

ANEXOS

Tabla de Contenido

ANEXO A - PUBLICACIONES, PROYECTOS Y DOCENCIA	1
1. PUBLICACIONES	1
2. PROYECTOS	2
3. DOCENCIA	2
ANEXO B- DESCRIPCIÓN DE ELEMENTOS DE PROCESO DE LA PAL-WIKI.....	3
1. ACTIVIDADES	4
1.1 Actividades del proceso Gestión de Historias de Usuario.....	4
1.2 Actividades del proceso Planificación	14
1.3 Actividades del proceso Desarrollo.....	20
1.4 Actividades del proceso Refactoring.....	25
1.5 Actividades del proceso Seguimiento Periódico	28
1.6 Actividades del proceso Integración Automática.....	31
1.7 Actividades del proceso Gestión de defectos	33
1.2 ROLES	41
1.2.1 Coordinador del Equipo	41
1.2.2 Ingeniero de Software	41
1.2.3 Cliente/Usuario.....	41
1.3 ELEMENTOS DE TRABAJO.....	42
1.3.1 Defecto	42
1.3.2 Historia de Usuario.....	45
1.3.3 Requisito de Calidad de Servicio	48
1.3.4 Recurso.....	51
1.3.5 Prueba de Aceptación.....	53
1.3.6 Prueba Unitaria.....	56
1.3.7 Clase	58
1.3.8 Tarea.....	60
1.3.9 Elemento de Integración Automática	63
ANEXO C - MANUAL DE INSTALACIÓN DE MEDIAWIKI	65
1. REQUISITOS PREVIOS.....	65

2.	DESCARGAR Y DESCOMPRIMIR MEDIAWIKI	66
3.	CREAR LA BASE DE DATOS	66
4.	INSTALACIÓN DE MEDIAWIKI	66
5.	CONFIGURACIÓN DE MEDIAWIKI	70
5.1	Ingreso de sólo usuarios registrados	71
5.2	Prevenir la creación de cuentas	71
5.3	Creación de Cuentas de usuarios	71
5.4	Permitir la subida de ficheros	72
5.5	Permitir otras extensiones de ficheros	72
5.6	Cambiar logo	73
6.	CONFIGURACIONES AVANZADAS DE MEDIAWIKI	73
6.1	Semantic MediaWiki	73
6.2	SemanticForms	75
6.3	HeaderTabs	75
6.4	Flash	76
7.	IMPORTACIÓN DE LA PAL-WIKI	76
7.1	Personalizar la barra de navegación	76
7.2	Importación del fichero XML	78
7.3	Subir los ficheros	79
7.4	Cambiar la página principal	79
	ANEXO D - MANUAL DE USUARIO DE LA PAL-WIKI	80
1.	INTERFAZ DE LA PAL-WIKI	80
1.1	La barra de navegación	81
1.2	Pestañas	82
1.3	Enlaces del usuario	82
2.	ADQUISICIÓN DEL CONOCIMIENTO	83
2.1	Creación de formularios	83
2.2	Definición de roles	85
2.3	Definición de procesos	86
2.4	Definición de actividades	88
2.5	Definición de elementos de trabajo	89
3.	ORGANIZACIÓN DEL CONOCIMIENTO	91
3.1	Asignar cada página a categoría	91

3.2 Asignar procesos y actividades a elementos de trabajo.....	92
4. DISTRIBUCIÓN DEL CONOCIMIENTO	93
4.1 Navegación de la PAL-Wiki	93
4.2 Búsqueda de activos en la PAL-Wiki.....	96
5. UTILIZACIÓN DEL CONOCIMIENTO	96
5.1 Instanciar activo de proceso	96
5.2 Publicar instancia de activo de proceso.....	98
5.3 Reuso de activo de proceso	100
5.4 Consultar ayuda de la PAL-Wiki	101
6. PRESERVACIÓN DEL CONOCIMIENTO	102
6.1 Edición de activo de proceso software.....	102
6.2 Retiro de activo de proceso	103
6.3 Restaurar activo de proceso.....	104
6.4 Discutir sobre activo de proceso	106
7. GESTIÓN DE USUARIOS	107
7.1 Crear grupo.....	107
7.2 Crear usuarios.....	108
7.3 Modificar derechos de acceso de los usuarios.....	109
8. GESTIÓN DE CAMBIOS	110
8.1 Consultar historial de versiones de activos de proceso	110
8.2 Comparar versiones de activos de proceso.....	111
8.3 Moderar contenidos.....	112
9. MEDICIÓN DE LA PAL-WIKI	114
9.1. Estadísticas generales.....	114
9.2. Usage Statistics	115
9.3. Google Analytics.....	116

Lista de Figuras

- Figura C-1.** Inicio de la instalación.
- Figura C-2.** Verificación del ambiente de instalación.
- Figura C-3.** Configuración del sitio.
- Figura C-4.** Configuración de notificación por e-mail.
- Figura CB-5.** Configuración de la base de datos.
- Figura C-6.** Instalación exitosa de MediaWiki.
- Figura C-7.** Página principal de la wiki.
- Figura C-8.** Ventana UserLogin.
- Figura C-9.** Creación de cuenta de usuario.
- Figura C-10.** Página de administración de Semantic MediaWiki.
- Figura C-11.** Creación de tablas de Semantic MediaWiki.
- Figura D-1.** Registrarse/Entrar.
- Figura D-2.** Página inicial de la PAL-Wiki.
- Figura D-3.** Registrarse/Entrar.
- Figura D-4.** Pestañas superiores de la PAL-Wiki.
- Figura D-5.** Enlaces del usuario
- Figura D-6.** Página de plantillas.
- Figura D-7.** Crear una plantilla.
- Figura D-8.** Formularios.
- Figura D-9.** Registrarse/Entrar.
- Figura D-10.** Crear un formulario.
- Figura D-11.** Enlace a formulario de definición de roles.
- Figura D-12.** Creación de la página de rol.
- Figura D-13.** Formulario de definición de roles.
- Figura D-14.** Rol definido.
- Figura D-15.** Enlace a formulario de procesos.
- Figura D-16.** Creación de la página de proceso.
- Figura D-17.** Formulario de definición de procesos.

Figura D-18. Proceso definido.

Figura D-19. Enlace a formulario de actividades.

Figura D-20. Creación de la página de actividad.

Figura D-21. Formulario de definición de actividades.

Figura D-22. Registrarse/Entrar.

Figura D-23. Enlace a formulario de elemento de trabajo.

Figura D-24. Creación de página de elemento de trabajo.

Figura D-25. Formulario de definición de elementos de trabajo.

Figura D-26. Elemento de trabajo definido.

Figura D-27. Registrarse/Entrar.

Figura D-28. Asignar categoría a una página.

Figura D-29. Categoría asignada.

Figura D-30. Enlace para editar elemento de trabajo con formulario.

Figura D-31. Formulario de elemento de trabajo.

Figura D-32. Asignación de actividades a un elemento de trabajo.

Figura D-33. Asignación de procesos a un elemento de trabajo.

Figura D-34. Asignación de procesos a un elemento de trabajo.

Figura D-35. Barra de navegación.

Figura D-36. Procesos de la PAL-Wiki.

Figura D-37. Contenidos de un proceso.

Figura D-38. Presentación de transparencias.

Figura D-39. Pestañas en páginas de elementos de trabajo.

Figura D-40. Motor de búsqueda.

Figura D-41. Resultados de búsqueda.

Figura D-42. Consulta de un activo.

Figura D-43. Descarga de un activo.

Figura D-44. Contenidos del activo de proceso.

Figura D-45. Instanciación del activo de proceso.

Figura D-46. Enlace para subir un fichero.

Figura D-47. Formulario para subir un fichero.

Figura D-48. Enlace a ejemplos de activos.

Figura D-49. Enlace para publicar una instancia de activo.

Figura D-50. Formulario de publicación de instancias de activos.

Figura D-51. Datos de la instancia de activo.

Figura D-52. Enlace para consultar ejemplos de activos.

Figura D-53. Enlace para ejemplos de activos.

Figura D-54. Descarga de del activo para reuso.

Figura D-55. Enlace para consultar la ayuda de la wiki.

Figura D-56. Página de ayuda de la PAL-Wiki.

Figura D-57. Enlaces edición de páginas.

Figura D-58. Formulario de edición.

Figura D-59. Finalización de la edición.

Figura D-60. Enlace para retiro de activos.

Figura D-61. Página de borrado de página.

Figura D-62. Confirmación de borrado de página.

Figura D-63. Eliminación de activo interno.

Figura D-64. Enlace a páginas especiales.

Figura D-65. Enlace para restaurar páginas.

Figura D-66. Página para restaurar una página.

Figura D-67. Selección de página a restaurar.

Figura D-68. Página de registro de borrados.

Figura D-69. Confirmación de restauración de una página.

Figura D-70. Creación de una discusión.

Figura D-71. Realización de comentarios en una discusión.

Figura D-72. Página de la discusión creada.

Figura D-73. Enlace para crear una cuenta de usuario.

Figura D-74. Datos para creación de una cuenta de usuario.

Figura D-75. Confirmación de creación de cuenta.

Figura D-76. Enlace a configuración de permisos.

Figura D-77. Ingreso de una nombre de cuenta a configurar.

Figura D-78. Página de configuración de permisos de usuarios.

Figura D-79. Enlace para consultar historial de una página.

Figura D-80. Historial de versiones.

Figura D-81. Selección de versiones a comparar.

Figura D-82. Reporte de comparación de versiones.

Figura D-83. Apartado de cambios recientes y registros.

Figura D-84. Cambios recientes.

Figura D-85. Páginas nuevas.

Figura D-86. Galería de imágenes nuevas.

Figura D-87. Enlace para consultar estadísticas generales.

Figura D-88. Estadísticas de la PAL-Wiki.

Figura D-89. Enlace para consultar estadísticas de uso.

Figura D-90. Datos para generar estadísticas de uso.

Figura D-91. Reporte de estadísticas de uso.

Figura D-92. Página de acceso a Google Analytics.

Figura D-93. Panel con resumen de datos de acceso a la wiki.

Figura D-94. Detalle de un resumen de informe.

Figura D-95. Detalle de una métrica por unidad de tiempo.

ANEXO A - PUBLICACIONES, PROYECTOS Y DOCENCIA

1. PUBLICACIONES

Antonio Amescua, **Leonardo Bermón**, Javier García, María-Isabel Sánchez-Segura. Knowledge Repository to Improve Agile Development Processes Learning. Social Semantic web support for software development. IET Software, 2010.

Antonio Amescua, **Leonardo Bermón**, Javier García, María-Isabel Sánchez. Knowledge Repository for Learning Improvement in Agile Development Processes. European Software Process Improvement - EuroSPI, Grenoble, Francia, 2010.

Leonardo Bermón, Antonio Amescua, Javier García-Guzmán, María Isabel Sánchez-Segura. Development of Process Asset Libraries using Knowledge Repositories. Handbook of Research on Digital Libraries: Design, Development, and Impact. Yin-Leng Theng; Schubert Foo; Dion Goh; Jin-Cheon Na (Eds.), Febrero 2009, pp. 465-475.

Leonardo Bermón, Antonio Amescua, Javier García-Guzmán, Paloma Díaz. Process Asset Library in Software Process Support Technology: A Review of the Literature. IRMA Proceedings del 2007 International Conference on Information Resources Management, Vancouver, Canada. Mayo 2007.

Leonardo Bermón, Javier García-Guzmán, Antonio Amescua Seco. Plataforma para la Gestión del Conocimiento sobre Procesos de Software. Tendencias Ingeniería de Software e Inteligencia Artificial Volumen 3, Universidad Nacional de Colombia, Sede Medellín, 2009, pp. 50-58.

Leonardo Bermón, Antonio Amescua, Javier García-Guzmán. Desarrollo de Biblioteca de Activos de Procesos de Software Utilizando Técnicas de Gestión del Conocimiento. Tendencias Ingeniería de Software e Inteligencia Artificial Volumen 1, Universidad Nacional de Colombia, Sede Medellín. Julio 2007.

2. PROYECTOS

Proyecto: Proyecto de investigación colaborativa aplicada. ELDER: Plataforma software AAL (Ambient Assisted Living). Entidad financiadora: Life quality technologies. Abril 2010 – Diciembre 2010.

Proyecto: Proyecto de investigación colaborativa aplicada. Zonnect-servicios: plataforma de gestión de servicios. Entidad financiadora: Zonnect Redes de Ingeniería. Abril 2010 – Diciembre 2010.

Proyecto: Proyecto de investigación colaborativa aplicada. Sugar: sistema software de guiado avanzado en recintos. Entidad financiadora: Life quality technologies. Abril 2010 – Diciembre 2010.

Proyecto: Acuerdo particular de colaboración para la realización de actividades de I+D+i en el ámbito del "Buen Gobierno de las Tecnologías de la Información". Enero 2008 – Diciembre 2008. Entidad Financiadora: ATOS ORIGIN.

Proyecto: Plataforma CMMI colaborativa, distribuida y multi-idioma. Abril 2007 - Mayo 2008. Instituciones Participantes: Zonnect Redes de Ingeniería, S.L., Universidad Carlos III de Madrid, Universidad Politécnica de Madrid.

3. DOCENCIA

Asignatura: Principios de Ingeniería del Software. Cursos 2008/2009 y 2009/2010.
Coordinadora: María Isabel Sánchez.

ANEXO B- DESCRIPCIÓN DE ELEMENTOS DE PROCESO DE LA PAL-WIKI

En este anexo se presenta un conjunto de formatos que describen los diferentes elementos del proceso ágil definidos en la PAL-Wiki.

Los elementos de proceso descritos son:

- Actividades
- Roles
- Elementos de trabajo

1. ACTIVIDADES

1.1 Actividades del proceso Gestión de Historias de Usuario



1.1.1 Identificación de necesidades

Visión general

Es necesario extraer, organizar y gestionar las necesidades de los usuarios, así como establecer un lenguaje común para la comunicación entre el usuario y el equipo de desarrollo.

Roles participantes

- **Responsables**
 - Cliente/Usuario
 - Coordinador del Equipo

Criterios de entrada

- **Sin criterios:** No hay criterios de entrada para este proceso.

Cuándo

Al comienzo del proyecto, se deben identificar las necesidades del usuario.

Tareas

Identificación de roles	Existen diferentes factores para la identificación de roles: <ul style="list-style-type: none"> • Según la frecuencia en la que el usuario utilizará el sistema. • Nivel de experiencia en el dominio de la aplicación. • Nivel general de experiencia del usuario en la utilización de sistemas de información. • Los objetivos que pretende el usuario en relación con la utilización del software a desarrollar.
Identificación de personas representativas	En principio, el interlocutor es el cliente que participa en el equipo de desarrollo del proyecto. Solamente es deseable incluir otras personas si ayudan al equipo de desarrollo a definir historias que son críticas para el sistema.
Definición de la metáfora a utilizar	La metáfora ayuda a cualquier persona del proyecto a entender los elementos básicos y sus relaciones. <ul style="list-style-type: none"> • Se debe comprobar que la metáfora es útil para todos los implicados en el desarrollo porque facilita la comunicación y es entendida del mismo modo por todos los implicados. • Los miembros del equipo de desarrollo se sienten cómodos hablando sobre el sistema en términos de la metáfora.
Identificación de las historias de usuario	El cliente o el usuario, en colaboración con los desarrolladores, definen cada una de las interacciones con el sistema. La determinación de las historias de usuario se puede realizar mediante entrevistas, reuniones informales o jornadas de trabajo específicas.

Salidas

- Una pila de historias de usuario.

Activos asociados

- **Elementos de trabajo**
 - Historia de Usuario
- **Ejemplos**
 - Identificación de necesidades.xls



1.1.2 Priorización de historias

Visión general

Las historias deben ser priorizadas para determinar cuáles se van a desarrollar en primer lugar. La prioridad se determina mediante la aplicación de distintos criterios: Riesgos del proyecto, Importancia para la arquitectura del sistema y Criticidad para el usuario.

Los elementos que se priorizan son todos los que hay que hacer. Por ejemplo, historias de usuario, características, requisitos no funcionales, defectos, etc.

La prioridad será determinada de forma conjunta por todos los miembros del equipo, puntuando para cada historia de usuario identificada, cada uno de los tres factores y obteniendo un valor único de prioridad de la historia.

Este valor será la suma ponderada de los valores obtenidos en cada uno de los criterios. El equipo deberá determinar el grado relativo de importancia que tendrá cada uno de los criterios en el valor final de la prioridad.

Los posibles valores que se podrán asignar para cada criterio se expresan en reglas MoSCoW, que representan rasgos del requisito:

- **Must have:** Requisitos fundamentales del sistema (valor asignado 3).
- **Should have:** Requisitos importantes para los que habrá una solución a corto plazo (valor asignado 2).
- **Coul have:** Requisitos que podrían quedar fuera del sistema si no hay más remedio (valor asignado 1).
- **Want to have but won't have this time around:** Requisitos que son valorados pero que pueden esperar (valor asignado 0).

Roles participantes

- **Responsables**
 - Coordinador del Equipo
 - Cliente/Usuario

Criterios de entrada

- **Historias de usuario:** Representaciones de los requisitos de software escritos utilizando el lenguaje común.

Cuándo

Siempre al comienzo del proyecto y de forma consensuada.

Tareas

Determinar la prioridad por riesgos del proyecto	Se determina la prioridad de acuerdo a los siguientes riesgos para el proyecto: <ul style="list-style-type: none"> • Incertidumbre en el esfuerzo • Especificación pobre • Problemas políticos • Nuevas herramientas/técnicas/librerías
Determinar la prioridad por importancia para la arquitectura	Se priorizan las historias de usuario de acuerdo a la importancia/cobertura de casos de uso o componentes significativos para la arquitectura.
Determinar la prioridad por criticidad para el usuario	Se priorizan las historias de usuario de acuerdo a la importancia para la obtención de beneficios por parte de la empresa y necesidades del cliente.
Consensuar los niveles de prioridad	Se debe obtener un consenso de los niveles de prioridad determinados para las historias de usuario identificadas.

Salidas

- Una lista de historias de usuario ordenada según la prioridad.

Activos asociados

- **Elementos de trabajo**
 - Historia de Usuario
- **Ejemplos**
 - Priorización de historias.xls



1.1.3 Estimación de historias

Visión general	Debe realizarse una estimación de cuánto tiempo conllevará la implantación de las historias de usuario. Es necesaria una reunión de lanzamiento, una estimación individual (basada en la estimación más probable, pesimista y optimista) y una reunión de debate de las estimaciones para poder realizar una estimación final.	
Roles participantes	<ul style="list-style-type: none"> • Responsables <ul style="list-style-type: none"> ○ Coordinador del Equipo ○ Cliente/Usuario 	
Criterios de entrada	<ul style="list-style-type: none"> • Las historias de usuario que hay que analizar: Es necesario tener claro las Historias de Usuario para poderlas analizar correctamente. 	
Cuándo	Las historias hayan sido identificadas y priorizadas.	
Tareas	Identificar un subconjunto de historias de usuario	<p>En una reunión de lanzamiento se debe:</p> <ul style="list-style-type: none"> • Determinar el desarrollador o el grupo de desarrolladores más adecuado para proporcionar una estimación. • Proporcionar estimaciones para una suficiente cantidad de historias de usuario de máxima prioridad.
	Estimación individual	<p>Cada miembro del equipo, deberá realizar de forma individual una estimación del esfuerzo requerido para completar cada historia de usuario identificada. Esta estimación se deberá realizar en función del tamaño y/o complejidad técnica de la historia. Se debe referenciar la unidad de tamaño utilizada, la técnica o heurística utilizada para determinar el tamaño y la fórmula para convertir el tamaño en unidades de esfuerzo (horas de trabajo). Cada miembro deberá proporcionar un valor de esfuerzo aproximado, una cifra de esfuerzo para los casos: optimista, pesimista y más probable.</p>
	Realizar reunión de debate	<p>Se realiza una reunión de debate donde el equipo discute las estimaciones realizadas. Cada miembro del equipo presenta al resto las tres estimaciones realizadas para cada historia identificada y las razones que le han llevado a realizar esa estimación.</p>
	Crear estimaciones de implementación	<p>Después de escuchar al resto de miembros, cada individuo vuelve a realizar las tres estimaciones de esfuerzo para cada historia y se obtiene el valor de esfuerzo estimado para cada una ellas mediante la aplicación de la fórmula:</p> $\text{Estimación} = (\text{Estimación Optimista} + \text{Estimación Pesimista} + 4 \times \text{Estimación Más Probable}) / 6$ $\text{Desviación} = (\text{Estimación Pesimista} - \text{Estimación Optimista}) / 6$
	Estimación final y registro de la estimación	<p>El valor de esfuerzo que se asignará a cada historia será la media de los valores proporcionados por cada uno de los miembros del equipo. El valor de esfuerzo obtenido se registrará en la Lista de Historias de Usuario.</p>
Salidas	<ul style="list-style-type: none"> • Estimaciones de las historias de usuario. 	
Activos asociados	<ul style="list-style-type: none"> • Elementos de trabajo <ul style="list-style-type: none"> ○ Historia de Usuario • Ejemplos <ul style="list-style-type: none"> ○ EstimaciónHistorias.xls 	



1.1.4 Determinación de alcance de iteración

Visión general	El objetivo es determinar el plan de iteración reflejando los conocimientos más actuales sobre lo que se debe llevar a cabo en una iteración. El plan se debe completar antes de que se inicie la iteración. Un plan inicial se crea atendiendo a estimaciones y se perfila conforme las historias de usuario.	
Roles participantes	<ul style="list-style-type: none"> • Responsables <ul style="list-style-type: none"> ○ Coordinador del Equipo ○ Ingeniero de Software 	
Criterios de entrada	<ul style="list-style-type: none"> • Estimación de historias de usuario: Se debe tener una buena estimación de historias de usuario para obtener las diferentes iteraciones y así poder definir la duración de cada una. 	
Cuándo	Las historias de usuario han sido estimadas.	
Tareas	Decidir duración de la iteración	Se debe decidir la duración de la iteración. La mayor parte de los métodos indican de 1 a 6 semanas. Para equipos pequeños (hasta 10 personas) de 1 a 3 semanas. Para equipos grandes (hasta 200 personas) de 4 a 6. Las primeras iteraciones deberán durar más y las finales pueden ser más cortas y estables.
	Estimar la capacidad de entrega de cada trabajador	Se estima la capacidad de entrega de cada trabajador por unidad de tiempo. Se deberá calcular la capacidad de trabajo disponible, teniendo en cuenta la cantidad de personas y las horas de jornada diaria y semanal.
	Seleccionar las historias de usuario	Se identifican en la Lista de Historias de Usuario las historias que se desarrollarán en el ámbito de la primera iteración y se eligen las historias de usuario con prioridad más alta que se desean resolver en la iteración.
Salidas	<ul style="list-style-type: none"> • Plan de iteración para las historias de usuario. 	
Activos asociados	<ul style="list-style-type: none"> • Elementos de trabajo <ul style="list-style-type: none"> ○ Historia de Usuario • Productos de trabajo <ul style="list-style-type: none"> ○ Lista de Historias de Usuario • Ejemplos <ul style="list-style-type: none"> ○ Determinación del Alcance.xls ○ Capacidad de Trabajo.xls 	



1.1.5 Especificación de historias

Visión general	Se trata de completar la especificación de las historias de usuario consideradas en el alcance de la primera iteración. Para ello se deberá especificar la secuencia de pasos que el usuario debe desempeñar en cada una de las historias de usuario.	
Roles participantes	<ul style="list-style-type: none"> • Responsables <ul style="list-style-type: none"> ○ Cliente/Usuario ○ Ingeniero de Software 	
Criterios de entrada	<ul style="list-style-type: none"> • Lista de Historias de Usuario: Es necesario tener una visión básica de las historias de usuario que se van a tratar para poder especificar cada una de ellas. 	
Cuándo	Después de la determinación del alcance de la iteración.	
Tareas	Definición detallada de las historias de usuario	<p>Descripción detallada de cada historia de usuario para un mejor entendimiento. Se debe especificar la secuencia de pasos que el usuario debe realizar en cada historia de usuario.</p> <p>Se creará una hoja de cálculo en la Lista de Historias de Usuario para la especificación detallada de cada historia de usuario. Cada hoja se nombrará con el mismo identificador asignado en la Lista de Historias de Usuario a la historia que se está especificando en detalle.</p>
Salidas	<ul style="list-style-type: none"> • Especificación de historias de usuario. 	
Activos asociados	<ul style="list-style-type: none"> • Elementos de trabajo <ul style="list-style-type: none"> ○ Historia de Usuario • Productos de trabajo <ul style="list-style-type: none"> ○ Lista de Historias de Usuario • Ejemplos <ul style="list-style-type: none"> ○ Especificación de Historias.xls 	



1.1.6 Identificación de requisitos de calidad del servicio

Visión general

Para determinar los requisitos de calidad de servicio, analizar cada historia de usuario y establecer cuáles iteraciones necesitan requisitos de calidad de servicio. La lista de requisitos de calidad de servicio identifica y recopila los requisitos no funcionales de la aplicación. Es una lista de restricciones sobre la funcionalidad del sistema. El Coordinador del Equipo vuelve a evaluar y modificar la lista de requisitos de calidad de servicio conforme surgen nuevos requisitos debido a las pruebas y la necesidad de cambio del proyecto.

