

Implementación de Procesos de Negocio para la Dirección del Parque Nacional Galápagos, Utilizando BPMN 2.0 y Workflow Joget

Gianina Vallejo Gordillo, Víctor Zambrano Ochoa, M.Sc. Carlos Mera Gómez
Facultad de Ingeniería en Electricidad y Computación (FIEC)
Escuela Superior Politécnica del Litoral (ESPOL)
Campus Gustavo Galindo, Km 30.5 vía Perimetral
Apartado 09-01-5863. Guayaquil-Ecuador
gianvall@espol.edu.ec, vxzambra@espol.edu.ec, cjmera@espol.edu.ec

Resumen

La optimización de tiempo y recursos en una organización es un objetivo fundamental en una empresa, alineados con el mismo existen herramientas que ayudan a este propósito. La Dirección del Parque Nacional Galápagos decidió optimizar los procesos de negocio del área de Conservación y Uso de Ecosistemas Marinos, para lo cual se utilizó como modelador de procesos Bizagi Modeler y workflow Joget. Adicionalmente, se realizaron reportes con los datos generados en Pentaho Report. Para asegurar la calidad del desarrollo de la aplicación se realizaron pruebas de carga, profiling e interfaz. El motor de procesos de Joget nos brindó flexibilidad; al implementar las reglas del negocio cambiantes en el tiempo permitiendo la optimización de recursos ya que no involucra grandes inversiones de tiempo en el cambio de los procesos.

Palabras Claves: BPMN, Joget, procesos de negocio, pruebas.

Abstract

The optimization of time and resources in an organization is a key objective for a company, aligned with it, there are tools that help to this purpose. The Directorate of the Galapagos National Park decided to optimize the business process in the area of Conservation and Use of Marine Ecosystems, for which Bizagi Modeler was used as process modeler and Joget workflow. Also, reports were made with the data generated with Pentaho Report. To ensure the quality of the application development load testing, profiling testing and interface testing were made. The Joget process engine gave us flexibility; when implementing the changing business rules among time, permitting the optimization of resources, since it doesn't involve big investments of time in the change of processes.

Keywords: BPMN, Joget, business process, testing

1. Introducción

El conocimiento de sus procesos de negocio le permite a una organización incrementar su productividad y mejorar la atención al cliente. Sin embargo, actualmente no todas las organizaciones los tienen claros.

La optimización de sus procesos de negocio le ofrece a una organización beneficios como: reconocer cuellos de botella; identificando el actor que tarda en ejecutar su actividad, ahorrar tiempo y recursos para la organización y el cliente. Además, se dispone de información actual del estado de cada proceso en la organización, junto con variables que medirán el desempeño de los actores.

Para diagramar procesos de negocio disponemos de la herramienta de modelado de BPMN, Bizagi Modeler (Community Edition) y para la implementación de los mismos, del Workflow Joget (Open Source).

En este proyecto de graduación se modeló e implementó los procesos del área de Conservación y Uso de Ecosistemas Marinos, de la Dirección de Ecosistemas de la Dirección del Parque Nacional Galápagos (DPNG).

2. Descripción del problema

En el área de Conservación y Uso de Ecosistemas Marinos de la Dirección del Parque Nacional Galápagos se ha evidenciado que los procesos de negocio no están documentados; perdiéndose así los beneficios de una adecuada administración de procesos.

La falta de documentación de procesos, y la carencia de un sistema confiable que administre los mismos; motivó un cambio en la Dirección del Parque Nacional Galápagos. Esta institución nacional optó por requerir: documentación de los procesos que funcionan en esta área y un sistema que administre los

mismos según los roles de cada usuario, incluyendo reportes con los datos que se generen.

Cuando una organización gestiona sus procesos: incrementa su productividad, reduce errores, conoce fácilmente el estado actual de sus procesos, se obtiene información real y oportuna. Si el proceso cambia, la gestión del mismo será transparente para el usuario y su implementación llevará menos tiempo de desarrollo.

3. Revisión bibliográfica

En esta sección se revisó conceptos de procesos de negocio, BPMN y workflow, también se realizaron comparaciones entre modeladores de procesos y workflows; luego de analizar estas comparaciones se seleccionó la herramienta que más se ajustaba a los requerimientos dados por la organización.

