

SIGPRE - Sistema de Gestión Presupuestaria

Plan de Pruebas

Histórico de Revisiones

| Fecha | Versión | Descripción | Autor |
|-----------|---------|------------------|------------------------|
| 10/1/2008 | 1.0 | Borrador | Roberto López Hinojosa |
| 3/11/2008 | 1.1 | Tipos de pruebas | Roberto López Hinojosa |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

Tabla de Contenidos

| | |
|--|------------|
| <i>Histórico de Revisiones</i> | <i>ii</i> |
| <i>Tabla de Contenidos</i> | <i>iii</i> |
| 1 Introducción | 4 |
| 1.1 Propósito | 4 |
| 1.2 Alcance | 4 |
| 1.3 Referencias | 4 |
| 2 Estrategia de Pruebas | 4 |
| 2.1 Tipos de Pruebas | 4 |
| 2.1.1 Pruebas de Integridad de Datos | 4 |
| 2.1.3 Pruebas del Sistema | 5 |
| 2.1.4 Pruebas del ciclo del negocio | 5 |
| 2.1.5 Pruebas de Interfaz de Usuario | 6 |
| 2.1.7 Pruebas de Desempeño | 7 |
| 2.1.8 Pruebas de Seguridad y Control de Acceso | 8 |
| 3 Herramientas | 9 |
| 4 Recursos | 9 |
| 5 Entregables | 9 |

1 Introducción

1.1 Propósito

El presente documento describe el Plan de Pruebas de la herramienta de software Sistema de Gestión Presupuestaria.

1.2 Alcance

Este Plan de Pruebas aplica a todos los componentes necesarios para registrar, modificar la proforma presupuestaria y registrar las transacciones presupuestarias durante el proceso de ejecución presupuestaria.

1.3 Referencias

- Plan de Iteración

2 Estrategia de Pruebas

En la sección anterior describe QUE será probado; La estrategia define *cómo* será realizarán las pruebas.

Las consideraciones principales para la estrategia de prueba son las técnicas a ser usadas y el criterio para saber cuando las pruebas están completas.

2.1 Tipos de Pruebas

2.1.1 Pruebas de Integridad de Datos

| | |
|------------------------------------|--|
| Objetivo: | Asegurar la integridad de los datos |
| Técnica: | Registrar datos validos Inspeccionar el esquema de base de datos para asegurarse que los datos se han guardado satisfactoriamente y de acuerdo a los estándares definidos |
| Criterio de completitud: | Todos los métodos de acceso y procesos de la Base de datos funcionan como fueron diseñados y sin corrupción de datos |
| Consideraciones Especiales: | <ul style="list-style-type: none">• Las pruebas pueden requerir un ambiente de DBMS o controladores para ingresar o modificar datos directamente en la Base de datos.• Se debe utilizar un conjunto pequeño de datos para incrementar la visibilidad de cualquier evento anormal o inesperado.• Los procesos pueden ser invocados manualmente. |

2.1.3 Pruebas del Sistema

| | |
|------------------------------------|---|
| Objetivo: | Asegurar la apropiada navegación dentro del sistema, ingreso de datos, procesamiento y recuperación. |
| Técnica: | Ejecute cada caso de uso, flujo básico o función utilizando datos válidos e inválidos, para verificar que: <ul style="list-style-type: none"> • Los resultados esperados ocurren cuando se utiliza un dato válido. • Los mensajes de error o de advertencia aparecen en el momento adecuado, cuando se utiliza un dato inválido. • Cada regla de negocios es aplicada adecuadamente. |
| Criterio de completitud: | <ul style="list-style-type: none"> • Todas las pruebas planeadas han sido ejecutadas. • Todos los defectos que se identificaron han sido tenidos en cuenta. |
| Consideraciones Especiales: | Identifique o describa aquellos aspectos (internos o externos) que impactan la implementación y ejecución de las pruebas del Sistema |

2.1.4 Pruebas del ciclo del negocio

| | |
|----------------------------------|--|
| Objetivo: | Asegurar que el sistema funciona de acuerdo con el modelo de negocios emulando todos los eventos en el tiempo y en función del tiempo. |
| Descripción de la Prueba: | Las pruebas del ciclo de negocio deberían emular las actividades ejecutadas en el a través del tiempo. Debería identificarse un periodo, como por ejemplo un año, y las transacciones y actividades que podrían ocurrir durante un periodo de un año deberían ejecutarse. Incluyendo todos los ciclos y eventos diarios, semanales y mensuales que sean datos sensitivos, como las agendas. |
| Técnica: | Ejecute cada caso de uso, flujo básico o función utilizando datos válidos e inválidos, para verificar que: <ul style="list-style-type: none"> • Incremente el número de veces en que una función es ejecutada para simular diferentes usuarios sobre un periodo especificado • Todas las fechas o funciones que involucren tiempos serán probadas con datos válidos e inválidos de fechas o periodos de tiempo. • Todas las funciones ocurren en un periodo de tiempo serán ejecutadas en el tiempo apropiado. • Los resultados esperados ocurren cuando los datos válidos son usados. • Los mensajes de error o de advertencia aparecen en el momento adecuado, cuando se utiliza un dato inválido. • Cada regla de negocios es aplicada adecuadamente. |

