la información necesaria para la conexión a la base de datos (usuario, contraseña, etc), por lo que es obligatorio que este oculto al usuario.

La carpeta META-INF estará en: **/raíz/apps/gauditint/META-INF**

### Información de la web

La carpeta WEB-INF contiene todos los archivos ocultos que no son meta-information de la página, así pues aquí encontraremos los archivos XML para la descripción de la web (web.xml), así como los dos ficheros necesarios para la configuración de spring (gauditint-global-servlet.xml y gauditint-admin-security.xml).

**JSP**: En esta carpeta encontraremos todos los ficheros de Java Server Pages (.jsp) que corresponden a cada una de las páginas web. Se encontraran en la carpeta: **/raíz/apps/gauditint/WEB-INF/jsp**

**Librerías**: En esta carpeta se almacenarán todas las librerías necesarias para el proyecto, se guardarán en la carpeta: **/raíz/apps/gauditint/WEB-INF/lib**

**Tags**: En esta carpeta se almacenaran los ficheros de descripción de los tags, se almacenaran en **/raíz/apps/gauditint/WEB-INF/tld/**

**Ficheros configuración:** Los ficheros de configuración del servidor y propios, como por ejemplo los correspondientes a los mensajes de Spring (messages.properties) , los necesarios para la configuración de los servicios web utilizados ( deploy.wssd) , los de configuración del servidor de correo y parámetros de la aplicación (web.config.properties), etc. Estos ficheros se almacenarán en la carpeta **/raíz/apps/gauditint/WEB-INF/classes/**

## Test

Los test se crearán en un proyecto distinto, que dependa del proyecto de la aplicación principal, tendrá un paquete **test** con las clases que correspondan a cada test. Se encontraran en la carpeta: **./auditint/proyecto/Test/**

Los ficheros fuente de los test iran en la carpeta **/Test/src**

## Software utilizado

El software utilizado irá instalado en: **./software**, mientras que el software descargado se localizará en **./descargas**, por si hubiera que re-instalar alguno de los componentes.

Si algún software ya estaba instalado en la maquina en un directorio diferente, se creara un acceso directo a él desde la carpeta de software instalado, no así en la de descargas porque posiblemente no este localizable.

## Backups

Se realizaran copias de seguridad de los archivos generados (documentación, etc) que se localizarán en **./auditint/backups**

Además de estar localizadas en el servidor remoto de subversión.

## Documentación

La documentación la dividiremos en dos apartados, la documentación redactada y la no redactada (generada durante la realización del proyecto fuera de la fase de documentación, ficheros de esquemas, hojas Excel, etc).

Todos los ficheros de documentación iran en la carpeta **./auditint/Documentacion/**

### Documentación redactada

tendrá una nomenclatura común, todos los ficheros serán .doc o .pdf, y su nombre empezará por "CGI-INT-2010-PFC Juan –", seguidos del nombre del fichero ó título.

La memoria final ira en la subcarpeta Memoria, mientras que los diferentes archivos necesarios para el proyecto fin de carrera, tales como la propuesta iran en la subcarpeta "PFC"

### Documentación no redactada

Estos ficheros iran en una carpeta correspondiente al nombre de la fase en la que se realizaron los archivos (Análisis, Diseño, etc).

El calendario que va recogiendo día a día lo que se va haciendo ira en la subcarpeta Realización.

# ANEXO 3 – Análisis

## Índice

ANEXO 3 – ANALISIS .....	21
ÁMBITO DEL DOCUMENTO .....	22
DESCRIPCION Y OBJETIVOS .....	23
SISTEMA ACTUAL .....	24
IDENTIFICACION Y DEFINICION DE REQUISITOS .....	25
Ámbito y alcance del proyecto.....	25
Catálogo de requisitos del sistema.....	26

## **Ámbito del documento**

El ámbito de este documento es el análisis y diseño del proyecto, en el se encuentran los primeros apartados donde se describen los problemas del sistema actual, se describe brevemente el proceso y los objetivos a alcanzar.

En el primer apartado se describe el análisis obtención de requisitos etc. En el segundo apartado se describe el diseño de la solución, casos de uso, arquitectura y diseño de la base de datos.

## Descripción y objetivos

El objetivo de este documento es la obtención de una especificación detallada del nuevo sistema de Gestión de auditorías internas, de forma que éste satisfaga todas y cada una de las necesidades funcionales de los usuarios y administradores de la aplicación.

En el apartado de este documento *Situación Actual* se realiza un estudio sobre cómo trata el sistema actual la gestión de las auditorías, para de esta manera poder detectar problemas y necesidades que el nuevo sistema tratará de solventar tanto a nivel de funcionalidad como de agilidad en el manejo de la herramienta.

La definición del sistema junto con la especificación de requisitos realizada en este documento permitirá describir con precisión el nuevo sistema de información, entrando en detalle en las tareas propias del Análisis Funcional:

Modelo de procesos del sistema.

Elaboración del modelo de datos, tanto conceptual como lógico.

Interfaces de usuario.

La participación activa de los usuarios durante las diferentes sesiones de trabajo para la completa definición del sistema habrá resultado decisiva e imprescindible para la actual fase de análisis del nuevo sistema, ya que dicha participación constituye una garantía de que los requisitos, datos, procesos e interfaces identificados han sido comprendidos e incorporados al sistema.

## Sistema actual

**Comex Grupo Ibérica Integración** realiza actualmente el proceso de gestión auditorías de forma manual, mediante checklist representados en hojas Excel.

Existe una checklist por tipo de auditoría y por tipo de proceso a auditar. Este checklist esta formado por diversas preguntas con tres posibles respuestas, si, no y no aplica, junto con un campo observaciones por si es necesario dar alguna explicación sobre los motivos de por que no se ha hecho algo.

Los resultados de las auditorías se reflejan posteriormente en el Informe General de Auditorías, replicando de esta forma la información. Además de éstos se calculan semestralmente una serie de métricas, que se realizan de forma manual, llevando consigo el problema de tener que revisar una a una las hojas Excel de auditorías anteriores, situándose cada una en un lugar diferente dependiendo del proyecto, lo cual hace casi imposible realizar una comparación entre proyectos y muy complicado el cálculo de diversas métricas.

Cada proyecto además tiene otra hoja Excel con el seguimiento de las no conformidades, que recogen las respuestas negativas de la auditoría, por lo que resulta complicado realizar el seguimiento dado que todo esta en hojas Excel independientes.

Otro de los problemas existentes con el sistema manual actual es que el responsable de calidad debe estar atento a las fechas planificadas para las auditorías y ejecutarlas conforme a la planificación.

Esta aplicación pretende solucionar toda esta problemática dado que toda la información resultante de las auditorías se encuentra centralizada en una base de datos única, por lo que la explotación de datos será mucho más sencilla. Además podrá enviar avisos a las personas interesadas a modo de recordatorio para cumplir con las planificaciones establecidas.

## Identificación y definición de requisitos

En este apartado se lleva a cabo la definición, análisis y validación de los requisitos a partir de la información facilitada por los usuarios participantes. Se obtiene un catálogo detallado de los requisitos, a partir del cual se pueda comprobar que los productos generados en las actividades de modelización se ajustan a los requisitos de usuario.

## Ámbito y alcance del proyecto

### Definición del proyecto

El Sistema Informático de Gestión de auditorías internas de calidad permitirá toda la funcionalidad básica del proceso manual actual, pudiendo gestionar las auditorías y seguimientos así como la generación de métricas e informes de resultados.

Las herramientas básicas que el sistema ofrecerá serán las siguientes:

Gestión de auditores y auditados:

Se gestionara la información del personal de la empresa que realiza auditorías o es auditada en un proyecto, bien sea como jefe del mismo o de otra forma.

Gestión de los procesos a auditar.

Los procesos de un proyecto pueden cambiar, por lo que se permitirá su administración desde la parte privada de la aplicación, los procesos son fases del proyecto, donde cada proceso tiene asignadas una serie de preguntas (checklist del sistema actual).

Gestión de las preguntas a realizar en cada auditoría.

Las preguntas a realizar en cada auditoría en un determinado proceso cambian, por lo que también se permitirá su administración desde la parte privada de la aplicación. Una pregunta esta asociada a un único proceso, pero puede estar en distintos tipos de auditoría.

Aviso recordatorio de la auditoría.

Unos días (configurables por el administrador) antes de la fecha planificada para la ejecución de una auditoría se avisara a los auditores para recordarles el evento.

Solicitud automatizada de la planificación de la auditoría.

Al dar de alta una o varias auditorías para un proyecto, se avisara al Jefe de proyecto mediante email para que introduzca las fechas para la planificación de cada una de las auditorías.

En caso de no haber respuesta en un plazo de días (configurables por el administrador) se reenviará dicho correo avisando de nuevo.

Tras un número (configurable por el administrador) de re-envíos, se enviará un aviso al auditor para ponerlo en su conocimiento.

Seguimiento de no conformidades.

Generación de informes.

Generación de métricas.

Existirán dos partes claramente diferenciadas

Una parte pública desde donde se ejecutan las auditorías, se consultan informes y se realiza el seguimiento.

Una parte privada desde donde se administra la aplicación completa. Esta sección será accesible únicamente mediante login/password por parte de uno o varios administradores.

## Catálogo de requisitos del sistema

El nuevo sistema a desarrollar tiene como objetivo cumplir una serie de requisitos funcionales y no funcionales que se resumen a continuación:

**F - Funcional**

**NF - No funcional**

### Requisitos funcionales de la parte pública

1.- **F** El nuevo aplicativo resultará claro y sencillo, y su operativa será lo más amigable, intuitiva y atractiva posible.

2.- **F** El sistema tendrá una gestión de avisos por email al personal correspondiente, ya sea auditor o auditado según la siguiente lógica:

2.1 Al dar de alta una nueva auditoría, se enviará un correo a los auditados. El correo contendrá un link a una página que permita modificar la fecha de la auditoría correspondiente.

2.2 Al modificarse las fechas de auditación de una auditoría se enviará un correo a los auditores. El correo contendrá información de las fechas de la auditoría, para recordárselas al auditor.

2.3 Unos días antes (también configurables) de que tenga lugar la fecha de auditación de una auditoría, se enviará un email tanto a los auditores como auditados para la realización de la misma.

2.4 Si no se contesta al aviso del punto 2.1 (pasan un número determinado y configurable de días sin obtener respuesta), se reenviará el aviso para realizar la planificación de nuevo, cuando se hayan enviado un número de avisos, también configurable, se enviará un aviso al auditor para su conocimiento.

2.5 Al terminar una auditoría (completarla) el auditor recibirá un email (o bien desde la aplicación) para acceder al seguimiento de no conformidades de la auditoría ó proyecto.

2.6.- Pasados un número de días configurable desde que se completa el seguimiento el auditor recibirá un email para recordarle de comprobar como se está llevando a cabo dicho seguimiento.

3.- **F** En la parte pública se permitirá la creación, modificación y ejecución de auditorías, realizadas o bien sobre un proyecto (No especiales) o bien que no requieran proyecto (especiales).

Por defecto en la creación de auditorías vendrán marcadas todas las no especiales.

En la ejecución de la auditoría se permitirá la navegación mediante pestañas, dónde cada pestaña corresponderá a un proceso distinto.

Se podrán buscar proyectos y personal a través del Servicio Web disponible y mediante el código de recurso.

4.- **F** Desde la parte pública se podrán consultar diversos informes

Resumen de resultados de todas las auditorías: contendrá información de cada auditoría indicando el número de no conformidades de cada tipo, el riesgo previo y posterior a la auditoría, etc. La información que aparece en la hoja Excel del resumen de resultados del método manual.

Resumen de resultados de las auditorías por tipo de proyecto.

Resumen de resultados de las auditorías por jefe de proyecto.

Resumen de una auditoría, se trata de la información de las no conformidades asociadas a una auditoría, con la descripción el tipo y la causa de la no conformidad, así mismo se podrá observar el estado del proyecto actual y el estado de esta auditoría según el número de no conformidades de cada tipo.

El estado/Riesgo de un proyecto/auditoría se calcula mediante una fórmula que relaciona el número de no conformidades de un determinado tipo con el estado a través de un número que será el máximo número de no conformidades de un tipo para el cual se puede estar en ese estado. El estado resultante será el PEOR de todos los estados que se cumplan, siendo el peor estado aquel que requiera MENOR número de no conformidades para un tipo de no conformidad cualquiera.

Top 10 de preguntas con resultado negativo y sus causas, aquí se mostrarán las diez preguntas que más veces se han contestado negativamente, en número de respuestas negativas y en porcentaje de respuestas negativas sobre el total de respuestas, es decir dos rankings distintos.

5.- **F** Desde la parte pública se podrá acceder a diversas métricas calculadas automáticamente.

Media de no conformidades de cada tipo por Auditoría entre dos fechas que por defecto será el último año, vendrán agrupadas por un número determinado de meses, semestres, trimestres, bimensual,

mensual, etc. además existirá la posibilidad de agruparlas también por el tipo de proyecto ó proceso de la auditoría.

Número de auditorías por mes entre fechas (por defecto por año), de la misma forma podrán agruparse los meses en bimensual, trimestral, semestral, etc.

Número de auditorías planificadas y no realizadas entre dos fechas (por defecto por año)

6.- **F** Desde la parte pública se podrán realizar los Seguimientos de no conformidades (Auditor), donde se contemplaran todas las no conformidades de las auditorías de un proyecto ó únicamente de la auditoría seleccionada, con la información relevante a cada no conformidad así como la posibilidad de añadir acciones correctivas / cierre, modificar el estado de la revisión, seleccionar al personal responsable, etc.

7.- **F** Desde la parte pública se podrán modificar las fechas de la auditoría lo cual corresponde a su planificación.

8.- **F** Antes de la ejecución de la auditoría se mostrará una agenda de la misma, mostrando la información relevante a la auditoría.

### **Requisitos funcionales del sistema en general**

1.- **F** La obtención de las preguntas para una auditoría se realiza de la siguiente forma:

Cada pregunta esta asociada a un único proceso, y a varios tipos de auditoría, para una auditoría se obtienen todas las preguntas que están asociadas al tipo de auditoría de la auditoría seleccionada, y se ordenan por proceso.

2.- **F** Las respuestas de una auditoría pueden ser: Si, No y No/Aplica, y podrán tener observaciones.

3.- **F** Una auditoría puede tener los siguientes estados

Creación: Una auditoría recién creada, sin fecha de planificación.

Planificada: Una auditoría que tiene fecha de planificación y aun no ha sido ejecutada.

En curso: Una auditoría que ha comenzado a ejecutarse y se ha dejado a mitad, por lo que aun no esta completada.

Completada: Una auditoría que se ha ejecutado correctamente, y se han definido los tipos de no conformidad de sus no conformidades.

Invalidada: Aquella que ha sido invalidada manualmente.

Caducada: Es una auditoría que aun no se ha realizado y la fecha para la que se planificó ya ha pasado.

4.- **F** Una revisión de no conformidad puede tener uno de los siguientes estados

Pendiente: Cuando aun no se han definido sus acciones correctivas y el auditor no la ha revisado.

En curso: Cuando el auditor ha planificado la misma y los responsables están llevando a cabo las medidas correspondientes.

Invalida: Cuando no es necesaria su corrección por diversas causas.

Cerrada: Cuando se ha completado su corrección.

5.- **F** Las preguntas serán históricas, es decir al modificarse una pregunta, se dará de baja (lógica) la pregunta y se dará de alta una nueva, así las auditorías anteriores a esta modificación mostrarán la pregunta dada de baja mientras que las nuevas auditorías usarán la pregunta nueva.

6.- **F** Del mismo modo que las preguntas los procesos serán históricos, solo que su modificación no afectará más que al nombre por lo que no requerirá baja, sin embargo si se podrán dar de baja los procesos de forma manual.

### **Requisitos de la parte privada**

1.- **NF** Se accederá a cualquier parte de la administración mediante login/password, manteniendo en lo posible la seguridad.

2.- **F** Se podrá gestionar al personal, buscando por código de recurso a través del Servicio Web.

3.- **F** Se permitirá gestionar los procesos a auditar, pueden sufrir cambios.

4.- **F** Se facilitará la gestión de las preguntas existentes, pudiendo cambiar ó eliminar preguntas.

5.- **F** Todos los parámetros configurables numéricos serán configurables desde la administración, a la que se accederá mediante un login/password almacenado en la propia base de datos.

6.- **F** Se podrán gestionar los tipos de auditoría existente permitiendo validar o invalidar los existentes y configurando si es especial o no.

## ANEXO 4 – Diseño de Base de Datos

### Índice

<b>ANEXO 4 – DISEÑO DE BASE DE DATOS .....</b>	<b>29</b>
ÁMBITO DEL DOCUMENTO .....	30
MODELO DE DATOS .....	31
<i>Modelo conceptual de los datos</i> .....	31
<i>Modelo lógico de los datos</i> .....	37
<i>Normalización</i> .....	37
<i>Modelo físico de los datos (Implementación de la Base de Datos)</i> .....	43
DICCCIONARIO DE DATOS .....	45

## **Ámbito del documento**

En este documento se describen los pasos realizados para la generación del modelo de datos del sistema, así como explicaciones de los apartados más interesantes. Se describen los tres pasos fundamentales, diseño conceptual, lógico y físico. Además se describe en el diccionario de datos el significado de cada uno de los nombres utilizados para los atributos.

## **Modelo de datos**

Dentro de este apartado se identifican todas las necesidades de información de cada uno de los procesos que conforman el sistema de información, con el fin de obtener un modelo de datos que contemple todas las entidades, relaciones y atributos necesarios para dar respuesta a dichas necesidades.

### **Modelo conceptual de los datos**

Hemos realizado un esquema conceptual entidad-relación de los datos, durante el análisis hubo muchos cambios y varias versiones del esquema.

Así mismo hay datos no relacionados con el esquema pero que requieren ser guardados, son por ejemplo los diversos parámetros configurables de la aplicación y las cuentas de administración.

Para las relaciones 1:N , la punta de la flecha indica la entidad a la que se arrastra la clave primaria como foránea, es decir Entidad A ----- 1 : N ---> Entidad B, significa que una instancia de la entidad A puede tener relación con 0 o más instancias de la entidad B.

Los atributos han sido omitidos para mayor claridad.

La versión final del esquema conceptual es el siguiente

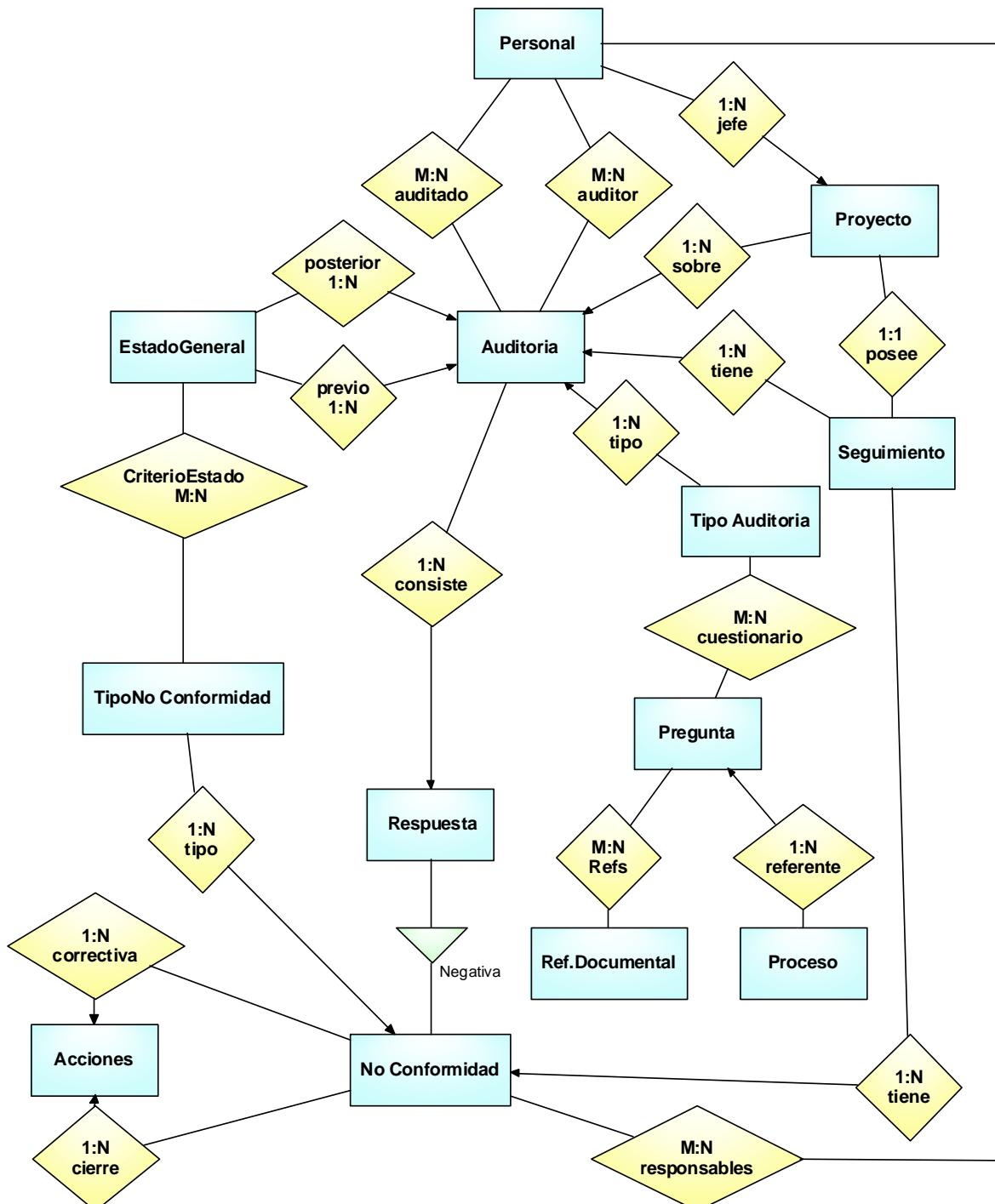


Figura 6 – Diseño de la base de datos relacional

## Auditoría

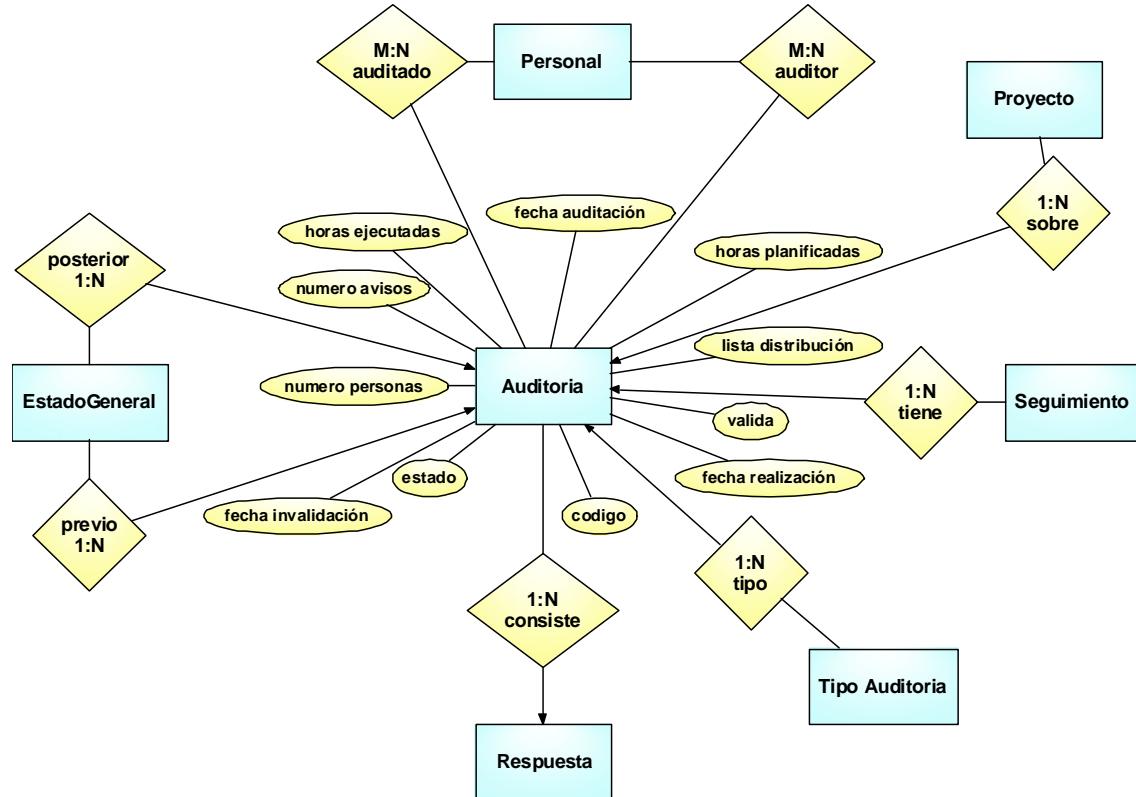


Figura 7 – Esquema conceptual Auditoría

La auditoría es la entidad más importante del sistema, dado que el objetivo final del sistema es poder gestionar auditorías de calidad, es por ello que queda en el centro del esquema entidad-relación y forma parte de muchas relaciones con el resto de entidades, así mismo posee una cantidad importante de atributos.

### Atributos

Horas ejecutadas: Son las horas que ha costado ejecutar la auditoría

Horas planificadas: Son las horas que se han planificado para la ejecución de la auditoría.

Fecha auditación: Fecha planificada por el auditado(o jefe de proyecto) para la realización de la auditoría según su calendario.

Fecha realización: Fecha en la que se ha realizado (ejecutado) la auditoría.

Fecha invalidación: Fecha en la que por ultima vez se dio de baja la auditoría.

Lista de distribución: Se trata de un listado de emails donde enviar un aviso para la visualización de la auditoría cuando esta concluya.

Valida: Validez de la auditoría

Código: código auto-generado en forma de texto que identifica a la auditoría

Estado: Estado de la auditoría (creación, planificada, completada, etc.)

Número de personas: Número de personas que ejecutaron la auditoría.

Número avisos: Número de avisos que se han realizado al auditado para que planifique la auditoría (Hay un máximo).

### Relaciones

Las relaciones de la entidad Auditoría se explican aquí:

Proyecto: Una auditoría se realiza sobre un proyecto generalmente, aunque si la auditoría es especial, como hemos dicho antes, no requiere proyecto.

Seguimiento: Una auditoría tiene un seguimiento propio, dado que algunas de las auditorías no tienen proyecto. Los seguimientos generalmente se realizan sobre un proyecto concreto.

Tipo Auditoría: El tipo de auditoría es el que indica si la auditoría requiere o no proyecto, hay tipos de auditoría especiales y tipos no especiales, según el tipo de auditoría las preguntas y los procesos que se auditán son distintos.

Respuesta: Evidentemente son las respuestas a las preguntas de la auditoría, una auditoría consta de diversas preguntas (que vienen definidas según el tipo de auditoría) y las respuestas son las respuestas a esas preguntas.

Estado General: Hay dos relaciones a esta entidad, un estado general depende del número de respuestas negativas, equivale semánticamente al riesgo de la auditoría, el riesgo previo es aquel que se estima para la auditoría, y el riesgo posterior es el riesgo real después de ejecutar la auditoría.

Personal: Son dos relaciones, una para indicar quienes están siendo auditados en una auditoría, y otra para indicar quienes son los auditores, que son responsables de la ejecución de la auditoría.

### No Conformidad

Otra de las entidades más importantes es la No Conformidad (equivalente a Revisión), dado que en una auditoría es muy importante tener presente las respuestas negativas para intentar solucionarlas.

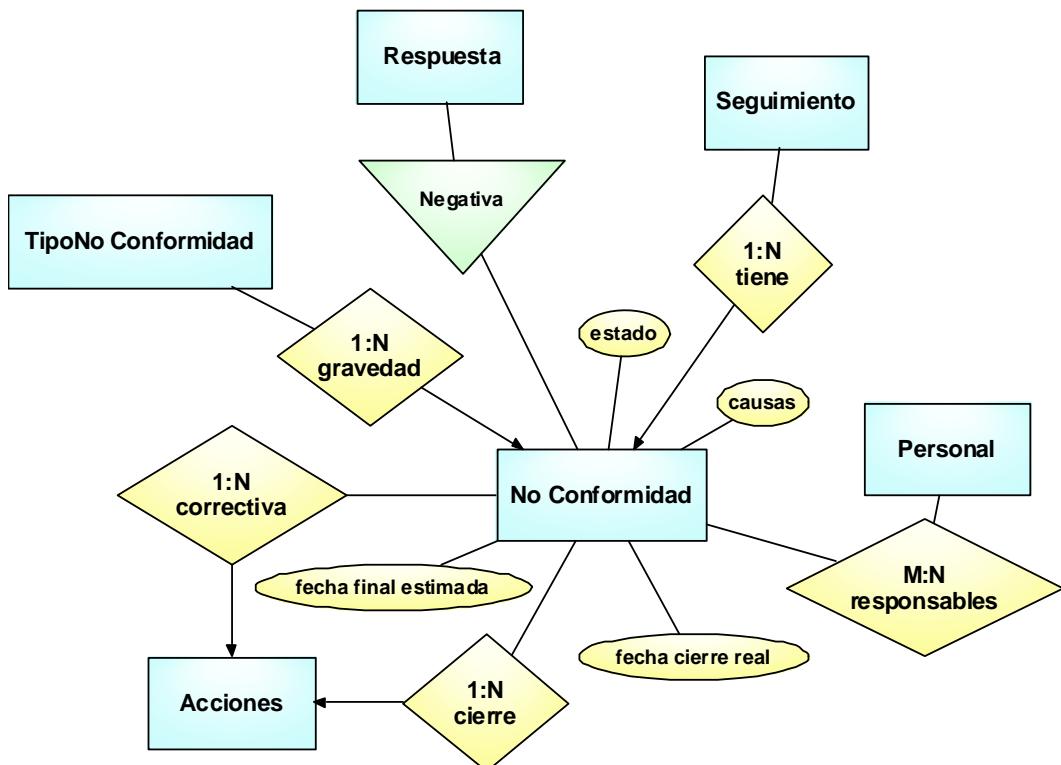


Figura 8 - Esquema conceptual No conformidad

### Atributos

Causas: Causas que han llevado a responder negativamente

Estado: Estado de la revisión (En curso, pendiente, finalizada, anulada)

Fecha final estimada: Fecha que el auditor cree conveniente tener solucionada la revisión

Fecha cierre real: Fecha en la cual se cerró la revisión dándola por finalizada/anulada.

### Relaciones

Personal: Se trata de los responsables que se asignan a una revisión para solucionarlo.

Acciones: Las acciones pueden ser para una revisión correctivas o de cierre, las primeras son aquellas que se plantean como solución, mientras que las de cierre son aquellas que finalmente se han realizado para solucionarlo.

Tipo de no conformidad: Se trata de la gravedad de una respuesta negativa.

### Estado general y Tipo de no conformidad

Estas dos entidades tienen sentido cuando por requisitos se explicitó que se requería la máxima configurabilidad posible, por lo que se permiten modificar los estados/riesgos y los tipos de no conformidad, así como la formula de calculo a través de la cual se establece un estado dado un tipo de no conformidad y un número de no conformidades (de ese tipo).

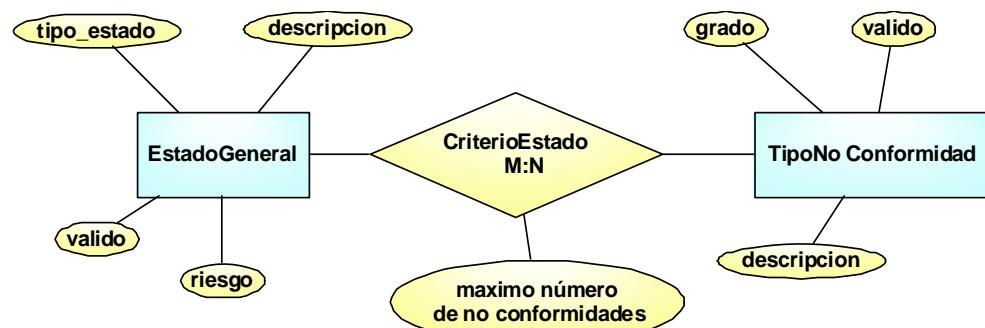


Figura 9 – Esquema conceptual Estado General y Tipo No Conformidad

#### Atributos: Estado General

Tipo\_estado: Clave primaria, nombre con el que se reconoce al estado

Descripción: Texto mostrado en la interfaz

Riesgo: Riesgo que incurre estar en dicho estado (Malo -> Alto, Bueno-> Bajo, etc.)

Valido: Si se usa o no dicho estado en el sistema.

#### Atributos: Tipo No Conformidad

Grado: Clave primaria, nombre con el que se reconoce al tipo

Descripción: Texto mostrado en la interfaz

Valido: Si se usa o no dicho estado en el sistema

#### Atributos: Relación (CriterioEstado)

Máximo número de no conformidades: Se trata del máximo número de no conformidades del tipo relacionado para el cual estamos en ese estado general. De esta forma podemos establecer la formula de cálculo obteniendo el peor estado como aquel que para un tipo de no conformidad cualquiera, el máximo numero de no conformidades es el más elevado para de los estados.