3.1 Procesos de negocio

Según el Manual de Gestión de Procesos de Negocios de Oracle [1]: “un proceso de negocio es un conjunto de actividades vinculadas que son realizadas por personas y sistemas que ofrecen un valor empresarial a los clientes internos o externos”.

A partir de esta definición podemos añadir que los procesos de negocio brindan un producto o servicio, al final de su proceso. Las personas y sistemas que intervienen en los procesos son llamados actores y estos realizan acciones dentro del proceso.

Según la WfMC (Workflow Management Coalition) en [2], BPM (Business Process Management) se define como: “una estrategia para gestionar y mejorar el desempeño de un negocio a través de una optimización continua de sus procesos; en un ciclo cerrado de modelamiento, ejecución y medición. Las actividades de BPM abarcan la concepción y el descubrimiento a través del desarrollo y gestión de la ejecución de los procesos de negocio con apropiados frameworks”

Para mejorar los procesos de negocio de la organización, se necesitan varias fases sucesivas; las cuales permiten actualizar los procesos a través del tiempo, ajustándolos a las necesidades competitivas de la organización.

3.2 BPMN

Para poder plasmar las actividades de cada proceso, se cuenta con BPMN un estándar para su modelamiento; regulado por la OMG (Object Management Group).

La OMG (Object Management Group) en [3], define BPMN (Business Process Modeling Notation – Notación para el Modelamiento de Procesos de Negocio) como: “una notación gráfica que describe la lógica de los pasos de un proceso de Negocio. Esta

notación ha sido especialmente diseñada para coordinar la secuencia de los procesos y los mensajes que fluyen entre los participantes de las diferentes actividades.”

La notación BPMN nos da un medio de comunicación sencillo entre los usuarios y los técnicos que modelan el proceso. Con este lenguaje común se plasma las actividades de los procesos, con los actores que intervienen en él. Actualmente, va por la versión 2.0 y goza de gran popularidad como notación para modelar procesos.

Los modeladores de procesos son los programas que nos permiten realizar representaciones gráficas de los procesos en un estándar de fácil comprensión para técnicos de TIC’s y usuarios.

Existe una gran variedad de modeladores de procesos en el mercado, entre los más destacados están: Bizagi, Process Maker y BonitaSoft.

Tabla 1. Comparativa entre los modeladores: Bizagi, Process Maker, BonitaSoft

	Bizagi Modeler	Process Maker	BonitaSoft
Licencia	Libre	Libre	Libre
BPMN 2.0	Sí	Sí	Sí
Fácil uso	Sí	Sí	Sí
Diagrama de colaboración	Sí	No	Sí
Mapeo de Datos	Sí	No	Sí
Transformaciones de Datos	Sí	No	Sí
Notificaciones	Sí	Sí	Sí
Análisis de Procesos	Sí	Sí	Sí
Captura de Procesos	Sí	Sí	Sí
Seguimiento de Cambios de Procesos	Sí	Sí	Sí
Modelamiento y Diseño de Procesos	Sí	Sí	Sí
Simulación de Procesos	Sí	No	Sí
Publicación de Procedimientos	Sí	Sí	Sí
Enfoque	Empresas medianas	Empresas medianas	Empresas grandes
Instalación	Sencilla	Sencilla	Complicada

Process Maker presenta deficiencias en comparación con Bizagi y BonitaSoft, ya que no cuenta con la opción de desarrollar diagramas de colaboración, mapeo y transformación de datos.

Bizagi y BonitaSoft comparten muchas características como: análisis, captura y modelamiento de procesos; sin embargo Bizagi presenta una interfaz más intuitiva con mayor acogida en el mercado, además amplía el número de usuarios para los que va dirigido: usuarios de negocio y programadores.

3.3 Workflow

Para ejecutar los modelos de procesos diagramados se necesita un motor; que gestione los mismos en una aplicación. El motor que cumple esta función es llamado Workflow.

Según la WfMC (Workflow Management Coalition) en [10], se define a Workflow como: “La automatización de procesos, en la cual en su totalidad o en parte, documentos, información o tareas son pasadas de un participante a otro para una acción, de acuerdo a un conjunto de normas de procedimiento.”

Los sistemas para representación de Workflow están basados en los flujos de trabajo, es decir en la continuación de actividades o tareas que han sido asignadas a un rol específico. De esta manera se busca que mediante este tipo de sistemas el usuario siga una secuencia de tareas determinadas, con la finalidad que se tenga una noción del estado en que se encuentran los procesos ejecutados y los trámites pendientes por realizar.

