**Plan de Pruebas Capstone**

**“STOCKWISE”**

**Tabla de contenido**

[**Histórico de Revisiones 3**](#_heading=h.y8zikj8zy6tx)

[**Información del Proyecto 3**](#_heading=h.30j0zll)

[**Integrantes 3**](#_heading=h.36m13n6ggb42)

[**Propósito del plan de pruebas 4**](#_heading=h.3znysh7)

[**Alcance de las pruebas 4**](#_heading=h.fsywjdwf5t4g)

[**Definición de roles y responsabilidades 6**](#_heading=h.xlz5ctv1qx2e)

[**Tipos de pruebas a realizar 8**](#_heading=h.3dy6vkm)

[**Estrategia y técnicas de pruebas a aplicar 9**](#_heading=h.g5jf1pkaaqi6)

[**Definición del proceso de testing 11**](#_heading=h.r5jtnrns8c8z)

[**Definición de ciclos de prueba a ejecutar 12**](#_heading=h.ig855cuwinfw)

[**Calendarización de las actividades de pruebas 14**](#_heading=h.1t3h5sf)

[**Resumen de riesgos 15**](#_heading=h.4d34og8)

[**Clasificación de los defectos 17**](#_heading=h.2s8eyo1)

[**Definición de artefactos 17**](#_heading=h.p4632zgaikom)

[**Condiciones de aceptación para cierre del proceso de pruebas 18**](#_heading=h.ducpojr5669z)

**Histórico de Revisiones**

| Versión | Fecha | Descripción/cambio | autor |
| --- | --- | --- | --- |
| 1.0 | 10/12/2024 | Versión inicial | Javier Martínez |
|  |  |  |  |
|  |  |  |  |

**Información del Proyecto**

| Organización | Duoc UC. Escuela de Informática y Telecomunicaciones |
| --- | --- |
| Sección | Vespertino |
| Proyecto (Nombre) | Stockwise |
| Fecha de Inicio | 06/08/2024 |
| Fecha de Término | 16/12/2024 |
| Docente | Javier Saldaño |

**Integrantes**

| Nombre | Correo |
| --- | --- |
| **Javier Martinez** | **java.martinez@duocuc.cl** |
| **Martin Salazar** | **m.salazarw@duocuc.cl** |
| **Jorge Muñoz** | **jo.munoza@duocuc.cl** |

**Propósito del plan de pruebas**

| Propósito, objetivo, visión que se espera de este plan de pruebas. |
| --- |
| El propósito del plan de pruebas es establecer una estrategia para evaluar la calidad y el rendimiento de la aplicación Stockwise, antes de su lanzamiento o implementación. De esta manera, se busca garantizar que el producto final cumpla con los requisitos establecidos, sea confiable, funcional y proporcione una experiencia satisfactoria a los usuarios.  El objetivo principal del plan de pruebas es identificar y corregir cualquier defecto, error o mal funcionamiento antes de que la aplicación llegue a los clientes finales. Se pretende asegurar que todas las funcionalidades y características estén implementadas correctamente, optimizar el rendimiento del sistema y asegurar la calidad del producto final.  La visión del plan de pruebas es lograr un producto final que cumpla con altos estándares de calidad y brinde una experiencia excepcional a los usuarios. Es fundamental identificar y corregir cualquier problema antes de que afecte a los clientes y asegurar que la aplicación cumpla con sus expectativas y necesidades.  En última instancia, el plan de pruebas tiene como objetivo asegurar la entrega de una aplicación confiable, funcional y de calidad a los clientes de Stockwise. Al lograr esto, se espera que el plan de pruebas contribuya a la satisfacción del cliente, fortalezca la reputación de la empresa y contribuya al éxito de la aplicación en el mercado de servicios de entrega de productos. |

