



UNIVERSIDAD
POLITÉCNICA
DE VALENCIA



UNIVERSIDAD POLITÉCNICA DE VALENCIA
ESCUELA TÉCNICA SUPERIOR DE INFORMÁTICA APLICADA

PROYECTO FINAL DE CARRERA

Desarrollo de una aplicación para la gestión de calidad de los procesos en
el entorno JIRA.

Alumno: Francesc Codina Navarro

Tutor: Felix Buendía García

La elaboración de este proyecto final de carrera no habría sido posible sin la ayuda de David Domínguez Tortajada, Eduardo Montón Sánchez y Ricardo Serafín.

Índice de contenido

1. INTRODUCCIÓN.....	5
1.1 DESCRIPCIÓN DEL PROYECTO.....	5
1.2 OBJETIVOS DEL PROYECTO.....	6
1.3 ESTRUCTURA.....	6
2. MARCO TEORICO.....	8
2.1 CMMI.....	8
3. ESPECIFICACIÓN DE REQUISITOS.....	17
3.1 INTRODUCCIÓN.....	17
3.1.1 Propósito.....	17
3.1.2 Ámbito.....	17
3.1.3 Visión global.....	18
3.1.4 Definiciones siglas y abreviaturas.....	18
3.2. DESCRIPCION GENERAL.....	21
3.2.1 Perspectiva del producto.....	21
3.2.2 Funciones del producto.....	22
3.2.3 Características del usuario.....	23
3.2.4 Reestrcciones generales.....	24
3.2.5 Supuestos y dependencias.....	24
3.3 REQUISITOS ESPECIFICOS.....	24
4. ANÁLISIS Y DISEÑO.....	29
4.1 CASOS DE USO.....	29
4.2 DIAGRAMAS DE CLASES.....	31
5. IMPLEMENTACION.....	37
5.1 ENTORNO DE DESARROLLO.....	37
5.3 DIAGRAMAS DE ESTRUCTURAS.....	38
5.2 ESTRUCTURA DEL PLUGIN.....	44
5.4 ARQUITECTURA POR CAPAS.....	48
6. RESULTADOS Y PRUEBAS.....	53
6.1 PRUEBAS.....	53
6.2 RESULTADOS.....	56
7. CONCLUSIONES.....	68
8. REFERENCIAS.....	69
9. ANEXOS.....	70
8.1 INDICADORES.....	70
8.2 CMMI METAS Y PRACTICAS.....	73
8.3 LOS PROCESOS EN LA EMPRESA.....	75
8.4 ACERCA DE JIRA.....	77
8.5 CASOS DE USO.....	85

Índice de ilustraciones

Figura 1: Evolución del modelo CMMI	9
Figura 2: Metas y practicas	16
Figura 3: Casos de uso del jefe de proyecto	29
Figura 4: Casos de uso del administrador de JIRA	30
Figura 5: Casos de uso de los usuarios de JIRA	31
Figura 6: Diagrama de clases de la aplicación JIRA	33
Figura 7: Diagrama de clases de la parte de los informes del plugin	34
Figura 8: Detalle del espacio de trabajo del entorno de desarrollo	37
Figura 9: Diagrama de estructuras de la aplicación JIRA	39
Figura 10: Componentes del sistema	43
Figura 11: Detalle de la estructura del proyecto	45
Figura 12: Detalle de la carpeta donde se encuentran los recursos	45
Figura 13: Estructura del sistema desde la perspectiva de capas	48
Figura 14: Estructura de la tabla maestro de indicadores de la base de datos	50
Figura 15: Estructura de la tabla objetivos de la base de datos	51
Figura 16: Estructura de la tabla medidas de la base de datos	52
Figura 17: Plantilla con datos para el cálculo de los indicadores de avance	53
Figura 18: Plantilla con el cálculo de los indicadores de estado	54
Figura 19: Plantilla con los cálculos de los indicadores de estado	54
Figura 20: Plantilla con las desviaciones y los pesos de las fases	55
Figura 21: Plantilla con el calculo de las velocidades de desarrollo	55
Figura 22: Vista de la pantalla de administración de workflows en JIRA	56
Figura 23: Detalle del workflow de pruebas donde se ven los pasos y las transiciones	57
Figura 24: Pantalla para añadir una condición a la transición	58
Figura 25: Pantalla para añadir un validador a la condición	58
Figura 26: Pantalla de configuración del validador de incidencia enlazada	59
Figura 27: Vista de los validadores de la transición	59
Figura 28: Mensaje de error que se produce cuando no se cumple la condición impuesta por el validador	59
Figura 29: Detalle del Issue del proyecto	59
Figura 30: Detalle de la incidencia del proyecto con la información de los pesos de las fases	60
Figura 31: Informe de avance, estado de las tareas	61
Figura 32: Informe de avance, estado del comienzo de las tareas	61
Figura 33: Informe de avance, Gráfico del informe de avance	62
Figura 34: Informe de avance, estado de finalización de las tareas	62
Figura 35: Informe de avance, plan de medición	63
Figura 36: Detalle de la incidencia del informe de estado	63
Figura 37: Informe de estado, identificación del proyecto	64
Figura 38: Informe de estado, avance del proyecto	64
Figura 39: Informe de estado, estado de desarrollo	65
Figura 40: Informe de estado, Aseguramiento de la calidad del proyecto	66
Figura 41: Informe de estado, Informe de riesgos	66
Figura 42: Informe de estado, plan de seguimiento	67
Figura 43: Plan de medición	67
Figura 44: JIRA de la fundación Apache	77
Figura 45: Incidencia en el JIRA de la compañía Atlassian	80
Figura 46: Detalle de un Bug en el JIRA de la compañía TSB	81
Figura 47: Flujo de trabajo de las incidencias en JIRA	82
Figura 48: Descripción del plugin en la pantalla de administración de plugins	84

1. INTRODUCCIÓN

Esta memoria describe el trabajo realizado durante el proyecto final de carrera.

En este apartado se va a hacer una breve introducción sobre el proyecto, la empresa para la cual se realiza, y de las motivaciones y objetivos que han llevado al desarrollo del mismo.

1.1 DESCRIPCIÓN DEL PROYECTO

El trabajo realizado consiste en la elaboración de una extensión (plugin) a una aplicación de gestión de proyectos. Dicha aplicación se basa en un conocido software denominado JIRA (Atlassian, 2010) que se encarga de gestionar y mantener información relacionada con las tareas involucradas en el desarrollo de productos software. En nuestro caso, la aplicación JIRA se ha utilizado para la gestión de proyectos en la empresa TSB.

TSB Tecnologías para la Salud y el Bienestar, es una empresa dedicada a la implantación y desarrollo de las nuevas tecnologías para el cuidado personalizado de la salud y el bienestar, mejorando la calidad de vida de las personas, y creando nuevas oportunidades de negocio a partir de sus capacidades tecnológicas y de investigación.

TSB fue fundada en enero de 2008 como empresa spin-off del Instituto ITACA de la Universidad Politécnica de Valencia. Partiendo de la experiencia de más de diez años de investigación para el sector socio-sanitario.

La empresa TSB, Tecnologías para la salud y el bienestar desea obtener la certificación del nivel 2 de CMMI, CMMI (Capability Maturity Model Integration) es un modelo para la mejora y evaluación del rendimiento de los procesos de una organización, que fue desarrollado inicialmente para los procesos relativos al desarrollo e implementación de software por la Universidad Carnegie-Mellon para el SEI (Software Engineering Institute).

Para la obtención de la certificación, la empresa TSB ha adquirido la herramienta JIRA. JIRA es un producto software desarrollado por la compañía Atlassian para la gestión de proyectos, seguimiento

de errores e incidencias (bug tracker, issue tracker), gestión de flujos de trabajo (workflows), gestión de metodologías ágiles de desarrollo. JIRA es altamente adaptable y configurable, permitiendo adaptar la mayoría de los aspectos, tipos de incidencias propios, campos, estados, resoluciones y flujos de trabajos.

JIRA esta escrito en java. Incorpora además un sistema de extensiones (plugins) y un interfaz de programación de aplicaciones que permite extender y adaptar JIRA según las necesidades.

La empresa TSB desea poder llevar a cabo la planificación, seguimiento, control y gestión de los proyectos software de manera ágil.

1.2 OBJETIVOS DEL PROYECTO

Los objetivos del proyecto son:

- El desarrollo de extensiones personalizadas sobre la herramienta JIRA.
- Se desarrollaran módulos de flujo de trabajo (Workflow) (validadores y condiciones)
- Se desarrollarán módulos de informes
- Se almacenará la información relativa al control de los proyectos a modo de histórico en una base de datos para su posterior recuperación.

Para la realización del proyecto se hará uso de diversas tecnologías dentro del marco del desarrollo web como Java EE, XML, HTML, CSS, C#, SQL, JavaScript así como el propio API de JIRA y otras tecnologías como PicoContainer, Apache velocity, Lucene, Apache maven, ASP .NET.

1.3 ESTRUCTURA

La estructura de este documento se compone de nueve secciones.

En la segunda sección, marco teórico, se trata de describir la base teórica sobre la que se va a sustentar el proyecto. El proyecto consiste en el desarrollo de una aplicación para la mejora de procesos. En el marco teórico se describe el modelo CMMI para la mejora de procesos.

En la especificación de requisitos se describen las características que debe cumplir la aplicación desarrollada durante el proyecto.

En análisis y diseño se describe el funcionamiento y el contenido de la aplicación. Esta sección contiene una descripción de los casos de uso y los diagramas de clase de la aplicación JIRA y de las extensiones desarrolladas.

En la sección de implementación se describe el proceso de desarrollo del proyecto. Una pequeña descripción de como se desarrollan extensiones para la aplicación JIRA, las tecnologías utilizadas, la estructura de la aplicación y la arquitectura utilizada.

En la sección resultados y pruebas se describe de forma breve el proceso de pruebas, y se muestran los resultados producidos por la aplicación.

En comentarios se exponen las impresiones obtenidas una vez concluido el proyecto sobre la realización del mismo.

En la sección de anexos se exponen documentos de interés para la comprensión del proyecto.

En el anexo de Indicadores se describe la función de los indicadores en los informes.

En el anexo de metas y practicas se expone un ejemplo de metas y practicas genéricas, útil para la comprensión del estándar CMMI.

En el anexo de los procesos en la empresa se definen como están estructurados los procesos en la empresa.

El anexo acerca de JIRA trata de explicar de manera sencilla la funcionalidad de la aplicación JIRA.

En el anexo casos de uso se muestran las tablas con los casos de uso del proyecto de manera mas extendida.

2. MARCO TEORICO

2.1 CMMI

Introducción.

Ahora, más que nunca, las compañías desean entregar mejores productos y servicios en menos tiempo y más baratos. Sin embargo, al mismo tiempo en el entorno de alta tecnología del siglo veintiuno, casi todas las organizaciones se han encontrado construyendo productos y servicios cada vez más complejos.

Hoy en día es raro que las compañías desarrollen por sí mismas todos los componentes que forman parte de un producto o servicio. Frecuentemente, algunos se construyen en la compañía y otros se adquieren, después todos los componentes se integran en el producto o servicio final. Por ello, las organizaciones deben ser capaces de gestionar y controlar este complejo proceso de desarrollo y de mantenimiento.

Los problemas que estas organizaciones encuentran implican soluciones que conciernen a toda la empresa y que requieren una aproximación integrada. La gestión eficaz de los activos de la organización es crítica para el éxito de su actividad.

Los CMM (Modelo de Capacidad y Madurez) se concentran en la mejora de los procesos de una organización. Contienen los elementos esenciales de eficacia de los procesos en una o más disciplinas y describen un camino de mejora evolutivo que permite pasar desde procesos inmaduros ad hoc a procesos disciplinados y maduros de mejor calidad y más eficaces.

En el mercado actual, existen modelos de madurez, estándares, metodologías y guías que pueden ayudar a una organización a mejorar su modo de operar. Sin embargo, la mayoría de las aproximaciones de mejora disponibles se centran en una parte específica de su actividad, concentrándose en mejorar un área de negocio.

El CMMI (Capability Maturity Model Integration) proporciona una oportunidad para evitar o

eliminar estos canales y barreras, apoyándose en los modelos integrados que trascienden disciplinas.

CMMI para el desarrollo

El modelo CMMI para el Desarrollo contempla las buenas prácticas relativas a las actividades de desarrollo y mantenimiento aplicadas a productos y servicios. Trata las prácticas que cubren el ciclo de vida del producto desde la concepción hasta la entrega y el mantenimiento. El énfasis lo pone en el trabajo necesario para construir y mantener el producto completo. Las organizaciones de numerosas industrias, incluyendo la aeroespacial, los bancos, la construcción de ordenadores, el software, la defensa, la fabricación del automóvil y las telecomunicaciones, utilizan el modelo CMMI para el desarrollo.

CMMI para el desarrollo contiene prácticas que cubren la gestión de proyectos, la gestión de procesos, la ingeniería de sistemas, la ingeniería del hardware, la ingeniería de software y otros procesos de soporte utilizados en el desarrollo y el mantenimiento.

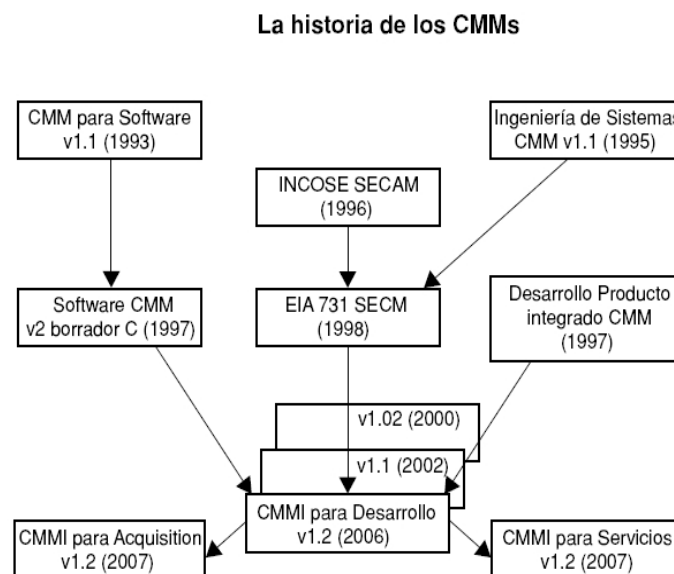


Figura 1: Evolución del modelo CMMI

Historia de los CMMs.

A partir de noviembre de 1986 el SEI (Software Engineering Institute), a requerimiento del Gobierno Federal de los Estados Unidos de América (en particular del Departamento de Defensa), desarrolló una primera definición de un modelo de madurez de procesos en el desarrollo de

software, que se publicó en septiembre de 1987. Este trabajo evolucionó al modelo CMM o SW-CMM (CMM for Software), cuya última versión (v1.1) se publicó en febrero de 1993.

CMMI fue desarrollado inicialmente para los procesos relativos al desarrollo e implementación de software por la Universidad Carnegie-Mellon para el SEI.

El proyecto de integración de CMM ha sido realizado para regular el problema de utilizar múltiples CMM. La misión inicial del equipo del producto CMMI fue combinar tres modelos fuente:

- SW-CMM (Capability Maturity Model for Software), version v2.0 draft C [SEI 1997b]
- SECM (Systems Engineering Capability Model) [EIA 1998]
- IPD-CMM (Integrated Product Development Capability Maturity Model), version v0.98[SEI 1997a]

Estos tres modelos fuente fueron seleccionados debido a su extensa adopción por las comunidades de desarrollo de sistemas y de software y también porque proponen diversos acercamientos a la mejora de procesos en el seno de una organización.

Representación continua y por etapas.

CMMI permite aproximarse a la mejora de procesos y a las evaluaciones usando dos representaciones diferentes: representación continua y por etapas.

La representación continua permite a una organización seleccionar un área de proceso (o un grupo de áreas de proceso) y mejorar los procesos relacionados con ésta. Esta representación utiliza unos niveles de capacidad para caracterizar la mejora concerniente a un área de proceso individual.

La representación por etapas ofrece una manera sistemática y estructurada de aproximarse a la mejora de procesos basada en el modelo etapa a etapa. El logro de cada etapa asegura que se ha establecido una infraestructura de proceso adecuada como fundamento para la etapa siguiente.

La representación por etapas prescribe un orden para implementar las áreas de proceso según unos niveles de madurez, que determinan el camino seguido por una organización para pasar del nivel inicial al nivel “en optimización”. Alcanzar cada nivel de madurez asegura que se ha establecido un fundamento adecuado para el siguiente nivel de madurez, lo que permite una mejora incremental y duradera.

En resumen CMMI soporta dos caminos de mejora. Un camino permite a las organizaciones mejorar de forma incremental los procesos que corresponden a un área o áreas de proceso individuales seleccionadas por la organización. El otro camino permite a las organizaciones mejorar un conjunto de procesos relacionados, tratando de forma incremental conjuntos sucesivos de áreas de proceso. Para la representación continua, se utiliza el término “nivel de capacidad”. Para la representación por etapas, se utiliza el término “nivel de madurez”.

Independientemente de qué representación se seleccione, el concepto de niveles es el mismo. Los niveles caracterizan a la mejora desde un estado mal definido hasta un estado que utiliza información cuantitativa para determinar y gestionar las mejoras que se necesitan para satisfacer los objetivos de negocio de una organización.

Para alcanzar un nivel particular, una organización debe satisfacer todas las metas apropiadas del área o conjunto de áreas de proceso que son objeto de la mejora, independientemente de si es un nivel de capacidad o de madurez. Ambas representaciones también proporcionan caminos para implementar la mejora de procesos que permiten lograr los objetivos de negocio. Ambas representaciones proporcionan el mismo contenido esencial y utilizan los mismos componentes del modelo.

Niveles de capacidad

Los niveles de capacidad, que pertenecen a la representación continua, se aplican al logro de mejora de procesos de una organización en áreas de proceso individuales. Estos niveles son un medio para mejorar de forma incremental los procesos que corresponden a un área de proceso dada.

Existen seis niveles de capacidad, numerados de 0 a 5:

0. Incompleto.
1. Realizado.
2. Gestionado.
3. Definido.
4. Gestionado cuantitativamente.
5. En optimización.

Niveles de madurez

Para dar soporte a aquellos que utilizan la representación por etapas, todos los modelos CMMI reflejan niveles de madurez en su diseño. El nivel de madurez de una organización proporciona un camino para predecir el rendimiento en una disciplina dada o en un conjunto de disciplinas.

Existen cinco niveles de madurez, denominados por los números 1 a 5:

1. Inicial.
2. Gestionado.
3. Definido.
4. Gestionado cuantitativamente.
5. En optimización.

Nivel de madurez 1: Inicial

En el nivel de madurez 1, los procesos son generalmente al vuelo y caóticos. La organización generalmente no proporciona un entorno estable para dar soporte a los procesos.

Nivel de madurez 2: Gestionado

En el nivel de madurez 2, los proyectos de la organización han asegurado que los procesos se planifican y realizan de acuerdo a políticas, los proyectos emplean personal cualificado que dispone de recursos adecuados para producir resultados controlados, involucran a las partes interesadas, se monitorizan, controlan y revisan, se evalúan en cuanto a su adherencia a las descripciones del proceso. Los proyectos se realizan y gestionan de acuerdo a sus planes documentados.

En el nivel de madurez 2, el estado de los productos de trabajo y la entrega de los servicios son visibles a la dirección en puntos definidos (en los hitos principales y al finalizar las tareas principales). Se establecen compromisos entre las partes interesadas y se revisan, según sea necesario. Los productos de trabajo se controlan de forma apropiada. Los productos de trabajo y servicios satisfacen sus descripciones de proceso especificadas, estándares y procedimientos.

Nivel de madurez 3: Definido

En el nivel de madurez 3, los procesos son bien caracterizados y comprendidos, y se describen en estándares, procedimientos, herramientas y métodos. El conjunto de procesos estándar de la organización, que es la base del nivel de madurez 3, se establece y mejora a lo largo del tiempo. Estos procesos estándar se usan para establecer la consistencia en toda la organización. Los proyectos establecen sus procesos definidos adaptando el conjunto de procesos estándar de la organización de acuerdo a las guías de adaptación.

Nivel de madurez 4: Gestionado cuantitativamente

En el nivel de madurez 4, la organización y los proyectos establecen objetivos cuantitativos en cuanto al rendimiento de calidad y del proceso, y los utilizan como criterios en la gestión de los procesos. Los objetivos cuantitativos se basan en las necesidades del cliente, usuarios finales y organización. El rendimiento de los procesos se comprende en términos estadísticos y se gestiona durante la vida de los mismos.

Nivel de madurez 5: En optimización

El nivel de madurez 5 se centra en mejorar continuamente el rendimiento de los procesos mediante mejoras incrementales. Se establecen los objetivos cuantitativos de mejora de procesos para una organización, se revisan continuamente para reflejar el cambio de los objetivos de negocio, y se utilizan como criterios para gestionar la mejora de los procesos. Se miden y evalúan los efectos de las mejoras de los procesos desplegadas frente a los objetivos cuantitativos de mejora de los procesos. Tanto los procesos definidos como el conjunto de procesos estándar de la organización son objeto de las actividades de mejora cuantitativa.

Áreas de proceso

Un área de proceso es un grupo de prácticas relacionadas que son ejecutadas de forma conjunta.

En CMMI las áreas de proceso se organizan en cuatro categorías: Gestión de procesos, Gestión de

proyectos, Ingeniería y Soporte.

La representación por etapas proporciona un camino predeterminado de mejora, desde el nivel de madurez 1 hasta el nivel de madurez 5. Las áreas de proceso se agrupan por nivel de madurez, indicando qué áreas de proceso implementar para lograr cada nivel de madurez. Por ejemplo, en el nivel de madurez 2 existe un conjunto de áreas de proceso que una organización debería usar. Una vez que se ha logrado el nivel de madurez 2 de esta forma, la organización enfoca sus esfuerzos en las áreas de proceso del nivel de madurez 3, y así sucesivamente.

Área de proceso	Categoría	Nivel de madurez
Análisis de Causas y Resolución - CAR	Soporte	5
Gestión de configuración - CM	Soporte	2
Análisis de decisiones y resolución - DAR	Soporte	3
Gestión integrada de proyecto + IPPD - IPM + IPPD	Gestión de proyectos	3
Medición y análisis - M	Soporte	2
Innovación y despliegue en la organización - OID	Gestión de procesos	5
Definición de procesos de la organización + IPPD - OPD + IPPD	Gestión de procesos	3
Enfoque en procesos de la organización - OPF	Gestión de procesos	3
Rendimiento de procesos de la organización - OPP	Gestión de procesos	4
Formación organizativa - OT	Gestión de procesos	3
Integración de producto - PI	Ingeniería	3
Monitorización y control de proyecto - PMC	Gestión de proyectos	2
Planificación de proyecto - PP	Gestión de proyectos	2
Aseguramiento de la calidad de proceso y de producto - PPQA	Soporte	2
Gestión cuantitativa de proyecto - QPM	Gestión de proyectos	4
Desarrollo de requerimientos - RD	Ingeniería	3
Gestión de requerimientos - RD	Ingeniería	2
Gestión de riesgos - RSKM	Gestión de proyectos	3
Gestión de acuerdos con proveedores - SAM	Gestión de proyectos	2
Solución técnica - TS	Ingeniería	3
Validación - VAL	Ingeniería	3
Verificación - VER	Ingeniería	3

Metas y practicas.

CMMI propone una serie de metas y practicas las cuales pueden mejorar los procesos de la organización.

Una meta específica describe las características únicas que deben estar presentes para satisfacer el área de proceso.

Las metas genéricas se aplican al conjunto de áreas de proceso. Una meta genérica describe las características que deben estar presentes para institucionalizar los procesos que implementan un área de proceso.

