

ANÁLISIS, DESARROLLO E IMPLEMENTACIÓN DE UN SOFTWARE INTRANET QUE PERMITA OPTIMIZAR LA PRODUCCIÓN DE LOS SERVICIOS QUE REALIZA EL C.A.L.E

Rolando Orquera Quirola, Prócel Silva Carlos, Fernando Solís Acosta

Departamento de Ciencias de la Computación, Universidad de las Fuerzas Armadas ESPE, Ecuador

rolandoOrquera@hotmail.com, {ctprocel, fesolis}@espe.edu.ec

RESUMEN

El presente proyecto ha sido diseñado para controlar, administrar e informar acerca del proceso para mantenimiento de los recursos tecnológicos militares que llegan al comando de apoyo logístico electrónico desde las diferentes unidades militares a nivel nacional, con el fin de realizar un trabajo adecuado por parte de los técnicos encargados. El sistema está desarrollado en un ambiente web basado en el lenguaje de programación Java sobre la plataforma JEE6 con una arquitectura distribuida a n – capas. Las herramientas tecnológicas utilizadas son Primefaces 3.5, servidor de aplicaciones JBOSS 6.1.0, HTML5, CSS 3, jQuery – ui – 1.10.3, base de datos MySQL 5.0 y Ajax. Para el análisis, desarrollo e implementación del software se ha tomado a la programación extrema como metodología a seguir. En todas las etapas de desarrollo del software se obtuvo resultados rápidos y satisfactorios, cubriendo todas las necesidades del cliente gracias a que ha sido involucrado activamente durante todo el proceso del proyecto. Finalmente se obtuvo un mejor control, orden y eficiencia en la logística que tiene el C.A.L.E producto de un ágil acceso a la información, registro de etapas de mantenimiento, generación de estadísticas y gráficas además de un seguimiento de actividades acorde al trabajo realizado en laboratorio.

Palabras clave: jQuery – ui – 1.10.3, Ajax, arquitectura n – capas, Primefaces 3.5.

ABSTRACT

This project has been designed to control, manage, and inform about the process for giving maintenance of the technological military resources that arrives to the electronic logistic support facility from the different military units around the country, in order of getting better results from the technicians in charge. The system has been developed within a web environment based on Java which works on the platform JEE6, it's distributed architecture has N-tier, The technological tools used, such as Primeface 3.5, also, an application server with JBOSS 6.1.0, EJB 3.0, HTML5, CSS 3, jQuery – ui – 1.10.3, a MySQL 5.0 database and Ajax. For the analysis, development, and implementation of this software, we have used a methodology called Extreme Programming (XP). In all these stages we obtained quick results and more important, satisfactory results, meeting all the needs of the client thanks to his active participation inside the whole project. Lastly, we obtained a better control, efficiency and better management in the logistic of the L.E.S.C (Logistic Electronic Support Commando), as a result of an agile access to the information, a registry of the stages of maintenance, the creation of statistics, and graphics, besides a tracing of all the activities of the work made within the laboratory

KeyWords: jQuery – ui – 1.10.3, Ajax, n – tier architecture, Primefaces 3.5

1. INTRODUCCIÓN

El comando de apoyo logístico electrónico (C.A.L.E), está encargado de brindar un servicio exclusivo de soporte técnico a las distintas unidades militares del Ecuador, actualmente no cuenta con un sistema capaz de llevar una administración apropiada de los equipos en mantenimiento, obligando al personal encargado a utilizar herramientas ofimáticas, para llevar un registro de todo el proceso dando como resultado pérdida de información, extravío de equipos, falta de control y un déficit en la confidencialidad de los recursos tecnológicos militares.

Por las razones anteriores es necesario implementar un software capaz de adaptarse al proceso de mantenimiento de los recursos tecnológicos militares que brinde información y seguridades apropiadas además de una correcta organización, dando como resultado un trabajo más ágil con un rápido acceso a los registros desde cualquier computador dentro de la red del C.A.L.E basados en los roles de usuario.

Con la implementación del sistema el consumo de papeles se ha reducido considerablemente gracias a la utilización de un gestor de base de datos y las características web de la aplicación, por lo cual la información siempre estará segura y a la mano del personal responsable.

