

# HISTORIAL DE CAMBIOS

| FECHA | VERSIÓN | DESCRIPCIÓN | RESPONSABLE |
| --- | --- | --- | --- |
| **23-05-24** | 1.0 | Avance del documento hasta el punto 5. | Juan David Soriano, Deivid Ruiz, kevin Jhohan Galindo Robles, Thomas Andres Bello Gonzalez, Dariana Marcela De la cruz Ortega. |

**Tabla 1: Historial de cambios**

# TABLA DE CONTENIDO

[1 HISTORIAL DE CAMBIOS 1](#_heading=h.gjdgxs)

[2 TABLA DE CONTENIDO 1](#_heading=h.1fob9te)

[ÍNDICE DE TABLAS 1](#_heading=h.3znysh7)

[3.1 Objetivos 1](#_heading=h.2et92p0)

[3.2 Estrategia de Pruebas 1](#_heading=h.tyjcwt)

[3.3 Alcance 1](#_heading=h.1t3h5sf)

[3.4 Referencias 1](#_heading=h.2s8eyo1)

[3.5 Definiciones, abreviaciones y acrónimos 1](#_heading=h.17dp8vu)

[4 ARTEFACTOS DE PRUEBA 1](#_heading=h.26in1rg)

[4.1 Módulos del Programa 1](#_heading=h.lnxbz9)

[4.2 Procedimientos de Usuario 1](#_heading=h.44sinio)

[5 CARACTERISTICAS A SER PROBADAS 1](#_heading=h.z337ya)

[6 CARACTERÍSTICAS QUE NO SERAN PROBADAS 1](#_heading=h.1y810tw)

[7 APROXIMACION 1](#_heading=h.2xcytpi)

[7.1 Pruebas Unitarias 1](#_heading=h.1ci93xb)

[7.2 Pruebas de Integración 1](#_heading=h.qsh70q)

[7.3 Pruebas de Sistema 1](#_heading=h.1pxezwc)

[7.4 Prueba Caja Blanca 1](#_heading=h.2p2csry)

[7.5 Pruebas Caja Negra 1](#_heading=h.147n2zr)

[7.6 Pruebas de Bases de Datos 1](#_heading=h.3o7alnk)

[8 PROCESO DE PRUEBAS 1](#_heading=h.23ckvvd)

[8.1 Pruebas Unitarias 1](#_heading=h.ihv636)

[8.1.1 Cliente 1](#_heading=h.32hioqz)

[8.1.2 Proveedor 1](#_heading=h.1hmsyys)

[8.1.3 Administrador 1](#_heading=h.41mghml)

[8.2 Pruebas De Uso 1](#_heading=h.2grqrue)

[8.2.1 Cliente 1](#_heading=h.vx1227)

[8.3 Pruebas De Integración 1](#_heading=h.3fwokq0)

[8.3.1 Cliente 1](#_heading=h.1v1yuxt)

[8.3.2 Proveedor 1](#_heading=h.4f1mdlm)

[8.4 Pruebas De Seguridad 1](#_heading=h.2u6wntf)

[9 ANEXO 1](#_heading=h.19c6y18)

[9.1 Anexo 1: Reportes de Pruebas 1](#_heading=h.3tbugp1)

# ÍNDICE DE TABLAS

[Tabla 1: Historial de cambios 2](#_heading=h.30j0zll)

[Tabla 2: Definiciones, acrónimos y abreviaciones 9](#_heading=h.3rdcrjn)

[Tabla 3: Módulos a probar en el sistema 11](#_heading=h.1ksv4uv)

[Tabla 4. Características a ser Probadas 13](#_heading=h.3j2qqm3)

[Tabla 5. Características que no serán probadas. 14](#_heading=h.4i7ojhp)

[Tabla 6. Pruebas Unitarias. 15](#_heading=h.3whwml4)

[Tabla 7. Pruebas de Frontera 15](#_heading=h.28h4qwu)

[Tabla 8. Entregable de la prueba. 16](#_heading=h.2bn6wsx)

[Tabla 9. Prueba de Integración. 16](#_heading=h.3as4poj)

[Tabla 10. Resultados Prueba Integración 16](#_heading=h.nmf14n)

[Tabla 11. Pruebas de Sistema 16](#_heading=h.49x2ik5)

[Tabla 12. Resultados Pruebas de Sistema. 17](#_heading=h.37m2jsg)

[Tabla 13: Caso de Prueba 1 18](#_heading=h.1mrcu09)

[Tabla 14: Caso de Prueba 2 19](#_heading=h.46r0co2)

[Tabla 15: Caso de Prueba 3 20](#_heading=h.2lwamvv)

[Tabla 16: Caso de Prueba 4 20](#_heading=h.111kx3o)

[Tabla 17: Caso de Prueba 5 21](#_heading=h.3l18frh)

[Tabla 18: Caso de Prueba 6 21](#_heading=h.206ipza)

[Tabla 19: Caso de Prueba 7 21](#_heading=h.4k668n3)

[Tabla 20: Caso de Prueba 8 22](#_heading=h.2zbgiuw)

[Tabla 21: Caso de Prueba 9 22](#_heading=h.1egqt2p)

[Tabla 22: Caso de Prueba 10 23](#_heading=h.3ygebqi)

[Tabla 23: Caso de Prueba 11 23](#_heading=h.2dlolyb)

[Tabla 24: Caso de Prueba 12 24](#_heading=h.sqyw64)

[Tabla 25: Caso de Prueba 13 24](#_heading=h.3cqmetx)

[Tabla 26: Caso de Prueba 14 25](#_heading=h.1rvwp1q)

[Tabla 27: Caso de Prueba 15 25](#_heading=h.4bvk7pj)

[Tabla 28: Caso de Prueba 16 25](#_heading=h.2r0uhxc)

[Tabla 29: Caso de Prueba 17 26](#_heading=h.1664s55)

[Tabla 30: Caso de Prueba 18 27](#_heading=h.3q5sasy)

[Tabla 31: Caso de Prueba 19 28](#_heading=h.25b2l0r)

[Tabla 32: Caso de Prueba 20 28](#_heading=h.kgcv8k)

[Tabla 33: Caso de Prueba 21 28](#_heading=h.34g0dwd)

[Tabla 34: Caso de Prueba 22 29](#_heading=h.1jlao46)

[Tabla 35: Caso de Prueba 23 29](#_heading=h.43ky6rz)

# INTRODUCCIÓN

## Objetivos

EL plan de pruebas de Software se elabora con el fin de especificar qué elementos o componentes se van a probar para que el grupo de trabajo pueda realizar el proceso de Validación y Verificación de los requerimientos funcionales y no funcionales de la herramienta SHOOP. Además, a través del plan de pruebas se puede continuar con la trazabilidad de los requerimientos, con lo cual el grupo de trabajo identifica el porcentaje de avance que se ha logrado hasta cierto momento.

Al desarrollar el plan de pruebas, se puede obtener información sobre los errores, defectos o fallas que tiene el prototipo, así se realizan las correcciones pertinentes, según el caso y se asegura la calidad del producto que se está entregando al cliente. El plan de pruebas se aplica sobre el producto, el código fuente de SHOOP. Los resultados se registran en el Anexo 1: Reportes de Pruebas. Las pruebas por implementar son básicas, esto incluye las pruebas caja blanca, caja negra, unitarias, integración y del sistema que son vitales para la validación del producto.

## Estrategia de Pruebas

A través de los diferentes documentos que se han realizado, se pretende retomar información directamente relacionada con las pruebas, para asegurar la calidad de estas y del producto. Además, le permite al responsable de las pruebas saber exactamente los criterios que se deben tener en cuenta para probar cada elemento del sistema. A continuación, se explica brevemente el aporte de cada documento con respecto al plan de pruebas.

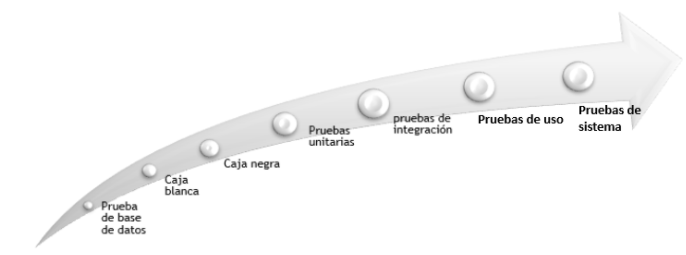


**Ilustración 1: Estrategia del plan de pruebas**

Con esta estrategia se asegura llevar el seguimiento de la trazabilidad que se ha manejado desde la especificación de requerimientos (documento SRS), además de mantener la consistencia entre la aplicación y su respectiva documentación.

## Alcance

Teniendo en cuenta los documentos hechos anteriormente, el grupo de trabajo pretende realizar las pruebas, de manera incremental, por módulo. Para una mejor comprensión, ver la Ilustración 2: Alcance del plan de pruebas (Prueba de base de datos, caja blanca, caja negra, pruebas unitarias, pruebas de integración, pruebas de uso, pruebas de sistemas), la cual muestra el alcance y el orden en que se realizaran.



**Ilustración 2: Alcance del plan de pruebas**

Las pruebas unitarias se realizarán inmediatamente después de haber implementado la aplicación, esto quiere decir, que el orden corresponde al descrito en la prioridad de requerimientos y por grafo de dependencias construidos desde el documento SRS Sección 2.9 PRIORIZACION DE REQUERIMIENTOS y 2.10 RELACIONES ENTRE LOS REQUERIMIENTOS.

## Referencias

1. ISO/IEC 9126 – Calidad del software.
   * ISO-IEC 14598.
   * ISO/IEC 9126-1.
   * ISO/IEC 9126-2 TR Métricas Externas.
   * ISO/IEC 9126-3 TR Métricas Internas.
   * ISO/IEC 9126-4 TR Métricas de Calidad en Uso.
2. ISO/IEC 2500n – División de Gestión de Calidad.
   * ISO/IEC 25000 - Guide to SQuaRE.
   * ISO/IEC 25001 - Planning and Management.
3. ISO/IEC 2501n – División de Modelo de Calidad.
   * ISO/IEC 25010 - System and software quality models.
   * ISO/IEC 25012 - Data Quality model.
4. ISO/IEC 2502n – División de Medición de Calidad
   * ISO/IEC 25020 - Measurement reference model and guide.
   * ISO/IEC 25021 - Quality measure elements.
   * ISO/IEC 25022 - Measurement of quality in use.
   * ISO/IEC 25023 - Measurement of system and software product quality.
   * ISO/IEC 25024 - Measurement of data quality.
5. ISO/IEC 2503n – División de Requisitos de Calidad
   * ISO/IEC 25030 - Quality requirements.
6. ISO/IEC 2504n – División de Evaluación de Calidad
   * ISO/IEC 25040 - Evaluation reference model and guide.
   * ISO/IEC 25041 - Evaluation guide for developers, acquirers and independent evaluators.
   * ISO/IEC 25042 - Evaluation modules.
   * ISO/IEC 25045 - Evaluation module for recoverability.

