**<Sistema de Control de Entregas>**

**Version <1.0>**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <27/oct/15> | 0.1 | Adaptación al proyecto | Borboa Lopez Jesus Bernardo  Lopez Lugo Alejandro  Michel Martínez |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

1. Introduction

1.1 Purpose

1.2 Scope

1.3 Intended Audience

1.4 Document Terminology and Acronyms

1.5 References

1.6 Document Structure

2. Evaluation Mission and Test Motivation

2.1 Background

2.2 Evaluation Mission

2.3 Test Motivators

3. Target Test Items

4. Outline of Planned Tests

4.1 Outline of Test Inclusions

4.2 Outline of Test Exclusions

5. Test Approach

5.1 Initial Test-Idea Catalogs and Other Reference Sources

5.2 Testing Techniques and Types

5.2.1 Prueba de datos e integridad de la base de datos

5.2.2 Prueba de funcionalidad

5.2.3 Prueba de Interfaz de Usuario (UI)

5.2.4 Prueba de desempeño

5.2.5 Prueba de carga

5.2.6 Prueba de volumen

5.2.7 Pruebas de seguridad y control de acceso

5.2.8 Prueba de fallas y recuperación

5.2.9 Pruebas de Configuración

5.2.10 Prueba de instalación

6. Deliverables

6.1 Additional Work Products

6.1.1 Detailed Test Results

6.1.2 Additional Automated Functional Test Scripts

6.1.3 Test Guidelines

6.1.4 Traceability Matrices

7. Environmental Needs

7.1 Base System Hardware

7.2 Base Software Elements in the Test Environment

7.3 Productivity and Support Tools

8. Responsibilities, Staffing, and Training Needs

8.1 People and Roles

9. Iteration Milestones

**Test Plan**

1. **Introducción**
   1. **Propósito**

El propósito del Plan de Pruebas de Sistema de control de entregas (SCE) es brindar toda la información necesaria para planear y para controlar el esfuerzo de la prueba. Describe el método para probar el software, y es el plan principal generado y usado por los encargados para dirigir el esfuerzo de la prueba

El plan de pruebas del proyecto SCE comprende los siguientes objetivos:

• Identificar los elementos que podrían ser sujetos a prueba

• Identifica la motivación y las ideas detrás de las áreas de prueba que serán cubiertas.

• Delinea el método de prueba que será usado.

• Identifica los recursos requeridos y proporciona un estimado del esfuerzo para la prueba.

• Enumera los elementos entregables del proyecto de prueba.

* 1. **Alcance**

Los niveles de prueba por cubrir son pruebas de Unidad, Integración y Sistema y los tipos de pruebas por cubrir son de Funcionalidad, Usabilidad, Confiabilidad y Soportabilidad.

* 1. **Intended Audience**

Este plan de pruebas está dirigido a los Probadores y a Administrador del proyecto como una guía para la planeación y control de las pruebas.

* 1. **Document Terminology and Acronyms**

[This subsection provides the definitions of any terms, acronyms, and abbreviations required to properly interpret the **Test Plan**. Avoid listing items that are generally applicable to the project as a whole and that are already defined in the project’s Glossary. Include a reference to the project’s Glossary in the References section.]

* 1. **References**

[This subsection provides a list of the documents referenced elsewhere within the **Test Plan**. Identify each document by title, version (or report number if applicable), date, and publishing organization or original author. Avoid listing documents that are influential but not directly referenced. Specify the sources from which the “official versions” of the references can be obtained, such as intranet UNC names or document reference codes. This information may be provided by reference to an appendix or to another document.]

* 1. **Document Structure**

[This subsection outlines what the rest of the **Test Plan** contains and gives an introduction to how the rest of the document is organized. This section may be eliminated if a Table of Contents is used.]

1. **Evaluation Mission and Test Motivation**

[Provide an overview of the mission and motivation for the testing that will be conducted in this iteration.]

* 1. **Background**

[Provide a brief description of the background surrounding why the test effort defined by this **Test Plan** will be undertaken. Include information such as the key problem being solved, the major benefits of the solution, the planned architecture of the solution, and a brief history of the project. Where this information is defined in other documents, you can include references to those other more detailed documents if appropriate. This section should only be about three to five paragraphs in length.]