## Resto de entidades

El resto de entidades tienen menor importancia en lo que al sistema se refieren, sin embargo deben almacenarse, por ello las explicamos juntas.

Hemos omitido las relaciones con el resto de entidades por dejar mayor claridad.

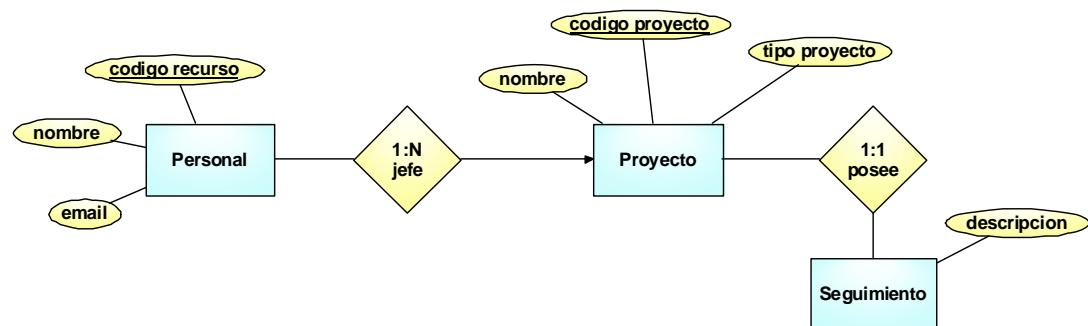


Figura 10 - Esquema conceptual Resto 1

Personal: Esta entidad tiene una importancia media, los datos almacenados son los explicitados en el documento de requisitos, añadiendo el email para poder enviar avisos.

Proyecto: Ocurre algo similar que al personal, los atributos de esta entidad son los requeridos según el informe de requisitos.

Seguimiento: Esta entidad sólo posee un campo descripción, es muy posible que carezca de fundamento, sin embargo simplifica el acceso a las revisiones de un proyecto y se especifica en el diagrama conceptual la importancia que tiene el seguimiento en el sistema, en el diseño físico podría omitirse y fusionarse con "Proyecto".



Figura 11 - Esquema conceptual Resto 2

Acciones: Esta entidad existe como tal debido a la adecuación a la primera forma normal, dado que sería un atributo multi-evaluado de la entidad "No conformidad".

Respuesta: Esta entidad es importante para las auditorías, dado que son las respuestas a las auditorías, podría en el diseño físico considerar omitir esta entidad y juntarla con "No conformidad", sin embargo es posible que muchas de las respuestas sean positivas y se guarde espacio para campos que nunca se rellenan.

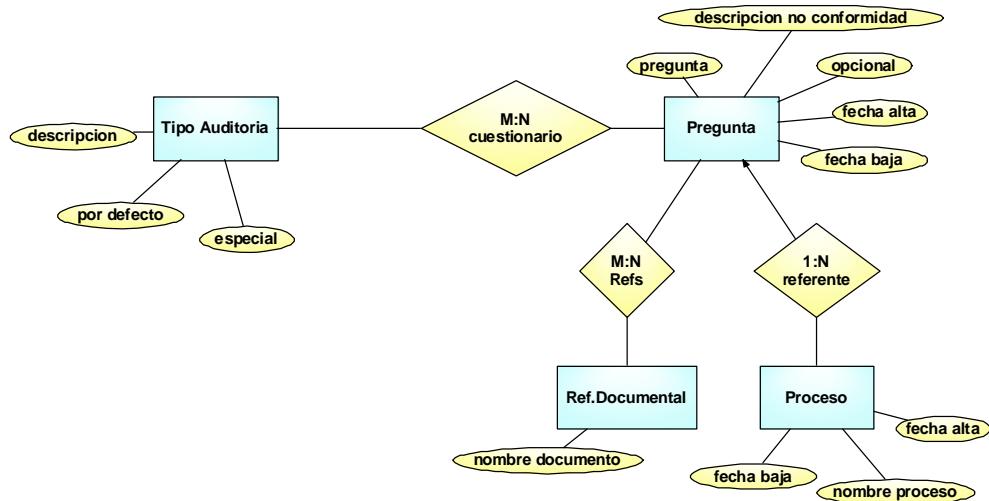


Figura 12 - Esquema conceptual Resto 3

**TipoAuditoría:** Esta entidad es configurable, por lo que es importante tenerla almacenada como entidad, además de que es la responsable de la generación del cuestionario para una auditoría.

**Pregunta:** Esta entidad tiene una importancia considerable y dado que son configurables e históricas (deben almacenarse las antiguas y poder darlas de baja) es fundamental que sean una entidad.

**Ref. Documental:** Con las referencias documentales pasa algo similar a las acciones, de no ser entidad serían atributo multi-evaluado, sin embargo tampoco tienen una importancia grande en el sistema para que sean entidad, pero deben serlo para cumplir la primera forma normal, de forma que queda constancia en el esquema conceptual (aunque realmente no haga falta).

**Proceso:** Los procesos son configurables e históricos, es por ello que deben ser entidad, de no ser así bastaría con un atributo en la pregunta indicando a qué proceso pertenecen.

## Modelo lógico de los datos

Tras el modelo conceptual de los datos se pasa al modelo lógico o esquema relacional de los datos, en el cual se transforma el esquema conceptual en esquema lógico, sobre él se realizan las comprobaciones de las formas normales (1<sup>a</sup> 2<sup>a</sup> y 3<sup>a</sup>), no se ha efectuado ni la 4<sup>a</sup> ni la 5<sup>a</sup> ni la de Boyce-Codd.

En la 3FN se encontraron las siguientes dependencias funcionales entre atributos:

**Auditoría:** Entre válida y estado (no para la fecha\_ininvalidación dado que puede tener fecha de invalidación pero haber sido re-validada y por lo tanto ser válida y con fecha de invalidación.)  
 estado = INVALIDA <-> valida = verdadero  
 estado != INVALIDA -> valida = falso

Sin embargo se mantuvo en el diseño físico para facilitar las consultas.

**EstadoGeneral:** Entre descripción y riesgo, existe una dependencia funcional entre estos atributos, sin embargo no es computable dado que es una dependencia semántica y que debe ser establecida por el administrador, por lo que el atributo se debe mantener.

## Normalización

### Acción

#### 1<sup>a</sup> Forma Normal

No existen atributos multi-evaluados y hay clave primaria

**Posee clave primaria: ok**

**descripción\_accion:** es atómico

**codigo\_revision:** clave foránea, atómica

**codigo\_revision\_cierre:** clave foránea, atómica

## 2<sup>a</sup> Forma Normal

Los atributos que no son parte de la clave dependen solo de la clave primaria.

**descripción\_accion:** No depende más que de la clave, dado que una revisión tiene múltiples acciones y pueden cambiar según el cliente decida.

**codigo\_revision:** Solo depende de la clave pues puede haber varias descripciones de acción para una misma revisión, lo mismo ocurre para **codigo\_revision\_cierre**.

## 3<sup>a</sup> Forma Normal

No existe dependencia funcional entre atributos que no son clave

No hay ninguna dependencia funcional

## Auditoría

### 1<sup>a</sup> Forma Normal

No existen atributos multi-evaluados y hay clave primaria

Mostramos los más conflictivos

**Tipo\_qa:** Una auditoría es de un único tipo.

**Riesgos:** Solo puede existir un riesgo posterior y un riesgo previo.

**Proyecto:** Las auditorías como máximo se realizan sobre un proyecto.

**Fecha\_auditación:** Es la fecha en la que se realizó, es única.

**Lista\_distribución:** Es una lista de emails, podría ser multi-evaluado pero se da en un formato de cadena compacto donde los emails están separados por comas, luego el atributo es único.

**Seguimiento:** Cada auditoría solo tiene un seguimiento.

## 2<sup>a</sup> Forma Normal

Los atributos que no son parte de la clave dependen solo de la clave primaria.

Avisado, inicio y acumulado no dependen de nada, salvo del **codigo\_auditoría**.

El **seguimiento depende del proyecto**, pero no siempre dado que algunas auditorías no tienen proyecto.

**Valida** depende del estado. Si estado = Invalidada -> valida=false. Si estado = Valida -> valida = trae. **ELIMINACIÓN DEL ATRIBUTO Válida**

Las horas ejecutadas y planificadas sólo dependen de la clave, así como la lista de distribución, fecha de auditación, proyecto, riesgos y el tipo de QA.

## 3<sup>a</sup> Forma Normal

No existe dependencia funcional entre atributos que no son clave

No existe ningún subconjunto del cual dependa algún atributo que no sea clave.

## AVISO

### 1<sup>a</sup> Forma Normal

No existen atributos multi-evaluados y hay clave primaria

Los avisos sólo tienen un tipo de aviso, una fecha de envío y un único email al cual enviarlo (de tenerlo) si no lo tiene significa que se utiliza la lista de distribución de la auditoría.

### 2<sup>a</sup> Forma Normal

Los atributos que no son parte de la clave dependen solo de la clave primaria.

**Seguimiento** depende (a veces) de **auditoría**, dado que se incorporó a la auditoría la clave foránea de seguimiento, y deberían coincidir.

Hay avisos que se generan por un seguimiento, donde se desconoce cual es la auditoría fuente de los cambios en el seguimiento, entonces seguimiento tiene un valor mientras que auditoría es nulo. Luego no podemos eliminarlo dado que la dependencia se rompe en estas circunstancias.

### 3<sup>a</sup> Forma Normal

No existe dependencia funcional entre atributos que no son clave

Un mismo aviso puede enviarse:

En distintas fechas al mismo email del mismo tipo y por la misma auditoría o seguimiento.

En la misma fecha, del mismo tipo y por la misma auditoría o seguimiento pero a distinto email.

Al mismo email y misma fecha por la misma auditoría o seguimiento pero de distinto tipo.

Al mismo email y misma fecha y tipo pero por un seguimiento ó auditoría distintos.

Luego no hay dependencias entre subconjuntos de atributos.

## CRITERIO ESTADO

Es una tabla de un único atributo, luego esta en 3<sup>a</sup> FN.

## ESTADO GENERAL

### 1<sup>a</sup> Forma Normal

No existen atributos multi-evaluados y hay clave primaria

Los atributos solo tienen un valor para cada tupla.

### 2<sup>a</sup> Forma Normal

Los atributos que no son parte de la clave dependen solo de la clave primaria.

Riesgo y Estado son semánticamente dependientes, pero no podemos saber semánticamente el riesgo que incurre estar en un estado. Por ejemplo estar en un estado "Bueno" incurre en un estado "Bajo", en un estado "Regular" incurre en un riesgo "Medio" y en un estado "Malo" incurre un riesgo "Alto", pero esto no podemos saberlo a priori ya que depende de la percepción del usuario para establecer el riesgo que incurre un determinado estado.

Existe dependencia funcional semántica por lo que la tabla estado general debería separarse en dos:

EstadoGeneral (Tipo\_estado, estado, valido)

EstadoGeneralRiesgo (estado, riesgo)

Sin embargo preferimos realizar lo siguiente: Establecer **estado** como clave primaria de la relación (llamándola **tipo\_estado**) y cambiar el significado del atributo estado, siendo este la descripción del estado, que es lo que realmente se muestra en la interfaz.

### 3<sup>a</sup> Forma Normal

No existe dependencia funcional entre atributos que no son clave

## **NO CONFORMIDAD**

### **1<sup>a</sup> Forma Normal**

No existen atributos multi-evaluados y hay clave primaria

Las causas de la no conformidad, son simplemente un campo de texto donde se describen las causas, no son varias causas separadas.

Una no conformidad pertenece a un solo seguimiento.

Una no conformidad es la respuesta negativa de una pregunta, luego es solo de una respuesta, y tiene una gravedad.

### **2<sup>a</sup> Forma Normal**

Los atributos que no son parte de la clave dependen solo de la clave primaria.

Si fecha de cierre no es nulo entonces el estado será finalizada o invalidada, luego no dependen.

### **3<sup>a</sup> Forma Normal**

No existe dependencia funcional entre atributos que no son clave.

No existen dependencias entre subconjuntos

## **PERSONAL**

### **1<sup>a</sup> Forma Normal**

No existen atributos multi-evaluados y hay clave primaria

El email es único, solo se almacena uno.

### **2<sup>a</sup> Forma Normal**

Los atributos que no son parte de la clave dependen solo de la clave primaria.

El email que se genera depende del nombre, pero no siempre dado que puede haber personas con el mismo nombre y apellido y no podrán tener el mismo email, pero no siempre se utiliza la misma forma de generar el email por lo que no existe la dependencia funcional.

### **3<sup>a</sup> Forma Normal**

No existe dependencia funcional entre atributos que no son clave

No hay subconjuntos, si esta en 2<sup>a</sup> FN también esta en 3<sup>a</sup> FN.

## **PERSONAL AUDITADO**

Tabla sin atributos, entonces esta en 3<sup>a</sup> FN

## **PERSONAL AUDITOR**

Tabla sin atributos, entonces esta en 3<sup>a</sup> FN

## **PREGUNTA**

### **1<sup>a</sup> Forma Normal**

No existen atributos multi-evaluados y hay clave primaria

Ninguno es conflictivo

### **2<sup>a</sup> Forma Normal**

Los atributos que no son parte de la clave dependen solo de la clave primaria.

**Pregunta** es clave candidata, pero debido a su longitud se estableció en el diseño físico no usarla como clave primaria y utilizar una clave autogenerada para que el rendimiento fuera mejor y el índice ocupara menos.

En el resto de atributos no hay conflictos

### **3<sup>a</sup> Forma Normal**

No existe dependencia funcional entre atributos que no son clave

No hay dependencias entre atributos.

## **PREGUNTAS POR TIPO DE AUDITORÍA**

Tabla sin atributos, entonces esta en 3<sup>a</sup> FN

### **PROYECTO**

#### **1<sup>a</sup> Forma Normal**

Existe un único jefe de proyecto por proyecto, y cada proyecto tiene un único seguimiento.

#### **2<sup>a</sup> Forma Normal**

Nombre de proyecto puede ser clave candidata, dado que no existen dos proyectos con el mismo nombre, luego todo depende del nombre de proyecto, al ser clave candidata.

No realizamos cambios para normalizar dado que la información de los proyectos se obtiene desde un servicio web al que se accede mediante el código de proyecto, por lo que necesitamos dicha clave.

#### **3<sup>a</sup> Forma Normal**

No existe dependencia funcional entre atributos que no son clave

El tipo de proyecto no depende del jefe ni del seguimiento ni de ambos, dado que el tipo de proyecto se refiere a las horas del proyecto.

## **REFERENCIA DOCUMENTAL**

En el diseño físico podría considerarse la **eliminación** de esta tabla

Tabla sin atributos, entonces esta en 3<sup>a</sup> FN

## **REFERENCIA DOCUMENTAL PREGUNTA**

Tabla sin atributos, entonces esta en 3<sup>a</sup> FN

### **RESPONSABLES REVISIÓN**

Tabla sin atributos, entonces esta en 3<sup>a</sup> FN

### **RESPUESTA**

#### **1<sup>a</sup> Forma Normal**

No existen atributos multi-evaluados y hay clave primaria

A una pregunta de una auditoría solo puede haber una respuesta con unas observaciones.

#### **2<sup>a</sup> Forma Normal**

Los atributos que no son parte de la clave dependen solo de la clave primaria.

No hay dependencias únicas, la respuesta no depende de la auditoría, ni de la pregunta

### **3<sup>a</sup> Forma Normal**

No existe dependencia funcional entre atributos que no son clave

La respuesta depende de lo que se haya o no hecho en el proyecto, lo cual no está almacenado. Las observaciones dependen de cuestiones que tampoco se almacenan en la base de datos.

### **SEGUIMIENTO**

Tabla con un único atributo, entonces está en 3<sup>a</sup> FN

### **TIEMPO-POR-DEFECTO**

Tabla con un único atributo, entonces está en 3<sup>a</sup> FN

### **TIPO DE AUDITORÍA**

#### **1<sup>a</sup> Forma Normal**

No existen atributos multi-evaluados y hay clave primaria

Si una auditoría es especial (actualmente) no viene marcada por defecto, mientras que si no lo es, si está marcada por defecto, pero esta opción de marcado por defecto puede variar, por lo que se mantiene este atributo de (**por\_defecto**) dado que en un futuro es muy probable que se creen auditorías extra que no sean muy comunes, por lo que no irán marcadas por defecto y podrán no ser especiales.

#### **2<sup>a</sup> Forma Normal**

Los atributos que no son parte de la clave dependen solo de la clave primaria.

La descripción es un campo que no depende de nada salvo del tipo\_qa y nada depende de él.

#### **3<sup>a</sup> Forma Normal**

No existe dependencia funcional entre atributos que no son clave

No existen conflictos

### **TIPO DE NO CONFORMIDAD**

No hay más que dos atributos que no dependen entre ellos ni son multi-evaluados, luego está en 3<sup>a</sup> FN.

### **TIPO DE PROCESO**

#### **1<sup>a</sup> Forma Normal**

No existen atributos multi-evaluados y hay clave primaria

Fecha de alta, baja y nombre son únicos.

#### **2<sup>a</sup> Forma Normal**

Los atributos que no son parte de la clave dependen solo de la clave primaria.

El nombre de proceso es una clave candidata, podría establecerse como clave.

#### **3<sup>a</sup> Forma Normal**

No existe dependencia funcional entre atributos que no son clave  
Las fechas no dependen entre sí y nombre es clave candidata.

## **Modelo físico de los datos (Implementación de la Base de Datos)**

En el diseño físico se han tomado las siguientes consideraciones:

Para tablas con claves candidatas grandes (Como en la relación **Pregunta, etc**) se ha optado por crear una clave autogenerada entera de forma que la indexación fuera más eficiente.

En cuanto a Tipo de Proceso y la clave candidata "Nombre", el sistema permite la opción de modificar dicho atributo y dado que SQL Server 2005 tiene problemas en la actualización en cascada, hemos optado por mantenerlo como atributo en el diseño físico y tener una clave primaria autogenerada.

La tabla Referencia Documental no ha sido eliminada dado que el mapeo de JPA obliga a tener dicha tabla.

Para poder entender la simbología del modelo físico de datos, se dan unas pautas básicas:

Las entidades vienen reflejadas con rectángulos que poseen una identificación en su cabecera la cual da nombre a la entidad marcada con una llave amarilla.

Las relaciones se indican mediante una llave en un extremo, que indica que en esa tabla estará la clave foránea y el otro extremo de la relación la tabla donde se encuentra la clave primaria de la relación.

Con todo ello se pueden interpretar los diagramas de entidad relación del sistema y observar más en detalle una descripción de cada entidad, los atributos propios de las entidades así como las relaciones que posee cada entidad con el resto. Es por ello que este modelo lógico de datos se acerca mucho más si cabe al almacenamiento físico de la información.

A continuación se representa el esquema lógico de datos del nuevo sistema de información.

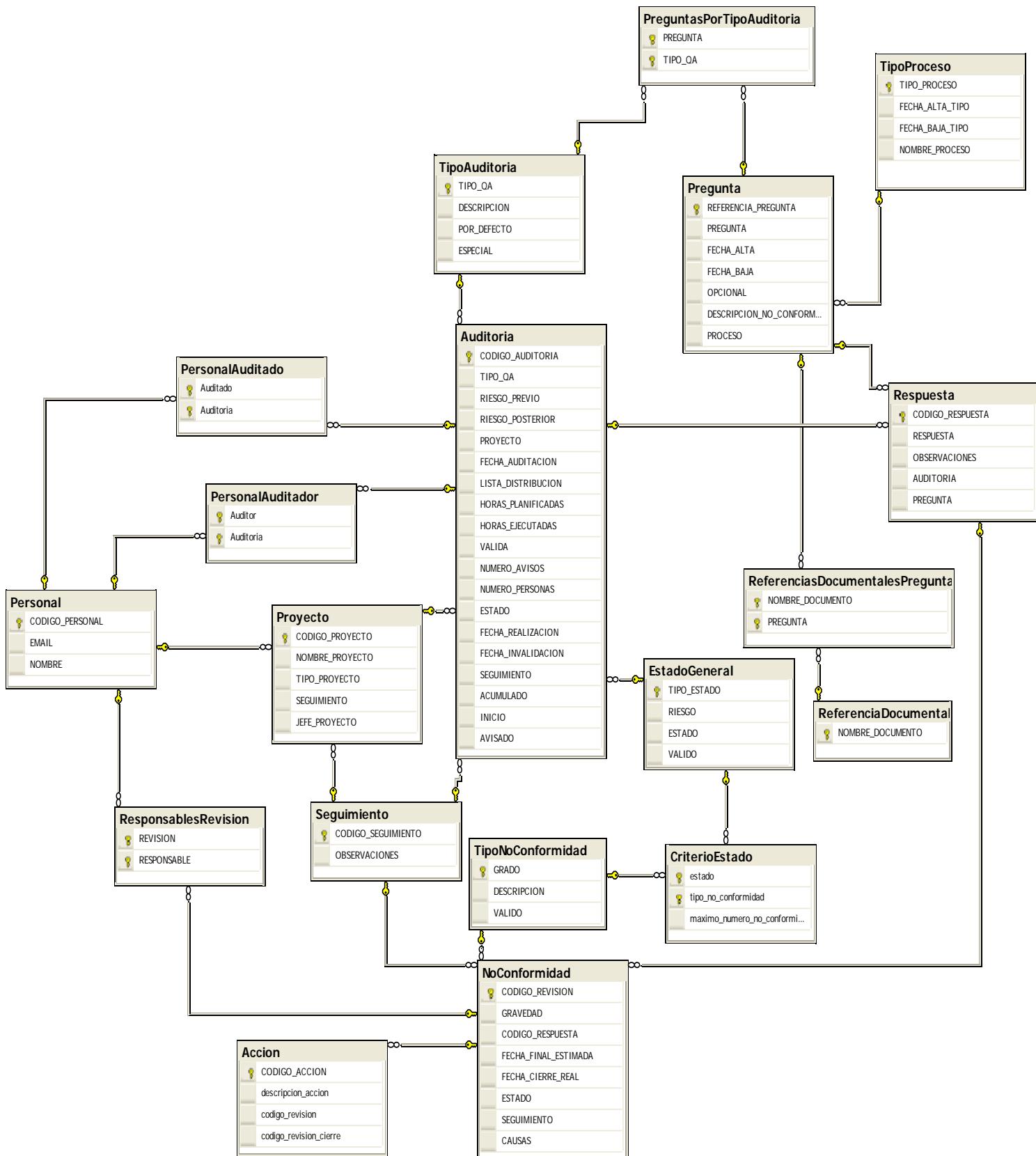


Figura 13 – Esquema físico de los datos

## Diccionario de Datos

Dentro de este apartado se define la estructura física de datos que utilizará el sistema, que consiste en una descripción de las entidades, detallándose nuevos conceptos, como son el tipo de dato de cada atributo de cada entidad, así como la longitud estipulada para ese campo. Además de todo esto en cada entidad aparecerán las claves externas de otras entidades que posibilitan la correlación de datos dentro del sistema de almacenamiento.

### ACCION

ENTIDAD: ACCION			
Entidad en la que se almacenan las acciones correctivas y de cierre de revisiones de no conformidad.			
Atributos	Tipo Dato	Clave	Descripción
CODIGO_ACCION	int	Primaria	código identificativo
descripcion_accion	varchar(255)	No	Descripción de la acción correctiva
codigo_revision	int	Externa	acción correctiva de una revisión.
codigo_revision_cierre	int	Externa	acción de cierre de una revisión

### AUDITORÍA

ENTIDAD: AUDITORÍA			
Entidad en la que se almacena la información relevante a cada auditoría			
Atributos	Tipo Dato	Clave	Descripción
CODIGO_AUDITORÍA	varchar(20)	Primaria	código identificativo
TIPO_QA	varchar(10)	Externa	Tipo de auditoría ( QA1, QA2.. )
RIESGO_PREVIO	varchar(20)	Externa	Riesgo previo a la auditoría
RIESGO_POSTERIOR	varchar(20)	Externa	Riesgo posterior a la auditoría
PROYECTO	varchar(50)	Externa	Proyecto que se audita
FECHA_AUDITACION	datetime	No	Fecha de planificación
LISTA_DISTRIBUCION	varchar(50)	No	Lista de distribución (email).
HORAS_PLANIFICADAS	int	No	Horas planificadas para realización
HORAS_EJECUTADAS	bigint	No	Horas ejecutadas reales
NUMERO_AVISOS	int	No	Numero de avisos realizados para la planificación
NUMERO_PERSONAS	int	No	Numero de personas que ejecutaron la auditoría.
ESTADO	int	No	Estado de la auditoría, en curso, planificada, completada, etc..
FECHA_REALIZACION	datetime	No	Fecha en la que se completó.
FECHA_INVALIDACION	datetime	No	Fecha en la que se invalidó.
SEGUIMIENTO	int	Externa	Seguimiento de la auditoría.
ACUMULADO	bigint	No	Tiempo acumulado hasta ahora en la ejecución de la auditoría
INICIO	bigint	No	Tiempo de inicio de la auditoría, para calcular las horas ejecutadas.
AVISADO	bit	No	Si ha sido avisada o no la auditoría para su planificación.

## AVISO

ENTIDAD: AVISO			
Entidad en la que se almacena la información de los avisos pendientes de realizar			
Atributos	Tipo Dato	Clave	Descripción
código	int	Primaria	código del aviso
tipo_aviso	int	No	Que clase de aviso es
email	varchar(50)	No	Email al que hay que enviarlo
fecha_envio	datetime	No	Fecha en la que hay que enviarlo
auditoría	varchar(20)	Externa	Auditoría por la que tuvo lugar el aviso
seguimiento	int	Externa	Seguimiento por el que tuvo lugar el aviso.

## CRITERIO ESTADO

ENTIDAD: CRITERIOESTADO			
Entidad en la que se almacena la formula de cálculo para estar en un estado.			
Atributos	Tipo Dato	Clave	Descripción
estado	varchar(20)	Externa Primaria	Estado objetivo
tipo_no_conformidad	varchar(20)	Externa Primaria	Tipo de no conformidad
maximo_numero_no_conformidades	int	Checked	Máximo número de no conformidades del tipo para estar en el estado.

## ESTADO GENERAL

ENTIDAD: ESTADO GENERAL			
Entidad en la que se almacenan los distintos estados generales posibles.			
Atributos	Tipo Dato	Clave	Descripción
TIPO_ESTADO	varchar(20)	Primaria	Clave primaria
RIESGO	varchar(20)	No	Riesgo, si el estado es bueno el riesgo es bajo. Etc.
ESTADO	varchar(20)	No	Descripción, lo que realmente se muestra.
VALIDO	bit	No	Dice si se usa o no este estado.

## NO CONFORMIDAD

ENTIDAD: NO CONFORMIDAD			
Entidad que equivale a Revisión, almacena los datos de las no conformidades de las respuestas.			
Atributos	Tipo Dato	Clave	Descripción
CODIGO_REVISION	int	Primaria	código de la no conformidad.
GRAVEDAD	varchar(20)	Externa	Tipo de no conformidad
CODIGO_RESPUESTA	int	Externa	Respuesta de la auditoría
FECHA_FINAL_ESTIMADA	datetime	No	Fecha final estimada de la revisión
FECHA_CIERRE_REAL	datetime	No	Fecha de cierre de la revisión
ESTADO	varchar(20)	No	Estado de la revisión
SEGUIMIENTO	int	Externa	Seguimiento al que pertenece la revisión
CAUSAS	varchar(300)	No	Causas de la no conformidad.

## PERSONAL

ENTIDAD: PERSONAL			
Entidad donde se almacenan los datos de las personas auditadas y auditores.			
Atributos	Tipo Dato	Clave	Descripción
CODIGO_PERSONAL	varchar(20)	Primaria	código identificativo.
EMAIL	varchar(50)	No	Email al que se envían los avisos
NOMBRE	varchar(50)	No	Nombre de la persona

## PERSONAL AUDITADO

ENTIDAD: PERSONAL AUDITADO			
Entidad donde se almacenan los datos de las personas auditadas y auditores.			
Atributos	Tipo Dato	Clave	Descripción
Auditado	varchar(20)	Primaria Externa	Auditado
Auditoría	varchar(20)	Primaria Externa	Auditoría

## PERSONAL AUDITOR

ENTIDAD: PERSONAL AUDITOR			
Entidad donde se almacenan los datos de las personas auditadas y auditores.			
Atributos	Tipo Dato	Clave	Descripción
Auditor	varchar(20)	Primaria Externa	Auditado
Auditoría	varchar(20)	Primaria Externa	Auditoría

## PREGUNTA

ENTIDAD: PREGUNTA			
Entidad donde se almacenan las preguntas de las auditorías.			
Atributos	Tipo Dato	Clave	Descripción

REFERENCIA_PREGUNTA	int	Primaria	código identificativo
PREGUNTA	varchar(300)	No	Descripción de la pregunta
FECHA_ALTA	datetime	No	Fecha en la que se insertó
FECHA_BAJA	datetime	No	Fecha en la que se dio de baja la pregunta.
OPCIONAL	bit	No	Indica si es opcional o no responderla
DESCRIPCION_NO_CONFORRIDAD	varchar(300)	No	Indica la descripción en caso de no conformidad.
PROCESO	int	Externa	Proceso en el que la pregunta tiene lugar.

#### PREGUNTAS POR TIPO DE AUDITORÍA

ENTIDAD: PREGUNTAS-POR-TIPO-DE-AUDITORÍA			
Entidad donde se almacena la relación de en qué tipos de auditoría se realiza una pregunta.			
Atributos	Tipo Dato	Clave	Descripción
PREGUNTA	int	Primaria Externa	Pregunta
TIPO_QA	varchar(10)	Primaria Externa	Tipo de auditoría

#### PROYECTO

ENTIDAD: PROYECTO			
Entidad en la que se almacena la información necesaria de los proyectos.			
Atributos	Tipo Dato	Clave	Descripción
CODIGO_PROYECTO	varchar(50)	Primaria	código identificativo.
NOMBRE_PROYECTO	varchar(50)	No	Nombre del proyecto
TIPO_PROYECTO	int	No	Tipo de proyecto según las horas.
SEGUIMIENTO	int	Externa	Seguimiento del proyecto
JEFE_PROYECTO	varchar(20)	Externa	código del jefe de proyecto

#### REFERENCIA DOCUMENTAL

ENTIDAD: REFERENCIA-DOCUMENTAL			
Entidad donde se almacena la información de las referencias documentales, para poder añadir/quitar referencias libremente se crea esta entidad, así como para poder usar JPA, que espera esta entidad.			
Atributos	Tipo Dato	Clave	Descripción
NOMBRE_DOCUMENTO	varchar(100)	Primaria	código identificativo que corresponde al nombre del documento.

#### REFERENCIA DOCUMENTAL PREGUNTA

ENTIDAD: REFERENCIA-DOCUMENTAL-PREGUNTA			
Entidad donde se almacena la relación entre una pregunta y las referencias documentales en las que la respuesta se documenta.			
Atributos	Tipo Dato	Clave	Descripción
NOMBRE_DOCUMENTO	varchar(100)	Externa Primaria	Nombre del documento
PREGUNTA	int	Externa Primaria	Pregunta

#### RESPONSABLES REVISIÓN

**ENTIDAD: RESPONSABLES-REVISION**

Entidad donde se almacena la relación existente entre una revisión y el personal, diciendo quienes son los responsables de la corrección de una no conformidad

<b>Atributos</b>	<b>Tipo Dato</b>	<b>Clave</b>	<b>Descripción</b>
REVISION	int	Externa Primaria	revisión de la que se hacen cargo
RESPONSABLE	varchar(20)	Externa Primaria	Persona responsable.

**RESPUESTA****ENTIDAD: RESPUESTA**

Entidad donde se almacenan las respuestas a las preguntas de una auditoría

<b>Atributos</b>	<b>Tipo Dato</b>	<b>Clave</b>	<b>Descripción</b>
CODIGO_RESPUESTA	int	Primaria	código identificativo
RESPUESTA	varchar(10)	No	Respuesta propiamente dicha. ( Si, No, No aplica )
OBSERVACIONES	varchar(300)	No	Observaciones de la respuesta
AUDITORÍA	varchar(20)	Externa	Auditoría en la cual tiene lugar
PREGUNTA	int	Externa	Pregunta a la cual se responde.

**SEGUIMIENTO****ENTIDAD: SEGUIMIENTO**

Entidad donde se almacenan los seguimientos

<b>Atributos</b>	<b>Tipo Dato</b>	<b>Clave</b>	<b>Descripción</b>
CODIGO_SEGUIMIENTO	int	Primaria	código identificativo
OBSERVACIONES	varchar(200)	No	Observaciones del seguimiento.

## TIEMPO-POR-DEFECTO

ENTIDAD: TIEMPO POR DEFECTO			
Atributos			
TIPO_TIEMPO			
TIPO_TIEMPO	varchar(40)	Primaria	código identificativo
TIEMPO_POR_DEFECTO	int	No	Numero de días / número de avisos.