El resto del artículo se ha estructurado de la siguiente manera: La sección 2 describe la metodología utilizada para la elaboración del proyecto. La sección 3 detalla las herramientas y métodos utilizados en la construcción del sistema. La sección 4 muestra el diseño e implementación de software. La sección 5 muestra los resultados obtenidos basados en la solución planteada y la aceptación del usuario. La sección 6 indica los trabajos relacionados a la aplicación propuesta. Finalmente, la sección 7 se presenta las conclusiones derivadas del análisis realizado y sus proyecciones a futuro.

2. METODOLOGÍA

En la realización del proyecto se ha utilizado como metodología a la programación extrema, debido a que es un proyecto a corto plazo, el equipo de trabajo es bastante limitado y además por los cambios pedidos sobre la marcha basada en el análisis de impacto del proyecto. Por estas razones se necesita una metodología ágil, abierta al cambio, que integre al cliente directamente con el proyecto y que no sea dependiente de una extensa documentación.

XP ha sido diseñado para aplicarse a proyectos que requieren tiempos cortos de desarrollo y cuentan con equipos pequeños para su implementación. (Arrieta, 2002)

Esta metodología involucra al cliente desde las primeras fases del proyecto hasta la finalización del mismo, permitiendo establecer reuniones periódicas en las cuales se tratan temas puntuales que permiten analizar cambios para dar soluciones basadas en la simplicidad sin dejar de lado los objetivos de cada requerimiento. (Kent, 2000)

Los valores que utiliza la programación extrema dentro de su filosofía son:

- **Comunicación:** es la constante comunicación entre el equipo de trabajo y el cliente, esto genera resultados rápidos y satisfactorios en la entrega del producto.
- **Simplicidad:** todos los requerimientos deben analizarse hasta llegar a una solución sencilla tanto en código como en diseño.
- **Retroalimentación:** se refiere al compromiso del cliente a integrarse activamente en el desarrollo del proyecto programando reuniones periódicas.
- **Valentía:** en este punto el desarrollador debe estar dispuesto a realizar cambios aunque esto signifique borrar parcial o totalmente código que haya demandado un gran esfuerzo. (Bolivariana, 2014)

La Figura 1 muestra cómo se lleva a cabo el proceso de la programación extrema.

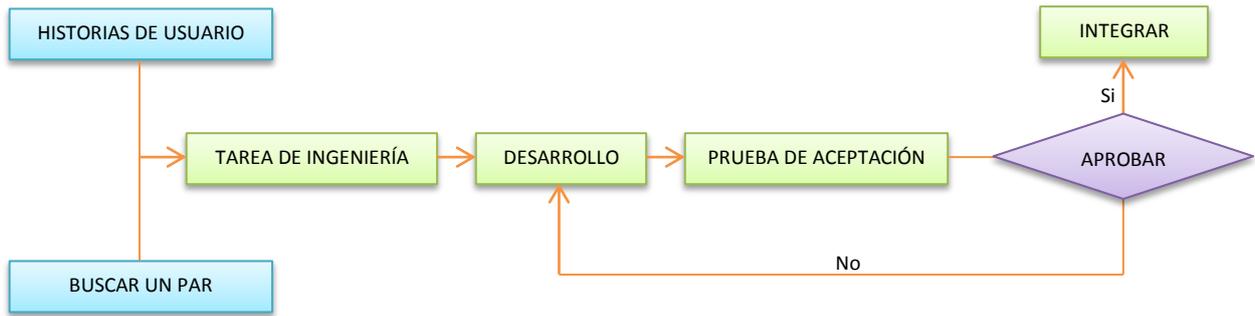


Figura 1: Proceso de la Metodología XP

2.1 Historias de usuario

En cada reunión el cliente puede realizar nuevas peticiones, las cuales son analizadas para brindar una solución sencilla que será documentada para su posterior implementación, a esto se conoce como historia de usuario.

No todas las historias de usuario son aplicables, solo se tomarán en cuenta aquellas que no afecten al core del sistema.

2.2 Buscar un par

Se refiere a encontrar una pareja de desarrolladores para que se encarguen de la programación del requerimiento analizado.