## Definiciones, abreviaciones y acrónimos

| CONCEPTO | DESCRIPCIÓN |
| --- | --- |
| Caja blanca | Es un tipo de prueba de software que se enfoca en verificar el funcionamiento de la estructura interna y el diseño del software. |
| Caja negra | Es una forma directa de prueba de software para evaluar la compatibilidad entre las interfaces de los diferentes componentes del software. |
| Casos de uso | Los casos de uso están diseñados como herramientas sencillas para describir el comportamiento del software o de los sistemas. |
| GUI | La interfaz gráfica de usuario |
| Normas ISO | Organización Internacional de Estandarización (conjunto de estándares internacionales compuesta por diversas organizaciones nacionales de normalización) |
| Shoop | Software de tienda virtual |
| SRS | Especificación de requisitos de software |
| Pruebas unitarias | Una prueba unitaria es un fragmento de código que comprueba la exactitud de un segmento más pequeño y aislado del código de la aplicación, generalmente una función o un método. |
| Pruebas de integración | Las pruebas de integración aseguran que los diversos módulos o servicios utilizados por tu aplicación operen correctamente en conjunto. |
| Pruebas de sistemas | Las pruebas de sistema son un tipo de prueba de software que verifica el funcionamiento del sistema. |
| Visual Studio Code | Es un software editor de código fuente, contiene diversos lenguajes de programación. |

**Tabla 2: Definiciones, acrónimos y abreviaciones**

# ARTEFACTOS DE PRUEBA

## Módulos del Programa

En esta sección se muestran los módulos que se pretenden probar, además de las especificaciones de las pruebas a realizar en cada uno. Cabe notar, que cada módulo representa un componente del sistema. Para obtener la descripción de cada componente ver [Documento Diseño](https://soysena-my.sharepoint.com/:w:/g/personal/jdsoriano98_soy_sena_edu_co/EWZJsFxaf5lOjiWQwEyGNgoBuAsaOP4aghklCabcEcXKaQ?e=scSGVd) V 1.0 sección 3.2. DIAGRAMA DE COMPONENTES. Los ítems a manejar en la tabla son los siguientes:



**Ilustración 3: Módulos del programa**

| Modulo | Pruebas | Descripción |
| --- | --- | --- |
| Usuario | Pruebas Unitarias | Se centran en verificar el funcionamiento correcto de unidades individuales de código, como funciones o métodos, a nivel de componente. |
| Pruebas Caja Blanca | Evalúan la lógica interna y la estructura del código, probando las rutas de ejecución y las condiciones lógicas dentro del software. |
| Pruebas Caja Negra | Se realizan sin conocer la implementación interna del código. En lugar de eso, se evalúa la funcionalidad del software a partir de sus entradas y salidas esperadas. |
| Pruebas de Integración | Verifican que los diferentes módulos o componentes del sistema funcionen correctamente juntos, probando la interacción entre ellos. |
| Pruebas de Sistema | Evalúan el sistema completo como una entidad única, asegurando que cumpla con los requisitos y expectativas del usuario final. |
| Proveedor | Pruebas Unitarias | Se centran en verificar el funcionamiento correcto de unidades individuales de código, como funciones o métodos, a nivel de componente. |
|  | Pruebas Caja Blanca | Evalúan la lógica interna y la estructura del código, probando las rutas de ejecución y las condiciones lógicas dentro del software. |
|  | Pruebas Caja Negra | Se realizan sin conocer la implementación interna del código. En lugar de eso, se evalúa la funcionalidad del software a partir de sus entradas y salidas esperadas. |
|  | Pruebas de Integración | Verifican que los diferentes módulos o componentes del sistema funcionen correctamente juntos, probando la interacción entre ellos. |
|  | Pruebas de Sistema | Evalúan el sistema completo como una entidad única, asegurando que cumpla con los requisitos y expectativas del usuario final. |
| Administrador | Pruebas Unitarias | Se centran en verificar el funcionamiento correcto de unidades individuales de código, como funciones o métodos, a nivel de componente. |
|  | Pruebas Caja Blanca | Evalúan la lógica interna y la estructura del código, probando las rutas de ejecución y las condiciones lógicas dentro del software. |
|  | Pruebas Caja Negra | Se realizan sin conocer la implementación interna del código. En lugar de eso, se evalúa la funcionalidad del software a partir de sus entradas y salidas esperadas. |
|  | Pruebas de Integración | Verifican que los diferentes módulos o componentes del sistema funcionen correctamente juntos, probando la interacción entre ellos. |
|  | Pruebas de Sistema | Evalúan el sistema completo como una entidad única, asegurando que cumpla con los requisitos y expectativas del usuario final. |

**Tabla 3: Módulos a probar en el sistema**

## Procedimientos de Usuario

Para utilizar la herramienta de manera adecuada se necesitan guías o manuales que sean claros, correctos, completos y coherentes, para que el usuario pueda manejar la herramienta de forma correcta y pueda comprender los conceptos tras la funcionalidad. A continuación, se muestran los diferentes atributos de calidad de estos procedimientos:

* **Clara**: las instrucciones proporcionadas en el documento deben ser lo suficientemente explicitas para que el usuario pueda desenvolverse dentro del entorno de la herramienta.
* **Correcta**: No existen errores semánticos, sintácticos, ortográficos ni de enlace dentro de la documentación proporcionada al usuario.
* **Completa**: la información debe estar completa, desde la parte técnica hasta la parte funcional.
* **Coherente**: no existen ambigüedades, ni incongruencias dentro del documento que puedan confundir al usuario.

Los documentos por entregar con la herramienta son:



**Ilustración 4: Documentos y responsables.**

# CARACTERISTICAS A SER PROBADAS

En esta sección se encuentran las características de la herramienta a ser probadas con un caso de estudio específico.

| **Característica** | **Descripción** | **Módulo** |
| --- | --- | --- |
| **Funcionalidad** | Capacidad del producto software para proporcionar las funcionalidades que satisfacen las necesidades explicitas e implícitas cuando el software se usa bajo unas ciertas condiciones | |
|  | Capacidad del producto software para proporcionar un conjunto de funciones apropiado para unas ciertas tareas y objetivos de usuario | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Adecuación |
| Capacidad del producto software para proporcionar los resultados o efectos correctos o acordados, con el grado necesario de precisión | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Exactitud |
| Capacidad del producto software para interactuar con uno o más sistemas | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Interoperabilidad |
| Capacidad del producto software para proteger información y datos de manera que las personas o sistemas no autorizados no puedan leerlos o modificarlos, al tiempo que no se deniega el acceso a las personas o sistemas autorizados | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Seguridad |
| Capacidad del producto software para adherirse a normas, convenciones o regulaciones en leyes y prescripciones similares relacionadas con la funcionalidad | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Cumplimiento funcional |
| **Fiabilidad** | Capacidad del producto software para mantener un nivel especificado de prestaciones cuando se usa bajo unas ciertas condiciones | |
|  | Capacidad del producto software para evitar fallar como resultado de fallos en el software | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Madurez |
| Capacidad del software para mantener un nivel especificado de prestaciones en caso de fallos software o de infringir sus interfaces | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Tolerancia a fallos |
| Capacidad del producto software para reestablecer un cierto nivel de prestaciones y de recuperar los datos directamente afectados en caso de fallo | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Capacidad de recuperación |
| Capacidad del producto software para adherirse a normas, convenciones o regulaciones relacionadas con la fiabilidad | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Cumplimiento de la fiabilidad |
| **Usabilidad** | Capacidad del producto software para ser entendido, aprendido, usado y ser atractivo para el usuario, cuando se usa bajo condiciones especificadas. | |
|  | Capacidad del producto software que permite al usuario entender si el software es adecuado y cómo puede ser usado para unas tareas o condiciones de uso particulares | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Capacidad para ser entendido |
| Capacidad del producto software que permite al usuario aprender sobre su aplicación | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Capacidad para ser aprendido |
| Capacidad del producto software que permite al usuario administrarlo y controlarlo | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Capacidad para ser administrado |
| Capacidad del producto software para ser atractivo al usuario | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Capacidad de ser atractivo |
| Capacidad del producto software para adherirse a normas, convenciones, guías de estilo o regulaciones relacionadas con la usabilidad | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Cumplimiento de la usabilidad |
| **Eficiencia** | Capacidad del producto software para proporcionar prestaciones apropiadas, relativas a la cantidad de recursos usados, bajo condiciones determinadas | |
|  | Capacidad del producto software para proporcionar tiempos de respuesta y de proceso y índices de respuesta al realizar sus funciones bajo unas ciertas condiciones | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Comportamiento temporal |
| Capacidad del producto software para usar las cantidades y tipos de recursos adecuados cuando el software lleva a cabo su función bajo condiciones determinadas | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Utilización de recursos |
| Capacidad del producto software para adherirse a normas o convenciones relacionadas con la eficiencia | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Cumplimiento de la eficiencia |
| **Mantenibilidad** | Capacidad del producto software para ser modificado. Las modificaciones podrían incluir correcciones, mejoras o adaptación del software a cambios en el entorno, y requisitos y especificaciones funcionales. | |
|  | Capacidad del producto software para serle diagnosticadas deficiencias o causas de los fallos en el software, o para identificar las partes que han de ser modificadas | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Capacidad de ser analizado |
| Capacidad del producto software que permite que una determinada modificación sea implementada | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Capacidad para ser cambiado |
| Capacidad del producto software para evitar efectos inesperados debidos a modificaciones del software | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Estabilidad |
| Capacidad del producto software que permite que el software modificado sea validado | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Capacidad para ser probado |
| Capacidad del producto software para adherirse a normas o convenciones relacionadas con la mantenibilidad | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Cumplimiento de la mantenibilidad |
| **Portabilidad** | Capacidad del producto software para ser migrado de un entorno a otro | |
|  | Capacidad del producto software para ser adaptado a diferentes entornos, sin aplicar acciones o mecanismos distintos de aquellos proporcionados para este propósito por el propio software | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Adaptabilidad |
| Capacidad del producto software para ser instalado en un cierto entorno | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Instalabilidad |
| Capacidad del producto software para coexistir con otro software independiente, en un entorno común, compartiendo recursos comunes | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Coexistencia |
|  | Capacidad del producto software para ser usado en lugar de otro producto software, para el mismo propósito, en el mismo entorno | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Capacidad para reemplazar |
| Capacidad del producto software para adherirse a normas o convenciones relacionadas con la portabilidad | Los módulos donde se puede probar esta característica son:  Usuario, administrador y proveedor  Cumplimiento de la portabilidad |

**Tabla 4. Características a ser Probadas**

# CARACTERÍSTICAS QUE NO SERAN PROBADAS

En esta sección se encuentran las características de la herramienta a ser probadas con un caso de estudio específico.

| Característica | Descripción | Módulo |
| --- | --- | --- |
| Escalabilidad | La aplicación debe poder manejar un incremento en la carga de usuarios y transacciones sin deterioro significativo del rendimiento. | Usuario – Proveedor - Administrador |
| Disponibilidad | La tienda en línea debe tener una disponibilidad de 99.9% para asegurar que los usuarios puedan acceder al sitio casi siempre. | Usuario – Proveedor - Administrador |
| Recuperación de Desastres | El sistema debe tener un plan de recuperación ante desastres que permita restaurar los datos y las operaciones en menos de 1 hora. | Usuario – Proveedor - Administrador |
| Tolerancia a Fallos | La aplicación debe ser capaz de seguir funcionando correctamente incluso si uno o más componentes fallan. | Usuario – Proveedor - Administrador |
| Autenticación y Autorización | Debe implementar autenticación robusta (e.g., autenticación multifactor) y autorización basada en roles para proteger las funciones del cliente, proveedor y administrador. | Usuario – Proveedor - Administrador |
| Protección de Datos | Toda la información sensible, como datos de clientes y proveedores, debe estar encriptada tanto en tránsito como en reposo. | Usuario – Proveedor - Administrador |
| Cumplimiento Normativo | El sistema debe cumplir con las normativas de protección de datos aplicables. | Usuario – Proveedor - Administrador |
| Interfaz de Usuario Intuitiva | El sistema debe proporcionar una interfaz de usuario intuitiva y fácil de usar para todos los actores (cliente, proveedor, administrador). | Usuario – Proveedor - Administrador |
| Accesibilidad | La tienda en línea debe ser accesible conforme a los estándares WCAG 2.1, asegurando que las personas con discapacidades puedan utilizar el sitio. | Usuario – Proveedor - Administrador |
| Documentación | El código debe estar bien documentado para facilitar la comprensión y mantenimiento por parte de nuevos desarrolladores. | Usuario – Proveedor - Administrador |
| Modularidad | El sistema debe estar diseñado de manera modular para facilitar la actualización y el reemplazo de componentes individuales. | Usuario – Proveedor - Administrador |
| Compatibilidad de Navegadores | La tienda en línea debe ser compatible con todos los navegadores web modernos (Chrome, Firefox, Safari, Edge). | Usuario – Proveedor - Administrador |
| Responsividad | El diseño debe ser responsive para asegurar una experiencia de usuario óptima en dispositivos móviles, tabletas y ordenadores. | Usuario – Proveedor - Administrador |
| Optimización de Base de Datos | Las consultas a la base de datos deben estar optimizadas para minimizar tiempos de espera y mejorar el rendimiento general. | Usuario – Proveedor - Administrador |
| Feedback del Usuario | El sistema debe incluir mecanismos para recibir y gestionar el feedback de los usuarios para mejorar continuamente la experiencia de usuario. | Usuario – Proveedor - Administrador |
| Servicio de Soporte | Debe haber un servicio de soporte disponible para resolver problemas técnicos y consultas de los usuarios dentro de un tiempo razonable. | Usuario – Proveedor - Administrador |

**Tabla 5. Características que no serán probadas.**

# APROXIMACION

En esta sección se exponen los tipos de pruebas a utilizar para la herramienta ERMT, cada una de ellas presenta un formato, el cual se va registrar los resultados.

