

Universidad de las Fuerzas Armadas ESPE

sede Latacunga



Departamento de Eléctrica y Electrónica

Ingeniería de Software

Plan de Pruebas del

sistema Prowessec

Tabla de Contenido

Historial de Versiones	4
Información del Proyecto	4
Aprobaciones	4
Resumen Ejecutivo	4
Alcance de las Pruebas	4
Elementos de Pruebas	4
Nuevas Funcionalidades a Probar	4
Pruebas de Regresión	4
Funcionalidades a No Probar	4
Enfoque de Pruebas (Estrategia)	4
Criterios de Aceptación o Rechazo	4
Criterios de Aceptación o Rechazo	4
Criterios de Suspensión	4
Criterios de Reanudación	4
Entregables	4
Recursos	4
Requerimientos de Entornos – Hardware	4
Requerimientos de Entornos – Software	4
Herramientas de Pruebas Requeridas	4
Personal	4
Entrenamiento	4
Planificación y Organización	4
Procedimientos para las Pruebas	4
Matriz de Responsabilidades	4
Cronograma	5
Premisas	5
Dependencias y Riesgos	5
Referencias	5
Glosario	5

Historial de Versiones

Fecha	Versión	Autor	Organización	Descripción
29/09/2021	v 1.0	❖ Carrillo José ❖ Romo Paola ❖ Sánchez Ana ❖ Semblantes Geraldyn	FUDELA - HIAS	Modelado de plantilla para Pruebas de Software

Información del Proyecto

Empresa / Organización	FUDELA - HIAS
Proyecto	ACNUR-HIAS-ESPE- (PARA POBLACIÓN EN MOVILIDAD HUMANA).
Fecha de preparación	10/09/2021
Cliente	FUDELA - HIAS
Patrocinador principal	Dr. Luis Enrique Simbaña Taipe
Gerente / Líder de proyecto	Karla Emilia Altamirano Cevallos
Gerente / Líder de pruebas de software	

Aprobaciones

Nombre y Apellido	Cargo	Departamento u organización	Fecha	Firma
Dr Luis Enrique Simbaña Taipe	Patrocinador Principal	ESPE		
Karla Emilia Altamirano Cevallos	Líder de proyecto	ESPE		

Resumen Ejecutivo

El presente plan de pruebas se ha desarrollado con la finalidad de responder a la necesidad de verificar de forma eficiente el correcto funcionamiento del sistema Prowessec, una aplicación web que tiene como objetivo principal ser un Marketplace.

El objetivo del Plan de Pruebas es evaluar el sistema Prowessec con los instrumentos adecuados para su mejoramiento o aprobación al mercado. Todo esto en base de proporcionar un producto software de calidad tanto para las organizaciones como para los usuarios.

A partir de este planteamiento se ha tratado de mejorar la eficiencia y eficacia de las funciones del sistema Prowessec a través de pruebas de las mismas como se presenta a continuación.

Alcance de las Pruebas

Elementos de pruebas.

A continuación se detallan los módulos sobre los cuales se va a realizar las pruebas y que integra a cada uno de ellos:

Inicio	<ul style="list-style-type: none">● Fotos● Novedades de curso● Novedades de productos
Acerca de Nosotros	<ul style="list-style-type: none">● Quienes somos● Misión● Visión● Objetivos● Valores
Servicios	<ul style="list-style-type: none">● Organizaciones Relacionadas<ul style="list-style-type: none">○ Ciudad○ Capacitaciones○ Incluir Fotos
Equipo de trabajo	<ul style="list-style-type: none">● Personas encargadas(descripción del staff)<ul style="list-style-type: none">○ Por Orden de Importancia
Documentación (Repositorio)	Documentos PDF divididos de la siguiente manera: HIAS <ul style="list-style-type: none">● marketing<ul style="list-style-type: none">○ Cursos

	<ul style="list-style-type: none"> ■ Material de planificación ■ Material de Apoyo <ul style="list-style-type: none"> ● Semanas <ul style="list-style-type: none"> ○ Diapositivas ○ Trabajo <p>FUDELA</p> <ul style="list-style-type: none"> ● Contabilidad <ul style="list-style-type: none"> ○ Cursos <ul style="list-style-type: none"> ■ Material de planificación ■ Material de Apoyo <ul style="list-style-type: none"> ● Semanas <ul style="list-style-type: none"> ○ Diapositivas ○ Trabajo ● marketing <ul style="list-style-type: none"> ○ Cursos <ul style="list-style-type: none"> ■ Material de planificación ■ Material de Apoyo <ul style="list-style-type: none"> ● Semanas <ul style="list-style-type: none"> ○ Diapositivas ○ Trabajo
Marketplace	<ul style="list-style-type: none"> ● Inicio <ul style="list-style-type: none"> ○ Nuevo Producto ○ Populares ● Servicio al cliente <ul style="list-style-type: none"> ○ Preguntas Frecuentes con respuesta ○ Contacto de Asesoramiento ● Categoría <ul style="list-style-type: none"> ○ Alimentación ○ Servicios ○ Varios ● Carrito de compra <ul style="list-style-type: none"> ○ Productos agregados ○ Subtotal ○ Botón “Pagar” <ul style="list-style-type: none"> ■ Pagar <ul style="list-style-type: none"> ● Direccion de envio ● Método de pago ● Perfil del Usuario <ul style="list-style-type: none"> ○ Formulario ○ Datos de envío ○ Historial ○ Método de pago

