

Implementación de un sistema para gestión y seguimiento de los documentos procesados por la Facultad de Ingeniería en Electricidad y Computación.

Karla Yadira Abad Sacoto, Larry Richard Beltrán Flores, Carmen Karina Vaca Ruiz, Msc.
Facultad de Ingeniería en Electricidad y Computación
Escuela Superior Politécnica del Litoral
Campus "Gustavo Galindo V.", Km 30.5, Vía Perimetral, Guayaquil, Ecuador
kabad@fiec.espol.edu.ec, lbeltran@fiec.espol.edu.ec, cvaca@espol.edu.ec

Resumen

La necesidad de contar con un sistema de gerencia de procesos de negocio, se debe a que en cualquiera de ellos es crucial la comunicación entre las partes. Un gran número de personas tienen que coordinar sus actividades o al menos compartir información durante el desarrollo del proceso. Aunque todos estos sistemas se implementan, en general, sobre paquetes o herramientas existentes, hay un trabajo importante de definición de los procedimientos y reglas, así como de integración con las herramientas habituales del cliente, en sus sistemas informáticos. Para desarrollar un sistema que administre procesos es recomendable utilizar un motor de Workflow que administre la ejecución de estos.

En este artículo se expone un análisis de distintos motores existentes, con el objetivo de mostrar las ventajas y limitaciones que cada uno provee. El análisis se realiza a través de la comparación de cinco motores de Workflow no comerciales y en base a características importantes que deben proveer los motores actuales. Luego se resume el desarrollo de la implementación del proyecto utilizando una de las herramientas de Workflow analizadas.

Palabras Claves: flujos de trabajo, análisis, proceso, motor, Bonita, sistema, solicitudes.

Abstract

The need for a business processes management system is because in any process the communication between the parties is critic. Many people have to coordinate their activities or at least share information during the development process. Although all these systems are implemented in general on existing tools or packages, there is a key task of defining the rules and procedures, as well as integration with usual tools of the customer, in their computer systems. In order to develop a system that manages processes you should use a Workflow engine to manage the implementation of these.

This article shows an analysis of various existing engines in order to show the advantages and limitations that each provides. The analysis is done through the comparison of five engines and non-commercial Workflows based on important features that should provide today's engines. After, you can find a summary about the development of project implementation using one of these analyzed Workflow tools.

1. Introducción

Los sistemas de Workflow (Flujos de Trabajo) sirven para automatizar procesos administrativos; los cuales encierran un conjunto de actividades que deben realizarse una tras otra, teniendo en cuenta la información obtenida en ellas para decidir el siguiente paso a dar. Las diferentes actividades son realizadas por los usuarios del sistema, dependiendo de cuál es su rol y cuáles sean sus funciones. Una aplicación implementada a través de esta tecnología guarda la información, reglas, roles y otros elementos sobre un servidor de base de datos relacionados, ejecutándose sobre una interfaz gráfica para interacción con los usuarios.

De elegir una buena herramienta de Workflow, depende que el sistema implementado sea robusto y que los desarrolladores puedan definir todas las reglas de negocio existentes en cada proceso que encierra una solicitud.

El propósito de este artículo es exponer un sistema que permita automatizar el procesamiento de documentos gestionados por la Facultad de Ingeniería en Electricidad y Computación de la ESPOL y que constituya una herramienta de comunicación entre la facultad y las personas involucradas en los procesos, de tal manera que estos puedan conocer el estado de todos sus trámites en cualquier momento.

En la Sección 2 del artículo se definen el análisis y selección de un motor de Workflow, en la Sección 3 se justifica y describe el motor escogido. En la Sección 4 se detalla la implementación del sistema. Luego se incluyen las conclusiones y recomendaciones en las Secciones 5 y 6, y se especifican las referencias en la Sección 7.

2. Análisis y selección de un motor de flujo de trabajo.

Un motor de flujo de trabajo es el software que permite controlar la ejecución de una instancia de proceso. Por medio de un “motor de Workflow” situado en el servidor de la red, se van encaminando los procesos de acuerdo con las reglas de transición establecidas, y facilitan la realización de actividades pendientes de los usuarios integrando en el proceso formularios electrónicos, acceso a bases de datos.

En el mercado actualmente encontramos muchos motores de Workflow, tanto comerciales como no comerciales.

Los motores comerciales son bastantes costosos y además impiden hacer cambios en su estructura debido a que no hay acceso al código con el cual fue implementado; por ello los usuarios de estos tipos de implementaciones obligan a sus sistemas a acoplarse al Workflow con poca oportunidad de hacerlo más ajustable a la situación real del negocio. Estas razones nos llevaron a analizar herramientas de software libre,

gratuitas y de ser posible de código abierto.

