Versión 3.0

Histórico de revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 21/11/2012 | 1.0 | Inicio del documento | Sergio Martínez Pérez |
| 03/12/2012 | 2.0 | Especificación de la versión a probar | Jorge Antonio Díaz-Benito Soriano |
| 18/12/2012 | 3.0 | Continuación del documento para la segunda iteración. | Sergio Martínez Pérez |
|  |  |  |  |

Índice

1. Introducción 1

1.1 Objetivos 1

1.2 Ámbito 1

1.3 Referencias 1

2. Elementos de prueba 1

3. Características que se deben probar 1

4. Características que no se van a probar 1

5. Aproximación 1

5.1 Pruebas funcionales 1

5.2 Pruebas de carga 1

5.3 Pruebas de rendimiento 2

5.4 Pruebas de seguridad 2

6. Criterios para decidir si pasa/falla un elemento 2

7. Criterios de suspensión y reanudación 2

7.1 Suspensión 2

7.2 Reanudación 2

8. Pruebas que se deben entregar 2

9. Tareas de prueba 2

10. Necesidades ambientales 2

10.1 Hardware 2

10.2 Software 2

10.3 Seguridad 2

10.4 Herramientas 2

10.5 Publicaciones 2

11. Responsabilidades. 3

12. Aprobación 3

# Introducción

## Objetivos

El objetivo de las pruebas es conseguir un software lo más depurado posible y con la funcionalidad que se espera del mismo, según se ha especificado en los flujos de cada uno de los casos de uso.

## Ámbito

Especificamos en este documento la metodología a seguir para el correcto desarrollo de las pruebas.

## Referencias

En este documento haremos referencia a los documentos del equipo de desarrollo que son el de Procedimientos de Pruebas, Casos de Prueba e Informe de Pruebas.

# Elementos de prueba

Se prueba la aplicación *OMS.jar*, revisión 946. (Iteración 1).

Se prueba la aplicación OMS.jar, revisión 1232. (Iteración 2).

# Características que se deben probar

Se debe probar que todos los casos de uso se han implementado tal y como se describieron en los documentos de casos de uso, es decir, verificar que cumplen con los flujos establecidos y con todas las características que se describieron. Dichos casos de uso son:

* CPCU1\_26-11-2012 – Caso de uso "Alta Usuario".
* CPCU2\_26-11-2012 – Caso de uso "Alta Tarea".
* CPCU3\_26-11-2012 – Caso de uso "Alta proyecto".
* CPCU4\_18-12-2012 - Caso de uso "Consultar Estado Proyecto".
* CPCU5\_18-12-2012 - Caso de uso "Actualizar Estado Tarea".

# Características que no se van a probar

En principio no dejaremos ninguna característica de ningún caso de uso sin probar. El caso de uso debe estar implementado tal y como se diseñó.

# Aproximación

## Pruebas funcionales

Cada caso de uso será probado por dos miembros del grupo.

En caso de encontrar un bug, éste se reportará al grupo N mediante Bugzilla. Se considerará una prueba satisfactoria si se han probado todos los caminos posibles del caso de uso, es decir, tanto los flujos básicos como los alternativos, en caso de haberlos.

Dichas pruebas serán las principales a realizar. Para ello, tendremos en cuenta el objetivo del caso de uso. Si al final de todos los posibles caminos previstos se llega al resultado esperado del caso de uso, se dirá que la prueba es exitosa. En caso contrario, de encontrar un resultado inesperado al final de cualquier camino, se reportará el bug, indicando el camino seguido hasta llegar a ese error. La técnica de generación de casos de prueba es la de generación de casos de prueba a partir de casos de uso.

# Criterios para decidir si pasa/falla un elemento

El criterio de acierto o fallo será que al final de la prueba el resultado de la misma sea el que se espera del caso de uso, según la documentación aportada del mismo. También consideraremos un fallo, en el caso de encontrar un fallo de carga, rendimiento o seguridad durante el proceso de la prueba.

# Criterios de suspensión y reanudación

## Suspensión

La prueba se suspenderá en caso de fallo completo del software, es decir, por fallo de la implementación.

## Reanudación

La prueba se reanudará una vez que se haya hecho efectiva la reparación del software por parte del otro grupo.

# Pruebas que se deben entregar

Se deben entregar los documentos de “Plan de Pruebas”, “Procedimiento de Pruebas”, “Logs” e “Informe de Pruebas”.

# Tareas de prueba

Antes de realizar la prueba, debemos especificar todos los posibles escenarios a probar. Se ha seguido el proceso de planificación, diseño, implementación y ejecución.

# Necesidades ambientales

## Hardware

No sabemos qué hardware hace falta, puesto que el equipo de desarrollo no nos ha comentado nada.

## Software

No sabemos qué software hace falta, puesto que el equipo de desarrollo no nos ha comentado nada.

## Seguridad

No sabemos hasta qué punto la seguridad es influyente en el proyecto en cuestión.

## Herramientas

Necesitaremos tener acceso a Bugzilla para reportar los posibles bugs encontrados.

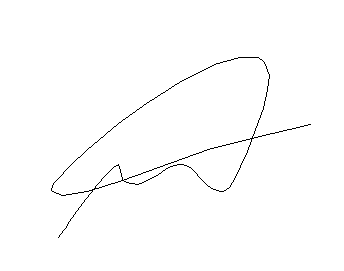
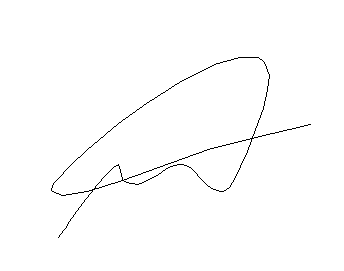
# Responsabilidades.

En el documento "Plan de Gestión de Configuración" ya se expuso quiénes eran los encargados de tratar todos los cambios pertinentes en el software. En cuanto a la realización de la pruebas se harán de forma individual.

|  |  |
| --- | --- |
| **Tarea** | **Responsable** |
| Planificación y diseño | Jorge Antonio Díaz-Benito Soriano  Sergio Martínez Pérez  Miguel Ángel Moraleda Sánchez  Jakob Ursic  Miguel Ángel Cuenca Navarro  Fernando Rubio Perona |
| Implementación | Jorge Antonio Díaz-Benito Soriano  Sergio Martínez Pérez |
| Ejecución | Jorge Antonio Díaz-Benito Soriano  Sergio Martínez Pérez |

El encargado de realizar las peticiones de cambio en Bugzilla será el jefe de proyecto, debido a que es el único que tiene los permisos necesarios.

# Aprobación



Jefe de proyecto Gestor de pruebas

Jorge Antonio Díaz-Benito Soriano Sergio Martínez Pérez