

Banco GATO FINANCIERO

Producto Ahorro Digital

Plan de pruebas

Introducción

El objetivo principal de este plan de pruebas es garantizar que el producto Ahorro Digital del Banco GATO FINANCIERO funciona de manera óptima en todas las situaciones posibles. Deseamos proporcionar un producto de alta calidad que satisfaga las necesidades de nuestros usuarios y supere sus expectativas. Para lograr esto, debemos identificar y corregir cualquier error o defecto antes de que el software sea lanzado al mercado.

Ahorro Digital es una plataforma web en donde los usuarios pueden explorar los productos de ahorro y simular cuánto podrían ganar con sus depósitos, generando información apropiada para la inversión y recaudo de nuestros productos.

Objetivos

- Garantizar y asegurar, mediante la ejecución de casos de prueba basados en la documentación entregada por el área funcional o de producto, que el usuario pueda consultar y simular correctamente los productos de ahorro de su dinero.
- Poder garantizar y validar la funcionalidad correcta del flujo de registro e inicio de sesión en el sistema de la funcionalidad ahorro digital.
- Verificar el funcionamiento del simulador de ahorros contemplando varios escenarios positivos y negativos.
- Asegurar el flujo y la visualización adecuada de manejo de errores dentro del sistema, los cuales son mostrados al usuario.
- Confirmar la accesibilidad y la navegación clara y adecuada para cada una de las funcionalidades que se muestran al usuario final.
- Garantizar la carga de manera apropiada de la información dentro de nuestra aplicación Web.

Alcance

Teniendo en cuenta los requerimientos suministrados para el análisis, diseño y ejecución de esta solución, menciono a continuación lo que hace parte del alcance de estas pruebas, lo que se contemplará durante la ejecución y validación del equipo de calidad de software.

- Validar el funcionamiento correcto del Onboarding, flujo de usuario de registro y el login de los usuarios de la aplicación web del producto de ahorro digital.
- Validar funcionamiento del simulador de ahorros del sistema de ahorro digital.
- Validacion del siluador de ahorros
- Validacion de funcionalidades de productos de ahorro
- Verificacion de formularios y navegacion de la UI de la aplicación
- Manejo de mensajes y de errores de respuesta

Fuera del alcance

- Integracion con plataformas externas no especificadas en el alcance
- Pruebas de carga y de estress
- Pruebas de seguridad no especificadas
- Porcesos de apertura de cuentas reales
- Aplicaciones moviles solo versión web

Riesgos

Riesgo	Impacto	Probabilidad	Plan de mitigaciòn
P0: Fallo en el registro de usuarios	Critico: bloque por completo el acceso a las funcionalidades de la solucion	Alta	Pruebas exhaustivas en cada una de las funcionalidades, validacion de campos, negativos, positivos
P0:Error en la autenticacion	Critico: usuarios no pueden ingresar	Alta	Validacion de tokens y sesiones
P0: Datos sensibles expuestos	Critico: Validaciòn de seguridad datos críticos expuestos de los usuarios	Alta	Verificar encriptaciòn, revisar logs
P1: Calculos incorrectos en el simulador	Alto: Informacion de datos financiera incorrecta	Alta	Validaciòn de fórmulas y cálculos con diferentes escenarios
P1: Timeout en API del simulador	Alto: experiencia de usuario lenta y degradada	Alta	Implementar mecanismos de retry y timeouts

			para esperas controladas
P2: Información de los productos no carga imágenes, descripciones	Medio: Afecta la presentación de los datos a consultar y visualizar por el usuario	Media	Pruebas con diferentes datos de prueba y diferentes condiciones.
P2: Filtros de productos no fusionan	Medio: uso reducido y mal experiencia al cliente	Media	Pruebas exhaustivas condiciones de datos diferentes.

Criterios de Aceptación

- Diseño de casos de prueba funcionales que cubran el 100% de las funcionalidades identificadas dentro del alcance.
- Los casos de prueba identificados como P1 y P2 deben pasar al 95%.
- La cobertura de pruebas debe ser mayor al 80%.
- Los requerimientos funcionales especificados y aprobados con alcances cerrados.
- Respuesta del API menor a 2 segundos.
- No hay bugs críticos identificados abiertos antes de salir a producción.