2.3 Tarea de ingeniería

En esta etapa se revisa cada historia de usuario para elaborar una o varias especificaciones funcionales debidamente documentadas para su posterior desarrollo.

2.4 Desarrollo

Se refiere a la entrega de las tareas de ingeniería a la pareja responsable de su desarrollo. Es importante recalcar que el cumplimiento de las mismas debe realizarse en los tiempos especificados para que no existan retrasos inesperados.

2.5 Prueba de aceptación

Es el proceso en el cual el usuario realiza pruebas sobre el cambio pedido con el fin de verificar el cumplimiento del mismo.

Estas pruebas serán documentadas especificando si la evaluación ha sido positiva o negativa para su respectiva integración o rectificación.

2.6 Integrar

Una vez que el cliente apruebe el cambio realizado, se lo integrará al sistema como parte de los requerimientos. Posteriormente se continuará con el desarrollo normal del proyecto.

(Sebastián, 2008)

3. HERRAMIENTAS Y MÉTODOS UTILIZADOS PARA LA CONSTRUCCIÓN DEL SISTEMA.

La Tabla 1 muestra todas las herramientas que han sido utilizadas para el desarrollo del presente proyecto la cual será presentada a continuación:

Herramienta	Descripción
Base de datos MySQL 5.0	Gestor de base de datos de gran aceptación que permite administrar toda la información de forma sencilla. Esta base de datos está enfocada para proyectos pequeños y medianos. En la actualidad existe mucha documentación y puede ser integrado en varios lenguajes de programación. (corporation, 2014)
Primefaces 3.5	Es un componente para la tecnología java server faces que brinda varias herramientas integradas para un desarrollo rápido de aplicaciones orientadas a la web.
JBOSS 6.1.0	Servidor de aplicaciones muy robusto multiplataforma desarrollado en lenguaje java. Su función es desplegar aplicaciones distribuidas orientadas a la web para su correcta implementación.
HTML5	Es el último estándar para HTML, fue diseñado para implementación de elementos multimedia sin necesidad de plugins adicionales.
CSS 3	Herramienta tecnológica que permite separar el diseño de una página web de la estructura principal. (w3schools, 2014)
jQuery – ui – 1.10.3	Herramienta basada en el lenguaje JavaScript utilizada en la capa de presentación para generar animaciones validaciones e implementar componentes gráficos calendarios, charts, botones, menus, etc.
Ajax	Es una técnica para enviar datos al servidor y recibir una respuesta de forma asíncrona, es decir sin necesidad de refrescar la página web. Ajax es el resultado del uso de tecnologías como DOM, html, xml, JavaScript, xmlhttp. (Ajax Ya, 2013)

Tabla 1: Herramientas tecnológicas

3.1 Método para desarrollo de requerimientos.

Se ha tomado el método de prototipo para la codificación de los requerimientos de usuario.

Este método consiste en realizar un programa totalmente funcional para ser presentado al cliente y este a su vez lo evalúe. De ser necesario se realizarán cambios sobre el modelo básico hasta obtener la solución deseada.

Con el método de prototipo la documentación del sistema es una actividad que se puede pasar por alto, lo importante es realizar un análisis correcto para dar soluciones rápidas y efectivas.

La Figura 2 muestra el proceso a seguir para la aplicación de este método.

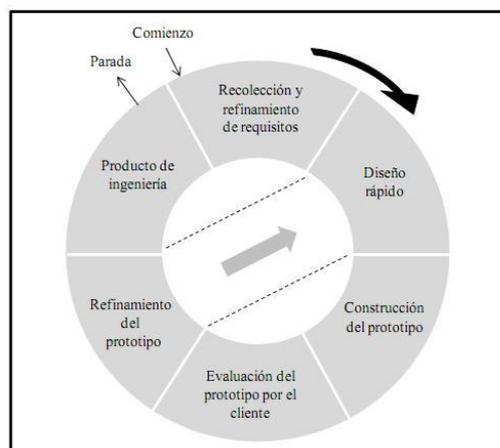


Figura 2: proceso del método de prototipo (Roger, 2009)

Para este caso se ha codificado un sistema piloto el cual se lo ha mejorado a lo largo del desarrollo del software hasta obtener una solución apropiada con la cual el cliente quede satisfecho.