**Alcance de las pruebas**

| Definición de requisitos de S.W., módulos de Software a probar, Requisitos ambiente de pruebas y Documentación Referenciada, etc. |
| --- |
| El alcance de las pruebas abarca todas las funcionalidades críticas y principales de la aplicación de Stockwise, incluyendo:   * Registro de usuarios. * Registros de productos. * Alerta de caducidad. * Reporte de movimientos. * Análisis de procesos. * Interfaz de usuario y usabilidad. * Rendimiento y tiempo de respuesta. * Seguridad y protección de datos.   Los requisitos de software para la aplicación de Stockwise incluirán los siguientes elementos:   * Módulo de registro de usuarios: Permitirá a los usuarios crear cuentas y acceder al sistema. * Módulo de productos: Permitirá revisar el inventario de productos que se encuentran en los almacenes del sistema. * Módulo de reportes: Permite revisar los movimientos realizados por los productos en el sistema y exportar los resultados obtenibles. * Módulo de kpi: Mostrará los KPI del sistema. * Interfaz de usuario: Deberá ser intuitiva, atractiva y fácil de usar para los usuarios. * Rendimiento: La aplicación deberá ser eficiente y proporcionar tiempos de respuesta rápidos para garantizar una experiencia fluida. * Seguridad: Deberán implementarse medidas de seguridad adecuadas para proteger la información confidencial de los usuarios.   Módulos de software a probar:   * Módulo de registro de usuarios. * Módulo de productos. * Módulo de reportes. * Módulo de kpi. * Interfaz de usuario.   Requisitos de ambiente de pruebas:   * Entorno de pruebas separado del entorno de producción. * Datos de prueba realistas para simular la creación de usuarios, productos y reportes. * Réplica fiel del entorno de producción para garantizar resultados realistas. * Herramientas de prueba adecuadas y actualizadas.   Documentación Referenciada:   * Especificaciones de requisitos de software. * Plan de pruebas detallado. * Casos de prueba y escenarios de prueba. * Resultados de las pruebas y registros de defectos. * Informes de calidad y cumplimiento de requisitos. * Documentación de diseño de interfaz de usuario. * Documentación de arquitectura y componentes del sistema. |

**Definición de roles y responsabilidades**

| Roles y responsabilidades de todos los participantes en el **proceso de pruebas** de SW. | | |
| --- | --- | --- |
| Rol | Responsabilidades | Relevancia |
| Calidad y Testing | Diseñar y planificar las estrategias de pruebas.  Coordinar y supervisar el proceso de pruebas.  Establecer los criterios de aceptación y calidad del software.  Identificar y gestionar los riesgos asociados a las pruebas.  Evaluar y seleccionar las herramientas de prueba adecuadas.  Analizar los resultados de las pruebas y generar informes de calidad.  Colaborar con los analistas programadores y diseñadores para mejorar la calidad del software. | Alta  Alta  Alta  Alta  Media  Alta  Alta |
| Analista Programador | Comprender los requisitos del software y los casos de uso.  Diseñar y desarrollar las pruebas unitarias para validar las funcionalidades y lógica del código.  Depurar y corregir los errores identificados durante las pruebas unitarias.  Colaborar con el responsable de calidad y testing en la identificación de escenarios de prueba y datos de prueba.  Realizar pruebas de integración para verificar la interoperabilidad entre los componentes del sistema.  Documentar y registrar los defectos encontrados durante las pruebas.  Participar en la revisión de los resultados de las pruebas y en la resolución de problemas. | Alta  Alta  Alta  Alta  Alta  Alta  Alta |
| Diseñador | Colaborar con el responsable de calidad y testing para comprender los requisitos y criterios de aceptación del software.  Asegurarse de que la interfaz de usuario sea intuitiva, accesible y cumpla con los estándares de diseño.  Participar en las pruebas de usabilidad para evaluar la experiencia del usuario.  Proporcionar retroalimentación sobre la apariencia visual y la usabilidad de la aplicación durante el proceso de pruebas.  Colaborar en la solución de problemas relacionados con la interfaz de usuario identificados durante las pruebas. | Alta  Alta  Alta  Alta  Media |

**Tipos de pruebas a realizar**

| Definir el tipo de pruebas que se debe desarrollar para este proyecto, actividades y responsables. |
| --- |
| Tipo de pruebas:   1. Pruebas de funcionalidad:    * Verificar que todas las funciones y características de la aplicación funcionen correctamente, como el registro de usuarios, productos en línea, movimientos de ellos y reportes del sistema.    * Validar que los flujos de trabajo principales estén implementados sin errores y cumplan con los requisitos especificados. 2. Pruebas de usabilidad:    * Evaluar la facilidad de uso de la aplicación, como la navegación intuitiva, la claridad de las instrucciones, la legibilidad de la interfaz de usuario, etc.    * Validar que los elementos de la interfaz sean comprensibles y se ajusten a las expectativas de los usuarios. 3. Pruebas de rendimiento:    * Evaluar el rendimiento de la aplicación en términos de tiempo de carga, tiempos de respuesta, capacidad de manejo de cargas de trabajo, etc.    * Verificar que la aplicación sea eficiente y responda rápidamente a las acciones de los usuarios.   Actividades de pruebas:   1. Planificación de pruebas:    * Definir el alcance y los objetivos de las pruebas.    * Establecer los criterios de entrada y salida de cada prueba.    * Determinar los recursos necesarios, como entornos de prueba, datos de prueba, herramientas de prueba, etc. 2. Diseño de casos de prueba:    * Identificar y documentar los escenarios de prueba basados en los requisitos del sistema y los flujos de trabajo.    * Especificar los datos de prueba necesarios para cada caso de prueba. 3. Ejecución de pruebas:    * Realizar pruebas según los casos de prueba diseñados.    * Registrar los resultados de las pruebas y documentar cualquier defecto encontrado. 4. Seguimiento y gestión de defectos:    * Registrar y hacer un seguimiento de los defectos encontrados durante las pruebas.    * Comunicarse con el equipo de desarrollo para abordar y solucionar los problemas identificados.   Responsables de pruebas:   1. Calidad y Testing: Encargado de diseñar, ejecutar y gestionar las pruebas, así como de reportar los resultados y defectos encontrados. 2. Analista Programador: Colabora estrechamente con el ingeniero de pruebas para comprender los requisitos y solucionar los defectos identificados. 3. Diseñador: Proporciona retroalimentación sobre la usabilidad y la experiencia del usuario durante las pruebas de usabilidad. |