## TIPO DE AUDITORÍA

ENTIDAD: TIPO-AUDITORÍA			
Atributos			
TIPO_QA			
TIPO_QA	varchar(10)	Primaria	código identificativo y nombre del tipo ("QA1").
DESCRIPCION	varchar(200)	No	Descripción del tipo de auditoría, cuando se realiza, etc.
POR_DEFECTO	bit	No	Define si este tipo de auditoría se marca por defecto o no.
ESPECIAL	bit	No	Define si este tipo de auditoría requiere proyecto o no.

## TIPO DE NO CONFORMIDAD

ENTIDAD: TIPO-NO-CONFORMIDAD			
Atributos			
GRADO			
GRADO	varchar(20)	Primaria	código identificativo.
DESCRIPCION	varchar(20)	No	Nombre que se muestra por defecto, del tipo de no conformidad.
VALIDO	bit	No	Indica si se utiliza este tipo de no conformidad o no.

## TIPO DE PROCESO

ENTIDAD: TIPO-PROCESO			
Atributos			
TIPO_PROCESO			
TIPO_PROCESO	int	Primaria	código identificativo.
FECHA_ALTA_TIPO	datetime	No	Fecha de alta del proceso
FECHA_BAJA_TIPO	datetime	No	Fecha de baja del proceso
NOMBRE_PROCESO	varchar(50)	No	Nombre que se muestra del proceso.

# ANEXO 5 – Diseño

## Índice

<b>ANEXO 5 – DISEÑO .....</b>	<b>51</b>
ÁMBITO DE ESTE DOCUMENTO .....	52
CASOS DE USO .....	53
<i>Casos de uso de la parte pública</i> .....	53
<i>Casos de uso de la parte privada</i> .....	58
ARQUITECTURA DE SOFTWARE DEL SISTEMA .....	61
<i>Capa de Modelo</i> .....	61
<i>Fachada y componentes</i> .....	62
<i>Servidor web</i> .....	63
DISEÑO DE CLASES .....	64
PATRONES DE DISEÑO .....	65
<i>Patron DAO</i> .....	65
<i>Patron Facade</i> .....	65

## Ámbito de este documento

En este documento se describen las tareas realizadas en la fase de diseño de la aplicación completa, describiendo la arquitectura software del sistema, los casos de uso generados en la fase de diseño, los patrones de diseño a utilizar en la implementación del sistema así como el diseño de las clases que forman el núcleo del modelo de negocio de la aplicación.

## Casos de uso

Se han definido los casos de uso de cada una de las partes del sistema (pública y privada), modelando de forma abstracta cada acción posible hacia el sistema.

### Casos de uso de la parte pública

Se han generado los diversos casos de uso a partir de los requisitos funcionales del sistema, de forma que queden ordenados.

#### Gestión de avisos

- RF2.-Inserción e inspección de avisos.
- RF2.- Existencia de tareas periódicas.
- RF2.- Enviar avisos por email.

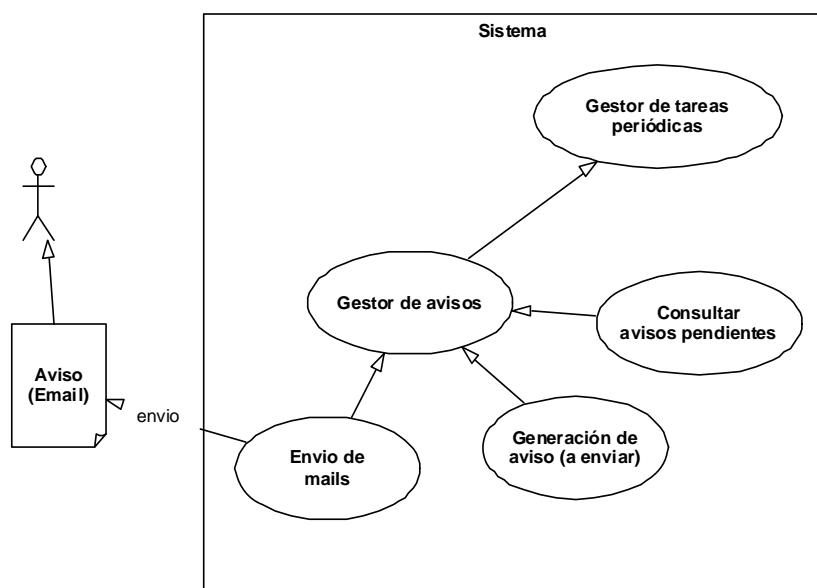


Figura 14 – Caso de uso Avisos

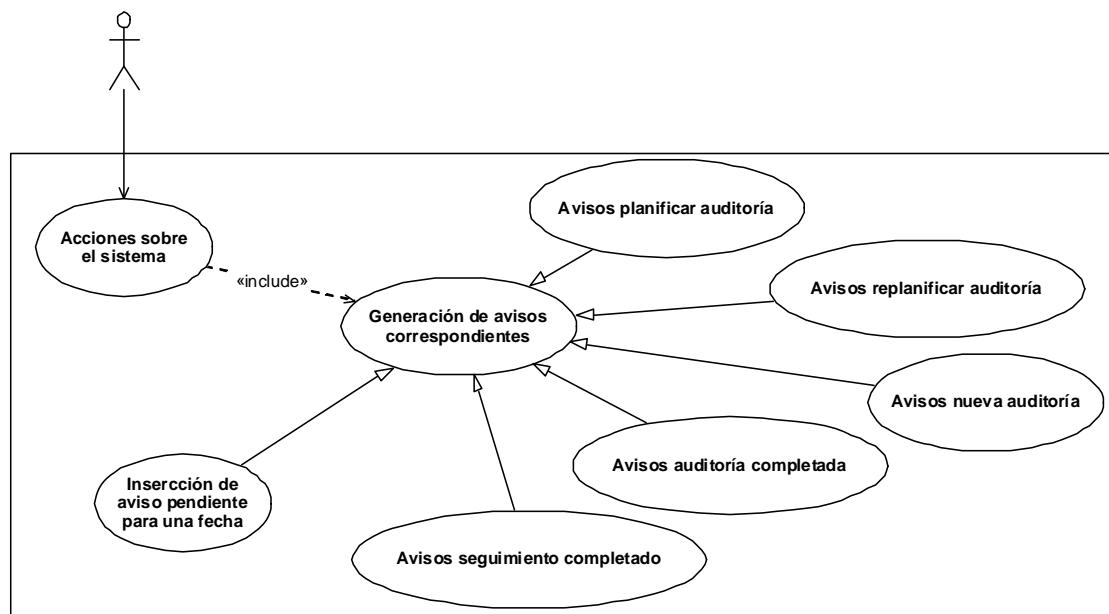


Figura 15

## Gestión de auditorías.

### Inserción

- RF3.- INSERCIÓN** Mostrar proyectos existentes  
**RF3.- INSERCIÓN** Añadir proyectos por código de proyecto obteniéndolos mediante el servicio web.  
**RF3.- INSERCIÓN** Mostrar personal existente.  
**RF3.- INSERCIÓN** Añadir personal por código de recurso obteniéndolo mediante el servicio web.  
**RF3.- INSERCIÓN** Posibilidad de seleccionar varios como auditores y varios como auditados.  
**RF3.- INSERCIÓN** Mostrar los tipos de auditoría validas existentes, permitiendo seleccionar varias especiales o varias no especiales.

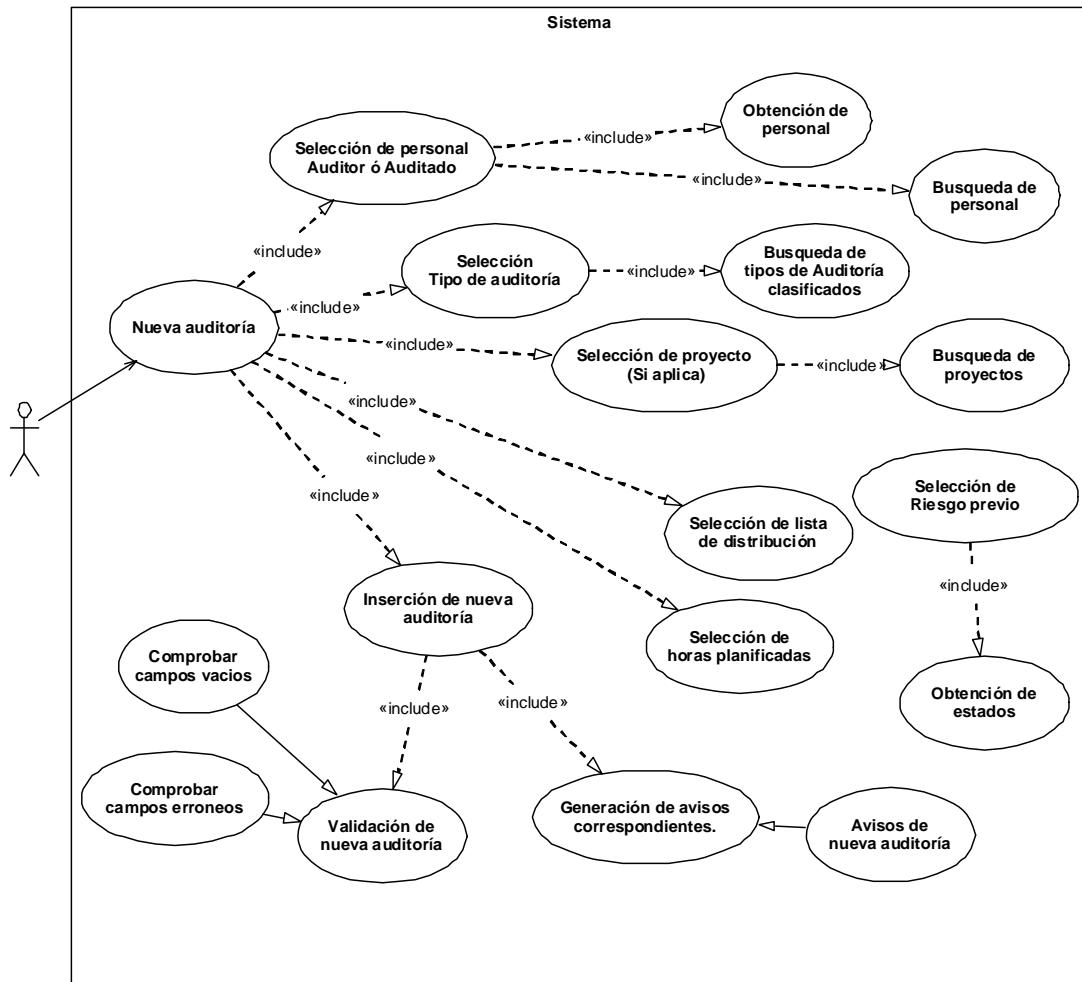


Figura 16 – Caso de uso inserción de auditoría

### Modificación

- RF3.- MODIFICACION** Mostrar información referente a una auditoría permitiendo modificarla.  
**RF3.- MODIFICACION** Posibilidad de invalidar auditorías.

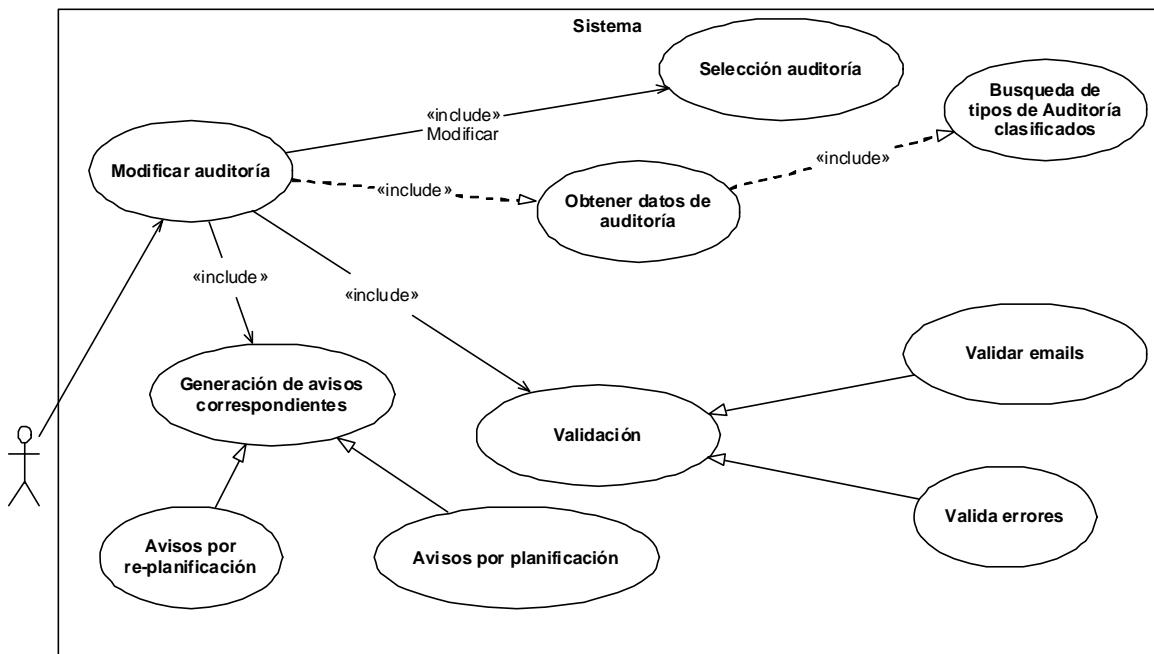


Figura 17 – Caso de uso modificación auditoría

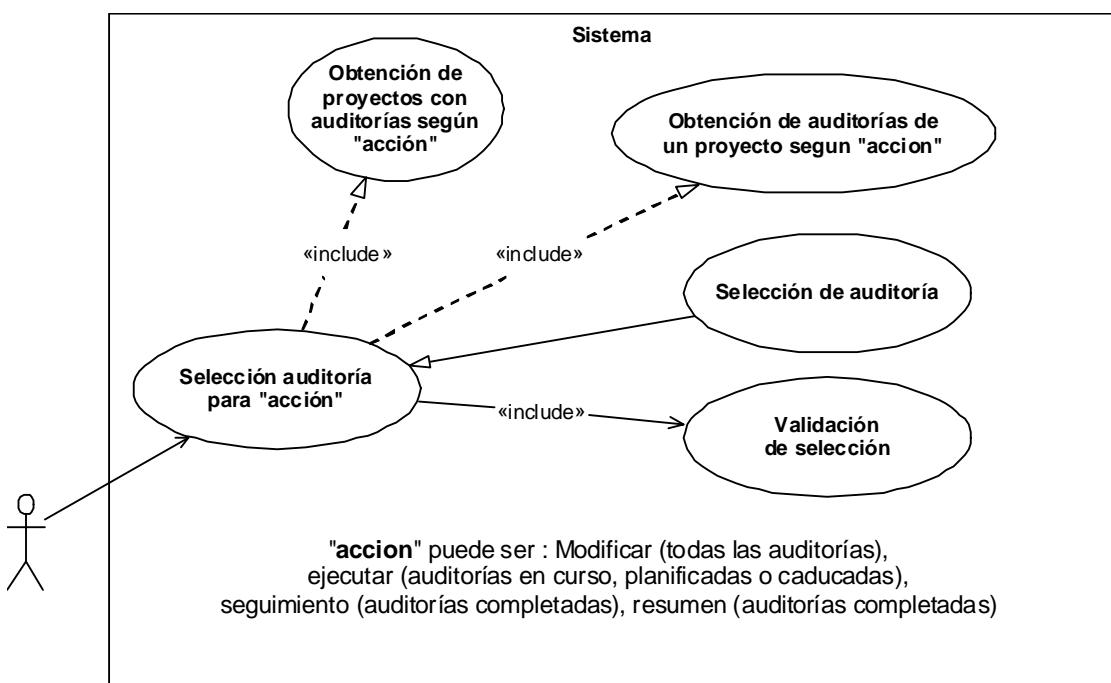


Figura 18 – Selección de auditoría

## Ejecución

**RF3.- EJECUCIÓN** Acceso al cuestionario (si existe), generación del cuestionario, ordenándolo por proceso y mostrando aquella página (preguntas de un proceso) seleccionado.

**RF3.- EJECUCIÓN** Saber cuantos procesos tienen lugar en la ejecución para la generación de las pestañas.

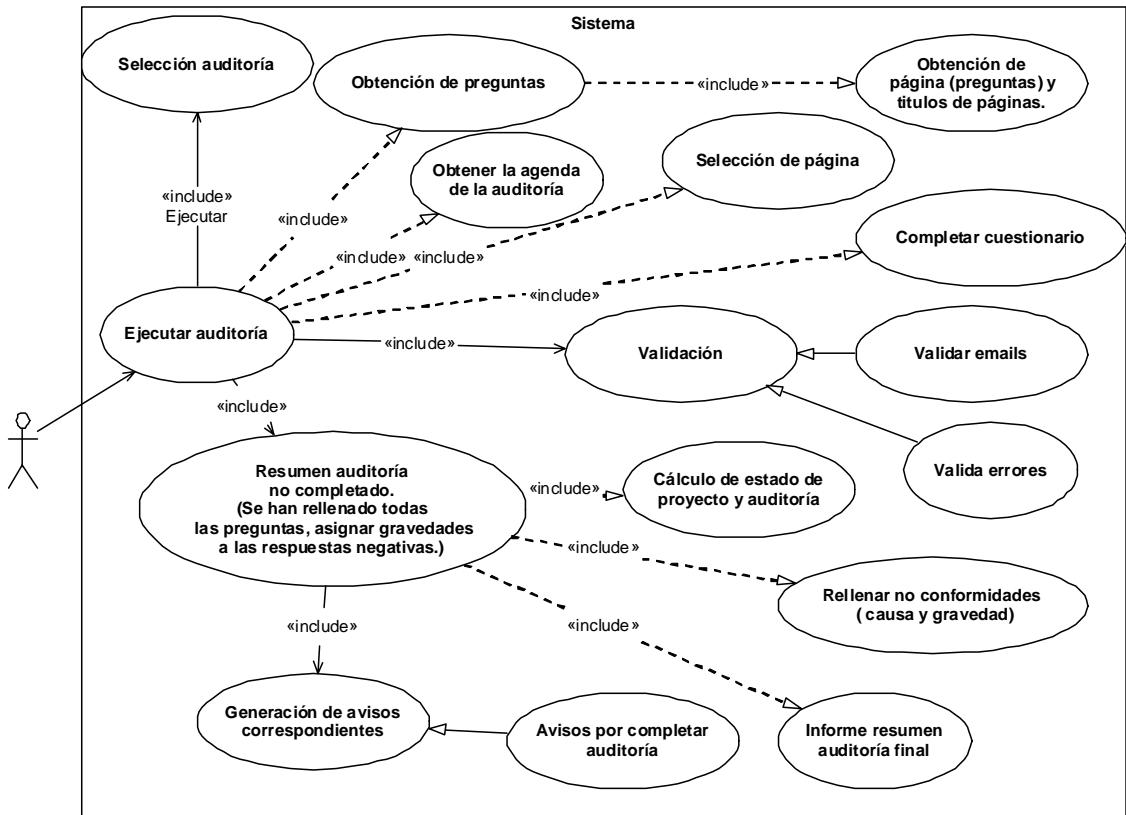


Figura 19 – Caso de uso ejecución auditoría

## Informes y métricas.

**RF4.-** Acceso a la información referente a los diversos informes.

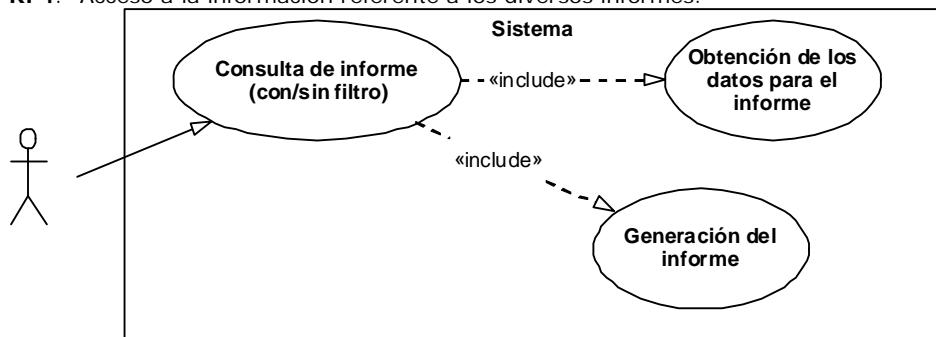


Figura 20 – Caso de uso informes

**RF5.-** Acceso a la información referente a las diversas métricas.

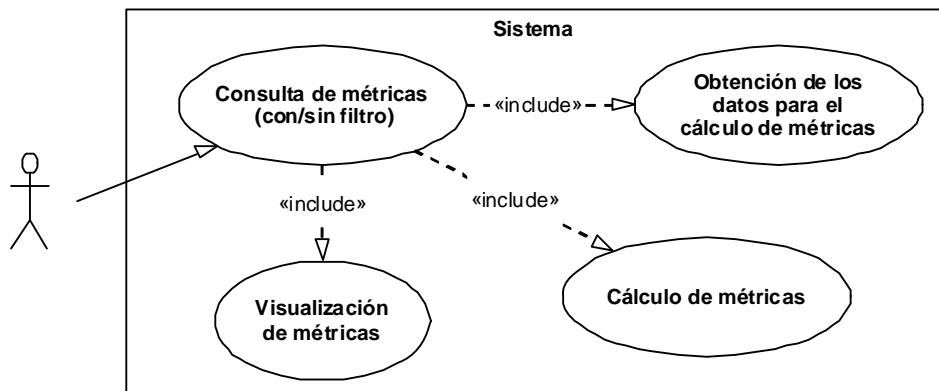


Figura 21 – Caso de uso metricas

### Seguimientos

**RF6.-** Para cada auditoría completada, el acceso a su seguimiento con la posibilidad de modificar los campos correspondientes a las acciones correctivas/cierre, responsables, fechas de cierre, estado de las revisiones, etc. Acceso a las revisiones de un seguimiento / auditoría determinado.

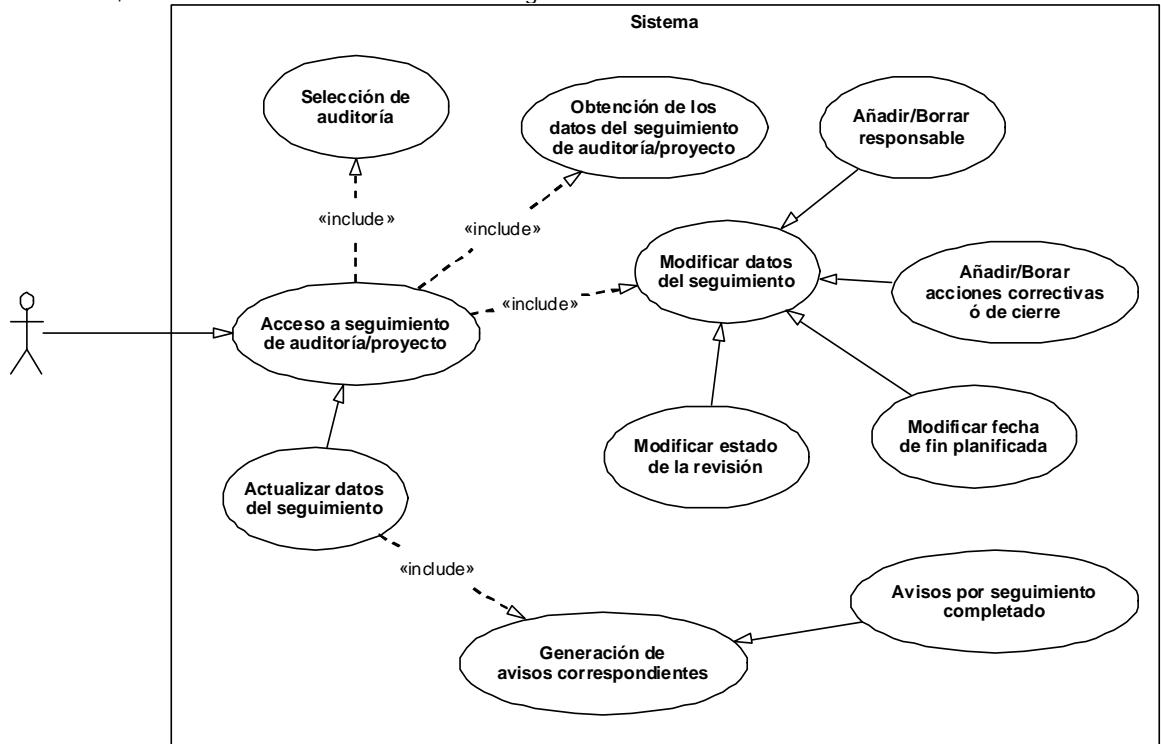


Figura 22 – Caso de uso seguimientos

**RF\*.-** Acceso al resultado de una auditoría ya completada sin posibilidad de modificación.

**RF\*.-** Acceso al seguimiento de un proyecto sin posibilidad de modificación.

**RF\*.-** Acceso a la administración.

**RF\*.-** Acceso al cuestionario de una auditoría completada sin posibilidad de modificación.

## Casos de uso de la parte privada

### Autenticación, acceso a la parte privada

RF1.- Comprobación de la corrección de usuario/contraseña

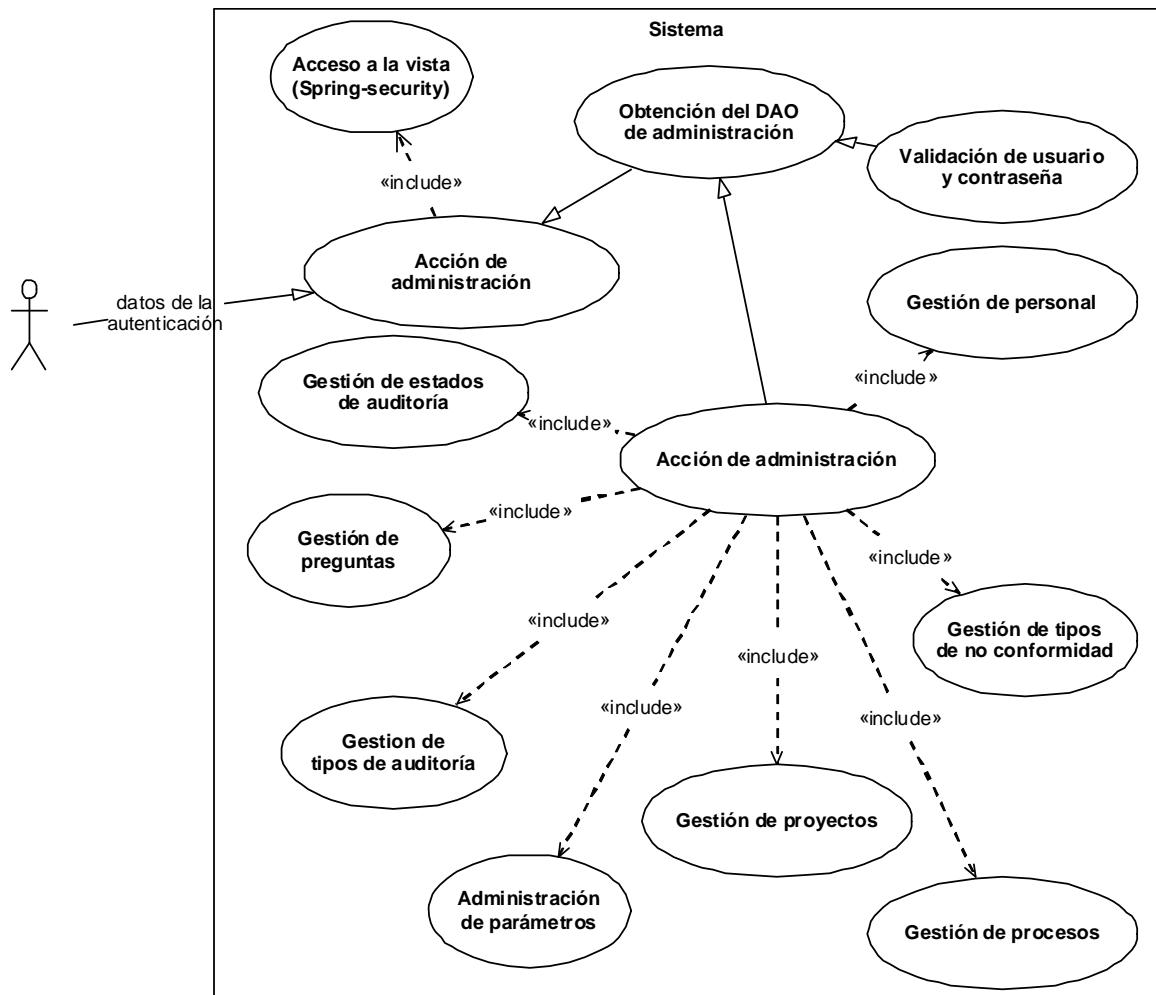


Figura 23 – Caso de uso autenticación

### Gestión de procesos

RF3.- Obtención de los procesos, tanto los datos de baja como los que no lo están.

RF3.- Modificación de un proceso.

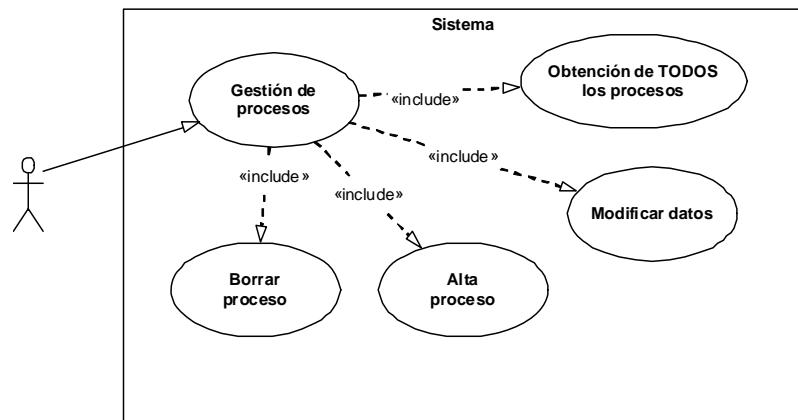


Figura 24 – Caso de uso gestión procesos

#### Gestión de preguntas

- RF4.- Obtención de las preguntas con la posibilidad de borrarlas (de forma lógica) o modificarlas.
- RF4.- modificación de una pregunta.
- RF4.- Obtención de las referencias documentales.
- RF4.- Posibilidad de añadir/quitar referencias documentales a una pregunta.

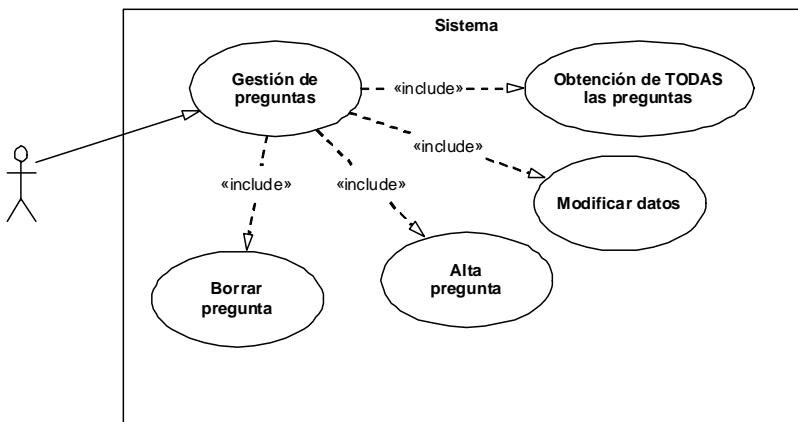


Figura 25 – Caso de uso gestión preguntas

#### Gestión de parámetros

- RF5.- Obtención de cada uno de los parámetros configurables.
- RF5.- Modificación de cada uno de los parámetros configurables.
- RF5.- Mostrar de alguna forma intuitiva la fórmula de cálculo para su modificación.

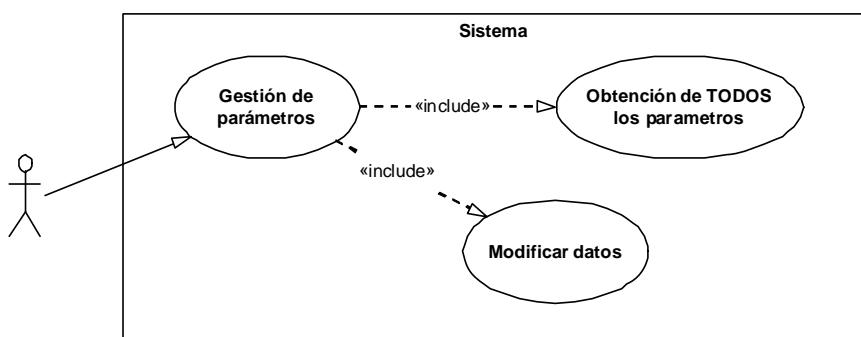


Figura 26 – Caso de uso gestión parámetros

Para poder tener una gran configurabilidad de los parámetros, será indispensable poder gestionar tanto los tipos de no conformidad como los estados de una auditoría/proyecto, de forma que la fórmula de cálculo que depende directamente de ellos, y el cálculo del estado, será completamente dinámico.

