



INGENIERÍA DE REQUISITOS

PRUEBAS DE ACEPTACIÓN DE REQUISITOS

PRUEBAS DE ACEPTACIÓN DE REQUISITOS

Las pruebas de aceptación de requisitos, son un proceso fundamental que permite verificar si un sistema cumple con las necesidades y expectativas definidas por el cliente. A través de estas pruebas, se valida que cada requisito especificado funcione correctamente en condiciones reales o simuladas, asegurando que el producto final sea aceptado antes de su entrega. Este mecanismo facilita la detección temprana de fallos o desviaciones, promoviendo la calidad y satisfacción del cliente.

Concepto y propósito de las pruebas de aceptación

Las pruebas de aceptación de requisitos, son una etapa fundamental dentro del proceso de aseguramiento de la calidad en el desarrollo de software. Su finalidad es garantizar que el producto cumple con las necesidades y expectativas del cliente o usuario final, validando que los requisitos previamente definidos han sido implementados correctamente.

1. Definición de las pruebas de aceptación

Las pruebas de aceptación se entienden como el conjunto de actividades dirigidas a evaluar si un sistema o producto satisface las condiciones y criterios establecidos en los requisitos funcionales y no funcionales acordados con el cliente. Se llevan a cabo generalmente al final del ciclo de desarrollo, antes de la entrega formal.

En un sistema de gestión de pedidos, la prueba de aceptación verificará si el cliente puede realizar una orden, modificarla y obtener una confirmación, asegurando que el flujo cumple con lo solicitado.

2. Propósito principal

El propósito principal de estas pruebas es validar la conformidad del producto con las especificaciones y requerimientos reales de negocio. Sirven para detectar fallos, omisiones o desviaciones que podrían afectar la experiencia del usuario o el funcionamiento del sistema en un entorno productivo.

Estas pruebas permiten, además, reducir riesgos asociados a entregas defectuosas, facilitar ajustes oportunos y garantizar la satisfacción del cliente.

3. Momentos y actores involucrados

Las pruebas de aceptación suelen realizarse una vez finalizadas las pruebas internas de desarrollo y pruebas de sistema. Los actores principales incluyen usuarios finales, clientes y a veces equipos de aseguramiento de calidad que facilitan el proceso.

- ✓ **Ejemplo.** En el caso de una aplicación móvil para reservas de vuelos, los usuarios finales podrían realizar pruebas donde reservan un vuelo, modifican datos o cancela una reserva, confirmando que todas las funciones cumplen con sus expectativas.

4. Tipos de pruebas de aceptación

- ✓ **Pruebas de aceptación de usuario (UAT).** Realizadas por los usuarios finales, para asegurar que el sistema funciona en un entorno similar al real.

- ✓ **Pruebas de aceptación operativas.** Valoran aspectos como la integración con sistemas existentes y la capacidad operativa.
- ✓ **Pruebas contractuales.** Verifican que se cumplen los términos acordados en contratos o acuerdos, de nivel de servicio.

5. Beneficios de realizar pruebas de aceptación

Entre los beneficios se encuentran la identificación temprana de discrepancias, mejora en la comunicación con el cliente, y aumento en la confianza sobre la calidad del producto. Facilitan la toma de decisiones sobre la liberación o ajustes finales antes del despliegue.

6. Ejemplo práctico

En un proyecto para crear un sistema de control de inventarios, la prueba de aceptación podría involucrar a los gerentes de inventario para que validen que se pueden ingresar productos, registrar ventas, realizar auditorías de stock y generar reportes. Si alguna función no se comporta como esperado, el equipo de desarrollo recibirá retroalimentación precisa para corregir el sistema.

Tipos de pruebas de aceptación

Las pruebas de aceptación son esenciales para validar que el software cumple con los requisitos y expectativas del cliente. Sin embargo, no existe un único tipo de prueba de aceptación; estas se clasifican según el enfoque, el objetivo y los actores involucrados. Cada tipo cumple un papel específico en asegurar la calidad y la funcionalidad del producto final.

