



INSTITUTO TECNOLÓGICO DE BUENOS AIRES
ESCUELA DE POSTGRADO

TESIS DE MAGISTER EN INGENIERÍA DE SOFTWARE

**HERRAMIENTA DE INTEGRACIÓN DE ACTIVIDADES PARA
PROYECTOS DE DESARROLLO DE SOFTWARE**

Autor: Ing. Gabriel Montes

Directores: Dr. Ramón García Martínez, M. Ing. Enrique J. Fernández

Buenos Aires, 2009

*A mis padres, Margarita y Juan Carlos,
por haberme enseñado a dar siempre lo mejor de mí.*

*A mis hijas Abril y Zoé,
por todo su cariño, amor y comprensión a lo largo de todo este tiempo.*

*A mi esposa Romina,
por su gran energía, por haber mantenido siempre viva la esperanza
y por haberme motivado a continuar aún en los momentos más difíciles.*

Resumen

El uso de múltiples herramientas de software, por lo general inconexas, para realizar la gestión de los proyectos de desarrollo de software presenta desventajas respecto del uso de una única herramienta de gestión integral. La adopción de una herramienta de ese tipo en entornos donde ya existen herramientas institucionalizadas podría ser resistida tanto por la organización como por los usuarios de la misma. El desarrollo de una herramienta muy simple que integre datos de gestión de diferentes sistemas complementándolos y no reemplazándolos podría tener algunas de las ventajas de una herramienta integral y, al mismo tiempo, facilitar su introducción a la organización.

Abstract

The usage of multiple software tools, usually not related to each other, to manage software development projects lacks of the advantages a single integrated management tool. The usage of such kind of integrated tools where the usage of other tools is already institutionalized could be difficult because of the resistance to change that can be seen at the organization or individual user level. The development of a tiny and very simple tool that just integrates management data from different sources, complementing and not replacing the tools being used can have some of the advantages of a large integrated tool and, at the same time, can facilitate its implementation into the organization.

TABLA DE CONTENIDOS

1	INTRODUCCIÓN	1
1.1	Sobre éste trabajo	1
1.2	Organización del documento	1
2	ESTADO DE LA CUESTIÓN	3
2.1	Introducción.....	3
2.2	Situación actual.....	4
3	HIPÓTESIS DE TRABAJO	11
4	PROPUESTA	13
4.1	Introducción a MÉTRICA Versión 3	13
4.2	Estudio de viabilidad del sistema (EVS).....	14
4.2.1	Establecimiento del alcance del sistema (EVS.1)	16
4.2.2	Estudio de la situación actual (EVS 2).....	22
4.2.3	Definición de requisitos del sistema (EVS 3)	25
4.2.4	Estudio de alternativas de solución (EVS 4).....	28
4.2.5	Valoración de las alternativas (EVS 5)	29
4.2.6	Selección de la solución (EVS 6).....	29
4.3	Análisis del sistema de información (ASI).....	29
4.3.1	Definición del sistema (ASI 1).....	32
4.3.2	Establecimiento de requisitos (ASI 2).....	34
4.3.3	Análisis de los casos de uso (ASI 4)	53
4.3.4	Análisis de clases (ASI 5).....	54
4.3.5	Definición de interfaces de usuario (ASI 8).....	56
4.3.6	Análisis de consistencia y especificación de requisitos (ASI 9)	62
4.3.7	Especificación del plan de pruebas (ASI 10)	65
4.3.8	Aprobación del análisis del sistema de información (ASI 11)	69
4.4	Diseño del sistema de información (DSI).....	70
4.4.1	Definición de la arquitectura del sistema (DSI 1)	73
4.4.2	Diseño de la arquitectura de soporte (DSI 2)	81
4.4.3	Diseño de casos de uso reales (DSI 3)	84
4.4.4	Diseño de clases (DSI 4).....	90
4.4.5	Diseño físico de datos (DSI 6)	101
4.4.6	Verificación y aceptación de la arquitectura del sistema (DSI 7)	111
4.4.7	Generación de especificaciones de construcción (DSI 8)	113
4.4.8	Especificación técnica del plan de pruebas (DSI 10)	116
4.4.9	Aprobación del diseño del sistema de información (DSI 12).....	124
4.5	Construcción del sistema de información (CSI).....	124
4.5.1	Preparación del entorno de generación y construcción (CSI 1)	127
4.5.2	Generación del código de los componentes y procedimientos (CSI 2)	130
4.5.3	Ejecución de las pruebas unitarias (CSI 3).....	132
4.5.4	Ejecución de las pruebas del sistema (CSI 5)	133
4.5.5	Elaboración de los manuales de usuario (CSI 6).....	135
4.5.6	Definición de la formación de usuarios finales (CSI 7)	135
4.5.7	Aprobación del sistema de información (CSI 9).....	137

4.6	Implantación y aceptación del sistema de información (IAS).....	137
4.6.1	Establecimiento del plan de implantación (IAS 1).....	139
4.6.2	Incorporación del sistema al entorno de operación (IAS 3)	142
4.6.3	Carga de datos al entorno de operación (IAS 4)	143
4.6.4	Pruebas de aceptación del sistema (IAS 6)	144
5	CONCLUSIONES	147
5.1	Sobre la metodología	147
5.2	Sobre el desarrollo realizado	147
6	FUTURAS LÍNEAS DE INVESTIGACIÓN	149
7	BIBLIOGRAFÍA	151
8	ANEXOS	153
8.1	Plantilla de código ASP	153
8.2	Estructuras XSD	153
8.3	Captura de pantallas	156
8.4	Registro de ejecución de pruebas de sistema.....	158
8.5	Registro de errores	159
8.6	Manual de usuario	159
8.6.1	Introducción	159
8.6.2	Trabajar con actividades	160
8.6.3	Configurar proyectos	163
8.6.4	Importar actividades.....	167
8.7	Interfaces de MÉTRICA Versión 3.....	168
8.7.1	Gestión de proyectos (GP)	169
8.7.2	Gestión de la configuración (GC)	172

1 INTRODUCCIÓN

1.1 Sobre éste trabajo

Este trabajo aborda la problemática del uso de múltiples herramientas de software para asistir la gestión de proyectos de desarrollo de software y sobre una solución propuesta basada en el desarrollo de una herramienta basada en tecnologías web que consolide la información dispersa en aquellas. El desarrollo de dicha herramienta se apoya en la metodología MÉTRICA Versión 3 [CSI, 1995].

El principal objetivo buscado es facilitar la búsqueda y consolidación de la información dispersa en las diferentes herramientas que se usan para la gestión del proyecto a quienes trabajan en el de software. Es de esperar que esto derive en una mejor calidad y una mayor productividad, tanto a nivel del proyecto como de la organización que lo contiene. Como objetivo secundario se encuentra el uso de una metodología de desarrollo reconocida como lo es MÉTRICA.

1.2 Organización del documento

El documento está compuesto por 7 capítulos y un conjunto de anexos, siguiendo los pasos de la metodología MÉTRICA.

El capítulo 1, Introducción, presenta los principales aspectos del trabajo mostrando el carácter metódico del mismo. También abarca la organización del documento a modo de resumen del mismo.

El capítulo 2, Estado de la cuestión, presenta la información recogida que se utilizó para establecer la situación actual que impulsa la necesidad de proveer una solución a la problemática existente.

El capítulo 3, Hipótesis de trabajo, presenta una definición concreta del tema, de los objetivos del trabajo y de los límites del mismo.

El capítulo 4, Propuesta, presenta una introducción a la metodología MÉTRICA y luego el desarrollo de la herramienta propuesta a través de los diferentes subprocesos del Proceso de

Desarrollo de Sistemas de Información de la metodología. Cada subproceso es abarcado en un subcapítulo para facilitar el seguimiento del trabajo.

El capítulo 5, Conclusiones, presenta una evaluación general de los procedimientos y recursos utilizados, notando aciertos y oportunidades de mejora encontradas. También analiza la contribución realizada y la experiencia adquirida en la realización del trabajo.

El capítulo 6, Futuras líneas de investigación, sugiere trabajos de investigación o desarrollos subsiguientes al presente trabajo.

El capítulo 7, Bibliografía, presenta una lista ordenada de todo el material que ha sido utilizado, organizando en bibliografía consultada y referenciada.

Finalmente, el capítulo 8, Anexos, presenta los documentos generados por la aplicación de la metodología MÉTRICA (planes de gestión, de gestión de configuración, de calidad, planillas y registros) para que el lector los tenga como referencia.

2 ESTADO DE LA CUESTIÓN

2.1 Introducción

Durante la ejecución del proceso de desarrollo de software, el equipo de trabajo se enfrenta a diferentes y variadas asignaciones que van más allá de lo que está planeado en un cronograma de actividades. Esto es debido a que en todo proceso de desarrollo ocurren eventos que requieren la ejecución de acciones para asegurar el cumplimiento de los objetivos del proyecto, minimizando los desvíos que se vayan identificando a lo largo de su ejecución. En ese contexto, cada miembro del equipo de desarrollo puede tener asignada una o más de las siguientes responsabilidades en forma simultánea:

- cumplir con asignaciones específicas en el cronograma de actividades;
- completar acciones derivadas de la gestión del proceso de desarrollo;
- completar acciones derivadas de actividades de aseguramiento de la calidad;
- completar acciones derivadas de auditorías al producto o al proceso de desarrollo; y
- completar acciones derivadas de requerimientos de cambio.

La lista anterior no pretende ser exhaustiva sino mostrar la diversidad de fuentes que derivan en actividades o acciones a ejecutar por parte de los miembros de un equipo de desarrollo.

Dependiendo del grado de formalidad que el proyecto u organización utilice para la gestión de esas acciones, los miembros del equipo deberán interactuar en mayor o menor medida con uno o más de los siguientes tipos de herramientas:

- software de planificación (planning/scheduling);
- software de gestión de riesgos (risk management);
- software de seguimiento de acciones (action tracking);
- software de seguimiento de revisiones de pares (peer review tracking);
- software de seguimiento de defectos (defect tracking); y

- software de gestión de cambios (change management).

En casos en los que la madurez del equipo u organización es baja, es posible que pocas o ninguna herramienta se utilice para darle seguimiento a las responsabilidades de los miembros del equipo. La consecuencia directa de esto es la falta de control sobre el mismo proceso de desarrollo, lo que puede llevar a problemas para cumplir los objetivos del proyecto. Es evidente que cuanto más complejo sea el proyecto, mayor riesgo existirá.

Por otro lado, a medida que la madurez es mayor, es común que se utilicen varias de estas herramientas, quizás integradas, para dar soporte al desarrollo y a la gestión y control del proyecto. En este caso, y relacionado con los conceptos de madurez y control, disminuyen los riesgos de no alcanzar los objetivos del proyecto.

En los entornos medianamente maduros, es común que los integrantes del equipo tengan que interactuar con varias herramientas para identificar sus responsabilidades. Al mismo tiempo, la gestión del proyecto también debe valerse de todas las herramientas en uso para relevar completamente el estado del mismo.

Lo anterior crea la necesidad de interacción en el día a día con diferentes herramientas. ¿Qué beneficios y dificultades se enfrentarían si, en vez, se interactuara con una sola herramienta que contuviera todo lo necesario para la ejecución y gestión efectiva del proyecto?

2.2 Situación actual

En la actualidad hay infinidad de herramientas comerciales, libres y desarrolladas en forma privada que se orientan a la planificación de proyectos, gestión de riesgos, de cambios, de incidencias, y demás actividades de la gestión de software. Existen muchos estudios que tratan de ordenar esta diversidad como ser [Mühlbauer, 2007], [Serrano et al., 2005], [Wicks, 2005] y [Wicks, 2006].

Estas herramientas proveen facilidades para gestionar varios tipos de actividades, sin embargo su uso integrado no es habitual. Inclusive, ha habido cierto grado de controversia respecto de la integración de herramientas para gestión de proyectos como se describe en [Thomas et al., 1992]. De cualquier manera, para validar la situación actual se realizó una encuesta para relevar las herramientas utilizadas en ambientes reales de desarrollo de software en el país. Los resultados se pueden ver en la Tabla 1.

Caso	Planificación	Riesgos	Acciones	Revisiones	Defectos	Cambios
1	Primavera Teamplay	IBM Rational ClearQuest	Desarrollo web propio	Desarrollo web propio	IBM Rational ClearQuest	IBM Rational ClearQuest
2	Microsoft Project, Altassian JIRA	Altassian JIRA	Altassian JIRA	N/A	Altassian JIRA	Altassian JIRA
3	N/A	Plantilla Microsoft Excel, Altassian JIRA	Altassian JIRA	Smart Bear Code Collaborator	Altassian JIRA	Altassian JIRA
4	Microsoft Project	Plantilla Microsoft Excel	Plantilla Microsoft Excel	Plantilla Microsoft Excel	Desarrollo web propio	Plantilla Microsoft Excel
5	Microsoft Project, Planner	Plantilla Microsoft Excel	Plantilla Microsoft Word	Plantilla Microsoft Word	Plantilla Microsoft Excel	Plantilla Microsoft Word
6	Microsoft Project	Plantilla Microsoft Excel	Plantilla Microsoft Word	Plantilla Microsoft Word	Plantilla Microsoft Excel	Plantilla Microsoft Word
7	dotProject	N/A	Plantilla Microsoft Excel	KnowledgeTree	Plantilla Microsoft Excel, dotProject	dotProject
8	Microsoft Project	N/A	Plantilla Microsoft Excel	N/A	Altassian JIRA	Plantilla Microsoft Excel, Altassian JIRA
9	Microsoft Project	N/A	Plantilla Microsoft Excel	N/A	Bugzilla	Bugzilla
10	Microsoft Project	N/A	N/A	N/A	HP TestDirector, BMC Remedy Action Request System	N/A

Tabla 1: Resultados de la encuesta de uso de herramientas. Estos resultados representan 10 ejemplos de utilización de herramientas en el país.

Si se asume que los resultados de la encuesta son relativamente representativos de la generalidad, se puede concluir que los equipos de desarrollo se enfrentan diariamente a muchos problemas relacionados con el uso de múltiples herramientas o la falta de las mismas para realizar su trabajo. Para reafirmar lo anterior, es necesario realizar un análisis que vaya desglosando los diferentes aspectos y problemas que la diversidad de herramientas en uso presenta.

Enfocándose en las herramientas que forman parte de las respuestas de la encuesta, surgen las siguientes conclusiones preliminares:

- no se ven casos claros en los que estén en uso herramientas para gestionar en forma integrada todas las actividades de los proyectos;
- Excel parecería ser la herramienta más usada, quizás por ser una herramienta muy poderosa, adaptable y de uso general;
- JIRA tendría la característica de ser lo más parecido a una herramienta de gestión integrada, que se usa de esa forma;
- Project sería la herramienta más usada para la planificación de proyectos;
- en muy pocas empresas se utilizan herramientas que cruzan información entre los distintos tipos de actividades, como ser JIRA y ClearQuest.

Para avanzar sobre esas conclusiones, es necesario un análisis más detallado. Como primer corte, se ve que hay muchas aplicaciones comerciales y desarrollos propios en uso y, en general, se nota una gran heterogeneidad en la selección de éstas para atacar cada tipo de actividad. En 10 respuestas se contabilizan 11 aplicaciones comerciales diferentes y 3 desarrollos propios. Estas aplicaciones comerciales pueden verse en la Tabla 2.

Herramienta	Referencia
Altassian JIRA	http://www.atlassian.com/software/jira/
BMC Remedy Action Request System	http://www.bmc.com/
Bugzilla	http://www.bugzilla.org/
dotProject	http://www.dotproject.net/
HP TestDirector	http://www.hp.com/go/software/
IBM Rational ClearQuest	http://www.ibm.com/software/awdtools/clearquest/
KnowledgeTree	http://www.knowledgetree.com/
Microsoft Project	http://office.microsoft.com/project/
Planner	http://live.gnome.org/Planner/
Primavera Teamplay	http://www.primavera.com/
Smart Bear Code Collaborator	http://www.smartbear.com/codecollab.php

Tabla 2: Aplicaciones comerciales referenciadas en la encuesta. Estas aplicaciones tienen propósitos específicos como ser la planificación, la gestión de cambios o la gestión de la configuración.

Asimismo, se ve que en 8 casos se respondió que se usan plantillas basadas aplicaciones genéricas (Excel y Word) como herramientas de apoyo para la gestión de ciertas actividades.

Un análisis más detallado de los casos, separando las respuestas entre aplicaciones comerciales, herramientas basadas en plantillas o desarrolladas internamente y actividades

que no tienen apoyo alguno, muestra que hay mucha disparidad en la forma de trabajar de los encuestados. Los resultados así organizados pueden verse en la Figura 1.

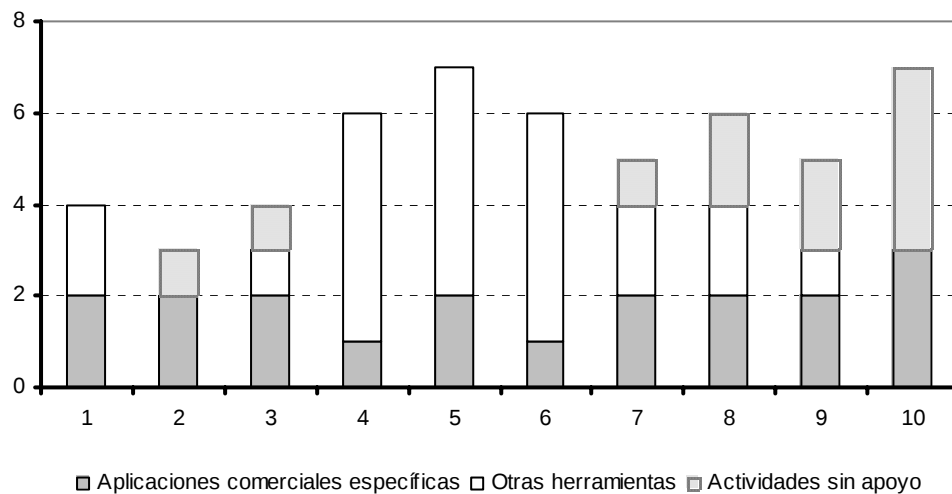


Figura 1: Uso de aplicaciones comerciales y herramientas por caso. Cada columna representa, por caso encuestado, la cantidad de aplicaciones comerciales, la cantidad de herramientas no basadas en aplicaciones comerciales y la cantidad de actividades que no tienen apoyo de aplicación o herramienta alguna.

En 7 de los 10 casos presentados se observa que son utilizadas dos aplicaciones comerciales. Es interesante notar que en 3 de esos 7 casos, estas aplicaciones comerciales tienen cierto grado de integración al abarcar diferentes aspectos de la gestión. Es notable, entonces, que en la mayoría de los casos, esta integración no está presente. A modo de resumen, los puntos más destacables, son:

- la baja integración que existe, en general, entre las herramientas utilizadas;
- la gran cantidad de fuentes de actividades que, tanto desarrolladores como quienes gestionan los proyectos, deben monitorear continuamente; y
- la baja capacidad de trazabilidad entre actividades, al combinar la problemática de baja integración y alta cantidad de fuentes.

El otro corte necesario para analizar las respuestas es por tipo de actividad. De este corte se pueden principalmente hacer varios análisis. El primero es analizar cuántas herramientas están en uso para dar soporte a un tipo específico de actividad y, de éstas, cuántas son aplicaciones comerciales diseñadas específicamente para darlo y cuántas lo reciben a través de plantillas

basadas en aplicaciones genéricas o desarrollos propios. En este análisis, se observa en la Figura 2.

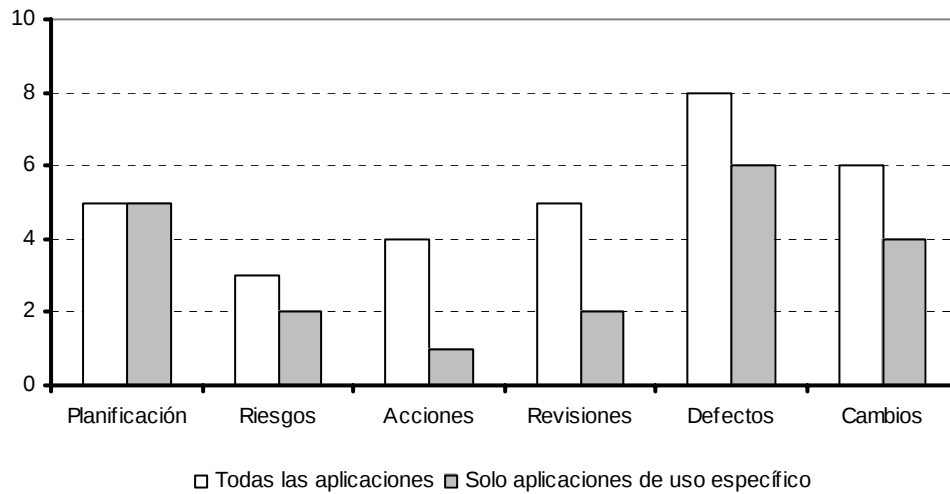


Figura 2: Cantidad de aplicaciones en uso por tipo de actividad. Las aplicaciones de uso específico son las aplicaciones comerciales. La diferencia entre estas y el total se explica en el uso de herramientas genéricas basadas Excel, Word o desarrollos propios.

De éste análisis se sacan las siguientes conclusiones:

- la planificación de actividades se hace en forma exclusiva a través de aplicaciones comerciales específicamente diseñadas para tal tarea;
- la mayor cantidad de aplicaciones o herramientas se concentra en la gestión de defectos y cambios;
- los riesgos y las acciones son las actividades con menos soporte de aplicaciones o herramientas; y
- los riesgos, las acciones y las revisiones son las actividades con menos soporte de aplicaciones comerciales.

Por último, vale analizar la capacidad que tienen las diferentes herramientas de dar soporte a los diferentes tipos de actividades. La cuenta de actividades para las que son usadas cada aplicación nombrada en la encuesta se muestra en la Figura 3.

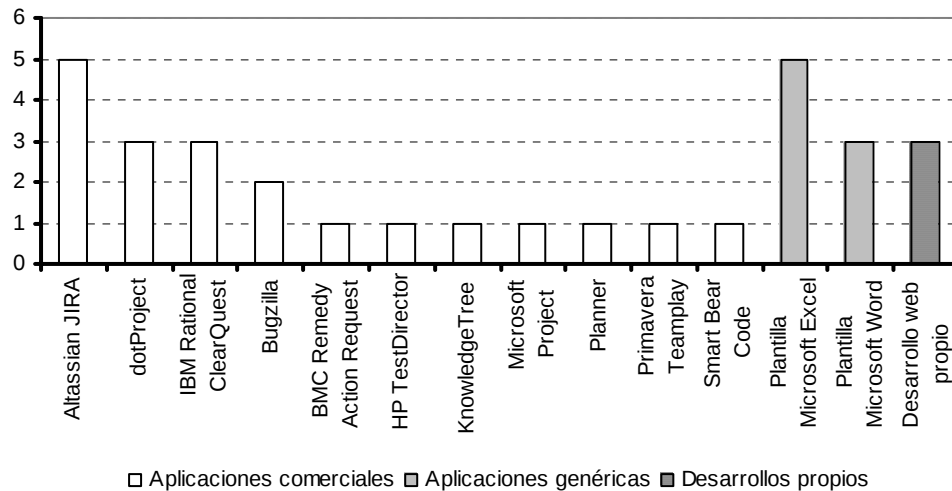


Figura 3: Cantidad de actividades para las que son utilizadas cada aplicación. Hay una notable tendencia hacia la especialización en las aplicaciones comerciales, salvo casos muy puntuales. La aplicación genérica más utilizada es Excel.

Del análisis de la utilización de las diferentes herramientas por tipo de actividad se sacan las siguientes conclusiones:

- solo tres aplicaciones comerciales se destacan por su uso para la gestión de diferentes tipos de actividades: JIRA, dotProject y ClearQuest;
- estas tres aplicaciones comerciales proveen facilidades para vincular las diferentes actividades entre sí;
- el resto de las aplicaciones comerciales muestran una gran tendencia a la especialización, dificultando también la capacidad de generar trazabilidad en forma automatizada; y
- las plantillas basadas en Excel son utilizadas como soporte para todos los tipos de actividades con excepción de la planificación de proyectos, convirtiéndose en la aplicación más versátil de las nombradas en la encuesta.

Volviendo a la pregunta inicial, en un caso ideal todas las herramientas estarían fusionadas en una sola interfaz para el usuario, ya sea este un miembro del equipo o quien esté encargado de la gestión del proyecto. Entre otras, las siguientes ventajas se derivarían de tener una herramienta integrada:

- tener un único punto de entrada y referencia donde estén contenidas todas las asignaciones de cada integrante del equipo;
- mayor facilidad de aprendizaje al tener que una sola herramienta, con una interfaz consistente y homogénea para todas las aplicaciones;
- posibilidad de centralizar y/o automatizar reglas y políticas vigentes en el proyecto u organización;
- posibilidad de integrar, documentar e incluso forzar la interrelación entre las diferentes actividades (revisiones y los defectos relacionados, riesgos y acciones preventivas relacionadas);
- recolección de métricas de forma automática, consistente y centralizada para su uso directo en la gestión como también para su posterior análisis y uso como datos históricos.

Por otro lado, existen varios factores que pueden influir en contra de la introducción de una herramienta única que gestione integralmente el trabajo del proyecto en un ambiente real:

- no existe gran variedad de herramienta que gestionen integralmente los proyectos de desarrollo: solo JIRA cumple parcialmente con esto;
- las herramientas comerciales que podrían usarse como base en la integración suelen ser muy costosas: JIRA, Project, ClearQuest, TestDirector pueden superar los USD 1.000,00 por licencia;
- el cambio de las herramientas que se encuentren actualmente en uso puede ser resistido a nivel de usuarios como también a nivel gerencial por haber invertido dinero, desarrollo y capacitación en las mismas

3 HIPÓTESIS DE TRABAJO

Del contexto descrito en la sección anterior se desprende que es habitual que un proyecto tenga problemas originados en:

- la necesidad de que los integrantes del equipo sepan utilizar o estén capacitados en las diferentes herramientas;
- la dificultad que puede haber en identificar correctamente las prioridades que pueden tener las asignaciones existentes simultáneamente en diferentes herramientas;
- la dificultad inherente al hecho de tener que monitorear constantemente diferentes herramientas; y
- la posibilidad de que, por olvido u omisión, queden actividades o acciones sin completar y esto ponga en riesgo el cumplimiento de los objetivos del proyecto.

Basado en todo lo anterior, es claro que una herramienta que permita gestionar integralmente todos estos aspectos del proceso de desarrollo de software es un complemento muy importante a la propia metodología utilizada.

Es importante notar que, dados los factores que pueden jugar en contra de dicha herramienta integrada, es necesario abordar el tema desde una perspectiva menos agresiva. No hacerlo podría crear desafíos lo suficientemente grandes como para que la introducción exitosa de tal herramienta se vuelva inviable.

Orientando la solución a producir pequeños cambios con gran impacto en los proyectos, este trabajo intentará mostrar que es posible realizar una herramienta que aproveche la infraestructura existente en el proyecto u organización y provea tanto a los miembros del equipo encargados del desarrollo técnico como a quienes realizan la gestión del mismo, de un panel centralizado donde se puedan consultar todas las asignaciones existentes, sin importar la fuente de las mismas. Esto permitiría obtener, en una primera etapa, las siguientes ventajas:

- tener un único punto de entrada y referencia donde estén contenidas todas las asignaciones de cada integrante del equipo;

- posibilidad de integrar las diferentes actividades (revisiones y los defectos relacionados, riesgos y acciones preventivas relacionadas);
- no impactar fuertemente en las herramientas y procesos existentes, facilitando su adopción.

La atención estará centrada en la integración de la información de actividades contenida en dos herramientas específicas, por ejemplo Project y Excel, presentando en forma consolidada las actividades de cada recurso del proyecto. La integración de otras herramientas quedará fuera del alcance de este desarrollo.

Es importante notar que el desarrollo estará limitado a la consulta y consolidación de todas las actividades presentes en otras herramientas, quedando expresamente excluida cualquier actualización o gestión de las tareas desde la herramienta de consolidación.

Todo lo anterior proveerá la base arquitectónica, funcional y ejemplos prácticos de integración de datos que saquen a la luz las dificultades del desarrollo y, al mismo tiempo, las ventajas de avanzar hacia una integración total de herramientas.

4 PROPUESTA

4.1 Introducción a MÉTRICA Versión 3

El presente trabajo se realizará aplicando la metodología MÉTRICA Versión 3. Esta metodología ofrece a las Organizaciones un instrumento útil para la sistematización de las actividades que dan soporte al ciclo de vida del software dentro del marco que permite alcanzar los siguientes objetivos:

- proporcionar o definir Sistemas de Información que ayuden a conseguir los fines de la Organización mediante la definición de un marco estratégico para el desarrollo de los mismos;
- dotar a la Organización de productos software que satisfagan las necesidades de los usuarios dando una mayor importancia al análisis de requisitos;
- mejorar la productividad de los departamentos de Sistemas y Tecnologías de la Información y las Comunicaciones, permitiendo una mayor capacidad de adaptación a los cambios y teniendo en cuenta la reutilización en la medida de lo posible;
- facilitar la comunicación y entendimiento entre los distintos participantes en la producción de software a lo largo del ciclo de vida del proyecto, teniendo en cuenta su papel y responsabilidad, así como las necesidades de todos y cada uno de ellos; y
- facilitar la operación, mantenimiento y uso de los productos de software obtenidos.

Esta versión de MÉTRICA contempla el desarrollo de Sistemas de Información para las distintas tecnologías que actualmente están conviviendo y los aspectos de gestión que aseguran que un Proyecto cumple sus objetivos en términos de calidad, coste y plazos.

Adicionalmente posee gran adaptabilidad, flexibilidad y sencillez, así como la estructura de actividades y tareas y la división en Procesos, más adecuada a la entrada-transformación-salida que se produce en cada una de las divisiones del ciclo de vida de un proyecto. Para cada tarea se detallan los participantes que intervienen, los productos de entrada y de salida así como las técnicas y prácticas a emplear para su obtención.

En la elaboración de MÉTRICA Versión 3 se han tenido en cuenta los métodos de desarrollo más extendidos, así como los últimos estándares de ingeniería del software y calidad, además de referencias específicas en cuanto a seguridad y gestión de proyectos. También se ha tenido en cuenta la experiencia de los usuarios de las versiones anteriores para solventar los problemas o deficiencias detectados.

En una única estructura la metodología MÉTRICA Versión 3 cubre distintos tipos de desarrollo: estructurado y orientado a objetos, facilitando a través de interfaces la realización de los procesos de apoyo u organizativos: Gestión de Proyectos, Gestión de Configuración, Aseguramiento de Calidad y Seguridad.

La automatización de las actividades propuestas en la estructura de MÉTRICA Versión 3 es posible ya que sus técnicas están soportadas por una amplia variedad de herramientas de ayuda al desarrollo.

Tanto la metodología como todas estas herramientas estarán disponibles en la web del Consejo Superior de Informática: www.csi.map.es.¹

4.2 Estudio de viabilidad del sistema (EVS)

El propósito de este proceso de MÉTRICA Versión 3 es analizar un conjunto concreto de necesidades, con la idea de proponer una solución a corto plazo. Los criterios con los que se hace esta propuesta no serán estratégicos sino tácticos y relacionados con aspectos económicos, técnicos, legales y operativos.

Los resultados del Estudio de Viabilidad del Sistema constituirán la base para tomar la decisión de seguir adelante o abandonar. Si se decide seguir adelante pueden surgir uno o varios proyectos que afecten a uno o varios sistemas de información. Dichos sistemas se desarrollarán según el resultado obtenido en el estudio de viabilidad y teniendo en cuenta la cartera de proyectos para la estrategia de implantación del sistema global.

Para valorar las alternativas planteadas y determinar una única solución, se estudiará el impacto en la organización de cada una de ellas, la inversión y los riesgos asociados.

¹ Texto adaptado de la metodología MÉTRICA Versión 3. Referirse a [CSI, 2005].

El resultado final de este proceso son los productos relacionados con la solución que se propone para cubrir la necesidad concreta que se planteó en el proceso, y que depende de si la solución conlleva desarrollo a medida o no.

Mientras que el Plan de Sistemas de Información tiene como objetivo proporcionar un marco estratégico que sirva de referencia para los Sistemas de Información de un ámbito concreto de una organización, el objetivo del Estudio de Viabilidad del Sistema es el análisis de un conjunto concreto de necesidades para proponer una solución a corto plazo, que tenga en cuenta restricciones económicas, técnicas, legales y operativas. La solución obtenida como resultado del estudio puede ser la definición de uno o varios proyectos que afecten a uno o varios sistemas de información ya existentes o nuevos. Para ello, se identifican los requisitos que se ha de satisfacer y se estudia, si procede, la situación actual.

A partir del estado inicial, la situación actual y los requisitos planteados, se estudian las alternativas de solución.

Dichas alternativas pueden incluir soluciones que impliquen desarrollos a medida, soluciones basadas en la adquisición de productos software del mercado o soluciones mixtas. Se describe cada una de las alternativas, indicando los requisitos que cubre.

Una vez descritas cada una de las alternativas planteadas, se valora su impacto en la organización, la inversión a realizar en cada caso y los riesgos asociados. Esta información se analiza con el fin de evaluar las distintas alternativas y seleccionar la más adecuada, definiendo y estableciendo su planificación.

Si en la organización se ha realizado con anterioridad un Plan de Sistemas de Información que afecte al sistema objeto de este estudio, se dispondrá de un conjunto de productos que proporcionarán información a tener en cuenta en todo el proceso.

Las actividades que engloba este proceso se muestran en la Figura 4, en la que se indican las actividades que pueden ejecutarse en paralelo y las que precisan para su realización resultados originados en actividades anteriores.

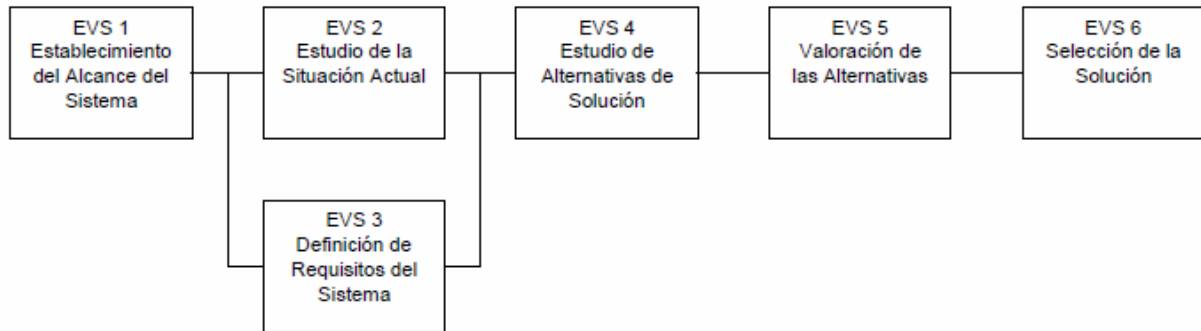


Figura 4: Actividades que componen el proceso de Estudio de Viabilidad del Sistema (EVS). Algunas de estas actividades pueden realizarse en paralelo.

4.2.1 Establecimiento del alcance del sistema (EVS.1)

En esta actividad se estudia el alcance de la necesidad realizando una descripción general de la misma. Se determinan los objetivos, se inicia el estudio de los requisitos y se identifican las unidades organizativas afectadas estableciendo su estructura.

Se analizan las posibles restricciones, tanto generales como específicas, que puedan condicionar el estudio y la planificación de las alternativas de solución que se propongan.

Si la justificación económica es obvia, el riesgo técnico bajo, se esperan pocos problemas legales y no existe ninguna alternativa razonable, no es necesario profundizar en el estudio de viabilidad del sistema, analizando posibles alternativas y realizando una valoración y evaluación de las mismas, sino que éste se orientará a la especificación de requisitos, descripción del nuevo sistema y planificación.

Se detalla la composición del equipo de trabajo necesario para este proceso y su planificación. Finalmente, con el fin de facilitar la participación activa de los usuarios en la definición del sistema, se identifican sus perfiles, dejando claras sus tareas y responsabilidades.

4.2.1.1 Estudio de la solicitud (EVS 1.1)

Se realiza una descripción general de la necesidad planteada por el usuario, y se estudian las posibles restricciones de carácter económico, técnico, operativo y legal que puedan afectar al sistema. También se establecen los objetivos generales del Estudio de Viabilidad, teniendo en cuenta las restricciones identificadas anteriormente.

Como el sistema no se encuentra en el ámbito de un Plan de Sistemas de Información, no existe información previa relativa a requisitos o arquitectura.

4.2.1.1.1 Descripción general del sistema

El sistema que se requiere construir tiene como objetivo consolidar la información referente a todas las actividades asignadas a los recursos de un proyecto, proveyendo a estos un único punto de entrada para satisfacer sus consultas.

El sistema estará compuesto principalmente una interfaz web² que permitirá a todos los usuarios acceder a la información sobre las actividades y responsabilidades asignadas, una base de datos que proveerá persistencia y acceso múltiple y simultáneo a la información y un conjunto de módulos de importación de datos, los cuales tomarán la información de actividades y responsabilidades de una o más herramientas e incorporarán esa información a la base de datos principal. Un esquema gráfico del sistema se encuentra en la Figura 5.

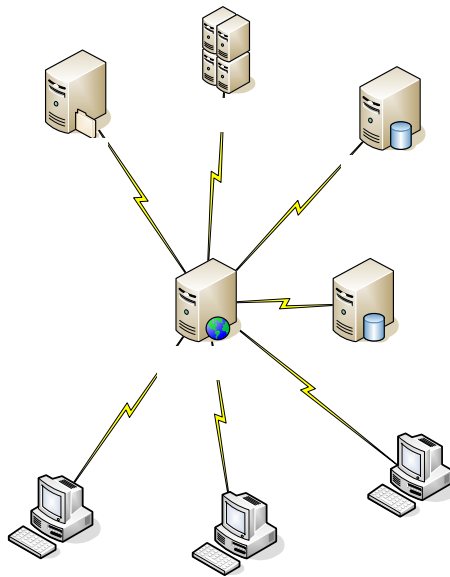


Figura 5: Arquitectura conceptual del sistema. El sistema está compuesto principalmente por un servidor web que provee a los usuarios la información consolidada de sus actividades. Ésta información está consolidada en una base de datos y se toma de diferentes fuentes externas.

Existen restricciones que afectarán el desarrollo del sistema. A continuación se listan las principales restricciones de carácter económico, técnico, operativo y legal:

² Para ahondar sobre los beneficios potenciales de una interfaz web, referirse a [Alshawi et al., 2003].

- [REP.1] Los fondos para realizar este desarrollo son muy escasos, por lo que un solo recurso estará afectado en forma parcial al mismo.
- [REP.2] Todas las herramientas utilizadas para el desarrollo deberán carecer de costo, debiéndose utilizar únicamente herramientas gratuitas o existentes en el actual ambiente de desarrollo.
- [RED.1] La solución deberá estar basada en tecnologías Microsoft (IIS, ASP, IE, Microsoft Office Automation, etc.) debido a que la implementación se realizará en ese entorno tecnológico.

4.2.1.1.2 Catálogo de objetivos del EVS

Los principales objetivos que se quieren alcanzar con el estudio de viabilidad son:

- [OEVS.1] Comprender la necesidad que da origen al proyecto en conjunto con los aspectos económicos, técnicos, legales y operativos relacionados.
- [OEVS.2] Brindar la base para decidir continuar o no con el proyecto en una etapa muy temprana del mismo, valorando alternativas.
- [OEVS.3] Obtener una propuesta de solución, identificando los requisitos que, a alto nivel, deben cumplirse.

4.2.1.1.3 Requisitos del sistema

Los requisitos funcionales del sistema, elaborados a partir de la descripción general del sistema, en conjunto con los atributos de calidad son:

- Funcionalidad:
 - o [RF.1] El sistema deberá mostrar en forma consolidada las actividades asignadas a un recurso específico.
 - o [RF.2] El sistema deberá mostrar en forma consolidada todas las actividades asignadas a un conjunto de recursos.
 - o [RF.3] El sistema deberá mostrar información básica en la lista principal de actividades y detalle a pedido del usuario.

- o [RF.4] El sistema deberá mostrar solo las actividades en curso o elegibles para ser ejecutadas.
- Fuentes de actividades:
 - o [RF.5] El sistema deberá importar actividades de archivos MS Project.
 - o [RF.6] El sistema deberá importar actividades de listas de acciones en archivos MS Excel.
 - o [RF.7] El sistema deberá importar actividades de listas de incidentes en archivos MS Excel.
- Seguridad:
 - o [RF.8] El sistema deberá proveer facilidades para manejar usuarios y roles.
 - o [RF.9] El sistema deberá mostrar a los usuarios solo la información que éstos tienen permitido ver.
 - o [RF.10] El sistema deberá proveer ayuda en línea al usuario.
- Atributos de calidad:
 - o [AC.1] El sistema debe presentar a los usuarios la información a través de interfaces web.
 - o [AC.2] El sistema deberá ser desarrollada sobre plataformas Microsoft: IIS e IE.
 - o [AC.3] El diseño del sistema debe ser modular, exponiendo una interfaz para desarrollar y acoplar módulos que permitan importar actividades de diversas fuentes.
 - o [AC.4] El sistema no debe requerir modificaciones o adaptaciones no-triviales de cualquiera de las fuentes de actividades.

Estos requisitos son la base que se tomará en las siguientes actividades y etapas del proceso para elaborar el modelo funcional y conceptual que debe cumplir el sistema a fin de obtener los objetivos para los que el proyecto ha sido iniciado.

4.2.1.2 Identificación del alcance del sistema (EVS 1.2)

Se analiza el alcance de la necesidad planteada y se identifican las restricciones relativas a la sincronización con otros proyectos, que puedan interferir en la planificación y futura puesta a punto del sistema objeto del estudio. Esta información se recoge en el catálogo de requisitos.

Una vez establecido el alcance, se identifican las unidades organizativas afectadas por el sistema, así como su estructura y responsables de las mismas. Para determinar los responsables se tiene en cuenta a quiénes afecta directamente y quiénes pueden influir en el éxito o fracaso del mismo.

4.2.1.2.1 Estructura organizativa

Dado que el sistema debe ser lo menos disruptivo posible en cuanto a la forma de trabajo de la organización en la que se implante, es de esperar que solo los equipos de trabajo de proyecto se vean afectados por el mismo. Otras unidades organizativas que compongan la organización no se verán afectadas.

4.2.1.2.2 Requisitos relativos a restricciones o dependencias con otros proyectos

Como no existen otros proyectos que afecten o se vean afectados por el desarrollo de este sistema, no se incluyen requisitos al respecto en el catálogo de requisitos del sistema.

4.2.1.2.3 Catálogo de usuarios

El desarrollo del sistema involucrará a los siguientes usuarios, los que serán tanto participantes del desarrollo como usuarios finales:

- [US.1] Líderes de proyecto: contribuirán al modelado y a la validación de los requisitos; también participarán en la evaluación del sistema terminado.
- [US.2] Miembros de proyecto: contribuirán al modelado y a la validación de los requisitos; también participarán en la evaluación del sistema terminado.

4.2.1.3 Especificación del alcance del EVS (EVS 1.3)

En función del alcance del sistema y los objetivos del Estudio de Viabilidad del Sistema, se determinan las actividades y tareas a realizar. En particular, hay que decidir si se realiza o no el estudio de la situación actual y, en el caso de considerarlo necesario, con qué objetivo. Si el sistema pertenece al ámbito de un Plan de Sistemas de Información, los criterios que pueden orientar sobre la necesidad de llevar a cabo el estudio de la situación actual dependen de la arquitectura de información propuesta, en cuanto a la identificación de los sistemas de información actuales, implicados en el ámbito de este estudio, que se haya decidido conservar.