4. DISEÑO E IMPLEMENTACIÓN

El proyecto está basado en un modelo vista controlador, es decir se han separado las capas de presentación, negocio y datos en diferentes niveles. La Figura 3 muestra la distribución del sistema.

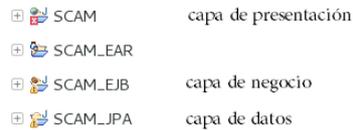


Figura 3: Distribución de las capas del sistema

4.1 Capa de Presentación

En este módulo se generan todos los backingbean necesarios para la lógica de las interfaces gráficas tal como lo muestra la Figura 4.

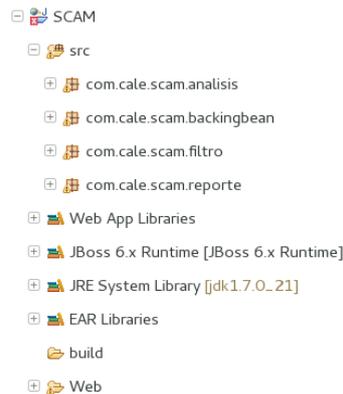


Figura 4: distribución de paquetes en la capa de presentación.

Se puede observar que las clases se encuentran encapsuladas en paquetes, en la Figura 5 se visualizará los niveles de la capa de presentación para el paquete com.cale.scam.backingbean.

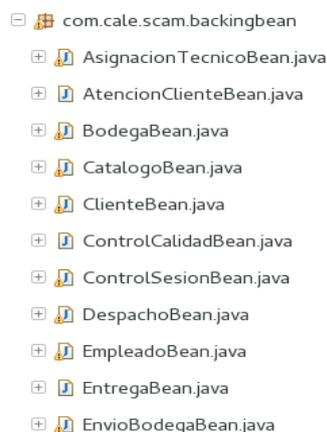


Figura 5: visualización de las capas del paquete com.cale.scam.backingbean

4.2 Capa de Negocio

Esta capa contiene todas las clases que contienen la lógica de negocio, es decir el lenguaje de manipulación de datos adaptado a los requerimientos del usuario. La Figura 6 muestra la distribución de la capa de negocio.

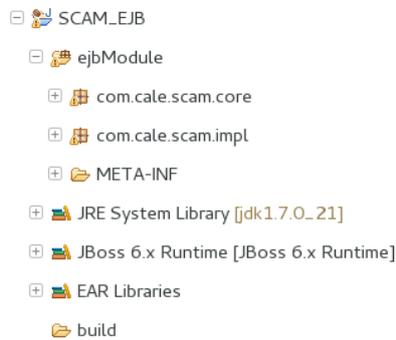


Figura 6: Distribución de la capa de negocio.

Los paquetes `com.cale.scam.core` y `com.cale.scam.impl` contienen las interfaces y sus implementaciones respectivamente tal como se muestra en la Figura 7.

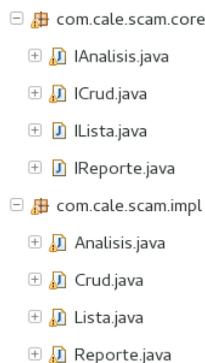


Figura 7: clases que representan la capa de negocio

4.3 Capa de Base de Datos

Esta capa encapsula el enmascaramiento de las entidades de la base de datos en un tipo de clases especiales llamadas “pojos” que permiten trabajar con la información de la base de datos. La Figura 8 muestra los pojos creados en el paquete `com.cale.scam.modelo` para la capa de datos.

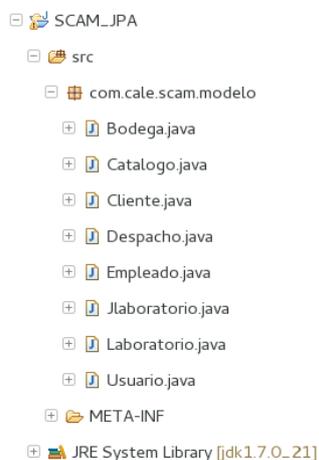


Figura 8: Enmascaramiento de pojos

El diseño completo de la arquitectura se muestra en la Figura 9.