Pruebas de aceptación

- ✓ **Cuándo.** Después de la implantación en el entorno de producción.
- ✓ **Objetivo.** Verifican que el sistema cumple con todos los requisitos indicados y permite que los usuarios del sistema den el visto bueno definitivo.
- ✓ **Herramientas.** Las mismas. Las pruebas se vuelven a ejecutar en el entorno real de producción y se añaden nuevas pruebas.

1. Pruebas de Aceptación del Usuario (UAT)

Este tipo de prueba se realiza al final del ciclo de desarrollo y está dirigido a usuarios finales o clientes reales. Su objetivo es verificar que el software cumple con los requisitos funcionales y no funcionales desde la perspectiva del usuario. Se prueba el sistema en condiciones reales de uso, evaluando la facilidad de uso, funcionalidad y rendimiento. Los usuarios proporcionan retroalimentación directa sobre la experiencia.

- ✓ **Ejemplo.** En una aplicación de gestión de tareas, los usuarios finales prueban si pueden crear, editar y eliminar tareas, y si el sistema envía notificaciones correctamente. Si un usuario reporta que la notificación no aparece, el equipo de desarrollo corrige el fallo antes de la entrega definitiva.

2. Pruebas de aceptación operacional

Este tipo de prueba se centra en asegurar que el software funcione correctamente en el entorno operativo real, incluyendo la infraestructura técnica y procesos asociados. Se evalúan aspectos como la instalación, configuración, copias de seguridad, recuperación ante fallos y rendimiento bajo carga real. Generalmente es ejecutado por personal de operaciones o de infraestructura.

- ✓ **Ejemplo.** Para un sistema bancario, se verifica que después de un corte de electricidad el sistema pueda restaurarse correctamente sin pérdida de datos, y que las tareas automatizadas de respaldo se ejecuten según lo programado.

3. Pruebas de aceptación contractual

Estas pruebas garantizan que el software cumple con los términos y condiciones pactados en el contrato entre proveedor y cliente. Esto puede incluir funcionalidades específicas, estándares de calidad, rendimiento y fechas de entrega. En caso de incumplimiento, se pueden activar cláusulas legales.

- ✓ **Ejemplo.** Una empresa contrató un sistema CRM que debe procesar al menos 5000 registros diarios. Durante la prueba contractual, se simulan cargas de datos para verificar que el sistema cumple ese requisito, validando además la respuesta dentro del tiempo acordado.

4. Pruebas de aceptación regulatoria

Dirigidas a verificar que el software cumple con normativas y estándares legales específicos del sector, como seguridad de datos, privacidad, accesibilidad o requisitos de auditoría. Su cumplimiento es fundamental para la legalidad y operatividad del sistema.

- ✓ **Ejemplo.** Un sistema de gestión hospitalaria debe cumplir con la Ley Portabilidad y Responsabilidad del Seguro Médico (HIPAA), que regula la confidencialidad y seguridad de la información médica. Se realizan pruebas para comprobar que solo usuarios autorizados acceden a datos sensibles y que los registros de acceso quedan almacenados para auditorías.

5. Pruebas de Aceptación alpha

Estas pruebas se realizan dentro de la organización de desarrollo, generalmente por testers y algunos usuarios internos seleccionados. Sirven para detectar fallos importantes antes de que el software salga a un público más amplio. Se ejecuta en un ambiente controlado.

- ✓ **Ejemplo.** Un equipo de desarrollo lanza una versión alpha de su nuevo software financiero, donde empleados internos realizan pruebas intensivas sobre las funcionalidades principales para detectar errores críticos que deben corregirse antes de avanzar.

6. Pruebas de aceptación beta

Se realizan con un grupo externo de usuarios en un entorno de producción o similar al real. El objetivo es obtener retroalimentación sobre la usabilidad, rendimiento

y detectar fallos que no se descubrieron en etapas previas. También se evalúa la satisfacción del usuario y se recopilan sugerencias para mejoras.