Se identifican los usuarios participantes de las distintas unidades organizativas afectadas para la realización del Estudio de Viabilidad del Sistema, determinando previamente sus perfiles y responsabilidades.

Debe comunicarse el plan de trabajo a los usuarios identificados como implicados en el Estudio de Viabilidad, solicitando su aceptación y esperando su confirmación.

4.2.1.3.1 Objetivos del estudio de la situación actual

El estudio de la situación actual dará el marco formal que permitirá identificar los demás sistemas, aplicaciones o herramientas con las que éste sistema deberá interactuar. Los objetivos son los siguientes.

- [OESA.1] Identificar las fuentes de datos que serán consumidas por el sistema.
- [OESA.2] Identificar las características del ambiente en el cual será implantado el sistema.

4.2.1.3.2 Catálogo de usuarios

Con motivo de completar el estudio, los siguientes participantes deberán ser contactados a fin de obtener información específica sobre el ambiente de implantación:

- [US.3] Responsable de TI: proveerá información técnica sobre la plataforma y tecnologías sobre las cuales se podrá desarrollar el sistema o que impongan restricciones a dicho desarrollo.

La identificación de un usuario adicional, provocará la actualización del catálogo de usuarios.

4.2.1.3.3 Plan de trabajo

La recolección de la información necesaria para proseguir con el estudio se obtendrá de entrevistas con los diferentes usuarios involucrados. Para concretar las entrevistas, cada usuario será contactado y se obtendrá una aceptación de la participación de las mismas.

4.2.2 Estudio de la situación actual (EVS 2)

La situación actual es el estado en el que se encuentran los sistemas de información existentes en el momento en el que se inicia su estudio.

Teniendo en cuenta el objetivo del estudio de la situación actual, se realiza una valoración de la información existente acerca de los sistemas de información afectados. En función de dicha valoración, se especifica el nivel de detalle con que se debe llevar a cabo el estudio. Si es necesario, se constituye un equipo de trabajo específico para su realización y se identifican los usuarios participantes en el mismo.

Si se decide documentar la situación actual, normalmente es conveniente dividir el sistema actual en subsistemas. Si es posible se describirá cada uno de los subsistemas, valorando qué información puede ser relevante para la descripción.

Como resultado de esta actividad se genera un diagnóstico, estimando la eficiencia de los sistemas de información existentes e identificando los posibles problemas y las mejoras.

4.2.2.1 Valoración del estudio de la situación actual (EVS 2.1)

En función de los objetivos establecidos para el estudio de la situación actual, y considerando el contexto del sistema especificado en la descripción general del mismo, se identifican los sistemas de información existentes que es necesario analizar con el fin de determinar el alcance del sistema actual. Asimismo, se decide el nivel de detalle con el que se va a llevar a cabo el estudio de cada uno de los sistemas de información implicados.

Para poder abordar el estudio, se realiza previamente una valoración de la información existente acerca de los sistemas de información afectados por el EVS. Se debe decidir si se realizan o no los modelos lógicos del sistema actual o si se describe el modelo físico.

La información relativa a los sistemas de información que se decida analizar, se obtiene mediante las sesiones de trabajo que se considere necesario.

4.2.2.1.1 Descripción de la situación actual

Los sistemas que es necesario analizar son aquellos sistemas que en la Organización sean utilizados para apoyar la gestión de proyectos. Dado que los sistemas son, principalmente, repositorios de actividades, es necesario relevar para cada uno la siguiente información:

- Modo o interfaz a utilizar para la obtención de la información de las actividades.
- Características o información disponible de cada una de las actividades.

Los repositorios de actividades que se deberán analizar son los siguientes:

- Archivos de cronogramas de actividades generados con MS Project.
- Archivos de documentación y seguimiento de acciones generados con MS Excel.
- Archivos de documentación y seguimiento de incidentes (reportes de defectos) generados con MS Excel.

También existen archivos conteniendo información relacionada con riesgos, revisiones de pares y requerimientos de cambio que no serán incluidos en el análisis sino que formarán parte de una posterior etapa de desarrollo del sistema.

Debido a que la información a relevar de cada repositorio es relativamente simple, no se realizará ninguna tarea de modelado adicional.

4.2.2.2 Identificación de los usuarios participantes en el estudio de la situación actual (EVS 2.2)

En función del nivel de detalle establecido para el estudio de la situación actual, se identifican los usuarios participantes de cada una de las unidades organizativas afectadas por dicho estudio. Se informa a los usuarios identificados como implicados en el Estudio de la Situación Actual, se solicita su aceptación y se espera su confirmación.

4.2.2.2.1 Catálogo de usuarios

Para realizar el estudio de la situación actual se requerirá la participación de los líderes de proyecto, ya identificados como usuarios clave anteriormente. Por lo tanto no es necesario realizar ninguna tarea adicional al respecto.

4.2.2.3 Descripción de los sistemas de información existentes (EVS 2.3)

En esta tarea se describen los sistemas de información existentes afectados, según el alcance y nivel de detalle establecido en la tarea Valoración del Estudio de la Situación Actual (EVS 2.1), mediante sesiones de trabajo con los usuarios designados para este estudio.

4.2.2.3.1 Descripción de la situación actual

Desde un punto de vista lógico, el sistema actual está compuesto por repositorios de actividades independientes, tanto a nivel sistema (no hay relación entre los repositorios), como a nivel de datos (no hay trazabilidad entre las actividades almacenadas). Por lo tanto, es trabajo quien lleva adelante la gestión del proyecto y de quienes ejecutan las actividades del mismo, el revisar y mantener actualizado cada uno de estos repositorios.

Desde un punto de vista físico, los datos repositorios están almacenados en servidores de archivos MS Windows que son compartidos por todos los miembros del equipo y, bajo ciertas normas de seguridad para mantener la confidencialidad de la información, por todos los equipos de trabajo.

Los usuarios poseen estaciones de trabajo MS Windows con MS Excel y MS Project instalados. Las estaciones de trabajo pueden acceder a los servidores de archivos previamente mencionados y, por medio de las aplicaciones instaladas localmente, tienen acceso a la información de actividades asignadas.

En algunos casos, los usuarios no poseen MS Project instalado. Para proveer la información del cronograma de actividades a estos usuarios, los cronogramas son pasados a formato PDF y almacenados junto al cronograma en formato MS Project. Por medio de la aplicación Adobe Reader, estos usuarios pueden finalmente acceder a la información que necesitan respecto de las actividades planificadas.

4.2.2.4 Realización del diagnóstico de la situación actual (EVS 2.4)

Con el fin de elaborar el diagnóstico de la situación actual se analiza la información de los sistemas de información existentes, obtenida en la tarea anterior y se identifican problemas, deficiencias y mejoras. Estas últimas deben tenerse en cuenta en la definición de los requisitos.

4.2.2.4.1 Descripción de la situación actual

De la descripción de la situación actual, se desprende que, tanto para los responsables del proyecto como para los demás miembros de equipo, es necesario acceder a múltiples aplicaciones y repositorios de información (cronograma, incidentes, acciones, etc.) para poder realizar sus actividades diariamente. Asimismo, es necesario controlar que estén actualizados estos múltiples repositorios, lo cual trae trabajo adicional al gerente del proyecto.

Como conclusión, se puede decir que, al completar este punto, se han cumplido los objetivos planteados en la tarea EVS 1.3.

4.2.3 Definición de requisitos del sistema (EVS 3)

Esta actividad incluye la determinación de los requisitos generales, mediante una serie de sesiones de trabajo con los usuarios participantes, que hay que planificar y realizar. Una vez finalizadas, se analiza la información obtenida definiendo los requisitos y sus prioridades, que se añaden al catálogo de requisitos que servirá para el estudio y valoración de las distintas alternativas de solución que se propongan.

4.2.3.1 Identificación de las directrices técnicas y de gestión (EVS 3.1)

La realización de esta tarea permite considerar los términos de referencia para el sistema en estudio desde el punto de vista de directrices tanto técnicas como de gestión. Con este fin, se recoge información sobre los estándares y procedimientos que deben considerarse al proponer una solución, relativos a: políticas técnicas y de seguridad y directrices de planificación, de gestión de cambios y de gestión de calidad.

4.2.3.1.1 Catálogo de normas

Para la realización del presente proyecto, se considerarán las siguientes normas:

- [NG.1] El desarrollo se realizará siguiendo la metodología MÉTRICA Versión 3.
- [NG.2] El desarrollo se realizará siguiendo un ciclo de vida en cascada.
- [NG.3] La gestión de cambios será planificada y ejecutada de acuerdo al respectivo plan.

- [NG.4] La gestión de calidad será planificada y ejecutada de acuerdo al respectivo plan.
- [ND.1] Se utilizarán los siguientes lenguajes de programación y tecnologías: HTML, XML, XSL/T, VBScript, ASP, T-SQL.
- [ND.2] Los accesos a la aplicación no podrán ser anónimos, debiendo los usuarios identificarse previamente.

4.2.3.2 Identificación de requisitos (EVS 3.2)

Para la obtención de las necesidades que ha de cubrir el sistema, se realizarán sesiones de trabajo en función de la disponibilidad de los usuarios participantes.

La información obtenida luego de realizadas las sesiones de acuerdo al plan previsto se presenta en las siguientes secciones.

4.2.3.2.1 Identificación de requisitos

A continuación se muestran los requisitos funcionales relevados para el sistema:

- [RFS.1] El sistema deberá tomar información relativa a las actividades planificadas de proyecto de cronogramas generados con MS Project.
- [RFS.2] El sistema deberá tomar información relativa a actividades de proyecto de planillas generadas con MS Excel.
- [RFS.3] El sistema deberá proveer al usuario una lista consolidada en forma de tabla de todas las actividades planificadas de proyecto.
- [RFS.4] El sistema deberá proveer al usuario una lista consolidada en forma de tabla de todas las otras actividades de proyecto.
- [RFS.5] El sistema deberá permitir al usuario filtrar las listas de actividades por proyecto y por usuario.
- [RFS.6] El sistema deberá identificar al usuario.

- [RFS.7] El sistema solo mostrará al usuario la información relativa a los proyectos de los cuales es miembro de equipo.
- [RFS.8] El sistema permitirá al usuario minimizar u ocultar las tablas de actividades.
- [RFS.9] El sistema permitirá al usuario reordenar las tablas de actividades por cada una de sus columnas.
- [RFS.10] El sistema permitirá solo al usuario identificado como líder de proyecto configurar las fuentes de actividades.
- [RFS.11] El sistema permitirá solo al usuario identificado como líder de proyecto actualizar la información de actividades en base a las fuentes de actividades.
- [RFS.12] El sistema permitirá distinguir a los usuarios entre miembros de equipo y líderes de proyecto.
- [RFS.13] El sistema permitirá al líder de proyecto definir la traducción de nombres de usuario desde la forma en la que se identifican en las fuentes de información a la forma en la que están registrados en el sistema.
- [RFS.14] El sistema no mostrará actividades cerradas, terminadas o cuyo comienzo esté planificado en el futuro.
- [RFS.15] El sistema proveerá un mecanismo interactivo para el alta, baja o modificación de usuarios.
- [RFS.16] El sistema proveerá un mecanismo interactivo para el alta, baja o modificación de proyectos.
- [RFS.17] El sistema proveerá un mecanismo interactivo para la asignación de roles (miembro de equipo, líder de proyecto) a los usuarios.
- [RFS.18] El sistema proveerá un mecanismo de actualización de actividades automático para fuentes del tipo MS Excel o MS PowerPoint.

En adición a los requisitos anteriores, se deben considerar los atributos de calidad (requisitos no funcionales) identificados en la tarea EVS 1.1.

4.2.3.3 Catalogación de requisitos (EVS 3.3)

Se analiza la información obtenida en las sesiones de trabajo para la Identificación de Requisitos, definiendo y catalogando los requisitos (funcionales y no funcionales) que debe satisfacer el sistema, indicando sus prioridades. Se incluirán también requisitos relativos al entorno tecnológico.

4.2.3.3.1 Catálogo de requisitos

Debido a que la identificación de requisitos realizada en la sección anterior tiene el suficiente nivel de detalle, se toma como primera versión del catálogo de requisitos. La prioridad dada a los requisitos es la siguiente:

- Obligatorios: RFS.1 al RFS.14
- Opcionales: RFS.15 al RFS.18

4.2.4 Estudio de alternativas de solución (EVS 4)

Este estudio se centra en proponer diversas alternativas que respondan satisfactoriamente a los requisitos planteados, considerando también los resultados obtenidos en el Estudio de la Situación Actual (EVS 2).

Teniendo en cuenta el ámbito y funcionalidad que debe cubrir el sistema, puede ser conveniente realizar, previamente a la definición de cada alternativa, una descomposición del sistema en subsistemas.

En la descripción de las distintas alternativas de solución propuestas, se debe especificar si alguna de ellas está basada, total o parcialmente, en un producto existente en el mercado. Si la alternativa incluye un desarrollo a medida, se debe incorporar en la descripción de la misma un modelo abstracto de datos y un modelo de procesos, y en orientación a objetos, un modelo de negocio y un modelo de dominio.

Debido a que el proyecto en sí mismo es el producto de una evaluación de alternativas de solución previa entre las que se consideró el uso de productos existentes en el mercado para reemplazar a los actuales, esta actividad no será llevada a cabo.

4.2.5 Valoración de las alternativas (EVS 5)

Una vez descritas las alternativas se realiza una valoración de las mismas, considerando el impacto en la organización, tanto desde el punto de vista tecnológico y organizativo como de operación, y los posibles beneficios que se esperan contrastados con los costes asociados. Se realiza también un análisis de los riesgos, decidiendo cómo enfocar el plan de acción para minimizar los mismos y cuantificando los recursos y plazos precisos para planificar cada alternativa.

Debido a que no hay alternativas a considerar en este proyecto, esta actividad no será realizada.

4.2.6 Selección de la solución (EVS 6)

Antes de finalizar el Estudio de Viabilidad del Sistema, se convoca al Comité de Dirección para la presentación de las distintas alternativas de solución, resultantes de la actividad anterior. En dicha presentación, se debaten las ventajas de cada una de ellas, incorporando las modificaciones que se consideren oportunas, con el fin de seleccionar la más adecuada. Finalmente, se aprueba la solución o se determina su inviabilidad.

Debido a que el proyecto contempla solo una alternativa de solución, las tareas Convocatoria de la Presentación y Evaluación de las Alternativas y Selección no serán realizadas.

4.2.6.1 Aprobación de la solución (EVS 6.3)

Luego de una reunión con líderes de proyecto involucrados en este desarrollo, se da la aprobación formal para avanzar con el desarrollo del sistema.

Como consideración especial, se decidió no proseguir con los requisitos opcionales, los cuales serán eventualmente implementados en una posterior etapa de desarrollo.

4.3 *Análisis del sistema de información (ASI)*

El objetivo de este proceso es la obtención de una especificación detallada del sistema de información que satisfaga las necesidades de información de los usuarios y sirva de base para el posterior diseño del sistema.

En la primera actividad, Definición del Sistema (ASI 1), se lleva a cabo la descripción inicial del sistema de información, a partir de los productos generados en el proceso Estudio de Viabilidad del Sistema (EVS). Se delimita el alcance del sistema, se genera un catálogo de requisitos generales y se describe el sistema mediante unos modelos iniciales de alto nivel. Así mismo se elabora el plan de trabajo a seguir.

La definición de requisitos del nuevo sistema se realiza principalmente en la actividad Establecimiento de Requisitos (ASI 2). El objetivo de esta actividad es elaborar un catálogo de requisitos detallado, que permita describir con precisión el sistema de información, y que además sirva de base para comprobar que es completa la especificación de los modelos obtenidos en las actividades Identificación de Subsistemas de Análisis (ASI 3), Análisis de Casos de Uso (ASI 4), Análisis de Clases (ASI 5), Elaboración del Modelo de Datos (ASI 6), Elaboración del Modelo de Procesos (ASI 7) y Definición de Interfaces de Usuario (ASI 8). Hay que hacer constar que estas actividades pueden provocar la actualización del catálogo, aunque no se refleja como producto de salida en las tareas de dichas actividades, ya que el objetivo de las mismas no es crear el catálogo sino definir modelos que soporten los requisitos.

Para la obtención de requisitos en la actividad Establecimiento de Requisitos (ASI 2) se toman como punto de partida el catálogo de requisitos y los modelos elaborados en la actividad Definición del Sistema (ASI 1), completándolos mediante sesiones de trabajo con los usuarios. Estas sesiones de trabajo tienen como objetivo reunir la información necesaria para obtener la especificación detallada del nuevo sistema. A continuación se identifican las facilidades que ha de proporcionar el sistema, y las restricciones a que está sometido en cuanto a rendimiento, frecuencia de tratamiento, seguridad y control de accesos, etc. Toda esta información se incorpora al catálogo de requisitos.

En la actividad Identificación de Subsistemas de Análisis (ASI 3), se estructura el sistema de información en subsistemas de análisis, para facilitar la especificación de los distintos modelos y la traza de requisitos.

En paralelo, se generan los distintos modelos que sirven de base para el diseño. Se elaboran el modelo de clases y el de interacción de objetos, mediante el análisis de los casos de uso. Se especifican, asimismo, todas las interfaces entre el sistema y el usuario, tales como formatos de pantallas, diálogos, formatos de informes y formularios de entrada.

En la actividad Análisis de Consistencia y Especificación de Requisitos (ASI 9), se realiza la verificación y validación de los modelos, con el fin de asegurar que son:

- completos, puesto que cada modelo obtenido contiene toda la información necesaria recogida en el catálogo de requisitos;
- consistentes, ya que cada modelo es coherente con el resto de los modelos; y
- correctos, dado que cada modelo sigue unos criterios de calidad predeterminados en relación a la técnica utilizada, calidad de diagramas, elección de nombres, normas de calidad, etc.).

En la actividad Especificación del Plan de Pruebas (ASI 10), se establece el marco general del plan de pruebas, iniciándose su especificación, que se completará en el proceso Diseño del Sistema de Información (DSI).

La participación activa de los usuarios es una condición imprescindible para el análisis del sistema de información, ya que dicha participación constituye una garantía de que los requisitos identificados son comprendidos e incorporados al sistema y, por tanto, de que éste será aceptado.

La Figura 6 muestra la relación de actividades del proceso Análisis del Sistema de Información, tanto para desarrollos estructurados (caso del presente proyecto) como para desarrollos orientados a objetos, distinguiendo las que se pueden realizar en paralelo de aquellas que han de realizarse secuencialmente.

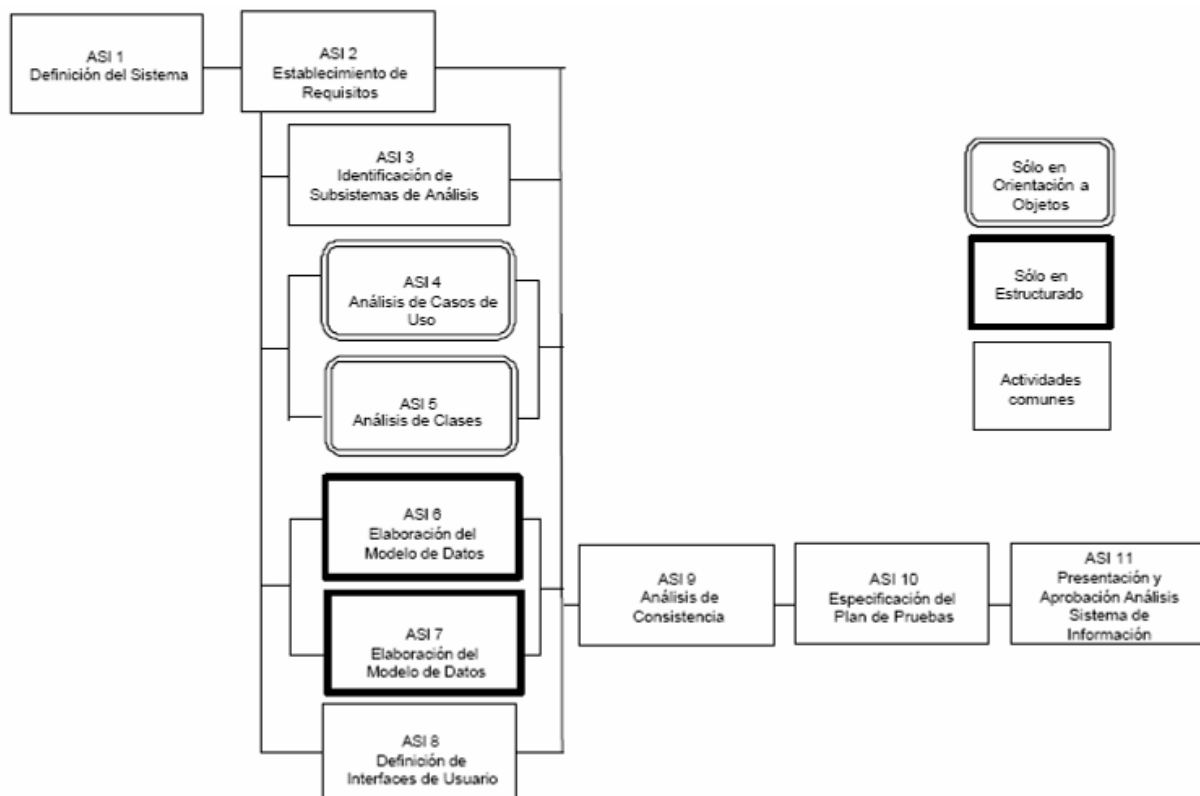


Figura 6: Actividades que componen el proceso de Análisis del Sistema de Información (ASI). Algunas de estas actividades se realizan en paralelo y otras son solo aplicables a un tipo de metodología: estructurada u orientada a objetos.

4.3.1 Definición del sistema (ASI 1)

Esta actividad tiene como objetivo efectuar una descripción del sistema, delimitando su alcance, estableciendo las interfaces con otros sistemas e identificando a los usuarios representativos. Las tareas de esta actividad se pueden haber desarrollado ya en parte en el proceso de Estudio de Viabilidad del Sistema (EVS), de modo que se parte de los productos obtenidos en dicho proceso para proceder a su adecuación como punto de partida para definir el sistema de información.

4.3.1.1 Determinación del alcance del sistema (ASI 1.1)

En esta tarea se delimita el sistema de información, utilizando como punto de partida el modelo de procesos especificado en la descripción de la solución del proceso Estudio de Viabilidad del Sistema (EVS). Se indica qué procesos pertenecen al ámbito del Sistema de Información y se identifican las entidades externas al sistema que aportan o reciben información. Asimismo, se obtiene un modelo conceptual de datos identificando las entidades y relaciones que forman parte del sistema de información objeto de este análisis.

4.3.1.1.1 Modelo de negocio

El modelo de negocio está compuesto por los usuarios del sistema y las diferentes fuentes de datos que proveerán información al sistema. Éstos son los actores principales en el sistema. En la Figura 7 se representa este modelo mediante un diagrama donde se pueden apreciar los usuarios principales, las fuentes de datos y los casos de uso de más alto nivel.

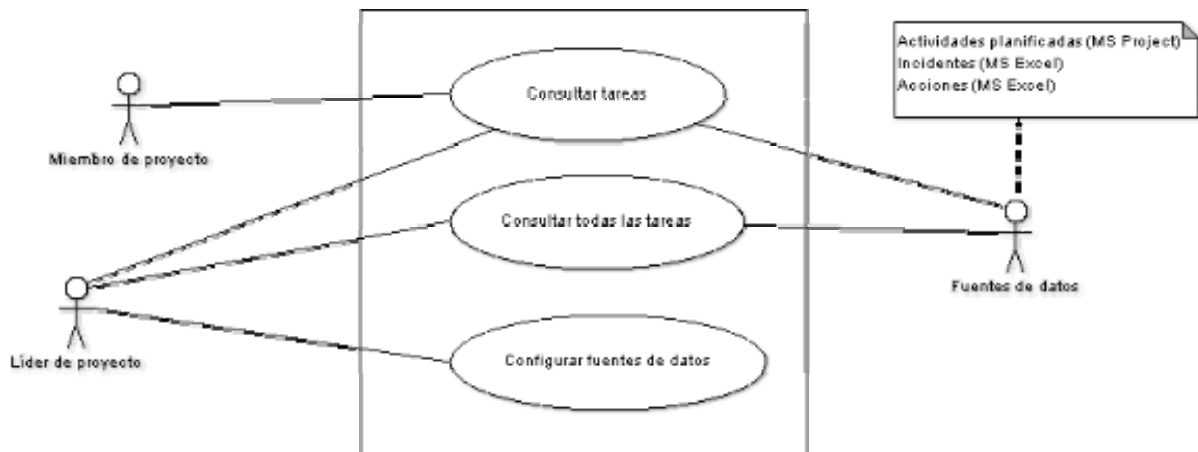


Figura 7: Este diagrama muestra el modelo de negocio del sistema, donde se aprecian los usuarios principales, líderes de proyecto y miembros de equipo, las fuentes de datos y los casos de uso de más alto nivel.

4.3.1.1.2 Modelo conceptual de datos

Del análisis inicial de los requisitos contenidos en el catálogo de requisitos, se obtienen los principales tipos de datos que el sistema manejará:

- **Proyectos:** son las entidades para las cuales las actividades están definidas.
- **Fuentes:** son los elementos externos al sistema que contienen las actividades. Cada fuente pertenece a un proyecto y cada proyecto puede tener una o más fuentes de actividades. A su vez, cada fuente puede contener o no actividades.
- **Actividades:** son las que definen qué, cuándo y cómo los miembros de los equipos de proyecto deben actuar. Las actividades están contenidas en una fuente y pueden tener uno o más usuarios asignados a la misma actividad.
- **Usuarios:** son los miembros de los equipos de proyecto, los cuales pueden llevar a cabo las actividades correspondientes a sus proyectos. Cada usuario puede tener o no actividades asignadas y puede o no pertenecer a un equipo de proyecto.

En la Figura 8 se representan estos conceptos. A través de sucesivos refinamientos del modelo, podrán aparecer nuevas entidades que deberán administrarse para poder cumplir con todos los requisitos asignados al sistema.

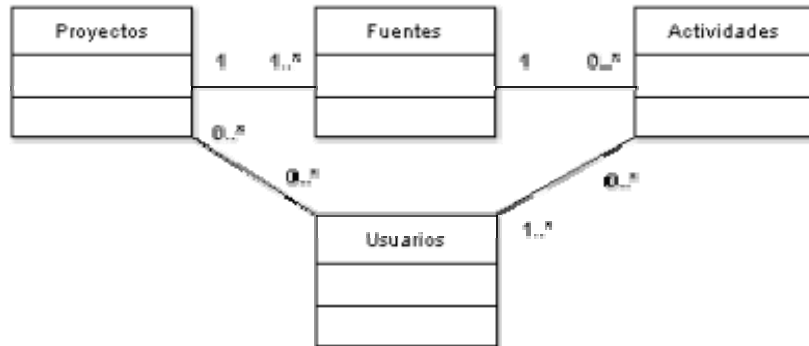


Figura 8: El sistema maneja cuatro tipos de datos: los proyectos, sus fuentes de actividades, las actividades en sí y los usuarios que llevan a cabo esas actividades y están asignados a los equipos de proyecto.

4.3.2 Establecimiento de requisitos (ASI 2)

En esta actividad se lleva a cabo la definición, análisis y validación de los requisitos a partir de la información facilitada por los usuarios, completándose el catálogo de requisitos. El objetivo de esta actividad es obtener 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.

Esta actividad se descompone en un conjunto de tareas que, si bien tienen un orden, exige continuas realimentaciones y solapamientos, entre sí y con otras tareas realizadas en paralelo. No es necesaria la finalización de una tarea para el comienzo de la siguiente. Lo que se tiene en un momento determinado es un catálogo de requisitos especificado en función de la progresión del proceso de análisis.

4.3.2.1 Obtención de requisitos (ASI 2.1)

En esta tarea comienza la obtención detallada de información mediante sesiones de trabajo con los usuarios.

Se recoge información de los requisitos que debe cumplir el software. En la definición de los requisitos, que sirven de base para establecer los niveles de servicios del sistema, hay que

tener en cuenta, si existen, las posibles restricciones del entorno, tanto hardware como software, que puedan afectar al sistema de información.

Los principales tipos de requisitos que se especificarán son:

- funcionales; y
- de seguridad.

4.3.2.1.1 Catálogo de requisitos

Tomando como base el catálogo de requisitos elaborado en la tarea de Identificación de Requisitos (EVS 3.2), se analiza cada uno de ellos y se identifican todos los requisitos funcionales que se desprenden de los mismos.

- [RFA.1] El sistema deberá importar información de actividades planificadas de proyecto.
 - o [RFA.1.1] El sistema deberá poder importar estas actividades de cronogramas generados con MS Project.
 - o [RFA.1.2] El sistema deberá importar el código de identificación (“ID”), el nombre, el estado, la fecha de comienzo, la fecha de fin y la lista de recursos asignados a la tarea.
 - o [RFA.1.3] El sistema solo importará datos de tareas que tengan recursos asignados.
 - o [RFA.1.4] El sistema solo importará datos de tareas que no estén completas.
 - [RFA.1.4.1] Cuando las tareas se importen de cronogramas generados con MS Project, el sistema solo importará datos de tareas que no estén completas (estado “Completed”).
 - o [RFA.1.5] El sistema no mostrará actividades que tengan fecha de comienzo en el futuro.

- o [RFA.1.6] El sistema traducirá, en base a la configuración de la fuente del proyecto correspondiente, el nombre del recurso en el cronograma para coincidir con el nombre con el que el recurso está registrado en el sistema.
 - o [RFA.1.7] El sistema importará los datos de cada tarea desde el archivo de cronograma de acuerdo con los parámetros definidos en la configuración de la fuente del proyecto.
- [RFA.2] El sistema deberá importar información de actividades de proyecto no planificadas (acciones preventivas y acciones correctivas).
 - o [RFA.2.1] El sistema deberá poder importar estas actividades de planillas generadas con MS Excel.
 - o [RFA.2.2] El sistema deberá importar el código de identificación (“ID”), la descripción, la fecha en la que se debe completar, la prioridad, el estado y la lista de responsables de completar la acción.
 - o [RFA.2.3] El sistema solo importará datos de acciones que tengan recursos asignados.
 - o [RFA.2.4] El sistema solo importará datos de acciones que no estén completas.
 - o [RFA.2.5] El sistema traducirá, en base a la configuración de la fuente del proyecto correspondiente, el nombre del recurso en la lista de acciones para coincidir con el nombre con el que el recurso está registrado en el sistema.
 - o [RFA.2.6] El sistema importará los datos de cada acción desde el archivo correspondiente de acuerdo con los parámetros definidos en la configuración de la fuente del proyecto.
- [RFA.3] El sistema mostrará al usuario una tabla conteniendo todas las actividades planificadas de proyecto.
 - o [RFA.3.1] La lista contendrá: proyecto, código de identificación (“ID”), nombre, estado, fecha de comienzo, fecha de fin y recurso asignado.

- o [RFA.3.2] El sistema mostrará el título “Actividades planificadas” sobre la tabla.
 - o [RFA.3.3] El sistema mostrará la cantidad de filas de la tabla entre paréntesis al lado derecho del título de la tabla de actividades planificadas.
- [RFA.4] El sistema mostrará al usuario una tabla conteniendo todas las actividades de proyecto no planificadas.
 - o [RFA.4.1] La lista contendrá: código de identificación (“ID”), descripción, fecha en la que se debe completar, prioridad, estado y responsables.
 - o [RFA.4.2] El sistema mostrará el título “Acciones” sobre la tabla.
 - o [RFA.4.3] El sistema mostrará la cantidad de filas de la tabla entre paréntesis al lado derecho del título de la tabla de acciones.
- [RFA.5] El sistema permitirá ocultar, volver a mostrar y reordenar el contenido de ambas tablas.
 - o [RFA.5.1] Los usuarios podrán reordenar la tabla en orden ascendente y descendente por cada una de sus columnas.
 - [RFA.5.1.1] Cuando el usuario haga clic sobre un nombre de columna, ésta se tomará como criterio de reordenamiento de la tabla.
 - [RFA.5.1.2] La primera solicitud de ordenamiento sobre una columna será considerada ascendente.
 - [RFA.5.1.3] Subsiguientes selecciones invertirán el orden definido por esa columna.
 - o [RFA.5.3] El usuario podrá ocultar y volver a mostrar la tabla.
 - [RFA.5.3.1] El sistema mostrará un signo “-” sobre el margen derecho a la altura del título de la tabla.
 - [RFA.5.3.2] Cuando el usuario haga clic sobre el “-”, la tabla se ocultará y el símbolo se transformará en “+”.

- [RFA.5.3.3] Cuando el usuario haga clic sobre el “+”, la tabla se volverá a mostrar y el símbolo se transformará en “-”.
- [RFA.6] Los usuarios solo podrán ver las actividades y acciones de los proyectos a los cuales están asignados.
- [RFA.7] Las tareas y acciones con varios recursos asignados, se mostrarán a razón de una fila por recurso (con el resto de la información duplicada).
- [RFA.8] El sistema deberá permitir al usuario filtrar las listas de actividades por proyecto y por usuario.
 - o [RFA.8.1] El sistema mostrará un menú desplegable para filtrar por proyecto las listas de actividades y acciones.
 - [RFA.8.1.1] El menú contendrá la opción “Todos los proyectos” y una opción por proyecto al cual el usuario esté asignado.
 - [RFA.8.1.2] Al cambiar la selección del filtro de proyecto, el sistema actualizará las tablas de actividades y acciones y actualizará las cantidades mostradas.
 - o [RFA.8.2] El sistema mostrará un menú desplegable para filtrar por usuario las listas de actividades y acciones.
 - [RFA.8.2.1] El menú contendrá la opción “Todos los usuarios” y una opción por cada usuario asignado a un proyecto al cual el usuario esté asignado.
 - [RFA.8.2.2] Al cambiar la selección del filtro de usuario, el sistema actualizará las tablas de actividades y acciones y actualizará las cantidades mostradas.
- [RFA.10] El sistema permitirá solo al usuario identificado como líder de proyecto trabajar sobre la configuración del proyecto.

- o [RFA.10.1] El sistema mostrará solo a los usuarios líderes de proyecto un botón “Configurar...” para acceder a las opciones de configuración del proyecto.
- [RFA.11] El sistema permitirá, a través de las opciones de configuración de proyecto, alterar la lista de fuentes de datos, la lista de usuarios y forzar la importación de datos desde las fuentes de datos.
 - o [RFA.11.1] El sistema permitirá ingresar y editar la configuración de cada proyecto, en formato XML, a través de una ventana de texto.
 - [RFA.11.1.1] La configuración contendrá la lista de fuentes conteniendo para cada una: nombre, tipo de fuente, dirección (URL) y tipo de actividades contenidas.
 - o [RFA.11.2] El sistema mostrará un botón titulado “Grabar cambios al proyecto” debajo de la ventana de edición de configuración.
 - [RFA.11.2.1] El botón estará habilitado solamente cuando que el texto de la configuración se corresponda con un XML bien formado.
 - o [RFA.11.3] El sistema aplicará la configuración del proyecto en forma permanente cuando el usuario haga clic en el botón “Grabar cambios al proyecto”.
 - [RFA.11.3.1] Si una fuente es agregada a la configuración de un proyecto (identificada por el nombre), ésta será registrada en el sistema.
 - [RFA.11.3.2] Si una fuente es modificada en la configuración de un proyecto (identificada por el nombre), los cambios serán registrados en el sistema.
 - [RFA.11.3.3] Si una fuente es eliminada de la configuración de un proyecto (identificada por el nombre), la misma será eliminada del sistema junto con todas las actividades que tuviera relacionadas.

- [RFA.11.3.5] Si la configuración tiene usuarios que no están registrados en el sistema, éste notificará al usuario de tal situación mediante un mensaje de alerta único por fuente registrada.
- [RFA.11.3.6] Los usuarios registrados que estén en la definición de una fuente, serán asignados al proyecto de la fuente si no hubieran estado previamente.
- o [RFA.11.4] El sistema mostrará la lista de fuentes de actividades de proyecto debajo del botón “Grabar cambios al proyecto”.
 - [RFA.11.4.1] Para cada fuente, la siguiente información será mostrada: nombre, tipo, acciones que el usuario puede disparar.
 - [RFA.11.4.2] Para fuentes del tipo MS Excel y MS Project, se mostrará un botón “Actualizar” como única opción que el usuario puede disparar.
 - [RFA.11.4.3] Cuando el usuario haga clic en el botón “Actualizar”, el sistema importará los datos de las actividades de la fuente correspondiente.
 - [RFA.11.4.4] Si el proceso de importación fuese exitoso, el sistema mostrará al usuario un mensaje de información comunicando dicha situación.
 - [RFA.11.4.5] Si durante el proceso de importación ocurriesen errores, el sistema mostrará al usuario un mensaje de alerta informando el error encontrado.
 - [RFA.11.4.6] Cuando la configuración del proyecto sea alterada, la lista de fuentes será regenerada para coincidir con la nueva configuración.
- o [RFA.11.5] En el diálogo de configuración, el sistema mostrará un menú desplegable conteniendo todos los proyectos para los cuales el usuario es líder de proyecto.
 - [RFA.11.6] La lista de proyectos aparecerá ordenada alfabéticamente.

- [RFA.11.7] Cuando el usuario cambie la selección del menú, se actualizará la configuración y la lista de fuentes mostradas para coincidir con las correspondientes al proyecto seleccionado.
- [RFA.11.6] Al salir del diálogo de configuración, el sistema refrescará toda la información de actividades mostrada previamente.

Los siguientes requisitos de seguridad se desprenden también de lo identificado previamente y del análisis correspondiente a esta tarea:

- [RFA.12] Los usuarios deberán estar identificados por el sistema antes de acceder a cualquier página web.
 - [RFA.12.1] Cualquier usuario miembro de equipo podrá acceder a la página conteniendo las listas de actividades.
 - [RFA.12.2] Solo los usuarios líder de proyecto podrán acceder a la página de configuración de proyectos.

4.3.2.1.2 Modelo de casos de uso

Del catálogo de requisitos elaborado en la tarea de Identificación de Requisitos (EVS 3.2) y de la evolución del mismo elaborado en esta misma tarea, se identificaron los casos de uso³ que describen en comportamiento funcional esperado del sistema⁴. La lista de casos de uso es la siguiente:

- [CU.1] Consultar actividades
- [CU.2] Filtrar actividades por proyecto
- [CU.3] Filtrar actividades por usuario
- [CU.4] Ocultar y mostrar lista de actividades
- [CU.5] Reordenar lista de actividades

³ Los casos de uso son elementos de modelado UML. Para más información referirse a [Rumbaugh et al., 1999] , [Booch et al., 2005] y [Jalloul, 2004].

⁴ El nombre asignado a los casos de uso es esencial para facilitar la comprensión del modelo. Para más información, referirse a [Probasco, 2001].

- [CU.6] Administrar configuración de proyecto
 - o [CU.6.1] Administrar fuentes de actividades
- [CU.7] Importar actividades manualmente
 - o [CU.7.1] Importar actividades desde MS Project
 - o [CU.7.2] Importar actividades desde MS Excel

Todos los casos de uso identificados están representados en los diagramas de la Figura 9, Figura 10, Figura 11 y Figura 12.

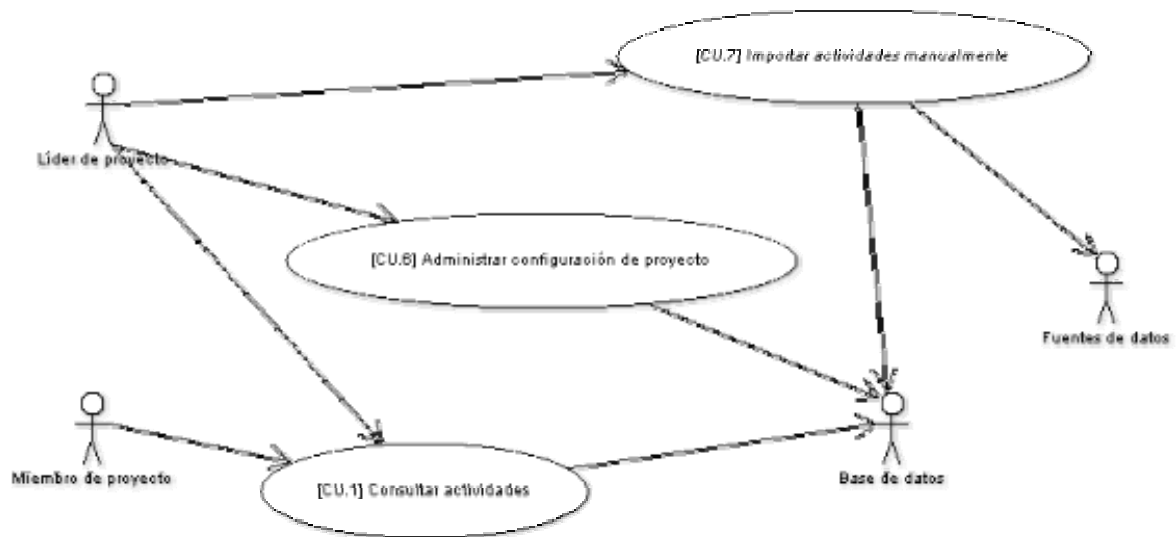


Figura 9: Diagrama de casos de uso del sistema. En el diagrama se observan los dos actores principales, los actores pasivos (fuentes de datos y base de datos) y los casos de uso de más alto nivel.

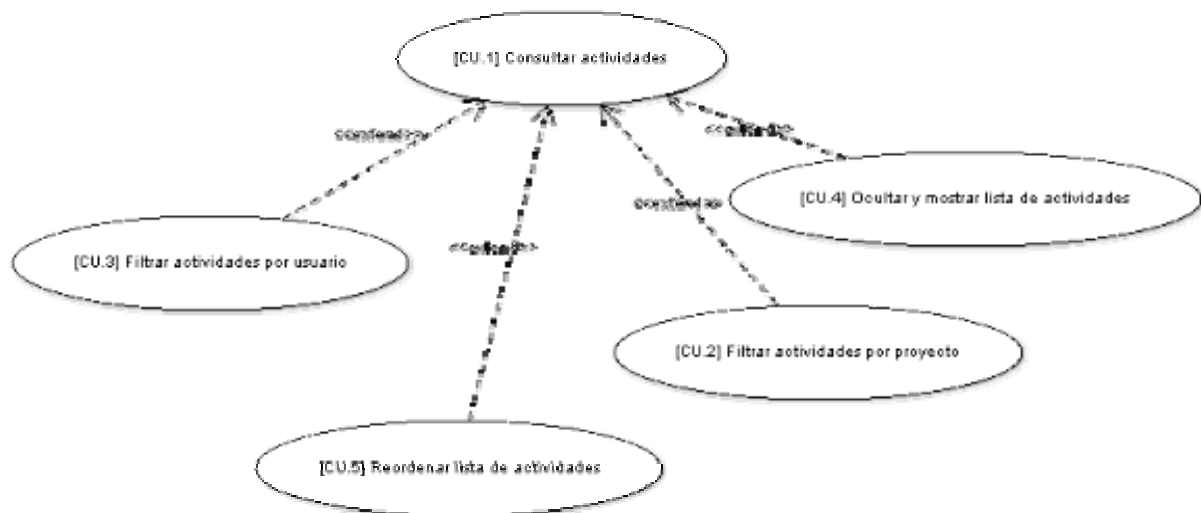


Figura 10: El caso de uso de consulta de actividades se extiende a través de otras operaciones que el usuario puede realizar: filtrar por proyecto, por usuario, ocultar y mostrar las tablas y reordenar las actividades de una tabla.