A continuación listamos las soluciones factibles (es decir son más robustas y brindan mayor soporte).

- JBPM [1]
- NetBeans BPEL [2]
- Object Web Bonita [3]
- Enhydra Shark [4]
- YAWL [5]

Con el objetivo de encontrar la mejor herramienta revisaremos las principales características de estos motores para luego analizar y concluir cuál es la mejor herramienta.

2.1. Características de un Workflow

Para analizar la mejor herramienta, hemos definido parámetros importantes que se deben cumplir, a fin de que sea provechosa su utilización.

- Ser libre, y de ser posible de código abierto para poder realizar pruebas y modificaciones en caso de ser necesario.
- Cumplir con los estándares propuestos por la WFMC.
- De fácil implantación y de ser posible integrable con diferentes tecnologías.
- Poseer una herramienta gráfica para el diseño de los modelos de proceso, la cual permita definir gráficamente la ruta de cada documento y los estados por los que va pasando.
- Proveer un conjunto de APIs que permiten un control completo de la definición y ejecución del proceso. Un API [6] es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.
- Que posea una herramienta para definir las reglas de negocio y asociación de roles a actividades.
- Permitir monitorear las instancias y actividades en ejecución.
- Permitir manejar tiempos plazos y eventos.
- Permitir anexar datos en cada actividad.
- Facilitar la interconexión con aplicaciones externas.
- Listar el trabajo asignado a cada usuario.
- Poseer gestión de históricos, es decir que permita revisar procesos que han sido realizados anteriormente.
- Ofrecer gran cantidad de documentación e información disponible.

Estas características han sido nuestros referentes para concluir cuál es la herramienta más óptima en la implementación de un sistema para manejo de procesos. Para realizar el análisis se ha revisado cada una de las características a través de lectura de documentos y pruebas de implementación.

A continuación se muestra una tabla comparativa entre las diferentes herramientas estudiadas y los parámetros de análisis.

	JBPM	BPEL	Bonita	YAWL	Enhydra
Herramienta Libre y de código abierto	✓		✓	✓	
Cumpla Estándares de la WFMC			✓	✓	✓
De fácil implantación e integrable con otras tecnologías	✓	✓	✓		
Herramientas gráficas para diseños de procesos	✓	✓	✓	✓	✓
Provee APIs	✓	✓	✓	✓	✓
Permite definir reglas de negocio	✓	✓	✓	✓	✓
Monitoreo de las instancias y actividades en ejecución	✓	✓	✓	✓	✓
Manejo de tiempos plazos y eventos	✓		✓		✓
Anexar datos en cada actividad	✓	✓	✓	✓	✓
Definir perfiles de usuario y roles	✓	✓	✓	✓	✓
Integración con bases de datos	✓	✓	✓	✓	✓
Facilidad para la interconexión con aplicaciones externas	✓	✓	✓		
Listado de trabajo asignado	✓		✓	✓	✓
Gestión de históricos			✓		
Gran cantidad de documentación e información disponible			✓		

Tabla1. Comparación entre diferentes motores de flujo de trabajo.

2.2. Análisis

Como se puede apreciar en la tabla anterior, las implementaciones que hemos analizado comparten muchas características comunes.

Todas proveen herramientas gráficas para diseño de procesos, aunque cabe mencionar que las herramientas ofrecidas por YAWL, Bonita y Enhydra son más completas y además permiten realizar diseños de procesos complejos (esto es debido a que permiten definir de mejor manera las reglas del negocio).

Otra característica común es que todas se pueden configurar con diferentes motores de base de datos y servidor de aplicaciones. Sin embargo, cada una utiliza un servidor de aplicaciones, base de datos e IDE (entorno de desarrollo integrado) por defecto. Un punto para tomar la decisión de que herramienta elegir ha sido precisamente la facilidad de la herramienta para configurarse sobre la plataforma que usaremos en la implementación.

Todas son libres pues se pueden conseguir gratuitamente sin embargo BPEL y Enhydra no permiten obtener el código fuente de su implementación.

Para este análisis realizamos pruebas con cada uno de los motores por lo que sabemos que Enhydra y YAWL son bastante complejos de instalar y probar. Además no existe suficiente información sobre cómo instalar la herramienta o cómo resolver los problemas que se pueden suscitar en el momento de la instalación. Por otra parte tanto Netbeans BPEL y JBPM vienen integrados en un IDE de desarrollo por lo que basta con conseguir el instalador adecuado del IDE y éste se encargara de la instalación y configuración por defecto del motor. En cuanto a Bonita podemos decir que instalarlo y probarlo bajo sus requerimientos por defecto (es decir servidor de aplicaciones y motor de base de datos) es sencillo.