- ✓ **Ejemplo.** Una empresa lanza una versión beta de su aplicación móvil para un grupo selecto de usuarios que la usan en su vida diaria y reportan problemas de usabilidad o fallos de rendimiento, información valiosa para la versión final.

Tabla 1. Tipos de prueba

Tipo de prueba	Objetivo principal	Actores involucrados	Ejemplo de uso	Momento en el ciclo de vida
Pruebas de Aceptación del Usuario (UAT).	Validar que el sistema satisface las necesidades reales de los usuarios.	Usuarios finales, clientes.	Usuarios verifican funcionalidades en una app de gestión de tareas.	Final del desarrollo, antes de lanzamiento.
Pruebas de aceptación operacional.	Comprobar que el sistema funciona correctamente en el entorno operativo real.	Equipo de operaciones IT.	Verificar recuperación ante fallos en un sistema bancario.	Antes del despliegue en producción.
Pruebas de aceptación contractual.	Confirmar que se cumplen los términos y condiciones del contrato.	Cliente, proveedor.	Validar que CRM procesa 5000 registros diarios.	Después de desarrollo y antes de entrega formal.
Pruebas de Aceptación Regulatoria.	Garantizar cumplimiento de normativas y leyes.	Audidores, expertos legales.	Comprobar cumplimiento de HIPAA en software hospitalario.	Durante pruebas finales o pre-lanzamiento.
Pruebas de aceptación alpha.	Detectar fallos críticos en un entorno controlado.	Equipo de desarrollo, testers internos.	Empleados prueban versión inicial de software financiero.	Etapa temprana de prueba.
Pruebas de aceptación beta.	Obtener retroalimentación de usuarios reales.	Usuarios externos seleccionados.	Usuarios finales prueban app móvil en uso cotidiano.	Después de pruebas alpha y antes de lanzamiento general.

Criterios de aceptación

Los criterios de aceptación son condiciones específicas y claras que un producto o requisito debe cumplir para ser considerado satisfactorio y listo para su entrega o implementación. Representan un conjunto de reglas o estándares que guían la validación y la prueba de cada requisito, asegurando que el resultado final cumpla con las expectativas tanto del cliente como del equipo de desarrollo.

Función y propósito

El propósito principal de los criterios de aceptación, es definir qué significa que un requisito es completo y funcional. Estos criterios ayudan a evitar ambigüedades

durante el desarrollo, orientan las pruebas de aceptación y sirven como referencia común entre los stakeholders (clientes, usuarios, desarrolladores y testers). Al establecer criterios claros, se minimizan malentendidos y se facilita la comunicación efectiva sobre qué funcionalidades deben entregarse.

Características de buenos criterios de aceptación

- ✓ **Claros y específicos.** Deben describir de manera precisa qué se espera que haga el sistema.
- ✓ **Medibles y verificables.** Deben poder ser evaluados mediante pruebas objetivas.
- ✓ **Alcanzables.** Realistas dentro del contexto del proyecto.
- ✓ **Relevantes.** Relacionados directamente con el requisito que describen.
- ✓ **Comprensibles para todos.** Deben ser entendidos tanto por técnicos como por no técnicos.

Ejemplo práctico. Supongamos que un requisito en un sistema de comercio electrónico es: "El usuario debe poder agregar productos al carrito de compras".

Los criterios de aceptación para este requisito, podrían ser:

- ✓ El usuario puede seleccionar un producto y hacer clic en "Agregar al carrito".
- ✓ El producto seleccionado aparece en el carrito ,con la cantidad correcta.
- ✓ El carrito actualiza el total correctamente, al agregar un producto.
- ✓ El usuario recibe una notificación o confirmación visual, de que el producto fue agregado.
- ✓ Si el producto está agotado, el sistema muestra un mensaje adecuado y no lo agrega al carrito.