Figura 11: El caso de uso correspondiente a la administración de la configuración del proyecto incluye la administración de las fuentes de actividades correspondientes a ese proyecto.

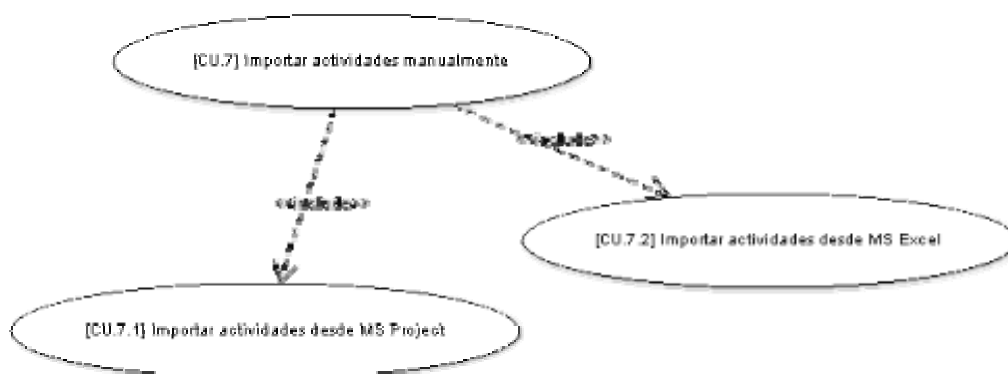


Figura 12: El caso de uso que describe la importación de actividades incluye, para la primera versión del sistema, la importación de datos de archivos generados con MS Project y con MS Excel.

4.3.2.2 Especificación de casos de uso (ASI 2.2)

El objetivo de esta tarea es especificar cada caso de uso identificado en la tarea anterior, desarrollando el escenario.

Para completar los casos de uso, es preciso especificar información relativa a:

- descripción del escenario, es decir, cómo un actor interactúa con el sistema, y cuál es la respuesta obtenida;
- precondiciones y postcondiciones; y
- condiciones de fallo que afectan al escenario, así como la respuesta del sistema (escenarios secundarios).

4.3.2.2.1 Especificación de casos de uso

A continuación se especifican cada uno de los casos de uso identificados. Se presenta una tabla por cada caso de uso especificado.

ID: CU.1	
Nombre: Consultar actividades	
Actores: - Líder de proyecto - Miembro de equipo	
Descripción: El usuario consulta las actividades a las que tiene acceso, es decir, a las asignadas a todos los usuarios de los proyectos de los cuales el mismo usuario es miembro.	
Precondiciones: - El usuario está registrado en el sistema.	
Flujo principal	Flujos alternativos
1. El usuario abre la página principal del sistema.	
2. El sistema muestra la siguiente información: un menú desplegable con la lista de proyectos del usuario, un menú desplegable con los usuarios miembros de esos proyectos, una tabla titulada “Actividades planificadas” con todas las actividades planificadas de esos usuarios, el número de actividades de la tabla, una tabla titulada “Acciones” con todas las actividades no planificadas de esos usuarios y el número de acciones de la tabla.	2.1. Adicionalmente, si el usuario es “Líder de proyecto” de algún proyecto, el sistema mostrará un botón “Configurar...”.
Postcondiciones: - No posee.	

Observaciones: - La información mostrada para las actividades planificadas es: proyecto, código de identificación (“ID”), nombre, estado, fecha de comienzo, fecha de fin y recurso asignado. - La información mostrada para las acciones es: código de identificación (“ID”), descripción, fecha en la que se debe completar, prioridad, estado y responsables. - Las actividades planificadas que se muestran son solamente aquellas que no están completas y cuya fecha de comienzo no es una fecha futura. - Las actividades con dos o más usuarios asignados, son repetidas tantas veces como usuarios tiene asignados, variando de fila en fila el nombre de usuario únicamente.
Asociaciones: - Este caso de uso es extendido por los casos CU.2, CU.3, CU.4 y CU.5.

Tabla 3: Caso de uso “Consultar actividades”.

ID: CU.2	
Nombre: Filtrar actividades por proyecto	
Actores: - Líder de proyecto - Miembro de equipo	
Descripción: El usuario cambia la selección del filtro de actividades y las listas de actividades son actualizadas en consecuencia.	
Precondiciones: - El usuario está registrado en el sistema. - El usuario está viendo las listas de actividades.	
Flujo principal	Flujos alternativos
1. El usuario despliega el menú de proyectos y ve como primera opción “Todos los proyectos” y a continuación, en orden alfabético, la lista de proyectos a los cuales está asignado.	
2. El usuario selecciona un proyecto.	
3. El sistema actualiza la lista de actividades planificadas y la de acciones mostrando solo las correspondientes a ese proyecto. También son actualizadas las cantidades de actividades que se muestran junto a los títulos de las listas.	3.1. El sistema mostrará una o ambas tablas vacías y se verán solo los títulos si el proyecto seleccionado no tuviera actividades planificadas y/o acciones. Asimismo, la cantidad de actividades mostrada será “0” en la tabla que corresponda.
4. El usuario despliega nuevamente el menú y selecciona “Todos los proyectos”.	
5. El sistema vuelve a mostrar todas las actividades.	
Postcondiciones: - No posee.	
Observaciones: - En todo momento se respetará la opción seleccionada en el filtro de usuarios. - La aplicación del filtro no alterará el estado de visualización (oculta o no) de las tablas.	
Asociaciones: - Este caso de uso extiende al caso CU.1.	

Tabla 4: Caso de uso “Filtrar actividades por proyecto”.

ID: CU.3	
Nombre: Filtrar actividades por usuario	
Actores: - Líder de proyecto - Miembro de equipo	
Descripción: El usuario cambia la selección del filtro de usuarios y las listas de actividades son actualizadas en consecuencia.	
Precondiciones: - El usuario está registrado en el sistema. - El usuario está viendo las listas de actividades.	
Flujo principal	Flujos alternativos
1. El usuario despliega el menú de usuarios y ve como primera opción “Todos los usuarios” y a continuación, en orden alfabético, la lista de nombres (completos) de los usuarios de los proyectos a los cuales está asignado.	
2. El usuario selecciona el nombre de un usuario.	
3. El sistema actualiza la lista de actividades planificadas y la de acciones mostrando solo las correspondientes a ese usuario. También son actualizadas las cantidades de actividades que se muestran junto a los títulos de las listas.	3.1. El sistema mostrará una o ambas tablas vacías y se verán solo los títulos si el usuario seleccionado no tuviera actividades planificadas y/o acciones. Asimismo, la cantidad de actividades mostrada será “0” en la tabla que corresponda.
4. El usuario despliega nuevamente el menú y selecciona “Todos los usuarios”.	
5. El sistema vuelve a mostrar todas las actividades.	
Postcondiciones: - No posee.	
Observaciones: - En todo momento se respetará la opción seleccionada en el filtro de proyectos. - La aplicación del filtro no alterará el estado de visualización (oculta o no) de las tablas.	
Asociaciones: - Este caso de uso extiende al caso CU.1.	

Tabla 5: Caso de uso “Filtrar actividades por usuario”.

ID: CU.4	
Nombre: Ocultar y mostrar lista de actividades	
Actores: - El usuario está registrado en el sistema. - El usuario está viendo las listas de actividades.	
Descripción: El usuario oculta o muestra las tablas de actividades planificadas o de acciones utilizando el símbolo que se encuentra sobre el ángulo superior derecho de la tabla correspondiente.	
Precondiciones: - El usuario está registrado en el sistema. - El usuario está viendo las listas de actividades.	
Flujo principal	Flujos alternativos
1. El usuario hace clic sobre el símbolo “[-]” que se encuentra sobre el ángulo superior derecho de la tabla.	
2. El sistema oculta la tabla correspondiente, tanto títulos como filas, y cambia el símbolo por “[+]”.	
3. El usuario hace clic sobre el símbolo “[+]” que se encuentra sobre el ángulo superior derecho de la tabla.	
2. El sistema vuelve a mostrar la tabla correspondiente, tanto títulos como filas, y cambia el símbolo por “[-]”.	
Postcondiciones: - No posee.	
Observaciones: - Al acceder por primera vez a la pantalla de actividades, el sistema mostrará las tablas y los correspondientes	

símbolos “[-]”. - El ocultamiento de ambas tablas debe ser independiente. - En todo momento se respetará la opción seleccionada en el filtro de proyectos. - En todo momento se respetará la opción seleccionada en el filtro de usuarios.
Asociaciones: - Este caso de uso extiende al caso CU.1.

Tabla 6: Caso de uso “Ocultar y mostrar lista de actividades”.

ID: CU.5	
Nombre: Reordenar lista de actividades	
Actores: - El usuario está registrado en el sistema. - El usuario está viendo las listas de actividades.	
Descripción: El usuario reordena el contenido de las listas de acuerdo a la columna seleccionada, en orden ascendente y descendente alternadamente.	
Precondiciones: - El usuario está registrado en el sistema. - El usuario está viendo las listas de actividades.	
Flujo principal	Flujos alternativos
1. El usuario selecciona el título de una columna de una de las tablas de actividades.	
2. Las filas de la tabla son reordenadas en orden ascendente en base a los datos de la columna seleccionada. Tanto textos, números y fechas serán ordenadas apropiadamente.	
3. El usuario vuelve a seleccionar el mismo título de columna.	3.1. El usuario selecciona otro título de columna.
4. Las filas de la tabla son reordenadas en orden inverso.	4.1 Las filas de la tabla son reordenadas en orden ascendente en base a los datos de la columna seleccionada.
Postcondiciones: - No posee.	
Observaciones: - En todo momento se mantendrán inalterados los datos mostrados. - El ocultar y volver a mostrar una tabla reordenada, no alterará su orden.	
Asociaciones: - Este caso de uso extiende al caso CU.1.	

Tabla 7: Caso de uso “Reordenar lista de actividades”.

ID: CU.6	
Nombre: Administrar configuración de proyecto	
Actores: - Líder de proyecto	
Descripción: El líder de proyecto altera la configuración de uno o más proyectos a través de un XML.	
Precondiciones: - El usuario está registrado en el sistema como líder de proyecto de uno o más proyectos. - El usuario está viendo las listas de actividades.	
Flujo principal	Flujos alternativos
1. El usuario presiona el botón “Configurar...” de la pantalla de actividades.	
2. El sistema abre una pantalla nueva donde se ve una lista ordenada alfabéticamente de todos los proyectos de los cuales el usuario es líder, un área de texto editable conteniendo el XML de configuración actual del proyecto seleccionado y un botón “Grabar cambios al proyecto”. Debajo se muestra una lista con las fuentes de actividades del proyecto.	2.1. Cuando el proyecto no tenga configuración asociada, el área de texto aparecerá vacía y el botón “Grabar cambios al proyecto” aparecerá deshabilitado.
3. El usuario altera el texto del área de texto editable y presiona el botón “Grabar cambios al proyecto”.	3.1. Mientras que la alteración que el usuario hace al XML genere un XML mal formado, el botón “Grabar cambios al proyecto” permanecerá deshabilitado y solo se habilitará al formar un XML correcto. 3.2. El usuario selecciona otro proyecto de la lista. 3.2.1. El sistema refresca el área de texto con el XML correspondiente.
4. El sistema almacena la nueva configuración e informa al usuario del éxito del cambio realizado mediante un cuadro de diálogo. También actualiza (sincroniza) la lista de fuentes de actividades del proyecto.	4.1. Si el sistema encontrara errores de estructura, la configuración no será almacenada y el usuario será advertido de la situación mediante un cuadro de diálogo. 4.2. Si uno o más usuarios existen en el XML de configuración pero no están dados de alta en el sistema, el usuario será notificado de tal situación mediante un cuadro de diálogo.
5. El usuario cierra la pantalla.	
6. El sistema vuelve a la pantalla de actividades y refresca el contenido de la misma.	
Postcondiciones: - La configuración de uno o más proyectos ha quedado alterada en forma permanente. - Los usuarios identificados en la configuración de un proyecto son vinculados al proyecto en forma automática.	
Observaciones: - No hay.	
Asociaciones: - Este caso de uso incluye al caso CU.6.1.	

Tabla 8: Caso de uso “Administrar configuración de proyecto”.

ID: CU.6.1	
Nombre: Administrar fuentes de actividades	
Actores: - Líder de proyecto	
Descripción: El líder de proyecto da de alta, modifica o da de baja fuentes de actividades modificando la configuración del proyecto.	
Precondiciones: - El usuario está registrado en el sistema como líder de proyecto de uno o más proyectos. - El usuario está viendo la pantalla de configuración de proyectos.	
Flujo principal	Flujos alternativos
1. El sistema muestra, en forma de tabla y debajo del área de texto que contiene el XML de configuración, la lista de fuentes de actividades del proyecto seleccionado. La siguiente información es suministrada: nombre de la fuente, el tipo y las acciones que el usuario puede disparar en forma de botones. Para las fuentes del tipo “MS Project” y “MS Excel”, la única acción disponible es “Actualizar”.	
2. El usuario altera el XML de configuración del proyecto, agregando, modificando o borrando una fuente, y selecciona “Grabar cambios al proyecto”.	
3. El sistema notifica el éxito de la modificación de la configuración y actualiza la tabla de fuentes de acuerdo a la nueva configuración.	
Postcondiciones: - Las fuentes nuevas y las modificaciones a fuentes existentes son registradas en forma permanente en el sistema. - Las fuentes modificadas o eliminadas arrastran la eliminación de todas las actividades relacionadas con las mismas.	
Observaciones: - Las fuentes son identificadas por nombre dentro de un proyecto.	
Asociaciones: - Este caso de uso está incluido en caso CU.6.	

Tabla 9: Caso de uso “Administrar fuentes de actividades”.

ID: CU.7	
Nombre: Importar actividades manualmente	
Actores: - Líder de proyecto	
Descripción: El líder de proyecto solicita al sistema que éste tome las actividades desde una fuente determinada y las almacena para su posterior consulta.	
Precondiciones: - El usuario está registrado en el sistema como líder de proyecto de uno o más proyectos. - El usuario está viendo la pantalla de configuración de proyectos. - El proyecto seleccionado posee al menos una fuente de actividades.	
Flujo principal	Flujos alternativos
1. El usuario presiona el botón “Actualizar” correspondiente a una fuente de datos.	
2. El sistema obtiene las actividades desde la fuente correspondiente.	2.1. El sistema no puede acceder a la fuente de datos. 2.1.1. El sistema cancela la operación y notifica al usuario acerca del problema mediante un cuadro de diálogo.
3. El sistema traduce la información obtenida a desde el formato de la fuente al del sistema, filtra las actividades completas o sin recursos y almacena la información.	3.1. El sistema detecta cualquier problema al almacenar la información. 3.1.1. El sistema cancela la operación y notifica al usuario acerca del problema mediante un cuadro de diálogo.
4. El sistema informa al usuario acerca del éxito de la operación a través de un cuadro de diálogo.	
Postcondiciones: - Las actividades que el sistema pudiera contener previamente a la solicitud de actualización son eliminadas. - Las actividades correspondientes a esa fuente y a ese proyecto son actualizadas en el sistema.	
Observaciones: - El sistema deberá poder traducir el estado y la prioridad de las actividades, las fechas y los nombres de los usuarios desde la forma en la que están almacenadas en la fuente y la forma que el sistema requiere para poder consolidar la información obtenida de distintas fuentes. - El sistema deberá poder reconocer qué actividades no debe importar a través de filtros que actuarán sobre los datos importados desde las fuentes (ej.: no importar actividades marcadas como cerradas).	
Asociaciones: - Este caso de uso incluye al caso CU.7.1 y al CU.7.2.	

Tabla 10: Caso de uso “Importar actividades manualmente”.

ID: CU.7.1	
Nombre: Importar actividades desde MS Project	
Actores: - Líder de proyecto	
Descripción: El líder de proyecto solicita al sistema importar actividades de un cronograma de actividades creado con MS Project.	
Precondiciones: - El usuario está registrado en el sistema como líder de proyecto de uno o más proyectos. - El usuario está viendo la pantalla de configuración de proyectos. - El proyecto seleccionado posee al menos una fuente de actividades del tipo “MS Project”. - El usuario presionó el botón “Actualizar” correspondiente a una fuente del tipo “MS Project”.	
Flujo principal	Flujos alternativos
1. El sistema abre el cronograma de acuerdo a la ruta especificada en la configuración.	1.1. El sistema no puede abrir el archivo. 1.1.1. El sistema cancela la operación y notifica al usuario acerca del problema mediante un cuadro de diálogo.
2. El sistema obtiene la siguiente información: código de identificación (“ID”), nombre, estado, fecha de comienzo, fecha de fin y lista de recursos asignados a la tarea.	
3. El sistema filtra las actividades de acuerdo a lo especificado en la configuración.	
4. El sistema traduce los datos al formato común: estado y nombre de los recursos.	4.1. El sistema no encuentra como traducir un nombre de usuario. 4.1.1. La actividad correspondiente para ese usuario es descartada. 4.2. El sistema no encuentra como traducir un estado. 4.2.1. La actividad correspondiente es descartada.
5. El sistema almacena la información de las actividades.	5.1. El sistema encuentra problemas al almacenar la información. 1.1.1. El sistema cancela la operación y notifica al usuario acerca del problema mediante un cuadro de diálogo.
Postcondiciones: - Las nuevas actividades están almacenadas en el sistema en reemplazo de las existentes.	
Observaciones: - Para que el sistema no importe las actividades cerradas, la configuración deberá contener un filtro que las identifique.	
Asociaciones: - Este caso de uso está incluido en caso CU.7, paso 3.	

Tabla 11: Caso de uso “Importar actividades desde MS Project”.

ID: CU.7.2	
Nombre: Importar actividades desde MS Excel	
Actores: - Líder de proyecto	
Descripción: El líder de proyecto solicita al sistema importar actividades de una planilla creada con MS Excel.	
Precondiciones: - El usuario está registrado en el sistema como líder de proyecto de uno o más proyectos. - El usuario está viendo la pantalla de configuración de proyectos. - El proyecto seleccionado posee al menos una fuente de actividades del tipo “MS Excel”. - El usuario presionó el botón “Actualizar” correspondiente a una fuente del tipo “MS Excel”.	
Flujo principal	Flujos alternativos
1. El sistema abre la planilla de acuerdo a la ruta especificada en la configuración.	1.1. El sistema no puede abrir el archivo. 1.1.1. El sistema cancela la operación y notifica al usuario acerca del problema mediante un cuadro de diálogo.
2. El sistema obtiene la siguiente información: código de identificación (“ID”), descripción, fecha en la que se debe completar, prioridad, estado y lista de responsables de completar la acción.	
3. El sistema filtra las actividades de acuerdo a lo especificado en la configuración.	
4. El sistema traduce los datos al formato común: estado, prioridad y nombre de los recursos.	4.1. El sistema no encuentra como traducir un nombre de usuario. 4.1.1. La actividad correspondiente para ese usuario es descartada. 4.2. El sistema no encuentra como traducir un estado o prioridad. 4.2.1. La actividad correspondiente es descartada.
5. El sistema almacena la información de las actividades.	5.1. El sistema encuentra problemas al almacenar la información. 1.1.1. El sistema cancela la operación y notifica al usuario acerca del problema mediante un cuadro de diálogo.
Postcondiciones: - Las nuevas actividades están almacenadas en el sistema en reemplazo de las existentes.	
Observaciones: - El sistema deberá hacer uso de la información provista en la configuración del proyecto para identificar la hoja de la planilla donde se encuentra la información a importar, la fila conteniendo la primera actividad y las columnas que contienen cada uno de los datos a importar. Adicionalmente para los usuarios, se deberá especificar el separador que se usa para los casos de múltiples asignaciones. - Para que el sistema no importe las actividades cerradas, la configuración deberá contener un filtro que las identifique.	
Asociaciones: - Este caso de uso está incluido en caso CU.7, paso 3.	

Tabla 12: Caso de uso “Importar actividades desde MS Excel”.

4.3.2.3 Validación de requisitos (ASI 2.4)

Mediante esta tarea, los usuarios confirman que los requisitos especificados en el catálogo de requisitos, así como los casos de uso, son válidos, consistentes y completos.

Luego de una sesión con usuarios de la aplicación (líderes de proyecto y miembros de equipo identificados previamente para colaborar con el proyecto), la lista de requisitos fue aceptada.

4.3.3 Análisis de los casos de uso (ASI 4)

El objetivo de esta actividad es identificar las clases cuyos objetos son necesarios para realizar un caso de uso y describir su comportamiento mediante la interacción dichos objetos.

Esta actividad se lleva a cabo para cada uno de los casos de uso contenidos en un subsistema de los definidos en la actividad Identificación de Subsistemas de Análisis (ASI 3). Las tareas de esta actividad no se realizan de forma secuencial sino en paralelo, con continuas realimentaciones entre ellas y con las realizadas en las actividades Establecimiento de Requisitos (ASI 2), Identificación de Subsistemas de Análisis (ASI 3), Análisis de Clases (ASI 5) y Definición de Interfaces de Usuario (ASI 8).

4.3.3.1 Identificación de clases asociadas a un caso de uso (ASI 4.1)

En esta tarea se comienzan a identificar los objetos necesarios para realizar el caso de uso, basándose en la especificación que tenemos del mismo.

A partir del estudio del caso de uso, se extrae una lista de objetos candidatos a ser clases. Es posible que, inicialmente, no se disponga de la información necesaria para identificar todas, por lo que se hace una primera aproximación que se va refinando posteriormente, durante esta actividad y en el proceso de diseño. Además, algunos de los objetos representan mejor la información del sistema si se les identifica como atributos en vez de como clases. Para poder diferenciarlas, es necesario estudiar el comportamiento de esos objetos en el diagrama de interacción y además se debe tener en cuenta una serie de reglas, como puede ser el suprimir palabras no pertinentes, con significados vagos o sinónimos.

Una vez definidas cada una de las clases, se incorporan al modelo de clases de la actividad Análisis de Clases (ASI 5), donde se identifican sus atributos, responsabilidades y relaciones.

Las clases que se identifican en esta tarea pueden ser:

- Clases de Entidad (representan la información manipulada en el caso de uso).
- Clases de Interfaz de Usuario (se utilizan para describir la interacción entre el sistema y sus actores. Suelen representar abstracciones de ventanas, interfaces de comunicación, formularios, etc.).

- Clases de Control (son responsables de la coordinación, secuencia de transacciones y control de los objetos relacionados con un caso de uso).

4.3.3.1.1 Modelo de clases de análisis

La Figura 13 contiene todas las clases de análisis identificadas a través del estudio de los casos de uso junto con sus atributos y operaciones más relevantes.

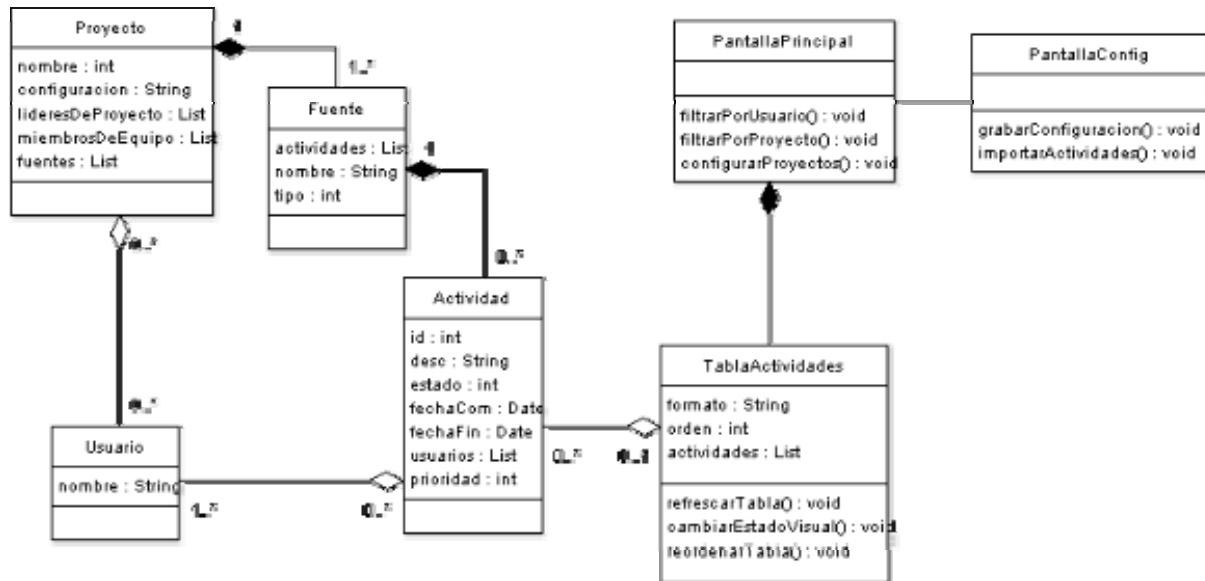


Figura 13: Modelo de clases de análisis conteniendo las clases identificadas y los atributos y operaciones más relevantes.

4.3.4 Análisis de clases (ASI 5)

El objetivo de esta actividad es describir cada una de las clases que ha surgido, identificando las responsabilidades que tienen asociadas, sus atributos, y las relaciones entre ellas.

Teniendo en cuenta las clases identificadas en la actividad Análisis de los Casos de Uso (ASI 4), se elabora el modelo de clases para cada subsistema. A medida que avanza el análisis, dicho modelo se va completando con las clases que vayan apareciendo, tanto del estudio de los casos de uso, como de la interfaz de usuario necesaria para el sistema de información.

4.3.4.1 Identificación de responsabilidades y atributos (ASI 5.1)

El objetivo de esta tarea es identificar las responsabilidades y atributos relevantes de una clase.

Las responsabilidades de una clase definen la funcionalidad de esa clase, y están basadas en el estudio de los papeles que desempeñan sus objetos dentro de los distintos casos de uso. A partir de estas responsabilidades, se puede comenzar a encontrar las operaciones que van a pertenecer a la clase. Estas deben ser relevantes, simples, y participar en la descripción de la responsabilidad.

Los atributos de una clase especifican propiedades de la clase, y se identifican por estar implicados en sus responsabilidades. Los tipos de estos atributos deberían ser conceptuales y conocidos en el dominio.

4.3.4.1.1 Modelo de clases de análisis

Para cada una de las clases identificadas en el modelo de clases de análisis, se describe cada uno de sus atributos y operaciones en la Tabla 13.

Clase	Atributos y operaciones		Descripción
Proyecto	Atributos	nombre	Es un texto descriptivo del nombre del proyecto.
		configuración	Es un XML (texto) que contiene toda la configuración del proyecto: nombre, fuentes de datos, sus actividades y usuarios.
		fuentes	En base a la configuración, el proyecto está vinculado a una lista de fuentes de actividades.
		lideresDeProyecto	Es una lista de usuarios identificados como líderes de proyecto de un proyecto específico.
		miembrosDeEquipo	Es una lista de usuarios identificados como miembros de equipo de un proyecto específico.
Fuente	Atributos	nombre	Es un texto descriptivo del nombre de la fuente de actividades del proyecto.
		tipo	Es el atributo que se utilizará para identificar el tipo de fuente, por ejemplo “MS Project” o “MS Excel”.
		actividades	Es una lista de todas las actividades contenidas en una fuente específica.
Actividad	Atributos	id	Es el identificador de actividad proveniente de la fuente de actividades. No es el identificador de actividad propio del sistema.
		desc	Es un texto que describe la actividad planificada o la acción que hay que llevar a cabo.
		fechaCom	Es la fecha de comienzo de la actividad planeada. Para las acciones, es la fecha en la que la acción se da de alta en la fuente.
		fechaFin	Es la fecha de fin de la actividad planeada. Para las acciones, es la fecha en la que la acción debería estar completa.
		prioridad	Es la prioridad de la acción. En general no se usa para actividades planeadas.
		estado	Es el estado de la actividad, por ejemplo “En curso” o “No iniciada”.
		usuarios	Es una lista de usuarios asociados o asignados a una actividad específica.

Clase	Atributos y operaciones		Descripción
Usuario	Atributos	nombre	Es un texto conteniendo el nombre del usuario como se lo ha registrado en el sistema. Si el nombre difiere en alguna de las fuentes, éste deberá ser traducido. Dicha traducción forma parte de la configuración de la fuente.
PantallaPrincipal	Operaciones	filtrarPorProyecto	Por medio de esta operación, las listas de actividades son filtradas por proyecto.
		filtrarPorUsuario	Por medio de esta operación, las listas de actividades son filtradas por usuario.
		configurarProyectos	Por medio de esta operación, los líderes de proyecto pueden acceder a la pantalla de configuración de proyectos.
PantallaConfig	Operaciones	grabarConfiguracion	Por medio de esta operación, la configuración de un proyecto específico es almacenada en forma permanente en el sistema.
		importarActividades	Por medio de esta operación, un líder de proyecto fuerza manualmente la importación de actividades desde una fuente de actividades determinada.
TablaActividades	Atributos	formato	Es un XSL (texto) que, en base a la información de las actividades, da formato a la tabla. Define títulos, columnas, visibilidad de actividades, etc.
		actividades	Es la lista de actividades relacionadas con la tabla que el usuario puede ver: todas las actividades de los proyectos a los que el usuario está vinculado.
		orden	Es un contenedor del criterio con el que la tabla debe estar ordenada. Esto incluye la columna que dicta el ordenamiento y la dirección (ascendente o descendente).
	Operaciones	refrescarTabla	Por medio de esta operación, la tabla es actualizada en base a los filtros de proyecto y usuarios. Eventualmente podría ser usada si hubiera cambios en las actividades, por ejemplo: luego de una importación manual.
		reordenarTabla	Por medio de esta operación, la tabla es actualizada en base al criterio de ordenamiento seleccionado por el usuario.
		cambiarEstadoVisual	Por medio de esta operación, la tabla es ocultada o mostrada en forma alternada.

Tabla 13: Descripción de los atributos y operaciones más importantes de cada una de las clases de análisis identificadas en el modelo de clases de análisis.

4.3.5 Definición de interfaces de usuario (ASI 8)

En esta actividad se especifican las interfaces entre el sistema y el usuario: formatos de pantallas y diálogos. El objetivo es realizar un análisis de los procesos del sistema de información en los que se requiere una interacción del usuario, con el fin de crear una interfaz que satisfaga todos los requisitos establecidos, teniendo en cuenta los diferentes perfiles a quienes va dirigido.

Al comienzo de este análisis es necesario seleccionar el entorno en el que es operativa la interfaz y establecer las directrices aplicables en los procesos de diseño y construcción. El

propósito es construir una interfaz de usuario acorde a sus necesidades, flexible, coherente, eficiente y sencilla de utilizar, teniendo en cuenta la facilidad de cambio a otras plataformas, si fuera necesario.

Se identifican los distintos grupos de usuarios de acuerdo con las funciones que realizan, conocimientos y habilidades que poseen, y características del entorno en el que trabajan. La identificación de los diferentes perfiles permite conocer mejor las necesidades y particularidades de cada uno de ellos.

Finalmente, se define el formato y contenido de cada una de las interfaces de pantalla especificando su comportamiento dinámico.

Como resultado de esta actividad se genera la especificación de interfaz de usuario, como producto que engloba los siguientes elementos:

- principios generales de la interfaz;
- catálogo de perfiles de usuario;
- descomposición funcional en diálogos;
- catálogo de controles y elementos de diseño de interfaz de pantalla;
- formatos individuales de interfaz de pantalla;
- modelo de navegación de interfaz de pantalla;
- formatos de impresión;
- prototipo de interfaz interactiva; y
- prototipo de interfaz de impresión.

4.3.5.1 Especificación de principios generales de la interfaz (ASI 8.1)

El objetivo de esta tarea es especificar los estándares, directrices y elementos generales a tener en cuenta en la definición de la interfaz de usuario.

En primer lugar, se selecciona el entorno de la interfaz interactiva (gráfico, carácter, etc.), siguiendo estándares internacionales y de la instalación, y se determinan los principios de diseño de la interfaz de usuario, contemplando:

- directrices generales en cuanto a la interfaz y aspectos generales de interacción;
- principios de composición de pantallas y criterios de ubicación de los distintos elementos dentro de cada formato;
- normas para los mensajes de error y aviso, codificación, presentación y comportamientos; y
- normas para la presentación de ayudas.

Hay que establecer criterios similares para la interfaz impresa:

- directrices generales;
- principios de composición de informes y formularios; y
- normas de elaboración, distribución y salvaguarda de la información.

4.3.5.1.1 Especificación de Interfaz de Usuario

La interfaz de usuario será gráfica e interactiva, del tipo estándar utilizado en las aplicaciones basadas en entornos web.

Los lineamientos principales para la construcción de la interfaz de usuario son los siguientes:

- [PGI.1] La activación de las distintas operaciones del sistema se realiza mediante el uso de controles estándar de interfaces web.
- [PGI.2] Los mensajes de error se mostrarán mediante cuadros de diálogo emergentes.
- [PGI.3] En todos los menús desplegables sólo estarán activadas las opciones que pueden ser utilizadas. Aquellas opciones que no pueden ser utilizadas en un momento determinado, no se mostrarán.

4.3.5.2 Especificación del comportamiento dinámico de la interfaz (ASI 8.4)

El objetivo de esta tarea es definir los flujos entre los distintos formatos de interfaz de pantalla, y también dentro del propio formato. Este comportamiento se describe mediante un modelo de navegación de interfaz de pantalla.

Para cada formato individual de pantalla o ventana se establece la entrada lógica de los datos y las reglas de validación, incluyendo dependencia de valores (reflejo de los requisitos de validación de sistema).

Se analiza y determina la secuencia de acciones específicas para completar cada diálogo, tal y como se ejecuta en el ámbito de la interfaz, así como las condiciones que se deben cumplir para su inicio, y las posibles restricciones durante su ejecución. El comportamiento está dirigido y representado por los controles y los eventos que provocan su activación.

Se identifican aquellos diálogos o formatos considerados críticos para el correcto funcionamiento del sistema, basándose en el número de usuarios, frecuencia de uso, datos implicados, alcance de las funciones asociadas al diálogo, diálogos comunes a diferentes funciones, marco de seguridad establecido en los requisitos del sistema, etc.

Para los diálogos o comportamientos complejos de interfaz se propone la técnica de diagrama de transición de estados, siendo suficiente en la mayoría de los casos una especificación del comportamiento con matrices control / evento / acción, detallándose la acción con una descripción textual.

4.3.5.2.1 Modelo de navegación de interfaz de pantalla

El sistema tiene dos pantallas principales: la que muestra las actividades y la que se usa para configurar los proyectos. También hay una serie de cuadros de diálogo relacionados con la pantalla de configuración: dos están relacionados con el almacenamiento o actualización de la configuración de los proyectos y dos con la importación de actividades. La relación entre las interfaces se puede ver en la Figura 14.

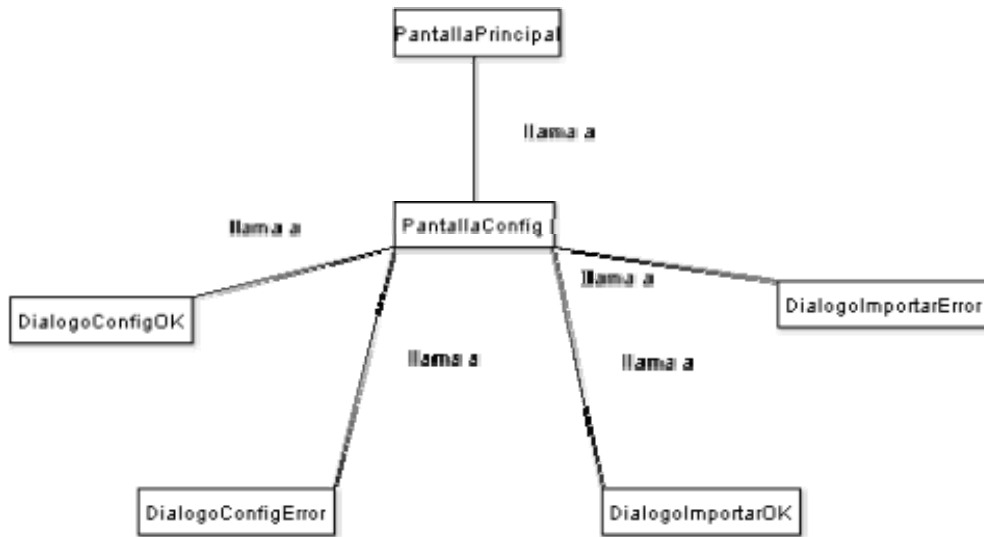


Figura 14: Las interfaces principales del sistema son la pantalla principal, donde se visualizan las actividades, y la pantalla de configuración de proyectos. Ésta, a su vez, llama a una serie de cuadros de diálogo.

La pantalla principal deberá contener los siguientes bloques:

- en la parte superior, sobre la izquierda, filtros de usuarios y proyectos; sobre la derecha, un botón para acceder a la ventana de configuración;
- debajo, una tabla donde se mostrarán en filas las actividades planificadas; y
- debajo de esta, una tabla similar donde se mostrarán las acciones.

El bosquejo de la Figura 15 representa la organización de los principales bloques que compondrán la pantalla principal.

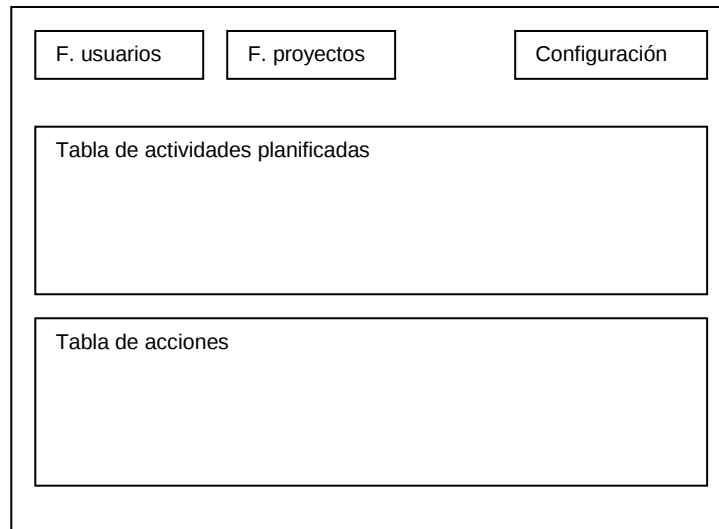


Figura 15: Bosquejo representativo de la disposición de los componentes de la pantalla principal.

Por su lado, la pantalla de configuración deberá estar compuesta por los siguientes bloques:

- en la parte superior izquierda un filtro de proyectos; sobre la derecha, un botón para cerrar el diálogo;
- debajo, un área de texto editable donde se visualizará y editará la configuración del proyecto junto con un botón para guardar la configuración; y
- por último, una tabla conteniendo todas las fuentes de datos del proyecto, junto con botones de acción por cada fuente.

El bosquejo de la Figura 16 representa la organización de los bloques que compondrán la pantalla de configuración.

Diagrama de la pantalla de configuración. El diseño incluye:

- Una barra superior con dos botones: "F. proyectos" a la izquierda y "Cerrar" a la derecha.
- Un área central grande etiquetada como "Área editable de configuración".
- Un botón "Botón guardar" situado debajo del área de configuración.
- Una sección inferior titulada "Tabla de fuentes de datos" que contiene una tabla con dos filas:

Tabla de fuentes de datos	
Fuente 1	Botón acción
Fuente n	Botón acción

Figura 16: Bosquejo representativo de la disposición de los componentes de la pantalla de configuración.

Las demás pantallas que componen el sistema son cuadros de diálogo simples y se utilizarán y seguirán los estándares disponibles. En todos estos casos, los diálogos contendrán la siguiente información y estructura:

- un ícono de información, exclamación o advertencia (error crítico);
- el mensaje de información o descripción del error; y
- un botón "OK" para cerrar el diálogo.

4.3.6 Análisis de consistencia y especificación de requisitos (ASI 9)

El objetivo de esta actividad es garantizar la calidad de los distintos modelos generados en el proceso de Análisis del Sistema de Información, y asegurar que los usuarios y los Analistas tienen el mismo concepto del sistema. Para cumplir dicho objetivo, se llevan a cabo las siguientes acciones:

- verificación de la calidad técnica de cada modelo;
- aseguramiento de la coherencia entre los distintos modelos; y

- validación del cumplimiento de los requisitos.

Esta actividad requiere una herramienta de apoyo para realizar el análisis de consistencia. También se elabora en esta actividad la Especificación de Requisitos Software (ERS), como producto para la aprobación formal, por parte del usuario, de las especificaciones del sistema.

La Especificación de Requisitos Software se convierte en la línea base para los procesos posteriores del desarrollo del software, de modo que cualquier petición de cambio en los requisitos que pueda surgir posteriormente, debe ser evaluada y aprobada.

4.3.6.1 Verificación de los modelos (ASI 9.1)

El objetivo de esta tarea es asegurar la calidad formal de los distintos modelos, conforme a la técnica seguida para la elaboración de cada producto y a las normas determinadas en el Catálogo de Normas.

4.3.6.1.1 Verificación

Se verificaron los distintos modelos de forma independiente con foco en la calidad de cada uno para garantizar su adecuación al problema a resolver y a la técnica seleccionada.

Como resultado, se efectuaron actualizaciones que fueron incorporadas a los diferentes modelos.

4.3.6.2 Análisis de consistencia entre modelos (ASI 9.2)

El objetivo de esta tarea es asegurar que los modelos son coherentes entre sí, comprobando la falta de ambigüedades o duplicación de información.

Las diferentes comprobaciones varían en función del tipo de desarrollo, aunque, en general, son matrices entre los elementos comunes de los distintos modelos. Estas comprobaciones forman parte del producto Resultado de Análisis de Consistencia.

Considerando que la interfaz de usuario incluye diagramas dinámicos y forma parte del modelo de clases, los análisis de consistencia con la interfaz pueden solaparse con los del resto de los modelos. Los análisis de consistencia propuestos son:

4.3.6.2.1 Matrices de trazabilidad

La primera matriz para analizar la consistencia entre los modelos es la que cruza los requisitos funcionales del Catálogo de Requisitos con los casos de uso del Modelo de Casos de Uso. Dicha matriz se muestra en la Tabla 14.

