**Plan de pruebas de software**

***[Angeles Scheduler]***

***Fecha: [16/11/2024]***

**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** |
| --- | --- | --- | --- | --- |
| 16/11/2024 | 0.1 | Felipe Concha |  | Modificación de plan de pruebas. |

**Aprobaciones**

| **Nombre y Apellido** | **Cargo** | **Fecha** | **Firma** |
| --- | --- | --- | --- |
| Felipe Concha | Jefe de Proyecto | 16/11/2024 | X |

**Resumen ejecutivo**

Análisis de un software utilizado para la creación y gestión de ventas, agenda y personal, realizando pruebas e inspecciones de las funcionalidades del sistema y su correcto uso. además de su estructuración o forma.

Se llevarán a cabo reuniones para estructurar hipótesis y levantamiento de información para realizar la planificación de un plan de pruebas para obtener los resultados esperados planteados en las hipótesis y así poder comparar los resultados obtenidos, utilizando las herramientas y estructuras necesarias.

Todos los resultados se estructuraron en una planilla excel con su categorización y clasificación para poder interpretar los datos de la manera correcta y más clara para su entendimiento.

Se mostrarán las pruebas realizadas de manera ordenadas y clasificadas por su importancia, además de mostrar los resultados de estas pruebas.

**Alcance de las pruebas**

**Elementos de pruebas**

El objetivo de las pruebas para evaluar los módulos por separado será para medir el rendimiento de ellas y la estabilidad de la aplicación web. Se utilizará Jmeter para generar las condiciones de carga a su máxima capacidad para asegurar que los módulos y componentes puedan manejar el tráfico esperado sin degradarse en los términos de tiempo de respuesta o disponibilidad.

Áreas de prueba:

* **Funcionalidad del sistema:** Validación del sistema realiza tareas básicas, como los registros de ventas.
* **Desempeño:** Evaluación de la capacidad de los módulos clave para soportar la carga esperada sin tiempos excesivos en respuesta.
* **Escalabilidad:** Comprobación de cómo el sistema responde al aumento de la carga durante un día de uso para identificar las horas de picks de uso.
* **Resiliencia:** Análisis de cómo se comporta el sistema ante fallos imprevistos para documentarlo y asegurarse que no haya pérdidas de datos.

**Nuevas funcionalidades a probar**

**Las funciones que se darán a probar son las siguientes:**

Administrar a los usuarios del sistema

* Agregar usuarios
* Modificar usuarios
* Eliminar usuarios
* Agregar clientes
* Modificar clientes
* Eliminar clientes
* Crear Venta
* Modificar Venta
* Eliminar Venta
* Gestión de roles de usuarios
* Controlador de autenticación
* Cálculos de comisiones de ventas
* Cálculos de comisiones de servicios
* Agregar Citas
* Modificar Citas
* Eliminar Citas
* Visualizar Citas
* Seguimientos de citas

**Pruebas de regresión**

El objetivo de este plan es asegurar que las funcionalidades existentes dentro de la plataforma no se vean afectadas tras realizar cambios, correcciones o mejoras en el código. Las pruebas de regresión son fundamentales para garantizar que el sistema sigue funcionando correctamente en su conjunto y que los errores no impacten en el comportamiento general. Es decir que en el caso de que se suba alguna actualización, mejora de algún componente o solución de algún bug se tendrá un espacio de trabajo para poder hacer estos cambios indirectamente sin que afecte al sistema general.

**El alcance de las pruebas de regresión toma en cuenta los siguientes casos:**

* Modificaciones en cualquier componente o módulo existente dentro del sistema.
* Corrección de error o bugs dentro de la plataforma.
* Mejoramientos de componentes o módulos dentro de la plataforma.
* Actualización de librerías o dependencias.

**Metodologías a seguir en las pruebas de regresión sería las siguientes:**

Determinar funcionalidades, componentes o módulos dentro de la plataforma que se han modificado. identificar las dependencias, librerías o framework que se actualizan o se han actualizado e identificar si han provocado algún conflicto dentro de la plataforma o si el rendimiento de algún componente o módulo está siendo afectado. Estas tareas serían las siguientes:

* **Pruebas unitarias:** Verificar si las funcionalidades modificadas funcionan correctamente de manera aislada.
* **Pruebas de integración:** asegurar que la integración de nuevos módulos o servicios se realicen de manera correcta.
* **Pruebas de aceptación:** Validar que las funcionalidades sigan cumpliendo con los requerimientos del usuario.
* **Pruebas de rendimiento:** Evaluar que el rendimiento del sistema no se vea afectado.

Luego de las pruebas y las ejecuciones de los nuevos cambios se revisa los resultados, se hace un análisis general de las pruebas documentando las y dándole el cierre.

**Funcionalidades a no probar**

Modificacion de usuario Usuarios

Razón 1: El Módulo para modificar el usuario desde el lanzamiento de la plataforma no se puede modificar por el hecho de que esto conlleva un cambio radical en cómo está diseñada tal componente, como está diseñado la tabla asociada a los usuario y como está implementada dentro del sistema.