FOOTER en todas las secciones	<ul style="list-style-type: none"> ● Botón flotante para contactar al WhatsApp del director del proyecto.
--------------------------------------	--

Nuevas Funcionalidades a Probar

Las funcionalidades se irán haciendo en base a los requerimientos planteados, todo esto tiene un enfoque directo al usuario final, que en este caso serán las personas que exploren en la página web, estas serán las siguientes:

Descripción	Estado
Buscar Producto	Ya implementado y funcionando
Comprar Producto	Ya se puede agregar al carrito de compras
Buscar Curso	Ya implementado
Acceder al botón de pagos	Ya implementado falta configurar ya la realización de pagos
Realizar un pago	No funcional
Registrarse	Ya implementado

Pruebas de Regresión

A continuación se listaran las funcionales no directamente involucradas en el desarrollo pero cuyos componentes están siendo afectados y por ende deben probarse para asegurar que estén funcionando correctamente:

- Validación de correos electrónicos.
- Encriptación de contraseñas
- Validación de Fechas
- Validación de Números de Teléfono
- No mostrar información de Administradores a Usuarios
- Validar que el Usuario no tenga acceso a las acciones del Administrador.
- Validar funcionalidad en dispositivos móviles o computadores.
- Validación de campos vacíos en el registro de información. No se permitirá el almacenamiento de información si los campos están vacíos o la información está incompleta.

Funcionalidades a No Probar

A continuación vamos a establecer las funcionalidades a no probar, que son muy pocas ya que todo se debería comprobar mucho más al manejar un sistema de pagos debe la página garantizar seguridad para los clientes tengan la confianza de ingresar sus datos.

Descripción	Estado
Paleta de colores	

Enfoque de Pruebas (Estrategia)

Prueba de Iteración	Prueba del sistema	Prueba de integración	Prueba de unidad	Porcentaje
Iteración 1	Especificación	Concordancia con los requerimientos documentados		
Iteración 2	Usabilidad	GUI		
Iteración 3	Prueba Alfa	Demostración del equipo de Programación		
Iteración 4	Prueba Beta	Ambiente natural		

Criterios de Aceptación o Rechazo

Criterios de Aceptación o Rechazo

Se dará por aceptada o rechazada la funcionalidad en base al porcentaje de prueba unitaria que haya obtenido en la sección anterior. Dicho porcentaje debe ser de un 100% para dar la aceptación y finalización de la misma, caso contrario se deberá realizar un cambio/modificación de dicha funcionalidad.

Prueba de Iteración	Porcentaje	Estado

Criterios de Suspensión

Los casos de pruebas sobre un módulo se detendrán solamente si el módulo presenta un 80% de errores o fallos que impiden la correcta ejecución del sistema caso contrario se procederá con la prueba del módulo siguiente.

Criterios de Reanudación

En caso de que una prueba haya superado el 80% de errores se reanudará con el caso de prueba solamente si todos los errores presentados son resueltos correctamente y permiten la ejecución correcta del sistema.

Entregables

Los documentos que se entregarán al finalizar las pruebas en el sistema web son:

- Documento de plan de pruebas
- Casos de pruebas
- Especificación del diseño de casos
- Evidencias de las pruebas
- Reportes emitidos por herramientas utilizadas para realizar pruebas de rendimiento.

Recursos

Requerimientos de Entornos – Hardware

Lo que se necesita en la red para realizar las pruebas que son

- Base de datos: Programada en MySQL
- Servidor: Hostinger
- Plataforma: Wordpress

Requerimientos de Entornos – Software

- Acceso al sistema como usuario
- Acceso al sistema como administrador.

Herramientas de Pruebas Requeridas

Personal

Nombre	Rol
Jose Gabriel Carrillo Sandoval	Automatización de Pruebas
Paola Lissette Romo Gavilanez	Analista de Pruebas
Ana Lissette Sánchez Pico	Analista de Pruebas
Geraldyn Nicol Semblantes Lozada	Lider de Pruebas

Entrenamiento

Para la evaluación del sistema Prowess se hará uso tanto de los requisitos documentados por el equipo de documentación como el uso de un test para evaluar la usabilidad del mismo.

