



INSTITUTO SUPERIOR POLITÉCNICO DE  
CÓRDOBA

TECNICATURA SUPERIOR EN CIENCIA DE  
DATOS E INTELIGENCIA ARTIFICIAL

Módulo de Testeo de Software

Evidencia 1

Integrantes:

- Galeano, Agustin
- López, Erick
- Nüesch, Christian
- Zurita Rojo, Abigail



# Índice

<b>Tipos de Prueba de Software.....</b>	<b>3</b>
Pruebas Unitarias.....	3
Pruebas de Integración.....	4
Pruebas Funcionales de Rendimiento.....	5
Pruebas Funcionales de Estrés.....	5
Pruebas Funcionales de Regresión.....	7
Pruebas Funcionales de Humo.....	8
Pruebas de Caja Blanca.....	9
Pruebas de Caja Negra.....	10

# Tipos de Prueba de Software

## Pruebas Unitarias

Ejemplo Nacional: Plataforma de Reservas de Eventos (Eventbrite Argentina)

Eventbrite Argentina usa pruebas unitarias para asegurar que cada módulo de su plataforma de reservas de eventos funcione correctamente:

- Validación de funciones clave: Se prueban funciones específicas como la creación de eventos, la gestión de entradas, y la emisión de boletos. Cada una de estas funciones se verifica en aislamiento para asegurar que opere como se espera.
- Cálculos y validaciones: Por ejemplo, se prueba la función que calcula el total de ingresos por ventas de entradas para eventos, incluyendo impuestos y descuentos. Se ingresan diferentes escenarios de ventas para verificar que el cálculo sea correcto.
- Prevención de errores en integración: Las pruebas unitarias ayudan a identificar errores en módulos individuales antes de integrarlos en el sistema completo, garantizando que las reservas y la gestión de eventos sean precisas y eficientes.

Estas pruebas aseguran que la plataforma de reservas de eventos funcione correctamente y que todas las funcionalidades relacionadas con eventos y ventas sean precisas.

Ejemplo Nacional: Sistema de Atención al Cliente en una Compañía de Telecomunicaciones (Movistar Argentina)

Movistar Argentina utiliza pruebas unitarias para validar que las funciones del sistema de atención al cliente funcionen adecuadamente:

- Verificación de funciones específicas: Se realizan pruebas unitarias en funciones como la gestión de tickets de soporte, la actualización del estado de los tickets, y la asignación de técnicos a los casos. Cada función se prueba por separado para asegurar su correcto funcionamiento.
- Control de casos de uso: Por ejemplo, se prueba la función que actualiza el estado de un ticket de "abierto" a "cerrado" después de que el técnico haya resuelto el problema. Se ingresan diferentes escenarios de actualización para asegurar que el estado cambie correctamente.

- Prevención de problemas en el sistema: Al detectar y corregir errores en estas funciones, Movistar puede evitar problemas que afecten la resolución de tickets y la satisfacción del cliente.

Estas pruebas son esenciales para mantener un alto nivel de servicio y asegurar que cualquier actualización o cambio en el sistema no afecte negativamente su rendimiento.

## Pruebas de Integración

Ejemplo Externo: Amazon

Amazon, como una de las mayores plataformas de comercio electrónico, realiza extensivas pruebas de integración para asegurar que todos los componentes de su sistema funcionen correctamente juntos. Un caso real se relaciona con la integración de su sistema de pagos con diferentes métodos de pago:

- Integración de pasarela de pago: Se realizaron pruebas de integración para asegurarse de que las transacciones realizadas a través de diferentes pasarelas de pago, como tarjetas de crédito, PayPal, y Amazon Pay, se procesarán correctamente. Se probó que cada componente se comunicara correctamente con los servicios externos y que las transacciones se registraran en la base de datos de pedidos.
- Comprobación de carrito de compras y sistema de inventario: Se integró la funcionalidad del carrito de compras con el sistema de inventario. Se llevaron a cabo pruebas para garantizar que al añadir o eliminar productos del carrito, el inventario se actualizará correctamente, evitando ventas de productos agotados.
- Notificaciones y confirmaciones: La integración entre el sistema de pedidos y el sistema de notificaciones fue probada para asegurar que, después de una compra exitosa, se enviaran las confirmaciones a los clientes vía email o notificaciones push.