## Pruebas Unitarias

Pruebas por cada unidad, en este caso una unidad es equivalente a un requerimiento. El requerimiento es aprobado y aprobado si este cumple con lo que está escrito en la especificación de requerimientos.

| NOMBRE | Pruebas Unitarias | | IDENTIFICADOR | UT01 |
| --- | --- | --- | --- | --- |
| ACTIVIDADES | | Verifica los procesos que se realizará en la prueba a cada tabla de la base de datos que sea capaz de hacer CRUD crear, leer, actualizar, borrar en total debemos hacer las pruebas con 15 tablas | | |
| TIEMPO ESTIMADO | | 20– 30 minutos por unidad | | |
| MÉTODOS O HERRAMIENTAS | | Visual STUDIO Sublime text | | |
| ENTREGABLES | | Cumplimiento de la prueba | | |

**Tabla 6. Pruebas Unitarias.**

En cuanto a los entregables de esta prueba se llenará la siguiente tabla:

| Nombre |  | Identificador | T01 | |
| --- | --- | --- | --- | --- |
| Valor máximo |  | **Valor mínimo** |  | |
| Resultado esperado | |  | | |
| Resultados obtenidos | |  | | |
| Estado | | Funciona: | | No funciona: |
| Comentarios | |  | | |

**Tabla 8. Entregable de la prueba.**

## Pruebas de Integración

Las pruebas de integración, como su nombre lo indica, son pruebas hechas a un conjunto de requerimientos, en este caso se distribuyen en los tipos de requerimientos que se definieron en el documento SRS.

| NOMBRE | Pruebas de Integración | | IDENTIFICADOR | IT01 |
| --- | --- | --- | --- | --- |
| ACTIVIDADES | | se comprueba la lógica, funcionalidad y especificación de cada unidad  del sistema de manera aislada con respecto al resto de unidades. | | |
| TIEMPO ESTIMADO | | 20 – 30 minutos por pruebas | | |
| MÉTODOS O HERRAMIENTAS | | Visual STUDIO Sublime text | | |
| ENTREGABLES | | Cumplimiento de la prueba | | |

**Tabla 9. Prueba de Integración.**

## Pruebas de Sistema

Las pruebas de sistema son pruebas realizadas a la herramienta como un conjunto, que casos de uso cumple a cabalidad, con rutas de éxito y fallo, que han sido definidas en el documento de Casos de Uso.

En este tipo de pruebas se verifica el cumplimiento de los requisitos

especificados, probando el sistema integrado en su entorno de hardware

y software.

| NOMBRE | Pruebas sistemas | | IDENTIFICADOR | ST01 |
| --- | --- | --- | --- | --- |
| ACTIVIDADES | | La prueba de sistemas es un tipo de prueba de software que realiza comprobaciones del sistema en su conjunto. Consiste en integrar todos los módulos y componentes individuales del software que has desarrollado, para comprobar si el sistema funciona conjuntamente como se esperaba. | | |
| TIEMPO ESTIMADO | | 15 minutos por prueba | | |
| MÉTODOS O HERRAMIENTAS | | Visual Studio Code – Sublime Text - Mysql | | |
| ENTREGABLES | | Cumplimiento de la prueba. | | |

**Tabla 11. Pruebas de Sistema**

## Prueba Caja Blanca

Las pruebas de caja blanca permiten probar la lógica interna del programa y su estructura realizando las siguientes acciones:

Ejecución de todas las sentencias (al menos una vez).

Recorrido de todos los caminos independientes de cada componente.

Comprobación de todas las decisiones lógicas.

Comprobación de todos los bucles.

Implementación de situaciones extremas o límites.

| NOMBRE | Pruebas Unitarias | | IDENTIFICADOR | UT01 |
| --- | --- | --- | --- | --- |
| ACTIVIDADES | | Verificar mántricas internas del back-end como bases datos conexiones entre vistas Y consultas de bases de datos | | |
| TIEMPO ESTIMADO | | 20– 30 minutos por unidad | | |
| MÉTODOS O HERRAMIENTAS | | Visual STUDIO -Sublime text ,MSQL | | |
| ENTREGABLES | | Cumplimiento de la prueba | | |

## Pruebas Caja Negra

La prueba de caja negra es una forma de prueba de software en la que la funcionalidad del software es comprobada por probadores que no tienen conocimiento de la estructura interna del código o de cómo implementar el código a un nivel más técnico.

Las pruebas de caja negra sólo comprueban los resultados externos del programa, es decir, lo que experimentará el usuario final cuando utilice el programa.

| NOMBRE | Pruebas sistemas | | IDENTIFICADOR | ST01 |
| --- | --- | --- | --- | --- |
| ACTIVIDADES | | Verifica las métricas externas como UI – Front End. Estas consisten en comprobar los resultados externos de la compilación del software, | | |
| TIEMPO ESTIMADO | | 15 minutos por prueba | | |
| MÉTODOS O HERRAMIENTAS | | Visual Studio Code – Sublime Text | | |
| ENTREGABLES | | Cumplimiento de la prueba | | |

**Tabla 11. Pruebas de Sistema**

## Pruebas de Bases de Datos

Las pruebas de bases de datos son procesos que verifican la estructura, integridad, seguridad y rendimiento de una base de datos. Aseguran que las tablas, datos, consultas y transacciones funcionan correctamente y cumplen con las reglas de negocio. También validan que los permisos de usuario y las copias de seguridad estén correctamente implementados para proteger la información.

| NOMBRE | Pruebas de Bases de Datos | | IDENTIFICADOR | ST01 |
| --- | --- | --- | --- | --- |
| ACTIVIDADES | | Verifica las métricas de pruebas de bases de datos: estructura correcta, datos válidos, consultas SQL precisas, buen rendimiento, transacciones ACID, integridad de datos, seguridad adecuada y backups funcionales. | | |
| TIEMPO ESTIMADO | | 20 - 30 minutos por prueba | | |
| MÉTODOS O HERRAMIENTAS | | MySQL Workbench, Oracle SQL Developer, SQL Server Management Studio (SSMS) | | |
| ENTREGABLES | | Cumplimiento de la prueba | | |

**Tabla 11. Pruebas de Sistema**

# PROCESO DE PRUEBAS

En esta sección se presentan los casos de pruebas generales para usarlos con la herramienta SHOOP. Cada cuadro está asociado a un caso de Uso, desde ahí se desglosa en los diferentes módulos involucrados para el funcionamiento y se evalúa el resultado obtenido. En las siguientes tablas, se muestran los casos de pruebas a realizar:

## Pruebas Unitarias

### Cliente

| **NOMBRE** | **Gestionar pedido** | PRUEBAS | **PUN 1[[1]](#footnote-0)** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El proveedor gestiona los pedidos recibidos, incluyendo la actualización del estado de los pedidos y la coordinación del envío. | | |
| **PRERREQUISITOS** | * El proveedor debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Vista vwgpo.php | | |
| **ENTRADA** | * El proveedor se dirige al apartado de gestionar sus pedidos | | |
| **ORÁCULO** | El sistema muestra todos los pedidos realizados | | |
| **PASOS** | * El proveedor accede al panel de administración de su cuenta. * El sistema muestra las opciones de gestión de pedidos. * El proveedor selecciona la opción para gestionar pedidos. * El sistema muestra una lista de pedidos recibidos, incluyendo detalles como ID del pedido, fecha de pedido, productos solicitados, cantidad, estado del pedido (pendiente, procesado, enviado, entregado) y detalles del cliente. * El proveedor puede realizar las siguientes acciones: * Actualizar el estado de un pedido: * El proveedor selecciona un pedido de la lista para actualizar su estado. * El sistema muestra los detalles actuales del pedido y las opciones de estado disponibles. * El proveedor selecciona el nuevo estado y envía la actualización. * El sistema actualiza el estado del pedido y notifica al cliente del cambio. * Coordinar el envío de un pedido: * El proveedor selecciona un pedido de la lista para coordinar el envío. * El sistema muestra los detalles del pedido y las opciones de envío. * El proveedor selecciona la opción de envío y proporciona los detalles necesarios. * El sistema actualiza el estado del pedido a "Enviado" y notifica al cliente. * Ver detalles de un pedido: * El proveedor selecciona un pedido de la lista para ver más detalles. * El sistema muestra los detalles completos del pedido, incluyendo la información del cliente y la dirección de envío. * El sistema confirma los cambios realizados y actualiza la lista de pedidos. | | |
| **Módulos Asociados** | * Proveedor | | |

| C**NOMBRE** | **Gestionar Direcciones de Envío** | PRUEBAS | **PUN 2** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario (cliente o proveedor) gestiona sus direcciones de envío en la plataforma. | | |
| **PRERREQUISITOS** | * El usuario debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Vista vwgde.php | | |
| **ENTRADA** | El proveedor o cliente se dirige a gestionar dirección de envío. | | |
| **ORÁCULO** | El sistema proporcionara dirección de envío. | | |
| **PASOS** | * El usuario accede a su cuenta en la plataforma. * El sistema muestra el panel de usuario. * El usuario selecciona la opción para gestionar direcciones de envío. * El sistema muestra una lista de direcciones guardadas. * El usuario puede realizar las siguientes acciones: * Agregar una nueva dirección: * El usuario selecciona la opción para agregar una nueva dirección. * El sistema muestra un formulario donde el usuario ingresa los detalles de la dirección. * El usuario completa el formulario y lo envía. * El sistema guarda la nueva dirección en el perfil del usuario. * Editar una dirección existente: * El usuario selecciona una dirección de la lista para editar. * El sistema muestra un formulario con los detalles actuales de la dirección. * El usuario realiza las modificaciones necesarias y envía el formulario. * El sistema actualiza la dirección en el perfil del usuario. * Eliminar una dirección: * El usuario selecciona una dirección de la lista para eliminar. * El sistema pide confirmación para eliminar la dirección. * El usuario confirma la eliminación. * El sistema elimina la dirección del perfil del usuario. * El sistema confirma los cambios realizados y actualiza la lista de direcciones. | | |
| **Módulos Asociados** | * Cliente * Proveedor | | |

| **NOMBRE** | **Registro usuario** | PRUEBAS | **PUN 3** |
| --- | --- | --- | --- |
| **PROPÓSITO** | Un usuario se registra en la plataforma para crear una cuenta nueva como cliente o proveedor. | | |
| **PRERREQUISITOS** | * El usuario debe tener acceso a una dirección de correo electrónico válida. | | |
| **UBICACIÓN** | Vista vwreusu.php | | |
| **ENTRADA** | El usuario accede a la página iniciar sección y dirigirse a registrar. | | |
| **ORÁCULO** | El sistema registra al usuario satisfactoria mente | | |
| **PASOS** | * El usuario accede a la página principal de la plataforma. * El usuario selecciona la opción para registrarse. * El sistema muestra un formulario de registro donde el usuario puede ingresar la siguiente información: * Nombre completo * Dirección de correo electrónico * Contraseña * Confirmación de contraseña * Rol deseado (Cliente o Proveedor) * El usuario completa el formulario de registro y envía la información. * El sistema valida la información proporcionada. * [Extensión 5a: Información incompleta o inválida] * Si la validación es exitosa, el sistema crea una nueva cuenta para el usuario y guarda la información en la base de datos. * El sistema envía un correo electrónico de confirmación a la dirección de correo electrónico proporcionada por el usuario. * El usuario accede a su correo electrónico y encuentra el mensaje de confirmación. * El usuario hace clic en el enlace de confirmación proporcionado en el correo electrónico. * El sistema verifica el enlace de confirmación y activa la cuenta del usuario. * El sistema muestra un mensaje de confirmación indicando que la cuenta ha sido creada y activada exitosamente, y redirige al usuario a la página de inicio de sesión. | | |
| **Módulos Asociados** | * Cliente * Proveedor (cuando actúa como cliente) |

| **NOMBRE** | **Ver balance de ventas** | PRUEBAS | **PUN 4** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El proveedor visualiza el balance de las ventas realizadas a través de la plataforma. | | |
| **PRERREQUISITOS** | * El proveedor debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Vista vwbv.php | | |
| **ENTRADA** | * El proveedor se dirige a la visualización del balance de ventas. | | |
| **ORÁCULO** | La plataforma muestra el balance de ventas. | | |
| **PASOS** | * El proveedor accede al panel de administración de su cuenta. * El sistema muestra las opciones de balance de ventas. * El proveedor selecciona la opción para ver el balance de ventas. * El sistema muestra un resumen del balance de ventas, incluyendo detalles como: * Total de ventas en un período específico. * Ingresos brutos y netos. * Comisiones aplicadas. * Detalles de ventas individuales (producto, cantidad, fecha, precio). * El proveedor puede filtrar y ordenar los datos del balance por diferentes criterios (por ejemplo, por fecha, por producto). * El proveedor selecciona una venta específica para ver más detalles si es necesario. * El sistema muestra los detalles completos de la venta seleccionada. | | |
| **Módulos Asociados** | * Proveedor | | |