Cada uno de estos puntos es verificable durante las pruebas y juntos definen cuándo el requisito se considera cumplido.

Ejemplo en contexto de desarrollo ágil

En metodologías ágiles, como Scrum, los criterios de aceptación se adjuntan comúnmente a las historias de usuario para facilitar su desarrollo y validación. Por ejemplo, una historia de usuario podría ser:

"Como cliente, quiero poder filtrar productos por precio para encontrar opciones dentro de mi presupuesto".

Los criterios de aceptación aquí pueden incluir:

- ✓ El filtro de precio permite establecer un rango mínimo y máximo.
- ✓ Al aplicar el filtro, la lista muestra sólo productos dentro del rango seleccionado.
- ✓ El sistema muestra un mensaje si no hay productos disponibles en ese rango.

Estos criterios guían al equipo en la implementación y prueba, asegurando que la funcionalidad cumpla las expectativas.

Tabla 2. Historia de usuario

Requisito / Historia de usuario	Criterio de Aceptación	Estado
Como usuario, quiero agregar productos al carrito de compras.	1. El usuario puede seleccionar un producto y hacer clic en "Agregar al carrito".	Cumplido.
	2. El producto seleccionado aparece en el carrito con la cantidad correcta.	Cumplido.
	3. El carrito actualiza el total correctamente al agregar un producto.	Pendiente.
	4. El usuario recibe una confirmación visual de que el producto fue agregado.	Cumplido.
	5. Si el producto está agotado, se muestra un mensaje y no se agrega al carrito.	Cumplido.

Diseño y planificación de pruebas de aceptación

El diseño y la planificación de pruebas de aceptación son procesos fundamentales para validar que el software cumple con los requisitos acordados con el cliente. Esta etapa se centra en preparar casos de prueba y organizar su ejecución para asegurar que el producto final sea funcional y satisfaga las expectativas del usuario final.

Identificación y análisis de requisitos para pruebas

Para diseñar pruebas de aceptación efectivas, primero es necesario analizar detalladamente, los requisitos funcionales y no funcionales del sistema. Esto implica identificar las funcionalidades claves que deben ser verificadas y desglosarlas en criterios de aceptación claros y medibles.

Por ejemplo, en un sistema bancario, un requisito podría ser que el usuario pueda transferir dinero entre cuentas. El análisis de este requisito determinará qué aspectos serán validados, como la autenticación del usuario, el monto transferido y la confirmación de la operación.

Diseño de casos de prueba de aceptación

En esta fase, se crean los casos de prueba basados en los criterios de aceptación previamente definidos. Cada caso describe un escenario específico que simula el comportamiento esperado del sistema en condiciones reales de uso.

Los casos de prueba deben ser claros, detallados y contener:

- ✓ Descripción del objetivo. Qué se prueba y por qué.

- ✓ Precondiciones. Estado del sistema antes de la prueba.
- ✓ Pasos a seguir. Secuencia de acciones a realizar.
- ✓ Resultados esperados. Comportamiento que debe observarse para aprobar la prueba.

Ejemplo. Para la función de compra en línea, un caso de prueba podría ser: “Verificar que un usuario registrado puede agregar un producto al carrito, proceder al pago y recibir confirmación del pedido”.

Planificación de recursos y cronograma

Planificar las pruebas implica asignar recursos humanos, técnicos y temporales. Esto incluye decidir quién ejecutará las pruebas, dónde se llevarán a cabo (entornos de prueba), y cuánto tiempo se dedicará a cada conjunto de pruebas.

Se recomienda definir un calendario con fechas específicas, para:

- ✓ Preparación del entorno de pruebas.
- ✓ Ejecución de cada caso de prueba.
- ✓ Registro y análisis de resultados.
- ✓ Revisión y retroalimentación con el cliente.

Este cronograma ayuda a mantener el proceso ordenado y asegura que se cumplan los plazos establecidos en el proyecto.