* 1. **Misión de la evaluación.**

Encontrar tantos errores como sean posibles

Encontrar problemas importantes, y determinar los riesgos percibidos

Advertir sobre los riesgos percibidos

Certificar un estándar

Verificar una especificación (requerimientos, diseño o cualquier otro elemento especificado)

Advertir sobre la calidad del producto, y la satisfacción de los stakeholders.

Satisfacer los procedimientos del proceso

* 1. **Test Motivators**

Riesgos de calidad

Riesgos técnicos

Riesgos del proyecto

Casos de uso

Requerimientos funcionales

Requerimientos no funcionales

Elementos de diseño

Sospechas de fallos

1. **Target Test Items**

The listing below identifies those test items⎯software, hardware, and supporting product elements ⎯that have been identified as targets for testing. This list represents what items will be tested.

[Provide a high level list of the major target test items. This list should include both items produced directly by the project development team, and items that those products rely on; for example, basic processor hardware, peripheral devices, operating systems, third-party products or components, and so forth. Consider grouping the list by category and assigning relative importance to each motivator.]

1. **Outline of Planned Tests**
   1. **Outline of Test Inclusions**

Esta lista representa qué será probado.

* + - 1. Pruebas a la Base de Datos

Verificar que los usuarios puedan registrar y recuperar la información

* + - 1. Pruebas de Funcionalidad

Verificar que los usuarios puedan ver la información que solicitan

Verificar que el conjunto de datos capturados por el usuario sean insertados correctamente

Verificar que los usuarios que no tienen vigencia no puedan ver la información

Verificar que los contenidos de los reportes sean correctos

Verificar que los reportes pueden ser descargados con un formato Excel.

* + - 1. Pruebas de interfaz de usuario (UI)

Navegar a través de todos los casos del uso, verificando que cada panel de la interfaz de usuario pueda ser entendido fácilmente

Verificar todas las funciones de ayuda en línea

Verificar que todas las pantallas cumplan con los estándares de diseño de interfaz

* + - 1. Pruebas de desempeño

Verificar que el tiempo de respuesta del sistema al recuperar la información en estación local

Verificar el tiempo de respuesta cuando está conectado a la red de área local.

Verificar el tiempo de respuesta de las conexiones en la red de acuerdo a los dispositivos de comunicación y sus velocidades (Modem, Internet, etc.)

* + - 1. Pruebas de carga

Verificar la respuesta del sistema con 1 usuario.

Verificar el tiempo de respuesta con 2 usuarios concurrentes.

Verificar la respuesta con 3 usuarios concurrentes.

* + - 1. Pruebas de stress

Ninguna.

* + - 1. Prueba de volumen.

Verificar el máximo número de clientes conectados todos desempeñándose de manera concurrente.

Verificar que múltiples queries se ejecutan simultáneamente.

* + - 1. Pruebas de seguridad y control de acceso

Verificar que los usuarios que tengan privilegios no tengan acceso al sistema

Verificar que los usuarios que no sean tipo director no aprueben los casos epidemiológicos

Verificar que los usuarios de sólo consulta solo vean la información a la que tienen privilegio

* + - 1. Pruebas de Fallas/Recuperación

Verificar la recuperación del sistema ante la presencia de fallos de energía.

Verificar la recuperación del sistema cuando exista corrupción de datos en la base de datos.

* + - 1. Pruebas de configuración

Verificar la operación del Excel 2003

Verificar la operación del Microsoft Internet Explorer V5.x

* + - 1. Pruebas de instalación

Verificar que la instalación del equipo se haga de acuerdo a sus características (nuevo o usado).

Verificar que la instalación del software se haga de acuerdo a las versiones del sistema correspondientes al equipo.

* 1. **Outline of Test Exclusions**

[Provide a high level outline of the potential tests that might have been conducted but that have been ***explicitly excluded*** from this plan. If a type of test will not be implemented and executed, indicate this in a sentence stating the test will not be implemented or executed and stating the justification, such as:

“These tests do not help achieve the evaluation mission.”

“There are insufficient resources to conduct these tests.”