Estrategia de pruebas

Nuestra estrategia de pruebas es exhaustiva y rigurosa, e incluirá varios niveles de pruebas para garantizar la calidad del software. Empezaremos con las pruebas back, que se centran en validar los servicios con los cuales se comunica la funcionalidad evaluada en este caso ahorro digital para asegurar que el flujo de la información de datos, contratos, esquema, estructura de campos y formatos funcionan correctamente. Seguiremos con las pruebas de integración, que verifican cómo estas partes individuales trabajan juntas. Luego realizaremos pruebas de sistema para evaluar el rendimiento del software como un todo. Finalmente, llevaremos a cabo pruebas de aceptación para confirmar que el software cumple con los requisitos y expectativas del usuario.

Se contemplará una estrategia Mixta usando pruebas manuales de sistema para revisar funcionalidades core de los cambios realizados y automatización para las pruebas de regresión del flujo base y crítico después de solucionar bugs reportados.

Tipos de pruebas

Para asegurar una cobertura de pruebas completa, realizaremos tanto pruebas funcionales como algunas no funcionales. Las pruebas funcionales evaluarán las funciones y características del software para asegurar que se comportan como se espera.

Las pruebas no funcionales evaluarán aspectos del software que no están directamente relacionados con una función específica, tales como la seguridad, la usabilidad y compatibilidad.

Funcionales

Para los casos de prueba de: P0: Onboarding, P1 Simulador, P2 Productos, realizando un primer ciclo de pruebas con un Smoke test contemplando una funcionalidad básica para cada funcionalidad dependiendo su prioridad en este caso será de la siguiente manera:

Prioridad 1: Smoke test Onboarding

Prioridad 2: Simulador

Prioridad 3: Productos

De acuerdo a los resultados realizaremos un segundo ciclo de prueba contemplando la funcionalidad core en este caso para cada una de los flujos: Onboarding, P1 Simulador, P2 Productos por su prioridad.

Reportaremos los errores usando una herramienta para el seguimiento adecuado en este caso Jira y priorizaremos por impacto.

Alto: para las funcionalidades críticas e importantes

Medio: funcionalidades secundarias pero para flujos también importantes

Baja: para defectos de funcionalidades consideradas de prioridad baja que no afectan o bloquean comportamientos importantes del sistema.

Pruebas de compatibilidad: verificaremos las funcionalidades en diferentes navegadores (Chrome, Edge, Firefox) y sistemas operativos con el objetivo de garantizar la experiencia al usuario.

Pruebas de usabilidad: verificaremos, botones, apariencia de navegación, sistema intuitivo, mensajes claros, funcionalidades de fácil uso al usuario, fácil legibilidad de la información.

Pruebas de Seguridad: verificaremos las vulnerabilidades del software usando herramientas como OWASP ZAP que nos permitan simular ataques al sistema y cómo responde frente a esto, proteger la información, verificación de contraseñas seguras combinando diferentes métodos de simulación de fugas de la información.

Este enfoque combinado nos permitirá identificar y solucionar cualquier problema potencial que pueda afectar la experiencia del usuario o la calidad del software. Validar desde etapas tempranas y poder brindar feedback al equipo de desarrollo de una forma rápida y oportuna, ya que se contemplan en diferentes etapas del desarrollo.

Recursos: Para garantizar un proceso de prueba efectivo, necesitaremos un equipo diverso y experto de pruebas, incluyendo los siguientes perfiles:

- Analistas QA
- Ingenieros de automatización
- Líder de pruebas o gestor.

El hardware necesario incluirá máquinas de alto rendimiento para ejecutar el software en diversas condiciones y entornos. Además, utilizaremos herramientas de prueba líderes en la industria, como Jira para la gestión de pruebas y playwright para la automatización de pruebas.

Responsabilidades: El equipo de pruebas será responsable de todas las pruebas y de la documentación de los resultados.

Cronograma: El proceso de pruebas se realizará por un periodo de 3 semanas. Este período incluirá el análisis de la documentación entregada, resolución de dudas, creación de datos, creación de casos de prueba, la realización de pruebas y la generación de informes. Se designará tiempo adicional para la corrección de errores y las pruebas de regresión.

Entorno de prueba: la ejecución de las diferentes pruebas se realizará en el ambiente de QA previamente configurado y desplegado con la versión del diseño y solución de ahorro digital y en ambiente pre-productivo donde se realizar pruebas de regresión una vez se hallan solucionado todos los defectos reportados para validar que no se hayan afectados funcionalidades básicas e importantes del sistema.

