

TEST STRATEGY

THIS DOCUMENT CONTAINS QA INFORMATION FOR MANUAL TESTING.

-

1 INTRODUCCIÓN

El equipo #4 se encarga de investigar, desarrollar y testear www.losvasquitos.com.ar

La UI ya existe. El equipo estará a cargo de testear sus funcionalidades y características.

2 OBJETIVO

El propósito de este documento es establecer los procedimientos, herramientas y recursos de QA para el proyecto www.losvasquitos.com.ar

3 RESUMEN DEL SISTEMA

El producto a testear es www.losvasquitos.com.ar. El producto se desarrollaría y probaría según los requisitos y los criterios de aceptación definidos en las historias de usuario.

4 ALCANCE DE LA PRUEBA

- 4.1 ALCANCE (IN SCOPE)

El software a probar será www.losvasquitos.com.ar. Se requiere el siguiente alcance para que los evaluadores lo logren:

A. Pruebas unitarias

B. Pruebas manuales funcionales:

- Pruebas de sistema, integración y regresión (cuando sea necesario)

C. Pruebas reglamentarias:

- CASL

- AODA: (WCAG) 2.0 Nivel AA

D. Compatibilidad del navegador:

- Último Firefox (navegador principal utilizado en el sitio)
- Chrome más reciente
- Microsoft Edge más reciente (trimestral. Prioridad más baja y no un éxito para los lanzamientos)

E. Compatibilidad del sistema operativo:

- Windows 10/11
- Android (Versión actual y anterior (12 y 13))
- iOS (versión actual y anterior)
- MacOS (versión actual y anterior)

F. Compatibilidad de dispositivos:

- Móvil (414x896 iPhone XR, 390x844 iPhone 12)
- Ordenador portátil (1920x1080)

4.2 FUERA DE ALCANCE

Los tipos de pruebas que se considerarán fuera del alcance son:

- Pruebas no funcionales:
- Rendimiento (carga, estrés)
- Pruebas de seguridad
- Pruebas de aceptación del usuario (realizadas por el cliente)

4.3 DEFINICIÓN DE HECHO

Conjunto de criterios definidos por el equipo Scrum que determinan si una historia de usuario se puede considerar realizada.

○ 4.3.1 DEFINICIÓN DE HISTORIA DE USUARIO DE LISTO

- El código se escribe, confirma y envía.
- Las solicitudes de extracción están completas, incluida la revisión de código, y se fusionaron con la rama de desarrollo.
- El proyecto se construye sin errores.
- Pruebas unitarias aprobadas (responsabilidad del desarrollador).
- Se aprobaron las pruebas en los dispositivos/navegadores enumerados en los supuestos del proyecto.
- Control de calidad realizado y problemas resueltos. (Los problemas de baja prioridad o gravedad menor se informarán y no afectarán la resolución de la historia)
- La función se prueba con criterios de aceptación y criterios técnicos.
- Cualquier configuración o cambios de construcción documentados.

- Documentación actualizada.

5 ENFOQUE DE PRUEBA

5.1 PRINCIPIOS Y APLICACIÓN

○

○ 5.1.1 PRUEBAS BASADAS EN RIESGO Y ENFOCADAS

5.1.1.1 Principio

Las pruebas deben estar enfocadas a los riesgos de www.losvasquitos.com.ar y la importancia del negocio. Lo más importante para www.losvasquitos.com.ar es el correcto funcionamiento de la compra y venta de productos para el hogar. Esto es importante porque estas dos características están directamente relacionadas con sus clientes.

5.1.1.2 Solicitud

www.losvasquitos.com.ar debe satisfacer ambas necesidades primarias. Las pruebas deben estar enfocadas primariamente en aquellas necesidades por ser de alta prioridad, pero también deben realizarse pruebas funcionales de acuerdo a la prioridad y posibilidad del caso de uso.

5.1.2 PRUEBA TEMPRANO

5.1.2.1 Principio

Las pruebas deben comenzar lo antes posible en el ciclo de vida del software (nivel de unidad y componente, así como revisiones estáticas).

5.1.2.2 Solicitud

El equipo de control de calidad participará en las reuniones de perfeccionamiento. Además, deben analizar los criterios de aceptación y los requisitos en las historias de usuario.

○

5.2 PRUEBA MANUAL

5.2.1 PREPARACIÓN DE LA PRUEBA

5.2.1.1 Criterio para entrar

Las pruebas se harán por boleto. El boleto debe tener todos los requisitos necesarios, criterios de aceptación y definición de hecho. Tan pronto como el ticket se refina en una reunión y el equipo de control de calidad lo analiza, el evaluador puede comenzar a preparar los casos de prueba.

5.2.1.2 Criterio de salida

Las pruebas deben cubrir todos los requisitos y criterios de aceptación y también deben tener pruebas funcionales para tener la mayor cobertura sobre los problemas. Todos los campos de entrada deben cubrirse al menos para los caminos felices.

5.2.2 EJECUCIÓN DE LA PRUEBA

- **5.2.2.1 Criterios de entrada**

Se deben realizar todos los casos de prueba para que se pruebe el ticket. Se debe desarrollar la característica a probar. El entorno de control de calidad debe ser estable y estar listo para usar. El equipo de control de calidad debe tener acceso a las principales credenciales para la autorización.

5.2.1.3 Criterio de salida

Se deben ejecutar al menos todas las pruebas de prioridad alta y media. Los casos de prueba deben informarse como aprobados, fallidos o bloqueados. Los errores deben informarse en los tickets como la guía de gestión de defectos e incidentes.

6 CRITERIOS DE SUSPENSIÓN

- Un error que no permite las pruebas de ruta feliz.
- El entorno de control de calidad no es lo suficientemente estable para continuar con la ejecución de la prueba.
- No se puede instalar una nueva versión o software.
- Los requisitos no son lo suficientemente claros para hacer los casos de prueba y ejecutarlos.
- Falta de credenciales y datos para crear y ejecutar casos de prueba.

- 7 REQUISITOS DE ENTORNOS DE PRUEBA

Las pruebas se realizarán en el entorno de control de calidad. Los desarrolladores deben configurar el entorno de control de calidad y los evaluadores deben tener acceso a las credenciales para acceder a ese entorno. El líder técnico será responsable de mantener el entorno de control de calidad. Las credenciales de control de calidad serán proporcionadas a los probadores por los propietarios del producto www.losvasquitos.com.ar

- 8 REQUISITOS DE DATOS DE PRUEBA

Los primeros datos de prueba requeridos son las credenciales del entorno. Los probadores deben tener acceso a los datos necesarios para las pruebas. Los datos a utilizar serán de la base de datos del entorno QA.

- 9 HERRAMIENTAS Y TÉCNICAS DE PRUEBA

9.1 Pruebas manuales

Herramientas para la práctica ágil y el seguimiento de errores: Jira.

Herramientas para la ejecución de pruebas: Selenium.

- 10 ROLES Y RESPONSABILIDADES DE PRUEBA

10.1 Probador manual: Matías Fasulino y Alejandro Zamora

Responsabilidad:

Pruebas estáticas

- Participa en la refinación de boletos.
- Analiza los requisitos, criterios de aceptación y definición de hecho de los boletos.

Pruebas funcionales

- A cargo de la gestión de la suite de casos de prueba.
- Encargado de realizar los casos de prueba para cada ticket a probar.
- A cargo de la ejecución de casos de prueba.
- Busque y cree datos para los casos de prueba necesarios.
- Informa y toma evidencia de errores.
- A cargo del seguimiento y la gestión de errores.

- 11 DINÁMICA DE PRUEBA

Las dinámicas de prueba son las acciones realizadas por los probadores durante los sprints. Las dinámicas se escriben aquí para reconocer el trabajo que están haciendo los probadores en este proyecto.

11.1 PRUEBAS ESTÁTICAS

CUÁNDO: Sprint anterior o Sprint inicial

COMPORTAMIENTO:

- Participar en el perfeccionamiento de las historias.
- Analizar las historias con los criterios de aceptación y la definición de hecho.

11.2 PRUEBAS FUNCIONALES MANUALES

CUÁNDO: Comenzando el sprint o tan pronto como el probador haya analizado la historia.

ACCIÓN:

- Redacción de casos de pruebas en excel template para cada relato necesario.
- Busque o cree los datos de prueba.

CUÁNDO: después de que la tarea se implemente y se fusione con el entorno de control de calidad.

ACCIÓN:

- Ejecutar casos de prueba escritos para la función fusionada.
- Toma evidencia para cada error.

CUÁNDO: Tan pronto como se encuentre un error

ACCIÓN:

- Hable con el desarrollador.
- Escribir ticket de error en Jira.

CUÁNDO: Tan pronto como se solucione un error.

ACCIÓN:

- Vuelva a probar el error encontrado.

CUÁNDO: Al final de cada sprint.

ACCIÓN:

- Pruebas de regresión (cuando sea necesario, debe hacerse con automatización).

- 12 GESTIÓN DE PRUEBAS

Los casos de prueba se administrarán con Selenium, creando un nuevo proyecto y construyendo una prueba basada en la funcionalidad.

- 13 GESTIÓN DE DEFECTOS E INCIDENTES

○ 13.1 PROCESO DE GESTIÓN DE DEFECTOS

13.1.2 GESTIÓN DE DEFECTOS

- El evaluador realizará la prueba de la historia en el estado "Revisión de control de calidad".
- Si se encuentra un defecto (**que no cumple con los criterios de aceptación de la historia**) mientras se prueba la historia durante el sprint, NO se creará un error/defecto por separado, sino que la historia se enviará de vuelta al desarrollador para que la corrija. . El evaluador establece el subestado "Comentarios" y mueve el ticket a la columna "En curso".
- Si se encuentra un defecto (**fuera de los criterios de aceptación de la historia**) mientras se prueba la historia durante el sprint, se creará un error/defecto separado en la acumulación. El evaluador establece el subestado en "Completo" y mueve el ticket a la columna "Listo para implementar".

- Si la prueba de la historia se completa sin que se encuentren problemas, el probador actualizará el subestado a "Completado" y moverá el ticket a la columna "Listo para implementar".

○ **13.1.4 GRAVEDAD**

El probador establecerá la gravedad de cualquier defecto de acuerdo con los siguientes criterios:

- Bloqueador: un error que es tan grave que bloquea cualquier actividad de prueba y debe corregirse lo antes posible con la máxima prioridad.
- Crítico: cualquier bloqueo, pérdida de datos o mal funcionamiento de una parte vital del software que debe corregirse para la próxima compilación.
- Mayor: pérdida de funcionalidad importante. Debe corregirse hasta la versión final (versión del cliente).
- Normal: problema de bajo impacto. Arreglo para ser discutido / planeado.
- Menor: problema de menor impacto/sin impacto. Arreglo para ser discutido / planeado.

Cada defecto también tendrá asignada una prioridad de acuerdo con su prioridad para el negocio.

13.1.5 PRIORIDAD

El probador establecerá la prioridad de cualquier defecto de acuerdo con los siguientes criterios:

- S1: crítico: falla que obstruye todas las pruebas en al menos una plataforma principal (navegador de escritorio o dispositivo móvil)
- S2: alto: falla en la función central que afecta a una plataforma principal pero no obstruye todas las pruebas O obstruye todas las pruebas solo en plataformas secundarias
- S3 – Medio: falla en la función principal que solo afecta a las plataformas secundarias
- S4 – Bajo: El error se puede solucionar una vez que se ha solucionado el Defecto más grave.