Planificación y Organización

Procedimientos para las Pruebas

A continuación se van a listar los pasos que se va a seguir para el desarrollo del plan de pruebas:

1. Analizar los requerimientos de desarrollo de software
Lo primero que se va a hacer es entender cuales son los requerimientos del usuario que componen el proyecto, se analizará la matriz de trazabilidad , requerimientos funcionales, no funcionales, casos de uso, historias de usuario.
2. Identificar nuevas funcionalidades
En caso de existir nuevas funcionalidades o requerimientos en el desarrollo del proyecto se las integrará acorde a las necesidades del cliente, además se adjuntará al documento de requerimientos, casos de uso, historias de usuario y documento de diseño.
3. Definir la estrategia de pruebas
Se van a desarrollar:
 - Pruebas Funcionales: Es decir se van a evaluar lo que el sistema hace, estas se definen a partir de funciones o características así como la interoperabilidad entre sistemas
 - Pruebas no Funcionales: Su objetivo es probar los requerimientos no funcionales.
 - Pruebas de estructura: Mediante las cuales se evaluará la cobertura del sistema, estas se basan en las pruebas de caja blanca y si no se obtiene una cobertura del 100% se diseñan pruebas adicionales.
 - Pruebas de Regresión: Estas se aplican a nuevas funcionalidades que ya han sido corregidas con la finalidad de comprobar que un defecto ya ha sido corregido
4. Elaboración de criterios de inicio, aceptación y suspensión de pruebas.
5. Se tomarán en cuenta las siguientes acciones preventivas para evitar retraso y retrabajo.
 - Acciones preventivas en la fase de análisis y diseño de sistemas
 - Elaborar el diseño de las pruebas durante la fase de análisis y diseño del sistema, para anticipar de esta forma los requerimientos de pruebas con antelación.
 - Determinar con antelación los requisitos de usuarios de base de datos y sistema operativo, incluyendo los privilegios de acceso.
 - Determinar con antelación los parámetros de configuración de componentes de software y sistemas.
 - Identificar los requerimientos de software, plugins y componentes.

- Mantener un control de las versiones sobre todo de los documentos como: documento de requerimientos, historias de usuario, casos de uso.
- Acciones preventivas en la fase de desarrollo
 - Definir con antelación los roles y responsabilidades de las personas encargadas de la programación de la página web.
 - Definir quién es el responsable de cada equipo de trabajo en este caso se definen 3 equipos documentación, base de datos y programación cada uno con su respectivo líder.
 - Realizar reuniones diarias para mayor facilidad en la implementación de nuevas funcionalidades.
 - Las cuentas de los usuarios a utilizar en el desarrollo deberán ser equivalentes a las que se usarán en el ambiente de desarrollo, pruebas y producción, con los mismos nombres, configuración y privilegios de acceso. Esto también aplicará para los usuarios del sistema operativo, base de datos y aplicación.
 - Se deben considerar lineamientos para el área de seguridad como el manejo de contraseñas de usuarios de sistema utilizados por los componentes de software.
 - Antes de hacer una entrega de algún módulo de programación para realizar las pruebas se debe constatar que toda la documentación esté acorde a lo que se está entregando.
 - Al realizar las pruebas se debe probar que existen los roles de usuarios, administradores, vendedores y shop manager.
 - Al instalar en ambiente de pruebas, debe verificarse si existen diferencias en las bases de datos, solicitando y revisando copias de la base de datos.
- Acciones preventivas antes de iniciar las pruebas
 - Cuando las pruebas sean end to end deben incluirse casos de prueba de todo componente intermedio, esto incluye configuraciones, plugins de instalación en base de datos y pruebas sobre componentes de plataforma.
 - Solicitar con antelación roles que actuarán dentro del sistema, nombre y privilegios que cada uno poseerá.
 - Solicitar con antelación las herramientas para la realización de las pruebas.
 - Identificar y solicitar con antelación la configuración de opciones de funcionalidad, en caso de existir funcionalidades habilitadas o deshabilitadas vía base de datos.
 - En caso de una instalación fallida en un ambiente de pruebas, reservar todos los cambios y dejar operativo el ambiente para evitar retrasos en otros equipos de proyectos.