“These tests are unnecessary due to the testing conducted by xxxx.”

As a heuristic, if you think it would be reasonable for one of your audience members to expect a certain aspect of testing to be included that you will not or cannot address, you should note it’s exclusion: If the team agrees the exclusion is obvious, you probably don’t need to list it.]

1. **Test Approach**

One aspect to consider for the test approach is the techniques to be used. This should include an outline of how each technique can be implemented, both from a manual and/or an automated perspective, and the criterion for knowing that the technique is useful and successful. For each technique, provide a description of the technique and define why it is an important part of the test approach by briefly outlining how it helps achieve the Evaluation Mission or addresses the Test Motivators.

Another aspect to discuss in this section is the Fault or Failure models that are applicable and ways to approach evaluating them.

As you define each aspect of the approach, you should update Section 8, Responsibilities, Staffing, and Training Needs, to document the test environment configuration and other resources that will be needed to implement each aspect.]

* 1. **Initial Test-Idea Catalogs and Other Reference Sources**

[Provide a listing of existing resources that will be referenced to stimulate the identification and selection of specific tests to be conducted. An example Test-Ideas Catalog is provided in the examples section of RUP.]

* 1. **Testing Techniques and Types**
     1. *Prueba de datos e integridad de la base de datos*

Las bases de datos y los procesos de la base de datos se deben probar como subsistema independiente. Esta prueba debe probar los subsistemas sin la interfaz de usuario. Se debe realizar investigación adicional acerca del Sistema de Administración de Base de Datos (DBMS) para conocer las herramientas y técnicas que pueden existir para apoyar esta prueba.

|  |  |
| --- | --- |
| Objetivo de la prueba: | Asegurarse que los métodos de acceso y los procesos de funcionan correctamente sin corrupción en los datos. |
| Técnica: | Invocar cada método y proceso de la base de datos, probando con datos validos e inválidos  Inspeccionar la base de datos para asegurarse que los datos han sido insertados según lo previsto, todos los acontecimientos de la base de datos ocurrieron correctamente, y revisar la recuperación de los datos para asegurarse que los datos correctos fueron recuperados. |
| Criterios de éxito: | Todos los métodos y procesos de acceso de base de datos funcionan según lo diseñado y sin ninguna corrupción de los datos. |
| Consideraciones especiales: | Proceso podría ser invocado manualmente.  Bases de datos pequeñas o como mínimo clasificadas (número limitado de expedientes) se deben utilizar para aumentar la visibilidad de cualquier acontecimiento no-aceptable. |

* + 1. *Prueba de funcionalidad*

La prueba de funcionalidad debe centrarse en cualquier requisito que pueda ser trazado directamente hacia los casos de uso o funciones del negocio y reglas de negocio. Las metas de estas pruebas son verificar la aceptación apropiada de datos, de proceso, y de recuperación, así como la implementación apropiada de las reglas de negocio. Este tipo de prueba se basa sobre técnicas de la caja negra; es decir verifica la aplicación y los procesos internos interactuando con la aplicación a través de las interfaces gráficas de usuario (GUI) y analizando la salida o los resultados.

|  |  |
| --- | --- |
| Objetivo de la prueba: | Asegurarse que la prueba de funcionalidad, incluyendo la navegación, datos de entrada, procesamiento y recuperación. |
| Técnica: | Ejecutar cada caso de uso, flujo de caso de uso, o función, utilizando datos válidos e inválidos, para verificar:  Los datos esperados ocurran cuando los datos validos son utilizados  El error apropiado/ mensajes de advertencia son desplegados cuando se han utilizado datos inválidos.  Cada regla de negocio es aplicada correctamente |
| Criterio de éxito: | Todas la pruebas planeadas han sido ejecutadas  Todos los defectos encontrados han sido resueltos |
| Consideraciones especiales: | Ninguna. |

* + 1. *Prueba de Interfaz de Usuario (UI)*

La prueba interfaz de usuario (UI) verifica la interacción de un usuario con el software. El propósito de la prueba es asegurarse que la UI provee del usuario el acceso y la navegación apropiados con base a las funciones del elemento de prueba. Además, la prueba de UI se asegura de que los objetos dentro de la UI funcionan según lo esperado de acuerdo a los estándares corporativos o de la industria.