| **NOMBRE** | **Gestionar formas de pago** | PRUEBAS | **PUN 5** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario (cliente o proveedor) gestiona sus formas de pago en la plataforma. | | |
| **PRERREQUISITOS** | * El usuario debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Vista gfp.php | | |
| **ENTRADA** | * El cliente/proveedor se ubica en la sesión de gestionar formas de pago. | | |
| **ORÁCULO** | El sistema gestiona las formas de pago | | |
| **PASOS** | * El usuario accede a su cuenta en la plataforma. * El sistema muestra el panel de usuario. * El usuario selecciona la opción para gestionar formas de pago. * El sistema muestra una lista de métodos de pago guardados. * El usuario puede realizar las siguientes acciones: * Agregar una nueva forma de pago: * El usuario selecciona la opción para agregar una nueva forma de pago. * El sistema muestra un formulario donde el usuario ingresa los detalles del método de pago (número de tarjeta, fecha de vencimiento, CVV, nombre del titular, etc.). * El usuario completa el formulario y lo envía. * El sistema guarda la nueva forma de pago en el perfil del usuario. * Editar una forma de pago existente: * El usuario selecciona una forma de pago de la lista para editar. * El sistema muestra un formulario con los detalles actuales del método de pago. * El usuario realiza las modificaciones necesarias y envía el formulario. * El sistema actualiza la forma de pago en el perfil del usuario. * Eliminar una forma de pago: * El usuario selecciona una forma de pago de la lista para eliminar. * El sistema pide confirmación para eliminar la forma de pago. * El usuario confirma la eliminación. * El sistema elimina la forma de pago del perfil del usuario. * El sistema confirma los cambios realizados y actualiza la lista de formas de pago. | | |
| **Módulos Asociados** | * Cliente * Proveedor | | |

| **NOMBRE** | **Buscar Producto** | PRUEBAS | **PUN 6** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario (cliente o proveedor actuando como cliente) busca productos en la plataforma utilizando diferentes criterios de búsqueda. | | |
| **PRERREQUISITOS** | * El usuario debe estar autenticado en la plataforma (opcional si la búsqueda básica está disponible sin iniciar sesión). | | |
| **UBICACIÓN** | Base de datos MYSQL y pantalla de vwBusPro.php | | |
| **ENTRADA** | El usuario accede a la página principal de la plataforma y busca  Medias  Tv  iphone | | |
| **ORÁCULO** | El sistema retorna el dato del producto buscado | | |
| **PASOS** | * El usuario ingresa palabras clave en la barra de búsqueda. * El sistema muestra resultados relevantes basados en las palabras clave ingresadas, incluyendo detalles básicos como el nombre del producto, la imagen, el precio y el proveedor. * El usuario puede refinar la búsqueda utilizando filtros adicionales, tales como: * Categorías de producto. * Rango de precios. * Calificaciones de usuarios. * Disponibilidad del producto. * Ofertas y descuentos. * El sistema actualiza los resultados de búsqueda según los filtros aplicados. * El usuario selecciona un producto específico de los resultados de búsqueda para ver más detalles. * El sistema muestra la página de detalles del producto seleccionado, incluyendo una descripción completa, especificaciones, imágenes adicionales, calificaciones y comentarios de otros usuarios, y opciones de compra. | | |
| **Módulos Asociados** | * Cliente * Proveedor | | |

### Proveedor

| NOMBRE | Gestionar Inventario | PRUEBAS | PUN 8 |
| --- | --- | --- | --- |
| PROPÓSITO | El proveedor gestiona el inventario de sus productos en la plataforma, incluyendo la actualización de cantidades y la adición o eliminación de productos. | | |
| PRERREQUISITOS | El proveedor debe estar autenticado en el sistema | | |
| UBICACIÓN | Base de datos MySQL y pantalla de vwDevRem.php | | |
| ENTRADA | * El administrador o proveedor accede al panel de administración del sistema. | | |
| ORÁCULO | El sistema debe permitir almacenar todos los datos proporcionados por el proveedor | | |
| PASOS | * El proveedor accede al panel de administración de su cuenta. * El sistema muestra las opciones de historial de ventas. * El proveedor selecciona la opción para ver el historial de ventas. * El sistema muestra una lista de todas las ventas realizadas por el proveedor, incluyendo detalles como ID del pedido, fecha de venta, productos vendidos, cantidad, precio total, y estado del pedido (pendiente, enviado, entregado). |
| Módulos Asociados | * Proveedor |

| **NOMBRE** | **VER KARDEX** | PRUEBAS | **PUN 9** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El proveedor puede ver el kardex de sus productos, que incluye información detallada sobre el inventario. | | |
| **PRERREQUISITOS** | * El proveedor debe estar autenticado en el sistema. * El proveedor debe tener productos registrados en la plataforma. | | |
| **UBICACIÓN** | Vista VRKX | | |
| **ENTRADA** | * El cliente/proveedor ingresa su correo registrado en el sistema. | | |
| **ORÁCULO** | El sistema logra ingresar a la vista del Kardex | | |
| **PASOS** | * El usuario accede a la página principal de la plataforma. * El sistema muestra una barra de búsqueda y opciones de filtrado. * El usuario ingresa palabras clave en la barra de búsqueda. * El sistema muestra resultados relevantes basados en las palabras clave ingresadas, incluyendo detalles básicos como el nombre del producto, la imagen, el precio y el proveedor. * El usuario puede refinar la búsqueda utilizando filtros | | |
| **Módulos Asociados** | * Cliente * Proveedor | | |

| **NOMBRE** | **REALIZAR ENVÍO** | PRUEBAS | **PUN 10** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El proveedor coordina y realiza el envío de un pedido recibido. | | |
| **PRERREQUISITOS** | * El proveedor debe estar autenticado en el sistema. * El proveedor debe tener un pedido pendiente de envío. | | |
| **UBICACIÓN** | Vista REN | | |
| **ENTRADA** | * El proveedor hace el proceso de envío | | |
| **ORÁCULO** | El provedor logra realizar el envío a traves del software | | |
| **PASOS** | * El proveedor accede al panel de administración de su cuenta. * El sistema muestra las opciones de gestión de pedidos. * El proveedor selecciona la opción para realizar el envío de un pedido. * El sistema muestra una lista de pedidos pendientes de envío. * El proveedor selecciona un pedido para realizar el envío. * El sistema muestra los detalles del pedido, incluyendo la información del cliente y la dirección de envío. * El proveedor ingresa los detalles del envío (por ejemplo, empresa de mensajería, número de seguimiento). * El proveedor confirma y envía la información de envío. * El sistema actualiza el estado del pedido a "Enviado" y notifica al cliente. | | |
| **Módulos Asociados** | * Proveedor | | |

| **NOMBRE** | **CANCELAR PEDIDOS** | PRUEBAS | **PUN 11** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario puede cancelar un pedido realizado en la plataforma. | | |
| **PRERREQUISITOS** | * El usuario debe estar autenticado en el sistema. * El pedido debe estar en un estado que permita su cancelación (por ejemplo, no enviado). | | |
| **UBICACIÓN** | Vista CPD | | |
| **ENTRADA** | * El cliente/proveedor cancela el pedido | | |
| **ORÁCULO** | El sistema logra cancelar el pedido correctamente | | |
| **PASOS** | * El usuario accede a su cuenta en la plataforma. * El sistema muestra el panel de usuario. * El usuario selecciona la opción para ver el historial de pedidos. * El sistema muestra una lista de pedidos del usuario. * El usuario selecciona un pedido para cancelar. * El sistema muestra los detalles del pedido y una opción para cancelar. * El usuario confirma la cancelación del pedido. * El sistema actualiza el estado del pedido a "Cancelado" y notifica a las partes involucradas (cliente, proveedor). * Si corresponde, el sistema inicia el proceso de reembolso. | | |
| **Módulos Asociados** | * Cliente * Proveedor * Administrador | | |

| **NOMBRE** | **SUBIR PRODUCTOS** | PRUEBAS | **PUN 11** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El proveedor sube un nuevo producto a la plataforma para su venta. | | |
| **PRERREQUISITOS** | * El proveedor debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Vista SPR | | |
| **ENTRADA** | * El proveedor debe subir sus productos correctamente | | |
| **ORÁCULO** | El sistema logra cancelar el pedido correctamente | | |
| **PASOS** | * El proveedor accede al panel de administración de su cuenta. * El sistema muestra las opciones de gestión de productos. * El proveedor selecciona la opción para subir un nuevo producto. * El sistema muestra un formulario para ingresar los detalles del producto, incluyendo: * Nombre del producto. * Descripción. * Categoría. * Precio. * Imágenes del producto. * Cantidad en inventario. * El proveedor completa el formulario con la información del producto. * El proveedor guarda y publica el producto. * El sistema valida la información ingresada y agrega el producto a la plataforma. * El sistema confirma la publicación del producto y lo hace visible para los clientes. | | |
| **Módulos Asociados** | * Proveedor | | |

| **NOMBRE** | **Ver historias de ventas** | PRUEBAS | **PUN 13** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El proveedor visualiza el historial de ventas de sus productos en la plataforma. | | |
| **PRERREQUISITOS** | El proveedor debe estar autenticado en el sistema.  Flujo Principal | | |
| **UBICACIÓN** | Base de datos MySQL y vista vwhven.php | | |
| **ENTRADA** | * El administrador o proveedor accede al panel de administración del sistema. | | |
| **ORÁCULO** | El sistema debe permitir ver el historial de ventas del proveedor | | |
| **PASOS** | * **El sistema muestra las opciones de historial de ventas.** * El proveedor selecciona la opción para ver el historial de ventas. * El sistema muestra una lista de todas las ventas realizadas por el proveedor, incluyendo detalles como ID del pedido, fecha de venta, productos vendidos, cantidad, precio total, y estado del pedido (pendiente, enviado, entregado). * El proveedor puede filtrar y ordenar la lista por diferentes criterios (por ejemplo, por fecha, por estado del pedido, por producto). * El proveedor selecciona una venta específica para ver más detalles. * El sistema muestra los detalles de la venta seleccionada, incluyendo la información del cliente, la dirección de envío, y el método de pago utilizado. |
| **Módulos Asociados** | * proveedor |

| **NOMBRE** | **Gestionar Perfil** | PRUEBAS | **PUN 14** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario (cliente o proveedor) gestiona su perfil, incluyendo la actualización de información básica, formas de pago y direcciones de envío. | | |
| **PRERREQUISITOS** | El proveedor debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Base de datos MySQL y vista vwgpef.php | | |
| **ENTRADA** | * El cliente o proveedor accede a su cuenta en plantaforma con el usuario: proveedro1 * Contraseña: proveedor2024 | | |
| **ORÁCULO** | El sistema debe permitir al usuario o proveedor ver y modificar su perfil | | |
| **PASOS** | * **El sistema muestra el panel de usuario.** * El usuario selecciona la opción para gestionar perfil. * El sistema muestra un formulario con la información actual del perfil, incluyendo detalles como nombre, dirección de correo electrónico, número de teléfono, formas de pago y direcciones de envío. * El usuario puede realizar las siguientes acciones: * Editar información básica: * El usuario selecciona la sección de información básica. * El sistema muestra un formulario con los detalles actuales. * El usuario realiza las modificaciones necesarias y envía el formulario. * El sistema valida y actualiza la información en el perfil. * Gestionar formas de pago: * El usuario selecciona la sección de formas de pago. * El sistema muestra una lista de métodos de pago guardados. * El usuario puede agregar, editar o eliminar métodos de pago. * El sistema guarda los cambios realizados. * Gestionar direcciones de envío: * El usuario selecciona la sección de direcciones de envío. * El sistema muestra una lista de direcciones guardadas. * El usuario puede agregar, editar o eliminar direcciones. |
| **Módulos Asociados** | * Proveedor cliente |

