**Plan de Pruebas de Software**

***Sistema Control de Combustible***

***Fecha: 29/05/2015***

**Tabla de contenido**

Historial de Versiones 4

Información del Proyecto 4

Aprobaciones 4

Resumen Ejecutivo 5

Alcance de las Pruebas 5

Elementos de Pruebas 5

Nuevas Funcionalidades a Probar 6

Pruebas de Regresión 6

Funcionalidades a No Probar 7

Enfoque de Pruebas (Estrategia) 7

Criterios de Aceptación o Rechazo 8

Criterios de Aceptación o Rechazo 8

Criterios de Suspensión 8

Criterios de Reanudación 9

Entregables 9

Recursos 10

Requerimientos de Entornos – Hardware 10

Requerimientos de Entornos – Software 10

Herramientas de Pruebas Requeridas 11

Personal 11

Entrenamiento 12

Planificación y Organización 12

Procedimientos para las Pruebas 12

Matriz de Responsabilidades 13

Cronograma 13

Premisas 14

Dependencias y Riesgos 14

Referencias 15

Glosario 15

# Historial de Versiones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| **29-05-2015** | **1.0** | **Ivan Linares Guerrero** | S.U.G.A. | **Versión N° 1** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Información del Proyecto

|  |  |
| --- | --- |
| Empresa / Organización | S.U.G.A. |
| Proyecto | “Sistema control de combustible” |
| Fecha de preparación | 29-05-2015 |
| Cliente | S.U.G.A. |
| Patrocinador principal |  |
| Gerente / Líder de Proyecto | Ivan Linares Guerrero |
| Gerente / Líder de Pruebas de Software | Ivan Linares Guerrero |

# Aprobaciones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nombre y Apellido** | **Cargo** | **Departamento u Organización** | **Fecha** | **Firma** |
| **Ivan Linares Guerrero** | **GP/LP** | S.U.G.A. | **29-05-2015** | **ilinaresg** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Resumen Ejecutivo

En el desarrollo de software, la fase de pruebas es crítica para asegurar que el producto sea enviado a ambiente de producción con la calidad esperada por el cliente.

Por esa razón se realizaran las pruebas unitarias y pruebas de carga al Sistema control de combustible.

# Alcance de las Pruebas

## Elementos de Pruebas

|  |  |  |
| --- | --- | --- |
| **Módulos** | **Componentes** | **Nivel** |
| Generar Orden de Abastecimiento | * Generar automáticamente el N° de Orden de Salida * Generar automáticamente la fecha * Seleccionar chofer * Seleccionar Unidad de Transporte * Datos de Abastecimiento | Alto |
| Generar Orden de Salida | * Generar automáticamente el N° de Orden de Salida * Generar automáticamente la fecha * Seleccionar el chofer * Seleccionar la unidad de transporte * Seleccionar la ruta * Ingresar los datos del cliente | Alto |

## Nuevas Funcionalidades a Probar

No aplica

## Pruebas de Regresión

No aplica

## Funcionalidades a No Probar

## Enfoque de Pruebas (Estrategia)

Se realizara Pruebas Unitarias y Pruebas de carga en los módulos de Generar Orden de Abastecimiento y Generar Orden de Salida.

Pruebas Unitarias

|  |  |
| --- | --- |
| Objetivos: | Se evaluarán las líneas de código de cada módulo, verificando que los métodos devuelvan los valores esperados. |
| Técnicas: | Se utilizará la herramienta Subversión |
| Criterios de finalización: | Los resultados esperados sean iguales a los obtenidos. Debe simplificar la integración. Localizar los errores, para su posterior corrección. |

Pruebas de Carga

|  |  |
| --- | --- |
| Objetivos: | Medir el rendimiento y la velocidad en la que sistema carga genera una orden de abastecimiento |
| Técnicas: | Se utilizará la herramienta Test Studio |
| Criterios de finalización: | Éxito en la carga de la información a la base de datos. |

# Criterios de Aceptación o Rechazo

## Criterios de Aceptación o Rechazo

|  |  |  |
| --- | --- | --- |
| **Modulo** | **Criterio de Aceptación** | **Criterio de Rechazo** |
| Generar Orden de Abastecimiento | Que genere la orden de abastecimiento con número xxxxx | Que no se genere la Orden de Abastecimiento |
| Generar Orden de Salida | Que genere la orden de salida con número xxxxx | Que no se genere la orden de Salida |

|  |  |  |
| --- | --- | --- |
| **CRITERIO** | **VALOR-ACEPTACION** | **VALOR-RECHAZO** |
| Porcentaje de las pruebas realizadas resultados exitosos | 90-100% | < 90% |
| Total de defectos encontrados que han sido solucionados | 90% | < 90% |

## Criterios de Suspensión

Que no se tenga acceso a la base de datos.

