**Plan de pruebas de software**

***[Sistema StyleHub]***

***Fecha: [25/09/2024]***

**Tabla de contenido**

[**Historial de versiones**](#_heading=h.30j0zll) **4**

[**Información del proyecto**](#_heading=h.1fob9te) **4**

[**Aprobaciones**](#_heading=h.3znysh7) **4**

[**Resumen ejecutivo**](#_heading=h.2et92p0) **5**

[**Alcance de las pruebas**](#_heading=h.tyjcwt) **5**

[**Elementos de pruebas**](#_heading=h.3dy6vkm) **5**

[**Nuevas funcionalidades a probar**](#_heading=h.1xnp6ea6cgml) **6**

[**Pruebas de regresión**](#_heading=h.4d34og8) **6**

[**Funcionalidades a no probar**](#_heading=h.2s8eyo1) **6**

[**Enfoque de pruebas (estrategia)**](#_heading=h.17dp8vu) **6**

[**Criterios de aceptación o rechazo**](#_heading=h.3rdcrjn) **7**

[**Criterios de aceptación o rechazo**](#_heading=h.26in1rg) **7**

[**Criterios de suspensión**](#_heading=h.lnxbz9) **7**

[**Criterios de reanudación**](#_heading=h.35nkun2) **7**

[**Entregables**](#_heading=h.1ksv4uv) **7**

[**Recursos**](#_heading=h.44sinio) **8**

[**Requerimientos de entornos – Hardware**](#_heading=h.2jxsxqh) **8**

[**Requerimientos de entornos – Software**](#_heading=h.z337ya) **8**

[**Herramientas de pruebas requeridas**](#_heading=h.3j2qqm3) **8**

[**Personal**](#_heading=h.1y810tw) **9**

[**Entrenamiento**](#_heading=h.4i7ojhp) **9**

[**Planificación y organización**](#_heading=h.2xcytpi) **9**

[**Procedimientos para las pruebas**](#_heading=h.1ci93xb) **9**

[**Matriz de responsabilidades**](#_heading=h.725hasz9hruw) **10**

[**Cronograma**](#_heading=h.2bn6wsx) **10**

[**Premisas**](#_heading=h.xjlmnmaggih7) **11**

[**Dependencias y Riesgos**](#_heading=h.gpc41f8jvhkq) **11**

[**Referencias**](#_heading=h.1pxezwc) **11**

[**Glosario**](#_heading=h.u3f93swmr0vm) **11**

**Historial de versiones**

| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| --- | --- | --- | --- | --- |
| **22/11/24** | **1.0** | **Yuri Durán** | **StyleHub** |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Información del proyecto**

| Empresa / Organización | Grupo Informático |
| --- | --- |
| Proyecto | Sistema |
| Fecha de preparación | 08/09/2024 |
| Cliente | StyleHub |
| Patrocinador principal | StyleHub |
| Gerente / Líder de proyecto | Yuri Durán |
| Gerente / Líder de pruebas de software | Bastian Albornoz |

**Aprobaciones**

| **Nombre y Apellido** | **Cargo** | **Departamento u organización** | **Fecha** | **Firma** |
| --- | --- | --- | --- | --- |
| **Bastian Albornoz** | **Lider de pruebas** | **Área Informática** | **22/11/2024** | **BastianA.** |
| **Yuri Duran** | **Analista de Pruebas** | **Área Informática** | **22/11/2024** | **DuránY.** |
| **Claudio Opazo** | **Analista de pruebas** | **Área Informática** | **22/11/2024** | **ClaudioO.** |

**Resumen ejecutivo**

El propósito del plan de pruebas es definir los objetivos de la prueba del sistema web solicitado, establecer y coordinar una estrategia de trabajo, y proveer un marco adecuado para así elaborar una planificación paso a paso las actividades de prueba, establecimos como equipo utilizar un plan detallado, para poder lograr los elementos clave de este proyecto.

El alcance que se tiene como objetivo es orientar el esfuerzo de pruebas, identificando y detallando las pruebas más importantes, para que el equipo de QA pueda enfocarse en su ejecución y pueda responder de forma adecuada a los cambios que tiene o tendrá el proyecto.

Las restricciones que podría presentar este proyecto serían la falta de recursos o presupuesto, falta de tiempo para la entrega final.

Este proyecto tendrá un alcance de esfuerzo de pruebas para así verificar y determinar su correcto funcionamiento, la estabilidad y confiabilidad del sistema WEB en su totalidad.

**Alcance de las pruebas**

El alcance de estas pruebas define qué partes del proyecto serán probadas y con qué propósito.

Incluye:

* Validación de la funcionalidad principal de la plataforma:
  + visualización interactiva de prendas en 3D.
  + Gestión de catálogos por Pymes.
  + Carrito de compras y sistema de pagos.
  + Filtro avanzado y recomendaciones personalizadas.
* Pruebas de compatibilidad en distintos navegadores y dispositivos.
* seguridad del portal de pagos y manejo de datos personales.