### Administrador

| NOMBRE | Gestionar usuarios | PRUEBAS | PUN 15 |
| --- | --- | --- | --- |
| PROPÓSITO | El administrador gestiona los usuarios registrados en la plataforma, incluyendo clientes y proveedores | | |
| PRERREQUISITOS | * El administrador debe estar autenticado y tener permisos adecuados para gestionar usuarios. | | |
| UBICACIÓN | Base de datos MySQL y pantalla vwgusu.php | | |
| ENTRADA | * + El administrador accede al panel de administración del sistema | | |
| ORÁCULO | El administrador gestiona a los usuarios registrados satisfactoriamente | | |
| PASOS | * El administrador accede al panel de administración del sistema. * El sistema muestra las opciones de gestión de usuarios. * El administrador selecciona la opción para gestionar usuarios. * El sistema muestra una lista de todos los usuarios registrados en la plataforma, incluyendo detalles como nombre, dirección de correo electrónico, rol (cliente o proveedor), estado de la cuenta (activa, inactiva), y fecha de registro. | | |
| Módulos Asociados | * Administrador | | |

| NOMBRE | Registrar Administrador | PRUEBAS | PUN 16[[2]](#footnote-1) |
| --- | --- | --- | --- |
| PROPÓSITO | Verificar si el perfil de administrador puede registrar otro administrador en la base de datos y dar acceso a la plataforma. | | |
| PRERREQUISITOS | * Debe estar autenticado en la base de datos. * Debe tener permisos adecuados para registrar otros administradores. | | |
| UBICACIÓN | Base de datos MySQL y pantalla de vwRegAdmin.php | | |
| ENTRADA | * Datos requeridos por el formulario.   + Nombre: Kevin   + Apellido: Galindo   + Tipo De Documento: C.C   + Número Documento: 104372732   + Correo: [kevingalindo@gmail.com](mailto:kevingalindo@gmail.com)   + Celular: 311 223 3211   + Género: Masculino | | |
| ORÁCULO | El administrador ha sido registrado con éxito. | | |
| PASOS | * Ingresar a la vista vwRegAdmin.php. * Digitar los datos solicitados por el formulario. * Dar Click en “Registrar”. | | |
| Módulos Asociados | * Administrador | | |

**Tabla 26: Caso de Prueba Unitaria 16**

| NOMBRE | Gestionar Comisiones | PRUEBAS | PUN 17 |
| --- | --- | --- | --- |
| PROPÓSITO | Verificar las comisiones obtenidas en la plataforma, generar reportes, editar valores, ver datos detalladamente, buscar, ordenar y filtrar diferentes tipos de criterios. | | |
| PRERREQUISITOS | * El administrador debe estar autenticado y tener permisos adecuados para gestionar comisiones. | | |
| UBICACIÓN | Base de datos MySQL y pantalla de vwGesCom.php | | |
| ENTRADA | * El administrador accede al panel de administración del sistema. | | |
| ORÁCULO | El sistema edita, guarda y retorna correctamente los valores solicitados o descritos por el administrador. | | |
| PASOS | * El administrador accede al panel de administración del sistema. * El sistema muestra las opciones de gestión financiera. * El administrador selecciona la opción para gestionar comisiones. * El sistema muestra una lista de todas las comisiones generadas por las ventas en la plataforma, incluyendo detalles como fecha, valor de la venta, porcentaje de comisión, y monto de la comisión. * El administrador puede filtrar y ordenar la lista por diferentes criterios (por ejemplo, por proveedor, por fecha, por monto de la comisión). * El administrador selecciona una comisión específica para ver más detalles. * El sistema muestra los detalles de la comisión seleccionada, incluyendo la transacción asociada y el proveedor correspondiente. * El administrador puede editar los detalles de la comisión si es necesario (ajustar el porcentaje de comisión). * El administrador guarda cualquier cambio realizado. * El sistema actualiza la base de datos con los cambios realizados y confirma la actualización. * El administrador puede generar un reporte de comisiones para un período específico. | | |
| Módulos Asociados | * Administrador | | |

**Tabla 27: Caso de Prueba Unitaria 17**

| **NOMBRE** | **Gestionar ganancias** | PRUEBAS | **PUN 18** |
| --- | --- | --- | --- |
| **PROPÓSITO** | Verifica que el proveedor pueda ver y gestionar las ganancias obtenidas de las ventas realizadas en la plataforma. | | |
| **PRERREQUISITOS** | * El proveedor debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Bases de datos MYsql | | |
| **ENTRADA** | * Se muestran las ganancias | | |
| **ORÁCULO** | El proveedor puede visualizar y gestionar la información de sus ganancias. | | |
| **PASOS** | El proveedor accede al panel de administración de su cuenta.  El sistema muestra las opciones de gestión financiera.  El proveedor selecciona la opción para gestionar ganancias.  El sistema muestra un resumen de las ganancias obtenidas, incluyendo detalles como:  Ingresos totales.  Comisiones y deducciones aplicadas.  Ganancias netas.  Transacciones individuales con detalles (producto, cantidad, fecha, precio).  El proveedor puede filtrar y ordenar los datos de las ganancias por diferentes criterios (por ejemplo, por fecha, por producto).  El proveedor puede descargar reportes de sus ganancias.  El sistema confirma y actualiza la vista del resumen de ganancias según las acciones realizadas. | | |
| **Módulos Asociados** | * Proveedor | | |

| **NOMBRE** | Realizar PQR | PRUEBAS | **PUN 19** |
| --- | --- | --- | --- |
| **PROPÓSITO** | Verificar si el usuario puede realizar peticiones, quejas y reclamos (PQR) en la plataforma. | | |
| **PRERREQUISITOS** | * El usuario debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Bases de datos MYsql | | |
| **ENTRADA** | * Ingresa archivo con la PQR | | |
| **ORÁCULO** | El historial de cambios ha sido actualizado | | |
| **PASOS** | Visitar la página de InfoReq ó RelacionesRequerimientos.  Realizar uno o más cambios en los requerimientos.  Clic en “modificar”.  Ingresar datos solicitados del historial (Responsable y Justificación)  Clic en “Aceptar”. | | |
| **Módulos Asociados** | * Cliente * Proveedor | | |

| **NOMBRE** | **Control vistas** | PRUEBAS | **PUN 20** |
| --- | --- | --- | --- |
| **PROPÓSITO** | Verificar si el administrador controla y visualiza las estadísticas de vistas y tráfico en la plataforma. | | |
| **PRERREQUISITOS** | * El administrador debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Bases de datos MYsql | | |
| **ENTRADA** | * Cambia el estado de las vistas | | |
| **ORÁCULO** | El administrador puede visualizar y analizar las estadísticas de vistas y tráfico en la plataforma. | | |
| **PASOS** | El administrador accede al panel de administración de la plataforma.  El sistema muestra las opciones de administración disponibles.  El administrador selecciona la opción para ver las estadísticas de vistas.  El sistema muestra un panel de control con las estadísticas de vistas y tráfico, que incluye:  Número total de visitas a la plataforma.  Visitas por página o sección (por ejemplo, página de inicio, categorías de productos, páginas de productos específicos).  Gráficos y tablas con datos históricos y tendencias de tráfico.  Datos demográficos y geográficos de los visitantes.  Fuentes de tráfico (por ejemplo, tráfico directo, motores de búsqueda, referencias).  El administrador puede aplicar filtros y segmentaciones para analizar los datos (por ejemplo, por rango de fechas, por tipo de dispositivo).  El administrador puede exportar los datos y reportes para un análisis más detallado.  El sistema confirma y actualiza la vista de las estadísticas según las acciones realizadas por el administrador. | | |
| **Módulos Asociados** | * Administrador | | |

## Pruebas De Uso

### Cliente

| **NOMBRE** | **RECIBIR NOTIFICACIONES** | PRUEBAS | **PUS 1** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario recibe notificaciones sobre eventos importantes en la plataforma. | | |
| **PRERREQUISITOS** | * El usuario debe estar autenticado en el sistema. * El usuario debe tener activadas las notificaciones en su perfil. | | |
| **UBICACIÓN** | Vista vwrcn.php | | |
| **ENTRADA** | * El sistema da alerta de notificación | | |
| **ORÁCULO** | El sistema logra cancelar el pedido correctamente | | |
| **PASOS** | * Un evento importante ocurre en la plataforma (por ejemplo, nuevo pedido, cambio de estado del pedido, mensaje del administrador). * El sistema genera una notificación correspondiente al evento. * El sistema envía la notificación al usuario a través de los canales configurados (por ejemplo, correo electrónico, notificaciones push, SMS). * El usuario recibe la notificación y puede acceder a los detalles a través del enlace proporcionado. * El sistema registra que la notificación ha sido enviada y recibida. | | |
| **Módulos Asociados** | * Cliente * Proveedor * Administrador | | |

| **NOMBRE** | **Configuración de notificaciones** | PRUEBAS | **PUS 2** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario configura sus preferencias de notificación en la plataforma. | | |
| **PRERREQUISITOS** | * El usuario debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Vista vwcdn.php | | |
| **ENTRADA** | * El cliente/proveedor o administrador se dirige al apartado de configuración. | | |
| **ORÁCULO** | El sistema muestra el apartado de configuración correctamente. | | |
| **PASOS** | * El usuario accede a su cuenta en la plataforma. * El sistema muestra el panel de usuario. * El usuario selecciona la opción para configurar notificaciones. * El sistema muestra un formulario con las opciones de notificación disponibles (por ejemplo, notificaciones de pedidos, promociones, mensajes del administrador). * El usuario selecciona sus preferencias de notificación, incluyendo: * Tipos de eventos para los cuales desea recibir notificaciones. * Canales de notificación preferidos (correo electrónico, notificaciones push, SMS). * El usuario guarda los cambios. * El sistema actualiza las preferencias de notificación del usuario en la base de datos. | | |
| **Módulos Asociados** | * Cliente * Proveedor * Administrador | | |

| **NOMBRE** | **Recibir Notificaciones de Pedido** | PRUEBAS | **PUS 3** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario (cliente o proveedor) recibe notificaciones relacionadas con sus pedidos en la plataforma. | | |
| **PRERREQUISITOS** | * El usuario debe estar autenticado en el sistema. * El usuario debe tener activadas las notificaciones en su perfil. | | |
| **UBICACIÓN** | Vista vwntf.php | | |
| **ENTRADA** | * El cliente se dirige al apartado de notificaciones. | | |
| **ORÁCULO** | Mostrar o Visualizar notificaciones correctamente | | |
| **PASOS** | * Un evento relacionado con un pedido ocurre en la plataforma (por ejemplo, un nuevo pedido, cambio de estado del pedido, etc.). * El sistema genera una notificación correspondiente al evento. * El sistema envía la notificación al usuario a través de los canales configurados (por ejemplo, correo electrónico, notificaciones push en la aplicación, SMS). * El usuario recibe la notificación y puede acceder a los detalles del pedido a través del enlace proporcionado en la notificación. * El sistema registra que la notificación ha sido enviada y recibida. | | |
| **Módulos Asociados** | * Cliente * Proveedor | | |

| NOMBRE | Recuperar Contraseña | PRUEBAS | PUS 4 |
| --- | --- | --- | --- |
| PROPÓSITO | El cliente/proveedor o administrador es capaz de recuperar su contraseña en caso de pérdida o ser olvidada. | | |
| PRERREQUISITOS | * El cliente/proveedor o administrador debe estar autenticado en el sistema. | | |
| UBICACIÓN | Base de datos MySQL y pantalla de vwRecCom.php | | |
| ENTRADA | * Kevingalindo@gmail.com | | |
| ORÁCULO | El sistema retorna todas las compras realizadas a través del tiempo. | | |
| PASOS | * El usuario accede a la página de inicio de sesión del sistema. * El usuario hace clic en el enlace o botón de recuperación de contraseña. * El usuario proporciona su dirección de correo electrónico y envía la solicitud de recuperación: [kevingalindo@gmail.com](mailto:kevingalindo@gmail.com). * Si la dirección de correo electrónico es válida, el sistema genera un token único y lo envía al correo electrónico del usuario junto con instrucciones para restablecer la contraseña. * El usuario accede a su correo electrónico, encuentra el correo electrónico de recuperación y hace clic en el enlace proporcionado para restablecer la contraseña. * El sistema redirige al usuario a una página donde puede ingresar una nueva contraseña. * El usuario ingresa y confirma la nueva contraseña. * El sistema confirma al usuario que la contraseña ha sido actualizada con éxito y redirige al usuario a la página de inicio de sesión para acceder con la nueva contraseña. | | |
| Módulos Asociados | * Cliente * Proveedor * Administrador | | |