Entregables: a continuación relaciono los entregables que se considerarán en cada fase del proceso.

- Plan de pruebas
- Diseño de casos de prueba
- Evidencias de las pruebas realizadas
- Informes de avances de las pruebas (Tableros de herramientas de ejecución)
- Informe de Cierre con la evaluación de los resultados y lecciones aprendidas.

Casos de prueba

El diseño de los casos de prueba es crucial para garantizar una cobertura completa de todas las funciones del software. Cada caso de prueba se desarrollará con un objetivo específico y se basará en la ejecución de diferentes escenarios por módulo a probar, incluyendo el flujo principal de operación y las condiciones extremas. Para hacer que el

proceso sea apropiado y eficiente, utilizaremos la herramienta Zephyr, una extensión de Jira, para documentar y rastrear todas las pruebas. Además de documentar los pasos necesarios para ejecutar cada caso de prueba, Zephyr también nos permite registrar los datos de entrada, las condiciones previas y los resultados esperados, proporcionando así un registro completo y detallado de cada prueba.

Id	Prioridad	Modulo	Descripcion de caso de prueba	Precondiciones	Pasos	Resultado esperado	Tipo de prueba
1	P0	Onboarding	Realizar registro en el sistema de manera correcta con datos validos	Datos validos de usuario	1.Ir a la pagina del banco 2.Dar clic en el boton ingresar al portal 3.Dar clic en el boton registrarse 4.Dar clic en confirmacion de la politica 5.Dar clic en el boton continuar 6.Diligenciar los campos solicitados	Aparece un mensaje de usuario creado exitosamente, redirecciona ala pagina de inicio del banco	Funcional
2	P0	Onboarding	Realizar registro en el sistema de forma fallida datos invalidos	Datos del correo duplicado e-mail existente en el sistema	1.Ir a la pagina del banco 2.Dar clic en el boton ingresar al portal 3.Dar clic en el boton registrarse 4.Dar clic en confirmacion de la politica 5.Dar clic en el boton continuar 6.Diligenciar los campos solicitados 7.Dar clic en el boton Continuar	Aparece un mensaje en rojo doden se indica error e-mail ya existe	Negativo
3	P0	Onboarding	Validar campos obligatorios en el formulario de registro	datos incompletos solo diligenciar campo nombre	1.Ir a la pagina del banco 2.Dar clic en el boton ingresar al portal 3.Dar clic en el boton registrarse 4.Dar clic en confirmacion de la politica 5.Dar clic en el boton continuar 6.En el formulario solo llenar el nombre los demás campos vacios.	Aparece un mensaje campos obligatorios y no se habilita el boton continuar	Validación
4	P0	Onboarding	Validar que cuando el recaptcha sea invalido bloquee el registro del usuario el sistema	Datos correctos del usuario	1.Ir a la pagina del banco 2.Dar clic en el boton ingresar al portal 3.Dar clic en el boton registrarse 4.Dar clic en confirmacion de la politica 5.Dar clic en el boton continuar 6.Diligenciar los campos solicitados 7.Dar clic en el boton Continuar 8. No resolver recaptcha	Aparece un mensaje indicando que debe resolver recaptcha y no habilita el boton registrar	Validación
5	P0	Onboarding	Validar que al realizar el registro con contraseñas incorrectas no permita acceso	Usuario registrado previamente	1.Ir a la pagina del banco 2.Dar clic en el boton ingresar al portal 3.Diligenciar Usuario 4.Dar clic en el boton siguiente 5.Diligenciar contraseña incorrecta	Aparece un mensaje indicando contraseña incorrecta y no permite el ingreso, no se habilita el boton para ingresar	Negativo