Roles participantes

- **Responsables**
 - Coordinador del Equipo
 - Ingeniero de Software

Criterios de entrada

- **Lista de Historias de Usuario:** Especificación de las historias de usuario definidas para el proyecto.

Cuándo

Se van a determinar las restricciones del sistema después de realizar la especificación de historias de usuario.

Tareas

- | | |
|---|---|
| Identificar objetivos de calidad de servicio | <p>Para la identificación de los objetivos de calidad del servicio:</p> <ul style="list-style-type: none"> • Examinar las categorías de calidad de servicio como plataforma, rendimiento, seguridad y capacidad. • Determinar los objetivos del sistema en cada una de estas categorías cuando sea posible. |
| Identificar requisitos de calidad de servicio | <p>Para la identificación de los requisitos de calidad del servicio:</p> <ul style="list-style-type: none"> • Analizar las historias de usuario y aplicar los objetivos de la calidad de servicio a los niveles de servicio y uso en todos los puntos en los que el sistema responda a un rol. • Expresar el requisito de calidad de servicio como una medida observable en el formato de contexto, estímulo y respuesta. Por ejemplo, el rendimiento de una búsqueda de catálogo puede requerir que los resultados de la búsqueda aparezcan en menos de 3 segundos. • Asignar los requisitos de calidad de servicio a un Ingeniero de Software. Guardar y sincronizar la lista de requisitos de calidad de servicio para crear los elementos de trabajo de los requisitos de calidad de servicio. Publicar la lista de requisitos de calidad de servicio. |

Salidas

- Lista de requisitos de calidad de servicio publicada.

Activos asociados

- **Elementos de trabajo**
 - Requisito de Calidad de Servicio



1.1.7 Creación de pruebas de aceptación

Visión general

Los clientes deben apreciar el avance del proyecto mediante la comprobación de que el software funciona. Para ello se deben escribir pruebas de aceptación que permitan que el cliente valide y verifique el funcionamiento correcto de las historias definidas.

Roles participantes

- **Responsables**
 - Cliente/Usuario

Criterios de entrada

- **Especificación de Historias de Usuario:** Es necesario tener especificado la secuencia de pasos que el usuario debe desempeñar en cada una de las historias de usuario.

Cuándo

Las pruebas de aceptación se deben definir periódica y repetidamente.

Tareas

Identificación de pruebas de aceptación	Es imprescindible especificar pruebas de aceptación para las historias dentro de una misma iteración. Las pruebas de aceptación deben ser fáciles de escribir y entender. Se utilizan notas de soporte para describir dónde están los datos de prueba y los resultados esperados. Cada prueba de aceptación tendrá: <ul style="list-style-type: none"> • Un identificador. • Un texto que describa el propósito de la prueba.
Especificación de pruebas de aceptación utilizando Pruebas Fit	En las notas de soporte donde se han introducido el identificador y texto descriptivo se añadirá un campo para indicar el fichero asociado a la prueba de aceptación. Este fichero va a contener una tabla de tipo Column Fixture, Action Fixture o Row Fixture, o una colección de tablas de estos tipos.
Creación de pruebas Column Fixture	Una tabla de tipo Column Fixture contiene las pruebas que se desean ejecutar sobre el resultado obtenido al realizar determinados cálculos de una clase. Las columnas de entrada de la tabla se van a corresponder con variables de la clase y las columnas calculadas con funciones.
Creación de pruebas Action Fixture	Una tabla Action Fixture define un flujo de trabajo a seguir y los valores que van a ir tomando los campos durante el proceso. Las filas contendrán una secuencia de comandos que se van a ejecutar en orden.
Creación de pruebas Row Fixture	Una tabla Row Fixture sirve para comprobar el valor de los elementos de una lista de objetos de la misma clase en un momento determinado. Cada fila se corresponderá con un objeto de la lista y cada columna con un campo o método del objeto.
Creación de una secuencia de pruebas Fit	En algunas ocasiones será necesario crear varios tipos de tablas para definir una misma prueba de aceptación. Las acciones definidas en tablas ActionFixtures pueden afectar a los elementos del sistema bajo pruebas, creando la necesidad de realizar pruebas sobre los fixtures implicados en el proceso.
Especificación de pruebas de aceptación sin utilizar Pruebas Fit	En los casos en los que no se puedan especificar las pruebas de aceptación mediante Pruebas Fit será necesario incluir en las notas de soporte, además del identificador y el texto descriptivo, un campo para poder indicar detalladamente los pasos a realizar en la aplicación para la ejecución de dicha prueba.

Salidas

- Listado de pruebas de aceptación para las diferentes historias de usuario.
- Ficheros creados mediante el complemento Pruebas Fit de Visual Studio que contengan la especificación de pruebas de aceptación en tablas Fit.
- Clases desarrolladas para la ejecución de pruebas Fit.

**Activos
asociados**

- **Elementos de trabajo**
 - Prueba de Aceptación
- **Productos de trabajo**
 - Lista de Historias de Usuario
- **Instrucciones técnicas**
 - Cómo crear Column Fixtures
 - Cómo crear Action Fixtures
 - Cómo crear Row Fixtures
 - Cómo crear una lista de tablas Fit
- **Transparencias**
 - Pruebas de Aceptación
 - Enunciado Ejercicio 1
 - Solución Ejercicio 1
 - Enunciado Ejercicio 2
 - Solución Ejercicio 2



1.1.8 Ejecución de pruebas de aceptación

Visión general

Los resultados de las pruebas varían, desde una ejecución completamente satisfactoria a un bloqueo total del sistema. Una vez que se ejecute un caso de prueba, es importante registrar los resultados.

Es necesaria una revisión por pares y tener un equipo especializado para ello.

Roles participantes

- **Responsables**
 - Coordinador del Equipo
 - Cliente/Usuario
 - Ingeniero de Software

Criterios de entrada

- **Pruebas de aceptación documentadas:** Se han escrito las pruebas de aceptación para las historias de usuario.
- **Clases desarrolladas con las historias de usuario implementadas:** Las historias de usuario han sido implementadas y están disponibles con la funcionalidad que se va a comprobar.
- **Ficheros de automatización de las pruebas de aceptación:** Se han creado pruebas de aceptación mediante el complemento Pruebas Fit de Visual Studio.
- **Clases desarrolladas para la ejecución de pruebas Fit.**

Cuándo

Las pruebas deben ejecutarse periódicamente y repetidamente cuando se haya terminado la funcionalidad de una historia de usuario.

Tareas

Determinar las pruebas que se van a ejecutar	<p>Tener en cuenta:</p> <ul style="list-style-type: none"> • En función de la tarea de prueba, identificar y establecer un orden de prioridad de las pruebas que se van a ejecutar. • Comprobar las notas de la última versión del sistema para asegurarse de que la funcionalidad que se va a comprobar está disponible.
Identificar la configuración de la prueba	Recuperar, instalar y definir la configuración inicial solicitada en la prueba. Esto incluye el hardware, software y los datos de las pruebas.
Obtener la última versión del sistema	Obtener la última versión del sistema desde un servidor de control de versiones o una fuente autorizada.
Ejecutar la prueba	<p>Según el tipo de prueba:</p> <ul style="list-style-type: none"> • Prueba fit: arrancar el complemento Pruebas Fit de Visual Studio y ejecutar el fichero asociado a la prueba de aceptación. • Prueba manual: ejecutar cada paso definido en la prueba. • Prueba automatizada: abrir el entorno de pruebas para ejecutar las pruebas automatizadas o el script de pruebas. • Registrar los resultados de la prueba en las notas de soporte (Product Backlog).
Analizar los resultados de las pruebas	<p>Para realizar el análisis:</p> <ul style="list-style-type: none"> • Adjuntar el requisito de calidad de servicio a los resultados de la prueba. • Comparar los resultados de las pruebas con los resultados que se esperaban. • Abrir un defecto si los resultados no satisfacen las expectativas. • Actualizar las pruebas para reflejar el comportamiento apropiado del sistema.

Salidas

- La tarea de pruebas se cierra porque todas las pruebas se han ejecutado sin que se produzca ningún bloqueo.
- Se han creado defectos a tener en cuenta en caso de un comportamiento imprevisto.

**Activos
asociados**

- **Elementos de trabajo**
 - Prueba de Aceptación
- **Productos de trabajo**
 - Lista de Historias de Usuario
- **Instrucciones técnicas**
 - Cómo ejecutar Pruebas Fit
- **Ejemplos**
 - Ejecución de pruebas de aceptación

1.2 Actividades del proceso Planificación



Visión general

1.2.1 Identificación de tareas

Las historias son descripciones de funcionalidades esperados por un usuario, no son listas de actividades para los desarrolladores.

Para iniciar la planificación de una iteración se debe discutir la historia y descomponer la historia en tareas constitutivas.

Para la identificación de las tareas:

- Crear una nueva tarea para cada una de las áreas principales del sistema afectadas por la historia de usuario.
- Para cada tarea, describir la funcionalidad nueva o modificada.
- Agrupar los cambios de menor nivel que estén lógicamente relacionados con tareas de mayor nivel.
- Dividir los cambios y adiciones de nivel superior en tareas de nivel más bajo en caso de que éstas sean demasiado amplias.
- Revisar el conjunto de tareas para asegurarse de que dichas tareas corresponden a la historia de usuario.

Roles participantes

- **Responsables**
 - Coordinador del Equipo
 - Ingeniero de Software

Criterios de entrada

- **Historias de Usuario:** Especificación detallada de las historias de usuario.

Cuándo

Después del desarrollo de la especificación de historias.

Tareas

Discutir las historias	<p>Reunir al equipo que esté trabajando en las historias. El Coordinador del Equipo facilitará esta reunión o el análisis por correo electrónico y registrará las tareas. Se discutirá desde el punto de vista técnico el funcionamiento esperado de la historia de usuario.</p> <ul style="list-style-type: none"> • El cliente comienza con la historia con prioridad más alta y la lee a los desarrolladores. • Los desarrolladores hacen preguntas hasta que entienden suficientemente la historia para descomponerla en tareas consecutivas. • Los desarrolladores podrán resolver con el cliente detalles concretos de las historias después de la reunión de planificación.
Identificar las tareas consecutivas	<p>Identificar las tareas necesarias para realizar la historia de usuario, teniendo en cuenta que los tipos de tareas son:</p> <ul style="list-style-type: none"> • Diseño técnico de componentes: Será necesario hacer una primera identificación de componentes/clases a desarrollar. Se puede utilizar un diagrama de secuencias para determinar los componentes a diseñar y las responsabilidades correspondientes. • Definición y automatización de pruebas unitarias para cada componente: Esta tarea se debe desglosar dependiendo de los componentes que se identifiquen. • Codificación y prueba unitaria de componentes: Esta tarea se debe desglosar dependiendo de los componentes que se identifiquen. • Ejecución de pruebas de aceptación.

Salidas

-
- Una lista de tareas para resolver la historia de usuario.
-

**Activos
asociados**

- **Elementos de trabajo**
 - Tarea
 - Historia de Usuario
 - **Productos de trabajo**
 - Lista de Tareas
 - **Ejemplos**
 - Identificación de Tareas.xls
-



1.2.2 Identificación de responsabilidades

Visión general	Un desarrollador acepta la responsabilidad de realizar cada tarea.	
Roles participantes	<ul style="list-style-type: none"> • Responsables <ul style="list-style-type: none"> ○ Coordinador del Equipo ○ Ingeniero de Software 	
Criterios de entrada	<ul style="list-style-type: none"> • Lista de tareas: Un listado de tareas para poder desarrollar una historia de usuario. 	
Cuándo	Se han identificado las tareas durante la planificación.	
Tareas	Asignar responsabilidad a cada tarea	Permitir que las tareas sean lo suficientemente amplias como para permitir que surja el diseño necesario pero con la seguridad suficiente de que una sola persona podría realizar dichas tareas entre uno y tres días.
	Generación de la planificación de la iteración	<p>Se genera la planificación de la iteración, la cual es una lista de tareas y responsabilidades.</p> <p>Para cada tarea se deberá incluir los siguientes campos:</p> <ul style="list-style-type: none"> • Identificador de la tarea. • Nombre representativo de la tarea. • Historia de usuario relacionada. • Responsable de la tarea. • Estado de realización de la tarea (sin comenzar, en curso, completada).
Salidas	<ul style="list-style-type: none"> • Una lista de tareas y responsabilidades. 	
Activos asociados	<ul style="list-style-type: none"> • Elementos de trabajo <ul style="list-style-type: none"> ○ Tarea ○ Recurso • Productos de trabajo <ul style="list-style-type: none"> ○ Lista de Tareas • Ejemplos <ul style="list-style-type: none"> ○ Identificación de Responsabilidades.xls 	



1.2.3 Determinación del esfuerzo de las tareas

Visión general	Después de que se han aceptado las tareas, los desarrolladores estiman individualmente las tareas para cerciorarse de que no están sobre-estimadas.	
Roles participantes	<ul style="list-style-type: none">• Responsables<ul style="list-style-type: none">○ Coordinador del Equipo○ Ingeniero de Software	
Criterios de entrada	<ul style="list-style-type: none">• Lista de tareas y responsabilidades: Lista de tareas y responsabilidades para realizar las historias de usuario.	
Cuándo	Durante la planificación.	
Tareas	Estimación del esfuerzo	Se realiza una estimación del esfuerzo de realización de las tareas para la siguiente iteración.
Salidas	<ul style="list-style-type: none">• Estimación del esfuerzo para la realización de las tareas planificadas.	
Activos asociados	<ul style="list-style-type: none">• Elementos de trabajo<ul style="list-style-type: none">○ Tarea○ Recurso• Productos de trabajo<ul style="list-style-type: none">○ Lista de Tareas• Ejemplos<ul style="list-style-type: none">○ Determinación del Esfuerzo.xls	



1.2.4 Ajustes al plan de iteración

Visión general

Cuando el desarrollador ha estimado cada una de sus tareas, necesita evaluar de manera realista si será capaz de completar todo el trabajo que tiene asignado para la iteración.

Si el desarrollador tiene dudas debe:

- Mantener todas las tareas y tener fe.
- Pedir a alguien más del equipo que realice sus tareas.
- Acordar con el usuario que una historia se acorte o dividirla y acortar alguna de sus partes.

Roles participantes

- **Responsables**
 - Coordinador del Equipo

Criterios de entrada

- **Estimaciones de esfuerzo:** Se han obtenido estimaciones de esfuerzo de las tareas para desarrollar una historia de usuario.

Cuándo

Después de la identificación de tareas y responsables.

Tareas

Ajustes al plan

Cuando se han agregado tareas de desarrollo al plan de iteración, debe asegurarse que el costo total de los elementos de trabajo del plan no supera la velocidad media del equipo de desarrollo.

- Si el requisito de calidad de servicio es un requisito de rendimiento, asegurarse que las tareas de desarrollo contienen presupuestos de rendimiento.
- Si el costo total de los elementos de trabajo supera el presupuesto, cambiar de lugar las historias de usuario, los requisitos de calidad de servicio o las adjudicaciones de errores a la siguiente iteración. Establecer el campo de iteración del elemento de trabajo en esa iteración.
- Si el costo de los elementos de trabajo es inferior a la velocidad media, pensar en la posibilidad de agregar nuevas historias de usuario, requisitos de calidad de servicio o adjudicaciones de errores para completar el programa.

Salidas

- Plan de iteración ajustado.

Activos asociados

- **Elementos de trabajo**
 - Tarea
 - Recurso
- **Productos de trabajo**
 - Lista de Tareas
- **Ejemplos**
 - Ajuste del Plan.xls



1.2.5 Celebrar reunión de planificación

Visión general	Finalice el plan de iteración con la celebración de una reunión sobre el planeamiento de iteraciones con el fin de presentar el contenido de la próxima iteración.	
Roles participantes	<ul style="list-style-type: none"> • Responsables <ul style="list-style-type: none"> ○ Coordinador del Equipo ○ Ingeniero de Software 	
Criterios de entrada	<ul style="list-style-type: none"> • Listado de tareas planificadas: Se tiene un conjunto de tareas identificadas junto con sus responsables y estimaciones de su desarrollo. 	
Cuándo	Durante la finalización de la planificación.	
Tareas	Celebrar la reunión sobre el inicio de la iteración	Celebración de una reunión sobre la planificación de iteraciones. Comunicar cualquier prioridad de la iteración. Cualquier riesgo que presente la iteración debería determinarse como una dependencia de un recurso externo.
Salidas	<ul style="list-style-type: none"> • El plan de iteración final refleja un conjunto de tareas programadas para la próxima iteración. 	
Activos asociados	<ul style="list-style-type: none"> • Elementos de trabajo <ul style="list-style-type: none"> ○ Tarea • Productos de trabajo <ul style="list-style-type: none"> ○ Lista de Tareas 	

1.3 Actividades del proceso Desarrollo



1.3.1 Diseño

Visión general

El diseño es rápido, la idea es identificar las principales clases del sistema, cuáles son sus responsabilidades y cómo implementarlas. Utilizarlo como intercambio de ideas, entre los miembros de desarrollo, acerca del funcionamiento interno del sistema a desarrollar.

El diseño adecuado para el software es aquel que:

- Funciona en todas las pruebas.
- No tiene lógica duplicada.
- Manifiesta cada intención importante para los programadores.
- Tiene el menor número posible de clases y métodos.

Roles participantes

- **Responsables**
 - Ingeniero de Software

Criterios de entrada

- **Historias de usuario:** Se tiene una especificación de las historias de usuario a implementar.

Cuándo

Se han seleccionado las tareas a desarrollar dentro del plan de iteración.

Tareas

Revisión de historias de usuario	Se revisa la especificación de historias de usuario para determinar conceptos y acciones que permitan la elaboración del diseño.
Identificación de clases y responsabilidades	<p>Para la identificación de clases y responsabilidades:</p> <ul style="list-style-type: none"> • Elaborar un diagrama de secuencias en el que se identifiquen las clases necesarias para implementar las historias de usuario. El diagrama de secuencias debe representar las responsabilidades de cada una de las clases identificadas mediante la definición de métodos con sus parámetros correspondientes. • Elaborar las fichas CRC de las clases identificadas.
Desarrollar modelo de rendimiento	<p>El modelo de rendimiento es un proceso que ayuda a identificar y abordar temas relacionados con el rendimiento de carácter potencial en la aplicación. Cualquier modelo de rendimiento se desarrolla a partir de un requisito de calidad de servicio que se divide en tareas de desarrollo. A cada tarea de desarrollo se le asigna un presupuesto de rendimiento para su implementación.</p> <p>Se debe:</p> <ul style="list-style-type: none"> • Revisar el requisito de calidad de servicio. • Identificar carga de trabajo. • Identificar objetivos de rendimiento. • Identificar el presupuesto de rendimiento. • Asignar presupuesto. • Evaluar presupuesto. • Validar modelo.
Crear prototipo de arquitectura	<p>La creación de prototipos iniciales reduce los riesgos generales del proyecto y los imprevistos. Riesgos menores en el proyecto y menos imprevistos hacen que la planificación y la estimación en posteriores iteraciones sean más precisas. Los prototipos pueden ser de carácter temporal, se pueden descartar una vez que se han controlado determinados temas</p>

	<p>que así lo requerían o se pueden generar como base de la arquitectura principal.</p> <ul style="list-style-type: none"> • Generar el prototipo. Se debe seguir centrándose en el problema que se debe solucionar. • Asegurar que el prototipo controla debidamente el aspecto que se va a explorar. • Implementar el prototipo en un entorno parecido al entorno de destino. • Comprobar el prototipo en aquellos aspectos que presentan alto nivel de riesgo. • Agregar cualquier nuevo elemento de trabajo que presente algún riesgo y que el prototipo pueda no haber cubierto.
Salidas	<ul style="list-style-type: none"> • Un contrato en el que se describen el comportamiento detallado del sistema en función de los cambios de estado de los objetivos de modelo de dominio.
Activos asociados	<ul style="list-style-type: none"> • Elementos de trabajo <ul style="list-style-type: none"> ○ Clase ○ Requisito de Calidad de Servicio • Productos de trabajo <ul style="list-style-type: none"> ○ Tarjetas CRC • Transparencias <ul style="list-style-type: none"> ○ Diseño Simple • Ejemplos <ul style="list-style-type: none"> ○ Tarjetas CRC ○ Diagrama de secuencias



1.3.2 Definición de pruebas unitarias

Visión general

Los programadores extremos deben probar todo lo que puede llegar a fallar, utilizando pruebas automatizadas que deben ejecutarse perfectamente en todo momento.

En general, para el diseño de las pruebas se tienen en cuenta dos enfoques:

- Enfoque Funcional (Caja negra): Se centra en los métodos, en sus entradas y sus salidas.
- Enfoque Estructural (Caja blanca): Se centra en la estructura interna del programa, analizando los caminos de ejecución.

Roles participantes

- **Responsables**
 - Cliente/Usuario
 - Ingeniero de Software

Criterios de entrada

- **Diseño preliminar:** Se han definido las clases del sistema junto con sus responsabilidades y cuestiones de implementación.

Cuándo

Antes de la codificación.

Tareas

Crear una clase de prueba	Una clase de prueba es una clase asociada con una clase del proyecto y contiene pruebas para los métodos de la clase del proyecto. La clase de prueba es utilizada para mantener las pruebas que se crearán.
Crear un método de prueba	A partir de la clase de prueba se crea un método para crear y configurar el objeto a probar. Un método de prueba implementa una única prueba (la prueba de una porción de funcionalidad).
Escribir una prueba	Para escribir la prueba unitaria: <ul style="list-style-type: none"> • Identificar las pruebas unitarias para que la tarea de desarrollo cubra la mayor funcionalidad posible. • Escribir o actualizar una sola prueba unitaria cada vez. • Utilizar un soporte temporal y objetos ficticios para maximizar el área de código probada. • Si el código ya se ha escrito, generar pruebas unitarias desde el código que funciona. • Si el método que se prueba cambia el estado del objeto probado, codificar una aserción.

Salidas

- Pruebas unitarias definidas.

Activos asociados

- **Elementos de trabajo**
 - Clase
 - Prueba Unitaria
- **Productos de trabajo**
 - Tarjetas CRC
- **Transparencias**
 - Pruebas Unitarias
 - Enunciado Ejercicio 1
 - Solución Ejercicio 1
 - Enunciado Ejercicio 2
- **Videos**
 - Pruebas Unitarias - Paso1
 - Pruebas Unitarias – Paso2
 - Pruebas Unitarias – Paso3
 - Pruebas Unitarias – Paso4

-
- Pruebas Unitarias – Paso5
 - Pruebas Unitarias – Paso6
 - Pruebas Unitarias – Paso7
 - Pruebas Unitarias – Paso8
 - Pruebas Unitarias – Paso9
 - Pruebas Unitarias – Paso10
 - **Ejemplos**
 - Ejemplo 1 de prueba unitaria
 - Ejemplo 2 de prueba unitaria
 - Código del Enunciado 1
 - Código de la solución del Enunciado 1
 - Código del Enunciado 2
 - Código de la solución del Enunciado 2
-



1.3.3 Codificación

Visión general

Al comenzar un proyecto de software, es necesario establecer un estándar de codificación, aceptado e implantado por todo el equipo, para asegurarse de que todos los programadores del proyecto trabajen de forma coordinada. Se pretende que este estilo ayude a construir programas correctos, entendibles y fáciles de mantener. Un código fuente completo debe reflejar un estilo armonioso, como si un único programador hubiera escrito todo el código de una sola vez.

Roles participantes

- **Responsables**
 - Ingeniero de Software

Criterios de entrada

- **Diseño:** Clases y responsabilidades definidas.
- **Pruebas de aceptación:** Pruebas de aceptación para las diferentes historias de usuario.
- **Plan de iteración:** Un plan de iteración con las tareas relacionadas con la codificación ha sido realizado.

Cuándo

Los desarrolladores han comenzado a trabajar sobre una nueva tarea de desarrollo escribiendo la primera prueba.

Tareas

Desarrollo de código fuente	<p>Se codifican las clases y responsabilidades diseñadas. Tener en cuenta que:</p> <ul style="list-style-type: none"> • Se deben cumplir las normativas de código, para que el código tenga un estilo consistente e independiente del autor y el software sea más fácil de entender y mantener. • Se debe programar en parejas. Todo el código desarrollado es escrito por dos desarrolladores sentados frente a una única máquina de trabajo. Un miembro de la pareja es el "conductor", es decir, tiene el control del ratón y del teclado. La otra persona observa continuamente el trabajo del conductor, mirando los defectos, pensando en alternativas, etc.
Ejecución de pruebas unitarias	<p>Se realizan las pruebas anteriormente definidas. Para ello:</p> <ul style="list-style-type: none"> • Ejecutar la prueba, asegurándose de que la prueba es errónea si algún elemento no está completado y se supera si el elemento funciona como está previsto. • Corregir las partes de la prueba unitaria que produzcan resultados incorrectos o ambiguos. • Realizar la prueba unitaria hasta que se completen todas las pruebas identificadas de la tarea.
Registro del código fuente	<p>Cuando se ejecuten satisfactoriamente todas las pruebas preestablecidas, registrar el código fuente y la prueba en el servidor de configuración.</p>

Salidas

- Un código bien estructurado.