## Criterios de Reanudación

Que se logre la conexión a la base de datos.

# Entregables

# Los entregables definidos para este plan de pruebas están conformados por:

# Documentos de Especificación de Casos de Uso de Prueba: Plasmará los diferentes caminos y el procedimiento de prueba para cada uno de los módulos.

# Logs de errores: Indicará el registro de errores el cual se generara en eclipse.

# Recursos

## Requerimientos de Entornos – Hardware

* Requerimientos Mínimos

|  |  |
| --- | --- |
| Requerimientos Mínimos | |
| Recurso | Descripción |
| Equipo | Procesador 2.7 GHz  6.00 GB Ram 500 GB HDD (Disco Duro) |
| Servidor | Servidor local Tomcat Apache |

* Equipo Usado para las pruebas

|  |  |
| --- | --- |
| Equipo usado para las pruebas | |
| Recurso | Nombre / Tipo |
| Equipo | Intel Core i5, 2.7 GHz, 6.00 GB Ram 500 GB HDD (Disco Duro) |
| Servidor utilizado | Servidor local Tomcat Apache |

## Requerimientos de Entornos – Software

|  |  |  |
| --- | --- | --- |
| **SOFTWARE** | **PRECIO** | **REQUERIMIENTOS** |
| Windows 7  http://oazeek.com/wp-content/uploads/2012/05/Windows-7.png | Software con licencia | Memoria RAM: 512MB (mínimo).  Espacio en Disco (mínimo): 4 GB  Procesador: desde Pentium 4. |
| Tomcat 7 | Software Libre | Memoria RAM: 520 MB (mínimo)  Procesador: desde Pentium 4. |
| MySql 5.6 | Software Libre | Memoria RAM: 512MB (mínimo).  Espacio en Disco (mínimo): 4GB  Procesador: desde Pentium 4. |
| JEE Indigo | Código Abierto | Memoria RAM: 512MB (mínimo).  Espacio de disco (mínimo): 4GB  Procesador: desde Dual Core |
| C:\Users\IVAN\Desktop\github-logo.jpg | Software Libre | Memoria RAM: 512 MB (mínimo)  Espacio en disco (mínimo): 4 GB  Memoria RAM: 512 MB (mínimo)  Espacio en disco (mínimo): 4 GB  Procesador: desde Dual Core |
| Resultado de imagen para mysql workbench 6.2 | Software Libre | Memoria RAM: 512 MB (mínimo)  Espacio en disco (mínimo): 4 GB  Procesador: desde Dual Core |

## Herramientas de Pruebas Requeridas

Herramienta de Automatización de Pruebas

Test Studio

* La herramienta de pruebas de carga Telerik Prueba Studio le permite obtener una mejor comprensión de cómo su sitio web podría realizar si la visita de un gran número de usuarios simultáneos. Esto le ayuda a evaluar si sus aplicaciones web cumplen con las necesidades del negocio de la disponibilidad y la satisfacción del usuario. Prueba Studio hace que sea fácil para que usted pueda empezar y encontrar los datos que necesita para tomar decisiones informadas. También tienen el poder con la flexibilidad para crear elaborados, escenarios de carga complejas para satisfacer sus necesidades más exigentes.