Estas pruebas de integración ayudaron a identificar y corregir problemas como la duplicación de transacciones o errores de sincronización de inventario, asegurando una experiencia de usuario fluida y confiable.

Ejemplo Nacional (Argentina): Plataforma de Gestión de Servicios Públicos (AFIP)

La Administración Federal de Ingresos Públicos (AFIP) realiza pruebas de integración para asegurar que los distintos módulos de su sistema de gestión de impuestos funcionen correctamente juntos:

- Procesos de declaración de impuestos: Se prueban las integraciones entre el módulo de declaración de impuestos, el sistema de validación de datos y el sistema de generación de informes. Por ejemplo, se verifica que una declaración de impuestos presentada se procese correctamente, se valide según las normativas y se refleje adecuadamente en los informes generados.
- Verificación de compatibilidad con sistemas externos: Se evalúa cómo el sistema de AFIP se integra con otras plataformas externas, como los sistemas bancarios para el pago de impuestos. Por ejemplo, se prueba que los pagos realizados a través de sistemas bancarios se reflejen correctamente en el sistema de AFIP.
- Aseguramiento de coherencia de datos: Las pruebas de integración garantizan que los datos de los contribuyentes se mantengan consistentes entre los diferentes módulos del sistema. Por ejemplo, se prueba que la información de un contribuyente actualizada en un módulo se refleje correctamente en todos los módulos relacionados

## Pruebas Funcionales de Rendimiento

Ejemplo Nacional (Argentina): Plataforma de Compra y Venta de Inmuebles (Zonaprop)

Zonaprop realiza pruebas funcionales de rendimiento para asegurar que su plataforma de búsqueda de propiedades funcione eficientemente:

- Control de tiempos de carga de listados: Se realizan pruebas para medir el tiempo que tarda en cargar los resultados de búsqueda de propiedades. Por ejemplo, se verifica que la lista de propiedades para una búsqueda específica se cargue en menos de 3 segundos.
- Gestión de imágenes: Se evalúa cómo la plataforma maneja la carga y visualización de imágenes de propiedades. Por ejemplo, se prueba que las fotos de alta resolución de las propiedades se carguen rápidamente y sin errores.
- Aseguramiento de rendimiento en picos de uso: Las pruebas de rendimiento aseguran que la plataforma mantenga su eficiencia durante picos de tráfico, como al final de mes cuando se realiza un alto número de búsquedas y consultas.

## Ejemplo Nacional: Mercado Libre Argentina

MercadoLibre, es un sitio de comercio electrónico, realiza pruebas de rendimiento para garantizar que su plataforma maneje grandes volúmenes de tráfico y transacciones:

- Pruebas de carga en días de promoción: Durante eventos como el "Hot Sale" o "CyberMonday," MercadoLibre realiza pruebas de carga para simular miles de transacciones por minuto. Estas pruebas ayudan a identificar cuellos de botella en la infraestructura, asegurando que la plataforma pueda manejar el aumento de tráfico sin caídas o ralentizaciones.
- Tiempo de respuesta de búsqueda y pago: Se mide el tiempo que toma cargar los resultados de búsqueda y completar el proceso de pago. El objetivo es mantener estos tiempos lo más bajos posible, mejorando la experiencia del usuario y reduciendo la tasa de abandono.
- Escalabilidad: Las pruebas de rendimiento evalúan la capacidad del sistema para escalar automáticamente, añadiendo recursos de servidor según la demanda. Esto incluye la distribución de la carga entre servidores y la adaptación a picos inesperados en el tráfico.