**Estrategia y técnicas de pruebas a aplicar**

| Definir las estrategias y técnicas de pruebas que se debe desarrollar para este proyecto, actividades y responsables. |
| --- |
| Estrategias de prueba   * Pruebas de unidad: Se centran en probar componentes individuales del software, como funciones específicas o módulos, para asegurar su correcto funcionamiento. Pueden ser realizadas por los analistas programadores responsables de cada componente. * Pruebas de integración: Verifican la interoperabilidad y la comunicación adecuada entre los diferentes componentes del sistema. Se pueden realizar pruebas de integración tanto horizontales como verticales. Los analistas programadores y el responsable de calidad y testing son responsables de estas pruebas. * Pruebas de sistema: Comprueban el comportamiento y el rendimiento del sistema en su conjunto. Se enfocan en probar los flujos de trabajo principales y garantizar que todas las funcionalidades están integradas correctamente. El responsable de calidad y testing, en colaboración con los analistas programadores, es responsable de estas pruebas. * Pruebas de aceptación: Se realizan para validar que el software cumple con los requisitos del cliente y las expectativas de los usuarios finales. Pueden incluir pruebas funcionales, pruebas de usabilidad y pruebas de rendimiento. El responsable de calidad y testing es el encargado principal de estas pruebas, en colaboración con los usuarios finales y el equipo de diseño.   Técnicas de pruebas:   * Pruebas funcionales: Se enfocan en verificar que el software cumpla con los requisitos funcionales especificados. Se pueden diseñar casos de prueba basados en los requisitos y escenarios de uso. El responsable de calidad y testing y los analistas programadores son responsables de estas pruebas. * Pruebas de usabilidad: Evalúan la facilidad de uso y la experiencia del usuario. Se pueden realizar pruebas de usabilidad con usuarios reales para recopilar comentarios y realizar mejoras. El diseñador y el responsable de calidad y testing son responsables de estas pruebas. * Pruebas de rendimiento: Evalúan el rendimiento y la capacidad del sistema bajo diferentes cargas y condiciones. Se pueden realizar pruebas de carga, pruebas de estrés y pruebas de rendimiento para identificar cuellos de botella y optimizar el rendimiento. El responsable de calidad y testing y los analistas programadores son responsables de estas pruebas. * Pruebas de seguridad: Verifican la protección de los datos y la seguridad del sistema. Se pueden realizar pruebas de penetración, pruebas de vulnerabilidad y análisis de seguridad para identificar posibles brechas de seguridad. El responsable de calidad y testing y el equipo de seguridad son responsables de estas pruebas.   Actividades y responsables de las pruebas:   * Planificación de pruebas: El responsable de calidad y testing es responsable de diseñar y planificar las estrategias de pruebas, así como de definir los casos de prueba y los criterios de aceptación. * Diseño y ejecución de pruebas: Los analistas programadores son responsables de diseñar y ejecutar las pruebas de unidad e integración, mientras que el responsable de calidad y testing lidera las pruebas de sistema y aceptación. * Registro y gestión de defectos: Los analistas programadores y el responsable de calidad y testing colaboran para registrar y gestionar los defectos encontrados durante las pruebas, así como para realizar un seguimiento y resolución de los mismos. * Seguimiento y control de las pruebas: El responsable de calidad y testing supervisa el progreso de las pruebas, monitorea los resultados y proporciona informes de calidad. |

