



TRABAJO INTEGRADOR

EQUIPO 3



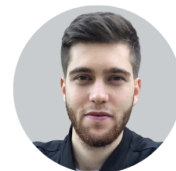
Paul Duarte



Fernando Ayala



Brian Retamal



Lisandro Fideleff



Alexis Henriquez



Mario Yustos

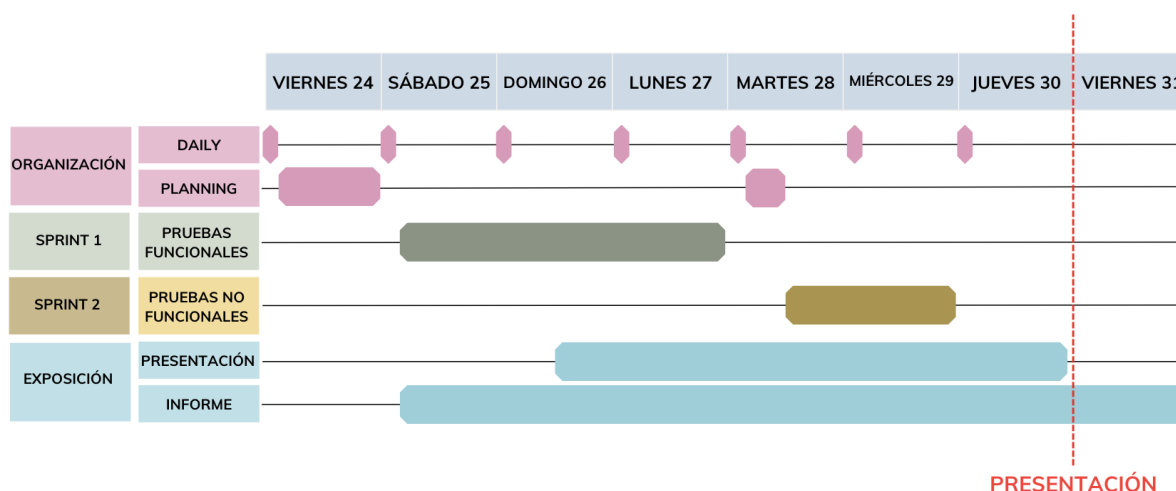


ÍNDICE

Organización del equipo.....	3
Diagrama de Gantt.....	3
El producto.....	4
Épica.....	4
Mapa del sitio.....	4
Diagrama de flujo acumulado.....	5
Ejecuciones totales de casos de prueba.....	5
Desarrollo del trabajo.....	6
Cantidad total de incidencias.....	6
Sprint 1.....	7
Sprint 2.....	8
Seguridad.....	9
API Testing.....	11
Introducción para el testing de apis.....	11
Objetivos.....	11
Automatización con Snippets.....	11
Resultados obtenidos en postman.....	11
Performance de API Testing.....	13
Automation.....	15
Bases de Datos.....	17
Diagrama ER original.....	17
BBDD Propuesta.....	18
UX/UI.....	19
Versión actual.....	19
Propuesta de Mockup.....	19
Accesibilidad.....	20
Conclusión.....	21

Organización del equipo

Diagrama de Gantt



A la hora de compartir cómo hemos organizado el trabajo dentro del equipo, decidimos valernos de un Diagrama de Gantt, ya que expresa gráficamente las diferentes instancias que transitamos para conseguir el objetivo final.

Para coordinar el trabajo a realizar, hemos llevado adelante Dailies donde los integrantes compartimos nuestros avances, bloqueos y los objetivos para el día. También hemos efectuado una Planning para definir y organizar el trabajo antes del inicio de cada Sprint. Todos estos componentes conforman la metodología Scrum que llevamos a cabo.

En el Sprint 1 hemos efectuado todas las pruebas funcionales que se encontraban relacionadas con los flujos principales que enunciaremos a continuación con la presentación del mapa del sitio.