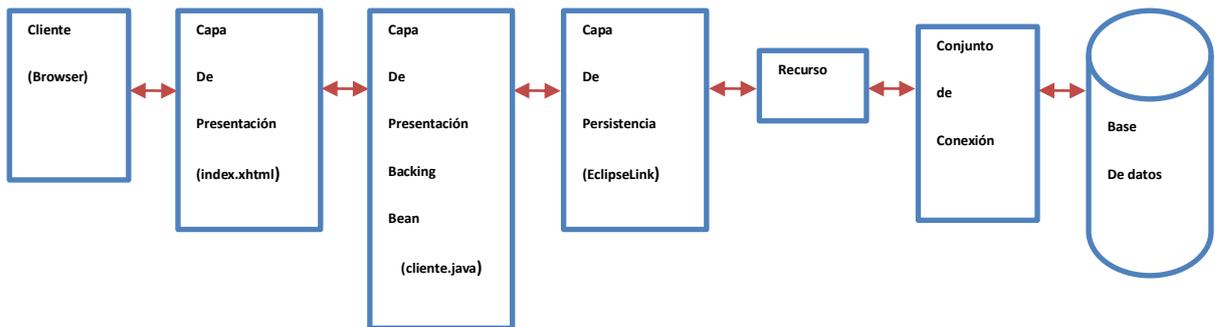


Figura 9: Arquitectura del modelo MVC (modelo vista controlador)

Finalmente se podrá apreciar el resultado obtenido al aplicar correctamente los conceptos detallados anteriormente en la Figura 10.



Figura 10: Implementación del sistema de control y administración militar

5. RESULTADOS

Se ha evaluado los resultados en base a lo siguiente:

- Registrar y verificar la etapa de mantenimiento en la que se encuentra el recurso tecnológico militar, este proceso se muestra en la Figura 11 y la Figura 12 respectivamente.

Código de registro en laboratorio
lab1

Tipo de mantenimiento
 Preventivo Correctivo Predictivo

Observación
Reparación de ventiladores

Etapa de mantenimiento
Detección de avería

Técnico Responsable: Raul Reyes

Figura 11: Registro

Búsqueda de la etapa de mantenimiento de recursos tecnológicos militares

Laboratorio

Código de laboratorio: lab1

Código de registro: reg1

Etapa de mantenimiento: **deteccion de averia**

Código recurso tecnológico militar: B045010704

Recurso tecnológico militar: AURICULAR Y MICRÓFONO TADIRAN CON CORDÓN H-140

Nombre jefe de laboratorio: Jorge

Apellido jefe de laboratorio: Ortiz

Laboratorio: Motorola

Fecha de ingreso a laboratorio: 2014-05-02

Fecha de salida de laboratorio:

Figura 12: Verificación de la etapa de mantenimiento

- Generar estadísticas y gráficas para control de producción. La Figura 13 muestra la gráfica de producción de cada laboratorio y la Figura 14 detalla la estadística del proceso generado.

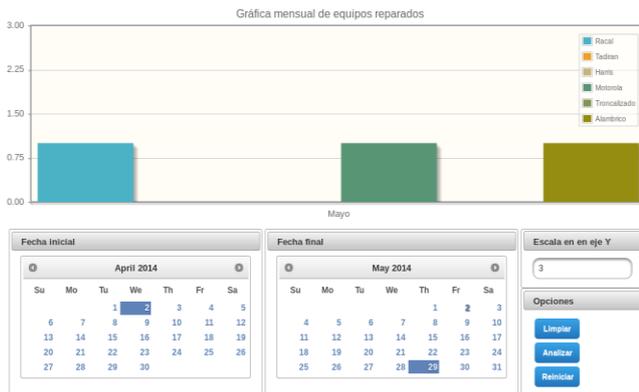


Figura 13: Gráficas de producción mensual variable

Número de equipos	Laboratorio	Fecha
1	Alambrico	2014-05-11
1	Motorola	2014-05-02
1	Racal	2014-05-12

Figura 14: Estadística de producción mensual

- Realizar un seguimiento a los técnicos de los laboratorios del C.A.L.E. La Figura 15 muestra una tabla con varios criterios que determinan el trabajo realizado por los técnicos.