Aunque todas las herramientas estudiadas proveen aplicaciones para monitoreo de los procesos en curso y de las actividades pendientes, se debe tener en consideración que unas son más sofisticadas que otras. Un ejemplo de ello es Bonita, el cual provee una aplicación (usada a través de un explorador) que bien podría ser usada a nivel empresarial sin necesidad de implementar algo más, debido a que esta herramienta permite realizar actividades, ver procesos en curso y procesos terminados, ver actividades pendientes, administración de usuario y roles, etc. En el caso del sistema a implementarse en esta tesis una herramienta de estas características sería muy valiosa para poder realizar pruebas de manera rápida y fácil.

Uno de los puntos más importantes a tomar en cuenta para la elección, ha sido el conjunto de APIs que provee cada herramienta. Los APIs son el medio de comunicación entre los componentes de software, que nos permiten acceder a las diferentes funciones y procedimientos de cada uno. En este caso se utilizarán para poder acceder a todas las funcionalidades que el

motor seleccionado otorgue durante la implementación del sistema.

Después de haber probado y analizado cada una de las herramientas presentadas concluimos que la mejor opción es Workflow es Object Web Bonita. Una de las principales ventajas de esta herramienta es la gran cantidad de documentos e información existentes, que se puede conseguir de manera libre y gratuita.

También cuenta con un gran número de componentes, entre ellos una herramienta gráfica para diseño de procesos que facilita de gran manera la tarea de modelar. Además es integrable con diferentes plataformas y ofrece un API completo que permite hacer uso de todas las funcionalidades del motor.

En la siguiente sección se presenta de manera general las principales características del motor seleccionado.

3. Workflow Bonita

Bonita es un sistema de flujo de trabajo que ofrece una gran cantidad de características innovadoras, como la infraestructura que permite a los usuarios estar informados de cualquier acontecimiento ocurrido durante la ejecución de un determinado proceso. También posee las características tradicionales de flujos de trabajo como la resolución dinámica de usuarios/roles, la ejecución secuencial y ejecutante de la actividades, que le permite dar soporte a procesos de flujo de trabajo cooperativos y administrativos.

Principales beneficios:

- Mejorar la eficiencia del trabajo en colaboración con Bonita: un equipo comparte un punto de vista común de las tareas actuales, cada individuo puede ver lo que sucede en tiempo real.
- Reducir los costos y los riesgos mediante la automatización de persona a persona y de sistema a los procesos del sistema. Los procesos pueden extenderse a lo largo de las organizaciones y ubicaciones geográficas.
- Manejar con eficiencia situaciones inesperadas: Bonita le permite modificar dinámicamente y de forma segura a la definición de un proceso en ejecución a fin de tener en cuenta los acontecimientos que no fueron planificados.
- Tomar beneficio de la cualidad de los servicios que provee la integración de J2EE con un servidor de aplicaciones. Estas cualidades incluyen las transacciones de servicios, roles y la autenticación basada en la gestión del ciclo de vida, además la conexión con sistemas de información externos.

En las últimas versiones de Bonita se ha estandarizado la definición del proceso del flujo de trabajo, a fin de permitir a los usuarios migrar fácilmente su definición de proceso con un mínimo

costo. Esa es la razón por la que se decidió prestar apoyo a la norma XPDL. XPDL es el acrónimo de XML Lenguaje de Definición de Proceso, un lenguaje basado en XML definido por la WfMC con el objetivo de tener interoperabilidad entre los distintos proveedores de flujos de trabajo. De esa manera podríamos fácilmente importar una definición de proceso desde un proveedor compatible con XPDL [7].

La mayoría de estas características representan un importante valor añadido con respecto a los otros Workflows de código abierto, en términos de flexibilidad, soporte para procesos corporativos y el beneficio que brinda el servidor de aplicaciones a través de la calidad de servicios. Estos servicios están disponibles para el usuario final a través de un conjunto de ricas APIs Java / Servicios Web. De esta forma, los usuarios pueden definir y ejecutar los procesos de flujo de trabajo a través del uso de los APIs.

3.1. Calidad de Servicios

Uno de los principales logros en Bonita Workflow es que todos los datos, incluidos los datos relevantes de flujo de trabajo y los estados internos, se persisten durante la ejecución de todo el proceso. Esto significa que cada vez que una actividad que modifica su estado este cambio se guarda en la base de datos de Bonita a lo largo de toda la información o datos que gestiona esta actividad.