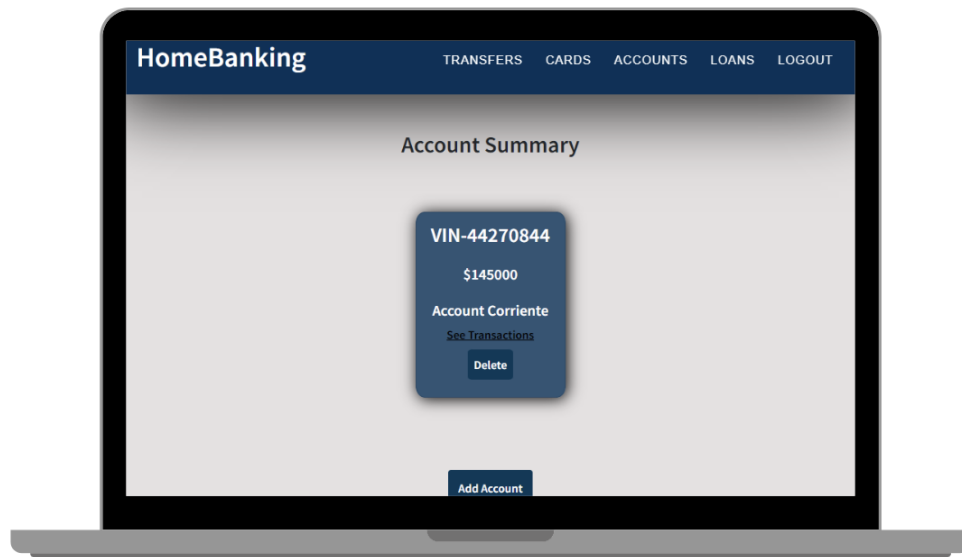
En el Sprint 2 hemos realizado todas las pruebas no funcionales que consideramos esenciales para intentar acercarnos al objetivo de asegurar la calidad del producto analizado. Hemos realizado pruebas de: APIs, Performance, Automation, Seguridad, VPN, Accesibilidad, UX/UI y recomendaciones sobre las BBDD.

A la par de la ejecución del trabajo, hemos desarrollado el presente informe y la presentación que, oportunamente, ha podido visualizarse en la exposición del análisis del producto.

Para llevar adelante la tarea solicitada hemos requerido utilizar una vasta cantidad de herramientas. A continuación listaremos los logos de las principales para que el lector pueda comprender cómo se han efectuado los diferentes análisis que se observarán en el trabajo realizado. Volveremos a cada una de las herramientas cuando sea oportuno a lo largo del informe.