Estas pruebas permitieron a MercadoLibre identificar áreas críticas para mejorar y asegurar que, durante picos de demanda, la plataforma permanezca rápida y fiable, manteniendo una experiencia de compra positiva para los usuarios.

## Pruebas Funcionales de Estrés

### Ejemplo Externo: Twitter

Twitter realiza pruebas de estrés para garantizar que su plataforma maneje eventos de tráfico masivo sin fallos:

- Simulación de eventos mundiales: Durante eventos globales como elecciones, eventos deportivos importantes, o crisis, Twitter puede experimentar un aumento súbito en el tráfico. Las pruebas de estrés simulan estos escenarios al generar millones de solicitudes y tweets en un corto periodo, evaluando cómo el sistema maneja la sobrecarga.
- Comportamiento bajo sobrecarga: Las pruebas evalúan cómo Twitter responde cuando se excede la capacidad normal del sistema, verificando si los servicios

críticos (como la publicación de tweets, el envío de notificaciones, y el streaming de datos) continúan funcionando o si se producen retrasos o errores.

- Mitigación de problemas: Durante las pruebas de estrés, se verifica la eficacia de las estrategias de mitigación, como la implementación de colas de mensajes y la activación de recursos adicionales. Estas estrategias ayudan a mantener la disponibilidad del servicio incluso bajo condiciones extremas.

Ahora Twitter puede identificar los límites de su infraestructura y mejorar la resiliencia del sistema, asegurando que el servicio permanezca operativo durante picos inesperados de tráfico.

Ejemplo Nacional: Banco Central de la República Argentina (BCRA)

El BCRA realiza pruebas de estrés en sus sistemas de pago y procesamiento de transacciones para asegurar su estabilidad:

- Simulación de altas transacciones financieras: Durante días de alta actividad financiera, como el cierre de mes o el período de presentación de impuestos, el sistema del BCRA experimenta un volumen masivo de transacciones. Las pruebas de estrés simulan este escenario al procesar millones de transacciones en poco tiempo, evaluando la capacidad del sistema para mantener la operación sin errores.
- Verificación de la estabilidad: Se evalúa si los sistemas pueden continuar operando correctamente bajo condiciones extremas, como picos en la carga o ataques cibernéticos. Esto incluye verificar la integridad de las transacciones y la seguridad de los datos durante situaciones de sobrecarga.
- Recuperación y resiliencia: Las pruebas también examinan la capacidad del sistema para recuperarse de fallos provocados por la sobrecarga, incluyendo la correcta reinicialización de servicios y la prevención de pérdida de datos.

Estas pruebas de estrés permiten al BCRA mantener la confiabilidad de sus sistemas financieros, garantizando la continuidad y seguridad de las operaciones incluso durante picos de actividad extrema.

## Pruebas Funcionales de Regresión

Ejemplo Externo: WhatsApp

WhatsApp implementa pruebas de regresión para asegurarse de que nuevas características no interfieran con las funciones existentes:

- Actualización de funciones: Cuando se introducen nuevas características, como la capacidad de editar mensajes después de enviarlos, se ejecutan pruebas de regresión para verificar que las funciones esenciales (como el envío de mensajes, notificaciones, y llamadas) no se vean afectadas por los cambios.
- Automatización de pruebas: WhatsApp utiliza suites automatizadas de pruebas de regresión que ejecutan cientos de casos de prueba para verificar las funcionalidades críticas de la aplicación. Esto incluye la revisión del cifrado de extremo a extremo, la correcta entrega de mensajes, y la calidad de las llamadas.
- Validación en diferentes plataformas: Como WhatsApp opera en múltiples plataformas (iOS, Android, Web), las pruebas de regresión se ejecutan en todas ellas para asegurar la coherencia y estabilidad de la aplicación en cualquier entorno.