|  |  |
| --- | --- |
| Objetivo de la prueba: | Verificar:  Que la navegación refleja las funciones del negocio y requerimientos, incluyendo cada pantalla, cada campo y métodos de acceso utilizados (tabs, movimientos del mouse, combinación de teclas y de acceso rápido)  Que los nombres de los objetos tales como nombres de menús, tamaño, posición, estén conforme a los estándares. |
| Técnica: | Crear/modificar las pruebas de cada pantalla para verificar la navegación correcta y los estados de cada objeto. |
| Criterio de éxito: | Cada pantalla es verificada de manera satisfactoria para mantener consistencia con las guías de diseño. |
| Consideraciones especiales: | No todas las características para los objetos del cliente y de los terceros pueden ser alcanzadas. |

* + 1. *Prueba de desempeño*

La prueba de desempeño es una prueba en donde los tiempos de respuesta, porcentajes de transacciones, y otros requerimientos sensibles al tiempo se miden y se evalúan. El propósito es verificar que el desempeño de los requisitos ha sido alcanzado. Esta prueba es implementada y ejecutada para obtener el comportamiento de los elementos a prueba en función de condiciones tales como configuraciones de la carga de trabajo o de hardware.

|  |  |
| --- | --- |
| Objetivo de la prueba: | Verificar el comportamiento del desempeño diseñando transacciones bajo las siguientes condiciones:  Carga normal prevista  Carga fuera de lo previsto |
| Técnica: | Modificar los archivos de datos (para aumentar el número de transacciones) o correr en más de una computadora (cliente).  Los scripts pueden correrse sobre una computadora (cuando se quiera comparar un usuario, un transacción simple) y repetirlo con múltiples clientes (virtual o actual) |
| Criterio de éxito: | Los resultados de las transacciones simples / un usuario o múltiples/ múltiples usuarios se concluyen sin fallas y de acuerdo al tiempo estimado. |
| Consideraciones especiales: | Existen varios métodos que pueden ser usados para desarrollar esta prueba:   * "Manejo de transacciones" directamente en el servidor, usualmente en la forma de llamados de SQL * Crear llamados de carga de usuario "virtual" para similar muchos clientes. Las herramientas como terminales remotas se usan en esta técnica que también puede ser usada para simular el "trafico" en la red. * Usar múltiples clientes físicos cada uno corriendo scripts de prueba para poner a carga el sistema.   La prueba de funcionamiento se debe realizar en una máquina dedicada o en un tiempo dedicado. Esto permite control completo y la medida exacta.  Las bases de datos usadas para la prueba de funcionamiento deben ser cualquier tamaño real, o escalado igualmente. |

* + 1. *Prueba de carga*

La prueba de la carga es una prueba del funcionamiento que sujeta al elemento de prueba a cargas de trabajo que varían para medir y evaluar los comportamientos de desempeño y las capacidades del elemento de prueba para continuar funcionando correctamente bajo estas diversas cargas de trabajo. El propósito de la prueba es determinar y asegurar que el sistema funciona correctamente más allá de la carga de trabajo máxima prevista. Además, la prueba de la carga evalúa las características de funcionamiento, tales como tiempos de reacción, porcentajes de transacción, y otras elementos sensibles al tiempo).

|  |  |
| --- | --- |
| Objetivo de la prueba: | Verifique el tiempo de los comportamientos de funcionamiento para las transacciones diseñadas o los casos del negocio bajo condiciones de la carga de trabajo que varían. |
| Técnica: | Modificar los archivos (para incrementar el número de transacciones) o las pruebas para incrementar el número de tiempo en que ocurre cada transacción. |
| Criterio de éxito: | Múltiples transacciones / múltiples usuarios: El criterio de éxito de las pruebas sin ninguna falla y con una aceptación de tiempo. |
| Consideraciones especiales: | La prueba de carga se debe realizar en una máquina dedicada o en un tiempo dedicado. Esto permite control completo y la medida exacta  Las bases de datos usadas para la prueba de funcionamiento deben ser cualquier tamaño real, o escalado igualmente. |