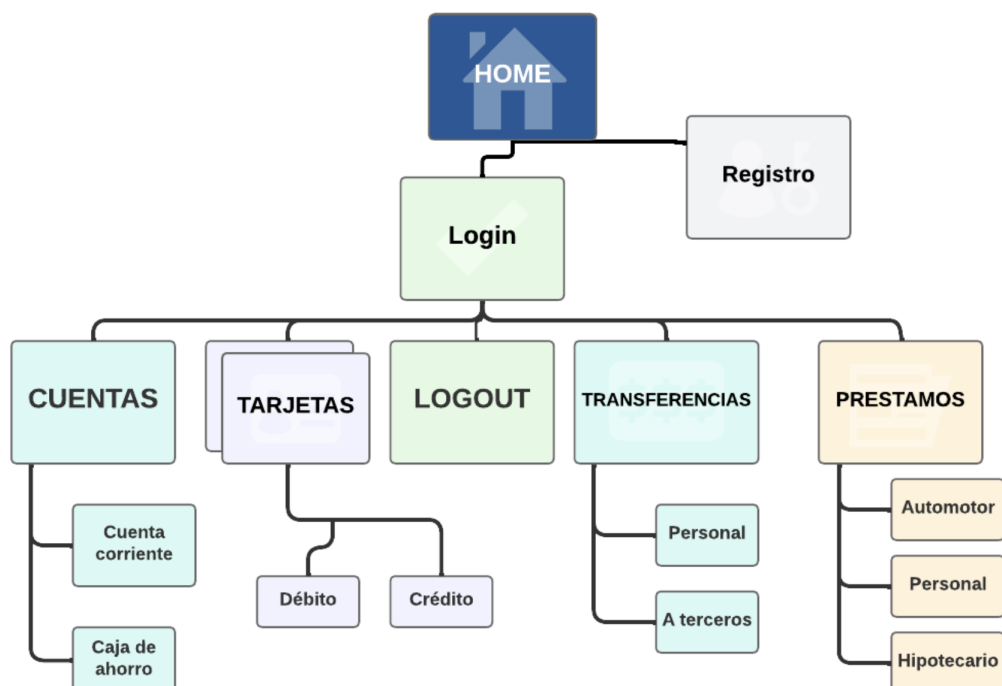
El producto



Épica

“Nos gustaría que construyan una aplicación que permita realizar operaciones bancarias desde cualquier dispositivo, pc, teléfono, tablet a través de una aplicación web. Si la aplicación web tiene éxito podríamos pensar en crear una aplicación móvil que utiliza los mismos servicios.”

Mapa del sitio



Como puede observarse en el mapa del sitio que hemos desarrollado, los flujos principales del Home Banking son: Registro, Login, Cuentas, Tarjetas y Préstamos.

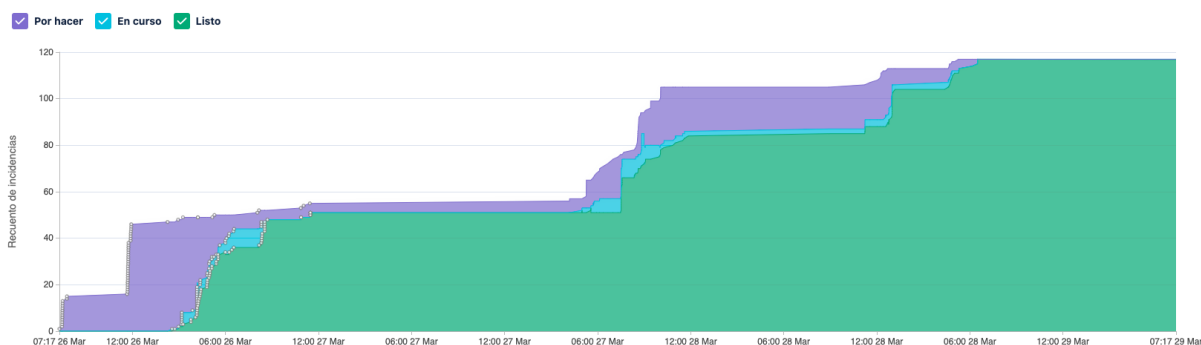
Registro y Login pueden entenderse como funcionalidades principales ya que son condición necesaria para acceder al resto de las funcionalidades que ostenta el sistema.

Por su parte, la administración, manipulación, visualización y eliminación de Cuentas y Tarjetas se comportan como flujos principales de este sistema.

Por último, la solicitud, visualización y ejecución de los diferentes tipos de Préstamos y Transferencias se consolidan como una parte fundamental del sistema, ya que hacen a la circulación del dinero dentro de la aplicación web analizada.