Razón 2: La dependencia entre tablas/relaciones y llaves foráneas, puede generar un conflicto en el registro histórico de los usuarios registrados.

**Enfoque de pruebas (estrategia)**

El enfoque se compondría por las pruebas de unidad que verifiquen el comportamiento individual. Pruebas de integración que verifique la integración de diferentes componentes y módulos, pruebas funcionales que verifique las funcionalidad esperados por el usuario. pruebas de sistema que validan el completo funcionamiento como un todo. Pruebas no funcionales que evalúen el desempeño, usabilidad y mantenibilidad, pruebas de validaciones que aseguren que el sistema cumpla con los requisitos y expectativas del cliente.

**Criterios de aceptación o rechazo**

**Criterios de aceptación o rechazo**

Se espera que la cobertura de pruebas unitarias se complete y ejecuten al 100%, al menos un 90% en la tasa de éxito de casos y un 10% de fallos, cobertura del código en un 80% o más, pruebas de regresión de un 90% de éxito y un 10% de fallos. Las pruebas funcionales corroboradas es de un 90% de éxito y solamente un 10% como criterio de aceptación.

**Criterios de suspensión**

Se establece como criterio de suspensión en el caso de que alguna prueba unitaria presente un comportamiento anómalo en su funcionamiento o fallo de vulnerabilidad en la fase de pruebas de tal módulo o componente. En este caso se analiza el fallo y se documenta además que se comunica con el equipo de Testing el problemas que se presentó para ver la acción a tomar para solucionar eventuales fallos.

**Criterios de reanudación**

* **Revalidación** de la funcionalidad antes de hacer las pruebas, verificar que las funcionalidades que causó la suspensiones ahora funcionen.
* **Pruebas incrementales** si una parte del sistema fue suspendida debido a un problema, no es necesario reiniciar todas las pruebas desde cero.
* **Informe de reanudación documentando** los motivos de las suspensiones, los cambios realizados y cómo se validan las correcciones de los problemas.

**Entregables**

* Planilla excel casos de prueba
* carta gantt

**Recursos**

**Requerimientos de entornos – Hardware**

* Se requiere acceso a la aplicación Web Angeles.
* Velocidad de internet de 20 mbps.
* Acceso a base de datos de servicio Ángeles Scheduler.

Requisitos hardware Computador

* CPU: Intel: Core i5-3300. AMD: Ryzen 3 1200. ...
* Características del procesador. SSE4.
* Memoria gráfica: 1GB.
* Espacio de almacenamiento disponible. 1 GB de SSD/HDD.
* Arquitectura del SO: x64./ x32.

**Requerimientos de entornos – Software**

* Gráficos integrados ,Nivel del hardware: DX11.
* Versión del SO: Win 10,win 8,win 8.1

**Herramientas de pruebas requeridas**

* CASO DE PRUEBAS (Unitarias) MANUALES por testers

**Personal**

* Analistas de prueba (testers): 2

**Entrenamiento**

* No aplica

**Planificación y organización**

**Matriz de responsabilidades**

| **Fase 4: Pruebas** | | | | |
| --- | --- | --- | --- | --- |
| Plan de pruebas | ED | JP | C | ED |
| Caso de pruebas | ED | JP | C | ED |

| **Sigla** | **Rol** | | **Nombre** | |
| --- | --- | --- | --- | --- |
| JP | Jefe de proyecto |  | Felipe Concha | |
| ED | Equipo de desarrollo | | Cristian Molina | |
| Jimmy Muñoz | |
| Felipe Concha | |
| C | Cliente | | Carmen Mardones | |

**Dependencias y Riesgos**

* Premisas que resultan no ser ciertas.
* Falta de procesos en el sistema, incluidos en el plan de pruebas.
* Conjeturas no estructuradas
* No tener conocimiento del uso del software
* No generar una lluvia de ideas con hipótesis

**Referencias**

* Plan de proyecto.
* Especificaciones de requerimientos.

**Glosario**

* **Reporte**: Documentos que contienen los detalles de un evento específico, por ejemplo, la visita a un sitio (museo, empresa, fábrica, etc.), o bien, sobre un documental, película, etc. Este escrito pretende dar respuesta a una(s) pregunta(s) de indagación que ayuda a ampliar el conocimiento respecto a un tema.
* **Premisa**: Resultado de un razonamiento que puede ser verdadero o falso y que permite determinar una conclusión. Sin embargo, por el uso del término, también se emplea como sinónimo de objetivo o de principio moral.
* **Hipótesis**: Suposición hecha a partir de unos datos que sirve de base para iniciar una investigación o una argumentación.
* **Stock**: Conjunto de mercancías o productos que se tienen almacenados en espera de su venta o comercialización.
* **Cronograma**: Representación gráfica de un conjunto de hechos en función del tiempo.
* **Requerimientos**: Especifican qué es lo que el sistema debe hacer (sus funciones) y sus propiedades esenciales y deseables.
* **Software**: Conjunto de programas y rutinas que permiten a la computadora realizar determinadas tareas.
* **Hardware**: Conjunto de elementos físicos o materiales que constituyen una computadora o un sistema informático.