* + 1. *Prueba de volumen*

La prueba de volumen sujeta al elemento de prueba a grandes cantidades de datos para determinar si se alcanzan los límites que hacen fallar al software. También identifica la carga máxima continua o volumen que el elemento de prueba puede manejar por un período dado. Por ejemplo, si el elemento de prueba está procesando un sistema de expedientes de la base de datos para generar un informe, una prueba de volumen utilizaría una base de datos grande para la prueba, y comprobaría que el software se comportó normalmente y que produjo el informe correcto.

|  |  |
| --- | --- |
| Objetivo de la prueba: | Verificar que las pruebas funcionen correctamente en los siguientes escenarios de gran volumen:  Máximo (actual o físicamente capaz) número de clientes conectados (o simulados) todos desempeñándose al mismo tiempo, en el peor de los casos (desempeño) de las funciones del negocio por un tiempo extendido.  El tamaño máximo de la base de datos ha sido (real alcanzado o escalado) y múltiples queries /reportes de transacciones se ejecutan simultáneamente. |
| Técnica: | Múltiples clientes pueden usar, aun corriendo la misma prueba o una prueba complementaria para producir el peor de los casos de un volumen de transacción por un tiempo extendido.  El tamaño máximo de la base de datos es creado (real, escalado, o llenado con datos representativos) y múltiples clientes se usan para ejecutar queries / reportes de transacciones simultáneamente por periodos extendidos. |
| Criterio de éxito: | Se han ejecutado todas las pruebas previstas y los límites especificados del sistema se alcanzan/se exceden sin el software o el software falla. |
| Consideraciones especiales: | ¿Qué período del tiempo sería considerado un tiempo aceptable para las condiciones del alto volumen (según lo observado arriba)? |

* + 1. *Pruebas de seguridad y control de acceso*

Estas pruebas se centran en dos áreas claves:

∙ Aplicación-nivel de seguridad, incluyendo acceso a los datos o funciones de negocio.

∙ Sistema- nivel de seguridad, incluyendo la entrada dentro o accesos remotos al sistema.

De acuerdo en la seguridad que se desea, el nivel de seguridad de la aplicación se asegura que a los usuarios se les restrinjan las funciones específicas o casos de uso, o que están limitados en los datos disponibles para ellos. Por ejemplo, a cada usuario se le puede permitir incorporar datos y crear nuevas cuentas, pero solamente los encargados pueden suprimirlas. Si hay seguridad a nivel de datos, la prueba asegura que el "tipo de usuario A" puede considerar toda la información del cliente, incluyendo datos financieros, sin embargo, del "usuario B" ve solamente la información para el mismo cliente. La seguridad a nivel sistema se asegura de que solamente esos usuarios con privilegios para acceder al sistema sean capaces de tener acceso a la aplicación y solamente a través de las entradas apropiadas.

|  |  |
| --- | --- |
| Objetivo de la prueba: | Aplicación- nivel de seguridad.: verificar que algún usuario pueda tener acceso solo a esas funciones / datos para los cuales su tipo del usuario sea permisos proporcionados.  Sistema- nivel de seguridad: Verificar que solamente los usuarios con acceso a el sistema y aplicaciones sean permitidos en éstos. |
| Técnica: | Aplicación-nivel: Identificar y enumerar cada tipo de usuario y sus funciones / tipos de datos a los que cada uno tiene permisos  Crear una prueba para cada tipo de usuario y verificar cada permiso creando transacciones específicas para cada tipo de usuario.  Modificar el tipo de usuarios y volver a ejecutar la prueba para el mismo usuario. En cada caso verificar las funciones adicionales / que los datos sean correctamente disponibles o denegados.  Sistema-nivel de acceso (ver consideraciones especiales) |
| Criterio de éxito: | Para cada tipo de usuario conocido, la función apropiada / los datos están disponibles y todas las transacciones funcionan según lo esperado y funcionan en pruebas de función anteriores |
| Consideraciones epeciales: | El acceso al sistema se debe repasar/discutir con el administrador apropiado de la red o de sistemas. |