Etapa de mantenimiento	Nombre del técnico asignado	Apellido del técnico asignado	Reparado
deteccion de averia	Raul	Reyes	no
ninguna	Raul	Reyes	no
ninguna	Raul	Reyes	no

Figura 15: detalle de trabajo realizado por técnicos con filtros de información por cada criterio

- Registrar los recursos tecnológicos militares ingresados. La Figura 16 muestra el proceso de registro para recursos tecnológicos militares ingresados a bodega y en la Figura 17 se puede observar el detalle del registro realizado.

Código de registro
reg2

Recurso tecnológico militar
B045010810 - BAJADA COAXIAL DE ANTENA 20M 710

Cliente
cli1 - Rolando Orquera

Usuario que realiza el registro
1234 - Camilo Sesto

Fecha de ingreso
May 2014

Su	Mo	Tu	We	Th	Fr	Sa
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Limpiar Buscar Editar Eliminar Registrar

Figura 16: Registro

Código de registro	Lugar	Código del recurso tecnológico militar	Nombre del recurso tecnológico
reg1	laboratorio	B045010704	AURICULAR Y MICRÓFONO TADIRAN CON CORDÓN
reg2	laboratorio	B045010810	BAJADA COAXIAL DE ANTENA 20M 710392
reg3	laboratorio	B045010810	BAJADA COAXIAL DE ANTENA 20M 710392
reg4	bodega	B045010810	BAJADA COAXIAL DE ANTENA 20M 710392
reg5	bodega	B045010704	AURICULAR Y MICRÓFONO TADIRAN CON CORDÓN

Figura 17: Detalle de registro

La Figura 18 detalla las salidas de todas las acciones realizadas mediante el historial del terminal de JBoss.

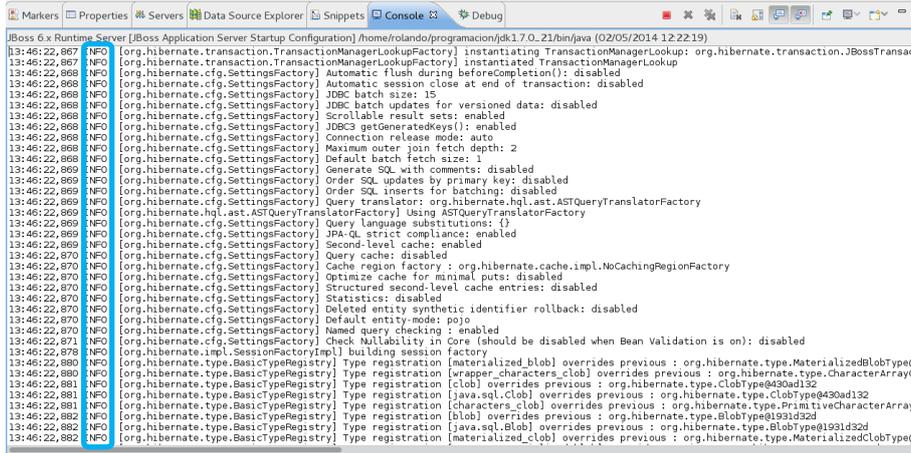


Figura 18: información de eventos realizados

Se puede observar que todas las salidas son de tipo informativo, es decir el sistema funciona correctamente.

La Tabla 2 muestra el análisis realizado en base a lo expuesto.

Objetivo	Evaluación	Prueba caja blanca	Prueba caja negra
Registrar y verificar la etapa de mantenimiento en la que se encuentra el recurso tecnológico militar	Positiva	Satisfactoria	Resultado de tipo informativo
Generar estadísticas y gráficas para control de producción	Positiva	Satisfactoria	Resultado de tipo informativo
Realizar un seguimiento a los técnicos de los laboratorios del C.A.L.E	Positiva	Satisfactoria	Resultado de tipo informativo
Registrar los recursos tecnológicos ingresados	Positiva	Satisfactoria	Resultado de tipo informativo

Tabla 2: análisis de resultados.