Excluye:

* Optimización de rendimiento para grandes volúmenes de datos (no prioritaria en esta fase).
* Integración con futuras funcionalidades no desarrolladas en este sprint (recomendaciones basadas en IA).

**Elementos de pruebas**

Lista de los componentes que serán evaluados.

* Frontend (Interfaz de usuario):
  + Diseño y usabilidad de la plataforma.
  + Compartimiento interactivo de los modelos 3D.
  + Formulario de registro y login.
* Backend(Funcionalidad):
  + Gestión de catálogos: creación, edición y eliminación de productos.
  + Algoritmos de recomendación y filtros de búsqueda.
  + procesamiento de pagos y transacciones seguras.
* Base de datos:
  + Almacenamiento y recuperación de datos (Catálogo, usuarios, historial de compras).
  + Consistencia de los datos tras las transacciones.
* Integraciones:
  + Visualización de modelos 3D con Three.js.
  + Portal de pagos (Integración con pasarelas como PayPal).

**Pruebas de regresión**

Las pruebas de regresión aseguran que nuevas modificaciones no afecten funcionalidades existentes.

* Áreas clave para pruebas de regresión:
  + Visualización 3D: Cambios en los modelos o su interacción no deben romper la funcionalidad.
  + Gestión de catálogos: La edición de productos debe mantener integridad en la base de datos.
  + Carrito de compras y pagos: Nuevas integraciones no deben causar errores en los flujos existentes.
* Estrategia:
  + Automatización de casos recurrentes con herramientas como Selenium para pruebas web.
  + Pruebas manuales específicas en áreas de alto impacto.

**Funcionalidades a no probar**

Definir qué partes del sistema no serán evaluadas en esta fase de pruebas.

* Exclusiones:
  + Funcionalidades planificadas pero no desarrolladas (por ejemplo, personalización basada en IA).
  + Escenarios de carga extrema o pruebas de estrés (no crítico para la fase inicial).
  + Versiones futuras de componentes integrados (como actualizaciones de librerías externas).

**Enfoque de pruebas (estrategia)**

Describe cómo se realizarán las pruebas para garantizar que cubren las necesidades del proyecto.

* Estrategia:
  + Pruebas funcionales: Validar que cada funcionalidad cumple con los requisitos (pruebas manuales y automatizadas).
  + Pruebas de integración: Asegurar que los módulos funcionan correctamente entre sí, como el frontend con el backend y el portal de pagos.
  + Pruebas de aceptación: Verificar que las funcionalidades cumplen las expectativas de los usuarios.
  + Pruebas de compatibilidad: Evaluar el rendimiento en diferentes navegadores y dispositivos.
  + Pruebas de seguridad: Validar que la plataforma protege datos sensibles como información de usuarios y pagos.

**Criterios de aceptación o rechazo**

**Criterios de aceptación o rechazo**

* Planeación de pruebas: que se planee el 100% de las pruebas posibles.
* Diseño de pruebas: que se diseñe como mínimo el 80% de las pruebas
* Implementación de pruebas: que se complete 100% de las pruebas que se hayan diseñado
* Evaluación de criterios de salida: se revisará el 100% de las evaluaciones realizadas para así poder obtener retroalimentaciones .
* Cierre del proceso: se cerrará el proyecto con un 85% de aprobación de la totalidad de los criterios de aceptación del proyecto.

**Criterios de suspensión**

Los criterios de suspensión indican las condiciones bajo las cuales debe ser detenidas.

1. Problemas críticos no resueltos:
   1. Se identifican defectos o fallos que bloquean funcionalidades clave, como, la visualización 3D no carga correctamente , fallos en el portal de pagos que impiden completar transacciones.
2. Problemas de infraestructura:
   1. El entorno de pruebas no está disponible o funcional (por ejemplo, fallos en el servidor, base de datos inaccesible).
3. Superación del tiempo estimado:
   1. El tiempo dedicado a una tarea de prueba excede significativamente la estimación sin lograr avances.
4. Alcance desviado:
   1. Las pruebas realizadas difieren de los objetivos planificados, lo que genera resultados irrelevantes.
5. Falta de recursos:
   1. insuficiencia de personal, herramientas o acceso a dispositivos necesarios para continuar las pruebas.

**Criterios de reanudación**

Los criterios de reanudación definen las condiciones que deben cumplirse para reiniciar las pruebas después de una suspensión.