* + 1. *Prueba de fallas y recuperación*

La prueba de fallas y recuperación se asegura de que el elemento de prueba pueda recuperarse con éxito de una variedad de malfuncionamientos del hardware, del software o de la red con la pérdida indebida de datos o de integridad de datos. Para esos sistemas que deban mantener el funcionamiento se asegura de que, cuando ocurre una condición de falla, los sistemas alternos o de backup "asuman el control correctamente" sin ninguna pérdida de datos o de transacciones. La prueba de recuperación es un proceso antagónico de la prueba en el cual la aplicación o el sistema se expone a las condiciones extremas, o condiciones simuladas, para causar una falla, tal como fallas en la entrada-salida de dispositivos (I/O), o apuntadores inválidos y llaves de la base de datos. Se invocan los procesos de recuperación, y la aplicación o sistema es monitoreado e inspeccionado para verificar su recuperación apropiada.

|  |  |
| --- | --- |
| Objetivo de la prueba: | Simular las condiciones de falla y ejercitar los procesos de la recuperación (manuales y automatizados) para restaurar la base de datos, la aplicación, y el sistema. Los siguientes de tipos condiciones se incluyen en la prueba para observar y para registrar comportamiento después de la recuperación:  interrupción de la energía al cliente   * interrupción de la energía al servidor * interrupción de la comunicación vía los servidores de la red * interrupción, comunicación, o apagón a los reguladores (dispositivos de almacenaje dinámicos del acceso) * ciclos incompletos (los procedimientos de filtrado de los datos interrumpidos, sincronización de los procesos de datos interrumpido) * punteros inválidos o llaves de la base de datos * elementos de datos inválidos o corrompidos en base de datos |
| Técnica: | interrupción de la energía al cliente: apagar la PC   * interrupción de la energía al servidor: simular o iniciar la baja de los procesos del servidor   interrupción de la comunicación vía los servidores de la red: simular o iniciar la pérdida de comunicación con la red (físicamente desconectar los cables de la red o los ruteadores)   * interrupción, comunicación, o apagón a los reguladores (dispositivos de almacenaje dinámicos del acceso): simular o físicamente eliminar la comunicación con uno o más de los reguladores. * ciclos incompletos (los procedimientos de filtrado de los datos interrumpidos, sincronización de los procesos de datos interrumpido) * punteros inválidos o llaves de la base de datos * elementos de datos inválidos o corrompidos en base de datos   Una vez que las condiciones anteriores se hayan alcanzado, se ejecutan ciertas transacciones y una vez terminado este segundo punto, los procedimientos de recuperación pueden ser invocados.  La prueba para los ciclos incompletos utiliza la misma técnica excepto que los procesos de la base de datos por si mismos pueden ser abortados o terminados prematuramente.  La prueba para las condiciones siguientes requiere que el estado de la base de datos sea conocido.  Varios campos, punteros, y llaves de la base de datos se deben corromper manualmente y directamente dentro de la base de datos (vía las herramientas de la base de datos). Las transacciones adicionales se deben ejecutar usando las pruebas de la función de. |
| Criterio de éxito: | Esta técnica soporta:  Unos o más desastres simulados que implican unas o más combinaciones de la aplicación, base de datos y del sistema.  Unas o más recuperaciones simuladas que implican unas o más combinaciones de la aplicación, base de datos y del sistema para conocer el estado deseado. |
| Consideraciones especiales: | La prueba de la recuperación es altamente intrusa. Los procedimientos para desconectar el cableado (simulando pérdida de la energía o de la comunicación) pueden no ser deseables o factibles. Los métodos alternativos, tales como herramientas del software de diagnóstico pueden ser requeridos. Los recursos de los grupos de los sistemas (o las operaciones de computadora), de la base de datos, y del establecimiento de una red se requieren. Estas pruebas se deben funcionar después de horas o en una máquina aislada. |

* + 1. *Pruebas de Configuración*

La prueba de la configuración verifica la operación del elemento de prueba sobre diversas configuraciones del software y de hardware. En la mayoría de los ambientes de la producción, las especificaciones particulares del hardware para los sitios de trabajo del cliente, las conexiones de red, y los servidores de la base de datos varían. Las estaciones de trabajo del cliente pueden tener diversos software cargados, tales como aplicaciones, drivers, etc, en cualquier momento diversas combinaciones pueden ser activadas usando diferentes recursos.