Requisito \ CU	CU.1	CU.2	CU.3	CU.4	CU.5	CU.6	CU.6.1	CU.7	CU.7.1	CU.7.2
RFA.1								x		
RFA.1.1									x	
RFA.1.2									x	
RFA.1.3								x		
RFA.1.4								x		
RFA.1.4.1									x	
RFA.1.5	x									
RFA.1.6								x		
RFA.1.7								x		
RFA.2								x		
RFA.2.1										x
RFA.2.2										x
RFA.2.3								x		
RFA.2.4								x		
RFA.2.5								x		
RFA.2.6								x		
RFA.3	x									
RFA.3.1	x									
RFA.3.2	x									
RFA.3.3	x	x	x							
RFA.4	x									
RFA.4.1	x									
RFA.4.2	x									
RFA.4.3	x	x	x							
RFA.5					x					
RFA.5.1					x					
RFA.5.1.1					x					
RFA.5.1.2					x					
RFA.5.1.3					x					
RFA.5.3				x						
RFA.5.3.1				x						
RFA.5.3.2				x						
RFA.5.3.3				x						
RFA.6	x									
RFA.7	x									
RFA.8		x	x							
RFA.8.1		x								
RFA.8.1.1		x								
RFA.8.1.2		x								
RFA.8.2			x							
RFA.8.2.1			x							
RFA.8.2.2			x							
RFA.10						x	x			
RFA.10.1	x					x				
RFA.11							x			
RFA.11.1						x				
RFA.11.1.1							x			

Requisito \ CU	CU.1	CU.2	CU.3	CU.4	CU.5	CU.6	CU.6.1	CU.7	CU.7.1	CU.7.2
RFA.11.2						x				
RFA.11.2.1						x				
RFA.11.3						x				
RFA.11.3.1							x			
RFA.11.3.2							x			
RFA.11.3.3							x			
RFA.11.3.5						x				
RFA.11.3.6						x				
RFA.11.4							x			
RFA.11.4.1							x			
RFA.11.4.2							x			
RFA.11.4.3								x		
RFA.11.4.4								x		
RFA.11.4.5								x		
RFA.11.4.6						x				
RFA.11.5						x				
RFA.11.6						x				
RFA.11.7						x				
RFA.11.6						x				
RFA.12	x					x				
RFA.12.1	x									
RFA.12.2						x	x			

Tabla 14: Matriz de trazabilidad entre los requerimientos del Catálogo de Requisitos y los casos de uso del Modelo de Casos de Uso

La segunda matriz para analizar la consistencia entre los modelos es la que cruza los casos de uso del Modelo de Casos de Uso con las clases del Modelo de Clases de Análisis. Dicha matriz se muestra en la Tabla 15.

Clase \ CU	CU.1	CU.2	CU.3	CU.4	CU.5	CU.6	CU.6.1	CU.7	CU.7.1	CU.7.2
Proyecto	x	x				x				
Fuente						x	x			
Actividad	x	x	x	x	x			x	x	x
Usuario	x		x					x	x	x
PantallaPrincipal	x	x	x	x	x					
PantallaConfig						x	x	x	x	x
TablaActividades	x	x	x	x	x					

Tabla 15: Matriz de trazabilidad entre los casos de uso del Modelo de Casos de Uso y las clases del Modelo de Clases Análisis.

4.3.7 Especificación del plan de pruebas (ASI 10)

En esta actividad se inicia la definición del plan de pruebas, el cual sirve como guía para la realización de las pruebas, y permite verificar que el sistema de información cumple las necesidades establecidas por el usuario, con las debidas garantías de calidad.

El plan de pruebas es un producto formal que define los objetivos de la prueba de un sistema, establece y coordina una estrategia de trabajo, y provee del marco adecuado para elaborar una planificación paso a paso de las actividades de prueba. El plan se inicia en el proceso Análisis del Sistema de Información (ASI), definiendo el marco general, y estableciendo los requisitos de prueba de aceptación, relacionados directamente con la especificación de requisitos.

Dicho plan se va completando y detallando a medida que se avanza en los restantes procesos del ciclo de vida del software, Diseño del Sistema de Información (DSI), Construcción del Sistema de Información (CSI) e Implantación y Aceptación del Sistema (IAS).

Se plantean los siguientes niveles de prueba:

- pruebas unitarias;
- pruebas de integración;
- pruebas del sistema;
- pruebas de implantación; y
- pruebas de aceptación.

En esta actividad también se avanza en la definición de las pruebas de aceptación del sistema. Con la información disponible, es posible establecer los criterios de aceptación de las pruebas incluidas en dicho nivel, al poseer la información sobre los requisitos que debe cumplir el sistema, recogidos en el catálogo de requisitos.

4.3.7.1 Definición del alcance de las pruebas (ASI 10.1)

En función de la solución adoptada en el desarrollo de un sistema de información, es posible que determinados niveles de pruebas sean especialmente críticos y otros no sean necesarios. Por ejemplo, puede haber grandes diferencias en función de una solución de desarrollo completo o un producto de mercado cerrado integrado con otros sistemas.

4.3.7.1.1 Plan de pruebas

Para realizar las pruebas del sistema, los siguientes niveles serán considerados: pruebas de sistema, pruebas de implantación y pruebas de aceptación. Las pruebas unitarias se realizarán junto con la construcción pero no se documentarán ni los casos ni sus resultados.

Durante las pruebas de sistema, el foco estará en asegurar que la funcionalidad definida para el sistema está bien implementada. Se ejecutarán casos de prueba relacionados con las funcionalidades disponibles para los miembros de equipo como para los líderes de proyecto. Estas pruebas se realizarán una vez que se haya completado la construcción del sistema. Se considerará pasado el nivel de prueba cuando todos los casos definidos para el nivel hayan pasado exitosamente y no haya defectos importantes sin resolver. Como producto final, se generará un registro de la ejecución exitosa de todos los casos definidos y un registro de defectos relacionados.

Durante las pruebas de implantación, el foco estará en asegurar que el sistema está bien implantado en el entorno de trabajo. Para ello se ejecutará un subconjunto de los casos de prueba de sistema que asegure la correcta interacción entre el sistema y el entorno. Estas pruebas se realizarán una vez que se hayan completado las pruebas de sistema y el sistema esté implantado en el entorno apropiado. Se considerará pasado el nivel de prueba cuando todos los casos definidos para el nivel hayan pasado exitosamente y no haya defectos importantes sin resolver. Como producto final, se generará un registro de la ejecución exitosa de todos los casos seleccionados y un registro de defectos relacionados.

Durante las pruebas de aceptación, el foco estará en asegurar que los usuarios del sistema estén conformes con la funcionalidad implementada. Para ello se permitirá a los usuarios utilizar el sistema en forma libre. Estas pruebas se realizarán una vez que se hayan completado las pruebas de implementación. Se considerará pasado el nivel de prueba cuando los usuarios brinden su aceptación. Como producto final, se solicitará a los usuarios un registro de su aceptación.

4.3.7.2 Definición de requisitos del entorno de pruebas (ASI 10.2)

El objetivo de esta tarea es la definición o recopilación de los requisitos relativos al entorno de pruebas, completando el plan de pruebas.

La realización de las pruebas aconseja disponer de un entorno de pruebas separado del entorno de desarrollo y del entorno de operación, garantizando cierta independencia y estabilidad en los datos y elementos a probar, de modo que los resultados obtenidos sean objetivamente representativos, punto especialmente crítico en pruebas de rendimiento.

En esta tarea se inicia la definición de las especificaciones necesarias para la correcta ejecución de las distintas pruebas del sistema de información. Entre ellas podemos citar las siguientes:

- requisitos básicos de hardware y software base: sistemas operativos, gestores de bases de datos, monitores de teleproceso, etc.;
- requisitos de configuración de entorno: librerías, bases de datos, ficheros, procesos, comunicaciones, necesidades de almacenamiento, configuración de accesos, etc.;
- herramientas auxiliares. por ejemplo, de extracción de juegos de ensayo, análisis de rendimiento y calidad, etc.; y
- procedimientos para la realización de pruebas y migración de elementos entre entornos.

4.3.7.2.1 Plan de pruebas

El entorno de pruebas en el que se realizarán las pruebas de sistema deberá cumplir con los siguientes requisitos de hardware y software:

- Servidor:
 - o Sistema operativo: Microsoft Windows XP SP2
 - o Servidor web: Microsoft Internet Information Services 5.1
 - o Servidor de base de datos: Microsoft SQL Server 6.0
- Cliente:
 - o Sistema operativo: Microsoft Windows XP SP2
 - o Navegador: Microsoft Internet Explorer 6.0
 - o Herramientas de oficina: Microsoft Office Excel 2003 y Microsoft Office Project 2003

4.3.7.3 Definición de las pruebas de aceptación del sistema (ASI 10.3)

En esta tarea se realiza la especificación de las pruebas de aceptación del sistema, labor fundamental para que el usuario valide el sistema, como último paso, previo a la puesta en explotación.

Se debe insistir, principalmente, en los criterios de aceptación del sistema que sirven de base para asegurar que satisface los requisitos exigidos.

Los criterios de aceptación deben ser definidos de forma clara, prestando especial atención a aspectos como:

- procesos críticos del sistema;
- rendimiento del sistema;
- seguridad; y
- disponibilidad.

4.3.7.3.1 Plan de pruebas

Dado que los usuarios podrán utilizar el sistema en forma libre durante las pruebas de aceptación, se les solicitará que presten su aceptación sobre los siguientes puntos fundamentales:

- configuración de proyectos;
- importación de actividades;
- visualización de actividades; y
- rendimiento general del sistema.

4.3.8 Aprobación del análisis del sistema de información (ASI 11)

Luego de una reunión con líderes de proyecto involucrados en este desarrollo, se da la aprobación formal para avanzar con el desarrollo del sistema.

4.4 Diseño del sistema de información (DSI)

El objetivo del proceso de Diseño del Sistema de Información (DSI) es la definición de la arquitectura del sistema y del entorno tecnológico que le va a dar soporte, junto con la especificación detallada de los componentes del sistema de información.

A partir de dicha información, se generan todas las especificaciones de construcción relativas al propio sistema, así como la descripción técnica del plan de pruebas, la definición de los requisitos de implantación.

Las actividades de este proceso se agrupan en dos grandes bloques. En un primer bloque de actividades, que se llevan a cabo en paralelo, se obtiene el diseño de detalle del sistema de información. La realización de estas actividades exige una continua realimentación. En general, el orden real de ejecución de las mismas depende de las particularidades del sistema de información y, por lo tanto, de generación de sus productos.

En la actividad Definición de la Arquitectura del Sistema (DSI 1), se establece el particionamiento físico del sistema de información, así como su organización en subsistemas de diseño, la especificación del entorno tecnológico, y sus requisitos de operación, administración, seguridad y control de acceso. Se completan los catálogos de requisitos y normas, en función de la definición del entorno tecnológico, con aquellos aspectos relativos al diseño y construcción que sea necesario contemplar. Asimismo, se crea un catálogo de excepciones del sistema, en el que se registran las situaciones de funcionamiento secundario o anómalo que se estime oportuno considerar y, por lo tanto, diseñar y probar. Este catálogo de excepciones se utiliza como referencia en la especificación técnica de las pruebas del sistema.

El particionamiento físico del sistema de información permite organizar un diseño que contemple un sistema de información distribuido, como por ejemplo la arquitectura cliente/servidor, siendo aplicable a arquitecturas multinivel en general. Independientemente de la infraestructura tecnológica, dicho particionamiento representa los distintos niveles funcionales o físicos del sistema de información. La relación entre los elementos del diseño y particionamiento físico, y a su vez, entre el particionamiento físico y el entorno tecnológico, permite una especificación de la distribución de los elementos del sistema de información y, al mismo tiempo, un diseño orientado a la movilidad a otras plataformas o la reubicación de subsistemas.

El sistema de información se estructura en subsistemas de diseño. Éstos a su vez se clasifican como de soporte o específicos, al responder a propósitos diferentes:

- los subsistemas de soporte contienen los elementos o servicios comunes al sistema y a la instalación, y generalmente están originados por la interacción con la infraestructura técnica o la reutilización de otros sistemas, con un nivel de complejidad técnica mayor;
- los subsistemas específicos contienen los elementos propios del sistema de información, generalmente con una continuidad de los subsistemas definidos en el proceso de Análisis del Sistema de Información (ASI).

También se especifica en detalle el entorno tecnológico del sistema de información, junto con su planificación de capacidades (capacity planning), y sus requisitos de operación, administración, seguridad y control de acceso.

El diseño detallado del sistema de información comprende un conjunto de actividades que se llevan a cabo en paralelo a la Definición de la Arquitectura del Sistema (DSI 1). El alcance de cada una de estas actividades se resume a continuación:

- diseño de Casos de Uso Reales (DSI 3), con el diseño detallado del comportamiento del sistema de información para los casos de uso, el diseño de la interfaz de usuario y la validación de la división en subsistemas;
- diseño de Clases (DSI 4), con el diseño detallado de cada una de las clases que forman parte del sistema, sus atributos, operaciones, relaciones y métodos, y la estructura jerárquica del mismo.

Una vez que se tiene el modelo de clases, se comienza el diseño físico en la actividad Diseño Físico de Datos (DSI 6), común con el enfoque estructurado.

Una vez finalizado el diseño de detalle, se realiza su revisión y validación en la actividad Verificación y Aceptación de la Arquitectura del Sistema (DSI 7), con el objeto de analizar la consistencia entre los distintos modelos y conseguir la aceptación del diseño por parte de los responsables de las áreas de Explotación y Sistemas.

El segundo bloque de actividades complementa el diseño del sistema de información. En él se generan todas las especificaciones necesarias para la construcción del sistema de información:

- Generación de Especificaciones de Construcción (DSI 8), fijando las directrices para la construcción de los componentes del sistema, así como de las estructuras de datos.
- Especificación Técnica del Plan de Pruebas (DSI 10), que incluye la definición y revisión del plan de pruebas, y el diseño de las verificaciones de los niveles de prueba establecidos. El catálogo de excepciones permite, de una forma muy ágil, establecer un conjunto de verificaciones relacionadas con el propio diseño o con la arquitectura del sistema.
- Establecimiento de Requisitos de Implantación (DSI 11), que hace posible concretar las exigencias relacionados con la propia implantación del sistema, tales como formación de usuarios finales, infraestructura, etc.

Finalmente, en la actividad de Presentación y Aprobación del Diseño del Sistema de Información (DSI 12), se realiza una presentación formal y aprobación de los distintos productos del diseño.

En la Figura 17 se muestra la relación de actividades del proceso Diseño del Sistema de Información (DSI).

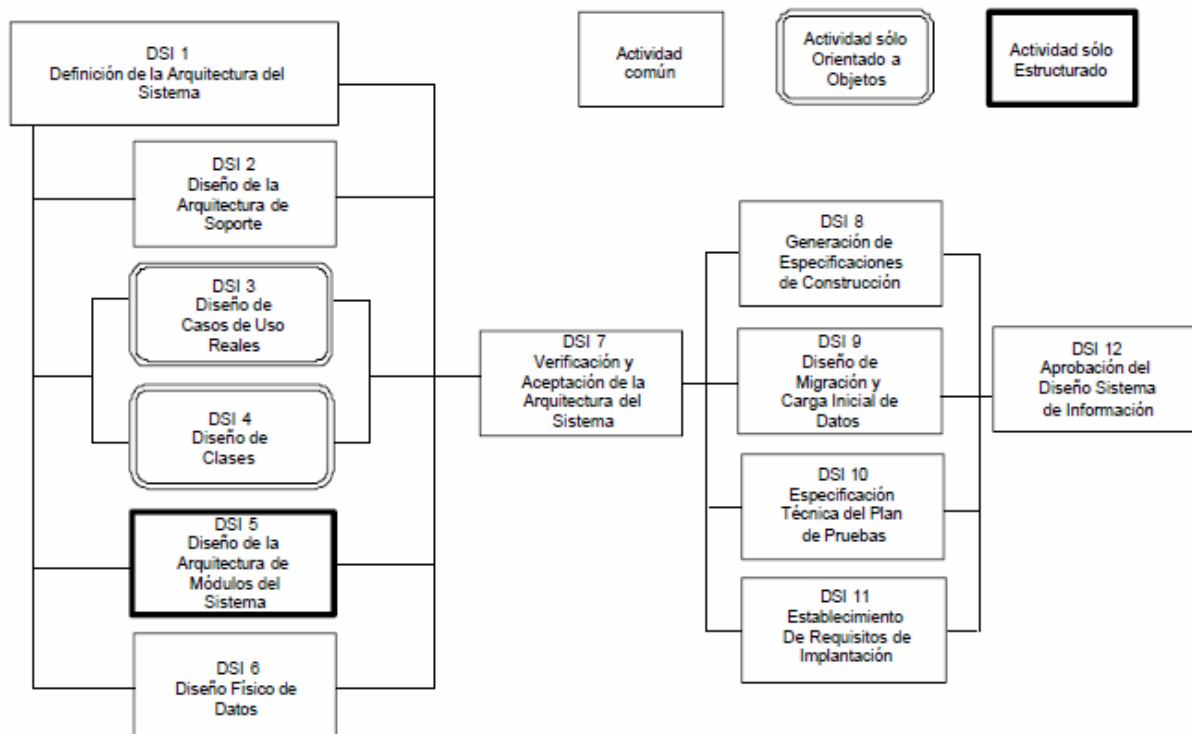


Figura 17: Las actividades que componen el proceso de Diseño del Sistema de Información (DSI) se separan principalmente en dos bloques: en el primero se trabaja sobre el diseño detallado del sistema, mientras que en el segundo se obtienen las especificaciones necesarias para la construcción.

4.4.1 Definición de la arquitectura del sistema (DSI 1)

En esta actividad se define la arquitectura general del sistema de información, especificando las distintas particiones físicas del mismo, la descomposición lógica en subsistemas de diseño y la ubicación de cada subsistema en cada partición, así como la especificación detallada de la infraestructura tecnológica necesaria para dar soporte al sistema de información.

El particionamiento físico del sistema de información se especifica identificando los nodos y las comunicaciones entre los mismos, con cierta independencia de la infraestructura tecnológica que da soporte a cada nodo.

Con el fin de organizar y facilitar el diseño, se realiza una división del sistema de información en subsistemas de diseño, como partes lógicas coherentes y con interfaces claramente definidas.

Se establece una distinción entre subsistemas específicos del sistema de información (en adelante, subsistemas específicos) y subsistemas de soporte, con la finalidad de independizar, en la medida de lo posible, las funcionalidades a cubrir por el sistema de información de la

infraestructura que le da soporte. En la mayoría de los casos, los subsistemas específicos provienen directamente de las especificaciones de análisis y de los subsistemas de análisis, mientras que los subsistemas de soporte provienen de la necesidad de interacción del sistema de información con la infraestructura y con el resto de los sistemas, así como de la reutilización de módulos o subsistemas ya existentes en la instalación.

Una vez identificados y definidos los distintos subsistemas de diseño, se determina su ubicación óptima de acuerdo a la arquitectura propuesta. La asignación de dichos subsistemas a cada nodo permite disponer, en función de la carga de proceso y comunicación existente entre los nodos, de la información necesaria para realizar una estimación de las necesidades de infraestructura tecnológica que da soporte al sistema de información. Este factor es especialmente crítico en arquitecturas multinivel o cliente/servidor, donde las comunicaciones son determinantes en el rendimiento final del sistema.

Se crea un catálogo de excepciones en el que se especifiquen las situaciones anómalas o secundarias en el funcionamiento y ejecución del sistema de información, y que se irá completando a medida que se avance en el diseño detallado de los subsistemas.

En esta actividad también se establecen los requisitos, normas y estándares originados como consecuencia de la adopción de una determinada solución de arquitectura o infraestructura, que serán aplicables tanto en este proceso como en la Construcción del Sistema de Información (CSI).

Se detallan a su vez, de acuerdo a las particularidades de la arquitectura del sistema propuesta, los requisitos de operación, seguridad y control, especificando los procedimientos necesarios para su cumplimiento.

Como resultado de esta actividad, se actualizan los catálogos de requisitos y normas, y se generan los siguientes productos:

- Diseño de la Arquitectura del Sistema, como producto que engloba el particionamiento físico del sistema de información y la descripción de subsistemas de diseño.
- Entorno Tecnológico del Sistema, que a su vez comprende la especificación del entorno tecnológico, las restricciones técnicas y la planificación de capacidades.

- Catálogo de Excepciones.
- Procedimientos de Operación y Administración del Sistema.
- Procedimientos de Seguridad y Control de Acceso.

4.4.1.1 Definición de niveles de arquitectura (DSI 1.1)

En esta tarea se describen los niveles de la arquitectura software, mediante la definición de las principales particiones físicas del sistema de información, representadas como nodos y comunicaciones entre nodos.

Se entiende por nodo cada partición física o parte significativa del sistema de información, con características propias de ejecución o función, e incluso de diseño y construcción.

Para facilitar la comprensión del sistema, se recomienda identificar como nodos los elementos de infraestructura más significativos de la arquitectura en la que se va a implementar el sistema de información. Los elementos que se aconseja especificar son los siguientes:

- gestores de datos;
- tipos de puesto cliente;
- servidores; y
- comunicaciones.

La comunicación se expresa por una conexión entre nodos, indicando su carácter bidireccional o unidireccional, con las principales características de los protocolos o tipo de mensajes utilizados.

La especificación de los niveles de la arquitectura se realiza con el detalle suficiente como para permitir un diseño dirigido hacia una solución concreta. En general, no es preciso indicar en cada nodo detalles relativos al hardware, capacidad, rendimiento o configuraciones de tolerancia a fallos, entre otros. Esta información se concreta en la tarea Especificación del Entorno Tecnológico (DSI 1.6).

Los criterios para diseñar la arquitectura se obtienen a partir de directrices tecnológicas o de integración, propias de la instalación, y del catálogo de requisitos del sistema de información.

Es necesario tener en cuenta, especialmente, aspectos relacionados con:

- usuarios: ubicación, movilidad, concurrencia, número, etc.;
- datos: variabilidad, volúmenes, necesidades de consolidación, seguridad, etc.; y
- procesos: distribución, reutilización, concurrencia, carácter crítico, etc.

4.4.1.1.1 Diseño de la arquitectura del sistema

Para el desarrollo del sistema se utilizará una arquitectura web cliente-servidor, es decir, una arquitectura de dos capas, donde la primera se corresponde con los clientes y la segunda con la aplicación y los servidores. Los clientes, en este contexto, se comunicarán con los servidores a través de la red de datos de la organización donde será implantado el sistema.

Son varias las razones que motivan la elección de esta arquitectura:

- la organización ya posee numerosas aplicaciones basadas en esta arquitectura, existiendo la posibilidad de implantar el sistema en servidores existentes y, a la vez, haciendo innecesaria la modificación o actualización de los clientes, ya que todas las estaciones de trabajo de los usuarios cuentan con navegadores para acceder a los servicios web;
- la necesidad de accesos concurrentes, en cualquier momento y desde cualquier estación de trabajo, se soluciona inmediatamente al residir los datos en los servidores centrales de la organización y hacerse disponibles a través de una interfaz web a cualquier usuario potencial del sistema; y
- el facilitar el mantenimiento y la resolución de incidentes de soporte, ya que la arquitectura tecnológica específica es la que utilizan otros sistemas en la organización y, al mantener la infraestructura e interfaces similares a lo existente, tanto usuarios como personal de soporte no verán un cambio significativo en su forma de trabajar o de resolver un problema puntual que pudieran enfrentar.

En la Figura 18 se muestra un diagrama de la arquitectura propuesta, donde se representan los tres principales componentes: los clientes, el servidor web y el servidor de base de datos, todos comunicados a través de la red de la organización.

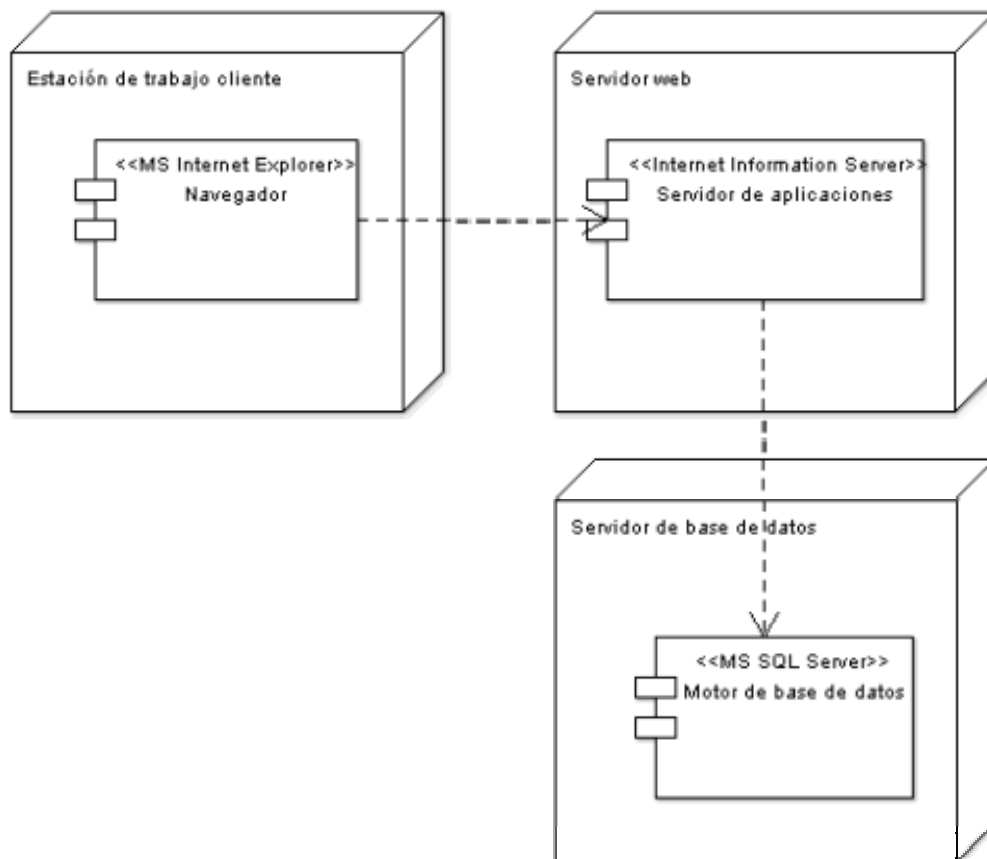


Figura 18: La arquitectura del sistema es de dos capas: en la primera está el cliente con el software de navegación y en la segunda, la aplicación junto con los servidores web y de base de datos

Los clientes son estaciones de trabajo PC estándar con sistema operativo Windows XP y navegadores MS Internet Explorer 6.0. Para el caso de los líderes de proyecto, también en dichas estaciones están instalados MS Office Excel 2003 y MS Office Project 2003. Como el sistema no requerirá una fuerte capacidad de procesamiento en los clientes, no se necesitarán modificaciones ni actualizaciones adicionales a las estaciones de trabajo existentes.

El servidor web es el equipo en el que estará implantado el sistema, el que procesará los requerimientos de los clientes y manejará la comunicación con la base de datos. Al ser un servidor existente en la organización, la arquitectura tecnológica del sistema se adaptará al mismo. Específicamente, el servidor de aplicaciones es Internet Information Services 5.1, y el sistema se desarrollará en Active Server Pages (ASP) 3.0 y VBScript 5.0 para asegurar la mayor compatibilidad posible.

El servidor de base de datos también es un equipo existente, el cual tiene instalado un motor de base de datos relacionales MS SQL Server 2000 (versión 8.0). Éste servirá como almacén

principal de los datos del sistema, tanto en lo referido a actividades como a la configuración misma del sistema.

Las comunicaciones se realizarán a través de la red IP de la organización, utilizando los protocolos estándar definidos para cada caso, por ejemplo, Hypertext Transfer Protocol (HTTP) entre los navegadores y el servidor web.

Todo lo anterior, es consistente con los requisitos, restricciones y normas identificadas en los procesos anteriores: Estudio de Viabilidad del Sistema (EVS) y Análisis del Sistema de Información (ASI).

4.4.1.2 Identificación de requisitos de diseño y construcción (DSI 1.2)

En esta tarea se realiza la especificación de los requisitos que están directamente relacionados con la adopción o diseño de una arquitectura o infraestructura concreta, y que pueden condicionar el diseño o la construcción del sistema de información.

Entre estos requisitos pueden estar los relacionados con lenguajes, rendimiento de los distintos elementos de la arquitectura, así como criterios de ubicación de módulos y datos en los distintos nodos.

Por tanto, como resultado de esta tarea se actualiza el catálogo de requisitos elaborado en el proceso Análisis de Sistemas de Información.

4.4.1.2.1 Catálogo de requisitos

Del análisis de la definición de la arquitectura, surgen requisitos adicionales a los ya identificados en el Catálogo de Requisitos de la actividad Establecimiento de requisitos (ASI 2). Dichos requisitos son representados en los siguientes atributos de calidad:

- [ACI.1] Los usuarios deberán poder interactuar con el sistema a través de estaciones de trabajo PC con sistema operativo Windows XP y las siguientes aplicaciones: MS Internet Explorer 6.0, MS Office Excel 2003 y MS Office Project 2003.
- [ACI.2] El sistema se deberá implantar en un servidor de web de aplicaciones Internet Information Services 5.1.

- [ACI.3] El sistema deberá utilizar los servicios de base de datos de un servidor MS SQL Server 2000 (versión 8.0).
- [ACI.4] Todos los nodos del sistema (clientes y servidores) deberán comunicarse a través de la red IP de la organización utilizando protocolos de comunicación estándar.

En adición a los puntos anteriores, los siguientes atributos de calidad deberán ser considerados durante el diseño y construcción del sistema:

- [ACC.1] La capa de presentación del sistema deberá cumplir con la especificación “HTML 4.01 Strict” de la W3C⁵.
- [ACC.2] Todo el código escrito en VBScript, tanto el correspondiente al manejo dinámico de la capa de presentación como a la lógica que resida en el servidor de aplicaciones, se desarrollará siguiendo las convenciones de codificación de VBScript publicadas por Microsoft⁶.

4.4.1.3 Especificación del entorno tecnológico (DSI 1.6)

En esta tarea se definen en detalle los distintos elementos de la infraestructura técnica que dan soporte al sistema de información, determinando la implementación concreta de los nodos y comunicaciones especificados en la tarea Definición de Niveles de Arquitectura (DSI 1.1).

Se propone agrupar los elementos de la infraestructura en los siguientes conceptos:

- hardware: procesadores, unidades de almacenamiento, estaciones de trabajo, etc.;
- software: sistemas operativos, subsistemas, middleware, gestores de bases de datos, sistemas de ficheros, software de base, herramientas y utilidades de gestión propias del sistema, etc.; y
- comunicaciones: diseño de la topología de la red, protocolos, nodos de red, etc.

La definición de los distintos elementos puede generar restricciones técnicas que afecten al diseño o construcción del sistema de información.

⁵ Para más información, referirse a las especificaciones de HTML 4.01 en [W3C, 1999].

⁶ Para más información, referirse a [Microsoft 1].

4.4.1.3.1 Entorno tecnológico del sistema

Las necesidades de infraestructura, tanto hardware, software como infraestructura de comunicaciones ya están disponibles en la organización donde se implantará el sistema. Dicha infraestructura está descrita en las secciones anteriores.

4.4.1.4 Especificación de requisitos de operación y seguridad (DSI 1.7)

El objetivo de esta tarea es definir los procedimientos de seguridad y operación necesarios para no comprometer el correcto funcionamiento del sistema y garantizar el cumplimiento de los niveles de servicios que exigirá el sistema en cuanto a la gestión de operaciones. Los niveles de servicio se especifican formalmente en el proceso Implantación y Aceptación del Sistema (IAS).

Tomando como referencia los requisitos establecidos para el sistema, y teniendo en cuenta la arquitectura propuesta y las características del entorno tecnológico definido en esta actividad, se lleva a cabo la definición de los requisitos de seguridad y control de acceso necesarios para garantizar la protección del sistema y minimizar el riesgo de pérdida, alteración o consulta indebida de la información. Para ello, se diseñan los procedimientos relacionados con:

- acceso al sistema y a sus recursos (datos, transacciones, librerías, etc.);
- mantenimiento de la integridad y confidencialidad de los datos;
- control y registro de accesos al sistema (logs, certificación, etc.);
- copias de seguridad y recuperación de datos y su periodicidad; y
- recuperación ante catástrofes.

4.4.1.4.1 Procedimientos de seguridad y control de acceso

El desarrollo del sistema deberá considerar que todos los accesos al mismo deberán identificar al usuario para poder brindarle solo la información que dicho usuario tiene permiso a acceder. Esto permitirá controlar la confidencialidad de los datos almacenados en el sistema. No se prevé la necesidad de mantener registros (logs) de acceso al sistema.

La integridad de los datos será mantenida a través del uso del motor de base de datos relacionales como único almacenamiento permanente de la información del sistema.

Asimismo, al utilizarse la infraestructura existente en la organización, el sistema quedará implantado bajo las normas organizacionales referidas a copias de seguridad, recuperación de datos y recuperación de catástrofes.

4.4.2 Diseño de la arquitectura de soporte (DSI 2)

En esta actividad se lleva a cabo la especificación de la arquitectura de soporte, que comprende el diseño de los subsistemas de soporte identificados en la actividad de Definición de la Arquitectura del Sistema (DSI 1), y la determinación de los mecanismos genéricos de diseño. Estos últimos sirven de guía en la utilización de diferentes estilos de diseño, tanto en el ámbito global del sistema de información, como en el diseño de detalle.

El diseño de los subsistemas de soporte, conceptualmente, es similar al diseño de los subsistemas específicos, aunque debe cumplir con unos objetivos claros de reutilización. De esta manera, se consigue simplificar y abstraer el diseño de los subsistemas específicos de la complejidad del entorno tecnológico, dotando al sistema de información de una mayor independencia de la infraestructura que le da soporte. Con este fin, se aconseja la consulta de los datos de otros proyectos existentes, disponible en el Histórico de Proyectos. Si esto no fuera suficiente, se puede contar en esta actividad con la participación de perfiles técnicos, con una visión global de la instalación.

Esta actividad se realiza en paralelo al diseño detallado, debido a que existe una constante realimentación, tanto en la especificación de los subsistemas con sus interfaces y dependencias, como en la aplicación de esqueletos o patrones en el diseño.

Los productos resultantes de esta actividad son:

- Diseño Detallado de los Subsistemas de Soporte; y
- Mecanismos Genéricos de Diseño y Construcción.

Para el desarrollo del presente sistema, solo se trabajará sobre los mecanismos genéricos de diseño y construcción.

4.4.2.1 Identificación de mecanismos genéricos de diseño (DSI 2.2)

El objetivo de esta tarea es identificar y diseñar, en el caso de no existir en la instalación, esqueletos, patrones de diseño o guías de diseño. Estos mecanismos genéricos se definen a

partir del estudio de comportamientos comunes relacionados, generalmente, con gestión de transacciones, persistencia de datos, control y recuperación de errores, utilización de recursos comunes, etc.

Los mecanismos genéricos de diseño son de aplicación tanto en la definición de la arquitectura del sistema como en el diseño de detalle de los subsistemas específicos y de soporte.

4.4.2.1.1 Mecanismos genéricos de diseño y construcción

Debido a que ASP y VBScript son los lenguajes de programación principales para el desarrollo de este sistema y considerando que ninguno de los dos es un lenguaje que brinda al programador todas las características de un lenguaje de programación orientado a objetos, se deberán tener ciertas consideraciones al momento de diseñar y construir el sistema basado en modelos de la metodología de desarrollo orientada a objetos. Las consideraciones más importantes son:

- las clases identificadas en los modelos de diseño que estén relacionadas con las interfaces o la lógica de la aplicación, se traducirán en archivos del sistema;
- las operaciones identificadas en dichas clases se corresponderán con subrutinas o funciones VBScript contenidas (o incluidas) en dichos archivos; y
- los atributos generales identificados en dichas clases se corresponderán con variables que tendrán un alcance global dentro del segmento de código VBScript de un archivo en particular.

Como guías generales de diseño y construcción del sistema, se tomarán en cuenta las siguientes consideraciones:

- [ACD.1] Las interfaces de usuario se construirán en archivos .asp siguiendo estrictamente el modelo general de página HTML que se adjunta en el anexo de la sección 8.1. Dicho modelo responde a la siguiente estructura:
 - o encabezado HTML;
 - bloque de definición de estilos (CSS), desarrollado en un archivo .css separado;

- o cuerpo HTML;
 - bloque de código a procesarse del lado del servidor (ASP), desarrollado en un archivo .asp separado;
 - bloque de presentación HTML; y
 - bloque de código a procesarse del lado del cliente (VBScript), desarrollado en un archivo .vbs separado.
- [ACD.7] Todos la información que maneje el sistema en memoria y que tengan estructura compleja deberán ser almacenados en estructuras XML.
 - o [ACD.7.1] En todos los casos se utilizará el parser de XML versión 1.0 que se provee a través de la interfaz “Microsoft.XMLDOM”.
- [ACD.2] Todo el envío de datos desde el servidor al cliente se realizará a través de estructuras XML.
 - o [ACD.2.1] Cuando éstas sean generadas como parte del flujo de procesamiento necesario para mostrar una página al usuario, la información será presentada al cliente por medio de islas de datos XML (“XML Data Islands”).
- [ACD.3] Todo el envío de datos desde el cliente al servidor se realizará a través de estructuras XML.
- [ACD.4] El envío y recuperación asíncrona de datos disparada desde el cliente al servidor se realizará a través de objetos XMLHttpRequest.
- [ACD.5] El código correspondiente a las tablas de actividades y fuentes de actividades se generará en forma dinámica a partir de los datos XML, transformados utilizando XSL/T.
- [ACD.6] Se desarrollarán funciones genéricas para realizar consultas a la base de datos y recuperar los resultados, debiendo ser estas el único punto de entrada utilizado en el sistema para dicha interacción.

- o [ACD.6.1] Todos los resultados recuperados se deberán proveer en formato XML.

4.4.3 Diseño de casos de uso reales (DSI 3)

Esta actividad tiene como propósito especificar el comportamiento del sistema de información para un caso de uso, mediante objetos o subsistemas de diseño que interactúan, y determinar las operaciones de las clases e interfaces de los distintos subsistemas de diseño.

Para ello, una vez identificadas las clases participantes dentro de un caso de uso, es necesario completar los escenarios que se recogen del análisis, incluyendo las clases de diseño que correspondan y teniendo en cuenta las restricciones del entorno tecnológico, esto es, detalles relacionados con la implementación del sistema. Es necesario analizar los comportamientos de excepción para dichos escenarios. Algunos de ellos pueden haber sido identificados en el proceso de análisis, aunque no se resuelven hasta este momento. Dichas excepciones se añadirán al catálogo de excepciones para facilitar las pruebas.

Algunos de los escenarios detallados requerirán una nueva interfaz de usuario. Por este motivo es necesario diseñar el formato de cada una de las pantallas o impresos identificados.

Es importante validar que los subsistemas definidos en la tarea Identificación de Subsistemas de Diseño (DSI 1.5) tienen la mínima interfaz con otros subsistemas. Por este motivo, se elaboran los escenarios al nivel de subsistemas y, de esta forma, se delimitan las interfaces necesarias para cada uno de ellos, teniendo en cuenta toda la funcionalidad del sistema que recogen los casos de uso. Además, durante esta actividad pueden surgir requisitos de implementación, que se recogen en el catálogo de requisitos.

Las tareas de esta actividad se realizan en paralelo con las de Diseño de Clases (DSI 4).

4.4.3.1 Identificación de clases asociadas a un caso de uso (DSI 3.1)

El objetivo de esta tarea es identificar las clases que intervienen en cada caso de uso, a partir del conjunto de clases definidas en la tarea Identificación de Clases Adicionales (DSI 4.1), ya que, como se ha señalado en la introducción de esta actividad, las actividades DSI 3 y DSI 4 se realizan en paralelo. Dichas clases se identifican a partir de las clases del modelo del análisis y de aquellas clases adicionales necesarias para el escenario que se está diseñando.

A su vez, a medida que se va estudiando la descripción de los casos de uso, pueden aparecer nuevas clases de diseño que no hayan sido identificadas anteriormente y que se incorporan al modelo de clases en la tarea Identificación de Clases Adicionales (DSI 4.1).

4.4.3.2 Diseño de la realización de los casos de uso (DSI 3.2)

El objetivo de esta tarea es definir cómo interactúan entre sí los objetos identificados en la tarea anterior para realizar, desde un punto de vista técnico, un caso de uso del sistema de información. Para ello, se parte de los escenarios especificados en el análisis, y se detallan teniendo en cuenta que se deben llevar cabo sobre un entorno tecnológico concreto y unos mecanismos genéricos de diseño.

Durante el desarrollo de esta tarea, es posible que surjan excepciones que se incluyen en el catálogo de excepciones, y que ahora quedan resueltas en los escenarios correspondientes.

Algunos de estos escenarios necesitan nueva interfaz de usuario. Por lo tanto, las clases de interfaz que se identifiquen se incorporan al modelo de clases de la tarea Identificación de Clases Adicionales (DSI 4.1), para realizar su diseño detallado.

También se realiza el estudio de los escenarios de los distintos casos de uso, para identificar comportamientos comunes sobre los que se aplican mecanismos genéricos de diseño identificados en la tarea de Identificación de Mecanismos Genéricos de Diseño (DSI 2.2), o se puede decidir diseñar un subsistema de soporte que contenga dicho comportamiento, como un servicio.

4.4.3.2.1 Diseño de la realización de los casos de uso

Para cada uno de los casos de uso identificados previamente, se analizó cuáles son las clases que intervienen en el desarrollo del mismo, que finalmente terminarán siendo implementadas como archivos separados.

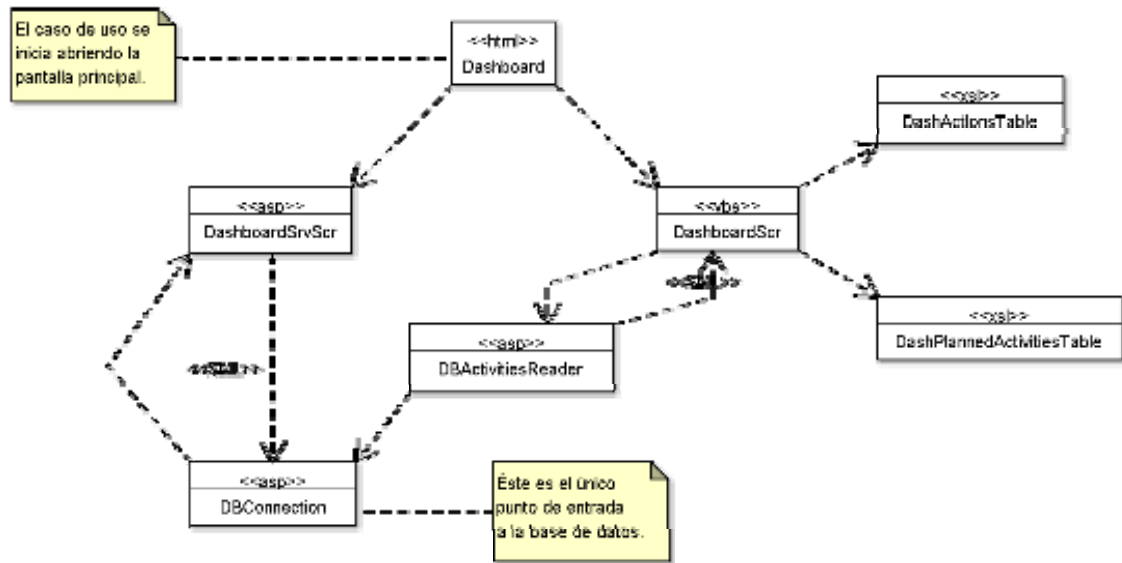


Figura 19: Diagrama que muestra las clases que intervienen en el CU.1, Consultar actividades, y la forma en la que interactúan.

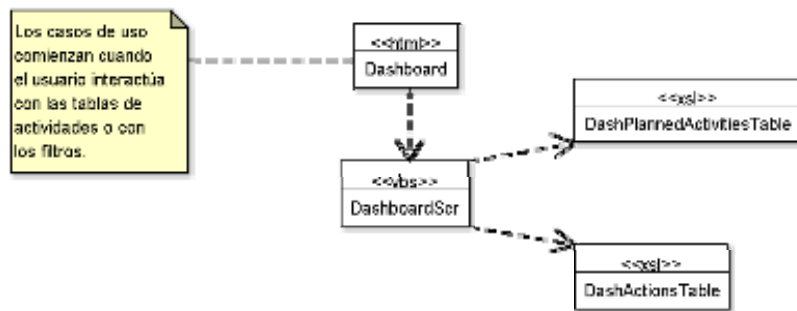


Figura 20: Diagrama que muestra las clases que intervienen en los CU.2, Filtrar actividades por proyecto, CU.3, Filtrar actividades por usuario, y CU.5, Reordenar lista de actividades, y la forma en la que interactúan.