Se han analizado tres workflow: Joget, BonitaSoft y Activiti, en la figura 1 se muestra las características comparadas.

	Enhydra	Joget	Bonita	Activiti
Generación de actividades	●	●	●	●
Ruteo de flujos	●	●	●	●
Diseñador de procesos	●	●	●	●
Calendario	●	●	●	●
Reglas de negocio	●	●	●	●
Variables del proceso	●	●	●	●
Notificaciones y alertas	●	●	●	●
Asignación de tareas mediante roles	●	●	●	●
Listas de trabajo	●	●	●	●
Asignación de tareas	●	●	●	●
Seguimiento de actividad y/o proceso	●	●	●	●
Lista de procesos y actividades	●	●	●	●

Figura 1. Características evaluadas en diferentes Workflows

- Soportada completamente.
- Soportada parcialmente
- No soportada

El cuadro comparativo refleja que la generación de actividades en el Workflow Activiti es soportada parcialmente, siendo un ítem de suma importancia para gestionar procesos. Lo que baja interés en la herramienta, junto con una pobre asignación de tareas, seguimiento de actividad y/o proceso y lista de procesos y actividades.

BonitaSoft tiene mejor evaluado los ítems considerados, destaca en reglas de negocio, variables del proceso y notificaciones y alerta. Soporta parcialmente el ruteo del flujo; esta característica le quita interés sobre la herramienta, por ser considerada

primordial en el desarrollo y ejecución de la aplicación.

Joget se destaca en varios ítems evaluados, como son: la generación de actividades y ruteo de flujos. El soporte parcial que brindan en estos ítems: diseñador de procesos, notificaciones y alertas, asignación de tareas son aceptables en comparación con otros Workflow.

4. Análisis y diseño de la solución

4.1 Análisis de la solución

La solución propuesta a la Dirección del Parque Nacional Galápagos, consiste en documentar la versión actual de los procesos de negocio correspondientes al módulo de Conservación y Uso de Ecosistemas Marinos usando BPMN 2.0, además de implementar estos procesos usando la herramienta de Workflow Joget.

4.2 Análisis de requerimientos

Como parte de la documentación e implementación se definieron requerimientos funcionales y no funcionales, con la finalidad de clarificar las necesidades y expectativas de los usuarios del sistema.

Los requerimientos funcionales son oraciones que expresan los servicios que el sistema debe proveer, como el sistema debe reaccionar ante entradas particulares. En algunos casos, los requerimientos funcionales también pueden explícitamente indicar lo que el sistema no debe hacer. [17]

Los requerimientos no funcionales son restricciones o limitaciones a los servicios o funciones que ofrecen el sistema. Incluyen restricciones de tiempo, restricciones en el proceso de desarrollo y limitaciones impuestas por estándares. Los requerimientos no funcionales generalmente aplican al sistema como un todo, en vez de a características individuales de servicios o características del sistema. [18]

4.3 Casos de uso

Para el sistema se consideraron los siguientes actores: Administrador, Responsable de CUEM, Director de Ecosistemas, Técnico de CUEM. En la figura 2, se muestran los casos de uso en los que estos actores interactúan con el sistema.

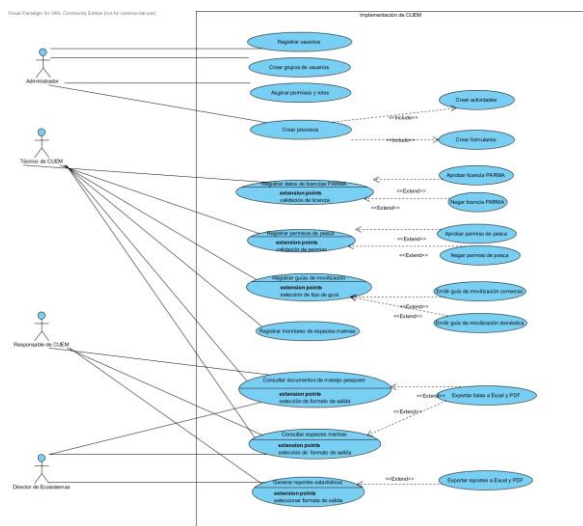


Figura 2. Diagrama de casos de uso

4.4 Diseño de la solución

Para llevar a cabo este proyecto se decidió utilizar dos servidores: uno para servidor de aplicaciones y otro para servidor de base de datos, tal como lo muestra el diagrama de despliegue en la figura 3.