|  |  |
| --- | --- |
| Objetivo de la prueba: | Verificar que la prueba funcione correctamente en las configuraciones requeridas del hardware/del software. |
| Técnica: | Abrir /cerrar varias pruebas relacionadas con el software relacionado a la aplicación, tales como Word, Excel.  Ejecute las transacciones seleccionadas para simular la interacción de un usuario con la aplicación y el software relacionado.  Repita estas pruebas reduciendo al mínimo la memoria convencional disponible en el cliente. |
| Criterio de éxito: | Para cada combinación de prueba (aplicación / software relacionado), todas las transacciones se terminan con éxito (sin falla). |
| Consideraciones especiales: | ¿Qué software es necesario, está disponible, accesible en el área de desktop de la pc?  Qué aplicaciones son típicamente utilizadas?  Qué datos son ejecutados por la aplicación (es decir hoja de balance en Excel, 100 páginas en documento Word)  Los sistemas, los servidores de la red, las bases de datos, el etc. pueden ser documentados como parte de esta prueba. |

* + 1. *Prueba de instalación*

La prueba de la instalación tiene dos propósitos. El primero es asegurarse de que el software se puede instalar bajo diversas condiciones tales como una nueva instalación, una actualización, e instalaciones personalizadas o completas bajo condiciones normales o anormales. Las condiciones anormales incluyen insuficiente espacio en disco, escasos privilegios de crear directorios, etcétera. El segundo propósito es verificar que, una vez que esté instalado, el software funciona correctamente. Esto generalmente significa correr un número de pruebas que fueron desarrolladas en las Pruebas de funcionalidad.

|  |  |
| --- | --- |
| Objetivo de la prueba: | Ejercitar la instalación sobre cada configuración de hardware requerida bajo las condiciones siguientes para observar y registrar los cambios del estado del comportamiento y de la configuración de la instalación del registro:  nueva instalación: una máquina nueva, nunca instalada previamente  actualización: una máquina instalada previamente, la misma actualización, misma versión  actualización: una máquina previamente instalada, versión anterior |
| Técnica: | * Desarrollar scripts manuales o automatizados para validar la condición de la máquina.   + nueva: nunca instalada   + la misma o versión anterior * Realice la instalación. |
| Criterio de éxito: | La técnica soporta la prueba de la instalación del producto desarrollado en unas o más configuraciones de la instalación. |
| Consideraciones especiales: | Qué transacciones se deben seleccionar para abarcar una prueba de confianza sobre el estado del equipo instalado de tal manera que los resultados sean exitosos y sin la pérdida de componentes de software importantes? |

1. **Deliverables**

Para cada prueba ejecutada, se creará un formato con el resultado de la prueba. Esto incluirá el nombre o la identificación de la prueba, el caso del uso o la especificación suplementaria con los cuales la prueba se relaciona, la fecha de la prueba, la identificación del probador, y los resultados de la prueba.

* 1. **Additional Work Products**

[In this section, identify the work products that are optional deliverables or those that should not be used to measure or assess the successful execution of the **Test Plan**.]

* + 1. *Detailed Test Results*

[This denotes either a collection of Microsoft Excel spreadsheets listing the results determined for each test case, or the repository of both test logs and determined results maintained by a specialized test product.]

* + 1. *Additional Automated Functional Test Scripts*

[These will be either a collection of the source code files for automated test scripts, or the repository of both source code and compiled executables for test scripts maintained by the test automation product.]

* + 1. *Test Guidelines*

[Test Guidelines cover a broad set of categories, including Test-Idea catalogs, Good Practice Guidance, Test patterns, Fault and Failure Models, Automation Design Standards, and so forth.]

* + 1. *Traceability Matrices*

[Using a tool such as Rational RequisistePro or MS Excel, provide one or more matrices of traceability relationships between traced items.]

1. **Environmental Needs**

Esta sección presenta los recursos no humanos requeridos para el plan de pruebas.

* 1. **Base System Hardware**

La siguiente tabla dispone los recursos del sistema para el esfuerzo del Plan de prueba.