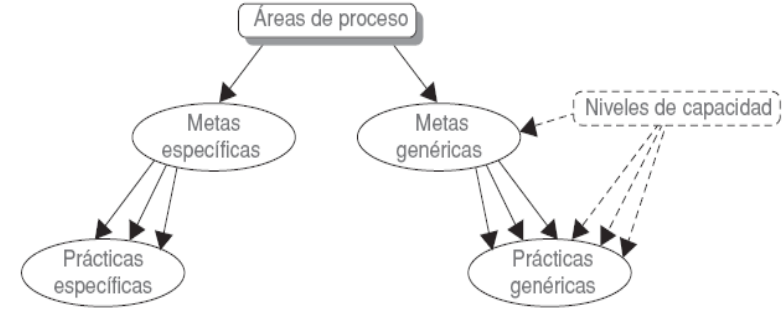
Tanto las metas específicas y genéricas son un componente requerido del modelo y se utilizan en las evaluaciones para determinar si se satisface un área de proceso. Los componentes requeridos describen lo que una organización debe realizar para satisfacer un área de proceso. Este logro se debe implementar de forma visible en los procesos de una organización.

Una práctica específica es la descripción de una actividad que se considera importante para alcanzar la meta específica asociada. Las prácticas específicas describen las actividades que se espera que produzcan la consecución de las metas específicas de un área de proceso.

Las prácticas genéricas se aplican al conjunto de áreas de proceso. Una práctica genérica es la descripción de una actividad que se considera importante para el logro de la meta genérica asociada.

Tanto las practicas especificadas y genéricas son componentes esperados del modelo. Los componentes esperados describen lo que una organización puede implementar para lograr un componente requerido. Los componentes esperados guían a los que implementan mejoras o realizan evaluaciones.

Representación continua



Representación por etapas

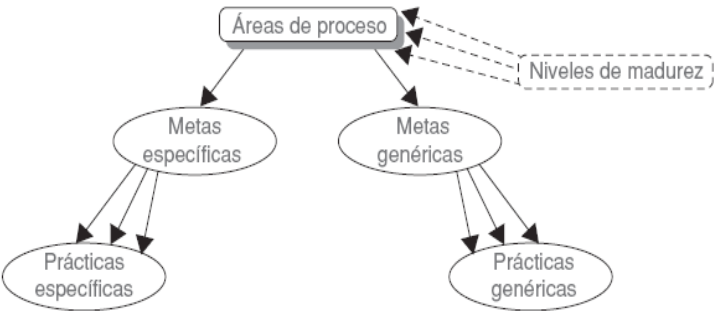


Figura 2: Metas y practicas

3. ESPECIFICACIÓN DE REQUISITOS

3.1 INTRODUCCIÓN

3.1.1 Propósito

El propósito de la especificación de requisitos expuesta a continuación es mostrar al usuario final cual va a ser la funcionalidad del proyecto.

Es útil para comprender lo que el cliente quiere, analizar sus necesidades, negociar una solución razonable y que no quede ambigua.

A través de esta especificación se determina lo que el sistema va a hacer y las restricciones que va a tener.

Además, reducirá el esfuerzo en el desarrollo, servirá como base para la estimación de costes y planificación a la hora de desarrollar la aplicación y será un punto de referencia para procesos de verificación y validación. En esta ERS se realizará una descripción general del proyecto, así como una especificación de los requisitos que debe cumplir.

3.1.2 Ámbito

El proyecto consiste en el desarrollo de una aplicación de uso interno para la empresa TSB.

TSB Tecnologías para la Salud y el Bienestar, es una empresa dedicada a la implantación y desarrollo de las nuevas tecnologías para el cuidado personalizado de la salud y el bienestar.

TSB fue fundada en enero de 2008 como empresa spin-off del Instituto ITACA de la Universidad Politécnica de Valencia. Partiendo de la experiencia de más de diez años de investigación para el sector socio-sanitario.

La empresa TSB, Tecnologías para la salud y el bienestar desea obtener la certificación del nivel 2 de CMMI.

CMMI (Capability Maturity Model Integration) es un modelo para la mejora y evaluación del rendimiento de los procesos de una organización.

La aplicación es una extensión de la herramienta JIRA utilizada por los trabajadores de la empresa para controlar los procesos relativos a los proyectos software.

JIRA: Es un producto software desarrollado por la compañía Atlassian con capacidad para la gestión de proyectos, seguimiento de errores e incidencias (bug tracker, Issue tracker), gestión de flujos de trabajo (workflows), gestión de metodologías ágiles de desarrollo.

3.1.3 Visión global

A partir de este momento, la especificación de requisitos se centrará en describir la funcionalidad del servicio, características del usuario, restricciones generales y requisitos detectados.

3.1.4 Definiciones siglas y abreviaturas

Acción correctiva: Acción tomada para eliminar la causa de una no conformidad detectada u otra situación no deseable NC ISO 9000: 2005 (3.6.2) .

La **auditoría** informática: Es el proceso de recoger, agrupar y evaluar evidencias para determinar si un sistema de información salvaguarda el activo empresarial, mantiene la integridad de los datos, lleva a cabo eficazmente los fines de la organización, utiliza eficientemente los recursos, y cumple con las leyes y regulaciones establecidas.

También permiten detectar de forma sistemática el uso de los recursos y los flujos de información dentro de una organización y determinar qué información es crítica para el cumplimiento de su misión y objetivos, identificando necesidades, duplicidades, costes, valor y barreras, que obstaculizan flujos de información eficientes.

CMMI (Capability Maturity Model Integration) es un modelo para la mejora y evaluación del rendimiento de los procesos de una organización.

La **gestión de proyectos**: Es la aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades de un proyecto para satisfacer los requisitos del proyecto.

JIRA: Es un producto software desarrollado por la compañía Atlassian con capacidad para la gestión de proyectos, seguimiento de errores e incidencias (bug tracker, Issue tracker), gestión de flujos de trabajo (workflows), gestión de metodologías ágiles de desarrollo.

La **matriz de la asignación de responsabilidades (RACI** por sus siglas en inglés): Se utiliza generalmente para relacionar actividades con recursos (individuos o equipos de trabajo). De esta manera se logra asegurar que cada uno de los componentes del alcance esté asignado a un individuo o a un equipo.

No conformidad: De acuerdo a la definición en la norma NC ISO 9000: 2005 (3.1.2), una no conformidad es el incumplimiento de un requisito.

Proceso: es un conjunto de actividades o eventos (coordinados u organizados) que se realizan o suceden (alternativa o simultáneamente) con un fin determinado.

Un **proyecto**: Es una planificación que consiste en un conjunto de actividades que se encuentran interrelacionadas y coordinadas, la razón de un proyecto es alcanzar objetivos específicos dentro de los límites que imponen un presupuesto y un lapso de tiempo previamente definidos.

Riesgo: La incertidumbre o probabilidad de que ocurra o se realice una eventualidad, la cual puede estar prevista, en este sentido podemos decir que el riesgo es la contingencia de un daño.

En función de lo anterior podemos afirmar que los riesgos informáticos se refieren a la incertidumbre existente por la posible realización de un suceso relacionado con la amenaza de daño respecto a los bienes o servicios informáticos.

Una **estructura de descomposición del trabajo** también conocido por su nombre en inglés **Work Breakdown Structure** o **WBS**, es una estructura exhaustiva, jerárquica y descendente formada por los entregables a realizar en un proyecto. La EDT es una herramienta muy común y crítica en la gestión de proyectos.

Una EDT es una presentación simple y organizada del trabajo requerido para completar el proyecto. Existen muchas maneras de organizar la presentación de este trabajo. Por ejemplo, se puede organizar de acuerdo a los Grupos de Proceso del ciclo de vida del proyecto o de las fases (Inicio, Planificación, Ejecución, Control y Cierre), mostrando cada fase como un elemento del nivel más alto. Otra forma de organizarla es teniendo en cuenta las responsabilidades funcionales. Algo importante de recordar es que la EDT documenta el alcance del proyecto, no su plan de ejecución.

Un **servicio web** (en inglés, *Web service*) es un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos.

La razón por la que los servicios Web son muy prácticos es que pueden aportar gran independencia entre la aplicación que usa el servicio Web y el propio servicio. De esta forma, los cambios a lo largo del tiempo en uno no deben afectar al otro. Esta flexibilidad será cada vez más importante, dado que la tendencia a construir grandes aplicaciones a partir de componentes distribuidos más pequeños es cada día más utilizada.

Flujo de trabajo (workflow): Es una secuencia de pasos conectados, Es una representación de una secuencia de operaciones, cómo se estructuran las tareas, cómo se realizan, cuál es su orden correlativo, cómo se sincronizan, cómo fluye la información que soporta las tareas y cómo se le hace seguimiento al cumplimiento de las tareas.

El flujo de trabajo puede considerarse como una vista del trabajo real bajo un cierto aspecto, sirviendo así como una representación virtual de trabajo real.

3.2. DESCRIPCION GENERAL

3.2.1 Perspectiva del producto

Mediante la aplicación JIRA los jefes de proyecto establecen:

- El calendario del proyecto.
- El calendario de las diferentes fases en que se compone el proyecto.
- La estimación de la duración de las tareas.
- La asignación de las diferentes tareas y casos de uso, a los desarrolladores.
- Control de los plazos del proyecto.
- Control de los costos del proyecto.
- Control y gestión de las auditorias realizadas sobre el proyecto.
- Identificación y gestión de los riesgos surgidos durante el proyecto.
- Identificación y control de las no conformidades que puedan surgir durante el proyecto.
- Gestión de las acciones correctivas sobre los riesgos y no conformidades encontrados.
- Gestión de la asignación de responsabilidades a terceras partes.

La aplicación se encargará de generar informes donde se muestra información relativa a los procesos anteriores, además guardará la información recogida en una base de datos para su posterior recuperación.

Los administradores son los encargados de adaptar la herramienta JIRA a las necesidades de la empresa mediante:

- Gestión de los proyectos, usuarios, roles, etc.
- Control de las políticas de seguridad de la herramienta.
- Creación de diferentes tipos de incidencias, estados, resoluciones, campos, etc.
- Creación de workflows personalizados para las diferentes incidencias.

La aplicación contendrá módulos con funciones de workflow con objetivo de ayudar a los administradores de JIRA en la labor de personalización de la herramienta.

3.2.2 Funciones del producto

Las funciones que llevará a cabo la aplicación son las siguientes:

Funciones de flujo de trabajo ("workflow"):

- Condición de archivo adjunto.

Crear una condición del flujo de trabajo que permita una transición de una incidencia solo si esta contiene al menos un archivo adjunto.

Asignar la condición a un flujo de trabajo.

Editar la condición perteneciente a un flujo de trabajo.

- Condición de incidencia enlazada.

Crear una condición del flujo de trabajo que verifique que la incidencia a realizar la transición contiene cierto número de enlaces a otras incidencias de un tipo dado.

Asignar la condición a un flujo de trabajo.

Editar la condición perteneciente a un flujo de trabajo.

- Validador de incidencia enlazada.

Actúa de manera similar al módulo de condición pero esta vez es un validador. Consiste en crear un validador de flujo de trabajo que verifique que la incidencia a realizar la transición contiene cierto número de enlaces a otras incidencias de un tipo dado.

Asignar la condición a un flujo de trabajo.

Editar la condición perteneciente a un flujo de trabajo.

- Validador de campo especificado.

Crear un validador que verifique que al menos un cierto campo de una determinada lista de campos de la incidencia a realizar la transición contiene algún valor.

Asignar la condición a un flujo de trabajo.

Editar la condición perteneciente a un flujo de trabajo.

Informes:

- Informe de estado.

Generar el informe de estado.

Generar el informe de estado y evaluar los indicadores.

Generar el informe de estado y almacenar los indicadores.

- Informe de avance

Generar el informe de avance.

Generar el informe de avance y evaluar los indicadores.

Generar el informe de avance y almacenar los indicadores.

3.2.3 Características del usuario

Los jefes de proyecto son los encargados de gestionar el proyecto al cual han sido asignados, son los encargados de generar los informes de avance y estado del proyecto.

Los administradores se encargan de la creación de los flujos de trabajo, para ello pueden contar con la ayuda de los modulos desarrollados.

3.2.4 Reestrcciones generales

Para la realización de los informes se tomarán como referencia una serie de plantillas y esquemas proporcionados por la empresa.

3.2.5 Supuestos y dependencias

La version de JIRA utilizada es la v3.13.

El web service funcionará sobre un servidor web IIS.

La base de datos utilizada será SQL Server 2005.

Las paginas web mostradas se deberán poder visualizar correctamente en los navegadores web firefox y safari.

3.3 REQUISITOS ESPECIFICOS.

Condición de archivo adjunto.

Crear una condición del flujo de trabajo que permita una transición de una incidencia solo si esta contiene al menos un archivo adjunto.

Condicion de incidencia enlazada.

Crear una condición del flujo de trabajo que verifique que la incidencia a realizar la transición contiene cierto número de enlaces a otras incidencias de un tipo dado.

Entradas:

- El número de enlaces.
- El tipo de enlace.
- El tipo de condición.

Salidas:

Se produce una salida para visualización y edición que muestra los datos de configuración.

- El número de enlaces.
- El tipo de enlace
- El tipo de condició

Validador de incidencia enlazada.

Actúa de manera similar al módulo de condición pero esta vez es un validador. Consiste en crear un validador de flujo de trabajo que verifique que la incidencia a realizar la transición contiene cierto número de enlaces a otras incidencias de un tipo dado.

Entradas :

- El número de enlaces.
- El tipo de enlace.
- El tipo de condición.

Salidas:

Se produce una salida para visualización y edición que muestra los datos de configuración.

- El número de enlaces.
- El tipo de enlace.
- El tipo de condición.

Cuando no se cumple la condición del validador se muestra un mensaje de error.

Validador de campo especificado.

Crear un validador que verifique que al menos un cierto campo de una determinada lista de campos de la incidencia a realizar la transición contiene algún valor.

Entradas:

- La lista de campos.

Salidas:

- La lista de campos.

Cuando no se cumple la condición del validador se muestra un mensaje de error.

Crear el informe de avance.

Este contendrá una serie de secciones.

- Estado de las tareas:

Esta sección contendrá una tabla resumen que muestra el estado en el que se encuentran las tareas pertenecientes a cada grupo de tareas dentro del árbol del proyecto.

- Estado del comienzo de las tareas:

La sección contendrá una tabla resumen de manera similar a la anterior. Esta vez la tabla describirá el estado del comienzo de las tareas.

- Estado de la finalización de las tareas:

La sección contendrá una tabla similar a la de la sección anterior, que describirá el estado de finalización de las tareas.

- Plan de medición:

La última sección del informe de avance es el plan de medición. El plan de medición consiste en una serie de indicadores que muestran el progreso del proyecto.

Informe de estado

El informe de estado muestra el estado del proyecto.

Contiene las siguientes secciones:

- Progreso del proyecto

Esta sección contiene dos gráficos de barras, estos gráficos muestran la evolución a través del tiempo los indicadores que se calculan en el informe de avance y que describen el progreso del proyecto.

- Estado del proyecto.

Esta sección muestra tres gráficos de barras. El primero muestra el estado de desarrollo para cada uno de los módulos del proyecto. El segundo muestra el estado de desarrollo por fase del proyecto, análisis, construcción, verificación y gestión del cambio. El tercero muestra el estado de desarrollo del proyecto por fases a través del tiempo.

- Resumen de las auditorías.

Esta sección contendrá una tabla que mostrará el estado de las auditorías del proyecto.

- Estado de las auditorías abiertas:

Esta sección contendrá una tabla en la que se mostraran las auditorías abiertas.

- Seguimiento de las no conformidades abiertas:

Esta sección contendrá una tabla donde se mostrarán todas las no conformidades del proyecto y sus acciones correctivas.

- Resumen de los riesgos:

Esta sección contendrá una tabla que mostrará el número de riesgos que se encuentran en cada uno de los estados. La tabla contendrá una columna para cada uno de los estados de los riesgos: "new", "in evaluation", "evaluated", "open", "in progress", "resolved", "verified", "closed", "approved", "released" y en cada columna se indicarán el número de riesgos para cada estado.

- Estado de la evaluación de los riesgos:

Esta sección contendrá una tabla donde se mostrarán todos los riesgos del proyecto y sus acciones correctivas.

- Matriz RACI:

La matriz de la asignación de responsabilidades se utiliza generalmente para relacionar actividades con recursos

Se trata nuevamente de una tabla que contiene tareas. Las tareas que aparecen en la tabla son las tareas pertenecientes al proyecto que tienen algún valor en alguno de los campos asociados a la asignación de responsabilidades: external assignee, informed, consulted o comment.

- Plan de medición:

Tiene dos secciones:

La primera se trata de una matriz en la que se muestran ciertos indicadores del proyecto dividido en las diferentes áreas de proceso del proyecto.

La segunda sección donde están incluidos los indicadores y medidas del maestro de indicadores que muestran información sobre estado del proyecto.

4. ANÁLISIS Y DISEÑO

A continuación se expone la fase de análisis del proyecto. El objetivo de la fase de análisis es determinar las metas o fines hacia las cuales se quiere llegar, los resultados deseados ¿Qué es lo que se requiere?.

Durante esta fase se obtendrá una descripción lógica o modelo del sistema que se va a desarrollar, así como del sistema en el que se encuentra incluido.

Para ello se hará uso de las herramientas de modelado UML. El lenguaje de modelado unificado es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad .

4.1 CASOS DE USO

Los diagramas de casos de uso documentan el comportamiento de un sistema desde el punto de vista del usuario. Por lo tanto los casos de uso determinan los requisitos funcionales del sistema, es decir, representan las funciones que un sistema puede ejecutar.

Su ventaja principal es la facilidad para interpretarlos, lo que hace que sean especialmente útiles en la comunicación con el cliente.

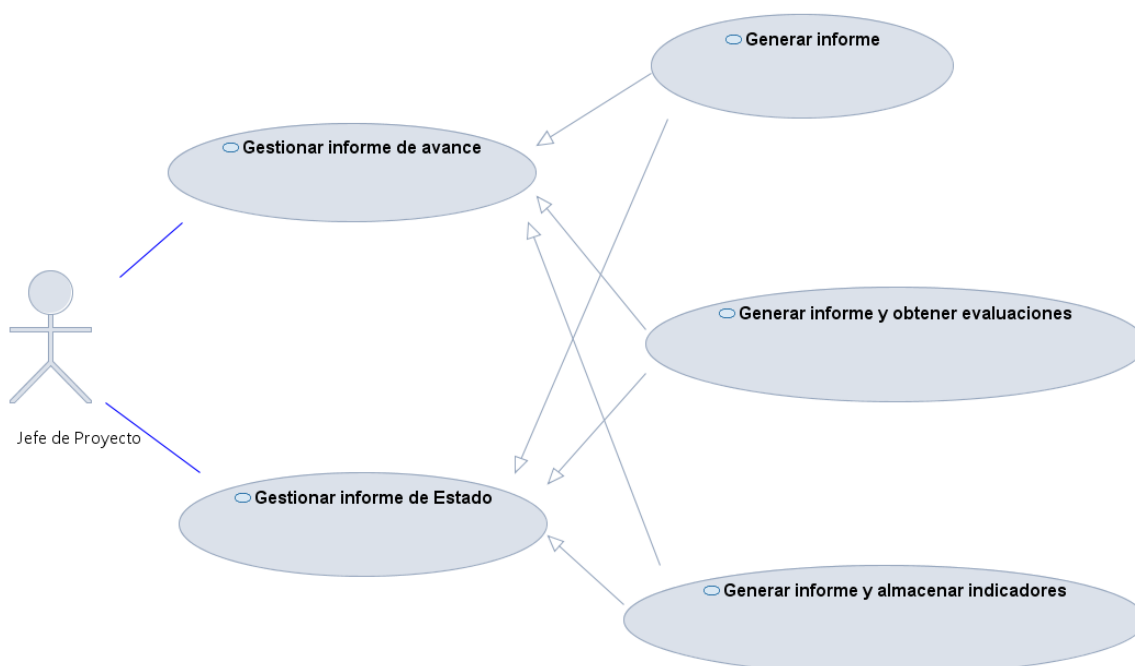


Figura 3: Casos de uso del jefe de proyecto

El jefe de proyecto gestiona los informes tanto de avance como de estado.

Generar el informe, genera todas las secciones de los informes de avance y estado que no precisan de evaluación, al no precisar de evaluación, no se produce la conexión con la base de datos de métricas y la penalización de tiempo que supone esta.

El proyecto cumple con una estructura en forma de árbol según la estructura de descomposición del trabajo WBS.

El jefe de proyecto selecciona el proyecto sobre el que quiere generar un informe ya sea de avance o de estado, configura el informe rellenando los parámetros de entrada si los hay y genera el informe.

Generar el informe y obtener evaluaciones, genera el proyecto, calcula los indicadores y calcula las evaluaciones para los indicadores. Un indicador se puede evaluar a rojo, verde o amarillo y depende de los rangos máximos y mínimos establecidos en la base de datos.

Se necesitan ciertos datos para el cálculo de los indicadores, estos datos se almacenan en incidencias (issues) especiales como la incidencia del proyecto, o la incidencia del informe de estado.

Generar el informe y almacenar los indicadores, genera el informe y calcula las evaluaciones como se ha descrito anteriormente. Además almacena las evaluaciones de los indicadores del proyecto en la base de datos de métricas.

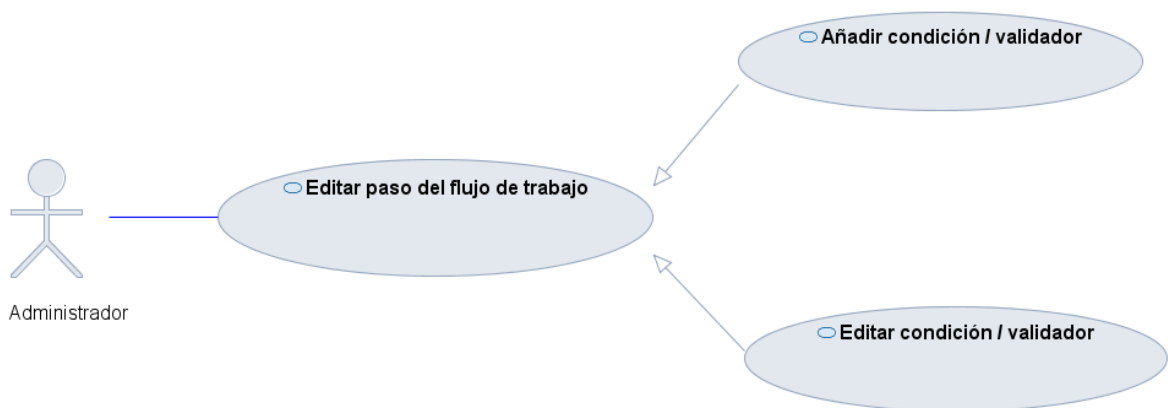


Figura 4: Casos de uso del administrador de JIRA

El administrador de la aplicación JIRA es el encargado de la creación y edición de los flujos de trabajo personalizados.

Añadir condición o validador. El administrador de sistemas añade una condición o validador a una transición de un flujo de trabajo. El administrador debe configurar si los hay, los parámetros de la condición o validador.

Editar condición o validador. El administrador de sistemas edita los parámetros de configuración de una condición o un validador ya existente en una transición de un flujo de trabajo.

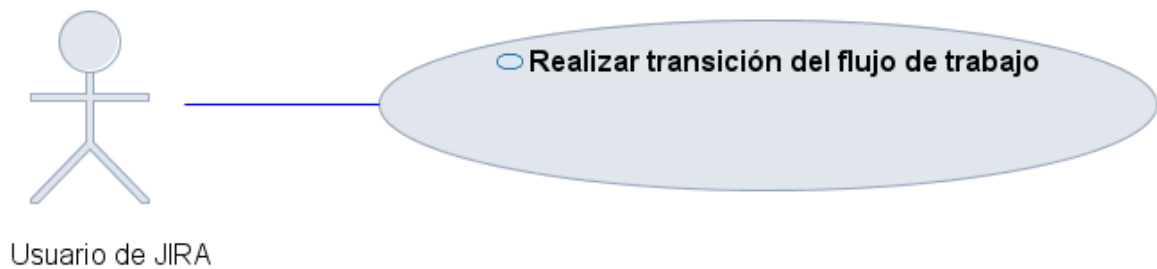


Figura 5: Casos de uso de los usuarios de JIRA

Los usuarios de la aplicación pueden realizar una transición del flujo de trabajo sobre una incidencia dada dependiendo de las condiciones y validadores. La transición se podrá producir si se cumple la condiciones y se llevará a cabo o se producirá un mensaje de error dependiendo de los validadores.