**Definición del proceso de testing**

| Listar y describir todas las actividades a desarrollar en el proceso general de testing, responsables, artefactos, etc. |
| --- |
| * Planificación de pruebas:   + Descripción: Definir la estrategia de pruebas, los objetivos, los alcances y los recursos necesarios.   + Responsables: Responsable de calidad y testing.   + Artefactos: Plan de pruebas, estrategia de pruebas, matriz de trazabilidad de requisitos y casos de prueba. * Diseño de casos de prueba:   + Descripción: Diseñar casos de prueba detallados basados en los requisitos y escenarios de uso.   + Responsables: Responsable de calidad y testing, analistas programadores.   + Artefactos: Casos de prueba, matrices de trazabilidad de casos de prueba y requisitos. * Preparación del entorno de pruebas:   + Descripción: Configurar el entorno de pruebas con los sistemas, datos y herramientas necesarios.   + Responsables: Responsable de calidad y testing, analistas programadores.   + Artefactos: Configuración del entorno de pruebas, datos de prueba. * Ejecución de pruebas:   + Descripción: Realizar las pruebas según los casos de prueba diseñados, registrar los resultados y los defectos encontrados.   + Responsables: Responsable de calidad y testing, analistas programadores.   + Artefactos: Registro de resultados de pruebas, informes de defectos. * Análisis y gestión de defectos:   + Descripción: Analizar y clasificar los defectos encontrados, priorizar su resolución y hacer un seguimiento de su estado.   + Responsables: Responsable de calidad y testing, analistas programadores.   + Artefactos: Registro de defectos, informes de seguimiento de defectos. * Testing y verificación de correcciones:   + Descripción: Volver a ejecutar los casos de prueba relacionados con los defectos corregidos para verificar que se han solucionado correctamente.   + Responsables: Responsable de calidad y testing, analistas programadores.   + Artefactos: Registro de resultados de pruebas de testing, informes de verificación de correcciones. * Pruebas de regresión:   + Descripción: Realizar pruebas en áreas afectadas por las correcciones para asegurar que las modificaciones no hayan introducido nuevos defectos.   + Responsables: Responsable de calidad y testing.   + Artefactos: Casos de prueba de regresión, registro de resultados de pruebas de regresión. * Pruebas de aceptación:   + Descripción: Validar que el software cumple con los requisitos y las expectativas del cliente y los usuarios finales.   + Responsables: Responsable de calidad y testing, usuarios finales, equipo de diseño.   + Artefactos: Resultados de pruebas de aceptación, comentarios y retroalimentación de los usuarios. * Informes de pruebas y cierre:   + Descripción: Documentar los resultados de las pruebas, generar informes de calidad y presentarlos al equipo de desarrollo y a los stakeholders.   + Responsables: Responsable de calidad y testing.   + Artefactos: Informes de pruebas, informes de calidad, informes de cierre de pruebas. |

**Definición de ciclos de prueba a ejecutar**

| Listar y describir cantidad de ciclos de prueba a realizar en este proyecto, las tareas y actividades para cada ciclo de prueba, responsables, artefactos, etc. |
| --- |
| Ciclo de Prueba:  Ciclo de Prueba Inicial:  Tareas y Actividades:   * Planificación de pruebas: Definir la estrategia de pruebas y los objetivos del ciclo de prueba. * Diseño de casos de prueba: Crear casos de prueba basados en los requisitos y escenarios de uso. * Preparación del entorno de pruebas: Configurar el entorno de pruebas con los sistemas, datos y herramientas necesarios. * Ejecución de pruebas: Realizar pruebas funcionales, pruebas de usabilidad y pruebas de rendimiento. * Análisis y gestión de defectos: Registrar y gestionar los defectos encontrados durante las pruebas. * Testing y verificación de correcciones: Volver a ejecutar los casos de prueba afectados por los defectos corregidos. * Pruebas de regresión: Realizar pruebas en áreas afectadas por las correcciones para asegurar que no se hayan introducido nuevos defectos. * Pruebas de aceptación: Validar que el software cumpla con los requisitos y expectativas del cliente y usuarios finales. * Informes de pruebas y cierre: Documentar los resultados de las pruebas y generar informes de calidad.   Responsables:   * Responsable de calidad y testing. * Analistas programadores. * Usuarios finales (participación en las pruebas de aceptación).   Artefactos:   * Plan de pruebas. * Casos de prueba. * Registro de defectos. * Informes de pruebas.   Ciclos de Prueba Iterativos:  Tareas y Actividades:     * Diseño y ejecución de pruebas: Crear nuevos casos de prueba y ejecutar pruebas en áreas modificadas o nuevas funcionalidades. * Análisis y gestión de defectos: Registrar y gestionar los defectos encontrados durante las pruebas. * Testing y verificación de correcciones: Volver a ejecutar los casos de prueba afectados por los defectos corregidos. * Pruebas de regresión: Realizar pruebas en áreas afectadas por las correcciones para asegurar que no se hayan introducido nuevos defectos. * Pruebas de aceptación: Validar que las funcionalidades iterativas cumplan con los requisitos y expectativas.   Responsables:   * Responsable de calidad y testing. * Analistas programadores. * Usuarios finales (participación en las pruebas de aceptación).   Artefactos:   * Casos de prueba. * Registro de defectos. * Informes de pruebas. |