El mecanismo de persistencia y de otros servicios, como la seguridad, el correo electrónico, conectores JCA (conecta servidores de aplicaciones y sistemas de información de empresa) o notificaciones de eventos son gestionados automáticamente por el servidor de aplicaciones de Bonita en la que el motor esté en marcha. Este enfoque tiene muchas ventajas en términos de calidad del servicio.

Servicios comunes: El sistema de Workflow Bonita integra un gran número de servicios para el control y simplificación de muchos aspectos relativos a la cooperación: Servicio de mensajes JMS: implementación para notificar la definición y cambios en la ejecución dentro de un proceso de flujo de trabajo. Cada interacción con el usuario es notificado al núcleo de Bonita y lanza un evento JMS (API de Servicios de Mensajería de Java). Servicio de plazos para la ejecución de actividades: que utilizan la gestión de extensiones de Java JMX para prevenir a los usuarios en caso de que una actividad no termine en la fecha prevista. Servicio de notificación Jabber: que permite a los usuarios recibir notificaciones en tiempo real e intercambio de diferentes tipos de mensajes.

4. Sistema solicitudes FIEC

El Sistema Solicitudes FIEC es una aplicación web, a la cual se podrá acceder a través de internet, con una cuenta de usuario de ESPOL.

El sistema fue implementado sobre la tecnología J2EE (Java 2 Enterprise Edition), utilizando el lenguaje de programación Java y el entorno de desarrollo Netbeans.

Aplicaciones externas:

Para proveer una interfaz de comunicación con sistemas externos se utilizaron Servicios Web o Webservices. Un **servicio web** [8] 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. Por ejemplo, en el sistema se ha utilizado esta tecnología para obtener información de los usuarios, proporcionada por el Centro de Servicios Informáticos de ESPOL (CSI).

Analizando el sistema encontramos que es posible dividirlo en tres grandes módulos que son: los flujos de proceso y el motor Bonita, la aplicación web encargada de interactuar con el usuario y la persistencia de los datos.

4.1. Flujos de procesos.

El primer paso para implementar un proceso es definir el modelo. Bonita usa el lenguaje de definición de procesos XPDL, pero ventajosamente no es necesario definir el modelo utilizando el lenguaje, dado que el motor provee una herramienta de modelado que permite definir el proceso de manera gráfica, llamada **proed**. El diagrama generado (modelo de proceso) es almacenado como un archivo XPDL y luego debe ser importado al motor de Workflow Bonita para poder iniciar nuevas instancias del proceso.

Para administrar los procesos se utilizaron las herramientas provistas por Bonita: la consola de administración **jiapAdmin** para gestionar usuarios, importar definiciones de procesos en formato XPDL, eliminar procesos, administrar instancias, etc., así como el editor gráfico **proed** para construir modelos de procesos.

Los principales elementos que se pueden definir en un modelo de proceso son:

Roles: responsabilidades de los participantes (usuarios a los que se les permite jugar algún rol) en una actividad.

Actividades: conjunto de acciones que generalmente son desempeñadas por una única persona dentro de los roles que pueden realizar dicha tarea.

Reglas de Transición: son reglas lógicas que determinan la navegación del procesos dentro del sistema. Expresan qué acción se va a tomar dependiendo del valor de expresiones lógicas.

Actividades de ruteo: permiten tomar decisiones que determinan la navegación (ruta) dentro del proceso. Expresan qué acción se va a tomar dependiendo de la evaluación de reglas de transición.

Rutas del Proceso: caminos que va tomando un proceso en base a reglas de transición.

Iteraciones: una iteración permite regresar a un punto determinado del proceso. Es decir, recorrer el proceso nuevamente con la posibilidad de cambiar ciertos parámetros que nos lleven a la ruta deseada.

Plazos: los plazos colocados en los procesos analizados determinarán la fecha límite para realizar una actividad.

Mappers: especifica qué personas pertenecen a un rol del proceso en particular.

Eventos (Hooks): un evento es una interrupción que contiene información, el mismo tiene un origen y uno o más destinatarios. Por ejemplo, cuando se decide rechazar una solicitud, se dispara un evento para que envíe un correo notificando a los involucrados.

4.2 Aplicación Web.

El uso de las aplicaciones Web tiene múltiples ventajas, no necesitan ser descargadas ni configuradas permitiéndoles a los usuarios acceder en cualquier momento y desde cualquier lugar utilizando su respectivo usuario y contraseña.