1. Resolución de problemas críticos:
   1. Se han corregido los defectos bloqueantes identificados y se han implementado en el entorno de prueba.
2. Restauración de la infraestructura:
   1. El entorno de pruebas vuelve a estar operativo y accesible para el equipo.
3. Revisión del alcance:
   1. Se ajustan los objetivos y se actualiza el plan de pruebas para alinearse con las prioridades.
4. disponibilidad de recursos:
   1. El equipo, herramientas y dispositivos necesarios están disponibles nuevamente.
5. validación de correcciones:
   1. Las correcciones implementadas han sido verificadas en un entorno controlado y son funcionales

**Entregables**

[Planilla de riesgos.xlsx](https://docs.google.com/spreadsheets/d/18Hmf6iJYNRhHwrAbbtkQUCeYIdWDxL0_/edit?usp=drive_link&ouid=116733068945474691895&rtpof=true&sd=true)

[Acta\_De\_Constitución\_Proyecto.docx](https://docs.google.com/document/d/1t1pOwsOWGxq1WpgrsQwSqUAWevpKtfcF/edit?usp=drive_link&ouid=116733068945474691895&rtpof=true&sd=true)

[Plan\_de\_Implantación.docx](https://docs.google.com/document/d/1vvcN043NwxURR5Oi-ag-Rt5VIe5ftFmw/edit?usp=drive_link&ouid=116733068945474691895&rtpof=true&sd=true)

[Requisitos\_funcionales.xlsx](https://docs.google.com/spreadsheets/d/1kIe2Rd6vj0pnf0spCb1ceK_il2HV_ECj/edit?usp=drive_link&ouid=116733068945474691895&rtpof=true&sd=true)

**Recursos**

**Requerimientos de entornos – Hardware**

**Componentes:**

- Procesador: Intel Core I9-7490 X 3.10 – 4.40 MHz.

- Almacenamiento: SSD NVMe 500 Gb.

- RAM: 16 Gb DDR4-2666 GHz.

- Teclado

- Monitor

- Mouse

**Requerimientos de entornos – Software**

**Software:**

**Server:**

* Ubuntu Linux 16.04.

**Sistema operativo:**

* Windows 10 y 11.
* Mac
* Linux

**Herramientas de pruebas requeridas**

* Pruebas unitarias (Jasmine).
* Prueba de componentes .
* Pruebas de sistema.
* Pruebas automatizadas (Katalon Studio y JMeter).
* Prueba de aceptación.

**Personal**

**El personal requerido para este proyecto fue:**

* Lider de proyecto (1)
* Lider de pruebas (1)
* Analista de pruebas (2)
* Especialista en automatización de pruebas (1)

**Entrenamiento**

* Las necesidades para realizar un correcto plan de pruebas es que sepan utilizar la metodología Ágil.
* Se necesitará conocimiento sobre las pruebas de tipo: desempeño del sistema, estres del sistema, arquitectura del sistema y código y testeos de caja blanca.
* Se necesitará conocimiento sobre automatización de las pruebas del sistema.

**Planificación y organización**

**Procedimientos para las pruebas**

1. Planeación de pruebas.
2. Diseño de pruebas:
   1. Revisión front-end.
   2. Pruebas unitarias de módulo.
   3. Pruebas de módulo en conjunto.
   4. Pruebas de todo el sistema.
3. Implementación de pruebas.
4. Evaluación de criterios de salida.
5. Cierre del proceso.

**Matriz de responsabilidades**

| **No.** | **ACTIVIDADES** | **Claudio Opazo** | **Bastián Albornoz** | **Yuri Durán** |
| --- | --- | --- | --- | --- |
| 1 | Planeación de pruebas. | **C** | **A** | **I** |
| 2 | Diseño de pruebas. | **R** | **A** | **R** |
| 3 | Implementación de pruebas. | **R** | **A** | **R** |
| 4 | Evaluación de criterios de salida. | **C** | **R** | **A** |
| 5 | Cierre del proceso. | **C** | **I** | **A** |
| **CLAVE: R = Responsable; A = Aprobador; C = Consultado ; I = Informado** | | | | |

**Cronograma**

| **N°** | **Actividades** | **Fecha** |
| --- | --- | --- |
| 1 | **Planeación de pruebas.** | 25/09/2024 |
| 2 | **Diseño de pruebas.** | 27/09/2024 |
| 3 | **Implementación de pruebas.** | 20/11/2024 |
| 4 | **Evaluación de criterios de salida.** | 21/11/2024 |
| 5 | **Cierre del proceso.** | 22/11/2024 |

**Premisas**

Las premisas relacionadas con las tareas de pruebas de software:

* Organización del Proyecto.
* Difusión de propuesta.
* Contratación de equipo.
* Acondicionamiento de sistemas o instalaciones.

Estas fueron las principales premisas relacionadas con las tareas de pruebas, ya que eran las principales limitaciones de tiempo. La metodología utilizada para las pruebas fue la metodología RUP (Rational Unified Process), ya que esta fue la más adecuada.

**Dependencias y Riesgos**

**Dependencias:**

* Dependencias con desarrollos.
* Dependencia basada en recursos.
* Dependencias con otros proyectos.
* Dependencias basada en preferencias.

**Riesgos:**

* Escasez de recursos.
* Falta de tiempo.
* Adaptabilidad.

**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:

[Requisitos\_funcionales.xlsx](https://docs.google.com/spreadsheets/d/1kIe2Rd6vj0pnf0spCb1ceK_il2HV_ECj/edit?usp=drive_link&ouid=116733068945474691895&rtpof=true&sd=true)

**Glosario**

RUP (Rational Unified Process): Es uno de los modelos tradicionales utilizados para el control de calidad.