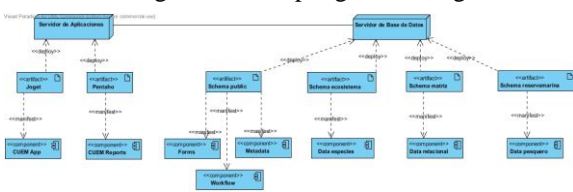


Figura 3. Diagrama de despliegue

En el servidor de aplicaciones se despliega la aplicación de Workflow Joget, que incluye el despliegue de los procesos y formularios de CUEM. Este servidor también contiene la aplicación de Pentaho para generar los reportes.

En el segundo servidor, se encuentra la base de datos creada en PostgreSQL. Aquí encontramos el esquema Public, en donde el Workflow Joget almacena los registros de formularios, datos de workflow, así como datos de las propiedades de la aplicación. Además este servidor cuenta con otros esquemas (ecosistema, matriz y reservamarina) en donde se almacenan la información transaccional, perteneciente a los distintos procesos con los que cuenta CUEM.

A continuación se presenta un mayor detalle de las herramientas que se utilizaron para el desarrollo de este proyecto:

Tabla 2. Herramientas y frameworks de desarrollo

Característica	Descripción
Workflow	Joget Workflow 4.x
Bases de datos del Workflow	PostgreSQL 9.x
Bases de datos del sistema	PostgreSQL 9.x

Framework de desarrollo	JEE
IDE	Netbeans 8.x
Modelador de procesos	Bizagi 2.8
Elaboración de prototipos	Axure RP Pro 7

Además las características mínimas de cada servidor pedidas a la DPNG, se presentan a continuación:

Tabla 3. Requerimientos del sistema

Característica	Descripción
Sistema Operativo	Windows Server o Linux 64 bits (recomendado CentOS)
Memoria RAM	12 GB
Disco Duro	Espacio disponible 2 TB
Conexión Ethernet	Al menos 1 Gbps

5. Definición y modelamiento de los procesos de conservación y uso de ecosistemas marinos

En el área de Conservación y Uso de Ecosistemas Marinos se encuentran dos departamentos: Manejo Pesquero y Monitoreo de Ecosistemas Marinos. En el departamento de Manejo Pesquero se emiten documentos donde se autorizan actividades de pesca y comercialización de especies permitidas. En el departamento de Monitoreo de Ecosistemas Marinos se registra la información sobre los monitoreos, por cada una de las especies marinas nativas, alrededor de las islas.

Para los procesos de los departamentos de CUEM se realizó:

- Un levantamiento de procesos, mediante entrevistas a los usuarios de cada área.
- Un modelamiento de procesos, usando notación BPMN 2.0, mediante el modelador Bizagi.
- Diseño de formularios de las actividades, mediante Axure RP Pro.

6. Implementación de los procesos de conservación y uso de ecosistemas marinos

Para llevar a cabo la implementación de los procesos, se realizaron los flujos de procesos usando la herramienta de workflow Joget.

Además se desarrollaron los reportes usando la herramienta Pentaho Reports, tomando en consideración la información estadística que se maneja y se presenta en cada proceso, para el área de Manejo Pesquero y de Monitoreo de Ecosistemas Marinos desarrollaron 10 reportes en cada área respectivamente.

7. Pruebas

7.1 Pruebas de carga

Las pruebas de carga ayudan a determinar el comportamiento y rendimiento que tiene el sistema dada las condiciones de prueba que se impongan. De esta manera probaremos el servidor para realizar varias transacciones del lado del cliente.

Este tipo de pruebas se llevan a cabo para verificar que la aplicación pueda cumplir con sus objetivos de rendimiento deseados. Además permite medir tiempos de respuesta, tasas de rendimiento y niveles de autorización de recursos, e identificar el punto de ruptura de la aplicación ante cargas máximas. [19]

Para realizar estas pruebas se utilizó el software JMeter [20], que es una aplicación de código abierto que simula la carga en el servidor para medir o analizar el rendimiento, mediante la grabación de cada URL visitada para simularlo varias veces.