6	P0	Onboarding	Validar que se permita la recuperacion de la contraseña	usuario previamente registrado	1.Ir a la pagina del banco 2.Dar clic en el enlace ¿olvido su contraseña? 3.Diligenciar datos solicitados 4. Dar clic en continuar 6. Se muestra el mensaje de confirmacion 7.Ingresar al correo y seguir los pasos solicitados	Se permite el ingreso correctamente al sistema con la contraseña nueva creada por el usuario	Funcional
7	P0	Onboarding	Validar que no se permita la recuperacion de contraseña con un usuario que no este previamente registrado	usuario no registrado en el sistema	1.Ir a la pagina del banco 2.Dar clic en el enlace ¿olvido su contraseña? 4.Diligenciar datos solicitados 5. Dar clic en continuar	Se muestra un mensaje indicando que el usuario no existe en sistema y que debe registrarse, se muestra el boton de registrarse	Validación(error 404)
8	P1	Simulador	Validar en el simulador cuando el monto es valido	1. Usuario registrado	1. Ingresar al sistema con usuario y contraseña 2. Seleccionar el monto valido 3. Seleccionar plazo 3. Dar clic en el botón simular	Se realiza el calculo correctamente con detalles del interes y montos de ganancia	Funcional
9	P1	Simulador	Validar el simulador cuando el monto es 0	1. Usuario registrado 2. usuario logueado en el sistema	1.Dar clic en el campo monto y colocar 0 2.Seleccionar plazo	Se muestra un mensaje indicando que el monto debe ser mayor a 0 y no habilita el botón simular	Negativo
10	P1	Simulador	Validar el simulador cuando el monto es el máximo y no se habilita el botón simular	1. Usuario registrado 2. usuario logueado en el sistema	1.Dar clic en el campo monto y colocar el valor maximo permitido 2.Seleccionar plazo	Se muestra un mensaje indicando que supera el valor limite permitido y no habilita el botón simular	Validación
11	P1	Simulador	Validar los calculos mostrados en el simulador con un valor permitido y plazos permitidos	1. Usuario registrado 2. usuario logueado en el sistema	1.Dar clic en el campo monto permitido 2.Seleccionar plazo 3. dar clic en el botón simular	Se muestra los calculos de la rentabilidad esperada y las tasa de interes aplicadas de acuerdo de las condiciones dadas	Funcional
12	P1	Simulador	Validar que al realizar cambios en los plazos y montos se recalculo los valores	1. Usuario registrado 2. usuario logueado en el sistema 3. Usuario previamente hizo un calculo con una tasa de 12 meses 4.Usuario no cambia el monto	1.Seleccionar plazo 6 meses 3. dar clic en el botón simular	Se muestra el recálculo de la rentabilidad esperada y las tasa de interes aplicada de 6 meses de acuerdo de las condiciones dadas	Funcional
13	P2	Productos	Realizar la validacion que los productos carguen correctamente en el sistema	1. Usuario registrado 2. usuario logueado en el sistema	1.Ir a la pagina de inicio 2. Dar clic en la opción productos	Se muestran los productos de forma correcta con su informacion de forma detallada	Funcional
14	P2	Productos	Realizar la validacion que no se muestren productos no activos o existentes	1. Usuario registrado 2. usuario logueado en el sistema 3. Contar con un producto inactivo	1.Ir a la pagina de inicio 2. Dar clic en la opción productos	No se muestra el producto inactivo, solo se muestra el listado de productos activos en el sistema	Negativo
15	P1	Onboarding	Realizar la validacion desde diferentes navegadores en el modulo de Onboarding	1.Tener url de aplicación correctamente cargada	1. Acceder ala Url de la web desde diferentes navegadores, Chrome, Edge, Firefox	El diseño de la pagina se ve correctamente, carga correctamente, permite la visualizacion de los elementos correctamente	Compatibilidad