| | |
|------------------------------------|---|
| Criterio de completitud: | <ul style="list-style-type: none"> • Todas las pruebas planeadas han sido ejecutadas. • Todos los defectos que se identificaron han sido tenidos en cuenta. |
| Consideraciones Especiales: | <ul style="list-style-type: none"> • Las fechas y eventos del sistema pueden requerir actividades especiales de soporte. • Se requiere un modelo de negocios para identificar requisitos y procedimientos de prueba apropiados. |

2.1.5 Pruebas de Interfaz de Usuario

| | |
|------------------------------------|---|
| Objetivo: | <p>Verifica lo siguiente:</p> <ul style="list-style-type: none"> • La navegación a través de los objetos de la prueba reflejan las funcionalidades del negocio y requisitos, se realiza una navegación forma por forma, usando los modos de acceso (tabuladores, movimientos del mouse, teclas rápidas, etc.) • Los objetos de las formas y características, tales como árbol de opciones de menú, medidas, posiciones, estados y focos se verifican conforme a los estándares. |
| Descripción de la Prueba: | <p>La prueba de interfaz de usuario verifica la interacción del usuario con el software. El objetivo es asegurar que la interfaz tiene apropiada navegación a través de las diferentes funcionalidades. Adicionalmente, las pruebas de interfaz aseguran que los objetos de la interfaz a ser probada se encuentra dentro de los estándares definidos, en este caso, estándares de oracle forms</p> |
| Técnica: | <p>Se selecciona un grupo de usuarios que ponen a trabajar el sistema en un ambiente real. Usan el sistema en sus actividades cotidianas, procesan transacciones y producen salidas normales del sistema. Las transacciones y personas que usan el sistema son reales y trabajan en su área de trabajo real. El desarrollador no está presente. Los usuarios están advertidos de que están usando un sistema que puede fallar. Los usuarios realizan pruebas a su antojo realizando uso de la aplicación.</p> |
| Criterio de completitud: | <p>Se establece un periodo de pruebas beta en el que los errores detectados no sean de carácter crítico para el sistema.</p> |
| Consideraciones Especiales: | <p>Se deben considerar mecanismos de comunicación entre los desarrolladores y los usuarios de manera que los errores detectados puedan ser corregidos.</p> |

2.1.7 Pruebas de Desempeño

| | |
|----------------------------------|--|
| Objetivo: | <p>Validar el tiempo de respuesta para las transacciones o funciones de negocios bajo las siguientes dos condiciones:</p> <ul style="list-style-type: none"> • Volumen normal anticipado • Volumen máximo anticipado. |
| Descripción de la Prueba: | <p>Las pruebas de desempeño miden tiempos de respuesta, índices de procesamiento de transacciones y otros requisitos sensibles al tiempo. El objetivo de las pruebas de desempeño es verificar y validar los requisitos de desempeño que se han especificado</p> <p>Las pruebas de desempeño usualmente se ejecutan varias veces, utilizando en cada una, carga diferente en el sistema. La prueba inicial debe ser ejecutada con una carga similar a la esperada en el sistema. Una segunda prueba debe hacerse utilizando una carga máxima.</p> <p>Adicionalmente, las pruebas de desempeño pueden ser utilizadas para perfilar y refinar el desempeño del sistema como una función de condiciones tales como carga o configuraciones de hardware</p> <p>Las principales actividades son:</p> <ul style="list-style-type: none"> • Comparar el desempeño del sistema actual con los requisitos, • Poner a punto el sistema para mejorar las métricas de desempeño y proyectar la capacidad futura de carga del sistema. <p>Los objetivos de nivel de servicio definidos deben guiar la prueba de performance. Algunas características que afectan el desempeño son:</p> <ul style="list-style-type: none"> • Errores lógicos • Procesamiento ineficiente • Diseño pobre: muchas interfaces, instrucciones y entradas/salidas. • Cuellos de botella en discos, CPU ó canales de entrada/salida • Salidas del sistema • Tiempos de respuesta • Capacidad de almacenamiento • Tasa de entrada/salida de datos • Número de transacciones que pueden ser manejadas simultáneamente. <p>Las pruebas de desempeño utilizan las técnicas de caja blanca y caja negra.</p> |