En la simulación se ha considerado dos pruebas con 100, 500 y 1000 usuarios concurrentes con tiempos entre cada petición de 4 segundos. Primero se probó con 100 usuarios, ya que se considera que como máximo estos serán los usuarios que normalmente usen el sistema al mismo tiempo. Luego, se aumentó el número de usuarios a 500 para probar los límites de carga y finalmente con 1000 usuarios para analizar el comportamiento del sistema ante una sobrecarga del servidor.

La primera prueba consiste en la consulta y registro de un informe de campo, en el siguiente orden:

- Acceder a la página principal de Monitoreo de Ecosistemas Marinos.
- Consultar la lista de informes de campo registrados.
- Abrir uno de los informes para consultar sus detalles.
- Regresar a la lista de informes de campo.
- Registrar un nuevo informe de campo.
- Regresar a la lista de informes de campo.

La segunda prueba consiste en la consulta de un reporte con sus respectivos parámetros, en el siguiente orden:

- Acceder a la página principal de Monitoreo de Ecosistemas Marinos.
- Consultar el reporte de avistamiento de cetáceos por mes.
- Escoger otro año en el parámetro de selección.
- Consultar el reporte con la nueva selección.

7.2 Pruebas de profiling

Las pruebas de profiling ayudan a medir el rendimiento de la aplicación, tomando en cuenta la cantidad de memoria usada, tiempos de ejecución, cantidad de recursos utilizados por la aplicación.

Para estas pruebas se usó JProfiler [21], que es una herramienta de tipo profiler utilizada permite medir el rendimiento de un programa recolectando y analizando los tiempos de ejecución de la aplicación. Un profiler puede identificar el tiempo utilizado o la frecuencia de uso de varias porciones de código y realizar su análisis comparativo. [22]

Durante estas pruebas se decidió analizar la cantidad de clases, componentes, paquetes, carga de CPU y memoria usada del sistema mientras se realizaban las pruebas de carga, ya que de esta manera se puede tener una idea clara de la variación del consumo de recursos a medida que aumentan las transacciones de los usuarios. Se tomaron los resultados de carga baja durante las pruebas entre 0 y 100 usuarios concurrentes, mientras que se tomaron los resultados de carga alta durante las pruebas entre 500 y 1000 usuarios.

7.3 Pruebas de interfaces

Las pruebas de interfaces nos permiten simular el ingreso de datos por parte de los usuarios y comprobar el ingreso de información correcta.

La herramienta que utilizamos en estas pruebas es Selenium IDE. Esta herramienta permite grabar las interacciones que realizamos en la aplicación y repetir los escenarios. Así podemos simular el ingreso de múltiples usuarios al sistema con diferentes roles, comprobando la respuesta del sistema a las diferentes entradas.

Las pruebas de interfaz se realizaron sobre el formulario de ingreso de pescador, este formulario incluye campos como: nombres, apellidos, cédula, entre otros.

8. Resultados

Los resultados de las pruebas se han dividido según las pruebas que se han realizado.

8.1 Resultados de carga

Durante la primera prueba se obtuvieron los siguientes resultados:

Tabla 4. Resultados de la primera prueba

# usuarios	# muestras	Error %	Rendimiento (transacciones por seg)	KB/seg
100	1700	0 %	37,7/seg	692,38
500	8500	16,72 %	44,9/seg	721,98
1000	16000	47,59 %	67,3/seg	804,42

Durante la segunda prueba se obtuvieron los siguientes resultados:

Tabla 5. Resultados de la segunda prueba

# usuarios	# muestras	Error %	Rendimiento (transacciones por seg)	KB/seg
100	1800	9 %	27,3/seg	424,79
500	9000	28,93 %	28,9/seg	318,58
1000	18000	47,47 %	45,7/seg	394,98

Con estos resultados se puede indicar que para 100 usuarios concurrentes no existen mayores inconvenientes para consultar formularios, registrar nuevos datos e incluso consultar una aplicación externa integrada al sistema como son los reportes, esto es algo bueno ya que aproximadamente esa será la cantidad máxima de usuarios que usen el sistema al mismo tiempo. Con las pruebas de 500 se pudo probar los límites de carga, probando que tan bueno es el rendimiento ante el aumento del número de usuarios que usan el sistema al mismo tiempo, obteniendo buenos resultados y porcentajes aceptables para esta cantidad. Mientras que con las pruebas de 1000 se pudo comprobar el comportamiento ante una sobrecarga hecha al servidor, obteniendo que alrededor de la mitad de los usuarios no tienen inconvenientes en realizar transacciones en el sistema.