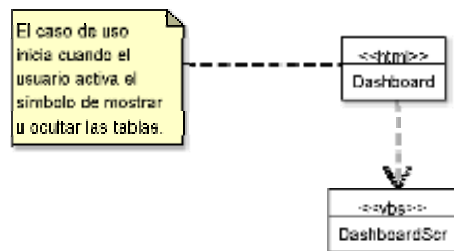


Figura 21: Diagrama que muestra las clases que intervienen en el CU.4, Ocultar y mostrar lista de actividades, y la forma en la que interactúan.

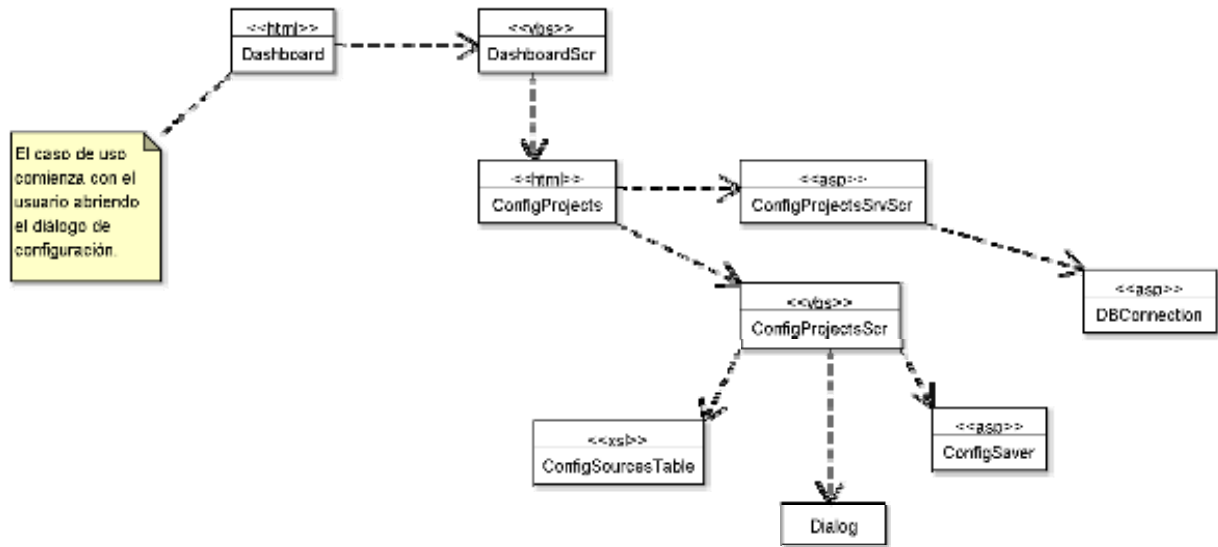


Figura 22: Diagrama que muestra las clases que intervienen en el CU.6, Administrar configuración de proyecto, y la forma en la que interactúan.

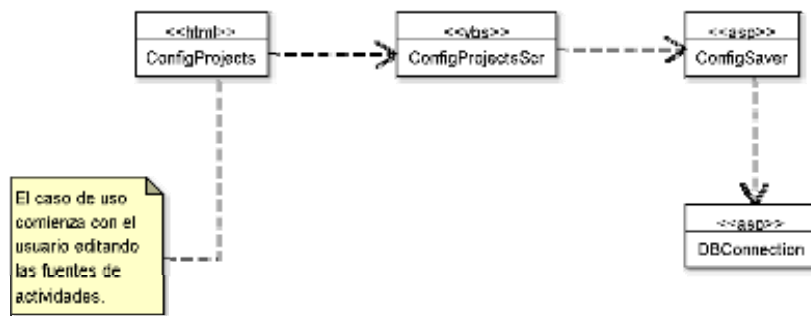


Figura 23: Diagrama que muestra las clases que intervienen en el CU.6.1, Administrar fuentes de actividades, y la forma en la que interactúan.

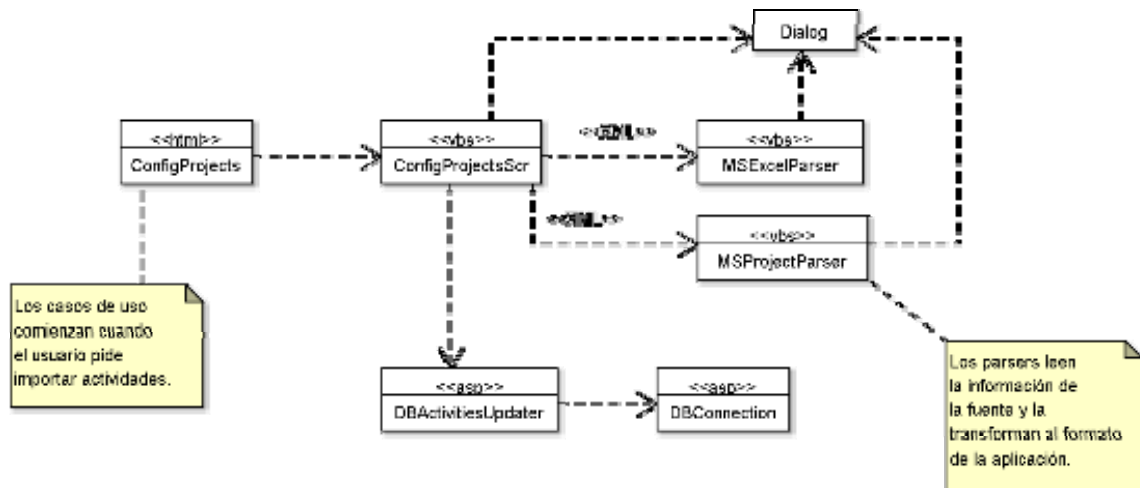


Figura 24: Diagrama que muestra las clases que intervienen en los CU.7, Importar actividades manualmente, CU7.1, Importar actividades desde MS Project, y CU7.2, Importar actividades desde MS Excel, y la forma en la que interactúan.

Finalmente, la lista de clases, archivos de código, queda constituida de la siguiente manera:

- Dashboard: es la pantalla principal de la aplicación escrita en HTML y contiene todo lo necesario para la visualización de las actividades y para la recepción de los eventos generados por los usuarios.
- DashboardScr: es la porción de código VBScript que responde a los eventos generados por los usuarios a través de la pantalla principal.
- DashboradSrvScr: es la porción de código ASP que valida la identidad del usuario y obtiene los datos necesarios para acotar qué actividades y proyectos puede éste ver.
- DBConnection: es el código ASP que interactúa con la base de datos, conectándose a la misma y actuando de capa de abstracción para el resto del código.
- DBActivitiesReader: es el código ASP que realiza la consulta a la base de datos, a través de la capa de abstracción, de acuerdo a los permisos del usuario.
- DashActionsTable y DashPlannedActivitiesTable: son dos estructuras XSL que servirán como base para la transformación de las actividades consultadas a la base de datos se conviertan en código HTML que las represente en forma de tablas al usuario.

- ConfigProjects: es la pantalla de configuración de la aplicación escrita en HTML y contiene todo lo necesario para la visualización de la configuración de los proyectos y para la recepción de los eventos generados por los usuarios.
- ConfigProjectsScr: es la porción de código VBScript que responde a los eventos generados por los usuarios a través de la pantalla de configuración.
- ConfigProjectsSrvScr: es la porción de código ASP que valida la identidad del usuario y obtiene los datos necesarios relativos a la configuración de los proyectos que el usuario tiene permisos de administrar.
- ConfigSourcesTable: es una estructura XSL que servirá como base para la transformación de las fuentes de actividades de un proyecto en código HTML que las represente en forma de tablas al usuario.
- ConfigSaver: es el código ASP que hará persistente a través de la base de datos, los cambios a la configuración de los proyectos que realice el usuario, incluyendo las modificaciones a las fuentes de datos de los mismos.
- Dialog: es la representación genérica de los cuadros de diálogo provistos por el sistema operativo para mostrar al usuario mensajes de información o de error.
- MSEXcelParser y MSProjectParser: son instanciaciones particulares de un mecanismo genérico de interpretación de fuentes externas y traducción de las actividades contenidas al formato necesario para su utilización en el sistema.
- DBActivitiesUpdater: es el código ASP que realiza la actualización de la base de datos, a través de la capa de abstracción, de las actividades importadas desde fuentes externas.

4.4.3.3 Revisión de la interfaz de usuario (DSI 3.3)

El objetivo de esta tarea es realizar el diseño detallado del comportamiento de la interfaz de usuario a partir de la especificación de la misma, obtenida en el proceso de análisis, y de acuerdo con el entorno tecnológico definido. Además, se incluyen las ventanas alternativas o elementos de diseño surgidos como consecuencia del diseño de los escenarios definidos en la tarea anterior.

Además, se revisa: la interfaz de usuario, la navegación entre ventanas, los elementos que forman cada interfaz, sus características (que deben ser consistentes con los atributos con los que están relacionadas), su disposición, y cómo se gestionan los eventos relacionados con los objetos.

En aquellos casos en los que se realizan modificaciones significativas sobre la interfaz de usuario, es conveniente que éste las valide, siendo opcional la realización de un nuevo prototipo.

4.4.3.3.1 Diseño de interfaz de usuario

Luego de un análisis de las interfaces definidas y al considerarse que el diseño de las mismas se encuentra fuertemente restringido por la tecnología a utilizar, se conviene que no aparecen nuevas clases de interfaz que deban ser tratadas en este punto. El análisis ha incluido una comparación entre las clases de interfaz antes definidas y las actuales clases de diseño y no se han encontrado diferencias significativas que justifiquen la modificación del actual modelo.

4.4.4 Diseño de clases (DSI 4)

El propósito de esta actividad es transformar el modelo de clases lógico, que proviene del análisis, en un modelo de clases de diseño. Dicho modelo recoge la especificación detallada de cada una de las clases, es decir, sus atributos, operaciones, métodos, y el diseño preciso de las relaciones establecidas entre ellas, bien sean de agregación, asociación o jerarquía. Para llevar a cabo todos estos puntos, se tienen en cuenta las decisiones tomadas sobre el entorno tecnológico y el entorno de desarrollo elegido para la implementación.

Se identifican las clases de diseño, que denominamos clases adicionales, en función del estudio de los escenarios de los casos de uso, que se está realizando en paralelo en la actividad Diseño de Casos de Uso Reales (DSI 3), y aplicando los mecanismos genéricos de diseño que se consideren convenientes por el tipo de especificaciones tecnológicas y de desarrollo. Entre ellas se encuentran clases abstractas, que integran características comunes con el objetivo de especializarlas en clases derivadas. Se diseñan las clases de interfaz de usuario, que provienen del análisis. Como consecuencia del estudio de los escenarios secundarios que se está realizando, pueden aparecer nuevas clases de interfaz.

También hay que considerar que, por el diseño de las asociaciones y agregaciones, pueden aparecer nuevas clases, o desaparecer incluyendo sus atributos y métodos en otras, si se considera conveniente por temas de optimización.

La jerarquía entre las clases se va estableciendo a lo largo de esta actividad, a medida que se van identificando comportamientos comunes en las clases, aunque haya una tarea propia de diseño de la jerarquía.

Otro de los objetivos del diseño de las clases es identificar para cada clase, los atributos, las operaciones que cubren las responsabilidades que se identificaron en el análisis, y la especificación de los métodos que implementan esas operaciones, analizando los escenarios del Diseño de Casos de Uso Reales (DSI 3). Se determina la visibilidad de los atributos y operaciones de cada clase, con respecto a las otras clases del modelo.

Una vez que se ha elaborado el modelo de clases, se define la estructura física de los datos correspondiente a ese modelo, en la actividad Diseño Físico de Datos (DSI 6).

Como resultado de todo lo anterior se actualiza el modelo de clases del análisis, una vez recogidas las decisiones de diseño.

4.4.4.1 Identificación de atributos de las clases (DSI 4.3)

El objetivo de esta tarea es identificar y describir, una vez que se ha especificado el entorno de desarrollo, los atributos de las clases.

Para identificar los atributos se revisa el modelo de clases obtenido en el proceso de Análisis del Sistema de Información (ASI 9.3), considerando que, a partir de uno de ellos, puede ser necesario definir atributos adicionales. Para cada atributo identificado se define su tipo, con formatos específicos, y si existieran, las restricciones asociadas a ese atributo.

Asimismo, se analiza la posibilidad de convertir un atributo en clase en aquellos casos en los que:

- el atributo se defina en varias clases de diseño; y
- la complejidad del atributo aumente la dificultad para comprender la clase a la que pertenece.

4.4.4.2 Identificación de operaciones de las clases (DSI 4.4)

El objetivo de esta tarea es definir, de forma detallada, las operaciones de cada clase de diseño. Para ello, se toma como punto de partida el modelo de clases generado en el análisis, así como el diseño de los casos de uso reales y los requisitos de diseño que pueden aparecer al definir el entorno de desarrollo.

Las operaciones de las clases de diseño surgen para dar respuesta a las responsabilidades de las clases de análisis y, además, para definir las interfaces que ofrece esa clase.

Según el entorno de desarrollo utilizado, se describe cada operación especificando: su nombre, parámetros y visibilidad (pública, privada, protegida). Si el entorno de desarrollo lo permite, se tiene en cuenta la posibilidad de simplificar el modelo de clases haciendo uso del polimorfismo y la sobrecarga de operaciones.

Para identificar las operaciones de aquellos objetos que presenten distintos estados, por lo que su comportamiento depende del estado en el que se encuentren, es recomendable realizar un diagrama de transición de estados, y traducir cada acción o actividad del mismo en una de estas operaciones.

4.4.4.3 Descripción de métodos de las operaciones (DSI 4.6)

En esta tarea se describen los métodos que se usan para detallar como se realiza cada una de las operaciones de una clase. Los métodos pueden especificarse mediante un algoritmo, usando pseudocódigo o lenguaje natural. Su implementación se basa en la secuencia de interacciones del diagrama de interacción en los que la clase aparezca o en la secuencia de transiciones del diagrama de transición de estados.

4.4.4.3.1 Modelo de clases de diseño

Para las operaciones más complejas de los casos de uso del sistema, se realizaron diagramas de actividad que modelan el comportamiento esperado. Los atributos y operaciones serán luego derivados de los mismos.

En el CU.1 se realizan principalmente dos operaciones: la inicialización de la pantalla principal y, dentro de esta, la obtención de todos los datos necesarios para hacerlo. En la Figura 25 se describe la secuencia de pasos necesarios para realizar una correcta inicialización

de la pantalla principal. En la Figura 26 se describe las particularidades que implican la obtención de los datos que serán mostrados en ésta misma pantalla.

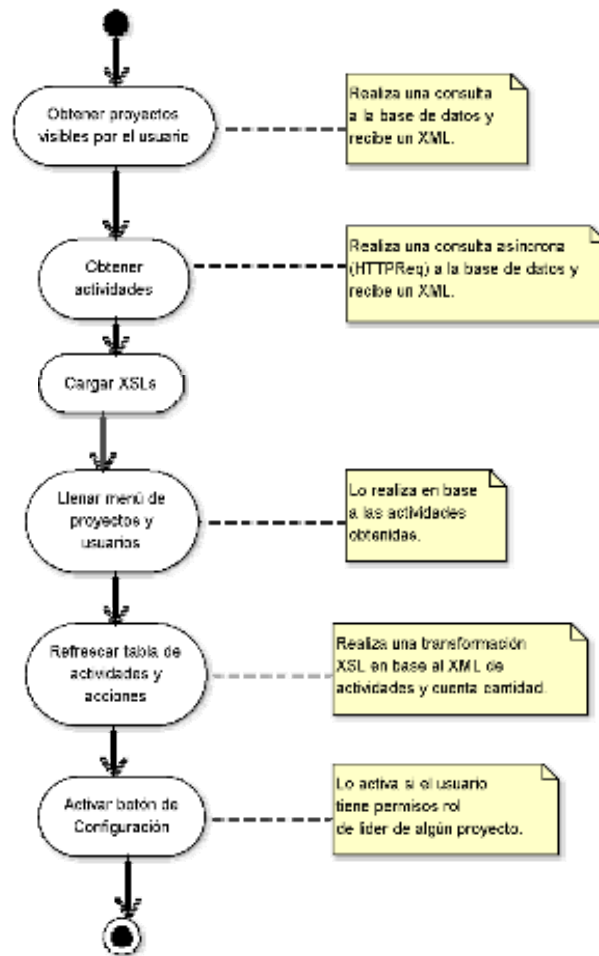


Figura 25: Secuencia de actividades necesarias para una correcta inicialización de la pantalla principal.

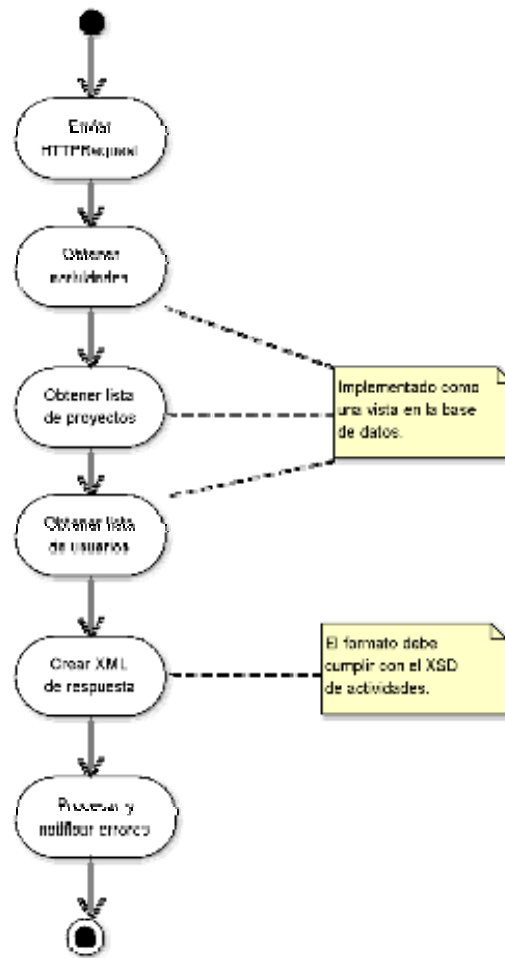


Figura 26: Particularidades de la obtención de los datos que se mostrarán en la pantalla principal.

En el caso del CU.6, también es necesario especificar claramente la secuencia de actividades necesarias para la correcta inicialización de la pantalla y para la grabación de la configuración de un proyecto. La secuencia de inicialización puede observarse en la Figura 27, mientras que la grabación está documentada en la Figura 28.

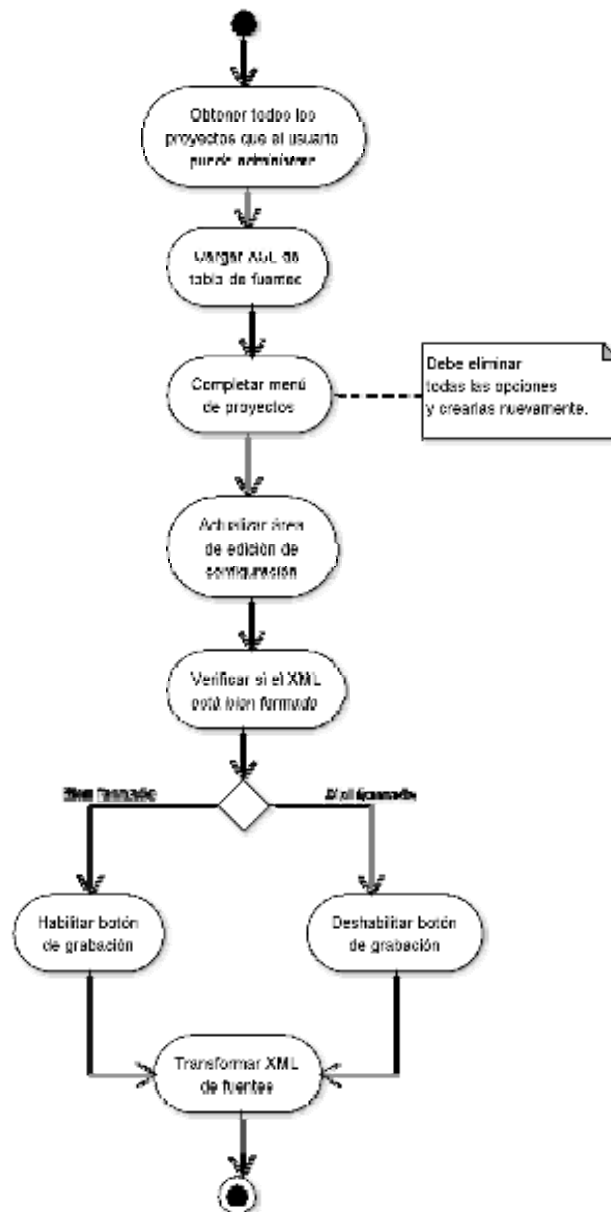


Figura 27: Secuencia de actividades que se deben realizar para la correcta inicialización de la pantalla de configuración de proyectos

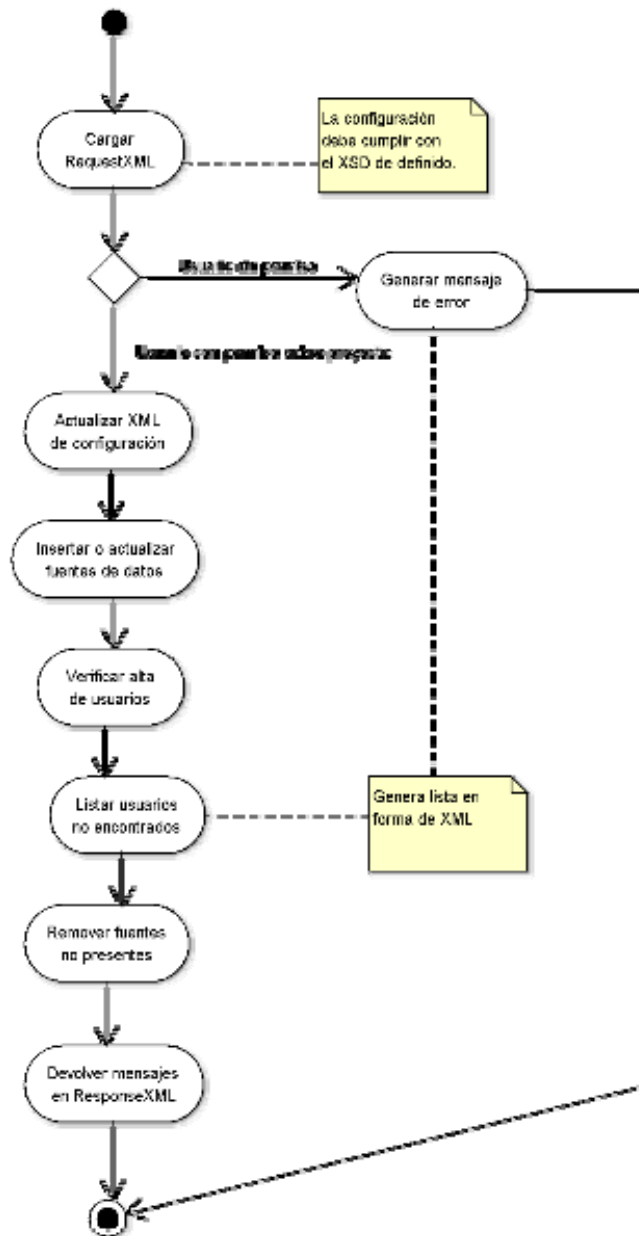


Figura 28: Secuencia de actividades correspondientes a la grabación en la base de datos de la configuración de un proyecto, incluyendo la actualización de sus fuentes de datos relacionadas.

El CU.7 es el que representa la importación de actividades al sistema y, por lo tanto, es necesaria la documentación de los procesos más importantes de dicho caso de uso: la importación de actividades desde una fuente externa y la grabación en la base de datos de esas actividades importadas. La secuencia de importación genérica para cualquier fuente se observa en la Figura 29, con detalles en la Figura 30 y Figura 31. La secuencia correspondiente a la grabación de las actividades importadas, se observa en la Figura 32.

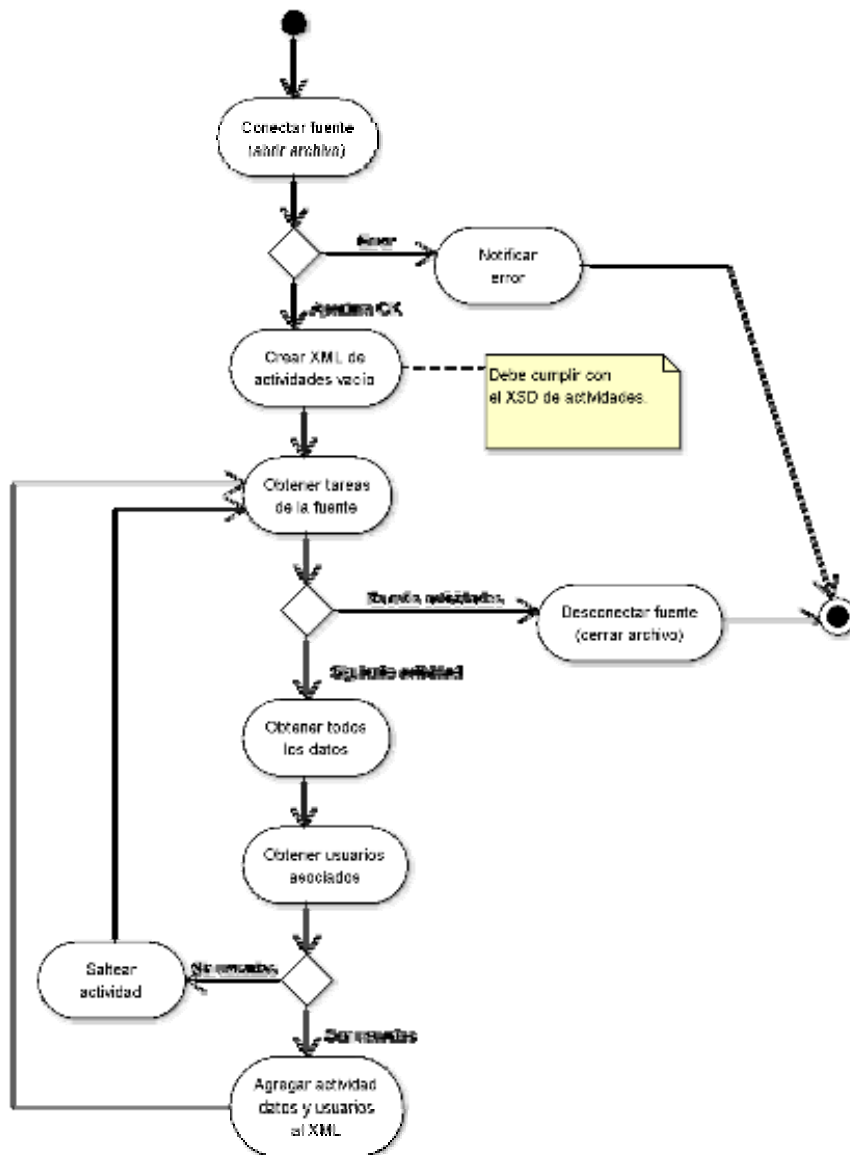


Figura 29: Secuencia de pasos requerida para importar actividades de una fuente de actividades externa. Esta figura se complementa con los detalles específicos para importar los datos de las actividades de la Figura 30 y los recursos asociados de la Figura 31.

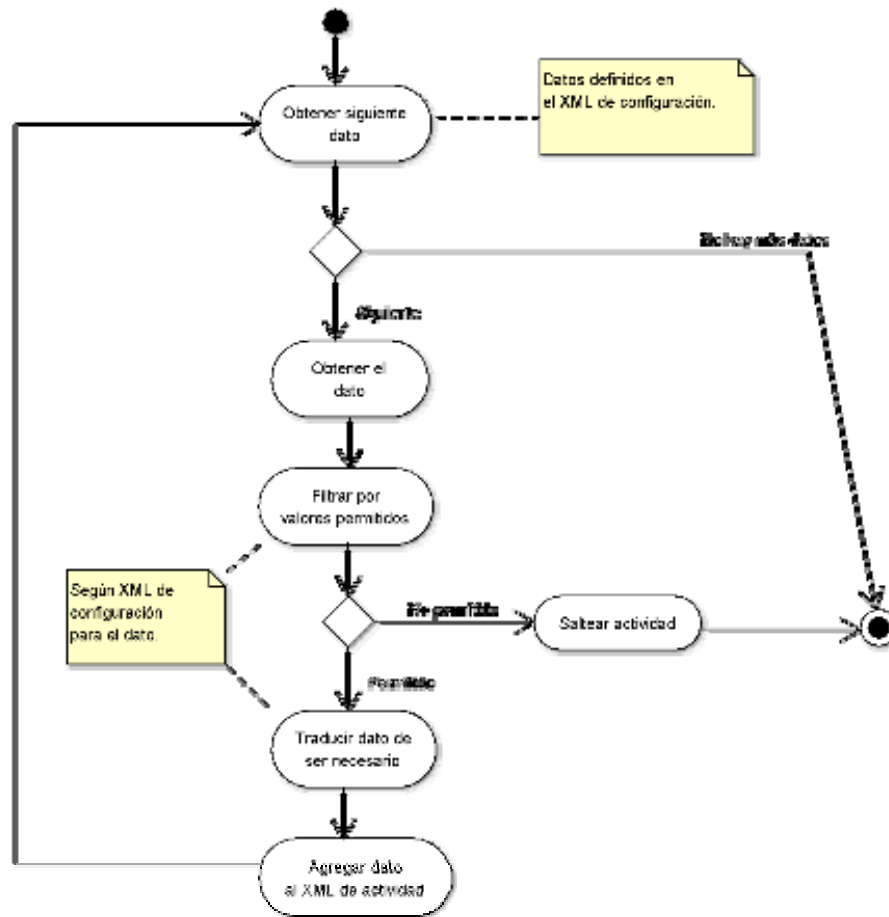


Figura 30: Los datos que forman parte de una determinada actividad deben ser importados de acuerdo a la secuencia de la figura, siempre de acuerdo a lo definido en la configuración del proyecto.

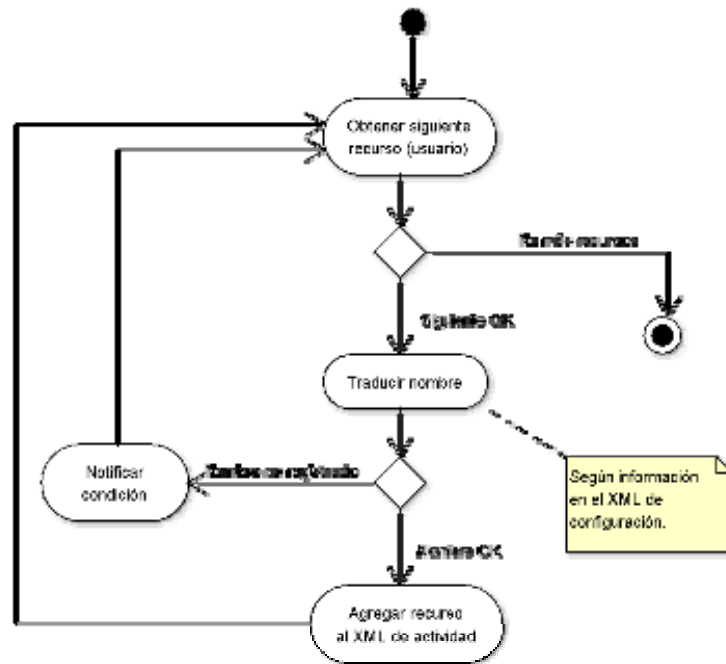


Figura 31: Para cada actividad importada, es necesario seguir esta secuencia de pasos para registrar adecuadamente los recursos relacionados con la misma.

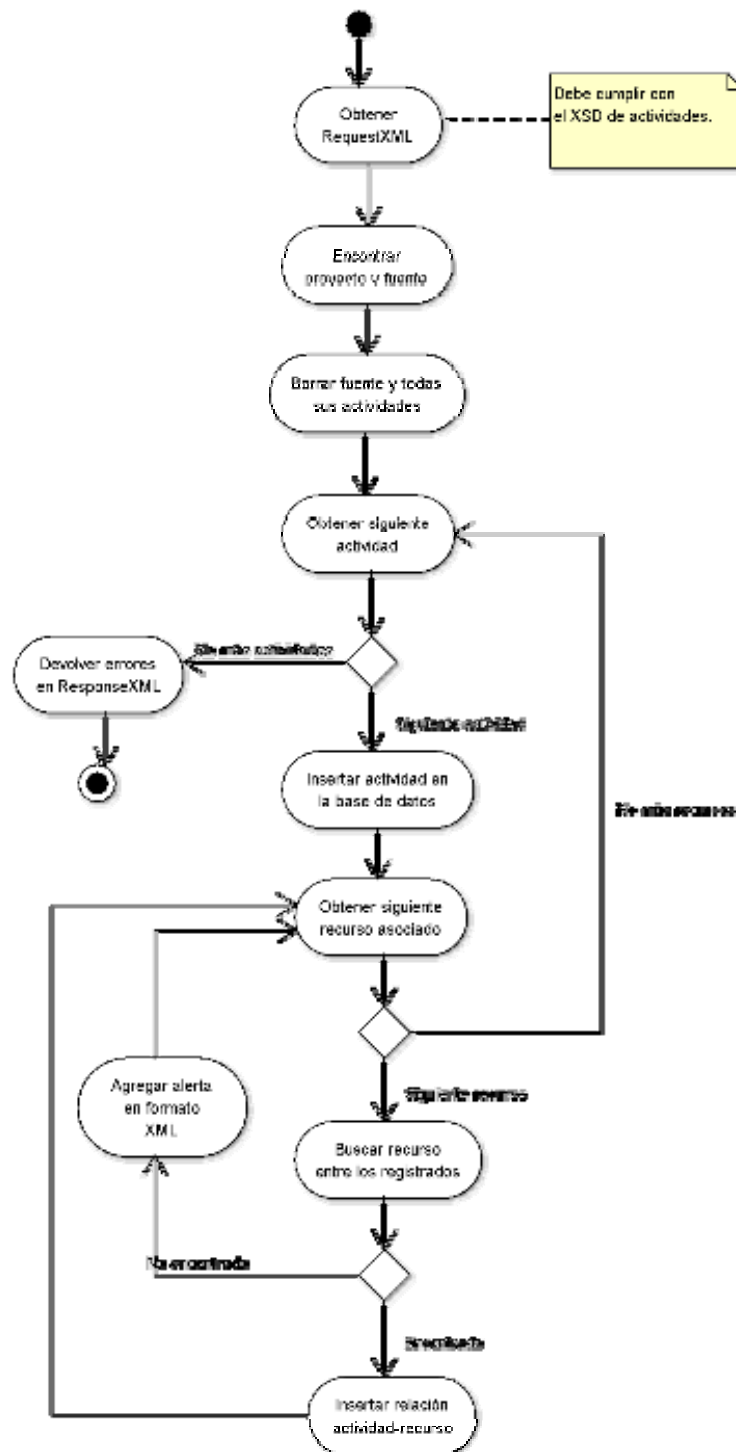


Figura 32: Secuencia de pasos que permiten al sistema grabar las actividades y recursos asociados que fueron importados desde una fuente de actividades externa.

Los CU.2, CU.3, CU.4 y CU.5 no fueron modelados con diagramas de actividad dado que sus operaciones son triviales. Por otro lado, los CU.7.1 y CU.7.2 tampoco tienen un diagrama específico por ser implementaciones específicas de lo definido para el CU.7, debiendo

respetar dicho comportamiento más allá de la fuente de datos particular que tengan como objetivo.

4.4.5 Diseño físico de datos (DSI 6)

En esta actividad se define la estructura física de datos que utilizará el sistema, a partir del modelo lógico de datos normalizado o modelo de clases, de manera que teniendo presentes las características específicas del sistema de gestión de datos concreto a utilizar, los requisitos establecidos para el sistema de información, y las particularidades del entorno tecnológico, se consiga una mayor eficiencia en el tratamiento de los datos.

Las tareas de esta actividad se realizan de forma iterativa y en paralelo con las realizadas en las actividades Definición de la Arquitectura del Sistema (DSI 1), dónde se especifican los detalles de arquitectura e infraestructura y la planificación de capacidades, Diseño de la Arquitectura de Soporte (DSI 2), dónde se determinan y diseñan los servicios comunes que pueden estar relacionados con la gestión de datos (acceso a bases de datos, ficheros, áreas temporales, sincronización de bases de datos, etc.), Diseño de Casos de Uso Reales y de Clases (DSI 3 y 4), para desarrollo orientado a objetos, y Diseño de la Arquitectura de Módulos del Sistema (DSI 5), para desarrollo estructurado, dónde se especifica la lógica de tratamiento y las interfaces utilizadas.

La obtención del modelo físico de datos se realiza aplicando una serie de reglas de transformación a cada elemento del modelo de clases que se está generando en la actividad Diseño de Clases (DSI 4).

Asimismo, en esta actividad hay que considerar los estándares y normas establecidos para el diseño aplicando, cuando proceda, los mecanismos genéricos de diseño identificados en la tarea Identificación de Mecanismos Genéricos de Diseño (DSI 2.2).

4.4.5.1 Diseño del modelo físico de datos (DSI 6.1)

El objetivo de esta tarea es realizar el diseño del modelo físico de datos a partir del modelo de clases.

Como paso previo al diseño de la estructura física de datos, se analizan las peculiaridades técnicas del gestor de bases de datos a utilizar, y las estimaciones sobre la utilización y volumen de las ocurrencias de cada clase del modelo de clases.

De acuerdo al análisis anterior, se determina cómo se van a convertir las clases en tablas, considerando las relaciones existentes entre ellas y los identificadores, definiendo sus claves primarias, ajenas, alternativas u otros medios de acceso en general.

También se definen aquellos elementos que, en función del gestor a utilizar, se considere necesario implementar.

4.4.5.1.1 Modelo físico de datos

Como primer paso en la definición del modelo físico de datos, se trabajó en las tablas relacionadas con el almacenamiento de las actividades. El modelo final propuesto se muestra en la Figura 33.

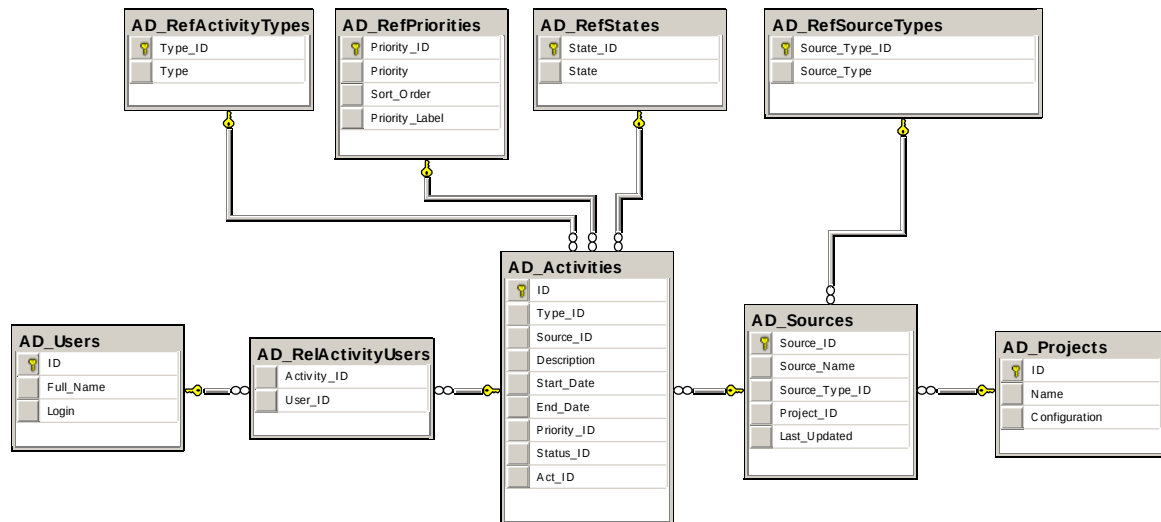


Figura 33: Modelo físico de las tablas relacionadas con el almacenamiento de las actividades.

Las tablas que se relacionan con el almacenamiento de las actividades son las siguientes:

- **Projects**: Esta tabla se corresponde con la clase que identifica a los proyectos. Sus campos son los siguientes:

Campo	Tipo	Descripción
ID	int	Identificador del proyecto en el sistema. Se utilizará como clave primaria de la tabla. Su valor es calculado por el motor de base de datos.
Name	nvarchar(50)	Texto para almacenar el nombre del proyecto.
Configuration	nvarchar(max)	Texto para almacenar la configuración del proyecto. Esta configuración será almacenada como un texto XML y contendrá la definición de las fuentes y de los usuarios del proyecto.

Tabla 16: Campos correspondientes a la tabla Projects.

- Sources: Esta tabla se corresponde con la clase que identifica a las fuentes de actividades. Sus campos son los siguientes:

Campo	Tipo	Descripción
ID	int	Identificador de la fuente de actividades en el sistema. Se utilizará como clave primaria de la tabla. Su valor es calculado por el motor de base de datos.
Project_ID	int	Identificador del proyecto al que corresponde la fuente de actividades. Hace referencia a la tabla Projects.
Source_Name	nvarchar(50)	Texto representativo del nombre de la fuente de actividades.
Source_Type_ID	int	Identificador que distingue el tipo de fuente de actividades. Se utiliza para seleccionar el parser apropiado e importar exitosamente el tipo específico de actividades almacenadas. Hace referencia a la tabla RefSourceTypes.
Last_Updated	datetime	Fecha y hora de la última importación de actividades de una fuente específica.

Tabla 17: Campos correspondientes a la tabla Sources.

- RefSourceTypes: Esta tabla almacena los diferentes tipos de fuentes de actividades con los que el sistema puede trabajar. Su creación está originada en la normalización del modelo de datos. Sus campos son los siguientes:

Campo	Tipo	Descripción
Source_Type_ID	int	Identificador del tipo de fuente de actividades. Se utilizará como clave primaria de la tabla. Su valor es calculado por el motor de base de datos.
Source_Type	nvarchar(20)	Texto representativo del tipo de fuente de actividades. Es un acrónimo: - “MSProject” para cronogramas creados con Microsoft Office Project. - “MSExcel” para fuentes basadas en archivos creados con Microsoft Office Excel.

Tabla 18: Campos correspondientes a la tabla RefSourceTypes.