Activos asociados

- **Elementos de trabajo**
 - Clase
 - Prueba Unitaria
- **Transparencias**
 - Normativas de Código
 - Programación por parejas
- **Ejemplos**
 - Ejemplo de normativa de código
 - Ejemplo 1 de clase (Clase Function)
 - Ejemplo 2 de clase (Clase FunctionTest)

1.4 Actividades del proceso Refactoring



1.4.1 Identificar áreas susceptibles de reorganización

Visión general	Se deben identificar las áreas susceptibles de reorganización en el sistema en aquellos elementos que presenten indicadores de problemas.	
Roles participantes	<ul style="list-style-type: none">• Responsables<ul style="list-style-type: none">○ Ingeniero de Software	
Criterios de entrada	<ul style="list-style-type: none">• Pruebas unitarias: Las pruebas unitarias para el área están actualizadas y se ejecutan correctamente.	
Cuándo	Existe código duplicado, aparecen métodos o clases muy extensas, las listas de parámetros de los métodos son grandes, es necesario cambiar el nombre de alguna variable, se quiere transformar un campo público en uno privado mediante una propiedad, etc.	
Tareas	Búsqueda de áreas susceptibles de organización	Cuando se agrega funcionalidad, se buscan las áreas del código que aumenten la complejidad de la arquitectura, identificando las áreas dónde aplicar refactoring.
	Definir y ejecutar pruebas	Ejecutar pruebas sobre las áreas identificadas. En caso de que no existan las pruebas, éstas se deben crear.
Salidas	<ul style="list-style-type: none">• Lista de áreas susceptibles de reorganización.• Definición y ejecución de pruebas sobre áreas a reorganizar.	
Activos asociados	<ul style="list-style-type: none">• Elementos de trabajo<ul style="list-style-type: none">○ Clase○ Prueba Unitaria	



1.4.2 Aplicar la refactorización

Visión general

Las técnicas de refactorización se aplican en pasos pequeños y de uno en uno, donde reorganizaciones mayores deben descomponerse en refactorizaciones más pequeñas. Cada reorganización debe realizarse en pocos minutos para minimizar el peligro de introducir defectos. Se recomienda realizar la refactorización que estén orientadas hacia los objetivos de los usuarios y clientes.

Roles participantes

- **Responsables**
 - Ingeniero de Software

Criterios de entrada

- **Áreas de reorganización:** Las áreas susceptibles de reorganización han sido identificadas.

Cuándo

Es una práctica que debe convertirse en un hábito y aplicarse todos los días por el equipo de desarrollo.

Tareas

Determinar refactorización	Determinar el tipo de reorganización de código que debe ser aplicado, a la vez que se mantiene igual la semántica del código.
----------------------------	---

Aplicar refactorización	<p>Aplicar cada técnica de refactorización de una en una. Cambiar el código y todas las referencias al área que se ha cambiado, según sea necesario.</p> <ul style="list-style-type: none"> • Si una clase no asigna una abstracción del mundo real o elemento de programación, cambiar el nombre de la clase a uno que asigne adecuadamente. Cambiar todas las referencias al código. • Si los métodos son demasiado complicados, es posible que sea necesario dividir un método en una serie de nuevos métodos. Utilizar técnicas como "extraer método" para sacar la funcionalidad que pertenece al propio método. Se debe asegurar que el nuevo método sea coherente con la abstracción (nombre de clase) en la que se agrega.
-------------------------	--

Ejecutar pruebas	Ejecutar las pruebas entre cada paso de refactorización para asegurar que el comportamiento del sistema no ha cambiado.
------------------	---

Salidas

- Áreas del sistema reorganizadas mediante la aplicación de la refactorización.

Activos asociados

- **Elementos de trabajo**
 - Clase
- **Productos de trabajo**
 - Normativa de Código



1.4.3 Realizar pruebas unitarias

Visión general

Las pruebas unitarias son una precondition necesaria para la refactorización. Una vez se hayan aplicado técnicas de refactorización se debe verificar si se han introducido o no defectos en el sistema. Las pruebas unitarias deben ejecutarse nuevamente, luego se buscan otras áreas susceptibles de reorganización y la refactorización continúa si es necesario.

Roles participantes

- **Responsables**
 - Ingeniero de Software

Criterios de entrada

- **Áreas reorganizadas:** Se han aplicado técnicas de refactorización para reorganizar determinadas áreas del sistema.

Cuándo

Se desea verificar el correcto funcionamiento de las áreas refactorizadas.

Tareas

- | | |
|----------------------------|--|
| Realizar pruebas unitarias | <p>Ejecutar las pruebas una vez más al finalizar con éxito la reorganización de código, integrar y ejecutar Pruebas unitarias completas y Pruebas de aceptación.</p> <p>Para la ejecución de las pruebas unitarias se debe realizar:</p> <ul style="list-style-type: none"> • Pruebas unitarias para que el área permanezca semánticamente equivalente después de la refactorización. Se deben corregir las pruebas unitarias que no funcionen. • Análisis de código y corregir las advertencias. • Volver a realizar pruebas unitarias si se realizan cambios en el código como resultado de un análisis del código. |
|----------------------------|--|

Salidas

- Las pruebas se ejecutan correctamente.

Activos asociados

- **Elementos de trabajo**
 - Prueba Unitaria

1.5 Actividades del proceso Seguimiento Periódico



1.5.1 Recopilar la información de seguimiento

Visión general

Se establece una barra de nivel mínimo de aceptación y a continuación, se mide el estado actual de las tareas y de las pruebas de aceptación en relación con ese nivel mínimo de aceptación.

El estado actual debe evaluarse constantemente en relación con las necesidades cambiantes, condiciones de mercado o expectativas del Cliente/Usuario. El nivel mínimo de aceptación se actualiza después de cada iteración.

Roles participantes

- **Responsables**
 - Coordinador del Equipo
 - Ingeniero de Software

Criterios de entrada

- **Lista de historias de usuario:** La lista de historias de usuarios está actualizada.
- **Lista de tareas:** La lista de tareas está actualizada y ordenada por prioridades.
- **Pruebas de aceptación:** Información sobre las pruebas de aceptación establecidas para las historias de usuario incluidas en la iteración.

Cuándo

Antes de la realización de la reunión de seguimiento periódico.

Tareas

- | | |
|---|---|
| Recopilar información | <p>Se recopila información sobre:</p> <ul style="list-style-type: none"> • Estado actual del esfuerzo empleado en la realización de las tareas. • Estado de las pruebas de aceptación que son superadas por la versión ejecutable actual del código de la aplicación. |
| Calcular estimaciones para finalizar las tareas | <p>Se debe:</p> <ul style="list-style-type: none"> • Proporcionar una estimación de esfuerzo para finalizar la iteración. • Estimar la fecha de finalización de los trabajos de la iteración. |

Salidas

- Información sobre el estado actual del proyecto.

Activos asociados

- **Elementos de trabajo**
 - Tarea
 - Prueba de Aceptación
- **Productos de trabajo**
 - Plantillas de Seguimiento
- **Ejemplos**
 - Seguimiento Periódico1.xls
 - Seguimiento Periódico2.xls



1.5.2 Celebrar reunión de seguimiento periódico

Visión general

Se celebra la reunión de seguimiento periódico teniendo en cuenta los siguientes aspectos:

La reunión tendrá lugar de pié para asegurar su brevedad.

- La duración de la reunión será de 15 ó 20 minutos y asistirán de 7 a 10 personas. Algunas reuniones al comienzo del proyecto pueden durar algo más.
- Los que no son miembros del equipo pueden asistir como oyentes pero sin voz ni voto.
- En la pizarra se escriben todas las tareas de las que se informa durante la reunión. Las tareas se borran cuando se finalicen.
- El Coordinador del Equipo asegura que se siguen las reglas y prepara el lugar para una reunión eficiente.
- La reunión debe comenzar a tiempo.
- No se permite otra discusión ajena a las preguntas anteriormente mencionadas.
- Si se necesitan discutir otros asuntos, se celebrarán reuniones secundarias después de la reunión diaria, con algunos miembros del equipo.

Roles participantes

- **Responsables**
 - Coordinador del Equipo
 - Ingeniero de Software
 - Cliente/Usuario

Criterios de entrada

- **Estado actual del proyecto:** Información sobre el estado actual de las tareas y de las pruebas de aceptación.

Cuándo

Periódicamente a la misma hora y en un mismo lugar.

Tareas

- | | |
|---------------------|---|
| Celebrar la reunión | Para celebrar la reunión: <ul style="list-style-type: none"> • Establecer las reglas de base en cuanto a interrupciones y compromisos personales. • Crear una lista con dos columnas enumerando lo positivo y lo negativo realizado. • A partir de estas ideas, trabajar con el equipo para crear soluciones y acciones correctoras en caso de ser necesarias. |
|---------------------|---|

Salidas

- Opiniones y aportes sobre cómo mejorar el proceso.

Activos asociados

- **Elementos de trabajo**
 - Tarea
- **Productos de trabajo**
 - Plantillas de Seguimiento



1.5.3 Publicar el informe de seguimiento

Visión general	Se realiza un resumen de las soluciones propuestas en la reunión de seguimiento del proyecto en una plantilla de comentarios. Luego, el documento se publica en el portal del proyecto.	
Roles participantes	<ul style="list-style-type: none">• Responsables<ul style="list-style-type: none">○ Coordinador del Equipo	
Criterios de entrada	<ul style="list-style-type: none">• Mejoras del proceso: Opiniones y aportes sobre cómo mejorar el proceso obtenidos de la reunión de seguimiento del proyecto.	
Cuándo	Al finalizar la reunión de seguimiento periódico.	
Tareas	Publicación del informe de seguimiento	<ul style="list-style-type: none">• El informe de seguimiento del proyecto de desarrollo y las conclusiones obtenidas de la celebración de la reunión de seguimiento son publicados.
Salidas	<ul style="list-style-type: none">• Publicación de las conclusiones de la reunión de seguimiento periódico.	
Activos asociados	<ul style="list-style-type: none">• Elementos de trabajo<ul style="list-style-type: none">○ Tarea• Productos de trabajo<ul style="list-style-type: none">○ Plantillas de Seguimiento	

1.6 Actividades del proceso Integración Automática



Visión general

1.6.1 Editar fichero de construcción del ejecutable

Para iniciar el proceso de integración automática se deben gestionar todos los ficheros fuente y de soporte en un repositorio centralizado.

Después de conectarse a dicho repositorio, se extraen las últimas versiones de los ficheros fuente y se extrae cualquier librería adicional. Las pruebas unitarias a realizar sobre el código deben ser identificadas y se codifican las secuencias de registro de resultados y reporte.

Roles participantes

- **Responsables**
 - Ingeniero de Software

Criterios de entrada

- **Servidor de gestión de configuración:** Todo el código reside en una fuente única.
- **Conjunto de cambios:** El conjunto de cambios para una tarea de desarrollo ha pasado las pruebas unitarias, el análisis de código y se ha revisado.

Cuándo

Por lo menos una vez al día.

Tareas

Identificación del código fuente a integrar	Todos ficheros fuentes son identificados y obtenidos en el repositorio de control de fuentes.
Configuración de la compilación	Se configuran los elementos que correspondan a la compilación de código fuente de la aplicación.
Identificación y ejecución de pruebas unitarias y de integración	Se escriben y ejecutan pruebas unitarias y de integración a realizar sobre el código a integrar, para comprobar el correcto funcionamiento del sistema.
Codificación de secuencias de registro de resultados y reporte	Se codifican las secuencias de registro de resultados y reporte para obtener los resultados de las pruebas unitarias como consecuencia de la integración.

Salidas

- Fichero de construcción del ejecutable.

Activos asociados

- **Elementos de trabajo**
 - Elemento de Integración Automática
- **Productos de trabajo**
 - Ejemplo de Fichero Nant.build



1.6.2 Editar el fichero de configuración de integración continua

Visión general

La integración automática es el proceso de ensamblar los componentes individuales para formar el producto de software. Se deben programar las tareas de despliegue en el entorno en que se va a ejecutar el proyecto. Las tareas de las compilaciones de integración continua se configuran en un fichero.

Para que la integración sea exitosa, los cambios deben ser pequeños. Por medio de la integración continua y automática se obtiene una retroalimentación inmediata sobre el sistema.

Roles participantes

- **Responsables**
 - Ingeniero de Software

Criterios de entrada

- **Fichero de construcción del ejecutable:** Se han identificado los elementos de integración automática.

Cuándo

Por lo menos una vez al día.

Tareas

- | | |
|--|--|
| Edición del fichero de configuración de integración continua | Se edita el fichero de configuración que se utilizará en el servidor para automatizar las integraciones continuas del proyecto. |
| Fallos en la integración | <p>Si la integración falla:</p> <ul style="list-style-type: none"> • Corregir los problemas inmediatamente: Si la construcción es corta, hay pocos cambios entre construcciones. Por lo tanto, los problemas son más fáciles de detectar. • Deshacer los cambios: Si los problemas de integración se refieren a flujos fundamentales, los desarrolladores son responsables de deshacer los cambios y regresar el sistema a un estado libre de errores. |

Salidas

- Se ha realizado la integración de un nuevo conjunto de cambios.

Activos asociados

- **Elementos de trabajo**
 - Elemento de Integración Automática

1.7 Actividades del proceso Gestión de defectos



1.7.1 Abrir un defecto

Visión general

Antes de abrir un defecto, se determina siempre si el problema ya se ha identificado. Para abrir un defecto, se registran los detalles en el formulario del sistema de seguimiento de defectos. Se incluyen los pasos para reproducir el defecto, de forma que se pueda examinar y asignarle una prioridad durante la clasificación. Incluir todos los detalles posibles para que el equipo de trabajo determine la mejor forma de tratar el defecto. Asignar un propietario a cada defecto abierto.

Roles participantes

- **Responsables**
 - Ingeniero de Software

Criterios de entrada

- **Comportamiento erróneo del sistema:** La realización de las pruebas ha descubierto un comportamiento del sistema que se ejecuta de forma contraria a lo previsto.

Cuándo

En cualquier momento del desarrollo de una tarea durante el proyecto.

Tareas

Determinar si el defecto está duplicado	<p>Para ello:</p> <ul style="list-style-type: none"> • Antes de abrir un nuevo defecto en el sistema de seguimiento de defectos, examinar la base de datos para ver si ya se ha identificado un defecto similar. • Si el problema ya se ha incluido en este sistema y el defecto ya está abierto, actualizar el formulario de seguimiento de defectos según sea necesario. • Si el problema ya se ha incluido en este sistema y el defecto está cerrado, cambiar el estado del elemento de trabajo Defecto de Cerrado a Reabierto. • Agregar nueva información y documentar los nuevos pasos para reproducir el defecto. • Indicar el número de versión de compilación donde se ha identificado el comportamiento incorrecto.
Registrar un nuevo defecto	<p>Tener en cuenta:</p> <ul style="list-style-type: none"> • Si el defecto no se ha incluido en el sistema de seguimiento de defectos y se puede reproducir, escribir los pasos para reproducirlo y los detalles sobre los resultados de lo que ocurre cuando se siguen los pasos en el elemento de trabajo. • Incluir todos los detalles posibles para ayudar a comprender el problema al equipo responsable de dar prioridad a los defectos. • Donde sea necesario, adjuntar al informe de defectos el caso de prueba, los resultados de la prueba, la configuración de la prueba, la historia de usuario o el requisito de calidad de servicio.
Asignar el defecto	<p>Para asignar el defecto:</p> <ul style="list-style-type: none"> • Determinar el área principal del problema. Si varias áreas pueden ser la causa del problema, seleccionar una de ellas. • Asignar el defecto al propietario del área afectada de la aplicación. • Guardar el defecto en la base de datos de seguimiento de

	defectos.
Salidas	<ul style="list-style-type: none">• El defecto se ha incluido correctamente en el sistema de seguimiento de defectos y contiene la información necesaria para su clasificación.• El defecto se ha asignado correctamente.
Activos asociados	<ul style="list-style-type: none">• Elementos de trabajo<ul style="list-style-type: none">○ Defecto○ Prueba Unitaria○ Prueba de Aceptación



1.7.2 Reproducir el defecto

Visión general

El primer paso para resolver un defecto es intentar reproducirlo. Cuando se reproduce un defecto, puede que se vea claramente que la descripción no incluye información suficiente o correcta, o que el defecto es intermitente. Es necesario establecer una forma confiable de desencadenar el defecto para localizarlo fácilmente e implementar una corrección.

Roles participantes

- **Responsables**
 - Ingeniero de Software

Criterios de entrada

- **Reproducción del defecto:** Existen instrucciones sobre cómo reproducir el defecto en el elemento de trabajo de defecto.

Cuándo

Se ha detectado un defecto y se desea resolverlo.

Tareas

Seguir la descripción del defecto	Para seguir la descripción del defecto: <ul style="list-style-type: none"> • Si el informe del defecto incluye pasos para reproducir el defecto, ejecutarlos. • Si el informe del defecto se ha obtenido en la ejecución de una prueba, seguir el procedimiento de la prueba.
Obtener más información	Tener en cuenta: <ul style="list-style-type: none"> • Si el defecto no se puede reproducir, reunir más información. Utilizar las configuraciones de las pruebas y los informes de defectos anteriores para el área del problema. Además, se pueden consultar los informes de defectos anteriores de otros desarrolladores o del personal de pruebas. • Experimentar con las posibles variaciones de la descripción del defecto para tener otras perspectivas. • Incluir comentarios sobre las mejoras o imperfecciones del defecto en las instrucciones del formulario de seguimiento del defecto.
Abandonar el defecto	Si se sigue sin poder reproducir el defecto, se puede tomar la decisión de establecer su estado como Corregido con un motivo de “No se puede reproducir”.

Salidas

- Se conocen los pasos que hacen que aparezca el defecto con bastante probabilidad.
- El defecto se abandona como no reproducible.

Activos asociados

- **Elementos de trabajo**
 - Defecto



1.7.3 Clasificar defectos

Visión general

Mediante este proceso se clasifican defectos nuevos o que se han vuelto a abrir y se les asigna una prioridad e iteración para trabajar en ellos. El Coordinador del Equipo se encarga de clasificar los defectos basándose en la información que aportan los demás miembros del equipo de desarrollo.

Roles participantes

- **Responsables**
 - Coordinador del Equipo

Criterios de entrada

- **Defecto nuevo o se ha vuelto a abrir:** Al menos un defecto es nuevo o se ha vuelto a abrir.

Cuándo

Se ha detectado un defecto y se desea resolverlo en cualquier momento del desarrollo de un proyecto.

Tareas

Generar lista de defectos	Consultar el sistema de seguimiento de defectos en busca de defectos nuevos o que se hayan vuelto a abrir.
Revisar defectos	Revisar todos los defectos que se encuentren y con la información que se obtenga del desarrollo, de las pruebas y de la administración del producto, determinar la importancia de corregir cada uno de los objetivos del proyecto y la iteración actual.
Asignar prioridad para corregir e iteración	Asignar una prioridad para corregir cada defecto y una iteración para que el equipo de desarrollo pueda programar y ocuparse del defecto cuando corresponda.
Reproducir defectos dudosos	Opcional - Si el informe de un defecto no está claro o el impacto real no es seguro, el equipo debe intentar reproducir el defecto para tener un buen conocimiento de su impacto y reproducibilidad real.

Salidas

- Todos los defectos nuevos y los que se han vuelto a abrir se revisan y se les asigna una prioridad y una iteración.

Activos asociados

- **Elementos de trabajo**
 - Defecto



1.7.4 Encontrar la causa de un defecto

Visión general

Cuando se informa de un defecto, es tarea del desarrollador encontrar su causa. El desarrollador puede emplear una serie de tácticas y herramientas para encontrar la causa principal del defecto. La estrategia para corregir el defecto a menudo se hace más evidente cuando se busca la causa. Utilizar herramientas de desarrollo como registros, un depurador, análisis de código o instrucciones “print” para determinar con exactitud la causa y registrar la evaluación de la ubicación.

Roles participantes

- **Responsables**
 - Ingeniero de Software

Criterios de entrada

- **Configuración de la prueba:** La configuración de la prueba es la misma que se utilizó cuando se encontró el defecto.
- **Código fuente actualizado:** El código fuente actualizado debe estar disponible.

Cuándo

Se ha detectado un defecto y se desea resolverlo en cualquier momento del desarrollo de un proyecto.

Tareas

Aislar área funcional	Mientras se busca el área del código que causa el defecto, eliminar las áreas del código que no le afecten.
Realizar seguimiento de áreas de código afectadas	La utilización de herramientas de depuración, marcan ligeramente el área de código identificada con etiquetas que proporcionan salida visualmente a un depurador o archivo de registro.
Analizar el sistema con herramientas externas	Opcional - Además de realizar un seguimiento del código, utilizar otras herramientas en el entorno para estimar los efectos en el sistema cuando se ejecutan ciertas áreas identificadas.
Determinar el problema con precisión	Ejecutar el programa las veces que sea necesario para determinar exactamente las líneas de código que causan el problema.
Analizar el código	Si no es posible depurar el código debido a que son cantidades a las que no se pueden realizar un seguimiento como el rendimiento o un problema de recursos, analizar el código línea por línea para identificar el problema.

Salidas

- Hay suficiente información desde la ubicación identificada del defecto para desarrollar una corrección.

Activos asociados

- **Elementos de trabajo**
 - Defecto



1.7.5 Reasignar un defecto

Visión general	Es necesario volver a asignar los defectos por falta de información, falta de conocimiento por parte del desarrollador en el área definida en la descripción del defecto o por un nuevo reparto de carga de trabajo. Cuando se reasigna un defecto, agregar descripciones explicando el motivo de la reasignación.	
Roles participantes	<ul style="list-style-type: none"> • Responsables <ul style="list-style-type: none"> ○ Ingeniero de Software 	
Criterios de entrada	<ul style="list-style-type: none"> • Defecto asignado: Se ha asignado previamente el defecto. 	
Cuándo	Se ha detectado un defecto y se desea resolverlo en cualquier momento del desarrollo de un proyecto.	
Tareas	Modificar descripción del defecto	Para modificar la descripción: <ul style="list-style-type: none"> • Escribir el motivo de la reasignación en el campo de Motivo en el formulario de seguimiento de defectos. • Proporcionar información para ayudar al siguiente desarrollador que examine el defecto.
	Actualizar propietario	Actualizar la información del propietario con el desarrollador adecuado.
Salidas	<ul style="list-style-type: none"> • El defecto se vuelve a asignar con un propietario actualizado. 	
Activos asociados	<ul style="list-style-type: none"> • Elementos de trabajo <ul style="list-style-type: none"> ○ Defecto 	



1.7.6 Comprobar una corrección

Visión general	Al comprobar una corrección se ve si el defecto se ha corregido correctamente y si es compatible con la funcionalidad existente. Cuando un desarrollador corrige un defecto, depende del personal de pruebas comprobar que el caso de prueba ahora funciona correctamente. Si el caso de prueba se ejecuta correctamente se puede cerrar el defecto. En caso contrario, el defecto se vuelve a asignar al desarrollador.	
Roles participantes	<ul style="list-style-type: none"> • Responsables <ul style="list-style-type: none"> ○ Ingeniero de Software 	
Criterios de entrada	<ul style="list-style-type: none"> • Versión identificada: Se identifica la versión del sistema en la que se corrige el defecto. • Casos de prueba identificados: Se identifican los casos de prueba que comprueban la funcionalidad relacionada con el defecto. 	
Cuándo	Se ha detectado un defecto y se desea resolverlo en cualquier momento del desarrollo de un proyecto.	
Tareas	Intentar reproducir el defecto	Seguir los pasos originales en la descripción del defecto para intentar reproducirlo.
	Buscar un comportamiento inesperado	Ejecutar la funcionalidad adyacente e intentar buscar un comportamiento inesperado alrededor del defecto. Esto es especialmente importante si la tarea de desarrollo no se implantó completamente.
	Obtener más información sobre el defecto	Opcional - Si la descripción del defecto no está clara o falta información, consultar con el desarrollador o el creador del defecto para obtener más información.
	Reasignar el defecto	Si el caso de prueba no se ejecuta correctamente, el defecto se vuelve a asignar al desarrollador.
Salidas	<ul style="list-style-type: none"> • El defecto se reasigna o se cierra en el sistema de seguimiento de defectos con una descripción actualizada. 	
Activos asociados	<ul style="list-style-type: none"> • Elementos de trabajo <ul style="list-style-type: none"> ○ Defecto 	



1.7.7 Cerrar el defecto

Visión general

Un defecto se cierra por muchos motivos. A menudo los defectos los corrige el desarrollador, los marca como resueltos y finalmente es el Coordinador del Equipo el que los cierra. Los defectos se cierran por varias razones: se aplazan hasta otra versión, se consideran no validos, no se pueden reproducir o están duplicados.

Roles participantes

- **Responsables**
 - Ingeniero de Software

Criterios de entrada

- **Versión y prueba:** Se identifica la versión del sistema con el defecto que se ha arreglado y la prueba que explica el defecto producido.
- **No se ha arreglado el defecto:** No se ha arreglado el defecto preparado para ello debido a restricciones de tiempo e implementación.
- **El defecto es un duplicado:** El defecto es un duplicado y no se puede reproducir o se aplazará para otra versión.

Cuándo

Se ha detectado un defecto y se desea resolverlo en cualquier momento del desarrollo de un proyecto.

Tareas

- | | |
|--------------------------------------|--|
| Actualizar un defecto como duplicado | Opcional - Si el defecto es un duplicado de otro existente, actualizar la descripción del defecto para hacer referencia al defecto ya existente. |
| Actualizar a “NO se corrige” | Opcional - Si el defecto no se va a corregir, actualizar su descripción detallando los motivos. Se puede informar de un defecto que se obtiene como resultado de una restricción del sistema o porque está diseñado para que funcione de esa manera. |
| Actualizar un defecto como corregido | Opcional <ul style="list-style-type: none"> • Si después de realizar los pasos para reproducir el defecto, éste ya no se reproduce, actualizar el defecto con esta información y marcarlo como cerrado. • En la descripción del defecto, documentar la versión que se utilizó para la comprobación. Esto sirve de gran ayuda para corregir futuras regresiones, así como para identificar los defectos que se detecten entre el momento de la corrección y la siguiente compilación. |

Salidas

- Se cierra el defecto en el sistema de seguimiento de defectos con una descripción del comportamiento corregido o una explicación de los motivos por los que no se corrigió.

Activos asociados

- **Elementos de trabajo**
 - Defecto

1.2 ROLES



Descripción

1.2.1 Coordinador del Equipo

Realiza labores de coordinación y dirección de equipos humanos especializados en la realización de actividades propias de un proceso. Realiza la estimación del esfuerzo necesario para llevar a cabo el proyecto, selecciona la estrategia de desarrollo, determina la estructura del mismo, fija el calendario de hitos y entregas y establece la planificación del proyecto.

Es el encargado de dirigir el proyecto, realizando las labores de seguimiento y control del mismo, revisión y evaluación de resultados y coordinación del equipo de proyecto. Se ocupa también de la gestión y resolución de incidencias que puedan surgir durante el desarrollo del proyecto así como de la actualización de la planificación inicial. Entre sus funciones se encuentran la elaboración de los informes de seguimiento y el archivo de la documentación de gestión del proyecto una vez que este ha finalizado.



Descripción

1.2.2 Ingeniero de Software

El Ingeniero de Software tiene como objetivo principal implementar la aplicación según lo especificado dentro del marco de tiempo previsto. Se encarga de especificar las características del diseño físico, calcular el tiempo y los trabajos necesarios para completar cada una de ellas, generar o supervisar la implementación de las mismas, preparar el producto para implementación y proporcionar toda su experiencia en materia de tecnología al equipo.



Descripción

1.2.3 Cliente/Usuario

La responsabilidad del Cliente/Usuario será proporcionar al equipo de trabajo las necesidades que debe satisfacer el sistema software a desarrollar.

1.3 ELEMENTOS DE TRABAJO

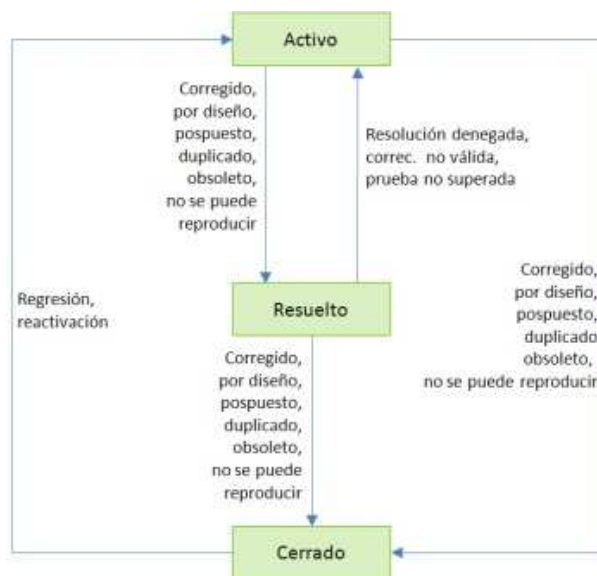


Visión general

1.3.1 Defecto

Un defecto es un elemento de trabajo que comunica que ha existido o existe un problema potencial en el sistema. El objetivo de abrir un defecto es crear un informe preciso de los defectos de forma que permita al lector comprender el impacto total del problema. Las descripciones del informe de defectos deben facilitar el seguimiento de los pasos utilizados cuando se ha encontrado un defecto, permitiendo así reproducirlo fácilmente. Los resultados de las pruebas deben mostrar el problema con claridad. La claridad y la comprensión de esta descripción influyen a menudo en la probabilidad de que se corrija el defecto.

Diagrama de Estados



Estados

Nuevo: Cuando los defectos se detectan en el producto de software, deben documentarse lo antes posible para que los desarrolladores puedan corregirlos. Antes de abrir un informe de defectos, se debe consultar los defectos existentes para asegurarse de que el defecto que se ha descubierto aún no está incluido en el informe.

Transiciones

- **Nuevo a Activo**
 - **Nuevo:** Un defecto se activa como defecto nuevo cuando se crea por primera vez.

Activo: Cuando se descubre un defecto nuevo y se incluye en el sistema de registro de defectos, el elemento de trabajo de defecto se define automáticamente en el estado Activo. Un defecto activo indica que existe un problema que se debe tratar.

Transiciones

- **Activo a Resuelto**
 - **Corregido:** Un defecto se resuelve como Corregido cuando el código modificado está protegido. Vincular el defecto al conjunto de cambios cuando la corrección esté protegida.
 - **Por diseño:** Un defecto adquiere el estado Por diseño si describe un

comportamiento o una condición previstos del sistema.

- **Pospuesto**: Un defecto se resuelve como Pospuesto si no se corrige en la iteración en curso. Se aplazará hasta que se vuelva a evaluar en futuras iteraciones o versiones del producto.
 - **Duplicado**: Un defecto se resuelve como Duplicado si describe el mismo problema que otro defecto. Incluir un vínculo al defecto duplicado para facilitar al autor del defecto la confirmación de la duplicación antes de cerrarlo.
 - **Obsoleto**: Un defecto adquiere el estado Obsoleto si ya no es aplicable al producto. Por ejemplo, si el defecto describe un problema en un área que ya no existe en el producto, se queda obsoleto.
 - **No se puede reproducir**: Un defecto se resuelve como No se puede reproducir, si el desarrollador no puede reproducir el defecto en su equipo.
-

Resuelto: Un defecto tiene el estado Resuelto cuando lo ha solucionado un desarrollador o durante la clasificación. Un defecto se resuelve como Corregido o Por diseño.

Transiciones

• Resuelto a Cerrado

- **Corregido**: Un defecto se cierra como Corregido cuando el autor del defecto ha comprobado que ya se ha corregido en una generación.
- **Por diseño**: Un defecto se cierra como Por diseño si el autor del defecto decide que el defecto describe algo que viene dado por el diseño.
- **Pospuesto**: Un defecto se cierra como Pospuesto si el autor del defecto decide que el defecto se debe aplazar.
- **Duplicado**: Un defecto se cierra como Duplicado si el autor del defecto confirma que el defecto describe el mismo problema que existe con otro defecto.
- **Obsoleto**: Un defecto se cierra como Obsoleto si el autor del defecto decide que el problema descrito ya no es aplicable al producto.
- **No se puede reproducir**: Un defecto se cierra como No se puede reproducir, si el autor del defecto no puede producir un ejemplo operativo del defecto ni proporcionar instrucciones más específicas para reproducir el defecto.

• Resuelto a Activo

- **Resolución denegada**: Un defecto vuelve al estado Activo si la resolución no es aceptable. Proporcionar información específica sobre por qué se ha denegado la resolución para ayudar a resolver el defecto correctamente a la persona encargada.
 - **Corrección no válida**: Un defecto vuelve al estado Activo si la corrección no se realizó de forma correcta. Proporcionar detalles sobre cómo y por qué la corrección no funcionó correctamente.
 - **Prueba no superada**: Un defecto se devuelve al estado Activo si la prueba demuestra que el defecto aún existe. Suministrar detalles sobre la prueba fallida y en qué generación se ha producido el defecto.
-

Cerrado: Un defecto cerrado significa que ya no se va a trabajar más con él en esa versión de producto. Un defecto se cierra después de que se haya comprobado la resolución.

Transiciones

• Cerrado a Activo

- **Regresión**: Si una prueba de regresión indica que el defecto persiste, activarlo y asignarlo a la clasificación. Establecer el campo Motivo en Regresión.
 - **Reactivación**: En caso de algún cambio que haga que el defecto cerrado se reactive nuevamente.
-

Campos	<ul style="list-style-type: none"> • Título: El título proporciona información general concisa del problema que hay que corregir. Debe ser lo bastante descriptivo como para que el equipo de clasificación pueda comprender qué área del producto está afectada y cómo lo está. • Área: El área se utiliza para incluir los defectos en la jerarquía del proyecto según la característica o el equipo. El área debe ser un nodo válido de la jerarquía del proyecto. • Iteración: La iteración identifica en qué iteración se corregirá el defecto. • Asignado a: Este campo identifica la persona a la que se ha asignado el defecto. • Estado: Un defecto puede tener el estado Activo, Resuelto o Cerrado. • Motivo: El motivo por el que un defecto tiene el estado actual. Por ejemplo, un defecto puede estar en estado Resuelto porque ya se ha Corregido. • Descripción: La descripción incluye un área para describir el problema y los pasos para reproducirlo. • Nombre de la prueba: Este campo identifica el nombre de la prueba asociada al defecto. • Id. de prueba: Este campo muestra el identificador de la prueba asociada con el defecto. • Ruta de acceso de la prueba: Este campo muestra la ruta de la prueba asociada con el defecto. • Vínculos: Vínculos a elementos de trabajo relacionados, hipervínculos, conjuntos de cambios o archivos de código fuente. • Datos adjuntos: Adjuntar archivos relacionados que proporcionan documentación adicional sobre el defecto. • Jerarquía: La prioridad relativa con respecto a otros elementos de trabajo. • Clasificación: El resultado de la reunión de clasificación de defectos. Una clasificación en blanco indica que el defecto no está incluido en la clasificación. 		
Actividades	Nuevo	Ingeniero de Software	Abrir un defecto
	Activo	Ingeniero de Software	Reproducir el defecto
			Reasignar un defecto
		Coordinador del Equipo	Clasificar defectos
	Resuelto	Ingeniero de Software	Encontrar la causa de un defecto
			Comprobar una corrección
	Cerrado	Coordinador del Equipo	Cerrar el defecto
Procesos	Gestión de Defectos	Coordinador del Equipo	Abrir un defecto
			Clasificar defectos
			Cerrar el defecto
		Ingeniero de Software	Abrir un defecto
			Reproducir el defecto
			Encontrar la causa de un defecto
			Reasignar un defecto
			Comprobar una corrección

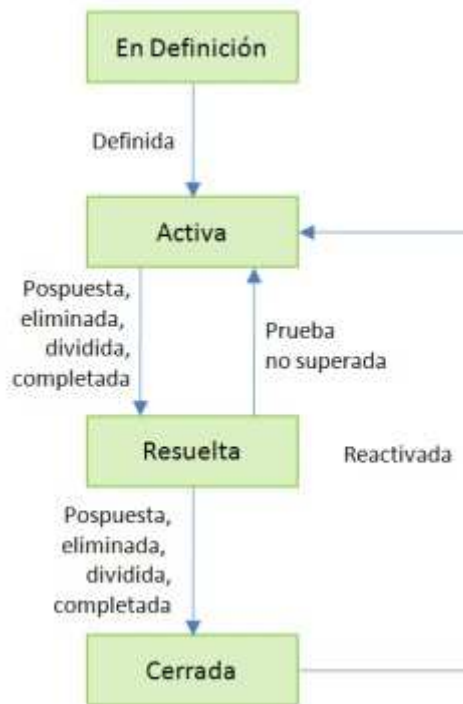


Visión general

1.3.2 Historia de Usuario

Una Historia de Usuario es una representación de un requisito de software escrito en una o dos frases utilizando el lenguaje común del usuario. Las historias de usuario son utilizadas en las metodologías de desarrollo ágiles para la especificación de requisitos (acompañadas de pruebas de aceptación). Cada historia de usuario debe ser limitada y debería poderse escribir sobre una nota adhesiva pequeña.

Diagrama de Estados



Estados

En definición: La nueva historia de usuario se puede crear en la Lista de Historias de Usuario del proyecto.

Transiciones

- **En Definición a Activa**

- **Definida:** Una historia de usuario se define cuando se crea por primera vez.

Activa: Se proporciona un título descriptivo y se rellena el campo Descripción con todos los detalles posibles acerca de la historia de usuario. Cuando la historia de usuario está completamente descrita, se le asigna al Coordinador del Equipo. La historia de usuario permanece en estado Activa mientras se está implementando. El Coordinador del Equipo coordina esfuerzos con otros desarrolladores para implementar la historia de usuario.

Transiciones

- **Activa a Resuelta**

- **Pospuesta:** Una historia de usuario se resuelve como Pospuesta si no se puede implementar en la iteración actual. Una historia de usuario se puede posponer porque el equipo no tiene tiempo suficiente o porque se han descubierto problemas de bloqueo. Actualizar el campo Iteración con la iteración correcta en la que se implementará la historia de usuario. Si la historia de usuario se aplaza a la versión siguiente de producto de software, dejar en blanco el campo

Iteración. Asegurarse de incluir una descripción detallada del motivo de aplazamiento de la historia de usuario y para cuándo está planeada la implementación.

- **Eliminada**: Una historia de usuario se resuelve como Eliminada si ya no se considera necesaria su implementación.
- **Dividida**: Una historia de usuario se resuelve con el valor Dividida cuando una revisión posterior indica que la historia es demasiado grande o que necesita una definición más específica. Cuando se divide una historia de usuario, se crean nuevas historias y se vinculan desde la historia original.
- **Completada**: Una historia de usuario se resuelve como Completada cuando el equipo de desarrollo completa la escritura de código para la historia de usuario. El Coordinador del Equipo asigna la historia de usuario al personal de pruebas.

Resuelta: Cuando la historia de usuario está implementada en código, el coordinador del equipo establece el estado en Resuelta. El coordinador del equipo asigna la historia de usuario al personal de pruebas para que la comprobación pueda iniciarse.

Transiciones

• Resuelta a Cerrada

- **Pospuesta**: Una historia de usuario se cierra como Pospuesta si no se puede implementar en la iteración actual.
- **Eliminada**: Una historia de usuario se cierra con el valor Eliminada si ya no se considera necesaria para implementación.
- **Dividida**: Una historia de usuario se cierra con el valor Dividida, cuando una revisión posterior indica que la historia de usuario es demasiado grande o que necesita una definición más granular.
- **Completada**: Una historia de usuario se cierra como Completada cuando el personal de pruebas indica que ha superado las pruebas correspondientes.

• Resuelta a Activa

- **Prueba no superada**: Si la historia de usuario no supera ninguna prueba, el personal de pruebas debe devolver la historia de usuario al estado Activa y reasignarla al Coordinador del Equipo. Además, el personal de pruebas debe crear defectos apropiados para los defectos de pruebas.

Cerrada: El personal de pruebas cierra una historia de usuario si ésta supera las pruebas correspondientes. Una historia de usuario también se cierra si está pospuesta, eliminada o dividida en más historias de usuario.

Transiciones

• Cerrada a Activa

- **Reactivada**: Una historia de usuario se debe reactivar por un cambio en la funcionalidad.

Campos

- **Título**: El título describe el propósito de la historia de usuario. El título debe ser lo bastante descriptivo como para permitir a los desarrolladores escribir varias iteraciones después de haber sido creada la historia de usuario.
- **ID**: El ID es el número de identificación único asignado a la historia de usuario.
- **Iteración**: La iteración programada es la iteración en que la historia de usuario se implementa en código.
- **Estado**: Una historia de usuario puede tener el estado Sin Comenzar, En Curso o Completada.
- **Prioridad**: Se indica la prioridad de la historia de usuario. La prioridad es un valor numérico determinado a partir de los criterios de Riesgos del proyecto, Importancia para la arquitectura del sistema y Criticidad para el usuario.
- **Tamaño**: Se indica el tamaño estimado de la historia de usuario. El tamaño de la historia está expresado en líneas de código (LOC). El tamaño incluye diseño,

codificación, pruebas e integración.

- **Esfuerzo**: Estimación del esfuerzo de realización de la historia de usuario. Las estimaciones están expresadas en horas de trabajo. El esfuerzo se estima a partir de un consenso de las estimaciones optimista, probable y pesimista de los miembros del equipo.
- **Especificación detallada**: La descripción de la historia de usuario debe contener información detallada sobre la secuencia de pasos que el usuario debe realizar en la historia de usuario

Actividades	<u>En Definición</u>	Coordinador del Equipo	Identificación de necesidades
		Cliente/Usuario	Identificación de necesidades
	<u>Activa</u>	Coordinador del Equipo	Priorización de historias Estimación de historias Determinación de alcance de iteración Identificación de tareas Especificación de historias
		Cliente/Usuario	
		Ingeniero de Software	Priorización de historias Estimación de historias Determinación de alcance de iteración Identificación de tareas Especificación de historias
	<u>Resuelta</u>	Ingeniero de Software	Codificación
	<u>Cerrada</u>	Coordinador del Equipo	Ejecución de pruebas de aceptación
		Cliente/Usuario	Ejecución de pruebas de aceptación
		Ingeniero de Software	Ejecución de pruebas de aceptación
Procesos	<u>Gestión de Historias de Usuario</u>	Coordinador del Equipo	Identificación de necesidades Priorización de historias Estimación de historias Determinación de alcance de iteración Ejecución de pruebas de aceptación
		Cliente/Usuario	Identificación de necesidades Especificación de historias Ejecución de pruebas de aceptación
		Ingeniero de Software	Priorización de historias Estimación de historias Determinación de alcance de iteración Especificación de historias Ejecución de pruebas de aceptación
	<u>Planificación</u>	Coordinador del Equipo	Identificación de tareas
		Ingeniero de Software	Identificación de tareas
	<u>Desarrollo</u>	Ingeniero de Software	Codificación



Visión general

1.3.3 Requisito de Calidad de Servicio

Los requisitos de calidad de servicio documentan características del sistema, como rendimiento, carga, disponibilidad, esfuerzo, accesibilidad, operatividad y capacidad de mantenimiento. Estos requisitos normalmente adquieren la forma de restricciones bajo las cuales debe funcionar el sistema.

Diagrama de Estados



Estados

En Definición: Los requisitos de calidad de servicio se pueden crear en la lista de requisitos de calidad de servicio ubicada en la carpeta de requisitos de la biblioteca de documentos.

Transiciones

- **En Definición a Activo**

- **Definido:** Un requisito de calidad de servicio se define como un requisito nuevo cuando se crea por primera vez.

Activo: El coordinador del equipo crea el requisito, proporciona un título descriptivo y rellena el campo de descripción con todos los detalles posibles acerca del requisito. Cuando el requisito está completamente descrito, el coordinador del equipo le asigna un ingeniero de software. El requisito permanece en estado Activo mientras se está implementando. El ingeniero de software coordina esfuerzos con otros desarrolladores para implementar el requisito.

Transiciones

- **Activo a Resuelto**

- **Pospuesto:** Un requisito de calidad de servicio se resuelve como Pospuesto si no se puede implementar en la iteración actual. Un requisito se puede aplazar porque el equipo no tiene tiempo suficiente o porque se han descubierto problemas de bloqueo. Actualice el campo Iteración con la iteración correcta en la que se implementará el requisito. Si el requisito se aplaza a la versión de producto software siguiente, deje en blanco el campo Iteración. Asegúrese de incluir una descripción detallada del motivo de aplazamiento del requisito y para cuándo está planeada la implementación.
- **Eliminado:** Un requisito de calidad de servicio tiene el valor Eliminado, si ya

no se considera necesario para la implementación.

- **Completado:** Un requisito de calidad de servicio se resuelve como Completado cuando el equipo de desarrollo completa la escritura de código para el requisito. El ingeniero de software asigna el requisito al personal de pruebas.

Resuelto: Cuando el requisito de calidad de servicio está implementado en código, el Coordinador del Equipo establece el estado en Resuelto. El Coordinador del Equipo asigna el requisito al personal de pruebas para que la comprobación pueda iniciarse.

Transiciones

• Resuelto a Cerrado

- **Pospuesto:** Un requisito de calidad de servicio se cierra como Pospuesto si no se puede implementar en la iteración actual.
- **Eliminado:** Un requisito de calidad de servicio se cierra con el valor Eliminado si ya no se considera necesario para la implementación.
- **Completado:** Un requisito de calidad de servicio se cierra como Completado cuando el personal de pruebas indica que ha superado las pruebas correspondientes.

• Resuelto a Activo

- **Prueba no superada:** Un requisito de calidad del servicio se devuelve al estado Activo si produce un defecto en una o más comprobaciones. El evaluador debe reasignarlo al responsable de desarrollo original. Además, el personal de pruebas debe crear defectos apropiados para los defectos de pruebas.

Cerrado: El personal de pruebas cierra un requisito de calidad de servicio si supera las pruebas correspondientes. Un requisito también se cierra si está pospuesto, eliminado o dividido en más requisitos.

Transiciones

• Cerrado a Activo

Reactivado: Un requisito de calidad de servicio pospuesto se reactiva cuando la iteración asignada se inicia. Si aún se debe escribir el requisito, asígnelo al coordinador del equipo. Si el requisito está listo para implementación, asígnelo a un ingeniero de software. Si se reactiva un requisito eliminado, siga el mismo proceso que para un requisito pospuesto

Campos

-
- **Título:** El título del requisito de calidad de servicio debe ser lo más descriptivo posible.
 - **Id:** El Id. es el número de identificación único asignado al requisito de calidad de servicio.
 - **Iteración:** La iteración en la que el requisito de calidad de servicio está implementado en código.
 - **Tipo:** Existen cinco tipos de requisitos de calidad de servicio: Carga, Esfuerzo, Rendimiento, Plataforma, Seguridad y Otros.
 - **Asignado a:** La persona actual a la que se asigna el requisito de calidad de servicio.
 - **Estado:** Un requisito de calidad de servicio puede tener el estado Activo, Resuelto o Cerrado.
 - **Descripción:** La descripción ofrece un área para describir el requisito de calidad de servicio. Proporcione los máximos detalles posibles para asegurarse de que el desarrollador puede implementar el requisito y de que el ingeniero de pruebas puede comprobarlo.
 - **Prioridad:** Indica el grado de importancia del requisito de calidad de servicio respecto a todos los requisitos del producto de software.
-

Actividades	<u>En Definición</u>	Coordinador del Equipo	Identificación de requisitos de calidad del servicio
		Ingeniero de Software	Identificación de requisitos de calidad del servicio
	<u>Activo</u>	Cliente/Usuario	Creación de pruebas de aceptación
	<u>Resuelto</u>	Ingeniero de Software	Diseño
	<u>Cerrado</u>	Coordinador del Equipo	Ejecución de pruebas de aceptación
		Cliente/Usuario	Ejecución de pruebas de aceptación
		Ingeniero de Software	Ejecución de pruebas de aceptación
Procesos	<u>Gestión de Historias de Usuario</u>	Coordinador del Equipo	Identificación de requisitos de calidad del servicio
			Ejecución de pruebas de aceptación
		Cliente/Usuario	Creación de pruebas de aceptación
			Ejecución de pruebas de aceptación
		Ingeniero de Software	Identificación de requisitos de calidad del servicio
	<u>Desarrollo</u>	Ingeniero de Software	Ejecución de pruebas de aceptación
			Diseño



**Visión
general**

**Diagrama de
Estados**

1.3.4 Recurso

Un recurso es un medio, tanto humano como material, necesario para la realización de las actividades de un proyecto.



Estados

Identificado: Este estado se produce cuando se identifica un recurso como necesario para la realización de un proyecto.

Transiciones

- **Identificado a Activo**
 - **Recurso incorporado:** Esta transición se produce cuando un recurso empieza a trabajar en el proyecto bajo las condiciones establecidas en el plan.

Activo: Este estado representa el periodo de tiempo en el que un recurso está dedicado al proyecto.

Transiciones

- **Activo a Liberado**
 - **Actividad finalizada:** Esta transición se produce cuando las responsabilidades del recurso en el proyecto se finalizan.

Liberado: Este estado representa a un recurso con disponibilidad para su asignación a otro proyecto.

Campos

- **Nombre:** Texto descriptivo e identificativo del recurso.
- **Propietario:** Unidad organizativa a la que pertenece el recurso.
- **Unidades disponibles:** Cantidad de tiempo (porcentaje de dedicación) que el recurso está disponible para el proyecto.
- **Coste:** Coste del recurso por unidad de tiempo.
- **Estado:** Estado en el que se encuentra el recurso con relación al proyecto.