Estas pruebas garantizan que las actualizaciones no introduzcan fallos y que la experiencia del usuario se mantenga estable y confiable.

Ejemplo Externo: Pruebas de Rendimiento y Regresión en una Aplicación de Streaming de Video

Durante el proceso de optimización del motor de reproducción de video, la aplicación de streaming realiza una serie de pruebas para asegurar tanto la mejora en el rendimiento como la estabilidad de las funciones existentes.

Al implementar mejoras en el motor de reproducción, se llevan a cabo pruebas de rendimiento para evaluar el impacto de estas optimizaciones en la carga y fluidez de la reproducción de video. Estas pruebas miden la rapidez con la que los videos se inician y se reproducen, así como la estabilidad del streaming después de las actualizaciones.

Simultáneamente, se ejecutan pruebas de regresión para garantizar que las mejoras no afecten negativamente a las funcionalidades existentes. Esto incluye verificar que la selección de la calidad del video y la reproducción en segundo plano continúen funcionando sin problemas.



Se espera una mejora en el rendimiento general del video, mientras que las funciones previas deben seguir operando correctamente y sin degradación, garantizando así una experiencia de usuario consistente y satisfactoria.

## Pruebas Funcionales de Humo

### Ejemplo Externo: Spotify

Spotify realiza pruebas de humo regularmente para asegurar que las funcionalidades básicas de la aplicación estén operativas:

- Verificación de funciones principales: Cada vez que se lanza una nueva versión de la aplicación, se ejecutan pruebas de humo para verificar que los usuarios puedan iniciar sesión, reproducir música, navegar por las listas de reproducción, y realizar búsquedas de canciones sin problemas.
- Validación del reproductor: Se prueban funciones críticas como la reproducción, pausa, y cambio de pistas para asegurarse de que el reproductor de música funcione correctamente. Estas pruebas rápidas aseguran que el núcleo de la experiencia del usuario esté intacto antes de proceder con pruebas más detalladas.
- Detección temprana de problemas: Si una prueba de humo falla, el equipo de desarrollo detiene el proceso de implementación para abordar los problemas identificados, evitando que errores graves lleguen a los usuarios finales.

Con esto Spotify ya puede lanzar actualizaciones con confianza, sabiendo que las funcionalidades esenciales de la aplicación están intactas y listas para pruebas más exhaustivas.

### Ejemplo Provincial: Sistema de Gestión de Tránsito en Salta

Un sistema de gestión de tránsito en Salta realiza pruebas de humo para validar rápidamente la funcionalidad principal de su aplicación:

- Chequeo de funciones clave: Antes de implementar cambios en el sistema, como la actualización del módulo de multas, se ejecutan pruebas de humo para verificar que los usuarios puedan acceder a información sobre el estado de las multas, realizar pagos, y ver historial de infracciones sin problemas.

- Validación de integridad del sistema: Las pruebas de humo también comprueban que los módulos esenciales, como la consulta de puntos en la licencia de conducir y el registro de vehículos, estén funcionando correctamente. Esto asegura que los servicios principales estén operativos y listos para pruebas más detalladas.
- Prevención de problemas mayores: Si una prueba de humo detecta un problema en una funcionalidad principal, se detiene la actualización y se corrige el error, evitando que se produzcan interrupciones en el servicio para los ciudadanos.

Estas pruebas rápidas aseguran que el sistema de gestión de tránsito mantenga la funcionalidad básica operativa y previenen la introducción de errores críticos en la plataforma.