Automatización

Para aumentar la eficiencia y precisión, utilizaremos Playwright para las pruebas de extremo a extremo (E2E). Esta herramienta nos permite automatizar acciones de usuario en navegadores web, probando así la funcionalidad completa de nuestra aplicación. Para las pruebas de API, utilizaremos Postman nos permitirá diseñar y ejecutar pruebas de API de forma rápida y sencilla.

Ajdunto Script

Reporte: evidencia de errores

Caso 1

Bug 1: Al intentar hacer el registro del usuario permite continuar sin validar campos obligatorios

Severidad: Alta

Estado: Abierto

- ✖ **Descripción actual:** Al realizar el registro en la pagina de registro de nuevo usuario y no seleccionar campo últimos 6 números de la tarjeta permite continuar sin hacer la validación de obligatoriedad
- ✚ **Resultado esperado:** Al intentar hacer un registro de un usuario nuevo y no diligenciar el campo últimos números de tarjeta , se muestre un mensaje que indique campos obligatorios y no permita continuar.

! **Precondición:**

- Estar en al pagina: <https://www.bancocajasocial.com/>
- Usuario nuevo no debe estar registrado

⚠ Pasos para reproducir el error:

1. Ingresar la formulario de registro
2. Dar clic en el campo usuario y diligenciar CC1013590769
3. Dar clic en el botón Continuar

ⓘ Evidencias:

Paso 2/5

Crear contraseña

A continuación, ingrese la siguiente información relacionada con su producto.

Usuario

Ejemplo: CC1234567890

Últimos 6 números de tarjeta débito

Puede ser cualquiera de sus tarjetas débito

Clave de tarjeta débito



Campo obligatorio

Clave de la tarjeta indicada en el campo anterior

¿Sin tarjeta débito? Realice el proceso con tarjeta de crédito

Continuar

Caso 2

Bug 2: la seguridad de las contraseñas no se realiza de forma adecuada y segura.

Severidad: **Alta**

Estado: Abierto

- ✖ **Descripción actual:** al verificar la seguridad de las contraseñas se evidencia que el sistema no realiza la validación que sean contraseñas fuertes que contengan caracteres o números combinaciones permite el registro de contraseñas débiles de fácil recordatorio.
- ✖ **Resultado esperado:** El sistema cuente con un sistema fuerte de validación de contraseñas, se muestren mensajes como contraseña débil y no habilite el botón registrar.

! Precondicion:

- Estar en al pagina: <https://www.bancocajasocial.com/>
- Usuario nuevo no debe estar registrado

⚠ Pasos para reproducir el error:

4. Ingresar la formulario de registro
5. Dar clic en el campo usuario y diligenciar CC1013590769
6. Diligenciar campo clave y colocar 1234
7. Se habilita el boton continuar y permite el registro

ℹ Evidencias:

Paso 2/5

Crear contraseña

A continuación, ingrese la siguiente información relacionada con su producto.

Usuario
CC1013590769

Ejemplo: CC1234567890

Últimos 6 números de tarjeta débito
417825

Puede ser cualquiera de sus tarjetas débito

Clave de tarjeta débito
12345 

Clave de la tarjeta indicada en el campo anterior

¿Sin tarjeta débito? Realice el proceso con tarjeta de crédito

Continuar 

Caso 3

Bug 3: El cálculo de los intereses en el simulador no es correcto.

Severidad: **Media**

Estado: Abierto

- ✖ **Descripción actual:** se está realizando una simulación de ahorro digital sin embargo al mostrar la información se evidencia que el cálculo no es correcto el interés está calculado sobre un porcentaje incorrecto.
- ➕ **Resultado esperado:** El sistema calcule correctamente el interés de acuerdo al tiempo dado por el usuario de forma correcta 500000×12 meses
- ❗ **Precondicion:**
 - Estar en la página: <https://www.bancocajasocial.com/>
 - Usuario registrado en el sistema
- ⚠ Pasos para reproducir el error:
 8. Ingresar a la opción productos
 9. Dar clic en la opción simulador
 10. Dar clic en ahorro digital
 11. Diligenciar el valor
 12. Seleccionar el plazo en días
 13. Dar clic en simular
- ℹ Evidencias:



Recomendaciones para mejorar el flujo

UI: Es fundamental resaltar que, para atraer usuarios y optimizar la experiencia de navegación, la aplicación o sitio web debe contar con una interfaz intuitiva, accesible y funcional. Una interfaz bien diseñada contribuye significativamente a mejorar la interacción del usuario, facilitando la exploración de productos y procesos de compra o distribución de manera ágil y eficiente.

Se recomienda estructurar la información de forma clara y jerárquica, emplear íconos visualmente atractivos y botones con textos precisos que orienten la acción del usuario. Asimismo, es importante minimizar la cantidad de redirecciones o pasos innecesarios entre páginas, garantizando una navegación fluida y coherente con los objetivos del negocio y las expectativas del usuario final.

En cuanto al simulador del producto usar colores llamativos ayuda a que los usuarios enfoquen su atención y puede dar claridad visual.

Técnico: en cuanto a automatización se recomienda crear flujos E2E entre el registro, inicio de sesión, simulación y visualización de productos que permite garantizar el flujo total base cuando se realice algún cambio en el sistema.

Implementar **pipeline CI/CD con ejecución automática** de pruebas de regresión, esto mejora los tiempos de feedback de una manera mucho más temprana sobre los errores encontrados al equipo de desarrollo permitiendo así su corrección y mejora la calidad de los productos.

