

**Universidad de las Ciencias Informáticas.
Facultad Regional Granma.**



Título: Set de pruebas para el proyecto Plataforma Libre para la Gestión de Procesos en la
Facultad Regional Granma versión 3.0.

Autora: Ing. Viana De la Cruz Leyva.

Ciudad de Manzanillo, junio 2012.
“Año 54 de la Revolución”.

Resumen

Construir un sistema que cumpla con las expectativas y necesidades de los clientes, se ha convertido en el mayor reto de las empresas productoras de *software* para lograr éxito mundialmente. En este grupo se incluye la Universidad de las Ciencias Informáticas (UCI), creada al calor de la batalla de ideas con la misión de contribuir a la informatización de las principales organizaciones y entidades de Cuba.

La Facultad Regional Granma de la UCI cuenta con un amplio rango de proyectos productivos de carácter nacional, dentro de estos se encuentra la Plataforma Libre para la Gestión de Procesos en la Facultad, que persigue implementar una aplicación para la gestión y organización de la información relativa a la institución. Tras un minucioso estudio del producto generado se concluyó que su calidad, dependerá en gran medida de la ejecución adecuada de un conjunto de pruebas de *software* previamente planificadas.

En este documento se propone un *set* de pruebas con el propósito de guiar el proceso en dicho proyecto cumpliendo con la necesidad imperante del tema relacionado con el aseguramiento de la calidad, del mismo se detalla: tipos de pruebas, enfocadas a las características propias del proyecto; herramientas automatizadas; recursos de *hardware* y *software* necesarios; artefactos generados; así como los involucrados en cada uno de los *test*.

PALABRAS CLAVES: *set* de pruebas, pruebas de *software*, calidad de *software*.

Índice General

Introducción	1
1. Fundamentación teórica	2
1.1. Estado actual del proceso de pruebas de <i>software</i> en la Facultad Regional Granma (FRG).	2
1.2. <i>Set</i> de pruebas.	3
1.3. Plan de pruebas.	4
2. Propuesta del <i>set</i> de pruebas	6
2.1. Descripción del sistema a probar	6
2.2. Selección de las pruebas y las herramientas de pruebas de <i>software</i> para automatizar el proceso	7
2.2.1. Explicación de las herramientas seleccionadas	8
2.2.2. Planeación de pruebas	9
3. Análisis de los resultados y valoración de la propuesta	11
3.1. Comparación de los resultados obtenidos en las pruebas realizadas a la versión 3.0 antes y después de aplicar la solución propuesta.	11
Conclusiones Generales	13
A. Encuesta	14

B. Comparación de resultados 18

Referencias bibliográficas 20

Introducción

Los diferentes campos de la ciencia y la tecnología se han perfeccionado a la par de la evolución de la humanidad, lo mismo se corrobora con la demanda de los productos de *software*. Cuba no está exenta de este proceso evolutivo, por ello el gobierno se dio a la tarea de desarrollar y perfeccionar la industria del *software* con el objetivo de crear sistemas para escalar en este mercado en el mundo. Como resultado del perfeccionamiento surge la Universidad de las Ciencias Informáticas (UCI) por idea del Comandante en Jefe Fidel Castro Ruz bajo dos propósitos claves: la informatización de la sociedad cubana y la exportación.

Dicha universidad abarca además la Facultad Regional Granma la que posee un gran cúmulo de proyectos productivos, dentro de los que está: la Plataforma Libre para la Gestión de Procesos del Departamento de Soluciones de Gestión; este proyecto, como el resto, debe ser revisado por el grupo de calidad, quien es el encargado de garantizar que los productos realizados estén acordes con las solicitudes y expectativas de los clientes. Después de una evaluación de la Plataforma Libre para la Gestión de Procesos en la Facultad Regional Granma versión 3.0, el grupo de calidad detectó un conjunto de errores que no se identificaron de forma temprana por no darle el seguimiento correspondiente, provocando desgaste de esfuerzos, tiempo y recursos, así como disgustos en los usuarios finales.