Asignación de roles y responsabilidades

Para que el proceso sea efectivo, es importante definir claramente los roles involucrados en las pruebas de aceptación:

- ✓ **Tester o evaluador.** Persona encargada de ejecutar los casos de prueba.
- ✓ **Cliente o usuario final.** Participa para validar que el software cumple con sus expectativas.
- ✓ **Analista de calidad.** Supervisa y documenta los resultados.
- ✓ **Equipo de desarrollo.** Atiende los problemas detectados para corregir errores.

Esta definición evita confusiones y facilita una comunicación fluida durante la ejecución de las pruebas.

Documentación y seguimiento de pruebas

Durante la planificación y ejecución, se debe mantener una documentación detallada que registre:

- ✓ Casos de prueba diseñados.
- ✓ Resultados de cada prueba (éxito o falla).

- ✓ Observaciones y defectos encontrados.
- ✓ Acciones correctivas tomadas.

Esta documentación sirve como evidencia del cumplimiento de los requisitos y es clave para la mejora continua del producto.

- ✓ **Ejemplo.** Un reporte que incluya:
- ✓ **Caso de prueba.** "Registro de usuario".
- ✓ **Resultado.** Exitoso.
- ✓ **Observación.** Ninguna incidencia.

Ejecución y resultados de las pruebas

La ejecución de las pruebas de aceptación representa una etapa crucial en el proceso de validación de requisitos, pues es el momento en que se verifica de manera práctica si el sistema cumple con las expectativas y criterios previamente definidos. Durante esta fase, el equipo encargado realiza las pruebas diseñadas, siguiendo el plan establecido, para detectar cualquier desviación o defecto que pudiera afectar la calidad y funcionalidad del producto.

Preparación para la ejecución de pruebas

Antes de iniciar la ejecución, es fundamental asegurar que todo está listo para que las pruebas se realicen de forma eficiente y fiable. Esto incluye:

- ✓ **Verificación de los casos de prueba.** Asegurar que cada caso esté correctamente definido, con pasos claros y resultados esperados específicos.
- ✓ **Disponibilidad del entorno de pruebas.** Confirmar que el ambiente donde se realizará la prueba está configurado igual que el entorno real donde el sistema será usado.
- ✓ **Recursos asignados.** Determinar quiénes participarán en la ejecución, ya sean testers, usuarios finales o clientes, y asegurar que cuentan con las herramientas necesarias.

Ejemplo. En un sistema de ventas, se verifica que el caso de prueba "Generar factura con datos correctos" contenga instrucciones precisas, como ingresar cliente válido, seleccionar productos y finalizar la venta.

Ejecución de casos de prueba

Esta fase consiste en aplicar los casos de prueba uno a uno siguiendo el plan y registrar los resultados:

- ✓ **Ejecución secuencial.** Cada caso de prueba se realiza paso a paso para verificar el comportamiento esperado.
- ✓ **Registro de resultados.** Se documenta si el sistema pasó o falló la prueba, junto con detalles importantes que ayuden a identificar posibles problemas.

- ✓ **Participación de usuarios.** En pruebas de aceptación, el cliente o usuario final suele estar involucrado para validar desde su perspectiva.

Ejemplo. Durante la prueba de “Agregar nuevo usuario”, el tester sigue los pasos para registrar un usuario y anota que la operación se completó exitosamente, marcando el caso como aprobado.

Manejo y clasificación de incidencias

Cuando el sistema no cumple con lo esperado, se registran las fallas o incidencias:

- ✓ **Documentación de fallos.** Cada error se describe con detalle, indicando el caso fallido, el comportamiento observado y el esperado.
- ✓ **Clasificación, según gravedad.** Se asigna una prioridad para corregir la incidencia (alta, media, baja).
- ✓ **Comunicación al equipo de desarrollo.** Los defectos son reportados para que se planifiquen las correcciones necesarias.

Ejemplo. Si durante la prueba el sistema no envía la confirmación automática tras la reserva, se registra el fallo y se notifica al equipo de desarrollo para solución.