6. TRABAJOS RELACIONADOS

El sistema desarrollado se compone de varios conceptos y tecnologías que podríamos relacionar con otros proyectos como es el caso de la tesis titulada “Desarrollo de un sistema de gestión de bodegas” de F. Mendoza y S. Tomalá (FMendoza, 2012) que indican la importancia de llevar un control sobre los artículos administrados, de la misma forma la lógica de negocio del sistema desarrollado sigue un proceso sistemático que permite gestionar los equipos en mantenimiento correctamente. En cuanto a la arquitectura del software la tesis titulada “Análisis, diseño e implementación de una tienda virtual con tecnología J2EE” de Marisol Rodríguez Goyanes (Goyanes, 2008) posee un diseño semejante al propuesto, se puede observar que en los sistemas actuales la distribución adecuada de las lógicas genera un performance eficiente y brinda características de escalabilidad, alta cohesión y bajo acoplamiento. Finalmente en cuanto a tecnologías y metodología se puede citar a la tesis titulada “Diseño e implementación de un sistema web para la gestión y administración empresarial basado en tecnología JEE y primefaces” de Jonathan Patricio Cárdenas Ruperti (RUPERTI, 2013) como un modelo adecuado para presente proyecto por integrar una planificación, componentes y plugins similares.

7. CONCLUSIONES Y TRABAJOS A FUTURO

Con el uso adecuado de una metodología, el trabajo fue desarrollado de forma lógica y sistemática, en el caso de este proyecto, la metodología XP contribuyó a organizar una correcta planificación sobre el sistema propuesto, generando eficiencia en cuanto a peticiones de usuario y codificación. La utilización de tecnologías como Ajax, jQuery, Primefaces entre otras, aportaron soluciones más sencillas que utilizando solamente lenguaje java, ya que tienen componentes listos para ser integrados al código fuente y brindan técnicas rápidas de desarrollo para un mejor desempeño. Finalmente con la implementación del sistema, se obtuvo un mejor control, orden y eficiencia en la logística que tiene el comando de apoyo logístico electrónico para el servicio de soporte técnico, producto de un acceso ágil a la información, organización coherente de datos, visualización de datos estadísticos para equipos reparados por laboratorio y un seguimiento apropiado del cumplimiento de las etapas de mantenimiento que atraviesa cada recurso tecnológico militar antes de ser entregado al cliente.

Como trabajo a futuro se propone generar un nuevo módulo que permita realizar una evaluación de desempeño para los técnicos encargados del mantenimiento. Esta evaluación puede basarse en variables como tiempo de reparación, conocimiento tecnológico, calidad de trabajo, manejo de herramientas, etc. Cada variable se cuantificará en una escala adecuada para generar un resultado estadístico.

Referencias Bibliográficas

- Ajax Ya. (2013). Recuperado el 02 de 05 de 2014, de <http://www.ajaxya.com.ar/>
- Arrieta, S. (2002). *Prácticas extremas en ORTs*. Universidad ORT Uruguay.
- Bolivariana, U. U. (2014). *Ingeniería de software*. Recuperado el 2014, de ingenieriadesoftware.mex.tl/52753_XP---Extreme-Programing.html
- corporation, O. (2014). *MySQL*. Recuperado el 2 de 05 de 2014, de <http://www.mysql.com/>
- FMendoza, S. (2012). *dspace Espol*. Recuperado el 13 de 05 de 2014, de http://www.dspace.espol.edu.ec/bitstream/123456789/9083/1/RESUMEN_CYCIT_sistema_Gestion_bodega.pdf
- Goyanes, M. R. (14 de 01 de 2008). *Universitat Oberta de Catalunya*. Recuperado el 13 de 05 de 2014, de <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/992/1/00790tfc.pdf>
- Kent, B. (2000). *Extreme Programming explaining*.
- Roger, P. (2009). *Ingeniería de software un enfoque práctico*. Madrid: Mc Graw Hill.
- RUPERTI, J. P. (01 de 2013). *Repositorio ESPE*. Recuperado el 13 de 05 de 2014, de <http://repositorio.espe.edu.ec/xmlui/bitstream/handle/21000/6457/T-ESPE-038234.pdf?sequence=1>
- Sebastián, P. (2008). *Programación Extrema*. Recuperado el 2 de 05 de 2014, de http://www.fcad.uner.edu.ar/jai/6JAI/XP_6JAI.pdf
- w3schools. (2014). *w3schools.com*. Recuperado el 02 de 05 de 2014, de <http://www.w3schools.com/>