**Tabla 34: Caso de Prueba de Uso 4**

| **NOMBRE** | **Login** | PRUEBAS | **PUS 5** |
| --- | --- | --- | --- |
| **PROPÓSITO** | Verifica que el usuario inicie sesión en la plataforma. | | |
| **PRERREQUISITOS** | * El usuario debe estar registrado en la plataforma | | |
| **UBICACIÓN** | Bases de datos MYsql | | |
| **ENTRADA** | * Ingresa usuario * Ingresa contraseña | | |
| **ORÁCULO** | El usuario está autenticado y tiene acceso a su panel principal. | | |
| **PASOS** | El usuario navega a la página de inicio de sesión de la plataforma.  El sistema muestra el formulario de inicio de sesión.  El usuario ingresa su correo electrónico y contraseña.  El usuario envía el formulario de inicio de sesión.  El sistema verifica las credenciales del usuario.  Si las credenciales son correctas, el sistema autentica al usuario y redirige al panel principal correspondiente a su rol (cliente, proveedor o administrador). | | |
| **Módulos Asociados** | * Cliente * Proveedor * Administrador | | |

| **NOMBRE** | **Cerrar sesión** | PRUEBAS | **PUS 6** |
| --- | --- | --- | --- |
| **PROPÓSITO** | Verificar si al hacer un cambio en los requerimientos, se pide el registro de cambios y actualiza el historial de cambios en la base de datos. | | |
| **PRERREQUISITOS** | * El usuario debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Bases de datos MYsql | | |
| **ENTRADA** | * Hace el cierre de sesión | | |
| **ORÁCULO** | El usuario cierra sesión y la autenticación se termina. | | |
| **PASOS** | El usuario selecciona la opción para cerrar sesión en el panel de usuario.  El sistema solicita confirmación para cerrar sesión.  El usuario confirma la acción de cerrar sesión.  El sistema termina la sesión del usuario y lo redirige a la página de inicio. | | |
| **Módulos Asociados** | * Cliente * Proveedor * Administrador | | |

| **NOMBRE** | **Gestionar configuración plataforma** | PRUEBAS | **PUS 7** |
| --- | --- | --- | --- |
| **PROPÓSITO** | Verificar si el administrador gestiona la configuración general de la plataforma. | | |
| **PRERREQUISITOS** | * El administrador debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Bases de datos MYsql | | |
| **ENTRADA** | * Ingresa la configuración que desea hacer | | |
| **ORÁCULO** | Las configuraciones de la plataforma se actualizan según los cambios realizados por el administrador. | | |
| **PASOS** | El administrador accede al panel de administración de la plataforma.  El sistema muestra las opciones de configuración.  El administrador selecciona la opción para gestionar la configuración de la plataforma.  El sistema muestra un formulario con las configuraciones disponibles (por ejemplo, configuración de comisiones, métodos de pago aceptados, políticas de envío y devolución).  El administrador realiza los cambios necesarios en lasconfiguraciones.  El administrador guarda los cambios.  El sistema actualiza las configuraciones de la plataforma y confirma los cambios realizados. | | |
| **Módulos Asociados** | * Administrador | | |

| **NOMBRE** | **GENERAR REPORTE** | PRUEBAS | **PUS 8** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario genera reportes detallados sobre diversas actividades en la plataforma. | | |
| **PRERREQUISITOS** | * El usuario debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Vista grr.php | | |
| **ENTRADA** | * El proveedor/administrador entra al apartado de detalles | | |
| **ORÁCULO** | * El sistema genera el reporte | | |
| **PASOS** | * El usuario accede al panel de administración de su cuenta. * El sistema muestra las opciones de generación de reportes. * El usuario selecciona el tipo de reporte que desea generar (por ejemplo, ventas, inventario, tráfico). * El sistema muestra un formulario para definir los parámetros del reporte (por ejemplo, rango de fechas, filtros específicos). * El usuario completa el formulario con los parámetros deseados. * El usuario envía la solicitud para generar el reporte. * El sistema procesa la solicitud y genera el reporte. * El sistema muestra el reporte generado, permitiendo al usuario visualizarlo en pantalla y/o descargarlo en formato PDF o Excel. | | |
| **Módulos Asociados** | * Administrador * Proveedor | | |

| **NOMBRE** | **Buscar Productos** | **PRUEBAS** | **PUS 9** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario busca productos en la plataforma utilizando filtros y parámetros de búsqueda específicos. | | |
| **PRERREQUISITOS** | El usuario debe estar autenticado en el sistema y tener acceso al catálogo de productos. | | |
| **UBICACIÓN** | vwHeader.php | | |
| **ENTRADA** | El usuario entra en el apartado de búsqueda de productos. | | |
| **ORÁCULO** | El sistema muestra los productos que coinciden con los parámetros de búsqueda establecidos por el usuario. | | |
| **PASOS** | · El usuario accede al panel de búsqueda de productos en su cuenta.  · El sistema muestra las opciones de búsqueda y filtros disponibles.  · El usuario selecciona o introduce los filtros deseados (por ejemplo, nombre del producto, categoría, rango de precio).  · El sistema muestra un formulario para ajustar los filtros o parámetros adicionales.  · El usuario completa el formulario con los criterios de búsqueda.  · El usuario envía la solicitud de búsqueda.  · El sistema muestra los productos que coinciden con los filtros aplicados. | | |
| **MODULOS ASOCIADOS** | · Administrador  · Proveedor  · Cliente | | |

| **NOMBRE** | **Ver más vendidos** | **PRUEBAS** | **PUS 10** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario consulta los productos más vendidos de la plataforma. | | |
| **PRERREQUISITOS** | El usuario debe estar autenticado en el sistema y tener acceso a la sección de productos más vendidos. | | |
| **UBICACIÓN** | vwTienda.php | | |
| **ENTRADA** | El usuario entra en la sección de productos más vendidos. | | |
| **ORÁCULO** | El sistema genera el reporte correspondiente. | | |
| **PASOS** | · El usuario accede al panel de productos más vendidos en su cuenta.  · El sistema muestra una lista con los productos más vendidos.  · El usuario puede aplicar filtros adicionales si lo desea (por ejemplo, por categoría o precio).  · El sistema actualiza la lista de productos mostrados en función de los filtros aplicados. | | |
| **MODULOS ASOCIADOS** | · Administrador  · Proveedor  · Cliente | | |

| **NOMBRE** | **Ver Ofertas** | **PRUEBAS** | **PUS 11** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario visualiza las ofertas o descuentos actuales disponibles en la plataforma. | | |
| **PRERREQUISITOS** | El usuario debe estar autenticado en el sistema y tener acceso a la sección de ofertas. | | |
| **UBICACIÓN** | vwTienda.php | | |
| **ENTRADA** | El usuario accede a la sección de ofertas. | | |
| **ORÁCULO** | El sistema muestra las ofertas disponibles, incluyendo productos con descuentos o promociones. | | |
| **PASOS** | · El usuario entra a la sección de ofertas en su cuenta.  · El sistema muestra una lista de productos en oferta.  · El usuario puede aplicar filtros por tipo de oferta o categoría.  · El sistema actualiza la lista de productos mostrados según los filtros aplicados. | | |
| **MODULOS ASOCIADOS** | · Administrador  · Proveedor  · Cliente | | |

| **NOMBRE** | **Ver productos vistos recientemente** | **PRUEBAS** | **PUS 12** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario consulta los productos que ha visto recientemente en la plataforma. | | |
| **PRERREQUISITOS** | El usuario debe estar autenticado en el sistema y haber visualizado productos previamente. | | |
| **UBICACIÓN** | vwTienda.php | | |
| **ENTRADA** | El usuario entra en la sección de productos vistos recientemente. | | |
| **ORÁCULO** | El sistema muestra una lista de los productos que el usuario ha visto recientemente. | | |
| **PASOS** | · El usuario accede a la sección de productos vistos recientemente.  · El sistema muestra los productos que el usuario ha visualizado en las últimas sesiones.  · El usuario puede hacer clic en los productos para obtener más detalles. | | |
| **MODULOS ASOCIADOS** | · Administrador  · Proveedor  · Cliente | | |

| **NOMBRE** | **Categorías del Producto** | **PRUEBAS** | **PUS 13** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario consulta las diferentes categorías de productos disponibles en la plataforma. | | |
| **PRERREQUISITOS** | El usuario debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | vwTienda.php | | |
| **ENTRADA** | El usuario entra a la sección de categorías del producto. | | |
| **ORÁCULO** | El sistema muestra una lista de categorías de productos disponibles para explorar. | | |
| **PASOS** | · El usuario accede a la sección de categorías.  · El sistema muestra una lista de categorías disponibles.  · El usuario selecciona una categoría para ver los productos relacionados.  · El sistema muestra los productos de la categoría seleccionada. | | |
| **MODULOS ASOCIADOS** | · Administrador  · Proveedor  · Cliente | | |

| **NOMBRE** | **Ayuda y Soporte** | **PRUEBAS** | **PUS 14** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario solicita asistencia o consulta la sección de preguntas frecuentes para resolver dudas. | | |
| **PRERREQUISITOS** | El usuario debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | vwSoport.php | | |
| **ENTRADA** | El usuario accede a la sección de Ayuda y Soporte. | | |
| **ORÁCULO** | El sistema muestra opciones de ayuda o un formulario de contacto para resolver dudas o problemas. | | |
| **PASOS** | · El usuario entra a la sección de Ayuda y Soporte.  · El sistema muestra un conjunto de preguntas frecuentes o un formulario de contacto.  · El usuario selecciona una pregunta frecuente o rellena el formulario de contacto si no encuentra respuesta.  · El sistema genera una respuesta automatizada o deriva el problema a un agente de soporte. | | |
| **MODULOS ASOCIADOS** | · Administrador  · Proveedor  · Cliente | | |

| **NOMBRE** | **Autenticación y autorización** | **PRUEBAS** | **PUS 15** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario edita y actualiza su perfil personal en la plataforma. | | |
| **PRERREQUISITOS** | El usuario debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | vwPerf.php | | |
| **ENTRADA** | El usuario entra a la sección de gestión de perfil. | | |
| **ORÁCULO** | El sistema muestra los datos del perfil del usuario y permite actualizarlos. | | |
| **PASOS** | · El usuario accede a la sección de gestionar perfil.  · El sistema muestra los detalles actuales del perfil del usuario (nombre, correo, dirección, etc.).  · El usuario realiza las modificaciones deseadas en el perfil.  · El usuario guarda los cambios realizados.  · El sistema confirma que el perfil ha sido actualizado correctamente. | | |
| **MODULOS ASOCIADOS** | · Administrador  · Proveedor  · Cliente | | |

## Pruebas De Integración

### Cliente

| **NOMBRE** | **Compra de producto** | PRUEBAS | **PIN 1** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El usuario compra un producto en la plataforma. | | |
| **PRERREQUISITOS** | * El usuario debe estar autenticado en el sistema. * El usuario debe tener al menos un método de pago y una dirección de envío registrados. | | |
| **UBICACIÓN** | Vista vwcdp.php | | |
| **ENTRADA** | El cliente se dirige a la sesión de comprar producto. | | |
| **ORÁCULO** | El cliente realiza una compra correctamente. | | |
| **PASOS** | * El usuario navega por la plataforma y selecciona un producto para comprar. * El sistema muestra la página de detalles del producto. * El usuario selecciona la opción para agregar el producto al carrito de compras. * El sistema agrega el producto al carrito de compras y muestra una confirmación. * El usuario accede al carrito de compras y revisa los productos agregados. * El usuario selecciona la opción para proceder al pago. * El sistema muestra una pantalla de pago donde el usuario puede seleccionar la dirección de envío y el método de pago. * El usuario confirma la dirección de envío y el método de pago. * El usuario revisa el resumen del pedido y confirma la compra. * El sistema procesa el pago. * [Extensión 10a: Pago rechazado] * Si el pago es exitoso, el sistema confirma la compra y envía una notificación al usuario con los detalles del pedido. * El sistema actualiza el inventario y registra la venta en el historial del proveedor. | | |
| **Módulos Asociados** | * Cliente * Proveedor (cuando actúa como cliente) | | |

| NOMBRE | Ver Historial Compras | PRUEBAS | PIN 2 |
| --- | --- | --- | --- |
| PROPÓSITO | El cliente o proveedor (cuando actúa como cliente) visualiza el historial de sus compras en la plataforma. | | |
| PRERREQUISITOS | * El cliente debe estar autenticado en el sistema * El cliente debe haber realizado algún tipo de compra. | | |
| UBICACIÓN | Base de datos MySQL y pantalla de vwHisCom.php | | |
| ENTRADA | * El cliente realiza una compra y observa esta compra en el historial. | | |
| ORÁCULO | El sistema retorna todas las compras realizadas a través del tiempo. | | |
| PASOS | * El cliente accede a su cuenta en la plataforma. * El sistema muestra el panel de usuario. * El cliente selecciona la opción para ver el historial de compras. * El cliente filtra las compras dependiendo la fecha, categoría de producto o estado del pedido. * El cliente selecciona un pedido específico para ver más detalles. * El cliente puede descargar una factura o recibo para el pedido seleccionado. * El cliente puede contactar al soporte o iniciar una solicitud de devolución o reembolso directamente desde la vista detallada del pedido si es necesario. | | |
| Módulos Asociados | * Cliente | | |