- Activities: Esta tabla se corresponde con la clase que representa a las actividades, identificada previamente. En esta tabla se almacenarán tanto las actividades planificadas como las acciones no planificadas. Sus campos son los siguientes:

Campo	Tipo	Descripción
ID	int	Identificador de la actividad en el sistema. Se utilizará como clave primaria de la tabla. Su valor es calculado por el motor de base de datos.
Source_ID	int	Identificador de la fuente de actividades. Hace referencia a la tabla Sources.
Type_ID	int	Identificador que distingue actividades planificadas de acciones no planificadas. Su objetivo es separar las actividades almacenadas en el sistema en las dos tablas que serán mostradas al usuario. Hace referencia a la tabla RefTypes.
Act_ID	nvarchar(10)	Texto para almacenar el identificador que tiene la tarea en la fuente de datos correspondiente, por ejemplo: código de actividad en un cronograma o código de acción en un repositorio de acciones.
Description	nvarchar(250)	Texto para almacenar la descripción de la actividad, tal como aparece en la fuente correspondiente, por ejemplo: nombre de la actividad en el cronograma o descripción de la acción no planificada.
Start_Date	datetime	Fecha de comienzo de la actividad planificada o fecha de creación de la acción no planificada.
End_Date	datetime	Fecha de fin de la actividad planificada o fecha comprometida de concreción de la acción no planificada.
Priority_ID	int	Identificador de la prioridad de la actividad. Se usa principalmente para acciones no planificadas. Hace referencia a la tabla RefPriorities.
Status_ID	int	Identificador del estado de la actividad. Hace referencia a la tabla RefStates.

Tabla 19: Campos correspondientes a la tabla Activities.

- **RefActivityTypes:** Esta tabla almacena los diferentes tipos de actividades que el sistema reconoce. Principalmente se trata de actividades planificadas y acciones no planificadas. Su creación está originada en la normalización del modelo de datos. Sus campos son los siguientes:

Campo	Tipo	Descripción
Type_ID	int	Identificador del tipo de actividades. Se utilizará como clave primaria de la tabla. Su valor es calculado por el motor de base de datos.
Type	nvarchar(20)	Texto representativo del tipo de actividades. Es un acrónimo: - “PLAN” para actividades planificadas. - “PREV” para acciones preventivas (no planificadas). - “CORR” para acciones correctivas (no planificadas).

Tabla 20: Campos correspondientes a la tabla RefActivityTypes.

- **RefPriorities:** Esta tabla almacena las diferentes prioridades aplicadas a las actividades almacenadas. Su creación está originada en la normalización del modelo de datos. Sus campos son los siguientes:

Campo	Tipo	Descripción
Priority_ID	int	Identificador de la prioridad de una actividad. Se utilizará como clave primaria de la tabla. Su valor es calculado por el motor de base de datos.
Priority	nvarchar(20)	Texto corto representativo de la prioridad, utilizado en el código del sistema y en los archivos de configuración.
Priority_Label	nvarchar(20)	Texto completo representativo de la prioridad, utilizado en la visualización de la prioridad en las interfaces de usuario.
Sort_Order	int	Número que permite ordenar las prioridades. Su existencia es necesaria porque un ordenamiento alfabético por los campos Priority o Priority_Label no asegura que el resultado sea semánticamente correcto.

Tabla 21: Campos correspondientes a la tabla RefPriorities.

- **RefStates:** Esta tabla almacena los diferentes estados en los que pueden estar las actividades del sistema. Principalmente identifica actividades planificadas o acciones no planificadas que han sido iniciadas o que no. Las actividades terminadas no se almacenan en el sistema. Su creación está originada en la normalización del modelo de datos. Sus campos son los siguientes:

Campo	Tipo	Descripción
State_ID	int	Identificador del estado de las actividades. Se utilizará como clave primaria de la tabla. Su valor es calculado por el motor de base de datos.
State	nvarchar(20)	Texto representativo del estado de las actividades que será mostrado al usuario: - “No iniciada” para actividades planificadas que comienzan en el futuro o acciones no planificadas que no han sido iniciadas por sus responsables. - “En progreso” para actividades planificadas que han comenzado o acciones no planificadas que han sido iniciadas por sus responsables.

Tabla 22: Campos correspondientes a la tabla RefStates.

- **Users:** Esta tabla se corresponde con la clase que representa a los usuarios del sistema, identificada previamente. En esta tabla se almacenarán tanto las actividades planificadas como las acciones no planificadas. Sus campos son los siguientes:

Campo	Tipo	Descripción
ID	int	Identificador único de un usuario del sistema. Se utilizará como clave primaria de la tabla. Su valor es calculado por el motor de base de datos.
Full_Name	nvarchar(50)	Texto completo del nombre y apellido del usuario, como se lo reconocerá en el sistema.
Login	nvarchar(50)	Texto correspondiente al nombre de usuario con el que el usuario se registra en el sistema.

Tabla 23: Campos correspondientes a la tabla Users.

- RelActivityUsers: Esta tabla almacena las relaciones entre actividades y usuarios del sistema. Debido a que un usuario puede tener muchas actividades asignadas y a que una actividad puede tener más de un usuario asignado, es necesaria su existencia para la correcta normalización del modelo de datos. Sus campos son los siguientes:

Campo	Tipo	Descripción
Activity_ID	int	Identificador de la actividad. Hace referencia a la tabla Activities.
User_ID	int	Identificador del usuario relacionado con la actividad. Hace referencia a la tabla Users.

Tabla 24: Campos correspondientes a la tabla RelActivityUsers.

A continuación se trabajó en las tablas relacionadas con la configuración del sistema. El modelo final propuesto se muestra en la Figura 34.

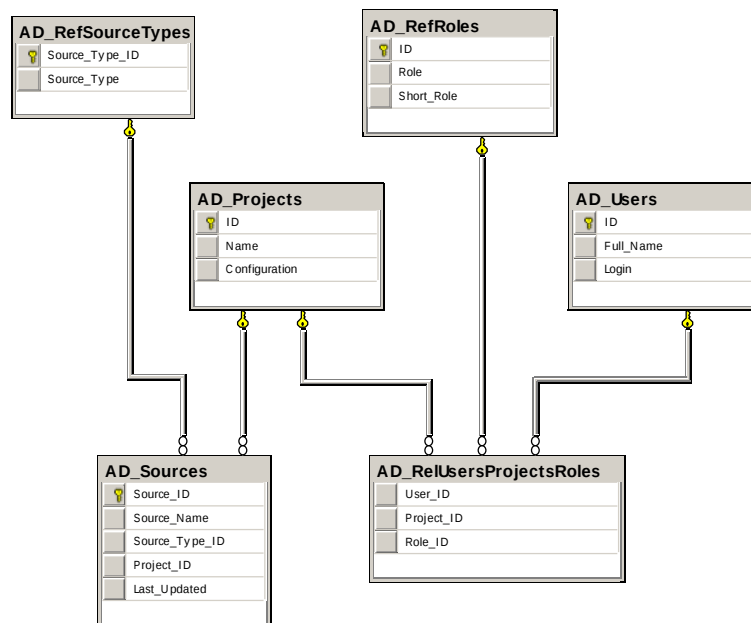


Figura 34: Modelo físico de las tablas relacionadas con la configuración del sistema.

Las tablas que se relacionan con la configuración del sistema y que no han sido descritas previamente son las siguientes:

- **RefRoles:** Esta tabla almacena los diferentes tipos de usuarios manejados por el sistema. Principalmente identifica a los líderes de proyecto y a los miembros de equipo. Su creación está originada en la normalización del modelo de datos. Sus campos son los siguientes:

Campo	Tipo	Descripción
ID	int	Identificador del rol del usuario. Se utilizará como clave primaria de la tabla. Su valor es calculado por el motor de base de datos.
Role	nvarchar(20)	Texto representativo del rol del usuario: - “Líder” se utilizará para identificar a los líderes de proyecto. - “Usuario” se utilizará para identificar a todos los miembros de equipo que no son líderes del mismo.
Short_Role	nvarchar(3)	Texto corto que identificará al rol y se utilizará en el código del sistema y en la configuración de los proyectos: - “PM”: representa a los líderes de equipo. - “USR”: representa a los miembros de equipo.

Tabla 25: Campos correspondientes a la tabla RefRoles.

- **RelUsersProjectsRoles:** Esta tabla almacena las relaciones entre los proyectos y los usuarios del sistema. Debido a que un usuario puede estar relacionado con muchos proyectos y a un proyecto puede tener más de un usuario asignado, es necesaria su existencia para la correcta normalización del modelo de datos. Adicionalmente, como atributo de esta relación, se define el rol que el usuario tiene en el proyecto relacionado. Sus campos son los siguientes:

Campo	Tipo	Descripción
Project_ID	int	Identificador del proyecto. Hace referencia a la tabla Projects.
User_ID	int	Identificador del asignado al proyecto. Hace referencia a la tabla Users.
Role_ID	int	Identificador del rol que el usuario tiene en el proyecto. Hace referencia a la tabla Refroles.

Tabla 26: Campos correspondientes a la tabla RelActivityUsers.

Todas las tablas que componen el modelo de datos del sistema, junto con sus relaciones, se encuentran representadas en la Figura 35.

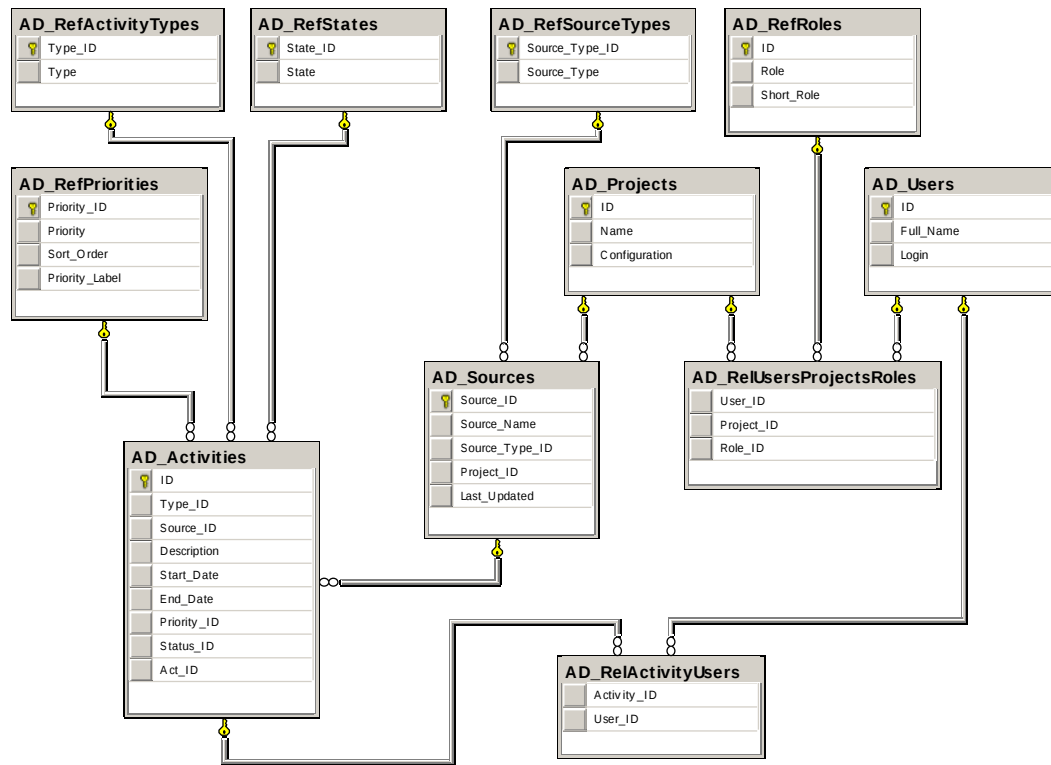


Figura 35: Modelo de datos completo del sistema.

En relación con el acceso a los datos almacenados en el motor de base de datos, se identificaron varias operaciones que se deberán realizar a través de consultas almacenadas (vistas) en dicho motor para simplificar la programación del sistema y mejorar el rendimiento del mismo. Estas operaciones son:

- **AllActivities**: Esta consulta proveerá toda la información necesaria para la correcta visualización de las actividades almacenadas (CU.1, CU.2, CU.3 y CU.5). Como el modelo de datos se encuentra normalizado, es necesario vincular nuevamente cada parámetro de las actividades con su respectivo texto para ser procesado (tipo de actividad, prioridad, estado). Asimismo, se requiere vincular a las actividades con los usuarios asignados y a las mismas con sus respectivos proyectos a través de las fuentes de datos. Por último, a través de la vinculación con la vista *GetUsersByUser*, se obtiene la lista de actividades que cada usuario puede ver. El diagrama de la consulta puede verse en la Figura 36.

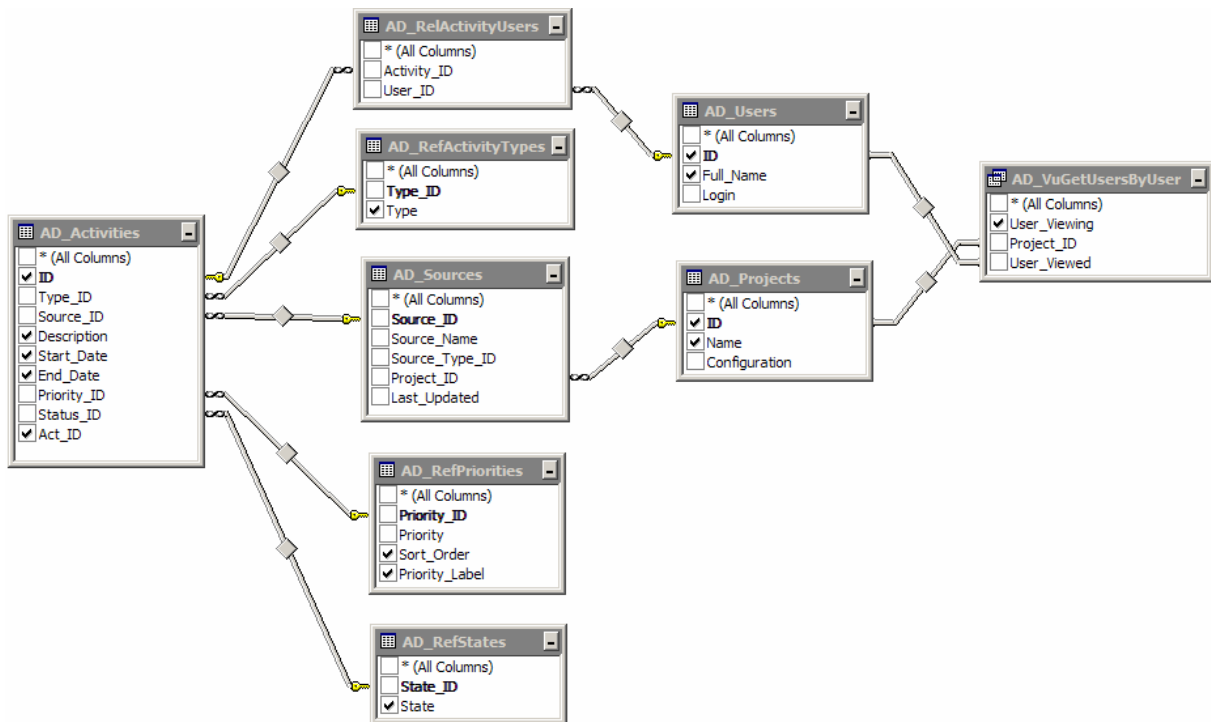


Figura 36: Diagrama de la vista AllActivities. Es utilizada para simplificar la construcción de los CU.1, CU.2, CU.3 y CU.5.

- **GetUsersByUser:** Esta consulta obtiene, a partir de un usuario determinado, todos los proyectos con los que está vinculado y, de estos, todos los usuarios de dichos proyectos. Se utiliza como auxiliar para la vista AllActivities, de forma tal de obtener eficientemente todas las actividades que un determinado usuario puede ver. El diagrama de la consulta puede verse en la Figura 37.

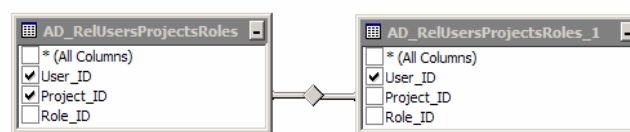


Figura 37: Diagrama de la vista GetUsersByUser. Es utilizada como ayuda para la programación de la vista AllActivities.

- **ProjectsByPM:** Esta consulta parte de los usuarios, filtrándolos por los que tienen roles de líder de proyecto, y los vincula a los proyectos de los cuales son líderes. Asimismo, obtiene la información adicional del proyecto que es necesaria para administrar la configuración de los proyectos (CU.6). El diagrama de la consulta puede verse en la Figura 38.

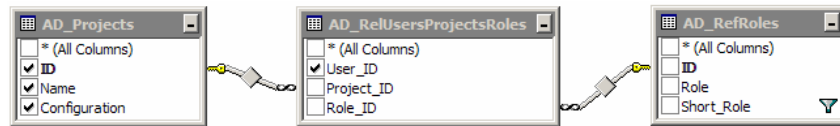


Figura 38: Diagrama de la vista ProjectsByPM. Es utilizada para obtener todos los proyectos que un usuario puede administrar, junto con la información necesaria para realizar dicha actividad.

- ProjectSources: Esta consulta obtiene, para cada proyecto, todas las fuentes de actividades relacionadas con el mismo. También obtiene información adicional de los proyectos y de las fuentes para facilitar la correcta importación de actividades (CU.7, CU.7.1 y CU.7.2). El diagrama de la consulta puede verse en la Figura 39.

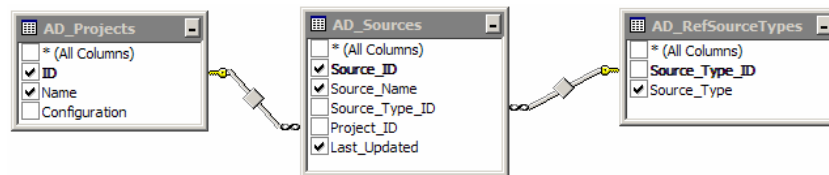


Figura 39: Diagrama de la vista ProjectSources. Es utilizada para obtener todas las fuentes de actividades de los proyectos, con el objeto de importar correctamente las actividades desde dichas fuentes al sistema.

Como último paso relacionado con el diseño relativo a la base de datos, será necesario que durante la construcción del sistema se preparen dos scripts que permitan realizar las siguientes funciones:

- Crear todas las tablas y vistas, junto con todas sus propiedades y relaciones.
- Inicializar las tablas de referencia con los valores necesarios para el correcto funcionamiento del sistema.

Finalmente, tras analizar los comportamientos identificados en los diferentes casos de uso, las siguientes consideraciones deben ser tenidas en cuenta con respecto al modelado de los datos que son transmitidos entre los diferentes componentes del sistema:

- Actividades: la información de actividades debe seguir un modelo específico al momento de ser importadas por el componente de importación y transferida al componente de almacenamiento. Como esta comunicación se realiza a través de estructuras XML, se documenta en un XSD.

- **Configuración:** la configuración de los proyectos debe también seguir un modelo específico ya que este es utilizado por los módulos de importación para interpretar los atributos expuestos por la fuente de datos externa. Como esta información se almacena en estructuras XML, se documenta en un XSD.

Ambos esquemas están incluidos en el anexo de la Sección 8.2.

4.4.6 Verificación y aceptación de la arquitectura del sistema (DSI 7)

El objetivo de esta actividad es garantizar la calidad de las especificaciones del diseño del sistema de información y la viabilidad del mismo, como paso previo a la generación de las especificaciones de construcción.

Para cumplir dicho objetivo, se llevan a cabo las siguientes acciones:

- verificación de la calidad técnica de cada modelo o especificación;
- aseguramiento de la coherencia entre los distintos modelos; y
- aceptación del diseño de la arquitectura.

4.4.6.1 Verificación de las especificaciones de diseño (DSI 7.1)

El objetivo de esta tarea es asegurar la calidad formal de los distintos modelos, conforme a la técnica seguida para la elaboración de cada producto y a las normas y estándares especificados en el catálogo de normas.

4.4.6.2 Análisis de consistencia de las especificaciones de diseño (DSI 7.2)

El objetivo de esta tarea es asegurar que las especificaciones del diseño son coherentes entre sí, comprobando la falta de ambigüedades o duplicación de información. Esta consistencia se asegura entre especificaciones de diseño, y con respecto a los modelos del análisis.

Las diferentes comprobaciones se fundamentan generalmente en técnicas matriciales o de revisión entre los elementos comunes de los distintos modelos.

4.4.6.2.1 Matrices de trazabilidad

La verificación de la completitud y consistencia de las especificaciones de diseño se realizó mediante la utilización de matrices de trazabilidad. La primera de ellas, Tabla 27, asegura que

todos los casos de uso identificados en el proceso de Análisis (ASI) sean implementados a través de una o más clases identificadas en la actividad de Diseño de Casos de Uso Reales (DSI 3). Asimismo, se validó que no hubiera clases sin su correspondiente caso de uso de origen.

Clase \ CU	CU.1	CU.2	CU.3	CU.4	CU.5	CU.6	CU.6.1	CU.7	CU.7.1	CU.7.2
Dashboard	x	x	x	x	x	x				
DashboardScr	x	x	x	x	x	x				
DashboardSrvScr	x									
DashPlannedActivitiesTable	x	x	x		x					
DashActionsTab	x	x	x		x					
ConfigProjects						x	x	x		
ConfigProjectsScr						x	x	x		
ConfigProjectsSrvScr						x				
ConfigSourcesTable						x				
ConfigSaver						x	x			
MSExcelParser								x		x
MSProjectParser								x	x	
DBConnection	x					x	x	x		
DBActivitiesReader	x									
DBActivitiesUpdater								x		
Dialog						x		x		

Tabla 27: Matriz de trazabilidad entre los casos de uso del Modelo de Casos de Uso y las clases del Modelo de Diseño.

La siguiente verificación asegura que el modelado de datos también tenga su correspondencia en el modelo de casos de uso identificado en el proceso de Análisis (ASI). Esta verificación se encuentra en la Tabla 28, donde se analiza la correspondencia entre los casos de uso y las tablas y vistas identificadas en la actividad Diseño Físico de Datos (DSI 6).

Clase \ CU	CU.1	CU.2	CU.3	CU.4	CU.5	CU.6	CU.6.1	CU.7	CU.7.1	CU.7.2
Projects	x					x	x	x		
Sources	x					x	x	x		
RefSourceTypes						x	x	x		
Activities	x							x		
RefActivityTypes	x							x		
RefPriorities	x							x		
RefStates	x							x		
Users	x					x		x		
RelActivityUsers	x							x		
RefRoles						x		x		
RelUsersProjectsRoles	x					x		x		
AllActivities (vista)	x	x	x		x					
GetUsersByUser (vista)	x									
ProjectsByPM (vista)						x				
ProjectSources (vista)							x			
Actividades (XSD)	x	x	x	x	x			x	x	x
Configuración (XSD)						x	x	x	x	x

Tabla 28: Matriz de trazabilidad entre los casos de uso del Modelo de Casos de Uso y las tablas y vistas del Modelo de Físico de Datos. Adicionalmente se incluyó la trazabilidad correspondiente a los XSD, los cuales definen la estructura de los contenedores de datos en XML.

4.4.6.3 Aceptación de la arquitectura del sistema (DSI 7.3)

El objetivo de esta tarea es obtener la aceptación de la arquitectura del sistema de información y de los requisitos de operación y seguridad, con el fin de poder valorar su impacto en la instalación.

4.4.6.3.1 Aceptación técnica del diseño

Luego de una reunión con líderes técnicos de la organización involucrados en este desarrollo, se da la aprobación formal del diseño del sistema.

4.4.7 Generación de especificaciones de construcción (DSI 8)

En esta actividad se generan las especificaciones para la construcción del sistema de información, a partir del diseño detallado.

Estas especificaciones definen la construcción del sistema de información a partir de las unidades básicas de construcción (en adelante, componentes), entendiendo como tales unidades independientes y coherentes de construcción y ejecución, que se corresponden con un empaquetamiento físico de los elementos del diseño de detalle, como pueden ser módulos, clases o especificaciones de interfaz.

La división del sistema de información en subsistemas de diseño proporciona, por continuidad, una primera división en subsistemas de construcción, definiendo para cada uno de ellos los componentes que lo integran. Si se considera necesario, un subsistema de diseño se podrá dividir a su vez en sucesivos niveles para mayor claridad de las especificaciones de construcción.

Las dependencias entre subsistemas de diseño proporcionan información para establecer las dependencias entre los subsistemas de construcción y, por lo tanto, definir el orden o secuencia que se debe seguir en la construcción y en la realización de las pruebas.

También se generan las especificaciones necesarias para la creación de las estructuras de datos en los gestores de bases de datos o sistemas de ficheros.

El producto resultante de esta actividad es el conjunto de las especificaciones de construcción del sistema de información, que comprende:

- especificación del entorno de construcción;
- descripción de subsistemas de construcción y dependencias;
- descripción de componentes;
- plan de integración del sistema de información;
- especificación detallada de componentes; y
- especificación de la estructura física de datos.

4.4.7.1 Especificación del entorno de construcción (DSI 8.1)

El objetivo de esta tarea es la definición detallada y completa del entorno necesario para la construcción de los componentes del sistema de información.

Se propone que la especificación del entorno se realice según los siguientes conceptos:

- entorno tecnológico: hardware, software y comunicaciones;
- herramientas de construcción, generadores de código, compiladores, etc.;
- restricciones técnicas del entorno;

- planificación de capacidades previstas, o la información que estime oportuno el departamento de sistemas para efectuar dicha planificación; y
- requisitos de operación y seguridad del entorno de construcción.

4.4.7.1.1 Especificaciones de construcción del sistema de información

El entorno requerido para la construcción del sistema debe cumplir con los requisitos identificados en la actividad de Identificación de requisitos de diseño y construcción (DSI 1.2). En adición a esos requisitos, es necesario asegurarse de cumplir con lo siguiente:

- La edición de código debe realizarse con Notepad++, incluyendo las extensiones para reconocer y editar archivos XML.
- El servidor web debe residir en la estación de trabajo de desarrollo y debe tener habilitadas las opciones de depuración del lado del cliente y del servidor.
- El acceso a la base de datos se podrá realizar con CuteSQL si se desea realizar un acceso más rápido que el que provee las propias herramientas del MS SQL Server utilizado.

4.4.7.2 Definición de componentes y subsistemas de construcción (DSI 8.2)

La especificación de los subsistemas de construcción se realiza a partir de los subsistemas de diseño, con una continuidad directa, permitiéndose a su vez un mayor nivel de detalle agrupando componentes en subsistemas dentro de un subsistema de construcción.

Los componentes se definen mediante la agrupación de elementos del diseño de detalle de cada subsistema de diseño. En principio, cada módulo o clase y cada formato individual de interfaz se corresponden con un componente, aunque se pueden agrupar o redistribuir módulos o clases en componentes, siguiendo otros criterios más oportunos.

Los subsistemas de construcción y las dependencias entre subsistemas y entre componentes de un subsistema recogen aspectos prácticos relativos a la plataforma concreta de construcción y ejecución.

La asignación de subsistemas de construcción a nodos, por continuidad con el diseño, determina la distribución de los componentes que lo integran.

4.4.7.2.1 Especificaciones de construcción del sistema de información

Debido a que el sistema es pequeño, no se dividirá la construcción en subsistemas.

4.4.8 Especificación técnica del plan de pruebas (DSI 10)

En esta actividad se realiza la especificación de detalle del plan de pruebas del sistema de información para cada uno de los niveles de prueba establecidos en el proceso Análisis del Sistema de Información:

- pruebas del sistema;
- pruebas de implantación; y
- pruebas de aceptación.

Para ello se toma como referencia el plan de pruebas, que recoge los objetivos de la prueba de un sistema, establece y coordina una estrategia de trabajo, y provee del marco adecuado para planificar paso a paso las actividades de prueba.

El catálogo de requisitos, el catálogo de excepciones y el diseño detallado del sistema de información, permiten la definición de las verificaciones que deben realizarse en cada nivel de prueba para comprobar que el sistema responde a los requisitos planteados. La asociación de las distintas verificaciones a componentes, grupos de componentes y subsistemas, o al sistema de información completo, determina las distintas verificaciones de cada nivel de prueba establecido.

Las pruebas del sistema, de implantación y de aceptación corresponden a verificaciones asociadas al sistema de información, y reflejan distintos propósitos en cada tipo de prueba.

Las pruebas del sistema son pruebas de integración del sistema de información completo. Permiten probar el sistema en su conjunto y con otros sistemas con los que se relaciona para verificar que las especificaciones funcionales y técnicas se cumplen.

Las pruebas de implantación incluyen las verificaciones necesarias para asegurar que el sistema funcionará correctamente en el entorno de operación al responder satisfactoriamente a los requisitos de rendimiento, seguridad y operación, y coexistencia con el resto de los

sistemas de la instalación, y conseguir la aceptación del sistema por parte del usuario de operación.

Las pruebas de aceptación van dirigidas a validar que el sistema cumple los requisitos de funcionamiento esperado, recogidos en el catálogo de requisitos y en los criterios de aceptación del sistema de información, y conseguir la aceptación final del sistema por parte del usuario.

Las pruebas del sistema se llevan a cabo en el proceso Construcción del Sistema de Información (CSI), mientras que las pruebas de implantación y aceptación se realizan en el proceso Implantación y Aceptación del Sistema (IAS).

Como resultado de esta actividad se actualiza el plan de pruebas con la información siguiente:

- especificación del entorno de pruebas;
- especificación técnica de niveles de prueba; y
- planificación de las pruebas.

4.4.8.1 Especificación del entorno de pruebas (DSI 10.1)

El objetivo de esta tarea es la definición detallada y completa del entorno necesario para la realización de las pruebas del sistema: unitarias, de integración, de implantación y de aceptación.

4.4.8.1.1 Plan de pruebas

El entorno de pruebas debe cumplir con los requisitos identificados en la actividad de Definición de requisitos del entorno de pruebas (ASI 10.2).

4.4.8.2 Especificación técnica de niveles de prueba (DSI 10.2)

El objetivo de esta tarea es el diseño detallado de los distintos niveles de prueba, especificados en el plan de pruebas elaborado en el proceso Análisis del Sistema de Información.

De acuerdo a la arquitectura del sistema propuesta y a las características intrínsecas del diseño del sistema de información, se definen en detalle las distintas verificaciones a realizar sobre el

sistema, conforme a los niveles de prueba establecidos, teniendo en cuenta que una verificación puede ser aplicable a varios componentes o grupos de componentes.

Estas verificaciones deben cubrir aspectos funcionales y no funcionales, considerando las excepciones que puedan producirse, así como las soluciones de diseño adoptadas, tanto del propio diseño de detalle del sistema de información, como de la utilización de subsistemas de soporte propios de la instalación.

Las verificaciones a realizar se especifican detallando:

- **Ámbito de aplicación** (prueba del sistema, de implantación o aceptación) y objetivo.
- **Casos de prueba asociados:** se definen en detalle los casos de prueba y se detalla cómo proceder en la ejecución de dichos casos, describiendo todas las entradas necesarias para ejecutar la prueba, y las relaciones de secuencialidad existentes entre las entradas, así como todas aquellas salidas que se espera obtener una vez ejecutado el caso de prueba, y las características especiales requeridas, como por ejemplo, tiempo de respuesta.
- **Procedimientos de prueba:** se determina el conjunto de pasos a seguir para asegurar que los casos de prueba se ejecutan adecuadamente, especificando:
 - casos de prueba a los que se aplica el procedimiento;
 - recursos hardware y software necesarios para ejecutar el procedimiento;
 - requisitos especiales o acciones necesarias para iniciar la ejecución; y
 - requisitos especiales o acciones necesarias a realizar durante la ejecución del procedimiento.
- **Entorno de prueba:** herramientas adicionales, condicionantes especiales de ejecución, etc.
- **Criterios de aceptación de la prueba.**
- **Análisis y evaluación de resultados.**

Como resultado final, se obtiene la relación de verificaciones que permiten comprobar:

- la integración del sistema de información en su totalidad (pruebas del sistema);
- el ajuste del sistema a las necesidades para las que fue creado, de acuerdo a las características del entorno en el que se va a implantar (pruebas de implantación); y
- la respuesta satisfactoria del sistema a los requisitos especificados por el usuario (pruebas de aceptación).

4.4.8.2.1 Plan de pruebas

Dado que durante las pruebas de sistema el foco estará en asegurar que la funcionalidad definida para el sistema está bien implementada.

Se espera que las pruebas sean ejecutadas por recursos con experiencia en pruebas funcionales de aplicaciones web, por lo que los casos de prueba no están escritos con el máximo nivel de detalle posible. Se espera que con cada paso definido para cada caso, quien realice la prueba explore las diferentes alternativas relacionadas con el respectivo paso.

Se definen a continuación los casos de prueba que se deberán ejecutar y se sugiere realizarlos en el orden en el que están numerados para minimizar el tiempo de ejecución.

ID: CP.1		
Objetivo: Asegurar que los controles de la pantalla principal funcionan correctamente.		
Precondiciones: - Debe existir el proyecto “B”. - El usuario debe estar dado de alta en el sistema. - El usuario debe tener permisos de miembro de equipo en “B”. - El usuario debe tener actividades planificadas y acciones asignadas en “B”.		
Paso	Descripción	Resultado esperado
1	El usuario ingresa al sistema.	El sistema muestra la pantalla principal con la siguiente información: - Menú proyectos con el valor “Todos los proyectos” seleccionado y la opción “B”. - Menú de usuarios con el valor “Todos los usuarios” seleccionado y una opción para el propio usuario. - Botón “Configurar...” inactivo. - Tabla de actividades planificadas y tabla de acciones con todas las actividades del usuario. - Cantidad de acciones en cada tabla a la derecha del título respectivo. - Controles de reducción de ambas tablas ([-]).
Postcondiciones: N/A		

Tabla 29: Caso de prueba para comprobar la pantalla principal.

ID: CP.2		
Objetivo: Asegurar que los controles de la pantalla principal funcionan correctamente.		
Precondiciones: - Deben existir los proyectos “A”, “B”, “C” y “D”. - Deben existir los usuarios “X”, “Y” y “Z”. - El usuario debe estar dado de alta en el sistema. - El usuario debe tener permisos de líder de equipo en “A” y “C”. - El usuario debe tener permisos de miembro de equipo en “B”. - El usuario debe tener actividades planificadas y acciones asignadas en “A” y “B”. - El usuario debe no tener actividades planificadas o acciones asignadas en “C”. - Deben haber actividades planificadas y acciones asignadas al usuario “X” en “A”. - Deben haber actividades planificadas y acciones asignadas al usuario “Y” en “C”. - Deben haber actividades planificadas y acciones asignadas al usuario “Z” en “D”.		
Paso	Descripción	Resultado esperado
1	El usuario ingresa al sistema.	El sistema muestra la pantalla principal con la siguiente información: - Menú proyectos con el valor “Todos los proyectos” seleccionado y las opciones “A”, “B” y “C”. - Menú de usuarios con el valor “Todos los usuarios” seleccionado y opciones para el propio usuario y los usuarios “X” e “Y”. - Botón “Configurar...” activo. - Tabla de actividades planificadas y tabla de acciones con todas las actividades del usuario y de los usuarios “X” e “Y”. - Cantidad de acciones en cada tabla a la derecha del título respectivo. - Controles de reducción de ambas tablas ([-]).
2	El usuario cambia la selección del menú de proyectos.	Las tablas de actividades planificadas y de acciones se refrescan con cada selección, se muestran correctamente las actividades y las cantidades.
3	El usuario cambia la selección del menú de usuarios.	Las tablas de actividades planificadas y de acciones se refrescan con cada selección, se muestran correctamente las actividades y las cantidades.
4	El usuario cambia en forma combinada la selección del menú de proyectos y del de usuarios.	Las tablas de actividades planificadas y de acciones se refrescan con cada selección, se muestran correctamente las actividades y las cantidades.
5	El usuario selecciona “Todos los proyectos” y “Todos los usuarios” de los respectivos menús.	Las tablas de actividades planificadas y de acciones se refrescan, mostrando nuevamente todas las actividades y las cantidades originales.
6	El usuario activa y desactiva los controles de reducción de tablas.	La tabla se ocultan y el control se convierte en “[+]” y, al volver a activarlo, la tabla se muestra y el control se convierte en “[-]”, siempre mostrando correctamente las actividades y las cantidades respectivas.
7	El usuario selecciona los encabezados de columna de las tablas.	La tabla se reordena de acuerdo a los valores de la columna seleccionada en forma ascendente y, al volver a seleccionar el encabezado, en forma descendente.
Postcondiciones: N/A		

Tabla 30: Caso de prueba para comprobar el funcionamiento de las tablas de actividades en la pantalla principal.

ID: CP.3		
Objetivo: Asegurar que los controles de la pantalla de configuración funcionan correctamente.		
Precondiciones: Ídem CP.2.		
Paso	Descripción	Resultado esperado
1	El usuario ingresa al sistema y presiona el botón “Configurar...”.	El sistema muestra la pantalla de configuración con la siguiente información: - Menú proyectos con una opción para el proyecto “A”. - Botón “Cerrar” activo. - Área de edición con la configuración en formato XML del proyecto “A”. - Botón “Guardar cambios al proyecto” activo. - Tabla de fuentes de datos en correspondencia con la configuración del proyecto, en cantidad, nombre y tipo de fuente. - Un botón “Actualizar” por cada fuente de actividad.
2	El usuario altera la configuración del proyecto, generando un XML mal formado y volviendo a generar un XML bien formado.	El botón “Guardar cambios al proyecto” se desactiva mientras el XML está mal formado.
3	El usuario altera la configuración de una de las fuentes y guarda los cambios.	El sistema informa al usuario del éxito de la operación.
4	El usuario presiona el botón “Cerrar” y, en la pantalla principal vuelve a presionar el botón “Configurar...”.	El sistema muestra nuevamente la configuración y los cambios realizados han sido almacenados correctamente.
5	El usuario cambia la selección del menú de proyectos.	El sistema actualiza el área editable con la configuración del proyecto respectivo y la tabla de fuentes con las entradas correspondientes al proyecto seleccionado.
Postcondiciones: - La configuración del proyecto “A” ha sido alterada y almacenada en la base de datos.		

Tabla 31: Caso de prueba para comprobar la pantalla de configuración.

ID: CP.4		
Objetivo: Asegurar que los cambios en la configuración de los proyectos son procesados correctamente.		
Precondiciones: - Deben existir los proyectos “A” y “B”. - Deben existir los usuarios “X”, “Y” y “Z”. - El usuario debe estar dado de alta en el sistema. - El usuario debe tener permisos de líder de equipo en “A”. - Los usuarios “X” e “Y” deben tener permisos de miembro de equipo en “A”. - Los usuarios “Y” y “Z” deben tener permisos de miembro de equipo en “B”. - El usuario se encuentra en la pantalla de configuración.		
Paso	Descripción	Resultado esperado
1	El usuario cambia la configuración de “A” agregando a “Z” como usuario del proyecto.	La configuración es almacenada correctamente y el usuario “Z” está ahora vinculado al proyecto “A” con rol de miembro de equipo. No hay otros cambios en la relación usuarios-proyectos.
2	El usuario cambia el nombre con el que el usuario “X” será reconocido en la fuente (valor “from”).	La configuración es almacenada correctamente y no hay cambios en la relación usuarios-proyectos.
3	El usuario remueve al usuario “Y” de la fuente correspondiente.	La configuración es almacenada correctamente y no hay cambios en la relación usuarios-proyectos.
4	El usuario agrega a una fuente un usuario no registrado en el sistema.	La configuración es almacenada correctamente y el sistema alerta al usuario del error al buscar el usuario inexistente. No hay cambios en la relación usuarios-proyectos.
Postcondiciones: - La configuración de “A” cambió. - Solo se agregó una relación entre “Z” y “A”.		

Tabla 32: Caso de prueba para comprobar la configuración de proyectos.

ID: CP.5		
Objetivo: Asegurar que los cambios en la configuración de los proyectos son procesados correctamente.		
Precondiciones: - Deben existir los proyectos “A” y “B”. - Deben existir las fuentes “F” y “G” en “A”. - Deben existir actividades planificadas o acciones para “G”. - El usuario debe estar dado de alta en el sistema. - El usuario debe tener permisos de líder de equipo en “A”. - El usuario se encuentra en la pantalla de configuración.		
Paso	Descripción	Resultado esperado
1	El usuario cambia la configuración de “A” agregando a “H” como fuente del proyecto.	La configuración es almacenada correctamente y “H” aparece en la lista de fuentes de “A”.
2	El usuario cambia los parámetros de “F”.	La configuración es almacenada correctamente.
3	El usuario elimina “G”.	La configuración es almacenada correctamente y “G” no aparece más en la lista de fuentes de “A”. Todas las actividades planificadas o acciones relacionadas con “G” son eliminadas del sistema.
Postcondiciones: - La configuración de “A” cambió. - “H” es una nueva fuente de “A”, “F” está modificada y “G”, junto con sus actividades y acciones fue borrada.		

Tabla 33: Caso de prueba para comprobar los cambios a la configuración de proyectos.

ID: CP.6		
Objetivo: Asegurar que las actividades son importadas desde fuentes MS Excel correctamente.		
Precondiciones: - Deben existir los proyectos “A” y “B”. - Deben existir las fuentes “F”, del tipo “MSEExcel” y “G” en “A”. - Deben no existir actividades planificadas o acciones para “F”. - Deben existir actividades planificadas o acciones para “G”. - El usuario debe estar dado de alta en el sistema. - El usuario debe tener permisos de líder de equipo en “A”. - El usuario se encuentra en la pantalla de configuración.		
Paso	Descripción	Resultado esperado
1	El usuario importa actividades de “F”.	El sistema notifica el éxito de la importación y se verifica el correcto procesamiento de lo siguiente: - Tipo de fuente (“PLAN”, “PREV”, “CORR”). - Identificador - Descripción, incluyendo caracteres de idiomas latinos. - Prioridad, con traducción. - Estado, con traducción y filtrado. - Recursos, con multiplicidad y traducción. Se verifica que no hubo alteraciones en las actividades de “G” ni de “B”.
2	El usuario agrega, modifica y quita actividades de “F” e importa las actividades nuevamente.	El sistema notifica el éxito de la importación y se verifica el correcto procesamiento de los cambios.
3	El usuario asigna a una actividad de “F” un recurso inexistente e importa las actividades nuevamente.	El sistema notifica el éxito de la importación y se verifica que esa actividad no fue importada y fue eliminada.
4	El usuario asigna una actividad de “F” una prioridad o un estado inexistente e importa las actividades nuevamente.	El sistema notifica el éxito de la importación y se verifica que esa actividad no fue importada y fue eliminada.
5	El usuario modifica la configuración, introduciendo una ruta de acceso incorrecta para la fuente.	El sistema notifica del problema al usuario. No se realizan alteraciones de ningún tipo a los datos existentes.
Postcondiciones: - La configuración de “A” queda modificada. - Las actividades de “F” son importadas al sistema.		

Tabla 34: Caso de prueba para comprobar la importación desde MS Excel.

ID: CP.7		
Objetivo: Asegurar que las actividades son importadas desde fuentes MS Project correctamente.		
Precondiciones: - Deben existir los proyectos “A” y “B”. - Deben existir las fuentes “F” y “G”, del tipo “MSProject” en “A”. - Deben existir actividades planificadas o acciones para “F”. - Deben no existir actividades planificadas o acciones para “G”. - El usuario debe estar dado de alta en el sistema. - El usuario debe tener permisos de líder de equipo en “A”. - El usuario se encuentra en la pantalla de configuración.		
Paso	Descripción	Resultado esperado
1	El usuario importa tareas de “G”.	El sistema notifica el éxito de la importación y se verifica el correcto procesamiento de lo siguiente: - Tipo de fuente (“PLAN”, “PREV”, “CORR”). - Identificador - Descripción, incluyendo caracteres de idiomas latinos. - Prioridad, con traducción. - Estado, con traducción y filtrado. - Recursos, con multiplicidad y traducción. Se verifica que no hubo alteraciones en las actividades de “F” ni de “B”.
2	El usuario agrega, modifica y quita tareas de “G” e importa las actividades nuevamente.	El sistema notifica el éxito de la importación y se verifica el correcto procesamiento de los cambios.
3	El usuario asigna a una actividad de “G” un recurso inexistente e importa las actividades nuevamente.	El sistema notifica el éxito de la importación y se verifica que esa actividad no fue importada y fue eliminada.
4	El usuario modifica la configuración, introduciendo una ruta de acceso incorrecta para la fuente.	El sistema notifica del problema al usuario. No se realizan alteraciones de ningún tipo a los datos existentes.
Postcondiciones: - La configuración de “A” queda modificada. - Las actividades de “G” son importadas al sistema.		