## Gestión de tipos de auditoría

RF6.- Obtener los tipos de auditoría.

RF6.- Modificación de los tipos de auditoría.

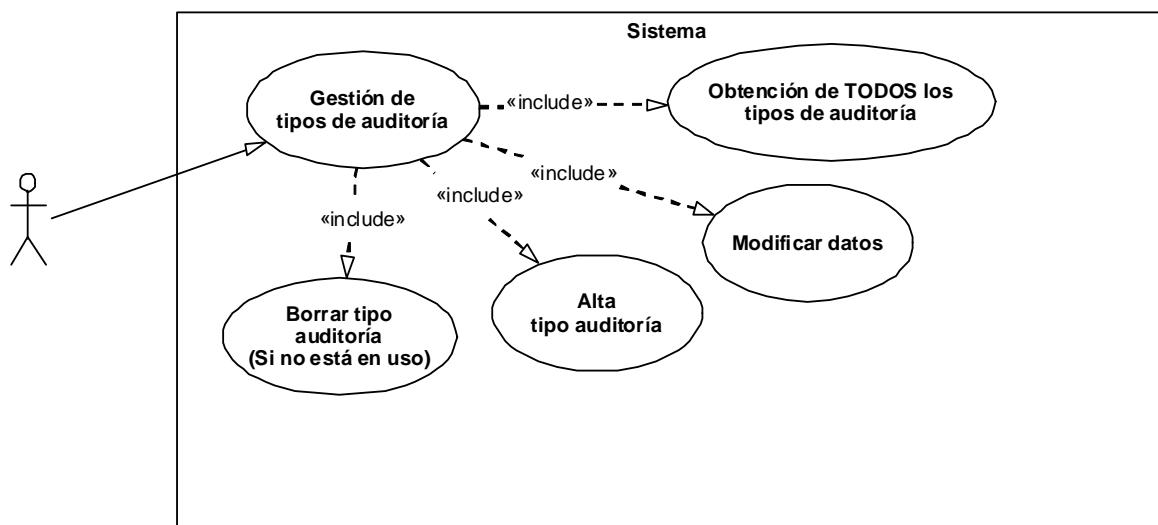


Figura 27 – Caso de uso gestión de tipos

## Arquitectura de software del sistema

Una vez especificados los casos de uso se procede al diseño técnico de la arquitectura del sistema. En la figura siguiente se muestra el Diagrama de Arquitectura de Software correspondiente al Entorno de la nueva aplicación.

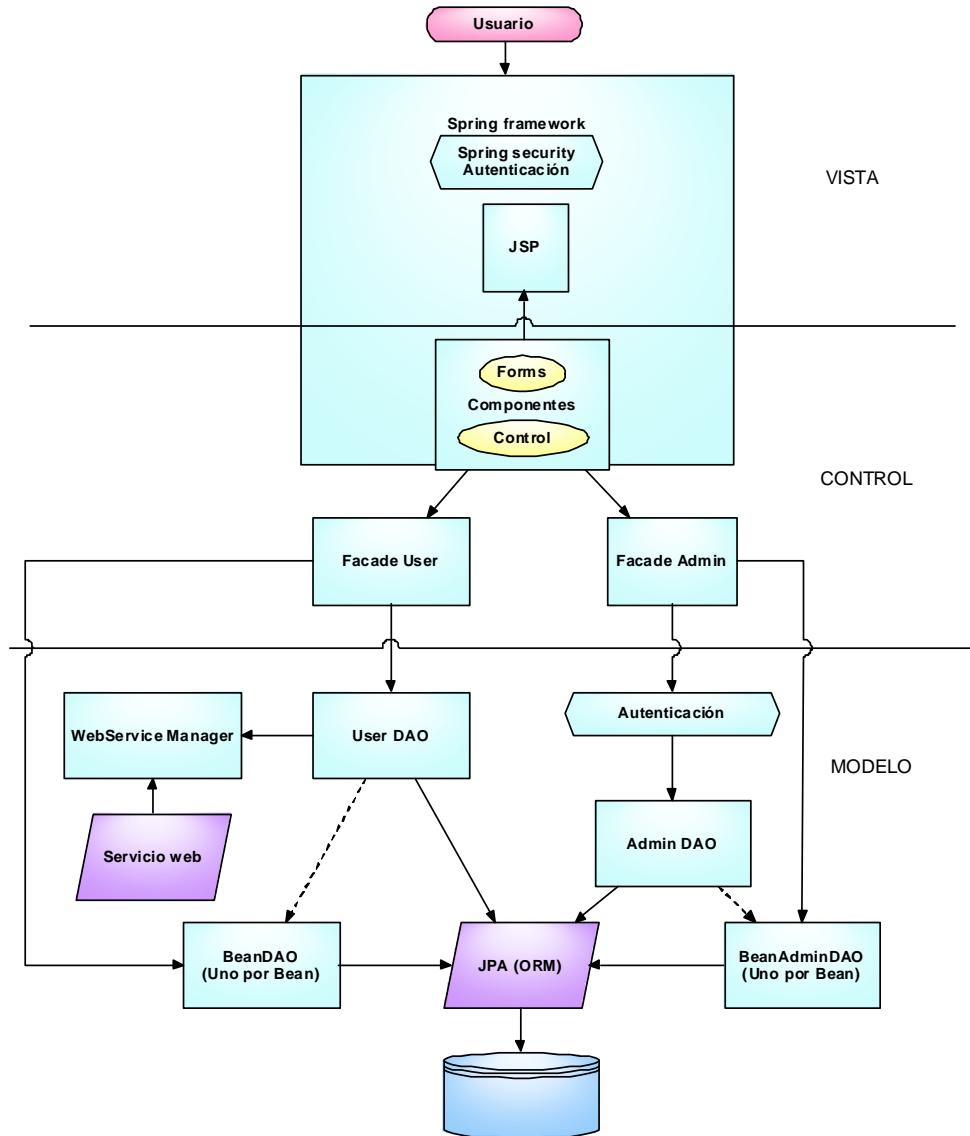


Figura 28 - Diagrama de Componentes Software

Se plantea un Sistema Centralizado que presenta una Arquitectura de tres capas, modelo/vista/controlador dónde se ocultan diversas capas de seguridad.

En las páginas siguientes se profundiza en los elementos más importantes de la arquitectura.

### Capa de Modelo

#### SGBD

La gestión de los repositorios de datos se delegará en un Gestor de Base de Datos reconocido de probada fiabilidad, que cumple al menos los siguientes requisitos:

Compatible SQL.  
Soporte de Conurrencia Cliente/Servidor.  
Soporte de Procedimientos Almacenados.  
Soporte Transaccional.  
Escalabilidad en Almacenamiento.  
Escalabilidad en Proceso.

## ORM

La gestión del mapeo objeto-relacional se delega en manos de JPA, reconocido estándar que cuenta con diversas implementaciones, por lo que resultará bastante fácil y rápido el cambio de proveedor.

## DAO

UserDAO y AdminDAO son dos formas separadas de acceso a datos, mientras que UserDAO no requiere autenticación para acceder a los datos y es de uso público, AdminDAO si lo requiere y necesita expresamente que exista una sesión con los datos de usuario y contraseña (ya sea sin cifrar, o cifrada mediante algún algoritmo de encriptación que deberá ser utilizado también para Spring-security). Las contraseñas se almacenarán en siempre cifradas en la Base de datos.

Las implementaciones de los DAO serán compatibles con JPA, para poder cambiar entre los distintos proveedores, en caso de no ser posible se realizaran dos implementaciones distintas, aquella donde todo sea compatible con JPA y aquella donde se utilicen partes excepcionales de alguna implementación de JPA, de forma que se requerirá poder obtener mediante el patrón Factory diversas implementaciones del DAO global, bien sea AdminDAO o UserDAO.

## Seguridad

Como se puede apreciar en la figura hay dos niveles en los que se pide autenticación, esto es así por si alguien consigue saltarse la primera de ellas, más cercana al usuario.

## Fachada y componentes

Los componentes son diversos controladores que gestionan los datos requeridos por la página solicitada, de forma que acceden a los datos a través de la fachada correspondiente. Estos componentes son llamados por Spring Framework accediendo a la página JSP oculta según el webflow definido en los ficheros xml de configuración de Spring.

### Seguridad

Si se intenta acceder a la fachada de administración cuando no se ha iniciado una sesión valida, no se podrán acceder a los datos.

### Lógica de negocio

La lógica de negocio está implementada en estas acciones o controladores, parte de la funcionalidad también está implementada en la fachada que ofrece diversos métodos más completos.

### Fachadas

Las dos fachadas proveen a los controladores acceso y modificación de datos, de una forma más o menos simplificada. De esta forma se encapsula el acceso a datos de la lógica de negocio y además se separa la parte pública de la privada.

## **Servidor web**

La implementación del software de servicio en el sistema central se basará en la utilización de un Servidor de Aplicaciones. Entre otros, el Servidor de Aplicaciones ofrecerá los siguientes recursos fundamentales:

### **Servidor Web**

Sistema capaz de servir páginas Web de construcción dinámica a petición de los usuarios remotos. Incluye mecanismos estándar para soportar la concurrencia y controlar distintas sesiones de trabajo.

Las páginas Web de construcción dinámica gestionadas por el Servidor Web permitirán desarrollar el interfaz gráfico de los aplicativos Web de la plataforma.

### **Seguridad**

La seguridad de esta capa se delega en manos de Spring-security, dado que se va a usar Spring como Framework para la aplicación web, de forma que existirá un rol admin y otro rol anónimo que no requiera autenticación.

### **Componentes**

Sistema capaz de ejecutar componentes de software. Incluye mecanismos estándar para soportar la concurrencia y controlar distintas sesiones de trabajo.

Los Componentes de Software gestionados por el Contenedor permitirán desarrollar la lógica de los Componentes Base y los Módulos Funcionales de la plataforma.

## Diseño de clases

El siguiente diseño se centra en las clases correspondientes a los Beans, estos son los Transfer Objects entre la base de datos y la capa de control.

Los métodos a implementar son los get/set de los atributos utilizando la nomenclatura estándar, así mismo se incorporan las anotaciones de JPA para que el proveedor de persistencia pueda persistir las clases.

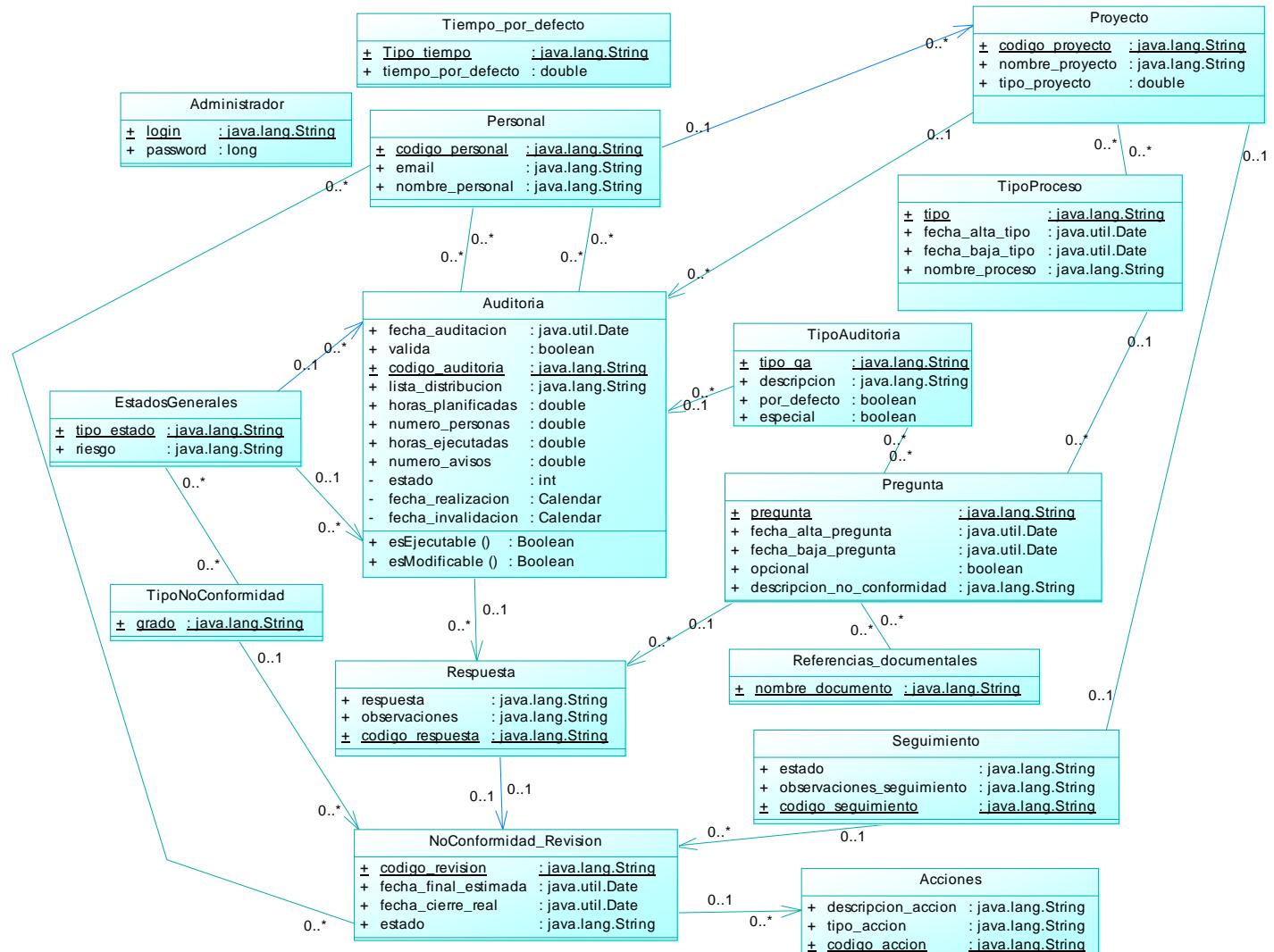


Figura 29 – Diagrama de clases

## Patrones de diseño

Los patrones de diseño son formas recomendables de implementar aspectos comunes a diversas aplicaciones, se ha diseñado para ser fácilmente implementable, reusable y mantenible, de forma que resulten ser una buena solución.

### Patrón DAO

Se utilizarán diversos patrones de diseño a la hora de implementar el sistema, uno de ellos será el patrón DAO para la capa de Modelo de datos, donde se encapsulará la forma de acceso a los datos.

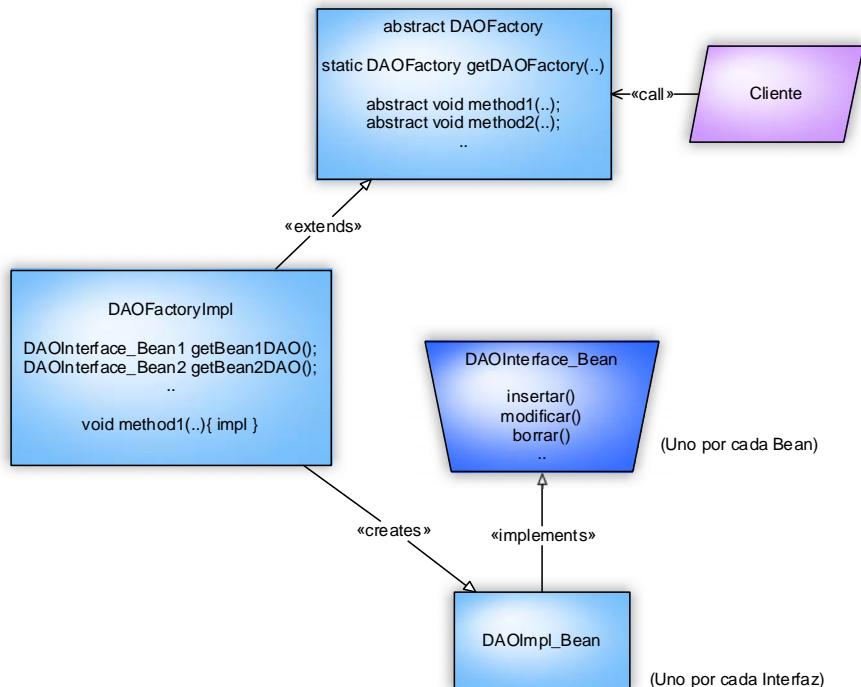


Figura 30 – Patrón DAO

De esta forma, el cliente lo único que conocerá de la capa del modelo, serán los métodos ofrecidos por la clase DAOFactory, sin llegar a conocer la implementación propia de los métodos, incluso podríamos tener varias implementaciones del mismo método para distintas bases de datos o distintas formas de implementarlo (mediante algún ORM, de forma manual con sentencias SQL, etc).

### Patrón Fachada

Este patrón lo hemos utilizado para abstraernos aun más de la jerarquía utilizada para el acceso a datos, debido a que para cada bean diferente, existe un DAO propio para poder actualizar, insertar y borrar el objeto de la BD, mientras que de esta forma una única clase nos provee de todos los métodos necesarios y además se encarga de facilitarnos ciertos métodos algo más complejos que incorporan cierta lógica de negocio.

Además tendremos implementadas dos fachadas diferentes, una para el acceso a datos como usuario y otra para el acceso a datos como administrador, de esta forma encapsulamos en cada fachada distinta funcionalidad y además simplificamos el uso de los DAO.

# ANEXO 6 – Plan de pruebas

## Índice

ANEXO 6 – PLAN DE PRUEBAS .....	66
DESCRIPCION Y OBJETIVOS .....	67
PRUEBAS DE VALIDACION Y ROBUSTEZ.....	67
<i>Pruebas sobre los requisitos funcionales de la parte pública .....</i>	67
<i>Pruebas sobre los requisitos no funcionales de la parte pública .....</i>	70
<i>Pruebas sobre los requisitos funcionales de la parte privada .....</i>	71
PRUEBAS UNITARIAS.....	72
<i>Pruebas sobre el acceso a datos.....</i>	72
<i>Pruebas sobre el acceso al servicio web .....</i>	72
PRUEBAS DE ACEPTACION .....	73
<i>Pruebas de la parte privada .....</i>	73
<i>Pruebas de la parte pública.....</i>	75

## Descripción y objetivos

El objetivo de este documento es elaborar un plan de pruebas general sobre los requisitos del sistema para identificar posibles carencias en el funcionamiento del sistema.

Se definirán los siguientes tipos de pruebas:

Pruebas de validación  
Pruebas unitarias  
Pruebas de aceptación

Por cada tipo de prueba se obtendrá un documento de resultados, donde se describirán los resultados obtenidos en las pruebas, su resolución y el estado final de la prueba.

Las pruebas se realizaran a través de la herramienta Testlink <http://blog.testlink.org/> , instalada en el servidor de la empresa <http://ramses:82/testlink/>

## Pruebas de validación y robustez

Las pruebas de validación corresponden a la verificación del cumplimiento de la funcionalidad recogida en el documento de requisitos obtenidos durante el análisis.

Las pruebas de robustez corresponden a comprobar que el sistema se comporta de manera correcta ante entradas imprevistas.

### Pruebas sobre los requisitos funcionales de la parte pública

Para todos estos apartados hay que comprobar que no se requiera de ninguna contraseña, ni usuario.

<b>REQ 1</b>	El nuevo aplicativo resultará claro y sencillo, y su operativa será lo más amigable, intuitiva y atractiva posible.
PRUEBA	Pruebas de usabilidad
Descripción	Recogidas en el documento de usabilidad.

<b>REQ 2</b>	El sistema tendrá una gestión de avisos por email al personal correspondiente, ya sea auditor o auditado ( 2.1 a 2.6 )
PRUEBA	Pruebas de generación de avisos.
Observaciones	Se realizará después de comprobar que se pueden crear/ejecutar/modificar auditorías y realizar el seguimiento. (punto 3) y (punto 6)
Descripción	<p>Se realizará un test del sistema en el cual se establecerán las tareas periódicas cada 2 minutos, de forma que el test pueda realizarse en un día.</p> <p>Se creará una nueva auditoría, de forma que se enviará el aviso 2.1, se comprobará su llegada, en caso de ser correcto se dará por bueno el punto 2.1</p> <p>Se modificará la fecha de auditación de la auditoría creada en el test, de forma que se enviará el aviso 2.2, se comprobará su llegada al email correspondiente, en caso de ser correcto se dará por bueno el punto 2.2, la fecha que se establecerá será una de no más de los días establecidos por defecto para el punto 2.3 ( días antes de la auditación ), de forma que el aviso del 2.3 se enviará automáticamente en la siguiente ejecución de la tarea periódica.</p> <p>Si hemos realizado bien el paso anterior, podremos comprobar la llegada del punto 2.3, se comprueba la llegada del email y se da por bueno/malo.</p> <p>Se completa la auditoría, de forma que podremos comprobar el punto 2.5, veremos si desde la aplicación se puede acceder al seguimiento.</p>

	<p>Se establecerá el tiempo del punto 2.6 ( tiempo después de completar el seguimiento para recordar su comprobación ) a 0 días, de forma que se enviará en el mismo día.</p> <p>Completamos el seguimiento de la auditoría ejecutada y comprobamos el email del auditor, para comprobar así el punto <b>2.6</b></p> <p>Para comprobar el punto <b>2.4</b> , se establecerán de tiempos :</p> <p>Numero avisos = 2</p> <p>Tiempo entre avisos = 1</p> <p>De forma que tendremos que esperar 3 días con el servidor encendido a la espera de los avisos , para ello creamos una nueva auditoría, esperamos 3 días y comprobamos la llegada de los emails comprobando que las fechas de llegada sean correctas, así como el destinatario, etc.</p> <p>En caso de haber algún error en el proceso se dará como erróneo el punto <b>2</b>, explicando los puntos en los que se falla.</p>
--	---

<b>REQ 3</b>	<p>En la parte pública se permitirá la creación, modificación y ejecución de auditorías, realizadas o bien sobre un proyecto (No especiales) o bien que no requieran proyecto (especiales).</p>
PRUEBA	<p>Pruebas de gestión de auditorías</p>
Descripción	<p>Se harán diversos tipos de pruebas</p> <p><b>Creación</b></p> <p>Se probará la robustez insertando en los diversos campos valores extraños, valores no esperados (enteros en lugar de cadenas, cadenas en lugar de enteros, etc)</p> <p>Se probará a evitar llenar cada uno de los campos</p> <p>Se probará a insertar sentencias SQL en algún campo de tipo cadena</p> <p>Se creará una auditoría con los campos necesarios de forma correcta y se comprobará que la ha creado accediendo a ella a través de modificación.</p> <p>Se comprobará que se permite la búsqueda de personal y de proyectos utilizando el código de recurso.</p> <p>Se comprobará que por defecto vienen marcadas las auditorías que requieren proyecto.</p> <p><b>Modificación</b></p> <p>Se modificará la auditoría creada con valores no esperados en los campos.</p> <p>Se probará a insertar sentencias SQL en algún campo de tipo cadena.</p> <p>Se cambiarán todos los campos de la auditoría a la vez y uno a uno por separado.</p> <p><b>Ejecución</b></p> <p>Se intentará ejecutar la auditoría creada.</p> <p>Se intentará ejecutar una auditoría no existente.</p> <p>Se intentará ejecutar una auditoría que no tenga preguntas.</p> <p>Se intentará ejecutar una auditoría que no pueda ser ejecutada (según su estado) ,se probará una vez por cada estado apuntando cuáles deja que no debería dejar ejecutar. (Solo se pueden ejecutar aquellas auditorías que tienen el estado: Planificada, Caducada ó En curso).</p> <p>Se accederá a todos los botones de forma indiscriminada, se probará que el cálculo de las horas ejecutadas se realice de forma correcta ( utilizando el guardar y continuar más tarde o completando la auditoría en una sola vez.</p> <p>Se comprobará que la navegación está pensada en forma de pestañas y que</p>

	corresponden a los procesos.
--	------------------------------

<b>REQ 4</b>	Desde la parte pública se podrán consultar diversos informes
PRUEBA	Pruebas de informes
Descripción	Se consultará cada informe, de forma que se consulten cuando haya datos para mostrar, cuando no haya datos y cuando falten algunos de los datos, de forma que se tendrán que generar diversos estados en la base de datos (Datos sobre las auditorías), para poder comprobar que los informes son los esperados, se muestran de una forma correcta, con datos correctos.

<b>REQ 5</b>	Desde la parte pública se podrá acceder a diversas métricas calculadas automáticamente.
PRUEBA	Pruebas de métricas
Descripción	Como en la prueba de informes, se consultarán las diversas métricas comprobando para cada estado de la base de datos ( Con datos, sin datos, carente de diversos datos) cada una de las métricas, comprobando manualmente que se calculan bien.

<b>REQ 6</b>	Desde la parte pública se podrán realizar los Seguimientos de no conformidades (Auditor), donde se contemplaran todas las no conformidades de las auditorías de un proyecto ó únicamente de la auditoría seleccionada, con la información relevante a cada no conformidad así como la posibilidad de añadir acciones correctivas / cierre, modificar el estado de la revisión, seleccionar al personal responsable, etc.
PRUEBA	Pruebas de seguimiento
Descripción	<p>Comprobar que se puede acceder al seguimiento, donde deben aparecer las no conformidades de una auditoría ejecutada que hayamos seleccionado.</p> <p>Probar para auditorías no existentes, no ejecutadas, invalidadas.</p> <p>Probar toda la funcionalidad del seguimiento, modificar fechas, acciones correctivas, acciones de cierre, estado de la revisión, abrir/cerrar revisiones, seleccionar responsables, etc.</p> <p>Comprobar que hay alguna forma de acceder al seguimiento sin poder modificarlo ( para los auditados ).</p>

<b>REQ 7</b>	Desde la parte pública se podrán modificar las fechas de la auditoría lo cual corresponde a su planificación
PRUEBA	Pruebas de gestión de auditorías.
Descripción	Recogido en la prueba del <b>REQ 3</b> , en modificación.

<b>REQ 8</b>	Antes de la ejecución de la auditoría se mostrará una agenda de la misma, mostrando la información relevante a la auditoría.
PRUEBA	Prueba de agenda.
Descripción	Comprobar que se muestra la agenda y se puede acceder a ella, validar la agenda comparándola con el fichero Excel suministrado del procedimiento manual anterior, que corresponde a la Agenda.

## Pruebas sobre los requisitos funcionales del sistema

<b>REQ NF 1</b>	método de obtención de preguntas
PRUEBA	Prueba de obtención de preguntas
Descripción	Comprobar que se obtienen bien las preguntas de una auditoría a la hora de ejecutarla.

<b>REQ NF 2</b>	Las respuestas de una auditoría pueden ser: Si, No y No/Aplica, y podrán tener observaciones.
PRUEBA	Prueba de cuestionario
Descripción	Comprobar que las respuestas posibles son Si, No y No/Aplica, donde se pueden acompañar las respuestas con un campo de observaciones.

<b>REQ NF 4</b>	Una revisión de no conformidad puede tener uno de los siguientes estados: Pendiente, En curso, Invalidada, Cerrada
PRUEBA	Prueba de revisiones
Descripción	Comprobar que existen estos cuatro estados y se pueden seleccionar correctamente a la hora de realizar el seguimiento.

<b>REQ NF 5</b>	Las preguntas serán históricas
PRUEBA	Prueba de preguntas históricas
Descripción	Eliminar o modificar una de las preguntas de alguna auditoría ejecutada, visualizar la auditoría y comprobar que se muestra la pregunta anterior a la modificación, crear una nueva auditoría idéntica a la anterior (mismo tipo sobre mismo proyecto) y comprobar que la pregunta mostrada es la modificada.

<b>REQ NF 6</b>	Los procesos serán históricos
PRUEBA	Prueba de procesos
Descripción	Eliminar algún proceso que esté en alguna auditoría ejecutada.  Comprobar que aparece dicho proceso en la visualización de la auditoría, crear una auditoría idéntica y comprobar que ese proceso ya no está en el cuestionario.  Modificar otro de los procesos cambiando el nombre y comprobar que sigue saliendo en ambas auditorías.

<b>REQ NF 7</b>	El sistema debe caducar auditorías que están planificadas para fechas anteriores.
PRUEBA	Prueba de caducación
Descripción	Programar una auditoría para una fecha pasada, esperar a que se ejecute el gestor de tareas periódicas.

## Pruebas sobre los requisitos funcionales de la parte privada

<b>REQ 1</b>	Se accederá a cualquier parte de la administración mediante login/password, manteniendo en lo posible la seguridad
PRUEBA	Prueba de seguridad
Descripción	<p>Intentar entrar sin utilizar usuario y contraseña a la parte de administración.</p> <p>Eliminar la seguridad de acceso a páginas de administración (Permitir el acceso a la página) y comprobar que no se pueda modificar nada sin estar realmente autenticado.</p> <p>De esta forma comprobamos que la autenticación es necesaria pese a que alguien consiga saltársela.</p>

<b>REQ 2</b>	Se podrá gestionar al personal, buscando por código de recurso a través del Servicio Web.
PRUEBA	Prueba de gestión personal
Descripción	<p>Probar la gestión del personal, con datos inesperados por el sistema en cada campo.</p> <p>Probar a buscar a través del servicio web a personas que se encuentren en la base de datos, que no se encuentren, que tengan email, no tengan email, que tengan nombre y que no tengan nombre.</p>

<b>REQ 3</b>	Se permitirá gestionar los procesos a auditar, pueden sufrir cambios.
PRUEBA	Prueba de gestión procesos
Descripción	<p>Probar la gestión de los procesos, con datos inesperados por el sistema en cada campo.</p> <p>Probar a crear un nuevo proceso, comprobando que aparecen en la gestión de preguntas.</p> <p>Probar que se modifican correctamente, el borrado ya se prueba en el requisito no funcional ( procesos históricos <b>REQ NF 6</b> )</p>

<b>REQ 4</b>	Se facilitará la gestión de las preguntas existentes, pudiendo cambiar ó eliminar preguntas.
PRUEBA	Prueba de gestión preguntas
Descripción	<p>Probar la gestión de las preguntas, con datos inesperados por el sistema en cada campo.</p> <p>Probar que se modifican correctamente, el borrado ya se prueba en el requisito no funcional ( preguntas históricas <b>REQ NF 5</b> )</p>

<b>REQ 5</b>	Todos los parámetros configurables numéricos serán configurables desde la administración, a la que se accederá mediante un login/password almacenado en la propia base de datos.
PRUEBA	Prueba de gestión general
Descripción	Probar la gestión general, modificar cada uno de los campos por separado comprobando que se muestran todos los parámetros configurables (Número de días para cada aviso, número de avisos, etc.)

<b>REQ 6</b>	Se podrán gestionar los tipos de auditoría existente permitiendo validar o invalidar los existentes y configurando si es especial o no.
PRUEBA	Prueba de gestión tipos de auditoría
Descripción	Probar la gestión de tipos de auditoría, insertando, modificando los existentes. Comprobar que aparecen a la hora de crear auditorías, gestionar preguntas.

## Pruebas unitarias

### Pruebas sobre el acceso a datos

Se realizaran para cada uno de los métodos de acceso a datos, tanto del usuario normal como de la administración.

Para ello se realizaran diversos test donde se llamen a cada uno de estos métodos, escribiendo en fichero/pantalla los resultados y comprobando que lo obtenido corresponde a lo que el método debe devolver.

### Pruebas sobre el acceso al servicio web

Se realizara un test para cada servicio web utilizado, comprobando que se ejecuten correctamente en cada caso y para todos los casos posibles en la respuesta.

Para ello se realizaran diversos test donde se llamen a cada uno de estos métodos, escribiendo en fichero/pantalla los resultados y comprobando que lo obtenido corresponde a lo que el método debe devolver.

## Pruebas de aceptación

Estas pruebas tienen como objetivo comprobar que el sistema cumple con los requisitos planteados siendo el cliente quien prueba el producto software, de esta forma se plantean diversos apartados a cumplimentar por el cliente anotando una serie de observaciones.

Aprovecharemos estas pruebas para integrarlas con las de usabilidad dado que es el cliente quien realiza ambas "encuestas".

### Pruebas de la parte privada

En este apartado se probaran las distintas funcionalidades requeridas para la parte de administración del sistema, cada una de las pruebas consistirá en ejercicios habituales durante el uso del software.

#### Administración y puesta a punto.

Entrar en la administración como:

usuario: admin  
contraseña: admin.

Modificar los tiempos establecidos para los avisos (número de días antes de realizar un aviso, número de avisos, etc) y definir en una hoja el significado entendido en cada uno de ellos.

Plantear en cada caso una descripción entendible.

Creación de los tipos de auditoría básicos preexistentes en el sistema manual.