Se recomienda que el sistema simule el ambiente de la producción, reduciendo el tamaño del acceso concurrente y de la base de datos, y así sucesivamente, siempre y cuando sea apropiado.

[**Note**: Add or delete items as appropriate.]

|  |  |  |
| --- | --- | --- |
| **System Resources** | | |
| **Resource** | **Quantity** | **Name and Type** |
| Database Server |  |  |
| —Network or Subnet |  | TBD |
| —Server Name |  | TBD |
| —Database Name |  | TBD |
| Client Test PCs |  |  |
| —Include special configuration requirements |  | TBD |
| Test Repository |  |  |
| —Network or Subnet |  | TBD |
| —Server Name |  | TBD |
| Test Development PCs |  | TBD |

* 1. **Base Software Elements in the Test Environment**

Los siguientes elementos del software se requieren en el ambiente de la prueba para este plan de prueba.

[Note: Add or delete items as appropriate.]

|  |  |  |
| --- | --- | --- |
| **Software Element Name** | **Version** | **Type and Other Notes** |
| NT Workstation |  | Operating System |
| Windows 2000 |  | Operating System |
| Internet Explorer |  | Internet Browser |
| Netscape Navigator |  | Internet Browser |
| MS Outlook |  | eMail Client software |
| Network Associates McAfee Virus Checker |  | Virus Detection and Recovery Software |

* 1. **Productivity and Support Tools**

Las siguientes herramientas serán empleadas para soportar el proceso de la prueba para este Plan de pruebas

[Note: Add or delete items as appropriate.]

|  |  |  |  |
| --- | --- | --- | --- |
| **Tool Category or Type** | **Tool Brand Name** | **Vendor or In-house** | **Version** |
| Test Management |  |  |  |
| Defect Tracking |  |  |  |
| ASQ Tool for functional testing |  |  |  |
| ASQ Tool for performance testing |  |  |  |
| Test Coverage Monitor or Profiler |  |  |  |
| Project Management |  |  |  |
| DBMS tools |  |  |  |

1. **Responsibilities, Staffing, and Training Needs**

[This section presents the required resources to address the test effort outlined in the **Test Plan**—the main responsibilities, and the knowledge or skill sets required of those resources.]

* 1. **People and Roles**

This table shows the staffing assumptions for the test effort.

[**Note**: Add or delete items as appropriate.]

|  |  |  |
| --- | --- | --- |
| **Human Resources** | | |
| **Role** | **Minimum Resources Recommended**  **(number of full-time roles allocated)** | **Specific Responsibilities or Comments** |
| Test Manager |  | Provides management oversight.  Responsibilities include:   * planning and logistics * agree mission * identify motivators * acquire appropriate resources * present management reporting * advocate the interests of test * evaluate effectiveness of test effort |
| Test Analyst |  | Identifies and defines the specific tests to be conducted.  Responsibilities include:   * identify test ideas * define test details * determine test results * document change requests * evaluate product quality |
| Test Designer |  | Defines the technical approach to the implementation of the test effort.  Responsibilities include:   * define test approach * define test automation architecture * verify test techniques * define testability elements * structure test implementation |
| Tester |  | Implements and executes the tests.  Responsibilities include:   * implement tests and test suites * execute test suites * log results * analyze and recover from test failures * document incidents |
| Test System Administrator |  | Ensures test environment and assets are managed and maintained.  Responsibilities include:   * administer test management system * install and support access to, and recovery of, test environment configurations and test labs |
| Database Administrator, Database Manager |  | Ensures test data (database) environment and assets are managed and maintained.  Responsibilities include:   * support the administration of test data and test beds (database). |
| Designer |  | Identifies and defines the operations, attributes, and associations of the test classes.  Responsibilities include:   * defines the test classes required to support testability requirements as defined by the test team |
| Implementer |  | Implements and unit tests the test classes and test packages.  Responsibilities include:   * creates the test components required to support testability requirements as defined by the designer |

1. **Iteration Milestones**

[Identify the key schedule milestones that set the context for the Testing effort. Avoid repeating too much detail that is documented elsewhere in plans that address the entire project.]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Milestone** | **Planned Start Date** | **Actual Start Date** | **Planned End Date** | **Actual End Date** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |