

Implementar pruebas de software en un proyecto asegura la calidad y el correcto funcionamiento de la aplicación. Existen diferentes tipos de pruebas, entre las más importantes podemos encontrar las pruebas de carga, estas miden el rendimiento de la aplicación bajo una gran demanda. Por otro lado tenemos las pruebas funcionales que están nos ayudan a validar que la aplicación cumpla los requisitos funcionales para que la aplicación funcione correctamente, y por último tenemos las pruebas de seguridad que están como su nombre lo dice aseguran la seguridad de la aplicación identificando las posibles vulnerabilidades para así proteger la información del sistema.

Estas pruebas las podemos realizar con herramientas como Selenium y JMeter que nos permiten automatizar estos procesos, haciéndonos más fácil la ejecución de pruebas funcionales y de rendimiento. Por otra parte, se pueden implementar recursos como una matriz de trazabilidad que nos puede ayudar a relacionar estos requisitos con el tipo de prueba que se debe implementar en cada uno de ellos y una CheckList que nos asegura que estas pruebas y actividades ya han sido realizadas y completadas correctamente.

Asignación de Roles:

Juan Pablo Leal Usuga: Presentador

Juan David Castaño Hoyos: Equipos de pruebas

Karen Yuliana Valencia Ochoa: Auditor de pruebas

Kelly Sandrith Diaz Bermudez: Documentador

ID	Tt Categoría	🔗 Prioridad	Tt Fuente/Source	Tt Objetivo	Tt Herramienta
F1	Registro de Usuario	Urgente	http://localhost:8000/wp-admin/user-new.php	Verificar que el flujo de registro de usuario funcione correctamente, para que los nuevos usuarios puedan crear su cuenta sin errores ni validaciones faltantes.	Selenium
F2	Busqueda de Productos	Moderado	http://localhost:8000/?post_type=product	Validar la búsqueda de productos por palabra clave, para que los resultados sean relevantes y muestren la información completa (imagen, precio, stock).	Selenium
F3	Verificar el Carrito	Urgente	http://localhost:8000/?page_id=10 (Carrito)	Probar el proceso de añadir y eliminar productos del carrito, para que el total de la compra se actualice correctamente y no queden residuos de artículos.	Selenium
F4	Pasarela de Pagos	Urgente	http://localhost:8000/?page_id=11 (Finalizar compra)	Comprobar que el flujo de pago con tarjeta (WooCommerce) se complete sin fallos, para que el pedido se genere correctamente y se notifique al usuario.	Selenium
F5	Notificacion de Correo	Critico	-http://localhost:8000/?page_id=12 (Inicio de sesion) -http://localhost:8000/?page_id=11 (Finalizar compra)	Validar el envío de correos transaccionales (confirmación de pedido, restablecer contraseña), para que los usuarios reciban siempre la notificación adecuada.	Selenium

ID	Tt Categoría	🔗 Prioridad	Tt Fuente/Source	Tt Objetivo	Tt Herramienta
C1	Tiempo de respuesta	Critico	http://localhost:8000/	Simular 500 usuarios concurrentes navegando por la página de inicio, para que el tiempo de respuesta se mantenga por debajo de 2 s bajo alta demanda.	JMeter
C2	Procesamiento Check Out	Urgente	http://localhost:8000/?page_id=11 (Finalizar compra)	Medir el tiempo de respuesta al procesar un checkout con 100 usuarios simultáneos, para que el sistema escale adecuadamente sin errores de timeout.	JMeter
C3	Soporte Cargas Masivas	Moderado	http://localhost:8000/wp-admin/post-new.php?post_type=product	Ejecutar un test de estrés subiendo archivos grandes (imágenes de producto) en paralelo, para que la aplicación soporte cargas masivas sin caídas.	JMeter
C4	Cuello Botella	Moderado	http://localhost:8000/?page_id=11 (Finalizar compra)	Analizar el rendimiento de la base de datos bajo 200 consultas/segundo, para que no haya cuellos de botella en la capa de datos.	JMeter
C5	Degradacion de Servicios	Urgente	http://localhost:8000/	Realizar un test de resistencia continuo durante 2 horas con 100 usuarios, para detectar fugas de memoria o degradación progresiva del servicio.	JMeter

ID	T↑ Categoría	🔍 Prioridad	T↑ Fuente/Source	T↑ Objetivo	T↑ Herramienta
S1	Vulnerabilidades SQL	Critico	http://localhost:8000/?page_id=12 (Inicio de sesion)	Ejecutar escaneos de vulnerabilidades (SQL Injection) en todos los formularios de entrada, para que la aplicación esté protegida contra inyecciones maliciosas.	Selenium
S2	Inicio de Sesion	Urgente	http://localhost:8000/?page_id=12 (Inicio de sesion)	Probar la fuerza bruta de inicio de sesión con un diccionario de contraseñas, para que validar que los mecanismos de bloqueo de cuenta y captcha funcionan.	Hydra
S3	Gestion de Permisos	Moderado	http://localhost:8000/wp-login.php?loggedout=true&wp_lang=es_CO (Inicio de sesion Admin)	Auditar la gestión de permisos (roles de usuario) intentando accesos no autorizados, para que sólo los roles adecuados puedan ver o modificar información sensible.	Selenium
S4	Seguridad HTTP	Critico	http://localhost:8000/	Revisar los encabezados HTTP de seguridad (CSP, HSTS, X-Frame-Options), para que esté mitigada la mayoría de ataques de inyección y clickjacking.	Manualmente
S5	Proteccion CSRF	Urgente	http://localhost:8000/?page_id=12 (Inicio de sesion)	Probar la protección contra CSRF enviando formularios y peticiones con y sin el token CSRF válido, para que se evite que atacantes realicen acciones no autorizadas en nombre de un usuario autenticado.	Selenium