La aplicación Web implementada para el proyecto sirve de nexo entre el motor de flujo de trabajos y los usuarios que harán uso del sistema. A través de la aplicación los usuarios podrán iniciar sus solicitudes, revisar el estado actual de estas; así como también los directivos podrán evaluarlas a través de formularios Web.

Para implementar esta aplicación se utilizó el marco de trabajo Java Server Faces, que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE.

4.3 Persistencia de datos

Para almacenar la información del sistema se ha escogido el servidor de base de datos My SQL.

My SQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario. Es muy utilizado en aplicaciones web porque hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

Para el sistema Solicitudes FIEC se han utilizado tres bases de datos las cuales almacenan la información del sistema de acuerdo a:

Base de datos Bonita: es administrada totalmente por el motor de Workflow Bonita y aquí se almacenan los registros correspondientes a los procesos, instancias de procesos, actividades, roles, usuarios y demás necesarios para describir un proceso.

Base de datos historial: el motor Bonita provee de historial, almacenando un archivo XML por cada instancia de proceso que finaliza. Para implementar el sistema se reemplazaron los archivos XML por una base de datos transaccional, debido a que manejar la información de las instancias a través de un archivo XML es mucho más complejo y requiere de mayor cantidad de tiempo para el acceso a los datos.

La base de datos para el historial replica información de las tablas más importantes de la base de datos de Bonita. Esta acción es realizada a fin de salvaguardar las instancias de proceso que finalizan su flujo, dado que Bonita las elimina para mantener su base de datos con pocos registros y así optimizar tiempos de respuesta.

Base de datos solicitudes: la base de datos Bonita almacena netamente información de los procesos tales como las actividades, transiciones, condiciones, atributos, etc. Por ello fue necesario crear una base de datos para almacenar los datos granulares de las solicitudes. El modelo de esta base contiene una tabla por cada solicitud, además de los datos personales de los usuarios y los roles a los que pertenecen, y también almacena datos necesarios para iniciar las solicitudes como las materias y carreras de la facultad.

5. Conclusiones

El sistema Solicitudes FIEC contribuye a la mejora de la calidad de servicio otorgada por la facultad, porque agiliza notablemente los trámites comparándolo con realizarlo de manera manual. Los participantes ya no deberán acercarse a la secretaría para conocer el estado actual de su solicitud, si no que podrán hacerlo en línea a través de la aplicación Web implementada, sin necesidad de encontrarse físicamente dentro de la universidad.

La selección de una adecuada herramienta de Workflow es de vital importancia para el desarrollo de un sistema que utiliza esta tecnología. Por ello es necesario realizar un análisis previo de las herramientas existentes en el mercado. Este análisis se debe llevar a cabo tomando en cuenta la naturaleza del sistema a implementar y de los procesos a automatizar.

Toda institución en la cual existan procesos que involucre manejo de documentos y que deban ser evaluados por varias personas en una secuencia determinada, deberían optar por implementarlos utilizando una herramienta de Workflow para conseguir mayor productividad y mejorar la calidad de los servicios que ofrezca.

Una empresa podría decidir implementar sus procesos desde cero, sin utilizar una herramienta de Workflow, pero perdería las ventajas que brindan estas herramientas y la implementación requeriría de mucho más tiempo para realizarse.

6. Recomendaciones

Se debe considerar que el motor de Workflow Bonita no incluye una herramienta para relacionar dos o más modelos de procesos. Esta relación debe ser realizada por los desarrolladores del sistema. Se recomienda crear una entidad en el modelo de base de datos que relacione estos procesos.

7. Referencias

- [1] JBoss JBPM, “*Project Home*”, <<http://www.jboss.com/products/jbpm>>
- [2] Netbeans Proyecto BPEL: Home, “*Project Home*”, <<http://enterprise.netbeans.org/soa/>>
- [3] YAWL: Home, “*Project Home*”, <<http://www.yawlfoundation.org/>>
- [4] Object Web Bonita: Home, “*Project Home*”, <<http://bonita.objectweb.org/>>
- [5] Enhydra Shark: Home, “*Project Home*”, <<http://shark.enhydra.org/>>
- [6] Wikipedia La Enciclopedia Libre, “*API*” Artículo, <http://es.wikipedia.org/wiki/Application_Programming_Interface>
- [7] BONITA Workflow System., “*The BONITA Workflow System XPDL Support*”, Mayo 26, 2006 p.4 <<http://wiki.bonita.objectweb.org/xwiki/bin/download/Main/Documentation/bonitaXPDL.pdf>>
- [8] Wikipedia La Enciclopedia Libre, “*Servicios Web*” Artículo, <http://es.wikipedia.org/wiki/Servicio_Web>