QA1  
QA2  
QA3  
PPQA  
QA4

Eliminación de la QA4

Creación de los procesos de cada tipo de auditoría

Creación de los procesos de la QA1  
Gestión de Riesgos  
Requisitos  
DAR  
Solución Técnica  
"Procesotes"

Eliminación del proceso "Procesotes"

Creación de una batería de preguntas para cada proceso (2 por proceso en 2 procesos)

Solución Técnica

Pregunta 1: ¿Se ha delimitado el alcance del sistema?  
Opcional: No  
Tipo de auditorías: QA1  
Referencias documentales: "Doc. Análisis"

Pregunta 2: ¿Se ha desarrollado la especificación mediante casos de uso del sistema? (Solo proyectos Tipo II y III)  
Opcional: Si  
Tipo de auditorías: QA1  
Referencias documentales: "Doc. Análisis"

DAR

Pregunta 3: ¿Se ha efectuado una toma de decisiones formal en el proyecto?

Opcional: No

Tipo de auditorías: Todas

Referencias documentales: Análisis de decisiones, Proceso DAR

Pregunta 4: ¿Se han estudiado las distintas alternativas?

Opcional: No

Tipo de auditorías: Todas

Referencias documentales: Ninguna

Creación de una pregunta TEST y posterior eliminación.

Creación de tipos de no conformidades

Generación de los tipos de no conformidades básicos pre-existentes

Menor

Mayor

Creación de tipos de estados/riesgos del proyecto/auditoría

Generación de los tipos de estado básicos pre-existentes

Bueno

Regular

Malo

Definición de la forma de calcular el estado de un proyecto a partir del número de no conformidades de cada tipo de no conformidad.

Generación de la formula pre-existente.

Tipo de no conformidad MAYOR

Con hasta 0(incluido) no conformidades se esta en el estado BUENO

Entre 1(incluido) y 2(incluido) no conformidades se esta en estado REGULAR

Con más de 3 no conformidades se esta en estado MALO

Tipo de no conformidad MENOR

Con hasta 4(incluido) no conformidades se esta en estado BUENO

Entre 5(incluido) y 9(incluido) no conformidades se esta en estado REGULAR

Con más de 10 no conformidades se esta en estado MALO

Gestionar al personal, buscar a dos personas cualesquiera mediante el servicio web. Se buscan por código de recurso, si no se conoce ninguno, probar con A0XX donde XX es un número, por ejemplo A020.

Gestionar los proyectos e intentar averiguar el significado de "auditabile" sin mirar en las pruebas posteriores.

Mantener al menos un proyecto auditabile.

Escribir en observaciones el significado de auditabile.

Crearse una cuenta propia para acceder a la administración, probarla.

## **Pruebas de la parte pública**

### **Proceso general de auditación**

Realización de una pequeña prueba del proceso de auditación general.

Crear una nueva auditoría para un proyecto determinado, crear (QA1, QA2 y QA3) (Apuntar el código de proyecto para luego).

Planificar la auditoría para una fecha cualquiera. (La QA1 que tendrá preguntas)

Ejecutar dicha auditoría (Contestar negativamente a alguna, para que haya seguimiento)

Acceder al seguimiento dónde pueda modificarse, llenar las no conformidades.

Acceder al seguimiento dónde solo pueda ser visualizado el seguimiento.

Visualizar las respuestas de la auditoría recién ejecutada.

Visualizar el resumen de la auditoría recién ejecutada (mostrado también al terminar la ejecución de la auditoría), pero accediendo por otro sitio ya que no se puede RE-ejecutar una auditoría.

Acceder al informe donde se muestran todas las auditorías (Informe general de auditorías).

Acceder al informe general de auditorías donde se pueda filtrar por proyecto (Elegir el mismo que se eligió al crear la auditoría).

Acceder a las distintas métricas y comprobar el funcionamiento.

# ANEXO 7 – Usabilidad

## Índice

<b>ANEXO 7 – USABILIDAD .....</b>	<b>76</b>
DESCRIPCION Y OBJETIVOS .....	77
<i>Pruebas con usuario inexperto .....</i>	<i>77</i>
<i>Pruebas con usuario experto.....</i>	<i>77</i>

## **Descripción y objetivos**

El objetivo de este documento es elaborar un plan para comprobar la usabilidad del sistema.

Se definirán los siguientes tipos de pruebas:

Pruebas con usuario inexperto.

Pruebas con usuario experto.

Los resultados de las pruebas iran en el documento de Resultado de pruebas, dado que se realizan de forma conjunta a las pruebas de aceptación, este documento será un documento de tipo encuesta que será rellenada por aquel que lleve a cabo las pruebas, se adjuntara una sección donde se den explicaciones de las medidas a tomar y de las finalmente llevadas a cabo.

### **Pruebas con usuario inexperto**

Se realizaran de forma conjunta a las pruebas de aceptación, donde para cada apartado se recogerá la siguiente información por parte del usuario:

Puntuaciones de 1 a 5, donde 5 = bueno (poco tiempo), 1 = malo (mucho tiempo).

Tiempo requerido

Facilidad de manejo

Facilidad de aprendizaje

Agradable a la vista

Texto a introducir por el usuario:

Observaciones

Qué cambiarias

### **Pruebas con usuario experto**

Se ejecutarán las pruebas de aceptación por parte de un usuario experto, observando en cada caso el tiempo requerido minuciosamente, si alguna cosa no se aprendió y por qué, etc, se tomara más énfasis en el campo observaciones.

# ANEXO 8 – Resultado de las pruebas

## Índice

ANEXO 8 – RESULTADO DE LAS PRUEBAS .....	78
DESCRIPCION Y OBJETIVOS .....	79
RESULTADOS DE LAS PRUEBAS DE VALIDACION Y ROBUSTEZ .....	79
<i>Pruebas sobre los requisitos funcionales de la parte pública</i> .....	79
<i>Resultados de las pruebas sobre los requisitos no funcionales de la parte pública</i> .....	81
<i>Resultados de las pruebas sobre los requisitos funcionales de la parte privada</i> .....	81
RESULTADOS DE LAS PRUEBAS UNITARIAS .....	82
<i>Resultados de las pruebas sobre el acceso a datos</i> .....	82
RESULTADOS DE LAS PRUEBAS DE ACEPTACION Y USABILIDAD .....	84
<i>Resultados de las pruebas de la parte privada</i> .....	84
<i>Pruebas de la parte pública</i> .....	87

## Descripción y objetivos

En este documento se recogerán los resultados de las distintas pruebas realizadas, así como las medidas adoptadas para la resolución de los errores detectados.

En este documento se redactan también los resultados obtenidos del plan de usabilidad, dado que las pruebas que se realizan van en conjunto con las pruebas de aceptación.

## Resultados de las pruebas de validación y robustez

### Pruebas sobre los requisitos de la parte pública

<b>REQ NF 1</b>	El nuevo aplicativo resultará claro y sencillo, y su operativa será lo más amigable, intuitiva y atractiva posible.
PRUEBA	Pruebas de usabilidad
Resultado	<b>Recogidas en el punto 4.</b>

<b>REQ 2</b>	El sistema tendrá una gestión de avisos por email al personal correspondiente, ya sea auditor o auditado ( 2.1 a 2.6 )
PRUEBA	Pruebas de generación de avisos.
Resultado	2.1 OK. 2.2 OK. 2.3 OK. 2.4 OK. 2.5 Desde la aplicación. OK 2.6 OK.  Avisos de visualización: No se enviaban, error al enviar a múltiples. <b>Solucionado</b>

<b>REQ 3</b>	En la parte pública se permitirá la creación, modificación y ejecución de auditorías, realizadas o bien sobre un proyecto (No especiales) o bien que no requieran proyecto (especiales).
PRUEBA	Pruebas de gestión de auditorías
Resultado	<b>Creación</b> <b>Errores encontrados:</b> Al mostrarse los errores de auditores/auditados, se descuadra el aspecto.  Errores en los listados de Auditores y Auditados, a veces los muestra otras no. <b>Solucionado</b> Si no se selecciona ninguno no muestra el error. <b>Solucionado</b> Los tipos de auditoría no dan errores al no seleccionar ninguno. <b>Solucionado</b> <b>Modificación</b>  No se muestran errores al insertar una lista de distribución incorrecta. <b>Solucionado</b> No se muestran errores al insertar una lista de distribución vacía. <b>Solucionado</b> <b>Ejecución</b> Si no se rellenan las observaciones de una respuesta Negativa ó No/Aplica, no se guarda en la ejecución.

<b>REQ 4</b>	Desde la parte pública se podrán consultar diversos informes
PRUEBA	Pruebas de informes
Resultado	<p><b>Informe filtrado</b></p> <p>La selección del jefe de proyecto debería ser con sugerencias, al igual que se hace para nueva auditoría.</p> <p><b>Solucionado</b></p>

<b>REQ 5</b>	Desde la parte pública se podrá acceder a diversas métricas calculadas automáticamente.
PRUEBA	Pruebas de métricas
Resultado	<p><b>Proyectos caducados</b></p> <p>El botón “Ver proyectos caducados” no tiene un aspecto similar al resto de la Web.</p> <p><b>Proyectos auditados</b></p> <p>El botón “Ver proyectos auditados” no tiene un aspecto similar al de la Web.</p> <p><i>Intervalo (meses)</i> no se sabe a que se refiere sin usarlo antes.</p>

<b>REQ 6</b>	Desde la parte pública se podrán realizar los Seguimientos de no conformidades (Auditor), donde se contemplaran todas las no conformidades de las auditorías de un proyecto ó únicamente de la auditoría seleccionada, con la información relevante a cada no conformidad así como la posibilidad de añadir acciones correctivas / cierre, modificar el estado de la revisión, seleccionar al personal responsable, etc.
PRUEBA	Pruebas de seguimiento
Resultado	<p><b>Seguimiento auditoría</b></p> <p>Resulta bastante incómodo manejarse cuando la barra se desplaza en cada submit.</p> <p>Se visualiza mal.</p> <p><b>Seguimiento proyecto</b></p> <p>Las acciones se visualizan mal</p>

<b>REQ 7</b>	Desde la parte pública se podrán modificar las fechas de la auditoría lo cual corresponde a su planificación
PRUEBA	Pruebas de gestión de auditorías.
Resultado	OK

<b>REQ 8</b>	Antes de la ejecución de la auditoría se mostrará una agenda de la misma, mostrando la información relevante a la auditoría.
PRUEBA	Prueba de agenda.
Resultado	El texto se sale fuera si es muy largo en alguno de sus campos.

## Resultados de las pruebas sobre los requisitos del sistema

<b>REQ 1</b>	Método de obtención de preguntas
PRUEBA	Prueba de obtención de preguntas
Resultado	OK

<b>REQ 2</b>	Las respuestas de una auditoría pueden ser: Si, No y No/Aplica, y podrán tener observaciones.
PRUEBA	Prueba de cuestionario
Resultado	OK

<b>REQ 4</b>	Una revisión de no conformidad puede tener uno de los siguientes estados: Pendiente, En curso, Invalidada, Cerrada
PRUEBA	Prueba de revisiones
Resultado	OK

<b>REQ 5</b>	Las preguntas serán históricas
PRUEBA	Prueba de preguntas históricas
Resultado	OK

<b>REQ 6</b>	Los procesos serán históricos
PRUEBA	Prueba de procesos
Resultado	OK

<b>REQ 7</b>	El sistema debe caducar auditorías que están planificadas para fechas anteriores.
PRUEBA	Prueba de caducación
Resultado	Error, no se actualiza la auditoría. <b>Solucionado</b>

## Resultados de las pruebas sobre los requisitos de la parte privada

<b>REQ NF 1</b>	Se accederá a cualquier parte de la administración mediante Lorin/password, manteniendo en lo posible la seguridad
PRUEBA	Prueba de seguridad
Resultado	OK, al entrar en las diversas páginas sin estar realmente autenticado nos devuelve resultados nulos.

<b>REQ 2</b>	Se podrá gestionar al personal, buscando por código de recurso a través del Servicio Web.
PRUEBA	Prueba de gestión personal
Resultado	OK

<b>REQ 3</b>	Se permitirá gestionar los procesos a auditar, pueden sufrir cambios.
PRUEBA	Prueba de gestión procesos
Resultado	OK

<b>REQ 4</b>	Se facilitará la gestión de las preguntas existentes, pudiendo cambiar ó eliminar preguntas.
PRUEBA	Prueba de gestión preguntas
Resultado	<p>La opcionalidad de las preguntas es más cómoda con checkbox.  <b>Solucionado</b></p> <p>Al añadir nueva pregunta, si añades referencias documentales, se pierde la información guardada en la selección de Proceso y Tipos de auditoría.  <b>Solucionado</b></p>

<b>REQ 5</b>	Todos los parámetros configurables numéricos serán configurables desde la administración, a la que se accederá mediante un Lorin/password almacenado en la propia base de datos.
PRUEBA	Prueba de gestión general
Resultado	<p>Al insertar un valor no numérico en la fórmula de cálculo, da una excepción en el servidor.</p> <p><b>Semi-solucionado:</b> A través de javascript validamos el formulario en el cliente, en el servidor el error lo da el Spring-binding.</p> <p>El servidor da la excepción pero nos devuelve a un estado conocido y seguro.</p>

<b>REQ 6</b>	Se podrán gestionar los tipos de auditoría existente permitiendo validar o invalidar los existentes y configurando si es especial o no.
PRUEBA	Prueba de gestión tipos de auditoría
Resultado	La gestión de requiere proyecto y validez, es más fácil de usar con checkboxes.

## Resultados de las pruebas unitarias

La mayoría de los errores de las pruebas unitarias se resuelven en el momento, la mayoría de ocasiones no se apuntan los resultados obtenidos.

## Resultados de las pruebas sobre el acceso a datos

Errores encontrados:

En el método: **public long** getNúmeroAuditorías(*fecha\_ini, fecha\_fin, estadoAuditoría* ) , se ha encontrado un error para el estado de auditoría "CREACIÓN"

**Solucionado:** El error esta solucionado, faltaba añadir uno de los parámetros solicitados en la consulta.

Incongruencias en nombres: Algunos nombres de parámetros no eran suficientemente claros.

Los casos han sido:

getAuditorías(int tipo)

Tipo se refiere al tipo de proyecto

getAuditorías(Personal persona, int tipo)

Persona se refiere al jefe de proyecto y tipo se refiere al tipo de proyecto.

**Solucionado:** Hemos cambiado “tipo” por tipoProyecto y persona por jefeProyecto.

## Resultados de las pruebas de aceptación y usabilidad

### Resultados de las pruebas de la parte privada

#### Administración y puesta a punto.

##### Entrar en la administración

<http://192.168.1.125:8080/gauditint-global/>

Usuario: admin  
Contraseña: admin.

1 = malo / mucho tiempo.

5 = bueno / poco tiempo.

**Modificar los tiempos establecidos para los avisos** (número de días antes de realizar un aviso, número de avisos, etc.) y anotar aquellos que no se entendió bien su significado.

Operación: Creación auditoría	
Tiempo requerido	5
Facilidad de manejo	5
Facilidad de aprendizaje	5
Observaciones	
Qué cambiarias	

#### Creación de los tipos de auditoría básicos

Crea varios tipos de auditoría, entre ellos tres que SI requieren proyecto

Nombre	Descripción
QA1	Tras el análisis y diseño
QA2	Tras la fase de desarrollo
QA3	Tras la fase de implantación

Uno de ellos que NO requiera proyecto

PPQA Auditoría de la gestión de auditorías

Crea otro tipo más y desactívalo.

QA4 - Descripción

Operación: Creación auditoría	
Tiempo requerido	3
Facilidad de manejo	5
Facilidad de aprendizaje	5
Observaciones	Ha costado entender que el tipo de auditoría corresponde a la administración, se buscaba en el menú de "AUDITORÍA" en lugar en el de ADMINISTRACIÓN
Qué cambiarias	

#### Creación de los procesos

Crea cada uno de estos procesos

Gestión de Riesgos

Requisitos

DAR

Solución Técnica

"Prueba"

Elimina el proceso "Prueba"

<b>Operación:</b> Gestión de los procesos	
Tiempo requerido	5
Facilidad de manejo	5
Facilidad de aprendizaje	5
Observaciones	OK
Qué cambiarias	

Creación de una serie de preguntas para cada proceso (2 por proceso en 2 procesos)

En el proceso: **Solución Técnica** crear estas dos preguntas:

**Pregunta 1:** ¿Se ha delimitado el alcance del sistema?

Descripción no conformidad: No se ha delimitado el alcance del sistema

Opcional: No

Tipo de auditorías: QA1

Referencias documentales: "Doc. Análisis"

**Pregunta 2:** ¿Se ha desarrollado la especificación mediante casos de uso del sistema?

Opcional: Si

Tipo de auditorías: QA1

Referencias documentales: "Doc. Análisis"

En el proceso: **DAR** crear las estas dos preguntas:

**Pregunta 3:** ¿Se ha efectuado una toma de decisiones formal en el proyecto?

Opcional: No

Tipo de auditorías: Todas

Referencias documentales: Análisis de decisiones, Proceso DAR

**Pregunta 4:** ¿Se han estudiado las distintas alternativas?

Opcional: No

Tipo de auditorías: Todas

Referencias documentales: Ninguna

Creación de una pregunta de **Test** inventada y posterior eliminación.

<b>Operación:</b> Gestión de las preguntas	
Tiempo requerido	5
Facilidad de manejo	3
Facilidad de aprendizaje	4
Observaciones	La referencia documental no funciona bien, al meter repetidos.
Qué cambiarias	

Creación de tipos de no conformidades

Generación de los tipos de no conformidades básicos

Menor

Mayor

<b>Operación:</b> Gestión de los tipos de no conformidades	
Tiempo requerido	4
Facilidad de manejo	2
Facilidad de aprendizaje	4

Observaciones	Ha costado entender que los tipos de no conformidades son los de tipos de no conformidades, debido a que existe un campo Descripción que no coincide con lo habitual.
Qué cambiarias	Eliminar el campo descripción y dejar simplemente el campo Nombre.

Creación de tipos de estados/riesgos del proyecto/auditoría

Generación de los tipos de estado básicos pre-existentes

Bueno ( riesgo Bajo)

Regular (riesgo Medio)

Malo (riesgo Alto)

Operación: Gestión de los tipos de estados	
Tiempo requerido	5
Facilidad de manejo	5
Facilidad de aprendizaje	5
Observaciones	Al igual que en tipo de no conformidad, no se entiende el atributo "descripción". Fallo al cambiar la descripción por una cadena muy larga. Cree que no se guarda pero lo que pasa es que falla al guardar algo tan grande.
Qué cambiarias	

Definición de la forma de calcular el estado de un proyecto a partir del número de no conformidades de cada tipo de no conformidad.

Generación de la formula de calculo siguiente.

#### **Tipo de no conformidad MAYOR**

Con hasta 0(incluido) no conformidades se esta en el estado BUENO

Entre 1(incluido) y 2(incluido) no conformidades se esta en estado REGULAR

Con más de 3 no conformidades se esta en estado MALO

#### **Tipo de no conformidad MENOR**

Con hasta 4(incluido) no conformidades se esta en estado BUENO

Entre 5(incluido) y 9(incluido) no conformidades se esta en estado REGULAR

Con más de 10 no conformidades se esta en estado MALO

Operación: Gestión de los fórmula de cálculo	
Tiempo requerido	5
Facilidad de manejo	1
Facilidad de aprendizaje	3
Observaciones	No se entiende la fórmula de cálculo, el significado de la modificación.
Qué cambiarias	Añadir alguna explicación, con una explicación externa se ha entendido el significado.

Gestionar al personal, buscar a dos personas cualesquiera mediante el servicio web. Se buscan por código de recurso, si no se conoce ninguno, probar con A0XX donde XX es un número, por ejemplo A020.

Gestionar los proyectos. Mantener al menos un proyecto auditabile.

Operación: Gestión del personal / proyectos y servicio Web.	
Tiempo requerido	5
Facilidad de manejo	5
Facilidad de aprendizaje	5
Observaciones	
Qué cambiarias	

Crear una cuenta propia para acceder a la administración y pruébala.

<b>Operación:</b> Gestión de la administración	
Tiempo requerido	5
Facilidad de manejo	5
Facilidad de aprendizaje	5
Agradable a la vista	5
Observaciones	
Qué cambiarias	

## Pruebas de la parte pública

### Proceso general de auditación

Realización de una pequeña prueba del proceso de auditación general.

Crear una nueva auditoría para un proyecto determinado, crear (QA1, QA2 y QA3) (Apuntar el código de proyecto para luego).

#### Código de proyecto:

<b>Operación:</b> Creación auditoría	
Tiempo requerido	4
Facilidad de manejo	4
Facilidad de aprendizaje	4
	Al seleccionar personal auditado y auditador, no se ha conseguido seleccionar a varios.
Observaciones	Utilizando la opción Cntrl + click se seleccionan varios. Tampoco se ha utilizado el input con sugerencias.
Qué cambiarias	

Planificar la auditoría para una fecha cualquiera. (La QA1 que tendrá preguntas)

<b>Operación:</b> Modificar auditoría	
Tiempo requerido	5
Facilidad de manejo	5
Facilidad de aprendizaje	5
Observaciones	
Qué cambiarias	

Ejecutar dicha auditoría (Contestar negativamente a alguna, para que haya seguimiento)

<b>Operación:</b> Ejecutar auditoría	
Tiempo requerido	5
Facilidad de manejo	3

Facilidad de aprendizaje	5
Observaciones	Incomodidad de no poder volver a atrás una vez has rellenado el cuestionario, para por ejemplo poder modificar observaciones o cambiar una respuesta.
Qué cambiarias	Añadir botón Terminar, y dejar el resto con una funcionalidad de navegación sobre el cuestionario.

Acceder al seguimiento dónde pueda modificarse, llenar las no conformidades con datos inventados.

<b>Operación:</b> Seguimientos	
Tiempo requerido	4
Facilidad de manejo	3
Facilidad de aprendizaje	5
Observaciones	<p>Resulta confusa la forma de selección de personal responsable.</p> <p>El botón guardar se esconde al mover la barra.</p> <p>Al darle a cerrar la revisión, no se guardan los cambios realizados. (Lo mismo ocurre al abrir)</p> <p>No se entiende la diferencia entre Fecha final y Fecha de cierre. (La fecha final es la estimada).</p>
Qué cambiarias	Modificaría la forma de seleccionar al personal responsable por una forma similar a la de las acciones correctivas/de cierre.

Visualizar las respuestas de la auditoría recién ejecutada.

<b>Operación:</b> Visualización	
Tiempo requerido	5
Facilidad de manejo	5
Facilidad de aprendizaje	5
Observaciones	

Visualizar el resumen de la auditoría recién ejecutada (mostrado también al terminar la ejecución de la auditoría), pero accediendo por otro sitio ya que no se puede RE-ejecutar una auditoría.

<b>Operación:</b> Resumen de auditoría	
Tiempo requerido	5
Facilidad de manejo	5
Facilidad de aprendizaje	5
Observaciones	
Qué cambiarias	

Acceder al informe general de auditorías donde se pueda filtrar por proyecto (Elegir el mismo que se eligió al crear la auditoría).

<b>Operación:</b> Informe general	
Tiempo requerido	5
Facilidad de manejo	5
Facilidad de aprendizaje	5
Observaciones	

Qué cambiarias
----------------

Acceder a las distintas métricas y comprobar el funcionamiento.

Operación: Métricas	
Tiempo requerido	5
Facilidad de manejo	5
Facilidad de aprendizaje	5
Observaciones	
Qué cambiarias	

# ANEXO 9 – Documentación

## Índice

ÁMBITO DEL DOCUMENTO .....	3
HERRAMIENTAS Y TECNOLOGIAS .....	4
<i>Herramientas</i> .....	4
<i>Tecnologías</i> .....	4
FASES DEL PROYECTO .....	6
PLANIFICACION DEL PROYECTO.....	7
<i>Hitos</i> .....	8
REALIZACION REAL.....	8
<i>Calendario</i> .....	8
<i>Leyenda</i> .....	15
<b>ANEXO 2 – GESTION DE CONFIGURACIONES</b>	<b>16</b>
ALCANCE DE ESTE DOCUMENTO .....	17
DESCRIPCION GENERAL .....	17
DISTRIBUCION.....	18
<i>Ficheros fuente</i> .....	18
<i>Contenido Web</i> .....	19
<i>Test</i> .....	20
<i>Software utilizado</i> .....	20
<i>Backups</i> .....	20
<i>Documentación</i> .....	20
<b>ANEXO 3 – ANALISIS</b>	<b>21</b>
ÁMBITO DEL DOCUMENTO .....	22
DESCRIPCION Y OBJETIVOS .....	23
SISTEMA ACTUAL.....	24
IDENTIFICACION Y DEFINICION DE REQUISITOS .....	25
<i>Ambito y alcance del proyecto</i> .....	25
<i>Catálogo de requisitos del sistema</i> .....	26
<b>ANEXO 4 – DISEÑO DE BASE DE DATOS</b>	<b>29</b>
ÁMBITO DEL DOCUMENTO .....	30
MODELO DE DATOS .....	31
<i>Modelo conceptual de los datos</i> .....	31
<i>Modelo lógico de los datos</i> .....	37
<i>Normalización</i> .....	37
<i>Modelo físico de los datos (Implementación de la Base de Datos)</i> .....	43
DICCCIONARIO DE DATOS.....	45
<b>ANEXO 5 – DISEÑO</b>	<b>51</b>
ÁMBITO DE ESTE DOCUMENTO.....	52
CASOS DE USO .....	53
<i>Casos de uso de la parte pública</i> .....	53
<i>Casos de uso de la parte privada</i> .....	58
ARQUITECTURA DE SOFTWARE DEL SISTEMA .....	61
<i>Capa de Modelo</i> .....	61
<i>Fachada y componentes</i> .....	62
<i>Servidor web</i> .....	63
DISEÑO DE CLASES .....	64
PATRONES DE DISEÑO .....	65
<i>Patrón DAO</i> .....	65
<i>Patrón Fachada</i> .....	65
<b>ANEXO 6 – PLAN DE PRUEBAS</b>	<b>66</b>
DESCRIPCION Y OBJETIVOS .....	67
PRUEBAS DE VALIDACION Y ROBUSTEZ.....	67
<i>Pruebas sobre los requisitos funcionales de la parte pública</i> .....	67
<i>Pruebas sobre los requisitos funcionales del sistema</i> .....	70
<i>Pruebas sobre los requisitos funcionales de la parte privada</i> .....	71
PRUEBAS UNITARIAS .....	72
<i>Pruebas sobre el acceso a datos</i> .....	72
<i>Pruebas sobre el acceso al servicio web</i> .....	72

PRUEBAS DE ACEPTACION .....	73
<i>Pruebas de la parte privada</i> .....	73
<i>Pruebas de la parte pública</i> .....	75
<b>ANEXO 7 – USABILIDAD</b>	<b>76</b>
DESCRIPCION Y OBJETIVOS .....	77
<i>Pruebas con usuario inexperto</i> .....	77
<i>Pruebas con usuario experto</i> .....	77
<b>ANEXO 8 – RESULTADO DE LAS PRUEBAS</b>	<b>78</b>
DESCRIPCION Y OBJETIVOS .....	79
RESULTADOS DE LAS PRUEBAS DE VALIDACION Y ROBUSTEZ .....	79
<i>Pruebas sobre los requisitos de la parte pública</i> .....	79
<i>Resultados de las pruebas sobre los requisitos del sistema</i> .....	81
<i>Resultados de las pruebas sobre los requisitos de la parte privada</i> .....	81
RESULTADOS DE LAS PRUEBAS UNITARIAS .....	82
<i>Resultados de las pruebas sobre el acceso a datos</i> .....	82
RESULTADOS DE LAS PRUEBAS DE ACEPTACION Y USABILIDAD .....	84
<i>Resultados de las pruebas de la parte privada</i> .....	84
<i>Pruebas de la parte pública</i> .....	87
<b>ANEXO 9 – DOCUMENTACION</b>	<b>90</b>
DESCRIPCION .....	93
ARQUITECTURA .....	94
<i>Spring security</i> .....	94
<i>Capa de vista (JSP)</i> .....	95
<i>Capa de control (Componentes)</i> .....	96
<i>Capa de fachada al Modelo (Facade)</i> .....	98
<i>Capa de acceso a datos (User DAO y Admin DAO)</i> .....	98
<i>Capa de mapping (ORM)</i> .....	98
<i>Capa de datos (SGBD)</i> .....	98
MODULO DE ACCESO A DATOS.....	99
<i>Beans</i> .....	99
<i>Data access object (DAO)</i> .....	101
<i>Periodic Task</i> .....	101
<i>Otros</i> .....	101
SERVICIO WEB .....	101
<b>ANEXO 10 – MANUAL DE USUARIO</b>	<b>103</b>
ÁMBITO DE ESTE DOCUMENTO.....	104
INDICE Y AYUDAS .....	104
RESUMEN DE LA PARTE PUBLICA .....	105
CREACION DE AUDITORIAS .....	106
<i>Selección de personal auditado ó auditor</i> .....	107
<i>Selección del tipo de auditoría</i> .....	108
MODIFICACION DE AUDITORIAS .....	109
EJECUCION DE AUDITORIAS .....	110
RESUMEN DE AUDITORIA .....	111
RESULTADO DE AUDITORIA .....	113
SEGUIMIENTO DE AUDITORIA.....	114
<i>Rellenando el seguimiento</i> .....	115
SEGUIMIENTO DE PROYECTO .....	116
SELECCION PREVIA DE AUDITORIA .....	117
INFORMES Y METRICAS .....	118
<i>Informe general de auditorías</i> .....	118
<i>Top ten de preguntas contestadas negativamente</i> .....	119
<i>Informe de revisiones</i> .....	121
<i>Media de no conformidades por auditoría</i> .....	121
<i>Proyectos y auditorías caducados</i> .....	122
<i>Estadísticas para las auditorías</i> .....	123
SERVICIOS WEB .....	124
RESUMEN DE LA PARTE PRIVADA .....	125
ADMINISTRACION DE PREGUNTAS.....	126
ADMINISTRACION DE PROCESOS.....	128
ADMINISTRACION DE PERSONAL .....	129
ADMINISTRACION DE PROYECTOS.....	129
ADMINISTRACION DE TIPOS DE NO CONFORMIDAD.....	130
ADMINISTRACION DE ESTADOS DE PROYECTO/AUDITORIA .....	130

ADMINISTRACION DE TIPOS DE AUDITORIA.....	131
ADMINISTRACION DE PARAMETROS Y FORMULA DE CALCULO .....	132
ADMINISTRACION DE LAS CUENTAS DE ADMINISTRADORES .....	133

## **Descripción**

El objetivo de este documento es explicar las partes del proyecto software, el significado de cada una de las partes para posibles futuras mejoras o ampliaciones de funcionalidad.

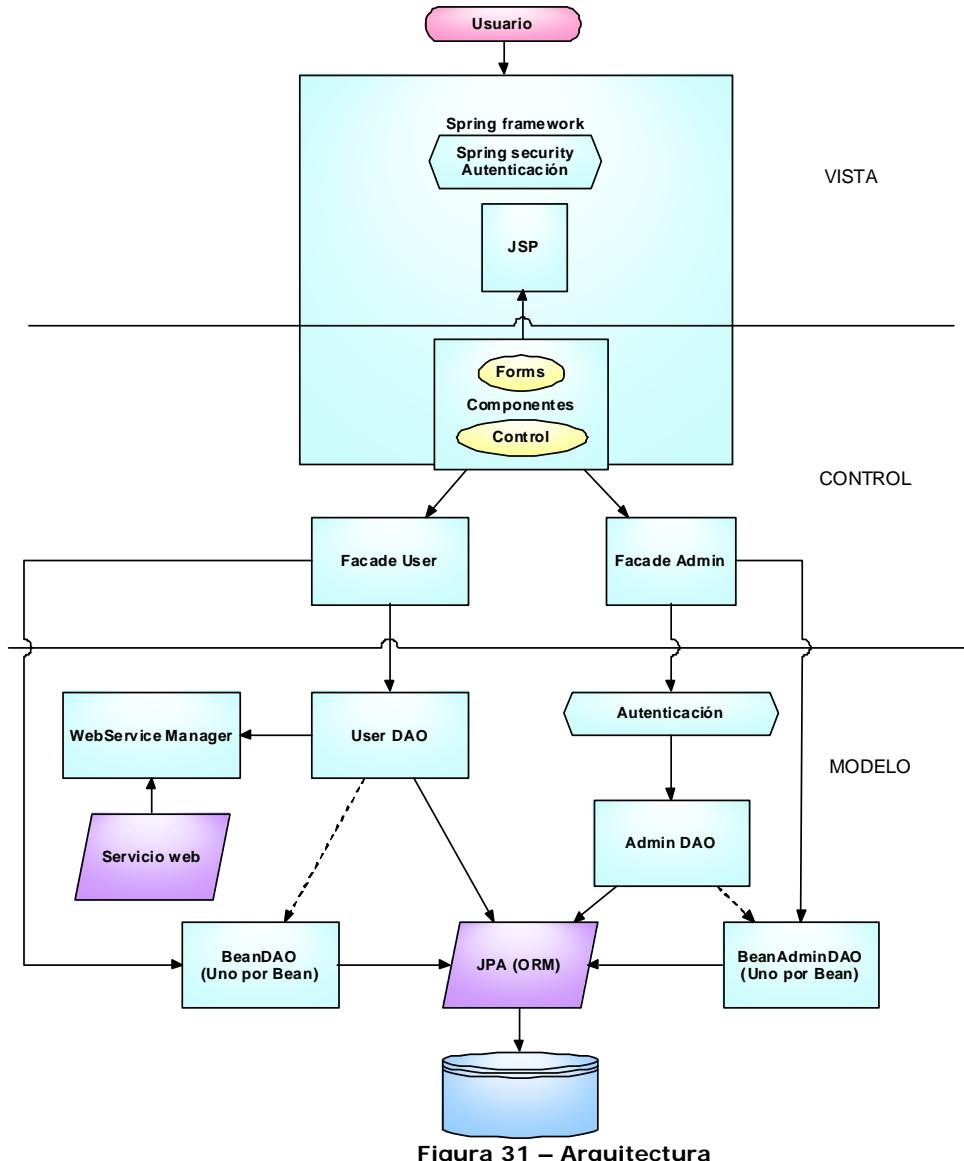
En primer lugar se explicará la arquitectura principal del sistema, y dónde están localizadas estas partes en el código fuente.