**Tabla 40: Caso de Prueba Integral 2.**

| NOMBRE | Calificar y Comentar Productos | PRUEBAS | PIN 3 |
| --- | --- | --- | --- |
| PROPÓSITO | Añadir una descripción, opinión y/o calificación a un producto despues de haberlo adquirido. | | |
| PRERREQUISITOS | * El cliente debe estar autenticado en el sistema. * El cliente debe haber comprado el producto que desea calificar y comentar. | | |
| UBICACIÓN | Base de datos MySQL y pantalla de vwCalPrd.php | | |
| ENTRADA | * El usuario adquiere un producto y procede a dar su opinión. | | |
| ORÁCULO | El sistema guarda la calificación u opinión el producto y es publicada en las reseñas del artículo. | | |
| PASOS | * El cliente accede a su cuenta en la plataforma. * El sistema muestra el panel de usuario. * El cliente selecciona la opción para ver sus pedidos completados. * El sistema muestra una lista de pedidos completados con los productos comprados. * El cliente selecciona un producto específico que desea calificar y comentar. * El cliente selecciona la opción para agregar una calificación y comentario. * El cliente ingresa la calificación y escribe el comentario. * El cliente envía el formulario. * El sistema muestra un mensaje de confirmación indicando que la calificación y el comentario han sido registrados con éxito y son visibles en la página del producto. | | |
| Módulos Asociados | * Cliente | | |

**Tabla 41: Caso de Prueba de Integración 3.**

| **NOMBRE** | **Gestionar productos** | PRUEBAS | **PIN 4** |
| --- | --- | --- | --- |
| **PROPÓSITO** | El proveedor gestiona sus productos en la plataforma, pudiendo actualizar, eliminar y ver detalles de los productos. | | |
| **PRERREQUISITOS** | * El proveedor debe estar autenticado en el sistema. | | |
| **UBICACIÓN** | Vista vwgpr.php | | |
| **ENTRADA** | * El sistema gestiona los productos del sistema | | |
| **ORÁCULO** | El sistema retorna todos los productos | | |
| **PASOS** | * El proveedor accede al panel de administración de su cuenta. * El sistema muestra las opciones de gestión de productos. * El proveedor selecciona la opción para gestionar productos. * El sistema muestra una lista de productos del proveedor. * El proveedor puede realizar las siguientes acciones: * Actualizar Producto: * El proveedor selecciona un producto para actualizar. * El sistema muestra un formulario con los detalles actuales del producto. * El proveedor actualiza la información del producto (por ejemplo, precio, descripción, imágenes). * El proveedor guarda los cambios. * El sistema valida y guarda los cambios. * Eliminar Producto: * El proveedor selecciona un producto para eliminar. * El sistema solicita confirmación para eliminar el producto. * El proveedor confirma la eliminación. * El sistema elimina el producto de la plataforma. * Ver Detalles de Producto: * El proveedor selecciona un producto para ver sus detalles. * El sistema muestra la información completa del producto. | | |
| **Módulos Asociados** | * Proveedor | | |

### Proveedor

| NOMBRE | Gestionar Devoluciones y Reembolsos | PRUEBAS | PIN 5 |
| --- | --- | --- | --- |
| PROPÓSITO | Verificar las devoluciones y reembolsos ocasionados por la venta de productos en la plataforma, editar valores, ver datos detalladamente, buscar, ordenar y filtrar diferentes tipos de criterios. | | |
| PRERREQUISITOS | * Debe estar autenticado en la base de datos. * Debe estar autenticado y tener permisos adecuados para gestionar devoluciones y reembolsos. | | |
| UBICACIÓN | Base de datos MySQL y pantalla de vwDevRem.php | | |
| ENTRADA | * El administrador o proveedor accede al panel de administración del sistema. | | |
| ORÁCULO | El sistema debe retornar, filtrar y buscar correctamente dependiendo se su ID, fecha, estado o motivo.  El sistema debe permitir cambiar el estado de devolución o reembolso ya sea aceptada o rechazada.  El sistema actualiza el estado de la devolución y reembolso.  El sistema debe iniciar con el proceso de devolución y reembolso según sea el caso.  El sistema debe notificar al cliente sobre el estado y procesos de su devolución y reembolso. | | |
| PASOS | * El administrador o proveedor accede al panel de administración del sistema. * El sistema muestra las opciones de gestión de pedidos. * El administrador o proveedor selecciona la opción para gestionar devoluciones y reembolsos. * El sistema muestra una lista de solicitudes de devolución y reembolso, incluyendo detalles como ID del pedido, producto, cliente, motivo de la devolución, y estado de la solicitud. * El administrador o proveedor puede filtrar y ordenar la lista por diferentes criterios (por ejemplo, por fecha, por estado, por motivo). * El administrador o proveedor selecciona una solicitud específica para ver más detalles. * El sistema muestra los detalles de la solicitud seleccionada, incluyendo la transacción original, el producto devuelto, el motivo de la devolución y cualquier comunicación previa con el cliente. * El administrador o proveedor revisa la solicitud y decide si aprueba o rechaza la devolución y el reembolso. * Si la solicitud es aprobada: * El sistema actualiza el estado de la solicitud a "Aprobada". * El sistema inicia el proceso de reembolso según el método de pago original. * El sistema notifica al cliente sobre la aprobación y el inicio del reembolso. * Si la solicitud es rechazada: * El sistema actualiza el estado de la solicitud a "Rechazada". * El sistema notifica al cliente sobre el rechazo y proporciona una razón para la decisión. * El administrador o proveedor puede agregar notas o comentarios adicionales a la solicitud. * El sistema registra todas las acciones y actualizaciones en la base de datos. | | |
| Módulos Asociados | * Administrador * Proveedor | | |

**Tabla 43: Caso de Prueba Integral 5.**

## Pruebas De Seguridad

| NOMBRE | Encriptación de claves de seguridad | PRUEBAS | PSE 1 |
| --- | --- | --- | --- |
| PROPÓSITO | Encriptar las contraseñas registradas en el sistema para que sean difíciles de desencriptar. | | |
| PRERREQUISITOS | * Existencia de base de datos. * Usuario registrado correctamente. | | |
| UBICACIÓN | Base de datos MySQL | | |
| ENTRADA | * Contraseña: proyectoSena | | |
| ORÁCULO | El sistema debe encriptar la contraseña y retornar una cadena de caracteres diferentes. | | |
| PASOS | * El usuario ingresa al registro de usuario. * El usuario digita la contraseña. * El sistema encripta la contraseña y la guarda en la base de datos. | | |
| Módulos Asociados | * Cliente * Proveedor * Administrador | | |

**Tabla 44: Caso de Prueba de Seguridad 1.**

| NOMBRE | Encapsulamiento de datos | PRUEBAS | PSE 2 |
| --- | --- | --- | --- |
| PROPÓSITO | Garantizar que los datos sensibles estén correctamente encapsulados y protegidos dentro de las aplicaciones. | | |
| PRERREQUISITOS | * Existencia de base de datos. * Código ya codificado. | | |
| UBICACIÓN | Base de datos MySQL | | |
| ENTRADA | * Codigo php | | |
| ORÁCULO | Los datos que se pasan por el modelo y controlador deben estar encapsulados correctamente. | | |
| PASOS | * Se ingresa al código fuente. * Se verifica la creación de getters y setters. * Se verifica que los datos tengan modificadores de acceso. * Se verifica que los objetos sean inmutables. | | |
| Módulos Asociados | * Cliente * Proveedor * Administrador | | |

**Tabla 45: Caso de Prueba de Seguridad 2.**

| NOMBRE | SQL Injection | PRUEBAS | PSE 3 |
| --- | --- | --- | --- |
| PROPÓSITO | Detectar y corregir vulnerabilidades donde un atacante podría ejecutar comandos SQL no autorizados. | | |
| PRERREQUISITOS | * Existencia de base de datos. * Usuario registrado correctamente. | | |
| UBICACIÓN | Base de datos MySQL | | |
| ENTRADA | * Usuario: ‘admin’ --’ * Contraseña: proyectoSena. * Formularios:   + OR 1=1   + ping -c 4 127.0.0.1; ls | | |
| ORÁCULO | El sistema No debe permitir el ingreso de codigo sql mediante | | |
| PASOS | * El usuario ingresa datos de login con el ataque sql. * El sistema no retorna ningún dato solicitado por la consulta. | | |
| Módulos Asociados | * Cliente * Proveedor * Administrador | | |

**Tabla 46: Caso de Prueba de Seguridad 3.**

| NOMBRE | Validación y Sanitización de Entradas | PRUEBAS | PSE 4 |
| --- | --- | --- | --- |
| PROPÓSITO | Prevenir ataques basados en la manipulación de entradas (como XSS, CSRF). | | |
| PRERREQUISITOS | * Comprender el Contexto de Uso de las Entradas. * Usuario registrado correctamente. | | |
| UBICACIÓN | No presenta vulnerabilidades de Cross-Site Scripting (XSS) y Cross-Site Request Forgery (CSRF). | | |
| ENTRADA | XSS  <a href="http://example.com/search?q=<script>alert('XSS');</script>">Haz clic aquí</a>  CSRF  <img src="http://example.com/transfer?amount=1000&to=attacker\_account" style="display:none;"> | | |
| ORÁCULO | El sistema No debe permitir el ingreso de Inyectar scripts maliciosos y enviar solicitudes manipuladas. | | |
| PASOS | * El usuario ingresa un script malicioso en una entrada de datos (Input). * El usuario ingresa una solicitud manipulada en una entrada de datos (Input). | | |
| Módulos Asociados | * Cliente * Proveedor * Administrador | | |

**Tabla 47: Caso de Prueba de Seguridad 4.**

| NOMBRE | Manejo de Errores | PRUEBAS | PSE 5 |
| --- | --- | --- | --- |
| PROPÓSITO | Prevenir caída de la plataforma debido a posibles errores generados mediante la ejecución. | | |
| PRERREQUISITOS | * Comprender el Entorno de la Aplicación. | | |
| UBICACIÓN | Base de datos MySQL y código fuente | | |
| ENTRADA | * Tipo de Dato erróneo: ‘uno’ | | |
| ORÁCULO | El sistema debe seguir funcionando aún si hay un error en algun sitio especifico y alertar del error. | | |
| PASOS | * El usuario ingresa un tipo de dato que no es correspondiente con el solicitado por el formulario y la base de datos. Ejemplo. Número entero por string. | | |
| Módulos Asociados | * Cliente * Proveedor * Administrador | | |