| | |
|------------------------------------|---|
| Técnica: | <p>Se selecciona un grupo de usuarios que ponen a trabajar el sistema en un ambiente real. Usan el sistema en sus actividades cotidianas, procesan transacciones y producen salidas normales del sistema. Las transacciones y personas que usan el sistema son reales y trabajan en su área de trabajo real. El desarrollador no está presente. Los usuarios están advertidos de que están usando un sistema que puede fallar. Los usuarios realizan pruebas a su antojo realizando uso de la aplicación.</p> |
| Criterio de completitud: | Se establece un periodo de pruebas beta en el que los errores detectados no sean de carácter crítico para el sistema. |
| Consideraciones Especiales: | Se deben considerar mecanismos de comunicación entre los desarrolladores y los usuarios de manera que los errores detectados puedan ser corregidos. |

2.1.8 Pruebas de Seguridad y Control de Acceso

| | |
|------------------------------------|--|
| Objetivo: | <p>Seguridad de Función y Datos: Verificar que los usuarios puedan acceder solo a aquellas funciones y datos para los cuales se le ha concedido permiso de acuerdo a su perfil</p> <p>Seguridad del Sistema: Verificar que solo aquellos usuarios con permisos puedan acceder las opciones del sistema..</p> |
| Descripción de la Prueba: | <p>Las pruebas se enfocan en dos áreas principales de seguridad:</p> <ul style="list-style-type: none"> • Seguridad de Aplicación, incluyendo en acceso a los datos o funciones del negocio, y • Seguridad del Sistema, que incluye el rastreo de accesos al sistema |
| Técnica: | <ul style="list-style-type: none"> • Seguridad de Datos y Funciones: Identificar y listar cada tipo de usuario y los datos/funciones a los que tiene permiso de acceso. • Crear pruebas para cada tipo de usuario y verificar los permisos creando transacciones específicas para cada tipo de usuario. • Modificar el tipo de usuario y ejecutar las mismas pruebas para los mismos usuarios. En cada caso verificar si las funciones o datos están correctamente disponibles o denegadas. |
| Criterio de completitud: | Para cada tipo de usuario están disponibles los datos y funciones apropiadas |
| Consideraciones Especiales: | El acceso al sistema debe ser revisado con el administrador de la red y de la base de datos. Probablemente estas pruebas no sean necesarias pues si son tareas de el administrado de la red. |

3 Herramientas

| | Herramienta |
|------------------------------|--|
| Registro de Defectos | MS Word |
| Otras herramientas de prueba | SQLPLUS |
| Gestión del Proyecto | Microsoft Project Microsoft Word Microsoft Excel |
| Herramientas DBMS | TOAD |

4 Recursos

La siguiente tabla muestra los colaboradores que se encargarán de la realización de las pruebas:

| Recursos Humanos | | |
|--------------------------------|---|--|
| Rol | Recursos Requeridos | Responsabilidades específicas/comentarios |
| Administrador de Pruebas | Roberto López | Proveer las directrices de las pruebas Responsabilidades: <ul style="list-style-type: none"> • Proveer dirección técnica • Adquirir los recursos necesarios |
| Diseñador de Pruebas | Roberto López | Identificar, priorizar e implementar los casos de pruebas Responsabilidades: <ul style="list-style-type: none"> • Generar el plan de pruebas • Evaluar la efectividad del esfuerzo en las pruebas |
| Evaluador del Sistema | Elena Flores Eduardo Lara Lorena Guzmán Santiago Angamarca | Ejecutar las pruebas. Responsabilidades: <ul style="list-style-type: none"> • Ejecutar pruebas • Registrar resultados • Documentar defectos |
| Administrador de Base de datos | Roberto López | Asegurar los datos de prueba (base de datos) Responsabilidades: <ul style="list-style-type: none"> • Administrar los datos de prueba (base de datos) |

5 Entregables

Los entregables de las actividades de pruebas definidas en este plan se describen en la siguiente tabla:

| Entregable | Propietario | Revisión / Distribución |
|-----------------------------|---------------|--------------------------|
| Plan de Pruebas | Roberto López | Coordinador del Proyecto |
| Conjunto de datos de prueba | Eduardo Lara | Responsable del Proyecto |

PROYECTO SIGPRE

| Entregable | Propietario | Revisión / Distribución |
|---|--------------------|--------------------------------|
| Scripts de Prueba | Roberto López | - |
| Reportes de defectos | Roberto López | Responsable del Proyecto |
| Resultado de Pruebas | Roberto López | - |
| Reporte de Evaluación de Pruebas | Roberto López | Responsable del Proyecto |