4.2 DIAGRAMAS DE CLASES

Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos.

A continuación se expondrá una breve descripción del la aplicación JIRA.

El proyecto está representado por la clase *Project*.

La clase *User* representa a los usuarios de JIRA, estos pueden pertenecer a uno o varios proyectos, además pueden pertenecer a uno o varios grupos de usuarios, grupos de usuarios típicos de JIRA pueden ser: jefe de proyecto, desarrolladores, administradores).

Los usuarios pertenecientes a un grupo de usuarios tienen una serie acciones que pueden realizar sobre un proyecto según el esquema de permisos asociado al proyecto.

Los esquemas de flujo de trabajo, *Workflow Scheme*, asocian flujos de trabajo con tipos de incidencias.

Los flujos de trabajo están representados por la clase *Workflow*. Los flujos de trabajo contienen una serie de pasos. Los pasos representan a un estado de la incidencia y contienen una serie de transiciones a otros estados.

La clase *Screen*, representa vistas, las vistas pueden contener pestañas, *Tab*, que contienen campos definidos por los usuarios *Custom Field* o campos propios de JIRA. Las vistas se pueden asociar a operaciones, como por ejemplo las operaciones que se producen sobre las incidencias, *Issue Operation*. De esta forma cada vez que se realiza una operación como por ejemplo crear una incidencia, editar una incidencia, ver la pantalla de una incidencia, realizar una transición de un flujo de trabajo etc.. se puede definir una serie de vistas que se mostraran.

Un proyecto tiene asociado un esquema de configuración de campos que define un campo para un cierto tipo de incidencia, estos campos pueden ser requeridos, *mandatory*, cuando un campo tiene la propiedad *mandatory* el campo no puede estar vacío, Los campos también pueden estar ocultos o visibles.

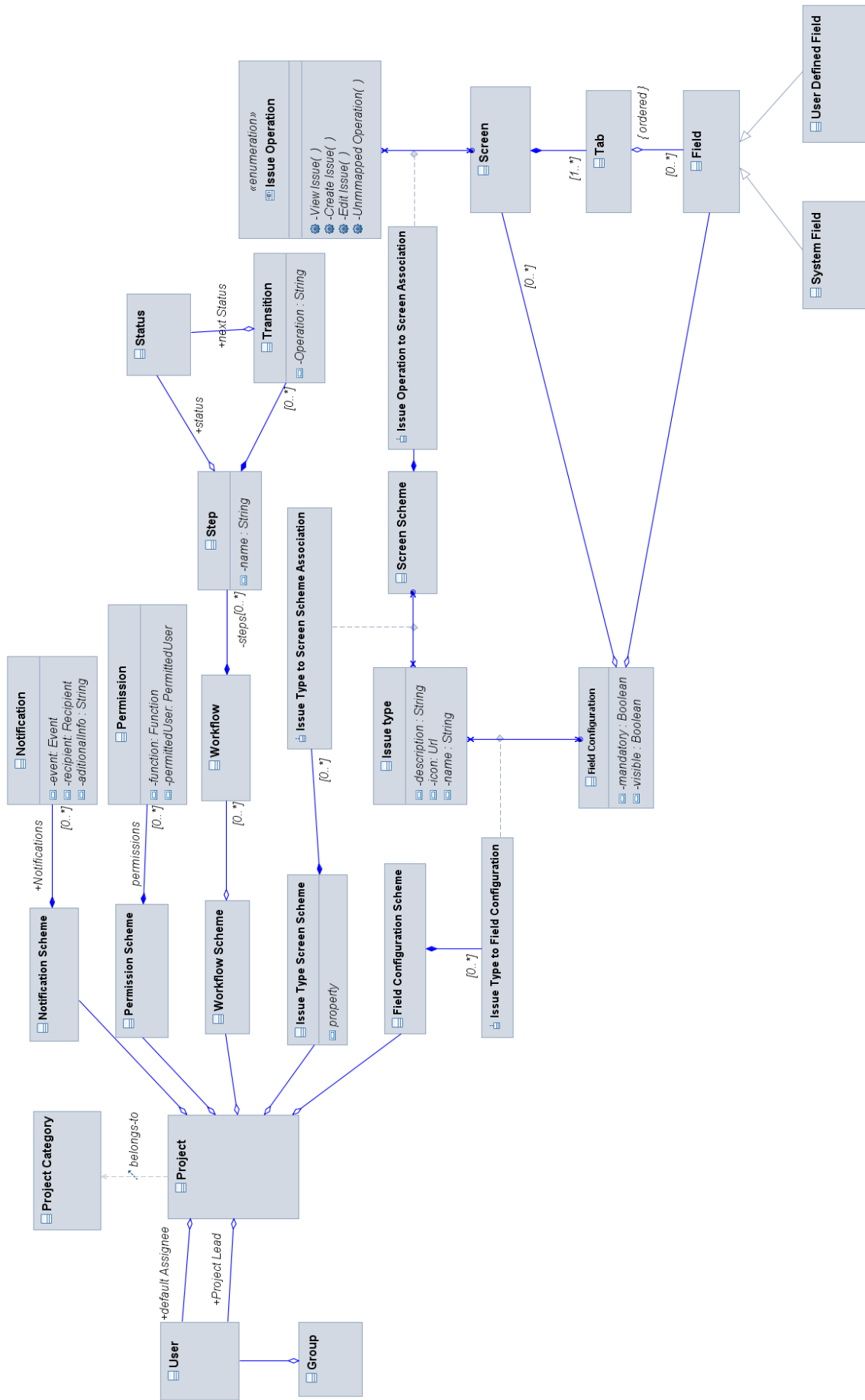


Figura 6: Diagrama de clases de la aplicación JIRA

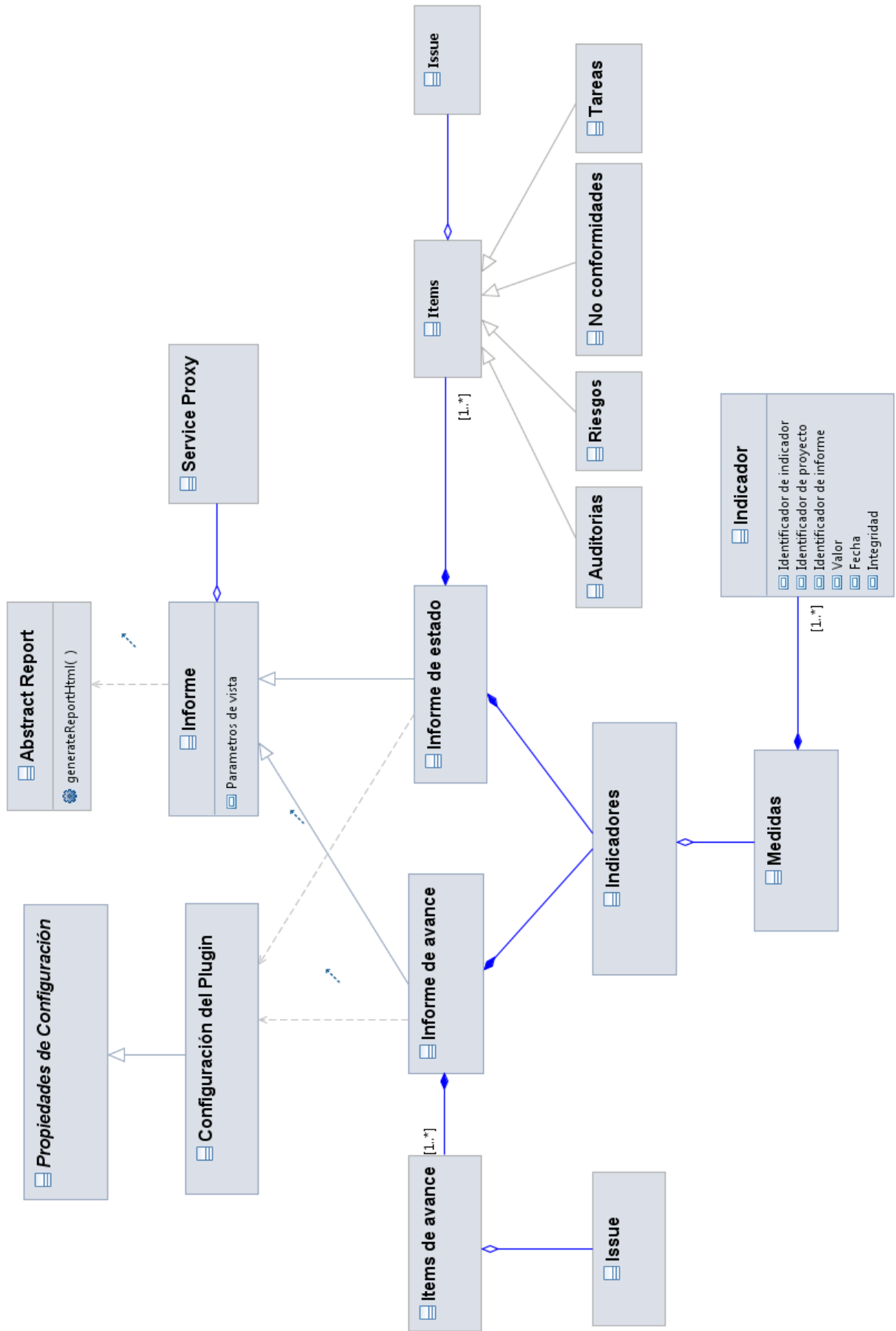


Figura 7: Diagrama de clases de la parte de los informes del plugin

La aplicación cuenta con dos clases encargadas de gestionar la configuración del plugin. Estas clases se encargan de identificar las personalizaciones propias (tipos de incidencias propias, campos propios, estados del flujo de trabajo propios, resoluciones de las incidencias propias), de la instancia de JIRA en la que el plugin se está ejecutando.

La clase *Propiedades de Configuración* contiene las funciones asociadas a la configuración, como por ejemplo leer el archivo de propiedades, mientras que la clase *Configuración del Plugin* contiene constantes utilizadas en el plugin.

Las clases *Informe de estado* e *Informe de avance* son las encargadas de modelizar los dos informes, ambas clases extienden la clase *Informe*.

La clase *Informe* contiene ciertas funciones y parámetros comunes a todos los informes del plugin, como por ejemplo la referencia a la clase *Service Proxi*, encargada de la comunicación con el servicio web y la base de datos, o los parámetros enviados a la capa de presentación, *Parámetros de vista*.

La clase *Informe* implementa la clase *Abstract Report*. Que es una clase perteneciente al API de JIRA y que le indica a JIRA que la clase ejecutada es un informe. La clase *Abstract Report* contiene el método `generateReportHTML` que debe ser implementado por todas las clases que modelan los informes, en este caso las clases *Informe de estado* e *Informe de Avance*.

Los informes contienen una clase *Indicadores* que se encarga de calcular los distintos indicadores. La clase *Indicadores* devuelve un objeto de la clase *Medidas* que es un conjunto de objetos de la clase *Indicador*.

Los indicadores contienen una serie de campos:

- Un identificador de indicador, cada indicador contiene un identificador que lo define tal como se muestra en el maestro de indicadores.
- Un identificador de proyecto que define el proyecto de donde se ha tomado la medida.
- Un identificador de informe, que identifica el tipo de informe, estado o avance, y si el informe es de estado y el modo es iteración I2, es decir el informe pertenece a una iteración

del proceso I2, identifica la iteración a la que pertenece el indicador.

- Valor es la medida tomada por el indicador.
- Fecha indica la fecha en la que se ha tomado la medida.
- Integridad indica si se ha podido tomar la medida de forma correcta o no se ha podido tomar, debido a la falta de datos o por si se ha producido un error o excepción.

Los informes contienen una serie de Items, estas clases envuelven a la clase *Issue* (*Incidencia*) del API de JIRA son las encargadas de formatear la información referente a las incidencias. Estas clases son enviadas a la capa de presentación y contienen funciones como *getFormatDate*.

5. IMPLEMENTACION

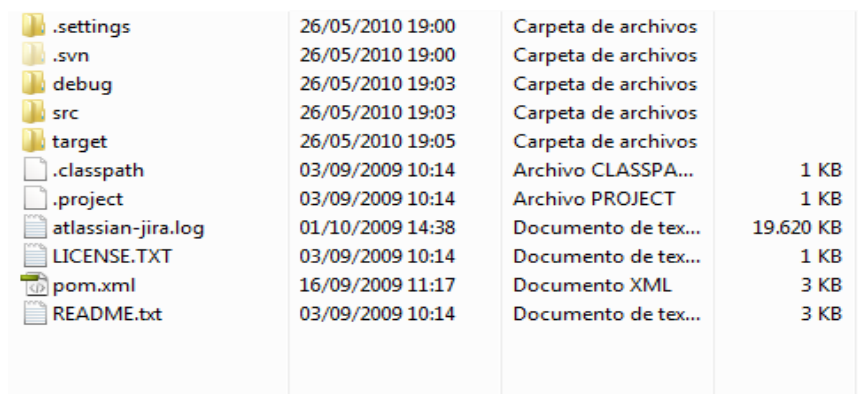
5.1 ENTORNO DE DESARROLLO

En este apartado se va a presentar el proceso de desarrollo de la aplicación.

Para el desarrollo del plugin de JIRA se ha utilizado el JIRA Plugin Development Kit. JIRA Plugin Development Kit es una herramienta proporcionada por Atlassian, se puede integrar con un IDE (Eclipse, IDEA, Netbeans) de forma que se crea un proyecto para el IDE escogido con todos los elementos necesarios para desarrollar el plugin. Para el desarrollo del plugin se ha elegido Eclipse como entorno de desarrollo.

JIRA Plugin Development Kit utiliza Apache Maven, el cual se encarga del proceso de compilado del plugin y de obtener las dependencias necesarias.

Maven es una herramienta de software para la gestión y construcción de proyectos Java. Maven utiliza un Project Object Model (POM), El Project Object Model es un archivo XML utilizado para describir el proyecto de software a construir, sus dependencias de otros módulos y componentes externos, y el orden de construcción de los elementos.



.settings	26/05/2010 19:00	Carpeta de archivos	
.svn	26/05/2010 19:00	Carpeta de archivos	
debug	26/05/2010 19:03	Carpeta de archivos	
src	26/05/2010 19:03	Carpeta de archivos	
target	26/05/2010 19:05	Carpeta de archivos	
.classpath	03/09/2009 10:14	Archivo CLASSPA...	1 KB
.project	03/09/2009 10:14	Archivo PROJECT	1 KB
atlassian-jira.log	01/10/2009 14:38	Documento de tex...	19.620 KB
LICENSE.TXT	03/09/2009 10:14	Documento de tex...	1 KB
pom.xml	16/09/2009 11:17	Documento XML	3 KB
README.txt	03/09/2009 10:14	Documento de tex...	3 KB

Figura 8: Detalle del espacio de trabajo del entorno de desarrollo

La carpeta *src* contiene el código del plugin. En la carpeta *debug* se encuentra una instancia de JIRA lista para ser ejecutada como servidor de pruebas.

5.3 DIAGRAMAS DE ESTRUCTURAS

Un diagrama de estructura compuesta es un tipo de *diagrama de estructura estática* en el Lenguaje de Modelado Unificado (UML), que muestra la estructura interna de una clase y las *colaboraciones* que esta estructura hace posibles. Esto puede incluir *partes* internas, *puertas* mediante las cuales, las partes interactúan con cada una de las otras o mediante las cuales, instancias de la clase interactúan con las partes y con el mundo exterior, y *conectores* entre partes o puertas. Una *estructura compuesta* es un conjunto de elementos interconectados que colaboran en tiempo de ejecución para lograr algún propósito. Cada elemento tiene algún *rol* definido en la colaboración.

La figura 9 muestra el diagrama de estructuras de la aplicación JIRA, es importante conocer el diagrama de estructuras para comprender el funcionamiento de JIRA, como interactúan sus partes y conocer las tecnologías asociadas.

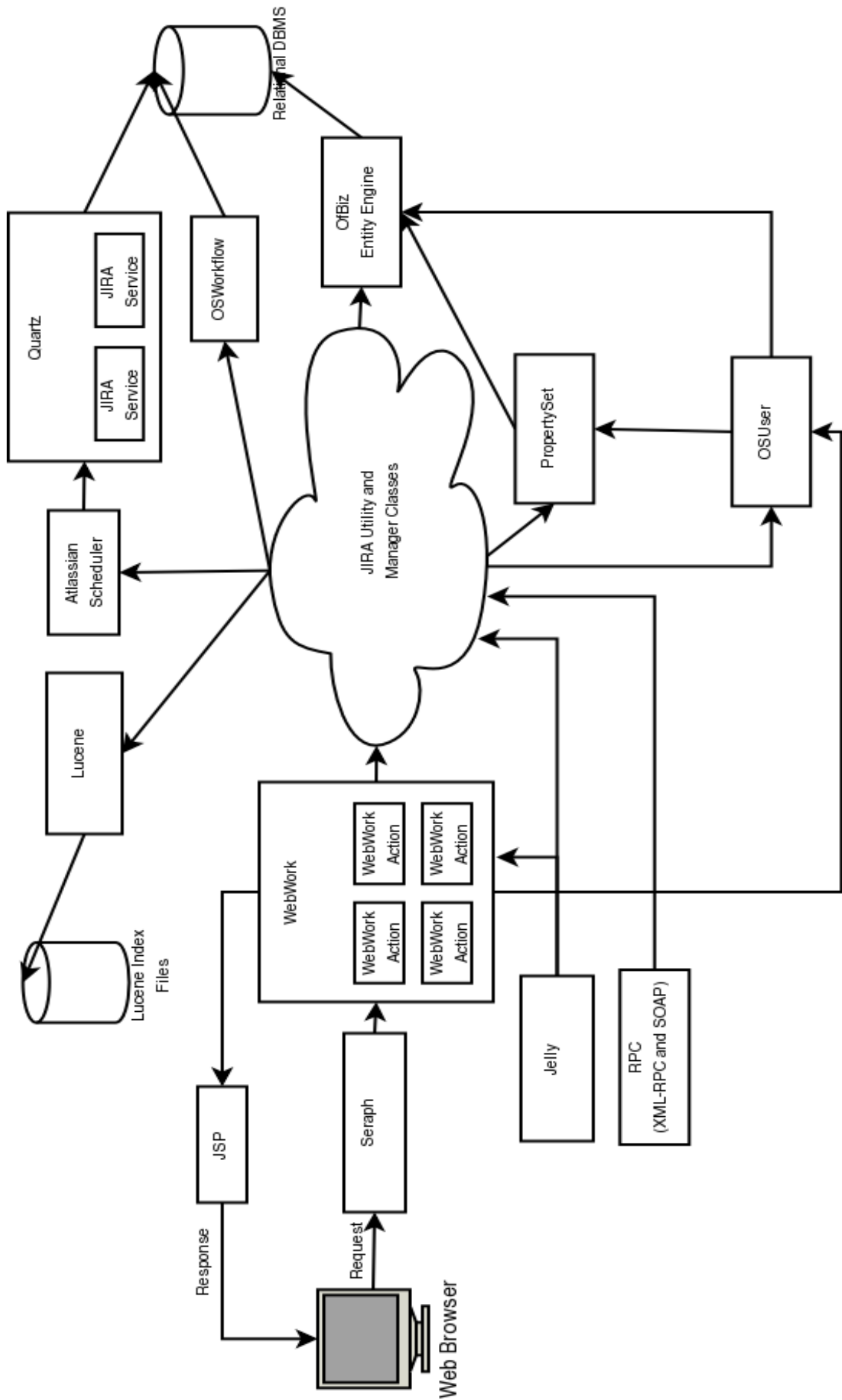


Figura 9: Diagrama de estructuras de la aplicación JIRA

Webwork.

JIRA es una aplicación WEB, los usuarios interactúan con la aplicación a través de un navegador. JIRA utiliza el marco de trabajo (Framework), WebWork del proyecto OpenSymphony para procesar las peticiones WEB enviadas por los usuarios. Cada petición es manejada por una acción de WebWork que normalmente usará otras clases, como utilidades y manejadores para cumplir su tarea.

JIRA utiliza JSP para la capa de presentación. Así que la mayoría de las páginas que se sirven al usuarios en respuesta a sus peticiones son generadas por un JSP. Por lo tanto, para generar una respuesta la acción de WebWork (WebWork Action) utiliza JSP.

El desarrollo de los plugins se simplifica, dado que JIRA se encarga de manejar las peticiones WEB. El plugin solo tiene que recoger los parámetros de entrada, y salida.

Seraph.

Toda la autenticación en JIRA se realiza a través de Seraph, Seraph es un marco de trabajo (Framework) que se encarga de la autenticación a través de la WEB. Seraph es implementado como un filtro, su único trabajo es dada una petición WEB, asociar esta con un usuario particular.

Dentro de la aplicación se puede hacer uso de la clase *JiraAuthenticationContext*, que se encarga del seguimiento de la sesión del usuario y del manejo de todos los parámetros personalizados como los archivos de localización i18n.

OSUser.

OSUser es un marco de trabajo (Framework) del proyecto OpenSymphony para la gestión de usuarios y grupos. OSUser utiliza PropertySet.

PropertySet.

PropertySet es un marco de trabajo (Framework) del proyecto OpenSymphony para almacenar un conjunto de propiedades (pares clave/valor) en una particular entidad con un ID único. Una entidad puede modelizar cualquier objeto. Por ejemplo OSUser utiliza PropertySet para almacenar el correo

electrónico, el nombre completo y las preferencias de los usuarios. Por lo tanto en el caso de OSUser la entidad representa a un usuario.

Utilidades de JIRA y los manejadores de clases.

Mucha de la lógica de negocio en JIRA está implementada en cientos de clases de java, estas clases pueden ser simples clases con utilidades o manejadores de objetos.

Los manejadores de objetos en JIRA son clases que suelen tener un objetivo específico, por ejemplo la clase *com.atlassian.jira.project.version.VersionManager* se utiliza para trabajar con las versiones de los proyectos, *com.atlassian.jira.issue.CustomFieldManager* se utiliza para trabajar con los campos personalizados de las incidencias.

La clase *ComponentManager* es la responsable de inicializar una gran cantidad de componentes de JIRA, *ComponentManager* utiliza *PicoContainer* para resolver todas las dependencias entre componentes, además posee varios métodos estáticos para los objetos que no se pueden instancia a través de *PicoContainer*.

PicoContainer.

JIRA utiliza PicoContainer como una factoría central de objetos. PicoContainer es el responsable de instanciar los objetos así como resolver sus dependencias, de esta forma cada objeto instanciado por PicoContainer puede instanciar otro objeto simplemente añadiéndolo como parámetro en su constructor. PicoContainer se encargara de instanciar el objeto, junto con los objetos que pudiera necesitar y devolvérselo al objeto que lo pidió en el constructor.

```
Public StatusReport(SearchProvider searchProvider, JiraAuthenticationContext authenticationContext,  
CustomFieldManager customFieldManager)
```

En el ejemplo se puede ver la cabecera del constructor de la clase *StatusReport* que modeliza el informe de estado.

Cuando se crea un objeto de la clase *StatusReport*, PicoContainer se encarga de instanciar los objetos necesarios junto con sus dependencias. En este caso son *SearchProvider* que provee métodos para realizar búsquedas de incidencias indexadas a través de Lucene, *JiraAuthenticationContext*

utilizado para obtener los datos relacionados la sesión, el usuario y sus permisos y *CustomFieldManager* utilizado para trabajar con los campos personalizados de la incidencias.

Lucene.

Lucene es un proyecto de la fundación Apache, JIRA utiliza Lucene en las búsquedas como apoyo de la base de datos. De esta forma se mejora el rendimiento en las búsquedas. Además JIRA provee de clases específicas para realizar búsquedas a través del interfaz de desarrollo.

Ofbiz.

JIRA usa el modulo Entity Engine de la suite Ofbiz para comunicarse con la base de datos. Entity Engine es un conjunto de herramientas y patrones para modelar y gestionar entidades específicas de datos. En este contexto una entidad es un conjunto de datos definidos por un conjunto de campos y un conjunto de relaciones con otras entidades.

La figura 10 muestra un diagrama de estructuras sobre el contexto en el que se encuentra el plugin.

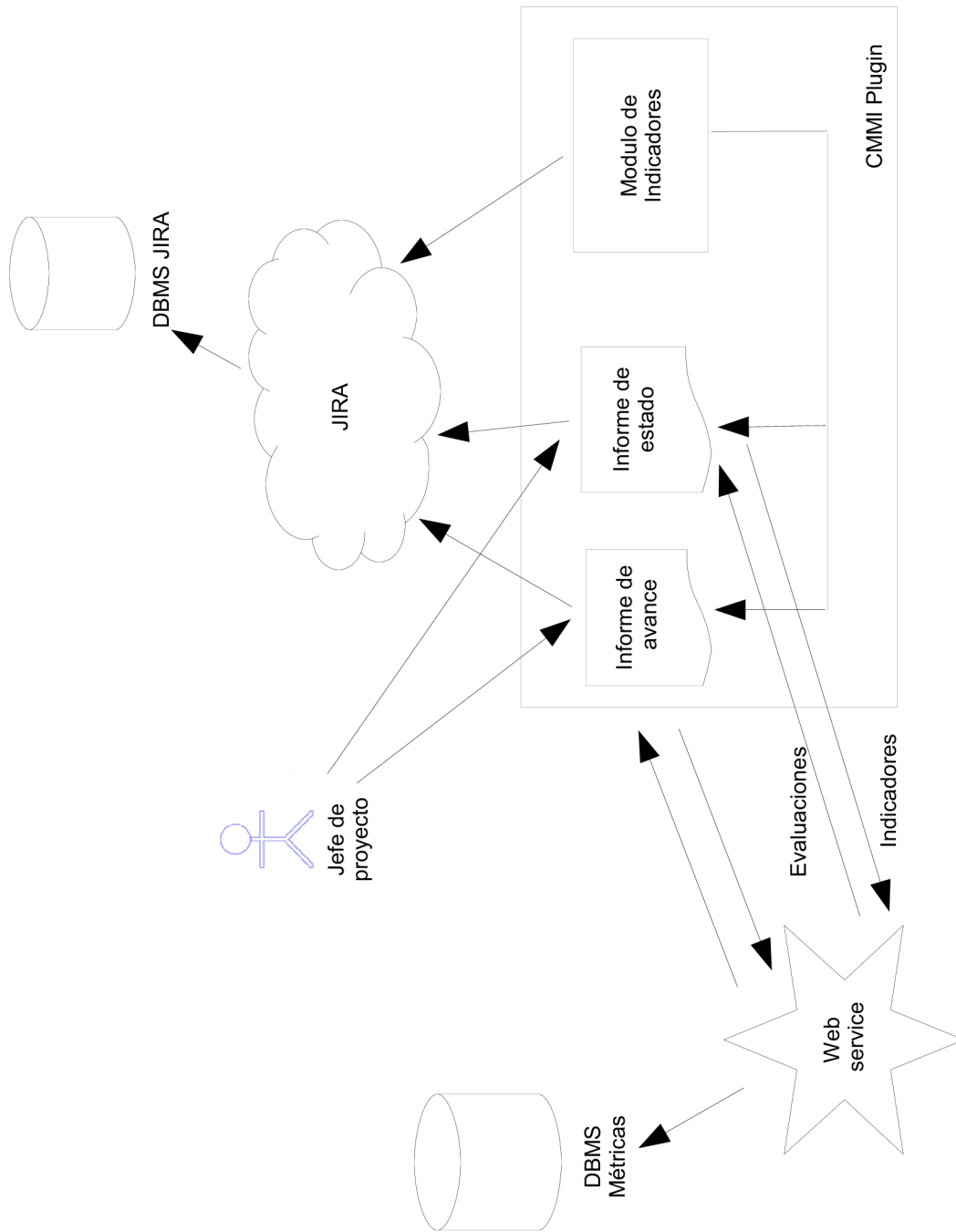


Figura 10: Componentes del sistema

El plugin CMMI esta integrado dentro de la aplicación JIRA. Tanto el informe de estado, el informe de avance y el modulo de indicadores obtienen la información necesaria a través del API de JIRA.

En la base de datos Métricas se encuentra el maestro de indicadores que contiene la definición de los indicadores así como los datos necesarios para obtener las evaluaciones.

Los jefes de proyecto generan el informe de avance o el informe de estado. Estos informes obtienen la información de los proyectos a través del API de JIRA, el modulo de indicadores calcula los indicadores para cada informe.

Los indicadores se envían al servicio web que según la acción a realizar calcula las evaluaciones o almacena los indicadores en la base de datos.

5.2 ESTRUCTURA DEL PLUGIN

Una vez configurado el entorno de desarrollo, un plugin de JIRA se compone de las clases java que contienen la lógica de negocio de la aplicación, plantillas de velocity que forman la capa de presentación junto con otros recursos externos que pudiera haber, archivos javascript, plantillas CSS archivos de propiedades. Por ultimo un plugin contiene un archivo descriptor en lenguaje XML.

El archivo descriptor define el plugin, este archivo es leído por la propia aplicación JIRA, y contiene la información referente al plugin, los módulos que contiene el plugin, tipos de módulos (informes, funciones de workflow, portlets) nombres e información referente a cada modulo y por ultimo las rutas de las clases java y recursos utilizados en el modulo.

Velocity es un motor de plantillas basado en Java. Le permite a los diseñadores de páginas hacer referencia a métodos definidos dentro del código Java de acuerdo al modelo de Modelo-Vista-Controlador (MVC). Velocity separa el código Java de las páginas Web, haciendo el sitio más mantenible a largo plazo y presentando una alternativa viable a Java Server Pages (JSP) o PHP.

La estructura del plugin tendrá la siguiente forma.

La carpeta Java: contiene las clases java.

La carpeta resources: donde se encuentra el archivo descriptor junto con los distintos recursos.

.svn	06/05/2010 14:33	Carpeta de archivos	
java	06/05/2010 14:33	Carpeta de archivos	
resources	06/05/2010 14:33	Carpeta de archivos	

Figura 11: Detalle de la estructura del proyecto

.svn	06/05/2010 14:33	Carpeta de archivos	
es	06/05/2010 14:33	Carpeta de archivos	
templates	06/05/2010 14:33	Carpeta de archivos	
atlassian-plugin.xml	04/05/2010 12:16	Documento XML	19 KB

Figura 12: Detalle de la carpeta donde se encuentran los recursos

A continuación se muestra una parte del archivo descriptor con la información del plugin y la información de una condición de flujo de trabajo.

```
<plugin-info>
  <description>${project.description}</description>
  <version>${project.version}</version>
  <vendor name="TSB" url="http://www.tsbtecnologias.es" />
</plugin-info>
```

Project referencia al archivo POM utilizado por maven, la aplicación JIRA lee las propiedades descripción y número de versión del archivo POM.

```
<workflow-condition key="link-condition" name="Link condition"
  class="es.tsbsoluciones.jira.cmmi.workflow.linkcondition.LinkConditionFactory">
  <description>
    Workflow condition to verify that the current issue contains at least one link of a given type
    (or at least n links, or exactly n links, etc.).
  </description>
  <condition-class>
    es.tsbsoluciones.jira.cmmi.workflow.linkcondition.LinkCondition
  </condition-class>
```

```

<resource type="velocity" name="view"
    location="templates/workflow/validator/linkvalidator/link-validator-view.vm" />
<resource type="velocity" name="input-parameters"
    location="templates/workflow/validator/linkvalidator/link-validator-edit-params.vm" />
<resource type="velocity" name="edit-parameters"
    location="templates/workflow/validator/linkvalidator/link-validator-edit-params.vm" />
</workflow-condition>

```

Esta parte del archivo descriptor pertenece a la condición de incidencia enlazada. La condición contiene dos clases, *LinkConditionFactory*, y *LinkCondition*.

Todos los módulos de flujo de trabajo (condiciones, validadores y postfunciones) pueden contener dos clases.

Una clase principal definida por la propiedad *condition-class* en el descriptor del ejemplo, que contiene la lógica de negocio. Esta clase en caso de ser una condición o un validador de flujo de trabajo debe implementar el interfaz *Condition* o *Validator* del API de JIRA.

Una clase factoría definida por la propiedad *class* dentro de *workflow-condition* que se encarga del envío y recepción de la información entre la clase principal y las plantillas de velocity.

La propiedad *resource* define los recursos del modulo que como se ha mencionado anteriormente pueden ser plantillas de velocity, plantillas de CSS, javascript, imágenes etc. En este caso solo se utilizan plantillas de velocity. La propiedad *name* define si las plantillas de velocity pertenecen a la entrada de parámetros, a la edición o a la vista.

A continuación se muestra la parte del descriptor referente al informe de avance.

```

<report key="Advance" name="Project progress report"
    class="es.tsbsoluciones.jira.cmmi.report.advance.AdvanceReport">
<description key="report.advance.description">i18n description</description>
<resource type="velocity" name="view"
    location="templates/report/advance/advance-report.vm"/>
<resource type="i18n" name="i18n"
    location="es.tsbsoluciones.jira.cmmi.report.advance.advance_report"/>
<label key="report.advance.label.extended" />

<properties>
    <property>

```

```

        <key>mode</key>
        <name></name>
        <type>select</type>
        <values>
            <value>
                <key>generate</key>
                <value>Generate the report</value>
            </value>

            <value>
                <key>measures</key>
                <value>Generate the report and calculate the measures</value>
            </value>

            <value>
                <key>save</key>
                <value>Generate the report and save the measures</value>
            </value>
        </values>
    </property>
</properties>
</report>

```

La propiedad *class* define la clase principal, que como se ha comentado anteriormente debe de extender la clase *Abstract Report* del API de JIRA y debe sobrescribir el método `generateReportHTML`.

Como recursos se encuentran el archivo de `velocity`, donde se muestra la salida del informe. Y un archivo `i18n` de internacionalización, en caso que se quiera mostrar el informe en diversos idiomas.

Por ultimo *properties* define los parámetros de entrada del informe, JIRA a través de WebWork se encarga de gestionar la entrada de parámetros para los informes.

5.4 ARQUITECTURA POR CAPAS

Los plugin desarrollados para JIRA, al contener una estructura muy marcada por la propia aplicación se integran perfectamente dentro de la arquitectura por capas.

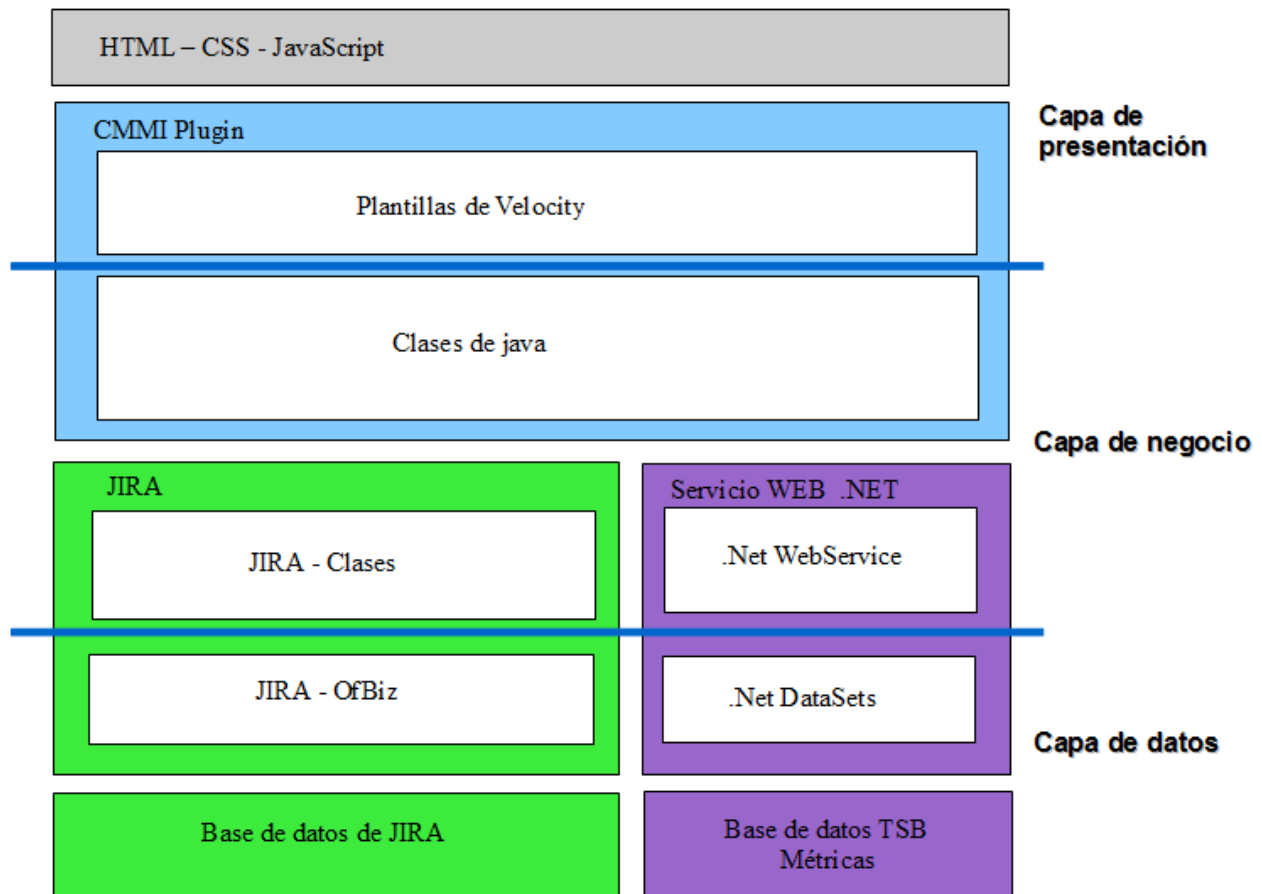


Figura 13: Estructura del sistema desde la perspectiva de capas

La capa de presentación esta formada por las plantillas de velocity junto con otros recursos como archivos CSS o javascript, como se ha comentado anteriormente.

La lógica de la aplicación esta implementada en las clases java del plugin y las clases de JIRA y del Servicio WEB.

Para el acceso a la base de datos JIRA utiliza el marco de trabajo OfBiz como se ha visto anteriormente.

El servicio web ha sido implementado con la tecnología .NET de Microsoft. El servicio WEB proporciona una serie de funciones accesibles a través de la red, estas funciones se comunican a

través del protocolo SOAP por lo que pueden ser llamadas por cualquier aplicación que entienda dicho protocolo.

SOAP (siglas de Simple Object Access Protocol) es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. SOAP fue creado por Microsoft, IBM y otros y está actualmente bajo el auspicio de la W3C. Es uno de los protocolos utilizados en los servicios Web.

Dentro del plugin de JIRA se ha implementado un cliente del servicio WEB para acceder a dichas clases. El servicio WEB interactúa con la base de datos mediante DataSets pertenecientes al marco de trabajo ADO .Net de Microsoft.

El DataSet de ADO.NET es una representación de datos residente en memoria que proporciona un modelo de programación relacional coherente independientemente del origen de datos que contiene. Un DataSet representa un conjunto completo de datos, incluyendo las tablas que contienen, ordenan y restringen los datos, así como las relaciones entre las tablas.

La base de datos se compone de tres tablas.

El maestro de indicadores contiene la información relativa a cada uno de los indicadores que se calculan en los informes de los proyectos.

La tabla del maestro de indicadores contiene los siguientes campos:

- Un id del indicador, que identifica a cada indicador de forma única.
- El nombre del indicador.
- La fórmula del indicador.
- El formato del resultado del indicador, si es de tipo numérico o es un porcentaje.
- Las unidades del resultado.
- El ámbito en el que se encuentra el indicador.

- El rango dentro del que se encuentra el resultado producido.
- El responsable encargado de supervisar el resultado producido por el indicador en el proyecto. El responsable puede ser el jefe de proyecto, el director técnico, el responsable de calidad, el gerente.
- Frecuencia con la que se calculan las medidas relativas al indicador.
- Método.
- Origen de datos, indica de donde se obtienen los datos necesarios para el calculo del indicador.
- Canal de comunicación.
- Comentarios.

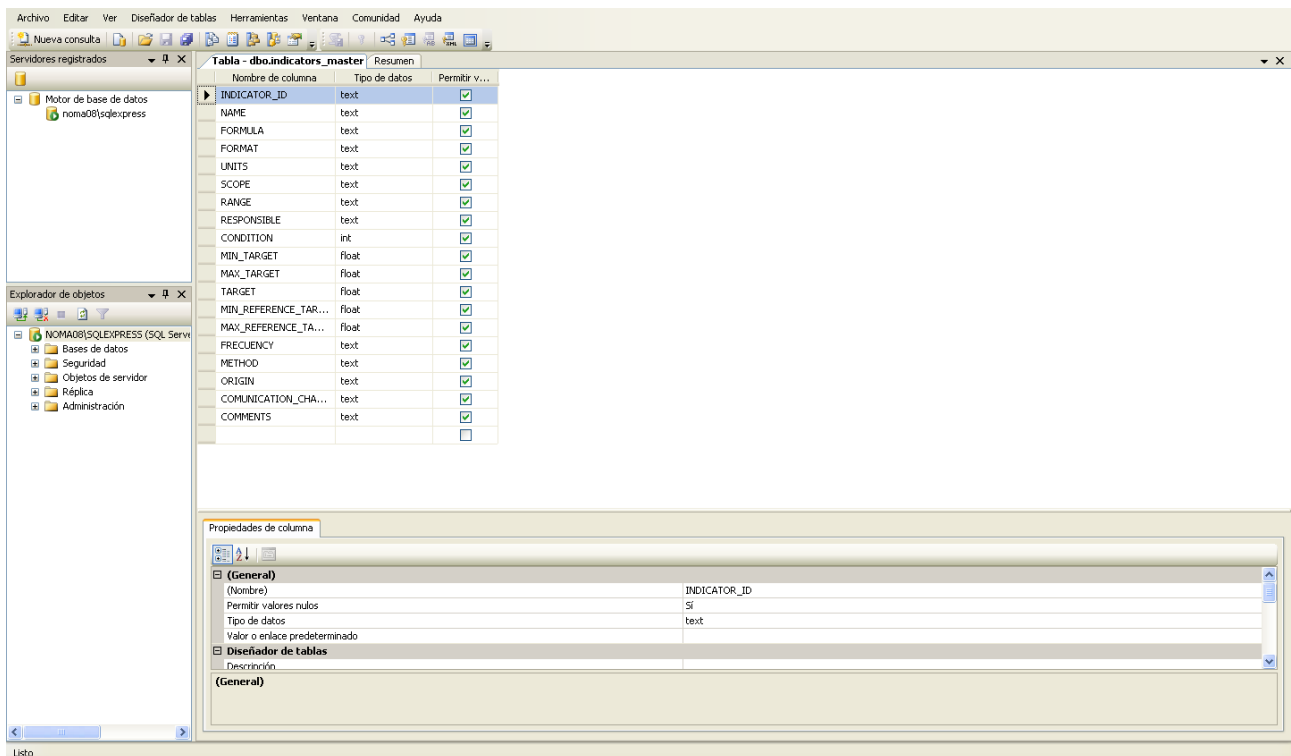


Figura 14: Estructura de la tabla maestro de indicadores de la base de datos

De tabla de los objetivos de los indicadores se obtiene la información necesaria para calcular las evaluaciones de los indicadores, las evaluaciones de los indicadores se calculan a partir de una condición y unos objetivos, estos objetivos pueden cambiar con el tiempo, los objetivos se almacenan a modo de histórico de objetivos.

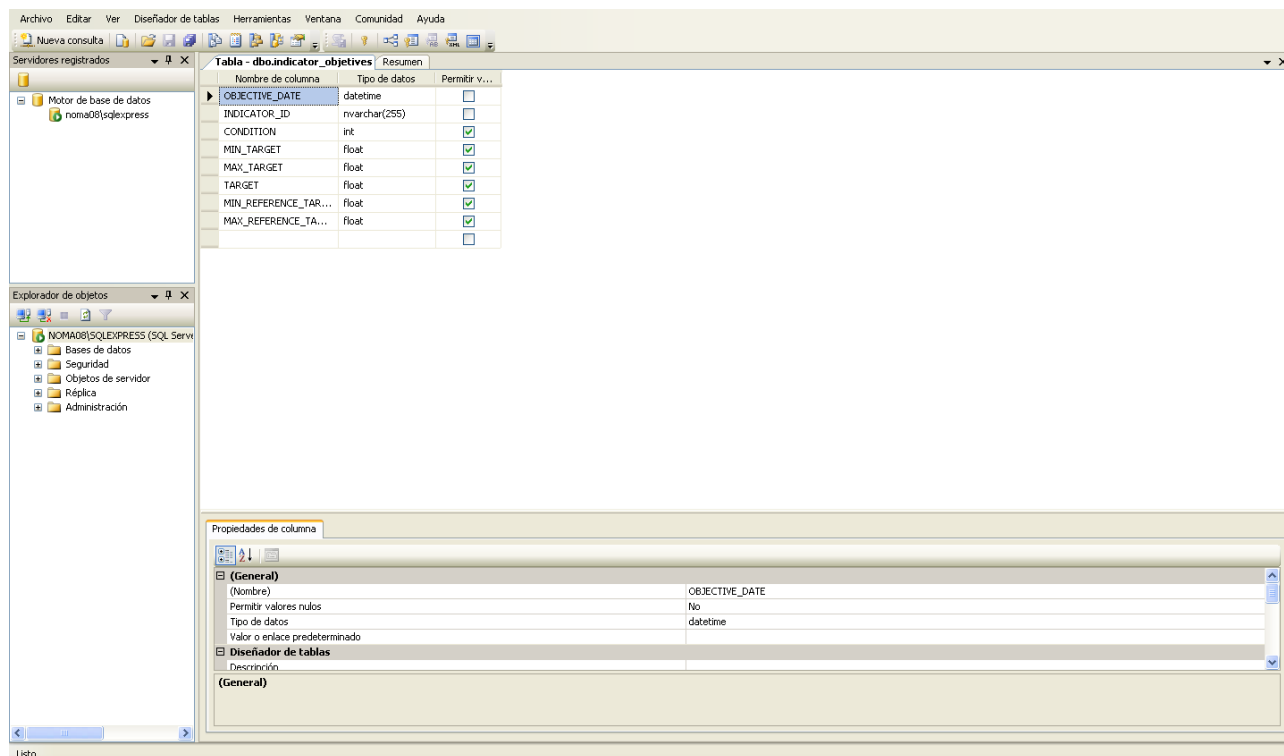


Figura 15: Estructura de la tabla objetivos de la base de datos

Cada indicador tiene un tipo de condición, esta condición puede ser; mayor que, mayor o igual que, menor que, menor o igual que, entre. Además cada indicador tiene unos objetivos mayores y menores y unos objetivos de referencia, a partir de estos objetivos se calculan las evaluaciones de los indicadores.

La tabla de medidas almacena las medidas obtenidas para cada indicador para cada proyecto en un punto dado de tiempo.

La tabla de medidas contiene los siguientes campos:

- Identificador de proyecto, identifica el proyecto al que pertenece la medida.
- Identificador de indicador, indica de que tipo de indicador es la medida almacenada.

- Identificador de informe, indica el tipo de informe en el que se ha tomado la medida, informe de estado o de avance, si el informe es de estado y de iteración, entonces en este campo se almacena la clave de la iteración a la que pertenece el informe.
- Evaluación, la evaluación que obtuvo la medida cuando fue tomada, este valor se almacena puesto que las evaluaciones para una medida pueden cambiar en el tiempo conforme cambian los objetivos.
- El valor obtenido en el cálculo del indicador.
- La fecha en la que se tomo la medida.
- El campo válido indica que la medición obtenida es válida para el cálculo de los indicadores generales de la organización.

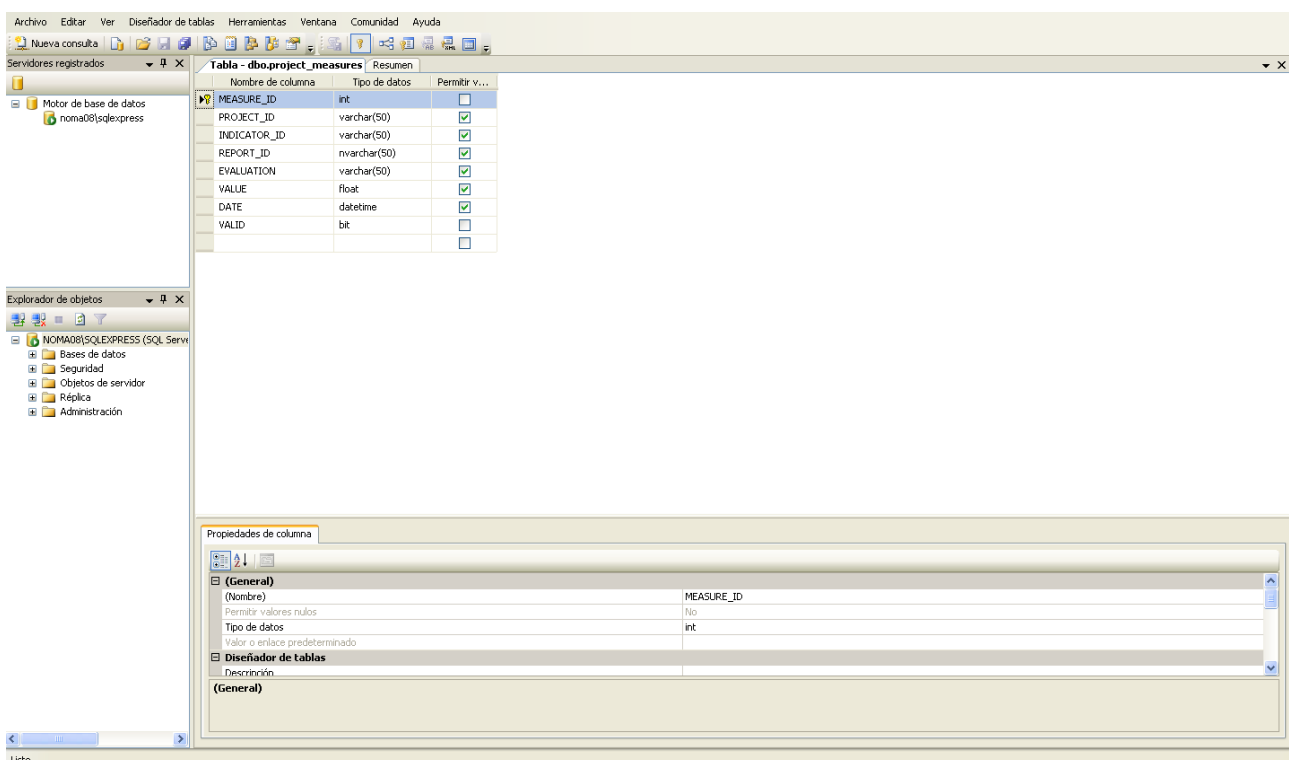


Figura 16: Estructura de la tabla medidas de la base de datos

6. RESULTADOS Y PRUEBAS

6.1 PRUEBAS

La instancia de JIRA de la empresa se ejecuta en un servidor de producción. Como se ha mencionado anteriormente JIRA Development Kit permite ejecutar en local un servidor junto con una instancia de JIRA integrado para la realización de las pruebas.

Con el Servicio Web y la base de datos se ha actuado de la misma forma, por una parte se ejecuta la instancia de producción y las pruebas se realizan en una instancia replicada en local.

Para las pruebas se ha creado un proyecto de pruebas. Sobre dicho proyecto se han ido efectuando las pruebas.

También se ha comprobado que los resultados eran correctos en el servidor de producción sobre proyectos reales. Para la comprobación de los resultados se han utilizado plantillas de excel.

Las siguientes figuras muestran ejemplos de las plantillas utilizadas en el proceso de verificación de los indicadores.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1									Analisis	Desarrollo	Validación	Cambio				
2								New	0	0	0	0				
3		Analysis Weight:	0,2		User Requirement Points:	104		In Analysis	0,1	0	0	0				
4		Development Weight:	0,65		Technical Complexity Factor:	1,02		Analysis Completed	0,2	0	0	0				
5		Verification Weight:	0,05		Use Actor Weights:	7		Awaiting Analysis Approval	0,2	0	0	0				
6		Change Management Weight:	0,1		Environment Factor:	0,905		Analysis approved	0,2	0	0	0				
7		Follow-up Weight:	0,06		Effort:	21		In design	0,2	0,325	0	0				
8		Initial Phase Weight:	0,015					Design completed	0,2	0,325	0	0				
9		Project Close Weight:	0,02					In development	0,2	0,325	0	0				
10		Deployment Weight:	0,06					Development completed	0,2	0,325	0	0				
11		Configuration Management Weight:	0,01					In test development	0,2	0,325	0					
12		Quality Assurance Weight:	0,01					Test development completed	0,2	0,65	0	0,033				
13								In review	0,2	0,65	0	0,033				
14								Review completed	0,2	0,65	0	0,033				
15								In internal testing	0,2	0,65	0,025	0,050				
16								Internal testing completed	0,2	0,65	0,05	0,100				
17								Awaiting approval	0,2	0,65	0,05	0,100				
18								Approved	0,2	0,65	0,05	0,100				
19								Released	0,2	0,65	0,05	0,100				
20								Internal testing failed	0,2	0,325	0	0				

Figura 17: Plantilla con datos para el cálculo de los indicadores de avance

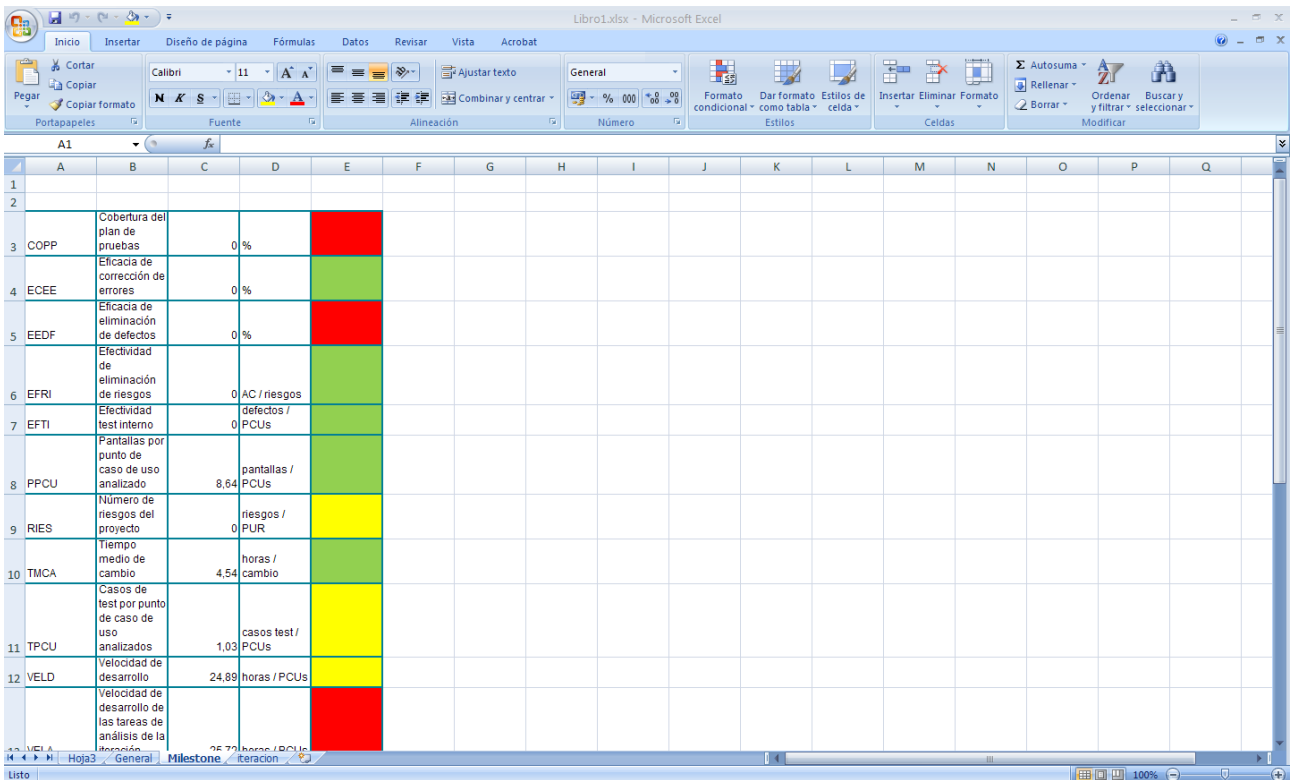


Figura 18: Plantilla con el cálculo de los indicadores de estado

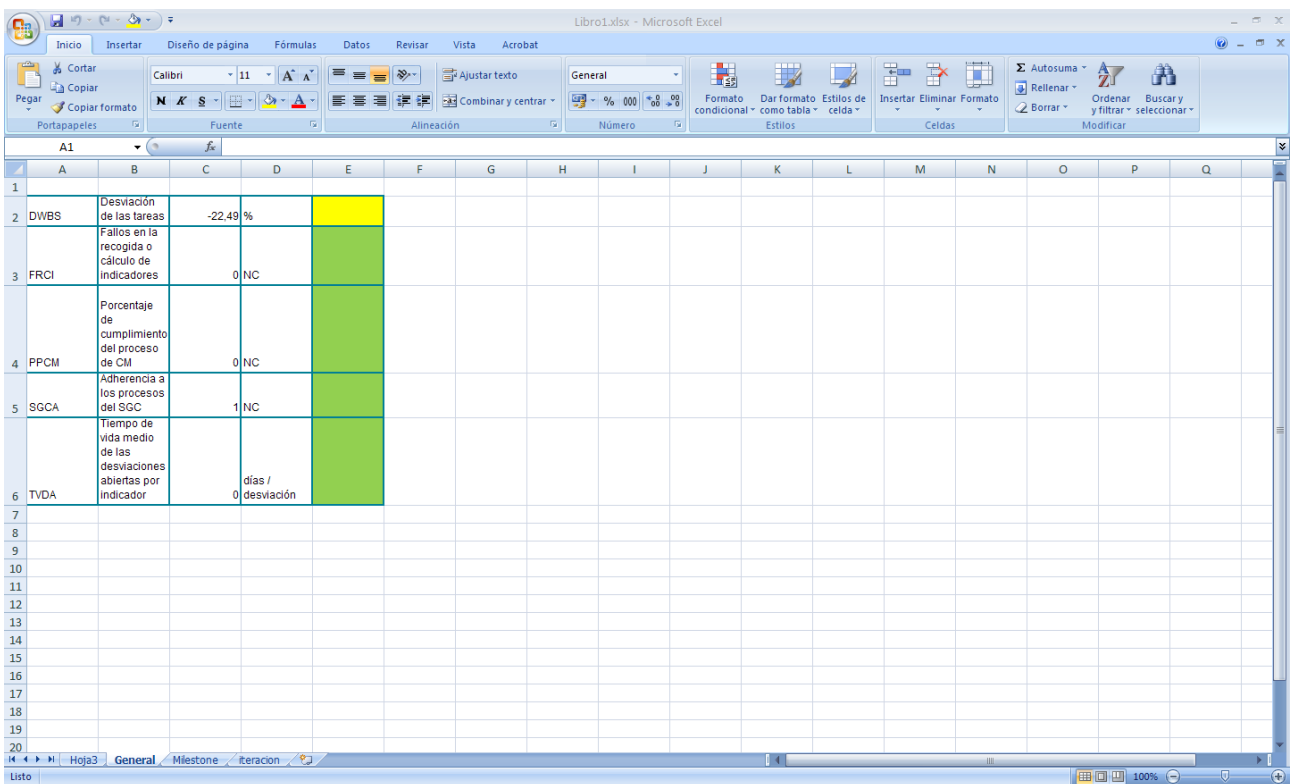


Figura 19: Plantilla con los cálculos de los indicadores de estado

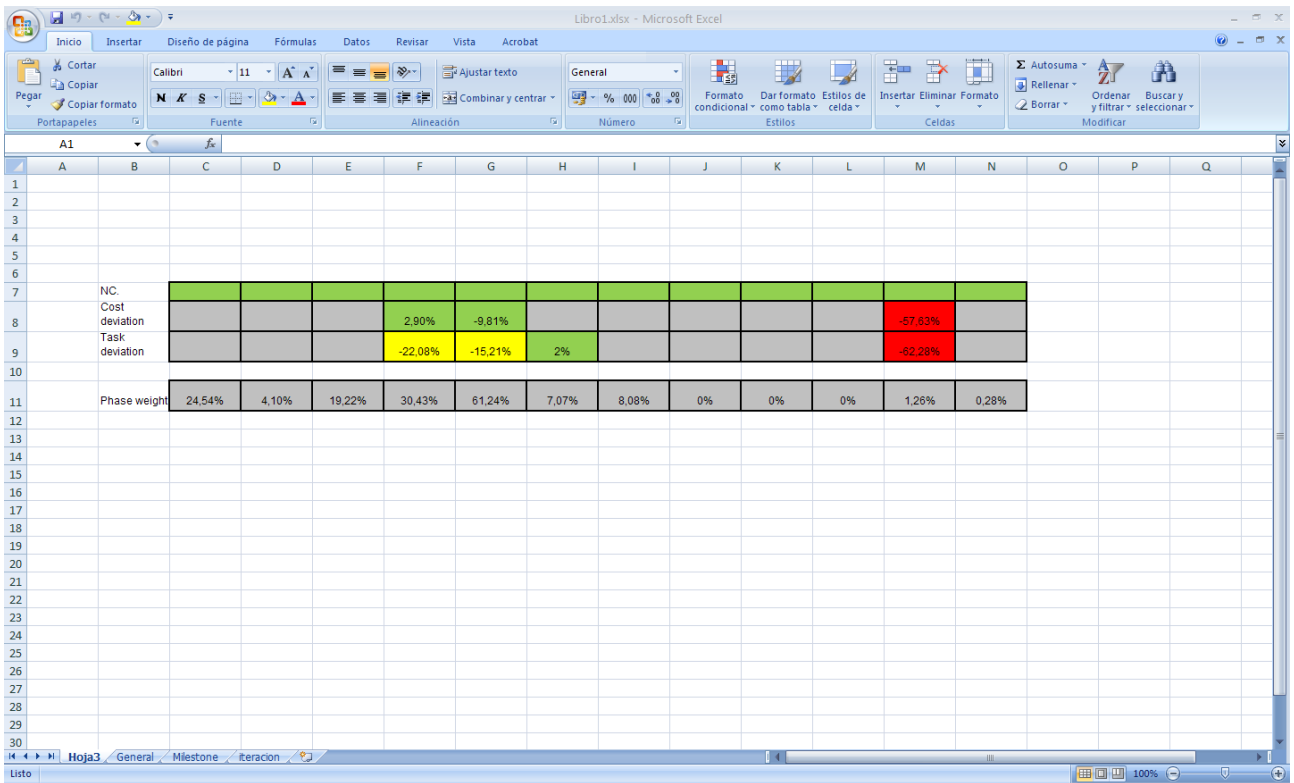


Figura 20: Plantilla con las desviaciones y los pesos de las fases

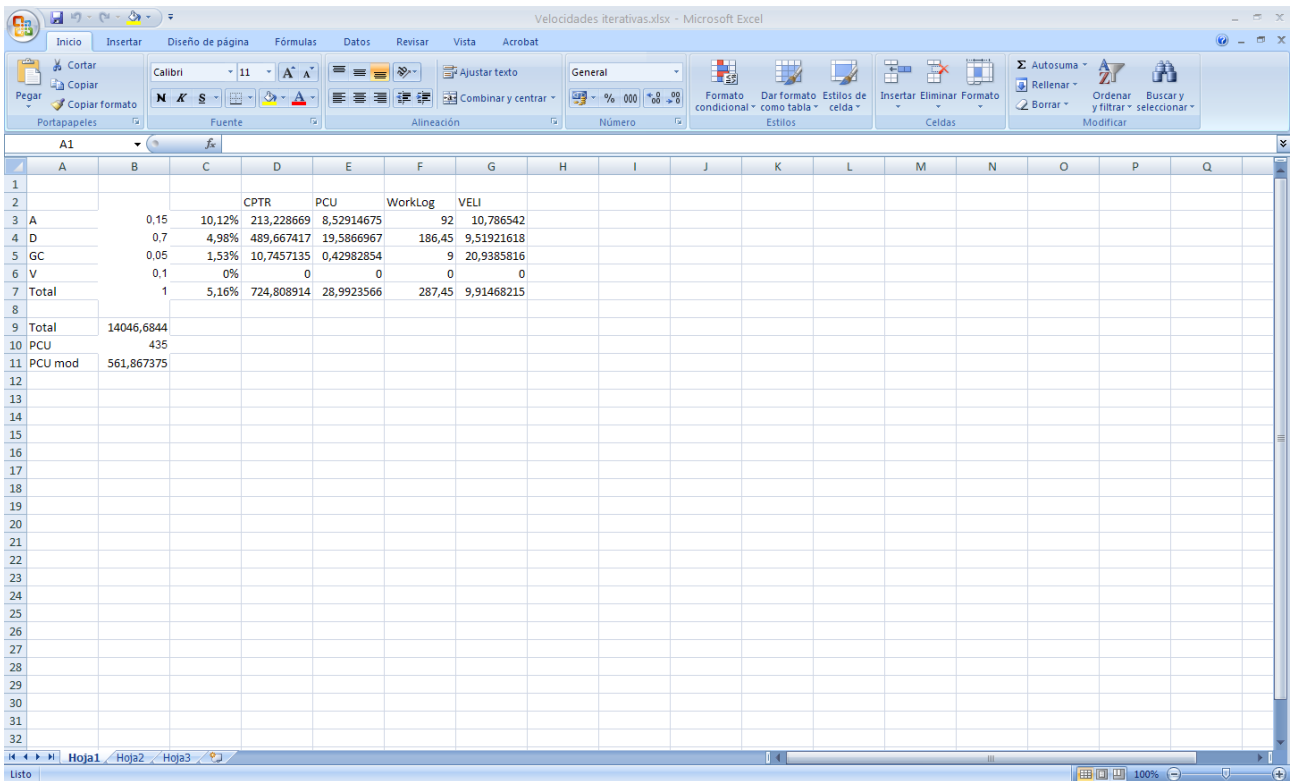


Figura 21: Plantilla con el calculo de las velocidades de desarrollo

6.2 RESULTADOS

JIRA
administrador Historial Filtros Salir

HOME NAVEGAR POR EL PROYECTO BUSCAR INCIDENCIAS NUEVA INCIDENCIA ADMINISTRACIÓN
BÚSQUEDA RÁPIDA:

▼ Proyecto

- Proyectos
- Project Categories

▼ Users, Groups & Roles

- User Browser
- Group Browser
- Project Role Browser

▼ Global Settings

- Attachments
- CVS Modules
- Default Dashboard
- Events
- General Configuration
- Global Permissions
- Issue Linking
- Look and Feel
- Mail Servers
- Subversion Repositories
- Sub-Tasks
- Time Tracking
- Trackbacks
- User Defaults
- Workflows

▼ Schemes

- Issue Security Schemes
- Notification Schemes
- Permission Schemes
- Workflow Schemes
- Scheme Tools

▼ Issue Fields

- Custom Fields
- Field Configurations
- Field Configuration Schemes
- Navigator Columns
- Screens
- Screen Schemes
- Issue Type Screen Schemes

▼ Issue Settings

- Issue Types
- Priorities
- Resolutions
- Statuses

▼ Import & Export

- Backup Data to XML
- Restore Data from XML
- External System Import
- Project Import

▼ Options & Settings

- Jelly Runner
- Send E-mail
- Announcement Banner

▼ System

- Indexing
- Integrity Checker
- LDAP
- License Details
- Listeners
- Logging & Profiling
- Mail Queue
- Plugins
- Scheduler Details
- Services
- System Info
- Support Request
- Trusted Applications

▼ Hide All

View Workflows

The table below shows the current workflows used in this version of JIRA and whether or not they are active.

All workflows have one of the following statuses:

- Active** - currently assigned to at least one scheme which is associated with one or more projects.
- Draft** - a draft version of an active workflow, which can be published to apply all changes.
- Inactive** - not assigned to any scheme, or assigned to schemes which are not associated with any projects.

To delete a workflow, you must first unassign it from any workflow schemes.

View the current [workflow schemes](#), showing how the workflows are assigned to projects and issue types.

Nombre	Descripción	Status	Schemes	Number of steps	Operaciones
jira (Read-only System Workflow)	The default JIRA workflow.	Inactive	Used by projects with no associated workflow scheme and by workflow schemes with unassigned issue types.	5	Steps XML Copy
Audits <small>Last modified on 18/mov/09 by administrador.</small>	Audits workflow	Active	<input type="checkbox"/> Scheme 1 <input type="checkbox"/> Copy 2 of Scheme 1 <input type="checkbox"/> Scheme 2	6	Steps XML Copy Create Draft
Baseline <small>Last modified on 23/jul/09 by administrador.</small>		Active	<input type="checkbox"/> Scheme 1 <input type="checkbox"/> Copy 2 of Scheme 1 <input type="checkbox"/> Scheme 2	2	Steps XML Copy Create Draft
Bug <small>Last modified on 25/sep/09 by administrador.</small>		Active	<input type="checkbox"/> Scheme 1 <input type="checkbox"/> Copy 2 of Scheme 1 <input type="checkbox"/> Scheme 2	3	Steps XML Copy Create Draft
Copy of jira <small>Last modified on 20/oct/09 by administrador.</small>	The default JIRA workflow.	Active	<input type="checkbox"/> Scheme 1 <input type="checkbox"/> Copy 2 of Scheme 1 <input type="checkbox"/> Scheme 2	5	Steps XML Copy
Copy of jira <small>Last modified on 21/ene/10 by administrador.</small>	The default JIRA workflow.	Draft	<input type="checkbox"/> Scheme 1 <input type="checkbox"/> Copy 2 of Scheme 1 <input type="checkbox"/> Scheme 2	5	Steps XML Copy Editar Borrar Publish
Copy of Task group <small>Last modified on 28/ene/09 by administrador.</small>	(This copy was automatically generated from a draft, when workflow 'Task group' was made inactive.)	Inactive		2	Steps XML Copy Editar Borrar
Risk <small>Last modified on 11/nov/09 by administrador.</small>		Active	<input type="checkbox"/> Scheme 1 <input type="checkbox"/> Copy 2 of Scheme 1 <input type="checkbox"/> Scheme 2	6	Steps XML Copy Create Draft
Task group <small>Last modified on 15/sep/09 by administrador.</small>		Inactive		2	Steps XML Copy Editar Borrar
Test Workflow <small>Last modified on 11/feb/10 by administrador.</small>	workflow for testing	Active	<input type="checkbox"/> Scheme 1 <input type="checkbox"/> Copy 2 of Scheme 1 <input type="checkbox"/> Scheme 2	5	Steps XML Copy
Test Workflow <small>Last modified on 11/feb/10 by administrador.</small>	workflow for testing	Draft	<input type="checkbox"/> Scheme 1 <input type="checkbox"/> Copy 2 of Scheme 1 <input type="checkbox"/> Scheme 2	5	Steps XML Copy Editar Borrar Publish
Use Case and Support Requirement <small>Last modified on 16/oct/09 by administrador.</small>		Inactive		20	Steps XML Copy Editar Borrar
Use Case and Support Requirements <small>Last modified on 28/dic/09 by administrador.</small>		Active	<input type="checkbox"/> Copy 2 of Scheme 1 <input type="checkbox"/> Scheme 2	20	Steps XML Copy Create Draft

Add New Workflow

To create a complete new workflow, you need to:

- provide a name and description to identify the workflow
- add the steps the workflow will have, and link them to statuses within JIRA
- create transitions between the different steps
- enable the workflow and assign it to a workflow scheme

You can create a new workflow below, or [import a workflow from XML](#).

Nombre:

Please use only ASCII characters.

Descripción:

This JIRA site is for demonstration purposes only. [Evaluate JIRA](#) - bug tracking software for your team.

Powered by [Atlassian JIRA](#) the Professional [Issue Tracker](#). (Enterprise Edition, Version: 3.13-#330) - [Bug/feature request](#) - [Atlassian news](#) - [Contact Administrators](#)

Figura 22: Vista de la pantalla de administración de workflows en JIRA

En la sección de administración (figura 17) se accede a la sección de administración de flujos de trabajo donde se muestran todos los flujos de trabajo, la descripción de estos, el estado que puede ser activo, inactivo o draft (copia de trabajo de un flujo de trabajo existente). Los flujos de trabajo activos no se pueden editar por lo que se puede hacer una copia de trabajo de un flujo de trabajo para editar y luego publicar.

Los esquemas asignan un flujo de trabajo a un proyecto y a un tipo de incidencia determinado.

Por ultimo se muestran las operaciones disponibles sobre cada flujo de trabajo.

La vista de pasos de un flujo (figura 18) de trabajo muestra los pasos del flujo de trabajo y las transiciones disponibles, donde se puede añadir una transición a un flujo de trabajo, eliminar una transición o editar una transición disponible.

View Workflow Steps — Test Workflow (Draft) ?

i You are editing a draft workflow. [View the original workflow](#) or [publish this draft](#).
This draft was last edited by you at 11/feb/10 09:29 AM.

This shows all of the steps for **Test Workflow (Draft)**. Steps that exist on the active workflow, can't be deleted from the draft workflow.

View all workflows.
 View all statuses.

Step Name (id)	Linked Status	Transitions (id)	Operaciones
Open (1)	Open	Start Progress (4) >> In Progress Resolve Issue (5) >> Resolved Close Issue (2) >> Closed	Add Transition Delete Transitions Editar View Properties
In Progress (3)	In Progress	Stop Progress (301) >> Open Resolve Issue (5) >> Resolved Close Issue (2) >> Closed	Add Transition Delete Transitions Editar View Properties
Resolved (4)	Resolved	Close Issue (701) >> Closed Reopen Issue (3) >> Reopened	Add Transition Delete Transitions Editar View Properties
Reopened (5)	Reopened	Resolve Issue (5) >> Resolved Close Issue (2) >> Closed Start Progress (4) >> In Progress	Add Transition Delete Transitions Editar View Properties
Closed (6)	Closed	Reopen Issue (3) >> Reopened	Add Transition Delete Transitions Editar View Properties

Add New Step

Step Name:

Linked Status:

Figura 23: Detalle del workflow de pruebas donde se ven los pasos y las transiciones

Como se ha visto anteriormente una transición de un flujo de trabajo puede contener condiciones validadores y postfunciones.

Al añadir una condición a un flujo de trabajo se muestra la pantalla con las condiciones disponibles (figura 19), las condiciones de flujo de trabajo de JIRA y las condiciones de flujo de trabajo añadidas a través de las extensiones (plugins). En la siguiente imagen se muestra la pantalla de condiciones disponibles.

Add Condition To Transition	
Nombre	Descripción
<input type="radio"/> Issue must have a file attached	The issue must have at least one file attached.
<input type="radio"/> Link condition	Workflow condition to verify that the current issue contains at least one link of a given type (or at least n links, or exactly n links, etc.).
<input type="radio"/> Only Assignee Condition	Condition to allow only the assignee to execute a transition.
<input type="radio"/> Only Reporter Condition	Condition to allow only the reporter to execute a transition.
<input type="radio"/> Permission Condition	Condition to allow only users with a certain permission to execute a transition.
<input type="radio"/> Related issue in status	Condition to allow to execute the transition only if all the issues related through a specific link type and complying a given pattern are in a specified status.
<input type="radio"/> Sub-Task Blocking Condition	Condition to block parent issue transition depending on sub-task status.
<input type="radio"/> User Is In Any Groups	It allows only users in any given groups to execute the transition.
<input type="radio"/> User Is In Custom field	It allows only users in custom field to execute the transition.
<input type="radio"/> User Is In Group	Condition to allow only users in a given group to execute a transition.
<input type="radio"/> User Is In Group Custom Field	Condition to allow only users in a custom field-specified group to execute a transition.
<input type="radio"/> User Is In Project Role	Condition to allow only users in a given project role to execute a transition.
<input type="radio"/> Value Field	It allows to only execute the transition if the given value of the field is equal to a specified value.

Figura 24: Pantalla para añadir una condición a la transición

De manera similar a la pantalla de condiciones, la pantalla de añadir un validador a la transición muestra los validadores disponibles.

Add Validator To Transition	
Nombre	Descripción
<input type="radio"/> Baseline validator	Validates that all configuration items with explicit versioning are included in a baseline.
<input type="radio"/> Date Compare	Compare two date fields during a workflow transition.
<input type="radio"/> Fields Required	Fields required during a workflow transition.
<input type="radio"/> Link validator	Workflow validator to verify that the current issue contains at least one link of a given type (or at least n links, or exactly n links, etc.).
<input type="radio"/> Linked issue validator	Workflow validator that checks that at least a link of a given type with an issue of a given type has been specified.
<input type="radio"/> One field has been specified	Workflow validator like the required field validator that check that at least one field among the configured ones has been specified.
<input type="radio"/> Permission Validator	Validates that the user has a permission.
<input type="radio"/> User Permission Validator	Validates that the user has a permission, where the OSWorkflow variable holding the username is configurable. Obsolete.
<input type="radio"/> Window Dates	It compares two fields dates, adding a window of days to one of them.

Figura 25: Pantalla para añadir un validador a la condición

Cuando añades un validador a una transición, se muestra la pantalla de edición del validador donde se edita la configuración del validador.

En la figura 21 se muestra la pantalla de edición del validador de incidencia enlazada, los parámetros del validador son el tipo de incidencia enlazada y el tipo de enlace.

Add Parameters To Validator

Add required parameters to the Validator.

Issue linked type:

Link type:

Figura 26: Pantalla de configuración del validador de incidencia enlazada

Una vez introducidos los parámetros en la pantalla de la transición de flujo de trabajo se muestra la vista de los validadores de la transición (figura 22).

[All](#) [Conditions \(1\)](#) [Validators \(1\)](#) [Post Functions \(7\)](#)

Add a new validator to check the input parameters before this transition is executed.

Check that the issue has a **Test Workflow** linked with a **groups** link

[Editar](#) | [Borrar](#)

Figura 27: Vista de los validadores de la transición

De esta forma cuando un usuario realiza una transición se ejecutan los validadores de esta, si no se cumple la condición impuesta por el validador se muestra un mensaje de error.

JIRA administrador Historial Filtros Salir

HOME NAVEGAR POR EL PROYECTO BUSCAR INCIDENCIAS NUEVA INCIDENCIA ADMINISTRACIÓN BÚSQUEDA RÁPIDA:

ERROR

It seems that you have tried to perform an illegal workflow operation.

If you think this message is wrong, please consult your [administrators](#) about getting the necessary permissions.

Form Errors:

- The issue must have at least a Test Workflow issue linked with a Groups link.

This JIRA site is for demonstration purposes only. [Evaluate JIRA](#) - bug tracking software for your team.

Powered by [Atlassian JIRA](#) the Professional Issue Tracker. (Enterprise Edition, Version: 3.13-#330) - [Bug/feature request](#) - [Atlassian news](#) - [Contact Administrators](#)

Figura 28: Mensaje de error que se produce cuando no se cumple la condición impuesta por el validador

(TSB) Francisco Codina Filters Log Out

HOME BROWSE PROJECT FIND ISSUES **CREATE NEW ISSUE** QUICK SEARCH:

Create Issue

Step 2 of 2: Enter the details of the issue...

Project: ORCONERA3

Issue Type:

Field Tab Estimation **Phases weights** Economical Information

Technical Complexity Factor:
Weight of the technical complexity factor from the original estimation

Environment Factor:
Weight of the environment factor from the original estimation

Use Actor Weights:
Weight of the actors factor from the original estimation

User Requirement Points:
Original user requirement points from the original estimation without any adjustment

Effort:
Value used in the estimation to calculate the effort (hours / PUR)

Powered by [Atlassian JIRA](#) the Professional Issue Tracker. (Enterprise Edition, Version: 3.13.4-#354) - [Bug/feature request](#) - [Atlassian news](#) - [Contact Administrators](#)

Figura 29: Detalle del Issue del proyecto

Create Issue

Step 2 of 2: Enter the details of the issue...

Project: ORCONERA3

 Issue Type: Project

 Field Tab **Estimation** Phases weights Economical Information

* Analysis Weight:	<input type="text"/>	Percentage of the analysis phase initially established for the project estimation
* Development Weight:	<input type="text"/>	Percentage of the development phase initially established for the project estimation
* Verification Weight:	<input type="text"/>	Percentage of the verification phase initially established for the project estimation
* Follow-up Weight:	<input type="text"/>	Percentage of the initial estimation for the follow-up tasks. This is a percentage of the general management percentage of the project
* Initial Phase Weight:	<input type="text"/>	Percentage of the initial estimation for the G2 phase tasks
* Deployment Weight:	<input type="text"/>	Percentage of the initial estimation assigned to the deployment phase
* Project Close Weight:	<input type="text"/>	Percentage of the initial estimation given to the tasks to close the project.
* Change Management Weight:	<input type="text"/>	Percentage of initial estimation for change management tasks of the project
* Configuration Management Weight:	<input type="text"/>	Percentage of the Project Management that is devoted to the configuration management
* Quality Assurance Weight:	<input type="text"/>	Percentage of the management of the project at the initial estimation

 Powered by Atlassian JIRA the Professional Issue Tracker. (Enterprise Edition, Version: 3.13.4-#354) - [Bug/feature request](#) - [Atlassian news](#) - [Contact Administrators](#)

Figura 30: Detalle de la incidencia del proyecto con la información de los pesos de las fases

Cada proyecto tiene una única incidencia de tipo project donde se guarda la información necesaria para calcular los indicadores. A continuación se muestra el informe de avance.

En la primera parte del informe de avance se muestra el estado de las tareas (figura 26).

La primera tabla se muestra el estado de las tareas para cada grupo de tareas. El proyecto se divide según la estructura de descomposición de tareas en tareas y grupos de tareas, en forma de árbol desde la raíz del proyecto. A la izquierda se encuentran los issues que representan a los grupos de tareas junto con el identificador del grupo de tareas.

La siguiente tabla muestra las tareas que todavía no se han planificado, el estado de la tarea es "new", para cada usuario.

Finalmente se muestra un gráfico con el estado de las tareas del proyecto

Informe de avance:

28-feb-2009

1. Estado de las tareas

Tabla resumen

Grupo de tareas	Identificador	Inicio	Fin	Total	Planned	In progress	Finished	Reopened	New
Project	TST-356	01-ene-2010	25-jun-2010	9	0	0	3	0	6
FASE I2	TST-324	15-ene-2009	08-abr-2009	6	0	0	1	0	5
Iteración Analisis inicial	TST-325	15-ene-2009	03-feb-2009	3	0	0	1	0	2
Iteración-I1	TST-329	04-feb-2009	24-feb-2009	0	0	0	0	0	0
Iteración I2	TST-330	25-feb-2009	18-mar-2009	3	0	0	0	0	3

Tareas sin planificar (new)

Tarea	Identificador	Asignado	Grupo de Tareas
administrador			
Task 2 - LAI	TST-327	admin	Iteración Analisis inicial
Tarea Analysis	TST-384	admin	Iteración I2
User interface 1	TST-391	admin	Iteración I2
Test Plan	TST-392	admin	Iteración I2
Task prueba	TST-371	admin	Fase G3
Test Me			
Task 3 - LAI	TST-328	test	Iteración Analisis inicial

Gráfico: estado de las tareas

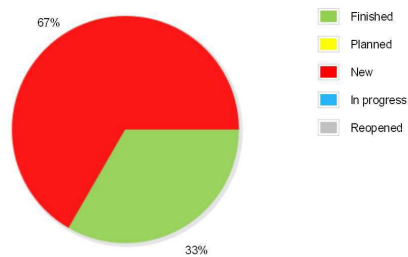


Figura 31: Informe de avance, estado de las tareas

La segunda sección del informe de avance tiene una estructura similar a la primera sección y muestra el estado del comienzo de las tareas (figuras 27, 28). Las tareas que han comenzado según el plan, antes de la fecha planificada, con retraso, después de la fecha planificada, y las tareas retrasadas, las tareas que deberían haber empezado según la planificación pero todavía no han empezado.

2. Estado del comienzo de las tareas

Tabla resumen

Grupo de tareas	Identificador	Inicio	Fin	Según plan	Con retraso	Retrasadas
Project	TST-356	01-ene-2010	25-jun-2010	1	4	2
FASE I2	TST-324	15-ene-2009	08-abr-2009	0	2	2
Iteración Analisis inicial	TST-325	15-ene-2009	03-feb-2009	0	2	2
Iteración-I1	TST-329	04-feb-2009	24-feb-2009	0	0	0
Iteración I2	TST-330	25-feb-2009	18-mar-2009	0	0	0

Tareas con comienzo retrasado

Tarea	Identificador	Asignado	Start Planned Date	End Planned Date	Start Date	Due Date	Maxim Due Date
administrador							
Task 2 - LAI	TST-327	admin	22-ene-2009	28-ene-2009	23-ene-2009	31-ene-2009	
Test Me							
Task 3 - LAI	TST-328	test	29-ene-2009	03-feb-2009		03-feb-2009	03-abr-2009

Figura 32: Informe de avance, estado del comienzo de las tareas

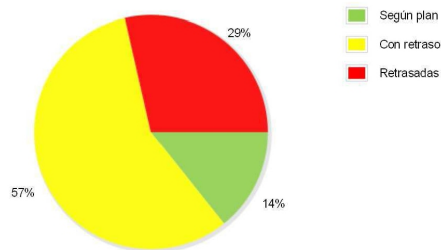


Figura 33: Informe de avance, Gráfico del informe de avance

La tercera sección del informe de avance muestra el estado de finalización de las tareas.

3. Estado de la finalización de las tareas

Tabla resumen

Grupo de tareas	Identificador	Inicio	Fin	Según plan	Con retraso	Retrasadas	Limite < 1 semana
Project	TST-356	01-ene-2010	25-jun-2010	0	3	2	0
↳ FASE I2	TST-324	15-ene-2009	08-abr-2009	0	1	2	0
↳ Iteración Analisis inicial	TST-325	15-ene-2009	03-feb-2009	0	1	2	0
↳ Iteración I1	TST-329	04-feb-2009	24-feb-2009	0	0	0	0
↳ Iteración I2	TST-330	25-feb-2009	18-mar-2009	0	0	0	0

Tareas con finalización retrasada

Tarea	Identificador	Asignado	Start Planned Date	End Planned Date	Start Date	Due Date	Maxim Due Date
administrador							
Task 2 - I.AI	TST-327	admin	22-ene-2009	28-ene-2009	23-ene-2009	31-ene-2009	
Test Me							
Task 3 - I.AI	TST-328	test	29-ene-2009	03-feb-2009		03-feb-2009	03-abr-2009

No existen tareas con fecha límite en una semana o cumplida

Gráfico: finalización de las tareas

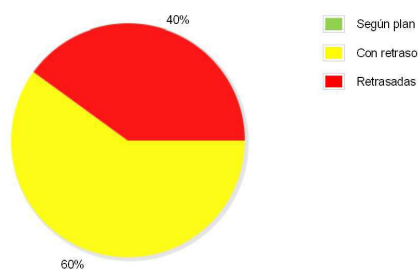


Figura 34: Informe de avance, estado de finalización de las tareas

Por ultimo se muestran los indicadores calculados en el informe de avance.

El esfuerzo en horas estimado del trabajo realizado: Las horas que se deberian de haber invertido en el proyecto según la planificación.

El esfuerzo real del trabajo realizado: las horas reales invertidas en el proyecto.

El esfuerzo estimado del trabajo realizado, las horas estimadas del trabajo terminado en el proyectos, también llamado valor conseguido.

También se muestran diversos índices. Junto con la evaluación de los indicadores, rojo, verde y amarillo.

4. Plan de medición

Esfuerzo estimado del trabajo planificado (CPTP):	345 horas
Esfuerzo real del trabajo realizado (CRTR):	164 horas
Esfuerzo estimado del trabajo realizado (CPTR):	239 horas
Variación de coste (CRTR - CPTR):	-76 horas
Índice de variación de coste (CPTP/CRTR):	(-46 %) 1,46 N / A
Variación de la planificación (CPTP - CPTR):	106 horas
Índice de variación de la planificación (CPTR/CPTP):	(31 %) 0,69 N / A

Figura 35: Informe de avance, plan de medición

Cada proyecto cuenta también con una incidencia de informe de estado. De manera similar a la incidencia del proyecto contiene información utilizada para la generación del informe.

The screenshot shows the JIRA interface for an issue titled 'Status report' (ID: TSI-58). The issue is of type 'Status Report', is currently 'Open', and has a 'Major' priority. It is assigned to 'administrador'. The 'Test Indicators' section includes a 'Status report' sub-section with the following details:

- Componente/s:** Ninguno
- Version(es) Afectadas:** Ninguno
- Version(es) Fijadas:** Ninguno
- Time Tracking:** Desconocido

The 'Project progress status' section provides a detailed explanation of the metrics:

- CPTP:** Es el coste programado del trabajo planificado, es el trabajo en horas que se ha planificado para este punto del proyecto.
- CRTR:** Es el coste real del trabajo realizado, es el coste que se ha realizado, las horas reales invertidas en el proyecto hasta ahora.
- CPTR:** Es el coste planificado del trabajo realizado, del trabajo realizado cuanto se había planificado que costaría.
- ICE e IEP:** son los índices de variación de coste e índice de variación de la planificación.

The 'Development status project' section explains that the progress is shown as a percentage of the project, calculated based on the number of use cases completed relative to the total number of use cases. It also mentions that the progress is measured by module in the first graph and by phase (analysis, development, verification, and change management) in the subsequent graphs.

The 'Quality report' section states that it shows a summary of audits, non-conformities, and corrective actions during the audit. The 'Risk report' section shows project risks along with corrective actions and risk evaluations. The 'Involvement plan' section lists related tasks from external sources. The 'Metrics' section shows measures for the project's indicators.

The sidebar on the left provides workflow actions such as 'Start Progress', 'Resolve Issue', 'Close Issue', and 'Operaciones' (Assign, Clone, Comment, Create sub-task, etc.).

Figura 36: Detalle de la incidencia del informe de estado

Finalmente se muestra el informe de estado.

En la primera sección del informe se muestra información general del proyecto.

Status report

1. Report identify

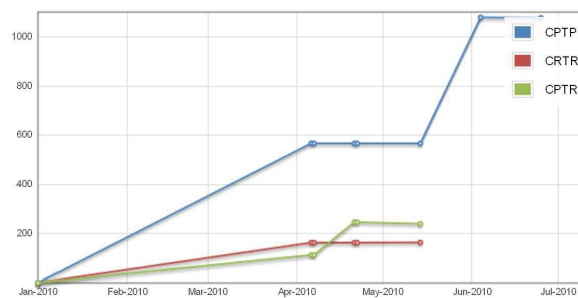
Project name	Test
Start Date	01-ene-2010
Client	
Report Date	28-feb-2009
Milestone	Iteration / Release

Figura 37: Informe de estado, identificación del proyecto

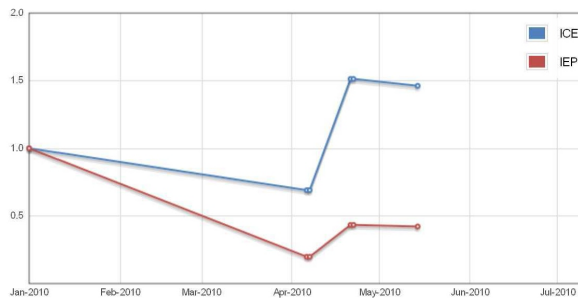
La segunda sección del informe de estado muestra una serie de gráficos con la evolución los indicadores de avance del proyecto durante la duración del mismo.

2. Project progress

Summary



Indexes



Date	CPTP	CRTR	CPTR	ICE	IEP
6-Apr-2010	567	162,5	112,21	0,69	0,2
7-Apr-2010	567	162,5	112,21	0,69	0,2
21-Apr-2010	567	162,5	245,94	1,51	0,43
22-Apr-2010	567	162,5	245,94	1,51	0,43
14-May-2010	567	163,5	239,11	1,46	0,42

Review of project progress

En esta sección se muestra el progreso del proyecto a lo largo del tiempo. Los gráficos muestran:

CPTP Es el coste programado del trabajo planificado, es el trabajo en horas que se ha planificado para este punto del proyecto.

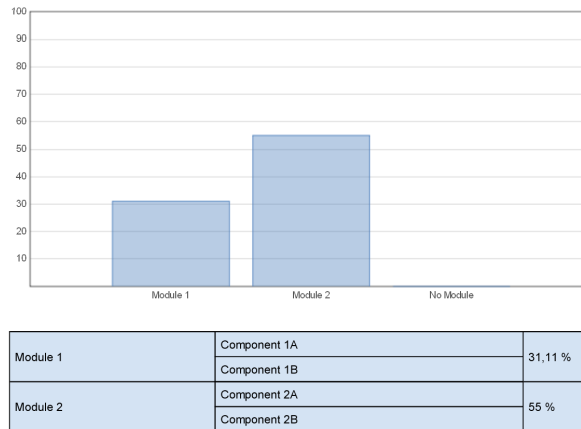
CRTR Es el coste real del trabajo realizado, es el coste que se ha realizado, las horas reales invertidas en el proyecto hasta ahora.

CPTR Es el coste planificado del trabajo realizado, del trabajo realizado cuanto se había planificado que costaría.

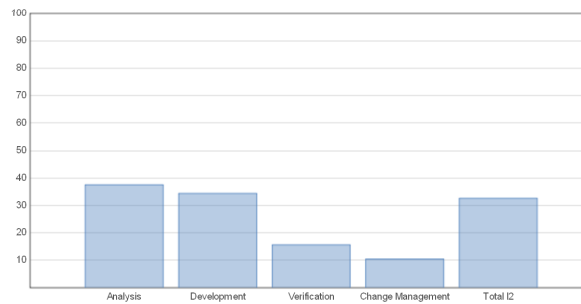
ICE e IEP son los índices de variación de coste e índice de variación de la planificación.

Figura 38: Informe de estado, avance del proyecto

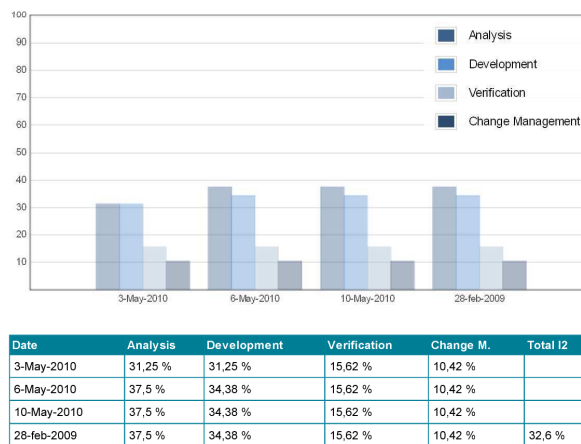
De forma similar en la tercera sección del informe de estado se muestran gráficos del estado de desarrollo del proyecto.