Esto indica que en líneas generales el comportamiento del sistema cumple con las expectativas de robustez y rendimiento, incluso ante una sobrecarga.

8.2 Resultados de profiling

En la tabla 6 se muestra un resumen de las instancias totales para ambas cargas. Se puede apreciar que las instancias totales para clases y para paquetes se incrementan a medida que aumentan las transacciones de los usuarios, mientras que las instancias de componentes se mantienen iguales:

Tabla 6. Instancias totales

	Carga baja de transacciones	Carga alta de transacciones
Instancias totales de clases	7539	31419
Instancias totales de componentes	4	4
Instancias totales de paquetes	7518	31255

En la tabla 7 se muestra un resumen de los tamaños totales para ambas cargas. Existe un aumento notable en el tamaño de clases y paquetes a medida que aumentaban las transacciones de usuarios, esto se debe a que las instancias totales también aumentaron en cantidad. Además, el tamaño de componentes se mantiene igual.

Tabla 7. Tamaños totales

	Carga baja de transacciones	Carga alta de transacciones
Tamaño total de clases	317 kB	1304 kB
Tamaño total de componentes	160 bytes	160 bytes
Tamaño total de paquetes	316 bytes	1318 bytes

JProfiler muestra de manera gráfica como se está realizando la carga de CPU. Mediante esta telemetría se puede observar que para carga baja los picos llegan a un máximo de 8%, tal como se lo muestra en la figura 4. A medida que se aumenta a una carga alta la mayor parte se mantiene por debajo de 30% y los picos llegan a un máximo de 65%, tal como se lo muestra en la figura 5, lo cual es un considerable aumento pero es entendible que resulte esto, debido al incremento de muestras que se están realizando.

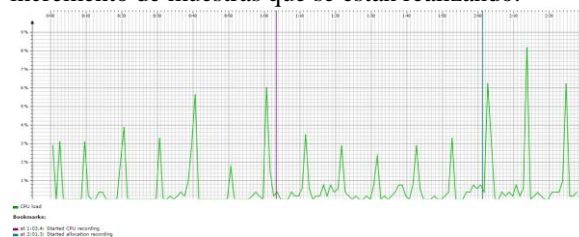


Figura 4. Telemetría de carga de CPU durante carga baja

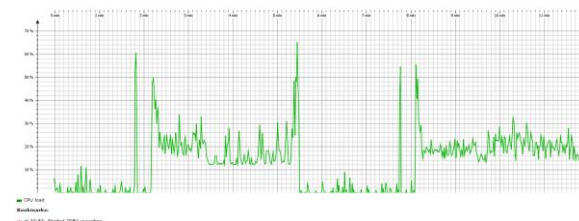


Figura 5. Telemetría de carga de CPU durante carga alta

JProfiler también muestra la telemetría para mostrar la memoria usada y la memoria libre. En la figura 6 se puede notar que mientras la carga es baja existe una gran cantidad de memoria libre y por lo general la memoria usada se mantiene por debajo de 300 MB. Al aumentar la carga se puede apreciar en la figura 7 que existe poca memoria libre en comparación a la carga baja, aunque la usada se mantiene por debajo de 500 MB.

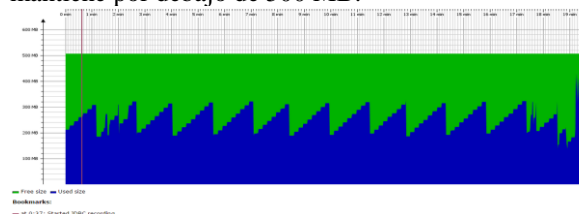


Figura 6. Telemetría de uso de memoria durante carga baja

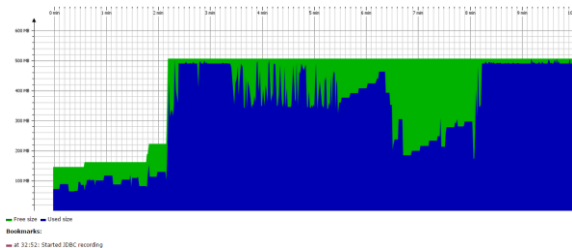


Figura 7. Telemetría de uso de memoria durante carga alta

Mediante estas pruebas se puede notar que a medida que se aumentan los usuarios también aumentan los recursos consumidos del sistema. Además se puede notar que existe un gran consumo de CPU y memoria, por lo que se puede optimizar el sistema para que estos porcentajes bajen y se muestren valores que contribuyan a la mejor utilización de recursos. Esto se puede obtener al analizar los formularios del sistema y tratar optimizar el uso de conexiones a base de datos, carga de imágenes e implementación de códigos JavaScript que han sido agregados como consecuencia de requerimientos adicionales en cada uno de los formularios.