En los siguientes apartados se explicará cada uno de los módulos implementados y su significado.

## Arquitectura

Podemos observar en la **Figura 25** que existe una capa de seguridad Spring-security, la cual nos permite o no el acceso a diferentes vistas del modelo; más abajo observamos los componentes, que se encargan de gestionar las llamadas correspondientes al control tras validar los formularios de ingreso accediendo a la fachada correspondiente.

Estas fachadas nos encapsulan de la capa de acceso a datos, tanto como usuario como administración, que se realiza a través de los módulos DAO.



## Spring security

Las propiedades de esta capa de seguridad vienen definidas en un fichero "gauditint-admin-security.xml" que utiliza una clase para codificar la contraseña (*utils.Encoder*).

Los archivos jsp están dentro de la carpeta *WEB-INF/jsp*, por lo que quedarán ocultos, salvo el *index.jsp* que simplemente redirecciona al inicio.

Este modulo de Spring nos ofrece una capa de seguridad en las vistas. Además de esta hay otra capa de seguridad más en el DAO, un poco más oculta y que restringe el acceso a los datos.

## Capa de vista (JSP)

Todos los archivos de la capa de vista se encuentran en la carpeta de contenido web:

**/apps/gauditint/**

Sin tener en cuenta las tres subcarpetas de código fuente:

gauditint-dao (Capa de datos)  
gauditint-webuser (Capa de control usuario)  
gauditintn-webadmin (Capa de control administración)

## Spring y JSP

Las vistas del modelo se han implementado con la tecnología de Java Server Pages, en su mayor parte se ha utilizado lo menos posible las inserciones de código java en las páginas mediante `<% %>` y `<%= %>`, y explotando la funcionalidad de los tags de **Spring**.

Los tags más utilizados son `<spring:bind>` y diversos tags del core `<c:xxxx>`, como por ejemplo `<c:forEach>` `<c:out>` `<c:if>`, así como el tag de formularios `<form:xxx>`.

Para aprender en el manejo de Spring hemos seguido el tutorial de su página web:  
<http://static.springsource.org/docs/Spring-MVC-step-by-step/>

Conforme se iba desarrollando el proyecto hemos ido ampliando la funcionalidad utilizada del framework buscando por la web cómo hacer ciertas cosas, como por ejemplo, asignar una lista de listas a un atributo de la clase que contiene los datos de un formulario.

## Jerarquía

Una vista cualquiera esta dividida en varios jsp, por lo general cinco jsp forman una página.

directivas.jsp (Importa las librerías de tags)  
menu.jsp / menuAdmin.jsp (Menú de la página) [ incluye a la cabecera ]  
cabecera.jsp (Cabecera simple de la página)  
pie.jsp (Pie de página)  
xxxx.jsp ( contenido de la propia página, que incluirá las directivas, el menú correspondiente y el pie)

## Diseño

Para el diseño de la página se han utilizado diversos archivos css, que contienen la descripción del diseño.

bloques.css (información de los bloques generales de la web, #cabecera, #contenido y #pie, así como estilos para tablas, botones, errores, etc.)  
menu.css (información de los menús)  
suggestion.css (información de las listas que salen en los inputs con sugerencias)  
calendarios.css (información para los calendarios mostrados en popup)

## Soporte para idiomas

Se ha desarrollado soporte para idiomas utilizando la funcionalidad de Spring para incorporar los mensajes que se encuentran definidos en un fichero **messages.properties**, por lo que implementar el cambio de idioma a sido bastante sencillo y cómodo, pese a no ser necesario se ha decidido realizarlo por aprender la metodología. Hemos seguido los pasos que hay en un ejemplo de este tutorial:

<http://www.programacionj2ee.com/2009/08/19/internacionalizacion-i18n-en-spring/>

## Recursos utilizados de la web

Se han utilizado diversos recursos obtenidos de la web:

Calendarios dinámicos para el ingreso de fechas en formularios. Hemos utilizado el calendario del repertorio de código en common.

Ajax Suggestion, para la búsqueda en la base de datos de personal:

(Lo hemos modificado para que sea genérico y sirva para cualquier parte de la web.)

<http://www.forosdelweb.com/f13/autocomplete-457652/>

## Custom Tags

Se han creado dos custom tags, ambos para generar el contenido de un select, uno inserta proyectos y el otro inserta auditorías, según una de las opciones seleccionadas.

Para los proyectos la opcionalidad está entre:

- proyectos que tienen alguna auditoría
- proyectos que tienen alguna auditoría completada
- proyectos que tienen alguna auditoría ejecutable
- proyectos que tienen seguimiento
- proyectos auditables
- todos los proyectos (por defecto)

Para las auditorías la opcionalidad es similar:

- auditorías ejecutables (para ejecutar la auditoría)
- todas las auditorías (para modificar la auditoría)
- auditorías completadas (para acceder al resumen de auditoría)
- auditoría con seguimiento (para acceder al seguimiento)

Esto nos sirve para poder seleccionar más cómodamente los proyectos y/o auditorías de una lista de selección más corta de lo que sería si mostrásemos todas.

## Capa de control (Componentes)

El framework de Spring utilizado, requiere de diversos componentes (Controladores, clases de soporte para formularios y validadores) que se encuentran dentro del modulo **apps/gauditint/gauditint-webuser/src/java** (para la parte de usuario) y **apps/gauditint/gauditintn-webadmin/src/java** (para la parte de administración).

Ambos tienen los **componentes** divididos en diversos paquetes java: control, forms, validators, servlets.

Los archivos control, forms y validators son los usados por el framework de spring, como controladores, soporte para formularios y validadores respectivamente, todos vienen definidos en el archivo: "**gauditint-global-servlet.xml**", como veremos más adelante.

El aspecto de este archivo es la configuración de Springs, y trata de mapear una URL lógica, con un **controlador**, que ofrece una capa vista (Jsp) a la que le añade datos traídos del modelo a través de la fachada correspondiente. De esta forma el controlador es un nexo de unión entre la vista y el modelo, siguiendo así el patrón Modelo-Vista-Controlador.

Además si el controlador extiende alguna clase controladora de formulario (*SimpleFormController* por ejemplo) significa que da soporte a formularios, recogiendo los datos a través de alguna de las clases que se encuentran en el paquete java **forms**

En el siguiente ejemplo observamos cual es el controlador, formulario y validador que se encargan de la url *"/nueva-auditoría.htm"*. Este es un trocito de ejemplo del fichero que nombramos anteriormente, *"gauditint-global-servlet.xml"*

```
<bean name="/nueva-auditoría.htm" class="control.NuevaAuditoríaController">
    <property name="sessionForm" value="true"/>
    <property name="commandName" value="NuevaAuditoría"/>
    <property name="commandClass" value="forms.NuevaAuditoría"/>
    <property name="validator">
        <bean class="validators.NuevaAuditoríaValidator"/>
    </property>
    <property name="formView" value="nuevaAuditoría"/>
    <property name="successView" value="Success"/>
</bean>
```

Podemos observar que para la url “/nueva-auditoría.htm”, la **clase controladora** será “**NuevaAuditoríaController**” (dentro del paquete control) que deberá implementar la interfaz Controller, además vemos que es un formulario (por las propiedades sessionForm, commandName, commandClass, validator, formView y successView) por lo que el controlador deberá extender alguna de las implementaciones de FormController (que ya implementan la interfaz Controller).

En el ejemplo, la **clase contenadora** de los datos del formulario es “**NuevaAuditoría**” (dentro del paquete forms) a la que se le asigna un nombre “NuevaAuditoría”, con este nombre podemos acceder desde los JSP: <form:form commandClass="NuevaAuditoría"> ó <spring:bind path="NuevaAuditoría.nombre-atributo">.

La **clase validador** en este ejemplo sencillo será “**NuevaAuditoríaValidator**” (dentro del paquete validators) que implementará la interfaz Validator, implementando el método validate(obj,errors), soportando la clase forms.NuevaAuditoría y añadiendo errores al objeto errors en cada campo que falle, de esta forma si una validación es erronea (Contiene errores), no llega a ejecutarse el método onSubmit, se regenera la vista (formView) con mensajes de error en los campos.

No todas las url requieren de formulario, ni todas las que requieren formulario requieren además validador.

## Control

Los diversos controladores se comportan de forma muy similar, en el método **handleRequest** utilizan el modelo generado por la clase padre (**super.handleRequest**), y añaden los datos necesarios para generar la vista al modelo.

Mientras que en el método **onSubmit** realizan las acciones requeridas según la acción que precise, obteniendo los datos del formulario a través de la clase definida en el fichero xml correspondiente.

Para los formularios con múltiples acciones, Spring no ofrecía ningún mecanismo (si lo hacia sin embargo para múltiples request mediante la clase MultiActionController, pero no para formularios, es decir, no existía MultiActionFormController), por esta razón y para evitar usar JavaScript para modificar parámetros/valores de atributos, decidimos implementar nuestro propio MethodNameResolver, para resolver el método al cual llamar dentro de un controlador dependiendo de la acción que va oculta en un input del formulario.

Este **MethodNameResolver** obtiene un formulario baseForm (El command del onSubmit del SimpleFormController) que es una de las clases formulario pero que tiene de especial que extiende baseForm, que es una clase con un atributo acción y otro una lista de parámetros.

Esta clase obtiene a partir del parámetro acción los parámetros y llama al método que corresponda con:

```
nombreMetodo(param1,param2,...,paramn,command,errors);  
nombreMetodo(param1,param2,...,paramn,request,response,command,errors);
```

Al primero se le llama desde onSubmit(controller,command,errors); y al segundo desde onSubmit(controller,command,errors,request,response);, dado que algunos de los SimpleFormController no necesitan los parámetros request,response.

### Ejemplo

De esta forma si el atributo acción vale : ‘borrar(1&2)’ , se ejecutara el método onSubmit del simpleFormController, y este simplemente llamará al método onSubmit de MethodNameResolver correspondiente :

```
return new MethodNameResolver().onSubmit(this,command,errors);
```

Y desde ahí se ejecutará el método:

```
controller.borrar(1,2,command,errors);
```

Donde el controller es la clase que llamo al MethodNameResolver.

Actualmente los parámetros son solo de tipo entero, podría modificarse a parámetros de cualquier tipo modificando esta clase y estableciendo los tipos de los argumentos como Objetos realizando el cast en cada uno de los métodos llamados a entero.

La **validación** del método a invocar se encuentra en `baseForm`, que es una clase abstracta con el método “`boolean validate(String acción)`” que indica que acciones pueden ser llamadas para su controlador, de esta forma se evita que puedan llamarse a cualquiera de los métodos de la clase, modificando el valor del input con nombre “acción”, lo cual sería un fallo de seguridad importante.

### Form

Son clases java planas, con atributos y métodos `get/set`, por cada input del formulario hace falta un atributo distinto, si es una lista de atributos iguales se utilizan listas, mientras que si es un listado de listas de valores, se utilizan listas de listas y acarrean ciertos problemas en su inicialización.

### Validator

Estas clases han sido usadas poco, dado que en la mayoría de formularios el método de validación es diferente dependiendo de la acción requerida, por ello la validación se realiza muchas veces en el propio método `onSubmit`, lo cual es algo más lento.

Una opción interesante para resolver esto sería usar un `MultiActionController`, estableciendo un validador para cada acción.

### Servlet

Los servlets sirven para obtener datos de llamadas Ajax, hay dos implementados, uno de ellos sirve para buscar personal con parte de nombre y/o parte de código (`buscaPersonal.java`), y el otro para obtener las auditorías de un proyecto (`obtenerAuditoríasProyecto.java`).

## Capa de fachada al Modelo (Facade)

La fachada de usuario y administrador ofrecen métodos, algunos más complejos que otros, para el control del sistema, por ejemplo el obtener un cuestionario para una auditoría tiene efectos co-laterales según la lógica de negocio, pues estos efectos están implementados en esta fachada.

En el ejemplo de la obtención del cuestionario para una auditoría, cuando éste se obtiene por primera vez, se crea para la Auditoría un objeto respuesta nuevo por cada pregunta del cuestionario, cuando se llama por segunda vez (para la misma auditoría) no deben generarse más respuestas, ni ninguna nueva.

Otras partes de la lógica de negocio están implementadas en los propios controladores.

## Capa de acceso a datos (User DAO y Admin DAO)

Ambas clases de acceso a datos están definidas en el modulo de acceso a datos (sección 0), y sirven para encapsular los métodos de acceso a datos y su implementación.

## Capa de mapping (ORM)

Se trata de un mapeador objeto relacional, en nuestro caso hemos usado JPA, con la implementación de Hibernate, la cual nos ha funcionado bastante bien. El software desarrollado cumple la especificación JPA, por lo que cambiar de proveedor no debería ser ningún problema, al menos en principio.

## Capa de datos (SGBD)

El sistema gestor de base de datos utilizado es SQL Server 2005. El diseño de la base de datos se puede obtener del fichero **Análisis y Diseño**.

## Módulo de Acceso a Datos.

En este modulo se encuentra el código fuente relacionado con el acceso a datos, beans, etc. Se encuentra en la carpeta de código fuente:

*"apps/gauditint/gauditint-dao/src/java".*

En el encontramos diversos paquetes java que explicaremos a continuación.

### Beans

El paquete **beans** contiene clases java con anotaciones JPA para el mapeo objeto-relacional a una base de datos, la información relevante a ésta se encuentra en el archivo "persistente.xml" donde encontramos las diferentes persistence unit, una para administración y otra para usuarios.

### Aviso

Almacena la información necesaria para la generación de avisos, existe una tarea que lee los avisos pendientes, los envía y los elimina de la base de datos. Las tareas periódicas están la parte de administración.

### Administrador

Información de los administradores del sistema, además de guardarse esta información en una tabla propia, se guardan también en las tablas necesitadas por Spring-security.

### Auditoría

Información relevante a una auditoría.

### CriterioEstado

Guarda la información relevante a la forma de calcular el estado general de un proyecto, es decir, para un tipo de no conformidad y un estado, se dice cual es el máximo número de no conformidades de ese tipo que puede haber para poder estar en dicho estado.

Es la relación M:N entre Tipo de no conformidad – Estado General, donde existe un atributo entero ("Máximo numero de no conformidades")

### CriterioEstadoPK

Necesaria para JPA al tener un atributo en la relación.

### EstadoGeneral

Son los estados de un proyecto / auditoría debido al número de no conformidades que tenga (que vendrá definido por el criterio de estado).

Explicación: Para que un proyecto tenga estado "Bueno", tiene que tener menos de 5 no conformidades "Leves" y menos de 1 no conformidad "Grave".

Donde "Leve" y "Grave" son los tipos de no conformidad existentes, "Bueno" es el estado general, y los números están en CriterioEstado.

**Personal**

Información de las personas que pueden ser auditores/auditados/jefes de proyecto.

**Pregunta**

Información de las preguntas

**Proceso**

Procesos de una auditoría, cada pregunta esta asignada a un proceso. Los procesos son las pestañas en la ejecución de una auditoría.

**Proyecto**

Información de los proyectos, tienen su seguimiento.

**ReferenciaDocumental**

Una pregunta debe estar documentada su respuesta en un determinado documento.

**Respuesta**

Guarda la información de la respuesta y observaciones.

**Revisión**

Son las respuestas negativas de las auditorías, almacena toda la información relevante a cada revisión, asignada a un Seguimiento concreto y a una respuesta de alguna auditoría.

**Seguimiento**

Tanto los proyectos como auditorías tienen un seguimiento, el seguimiento se almacena también en la auditoría dado que algunas auditorías no tienen proyecto, y también se almacena a en proyecto porque el seguimiento de un proyecto comprende todas las no conformidades de todas sus auditorías, mientras que el seguimiento de la auditoría solo comprende las no conformidades de la auditoria.

**TipoAuditoría**

Existen diversos tipos de auditoría y se almacenan como entidad debido a su relación con las preguntas, a través de esta relación es como se obtiene el cuestionario de una auditoría.

El cuestionario solo se obtiene una vez y en el instante anterior a su ejecución.

**TipoNoConformidad**

Se refiere a la gravedad de una respuesta negativa, para que pueda ser configurable, así como el estado del proyecto y la formula de cálculo.

## Data access object (DAO)

El DAO (Data access object) es un patrón de diseño para el acceso a datos a través de un objeto. En este proyecto se ha dividido en dos:

### DAOGlobal y DAOGlobalAdmin

#### DAOGlobal

Se ha implementado utilizando el patrón factoría, donde *GlobalDAOFactory* es una clase abstracta que nos inicializa *LocalUserDAOImpl* como implementación de JPA Hibernate, la implementación realizada en *LocalUserDAOImpl* es genérica para cualquier implementación de JPA, sin embargo hay que tener cierto cuidado en la elección de proveedor dado que en pruebas con *eclipseLink* (otra implementación de JPA) ha habido problemas.

Esta clase se instancia mediante el método “getDAOFactory(int witchFactory)”, en este caso solo se ha realizado una implementación de dicha factoría *LocalUserDAOImpl*, que extiende esta clase abstracta e implementa todos sus métodos directos así como la obtención de cada uno de los DAO necesarios para insertar/actualizar/borrar entidades.

Cada uno de estos DAO implementa una interfaz, que se encuentran en DAO.DAOInterfaces. En el paquete DAO.DAOImplementations se encuentran estas implementaciones.

Si se desea cambiar la forma de acceso a datos, solo hay que implementar las interfaces de los DAO de cada uno de los beans, y crear una clase que extienda la clase abstracta *GlobalDAOFactory*, añadir esta implementación al método *getDAOFactory(int witchFactory)* añadiendo una constante entera distinta a *USER\_FACTORY*, que es la que existe actualmente como única implementación elegible. Y utilizarla del mismo modo que se utiliza actualmente.

```
GlobalDAOFactory gdf = GlobalDAOFactory.getDAOFactory(GlobalDAOFactory.MY_FACTORY);
```

De forma análoga se ha generado *GlobalAdminDAOFactory*, con su respectiva implementación *LocalAdminDAOImpl*.

En esta implementación el constructor de la clase requiere un parámetro Administrador admin, de forma que si el administrador no es correcto no se pueda acceder a ninguno de los métodos de la clase.

## Periodic Task

Este paquete java Periodic Task se encuentran las clases que ejercen diversas tareas periódicas, una de ellas sirve para enviar por email los avisos generados por la aplicación, mientras que la otra simplemente se encarga de actualizar el estado de las auditorías que han caducado.

## Otros

En este modulo de acceso a datos se encuentran algunas clases contenedores (Holders) que sirven para facilitar el manejo de diversas consultas a la base de datos que se almacenan en estas clases contenedoras, se encuentran en beans.estado, beans.informes, beans.metricas.

Así mismo también se encuentra un paquete de comparadores que sirven para ordenar las respuestas por proceso y ordenar las preguntas por el porcentaje de respuestas negativas.

## Servicio Web

El servicio web y acceso a servicio web esta implementado en el paquete com.grupocomex.ws, la clase principal se llama **WebServiceManager** y ofrece los métodos de **buscarPersonal(String codigoRecurso)** y **buscarProyecto (String codigoRecurso)** que devuelven un bean Personal y un listado de bean Proyecto respectivamente.

Esta clase se encarga de hacer peticiones al servicio web correspondiente para obtener los datos solicitados.

Se ha utilizado WSS4J para el acceso al servicio web, debido a que se usa WSE 3.0 en la parte del servidor, se ha utilizado la librería addressing-1.1.jar para poder compatibilizar las cabeceras.

Las cabeceras de seguridad no son generadas en WSE 3.0 sobre el WSDL, por lo que el cliente que genera las clases a partir del WSDL carece de esta información. Si además cuando se ha establecido en el servidor la opción *requireActionHeader*, el servidor espera algo que el cliente por defecto no pone si en el wsdl no se pide.

Para solucionarlo hemos añadido en el deploy.wsdd las siguientes líneas:

```
<handler type="java:org.apache.axis.message.addressing.handler.AddressingHandler">
    <parameter name="referencePropertyNames" value="*"/>
</handler>
```

# ANEXO 10 – Manual de usuario

## Índice

ANEXO 10 – MANUAL DE USUARIO .....	103
ÁMBITO DE ESTE DOCUMENTO .....	104
ÍNDICE Y AYUDAS .....	104
RESUMEN DE LA PARTE PÚBLICA .....	105
CREACIÓN DE AUDITORIAS .....	106
Selección de personal auditado ó auditor .....	107
Selección del tipo de auditoría .....	108
MODIFICACIÓN DE AUDITORIAS .....	109
EJECUCIÓN DE AUDITORIAS .....	110
RESUMEN DE AUDITORIA .....	111
RESULTADO DE AUDITORIA .....	113
SEGUIMIENTO DE AUDITORIA .....	114
Rellenando el seguimiento .....	115
SEGUIMIENTO DE PROYECTO .....	116
SELECCIÓN PREVIA DE AUDITORIA .....	117
INFORMES Y METRICAS .....	118
Informe general de auditorías .....	118
Top ten de preguntas contestadas negativamente .....	119
Informe de revisiones .....	121
Media de no conformidades por auditoría .....	121
Proyectos y auditorías caducados .....	122
Estadísticas para las auditorías .....	123
SERVICIOS WEB .....	124
RESUMEN DE LA PARTE PRIVADA .....	125
ADMINISTRACIÓN DE PREGUNTAS .....	126
ADMINISTRACIÓN DE PROCESOS .....	128
ADMINISTRACIÓN DE PERSONAL .....	129
ADMINISTRACIÓN DE PROYECTOS .....	129
ADMINISTRACIÓN DE TIPOS DE NO CONFORMIDAD .....	130
ADMINISTRACIÓN DE ESTADOS DE PROYECTO/AUDITORIA .....	130
ADMINISTRACIÓN DE TIPOS DE AUDITORIA .....	131
ADMINISTRACIÓN DE PARÁMETROS Y FÓRMULA DE CALCULO .....	132
ADMINISTRACIÓN DE LAS CUENTAS DE ADMINISTRADORES .....	133

## Ámbito de este documento

El objetivo principal de este documento es orientar al usuario ante las posibles dudas que el sistema pueda causar, su principal objetivo es aclarar los diversos puntos de la aplicación de una forma un poco más gráfica, de forma que se especifique toda la funcionalidad de la interfaz de usuario.

## Índice y ayudas

La aplicación dispone de un pequeño manual en las páginas iniciales, tanto para usuarios normales como para administradores (en cuyo caso es más extenso).

Markado en color rojo en la siguiente figura, observamos el pequeño manual incorporado en la página inicial de la aplicación.

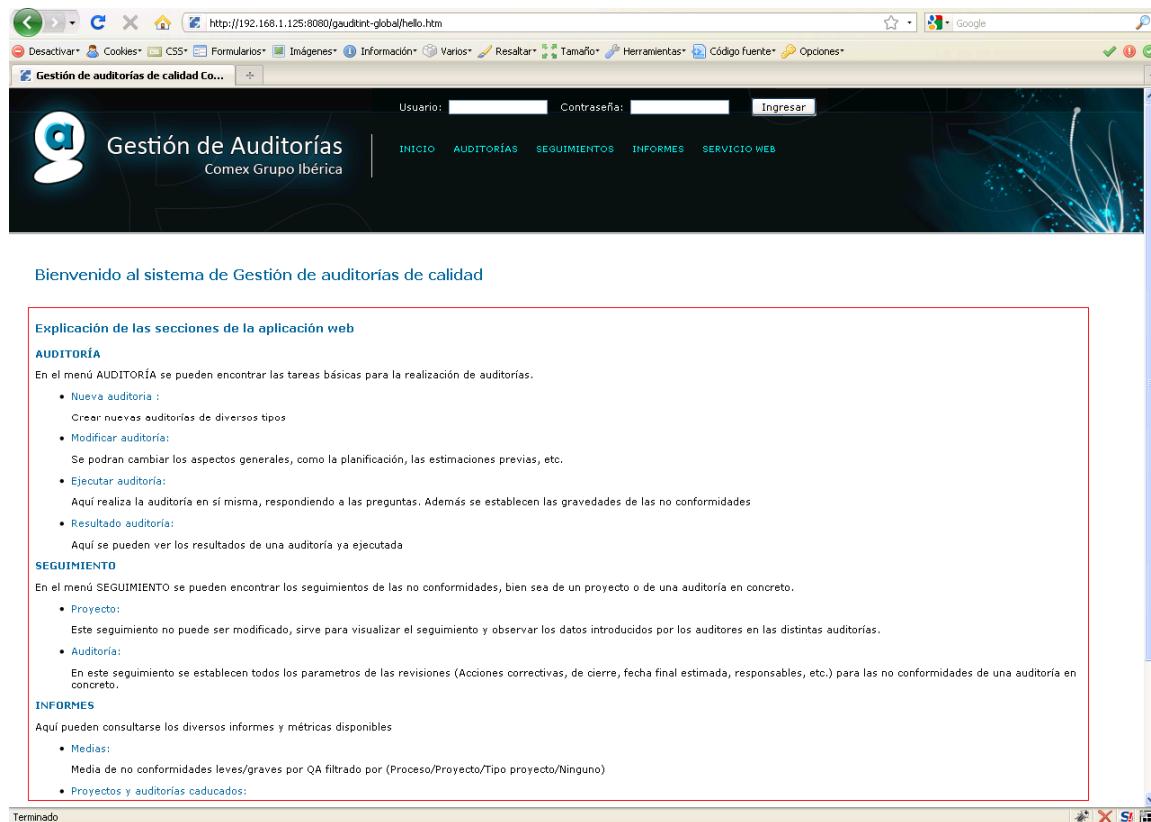


Figura 32

Las ayudas, se reparten en diferentes sitios conflictivos, donde se han detectado problemas en las pruebas de usabilidad, se requiere tener Javascript activado para su visualización. Las ayudas se esconden tras una imagen de "información" marcada con el símbolo 'i'.

Al disponer el ratón sobre el ícono, se nos muestra un rótulo con la información relevante.

Vemos en la siguiente figura, en rojo marcados los iconos de información y el rótulo con la información de uno de ellos

**Nueva Auditoria**

**Auditores y auditados** ? Pueden seleccionarse multiples auditores y auditados.

**Auditores QA:**

Utilizando el texto para buscar por código o nombre (o apellido) y posteriormente dándole a "Seleccionar".

Buscar:

Utilizando el listado con "Control + Click" para selección múltiple.  
Un solo click para seleccionar solo a uno.

E705 - Ignacio VILALTA Esteban	E705 - Ignacio VILALTA Esteban
E550 - Jose Antonio PERNA Dominguez	E550 - Jose Antonio PERNA Dominguez
E505 - Manuel MAZARICO Torres	E505 - Manuel MAZARICO Torres
E600 - Raul NOVOA Minguiez	E600 - Raul NOVOA Minguiez
E606 - Vicente GIL Zuazo	E606 - Vicente GIL Zuazo

**Tipo de auditoría** ?

**Tipos normales**

Q41 Tras el análisis y el diseño

Q42 Tras la fase de desarrollo

Q43 Tras la fase de implantación

**Tipos Especiales (no requieren proyecto)**

PPOA Auditoría de la gestión de auditorías

**Proyecto:**

**Otros parámetros**

Horas planificadas:

Riesgo previo de la auditoría:

Lista de distribución ?

Terminado

**Figura 33**

## Resumen de la parte pública

En la parte pública encontramos diversas funcionalidades ordenadas en los menús de la siguiente forma:

### AUDITORÍA

En el menú AUDITORÍA se pueden encontrar las tareas básicas para la realización de auditorías.

- Nueva auditoria :  
Crear nuevas auditorías de diversos tipos
- Modificar auditoría:  
Se podrán cambiar los aspectos generales, como la planificación, las estimaciones previas, etc.
- Ejecutar auditoría:  
Aquí realiza la auditoría en sí misma, respondiendo a las preguntas. Además se establecen las gravedades de las no conformidades
- Resultado auditoría:  
Aquí se pueden ver los resultados de una auditoría ya ejecutada

## SEGUIMIENTO

En el menú SEGUIMIENTO se pueden encontrar los seguimientos de las no conformidades, bien sea de un proyecto o de una auditoría en concreto.

- Proyecto:

Este seguimiento no puede ser modificado, sirve para visualizar el seguimiento y observar los datos introducidos por los auditores en las distintas auditorías.

- Auditoría:

En este seguimiento se establecen todos los parámetros de las revisiones (Acciones correctivas, de cierre, fecha final estimada, responsables, etc.) para las no conformidades de una auditoría en concreto.

## INFORMES

Aquí pueden consultarse los diversos informes y métricas disponibles

- Medias:

Media de no conformidades leves/graves por QA filtrado por (Proceso/Proyecto/Tipo proyecto/Ninguno)

- Proyectos y auditorías caducados:

Métrica en la cual podemos ver cuantas auditorías están caducadas, en cuantos proyectos, en número y en porcentaje.

- Estadísticas auditorías:

Informe en el cual vemos el número de las auditorías (Caducadas, planificadas, ejecutadas, etc.) entre dos fechas (por defecto por año) agrupadas por semestres (o cualquier otro intervalo, bien sea bimensual, trimestral, etc.)

- Informe de revisiones:

Aquí podremos ver un resumen de las revisiones (no conformidades) que están pendientes o en curso, y desde aquí tendremos un acceso directo a su seguimiento para poder terminarlas.

## Creación de auditorías

Para la creación de auditorías hay que ir en el menú a:

**AUDITORIA -> Nueva auditoría**

De esta forma nos encontraremos una pantalla en la cual deberemos llenar la información correspondiente.

En la siguiente figura vemos diferentes marcas, en rojo los auditores y auditados, deberemos seleccionar al personal que va a ser auditor y auditado respectivamente.

En azul el tipo de auditoría a seleccionar, se creará una nueva auditoría por cada tipo de auditoría seleccionada, pero solo podemos seleccionar tipos normales (que requieren proyecto) o tipos especiales, pero nunca los dos a la vez. Por defecto irán marcados los tipos que el administrador considere.

En morado tenemos otros parámetros que son opcionales, que podrán ser modificados más tarde, no así ocurre con los auditores, auditados ni tipo de auditoría. En caso de error procederemos a invalidar la auditoría.

The screenshot shows the 'Nueva Auditoria' (New Audit) page. It has two main sections: 'Auditores y auditados' (Auditors and Auditees) and 'Tipo de auditoría' (Audit Type). The 'Auditores y auditados' section contains two lists: 'Auditores QA' and 'Auditados'. Both lists have a 'Buscar:' input field and a 'Seleccionar' button. The 'Auditores QA' list contains: E705 - Ignacio VILALTA Esteban, E550 - Jose Antonio PERNA Dominguez, E505 - Manuel MAZARICO Torres, E600 - Raul NOVOA Minguez, E806 - Vicente GIL Zuazo. The 'Auditados' list contains: E705 - Ignacio VILALTA Esteban, E550 - Jose Antonio PERNA Dominguez, E505 - Manuel MAZARICO Torres, E600 - Raul NOVOA Minguez, E806 - Vicente GIL Zuazo. The 'Tipo de auditoría' section has a radio button for 'Tipos normales' (selected) and another for 'Tipos Especiales (no requieren proyecto)'. Under 'Tipos normales', there are four checkboxes: Q1 Tras el análisis y el diseño, Q2 Tras la fase de desarrollo, Q3 Tras la fase de implantación, and PPQA Auditoría de la gestión de auditorías (unchecked). The 'Otros parametros' section includes: 'Horas planificadas: 0', 'Riesgo previo de la auditoría: Seleccionar', and 'Lista de distribución:'. A 'Guardar' button is at the bottom.

Figura 34

### Selección de personal auditado ó auditor

Para la selección del personal tenemos dos opciones:

La **primera**, la más sencilla cuando hay pocas personas es la selección manual a través del select multiple dispuesto, donde se ven los nombres de las personas. Mediante Cntrl + click podemos seleccionar varios, para des-seleccionar todos, basta con **soltar** la tecla Cntrl y pulsar con el ratón en una persona hasta que quede seleccionada ella sola y nuevamente Cntrl + click sobre la persona seleccionada.