Actividades

<u>Identificado</u>	Coordinador del Equipo	Identificación de tareas Identificación de responsabilidades Determinación del esfuerzo de las tareas Ajustes al plan de iteración Celebrar reunión de planificación
---------------------	------------------------	--

Procesos	<u>Activo</u>	Ingeniero de Software	Ajustes al plan de iteración Identificación de tareas Identificación de responsabilidades Determinación del esfuerzo de las tareas Celebrar reunión de planificación
		Coordinador del Equipo	Recopilar la información de seguimiento
		Ingeniero de Software	Recopilar la información de seguimiento Publicar el informe de seguimiento
		Coordinador del Equipo	Recopilar la información de seguimiento
		Ingeniero de Software	Recopilar la información de seguimiento Publicar el informe de seguimiento
	<u>Liberado</u>	Coordinador del Equipo	Identificación de tareas Identificación de responsabilidades Determinación del esfuerzo de las tareas Ajustes al plan de iteración Celebrar reunión de planificación
		Ingeniero de Software	Identificación de tareas Identificación de responsabilidades Determinación del esfuerzo de las tareas Celebrar reunión de planificación
		Coordinador del Equipo	Recopilar la información de seguimiento
		Ingeniero de Software	Recopilar la información de seguimiento Celebrar reunión de seguimiento periódico Publicar el informe de seguimiento

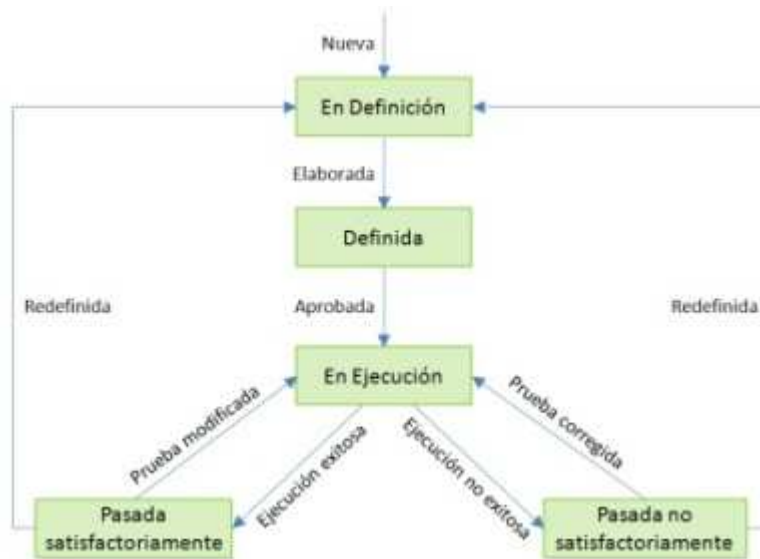


Visión general

1.3.5 Prueba de Aceptación

Durante una iteración, para las historias de usuario seleccionadas durante la planificación se identificarán pruebas de aceptación que permitan que el cliente valide y verifique el funcionamiento correcto de las historias de usuario definidas. Una historia de usuario no se considera completa hasta que haya pasado su prueba de aceptación. Nuevas pruebas de aceptación deben ser creadas en cada iteración o el equipo de desarrollo reportará un progreso nulo.

Diagrama de Estados



Estados

En Definición: Una nueva prueba de aceptación se crea a partir de una historia de usuario.

Transiciones

- **Nueva a En Definición**
 - **Nueva:** Se crea una nueva prueba de aceptación.
- **En Definición a Definida**
 - **Elaborada:** Una prueba de aceptación se elabora cuando se le asigna a una historia de usuario.

Definida: Una prueba de aceptación se considera definida cuando se están creando los datos de prueba y los resultados esperados de su ejecución.

Transiciones

- **Definida a En Ejecución**
 - **Aprobada:** Los datos de prueba y los resultados esperados de su ejecución han sido completamente definidos.

En Ejecución: Las pruebas de aceptación se ejecutan, realizando cada paso especificado en la prueba. Se comparan los resultados de las pruebas con los resultados que se esperaban y se registran los resultados de la prueba.

Transiciones

- **En Ejecución a Pasada satisfactoriamente**
 - **Ejecución exitosa:** La ejecución de la prueba está libre de defectos y cumple los resultados esperados.
- **En Ejecución a No pasada satisfactoriamente**

- **Ejecución no exitosa:** La ejecución de la prueba presenta algún tipo de defecto.

Pasada satisfactoriamente: La prueba de aceptación pasa satisfactoriamente los resultados esperados demostrando que la historia de usuario asociada se ha desarrollado correctamente.

Transiciones

- **Pasada satisfactoriamente a En Ejecución**
 - **Prueba modificada:** Si se tiene un nuevo conjunto de datos de prueba, se debe ejecutar nuevamente la prueba de aceptación.
- **Pasada satisfactoriamente a En Definición**
 - **Redefinida:** Si se realizan cambios en la definición de los datos de prueba o en la secuencia de pasos, la prueba debe ser redefinida.

No pasada satisfactoriamente: La prueba de aceptación no ha sido superada satisfactoriamente al comparar los resultados de las pruebas con los resultados que se esperaban. Se debe abrir un defecto si los resultados no satisfacen las expectativas.

Transiciones

- **No pasada satisfactoriamente a En Ejecución**
 - **Prueba corregida:** Los defectos presentados por la prueba han sido corregidos y la prueba debe ser nuevamente ejecutada para verificar que los defectos han sido eliminados.
- **No pasada satisfactoriamente a En Definición**
 - **Redefinida:** Para corregir los defectos detectados se debe definir nuevamente los casos de prueba con sus datos y resultados esperados.

Campos

- **ID:** El ID es el número de identificación único asignado a la prueba de aceptación.
- **Propósito:** Motivación o causa para realizar la prueba de aceptación y qué se debe comprobar durante la prueba.
- **Pasos:** Lista de pasos que se deben realizar para ejecutar la prueba de aceptación.
- **Fecha:** Fecha de realización de la prueba de aceptación.
- **Resultados:** Describe los resultados obtenidos con la realización de la prueba de aceptación comparándolos con los resultados que se esperaban.

Actividades

<u>En Definición</u>	Cliente/Usuario	Creación de pruebas de aceptación
<u>Definida</u>	Cliente/Usuario	Creación de pruebas de aceptación
<u>En Ejecución</u>	Coordinador del Equipo	Ejecución de pruebas de aceptación
	Cliente/Usuario	Ejecución de pruebas de aceptación
	Ingeniero de Software	Ejecución de pruebas de aceptación
<u>Pasada satisfactoriamente</u>	Coordinador del Equipo	Ejecución de pruebas de aceptación
	Cliente/Usuario	Ejecución de pruebas de aceptación
	Ingeniero de Software	Ejecución de pruebas de aceptación
<u>No pasada satisfactoriamente</u>	Coordinador del Equipo	Ejecución de pruebas de aceptación

Procesos		Cliente/Usuario	Ejecución de pruebas de aceptación
		Ingeniero de Software	Ejecución de pruebas de aceptación
	<u>Gestión de Historias de Usuario</u>	Coordinador del Equipo	Ejecución de pruebas de aceptación
		Cliente/Usuario	Creación de pruebas de aceptación Ejecución de pruebas de aceptación
		Ingeniero de Software	Ejecución de pruebas de aceptación



Visión general

1.3.6 Prueba Unitaria

Las pruebas unitarias deben probar todas las clases del sistema. Las pruebas deben crearse antes de la codificación. Estas pruebas automatizadas pueden ahorrar mucho esfuerzo en encontrar y corregir defectos. Se requiere que todo el código pase todas las pruebas unitarias antes que sea liberado para asegurar que toda la funcionalidad siempre sea correcta.

Diagrama de Estados



Estados

En Definición: Durante este estado se identifican las pruebas unitarias para que la tarea de desarrollo cubra la mayor funcionalidad posible.

Transiciones

- **Nueva a En Definición**
 - **Nueva:** Una prueba se activa como Nueva cuando se crea por primera vez.
- **En Definición a Programada**
 - **Definida:** Se ha definido una clase de prueba y se ha creado un método para crear y configurar el objeto a probar.

Programada: Durante este estado se escribe la prueba.

Transiciones

- **Programada a En Ejecución**
 - **Aprobada:** La clase de prueba se ha escrito.

En Ejecución: La prueba unitaria se ejecuta para verificar su comportamiento. Si el código falla, corregir el código fuente y repetir la prueba. Si el código no falla, pasar a la siguiente prueba.

Transiciones

- **En Ejecución a Clase definida exitosamente**
 - **Ejecución exitosa:** Las pruebas se han ejecutado en forma exitosa.

Clase definida exitosamente: Después que se ha verificado que las pruebas unitarias se han ejecutado exitosamente, se aceptan y se debe registrar el código fuente y la prueba en el servidor de configuración.

Transiciones

- **Clase definida exitosamente a En Ejecución**
 - **Clase modificada:** Se han realizado cambios en la clase de prueba y debe ser ejecutada nuevamente.
- **Clase definida exitosamente a Programada**
 - **Programación modificada:** Se han realizado cambios que afectan la programación de la clase de prueba.

Campos	<ul style="list-style-type: none">• <u>ID</u>: El ID es el número de identificación único asignado a la prueba unitaria.• <u>Propósito</u>: Motivación o causa para realizar la prueba unitaria.• <u>Clase</u>: Nombre de la clase que se está probando.• <u>Método de prueba</u>: Nombre del método para crear y configurar el objeto a probar.• <u>Resultados</u>: Descripción de los resultados obtenidos con la realización de la prueba unitaria.		
Actividades	<u>En Definición</u>	Cliente/Usuario	Definición de pruebas unitarias
		Ingeniero de Software	Definición de pruebas unitarias
	<u>Programada</u>	Cliente/Usuario	Definición de pruebas unitarias
		Ingeniero de Software	Definición de pruebas unitarias
			Identificar áreas susceptibles de reorganización
	<u>En Ejecución</u>	Ingeniero de Software	Codificación
		Aplicar la refactorización	
		Realizar pruebas unitarias	
	<u>Clase definida exitosamente</u>	Ingeniero de Software	Codificación
Procesos	<u>Desarrollo</u>	Cliente/Usuario	Definición de pruebas unitarias
		Ingeniero de Software	Definición de pruebas unitarias
			Codificación
	<u>Refactoring</u>	Ingeniero de Software	Identificar áreas susceptibles de reorganización
		Aplicar la refactorización	
		Realizar pruebas unitarias	



Visión general

1.3.7 Clase

Una clase es una descripción de un conjunto de objetos que comparten los mismos atributos, operaciones, métodos, relaciones y significado. La posibilidad de abstraer métodos y descripciones de datos comunes de un conjunto de objetos y almacenarlos en una clase es esencial para el desarrollo de software orientado a objetos. Definir clases significa situar código reutilizable en un depósito común en lugar de volver a expresarlo una y otra vez.

Diagrama de Estados



Estados

En definición: Una clase se considera en En definición cuando se están identificando sus características estáticas y dinámicas.

Transiciones

- **Nueva a En Definición**
 - **Nueva:** Una clase se activa como clase Nueva cuando se crea por primera vez.
- **En Definición a Diseñada**
 - **Definida:** Un clase se considera diseñada cuando se han definido sus atributos y responsabilidades.

Diseñada: Una clase se encuentra en estado Diseñada cuando se ha realizado un definición completa de sus atributos y métodos.

Transiciones

- **Diseñada a Programada**
 - **Paso a desarrollo:** Una clase pasa al estado de Programada cuando se comienzan las actividades de codificación necesarias para su implementación.

Programada: Una clase se encuentra en estado Programada cuando se tiene una implementación en código de la clase.

Transiciones

- **Programada a Probada**
 - **Finalizada:** Cuando se realizan pruebas acerca de la funcionalidad de la clase.

Probada: Una clase está en estado Probada cuando el comportamiento esperado de la clase está de acuerdo con la especificación de historias de usuario.

Transiciones			
<ul style="list-style-type: none"> • Probada a Programada <ul style="list-style-type: none"> ○ Reprogramación: Si se realizan cambios a nivel de código de la clase, estos cambios debe codificarse. • Probada a Diseñada <ul style="list-style-type: none"> ○ Rediseño: Si se realizan cambios cuyo origen está en el diseño, la clase debe ser rediseñada. 			
Campos	<ul style="list-style-type: none"> • Identificador: Es el número de identificación único asignado a la clase. • Nombre: Texto descriptivo e identificativo de la clase. • Descripción: La descripción ofrece un área para describir detalladamente el propósito de la clase y requisitos para su implementación. • Paquete: Nombre del paquete o módulo al que pertenece la clase. • Atributos: Lista de propiedades de la clase junto con su tipo de dato. • Métodos: Lista de operaciones de la clase. • Asignada a: La persona actual a la que se ha asignado la clase. • Estado: Estado en el que se encuentra la clase con relación al proyecto. 		
Actividades	<u>En definición</u>	Ingeniero de Software	Diseño
		Cliente/Usuario	Definición de pruebas unitarias
			Definición de pruebas unitarias
	<u>Diseñada</u>	Ingeniero de Software	Diseño
			Identificar áreas susceptibles de reorganización
	<u>Programada</u>	Ingeniero de Software	Codificación
	<u>Probada</u>	Ingeniero de Software	Aplicar la refactorización
			Codificación
			Aplicar la refactorización
Procesos	<u>Desarrollo</u>	Cliente/Usuario	Definición de pruebas unitarias
			Codificación
		Ingeniero de Software	Diseño
			Definición de pruebas unitarias
	<u>Refactoring</u>	Ingeniero de Software	Identificar áreas susceptibles de reorganización
			Aplicar la refactorización



Visión general

1.3.8 Tarea

Un elemento de trabajo de tarea comunica la necesidad de realizar algún trabajo. Cada función tiene sus propios requisitos para una tarea. Por ejemplo, el desarrollador utiliza tareas de desarrollo para asignar trabajo derivado de las historias de usuario o los requisitos de calidad de servicio a los propietarios de los componentes. El ingeniero de software utiliza las tareas de prueba para asignar el trabajo de escribir y ejecutar casos de pruebas. Una tarea también se puede utilizar para señalar regresiones o para sugerir que se realicen pruebas exploratorias. Por último, se puede utilizar genéricamente una tarea para asignar trabajo dentro de un proyecto. En el formulario de elemento de trabajo, ciertos campos se utilizan sólo en los casos en que una tarea se refiere a una función determinada.

Diagrama de Estados



Estados

En Definición: Las tareas en definición representan trabajo sobre el cual un acuerdo aún no ha sido alcanzado. Las tareas en definición son priorizadas y luego aceptadas o rechazadas.

Transiciones

- **En Definición a Activa**
 - **Acordada:** Una tarea es Acordada cuando es revisada y se determina que el trabajo representado por la tarea es apropiado para la iteración actual.
- **En Definición a Cerrada**
 - **Rechazada:** Una tarea es Cerrada cuando es revisada y se determina que el trabajo representado por la tarea no es apropiado debido a algún factor como información faltante o análisis incorrecto.

Activa: Una tarea Activa indica si hay algún elemento de trabajo a ser realizado. Todas las tareas activas deben tener asignado un propietario.

Transiciones

- **Activa a Finalizada**
 - **Completada:** Una tarea es completada cuando se finaliza el trabajo representado por la tarea, requiriéndose en ese momento una revisión por medio de pruebas o revisión de pares.
- **Activa a Cerrada**

Cancelada: Una tarea es Cancelada si el trabajo que representa ya no es necesario para terminar el producto.
- **Activa a En Definición**
 - **Redefinir alcance:** Si una tarea ha sido activada para investigación, debe

	regresar al estado de En Definición para ser priorizada una vez la investigación se complete.		
	Cerrada: Una tarea se considera cerrada cuando después de su definición la actividad no ha sido aceptada en la planificación o cancelada antes de su finalización. Transiciones <ul style="list-style-type: none"> • Cerrada a Activa <ul style="list-style-type: none"> ○ Reactivada: Una tarea es Reactivada si es aplazada en una iteración previa y puede ser completada en la iteración actual. 		
	Finalizada: Una tarea Finalizada está completada si los productos de salida no son aceptables, la tarea retorna al estado Activa para trabajo adicional. Transiciones <ul style="list-style-type: none"> • Finalizada a Activa <ul style="list-style-type: none"> ○ Revisión no satisfactoria: Una tarea es finalizada como Activa si la revisión o prueba no acepta la salida de la tarea. 		
Campos	<ul style="list-style-type: none"> • Título: El título proporciona información general concisa de la tarea que se va a realizar. Debe ser lo bastante descriptivo como para que el equipo pueda comprender qué área del producto está afectada y en qué sentido. • ID: El ID es el número de identificación único asignado a una tarea. • Historia: Historia de usuario relacionada con la tarea. La tarea está relacionada con áreas principales del sistema afectadas por una historia de usuario. • Responsable: Persona a la que está asignada la tarea actualmente. • Estado: Una tarea se puede encontrar en el estado Sin Comenzar, En Curso o Completada. • Esfuerzo: Estimación del esfuerzo de realización de la tarea. La estimación del esfuerzo de la tarea está expresada en horas de trabajo. 		
Actividades	<u>En Definición</u>	Coordinador del Equipo	Identificación de tareas Identificación de responsabilidades Determinación del esfuerzo de las tareas Ajustes al plan de iteración Celebrar reunión de planificación Celebrar reunión de seguimiento periódico
		Cliente/Usuario	Celebrar reunión de seguimiento periódico
		Ingeniero de Software	Identificación de tareas Identificación de responsabilidades Determinación del esfuerzo de las tareas Celebrar reunión de planificación Celebrar reunión de seguimiento periódico
	<u>Activa</u>	Coordinador del Equipo	Recopilar la información de seguimiento Celebrar reunión de seguimiento periódico Publicar el informe de seguimiento
		Cliente/Usuario	Celebrar reunión de seguimiento periódico

Procesos	<u>Cerrada</u>	Ingeniero de Software	Recopilar la información de seguimiento Celebrar reunión de seguimiento periódico
		Coordinador del Equipo	Recopilar la información de seguimiento Celebrar reunión de seguimiento periódico Publicar el informe de seguimiento
		Cliente/Usuario	Celebrar reunión de seguimiento periódico
	<u>Finalizada</u>	Ingeniero de Software	Recopilar la información de seguimiento Celebrar reunión de seguimiento periódico
		Coordinador del Equipo	Recopilar la información de seguimiento Celebrar reunión de seguimiento periódico Publicar el informe de seguimiento
		Cliente/Usuario	Celebrar reunión de seguimiento periódico
		Ingeniero de Software	Recopilar la información de seguimiento Celebrar reunión de seguimiento periódico
	<u>Planificación</u>	Coordinador del Equipo	Identificación de tareas Identificación de responsabilidades Determinación del esfuerzo de las tareas Ajustes al plan de iteración Celebrar reunión de planificación
		Ingeniero de Software	Identificación de tareas Identificación de responsabilidades Determinación del esfuerzo de las tareas Celebrar reunión de planificación
		Coordinador del Equipo	Recopilar la información de seguimiento Celebrar reunión de seguimiento periódico Publicar el informe de seguimiento
		Cliente/Usuario	Celebrar reunión de seguimiento periódico
		Ingeniero de Software	Recopilar la información de seguimiento Celebrar reunión de seguimiento periódico



Visión general

1.3.9 Elemento de Integración Automática

La integración es el proceso de ensamblar los componentes individuales desarrollados y construirlos en el producto software final. Los elementos de integración deben pasar también las pruebas unitarias. Es preferible que se pueda automatizar este proceso y de esta manera acceder a un ejecutable confiable del sistema.

Diagrama de Estados



Estados

Scripts en construcción: Durante el estado de Scripts en Construcción los diferentes elementos de integración automática se obtienen del sistema de gestión de configuración, se compilan y enlazan, y se ejecutan pruebas automatizadas para comprobar su correcto funcionamiento.

Transiciones

- **Nuevo a Scripts en construcción**
 - **Nuevo:** Los elementos de integración son obtenidos del sistema de gestión de configuración.
- **Scripts en construcción a Scripts en ejecución**
 - **Construcción exitosa:** La construcción del sistema fue correcta.

Scripts en ejecución: Durante este estado se programan las tareas de despliegue en el entorno en que se va a ejecutar el proyecto.

Transiciones

- **Scripts en ejecución a Integración exitosa**
 - **Ejecución exitosa:** La integración se realizó en forma correcta y la versión del sistema obtenida es estable.
- **Scripts en ejecución a Integración no exitosa**
 - **Ejecución no exitosa:** La integración continua presenta errores en su configuración.

Integración satisfactoria: Los componentes individuales han sido ensamblados correctamente para formar el producto de software en el entorno de ejecución.

Transiciones

- **Integración satisfactoria a Scripts en Construcción**
 - **Cambios en construcción:** Nuevos componentes han sido solicitados a nivel de construcción del producto.
- **Integración satisfactoria a Scripts en Ejecución**
 - **Cambios en integración:** Nuevos componentes han sido solicitados a nivel de integración del producto.

Integración no satisfactoria: Los componentes individuales no han sido ensamblados correctamente y presentan defectos a ser corregidos para formar un producto de

software en el entorno de ejecución.

Transiciones

- **Integración no satisfactoria a Scripts en Construcción**
 - **Scripts modificados:** Los defectos detectados han sido corregidos y el sistema está listo para repetir el proceso de construcción.
- **Integración no satisfactoria a Scripts en Ejecución**
 - **Scripts modificados:** Los defectos detectados han sido corregidos y el sistema está listo para repetir el proceso de ejecución.

Campos	<ul style="list-style-type: none"> • <u>Script de construcción</u>: Fichero que configura la realización de la construcción del sistema. • <u>Script de integración</u>: Fichero que permite la automatización de las tareas de integración que se encuentra localizado en el servidor de control de versiones. • <u>Registro de resultados</u>: Registro que mantiene información del resultado de todos los intentos de integración automática. 		
Actividades	<u>Scripts en construcción</u>	Ingeniero de Software	Editar fichero de construcción del ejecutable
	<u>Scripts en ejecución</u>	Ingeniero de Software	Editar el fichero de configuración de integración continua
	<u>Integración satisfactoria</u>	Ingeniero de Software	Editar fichero de construcción del ejecutable
	<u>Integración no satisfactoria</u>	Ingeniero de Software	Editar el fichero de configuración de integración continua
Procesos	<u>Integración Automática</u>	Ingeniero de Software	Editar fichero de construcción del ejecutable
			Editar el fichero de configuración de integración continua

ANEXO C - MANUAL DE INSTALACIÓN DE MEDIAWIKI

En este manual se describe un conjunto de procedimientos para realizar la instalación del software **MediaWiki 1.16.0** y un conjunto de extensiones adicionales para implementar la PAL-Wiki.

1. REQUISITOS PREVIOS

Los requisitos para la instalación de MediaWiki incluyen:

- Un servidor web para enviar las páginas generadas hacia un navegador web. La mayoría de las instalaciones utilizan **Apache**, aunque también se puede utilizar **Internet Information Server 6.0**.
- **PHP** versión 5.0 o superior, se requiere para ejecutar el software MediaWiki.
- Un servidor de base de datos es requerido para almacenar las páginas y los datos del sitio. Servidores de bases de datos como **MySQL 4.0** o superior, **PostgreSQL 8.1** o superior o **SQLite**.

Estos requisitos previos pueden ser instalados individualmente o a través de paquetes como **AppServ**, **Xampp** o **Wamp**.

A nivel de requisitos de hardware se requieren como mínimo 256Mb de RAM y 40Mb de almacenamiento.

2. DESCARGAR Y DESCOMPRIMIR MEDIAWIKI

Se debe descargar la última versión de MediaWiki desde la página <http://www.mediawiki.org/wiki/MediaWiki> y extraer el fichero en un directorio accesible por web.

3. CREAR LA BASE DE DATOS

Se debe crear una base de datos que apoye el motor del MediaWiki. Por ejemplo, se puede ingresar a MySQL desde AppServ y crear una base de datos denominada: *palwiki*.

4. INSTALACIÓN DE MEDIAWIKI

Para realizar la instalación, se debe acceder al servidor web con un navegador de Internet y abrir la ruta donde se ha descomprimido MediaWiki. Si hemos extraído MediaWiki en la carpeta raíz de los sitios web, se habrá creado una carpeta llamada **mediawiki-1.16.0**. Se puede cambiar el nombre de esta carpeta, por el nombre de la wiki, en este caso: *palwiki*. Por lo tanto, se debe ir a la siguiente dirección:

<http://localhost/palwiki>

Surge una pantalla inicial de MediaWiki para que se configure la aplicación. Se debe hacer clic en el enlace “**set up the wiki**” (ver Figura C-1).

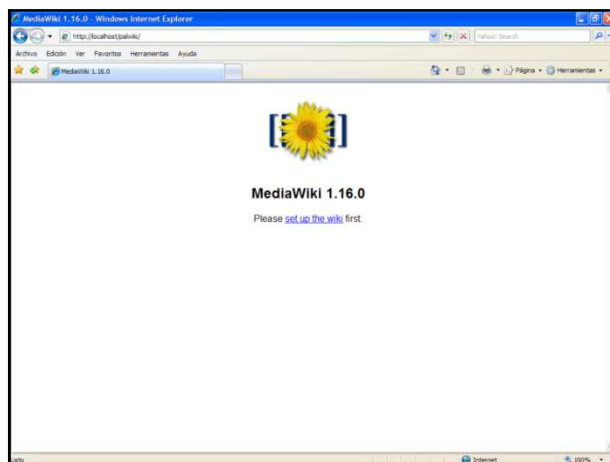


Figura C-1. Inicio de la instalación.

El instalador realiza unas comprobaciones para verificar que el servidor cumple los requisitos necesarios para instalar la aplicación. Si todo está correcto se verá el mensaje: **“Environment Checked. You can install MediaWiki”** (ver Figura C-2). Si el servidor no satisface los requisitos, no se puede instalar MediaWiki. En tal caso, habrá que actualizar o instalar las aplicaciones que se indiquen.

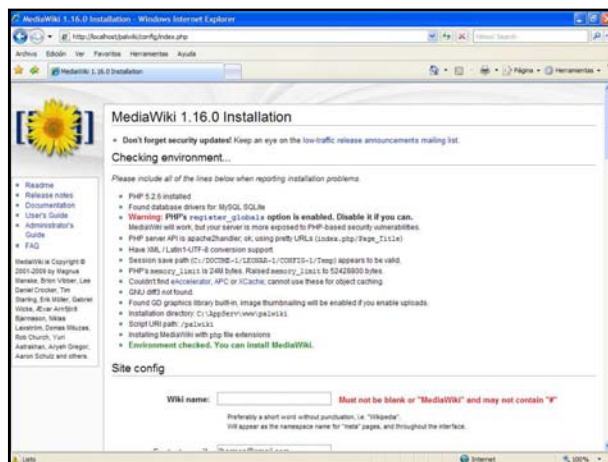


Figura C-2. Verificación del ambiente de instalación.