Por lo que en la presente investigación se define como **objetivo general** proponer un *set* de pruebas para el aseguramiento de la calidad de *software* en el proyecto Plataforma Libre para la Gestión de Procesos en la Facultad Regional Granma versión 3.0.

Capítulo 1

Fundamentación teórica

Introducción

En este capítulo se definen algunos conceptos necesarios para mayor comprensión de la investigación.

1.1. Estado actual del proceso de pruebas de *software* en la Facultad Regional Granma (FRG).

En la FRG para realizar el proceso de pruebas, auditorías y revisiones, se parte de la planificación de la actividad; posteriormente se analiza en que etapa de desarrollo se encuentra el producto, pues en dependencia de ello se realizan las revisiones, las pruebas (funcionales, de carga y estrés generalmente) o las dos de forma simultánea, en caso de que el proyecto no esté en condiciones se aborta la actividad, de lo contrario, le debe facilitar al grupo de calidad todos los documentos necesarios para proceder a una evaluación; por último se informan las no conformidades detectadas durante todo el proceso y se emite una nota del equipo revisor. En caso de que el proceso que se esté realizando sea una auditoría, al concluir se da a conocer la evaluación final del proyecto objeto de análisis.

1.2. Set de pruebas.

Para poder conformar un concepto completo sobre *set* de pruebas, primeramente se debe tener presente, que un *set*, es un conjunto de varios elementos que tienen en común la función que realizan; en este caso particular esos elementos a los que se hace mención serían las pruebas de *software* y la función correspondería al análisis de los errores de una aplicación. Existen varias ideas con respecto a los *sets* de prueba, aunque ninguna de ellas hace referencia a un autor específico. A continuación se listan algunas:

- “Una *suite* de prueba contiene: un identificador único; responsables; referencia al documento(s) de requerimiento de prueba, fuente de la *suite*; la configuración de *software* a probar; referencia a los casos de prueba que conforman la *suite*; requerimientos sobre el ambiente de prueba (especificación de la configuración del *hardware* en que se ejecutará la prueba, especificación del *software* de sistema y de apoyo a la prueba, especificación de recursos especiales requeridos, especificación del procedimiento de prueba, etc) y por último, las precondiciones”. [1]
- “Las *suites* de prueba (*test suites*) permiten ejecutar de manera colectiva casos, métodos y/o *suites* de pruebas relacionados”. [2]
- “Una *suite* corresponde a un conjunto de *test* que se realizan de manera secuencial”. [3]

Aunque ninguna de las ideas es errónea, no abarcan por sí sola todos los elementos necesarios para establecer lo que es un *set* de pruebas, por ello, en la presente investigación se define como una colección de pruebas que pueden realizarse de forma secuencial a una aplicación determinada a lo largo de su ciclo de desarrollo, este recoge aspectos como: el modo en que las mismas deben ser realizadas, las herramientas que se proponen, así como los requisitos de *hardware* y *software* necesarios en dependencia del tipo de aplicación.

Los *sets* de pruebas exigen la elaboración de un plan de pruebas que permita señalar los elementos

que se desean probar en dependencia de las características específicas de la etapa de desarrollo del producto.

1.3. Plan de pruebas.

El plan de pruebas puede ser aplicado en cualquier *release*¹ del proyecto, el mismo está conformado por un determinado número de pruebas que no debe coincidir obligatoriamente con las indicadas en el *set* al que pertenece. Los elementos que nunca deben faltar a la hora de confeccionar este documento son:

1. Introducción

a) Propósito

Se deben especificar los objetivos.

b) Contexto

Se debe escribir una breve descripción del objetivo de la prueba (componentes, aplicaciones, sistemas, etc), incluyendo además, principales funcionalidades, arquitectura e historia del proyecto.

c) Alcance

Se describen los niveles de prueba. Se provee una lista con cualquier hecho durante el desarrollo del documento que pueda impactar en el diseño, desarrollo o implementación de las pruebas.

d) Identificación del proyecto

En este acápite se identifica la documentación disponible para desarrollar el plan de pruebas, la información se puede recoger en forma de tabla.