**Tabla 48: Caso de Prueba de Seguridad 5.**

| **NOMBRE** | **Autenticación y autorización** | **PRUEBAS** | **PSE 6** |
| --- | --- | --- | --- |
| **PROPÓSITO** | **Garantizar que el sistema permita el acceso únicamente a usuarios autorizados, probar la robustez de los métodos de autenticación y validar que los roles y permisos se asignen y gestionen correctamente.** | | |
| **PRERREQUISITOS** | **Conocer los métodos de autenticación implementados en el sistema.**  **Tener credenciales válidas e inválidas, así como usuarios con diferentes roles.** | | |
| **UBICACIÓN** | **Base de datos MySQL y módulo de autenticación en el código fuente.** | | |
| **ENTRADA** | **Usuario: cliente1**    **Contraseña: Pass123!**    **Rol: Cliente**    **Usuario: admin1**    **Contraseña: Admin789#**    **Rol: Administrador** | | |
| **ORÁCULO** | **Solo usuarios con credenciales válidas deben acceder al sistema.**  **Los métodos de autenticación deben rechazar accesos no autorizados.**  **Los roles deben restringir o permitir funciones y datos según el perfil del usuario.** | | |
| **PASOS** | **· Ingresar credenciales válidas para diferentes tipos de usuarios y verificar el acceso exitoso.**  **· Intentar acceder al sistema con credenciales incorrectas y comprobar que el sistema rechace el acceso.**  **· Activar la autenticación de dos factores (si está disponible) e intentar iniciar sesión sin completar el segundo paso.**  **· Validar que un usuario con rol de "Cliente" no pueda acceder a funciones exclusivas de "Administrador".**  **· Confirmar que los roles configurados habiliten solo las acciones asignadas a cada perfil.** | | |
| **MODULOS ASOCIADOS** | **· Cliente**  **· Proveedor**  **· Administrador** | | |

| **NOMBRE** | **Control de acceso** | **PRUEBAS** | **PSE 7** |
| --- | --- | --- | --- |
| **PROPÓSITO** | **Comprobar que las restricciones de acceso a datos y funciones estén implementadas correctamente y que los usuarios no puedan acceder a recursos fuera de sus permisos.** | | |
| **PRERREQUISITOS** | **Tener configurados roles y permisos en el sistema.**  **Contar con credenciales de usuarios con diferentes niveles de acceso.** | | |
| **UBICACIÓN** | **Base de datos MySQL y código fuente encargado de la gestión de permisos.** | | |
| **ENTRADA** | **Datos de prueba:**    **Usuario con permisos limitados:**    **Usuario: cliente1**  **Contraseña: Pass123!**  **Rol: Cliente**  **Acciones probadas:**    **Intentar acceder a la página de administración: shoop/home.php?pg=22**  **Intentar modificar información de otro usuario desde: shoop/home.php?pg=15&idusu=2.**  **Visualizar detalles de órdenes de otros clientes en: shoop/home.php?pg=17&idusu=2** | | |
| **ORÁCULO** | **Los usuarios deben acceder únicamente a los recursos y funciones permitidas por su rol.**  **El sistema debe impedir accesos no autorizados, redirigiendo al usuario o mostrando un mensaje de error adecuado.**  **Los intentos de acceso no autorizado deben registrarse en el sistema.** | | |
| **PASOS** | **· Iniciar sesión con cada uno de los usuarios de prueba.**  **· Intentar realizar acciones restringidas según el rol y verificar que el sistema las bloquee.**  **· Confirmar que los usuarios con permisos limitados no puedan visualizar ni modificar recursos que no les pertenecen.**  **· Asegurar que el usuario con permisos elevados (Administrador) pueda acceder a todas las funciones sin restricciones.**  **· Revisar los logs del sistema para verificar que se registraron los intentos de acceso no autorizados.** | | |
| **MODULOS ASOCIADOS** | **· Cliente**  **· Proveedor**  **· Administrador** | | |

| **NOMBRE** | **Fuerza bruta** | **PRUEBAS** | **PSE 8** |
| --- | --- | --- | --- |
| **PROPÓSITO** | **Simular intentos masivos de acceso no autorizado para evaluar la resistencia del sistema ante ataques de fuerza bruta y verificar si existen medidas de mitigación, como bloqueos temporales tras múltiples intentos fallidos.** | | |
| **PRERREQUISITOS** | **Tener configurado un sistema de autenticación en la aplicación.**  **Contar con herramientas para automatizar los intentos de acceso (por ejemplo, scripts o herramientas como Burp Suite o Hydra).**  **Entender las políticas de bloqueo implementadas en el sistema.** | | |
| **UBICACIÓN** | **Módulo de autenticación en la aplicación** | | |
| **ENTRADA** | **Datos de prueba:**    **Usuario: cliente1**    **Contraseña: Intento1**  **(Repetir hasta generar al menos 50 intentos consecutivos con contraseñas aleatorias).**    **Credenciales válidas:**    **Usuario: cliente1**  **Contraseña: Pass123!**  **(Probar acceso después de realizar múltiples intentos fallidos para verificar si el bloqueo temporal afecta al usuario legítimo.)** | | |
| **ORÁCULO** | **El sistema debe bloquear temporalmente al usuario o la IP después de un número definido de intentos fallidos consecutivos.**  **Debe mostrarse un mensaje indicando el bloqueo, como "Demasiados intentos fallidos, intente nuevamente en X minutos".**  **El sistema debe permitir nuevamente el acceso después del tiempo de bloqueo o tras acciones específicas (como restablecer contraseña).** | | |
| **PASOS** | **· Enviar múltiples solicitudes de inicio de sesión consecutivas utilizando credenciales incorrectas.**  **· Observar la respuesta del sistema tras X intentos (por ejemplo, 10 o más).**  **· Intentar iniciar sesión con credenciales válidas y verificar si el sistema permite el acceso o aplica un bloqueo temporal al usuario o la IP.**  **· Realizar pruebas adicionales desde otra IP para confirmar que el bloqueo sea específico a la cuenta/IP, no global.**  **· Revisar los logs de seguridad del sistema para confirmar el registro de los intentos fallidos.** | | |
| **MODULOS ASOCIADOS** | **· Cliente**  **· Seguridad del sistema**  **· Logs y monitoreo** | | |

| **NOMBRE** | **Pruebas de vulnerabilidad** | **PRUEBAS** | **PSE 9** |
| --- | --- | --- | --- |
| **PROPÓSITO** | **Identificar problemas de seguridad conocidos en el sistema mediante herramientas de escaneo y generar informes detallados sobre las vulnerabilidades detectadas.** | | |
| **PRERREQUISITOS** | **Tener acceso al sistema para ejecutar las herramientas de análisis.**  **Contar con herramientas específicas como OWASP ZAP, Nessus, Burp Suite o Nikto.**  **Entender las configuraciones de red, aplicaciones y base de datos para interpretar los resultados.** | | |
| **UBICACIÓN** | **Se enfoca en el sistema completo: servidor, base de datos y módulos de código fuente.** | | |
| **ENTRADA** | **Datos de prueba:**    **URL o IP del servidor de prueba (por ejemplo: 192.168.1.10).**  **Credenciales necesarias para áreas autorizadas (si aplica).**  **Configuración básica de herramientas de escaneo:**  **Alcance: Escaneo completo o parcial.**  **Exclusiones: Áreas sensibles donde el escaneo podría causar fallos no deseados.** | | |
| **ORÁCULO** | **Todas las vulnerabilidades detectadas deben registrarse en un informe con detalles claros, como nivel de riesgo (bajo, medio, alto), ubicación y descripción.**  **El sistema no debe permitir accesos no autorizados ni mostrar datos confidenciales durante el escaneo.** | | |
| **PASOS** | **· Configurar y ejecutar herramientas de escaneo en el sistema para detectar:**  **· Inyecciones SQL.**  **· Configuraciones de seguridad incorrectas.**  **· Exposición de datos sensibles.**  **· XSS (Cross-Site Scripting).**  **· Analizar los resultados generados por la herramienta y priorizar las vulnerabilidades según su criticidad.**  **· Generar un informe detallado con las vulnerabilidades encontradas, su impacto, pasos para reproducirlas y posibles soluciones.**  **· Validar las correcciones aplicadas realizando un nuevo escaneo y confirmando la resolución de los problemas.** | | |
| **MODULOS ASOCIADOS** | **· Seguridad del sistema**  **· Cliente**  **· Proveedor**  **· Administrador** | | |

| **NOMBRE** | **Resistencia a ataques DoS/DDoS** | **PRUEBAS** | **PSE 10** |
| --- | --- | --- | --- |
| **PROPÓSITO** | **Evaluar la capacidad del sistema para manejar una carga excesiva de solicitudes y verificar si existen medidas implementadas para prevenir la interrupción del servicio ante ataques DoS o DDoS.** | | |
| **PRERREQUISITOS** | **Acceso al entorno de prueba del sistema (nunca realizar estas pruebas en el entorno de producción).**  **Herramientas de simulación de tráfico como Apache JMeter, LOIC, Hping3 o Slowloris.**  **Conocimiento de las configuraciones de mitigación como firewalls, rate limiting y balanceadores de carga.** | | |
| **UBICACIÓN** | **Infraestructura de red y servidores que soportan el sistema.** | | |
| **ENTRADA** | **Datos de prueba:**    **Configuración de las herramientas para simular tráfico excesivo:**    **Número de solicitudes: 10,000 por minuto.**  **Duración de la prueba: 5 minutos.**  **Tipo de solicitud: GET y POST hacia el sistema.**  **Niveles de tráfico adicionales:**    **Incrementar gradualmente el tráfico desde 1,000 a 50,000 solicitudes por minuto para medir el umbral de tolerancia.** | | |
| **ORÁCULO** | **El sistema debe responder a las solicitudes legítimas durante la prueba sin interrupciones significativas.**  **Las solicitudes maliciosas deben ser bloqueadas automáticamente por las medidas de mitigación.**  **Los logs del sistema deben registrar intentos de saturación y las acciones tomadas para mitigarlos.** | | |
| **PASOS** | **· Configurar las herramientas de simulación para enviar un alto volumen de solicitudes al sistema.**  **· Observar el comportamiento del sistema bajo carga, prestando atención a:**  **· Respuesta del servidor (errores 429 o 503).**  **· Tiempos de respuesta para solicitudes legítimas.**  **· Verificar si se activan medidas de mitigación como:**  **· Bloqueo de IPs sospechosas.**  **· Limitación de solicitudes por usuario.**  **· Distribución del tráfico a través de balanceadores de carga.**  **· Revisar los logs para confirmar que los intentos de saturación y las acciones de mitigación fueron registrados.**  **· Generar un informe detallado sobre el impacto de la prueba, medidas exitosas y posibles mejoras.** | | |
| **MODULOS ASOCIADOS** | **· Infraestructura de red**  **· Seguridad del sistema**  **· Servidores web** | | |

| **NOMBRE** | **Pruebas de integridad de datos** | **PRUEBAS** | **PSE 11** |
| --- | --- | --- | --- |
| **PROPÓSITO** | **Verificar que los datos almacenados y transmitidos en el sistema no sean alterados de manera no autorizada, garantizando su consistencia y seguridad.** | | |
| **PRERREQUISITOS** | **Almacenamiento de datos en bases de datos y comunicación entre cliente y servidor.** | | |
| **UBICACIÓN** | **Infraestructura de red y servidores que soportan el sistema.** | | |
| **ENTRADA** | **Nombre: ClientePrueba**  **Documento: 123456789**  **Teléfono: 3214567898**  **Sexo: Masculino**  **Tipo Documento: CC**  **Correo: corre0@gmail.com**  **Contraseña: 2141524**  **Intento de alteración de datos mediante SQL Injection:**  **Entrada: ' OR '1'='1**  **Objetivo: Verificar si el sistema permite modificar o acceder a datos sin autorización.**  **Datos transmitidos entre cliente y servidor:**  **Realizar una solicitud POST con datos cifrados y verificar la consistencia.** | | |
| **ORÁCULO** | **Los datos en la base de datos deben permanecer sin alteraciones no autorizadas.**  **Las transmisiones de datos deben mantenerse íntegras, sin modificación por terceros.**  **Las solicitudes de acceso no válidas deben ser rechazadas y registradas en los logs del sistema.** | | |
| **PASOS** | **· Insertar datos válidos en la base de datos a través del sistema.**  **· Intentar modificar los datos utilizando métodos no autorizados, como:**  **· Consultas SQL maliciosas.**  **· Manipulación de paquetes de datos en tránsito.**  **· Verificar en la base de datos que los datos originales permanecen sin alteración.**  **· Monitorizar las transmisiones de datos utilizando herramientas como Wireshark para asegurarse de que no se producen modificaciones no autorizadas durante el proceso de envío/recepción.**  **· Realizar pruebas adicionales con roles de usuario limitados para verificar que no pueden alterar datos que no les corresponden.**  **· Generar un informe con los resultados, incluyendo cualquier inconsistencia detectada y las acciones necesarias para resolverlas.** | | |
| **MODULOS ASOCIADOS** | **· Cliente**  **· Proveedor**  **· Base de datos**  **· Seguridad del sistema** | | |

# ANEXO

## Anexo 1: Reportes de Pruebas

1. [↑](#footnote-ref-0)
2. [↑](#footnote-ref-1)