En la parte inferior de la pantalla se ingresan los datos de Configuración del sitio (**Site Config**, ver Figura C-3). En este apartado se configuran los siguientes parámetros:

- **Wikiname:** Nombre del sitio. Para este caso: *palwiki*.
- **Contact-email:** e-mail del administrador.
- **Language:** Idioma. Para este caso: *español*.
- **Copyright/license:** Tipo de licencia dependiendo si los contenidos tienen copyright o pueden ser copiarlos. Se coloca como privada ya que la wiki no es pública.
- **Admin username/password:** Nombre del usuario administrador del MediaWiki y su contraseña (habrá que ingresar la contraseña dos veces).
- **Shared memory caching:** Si se desea activar el sistema de cacheo de páginas para acelerar el sitio cuando el número de usuarios es muy elevado. Como es un sitio privado, no merece la pena cachear.

Figura C-3. Configuración del sitio.

A continuación, se debe configurar la notificación por email (**Email, email notification and authentication setup**) como se observa en la Figura C-4. MediaWiki permite la interactividad mediante correo electrónico. También envía correos automáticos cuando se han actualizado las páginas que se han introducido en listas de seguimiento. Para utilizar las funciones de email, es necesario disponer de un servidor SMTP. Para el caso de la PAL-Wiki, se deshabilitarán estas funciones de email.

Figura C-4. Configuración de notificación por e-mail.

Por último, se realiza la configuración de la base de datos (**Database config**) como se observa en la Figura C-5.

- **Database type:** Tipo de base de datos. Para este caso, *MySQL*.
- **Database host:** PC que tiene la base de datos. Para este caso, *localhost*.
- **Database name:** Nombre de la base de datos creada en MySQL: *palwiki*.

- **DB username:** Nombre de usuario que utilizará MediaWiki para el acceso a la base de datos. Para este caso, *admin*.
- **DB password:** Contraseña para acceso a la base de datos.
- **DB password confirm:** Se repite la contraseña.
- **Superuser account:** Administrador de la base de datos. Generalmente es el *root*.
- **Superuser password:** Contraseña. En blanco si no se ha establecido.

Luego, se debe pulsar el botón “**Install MediaWiki**” y si toda la información introducida es correcta, la aplicación estará lista para ser utilizada.

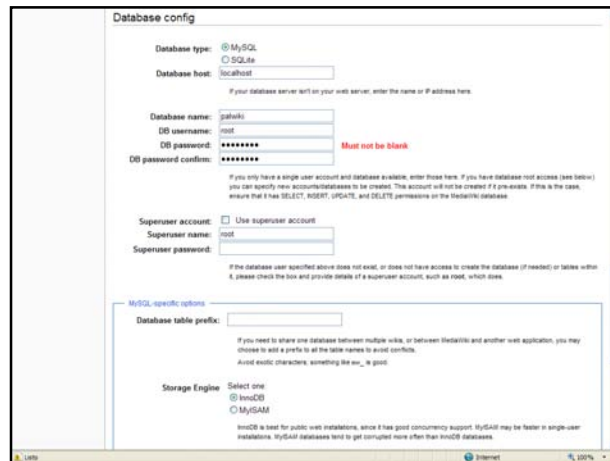
The image shows a web browser window displaying the 'Database config' form for MediaWiki. The form is titled 'Database config' and contains several sections. The first section, 'Database type', has two radio buttons: 'MySQL' (selected) and 'SQLite'. The 'Database host' field is set to 'localhost'. Below this, there is a note: 'If your database server isn't on your web server, enter the name or IP address here.' The next section, 'Database name', has a text input field containing 'pathwiki'. The 'DB username' field is set to 'root', and the 'DB password' field is filled with asterisks. A red error message 'Must not be blank' is visible next to the password field. Below the password field, there is a note: 'If you only have a single user account and database available, enter those here. If you have database root access (see below) you can specify new accounts/databases to be created. This account will not be created if it pre-exists. If this is the case, ensure that it has SELECT, INSERT, UPDATE, and DELETE permissions on the respective database.' The 'Superuser account' section has a checkbox 'Use superuser account' which is unchecked. The 'Superuser name' field is set to 'root', and the 'Superuser password' field is empty. Below this, there is a note: 'If the database user specified above does not exist, or does not have access to create the database (if needed) or tables within it, please check the box and provide details of a superuser account, such as root, which does.' The 'MySQL-specific options' section has a 'Database table prefix' field which is empty. Below this, there is a note: 'If you need to share one database between multiple wikis, or between MediaWiki and another web application, you may choose to add a prefix to all the table names to avoid conflicts. Avoid exotic characters, something like _w_, is good.' The 'Storage Engine' section has a 'Select one' dropdown menu with 'InnoDB' selected. Below this, there is a note: 'InnoDB is best for public web installations, since it has good concurrency support. MyISAM may be faster in single-user installations. MyISAM databases tend to get corrupted more often than InnoDB databases.' The form is displayed in a browser window with a status bar at the bottom showing '100%' zoom and 'Internet' connection.

Figura C-5. Configuración de la base de datos.

Si se presenta algún error al ingresar la información de instalación, aparecerá un error. Una vez que todo está correcto, al pulsar el botón “**Install MediaWiki**” la aplicación comenzará el proceso de creación de la base de datos y las tablas necesarias. En la pantalla se observarán las diferentes tareas que se realizan y al final del proceso se observa el mensaje “**Installation successful!**” que indica que la instalación se ha realizado de forma satisfactoria (ver Figura C-6).

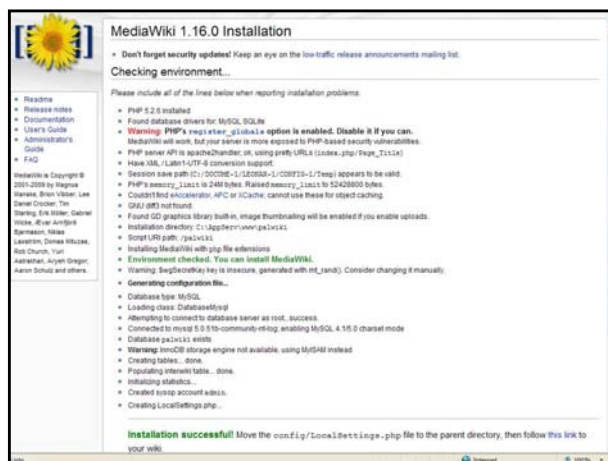


Figura C-6. Instalación exitosa de MediaWiki.

También se indica que se debe mover el fichero **LocalSettings.php** que hay dentro de la carpeta “**Config**” a la carpeta raíz de MediaWiki y seguir el enlace correspondiente. Al hacer clic en el enlace, surgirá la página principal como se observa en la Figura C-7.

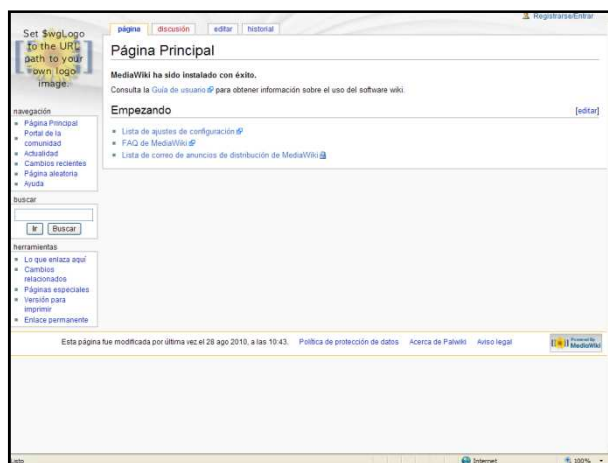


Figura C-7. Página principal de la wiki.

5. CONFIGURACIÓN DE MEDIAWIKI

La configuración por defecto de MediaWiki se establece en el fichero **includes/DefaultSettings.php**. Este fichero no se debe editar ya que si se hace, se pierde la personalización del sitio. En su lugar, se realiza una copia en el fichero **LocalSettings.php** de las líneas que se quieren modificar, y se editan sobre éste último fichero. Como primero se procesa el fichero **DefaultSettings.php** y luego **LocalSettings.php**, cuando dos

parámetros se repiten en ambos, prevalecerán los valores establecidos en LocalSettings.php. Se deben realizar las siguientes configuraciones de parámetros:

5.1 Ingreso de sólo usuarios registrados

Para desactivar la lectura de páginas por parte de usuarios anónimos que ingresan a la wiki se modifica el parámetro **\$wgGroupPermissions** en el fichero **LocalSettings.php**. Así, los usuarios para ingresar al sitio deben tener una cuenta creada.

```
$wgGroupPermissions['*']['read'] = false;
```

5.2 Prevenir la creación de cuentas

Para restringir la creación de cuentas y permitir sólo a los usuarios administradores su creación, en el fichero **LocalSettings.php** se debe agregar el siguiente parámetro:

```
$wgGroupPermissions['*']['createaccount'] = false;
```

5.3 Creación de Cuentas de usuarios

Para crear las cuentas de usuarios se debe ingresar a la siguiente dirección de la wiki:

<http://localhost/palwiki/index.php/Special:Userlogin>

Surgirá la página que se muestra en la Figura C-8 y se selecciona el enlace “**Crear una cuenta**”.

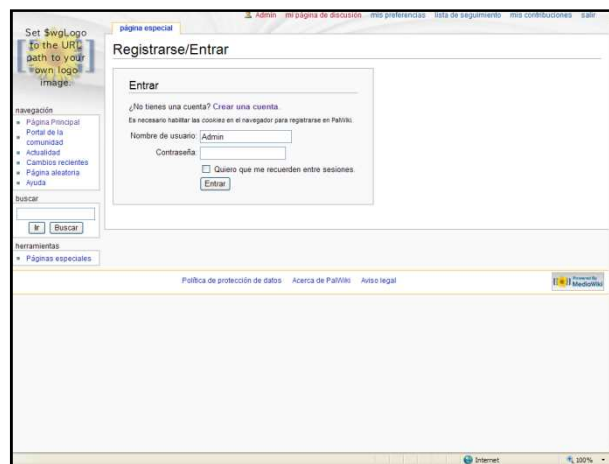


Figura C-8. Ventana UserLogin.

Al seleccionar el enlace se abre un formulario con los datos de la cuenta de usuario a crear (ver Figura C-9):

- Nombre de usuario.
- Contraseña.
- Repetir contraseña.
- Correo electrónico: Opcional, como el servidor de correo no ha sido habilitado no es necesario ingresar este dato.
- Nombre real del usuario.

Figura C-9. Creación de cuenta de usuario.

Se pulsa el botón “**Crear una nueva cuenta**” para finalizar el proceso de creación de una cuenta de usuario. Aparecerá un mensaje confirmando la creación de la cuenta de usuario.

5.4 Permitir la subida de ficheros

En el fichero **LocalSettings.php** se debe modificar un parámetro para permitir la subida de ficheros para que los usuarios puedan subir imágenes, documentos, etc. Para ello, al parámetro **\$wgEnableUploads** se le asigna el valor “**true**”.

```
$wgEnableUploads = true;
```

5.5 Permitir otras extensiones de ficheros

Por defecto, MediaWiki solo permite subir ficheros de tipo png, gif, jpg y jpeg. Para subir otros tipos de archivos como pdfs, zip, etc., en el parámetro **\$wgFileExtensions** de

LocalSettings.php se establecen las extensiones de fichero permitidas. Para el caso de la PAL-Wiki, se habilitan ficheros tipo Word, Excel, Flash, PowerPoint, pdf y zip:

```
$wgFileExtensions = array( 'png', 'gif', 'jpg', 'jpeg', 'avi', 'doc', 'xls', 'swf', 'ppt', 'pdf', 'zip' );
```

5.6 Cambiar logo

El logo que aparece en la esquina superior izquierda de cada página es determinado por el parámetro **\$wgLogo** en el fichero **LocalSettings.php**.

Primero, se debe subir la imagen que será el logo del sitio. Luego, el parámetro **\$wgLogo** se debe añadir en el fichero **LocalSettings.php** con la ruta donde se encuentra localizado el nuevo logo. Por ejemplo, el fichero *logo.jpg* se ha guardado en la carpeta *images/e/e1*. Por lo tanto, en el fichero **LocalSettings.php** se debe agregar la siguiente línea:

```
$wgLogo = 'http://localhost/palwiki/e/e1/logo.jpg';
```

6. CONFIGURACIONES AVANZADAS DE MEDIAWIKI

Las configuraciones avanzadas se refieren a funcionalidades mucho más elaboradas y que generalmente no son muy empleadas en las wikis tradicionales pero deben ser instaladas para el correcto funcionamiento de la PAL-Wiki.

6.1 Semantic MediaWiki

Semantic MediaWiki es una extensión libre de MediaWiki que ayuda a buscar, organizar, poner etiquetas, navegar, evaluar y compartir el contenido del wiki. Los pasos para la instalación de Semantic MediaWiki son los siguientes:

- Descargar Semantic MediaWiki. Descargar el software desde la página web http://semantic-mediawiki.org/wiki/Semantic_MediaWiki
- Copiar los ficheros. Extraer los contenidos del fichero descargado y colocarlos en la carpeta “[**wikipath**]/**extensions**”.
- Activar la extensión. Insertar las siguientes dos líneas en el fichero “**LocalSettings.php**”:

```
include_once("$IP/extensions/SemanticMediaWiki/SemanticMediaWiki.php");
```

```
enableSemantics('example.org');
```

El identificador “*example.org*” debe ser reemplazado por el nombre del servidor web. Es utilizado como un nombre único para identificar los datos exportados de la wiki sobre la Web Semántica, y un nombre de servidor válido funciona para este propósito.

- d) Inicializar la base de datos. Se debe ingresar a la wiki como usuario administrador e ir a la página “**Special:SMWAdmin**” para realizar los pasos finales de la instalación. Dos pasos son necesarios (ver Figura C-10).



Figura C-10. Página de administración de Semantic MediaWiki.

Primero, iniciar la instalación de la base de datos pulsando el botón “**Iniciar o actualizar tablas**”. Se crearán las tablas correspondientes en la base de datos como se observa en la Figura C-11. Al final de este proceso, se genera el siguiente mensaje “**El motor de almacenamiento fue configurado con éxito**” y se debe regresar a la página de “**Special:SMWAdmin**” por medio del enlace presentado.



Figura C-11. Creación de tablas de Semantic MediaWiki.

Luego, activar la actualización automática de datos pulsando el botón “**Comenzar actualización de datos**”. Al finalizar el proceso se genera un mensaje de inicio de actualización (ver Figura C-12) que puede tardar varios minutos.



Figura C-12. Actualización de datos de Semantic MediaWiki.

6.2 SemanticForms

Semantic Forms es una extensión basada en Semantic MediaWiki, que permite a los usuarios crear formularios para añadir y editar páginas que utilizan plantillas para almacenar datos semánticos. Se requiere tener instalada Semantic MediaWiki para la instalación de Semantic Forms. Los pasos para su instalación son los siguientes:

- Descargar Semantic Forms. La extensión se descarga de la página web de Semantic Forms en: http://www.mediawiki.org/wiki/Extension:Semantic_Forms
- Copiar el fichero. El fichero descargado se copia en la carpeta “[**wiki**path]/extensions/”.
- Modificar el fichero **LocalSettings.php**. Añadir la siguiente línea debajo de las correspondientes a la extensión de MediaWiki ('**include_once**' y '**enableSemantics**')

```
include_once("$IP/extensions/SemanticForms/SemanticForms.php");
```

6.3 HeaderTabs

HeaderTabs es una extensión que permite reemplazar los encabezados de primer nivel de las páginas de la wiki por pestañas. Los pasos para su instalación son los siguientes:

- a) Descargar HeaderTabs. La extensión se descarga de la página web de HeaderTabs en:
http://www.mediawiki.org/wiki/Extension:Header_Tabs
- b) Copiar el fichero. El fichero descargado se copia en la carpeta “[wiki-path]/extensions/”.
- c) Modificar el fichero **LocalSettings.php**. Añadir la siguiente línea en el fichero:

```
require_once("$IP/extensions/HeaderTabs/HeaderTabs.php");
```

6.4 Flash

Esta extensión permite la inserción de vídeos dentro de una página de la wiki. Los pasos para su instalación son los siguientes:

- a) Descargar el Fichero Flash.php. La extensión se descarga de la página web de la extensión Flash en: <http://www.mediawiki.org/wiki/Extension:Flash>.
- b) Copiar fichero. El fichero descargado se copia en la carpeta “[wiki-path]/extensions/”.
- c) Modificar el fichero **LocalSettings.php**. Añadir la siguiente línea en el fichero:

```
require_once("$IP/extensions/Flash.php");
```

- d) Si no se ha realizado aún, se debe permitir subir ficheros tipo Flash, agregando la siguiente línea en LocalSettings.php:

```
$wgFileExtensions[] = 'swf';
```

7. IMPORTACIÓN DE LA PAL-WIKI

Para importar los contenidos de la PAL-Wiki se debe modificar la barra de navegación de la wiki, importar un fichero xml con los contenidos de las páginas, subir las imágenes y documentos, y asignar una nueva página principal.

7.1 Personalizar la barra de navegación

Los contenidos de la barra de navegación, los cuales aparecen a la izquierda de cada página son determinados por la página **MediaWiki:Sidebar**. Para modificar sus contenidos se accede por medio de la siguiente dirección:

<http://localhost/palwiki/index.php/MediaWiki:Sidebar>

La página se debe editar y se deben cambiar sus contenidos por los siguientes elementos que contienen la estructura de elementos de proceso de la PAL-Wiki.

- * biblioteca de documentos
- ** Niveles de calidad|Niveles de calidad
- ** <http://selcampus.sel.inf.uc3m.es/SelCampus>
- * visión general
- ** Conceptos|Conceptos
- ** Gobierno de la Organización Software|Gobierno de la Organización Software
- ** Ciclos e Iteraciones|Ciclos e Iteraciones
- ** Modelo de Equipo|Modelo de Equipo
- ** Principios|Principios
- ** Cultura Organizativa|Cultura Organizativa
- * roles
- ** Coordinador del Equipo|Coordinador del Equipo
- ** Ingeniero de Software|Ingeniero de Software
- ** Cliente/Usuario|Cliente/Usuario
- * elementos de trabajo
- ** Defecto|Defecto
- ** Historia de Usuario|Historia de Usuario
- ** Requisito de Calidad de Servicio|Requisito de Calidad de Servicio
- ** Recurso|Recurso
- ** Prueba de Aceptación|Pruebas de Aceptación
- ** Prueba Unitaria|Prueba Unitaria
- ** Clase|Clase
- ** Tarea|Tarea
- ** Elemento de Integración Automática|Elemento de Integración Automática
- * índice de procesos
- ** Procesos|Procesos
- ** Actividades|Actividades
- ** Roles|Roles
- ** Elementos de Trabajo|Elementos de Trabajo
- ** Productos de Trabajo|Productos de trabajo
- ** Instrucciones Técnicas|Instrucciones Técnicas
- * ayuda
- ** Ayuda|Tutorial

La nueva barra de navegación se muestra como aparece en la Figura C-13.

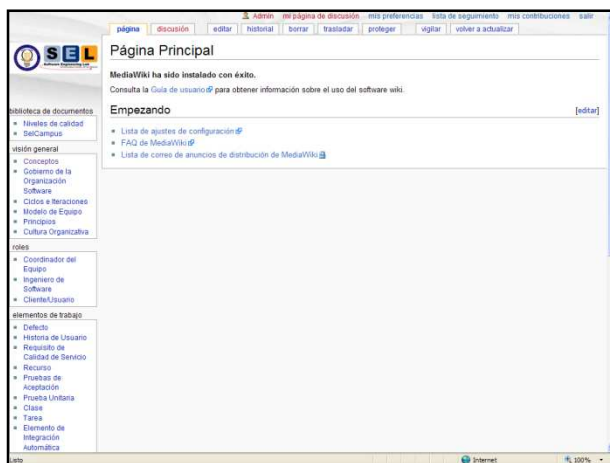


Figura C-13. Barra de navegación de la PAL-Wiki.

7.2 Importación del fichero XML

Los contenidos de la PAL-Wiki se incluyen en el fichero *palwiki.xml* presente en el CD anexo a esta tesis. Para importar sus contenidos se debe ingresar a la wiki como administrador y en la barra de navegación hacer clic en el enlace “**Páginas especiales**” y luego en “**Importar páginas**”. Surgirá una página de importación como se muestra en la Figura C-14. En esta página se selecciona el fichero xml a importar haciendo clic en el botón “**Examinar**”. Luego, se hace clic en el botón “**Subir un archivo**”. El proceso de creación de las páginas a partir del fichero xml puede tardar algunos minutos.



Figura C-14. Importación de páginas de la wiki.

Se genera una página que resume las páginas importadas (Figura C-15), y al finalizar el proceso se genera un mensaje informando que la importación se ha realizado con éxito.



Figura C-15. Resultados de importación de páginas a la wiki.

7.3 Subir los ficheros

Los ficheros de la PAL-Wiki incluidos en la carpeta “**Images**” en el CD anexo a esta tesis doctoral deben ser subidos a la PAL-Wiki.

7.4 Cambiar la página principal

Por último, se debe asignar una de las páginas importadas como la página principal. Por defecto, MediaWiki busca una página con el título de **Página Principal** y la muestra después la instalación. Esta página se puede cambiar alterando los contenidos de la página **MediaWiki:Mainpage** para que señale a un título diferente (ver Figura C-16).

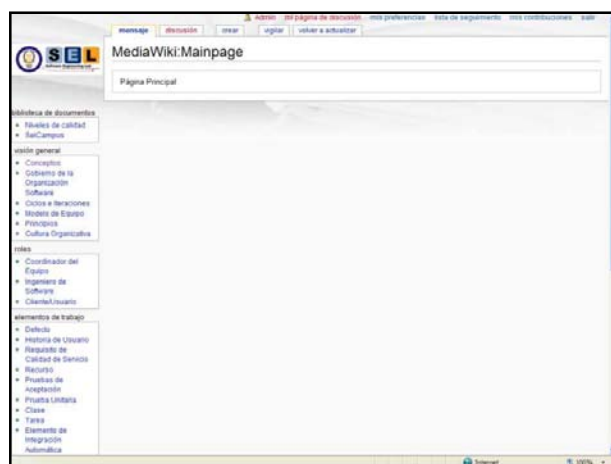


Figura C-16. Cambio de página principal.

Se debe seleccionar la pestaña “**Crear**”, y editar su contenido, cambiándolo a “**Conceptos**” como se observa en la Figura C-17.



Figura C-17. Página principal de la Pal-Wiki.

ANEXO D - MANUAL DE USUARIO DE LA PAL-WIKI

Este manual describe las principales funcionalidades que un usuario puede realizar en la PAL-Wiki.

1. INTERFAZ DE LA PAL-WIKI

Para ingresar como un usuario autorizado a la PAL-Wiki, en la página de acceso (Figura D-1) se deben ingresar los campos “Nombre de usuario” y “Contraseña”. Luego, pulsar el botón “Registrarse/Entrar”.



Figura D-1. Registrarse/Entrar.

Si el usuario es válido, se presentará la página inicial de la PAL-Wiki (Figura D-2), titulada “Conceptos”.



Figura D-2. Página inicial de la PAL-Wiki.

Cualquier página de la PAL-Wiki tiene tres elementos (Figura D-3):

- En la parte lateral izquierda se encuentra **la barra de navegación**, con acceso a las páginas más importantes de la PAL-Wiki.
- En la parte superior de la página se encuentran las **pestañas**, con enlaces útiles para realizar acciones con respecto a la página actual.
- En la esquina superior derecha se encuentran los **enlaces del usuario**, con enlaces a su página de discusión, preferencias, lista de seguimiento, contribuciones, y salir de la PAL-Wiki.



Figura D-3. Registrarse/Entrar.

1.1 La barra de navegación

La barra de navegación tiene las siguientes secciones:

- **Biblioteca de documentos:** Para acceder al repositorio de proyectos.
- **Visión general:** Introducción a la PAL-Wiki acerca del proceso software.
- **Roles:** Lista de roles definidos para realizar los diferentes procesos y actividades.
- **Elementos de trabajo:** Lista de los diferentes ítems de trabajo que se tienen que crear, modificar y gestionar para la gestión del proceso.
- **Índice de procesos:** Con los diferentes elementos que configuran la guía de procesos como procesos, actividades y productos de trabajo.
- **Ayuda:** Tutorial sobre el uso y manejo de la PAL-Wiki.
- **Buscar:** Motor de búsqueda de la PAL-Wiki.
- **Herramientas:** Lista de herramientas como consulta de cambios recientes, subir archivos, páginas especiales, etc.

1.2 Pestañas

En la parte superior, se encuentran las pestañas (Figura D-4):

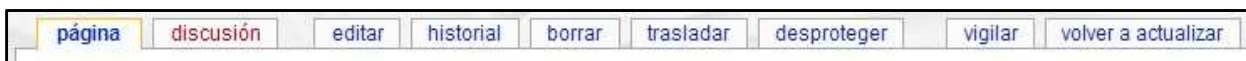


Figura D-4. Pestañas superiores de la PAL-Wiki.