¹Nueva versión de una aplicación informática.

2. **Requerimientos de prueba**

Se identifican los elementos (casos de uso, requerimientos funcionales y requerimientos no funcionales) que serán probados.

3. **Estrategia de prueba**

Para cada tipo de prueba se debe proporcionar una descripción y el porqué está siendo implementada y ejecutada. En caso de que alguna no se implemente se indicará la justificación. También se listarán las herramientas a utilizar, especificando la versión.

4. **Recursos**

Esta sección representa los recursos que se recomiendan utilizar.

a) **Roles**

Detallar los recursos de personal.

b) **Sistema**

Detallar los recursos de sistema.

5. **Entregables**

Se enumeran los documentos, las herramientas y los informes que serán creados.

Capítulo 2

Propuesta del *set* de pruebas

Introducción

En el presente capítulo se describe el *set* de pruebas que se propone aplicar a la Plataforma Libre para la Gestión de Procesos en la Facultad Regional Granma versión 3.0.

2.1. Descripción del sistema a probar

El proyecto Plataforma Libre para la Gestión de Procesos en la Facultad Regional Granma surge debido a la ausencia de un sistema automatizado para realizar la gestión del caudal de información en formato duro que contiene el centro, además resultaba necesario y de vital importancia recoger y centralizar esta información para su posterior uso y explotación.

El sistema puede accederse a través de la *web* y la metodología escogida para su desarrollo es SXP. Todas las herramientas de desarrollo utilizadas son libres (Ver Figura 2.1).

Herramientas	
Sistema Gestor de Bases de Datos	Postgre SQL 8.3.
IDE de desarrollo	Netbeans
Framework de desarrollo	Django
Framework de Interfaz de usuario	JQuery
Herramienta de Modelado	Power Architect, Visual Paradigm
Herramienta para Versionado	Bazaar
Herramienta de Gestión de Proyecto	Redmine
Sistema para Seguimiento de Errores	Django

Figura 2.1: Herramientas utilizadas por el proyecto Plataforma Libre para la Gestión de Procesos

2.2. Selección de las pruebas y las herramientas de pruebas de *software* para automatizar el proceso

Con el propósito fundamental de evaluar el conocimiento que existe en cuanto al proceso de pruebas de *software* y las herramientas automatizadas por parte de los integrantes del proyecto Plataforma Libre para la Gestión de Procesos en la Facultad Regional Granma, se llevó a cabo una encuesta, la misma consta de 4 preguntas previamente elaboradas (Ver Anexo A) y los encuestados suman un total de 8 personas seleccionadas de forma aleatoria.



Figura 2.2: Distribución de los encuestados

Una vez concluida la encuesta, el resultado arrojado fue que a pesar de haber conocimiento sobre la existencia de las pruebas de *software*, no se ha ampliado el espectro hacia la

variedad de las mismas, este mismo comportamiento fue evidenciado en el caso de las herramientas y a pesar del desconocimiento se aboga porque sean utilizadas sabiendo que agilizarían exponencialmente el trabajo.

Para seleccionar las herramientas se hizo mayor énfasis en las libres y de código abierto, atendiendo a que el país y la Facultad Regional Granma abogan por esta alternativa, pues el ahorro económico que representa es notable. Otro de los aspectos que se tuvo en cuenta fue el nivel de experiencia sobre el uso de algunas herramientas en el centro, valorándose cuán eficiente son con respecto a las demás para analizar si podrían seguir utilizándose, ahorrando tiempo a la hora de revisar el producto, pues las capacitaciones serían mínimas.