8.3 Resultados de interfaces

Se realizaron las siguientes pruebas:

TDS_01: Registrar Pescador

- TC_L1_01: Ingresar los siguientes datos correctos del pescador: nombres, apellidos, cédula, número de hijos del pescador, isla de origen y número de residencia.
- TC_L1_02: Ingresar los siguientes datos correctos del pescador: nombres y apellidos; ingresar los siguientes datos incorrectamente: cédula, número de hijos del pescador. No ingresar la isla de origen y el número de residencia.
- TC_L1_03: Ingresar los siguientes datos correctos del pescador: cédula, número de hijos del pescador; ingresar los siguientes datos incorrectamente: nombres y apellidos. Ingresar la isla de origen y el número de residencia.

Dando los siguientes resultados:

Tabla 8. Resultado de pruebas realizadas

<i>Pruebas</i>	Resultados Esperados	Resultados Obtenidos
<i>TC_L1_01</i>	Formulario Enviado	Se obtienen los resultados esperados.
<i>TC_L1_02</i>	Formulario No Enviado	Se obtienen los resultados esperados.
<i>TC_L1_03</i>	Formulario	Se obtienen los

	No Enviado	resultados esperados.
--	------------	-----------------------

9. Conclusiones

Este proyecto de graduación tuvo su origen en la necesidad de implementar los procesos de Conservación y Uso de Ecosistemas Marinos, de la Dirección de Ecosistemas del Parque Nacional Galápagos, sobre un Workflow.

Con el trabajo realizado en este proyecto podemos concluir lo siguiente:

1. Bizagi modeler es una herramienta con una curva de aprendizaje corta e interfaz intuitiva; implementa BPMN 2.0, lo cual nos facilitó el modelamiento de procesos. Se tuvo una media de 1 día en el modelamiento del proceso en base a las entrevistas realizadas.
2. Bizagi Modeler permite exportar el modelo del proceso en varios tipos de archivos como: XPDL, .jpg; proporcionando un medio para compartir el mismo con otros desarrolladores. También la imagen generada la pudimos incluir en la documentación de los procesos que entregaremos a la organización.
3. El modelamiento de los procesos en base a los roles que intervienen en los mismos, facilita la implementación en la herramienta Workflow Joget al asociar los usuarios al rol que pertenece y otorgar o negar permisos en acciones según los roles. Esta característica permite adaptarse a las necesidades de la organización.
4. La herramienta Joget nos permite el rápido diseño de componentes dentro de un Workflow como: formularios, listas, vistas de usuarios, y modelos de procesos de ejecución. Brindándonos un mantenimiento ágil de los procesos para la organización.
5. El motor de procesos de Joget nos brinda flexibilidad; al implementar reglas del negocio cambiantes en el tiempo permitiendo la optimización de recursos ya que no involucra grandes inversiones de tiempo en el cambio de los procesos.
6. El uso de Pentaho Reports fue importante para la mejorar la visualización de datos almacenados, mediante el uso de gráficos, tablas y parámetros dinámicos, que fueron requeridos por los usuarios de los diferentes procesos. Además su plataforma web facilitó la integración al sistema.
7. Las pruebas de carga permitieron analizar el rendimiento del sistema mediante la simulación de varias transacciones. En estas pruebas se comprobó que no existen inconvenientes para que 100 usuarios puedan consultar y registrar información al mismo tiempo. Además de cumplir con la robustez y el rendimiento

esperados ante una cantidad de usuarios mayores a los que usarán el sistema, como 500 y 1000.

- Las pruebas de profiling permitieron analizar el rendimiento que tiene el sistema, tomando en cuenta el consumo de recursos. Se pudo comprobar que cuando se aumentan los usuarios, aumentan los recursos consumidos del sistema; también la existencia de un gran consumo de CPU y memoria que puede ser optimizado al analizar los formularios que han sido modificados con conexiones a base de datos, carga de imágenes e implementación de códigos JavaScript.