Software de Gestíon de pruebas

Subversion

* Para la administración de los archivos versionados, *Subversion* maneja los conceptos de “repositorio” y “copia de trabajo”. Por repositorio se entiende aquel espacio, común y compartido por el equipo desarrollador, en el que se conservan todas y cada una de las versiones de los archivos producidos durante el ciclo de vida del proyecto, mientras que la copia de trabajo es aquel espacio perteneciente al entorno local de cada desarrollador, sobre el cual posee dominio total y que necesita periódicamente sincronizarse con el repositorio. Esta sincronización se traduce en las tres operaciones básicas de la herramienta:
* Importar (*Checkout*): a través de este comando se descarga del repositorio una copia exacta de sus contenidos, tanto de los archivos como de su estructura. Es posible descargar la última versión o versiones más antiguas.
* Actualizar (*Update*): esta operación permite descargar únicamente aquellas modificaciones realizadas al repositorio, ocurridas desde la última sincronización con la copia de trabajo.
* Confirmar (*Commit*): esta operación ocurre en vía inversa a las dos anteriores, ya que permite actualizar el contenido del repositorio con los cambios realizados en la copia de trabajo.

## Personal

|  |  |
| --- | --- |
| Personal de pruebas | |
| Cargo | Nombre |
| 1. Lider de Pruebas     Nombre del servidor  Nombre de la base de datos | Ivan Linares Guerrero |
| 1. Analista de Pruebas |
| 1. Analista de Pruebas |

## Entrenamiento

* Subversión
* Test Studio

# Planificación y Organización

## Procedimientos para las Pruebas

1. Planeación de Pruebas: Se definirá el alcance de las pruebas, los tipos de prueba a emplear, los criterios de salida, la estrategia a seguir dado que pueden existir funcionalidades que requieran una mayor dedicación. Realizar también estimación de tiempos, asignación de roles y prepara el entorno de pruebas.
2. Diseño de Pruebas: Se deberá analizar la documentación existente para así poder definir un diseño de pruebas, definir los tipos de casos de prueba y los datos de pruebas necesarios.
3. Implementación de Pruebas: Ejecutar de acuerdo a como han sido diseñadas, la implementación se llevará a cabo de manera automatizada utilizando herramientas de pruebas. Una vez localizados los errores se procede a la corrección de los mimos y a realizar una prueba de regresión para confirmar si es que todo sigue en orden luego de los cambios realizados.
4. Evaluación de Criterios de Salida: Evaluar según lo definido en Planeación las salidas de las pruebas, si es que los resultados obtenidos no alcanzan a los esperados no será posible continuar con la siguiente etapa.
5. Cierre del proceso de pruebas: Se deben finalizar los entregables, cerrar las incidencias reportadas, evaluar y analizar las lecciones aprendidas.

## Matriz de Responsabilidades

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Proyecto:** | **Sistema control de combustible** | | | | |
| **Producto o Entregable** | | **Responsable** | | | |
| **R** | **A** | **C** | **I** |
| Elaboración del Plan de Pruebas | | Ivan Linares Guerrero | | | |
| Elaboración de Pruebas de Carga | |
| Elaboración de Pruebas Unitarias | |
| Análisis de los Log de Errores y Reportes de la Herramientas | |
| Elaboración del Reporte de Resultado de las Pruebas | |

## Cronograma

|  |  |  |  |
| --- | --- | --- | --- |
| Actividad | 09/05/15 | 16/05/15 | 23/05/15 |
| Capacitación Herramienta P. Unitaria | X |  |  |
| Capacitación Herramienta P. Funcionales |  | X |  |
| Capacitación Herramienta P. Carga |  |  | X |

## Premisas

Se tiene como premisas que:

* El equipo conoce detalladamente el proyecto, así como las funcionalidades de cada modulo.

## Dependencias y Riesgos

Aquí se listan los riesgos asociados con el proceso de pruebas de software.

* Dependencias con Desarrollos.
* Dependencias con otros proyectos.
* Disponibilidad de recursos.
* Restricciones de tiempo.

# Referencias

Lista de todos los documentos que pueden citarse como apoyo o para ampliar el contenido del plan de pruebas. Algunos ejemplos de lo que se puede hacer referencia aquí son:

* Plan de Proyecto.
* Especificaciones de Requerimientos.
* Diagrama de Secuencias.
* Especificación de Caso de Uso.
* Diagrama de Casos de Uso.

# Glosario

* Performance: Rendimiento
* Módulo: Elemento con función propia concebido para poder ser agrupado de distintas maneras con otros elementos constituyendo una unidad mayor.