2.2.1. Explicación de las herramientas seleccionadas

JMeter

Es una de las herramientas que según los resultados arrojados por la encuesta es más conocida por parte de los integrantes del proyecto Plataforma Libre para la Gestión de Procesos en la Facultad, cuestión que sumada a las ventajas que la misma tiene con respecto a otras herramientas de carga la convierte en la más idónea. Entre sus ventajas se pueden mencionar las siguientes:

- a) “De las herramientas gratis, es la más completa y útil para el tipo de pruebas en cuestión”. [4]
- b) “Brinda mayor cantidad de variantes para recoger los resultados obtenidos que el resto de las herramientas gratis, lo que permiten hacer un análisis exhaustivo de las pruebas realizadas” [5], aunque debe aclararse que los gráficos no son muy fáciles de entender.
- c) “Necesita muy pocos requerimientos de sistema (aunque estos aumenten a medida que aumentan la cantidad de VU – Virtual Users-)”. [5]

d) “Se cuenta con bastante documentación”. [5]

Selenium IDE

Selenium, es de todas las herramientas analizadas durante la encuesta la más conocida por los integrantes del proyecto, convirtiéndola por consiguiente en la mayor candidata dentro de las herramientas para pruebas funcionales. “...Es una herramienta simple, potente y flexible, además de que provee un lenguaje fácil de usar y de aprender. Una de las ventajas de Selenium es que es multinavegador...” [6], facilitando así que los resultados visuales obtenidos sean iguales y que las funcionalidades no cambien. Además, con la misma se pueden grabar las acciones del usuario en la mayoría de los lenguajes populares como Java, C, Perl, entre otros, lo que elimina la necesidad de aprender un nuevo lenguaje de programación.

2.2.2. Planeación de pruebas

Según los resultados arrojados una vez finalizada la encuesta, el desconocimiento sobre las pruebas de *software* es relevante, por ello, teniendo en cuenta que el grupo de calidad del centro responde a Calisoft, las pruebas seleccionadas para conformar el *set* coinciden con las precisadas por el mismo, a excepción de las pruebas comparativas que no se adaptan a los sistemas de código abierto. A continuación se describen las pruebas seleccionadas:

Prueba funcional.

- Propósito de la prueba

- Lugar, fecha y duración

- *Hardware* y *software* necesarios

- Facilitadores y nivel de participación
- Usuarios participantes
- Tareas a desarrollar por los usuarios
- Métodos de recolección y análisis de datos

Este mismo procedimiento, con igual nivel de detalle es aplicado a las restantes pruebas:

Prueba de seguridad.

Prueba de recuperación y tolerancia a fallos.

Prueba de usabilidad.

Prueba de carga.

Prueba de estrés.

Prueba de configuración.

Prueba de instalación.

Capítulo 3

Análisis de los resultados y valoración de la propuesta

Introducción

Para que un proceso de pruebas sea exitoso se requiere de un análisis final de los resultados arrojados.

3.1. Comparación de los resultados obtenidos en las pruebas realizadas a la versión 3.0 antes y después de aplicar la solución propuesta.

Anteriormente solo se habían realizado pruebas funcionales a la aplicación, los resultados arrojados serán utilizados como punto de referencia para establecer una comparación con los obtenidos durante las pruebas planificadas en la presente investigación, para ello se debe ver la tabla de comparación de resultados (Ver Anexo B).

Aunque se han reducido considerablemente los errores detectados referentes a las funciones del sistema, los resultados continúan siendo inaceptables, pues no se considera que un producto que esté en explotación persista con tantos defectos, lo mismo se corrobora con el fallo de los requisitos:

- Adicionar Noticia.
- Modificar Noticia.
- Eliminar Noticia.
- Mostrar Noticia.

A estas imperfecciones se suman el resto de las NC encontradas durante las pruebas realizadas como parte de la propuesta.