Tabla 35: Caso de prueba para comprobar la importación desde MS Project.

4.4.9 Aprobación del diseño del sistema de información (DSI 12)

Luego de una reunión con líderes de proyecto y usuarios involucrados en este desarrollo, se da la aprobación formal para avanzar con la construcción del sistema.

4.5 Construcción del sistema de información (CSI)

En este proceso se genera el código de los componentes del Sistema de Información, se desarrollan todos los procedimientos de operación y seguridad y se elaboran todos los manuales de usuario final y de explotación con el objetivo de asegurar el correcto funcionamiento del Sistema para su posterior implantación.

Para conseguir dicho objetivo, en este proceso se realizan las pruebas de acuerdo al plan de pruebas establecido.

Asimismo, se define la formación de usuario final y, si procede, se construyen los procedimientos de migración y carga inicial de datos.

El producto Especificaciones de Construcción del Sistema de Información, obtenido en la actividad de Generación de Especificaciones de Construcción (DSI 8), es la base para la construcción del sistema de información. En dicho producto se recoge la información relativa al entorno de construcción del sistema de información, la especificación detallada de los componentes y la descripción de la estructura física de datos, tanto bases de datos como sistemas de ficheros. Opcionalmente, incluye un plan de integración del sistema de información, en el que se especifica la secuencia y organización de la construcción de los distintos componentes.

En la actividad Preparación del Entorno de Generación y Construcción (CSI 1), se asegura la disponibilidad de la infraestructura necesaria para la generación del código de los componentes y procedimientos del sistema de información.

Una vez configurado el entorno de construcción, se realiza la codificación y las pruebas de los distintos componentes que conforman el sistema de información, en las actividades:

- Generación del Código de los Componentes y Procedimientos (CSI 2), que se hace según las especificaciones de construcción del sistema de información, y conforme al plan de integración del sistema de información;
- Ejecución de las Pruebas Unitarias (CSI 3), dónde se llevan a cabo las verificaciones definidas en el plan de pruebas para cada uno de los componentes; y
- Ejecución de las Pruebas de Integración (CSI 4), que incluye la ejecución de las verificaciones asociadas a los subsistemas y componentes, a partir de los componentes verificados individualmente, y la evaluación de los resultados.

Una vez construido el sistema de información y realizadas las verificaciones correspondientes, se lleva a cabo la integración final del sistema de información en la actividad Ejecución de las Pruebas del Sistema (CSI 5), comprobando tanto las interfaces

entre subsistemas y sistemas externos como los requisitos, de acuerdo a las verificaciones establecidas en el plan de pruebas para el nivel de pruebas del sistema.

En la actividad Elaboración de los Manuales de Usuario (CSI 6), se genera la documentación de usuario final o explotación, conforme a los requisitos definidos en el proceso Diseño del Sistema de Información.

La formación necesaria para que los usuarios finales sean capaces de utilizar el sistema de forma satisfactoria se especifica en la actividad Definición de la Formación de Usuarios Finales (CSI 7).

La carga inicial de datos se realiza en la actividad Construcción de los Componentes y Procedimientos de Migración y Carga Inicial de Datos (CSI 8).

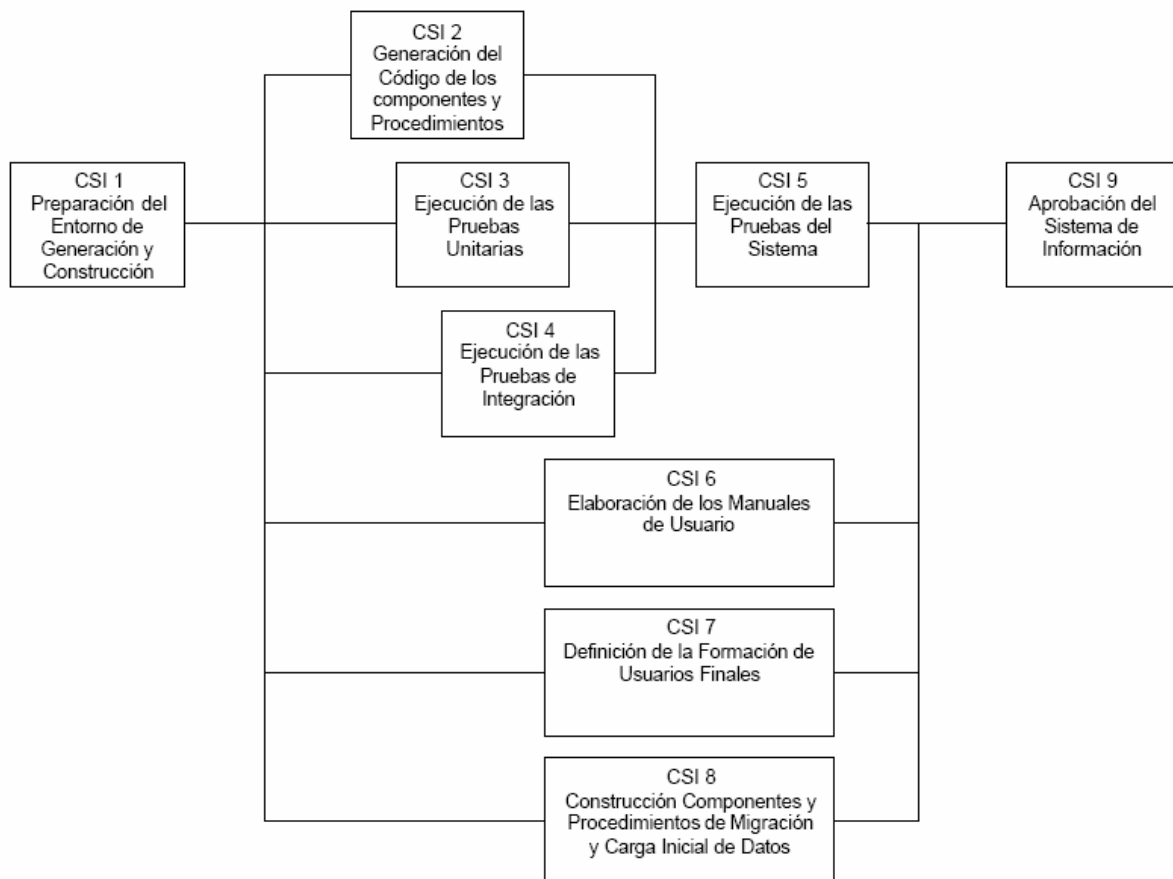


Figura 40: Actividades que componen el proceso de Construcción del Sistema de Información (CSI). Se pueden dividir en actividades exclusivamente técnicas: generación de código y pruebas; y actividades de soporte: elaboración de manuales, formación de usuarios y migración y carga inicial de datos.

4.5.1 Preparación del entorno de generación y construcción (CSI 1)

El objetivo de esta actividad es asegurar la disponibilidad de todos los medios y facilidades para que se pueda llevar a cabo la construcción del sistema de información. Entre estos medios, cabe destacar la preparación de los puestos de trabajo, equipos físicos y lógicos, gestores de bases de datos, bibliotecas de programas, herramientas de generación de código y bases de datos, entre otros.

Las características del entorno de construcción y sus requisitos de operación y seguridad, así como las especificaciones de construcción de la estructura física de datos, se establecen en la actividad Generación de Especificaciones de Construcción (DSI 8), y constituyen el punto de partida para la realización de esta actividad.

4.5.1.1 Implantación de la base de datos física (CSI 1.1)

En esta tarea hay que completar los siguientes pasos:

- crear los elementos del sistema gestor de base de datos o sistema de ficheros;
- inicializar la base de datos, cargando los datos considerados necesarios en el espacio de almacenamiento previamente definido.

4.5.1.1.1 Base de datos física

Tal como está definido en la actividad Generación de Especificaciones de Construcción (DSI 8), el motor de base de datos elegido para almacenar la información del sistema es un MS SQL Server 2000.

Una vez que el motor de base de datos estuvo disponible, se creó un espacio de trabajo (database) con el nombre “actdash” para contener la información del sistema.

Para la inicialización de la estructura del espacio de trabajo, se crearon manualmente todas las tablas y vistas del sistema junto con sus respectivas relaciones siguiendo las especificaciones delineadas en la actividad Diseño Físico de Datos (DSI 6). Las tablas creadas son las siguientes:

- AD_Activities
- AD_Projects

- AD_RefActivityTypes
- AD_RefPriorities
- AD_RefRoles
- AD_RefSourceTypes
- AD_RefStates
- AD_RelActivityUsers
- AD_RelUsersProjectsRoles
- AD_Sources
- AD_Users
- AD_VuAllActivities
- AD_VuGetUsersByUser
- AD_VuProjectsByPM
- AD_VuProjectSources

De manera similar, se procedió a la inicialización de los datos de las tablas de referencia, paso necesario para el correcto funcionamiento del sistema. Las tablas actualizadas son las siguientes:

- AD_RefActivityTypes:

Type_ID	Type
<auto>	PLAN
<auto>	CORR
<auto>	PREV
<auto>	CHAN

Tabla 36: Datos iniciales de la tabla RefActivityTypes.

- AD_RefPriorities

Priority_ID	Priority	Priority_Label	Sort_Order
<auto>	VH	10	Muy Alta
<auto>	HI	20	Alta
<auto>	ME	50	Media
<auto>	LO	80	Baja
<auto>	VL	90	Muy Baja

Tabla 37: Datos iniciales de la tabla RefPriorities.

- AD_RefRoles

ID	Role	Short_Role
<auto>	Administrador	ADM
<auto>	Líder	PM
<auto>	Usuario	USR

Tabla 38: Datos iniciales de la tabla RefRoles.

- AD_RefSourceTypes

Source_Type_ID	Source_Type
<auto>	MSExcel
<auto>	MSProject

Tabla 39: Datos iniciales de la tabla RefSourceTypes.

- AD_RefStates

State_ID	State
<auto>	No Iniciada
<auto>	En Progreso

Tabla 40: Datos iniciales de la tabla RefStates.

Como paso final de la preparación de la base de datos, se procedió a la creación de dos scripts para automatizar todo este procedimiento en caso de ser necesario replicar la instalación en otro ambiente. Los scripts son:

- DB_CreateTables.sql: Crea todas las tablas y sus relaciones, como así también las vistas.
- DB_InitData.sql: Inicializa los datos de las tablas de referencia.

4.5.1.2 Preparación del entorno de construcción (CSI 1.2)

En esta tarea se prepara el entorno en el que se construirán los componentes del sistema de información, contemplando aspectos tales como:

- herramientas, como ser generadores de código, editores, compiladores, verificadores sintácticos, montadores de enlace;
- implementación de los procedimientos de operación y seguridad propios del entorno de construcción, de acuerdo a los requisitos de seguridad y operación establecidos en la tarea Especificación del Entorno de Construcción (DSI 8.1).

4.5.1.2.1 Entorno de construcción

En la estación de trabajo de desarrollo, la cual debe ser un equipo con sistema operativo Windows XP, se instalan y configuran los siguientes programas y herramientas:

- servidor web Internet Information Services 5.1, habilitando las opciones de depuración del lado del cliente y del servidor;
- aplicaciones de oficina MS Office Excel 2003 y MS Office Project 2003;
- editor Notepad++ y extensiones para archivos XML; y
- herramienta CuteSQL.

4.5.2 Generación del código de los componentes y procedimientos (CSI 2)

El objetivo de esta actividad es la codificación de los componentes del sistema de información, a partir de las especificaciones de construcción obtenidas en el proceso Diseño del Sistema de Información (DSI), así como la construcción de los procedimientos de operación y seguridad establecidos para el mismo.

En paralelo a esta actividad, se desarrollan las actividades relacionadas con las pruebas unitarias y de integración del sistema de información. Esto permite una construcción incremental, en el caso de que así se haya especificado en el plan de pruebas y en el plan de integración del sistema de información.

4.5.2.1 Generación del código de componentes (CSI 2.1)

En esta tarea se genera el código correspondiente a cada uno de los componentes del sistema de información, identificados en la tarea Definición de Componentes y Subsistemas de Construcción (DSI 8.2).

Para generar el código fuente se tienen en cuenta los estándares de nomenclatura, codificación y calidad recogidos en el catálogo de normas.

Con el fin de verificar que el código fuente especifica de forma correcta el componente, se realiza su compilación, verificando y corrigiendo los errores sintácticos.

4.5.2.1.1 Código fuente de los componentes

Basado en las especificaciones de la actividad de Diseño de Clases (DSI 4), se crearon todos los archivos que componen la interfaz gráfica, los scripts necesarios para procesar el comportamiento de las mismas y los archivos de transformación XSL.

El listado de archivos generados es el siguiente:

- DBConnection.asp
- dashboard.asp, dashboard.css, dashboard.vbs y dashboard_srv.asp
- DashTableActions.xsl
- DashTablePlan.xsl
- DBActivitiesReader.asp
- configProjects.asp, configProjects.vbs y configProjects_srv.asp
- ConfigSources.xsl
- DBActivitiesUpdater.asp
- DBConfigSaver.asp
- ParserMSExcel.vbs
- ParserMSProject.vbs

4.5.3 Ejecución de las pruebas unitarias (CSI 3)

En esta actividad se realizan las pruebas unitarias de cada uno de los componentes del sistema de información, una vez codificados, con el objeto de comprobar que su estructura es correcta y que se ajustan a la funcionalidad establecida.

En el plan de pruebas se ha definido el entorno necesario para la realización de cada nivel de prueba, así como las verificaciones asociadas a las pruebas unitarias, la coordinación y secuencia a seguir en la ejecución de las mismas y los criterios de registro y aceptación de los resultados.

4.5.3.1 Realización y evaluación de las pruebas unitarias (CSI 3.2)

El objetivo de esta tarea es comprobar el correcto funcionamiento de los componentes del sistema de información, codificados en la actividad Generación del Código de los Componentes y Procedimientos (CSI 2), conforme a las verificaciones establecidas en el plan de pruebas para el nivel de pruebas unitarias, en la actividad Especificación Técnica del Plan de Pruebas (DSI 10).

Para cada verificación establecida, se realizan las pruebas con los casos de pruebas asociados, efectuando el correspondiente análisis y evaluación de los resultados, y generando un registro conforme a los criterios establecidos en el plan de pruebas.

Seguidamente, se analizan los resultados de las pruebas unitarias, evaluándose las mismas para comprobar que los resultados son los esperados. Si los resultados no son los esperados hay que proceder a realizar las correcciones pertinentes.

4.5.3.1.1 Resultado y evaluación de las Pruebas Unitarias

Como fue especificado en la actividad Especificación Técnica del Plan de Pruebas (DSI 10), las pruebas unitarias se realizaron sin escribir casos de prueba ni registrar sus resultados.

Solo una vez finalizada la construcción y que las pruebas unitarias hayan sido satisfactorias, se procedió a la siguiente tarea.

4.5.4 Ejecución de las pruebas del sistema (CSI 5)

El objetivo de las pruebas del sistema es comprobar la integración del sistema de información globalmente, verificando el funcionamiento correcto de las interfaces entre los distintos subsistemas que lo componen y con el resto de sistemas de información con los que se comunica.

En la realización de estas pruebas es importante comprobar la cobertura de los requisitos, dado que su incumplimiento puede comprometer la aceptación del sistema por el equipo de operación responsable de realizar las pruebas de implantación del sistema, que se llevarán a cabo en el proceso Implantación y Aceptación del Sistema.

4.5.4.1 Preparación del entorno de las pruebas del sistema (CSI 5.1)

En esta tarea se preparan todos los recursos necesarios para realizar las pruebas del sistema, de acuerdo a las características del entorno establecidas en el plan de pruebas.

Para ello se asegura la disponibilidad del entorno y de los datos necesarios para ejecutar estas pruebas, se preparan las bibliotecas o librerías que se estimen oportunas para la realización de las mismas, así como los procedimientos manuales o automáticos asociados.

4.5.4.1.1 Entorno de pruebas del sistema

Las pruebas de sistema se realizaron en el mismo entorno de desarrollo, por lo la preparación de dicho entorno consistió en completar los siguientes pasos:

- eliminar la base de datos de desarrollo y recrearla utilizando los scripts generados en la actividad Preparación del Entorno de Generación y Construcción (CSI 1); y
- tomar la línea base de código generada en la actividad Ejecución de las Pruebas Unitarias (CSI 3), implantarla nuevamente en el servidor de aplicaciones web y reiniciar dicho servidor.

Al completar los pasos anteriores, se asegura que el ambiente de pruebas coincidirá exactamente con la línea base de código y estructura y contenido inicial de base de datos que pasó exitosamente las pruebas unitarias ejecutadas en la actividad Ejecución de las Pruebas Unitarias (CSI 3).

4.5.4.2 Realización de las pruebas del sistema (CSI 5.2)

El objetivo de esta tarea es comprobar la integración de todos los subsistemas y componentes del sistema de información, así como la interacción del mismo con otros sistemas de información con los que se relaciona, de acuerdo a las verificaciones establecidas para el nivel de pruebas del sistema.

Para cada verificación establecida, se realizan las pruebas con los casos de pruebas asociados, efectuando el correspondiente análisis e informe de los resultados y generando un registro conforme a los criterios establecidos en el plan de pruebas.

4.5.4.2.1 Resultado de las pruebas del sistema

Las pruebas de sistema se ejecutaron siguiendo la planificación de ciclos de prueba-corrección de errores. Los registros de ejecución pueden verse en el anexo Registro de ejecución de pruebas de sistema.

4.5.4.3 Evaluación del Resultado de las Pruebas del Sistema (CSI 5.3)

El objetivo de esta actividad es analizar los resultados de las pruebas del sistema de información y efectuar su evaluación. Dicha evaluación recoge el grado de cumplimiento de las mismas, y consiste en:

- comparar los resultados obtenidos con los esperados;
- identificar el origen de cada problema detectado para poder remitirlo a quien proceda, determinar la envergadura de las modificaciones y qué acciones deben llevarse a cabo para resolverlo de forma satisfactoria; e
- indicar si el plan de pruebas debe volver a realizarse total o parcialmente, y si será necesario contemplar nuevos casos de prueba no considerados anteriormente.

4.5.4.3.1 Evaluación del Resultado de las Pruebas del Sistema

En el primer ciclo de pruebas se encontraron los errores que pueden verse en el anexo Registro de errores. Debido a ello, se procedió al análisis de cada error, identificando la causa, corrigiéndola y verificando que la corrección fue correcta por medio de pruebas unitarias adicionales. Una vez corregidos todos los errores, se volvieron a ejecutar todos los casos de prueba sin encontrar nuevos errores.

4.5.5 Elaboración de los manuales de usuario (CSI 6)

4.5.5.1 Elaboración de los manuales de usuario (CSI 6.1)

El objetivo de esta tarea es elaborar la documentación de usuario, tanto usuario final como de explotación, de acuerdo a los requisitos recogidos en el catálogo de requisitos.

Los requisitos de documentación especifican aspectos relativos a los tipos de documentos a elaborar y estándares a seguir en la generación de los mismos, y para cada uno de ellos:

- formato y soporte en el que se desarrollarán;
- estructura; y
- distribución y mantenimiento de la documentación y número de copias a editar.

4.5.5.1.1 Manuales de Usuario

Siguiendo la línea de los manuales de usuario desarrollados para otras aplicaciones web de la organización, este estará desarrollado con base en un documento creado con MS Word y exportado a un archivo PDF. Éste estará disponible a través de un botón en la página principal del sistema.

El documento tiene una sección por cada pantalla y funcionalidad principal de la aplicación y su mantenimiento se realizará en conjunto con el mantenimiento del sistema mismo. De esta forma se asegura que haya una correspondencia permanente entre la funcionalidad y la interfaz del sistema y el contenido del manual.

El manual puede verse en el anexo Manual de usuario.

4.5.6 Definición de la formación de usuarios finales (CSI 7)

En esta actividad se establecen las necesidades de formación del usuario final, con el objetivo de conseguir la explotación eficaz del nuevo sistema.

Para la definición de la formación hay que tener en cuenta las características funcionales y técnicas propias del sistema de información, así como los requisitos relacionados con la formación del usuario final.

El producto resultante de esta actividad es la especificación de la formación de usuarios finales, que consta de los siguientes elementos:

- esquema de formación; y
- materiales y entornos de formación.

En el proceso Implantación y Aceptación del Sistema (IAS), se unifican las especificaciones de formación de cada sistema de información implicado en la implantación y se elabora un único plan de formación que esté alineado con el plan de implantación del sistema.

4.5.6.1 Definición del esquema de formación (CSI 7.1)

El objetivo de esta tarea es la definición del contenido de la formación del usuario final del sistema, realizando, a su vez, una estimación de la duración de los distintos apartados o acciones formativas que se contemplen.

La especificación de los esquemas de formación se puede particularizar para cada perfil de usuario identificado en el proceso Análisis del Sistema de Información (ASI), si se considera necesario.

4.5.6.1.1 Esquema de Formación

Para la formación de los usuarios del sistema se distinguirán dos perfiles: los líderes de proyecto y los miembros de equipo. Para cada perfil, se realizará un esquema de formación similar pero de contenido diferente.

A los miembros de equipo se los capacitará en la funcionalidad básica de la pantalla principal de la aplicación. Esto se realizará a través de dos reuniones de formación presenciales de 1 hora de duración, dando la posibilidad a los usuarios de elegir a qué evento asistir.

A los líderes de proyecto se los capacitará adicionalmente en la configuración del sistema y en las capacidades específicas de importación de datos desde archivos MS Project y MS Excel. Esto se realizará a través de una reunión de formación presencial de 2 horas de duración.

4.5.6.2 Especificación de los recursos y entornos de formación (CSI 7.2)

El objetivo de esta tarea es detallar los recursos necesarios para llevar a cabo la formación, relativos a los materiales de formación, equipos físicos y lógicos, aulas, etc.

También se determinan las características que debe reunir el entorno para realizar la formación, en cuanto a la necesidad de hacer cargas iniciales o migración de datos, activar los procedimientos de seguridad y control de acceso específicos etc.

4.5.6.2.1 Materiales y Entornos de Formación

Para la realización de las reuniones de formación se utilizará la Sala de Capacitación de la Organización, provista con estaciones de trabajo para todos los participantes y demás materiales didácticos necesarios.

Para la reunión de formación de los miembros de equipo, será necesario que la base de datos del sistema esté libre de datos y tenga configurado un usuario para el instructor y uno genérico para todos los participantes.

Para la reunión de formación de los líderes de proyecto, será necesario que la base de datos del sistema esté libre de datos y tenga configurado un usuario para el instructor y uno con permisos de líder para cada uno de los participantes.

En todos los casos también será necesario contar con archivos MS Project y MS Excel para utilizar de ejemplos y fuentes de datos durante las reuniones.

4.5.7 Aprobación del sistema de información (CSI 9)

4.5.7.1 Presentación y aprobación del sistema de información (CSI 9.1)

En esta tarea se recopilan los productos del sistema de información y se presentan al Comité de Seguimiento para su aprobación.

4.5.7.1.1 Aprobación del Sistema de Información

Luego de analizar los resultados de las pruebas de sistema en conjunto con el manual de usuarios y las especificaciones de formación, los líderes de proyecto y usuarios involucrados en este desarrollo dieron la aprobación formal del sistema.

4.6 Implantación y aceptación del sistema de información (IAS)

Este proceso tiene como objetivo principal la entrega y aceptación del sistema en su totalidad, y la realización de todas las actividades necesarias para el paso a producción del mismo.

En primer lugar, se revisa la estrategia de implantación que ya se determinó en el proceso Estudio de Viabilidad del Sistema (EVS). Se estudia su alcance y, en función de sus características, se define un plan de implantación y se especifica el equipo que lo va a llevar a cabo. Conviene señalar la participación del usuario de operación en las pruebas de implantación, del usuario final en las pruebas de aceptación, y del responsable de mantenimiento.

Las actividades previas al inicio de la producción incluyen la preparación de la infraestructura necesaria para configurar el entorno, la instalación de los componentes, la activación de los procedimientos manuales y automáticos asociados y, cuando proceda, la migración o carga inicial de datos. Para ello se toman como punto de partida los productos software probados, obtenidos en el proceso Construcción del Sistema de Información (CSI) y su documentación asociada.

Se realizan las pruebas de implantación y de aceptación del sistema en su totalidad. Responden a los siguientes propósitos:

- Las pruebas de implantación cubren un rango muy amplio, que va desde la comprobación de cualquier detalle de diseño interno hasta aspectos tales como las comunicaciones. Se debe comprobar que el sistema puede gestionar los volúmenes de información requeridos, se ajusta a los tiempos de respuesta deseados y que los procedimientos de respaldo, seguridad e interfaces con otros sistemas funcionan correctamente. Se debe verificar también el comportamiento del sistema bajo las condiciones más extremas.
- Las pruebas de aceptación se realizan por y para los usuarios. Tienen como objetivo validar formalmente que el sistema se ajusta a sus necesidades.

Asimismo, se llevan a cabo las tareas necesarias para la preparación del mantenimiento, siempre y cuando se haya decidido que éste va a efectuarse. En cualquier caso, es necesario que la persona que vaya a asumir el mantenimiento conozca el sistema, antes de su incorporación al entorno de producción.

Además hay que determinar los servicios (y niveles para cada uno) que requiere el sistema que se va a implantar, y el acuerdo que se adquiere una vez que se inicie la producción. Hay que distinguir entre servicios de gestión de operaciones (servicios por lotes, seguridad,

comunicaciones, etc.) y servicios al cliente (servicio de atención a usuarios, mantenimiento, etc.) que se deben negociar en cuanto a recursos, horarios, coste, etc. Se fija el nivel con el que se prestará el servicio como indicador de la calidad del mismo.

Conviene señalar que la implantación puede ser un proceso iterativo que se realiza de acuerdo al plan establecido para el comienzo de la producción del sistema en su entorno de operación. Para establecer este plan se tiene en cuenta:

- el cumplimiento de los requisitos de implantación definidos en la actividad Establecimiento de Requisitos (ASI 2) y especificados en la actividad Establecimiento de Requisitos de Implantación (DSI 11); y
- la estrategia de transición del sistema antiguo al nuevo.

Finalmente, se realizan las acciones necesarias para el inicio de la producción.

En la Figura 41 se muestra la relación de actividades de este proceso.

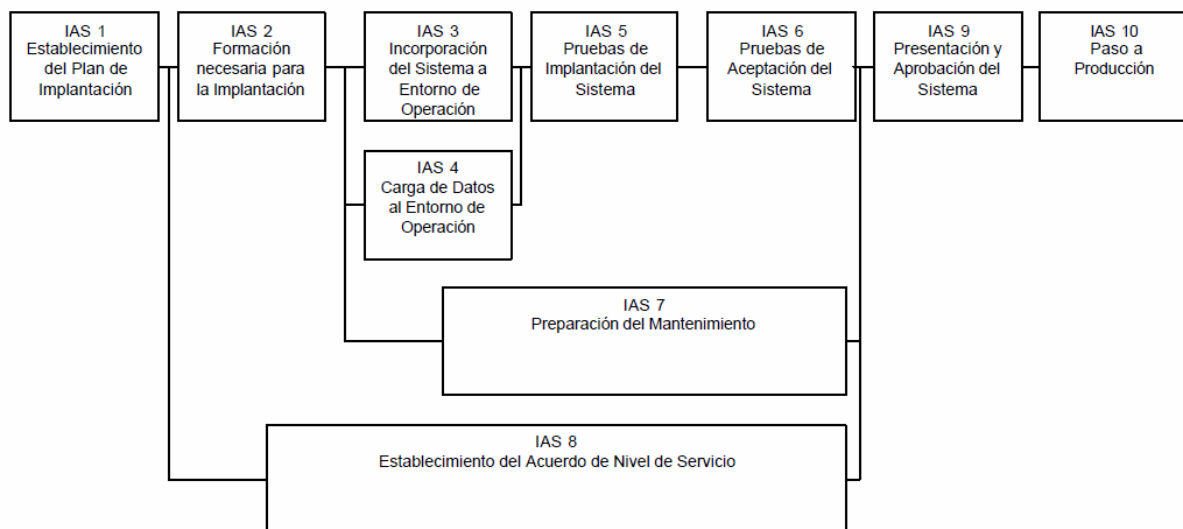


Figura 41: Actividades que componen el proceso de Implantación y Aceptación del Sistema de Información (IAS). Estas actividades parten de la planificación de la implantación, pasan por la formación de los usuarios, las pruebas y terminan con el paso a producción del sistema. Como paso previo se definen los niveles de servicio y se prepara el esquema de mantenimiento.

4.6.1 Establecimiento del plan de implantación (IAS 1)

En esta actividad se revisa la estrategia de implantación para el sistema, establecida inicialmente en el proceso Estudio de Viabilidad del Sistema (EVS). Se identifican los

distintos sistemas de información que forman parte del sistema objeto de la implantación. Para cada sistema se analizan las posibles dependencias con otros proyectos, que puedan condicionar el plan de implantación.

Una vez estudiado el alcance y los condicionantes de la implantación, se decide si ésta se puede llevar a cabo. Será preciso establecer, en su caso, la estrategia que se concretará de forma definitiva en el plan de implantación.

Se constituye el equipo de implantación, determinando los recursos humanos necesarios para la propia instalación del sistema, para las pruebas de implantación y aceptación, y para la preparación del mantenimiento. Se identifican, para cada uno de ellos, sus perfiles y niveles de responsabilidad.

4.6.1.1 Definición del plan de implantación (IAS 1.1)

La estrategia de implantación del sistema se habrá determinado en función de la envergadura del sistema, es decir el número de sistemas de información implicados en la implantación y la cobertura geográfica, cuyo alcance depende de las características y complejidad de los sistemas de información que conforman el sistema objeto de la implantación.

Se revisan los requisitos de implantación (instalación, infraestructura, formación) y los procedimientos implicados en la implantación, establecidos para cada uno de los sistemas de información en la tarea Especificación de Requisitos de Operación y Seguridad (DSI 1.7), con el fin de asegurar su adecuación a la estrategia global de implantación.

Una vez analizada la información anterior, se define un plan de implantación que permita calcular adecuadamente el esfuerzo y los recursos necesarios para llevar a cabo con éxito la implantación. Dicho plan debe contemplar todas las tareas relacionadas con:

- la formación necesaria para la implantación, tanto a usuarios finales como al equipo que se encarga de realizar las pruebas de aceptación del sistema;
- la preparación de la infraestructura necesaria para la incorporación del sistema al entorno de operación;
- la instalación de todos los componentes y procedimientos manuales y automáticos asociados a cada sistema de información implicado en la implantación;

- la ejecución de los procedimientos de carga inicial de datos;
- la realización de las pruebas de aceptación del sistema; y
- la formalización del plan de mantenimiento.

4.6.1.1.1 Plan de Implantación

Las pruebas de aceptación del sistema serán realizadas por un subconjunto de los líderes de proyecto involucrados en el desarrollo del presente sistema. Para poder realizar esta tarea, serán capacitados previamente de acuerdo a lo definido en la tarea Definición de la Formación de Usuarios Finales (CSI 7). Una vez finalizada la formación, se procederá a la ejecución de las pruebas de aceptación del sistema.

El sistema será incorporado a la organización en dos etapas: en la primera, se realizará un piloto con usuarios y proyectos seleccionados a fin de validar la adecuación del sistema a las necesidades de la organización; en la segunda, partiendo de un piloto exitoso, se procederá a la incorporación masiva del sistema a todos los proyectos. Por lo tanto, para la etapa de piloto, se utilizará la aplicación en el entorno que fuera utilizado para realizar las pruebas de sistema. Solo cuando se haya declarado completo el piloto, se procederá a la implantación del sistema en un entorno completamente productivo.

Durante la ejecución del piloto, se definirá el esquema de mantenimiento del sistema, el cual será aplicado solamente una vez que el sistema sea implantado en el entorno productivo y comience su utilización masiva en la organización.

4.6.1.2 Especificación del equipo de implantación (IAS 1.2)

Se constituye el equipo de trabajo necesario para llevar a cabo la implantación y aceptación del sistema, según el plan de implantación establecido en la tarea anterior.

Para ello se identifican, en función del nivel de esfuerzo requerido, los distintos participantes implicados en la implantación del sistema (usuarios, equipo técnico y responsable de mantenimiento), determinando previamente sus perfiles, responsabilidades, nivel de implicación y fechas previstas de participación a lo largo de toda la implantación.

4.6.1.2.1 Equipo de implantación

El equipo de implantación estará conformado por las siguientes personas y perfiles:

- El líder de proyecto del sistema, quien deberá velar por la calidad final del producto y por la terminación en tiempo de todas las tareas necesarias para realizar correctamente la implantación del sistema.
- El desarrollador del sistema, quien deberá estar disponible de tiempo completo y será responsable de la preparación del entorno de aceptación y de la resolución de todos los problemas funcionales y técnicos que pudieran surgir durante las pruebas de aceptación.
- Dos líderes de equipo que hayan estado involucrados en el proyecto de desarrollo, quienes deberán estar con una disponibilidad parcial ya que serán responsables de ejecutar las pruebas de aceptación y de reportar todos los incidentes que encuentren durante las mismas.

En ambos casos, los recursos deberán estar disponibles durante toda la etapa de implantación a fin de asegurarse un cumplimiento de las fechas estipuladas para la terminación del proyecto.

4.6.2 Incorporación del sistema al entorno de operación (IAS 3)

En esta actividad se realizan todas las tareas necesarias para la incorporación del sistema al entorno de operación en el que se van a llevar a cabo las pruebas de implantación y aceptación del sistema.

Mientras que las pruebas unitarias, de integración y del sistema se pueden ejecutar en un entorno distinto de aquél en el que finalmente se implantará, las pruebas de implantación y aceptación del sistema deben ejecutarse en el entorno real de operación. El propósito es comprobar que el sistema satisface todos los requisitos especificados por el usuario en las mismas condiciones que cuando se inicie la producción.

Por tanto, como paso previo a la realización de dichas pruebas y de acuerdo al plan de implantación establecido, se verifica que los recursos necesarios están disponibles para que se pueda realizar, adecuadamente, la instalación de todos los componentes que integran el sistema, así como la creación y puesta a punto de las bases de datos en el entorno de operación.

Asimismo, se establecen los procedimientos de explotación y uso de las bases de datos de acuerdo a la normativa existente en dicho entorno.

4.6.2.1 Realización de la instalación (IAS 3.2)

Se realiza la instalación de todos los componentes del nuevo sistema, incluidos los procedimientos manuales y automáticos, de acuerdo al plan de implantación y a su ubicación física, establecida en el proceso Diseño del Sistema de Información. Se deben tener en cuenta los estándares y normativas por los que se rige la organización en los entornos de operación.

Asimismo, se prepara el entorno de datos identificando los sistemas de información que forman parte del sistema objeto de la implantación. Para cada uno de ellos:

- se recrean las bases de datos a partir del esquema físico elaborado en el proceso de construcción; y
- se preparan las autorizaciones de acceso a los datos para los distintos perfiles de usuarios.

4.6.3 Carga de datos al entorno de operación (IAS 4)

Teniendo en cuenta que los sistemas de información que forman parte del sistema a implantar pueden mejorar, ampliar o sustituir a otros ya existentes en la organización, puede ser necesaria una carga inicial y/o una migración de datos cuyo alcance dependerá de las características y cobertura de cada sistema de información implicado. Por tanto, la necesidad de una migración de datos puede venir determinada desde el proceso Estudio de Viabilidad del Sistema (EVS), en la actividad Selección de la Solución (EVS 6). Allí se habrá establecido la estrategia a seguir en la sustitución, evaluando las opciones del enfoque de desarrollo e instalación más apropiados para llevarlo a cabo.

En cualquier caso, en la actividad Diseño de la Migración y Carga Inicial de Datos (DSI 9) se habrán definido y planificado los procesos y procedimientos necesarios para llevar a cabo la migración, realizándose su codificación en la actividad Construcción de los Componentes y Procedimientos de Migración y Carga Inicial de Datos (CSI 8).

4.6.3.1 Carga inicial de datos (IAS 4.1)

Se realiza la carga inicial de datos del nuevo sistema, y se comprueba que ha finalizado correctamente.

4.6.4 Pruebas de aceptación del sistema (IAS 6)

Las pruebas de aceptación tienen como fin validar que el sistema cumple los requisitos básicos de funcionamiento esperado y permitir que el usuario determine la aceptación del sistema. Por este motivo, estas pruebas son realizadas por el usuario final que, durante este periodo de tiempo, debe plantear todas las deficiencias o errores que encuentre antes de dar por aprobado el sistema definitivamente.

Los Directores de los Usuarios revisan los criterios de aceptación, especificados previamente en el plan de pruebas del sistema, y dirigen las pruebas de aceptación final que llevan a cabo los usuarios expertos. A su vez, éstos últimos deben elaborar un informe que los Directores de los Usuarios analizan y evalúan para determinar la aceptación o rechazo del sistema.

4.6.4.1 Preparación de las pruebas de aceptación (IAS 6.1)

Se analizan los criterios de aceptación establecidos por el usuario y recogidos en las verificaciones del plan de pruebas, por si fuera necesario incorporar algún caso de prueba adicional.

Se comunica el plan de pruebas de aceptación una vez actualizado, a los usuarios implicados según los distintos perfiles identificados en la tarea Especificación del equipo de implantación (IAS 1.2).

4.6.4.1.1 Plan de pruebas

Una vez terminada la preparación del entorno donde se realizarán las pruebas de aceptación, se revisará el plan de pruebas y se comunicará a los usuarios que participarán, junto con la invitación a la sesión de pruebas.

Debido a que las pruebas de aceptación se realizarán siguiendo un esquema de pruebas libres, no es necesario alterar el plan de pruebas original.

4.6.4.2 Realización de las pruebas de aceptación (IAS 6.2)

Se llevan a cabo las pruebas de aceptación final del sistema para asegurar que todos los componentes responden a los criterios de aceptación especificados.

Se registra la realización de las pruebas, incluyendo un informe que recoja la desviación de los requisitos establecidos y los problemas que quedan sin resolver.

4.6.4.2.1 Resultado de las pruebas de aceptación

Las pruebas de aceptación fueron realizadas sin encontrarse problemas que impidan un correcto funcionamiento y uso del sistema.

4.6.4.3 Evaluación del resultado de las pruebas de aceptación (IAS 6.3)

Se evalúan los resultados de las pruebas, analizando las incidencias recibidas y comprobando que se han llevado a cabo todos los casos de pruebas establecidos en el plan de pruebas. Dicha evaluación consiste en:

- comparar los resultados obtenidos con los esperados;
- identificar el origen de cada problema para poder remitirlo a quién proceda y determinar qué acciones o medidas correctoras es preciso llevar a cabo para resolverlo de forma satisfactoria; e
- indicar qué pruebas se debe volver a realizar, o si será necesario contemplar nuevos casos de prueba.

Una vez realizadas las medidas correctoras necesarias, y comprobado que su comportamiento es adecuado, se documenta el resultado global de la evaluación de las pruebas de aceptación que incluye la aprobación del sistema por parte del usuario final.

4.6.4.3.1 Evaluación del Resultado de las Pruebas de Aceptación

Luego de analizar los resultados de las pruebas de aceptación, los líderes de proyecto y usuarios involucrados en este desarrollo dieron la aceptación formal del sistema.

5 CONCLUSIONES

5.1 *Sobre la metodología*

La metodología MÉTRICA es una parte integral y fundamental de los contenidos de la Maestría. A lo largo de las diferentes unidades de estudio, se tratan los procesos que la componen y las técnicas que son utilizadas en la ejecución de estos procesos. El estudio de estas unidades permite al alumno conocer en detalle una metodología de desarrollo basada en normas internacionales⁷ que es aplicable a cualquier tipo de proyecto de desarrollo y mantenimiento de sistemas de información. Asimismo, permite entender las ventajas de utilizar una metodología en estos proyectos: resultados predecibles, de calidad y capacidad de mejora continua.

Sin embargo, solo la ejecución de un proyecto concreto utilizando esta metodología da la experiencia necesaria para afianzar definitivamente los conocimientos antes adquiridos y estar completamente listos para promover el uso de esta metodología con argumentos no solo teóricos sino también prácticos.

5.2 *Sobre el desarrollo realizado*

El sistema desarrollado, aún en una fase temprana de integración con otras herramientas, ha permitido a los equipos de trabajo tener más visibilidad y control sobre las actividades que debían realizar para asegurarse el cumplimiento de los objetivos de sus proyectos. Los líderes de proyecto han podido tener más claridad sobre la totalidad de actividades que están en proceso en un determinado punto del tiempo y más claridad sobre la real carga de trabajo de cada uno de los miembros de sus equipos. Asimismo, ha facilitado la comunicación entre los líderes y los miembros de los equipos y ha facilitado que éstos realicen una autogestión mucho más efectiva de sus actividades. Si bien el proyecto está en una fase de piloto, es esperable que continúe evolucionando y sea adaptado a otros proyectos de la organización, siempre manteniendo el nivel de disrupción en los proyectos y en la organización misma al mínimo posible.

Es importante aclarar que el sistema en sí es de diseño y construcción sumamente simple. Esto es así debido a que esas son características fundamentales para que sea rápidamente

⁷ Normas ISO/IEC 12207, [ISO, 1995] e ISO/IEC 15504 [ISO, 1998].

aceptado por los usuarios y por la organización. El haber elegido un diseño más complejo, seguramente, habría demorado la construcción del sistema y la adopción por parte de la comunidad de usuarios. A lo largo del presente trabajo se pudo percibir como los usuarios mostraban una predisposición mucho mayor al ver que el sistema era realmente simple en diseño y características de uso y que solucionaba un problema muy puntual que la misma comunidad de usuarios, en más de una ocasión, había identificado.

6 FUTURAS LÍNEAS DE INVESTIGACIÓN

En base a la experiencia adquirida durante el trabajo y en base a todos aquellos aspectos del sistema que tuvieron que quedar de lado para llegar a una primera versión del mismo, se sugiere continuar investigando de acuerdo a los siguientes puntos:

- Desarrollar nuevos requerimientos. Uno de los puntos más importantes al momento de evaluar la utilidad de este sistema es su capacidad para conectarse e importar datos de otras herramientas. En ese sentido, sería muy importante desarrollar nuevos módulos que importen datos de otros sistemas, en adición a los dos ya desarrollados. Asimismo, en una etapa posterior, sería muy importante el dotar al sistema de la habilidad de capturar información de progreso provista por los usuarios y actualizar los sistemas fuente. Finalmente, la adición de habilidades de generación de reportes sería un complemento ideal para que los líderes de proyecto puedan evaluar el desempeño de sus equipos de trabajo.
- Convertir el sistema a tecnologías de código abierto y multiplataforma. Dado que el sistema fue desarrollado con restricciones de carácter técnico en cuanto al ambiente donde se implantó, no se consideró el uso de tecnologías más modernas, de código abierto y multiplataforma. Esto podría ser un valor agregado muy importante ya que permitiría que otras organizaciones hagan uso de la herramienta con costos muy bajos de implantación.
- Estudiar el impacto real de la implementación a gran escala y a largo plazo de este sistema. En el momento de completar este trabajo el sistema está en una etapa de piloto en la organización donde se implementó. Se sugiere continuar con una evaluación continua del desempeño de la misma y de los equipos de trabajo que la utilizan para identificar de la manera más clara posible las reales ventajas de su uso y, al mismo tiempo, las necesidades de mejoras más importantes. Esto debería ser la guía principal del futuro uso y desarrollo del sistema.
- Ahondar en las nuevas tecnologías⁸, tendencias⁹ y últimos estudios sobre integración de herramientas. Esto podría facilitar el trabajo o desarrollo de nuevas versiones de la

⁸ Ver tecnologías Web 2.0 en [Wikipedia 1] y ahondar en las técnicas de *mashup* en [Wikipedia 2].