**Clasificación de los defectos**

| Definir la clasificación de los defectos según su nivel de severidad | |
| --- | --- |
| Nivel de Severidad | Descripción |
| **Crítica** | * Defectos que impiden por completo el uso o funcionamiento del sistema. * Causan pérdida de datos crítica o corrupción de datos. * Ponen en riesgo la seguridad del sistema. |
| **Alta** | * Defectos que afectan significativamente el uso o funcionamiento del sistema. * Pueden causar pérdida de datos importantes o errores graves en el sistema. * Comprometen la funcionalidad principal del sistema. |
| **Media** | * Defectos que afectan parcialmente el uso o funcionamiento del sistema. * Pueden causar molestias o dificultades para los usuarios. * Comprometen funcionalidades secundarias o de menor importancia. |
| **Baja** | * Defectos que tienen un impacto mínimo en el uso o funcionamiento del sistema. * No causan interrupciones significativas ni errores graves. * Pueden ser mejoras o correcciones menores en la funcionalidad. |

**Definición de artefactos**

| Listar y describir los artefactos que serán administrados y entregados durante este proceso de prueba. | | |
| --- | --- | --- |
| Artefacto | Descripción | Propósito |
| Plan de pruebas | Documento que establece los objetivos, alcance, estrategia y cronograma de las pruebas. | Proporciona una guía para la realización de las pruebas y asegura una cobertura adecuada. |
| Casos de prueba | Documentos que describen los pasos detallados para ejecutar una prueba específica. | Proporciona una estructura clara y repetible para realizar las pruebas y permite una evaluación precisa de los resultados. |
| Matriz de trazabilidad de requisitos y casos de prueba | Documento que relaciona los requisitos del sistema con los casos de prueba correspondientes. | Permite rastrear y asegurar que todos los requisitos sean probados y que los casos de prueba cubran todos los escenarios necesarios. |
| Registro de defectos | Documento o sistema de seguimiento utilizado para registrar y rastrear los defectos encontrados durante las pruebas. | Facilita la comunicación y el seguimiento de los defectos para su resolución y verificación posterior. |
| Informes de pruebas | Documentos que resumen los resultados de las pruebas, incluyendo métricas, estadísticas y conclusiones. | Proporciona una visión general de la calidad del software probado y ayuda a tomar decisiones informadas sobre su lanzamiento o implementación. |
| Informes de calidad | Documentos que evalúan la calidad del software probado en función de métricas y estándares predefinidos. | Proporciona una evaluación objetiva de la calidad del software y ayuda a identificar áreas de mejora o riesgos. |
| Informes de seguimiento y cierre de pruebas | Documentos que resumen las actividades realizadas durante las pruebas y las lecciones aprendidas del proceso. | Proporciona una visión general del proceso de pruebas y destaca las áreas de mejora y las mejores prácticas identificadas. |

**Condiciones de aceptación para cierre del proceso de pruebas**

| Condiciones que se deben cumplir para dar término al proceso de pruebas y margen de tolerancia de aceptación de defectos. |
| --- |
| Ejecución de todos los casos de prueba planificados:   * Todas las pruebas diseñadas y definidas en el plan de pruebas se han ejecutado según lo programado. * No quedan casos de prueba pendientes de ejecución.   Cumplimiento de los criterios de aceptación de pruebas:   * Los resultados de las pruebas cumplen con los criterios de aceptación predefinidos y acordados. * Los resultados se comparan con los criterios de aceptación establecidos y se encuentran dentro del margen de tolerancia definido.   Resolución de defectos críticos y de alta severidad:     * Todos los defectos críticos y de alta severidad encontrados durante las pruebas han sido resueltos y verificados.   Cumplimiento de los plazos y presupuesto de pruebas:     * El proceso de pruebas se ha completado dentro del marco de tiempo y el presupuesto asignados.   Cumplimiento de los objetivos y alcance de pruebas:     * Se han logrado los objetivos y se ha cubierto el alcance de pruebas establecidos en el plan de pruebas. |