Conclusiones Generales

Con la realización de este trabajo se logró elaborar un *set* que guiará el proceso de pruebas durante el ciclo de vida del proyecto Plataforma Libre para la Gestión de Procesos en la Facultad Regional Granma versión 3.0 con el objetivo de asegurar la calidad del producto. Para ello se diseñó y ejecutó un conjunto de pruebas que permitieron identificar varios defectos, posibilitando emitir una valoración crítica con respecto a la calidad del producto. La ejecución de pruebas de carga y estrés demostraron que la aplicación solo es capaz de soportar 40 usuarios online sin presentar ningún inconveniente durante la navegación, número demasiado pequeño considerando la cantidad de personas que lo utilizan, además el tiempo de espera para que carguen las páginas no es aceptable. La aplicación no presenta ningún problema crítico relacionado con la usabilidad y recuperación, pues los pocos errores detectados no influyen considerablemente en el sistema, asegurando así, mayor aceptación por los que acceden a este. La realización de diferentes tipos de pruebas al sistema creado por el proyecto Plataforma Libre para la Gestión de Procesos en la Facultad Regional Granma versión 3.0 demostró que la obtención de un *software* con calidad depende de la detección y corrección de la mayor cantidad de defectos, elemento que no se tuvo en cuenta durante la realización de las primeras pruebas.

El set de pruebas propuesto es necesario porque actualmente el proyecto no cuenta con dicho artefacto, además, facilitará el proceso de aseguramiento de la calidad, convirtiéndolo en una actividad más organizada y eficiente.

Anexo A

Encuesta

Encuesta para obtener información sobre el conocimiento de las pruebas de *software* y las herramientas automatizadas para tal fin.

Con el propósito principal de juzgar el nivel de conocimiento existente sobre el proceso de pruebas de *software* en la Facultad Regional Granma y por la importancia que tiene dicho proceso en el desarrollo de *software*; se necesita de su colaboración a través de esta encuesta en la que se recogerá su opinión y conocimiento acerca del tema de forma anónima.

De antemano se le agradece su colaboración y el tiempo que ha dedicado a la misma.

a) ¿Conoce qué es una prueba de *software*?

_____ Si _____ No

En caso de que su respuesta sea afirmativa diga cuáles de estas pruebas que a continuación se listan conoce:

- — Pruebas de unidad.
- — Pruebas de integración.
- — Pruebas de caja blanca.
- — Pruebas de caja negra.
- — Pruebas funcionales.

- — Pruebas de seguridad.
- — Pruebas de recuperación y tolerancia a los fallos.
- — Pruebas comparativas.
- — Pruebas de usabilidad.
- — Pruebas de carga.
- — Pruebas de estrés.
- — Pruebas de configuración.
- — Pruebas de instalación.
- — Pruebas de robustez.
- — Pruebas de aceptación.

b) **¿Se realiza en su centro (o específicamente en su proyecto) alguna prueba de *software*?**

_____ Si _____ No

En caso de que su respuesta sea afirmativa diga cuáles de estas pruebas se realizan:

- — Pruebas de unidad.
- — Pruebas de integración.
- — Pruebas de caja blanca.
- — Pruebas de caja negra.
- — Pruebas funcionales.
- — Pruebas de seguridad.
- — Pruebas de recuperación y tolerancia a los fallos.
- — Pruebas comparativas.
- — Pruebas de usabilidad.
- — Pruebas de carga.