Figura 35

La **segunda** opción, más cómoda cuando existen múltiples personas en el sistema, es acceder mediante el input de texto que se encuentra en la zona superior, donde podremos buscar a las personas tanto por nombre, apellido o código en la empresa. Si la persona no se encontrara en la base de datos, el sistema buscará su código por el servicio web, pero únicamente su código.

Para seleccionarlo finalmente hay que pulsar “seleccionar”, lo cual nos seleccionará la persona seleccionada como auditor o auditado **SIN** modificar la selección que hubieramos realizado antes, es decir si teníamos a alguien ya seleccionado, seleccionar a otro más significa tener a ambos seleccionados.

Figura 36

Es obligatorio al menos seleccionar un auditor y al menos un auditado.

### Selección del tipo de auditoría

Para las auditorías normales, hay que escoger proyecto obligatoriamente, en el caso de ser auditorías especiales bastará con indicar que tipos escoger, para ello hay que desplegar el select que tenemos debajo de “Proyecto:”.

Para cambiar entre auditorías normales y especiales existe un botón de radio, marcado en rojo en la siguiente figura:

Figura 37

Cuando seleccionemos las auditorías especiales se nos deshabilitará la opción de elegir proyecto.

El resto de parámetros no requieren mención especial, si acaso la lista de distribución, que como explica en el rótulo de información, es una lista de emails separados por comas.

Los emails del auditor y auditado ya están incluidos, esta lista de distribución se usará en los siguientes casos:

- Cuando un aviso cuyo receptor no tenga email asignado.
- Cuando se envíe el aviso para visualizar los resultados de una auditoría ejecutada.

## Modificación de auditorías

Para acceder a la modificación, puede hacerse directamente desde el email enviado por la aplicación, o desde el menú: **AUDITORIA -> Modificar auditoría**

Antes de modificar una auditoría hay que seleccionarla, este proceso se explica en el apartado de **Selección previa de auditoría**. (Salvo que se acceda desde el aviso)

En este apartado será donde se planifique una auditoría o se invalide, así mismo se podrán modificar algunos de los datos de la misma, como la lista de distribución y los parámetros de previsiones.

Vemos en la siguiente figura la pantalla de modificación, donde disponemos de, en rojo, un apartado con los datos de la auditoría. En azul una zona que podemos modificar.

No obstante en este ejemplo, la auditoría está invalidada, por ello disponemos de un botón para revalidarla (marcada en rosa), si la validamos podremos cambiar la fecha de planificación y ejecutar la auditoría.

**Datos de la auditoría**

Datos de la auditoría	
Código de la auditoría:	2010-5-12-0001:QA1
Tipo de auditoría:	QA1
Estado de la auditoría:	Invalidada
Horas ejecutadas:	0
Número de avisos realizados:	1
Auditados:	Vicente GIL Zuazo - E806
Auditores QA:	Jose Antonio PERRA Dominguez - E550
Auditoría no realizada	

**Datos modificables**

Datos modificables:	
Horas planificadas	<input type="text" value="1"/>
Lista de distribución	<input type="text" value="juankarrey@gmail.com"/>
Número de personas que ejecutaron la auditoría	<input type="text" value="0"/>
Cambiar número de personas	<input type="text" value="0"/>
Auditoría no realizada	
Proyecto:	Cam 1 - 2007-557-55250
Riesgo previo de la auditoría:	Bajo
Cambiar riesgo previo:	<input type="button" value="Selección"/>
Fecha de invalidación:	<input type="text" value="12/05/2010"/>
La auditoría está invalidada, puede revalidarla.	
<input type="button" value="Revalidar Auditoría"/>	

Guardar Cancelar Terminado

Figura 38

Para planificar la auditoría le daremos al ícono de calendario y seleccionaremos la fecha que se considere oportuna, ó escribiremos directamente la fecha en el formato marcado, como se ve en la siguiente figura, de una auditoría válida:

## Datos modificables

Datos modificables:																																															
Horas planificadas	1																																														
Lista de distribución	juankarrey@gmail.com																																														
Número de personas que ejecutaron la auditoría	0																																														
Cambiar numero de personas	0																																														
Auditoría no realizada																																															
Fecha de planificación (dd/mm/yyyy)	<input type="text"/>																																														
Proyecto:	<input type="text"/>																																														
Riesgo previo de la auditoría:	<input type="text"/>																																														
Cambiar riesgo previo:	<input type="text"/>																																														
<input type="button" value="Guardar"/> <input type="button" value="Cancelar"/> <input type="button" value="Invalidar"/>																																															
<table border="1" style="width: 100px; text-align: center;"> <tr> <td></td> <td>Mayo</td> <td>2010</td> <td></td> </tr> <tr> <td>L</td><td>M</td><td>X</td><td>J</td><td>V</td><td>S</td><td>D</td></tr> <tr> <td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td></tr> <tr> <td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td></tr> <tr> <td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td></tr> <tr> <td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td></tr> <tr> <td>31</td><td></td><td></td><td></td><td></td><td></td><td></td></tr> </table>			Mayo	2010		L	M	X	J	V	S	D	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31						
	Mayo	2010																																													
L	M	X	J	V	S	D																																									
3	4	5	6	7	8	9																																									
10	11	12	13	14	15	16																																									
17	18	19	20	21	22	23																																									
24	25	26	27	28	29	30																																									
31																																															
<input type="button" value="Cerrar"/>																																															

Figura 39

## Ejecución de auditorías

Para acceder a la ejecución de una auditoría, puede hacerse directamente desde el email enviado por la aplicación, o desde el menú **AUDITORIA-> Ejecutar auditoría**

Antes de ejecutar una auditoría hay que seleccionarla, este proceso se explica en el apartado de **Selección previa de auditoría**. (Salvo que se acceda desde el aviso)

Una vez tenemos la auditoría a ejecutar, se nos mostrará una página inicial con la agenda de la auditoría, que nos indica un poco los datos de la auditoría a ejecutar.

Agenda de la auditoría	
Nombre del proyecto	2007-607 Expedientes ARI-ARCH
Jefe del proyecto	Raul NOVOA Minguet
Código de la auditoría	2010-5-12-0003:QA1
Tipo de auditoría	QA1
Grupo QA (Auditores)	- Ignacio VILALTA Esteban
Grupo de Auditados	- Jose Antonio PERNIA Dominguez
Lista de distribución	juankarrey@gmail.com

[Ejecutar auditoria](#)

Figura 40

En la parte superior del contenido, vemos una serie de pestañas, estas pestañas corresponden a los procesos o temáticas en los cuales esta subdividido el cuestionario global.

A parte de estas pestañas disponemos de unos botones que nos guiaran secuencialmente por los procesos 'Siguiente' y 'Anterior'.

Una vez queramos dar por finalizada una auditoría deberemos darle a 'Terminar', si la auditoría esta completa nos llevará al resumen de la auditoría, si no lo está no hará nada, si queremos continuar más tarde deberemos darle a 'Guardar y salir'.

Vemos los botones en la siguiente figura (abajo a la derecha), que se encuentran en cualquiera de las pestañas salvo en la de la Agenda, donde disponemos de un botón "Ejecutar auditoría" que nos lleva a la primera página (pestaña).

Gestión de Riesgos							
Referencia	Pregunta	Referencias doc.	Tipo QA	Si	No	No Aplica	Observaciones
659	¿ Se han definido los riesgos de la organización ?	Riesgos de la organización	QA1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
660	¿ Se han evaluado los riesgos ? (Asignar impacto y probabilidad)	Riesgos del proyecto	QA1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
661	¿ Se han definido las acciones de mitigación para cada uno de los riesgos ?	Riesgos del proyecto	QA1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
662	¿ Se han definido los planes de contingencia para cada uno de los riesgos ?	Riesgos del proyecto	QA1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
663	¿ Se hace el seguimiento de los riesgos del proyecto ? (Solo tipo II y Tipo III)	Bitácora Riesgos del proyecto	QA1 QA2 QA3	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
672	¿ Las acciones de mitigación aparecen reflejadas en el plan del proyecto ? (Solo aquellas acciones que puedan estar en el plan)		QA1 QA2 QA3	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
666	¿ Se han definido los riesgos del proyecto en el inicio del proyecto ?	Riesgos del proyecto	QA1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	

[Siguiente](#) [Anterior](#) [Terminar](#) [Guardar y salir](#)

**Figura 41**

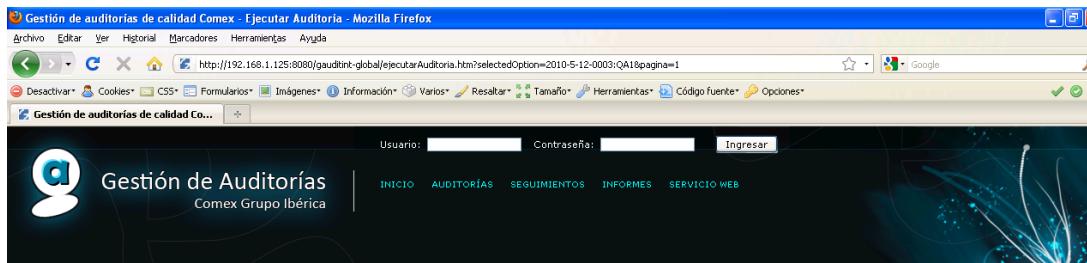
Para realizar la ejecución deberemos contestar Si, No ó No Aplica a todas las preguntas, rellenando en cada caso las observaciones que se consideren oportunas.

Una vez hemos llenado **todas** las preguntas de **todos** los procesos, podremos darle a terminar y pasaremos al resumen de auditoría, para completar finalmente la auditoría marcando la gravedad de las no conformidades e indicando sus causas.

## Resumen de auditoría

A este sub-apartado de la ejecución de auditoría se accede después de terminar de contestar el cuestionario de una auditoría.

En la siguiente figura podemos ver que debemos llenar ocho no conformidades, escogiendo su gravedad (grave ó leve) y escribir sus causas.



#### Completar auditoría

##### Descripción de las No Conformidades

Seleccione el grado de las no conformidades y sus causas a fin de terminar la auditoría.

Código	Descripción de la no conformidad	Tipo	Causas de la no conformidad
165	No se han definido los planes de contingencia para cada uno de los riesgos	Grave: <input type="radio"/>	Leve: <input type="radio"/>
166	No se ha utilizado la plantilla de recogida y toma de requisitos	Grave: <input type="radio"/>	Leve: <input type="radio"/>
167	No se han obtenido y analizado los requisitos del sistema por parte del usuario	Grave: <input type="radio"/>	Leve: <input type="radio"/>
168	No se han registrado las peticiones de cambio de requisitos en Tareeitor	Grave: <input type="radio"/>	Leve: <input type="radio"/>
169	No se ha efectuado una toma de decisiones formal en el proyecto	Grave: <input type="radio"/>	Leve: <input type="radio"/>
170	No se ha planificado alguna PR en el plan de proyecto	Grave: <input type="radio"/>	Leve: <input type="radio"/>
171	No existe una consultoría previa del sistema de la información	Grave: <input type="radio"/>	Leve: <input type="radio"/>
172	Los contactos no tienen asignado el cargo y la responsabilidad que tienen dentro del proyecto	Grave: <input type="radio"/>	Leve: <input type="radio"/>

**Figura 42**

Una vez completado, pulsamos el botón de "Completar auditoría", la cual nos llevará al resumen final (ya completado), donde se nos mostrará la información general de la auditoría.

En la siguiente figura observamos este resumen, con dos cálculos del estado de un proyecto a partir del número de no conformidades de cada tipo de gravedad, así para el ejemplo tenemos 4 leves y 4 graves, lo cual nos lleva a un estado Bueno (según las leves) y Malo (según las graves) lo cual en definitiva infiere en un estado Malo.

Como el proyecto solo tiene una auditoría, los resultados para el proyecto y para la auditoría son los mismos, si hubiera más de una auditoría para el proyecto, posiblemente el proyecto tuviera mayor número de no conformidades.

Resumen de auditoría

Descripción de las revisiones			
Código	Descripción de la no conformidad	Tipo	Causas
165	No se han definido los planes de contingencia para cada uno de los riesgos	Grave	Faltan planes para algunos riesgos
166	No se ha utilizado la plantilla de recogida y toma de requisitos	Leve	No se utilizó plantilla
167	No se han obtenido y analizado los requisitos del sistema por parte del usuario	Leve	No se realizó
168	No se han registrado las peticiones de cambio de requisitos en Tareitor	Grave	Hubo algunas peticiones que no estan en Tareitor
169	No se ha efectuado una toma de decisiones formal en el proyecto	Leve	La toma de decisiones quedó relegada
170	No se ha planificado alguna PR en el plan de proyecto	Leve	No está planificada
171	No existe una consultoría previa del sistema de la información	Grave	No hubo consultores dispuestos a realizarla
172	Los contactos no tienen asignado el cargo y la responsabilidad que tienen dentro del proyecto	Grave	No se han asignado

Estado del proyecto: '2007-607 Expedientes ARI-ARCH' (codigo: 2007-607-55320): Malo

Tipo	#No conf	Estado (del proyecto)
Grave	4	Malo
Leve	4	Bueno

Resultados generales de la auditoría.

Tipo	#No conf	Estado (de la auditoría)
Grave	4	Malo
Leve	4	Bueno

GESTIÓN DE AUDITORÍAS | Comex Grupo Ibérica

Figura 43

## Resultado de auditoría

Para acceder a la visualización de los resultados de una auditoría, ir al menú :  
**AUDITORIA -> Resultado auditoría**

Antes de acceder al resultado de una auditoría hay que seleccionarla, este proceso se explica en el apartado de **Selección previa de auditoría**. (Salvo que se acceda desde el aviso)

Se puede acceder desde la selección, tanto al Resumen de auditoría como a la visualización de las respuestas, en el primero se accederá al resumen final de auditoría explicado en la sección anterior (figura 43), mientras que en las respuestas se mostrarán todas las respuestas agrupadas según la contestación (Afirmativas, Negativas, No aplica) como vemos en la siguiente figura.

Negativas No Aplica Afirmativas

Preguntas con respuesta : No

Ref.	Pregunta	Proceso	Ref. Documentales	Tipo QA	Respuesta	Observaciones
663	¿Se hace el seguimiento de los riesgos del proyecto ? (Solo tipo II y Tipo III)	Gestión de Riesgos	Bitacora Riesgos del proyecto	QA1 QA2 QA3	No	
558	¿Se ha utilizado la plantilla de recogida y toma de requisitos?	Requisitos	Doc. toma de requisitos	QA1	No	
560	¿Se han obtenido y analizado los requisitos del sistema por parte del usuario ?	Requisitos	Doc. Análisis	QA1	No	
579	¿Se han registrado las peticiones de cambio de requisitos en Tareitor?, (Sólo aquellas aprobadas y con suficiente entidad)	Requisitos	Doc. Análisis Gestión de Requisitos en tareitor	QA1 QA2 QA3	No	
455	¿ Se ha efectuado una toma de decisiones formal en el proyecto ?	DAR	Análisis de decisiones	QA1 QA2 QA3	No	
382	¿ Se ha planificado alguna PR en el plan de proyecto ?	Peer Review	Plan de proyecto	QA1 QA2 QA3	No	
469	¿ Existe una consultoría previa del sistema de la información ?	Anteproyecto	Doc. Consultoría	QA1	No	
492	¿Los contactos tienen asignado el cargo y la responsabilidad que tienen dentro del proyecto ?	Gestión de Proyectos	Site del proyecto en sharepoint	QA1	No	

Figura 44

## Seguimiento de auditoría

El seguimiento de una auditoría sirve para agrupar las respuestas negativas de una auditoría de manera que se asignen acciones correctivas para su corrección por parte de un personal responsable antes de una fecha estimada, todo esto puede modificarse desde el seguimiento de una auditoría, al que se accede desde el menú: **SEGUIMIENTO -> Seguimiento auditoría**

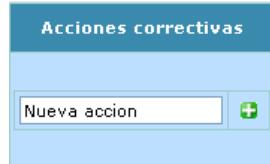
En la siguiente figura vemos el seguimiento de la auditoría, observamos unos cuadros de información, en los azules al hacer click sobre ellos mostramos la información extendida de una revisión, lo cual nos enseña la información relevante para saber qué revisión estamos modificando. Mientras que en el cuadro de información rosa, nos muestra un bloque de información flotante como el que vemos en la figura 46.

Figura 45

Figura 46

## Rellenando el seguimiento

Para seleccionar acciones correctivas o de cierre, hay que escribir la acción correctiva a asignar y pulsar al botón [+] como vemos en la figura

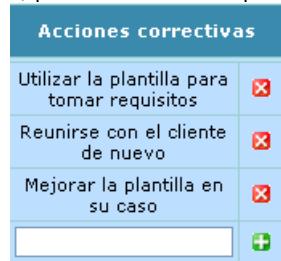


Acciones correctivas

Nueva acción +

Figura 47

Una vez hemos añadido algunas acciones, podemos borrarlas pulsando en la [x] como se ve en esta figura



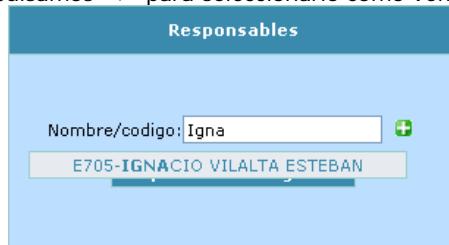
Acciones correctivas

Utilizar la plantilla para tomar requisitos	x
Reunirse con el cliente de nuevo	x
Mejorar la plantilla en su caso	x

+ +

Figura 48

La asignación de personal responsable es similar al caso de nueva auditoría, esta vez sólo disponemos del input con sugerencias, de esta forma escribimos el código de personal ó el nombre ó uno de los apellidos, ó parte del nombre/apellido, y pulsamos "+" para seleccionarlo como vemos en la figura.



Responsables

Nombre/código: Igná +

E705-IGNACIO VILALTA ESTEBAN

Figura 49

Una vez seleccionado personal, podemos desseleccionarlo pulsando la "x", del mismo modo que para las acciones correctivas.



Responsables

Nombre/código: +

Responsables asignados

Manuel MAZARICO Torres	x
Jose Antonio PERNA Dominguez	x
Ignacio VILALTA Esteban	x

Figura 50

La revisión puede abrirse/cerrarse mediante el botón situado más a la izquierda de una revisión, si está abierta, es susceptible a cambios, mientras que si está cerrada, no podrán realizarse cambios sobre ella.

También puede modificarse su estado, entre "Pendiente, en curso, finalizada, anulada", en caso de estar pendiente ó en curso, ésta revisión aparecerá en el Informe de revisiones.

Así mismo también podrá modificarse la fecha mediante el calendario de más a la derecha de la figura 45, que observamos más en detalle en la siguiente figura.



Figura 51

## Seguimiento de proyecto

Del mismo modo que para la auditoría, también es posible acceder al seguimiento de un proyecto, en este ámbito sólo es posible visualizar los resultados de un seguimiento, sirve para la consulta de los auditados.

Se puede acceder desde el menú **SEGUIMIENTO -> Seguimiento** de proyecto, o mediante el link enviado por email (avisos). En el primer caso deberemos realizar la selección previa de proyecto, muy similar a la de la auditoría explicada en la sección "**Selección previa de auditoría**"

Vemos en la siguiente figura un ejemplo de seguimiento de proyecto.

Nº	Código	Proyecto	Auditados	Tipo QA	Fecha de revisión	Auditores QA	Descripción de la no conformidad	Proceso
4	169	2007-607 Expedientes ARI-ARCH	Jose Antonio PERNA Dominguez	QA1	13/05/2010	Ignacio VILALTA Esteban	No se ha efectuado una toma de decisiones formal en el proyecto	DAR
Estado	Acciones correctivas	Acciones de cierre	Responsables	Fecha final	Estado	Fecha de cierre	Causas	
Abierta	Efectuar una toma de decisiones formal		Manuel MAZARICO Torres	19/05/2010	Pendiente		La toma de decisiones quedó relegada	
Nº	Código	Proyecto	Auditados	Tipo QA	Fecha de revisión	Auditores QA	Descripción de la no conformidad	Proceso
5	170	2007-607 Expedientes ARI-ARCH	Jose Antonio PERNA Dominguez	QA1	13/05/2010	Ignacio VILALTA Esteban	No se ha planificado alguna PR en el plan de proyecto	Peer Review
Estado	Acciones correctivas	Acciones de cierre	Responsables	Fecha final	Estado	Fecha de cierre	Causas	
Abierta	Planificar al menos una PR		Manuel MAZARICO Torres	19/05/2010	Pendiente		No está planificada	
Nº	Código	Proyecto	Auditados	Tipo QA	Fecha de revisión	Auditores QA	Descripción de la no conformidad	Proceso
6	171	2007-607 Expedientes ARI-ARCH	Jose Antonio PERNA Dominguez	QA1	13/05/2010	Ignacio VILALTA Esteban	No existe una consultoría previa del sistema de la información	Anteproyecto
Estado	Acciones correctivas	Acciones de cierre	Responsables	Fecha final	Estado	Fecha de cierre	Causas	
Abierta	Realizar la consultoría		Jose Antonio PERNA Dominguez	19/05/2010	Pendiente		No hubo consultores dispuestos a realizarla	
Nº	Código	Proyecto	Auditados	Tipo QA	Fecha de revisión	Auditores QA	Descripción de la no conformidad	Proceso
7	172	2007-607 Expedientes ARI-ARCH	Jose Antonio PERNA Dominguez	QA1	13/05/2010	Ignacio VILALTA Esteban	Los contactos no tienen asignado el cargo y la responsabilidad que tienen dentro del proyecto	Gestión de Proyectos
Estado	Acciones correctivas	Acciones de cierre	Responsables	Fecha final	Estado	Fecha de cierre	Causas	
Cerrada	No puede corregirse.		Jose Antonio PERNA Dominguez		Finalizada	13/05/2010	No se han asignado	

Figura 52

## Selección previa de auditoría

La selección previa de auditoría en un sistema mediante el cual seleccionamos la auditoría que deseamos para realizar una acción (Modificarla, ejecutarla, acceder a su seguimiento, etc.).

Podemos seleccionar auditorías normales (Que requieren proyecto y se accede a ella a través de su proyecto) o especiales (No requieren proyecto).

Cuando seleccionamos el proyecto deseado el segundo desplegable se llenará automáticamente, por lo general sólo nos mostrará aquellos proyectos y auditorías sobre los que se **PUEDA** acceder a la acción requerida.

Por ejemplo, no dejará ejecutar una auditoría que ya ha sido ejecutada, o que aún no ha sido planificada. Tampoco dejará acceder al seguimiento a aquellas que no estén completadas o estén invalidadas.

De esta forma se crea automáticamente un filtro que nos evita desplegables demasiado grandes.

Vemos en la figura un ejemplo de selección.

Seleccione auditoría

Tipos normales

Proyecto: Seleccionar

Auditorías: Primero elija proyecto

Tipos Especiales (no requieren proyecto)

Auditorías: Seleccionar

Modificar

Figura 53

Seleccione auditoría

Tipos normales

Proyecto: 2007-607-55320 - 2007-607 Expedientes ARI-ARCH

Auditorías: 2010-5-12-0003:QA1  
2010-5-12-0003:QA1  
2010-5-12-0004:QA2  
2010-5-12-0005:QA3

Tipos Especiales (no requieren proyecto)

Auditorías: Seleccionar

Modificar

Figura 54

En la primera de las figuras (53), aún no se ha seleccionado proyecto, mientras que en la segunda sí y se ha llenado el desplegable.

## Informes y métricas

Existen diversos informes y métricas en el sistema, explicaremos uno a uno todos ellos. Para el acceso a los diferentes informes y métricas hay que acceder al menú **INFORMES -> \***, donde el \* corresponde a cualquiera de los submenús mostrados.

### Informe general de auditorías

En este informe podremos visualizar un resumen de todas las auditorías, tenemos en la parte superior opciones para poder filtrar el informe por diversas formas:

**Jefe de proyecto (rojo)** -> Muestra las auditorías realizadas sobre proyectos donde la persona seleccionada es el jefe.

**Tipo de proyecto (azul)** -> Muestra las auditorías realizadas sobre proyectos del tipo seleccionado.

Estos dos anteriores son acumulables, por lo que resultaría en: Auditorías realizadas sobre proyectos del tipo seleccionado en los que la persona seleccionada es jefe.

Para seleccionar al jefe tenemos un desplegable normal ó un input con sugerencias como los vistos en "Crear auditoria" y "Seguimiento de auditoría".

**Por proyecto (rosa)** -> Muestra las auditorías realizadas sobre el proyecto seleccionado.

**Estados (naranja)** -> Muestra las auditorías que están en un estado de los seleccionados, sirve para no mostrar las auditorías invalidas, o recién creadas. Además nos sirve para poder visualizar todas las auditorías caducadas, o las planificadas para su ejecución.

Vemos en la siguiente figura las opciones de filtro encuadradas.

The screenshot shows the 'Informe general de auditorías' page. At the top, there are three main filter sections: 'Por jefe y/o tipo de proyecto' (highlighted with a red box), 'Filtrar por proyecto' (highlighted with a pink box), and 'Estados que desea mostrar' (highlighted with an orange box). The 'Por jefe y/o tipo de proyecto' section contains a radio button for 'Por jefe y/o tipo de proyecto' and a dropdown menu for 'Filtrar por jefe'. The 'Filtrar por proyecto' section contains a radio button for 'Filtrar por proyecto' and a dropdown menu for 'No filtrar por proyecto'. The 'Estados que desea mostrar' section contains a list of checkboxes for audit states: Creación, Caducada, En curso, Invalidada, Completada, and Planificada. Below these sections is a 'Ver resumen de auditorías' button. The main content area is titled 'Informe general de Auditorías' and contains a table with columns: Tipo de revisión, Código, Estado, H. plan., H. ejec., #No conf. Grave, #No conf. Leve, Riesgo previo, Riesgo posterior, Fecha planificada, Fecha de revisión, Auditores QA, Auditados, and a status column. One row in the table is highlighted with a blue box and labeled 'Terminado'.

El informe final lo podemos ver en detalle en la siguiente figura explicativa (marcado en rojo el informe).

Estado que desea mostrar

Marcar todos - Desmarcar

- Creación
- Caducada
- En curso
- Invalidada
- Completada
- Planificada

Ver resumen de auditorías

**Informe general de Auditorías**

Tipo de revisión	Código	Estado	H. plan.	H. ejec.	#No conf. Grave	#No conf. Leve	Riesgo previo	Riesgo posterior	Fecha planificada	Fecha de revisión	Auditores QA	Auditados
QA3	2010-5-12-0003:QA3	Creación	1	0	0	0	Bajo				Jose Antonio PERRA Dominguez	Vicente GIL Zuazo
QA2	2010-5-12-0004:QA2	Creación	0	0	0	0					Ignacio VILALTA Esteban	Jose Antonio PERRA Dominguez
QA3	2010-5-12-0004:QA3	Creación	0	0	0	0					Ignacio VILALTA Esteban	Ignacio VILALTA Esteban
QA3	2010-5-12-0005:QA3	Creación	0	0	0	0					Ignacio VILALTA Esteban	Jose Antonio PERRA Dominguez
QA1	2010-5-12-0001:QA1	Creación	1	0	0	0	Bajo				Jose Antonio PERRA Dominguez	Vicente GIL Zuazo
QA1	2010-5-12-0002:QA1	Caducada	0	0	0	0		01/05/2010			Ignacio VILALTA Esteban	Ignacio VILALTA Esteban
PPQA	2010-5-12-0001:PPQA	Caducada	0	0	0	0		10/05/2010			Ignacio VILALTA Esteban	Ignacio VILALTA Esteban
QA2	2010-5-12-0002:QA2	Caducada	1	0	0	0	Bajo		11/05/2010		Jose Antonio PERRA Dominguez	Vicente GIL Zuazo
QA1	2010-5-12-0003:QA1	Completada	0	1	4	4		Bajo	28/05/2010	13/05/2010	Ignacio VILALTA Esteban	Jose Antonio PERRA Dominguez
PPQA	2010-5-12-0002:PPQA	Planificada	0	0	0	0			30/05/2010		Ignacio VILALTA Esteban	Ignacio VILALTA Esteban
QA2	2010-5-12-0003:QA2	Planificada	0	0	0	0			31/05/2010		Ignacio VILALTA Esteban	Ignacio VILALTA Esteban

GESTIÓN DE AUDITORÍAS | Comex Grupo Ibérica

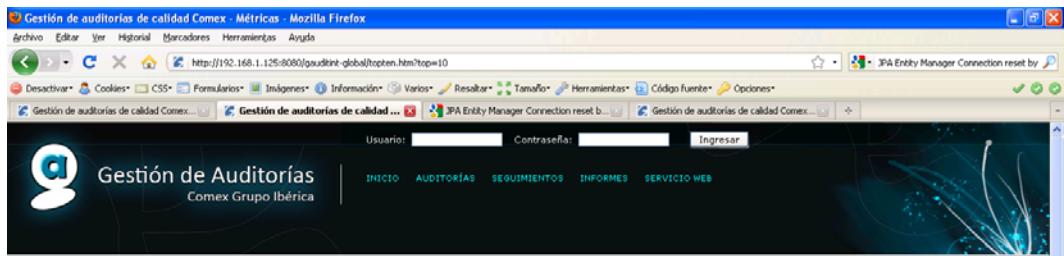
Figura 55

## Top ten de preguntas contestadas negativamente

Se trata de un informe en el cual se muestra el ranking de preguntas contestadas el mayor número de veces de forma negativa, así mismo también se muestra este mismo ranking en porcentaje de preguntas negativas sobre el total de respuestas.

Para poder configurar el número de preguntas mostradas, podemos hacerlo modificando la URL mostrada en la parte superior, modificando el número 10, por el número que deseemos.

Vemos en la siguiente figura el ranking en número.



Ranking de preguntas contestadas negativamente el mayor número de veces.

Posición	Ref.	Pregunta	#Respuestas Negativas
1	382	¿ Se ha planificado alguna PR en el plan de proyecto ? Mostrar causas (1)	1
2	455	¿ Se ha efectuado una toma de decisiones formal en el proyecto ? Mostrar causas (1)	1
3	469	¿ Existe una consultoría previa del sistema de la información ? Mostrar causas (1)	1
4	492	¿ Los contactos tienen asignado el cargo y la responsabilidad que tienen dentro del proyecto ? Mostrar causas (1)	1
5	550	¿ Se ha utilizado la plantilla de recogida y toma de requisitos ? Mostrar causas (1)	1
6	560	¿ Se han obtenido y analizado los requisitos del sistema por parte del usuario ? Mostrar causas (1)	1
7	579	¿ Se han registrado las peticiones de cambio de requisitos en Tareector ?, (Sólo aquellas aprobadas y con suficiente entidad) Mostrar causas (1)	1
8	663	¿ Se hace el seguimiento de los riesgos del proyecto ? (Solo tipo II y Tipo III) Mostrar causas (1)	1

Ranking de preguntas contestadas negativamente en porcentaje.

Posición	Ref.	Pregunta	#Respuestas Negativas	Total de respuestas	% de Respuestas negativas
Terminado					

Figura 56

Si pulsamos en 'Mostrar causas' se extenderán las causas que han llevado a responder negativamente a esa pregunta.

En la siguiente figura vemos el ranking en porcentaje.

Posición	Ref.	Pregunta	Mostrar causas (1)
7	579	¿ Se han registrado las peticiones de cambio de requisitos en Tareector ?, (Sólo aquellas aprobadas y con suficiente entidad)	Mostrar causas (1)
8	663	¿ Se hace el seguimiento de los riesgos del proyecto ? (Solo tipo II y Tipo III)	Mostrar causas (1)

Ranking de preguntas contestadas negativamente en porcentaje.

Posición	Ref.	Pregunta	#Respuestas Negativas	Total de respuestas	% de Respuestas negativas
1	469	¿ Existe una consultoría previa del sistema de la información ? Mostrar causas (1)	1	2	50.0
2	492	¿ Los contactos tienen asignado el cargo y la responsabilidad que tienen dentro del proyecto ? Mostrar causas (1)	1	2	50.0
3	550	¿ Se ha utilizado la plantilla de recogida y toma de requisitos ? Mostrar causas (1)	1	2	50.0
4	560	¿ Se han obtenido y analizado los requisitos del sistema por parte del usuario ? Mostrar causas (1)	1	2	50.0
5	382	¿ Se ha planificado alguna PR en el plan de proyecto ? Mostrar causas (1)	1	3	33.333333333333336
6	455	¿ Se ha efectuado una toma de decisiones formal en el proyecto ? Mostrar causas (1)	1	3	33.333333333333336
7	579	¿ Se han registrado las peticiones de cambio de requisitos en Tareector ?, (Sólo aquellas aprobadas y con suficiente entidad) Mostrar causas (1)	1	3	33.333333333333336
8	663	¿ Se hace el seguimiento de los riesgos del proyecto ? (Solo tipo II y Tipo III) Mostrar causas (1)	1	3	33.333333333333336
9	383	¿ Se ha ejecutado la PR conforme a la planificación ? No hay causas	0	3	0.0
10	304	¿ Se ha enviado con antelación el producto a revisar al revisor ? No hay causas	0	3	0.0

GESTIÓN DE AUDITORÍAS | Comex Grupo Ibérica

Terminado

Figura 57

## Informe de revisiones

En el informe de revisiones podremos observar las revisiones pendientes ó en curso para poder acceder al seguimiento de las mismas para poder finalizarlas, hasta estar finalizadas ó anuladas.