Matriz de Responsabilidades

Se hará uso de una matriz RACI como se detalla a continuación

Actividades	Personas				
	José Carrillo	Paola Romo	Ana Sánchez	Geraldyn Semblantes	Karla Altamirano
Aprobar finalización de aplicación Web					R
Evaluar el funcionamiento de Prowessec	C	C	R	A	I
Verificar cumplimiento de requisitos	C	C	A	R	I
Estimar usabilidad	R	R	A	C	I

(R) Responsable (A) Aprueba (C) Consultado (I) Informado

Cronograma

Se estima las siguientes fechas y actividades:

Fecha	Actividad	Responsable
06-Oct-2021	Finalización de aplicación web Prowessec	Karla Altamirano
07-Oct-2021	Revisión de Prowessec	Equipo documentación
08-Oct-2021	Corrección y retroalimentación de Prowessec	Equipo Software

Premisas

En el ítem anterior de cronograma se indica las fechas de finalización de la página web, y las respectivas revisiones a realizarse

- Hay que analizar las herramientas que se utilizaran para realizar las respectivas pruebas
- El proyecto debe cumplir un total de 90 horas es decir en este tiempo debemos distribuirlo para la finalización de la página web y las respectivas pruebas a realizarse

- La metodología de pruebas se analizará más adelante, primero debemos buscar la herramienta más óptima para realizar pruebas.
- Entre las herramientas a ocupar puede ser:
 - Las herramientas de software nos permiten realizar pruebas automatizadas y en especial realizar pruebas funcionales y de regresión en la aplicación tenemos herramientas como:
 - Selenium
 - Test Studio
 - Testim
 - También debemos realizar pruebas de carga, para analizar qué tan rápido es acceder a la página web su rendimiento, y garantizar que tan se tendrá un acceso de ida y vuelta son:
 - Load Runner
 - Web Load
 - Blazemeter
 - JMeter
- Para la elección de una de las herramientas deberíamos tener una reunión con el equipo de programación para definir la herramienta que se va a utilizar y el tiempo que llevará realizar las pruebas.

Dependencias y Riesgos

Aquí vamos a analizar de qué depende nuestro proyecto y cómo esas dependencias pueden afectar a la funcionalidad del mismo:

- La página web depende de hostinger y un error puede hacer que se caiga la plataforma y deje de funcionar.
- El botón de pagos depende de datafast
- Si datafast se cae o no nos brindan acceso entonces los usuarios no podrán realizar los pagos en la página web y sería un impedimento para que surja el proyecto
- Problemas de los bancos, es algo que no controlamos pero al usar datafast este buscará nuestros datos bancarios y si el banco tiene problemas entonces no podrá proceder al pago.
- El servicio de transporte igual es algo que no controlamos entonces un problema con el servicio y haría que se retrasen entregas y habría molestia de parte de los usuarios.

Referencias

MySQL. (s. f.). Recuperado 01 de octubre de 2021, de <https://www.mysql.com/>

WordPress.com: Crea un sitio web o blog gratuito. (s. f.). Recuperado 01 de octubre de 2021, de <https://wordpress.com/es/>

Glosario

Aplicación Web: Son aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de internet mediante un navegador. En otras palabras, es un programa que se codifica en un lenguaje interpretable por los navegadores web en la que se confía la ejecución al navegador.

Marketplace: Es un tipo de sitio web de comercio electrónico en el que la información sobre productos o servicios es proporcionada por múltiples terceros. Marketplace es el principal tipo de comercio electrónico multicanal y puede ser una forma de racionalizar el proceso de producción.

Requerimientos: Es una descripción completa del comportamiento del sistema que se va a desarrollar. Incluye un conjunto de casos de uso que describe todas las interacciones que tendrán los usuarios con el software.

Encriptación: La encriptación es un método de codificación de datos (mensajes o archivos) de modo que solo las partes autorizadas puedan leer la información o acceder a ella. La encriptación utiliza algoritmos complejos para codificar la información que se envía.

Funcionalidad: Es la capacidad del software de cumplir y proveer las funciones para satisfacer las necesidades explícitas e implícitas cuando es utilizado en condiciones específicas.

Usabilidad: La usabilidad persigue que la experiencia del usuario sea lo más satisfactoria posible al interactuar con un producto o servicio. Una buena usabilidad requiere que el sitio sea fácil de navegar, tenga un diseño decente, sea coherente en todas las páginas y que también sea informativo y útil para el visitante.

GUI: Es una interfaz entre la persona y la máquina. El objetivo de esta es representar el código del backend de un sistema de la forma más clara posible para el usuario.

MySQL: Es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual.

Hostinger: Es un proveedor de alojamiento web y dominios web para el desarrollo de sistemas web.

Wordpress: Es un sistema de gestión de contenidos lanzado el 27 de mayo de 2003, enfocado a la creación de cualquier tipo de página web.

Casos de prueba: Es un conjunto de valores de entrada, pre condiciones de ejecución, resultados esperados y post condiciones de ejecución sobre los cuales se evaluarán requerimientos y sus funcionalidades.

Matriz de trazabilidad: Es una matriz que relaciona cada uno de los requerimientos con el entregable que se haya solicitado, ayudará a identificar el resultado que se alcanza a través de cada requisito y a la vez qué requisitos son los que se permiten obtener en un entregable.

Historias de usuario: Es una explicación general e informal de una función de software escrita desde la perspectiva del usuario final.