Progress by phase



Progress summary



Review of development status

Esta sección muestra el progreso en tanto por ciento del proyecto. El progreso se calcula a partir de los puntos de caso de uso. Los puntos de caso de uso son una medida de la complejidad de los casos de uso. Se mide el porcentaje de casos de uso realizados con respecto al total de los puntos de caso de uso del proyecto.

En la primera gráfica se miden el progreso por módulo.

En las siguientes se mide el progreso por fase análisis, desarrollo, verificación y gestión del cambio.

Figura 39: Informe de estado, estado de desarrollo

La cuarta sección del informe de estado muestra el estado de las auditorías del proyecto, un resumen de las no conformidades surgidas y las acciones correctivas para cada no conformidad.

4. Quality assurance project

Summary of audits

Issue	Planned	Open	Ready	Completed	Closed
Audits	1	2	0	1	1

Tracking of nonconformities

Summary	Identifier	Process - subprocess	Status	Severity	Reason
Auditoria 4	TST-313				
Nonconformity - Completar el acta viabilidad	TST-283	Analisis inicial - I1.1 Contacto Inicial y preparacion del proyecto	Closed	Minor	Not realized
	Identifier	Corrective Actions	Status	Resolution	Assignee
	TST-285	corrective action 1	Resolved	Fixed	administrador

Summary	Identifier	Process - subprocess	Status	Severity	Reason
No parent non conformity	TST-450		Open		
Auditoria C1	TST-274				
Nonconformity 2	TST-288	Analisis - I2.1 Analisis inicial de requisitos de sistema	Open	Scale	Reason 3
Auditoria A1	TST-286				
NC 3	TST-308	Estimacion, viabilidad y oferta - G1.1 Estimacion y viabilidad inicial	Open	Major	
Auditoria 4	TST-313				
NC 1	TST-306	Calidad - E1.3 Informe semestral de SQA	Open	Minor	
Nonconformity - Existen tareas externas que no tienen matriz RACI actualizada	TST-311	Estimacion, viabilidad y oferta - G1.1 Estimacion y viabilidad inicial	Open	Major	

Review of the status of the quality assurance of the project

En esta sección se muestra un resumen de la auditorias, las noconformidades surgidas durante la auditoria y las acciones correctivas.

Figura 40: Informe de estado, Aseguramiento de la calidad del proyecto

figura 36 muestra los riesgos identificados en el proyecto junto con las acciones correctivas tomadas para cada riesgo.

5. Identified risks

Summary of risk

Issue	New	In Evaluation	Evaluated	Open	In progress	Resolved	Verified	Closed	Approved	Released
Risks	2	0	1	0	0	0	0	1	0	0

Status of risk assessments

Summary	Identifier	Category	Strategy	Evaluation			Status
				Impact	Prob.	assess.	
Disponibilidad del jefe de proyecto	TST-449	Category 2, Category 2 1	Strategy 2	0.2	0.2		Closed

Summary	Identifier	Category	Strategy	Evaluation			Status
				Impact	Prob.	assess.	
Comunicacion stakeholders externos	TST-300	Category 1, Category 1 1	Strategy 1	0.2	0.4		Evaluated
	Identifier	Corrective Actions	Status	Resolution	Assignee		
	TST-302	Corrective action 4	Open	None	administrador		
Risk 2	TST-301		Strategy 3	0.4	0.7		Open
Risk 3	TST-316	Category 1, Category 1 2	Strategy 1	0.2	0.1		Open

Review of the status of the project's risks

En esta sección se muestran los riesgos del proyecto junto con sus acciones correctivas y la evaluacion de los riesgos.

Figura 41: Informe de estado, Informe de riesgos

La figura 37 muestra el plan de seguimiento. Se muestra la matriz de asignación de responsabilidad.

6. Follow up of the involvement plan

RACI Matrix

Task Summary	Identifier	External Assignee	Assignee	Informed	Consulted	Status	Due Date	Comment
Task R3 2	TST-293	External assignee	admin	Informed	Consulted	Resolved	06-nov-2009	Comment
Raci task 2	TST-396	Hey Hey!	admin		Consulted	Resolved	14-ene-2010	

Review of the involvement plan

Plan de involucración, en esta sección aparecen aquellas tareas relacionadas con fuentes externas.

Figura 42: Informe de estado, plan de seguimiento

Por ultimo en el plan de medición se muestran los indicadores calculados durante el informe de estado junto con la evaluación de estos.

7. Measurement plan

	Analisis inicial	Estimacion, viabilidad y	Planificacion	Analisis	Construccion	Verificacion	Seguimiento	Implantacion	Cierre	Calidad	Gestion del cambio	Gestion de la configuracion
NC.												
Cost deviation				-94,3 %	-97,87 %						-47,28 %	
Task deviation					0 %						-50 %	
Phase weight	19,61 %	0 %	35,29 %	41,18 %	23,53 %	0 %	0 %	588,24 %	58,82 %	0 %	35,29 %	23,53 %

ID	Name	Value	Units	Evaluation
General project indicators				
DWBS	Desviación de las tareas planificadas	-7,54	%	
FRCI	Fallos en la recogida o cálculo de indicadores	0	NC	
PPCM	Porcentaje de cumplimiento del proceso de CM	0	NC	
SGCA	Adherencia a los procesos del SGC	1	NC	
TVDA	Tiempo de vida medio de las desviaciones abiertas por indicador	0	días / desviación	
General milestone indicators				
COPP	Cobertura del plan de pruebas	50	%	
ECEE	Eficacia de corrección de errores	50	%	
EEDF	Eficacia de eliminación de defectos	50	%	
EFRI	Efectividad de eliminación de riesgos	0,5	AC / riesgos	
EFTI	Efectividad test interno	0	defectos / PCU	
PPCU	Pantallas por punto de caso de uso analizados	0,6	pantallas / PCU	
RIES	Número de riesgos del proyecto	0,45	riesgos / PUR	
TMCA	Tiempo medio de cambio	1,5	horas / cambio	
TPCU	Casos de test por punto de caso de uso analizados	0,6	casos test / PCU	
VELD	Velocidad de desarrollo	0,72	horas / PCUs	
VELA	Velocidad de desarrollo de las tareas de análisis	0,85	horas / PCUs	
VELC	Velocidad de desarrollo de las tareas de construcción	0,32	horas / PCUs	
VELV	Velocidad de desarrollo de las tareas de verificación	0	horas / PCUs	
VELGC	Velocidad de desarrollo de las tareas de gestión del cambio	7,91	horas / PCUs	
Specific iteration indicators				
DESIA	Desviación de las tareas de analisis de la iteración	20	% (1 / 2)	
DESIC	Desviación de las tareas de construcción de la iteración	-	% (0 / 1)	
DESIV	Desviación de las tareas de verificación de la iteración	-	% (0 / 0)	
DESIGC	Desviación de las tareas de gestion del cambio de la iteración	-	% (0 / 0)	
VELID	Velocidad de desarrollo de las tareas de la iteración	0,21	horas / PCUs	
VELIA	Velocidad de desarrollo de las tareas de análisis	0	horas / PCUs	
VELIC	Velocidad de desarrollo de las tareas de construcción	0	horas / PCUs	
VELIV	Velocidad de desarrollo de las tareas de verificación	0	horas / PCUs	
VELIGC	Velocidad de desarrollo de las tareas de gestión del cambio de la iteración	0	horas / PCUs	

Review of the measurement plan

En esta sección aparecen las medidas del proyecto para los indicadores.

Figura 43: Plan de medición

7. CONCLUSIONES

Durante la realización del proyecto he obtenido una gran experiencia laboral, como acercamiento al desarrollo de aplicaciones profesionales. He aprendido una gran cantidad de nuevas tecnologías, entornos de desarrollo, herramientas de construcción de aplicaciones. He aprendido a manejarme y saber adaptarme ante nuevas tecnologías.

La experiencia también me ha servido como un acercamiento a la gestión de proyectos, desde la propia terminología utilizada hasta las metodologías y estándares propios de la gestión de proyectos.

JIRA se desvela como una herramienta muy potente para la gestión de proyectos, especialmente para la gestión de proyectos software por su gran capacidad de integración con otros componentes y la facilidad de personalización y de desarrollo que ofrece.

Ha sido muy importante el factor humano durante la realización del proyecto, trabajar con un equipo de personas.

Por último no quisiera dejar la oportunidad de agradecer al equipo humano de TSB por la experiencia vivida junto a ellos.

Ampliaciones futuras.

El desarrollo de la aplicación, continua. Entre los planes de futuro se encuentran, la mejora de la localización, la posibilidad de generar los informes en distintos idiomas, creación de nuevos informes, la mejora de las búsquedas en JIRA, mediante algún módulo buscador. Se están estudiando nuevas mejoras en el sistema.

También se está estudiando la creación de una aplicación para la gestión del maestro de indicadores que realice el cálculo de los indicadores organizativos de la empresa etc...

8. REFERENCIAS

- <http://java.sun.com/javaee/reference/>
- <http://www.atlassian.com/software/jira/>
- <http://confluence.atlassian.com/display/JIRA03x/JIRA+Development+Hub>
- <http://jira.atlassian.com/secure/Dashboard.jspa>
- <http://maven.apache.org/>
- <http://velocity.apache.org/>
- <http://www.picocontainer.org/>
- <http://www.w3c.es/>
- <http://www.sei.cmu.edu/cmmi/>
- <http://lucene.apache.org/java/docs/>
- <https://issues.apache.org/jira/secure/Dashboard.jspa>
- <http://www.tsbtecnologias.es/inicio/index.php>
- <http://msdn.microsoft.com/es-es/default.aspx>
- <http://en.wikipedia.org/>
- <http://www.bibliojuridica.org/libros/2/909/5.pdf>
- <http://www.monografias.com/trabajos55/organizaciones-con-control-calidad/organizaciones-con-control-calidad.shtml>

9. ANEXOS

8.1 INDICADORES

Un proceso efectivo de medición y análisis (MA) proporciona una base adecuada de entendimiento de las capacidades de desarrollo, lo que permite definir planes viables para el desarrollo de productos y la prestación de servicios de calidad. Las medidas permiten detectar tendencias y anticipar problemas y, por lo tanto, permite establecer un mejor control de los costes, una reducción de los riesgos, mejorar la calidad y asegurar la consecución de los objetivos de negocio.

La política de Medición y Análisis especifica las particularidades del proceso S2 de medición y análisis de TSB Tecnologías, determinando como se deben especificar los indicadores, como se deben realizar las medidas y cuales son los criterios para analizar y comunicar los indicadores. Toda esta información será recogida en el Maestro de Indicadores, que se mantendrá actualizado en cada ciclo de Medición y Análisis de la empresa.

El responsable de la recogida de los indicadores de proyecto será el Jefe de Proyecto dentro de sus tareas de seguimiento del mismo. Para ello, cada vez que un indicador deba ser calculado según su frecuencia, el Jefe de Proyecto rellena una columna nueva en el Cuadro de Mando de Indicadores del Proyecto. Estos valores deben ser incorporados a los informes de seguimiento del proyecto en cuestión.

Maestro de indicadores:

ID	Nombre	Unidades
COPP	Cobertura del plan de pruebas	%
COST1	Desviación del coste de las tareas de análisis inicial	%
COST2	Desviación del coste de las tareas de estimación, viabilidad y oferta comercial	%
COST3	Desviación del coste de las tareas de planificación	%
COST4	Desviación del coste de las tareas de análisis	%
COST5	Desviación del coste de las tareas de construcción	%
COST6	Desviación del coste de las tareas de verificación	%
COST7	Desviación del coste de las tareas de seguimiento	%
COST8	Desviación del coste de las tareas de implantación	%
COST9	Desviación del coste de las tareas de cierre	%
COST10	Desviación del coste de las tareas de calidad	%
COST11	Desviación del coste de las tareas de gestión del cambio	%
COST12	Desviación del coste de las tareas de gestión de la configuración	%
CPTP	Esfuerzo estimado del trabajo planificado	horas
CPTR	Esfuerzo estimado del trabajo realizado	horas
CRTR	Esfuerzo real del trabajo realizado	horas
DESIA	Desviación de las tareas de análisis de la iteración	%
DESIC	Desviación de las tareas de construcción de la iteración	%
DESIV	Desviación de las tareas de verificación de la iteración	%
DESV1	Desviación de las tareas de análisis inicial	%
DESV2	Desviación de las tareas de estimación, viabilidad y oferta comercial	%
DESV3	Desviación de las tareas de planificación	%
DESV4	Desviación de las tareas de análisis	%
DESV5	Desviación de las tareas de construcción	%
DESV6	Desviación de las tareas de verificación	%
DESV7	Desviación de las tareas de seguimiento	%
DESV8	Desviación de las tareas de implantación	%
DESV9	Desviación de las tareas de cierre	%
DESV10	Desviación de las tareas de calidad	%
DESV11	Desviación de las tareas de gestión del cambio	%
DESV12	Desviación de las tareas de gestión de la configuración	%
DWBS	Desviación de las tareas	%
ECEE	Eficacia de corrección de errores	%
EEDF	Eficacia de eliminación de defectos	%
EFG1	Efectividad de las fases de concepción del proyecto	N / A
EFRI	Efectividad de eliminación de riesgos	AC / riesgos
EFTI	Efectividad test interno	defectos / PCUs
ENG1	Eficiencia de las fases de concepción del proyecto	%
EOFR	Coste de la oferta por proyecto	€ / h
FRCI	Fallos en la recogida o cálculo de indicadores	NC
ICE	Índice de variación del coste	N / A
IEP	Índice de variación de la planificación	N / A
OHG1	Overhead de las fases de concepción del proyecto	%

PPCM	Porcentaje de cumplimiento del proceso de CM	NC
PCU	Pantallas por punto de caso de uso analizado	pantallas / PCUs
PRFA	Progreso de la fase de análisis	%
PRFC	Progreso de la fase de construcción	%
PRFGC	Progreso de la fase de gestión del cambio	%
PRFV	Progreso de la fase de verificación	%
RIES	Número de riesgos del proyecto	riesgos / PUR
RIPO	Riesgos identificados tras la fase de planificación	riesgos / PUR
SGCA	Adherencia a los procesos del SGC	NC
TMCA	Tiempo medio de cambio	horas / cambio
TPCU	Casos de test por punto de caso de uso analizados	casos test / PCUs
TVDA	Tiempo de vida medio de las desviaciones abiertas por indicador	días / desviación
VELA	Velocidad de desarrollo de las tareas de análisis	horas / PCUs
VELC	Velocidad de desarrollo de las tareas de construcción	horas / PCUs
VELD	Velocidad de desarrollo	horas / PCUs
VELGC	Velocidad de desarrollo de las tareas de gestión del cambio	horas / PCUs
VELV	Velocidad de desarrollo de las tareas de verificación	horas / PCUs
VELIA	Velocidad de desarrollo de las tareas de análisis de la iteración	horas / PCUs
VELIC	Velocidad de desarrollo de las tareas de construcción de la iteración	horas / PCUs
VELIGC	Velocidad de desarrollo de las tareas de gestión del cambio de la iteración	horas / PCUs
VELIV	Velocidad de desarrollo de las tareas de verificación de la iteración	horas / PCUs
WEIG1	Peso de las tareas de análisis inicial	%
WEIG2	Peso de las tareas de estimación viabilidad y oferta	%
WEIG3	Peso de las tareas de planificación	%
WEIG4	Peso de las tareas de análisis	%
WEIG5	Peso de las tareas de construcción	%
WEIG6	Peso de las tareas de verificación	%
WEIG7	Peso de las tareas de seguimiento	%
WEIG8	Peso de las tareas de implantación	%
WEIG9	Peso de las tareas de cierre	%
WEIG10	Peso de las tareas de calidad	%
WEIG11	Peso de las tareas de gestión del cambio	%
WEIG12	Peso de las tareas de gestión de la configuración	%

8.2 CMMI METAS Y PRACTICAS

En este apéndice se muestra un ejemplo de práctica y meta genérica para la mejora de procesos propuesta por CMMI.

Un ejemplo de meta genérica sería:

GG 2 INSTITUCIONALIZAR UN PROCESO GESTIONADO

El proceso está institucionalizado como un proceso gestionado.

Un ejemplo de práctica genérica para esta meta sería:

GP 2.8 MONITORIZAR Y CONTROLAR EL PROCESO

Monitorizar y controlar el proceso frente al plan para realizar el proceso y tomar las acciones correctivas apropiadas.

El propósito de esta práctica genérica es realizar la monitorización y el control directo del proceso día a día. Se mantiene una visibilidad apropiada del proceso, por lo que se pueden tomar acciones correctivas apropiadas cuando sea necesario. Monitorizar y controlar el proceso involucra medir los atributos apropiados del proceso o de los productos de trabajo producidos por el proceso.

Para más información sobre la monitorización y el control del proyecto y la toma de acciones correctivas, consúltese el área de proceso de Monitorización y control de proyecto.

Para más información sobre la medición consúltese el área de proceso de Medición y análisis.

Subprácticas

1. Medir el rendimiento real frente al plan de realización del proceso. Las medidas son del proceso, de sus productos de trabajo y de sus servicios.
2. Revisar los logros y los resultados del proceso frente al plan de realización del proceso.
3. Revisar las actividades, el estado y los resultados del proceso con el nivel de gerencia inmediato responsable del proceso e identificar los problemas. Las revisiones pretenden proporcionar al nivel de gerencia inmediato la visibilidad apropiada del proceso. Las revisiones pueden ser periódicas o por eventos.
4. Identificar y evaluar los efectos de las desviaciones significativas del plan de realización del proceso.
5. Identificar los problemas en el plan de realización del proceso y en la ejecución del mismo.
6. Tomar acciones correctivas cuando los requerimientos y los objetivos no se satisfacen, cuando se identifican problemas o cuando el progreso difiere significativamente del plan de realización del

proceso.

Hay riesgos inherentes que deberían considerarse antes de que se tome cualquier acción correctiva.

Una acción correctiva puede incluir:

- Tomar acciones correctoras para reparar los productos de trabajo o los servicios defectuosos.
- Cambiar el plan de realización del proceso.
- Ajustar los recursos incluyendo personas, herramientas y otros recursos.
- Negociar los cambios de los compromisos establecidos.
- Asegurar el cambio a los requerimientos y objetivos que tienen que satisfacerse.
- Finalizar el esfuerzo.

7. Seguir las acciones correctivas hasta el cierre.

8.3 LOS PROCESOS EN LA EMPRESA

El Sistema de Gestión de TSB divide sus procesos en tres tipos fundamentales:

Procesos estratégicos: son aquellos que proporcionan directrices a todos los demás procesos y son realizados por la dirección o por otras entidades. Se suelen referir a las leyes, normativas,... aplicables a la empresa y que no son controladas por los servicios o proyectos.

Son procesos estratégicos de TSB:

- E1 - Auditorias internas
- E2 - Control de no conformidades
- E3 - Acciones correctivas y preventivas
- E4 - Seguimiento y medición de procesos
- E5 - Revisión del sistema

Procesos clave: atañen a diferentes áreas de desarrollo/servicio/producción y tienen impacto en el cliente creando valor para éste. Son las actividades esenciales de la empresa, su razón de ser.

Son procesos clave de la empresa:

Desarrollo y servicios:

- I1 - Análisis inicial de requisitos
- G1 - Estimación, viabilidad inicial y elaboración de oferta comercial
- G2 - Fase inicial de proyecto
- I2 - Análisis, diseño, desarrollo y validación
- G3 - Entrega y cierre

Producción:

- I3 - Producción e integración
- G4 - Gestión de proveedores

- G5 - Recepción y almacén

I+D+i:

- G6 - Gestión de ideas
- G7 - Planificación, seguimiento y control de la cartera de proyectos
- I4 - Desarrollo de los proyectos de I+D+i

Procesos de soporte: dan apoyo a los procesos fundamentales que realiza un servicio desarrollo o producto. TSB tiene desplegados los siguientes procesos de soporte:

- S1 - Gestión de la configuración
- S2 - Control de los documentos
- S3 - Control de los registros
- S4 - Seguimiento del proyecto
- S5 - Gestión del cambio

8.4 ACERCA DE JIRA

JIRA es un producto software desarrollado por la compañía Atlassian con capacidad para la gestión de proyectos, seguimiento de errores e incidencias (bug tracker, Issue tracker), gestión de flujos de trabajo (workflows), gestión de metodologías ágiles de desarrollo.

The screenshot shows the Apache Software Foundation JIRA issue tracker homepage. At the top, there is a navigation bar with links for HOME, BROWSE PROJECTS, and FIND ISSUES. A search bar is located on the right. The main content area is divided into two columns. The left column contains a welcome message, instructions on how to use JIRA, and a list of latest releases. The right column contains a login form with fields for Username and Password, a checkbox for 'Remember my login on this computer', and a 'Log In' button. Below the login form, there is a link for 'Forgot Password' and a 'Not a member? Signup' link. At the bottom, there is a footer with information about the Atlassian JIRA license and version.

Welcome to Apache's JIRA issue tracker!

Anyone is free to [find](#) issues. You must [register](#) and [login](#) if you want to [create](#), [comment](#), [vote](#), or [watch](#) issues. Only developers can edit, prioritize, schedule and resolve issues.

Some projects were migrated here from [Bugzilla](#). If you had a Bugzilla account, log in using your email address as username. You will need to have a new password mailed to you. You can search for issues by their old Bugzilla ID in the portlet below.

Got a question about JIRA? See the [ASF JIRA FAQ](#) on the Wiki.

If your ASF project wants to use JIRA, either read this wiki page about [migrating from Bugzilla](#) or please contact jira at apache.org to setup a new project.

ASF JIRA

Log In

HOME BROWSE PROJECTS FIND ISSUES QUICK SEARCH:

ASF JIRA

Welcome to Apache's JIRA issue tracker!

Anyone is free to [find](#) issues. You must [register](#) and [login](#) if you want to [create](#), [comment](#), [vote](#), or [watch](#) issues. Only developers can edit, prioritize, schedule and resolve issues.

Some projects were migrated here from [Bugzilla](#). If you had a Bugzilla account, log in using your email address as username. You will need to have a new password mailed to you. You can search for issues by their old Bugzilla ID in the portlet below.

Got a question about JIRA? See the [ASF JIRA FAQ](#) on the Wiki.

If your ASF project wants to use JIRA, either read this wiki page about [migrating from Bugzilla](#) or please contact jira at apache.org to setup a new project.

Login

Username

Password

Remember my login on this computer

[Forgot Password](#)

Not a member? [Signup](#) for an account.

All Projects

Select project..

Latest Releases

Wicket	1.4.9	22/May/10
Felix	karaf 1.6.0	21/May/10
HttpComponents HttpClient	4.1 Alpha2	19/May/10
Derby	10.6.1.0	18/May/10
jUDDI	3.0.2	17/May/10
Zookeeper	3.3.1	17/May/10

Search by Bugzilla ID

[Bugzilla ID](#)

Powered by a free Atlassian [JIRA](#) open source license for Apache Software Foundation. Try JIRA - [bug tracking software](#) for your team.

Atlassian JIRA the Professional Issue Tracker. (Enterprise Edition, Version: 3.13.5-#360) - [Bug/feature request](#) - [Atlassian news](#) - [Contact Administrators](#)

Figura 44: JIRA de la fundación Apache

Inicialmente Jira se utilizó para el desarrollo de software, sirviendo de apoyo para la gestión de requisitos, seguimiento del estatus y más tarde para el seguimiento de errores. Jira puede ser utilizado para la gestión de procesos y para la mejora de procesos gracias a sus funciones para la organización de flujos de trabajo (Workflows).

Issues en JIRA.

Los issues son la base de JIRA, un issue puede utilizarse para modelar diferentes tipos de incidencias. Dependiendo de cómo una organización este usando JIRA, un issue puede representar un bug de software, una tarea de un proyecto, un grupo de tareas, una petición...