- **Página:** Permite ver el artículo o página de la PAL-Wiki.
- **Discusión:** Permite ir a la página de discusión del artículo.
- **Editar:** Permite modificar la página que actualmente se está viendo.
- **Editar con formulario:** Permite modificar la página utilizando un formulario.
- **Historial:** Permite ver el historial de modificaciones de la página.
- **Trasladar:** Permite cambiar de nombre al artículo, dejando una redirección al nuevo nombre en el antiguo.
- **Proteger o desproteger:** Para prevenir o aceptar la edición por usuarios no autorizados.
- **Vigilar o dejar de vigilar:** Permite añadir o quitar un artículo de la lista de seguimiento.
- **Volver a actualizar:** Actualiza los contenidos de la página.

1.3 Enlaces del usuario

En la parte superior, se encuentran los siguientes enlaces de usuario (Figura D-5):

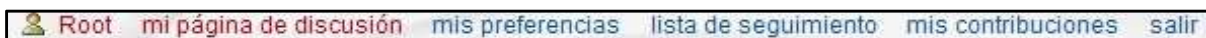


Figura D-5. Enlaces del usuario

- **Nombre de usuario:** Enlaza con la página de usuario.
- **Mi página de discusión:** Página de discusión propia de cada usuario.
- **Preferencias:** Página de configuración de las preferencias de usuario.
- **Lista de seguimiento:** Permite ver las últimas ediciones de los artículos que se han marcado para vigilar.
- **Mis contribuciones:** Muestra una lista de las contribuciones del usuario, ordenadas por fecha.
- **Salir:** Permite salir de la cuenta de usuario.

2. ADQUISICIÓN DEL CONOCIMIENTO

2.1 Creación de formularios

1. Ingresar a la PAL-Wiki como administrador.

2. Para crear un formulario se deben crear primero un conjunto de plantillas que contienen los campos del formulario. Para crear las plantillas se debe acceder a la página **Special:Plantillas** (por ejemplo, <http://localhost/wiki/index.php/Special:Plantillas>) y hacer clic en “**Crear una plantilla**” (Figura D-6).



Figura D-6. Página de plantillas.

3. Ingresar los siguientes datos para la plantilla a crear (Figura D-7):

- **Nombre de la plantilla:** El nombre de la plantilla que contiene varios campos.
- **Categoría:** Categoría a la que pertenece la plantillas. Es opcional.

Para cada campo a definir se deben ingresar los siguientes datos:

- **Nombre del campo:** Nombre del campo de la plantilla, se utiliza para referenciar el valor ingresado por el usuario.
- **Mostrar etiqueta:** La etiqueta que se mostrará al usuario.
- **Propiedad semántica:** Dejar en blanco, ya que la PAL-Wiki no trabaja con propiedades semánticas.

Se pueden agregar más campos a la plantilla, pulsando el botón “**Agregar campo**”. Cuando se termine de definir todos los campos de la plantilla, pulsar el botón “**Grabar la página**”. De esta forma se pueden crear todos los campos de un formulario.

Figura D-7. Crear una plantilla.

4. Ingresar a la página “**Special:Forms**” (Figura D-8) y hacer clic en el enlace de “**Crear un formulario**”.

Figura D-8. Formularios.

5. Ingresar el nombre del formulario a crear y luego agregar las diferentes plantillas que conformarán el formulario (Figura D-9). Las plantillas se deben agregar una por una en el orden en que se vayan seleccionando. Para ello, en la lista desplegable se selecciona una plantilla y se pulsa el botón “**Agregar**”.

Figura D-9. Registrarse/Entrar.

6. Ingresar la etiqueta de la plantilla (Figura D-10). En el ejemplo, la visión general es un campo único, así que no es necesario que tenga una etiqueta que agrupe sus elementos. Por esta razón, la casilla de chequeo “**Permitir múltiples instancias de esta plantilla**” no se selecciona. Para cada campo de la plantilla agregada, se ingresa su etiqueta y tipo de entrada. En el ejemplo, por defecto se muestra el tipo de entrada *text*. La visión general es un campo de texto mucho más amplio y se debe seleccionar *textarea*. Se puede eliminar una plantilla agregada, pulsando el botón “**Sacar plantilla**”. Se pueden agregar más plantilla, antes o después de la actual, por medio del botón “**Agregar**”. Por último, el formulario se crea pulsando el botón “**Grabar la página**”.

Figura D-10. Crear un formulario.

7. El formulario creado se agrega a la lista de formularios de la wiki.

2.2 Definición de roles

1. Ingresar a la PAL-Wiki como administrador.

2. Acceder a la página **Special:Forms** (Figura D-11) que contiene un conjunto de formularios y hacer clic en “**Formulario Rol**”.

Figura D-11. Enlace a formulario de definición de roles.

3. Ingresar el nombre del rol en la caja de texto y pulsar el botón “**Agregar o Editar**” (Figura D-12).

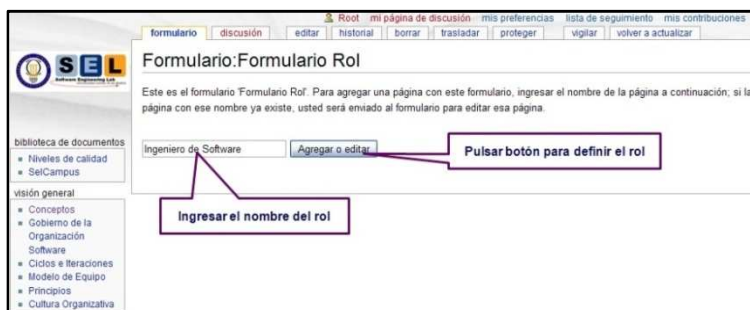


Figura D-12. Creación de la página de rol.

4. Ingresar en los campos correspondientes el nombre y descripción del rol a definir (Figura D-13). Luego, pulsar el botón “**Guardar la página**”.

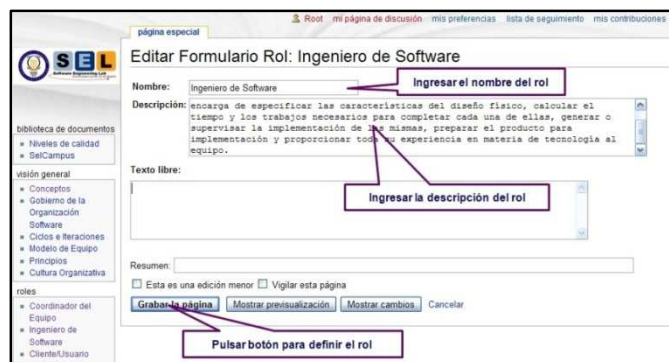


Figura D-13. Formulario de definición de roles.

5. Se genera la página de definición del rol (Figura D-14).



Figura D-14. Rol definido.

2.3 Definición de procesos

1. Ingresar a la PAL-Wiki como administrador.

2. Acceder a la página **Special:Forms** (Figura D-15) que contiene un conjunto de formularios y hacer clic en el enlace “**Formulario Proceso**”.



Figura D-15. Enlace a formulario de procesos.

3. Ingresar el nombre del proceso y pulsar el botón “**Agregar o editar**” (Figura D-16).



Figura D-16. Creación de la página de proceso.

4. Ingresar los campos del formulario de procesos: Visión general, Roles participantes, Criterios de entrada, Cuándo, Actividades, Criterios de salida, Elementos de trabajo, Productos de trabajo, Transparencias, Vídeos, Ejemplos (Figura D-17). Se pueden agregar varios elementos del formulario pulsando el botón “**Agregar otro**”. Se pueden eliminar elementos del formulario pulsando el botón “**Sacar**”. Al finalizar la definición de los campos, pulsar el botón “**Grabar la página**” en la parte final del formulario.



Figura D-17. Formulario de definición de procesos.

5. Se genera la página del proceso definido (Figura D-18).



Figura D-18. Proceso definido.

2.4 Definición de actividades

1. Ingresar a la PAL-Wiki como administrador.

2. Acceder a la página **Special:Forms** (Figura D-19) que contiene un conjunto de formularios y hacer clic en el enlace “**Formulario Actividad**”.



Figura D-19. Enlace a formulario de actividades.

3. Ingresar el nombre de la actividad y pulsar el botón “**Agregar o editar**” (Figura D-20).



Figura D-20. Creación de la página de actividad.

4. Ingresar los campos del formulario de actividades (Figura D-21): Visión general, Roles participantes, Criterios de entrada, Cuándo, Tareas, Criterios de salida, Elementos de trabajo, Productos de trabajo, Transparencias, Vídeos, y Ejemplos. Se pueden agregar varios elementos del formulario pulsando el botón “**Agregar otro**”. Se pueden eliminar elementos del formulario pulsando el botón “**Sacar**”. Al finalizar la definición de los campos, pulsar el botón “**Grabar la página**” en la parte final del formulario.

Figura D-21. Formulario de definición de actividades.

5. Se genera la página de la actividad definida (Figura D-22).

Figura D-22. Registrarse/Entrar.

2.5 Definición de elementos de trabajo

1. Ingresar a la PAL-Wiki como administrador.
2. Acceder a la página **Special:Forms** (Figura D-23) que contiene un conjunto de formularios y hacer clic en el enlace “**Formulario Actividad**”.

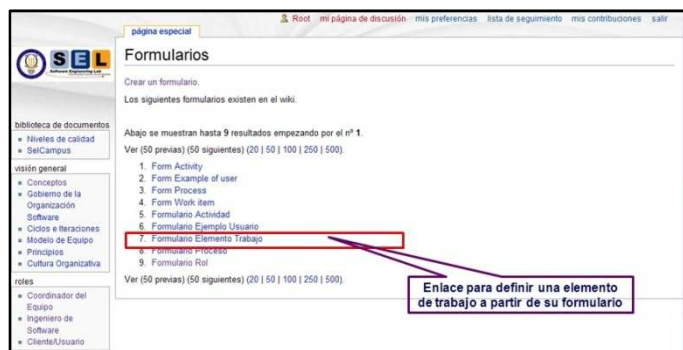


Figura D-23. Enlace a formulario de elemento de trabajo.

3. Ingresar el nombre del elemento de trabajo y pulsar el botón “**Agregar o editar**” (Figura D-24).

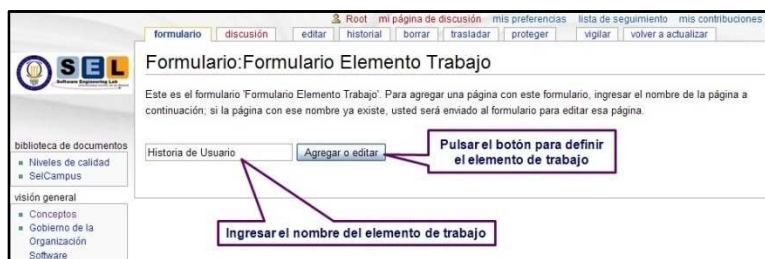


Figura D-24. Creación de página de elemento de trabajo.

4. Ingresar los campos del formulario de actividades (Figura D-25). En la pestaña “**Visión general**” se ingresa el campo Visión general del elemento de trabajo; en la pestaña “**Estados**” se ingresa el fichero que contiene el diagrama de estados del elemento de trabajo e información de cada estado (con su nombre, descripción, y transiciones); y en la pestaña “**Campos**” se ingresa cada campo (con su nombre y descripción). Se pueden agregar varios elementos del formulario pulsando el botón “**Agregar otro**”. Se pueden eliminar elementos del formulario pulsando el botón “**Sacar**”. Al finalizar la definición de los campos, pulsar el botón “**Grabar la página**” en la parte final del formulario.



Figura D-25. Formulario de definición de elementos de trabajo.

5. Se genera la página del elemento de trabajo definido (Figura D-26).

Figura D-26. Elemento de trabajo definido.

3. ORGANIZACIÓN DEL CONOCIMIENTO

3.1 Asignar cada página a categoría

1. Hacer clic en la pestaña “**Editar con formulario**” de una determinada página (Figura D-27).

Figura D-27. Registrarse/Entrar.

2. Agregar la categoría a la que pertenece la página en el área de texto “**Texto libre**” ubicada en la parte inferior del formulario (Figura D-28) con el formato `[[Category:Nombre de la Categoría]]`. Las categorías definidas en la PAL-Wiki son: Procesos, Actividades, Ejemplos de usuarios, Elementos de trabajo, Roles y Visión General. Luego, pulsar el botón “**Grabar la página**”.

Figura D-28. Asignar categoría a una página.

3. En la parte inferior de la página se presenta la categoría asignada (Figura D-29).

Figura D-29. Categoría asignada.

3.2 Asignar procesos y actividades a elementos de trabajo

1. Hacer clic en la pestaña “**Editar con formulario**” de un elemento de trabajo (Figura D-30).

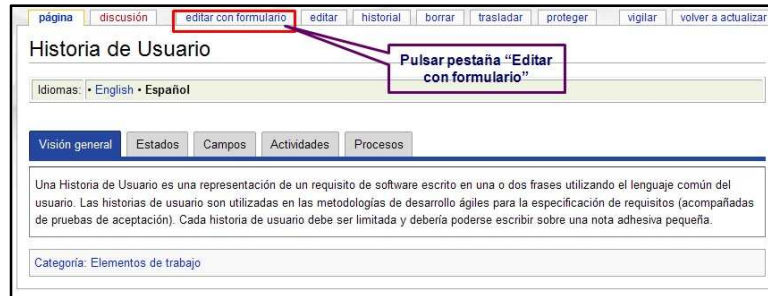


Figura D-30. Enlace para editar elemento de trabajo con formulario.

1. Hacer clic en la pestaña “**Actividades**” (Figura D-31).



Figura D-31. Formulario de elemento de trabajo.

2. Definir las actividades del elemento de trabajo (Figura D-32). En el formulario, ingresar los datos de cada “Estado” junto con su listado asociado de “Actividades”. Luego, pulsar el botón “Grabar la página”.

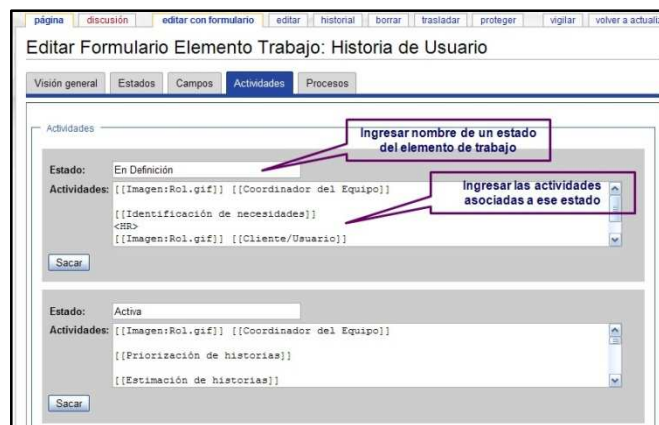


Figura D-32. Asignación de actividades a un elemento de trabajo.

3. Hacer clic en la pestaña “**Procesos**” (Figura D-33).



Figura D-33. Asignación de procesos a un elemento de trabajo.

5. Definir los procesos del elemento de trabajo (Figura D-34). En el formulario, ingresar los datos de cada “**Proceso**” asociado al elemento de trabajo y los “**Roles y Actividades**” relacionados. Luego, pulsar el botón “**Grabar la página**”.



Figura D-34. Asignación de procesos a un elemento de trabajo.

4. DISTRIBUCIÓN DEL CONOCIMIENTO

4.1 Navegación de la PAL-Wiki

1. La navegación de la PAL-Wiki se realiza principalmente a través de la barra de navegación localizada en la parte izquierda (Figura D-35). Los principales elementos son la visión general, los roles, los elementos de trabajo y el índice de procesos.



Figura D-35. Barra de navegación.

2. Hacer clic en el enlace “**Procesos**” ubicado en el “**Índice de procesos**” de la barra de navegación para acceder al conjunto de procesos definidos en la PAL-Wiki (Figura D-36).



Figura D-36. Procesos de la PAL-Wiki.

3. Seleccionar un proceso para consultar sus contenidos y activos asociados. Se genera una página con los contenidos del proceso. Por defecto los contenidos se encuentran en español, pero pueden consultarse en inglés, seleccionando el enlace “**English**” en la parte superior de la página. Los procesos tienen una tabla de **Contenido** para un acceso rápido a dichos contenidos. Después de dicha tabla aparecen los contenidos del proceso (Figura D-37).



Figura D-37. Contenidos de un proceso.

4. Si la página contiene transparencias, se pueden consultar con la barra de control localizada en la parte inferior de cada presentación (Figura D-38).



Figura D-38. Presentación de transparencias.

5. La navegación en los elementos de trabajo se realiza a través de pestañas para que sus contenidos estén mejor organizados debido a que son bastante extensos (Figura D-39).



Figura D-39. Pestañas en páginas de elementos de trabajo.

4.2 Búsqueda de activos en la PAL-Wiki

1. Si los activos no se encuentran navegando en la estructura de la PAL-Wiki, se puede utilizar el motor de búsqueda (Figura D-40) ubicado en la parte inferior de la barra de navegación, con la etiqueta “**buscar**”. Se debe ingresar el texto de búsqueda en el campo de texto y pulsar el botón “**Ir**” para ir a la página con el título exacto buscado o “**Buscar**” para buscar ese texto en las páginas de la PAL-Wiki.

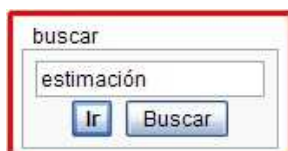


Figura D-40. Motor de búsqueda.

2. El motor de búsqueda genera un conjunto de resultados organizados por coincidencias del texto de búsqueda con títulos de páginas y por texto en los contenidos de las páginas (Figura D-41).



Figura D-41. Resultados de búsqueda.

5. UTILIZACIÓN DEL CONOCIMIENTO

5.1 Instanciar activo de proceso

1. Encontrar el activo para la realización de una tarea o actividad específica. Por ejemplo, durante la gestión de historias de usuario, se requiere una plantilla para priorizar las historias de usuario. La página de la PAL-Wiki que contiene conocimiento sobre la priorización se encuentra por medio de la navegación o búsqueda. Una vez encontrada la página, en este caso *Priorización de Historias*, se hace clic en el enlace que contiene la

plantilla. En el apartado **Productos de trabajo**, se hace clic en el enlace **Lista de Historia de Usuarios** (Figura D-42).



Figura D-42. Consulta de una activo.

2. Pulsar el botón “**Save**” para guardar el fichero que contiene la plantilla seleccionada. Se debe indicar la ruta donde se guardará el fichero (Figura D-43).

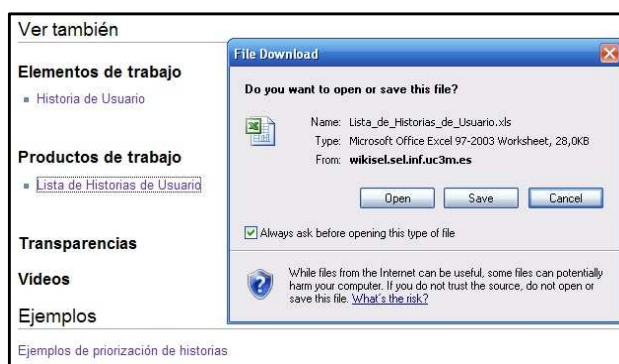


Figura D-43. Descarga de un activo.

3. Abrir el fichero descargado (Figura D-44). En este caso, el fichero es una hoja electrónica con los datos de priorización de historias de usuario.

ID	Descripción	Estado	Prioridad	Tamaño	Esfuerzo
1	Historia de Usuario	0	30	5,761.1,74	
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					
21					
22					
23					
24					
25					
26					
27					
28					
29					
30					

Los posibles estados de las historias son: "Sin comenzar", "En curso", y "Completada".
 El tamaño de las historias está expresado en líneas de código (LOC). El tamaño incluye diseño, codificación, pruebas e integración.
 Las estimaciones del esfuerzo de las historias están expresadas en horas de trabajo.

Figura D-44. Contenidos del activo de proceso.

4. Modificar y guardar la plantilla de acuerdo a las instrucciones dadas en la página de la actividad específica (Figura D-45).

	ID	Iteración	Estado	Prioridad	Tamaño	Esfuerzo
1. Crear juego Sudoku	HU-01		Pendiente	3		
2. Aplicar la técnica Eliminación de candidatos para resolver Sudoku	HU-02		Pendiente	15		
3. Aplicar la técnica Reducción de candidatos ocultos para resolver Sudoku	HU-03		Pendiente	13		
4. Aplicar la técnica Búsqueda de pares/múltiples para resolver Sudoku	HU-04		Pendiente	9		
5. Aplicar la técnica Búsqueda de triplete/múltiples para resolver Sudoku	HU-05		Pendiente	10		
6. Aplicar la técnica Búsqueda de cuartetos/múltiples para resolver Sudoku	HU-06		Pendiente	3		
7. Aplicar la técnica Búsqueda de pares ocultos para resolver Sudoku	HU-07		Pendiente	7		
8. Aplicar la técnica Búsqueda de triplete ocultos para resolver Sudoku	HU-08		Pendiente	8		
9. Aplicar la técnica Búsqueda de pares/múltiples ocultos para resolver Sudoku	HU-09		Pendiente	11		
10. Aplicar la técnica Reducción de cajas/lineas para resolver Sudoku	HU-10		Pendiente	3		
11. Las prioridades de las historias siguen la regla MoSCoW: "Must have", "Should have", "Could have", y "Don't have".						
12.						
13.						
14.						
15.						
16.						
17.						
18.						
19.						
20. La metáfora será que el Sudoku es un gran cajón en cuyo interior hay 81 cajitas (celdas) ordenadas en forma de cuadrícula de 9x9.						
21. Las cajitas se pueden agrupar en conjuntos de 9 elementos, tomando todas las de una misma línea o un grupo de 3x3.						
22. Dentro de cada cajita podrá haber hasta 9 puntos (candidatos), cada uno con un número entre 1 y 9.						
23. Cuando se descarte algún candidato de una de las celdas, se extrae de la cajita la tarjeta correspondiente.						
24.						
25.						
26.						
27.						
28.						

Figura D-45. Instanciación del activo de proceso.

5.2 Publicar instancia de activo de proceso

1. Cambiar el nombre del fichero que tiene la instancia de activo creada.
2. Ingresar a la PAL-Wiki y hacer clic en el enlace “**Subir archivo**” en el apartado de “**Herramientas**” ubicado en la parte inferior de la barra de navegación (Figura D-46).



Figura D-46. Enlace para subir un fichero.

3. En la página “**Subir archivo**”, hacer clic en el botón “**Examinar**” para seleccionar el fichero que contiene el ejemplo de activo a publicar. En la caja de texto “**Nombre del archivo de destino**”, ingresar el nombre como será almacenado en la wiki. Luego, pulsar en el botón “**Subir el archivo**” (Figura D-47).

Figura D-47. Formulario para subir un fichero.

4. Acceder a la página del proceso o actividad de la PAL-Wiki que va a contener el ejemplo a publicar. Para este caso, la página es *Priorización de historias*, y hacer clic en el enlace “*Ejemplo de priorización de historias*” en el apartado “**Ejemplos**” (Figura D-48).



Figura D-48. Enlace a ejemplos de activos.

5. Hacer clic en la pestaña “**Editar con formulario**” para acceder al formulario que permite publicar el ejemplo de activo creado (Figura D-49).



Figura D-49. Enlace para publicar una instancia de activo.

6. Hacer clic en el botón “**Agregar otro**” para publicar el ejemplo de activo de proceso (Figura D-50).

Figura D-50. Formulario de publicación de instancias de activos.

7. Ingresar los datos del formulario de publicación de ejemplos (Figura D-51). En el campo “**Nombre del ejemplo**”, ingresar el nombre del ejemplo y en “**Nombre del fichero**”, ingresar el nombre del fichero que contiene el ejemplo que previamente fue subido a la wiki.

Figura D-51. Datos de la instancia de activo.

5.3 Reuso de activo de proceso

1. Acceder a una página de la PAL-Wiki y hacer clic en el enlace “**Ejemplos**” (Figura D-52).

Figura D-52. Enlace para consultar ejemplos de activos.

2. En el apartado de “**Ejemplos**”, hacer clic en el enlace de ejemplos específicos acerca de ese proceso o actividad (Figura D-53).



Figura D-53. Enlace para ejemplos de activos.

3. Hacer clic en el ejemplo específico a descargar (Figura D-54). El fichero descargado es reutilizado para realizar alguna actividad durante el proceso software.

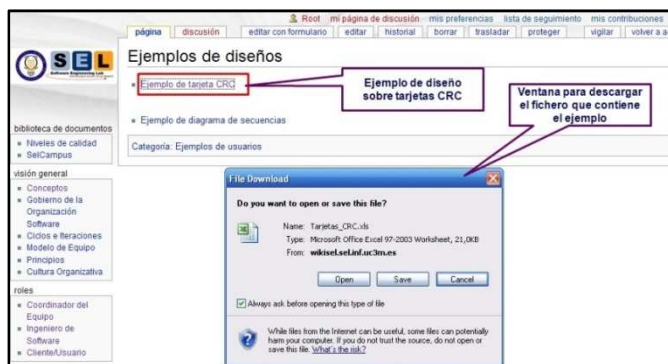


Figura D-54. Descarga de del activo para reuso.

5.4 Consultar ayuda de la PAL-Wiki

1. Hacer clic en el enlace “**Tutorial**” del apartado “**Ayuda**” en la parte inferior de la barra de navegación (Figura D-55).