Análisis y evaluación de resultados

Al finalizar la ejecución, se analizan los resultados globales para determinar el estado del producto:

- ✓ **Porcentaje de casos aprobados.** Se calcula la proporción de pruebas exitosas frente al total.
- ✓ **Impacto de las incidencias.** Se evalúa si los errores encontrados afectan funcionalidades críticas o son detalles menores.
- ✓ **Decisión de aceptación.** Basándose en el análisis, se decide si el producto cumple con los criterios para ser aprobado o si requiere ajustes.

Ejemplo. Un sistema con 95 % de pruebas aprobadas y fallos en funciones secundarias puede considerarse listo, mientras que si las pruebas críticas fallan, se debe continuar con correcciones.

Reporte y documentación final

La información recogida durante la ejecución se consolida en informes:

- ✓ **Informe de resultados.** Documento que resume las pruebas realizadas, resultados, fallos encontrados y recomendaciones.
- ✓ **Registro para trazabilidad.** Los informes permiten hacer seguimiento a las pruebas y a la evolución del proyecto.
- ✓ **Retroalimentación para stakeholders.** El reporte se comparte con clientes, gerentes y desarrolladores para tomar decisiones.

Ejemplo. Un reporte final incluye una tabla con cada caso de prueba, resultado, observaciones, responsable y estado actual de corrección.

Ejemplos prácticos y herramientas recomendadas

- ✓ **Ejemplo práctico.** En la validación de un sistema bancario, se ejecutan pruebas para verificar transferencias, consulta de saldo y generación de estados de cuenta, registrando todos los resultados para la toma de decisiones.
- ✓ **Herramientas recomendadas.** TestRail para gestionar casos y resultados; Jira para seguimiento de incidencias; Excel o Google Sheets, para reportes rápidos.

Validación final y aprobación del sistema

El principal propósito de la validación final es comprobar, de manera integral y exhaustiva, que el sistema satisface las necesidades funcionales, técnicas y de usabilidad establecidas durante la etapa de análisis y diseño. A través de esta validación, se verifica que no existan defectos críticos ni incumplimientos que puedan afectar la operación real del software.

Proceso de validación final

La validación final se lleva a cabo tras completar las pruebas de aceptación y después de que se hayan solucionado los errores más importantes detectados. En esta etapa se realizan las siguientes actividades:

- ✓ **Revisión exhaustiva.** Se revisan todos los documentos, requisitos y especificaciones, para asegurar que cada aspecto ha sido cubierto y verificado.
- ✓ **Pruebas integrales.** Se ejecutan pruebas finales en un entorno lo más parecido posible al entorno real de uso, considerando casos reales de operación.
- ✓ **Demostración al cliente.** Se presenta el sistema al cliente o usuario final, para que este pueda validar que el software cumple con sus expectativas y necesidades.
- ✓ **Confirmación formal.** Se solicita la aprobación formal del sistema mediante firmas o documentos de aceptación que reflejen la conformidad.

Ejemplos prácticos:

1. **Sistema de gestión hospitalaria.** Después de ejecutar todas las pruebas de aceptación y corregir los errores reportados, se invita al personal médico y administrativo a validar el sistema con casos de uso reales, como agendar citas, ingresar pacientes y generar reportes. La validación final se concluye con la firma del gerente del hospital que confirma la conformidad del sistema.
2. **Aplicación móvil bancaria.** Tras múltiples ciclos de pruebas, el banco realiza una sesión de validación final con usuarios seleccionados, quienes interactúan con la app para realizar transferencias y consultar saldos. Una vez satisfechos, emiten su aprobación para liberar la app al público general.

Importancia de la validación final

Esta fase asegura que el sistema no solo es técnicamente correcto, sino que también es aceptable para los usuarios y cumple con el propósito para el cual fue diseñado. Un sistema aprobado evita retrabajos costosos y garantiza la satisfacción del cliente, lo que contribuye a la reputación y éxito del proyecto.