Los issues tienen una serie de campos y propiedades:

Key es un identificador único para cada incidencia, el identificador de un issue está formado por el identificador del proyecto al que pertenece seguido de un número de identificación.

Issue type es el tipo de issue, la incidencia que representa, como por ejemplo una tarea perteneciente a un proyecto o un bug software.

Status, cada issue tiene un estado dependiendo de donde se encuentra en su ciclo de vida, los issues pueden empezar con el estado de open y cambiar de estado mediante transiciones del workflow.

Priority, La prioridad es la importancia relativa de un issue con respecto a los otros.

Resolution, determina como se ha resuelto un issue, un bug software corregido puede resolverse como "fixed", o una incidencia podría resolverse "duplicada" cuando dos personas informan del mismo bug.

Assignee, la persona a quien está actualmente asignada la incidencia, la persona asignada a una tarea será la persona responsable de llevarla a cabo.

Reporter, la persona que crea la incidencia y se la asigna al assignee.

Project, el proyecto al cual pertenece la incidencia.

Summary, el nombre de la incidencia, una breve descripción de la incidencia.

Description, una descripción detallada de la incidencia.

Comments, cada incidencia puede contener comentarios, que pueden ser añadidos por las personas

envueltas en la incidencia.

Original Estimated. Es una estimación inicial del tiempo que será necesario para resolver la incidencia.

Time Spent, Es el tiempo actual de trabajo realizado en la incidencia, este tiempo aumenta a medida que se incluye tiempo de trabajo sobre la incidencia.

Remaining Estimate, el tiempo de trabajo estimado restante para resolver una incidencia.

Proyectos en JIRA.

Un proyecto es una colección de issues, Los proyectos son definidos según las necesidades de la organización, un proyecto de desarrollo de software, una campaña de marketing, un proyecto administrativo...

Los proyectos tienen un contexto propio pudiendo definir tipos de issues, campos, estados, resoluciones, workflows, permisos, propios.

Los componentes son agrupaciones lógicas de incidencias en secciones más pequeñas dentro de un proyecto. Cada proyecto puede consistir en una serie de componentes o puede no contener ninguno. Una incidencia puede no pertenecer a ningún componente o puede pertenecer a más de un componente.

The screenshot shows a JIRA issue page for 'HTTPS cache header settings'. The issue is categorized as 'Improvement' and is currently 'Open'. It is a 'Blocker' priority issue, currently 'Unassigned', and was reported by 'Brian Lane'. The issue has 2 votes and 6 watchers. The component is 'Performance', it affects no versions, and the fix version is '4.1'. The issue was created on 23/Nov/08 at 08:10 PM and last updated on 07/Nov/09 at 07:58 AM. The time tracking is 'Not Specified'. The issue is linked to a reference issue, 'JIRA-16021', with the note 'Form data lost when using back and forward web browser button:'. The participants are 'Anton Mazkovi [Atlassian]' and 'Brian Lane'. The issue was last commented on 44 weeks and 6 days ago. The description states: 'Conf 3 is doing some html changes to improve performance when being used with HTTPS. We should review these changes and look to add them to JIRA 4 as well.' The comments section shows two entries: one by 'Brian Lane' on 27/Nov/08 at 09:24 PM, which includes a link to a blog post about changes, and another by 'Anton Mazkovi [Atlassian]' on 26/Mar/09 at 10:38 PM, which discusses the problem of JavaScript modifying the DOM and how it affects browser behavior when navigating back and forth.

Figura 45: Incidencia en el JIRA de la compañía Atlassian

Las versiones son puntos en el tiempo de un proyecto. Los issues pueden estar asociados a diferentes versiones de un proyecto, esto puede ser útil en proyectos que evolucionan a través de versiones, como por ejemplo los proyectos software. Para estos casos los issues contienen dos campos que los relacionan con la versión de un proyecto.

Affects version(s), la versión, versiones de un proyecto en las que se manifiesta un issue (por ejemplo un bug software).

Fix version(s), la versión en la que un issue se resuelve.

Las versiones pueden estar en el estado Released, Unreleased o Achieved.

The screenshot shows the JIRA interface for a bug report. At the top, the TSB logo is on the left, and user information (Francisco Codina) and navigation links (History, Filters, Log Out) are on the right. Below the navigation bar, there are tabs for HOME, BROWSE PROJECT, FIND ISSUES, and CREATE NEW ISSUE, along with a QUICK SEARCH field.

The main content area is divided into several sections:

- Issue Details:** Shows the key (OTLM-753), type (CR - Bug), status (Open), priority (Medium/Desired), assignee (Francisco Codina), reporter (Susana Pomés Burguera), votes (0), and watchers (0).
- Available Workflow Actions:** A list of actions such as Start, Won't fix, Assign, Attach file, Attach screenshot, Clone, Comment, Create sub-task, Edit, Link, Convert, Voting, and Watching.
- Time Tracking:** A table showing Original Estimate (2 hours), Remaining Estimate (2 hours), and Time Spent (Not Specified).
- Issue Links:** A section for linking related issues, including 'Affect', 'Executed', and 'Raise'.
- Details:** A table with fields like Change Request Type (Internal), Modulos y Componentes (PC - Transfusiones), Test plan reference (OTLM-542 - TP - Trasfusiones), and Test plan execution (2).
- Description:** A text area containing the bug description in Spanish, detailing issues with report formatting and alignment.
- Comments:** A section for adding comments, currently showing 'There are no comments yet on this issue.'

At the bottom, there is a footer with the text: 'Powered by Atlassian JIRA the Professional Issue Tracker. (Enterprise Edition, Version: 3.13.4-#354) - Bug/feature request - Atlassian news - Contact Administrators'.

Figura 46: Detalle de un Bug en el JIRA de la compañía TSB

Usuarios en JIRA.

Representan a las personas que usarán la aplicación, los usuarios son quienes han de asignar incidencias y trabajar en ellas. Los administradores se encargan del adecuado funcionamiento de la aplicación.

Permisos en JIRA.

JIRA posee un sistema de permisos que permite configurar quien accede a los recursos de JIRA y que acciones pueden llevar a cabo.

Los permisos son gestionados por los esquemas de permisos, los esquemas de permisos permiten definir permisos para usuarios, grupos y roles en cierto contexto, permisos globales, permisos para un proyecto o niveles de seguridad de los issues.

Workflows en JIRA.

El workflow (flujo de trabajo) Es un conjunto de pasos y transiciones que atraviesa una incidencia durante su ciclo de vida. Los workflows normalmente representan “procesos de negocio”.

Un estado representa una etapa en el flujo de trabajo para una incidencia. Una incidencia solo puede estar en un único paso en un instante de tiempo. Cada paso tiene asociado un estado.

El workflow por defecto de JIRA tiene la siguiente forma, los recuadros representan los pasos del workflow mientras que las líneas representan las transiciones.

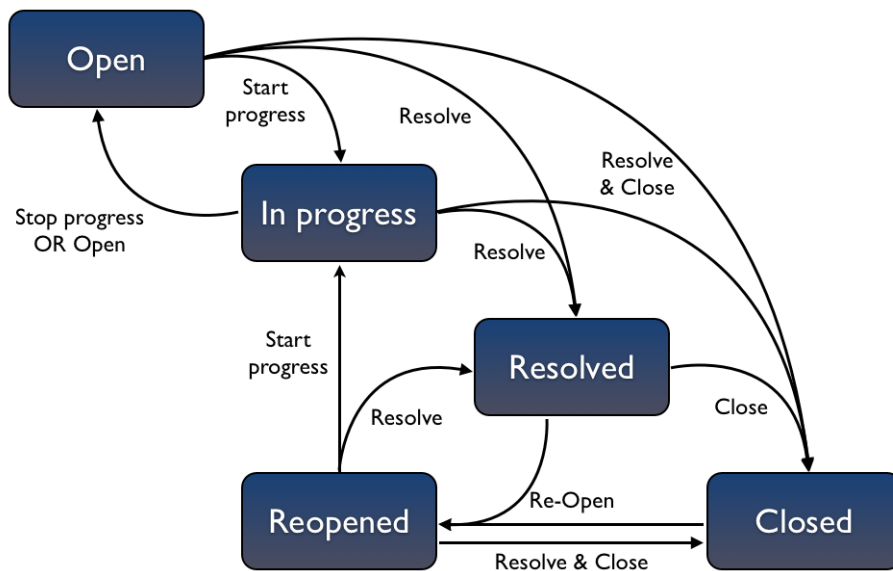


Figura 47: Flujo de trabajo de las incidencias en JIRA

Las transiciones llevan a una incidencia de un estado al siguiente, las incidencias cambian de estado mediante transiciones.

Las transiciones pueden tener una serie de propiedades:

Conditions, Determinan si un issue puede empezar la transición desde un paso al siguiente, se debe de cumplir con la condición definida para poder realizar la transición.

Validator, Los validadores son parecidos a los Conditions, se diferencian en que en los validadores se permite comenzar la transición. Se suelen usar para una vez cumplidas las condiciones, comprobar que cualquier campo requerido es introducido durante la transición.

Post Functions, son eventos automáticamente lanzados inmediatamente después de la transición. Cada proyecto tiene asociado un workflow scheme, los workflow schemes a su vez asocian tipos de issues y workflows. Un proyecto por lo tanto podrá tener diferentes workflows para diferentes tipos de issues. O dicho de otra forma las incidencias de diferentes tipos pueden estar asociadas a diferentes workflows y por lo tanto tener estados y transiciones diferentes.

Plugins en JIRA.

Como se ha comentado anteriormente JIRA incorpora un sistema de plugins propio. Además incorpora un API propio. Los plugins son archivos .jar que contienen el código, recursos (principalmente plantillas de velocity, aunque puede contener otros recursos) y un archivo descriptor XML, sirven para añadir nueva funcionalidad o extender una funcionalidad existente. Cada plugin está compuesto de módulos, existen distintos tipos de módulos en JIRA dependiendo de la funcionalidad que se desea añadir.

administrador Filtros Salir

HOME BROWSE PROJECTS BUSCAR INCIDENCIAS NUEVA INCIDENCIA ADMINISTRACIÓN BÚSQUEDA RÁPIDA:

▼ Proyecto

- Proyectos
- Project Categories

▼ Users, Groups & Roles

- User Browser
- Group Browser
- Project Role Browser

▼ Global Settings

- Attachments
- CVS Modules
- Default Dashboard
- Events
- General Configuration
- Global Permissions
- Issue Linking
- Look and Feel
- Mail Servers
- Subversion Repositories
- Sub-Tasks
- Time Tracking
- Trackbacks
- User Defaults
- Workflows

▼ Schemes

- Issue Security Schemes
- Notification Schemes
- Permission Schemes
- Workflow Schemes
- Scheme Tools

▼ Issue Fields

- Custom Fields
- Field Configurations
- Field Configuration Schemes
- Navigator Columns
- Screens
- Screen Schemes
- Issue Type Screen Schemes

▼ Issue Settings

- Issue Types
- Priorities
- Resolutions
- Statuses

▼ Import & Export

- Backup Data to XML
- Restore Data from XML
- External System Import
- Project Import

▼ Options & Settings

- Jelly Runner
- Send E-mail
- Announcement Banner

▼ System

- Indexing
- Integrity Checker
- LDAP
- License Details
- Listeners
- Logging & Profiling
- Mail Queue
- Plugins
- Scheduler Details
- Services
- System Info
- Support Request
- Trusted Applications

▼ Hide All

Current Plugins

Plugins are used to extend the functionality of JIRA in different ways. The [JIRA Extension](#) space is a great place to download extra plugins contributed by the JIRA community.

Installed Plugins

- [Workflow Plugin](#) (20 module(s))
- [Issue Operations Plugin](#) (18 module(s))
- [Custom Field Types & Searchers](#) (36 module(s))
- [Portlets Plugin](#) (18 module(s))
- [JIRA Subversion plugin](#) (5 module(s))
- [Wiki Renderer Macros Plugin](#) (8 module(s))
- [User Format](#) (6 module(s))
- [JIRA Suite Utilities](#) (13 module(s))
- [View Project Operations Sections](#) (3 module(s))
- [Top Navigation Bar](#) (7 module(s))
- [Project Role Actors Plugin](#) (4 module(s))
- TSB CMMI Plugin** (23 module(s))
- [Renderer Component Factories Plugin](#) (22 module(s))
- [Web Resources Plugin](#) (55 module(s))
- [JIRA Footer](#) (1 module(s))
- [Preset Filters Sections](#) (9 module(s))
- [Admin Menu Sections](#) (67 module(s))
- [User Profile Links](#) (13 module(s))
- [Reports Plugin](#) (4 module(s))
- [Webwork Plugin](#) (1 module(s))
- [User Navigation Bar Sections](#) (8 module(s))
- [Issue Tab Panels Plugin](#) (5 module(s))
- [Content Link Resolvers Plugin](#) (4 module(s))
- [Browse Project Operations Sections](#) (3 module(s))
- [Custom Field Search](#) (1 module(s))
- [Issue Views Plugin](#) (11 module(s))
- [Renderer Plugin](#) (4 module(s))
- [Project Panels Plugin](#) (12 module(s))

TSB CMMI Plugin

Descripción: Jira plugin with customized element for TSB cmmi processes.
Vendor: TSB
Plugin Version: 0.0.14
JIRA Version: 0.0

Disable plugin

Create related task (createrelatedtask-function) Creates a related task, linking it with the current issue and a task group.	Disable module
Auto link with issue (autolink-function) Automatically creates a link between the current issue and another issue specified by a custom field.	Disable module
Sets field to current date (setcurrentdate-function) Sets an issue field to the date and time of the transition.	Disable module
Copy value to related issues (copyrelated-function) Copies the value of an issue field to all related issues (following a specific link type) possibly recursively.	Disable module
Related issue in status (relatedissueinstatus-condition) Condition to allow to execute the transition only if all the issues related through a specific link type and complying a given pattern are in a specified status.	Disable module
Issue must have a file attached (hasattachment-condition) The issue must have at least one file attached.	Disable module
Link condition (link-condition) Workflow condition to verify that the current issue contains at least one link of a given type (or at least n links, or exactly n links, etc.).	Disable module
Baseline validator (baseline-validator) Validates that all configuration items with explicit versioning are included in a baseline.	Disable module
Linked issue validator (linkedissue-validator) Workflow validator that checks that at least a link of a given type with an issue of a given type has been specified.	Disable module
One field has been specified (fieldhasbeenspecified-validator) Workflow validator like the required field validator that check that at least one field among the configured ones has been specified.	Disable module
Link validator (link-validator) Workflow validator to verify that the current issue contains at least one link of a given type (or at least n links, or exactly n links, etc.).	Disable module
Baseline notes (baseline) Baseline notes	Disable module
Project progress report (Advance) Project progress	Disable module
I2 progress report (AdvanceI2) I2 progress report	Disable module
Configuration items report (configurationitems) Configuration items report	Disable module
Status report (status) Status report	Disable module
Status report progress (statusProgress) Status report. 2. Project Progress	Disable module
Status report development status (developmentStatus) Status report. 3. Development Status	Disable module
Status report assurance (statusAssurance) Status report. 4. Quality Assurance Project	Disable module
Status report risks (statusRisk) Status report. 5. Identified Risks	Disable module
Status report RACI (statusRaci) Status report. 6. Follow up of the Involvement Plan	Disable module
Status report project status (projectStatus) Status report. 7. Measurement Plan	Disable module
Audit report (auditreport) Audit Report	Disable module

This JIRA site is for demonstration purposes only. [Evaluate JIRA](#) - bug tracking software for your team.

Powered by Atlassian JIRA the Professional Issue Tracker. (Enterprise Edition, Version: 3.13-#330) - [Bug/feature request](#) - [Atlassian news](#) - [Contact Administrators](#)

Figura 48: Descripción del plugin en la pantalla de administración de plugins

8.5 CASOS DE USO

Caso de Uso	Generar el informe.
Descripción	Generar las secciones del informe que no precisan evaluación.
Actor	Jefe de proyecto.
Resumen	<p>El jefe de proyecto accede a la pantalla del informe a través del proyecto.</p> <p>Introduce los datos del informe.</p> <p>La aplicación muestra un informe con la información referente al progreso del proyecto.</p>
Precondiciones	<p>Estar dado de alta en la aplicación JIRA y pertenecer al grupo jefes de proyecto del proyecto especificado.</p> <p>El proyecto modelado en JIRA debe cumplir con ciertas especificaciones:</p> <p>Debe estar modelado conforme a una estructura de descomposición del trabajo (WBS) en árbol, grupos de tareas que representan fases del proyecto y que agrupan a otros grupos de tareas y tareas.</p> <p>Debe contener una incidencia con la información del proyecto, con los campos requeridos.</p>
Postcondiciones	
Flujo de eventos	<p>En la aplicación JIRA:</p> <p>Seleccionar el proyecto.</p> <p>Entrar en la pantalla de proyecto.</p> <p>Seleccionar el informe (Informe de avance, informe de estado).</p> <p>En la pantalla del informe:</p> <p>Seleccionar el Modo “generar el informe”.</p> <p>Seleccionar el tipo de informe a generar (Informe de iteración / release, final del proyecto, final de la fase G1, final de la fase G2).</p> <p>Seleccionar la incidencia de la iteración o release.</p>
Excepciones	<p>La selección del modo de informe como del tipo de informe solo se realiza si se ha seleccionado el informe de estado.</p> <p>La selección de la iteración release solo se realiza si el tipo de informe es de iteración / release</p>

Caso de Uso	Generar el informe y obtener las evaluaciones.
Descripción	Generar todas las secciones del informe junto con los indicadores y obtiene la evaluación de los indicadores teniendo en cuenta las condiciones que obtiene del maestro de indicadores almacenado en la base de datos.
Actor	Jefe de proyecto.
Resumen	El jefe de proyecto accede a la pantalla del informe a través del proyecto. Introduce los datos del informe. La aplicación muestra un informe con la información referente al progreso del proyecto y muestra la evaluación de los indicadores del proyecto.
Precondiciones	Estar dado de alta en la aplicación JIRA y pertenecer al grupo jefes de proyecto del proyecto especificado. El proyecto modelado en JIRA debe cumplir con ciertas especificaciones: Debe estar modelado conforme a una estructura de descomposición del trabajo (WBS) en árbol, grupos de tareas que representan fases del proyecto y que agrupan a otros grupos de tareas y tareas. Debe contener una incidencia del proyecto, con los campos requeridos.
Postcondiciones	
Flujo de eventos	En la aplicación JIRA: Seleccionar el proyecto. Entrar en la pantalla de proyecto. Seleccionar el informe (Informe de estado, Informe de avance). En la pantalla del informe: Seleccionar el Modo “generar el informe y almacenar indicadores”. Seleccionar el tipo de informe a generar (Informe de iteración / release, final del proyecto). Seleccionar la incidencia de la iteración o release.
Excepciones	La selección del modo de informe como del tipo de informe solo se realiza si se ha seleccionado el informe de estado. La selección de la iteración release solo se realiza si el tipo de informe es de iteración / release

Caso de Uso	Generar el informe y almacenar los indicadores.
Descripción	Generar todas las secciones del informe, además almacena los indicadores generados en una base de datos para su posterior uso.
Actor	Jefe de proyecto.
Resumen	El jefe de proyecto accede a la pantalla del informe a través del proyecto. Introduce los datos del informe. La aplicación muestra un informe con la información referente al progreso del proyecto y muestra la evaluación de los indicadores del proyecto.
Precondiciones	Estar dado de alta en la aplicación JIRA y pertenecer al grupo jefes de proyecto del proyecto especificado. El proyecto modelado en JIRA debe cumplir con ciertas especificaciones: Debe estar modelado conforme a una estructura de descomposición del trabajo (WBS) en árbol, grupos de tareas que representan fases del proyecto y que agrupan a otros grupos de tareas y tareas. Debe contener una incidencia del proyecto, con los campos requeridos.
Postcondiciones	Los indicadores generados por el informe se almacenan en la base de datos.
Flujo de eventos	En la aplicación JIRA: Seleccionar el proyecto. Entrar en la pantalla de proyecto. Seleccionar el informe (Informe de estado, Informe de avance). En la pantalla del informe: Seleccionar el Modo “generar el informe y almacenar indicadores”. Seleccionar el tipo de informe a generar (Informe de iteración / release, final del proyecto). Seleccionar la incidencia de la iteración o release.
Excepciones	La selección del modo de informe como del tipo de informe solo se realiza si se ha seleccionado el informe de estado. La selección de la iteración release solo se realiza si el tipo de informe es de iteración / release

Caso de Uso	Añadir condición / validador
Descripción	Añade una condición o validador entre los disponibles a un paso de un flujo de trabajo concreto.
Actor	Administrador
Resumen	El administrador accede a la pantalla de edición del flujo de trabajo. Añade una condición o validador. Introduce los datos de configuración.
Precondiciones	Estar dado de alta en la aplicación JIRA y poseer permisos globales de administración de JIRA. Solo se pueden editar los flujos de trabajo inactivos, se pueden crear borradores de los flujos de trabajo activos para ser editados.
Postcondiciones	Se crea una nueva condición o validador en un paso de un flujo de trabajo concreto.
Flujo de eventos	El administrador accede a la pantalla de edición del paso del flujo de trabajo. Añade una condición o validador. La aplicación muestra la pantalla de configuración de la condición o validador junto con los campos de configuración. El administrador introduce los datos de configuración. La aplicación muestra la pantalla del paso del flujo de trabajo con la nueva condición o validador en la lista de condiciones y validadores del paso de flujo de trabajo.
Excepciones	Algunas condiciones o validadores pueden no necesitar parámetros de configuración, como por ejemplo la condición de archivo adjunto.

Caso de Uso	Editar condición / validador
Descripción	Editar una condición o validador asignado a un paso de un flujo de trabajo concreto.
Actor	Administrador
Resumen	<p>El administrador accede a la pantalla de edición del flujo de trabajo.</p> <p>Elige una condición o validador existente.</p> <p>Edita los datos de configuración.</p>
Precondiciones	<p>Estar dado de alta en la aplicación JIRA y poseer permisos globales de administración de JIRA.</p> <p>Solo se pueden editar los flujos de trabajo inactivos, se pueden crear borradores de los flujos de trabajo activos para ser editados.</p>
Postcondiciones	
Flujo de eventos	<p>El administrador accede a la pantalla de edición del paso del flujo de trabajo.</p> <p>Elige una condición o validador de entre la lista de condiciones y validadores del paso del flujo de trabajo.</p> <p>La aplicación muestra la pantalla de configuración de la condición o validador junto con los campos de configuración.</p> <p>El administrador introduce los datos de configuración.</p> <p>La aplicación muestra la pantalla del paso del flujo de trabajo con la nueva condición o validador</p>
Excepciones	

Caso de Uso	Realizar transición del flujo de trabajo
Descripción	El usuario realiza una transición del flujo de trabajo para una incidencia
Actor	Usuario
Resumen	El usuario realiza una transición en el flujo de trabajo de una incidencia dada.
Precondiciones	El usuario debe estar dado de alta en la aplicación. El usuario debe de tener permisos para poder ejecutar la acción.
Postcondiciones	La incidencia pasa del estado del flujo de trabajo actual al estado objetivo en la transición.
Flujo de eventos	<p>El usuario accede a la pantalla de la incidencia.</p> <p>Si se cumplen las condiciones establecidas la aplicación mostrara las transiciones disponibles.</p> <p>El usuario accede a la transición.</p> <p>La aplicación muestra la pantalla de transición del flujo de trabajo.</p> <p>El usuario introduce los datos.</p> <p>Si no se cumplen los validadores del paso del flujo de trabajo se muestra un mensaje de error.</p> <p>Si se cumple los validadores del paso del flujo de trabajo la aplicación realiza las postfunciones.</p> <p>El estado de la incidencia se actualiza al nuevo estado.</p> <p>La aplicación muestra la pantalla de la incidencia.</p>
Excepciones	Pueden haber pasos del flujo de trabajo que no requieran datos de entrada en ese caso se obvian dichos pasos.