En la siguiente figura observamos un ejemplo del informe de revisiones, donde se nos muestran unas pequeñas métricas acompañadas del listado total organizado por proyectos de las revisiones.

The screenshot shows the 'Informe de revisiones' (Audit Report) page. At the top, there is a summary table for metrics:

Estado	Gravedad	Cantidad	Porcentaje
Pendiente	Mayor	3	42.857142857142854 %
Pendiente	Menor	4	57.142857142857146 %

Below this is a list of audit items for project 2007-607, grouped by audit type (ARI-ARCH):

Estado	Gravedad	Auditoría	Seguimiento
Pendiente	Mayor	2010-5-12-0003-QA1	Ir al seguimiento
Pendiente	Menor	2010-5-12-0003-QA1	Ir al seguimiento
Pendiente	Menor	2010-5-12-0003-QA1	Ir al seguimiento
Pendiente	Mayor	2010-5-12-0003-QA1	Ir al seguimiento
Pendiente	Menor	2010-5-12-0003-QA1	Ir al seguimiento
Pendiente	Menor	2010-5-12-0003-QA1	Ir al seguimiento
Pendiente	Mayor	2010-5-12-0003-QA1	Ir al seguimiento

Figura 58

## Media de no conformidades por auditoría

En este informe se calcula la media de respuestas negativas por auditoría, agrupados según el criterio escogido: Por proceso ó por tipo de proyecto. También existe la opción de no agrupar, seleccionando "Por tipo de auditoría", dado que siempre las agrupa por tipo de auditoría y por el criterio escogido.

Es decir, si agrupamos por proceso tendremos: (Similar a por tipo de proyecto).

Tipo auditoria 1 – Proceso 1 – Media  
Tipo auditoria 1 – Proceso 2 – Media

Tipo auditoria 2 – Proceso 1 - Media  
Tipo auditoria 1 – Proceso 2 – Media

Si agrupamos por tipo auditoría:

Tipo auditoria 1 –Media  
Tipo auditoria 2 –Media

También deberemos escoger dos fechas (inicio y fin) que por defecto marcarán el último año, que indica que se recojan en el informe todas las auditorías realizadas entre esas fechas.

Vemos en la imagen un ejemplo:

**Método de agrupación:**

- Por tipo de auditoría
- Por tipo de proyecto
- Por proceso

Fecha inicial: 13/05/2009

Fecha final: 31/05/2010

[Ver métricas](#)

**Medias de no conformidades Grave por QA.:**

Tipo QA	Proceso	Tipo Proyecto	#No conf	Número de auditorías	Media
QA1	Gestión de Riesgos		1	1	1.0
QA1	Requisitos		1	1	1.0
QA1	Anteproyecto		1	1	1.0
QA1	Gestión de Proyectos		1	1	1.0

**Medias de no conformidades Leve por QA.:**

Tipo QA	Proceso	Tipo Proyecto	#No conf	Número de auditorías	Media
QA1	Requisitos		2	1	2.0
QA1	DAR		1	1	1.0
QA1	Peer Review		1	1	1.0

**Figura 59**

## Proyectos y auditorías caducados

Este apartado es una mezcla entre métrica e informe, nos muestra el número de proyectos y auditorías que han caducado entre dos fechas escogidas, en número y en porcentaje sobre el total.

Posteriormente nos sale un listado con las auditorías caducadas para cada proyecto permitiendo el acceso a su ejecución.

Lo vemos en la siguiente figura.

**Fechas**

Fecha inicial: 13/05/2009

Fecha final: 31/05/2010

[Ver proyectos/auditorías caducados](#)

**Estadísticas**

**Proyectos**  
Número total de proyectos caducados : 2  
Total de proyectos planificados : 3  
Porcentaje de proyectos planificados y no auditados : 66.6666666666667%

**Auditorías**  
Número total de auditorías caducadas : 3  
Total de auditorías planificadas : 6  
Porcentaje de auditorías planificadas y no auditadas : 50.0%

**Listado de proyectos**

**2007-557-55250 - Cam 1**  
**Auditorías caducadas para éste proyecto:**  
Auditoría caducada: 2010-5-12-0002:QA2 [Ejecutar](#)

**2007-999-55180 - Trabajos generales de Diseño**  
**Auditorías caducadas para éste proyecto:**  
Auditoría caducada: 2010-5-12-0002:QA1 [Ejecutar](#)

**Listado de auditorías especiales caducadas**  
Auditoría caducada: 2010-5-12-0001:PPQA [Ejecutar](#)

**Figura 60**

Observamos que estan ordenadas por proyecto, salvo en el caso de las especiales que están situadas al final del informe.

### Estadísticas para las auditorías

En este informe/métrica nos muestra el número de auditorías ordenadas por un intervalo de meses (por defecto semestral) y agrupadas por su estado (recien creada, planificada, invalidada, caducada, realizada ó en curso).

Hay que seleccionar dos fechas (inicio y final) y un intervalo (en número de meses) por ejemplo si ponemos 1, el informe marcará el número de auditorías que se encuentran en cada estado por mes, si ponemos 2 bimensual, 6 semestral, 12 anual, si ponemos 0 o un número erroneo, se pondrá por defecto semestral.

Lo vemos en la siguiente figura:

#### Estadísticas de auditorias

##### Fechas

Fecha inicial:

13/05/2009

Fecha final :

01/06/2010

Intervalo (en meses) de agrupación: 0

[Ver proyectos/auditorías realizados](#)

#### Auditorías caducadas (estados)

Fecha inicial	Fecha final	Realizadas	Caducadas	Invalidadas	En curso	Planificadas
13/05/2009	12/11/2009	0	0	0	0	0
13/11/2009	12/05/2010	0	3	0	0	0
13/05/2010	12/11/2010	1	0	0	0	2

**Figura 61**

## Servicios Web

Los servicios web sirven para obtener personal de la empresa e insertarlo en la base de datos local, así como la obtención de los proyectos a auditar, se accede a ellos a través del código de recurso.

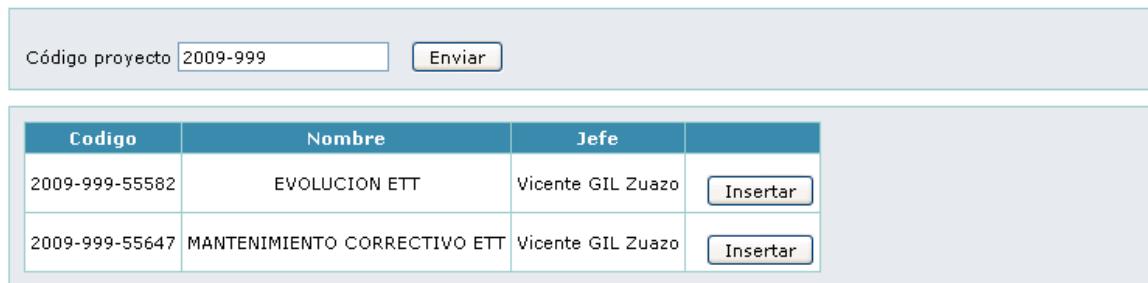
Basta con insertar el código de recurso y darle a enviar.

Para el caso del personal, en cualquier punto de la aplicación donde se requiera personal nuevo, basta con poner el código de recurso en el input con sugerencias y darle a enviar, el propio sistema buscara al personal en el servicio web, el principal problema de esta característica, es que el personal no dispone de email, por lo que **no le llegarán** los avisos para la **planificación de la auditoría**.

Para los proyectos, es posible que existan varias Operaciones de trabajo (OT) con el mismo código de recurso, por ello el sistema le permitira escoger entre ellas una, diferenciandose principalmente en el nombre.

Vemos en la siguiente figura un ejemplo del uso del servicio web:

### Buscar proyecto por Servicio Web.



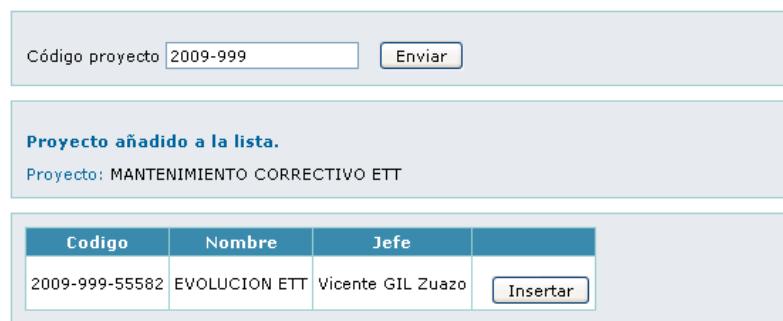
Código	Nombre	Jefe	
2009-999-55582	EVOLUCION ETT	Vicente GIL Zuazo	<input type="button" value="Insertar"/>
2009-999-55647	MANTENIMIENTO CORRECTIVO ETT	Vicente GIL Zuazo	<input type="button" value="Insertar"/>

Figura 62

En el ejemplo salen dos proyectos con el mismo código de recurso (2009-999), la otra parte del código es un subcódigo generado por el sistema para poder diferenciarlos internamente.

Dandole a "Insertar" en aquel que deseemos, lo introduciremos en la base de datos local, quedando fuera los restantes como se puede ver en la figura siguiente:

### Buscar proyecto por Servicio Web.



Código	Nombre	Jefe	
2009-999-55582	EVOLUCION ETT	Vicente GIL Zuazo	<input type="button" value="Insertar"/>

Figura 63

## Resumen de la parte privada

Para acceder a la parte privada de administración, debemos introducir nuestro usuario y contraseña en la parte superior de la pantalla, en la cabecera como vemos en la siguiente figura marcado en rojo.

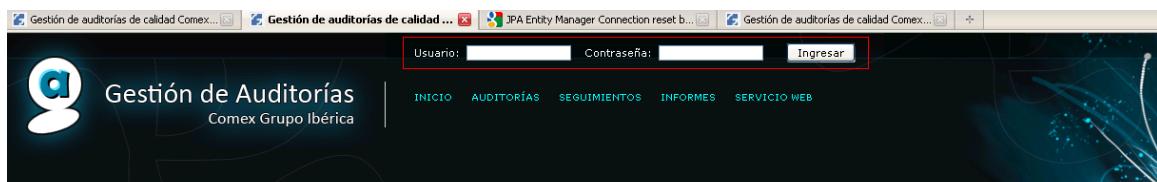


Figura 64

Desde la parte privada tenemos acceso a toda la parte pública y además al sistema de administración y configuración del sistema de gestión. Vemos en la siguiente figura que los menús ahora tienen más opciones, los de administración situados a la derecha del todo, que se explican brevemente a continuación.



Figura 65

Vemos también que ahora en lugar del formulario de ingreso a administración disponemos de un botón para salir de la administración, lo que equivale a cerrar la sesión.

La sesión caduca a los 30 minutos de inactividad.

Resumen de la funcionalidad organizada en los menús, que corresponde a la información disponible en la página de inicio de administración.

### ADMIN. ENTIDADES

En este menú se administran las entidades básicas presentes en la auditoría, que son las Preguntas, Procesos, Proyectos y Personal.

- Preguntas:  
Ordenadas por proceso en pestañas, sirve para crear preguntas. OJO! La modificación de una pregunta no interfiere en las auditorías pasadas, pero si en las métricas, dado que se crea una nueva pregunta y se da de baja la modificada.
- Procesos:  
La modificación del nombre de un proceso modifica las auditorías pasadas (Los nombres de los procesos). OJO! Si se da de baja un proceso, también lo hacen sus preguntas!
- Proyectos:  
La administración de los proyectos es simplemente para dejar como "No auditabile" un proyecto cuando éste no va a recibir más auditorías, por lo que no se podrán crear auditorías sobre él, pero si consultar las realizadas.
- Personal:

Es muy importante que todo el personal tenga email, dado que si ésta persona ha de recibir algún tipo de aviso, éste será enviado a la lista de distribución completa.

## ADMIN. TIPOS

En este apartado están otras entidades que corresponden a los tipos de una entidad (tipo de auditoría, de no conformidad y de estado).

- Tipo de Auditoría:

Corresponde al tipo de una auditoría (QA1, QA2, QA3, PPQA), se pueden crear distintos tipos y requerirán o no proyecto. Las que no requieran proyecto irán seleccionadas por defecto en la creación de una auditoría.

- Tipo de Estado de proyecto:

Los estados son la consecuencia directa de tener varias no conformidades de cierta gravedad, en principio son Bueno/Regular/Malo, y cada uno conveva su riesgo Bajo/Medio/Alto, se pueden crear más tipos de gravedad. La descripción corresponde al texto mostrado en la parte pública de la aplicación.

- Tipo de No Conformidad:

Equivalen a la gravedad de una respuesta negativa (Grave/Leve ó Mayor/Menor según cómo se prefiera nombrarlo), de nuevo la descripción corresponde con el texto mostrado en la parte pública de la aplicación.

## ADMIN. PARAMETROS

Aquí se administran diversos parámetros que se especificaron como configurables, por ejemplo el número de días previos o posteriores a una fecha para realizar un aviso. Así como la definición de la fórmula de cálculo con la que se calcula el estado de un proyecto/auditoría basándose en el número de no conformidades de un tipo de gravedad.

- Tiempos:

Días configurables y número de avisos para la planificación.

- Fórmula de cálculo:

Imprescindible para establecer la fórmula, si dos estados tienen el mismo número de no conformidades asignado para el mismo tipo de no conformidad, el sistema elegirá uno de forma aleatoria si se da que ambos se cumplen.

## Administración de preguntas

Para acceder a la administración de preguntas ir al menú: **ADMIN. ENTIDADES -> Gestión de preguntas**

Aquí dispondremos de un listado de las preguntas actuales (en la parte inferior), ordenadas por proceso (pestañas) y un formulario pequeño para la inserción de nuevas preguntas (en la parte superior).

En el listado de preguntas podremos modificar todo lo referente a la pregunta, así como la ordenación de las mismas en el cuestionario.

Al modificar una pregunta, la anterior existente se dará de baja (o se borrará si no se ha usado nunca) y se creará una nueva versión de la pregunta (Salvo que solo se modifique la descripción, la optionalidad ó el orden de la pregunta en el cuestionario).

Vemos en la siguiente figura toda la funcionalidad que explicamos a continuación

The screenshot shows the 'Gestión de auditorías de calidad Comex - Gestión de preguntas' application. At the top, there is a navigation bar with links like 'INICIO', 'AUDITORÍAS', 'SEGUIMIENTOS', 'INFORMES', 'SERVICIO WEB', 'ADMIN. ENTIDADES', 'ADMIN. TIPOS', and 'ADMIN. PARÁMETROS'. Below the navigation, there is a search bar and a 'Salir de la administración' link. The main content area is divided into two sections: 'Añadir una nueva Pregunta' (Add a new question) and 'Listado de preguntas : Gestión de Riesgos' (List of questions: Risk Management). The 'Añadir una nueva Pregunta' section contains a table with columns: 'Pregunta' (Question), 'Descripción de la no conformidad' (Non-conformity description), 'Proceso' (Process), 'Opcional' (Optional), 'Ref. Documentales' (Document references), 'Tipo de auditoría' (Audit type), and 'Insertar' (Insert). The 'Listado de preguntas' section shows a table with columns: 'Ref.', 'Orden' (Order), 'Pregunta' (Question), 'Descripción de la no conformidad' (Non-conformity description), 'Proceso' (Process), 'Opcional' (Optional), 'Ref. Documentales' (Document references), 'Tipo de auditoría' (Audit type), and 'Borrar' (Delete). The 'Nuevo' row (row 662) is highlighted with a red border. The 'Proceso' column for this row shows 'Gestión de Riesgos'.

Figura 66

### Nueva pregunta (rojo)

Tanto la pregunta como la descripción de no conformidad (cuando se responde "No" a una pregunta, la descripción de lo que no se ha hecho) son campos de texto normales.

El proceso corresponde a la pestaña a la que se añadirá la pregunta (marcado en azul están los procesos).

Las referencias documentales funcionan del mismo modo que las acciones correctivas/cierre en el seguimiento de auditoría, escribimos el documento donde debemos mirar para responder a la pregunta y le damos al botón [+].

El tipo de auditoría corresponde a en que tipos de auditoría se realizará esta pregunta.

Borrar una pregunta tiene el mismo efecto que la baja, si no se ha usado se borrará para siempre, mientras que si la pregunta ha sido respondida en alguna auditoría, esta se dará de baja para futuras auditorías pero se mantendrá en las realizadas anteriormente.

### Gestión del listado (rosa)

En el listado disponemos de las mismas opciones que en la nueva pregunta además del orden, que son las flechas azules con un número (que corresponde a la posición actual). Dandole hacia arriba (subir) intercambiamos la posición con la de arriba, análogamente ocurriría si le dieramos hacia abajo.

Podemos cambiar de proceso mediante las pestañas (en azul) para gestionar todas las preguntas.

## Administración de procesos

Las pestañas en las que se organizan las preguntas se administran desde la gestión de procesos, para ello acceder al menú **ADMIN. ENTIDADES -> Gestión de procesos**

Aquí podemos crear / modificar y borrar los procesos existentes, hay que tener cuidado porque al borrar un proceso también se borran (o se dan de baja) todas las preguntas del proceso.

La gestión de los procesos es muy intuitiva.

### Insertar nuevo Proceso

Los procesos corresponden a distintas secciones del cuestionario de una auditoría. Corresponden a las hojas excel del sistema manual previo.

Al borrar un proceso se dan de baja todas las preguntas contenidas en él.

Nombre	
<input type="text"/>	<input type="button" value="Insertar"/>

Listado de procesos		
Código	Nombre	Borrar
34	Gestión de Riesgos	<input type="button" value="Borrar"/>
35	Requisitos	<input type="button" value="Borrar"/>
36	DAR	<input type="button" value="Borrar"/>
37	Solución técnica	<input type="button" value="Borrar"/>
45	Peer Review	<input type="button" value="Borrar"/>
46	Anteproyecto	<input type="button" value="Borrar"/>
47	Gestión de Proyectos	<input type="button" value="Borrar"/>
60	PPQA Proceso	<input type="button" value="Borrar"/>
61	test	<input type="button" value="Borrar"/>

Figura 67

## Administración de personal

Para gestionar al personal del sistema ir al menú **ADMIN. ENTIDADES -> Gestión de personal**

En esta sección podremos modificar los nombres de las personas así como su email, o incluso buscar por el servicio web.

Es bastante importante que todo el personal disponga de email, para que lleguen los avisos, en caso de no disponer de email el receptor del aviso, éste se enviará a la lista de distribución de la auditoría.

**Gestión del personal**

**Buscar personal**

Buscar por código de recurso:	<input type="text"/>	<input type="button" value="Buscar"/>
-------------------------------	----------------------	---------------------------------------

**Listado de personas**

Código	Nombre	Email
E705	Ignacio VILALTA Esteban	email@email.es
E550	Jose Antonio PERNA Dominguez	email@email.es
E505	Manuel MAZARICO Torres	email@email.es
E600	Raul NOVOA Minguez	email@email.es
E806	Vicente GIL Zuazo	email@email.es

**Figura 68**

## Administración de proyectos

Para gestionar los proyectos al menú **ADMIN. ENTIDADES -> Gestión de personal**

La gestión de los proyectos, únicamente sirve para filtrar los proyectos sobre los que se pueden crear auditorías, de forma que quede más usable el formulario de creación donde hay que seleccionar el proyecto. Esta opción es la de "Auditabile" que significa que el proyecto puede recibir auditorías nuevas.

Es recomendable desmarcar aquellos proyectos terminados, puede modificarse sólo uno (mediante el botón "Modificar") o todos a la vez (mediante el botón "Guardar").

Vemos en la siguiente figura la administración de proyectos

**Gestión de proyectos**

**Buscar Proyectos**

Buscar por código de recurso:	<input type="text"/>	<input type="button" value="Buscar"/>
-------------------------------	----------------------	---------------------------------------

**Listado de proyectos**

Código	Nombre	Tipo	Jefe	Auditabile	
2007-520-55220	Portal La Calle Indiscreta	1	Jose Antonio PERNA Dominguez	<input checked="" type="checkbox"/>	<input type="button" value="Modificar"/>
2007-536-55246	CGI-INT-2007-536 Web de Educación Permanente	1	Manuel MAZARICO Torres	<input checked="" type="checkbox"/>	<input type="button" value="Modificar"/>
2007-557-55250	Cam 1	1	Ignacio VILALTA Esteban	<input checked="" type="checkbox"/>	<input type="button" value="Modificar"/>
2007-607-55320	2007-607 Expedientes ARI-ARCH	3	Raul NOVOA Minguez	<input checked="" type="checkbox"/>	<input type="button" value="Modificar"/>
2007-999-55180	Trabajos generales de Diseño	1	Ignacio VILALTA Esteban	<input checked="" type="checkbox"/>	<input type="button" value="Modificar"/>
2009-999-55583	MANTENIMIENTO - ETT	1	Vicente GIL Zuazo	<input checked="" type="checkbox"/>	<input type="button" value="Modificar"/>
2009-999-55647	MANTENIMIENTO CORRECTIVO ETT	1	Vicente GIL Zuazo	<input checked="" type="checkbox"/>	<input type="button" value="Modificar"/>

\* Si un proyecto es auditabile, entonces se pueden crear nuevas auditorías sobre él. Sirve para filtrar en la selección de proyecto a la hora de crear auditorías.

**Figura 69**

## Administración de tipos de no conformidad

Los tipos de no conformidad equivalen a las gravedades de una respuesta negativa, veámos en la sección de ejecución de auditoría en el apartado de resumen final, que debíamos seleccionar la gravedad entre grave y leve, estas opciones de gravedad se configuran aquí.

Para acceder a la gestión de tipos de no conformidad ir al menú  
**ADMIN. TIPOS -> Gestión de tipos de no conformidad.**

Aquí nos dejará borrar únicamente aquellos que no hayan sido utilizados, sin embargo podremos dejarlos como inválidos, lo cual significa que para las próximas auditorías no podrá utilizarse.

Lo vemos en la siguiente figura:

### Nuevo tipo de no conformidad

Los tipos de no conformidad corresponden a las distintas gravedades de una respuesta negativa, por ejemplo Grave, Leve, etc.

Al crear un tipo de no conformidad, debe de definirse la fórmula de cálculo para cada estado definido hasta el momento.

Al validar un tipo de no conformidad antiguo, es posible que no tenga definida la fórmula de cálculo para ese estado, por lo que es conveniente mirarlo.

Nombre			
		<input type="button" value="Insertar"/>	

**Tipos de no conformidad existentes**

Nombre	Descripción (Lo que se muestra)	Válido	Borrar
Mayor	Grave	<input checked="" type="checkbox"/> (Válido)	<input type="button" value="Borrar"/>
Menor	Leve	<input checked="" type="checkbox"/> (Válido)	<input type="button" value="Borrar"/>

**Figura 70**

Podemos cambiar la descripción de no conformidad, que es lo que se nos muestra en la parte pública de la aplicación.

Muy relacionado con el tipo de no conformidad están los estados de un proyecto/auditoría, mediante la fórmula de cálculo, que es conveniente mirar cuando se han modificado los tipos de no conformidad y los estados.

## Administración de estados de proyecto/auditoría

Los estados de proyecto ó auditoría corresponden al estado en el que se encuentran debido al número de no conformidades (de un tipo de gravedad) que tienen. Veámos en el resumen de auditoría el estado "Malo" y "Bueno".

Para acceder a la gestión de tipos de no conformidad ir al menú  
**ADMIN. TIPOS -> Tipos de ESTADO de auditoría.**

De la misma forma que para los tipos de no conformidad, es conveniente mirar la fórmula de cálculo después de gestionar los estados.

Los parámetros a modificar es la descripción del estado (Lo que se muestra en la parte pública) y el riesgo que incurre estar en dicho estado, por ejemplo estar en el estado Malo incurre un riesgo Elevado ó Alto.

El borrado de un estado es equivalente al de las gravedades, si ninguna auditoría tiene este estado como riesgo previo ó posterior asignado, podrá borrarse, en otro caso deberá marcarse como inválido para futuras auditorías.

Vemos en la siguiente figura la página de administración de estados de un proyecto/auditoría:

#### Nuevo estado

Los estados de un proyecto o auditoría son aquellos estados que incurren el tener un número determinado (por la fórmula de cálculo) de no conformidades en sus auditorías (o en la propia auditoría). Un estado puede ser por ejemplo : Malo, que implica un riesgo Alto (Riesgo de la auditoría ó proyecto).

Nombre	Riesgo		
<input type="text"/>	<input type="text"/>	<input checked="" type="checkbox"/> (Válido)	<input type="button" value="Insertar"/>

**Listado de estados**

Nombre	Descripción (Lo que se muestra)	Riesgo (Riesgo que incurre)	Válido	Guardar
Bueno	<input type="text" value="Bueno"/>	<input type="text" value="Bajo"/>	<input checked="" type="checkbox"/> (Válido)	<input type="button" value="Borrar"/>
Malo	<input type="text" value="Malo"/>	<input type="text" value="Alto"/>	<input checked="" type="checkbox"/> (Válido)	<input type="button" value="Borrar"/>
Regular	<input type="text" value="Regular"/>	<input type="text" value="Medio"/>	<input checked="" type="checkbox"/> (Válido)	<input type="button" value="Borrar"/>

**Figura 71**

## Administración de tipos de auditoría

Los tipos de auditoría corresponden a diferentes listados de preguntas, normalmente para diferentes períodos de un proyecto, o diferentes apartados. Podemos crear cuantas deseemos.

Existen varias opciones en los tipos de auditoría.

**Marcado por defecto** -> Sirve para que en la creación de nuevas auditorías, este tipo de auditoría esté marcado por defecto siempre, sólo sirve para aquellas que requieran proyecto.

**Requiere proyecto** -> Es la forma de diferenciar entre las auditorías normales y las especiales, aquellas que requieran proyecto necesitarán un proyecto sobre el cual auditar.

**Valido y Borrar** -> Funcionan igual que en los estados y tipos de no conformidad, se podrá borrar únicamente en el caso de que ninguna auditoría sea de ese tipo, mientras que si no queremos que salga más en los tipos de auditoría, deberemos marcarlo como inválido.

#### Nuevo tipo de auditoría

Los tipos de auditorías corresponden a diferentes listados de preguntas, cada tipo de auditoría tiene asignado un cuestionario subdividido en :

Por defecto en la creación de auditorías siempre vendrán marcadas las que estén marcadas como "Por defecto" = SI, pero SOLO aquellas que requieren proyecto, mientras que aquellas que no requieren proyecto se les denomina "Especiales".

Nombre	Descripción	Requiere proyecto		
<input type="text"/>	<input type="text"/>	<input checked="" type="radio"/> Si <input type="radio"/> No	<input type="button" value="Insertar"/>	

#### Listado de los tipos de auditoría

Nombre	Descripción	Requiere proyecto	Válido	Marcado por defecto	Borrar
QA1	<input type="text" value="Tras el análisis y el diseño"/>	<input checked="" type="checkbox"/> (Si)	<input checked="" type="checkbox"/> (Válido)	<input checked="" type="checkbox"/> (Si)	<input type="button" value="Borrar"/>
QA2	<input type="text" value="Tras la fase de desarrollo"/>	<input checked="" type="checkbox"/> (Si)	<input checked="" type="checkbox"/> (Válido)	<input checked="" type="checkbox"/> (Si)	<input type="button" value="Borrar"/>
QA3	<input type="text" value="Tras la fase de implantación"/>	<input checked="" type="checkbox"/> (Si)	<input checked="" type="checkbox"/> (Válido)	<input checked="" type="checkbox"/> (Si)	<input type="button" value="Borrar"/>
PPQA	<input type="text" value="Auditoría de la gestión de auditorías"/>	<input type="checkbox"/> (No)	<input checked="" type="checkbox"/> (Válido)	<input type="checkbox"/> (No)	<input type="button" value="Borrar"/>

**Figura 72**

## Administración de parámetros y fórmula de cálculo

En esta sección podremos administrar los parámetros (son para los avisos) y la fórmula de cálculo existente entre el tipo de no conformidad y el estado de proyecto/auditoría.

Vemos en la siguiente figura los parámetros configurables así como la forma que tiene la fórmula de cálculo:

### Tiempos para avisos

Descripción	Cantidad (días)
Número de días para avisar antes de la ejecución de la auditoría	15
Número de días entre avisos para que el auditado planifique la auditoría	5
Número máximo de avisos para que el auditado planifique, antes de avisar al auditor	3
Número de días para la realización del seguimiento y avisar al auditor	15

### Descripción de la fórmula de cálculo

Modificar fórmula de cálculo			
Tipo \\\ Estado	Bueno	Malo	Regular
Mayor	<= 0	<= 10000	<= 3
Menor	<= 4	<= 10000	<= 9
<input type="button" value="Guardar"/>			

Figura 73

Observar que para la fórmula de cálculo hay otro bloque de información que nos explica el significado de la fórmula, dado que no es fácilmente comprensible su significado.

Los parámetros configurables si son fácilmente entendibles y sirven para organizar el sistema de gestión de avisos.

### Fórmula de cálculo

La formula de calculo se define como el número de no conformidades MÁXIMO de un tipo de gravedad para el cual podemos estar en un estado. De esta forma, los estados se ordenan de mejor a peor según este número máximo, por ejemplo:

**Tipo/Estado Mejor estado 0 Peor estado 3 estado 2 estado 1**

**Grave 0 10000 5 2**

Esta formula significa que, los estados se ordenan de mejor a peor según el número de no conformidades.

**Número no conformidades: 0 2 5 1000**

**Estado: Mejor estado 0 Estado 1 Estado 2 Peor estado 3**

Y su significado es el siguiente:

Siendo x el número de no conformidades del tipo "Grave":

Si  $x \leq 0 \rightarrow$  Mejor estado 0

Si  $0 < x \leq 2 \rightarrow$  Estado 1

Si  $2 < x \leq 5 \rightarrow$  Estado 2

Si  $x > 5 \rightarrow$  Peor estado 3

Por ejemplo, si tengo 4 no conformidades graves, estaré en el "estado 2".

Esta misma descripción es la que se muestra en el bloque de información de la página como vemos en la figura:

#### Descripción de la fórmula de cálculo

**Mostrar Explicación de la fórmula de cálculo**

**Tip** La fórmula de cálculo se define como el número de no conformidades MÁXIMO de un tipo de gravedad para el cual podemos estar en un estado. De esta forma, los estados se ordenan de mejor a peor según este número máximo, por ejemplo:

Tipo/Estado	Mejor estado 0	Peor estado 3	estado 2	estado 1
Grave	0	1000	5	2

Esta fórmula significa que, los estados se ordenan de mejor a peor según: 0 - 2 - 5 - 1000 (Número de no conformidades). "Mejor estado 0" - "estado 1" - "estado 2" - "Peor estado 3"

Y su significado es:

Siendo  $x$  el número de no conformidades del tipo "Grave":

- Si  $x \leq 0 \rightarrow$  Mejor estado 0
- Si  $0 < x \leq 2 \rightarrow$  estado 1
- Si  $2 < x \leq 5 \rightarrow$  estado 2
- Si  $x > 5 \rightarrow$  Peor estado 3

Figura 74

Cuando se crea un tipo de no conformidad ó un estado nuevo, se pone toda su fila/columna al valor 10.000 (cifra nunca alcanzable). Mientras que cuando se invalidan, la fórmula se guarda (para los estados actuales) y al validarla se mantiene, sin embargo es muy posible que no sea lo que nosotros deseamos, por ello es por lo que es siempre recomendable comprobarla.

## Administración de las cuentas de administradores

Las cuentas de los administradores pueden gestionarse desde la cuenta del superadministrador, o bien desde la propia cuenta, es decir el superadministrador es el único que puede modificar al resto de administradores, mientras que el resto sólo puede gestionarse su propia cuenta.

Tanto administradores como superadministrador pueden crear nuevos administradores normales, no se podrán crear superadministradores.

La gestión de una cuenta consiste simplemente en el cambio de contraseña, como vemos en la siguiente figura:

**Nuevo Administrador**

La gestión de administradores permite crear nuevos administradores o cambiar la contraseña a los administradores actuales.

Usuario	Contraseña	Repita contraseña
<input type="text"/>	<input type="password"/>	<input type="password"/>

**Insertar**

**Listado de administradores**

Usuario	Vieja contraseña	Contraseña	Repita contraseña	Borrar	Errores
admin	*****	<input type="password"/>	<input type="password"/>	<b>Borrar</b>	
ivilalta	*****	<input type="password"/>	<input type="password"/>	<b>Borrar</b>	

**Modificar**

Figura 75