10. Agradecimientos

A la Dirección del Parque Nacional Galápagos, por su colaboración en la elaboración de este proyecto.

A nuestro director del proyecto de graduación, M.Sc. Carlos Mera por su guía y conocimientos compartidos.

Al Ing. Francisco Ramírez, por su apoyo y tiempo con nuestras inquietudes.

11. Referencias

- Das M., Deb M. y Wilkins M, “Manual Oracle Business Process Management Suite 11g Handbook”, URL: http://cdn.tgtmedia.com/searchOracle/downloads/449-0_ch01.pdf, fecha de consulta: Enero del 2015.
- Workflow Management Coalition, “What is BPM?”, URL: <http://www.wfmc.org/what-is-bpm>, fecha de consulta: Enero del 2015.
- Object Management Group, “What is BPMN?”, URL: <http://www.bpmn.org/>, fecha de consulta: Enero del 2015.
- Object Management Group, “Business Process Model and Notation”, URL: <http://www.omg.org/spec/BPMN/2.0/>, fecha de consulta: Enero del 2015.
- Bizagi Process Model, “Business Process Modeling Notation”, URL: <http://www.bizagi.com/esp/descargas/BPMNbyExample.pdf>, fecha de consulta: Enero del 2015.
- Camunda, “BPMN 2.0 Best Practices”, URL: <http://camunda.org/bpmn/examples/>, fecha de consulta: Enero del 2015.
- Bizagi, “BPM Suite”, URL: <http://www.bizagi.com/es/bpm-suite-es>, fecha de consulta: Enero del 2015.
- Zeinelbdeen F., Salih O., Elhafiz M., “Comparison of The Workflow Management Systems Bizagi, ProcessMaker, and Joget”, The International Arab Conference on Information Technology, 2013.
- BonitaSoft, “Bonita BPM”, URL: <http://www.bonitasoft.com/how-we-do-it>, fecha de consulta: Enero del 2015.
- WfMC, “Workflow”, URL: http://glossary.businessprocessincubator.com/index.php/bpifeed_ask_similar?qanda-popup=1&qanda-embedded=1&qanda-show-my-qs=0&qanda-bpifeed=1, fecha de consulta: Enero del 2015.
- Joget, “Joget Workflow Architecture”, URL: <http://www.joget.org/bpm-software>, fecha de consulta: Enero del 2015.
- Manageability, “Open Source Workflow Engines Written in Java”, URL: http://www.manageability.org/blog/stuff/workflow_in_java/view, fecha de consulta: Enero del 2015.
- BonitaSoft, “Bonita BPM Engine architecture”, URL: <http://documentation.bonitasoft.com/bonita-bpm-engine-architecture>, fecha de consulta: Enero del 2015.
- InfoWorld, “Bossie Awards 2013: The best open source applications”, URL: <http://www.infoworld.com/article/2606353/open-source-software/119652-Bossie-Awards-2013-The-best-open-source-applications.html#slide23>, fecha de consulta: Enero del 2015.
- Activiti, “Activiti Components”, URL: <http://activiti.org/components.html>, fecha de consulta: Enero del 2015.
- Calegari D., Delgado A., “Aspectos claves para la construcción de software de soporte e impacto en la mejora continua de las organizaciones”, Grupo COAL, Instituto de Computación, Facultad de Ingeniería, Universidad de la República de Uruguay, 2014.
- Sommerville I., “Software Engineering”, Addison-Wesley, Ninth Edition, 2011
- Sommerville I., “Software Engineering”, Addison-Wesley, Ninth Edition, 2011
- Meier J.D., Farre C., Bansode P., Barber S., Rae D., “Performance Testing Guidance for Web Applications”, Microsoft Corporation, 2007.
- Apache JMeter, “Apache JMeter Overview”, URL: <http://jmeter.apache.org/index.html>, fecha de consulta: Marzo del 2015.
- JProfiler, “JProfiler Overview”, URL: <https://www.ej-technologies.com/products/jprofiler/overview.html>, fecha de consulta: Marzo del 2015.
- Russo J., “Tutorial Herramientas de Performance”, Universidad ORT Uruguay, 2006.