- — Pruebas de estrés.
- — Pruebas de configuración.
- — Pruebas de instalación.
- — Pruebas de robustez.
- — Pruebas de aceptación.

c) **¿Conoce qué es una herramienta de prueba de *software*?**

_____ Si _____ No

En caso de que su respuesta sea afirmativa diga cuáles de estas herramientas que se listan a continuación conoce:

- — JMETER
- — SELENIUM IDE
- — WEBKING DE PARASOFT
- — OPEN LOAD
- — OPENSTA
- — FUNKLOAD
- — WEBLOAD
- — DB UNIT
- — CRUISE CONTROL
- — NESSUS
- — PYLOT
- — DATA GENERATOR FOR POSTGRES

d) **¿Está de acuerdo usted con el uso de herramientas de pruebas?**

_____ Si _____ No

En caso de que su respuesta sea afirmativa, diga por qué:

Anexo B

Comparación de resultados

Funcionalidades defectuosas antes	Funcionalidades defectuosas después
Editar Usuario	
Eliminar Usuario	Eliminar Usuario
Adicionar Grupo	
Editar Grupo	
Eliminar Grupo	Eliminar Grupo
Loguear Usuario	
Editar Avisos	
Adicionar Efeméride	
Modificar Efeméride	
Eliminar Efeméride	
Adicionar Menú del Comedor	Adicionar Menú del Comedor
Modificar Menú del Comedor	Modificar Menú del Comedor
Eliminar Menú del Comedor	Eliminar Menú del Comedor
Adicionar Consumo Diario	
Modificar Consumo Diario	
Eliminar Consumo Diario	
Adicionar Consumo Mensual	
Modificar Consumo Mensual	
Eliminar Consumo Mensual	
	Adicionar Noticia
	Modificar Noticia
	Eliminar Noticia
	Mostrar Noticia
19 fallos	9 fallos

Tabla B.1: Comparación de resultados

Referencias bibliográficas

- [1] **TERUEL**, Alejandro. Pruebas repetibles y mantenibles. *Suites de Pruebas* [en línea]. Universidad Simón Bolívar. Enero-Marzo, 2001. Disponible en url: <<http://ldc.usb.ve/teruel/ci4713/clases2001/pruebasRep.htmlsuites>>.
- [2] **LÓPEZ, RODRÍGUEZ D.** *Prácticas del curso de Prueba de Software* [en línea]. [citado 1 febrero 2012]. Disponible en url: <<http://www.google.com/cu/url?sa=trct=jq=Las+suites+de+prueba+%28test+suites%29+p+ejecutar+de+manera+colectiva+casos%2C+m%C3%A9todos+y%2Fo+suites+de+prue+relacionadossource=webcd=1ved=0CCYQFjA Aurl=http%3A%2F%2Fwww.cc.uah.es%2Fdocencia%2FPruebas%2FPruebas4x1.pdf&ei=6oSFT8SkF9T3gAfhnJnaBwusg=AFQJrSwZBR172wEYeSrx-rTJJ8w>>.
- [3] Ingeniería de Software. *Practica 5: Realización de pruebas* [en línea]. 2005. [citado 1 febrero 2012]. Disponible en url: <<http://www.humbertocervantes.net/cursos/ingsoft/practica5/practica5.html>>.
- [4] **SARCO**, José Pablo. *JMeter: Performance Testing con JMeter Parte 1 Testing en Español* [en línea]. [citado 7 marzo 2012]. Disponible en url: <<http://josepablosarco.wordpress.com/2009/01/12/jmeter-performance-testing-con-jmeter-parte-1>>.
- [5] **ArticuloZ.** *Herramienta Para La Ejecución De Las Pruebas* [en línea]. [citado 7 marzo 2012]. 2009. Disponible en url: <<http://www.articuloz.com/software-articulos/herramienta-para-la-ejecucion-de-las->

pruebas-886032.html>.

- [6] **ESMITE**, Ignacio. *Automatización y Gestión de las Pruebas Funcionales usando Herramientas Open Source* [en línea]. Universidad de la República Montevideo, Uruguay. [citado 7 marzo 2012]. Disponible en url:
<<http://www.google.com.cu/url?sa=trct=jq=Selenium+simple%2C+potente+y+flexiblesourc=webcd=1ved=0CFUQFjAAurl=http%3A%2F%2Fwww.ces.com.uy>>