⁹ Ver tendencias sobre el uso de nuevas tecnologías web en [PMI, 2008].

herramienta, como así también indicar un reemplazo de la misma por otras como más y mejores capacidades o con mayor adhesión a estándares de la industria.

7 BIBLIOGRAFÍA

- [Alshawi et al., 2003] Alshawi, Mustafa; Ingirige, Bingunath: “Web-enabled project management: an emerging paradigm in construction”, *Automation in Construction* 12, 2003, pág. 349-364. (<http://www.ereviewonline.de/press/news/CaseStudyCollaboration.pdf>)
- [Booch et al., 2005] Booch, Grady; Rumbaugh, James; Jacobson Ivar: *The Unified Modeling Language User Guide*, Addison-Wesley, 1998. (ISBN 9780321267979)
- [W3C, 1999] W3C, *HTML 4.01 Specification*, W3C Recommendation, 1999. (<http://www.w3.org/TR/html4>)
- [ISO, 1995] ISO, *ISO/IEC 12207 Systems and software engineering - Software life cycle processes*, ISO Standards, 1995.
- [ISO, 1998] ISO, *ISO/IEC 15504 SPICE (Software Process Improvement And Assurance Standards Capability dEtermination)*, ISO Standards, 1998.
- [Jalloul, 2004] Jalloul, Ghinwa: *UML by Example*, Cambridge University Press, 2004. (ISBN 9780521008815)
- [CSI, 1998] Consejo Superior de Informática: *EUROMÉTODO Versión 1*, Ministerio para las Administraciones Públicas de España. (<http://www.csae.map.es/csi/pg5e40.htm>)
- [CSI, 2005] Consejo Superior de Informática: *MÉTRICA Versión 3: Metodología de Planificación, Desarrollo y Mantenimiento de sistemas de información*, Ministerio para las Administraciones Públicas de España. (<http://www.csi.map.es/csi/metrica3/index.html>)
- [Mühlbauer, 2007] Mühlbauer, Rudolf: “Comparison of Project Management Software”, 2007.
- [PMI, 2008] PMI, “Wasted Web?”, *PM Network*, Volume 22, Number 11, noviembre, 2008, pág. 16-17. (ISSN 10408754)
- [Probasco, 2001] Probasco, Leslee: “What Makes a Good Use-Case Name?”, *the Rational edge*, 2001.

(<http://download.boulder.ibm.com/ibmdl/pub/software/dw/rationaledge/mar01/WhatMakesaGoodUseCaseNameMar01.pdf>)

[Robbins et al., 2000] Robbins, Jason E; Redmiles, David F.: “Cognitive Support, UML Adherence, and XMI Interchange in Argo/UML”, *Information and Software Technology*, 2000. (<http://www.ics.uci.edu/~redmiles/publications/J005-RR99.pdf>)

[Rumbaugh et al., 1999] Rumbaugh, James; Jacobson, Ivar; Booch, Grady: *The Unified Modeling Language Reference Manual*, Addison-Wesley, 1999. (9780201309980)

[Serrano et al., 2005] Serrano, Nicolás; Ciordia, Ismael: “Bugzilla, ITracker, and Other Bug Trackers”, *IEEE Software*, 2005, pág. 11-13.

[Thomas et al. 1992] Thomas, Ian; Nejme, Brian A.: “Definitions of Tool Integration for Environments”, *IEEE Software*, 1992, pág. 29-35.
(<http://www.ics.uci.edu/~taylor/ics228/thomas.pdf>)

[Microsoft 1] Microsoft, *VBScript Coding Conventions*, Microsoft Developer Network.
([http://msdn.microsoft.com/en-us/library/ektke1b0\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/ektke1b0(VS.85).aspx))

[Wicks, 2005] Wicks, Michael N.: “Tool Integration in Software Engineering Environments”, Heriot-Watt University, 2005.
(<http://www.macs.hw.ac.uk/~mnw1/pdf/esecfse2005/DocSymp.pdf>)

[Wicks, 2006] Wicks, Michael N.: “Tool Integration within Software Engineering Environments: An Annotated Bibliography”, Technical Report HW-MACS-TR-0041, Heriot-Watt University, 2006. (<http://www.macs.hw.ac.uk:8080/techreps/docs/files/HW-MACS-TR-0041.pdf>)

[Wikipedia 1] Wikipedia, *Web 2.0*. (http://es.wikipedia.org/wiki/Web_2.0)

[Wikipedia 2] Wikipedia, *Mashup (web application hybrid)*.
([http://en.wikipedia.org/wiki/Mashup_\(web_application_hybrid\)](http://en.wikipedia.org/wiki/Mashup_(web_application_hybrid)))

8 ANEXOS

8.1 Plantilla de código ASP

La siguiente plantilla de código ASP es la que deberá ser utilizada para la creación de todas las páginas HTML/ASP del sistema:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<title></title>
<link rel="stylesheet" type="text/css" href="dashboard.css" />
</head>
<body>
  <!--#include file="dashboard_srv.asp"-->
  <!--HTML code here-->
  <script type="text/vbscript" src="dashboard.vbs"></script>
</body>
</html>
```

8.2 Estructuras XSD

La siguiente estructura XSD es la que deberán respetar los XML creados para contener las actividades importadas de fuentes externas al sistema:

```
<?xml version="1.0" encoding="UTF-8" ?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="activities">
    <xs:complexType>
      <xs:attribute name="project" type="xs:string" use="required" />
      <xs:attribute name="source" type="xs:string" use="required" />
      <xs:attribute name="type" use="required" >
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="PLAN" />
            <xs:enumeration value="PREV" />
            <xs:enumeration value="CORR" />
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name="timestamp" type="xs:string" use="required" />
      <xs:sequence>
        <xs:element ref="activity" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:element name="activity">
```

```

<xs:complexType>
  <xs:attribute name="id" type="xs:string" use="required" />
  <xs:attribute name="desc" type="xs:string" use="required" />
  <xs:attribute name="start" type="xs:date" use="optional" />
  <xs:attribute name="end" type="xs:date" use="optional" />
  <xs:attribute name="prior" use="optional">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="VH" />
        <xs:enumeration value="HI" />
        <xs:enumeration value="ME" />
        <xs:enumeration value="LO" />
        <xs:enumeration value="VL" />
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="stat" type="xs:string" use="required" />
  <xs:sequence>
    <xs:element ref="resource" />
  </xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="resource">
  <xs:complexType>
    <xs:attribute name="name" type="xs:string" use="required" />
  </xs:complexType>
</xs:element>
</xs:schema>

```

La siguiente estructura XSD es la que deberán respetar los XML creados para contener la configuración de los proyectos del sistema:

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="configuration">
    <xs:complexType>
      <xs:attribute name="project" type="xs:string" use="required" />
      <xs:sequence>
        <xs:element ref="source" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:element name="source">
    <xs:complexType>
      <xs:attribute name="name" type="xs:string" use="required" />
      <xs:attribute name="type" use="required">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="MSProject" />
            <xs:enumeration value="MSExcel" />
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name="URL" type="xs:string" use="required" />
      <xs:sequence>

```

```

        <xs:element ref="activities" />
        <xs:element ref="users" />
    </xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="activities">
    <xs:complexType>
        <xs:attribute name="type" use="required">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:enumeration value="PLAN" />
                    <xs:enumeration value="PREV" />
                    <xs:enumeration value="CORR" />
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="dateFormat" type="xs:string" use="optional" />
        <xs:attribute name="sheet" type="xs:string" use="optional" />
        <xs:attribute name="initRow" type="xs:string" use="optional" />
        <xs:sequence>
            <xs:element ref="data" />
        </xs:sequence>
    </xs:complexType>
</xs:element>

<xs:element name="data">
    <xs:complexType>
        <xs:attribute name="desc" type="xs:string" use="required" />
        <xs:attribute name="prop" type="xs:string" use="optional" />
        <xs:attribute name="column" type="xs:string" use="optional" />
        <xs:attribute name="isDate" use="optional">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:enumeration value="True" />
                    <xs:enumeration value="False" />
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:choice>
            <xs:element ref="filters" />
            <xs:element ref="traslationKeys" />
        </xs:choice>
    </xs:complexType>
</xs:element>

<xs:element name="filters">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="allow" />
        </xs:sequence>
    </xs:complexType>
</xs:element>

<xs:element name="allow">
    <xs:complexType>
        <xs:attribute name="val" type="xs:string" use="required" />
    </xs:complexType>
</xs:element>

<xs:element name="traslationKeys">

```

```

<xs:complexType>
  <xs:sequence>
    <xs:element ref="value" />
  </xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="value">
  <xs:complexType>
    <xs:attribute name="to" type="xs:string" use="required" />
    <xs:attribute name="from" type="xs:string" use="required" />
  </xs:complexType>
</xs:element>

<xs:element name="users">
  <xs:complexType>
    <xs:attribute name="separator" type="xs:string" use="optional" />
    <xs:attribute name="column" type="xs:string" use="optional" />
    <xs:sequence>
      <xs:element ref="value" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

8.3 Captura de pantallas

En la Figura 42 se muestra una captura de pantalla principal del sistema funcionando en el entorno de pruebas. Por otro lado, en la Figura 43 se muestra una captura de la pantalla de configuración de proyectos del sistema, también ejecutándose en el entorno de pruebas.

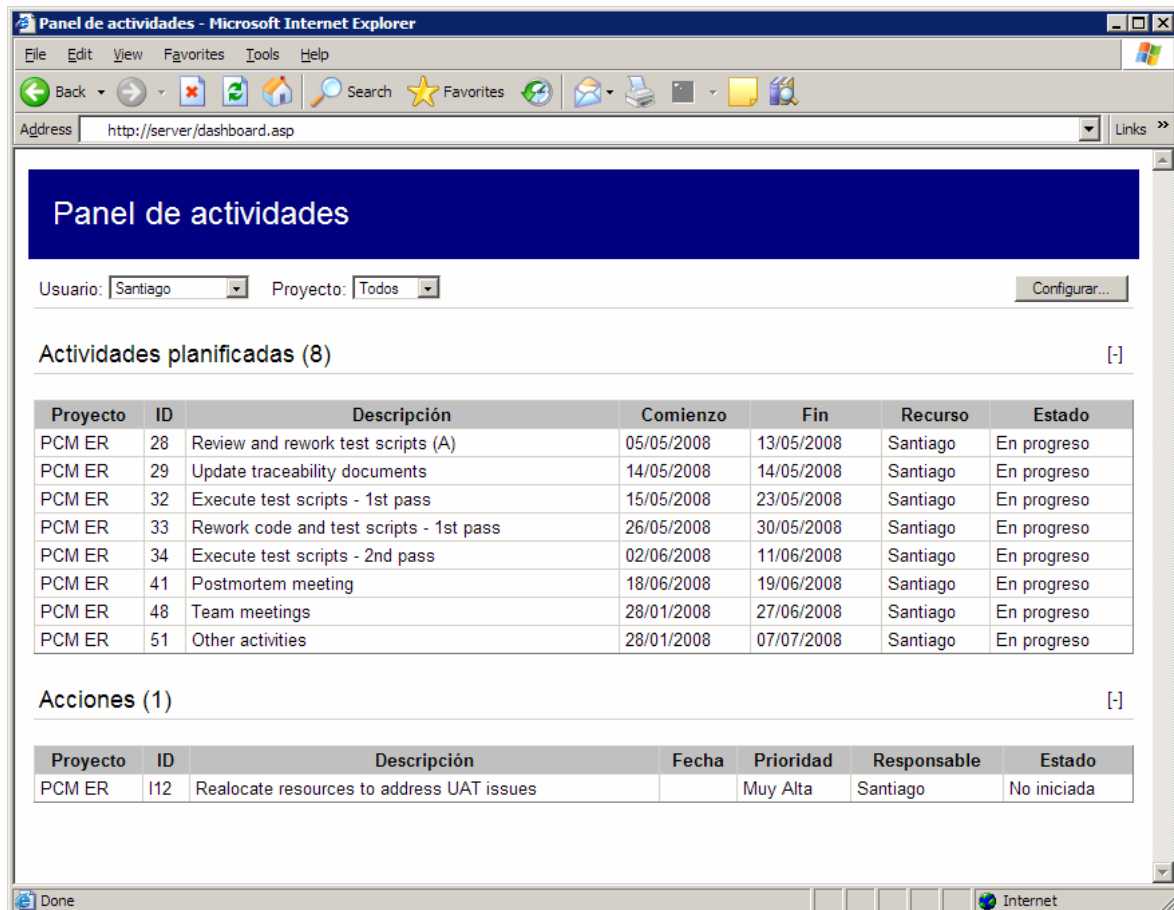


Figura 42: Pantalla principal del sistema.

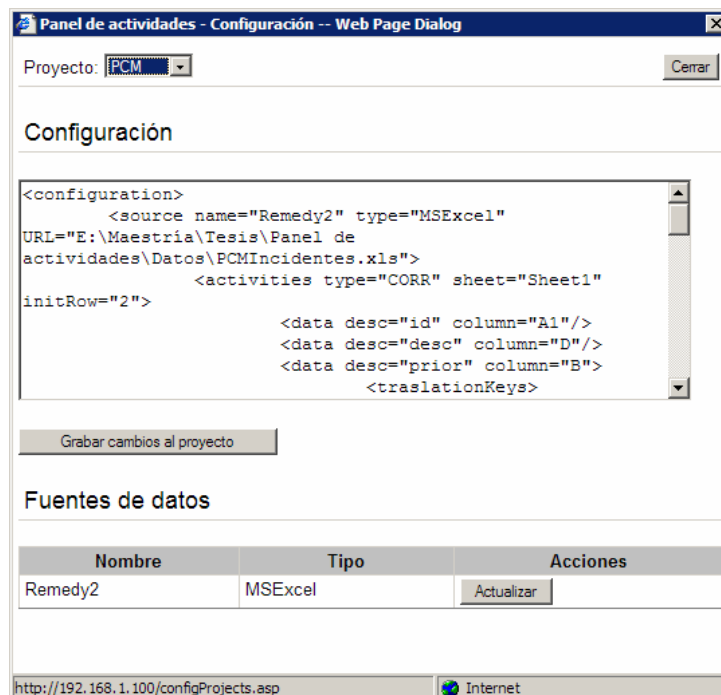


Figura 43: Pantalla de configuración del sistema.

8.4 Registro de ejecución de pruebas de sistema

En la Tabla 41 se muestra el registro de la ejecución de pruebas del sistema, realizadas en el ambiente de pruebas. En una primera pasada de pruebas se ejecutaron todos los casos de prueba definidos y se encontraron cuatro errores. Al finalizar el ciclo, se procedió a la corrección de los errores y, debido a que la criticidad de los mismos no fue determinada como alta, sobre una nueva versión de código se ejecutaron nuevamente solo los casos de prueba que habían fallado en la primera pasada. Finalmente, todos los casos de prueba resultaron exitosos.

Versión de código	Fecha	Caso de prueba	Paso	Resultado	Comentarios
VER.1	08/10/2008	CP.1	1	Pasó	[ERR.1]
VER.1	08/10/2008	CP.2	1	Pasó	
VER.1	08/10/2008	CP.2	2	Pasó	
VER.1	08/10/2008	CP.2	3	Pasó	
VER.1	08/10/2008	CP.2	4	Pasó	
VER.1	08/10/2008	CP.2	5	Pasó	
VER.1	08/10/2008	CP.2	6	Falló	[ERR.2]
VER.1	08/10/2008	CP.2	7	Pasó	
VER.1	08/10/2008	CP.3	1	Falló	[ERR.3]
VER.1	08/10/2008	CP.3	2	Pasó	
VER.1	08/10/2008	CP.3	3	Pasó	
VER.1	08/10/2008	CP.3	4	Falló	[ERR.4]
VER.1	08/10/2008	CP.3	5	Pasó	
VER.1	08/10/2008	CP.4	1	Pasó	
VER.1	08/10/2008	CP.4	2	Pasó	
VER.1	08/10/2008	CP.4	3	Pasó	
VER.1	08/10/2008	CP.4	4	Pasó	
VER.1	08/10/2008	CP.5	1	Pasó	
VER.1	08/10/2008	CP.5	2	Pasó	
VER.1	08/10/2008	CP.5	3	Pasó	
VER.1	08/10/2008	CP.6	1	Pasó	
VER.1	08/10/2008	CP.6	2	Pasó	
VER.1	08/10/2008	CP.6	3	Pasó	
VER.1	08/10/2008	CP.6	4	Pasó	
VER.1	08/10/2008	CP.6	5	Pasó	
VER.1	08/10/2008	CP.7	1	Pasó	
VER.1	08/10/2008	CP.7	2	Pasó	
VER.1	08/10/2008	CP.7	3	Pasó	
VER.1	08/10/2008	CP.7	4	Pasó	
VER.2	02/11/2008	CP.1	1	Pasó	
VER.2	02/11/2008	CP.2	6	Pasó	
VER.2	02/11/2008	CP.3	1	Pasó	
VER.2	02/11/2008	CP.3	4	Pasó	

Tabla 41: Registro de ejecución de pruebas de sistema. El registro contiene los resultados de los dos ciclos de prueba ejecutados sobre el sistema.

8.5 Registro de errores

La Tabla 42 contiene el registro de todos los errores encontrados durante las pruebas del sistema. En la tabla se encuentra la información referente al error en sí: identificador, descripción, fecha en la cual fue detectado y versión de código que se estaba probando al momento de ser detectado; también se encuentra información referente al estado y resolución del error: estado de resolución, fecha de corrección y versión de código en la cual el error fue verificado como corregido.

Id.	Descripción	Fecha de detección	Versión de código	Estado	Fecha de corrección	Versión de código
ERR.1	El menú de proyectos trae múltiple instancias del mismo proyecto.	08/10/2008	VER.1	Corregido	02/11/2008	VER.2
ERR.2	Los controles para ocultar y mostrar las tablas no funcionan bien cuando se los activa en forma intercalada.	08/10/2008	VER.1	Corregido	11/10/2008	VER.2
ERR.3	La configuración mostrada no coincide con la que el usuario debería ver.	08/10/2008	VER.1	Corregido	09/10/2008	VER.2
ERR.4	Al presionar el botón “Cerrar”, la ventana no se cierra y el sistema muestra un error.	08/10/2008	VER.1	Corregido	11/10/2008	VER.2

Tabla 42: Registro de errores encontrados durante las pruebas de sistema.

8.6 Manual de usuario

8.6.1 Introducción

El Panel de Actividades es una herramienta que permite consolidar información desde diferentes fuentes acerca de las actividades que están asignadas a los miembros de los equipos de trabajo. A través de esta consolidación, la herramienta permite:

- Consolidar información dispersa en diferentes documentos o sistemas, como ser: cronograma de actividades, registros de riesgos, lista de acciones, registro de incidentes, registro de defectos, registro de cambios, etc.
- Acceder a la lista completa de actividades asignadas, tanto propias como de todos los demás miembros del equipo de trabajo, a través de un único punto de entrada con interfaz web.

- Facilitar la priorización de actividades, tanto propias como de todos los miembros del equipo de trabajo.
- Facilitar la evaluación y la adecuada distribución de la carga de trabajo entre los miembros del equipo de trabajo.
- Reducir la posibilidad de que, por olvido u omisión, queden actividades sin completar y esto ponga en riesgo el cumplimiento de los objetivos de los proyectos.

La herramienta cuenta con dos pantallas: una para listar las actividades, permitiendo aplicar filtros por proyecto y recurso y permitiendo ordenar las actividades de acuerdo a sus características; y otra para la configuración del sistema, donde se especifica qué y cómo se importarán las actividades desde otros sistemas o repositorios.

La herramienta está diseñada para reconocer a dos tipos de usuarios: los miembros de equipo, quienes tienen acceso solamente a la pantalla de actividades y pueden ver solo las actividades asignadas a miembros de sus equipos de trabajo; y los líderes de proyecto, quienes además tienen acceso a la pantalla de configuración de proyectos, donde definen qué información, cómo y de dónde se importa a la herramienta.

8.6.2 Trabajar con actividades

La pantalla de actividades muestra todas las actividades que tiene permisos para ver. Las actividades se agrupan en dos tablas: la tabla de actividades planificadas y la tabla de acciones. Éstas últimas no están planificadas e irán identificándose a lo largo del desarrollo del proyecto. Puede filtrar las actividades por proyecto y por recurso y también puede reordenar las listas de las tablas para facilitar la administración de las actividades asignadas.

8.6.2.1 Identificar información de actividades planificadas

En la tabla de actividades planificadas, las cuales provendrán del cronograma de actividades del proyecto, tiene la siguiente información:

- **Proyecto:** Es el proyecto al cual pertenece la actividad.
- **ID:** Es el identificador de la actividad. Su valor es de tipo alfanumérico.
- **Descripción:** Es la descripción o título de la actividad.

- **Comienzo:** Es la fecha en la cual la actividad deberá ser iniciada según el plan.
- **Fin:** Es la fecha en la cual la actividad deberá ser finalizada según el plan.
- **Recurso:** Es el recurso asignado a la actividad. En caso de haber más de un recurso, se verán tantas filas en la tabla como recursos asignados tenga la actividad original.
- **Estado:** Es el estado de la actividad. Este puede ser No iniciada o En progreso. Las actividades completadas no se muestran en la tabla.

8.6.2.2 Identificar información de acciones

En la tabla de acciones, las cuales provendrán de las listas de acciones, registros de incidentes, registros de defectos del proyecto, etc., tiene la siguiente información:

- **Proyecto:** Es el proyecto al cual pertenece la acción.
- **ID:** Es el identificador de la acción. Su valor es de tipo alfanumérico.
- **Descripción:** Es la descripción o título de la acción.
- **Fecha:** Es la fecha comprometida de cierre de la acción.
- **Prioridad:** Es la prioridad de la acción. La prioridad podrá ser Muy Alta, Alta, Media, Baja o Muy Baja.
- **Responsable:** Es el recurso responsable de realizar la acción. En caso de haber más de un responsable, se verán tantas filas en la tabla como responsables asignados tenga la acción original.
- **Estado:** Es el estado de la acción. Este puede ser No iniciada o En progreso. Las acciones completadas no se muestran en la tabla.

8.6.2.3 Filtrar actividades por usuario

Tiene la posibilidad de ver todas las actividades de todos los demás miembros de equipo de cada uno de los proyectos a los cuales esté asignado o de filtrarlas por usuario:

- Seleccione la opción Todos del menú de usuarios para que las tablas de actividades planificadas y de acciones muestren todas las actividades de todos los usuarios de todos los proyectos a los cuales esté asignado.
- Seleccione la opción correspondiente a un usuario específico para que las tablas de actividades planificadas y de acciones muestren solo las actividades asignadas al usuario seleccionado.

8.6.2.4 Filtrar actividades por proyecto

Si está asignado a más de un proyecto, tiene la posibilidad de ver todas las actividades o de filtrarlas por cada uno de los proyectos:

- Seleccione la opción Todos del menú de proyectos para que las tablas de actividades planificadas y de acciones muestren todas las actividades de todos los proyectos a los cuales esté asignado.
- Seleccione la opción correspondiente a un proyecto específico para que las tablas de actividades planificadas y de acciones muestren solo las actividades del proyecto seleccionado.

8.6.2.5 Reordenar información de las tablas

Puede reordenar la información de la tabla de actividades planificadas y de la tabla de acciones:

- Haga clic en el encabezado de una columna para que la tabla se reordene inmediatamente de menor a mayor de acuerdo a los valores de esa columna.
- Haga clic nuevamente en el encabezado de la misma columna para que la tabla se reordene en el sentido inverso, por ejemplo, de mayor a menor.
- Haga clic en el encabezado de otra columna para reordenar la tabla de acuerdo a los valores de otra columna.

8.6.2.6 Ocultar la información de las tablas

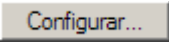
En el caso de que las tablas sean muy extensas y desee tener una mejor utilización del espacio disponible en la ventana, podrá ocultar una o ambas tablas.

- Haga clic en el símbolo [-] sobre la derecha de la tabla que quiera ocultar para que ésta se oculte. Solo quedará visible el título de la tabla y la cantidad de actividades o acciones que contiene.
- Haga clic en el símbolo [+] a la derecha del título de la tabla que quiera volver a mostrar para que ésta se vuelva a mostrarse completamente.

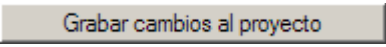
8.6.3 Configurar proyectos

Solo los líderes de proyecto tienen acceso a la ventana de configuración de proyectos. En esta ventana se edita la configuración de los proyectos y se dispara el proceso de importación de actividades desde fuentes externas, como ser otros sistemas o repositorios.

8.6.3.1 Abrir la ventana de configuración

- Haga clic en el botón  que se encuentra en la esquina superior derecha en la ventana principal para acceder a la ventana de configuración. Solo será visible si tiene permisos de líder de proyecto.

8.6.3.2 Configurar un proyecto

1. Seleccione el proyecto que desea configurar del menú de proyectos, en la esquina superior izquierda de la ventana de configuración. Solo los proyectos en los cuales tiene permisos de líder de proyecto estarán listados en el menú. Al cambiar la selección, la configuración del proyecto seleccionado y sus fuentes de datos se mostrarán en la ventana.
2. Edite la configuración en el área editable del centro de la ventana.
3. Haga clic en el botón  para guardar los cambios. Este botón permanecerá habilitado solo mientras la configuración del proyecto cumpla con la estructura de un XML bien formado. Si hubiera habido cambios en las fuentes de datos, estos cambios se verán reflejados en la tabla inferior de la ventana.

8.6.3.2.1 Notas

- Si se edita la configuración de un proyecto y se selecciona otro proyecto del menú sin haber guardado los cambios, éstos se perderán.
- Si la configuración hace referencia a usuarios que no están dados de alta en el sistema, la configuración será almacenada y se notificará la lista de usuarios en esa situación.
- Si una fuente de datos fuera modificada o eliminada de la configuración, todas las actividades relacionadas a esa fuente serán eliminadas del sistema.
- La creación inicial del proyecto y la asignación del usuario del sistema que tiene permisos de líder de ese proyecto debe hacerse manualmente a través del administrador del sistema.

8.6.3.3 Opciones de configuración

La configuración de los proyectos debe estar especificada en formato XML. Los nodos y atributos permitidos son:

- configuration: nombre del nodo raíz. Este nodo contendrá todas las fuentes de datos del proyecto en nodos source.
- source: nombre del nodo que contiene toda la información relativa a una fuente de datos de un proyecto. Tiene los siguientes atributos y nodos:
 - name: atributo que especifica el nombre de la fuente de datos. El nombre de la fuente debe ser único dentro del proyecto para poder identificarla correctamente en el sistema.
 - type: atributo que define el tipo de fuente de datos. El sistema soporta actualmente los tipos MSProject y MExcel.
 - URL: atributo que especifica la ubicación de la fuente de datos.
 - activities: nodo que contiene la definición de los datos que deberán ser importados al sistema.

- users: nodo que contiene la definición de los usuarios que deberán ser importados al sistema.
- activities: nombre del nodo que contiene la especificación de los datos de la fuente de datos del proyecto. La información contenida en este nodo permitirá al sistema interpretar adecuadamente el contenido de la fuente de datos e importar los mismos convirtiéndolos al formato propio del sistema. Tiene los siguientes atributos y nodos:
 - type: atributo que define el tipo de actividades contenido en la fuente de datos. Si hubiera más de un tipo de actividad en la misma fuente de datos, se deberán crear fuentes de datos adicionales en la configuración del proyecto. El sistema soporta actualmente los tipos PLAN, para actividades planificadas, PREV, para acciones preventivas y CORR, para acciones correctivas. Estos últimos dos tipos se mostrarán juntos en la tabla de acciones de la pantalla principal del sistema.
 - dateFormat: atributo que se utiliza exclusivamente para importar correctamente las fechas de las actividades desde MS Project. Deberá especificar el formato en el cual las fechas son presentadas por MS Project. Si por ejemplo las fechas se muestran como “Mie 22-07-08”, el formato deberá ser “----dd-mm-yy”. El sistema solo reconocerá las secuencias “dd”, “mm” e “yy” para interpretar correctamente la fecha provista por MS Project.
 - sheet: atributo que se utiliza exclusivamente para importar actividades desde planillas MS Excel y sirve para especificar la hoja de la planilla que contiene la información.
 - initRow: atributo que se utiliza exclusivamente para importar actividades desde planillas MS Excel y sirve para especificar el número de la primera fila que contiene datos válidos.
 - data: nodos que especifican el detalle de la información que debe ser extraída de la fuente. Deberá existir un nodo data por cada atributo a extraer de la fuente.

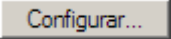
- data: nombre del nodo que contiene la información específica de un atributo de actividad a ser importada al sistema. Los nodos data tienen los siguientes atributos y nodos asociados:
 - desc: nombre del atributo que especifica qué tipo de dato se está especificando. El sistema reconoce solo ciertos atributos:
 - id: especifica que el dato es el identificador de la actividad.
 - desc: especifica que el dato es la descripción de la actividad.
 - stat: especifica que el dato es el estado de la actividad. Por lo general este dato se utilizará para filtrar los valores que indiquen actividades abiertas utilizando nodos `filters` y se traducirá el contenido con nodos `translationKeys`. Los valores de traducción aceptados son “No iniciada” o “En progreso”.
 - start: especifica que el dato es la fecha de comienzo de la actividad. Es solo aplicable a actividades planificadas.
 - end: especifica que el dato es la fecha de fin de la actividad para actividades planificadas o la fecha de compromiso de cierre de acciones no planificadas.
 - prior: especifica que el dato es la prioridad de la actividad. Es solo aplicable a acciones no planificadas. Por lo general será utilizado en conjunto con nodos `translationKeys`. Los valores de traducción aceptados son “VH”, “HI”, “ME”, “LO” y “VL”.
 - prop: nombre del atributo que especifica el número de propiedad que contiene el dato referenciado por el atributo desc del nodo data. Es solo aplicable a datos de fuentes MS Project.
 - column: nombre del atributo que especifica el nombre de la columna que contiene el dato referenciado por el atributo desc del nodo data. Es solo aplicable a datos de fuentes MS Excel.

- `isDate`: nombre del atributo que especifica que el dato es una fecha. Es solo aplicable a datos de fuentes MS Project y se utiliza en conjunto con el atributo `dateFormat` del nodo `activities`.
 - `filters`: nodo que contiene todos los valores permitidos para un determinado dato. Cada valor permitido deberá contenerse en un nodo `allow`, especificando el valor en el atributo `val`. Los valores no encontrados en esta lista provocarán que la actividad sea ignorada en el proceso de importación.
 - `translationKeys`: nodo que contiene las reglas para traducir valores de la fuente de origen a valores reconocidos por el sistema. Este nodo contendrá nodos `value` con atributos `from` y `to` que especificarán la regla de traducción. De no encontrarse una regla válida, los nodos no serán traducidos y su valor se importará sin modificar.
- `users`: nombre del nodo que contiene la información específica de los recursos asignados al proyecto. Los nodos `users` tienen los siguientes atributos y nodos asociados:
 - `separator`: en el caso de que exista la posibilidad de que varios recursos estén asignados a la misma actividad, este atributo especifica cuál es el carácter que se utilizará para separar el nombre de los diferentes recursos.
 - `column`: nombre del atributo que especifica el nombre de la columna que contiene el o los recursos asignados a cada actividad. Es solo aplicable a datos de fuentes MS Excel.
 - `value`: nodo que especifica el nombre del recurso tal como se especifica en la fuente y que permite traducirlo al nombre con el que el recurso está registrado en el sistema. Para ello, cuenta con los atributos `from` y `to`.

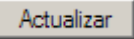
8.6.4 Importar actividades

Solo los líderes de proyecto tienen acceso a la ventana de configuración de proyectos, desde donde se puede forzar la importación de actividades desde fuentes externas, como ser otros sistemas o repositorios.

8.6.4.1 Abrir la ventana de configuración

- Haga clic en el botón  que se encuentra en la esquina superior derecha en la ventana principal para acceder a la ventana de configuración. Solo será visible si tiene permisos de líder de proyecto.

8.6.4.2 Importar actividades manualmente

1. Seleccione el proyecto que desea configurar del menú de proyectos, en la esquina superior izquierda de la ventana de configuración. Solo los proyectos en los cuales tiene permisos de líder de proyecto estarán listados en el menú. Al cambiar la selección, la configuración del proyecto seleccionado y sus fuentes de datos se mostrarán en la ventana.
2. Haga clic en el botón  de la fuente de datos que desea importar.

8.6.4.2.1 Notas

- Las actividades relacionadas con la fuente de datos que el sistema pudiera contener previamente a la solicitud de actualización son eliminadas y reemplazadas por los nuevos datos.
- Si el sistema no encuentra como traducir un nombre de usuario, un estado o prioridad, la actividad correspondiente no será importada.

8.7 Interfaces de MÉTRICA Versión 3

La estructura de MÉTRICA Versión 3 incluye también un conjunto de interfaces que definen una serie de actividades de tipo organizativo o de soporte al proceso de desarrollo y a los productos, que en el caso de existir en la organización se deberán aplicar para enriquecer o influir en la ejecución de las actividades de los procesos principales de la metodología y que si no existen habrá que realizar para complementar y garantizar el éxito del proyecto desarrollado con MÉTRICA Versión 3.

La aplicación de MÉTRICA Versión 3 proporciona sistemas con calidad y seguridad, no obstante puede ser necesario en función de las características del sistema un refuerzo especial en estos aspectos, refuerzo que se obtendría aplicando la interfaz.

Las interfaces descritas en la metodología son:

- Gestión de Proyectos (GP)
- Seguridad (SEG)
- Aseguramiento de la Calidad (CAL)
- Gestión de la Configuración (GC)

8.7.1 Gestión de proyectos (GP)

La Gestión de Proyectos tiene como finalidad principal la planificación, el seguimiento y control de las actividades y de los recursos humanos y materiales que intervienen en el desarrollo de un Sistema de Información. Como consecuencia de este control es posible conocer en todo momento qué problemas se producen y resolverlos o paliarlos lo más pronto posible, lo cual evitará desviaciones temporales y económicas.

La Interfaz de Gestión de Proyectos de MÉTRICA Versión 3 contempla proyectos de desarrollo de Sistemas de Información en sentido amplio, acorde con EUROMÉTODO¹⁰ se consideran proyectos de desarrollo de nuevos Sistemas de Información y también los proyectos de ampliación y mejora de los ya existentes.

Las actividades de la Interfaz de Gestión de Proyectos son de tres tipos:

- Actividades de Inicio del Proyecto (GPI), que permiten estimar el esfuerzo y establecer la planificación del proyecto.
- Actividades de Seguimiento y Control (GPS), supervisando la realización de las tareas por parte del equipo de proyecto y gestionando las incidencias y cambios en los requisitos que puedan presentarse y afectar a la planificación del proyecto.
- Actividades de Finalización del Proyecto, cierre y registro de la documentación de gestión.

Estas actividades pueden requerir, en función de la complejidad del proyecto, el soporte de herramientas comerciales de gestión de proyectos.

¹⁰ Para más información, referirse a [CSI, 1998].

8.7.1.1 Actividades de inicio del proyecto (GPI)

Las actividades al inicio de un proyecto tienen un doble objetivo: estimar el esfuerzo a realizar para desarrollar el sistema y planificar las actividades de dicho desarrollo. Para ello, tomando como punto de partida la Solución Propuesta en el Estudio de Viabilidad del Sistema (EVS 6), se identifican los elementos a desarrollar, se calcula el esfuerzo a realizar, y se planifican las actividades del proyecto comprendiendo los aspectos de recursos, programación de tareas y establecimiento de un calendario de entregas y recepciones entre el cliente y los proveedores.

8.7.1.1.1 Estimación de esfuerzo (GPI 1)

El objetivo de esta actividad es conocer el tamaño aproximado del sistema a desarrollar, y establecer el coste, la duración y los recursos necesarios para conseguir desarrollarlo.

Es muy difícil calcular con absoluta precisión el esfuerzo requerido para desarrollar cualquier proyecto informático, debido a la gran cantidad de factores que intervienen en su realización, algunos de ellos inciertos o desconocidos. Sin embargo, las técnicas existentes para realizar los cálculos proporcionan un valor aproximado suficiente para el alcance del desarrollo del proyecto. Será siempre útil la experiencia anterior que hubiese, extraída de la realización de proyectos similares en la organización, así como la existencia de una base de datos con información relativa a métricas, en el sentido del término en ingeniería del software.

8.7.1.1.1.1 Catálogo de clases

Del análisis preliminar desarrollado en el anteproyecto del presente trabajo y de los resultados parciales del proceso de Estudio de Viabilidad del Sistema (EVS), se estima que el éste estará compuesto por los siguientes casos de uso:

- Visualizar actividades: caso de complejidad media, debido a la interacción que el usuario podrá realizar con la información presentada.
- Importar actividades: caso de complejidad alta, debido a las necesidades de interpretación y traducción de la información contenida en otras herramientas.

8.7.1.1.1.2 Esfuerzo estimado

Utilizando técnicas de estimación por analogía y juicio de expertos y tomando como base histórica la información de proyectos anteriores desarrollados en la organización donde se va a implantar el sistema, se obtiene la siguiente información:

- Casos de uso de complejidad media: 30 horas de esfuerzo de construcción
- Casos de uso de complejidad alta: 40 horas de esfuerzo de construcción

También, basándose en información histórica, los procesos del ciclo de vida tienen la siguiente distribución aproximada:

- Análisis: 25%
- Diseño: 30%
- Construcción: 35%
- Implantación: 10%

Extrapolando el esfuerzo de construcción en el resto de las fases, el total del esfuerzo del proyecto debería ser de aproximadamente 200 horas.

Debido a que el proyecto es de un tamaño muy chico, donde los desvíos pequeños pueden llegar a ser muy significativos y debido a que el recurso asignado estará trabajando en tiempo parcial, se estima una contingencia adicional de un 50% sobre el esfuerzo total. Por lo tanto, el esfuerzo total del proyecto se estima en 300 horas.

8.7.1.1.2 Planificación (GPI 2)

El objetivo de esta actividad es definir y preparar las condiciones de trabajo, estableciendo recursos, fechas y costes, para lograr los objetivos que se persiguen con el proyecto.

La planificación de un proyecto establece las fechas previstas para la realización del conjunto de actividades que lo componen, teniendo en cuenta que se deben emplear para ello unos recursos que implican unos costes, estimados inicialmente en EVS 5.1, cuyo conjunto forma el presupuesto base para lograr el resultado comprometido con el Cliente o Usuario. Así mismo deberán reflejarse hitos y calendario de entregas de productos al cliente.

8.7.1.1.2.1 Planificación del proyecto

Tomando como base el esfuerzo estimado para el proyecto y la distribución esperada entre los diferentes procesos a ejecutar a lo largo del desarrollo, y considerando una disponibilidad del recurso asignado de aproximadamente 20 horas semanales, en la Tabla 43 se muestra el cronograma de trabajo.

	Semana														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Procesos de desarrollo															
EVS + ASI	75 horas														
DSI					90 horas										
CSI										105 horas					
IAS															30 hs

Tabla 43: Cronograma de trabajo del proyecto.

8.7.2 Gestión de la configuración (GC)

La interfaz de gestión de la configuración consiste en la aplicación de procedimientos administrativos y técnicos durante el desarrollo del sistema de información y su posterior mantenimiento. Su finalidad es identificar, definir, proporcionar información y controlar los cambios en la configuración del sistema, así como las modificaciones y versiones de los mismos. Este proceso permitirá conocer el estado de cada uno de los productos que se hayan definido como elementos de configuración, garantizando que no se realizan cambios incontrolados y que todos los participantes en el desarrollo del sistema disponen de la versión adecuada de los productos que manejan.

La interfaz de gestión de configuración de MÉTRICA Versión 3 permite definir las necesidades de gestión de configuración para cada sistema de información, recogiendo en un plan de gestión de configuración, en el que se especifican actividades de identificación y registro de productos, que se realizan durante todas las actividades de MÉTRICA Versión 3 asociadas al desarrollo y mantenimiento del sistema de información.

Asimismo, permite controlar el sistema como producto global a lo largo de su creación, obtener informes sobre el estado de desarrollo en que se encuentra y reducir el número de errores durante el mismo, lo que se traduce en un aumento de calidad del proceso de desarrollo y de mejora de la productividad en la organización.

La gestión de configuración facilita además el mantenimiento del sistema, aportando información precisa para valorar el impacto de los cambios solicitados y reduciendo el tiempo de implementación de un cambio, tanto evolutivo como correctivo.

8.7.2.1 Establecimiento del plan de gestión de la configuración (GC 2)

El objetivo de esta actividad es definir el Plan de Gestión de Configuración para la Solución propuesta (EVS 6.2) y especificar el entorno tecnológico de soporte a la gestión de configuración.

Una vez establecido el Plan de Gestión de la Configuración, se irán registrando los productos que se obtengan en los procesos de Análisis, Diseño, Construcción, Implantación y Aceptación del Sistema de Información y que se hayan determinado en el plan como productos a incluir en el sistema de gestión de configuración.

8.7.2.1.1 Plan de gestión de la configuración

Debido a la baja disponibilidad de recursos asignados al proyecto y al tamaño relativamente pequeño del desarrollo, la gestión de configuración se realizará de la manera más sencilla posible.

Se controlarán principalmente dos clases de elementos: los documentos de trabajo y el código fuente. Para los documentos, se utilizará la siguiente convención de nombres:

- `<ID_Documento>_<Versión>.<Extensión>`

Donde:

- ID_Documento es el nombre del documento, el cual debe ser único entre todos los elementos de configuración.
- Versión es un número de secuencia. Con cada revisión del documento, este número se incrementará en 1.
- Extensión es el tipo de archivo.

El código fuente se almacenará en un directorio que utilizará la siguiente convención:

- `Código\VER.<Versión>`

Donde:

- Versión es un número de secuencia. Con cada nueva versión del código fuente, este número se incrementará en 1.

En cada proceso del ciclo de vida de desarrollo del sistema se crearán uno o más de los elementos de configuración definidos anteriormente. Una vez terminado cada uno de estos procesos, los elementos de configuración pasarán a formar parte de la línea base de ese proceso. La definición de los procesos, líneas base y elementos que las conforman se pueden ver en la Tabla 44.

Proceso y Línea Base	Elementos de Configuración	ID Documentos
Estudio de Viabilidad del Sistema (EVS)	Viabilidad del Sistema	EVS
	Plan de Gestión	PGS
	Plan de Configuración	PCS
Análisis del Sistema de Información (ASI)	Análisis del Sistema	ASI
Diseño del Sistema de Información (DSI)	Diseño del Sistema	DSI
Construcción del Sistema de Información (CSI)	Construcción del Sistema	CSI
	Código Fuente	-
	Registro de Pruebas	RPS
	Registro de Errores	RES
	Manual de Usuario	MDU
Implantación y Aceptación del Sistema (IAS)	Implementación y Aceptación	IAS

Tabla 44: Lista de líneas base, elementos que las componen e identificadores de esos documentos.