## Pruebas de Caja Blanca

Ejemplo Nacional: Sistema de Pago Electrónico (Pim)

Pim, una plataforma de pago electrónico utilizada en Argentina, implementa pruebas de caja blanca para garantizar la seguridad y eficiencia de sus transacciones:

- Validación del código fuente: Los ingenieros de Pim realizan pruebas de caja blanca para revisar el código que maneja las transferencias de fondos. Esto incluye verificar que los algoritmos de cifrado y las funciones de autenticación se implementen correctamente, garantizando la seguridad de las transacciones.
- Chequeo de lógica y flujos de control: Se analizan los flujos de control para asegurarse de que los fondos se deducen correctamente de la cuenta del remitente y se acreditan en la del receptor. Además, se examinan las condiciones de error para confirmar que se manejen adecuadamente, como en casos de fondos insuficientes o errores en la red.
- Optimización de procesos: Al realizar estas pruebas, se detectan y corrigen posibles cuellos de botella en el código, mejorando el rendimiento del sistema. Esto garantiza que las transacciones se procesen de manera eficiente y segura.

Para mantener un sistema confiable, Pim implementa rigurosos procesos de análisis y verificación de cada parte del código. Esto asegura la integridad y seguridad de todas las operaciones financieras.

Ejemplo Personal: Proyecto de Control de Inventario

En un proyecto personal de control de inventario, se llevaron a cabo pruebas de caja blanca para garantizar que los algoritmos de cálculo de stock funcionaran correctamente:

- Revisión de algoritmos: Se analizaron las funciones de cálculo de stock para asegurarse de que las operaciones matemáticas, como la adición y sustracción de cantidades, se ejecutarán correctamente. Esto involucró recorrer el código línea por línea, verificando que no hubiera errores lógicos.
- Cobertura de código: Las pruebas incluyeron la verificación de todos los caminos posibles en el código, como la actualización del stock al agregar nuevos productos o al registrar una venta. Esto aseguró que todas las condiciones y bucles se ejecutarán según lo esperado.
- Prevención de errores: Durante la revisión, se detectaron y corrigieron errores potenciales que podrían haber llevado a cálculos incorrectos del inventario, evitando problemas futuros en la gestión del stock.

Con ello garantizamos que el sistema de control de inventario funcionara de manera precisa y eficiente, previniendo discrepancias en los registros.

## Pruebas de Caja Negra

Ejemplo Local: Sistema de Registro de Estudiantes en una Universidad (Universidad Nacional de Córdoba)

La Universidad Nacional de Córdoba utiliza pruebas de caja negra para asegurar el correcto funcionamiento de su sistema de registro de estudiantes:

- Validación de funcionalidades del usuario: Se prueban funciones como la inscripción en cursos, la consulta de horarios y la generación de certificados de inscripción sin tener acceso al código fuente. Por ejemplo, se verifica que los estudiantes puedan inscribirse en cursos, consultar los horarios de clases y descargar sus certificados de inscripción.
- Pruebas de casos de uso: Por ejemplo, se prueba que al inscribirse en un curso, el sistema actualice correctamente el historial académico del estudiante y que los horarios de clases se muestren correctamente en la interfaz del usuario. También se verifica que el certificado de inscripción sea generado y descargado correctamente.

- Aseguramiento de funcionalidades críticas: Las pruebas de caja negra garantizan que el sistema de registro funcione según lo esperado desde la perspectiva del usuario, asegurando una experiencia de inscripción y consulta eficiente.

Ejemplo Nacional: Prueba de Caja Negra y de Estrés del Sistema de Reservas de Vuelos (Despegar.com)

Se llevan a cabo pruebas de caja negra para asegurar que las funcionalidades de búsqueda de vuelos, selección de asientos y reserva de billetes operen correctamente sin necesidad de conocer el código interno del sistema. Posteriormente, se realizan pruebas de estrés para evaluar cómo el sistema maneja un alto volumen de solicitudes simultáneas durante temporadas de alta demanda. Estas pruebas garantizan que las funciones de reserva y pago continúen funcionando sin problemas bajo carga.

Las pruebas confirman que las funcionalidades específicas operan correctamente desde la perspectiva del usuario y que el sistema maneja eficientemente el alto volumen de tráfico, asegurando una experiencia de reserva fluida incluso durante picos de alta demanda.