Figura D-55. Enlace para consultar la ayuda de la wiki.

2. Consultar los diferentes contenidos de la ayuda que están estructurados en las pestañas de la página (Figura D-56). Se puede consultar información sobre la edición de páginas (“**Edición**”), dar formato a las páginas (“**Formato**”), creación de enlaces (“**Enlaces**”), participación en discusiones (“**Discusiones**”) y subida de ejemplos (“**Subir ejemplos**”).

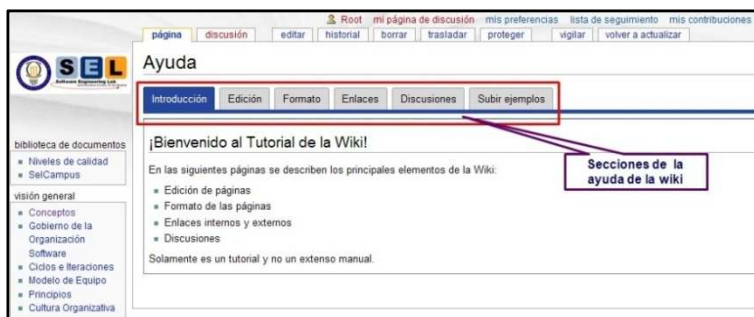


Figura D-56. Página de ayuda de la PAL-Wiki.

6. PRESERVACIÓN DEL CONOCIMIENTO

6.1 Edición de activo de proceso software

1. La edición se puede realizar de dos formas (Figura D-57). Haciendo clic en la pestaña “**Editar con formulario**” en la página de un proceso o actividad de la PAL-Wiki o en la pestaña “**Editar**”. Se recomienda utilizar la primera opción porque el uso de formularios permite una edición más estructurada y sin utilizar el lenguaje de marcación de la wiki.



Figura D-57. Enlaces edición de páginas.

2. Modificar los contenidos de los campos correspondientes del formulario (Figura D-58).

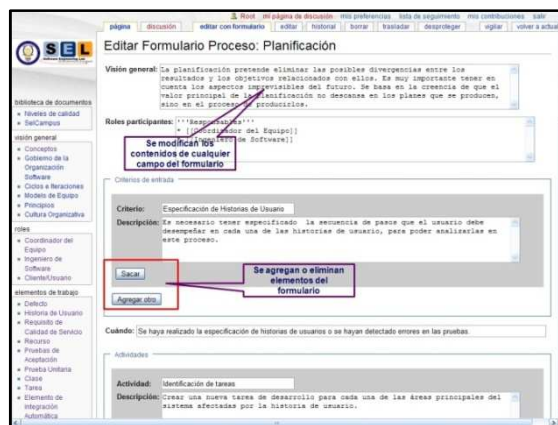


Figura D-58. Formulario de edición.

3. Pulsar el botón “**Grabar la página**” para efectuar los cambios realizados (Figura D-59). Se puede pulsar el botón “**Mostrar pre-visualización**” para observar cómo se vería la página sin haberlo grabado.



Figura D-59. Finalización de la edición.

6.2 Retiro de activo de proceso

1. El retiro de activos puede realizarse de varias formas. Si el activo se refiere a una página específica de la PAL-Wiki, se debe ingresar como administrador y hacer clic en la pestaña “**borrar**” ubicada en la parte superior de la página (Figura D-60).



Figura D-60. Enlace para retiro de activos.

2. Seleccionar de la lista el “**Motivo de borrado**” de la página. Los motivos pueden ser: A petición del autor, Violación del copyright, vandalismo u otra razón. Pulsar el botón “**Borrar esta página**” para confirmar la acción (Figura D-61).



Figura D-61. Página de borrado de página.

3. Se genera una página con la confirmación de la página que ha sido retirada (Figura D-62). Se puede consultar el “**Registro de borrados**”, para consultar las páginas retiradas y restauradas de la PAL-Wiki.



Figura D-62. Confirmación de borrado de página.

4. La otra opción es eliminar un activo que se presenta como un fichero, transparencia o vídeo en la página de un proceso o actividad. Para ello, en la página de dicho proceso o actividad, editar el formulario y eliminar el fichero, haciendo clic en el botón “**Sacar**” del elemento respectivo (Figura D-63).



Figura D-63. Eliminación de activo interno.

6.3 Restaurar activo de proceso

1. Hacer clic en el enlace “**Páginas especiales**” ubicado en la parte inferior de la barra de navegación (Figura D-64).

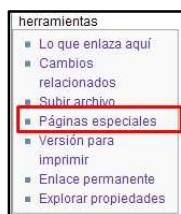


Figura D-64. Enlace a páginas especiales.

2. Hacer clic en el enlace “**Restaurar una página borrada**” ubicado en el apartado “**Herramientas de páginas**” (Figura D-65).



Figura D-65. Enlace para restaurar páginas.

3. Se puede buscar la página borrada de dos formas. Ingresando el texto inicial del título de la página borrada y pulsando luego el botón “**Buscar**”, o haciendo clic en el enlace “**Registro de borrados**” (Figura D-66).

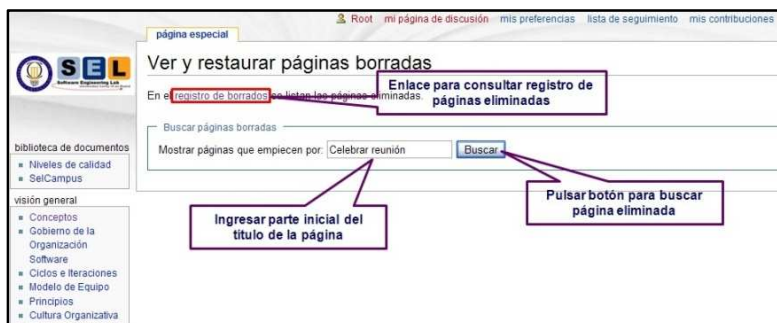


Figura D-66. Página para restaurar una página.

4. En la página “**Ver y restaurar páginas borradas**”, seleccionar la página a restaurar (Figura D-67).

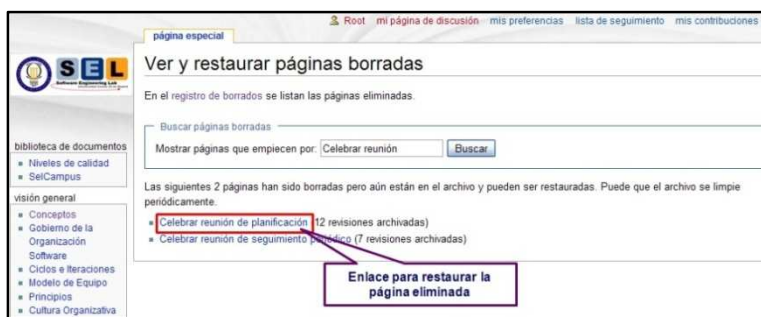


Figura D-67. Selección de página a restaurar.

5. Hacer clic en el enlace “**Ver/restaurar**” de la versión de la página a restaurar para comprobar si es la página que se desea restaurar (Figura D-68). Si esa es la página, seleccionarla por medio de la casilla de chequeo en el apartado “**Historial**”. Luego, pulsar el botón “**Restaurar**”.



Figura D-68. Página de registro de borrados.

6. Se confirma que la página ha sido restaurada (Figura D-69). Hacer clic en el enlace de la página para verificar sus contenidos.



Figura D-69. Confirmación de restauración de una página.

6.4 Discutir sobre activo de proceso

1. Cada página de la wiki tiene asociada una página de discusión. Hacer clic en la pestaña “**Discusión**” ubicada en la parte superior de la página donde se desea iniciar una discusión o participar en una discusión ya creada (Figura D-70).

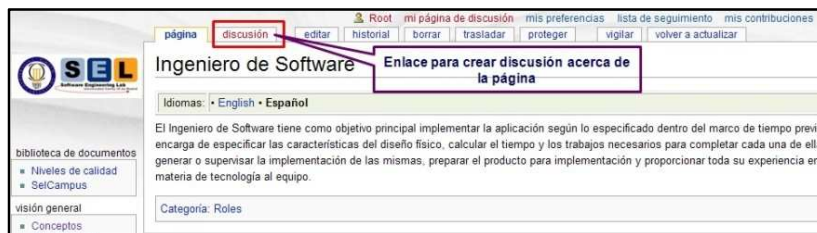


Figura D-70. Creación de una discusión.

2. Escribir los comentarios con respecto a los contenidos de la página (Figura D-71). Existen algunas convenciones para la edición de las páginas de discusión.

- Firmar siempre después de realizar un comentario. Utiliza las cuatro tildes “~~~~” en la sintaxis (o en el botón de la barra de herramientas).
- Comienza una nueva discusión con “==**Encabezado de Nivel 2**==” en la parte baja de la página (o utiliza la pestaña “+”).

- Responder a un comentario con dos puntos (“:”) al inicio de la línea.



Figura D-71. Realización de comentarios en una discusión.

3. La página de discusión es creada y el comentario realizado se muestra junto con el nombre del usuario que realizó el comentario y la fecha y hora de la participación (Figura D-72).



Figura D-72. Página de la discusión creada.

7. GESTIÓN DE USUARIOS

7.1 Crear grupo

Los grupos de usuarios por defecto en la PAL-Wiki son:

- **Bot:** Cuentas con el derecho a realizar ediciones ocultas por medio de scripts automatizados.
- **Administrador:** Todos los privilegios de edición y control de la wiki (bloquear, eliminar, restaurar, proteger paginas, revertir, etc.).
- **Burócrata:** Usuarios con privilegios de cambiar los grupos a los que pertenecen otros usuarios, no tienen permisos administrativos.
- **Root:** Los usuarios de ese grupo pueden bloquear y desbloquear la base de datos.
- **Users:** Los usuarios registrados.

Cuando se crean una cuenta de usuario, esta pertenece al grupo “**Users**”, por defecto. Si se va a crear un nuevo grupo se debe acceder al fichero “**LocalSettings.php**” y agregar líneas como:

```
$wgGroupPermissions['nombre_de_grupo']['nombre_permiso'] = valor;
```

Por ejemplo:

```
$wgGroupPermissions['gestionProcesoOrganizacion']['block'] = true;
```

Se crea el grupo de usuarios “**Gestión del proceso de la organización**” y se le permite bloquear páginas.

7.2 Crear usuarios

1. Ingresar a la PAL-Wiki como administrador.
2. Ingresar a la página “**Special:Userlogin**” (Figura D-73). Hacer clic en el enlace “**Crear una cuenta**”.



Figura D-73. Enlace para crear una cuenta de usuario.

3. Ingresar los datos de la nueva cuenta de usuario (Figura D-74):

- Nombre de la cuenta del usuario del proceso.
- Contraseña de acceso.
- Repetir la contraseña
- Nombre real del usuario.

Figura D-74. Datos para creación de una cuenta de usuario.

4. Se genera un mensaje confirmando la creación de la cuenta de usuario (Figura D-75).

Figura D-75. Confirmación de creación de cuenta.

7.3 Modificar derechos de acceso de los usuarios

1. Hacer clic en el enlace “**Páginas especiales**” ubicado en la parte inferior de la barra de navegación (Figura D-64).

2. Hacer clic en el enlace “**Configuración de permisos de usuario**” en el apartado “**Usuarios y permisos**” en “**Páginas especiales**” (Figura D-76).

Figura D-76. Enlace a configuración de permisos.

3. Escribir el nombre de la cuenta de usuario en la caja de texto, y luego pulsar el botón “**Modificar el grupo de usuarios**” (Figura D-77).

Figura D-77. Ingreso de una nombre de cuenta a configurar.

4. Modificar los permisos del usuario, seleccionando las casillas de chequeo del grupo “**Grupos que puede cambiar**” (Figura D-78). Luego, para confirmar la configuración de permisos, pulsar el botón “**Grupos de usuarios**”. En el apartado “**Cambios en el perfil de usuario**” se registra la fecha y hora de las modificaciones realizadas a la cuenta del usuario y cuál usuario las realizó.

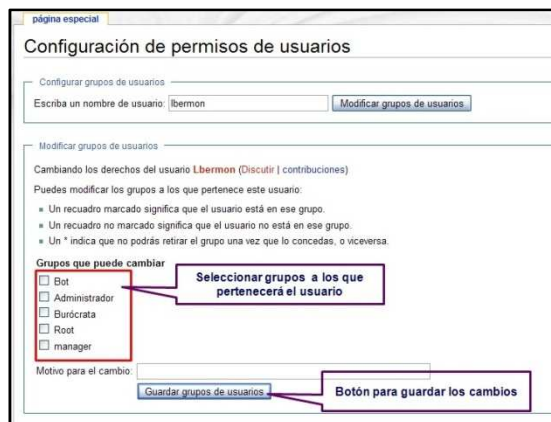


Figura D-78. Página de configuración de permisos de usuarios.

8. GESTIÓN DE CAMBIOS

8.1 Consultar historial de versiones de activos de proceso

1. Hacer clic en la pestaña “**historial**” ubicada en la parte superior de la página para la cual se consultará el historial de versiones (Figura D-79).

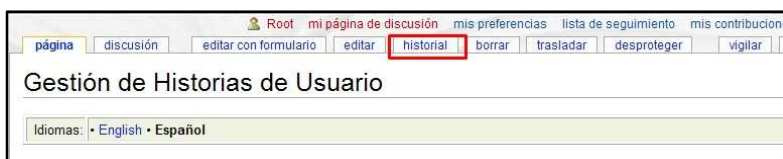


Figura D-79. Enlace para consultar historial de una página.

2. Se genera el historial de versiones de la página (Figura D-80). Si el historial es muy extenso se puede filtrar por fecha (año y mes), ingresando el año en el campo “**Desde el año (y anterior)**” y seleccionando en la lista “**Desde el mes (y anterior)**”. Luego, pulsar el botón “**Mostrar**”.

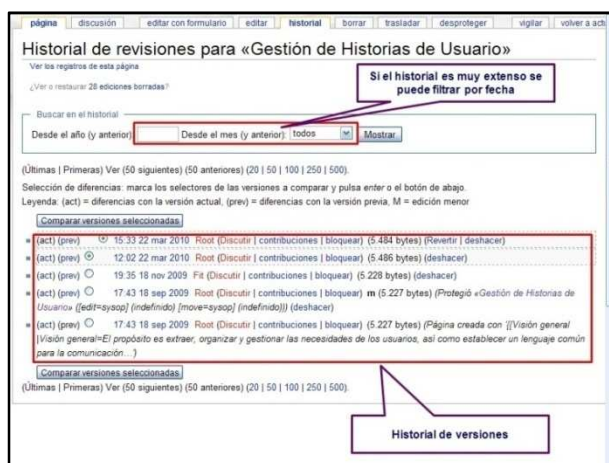


Figura D-80. Historial de versiones.

8.2 Comparar versiones de activos de proceso

1. Seleccionar en el historial de versiones, las casillas de chequeo de las dos versiones que se desean comparar (Figura D-81). Luego, pulsar el botón “**Comparar versiones seleccionadas**”. Al hacer clic sobre “**act**” se pueden ver las diferencias con la versión actual y al hacer clic sobre “**prev**” se pueden ver las diferencias con la versión previa.



Figura D-81. Selección de versiones a comparar.

2. Se genera la comparación entre las dos páginas mostrando en diferentes colores las diferencias (Figura D-82). Además, al final aparecerá la vista de la página como se presenta actualmente.

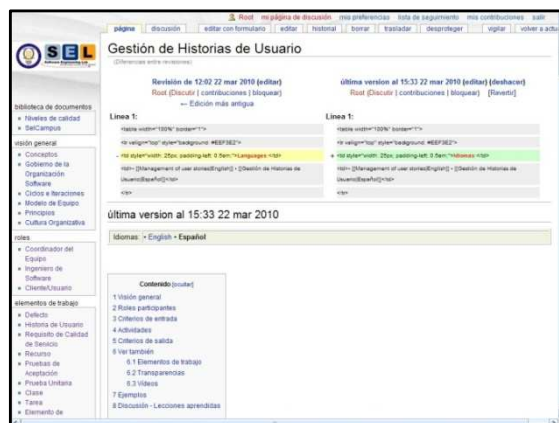


Figura D-82. Reporte de comparación de versiones.

8.3 Moderar contenidos

1. Hacer clic en el enlace “**Páginas especiales**” ubicado en la parte inferior de la barra de navegación (Figura D-64).

2. El apartado “**Cambios recientes y registros**” (Figura D-83) ofrece una colección especial de páginas y herramientas para llevar un seguimiento de lo que ha sucedido en el wiki. Así, un usuario Administrador puede revisarlos y moderar sus contenidos.



Figura D-83. Apartado de cambios recientes y registros.

3. Si se hace clic en el enlace “**Cambios recientes**” se genera una página que muestra todas las ediciones, ficheros subidos, movimiento de páginas, elementos borrados y otras acciones hechas en el wiki (Figura D-84). En el grupo de “**Opciones sobre cambios recientes**” en la parte superior se presenta una colección de enlaces para filtrar la vista: limitar el número de cambios mostrados, el número de días o restringirlo a cierto espacio de nombre. También se pueden ocultar ediciones marcadas como menores.

Una línea en cambios recientes consiste en varios enlaces:

- **diff**: muestra las diferencias con la revisión previa de la página.
- **hist**: enlaza al historial de revisiones de la página,

El enlace con el título completo de la página lleva a la versión más reciente. También, hay señales describiendo el tipo de modificaciones a las páginas:

- **N**: señala una nueva página.
- **m**: una edición menor.
- **b**: una edición hecha por un **bot**.

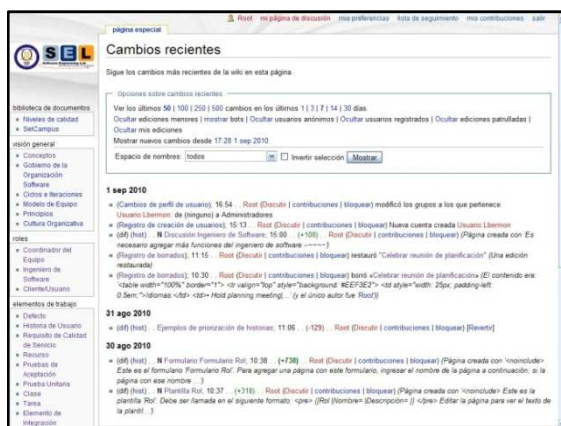


Figura D-84. Cambios recientes.

4. Si se hace clic en el enlace “**Páginas nuevas**” se genera un listado con las páginas nuevas creadas en la wiki (Figura D-85). El listado se puede filtrar por páginas creadas por un usuario específico o por espacio de nombre (para diferenciar diferentes tipos de página: discusiones, plantillas, ficheros, etc.).

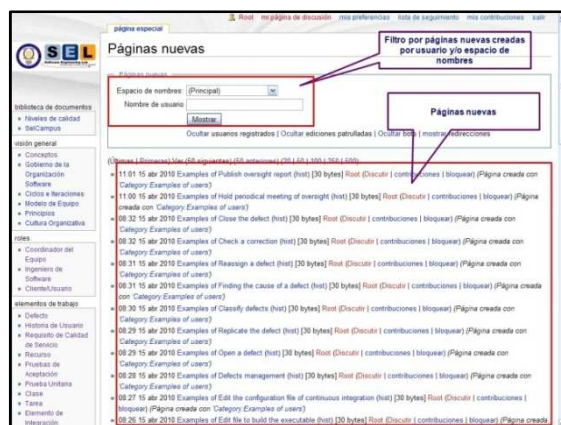


Figura D-85. Páginas nuevas.

5. Si se hace clic en el enlace “**Galería de imágenes nuevas**” se genera una página con un listado de las imágenes y los ficheros recientes que se hayan subido a la PAL-Wiki (Figura D-86).



Figura D-86. Galería de imágenes nuevas.

9. MEDICIÓN DE LA PAL-WIKI

La medición de la PAL-Wiki se realiza por medio de varias formas. La primera está incluida en la instalación; el resto son extensiones que deben ser instaladas separadamente.

- **Estadísticas generales:** Esta funcionalidad está incluida por defecto en la instalación.
- **Usage Statistics:** Permite averiguar cómo el sistema wiki crece con el tiempo proporcionando la capacidad de ver esta información en formato gráfico.
- **Google Analytics:** Para añadir códigos rastreadores a todas las páginas de la wiki y así obtener estadísticas utilizando la herramienta de Google.

9.1. Estadísticas generales

1. Hacer clic en el enlace “**Páginas especiales**” en la parte inferior de la barra de navegación (Figura D-64).
2. Hacer clic en el enlace “**Estadísticas**” en el apartado “**Datos sobre la wiki y herramientas**” en “**Páginas especiales**” (Figura D-87).



Figura D-87. Enlace para consultar estadísticas generales.

3. Se genera la página de estadísticas con información sobre páginas, ediciones, usuarios y visitas (Figura D-88).

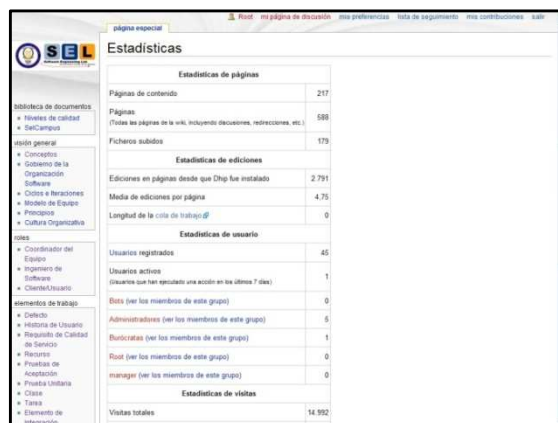


Figura D-88. Estadísticas de la PAL-Wiki.

9.2. Usage Statistics

1. Hacer clic en enlace “**Páginas especiales**” en la parte inferior de la barra de navegación (Figura D-64).
2. Hace clic en el enlace “**Estadísticas de uso**” (Figura D-89).



Figura D-89. Enlace para consultar estadísticas de uso.

3. Seleccionar “**Intervalo**”, “**Tipo de intervalo**”, “**Fecha de inicio**” y “**Fecha de fin**” (Figura D-90). Luego, pulsar el botón “**Generar estadísticas**”.

Figura D-90. Datos para generar estadísticas de uso.

4. Se genera un conjunto de gráficas con datos de uso para el usuario actual y de todos los usuarios sobre las páginas editadas y total de ediciones en el rango de fechas seleccionado (Figura D-91).



Figura D-91. Reporte de estadísticas de uso.

9.3. Google Analytics

1. Ingresar a la página de **Google Analytics** con una cuenta de acceso creada previamente (Figura D-92). Ingresar datos de “**Correo electrónico**” y “**Contraseña**”.



Figura D-92. Página de acceso a Google Analytics.

2. Se abre la página principal de la herramienta donde se puede consultar datos sobre:

- Cómo son los visitantes de la wiki (**Usuarios**).
- Cómo llegan los visitantes a la wiki (**Fuentes de tráfico**).
- Cómo navegan por la wiki (**Contenido**).

La página inicial llamada “**Panel**” ofrece una visión global del sitio y acceso a todas las funcionalidades del sistema (Figura D-93).

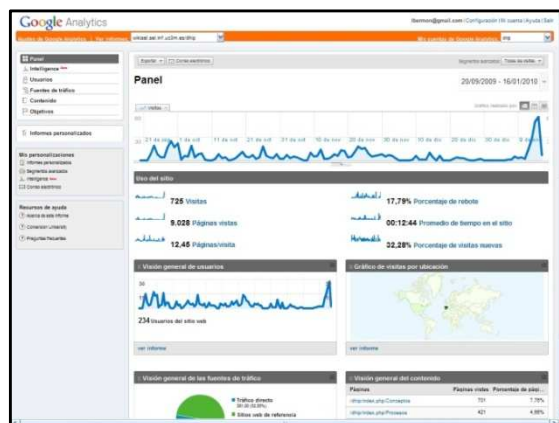


Figura D-93. Panel con resumen de datos de acceso a la wiki.

El panel está formado por un conjunto personalizable de resúmenes de informes. El usuario puede añadir, eliminar y arrastrar los diferentes informes hasta adaptarlo a sus necesidades. Cada informe puede ser examinado en detalle pulsando sobre el enlace “**Ver informe**” (Figura D-94).



Figura D-94. Detalle de un resumen de informe.

Cualquier métrica puede verse con mayor detalle. Por ejemplo, se pueden desglosar las visitas para identificarlas por semana o día (Figura D-95).



Figura D-95. Detalle de una métrica por unidad de tiempo.

La herramienta ofrece un conjunto de indicadores básicos.

- **Visitas:** Número de personas han accedido a la wiki en un periodo de tiempo. Sólo se contabiliza una visita por cada sesión, independientemente del número de páginas a las que haya accedido durante la sesión.
- **Usuario único absoluto:** Número de individuos únicos que visitan la wiki durante un periodo de tiempo.
- **Páginas vistas:** Número total de páginas que han sido visionadas durante un periodo de tiempo.
- **Fuentes de entrada:** Procedencia de las visitas: desde qué páginas se enlaza, con cuáles palabras clave se encontró el sitio, etc.
- **Tiempo en el sitio:** Tiempo medio que dedican los visitantes a navegar por la wiki en cada visita.
- **Porcentaje de abandonos:** Porcentajes de usuarios se ha limitado a visitar una sola página.
- **Nuevas visitas:** Número de usuario que acceden por primera vez a la wiki.