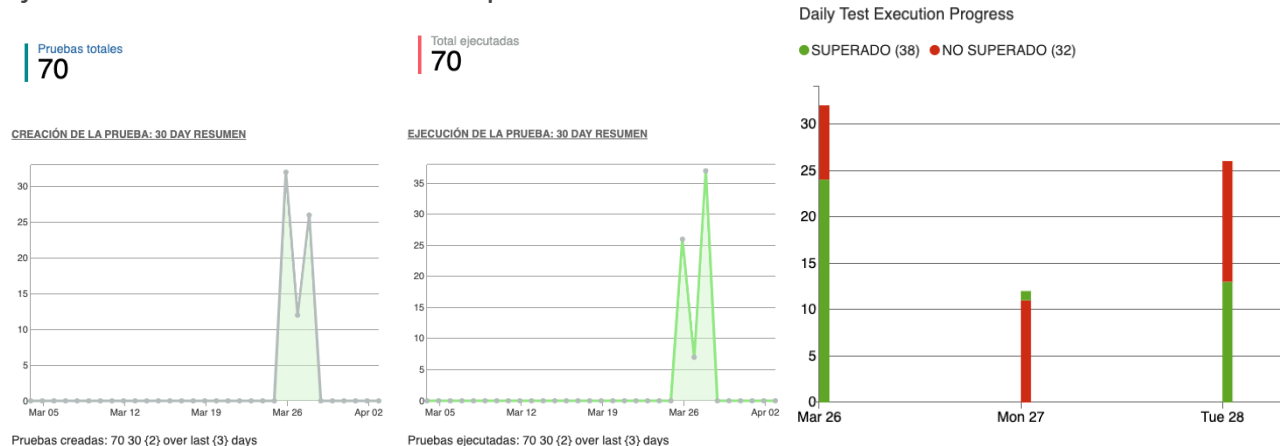
Reportes generales de Jira

Diagrama de flujo acumulado



*Se tomó el criterio de finalizar los Bugs en proceso a modo de cierre del proyecto en Jira, pero en una situación laboral esta instancia debe continuar hasta que los desarrolladores solucionen los defectos y fallos y asignen los casos para ser testeados nuevamente.

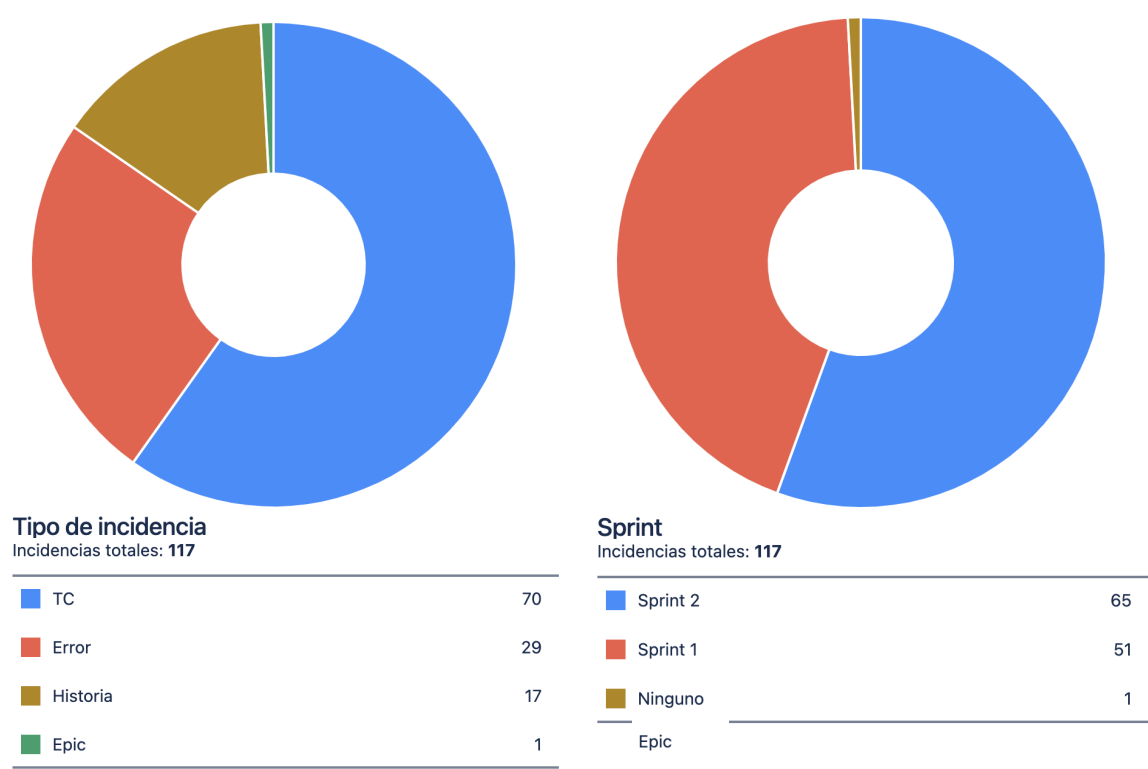
Ejecuciones totales de casos de prueba



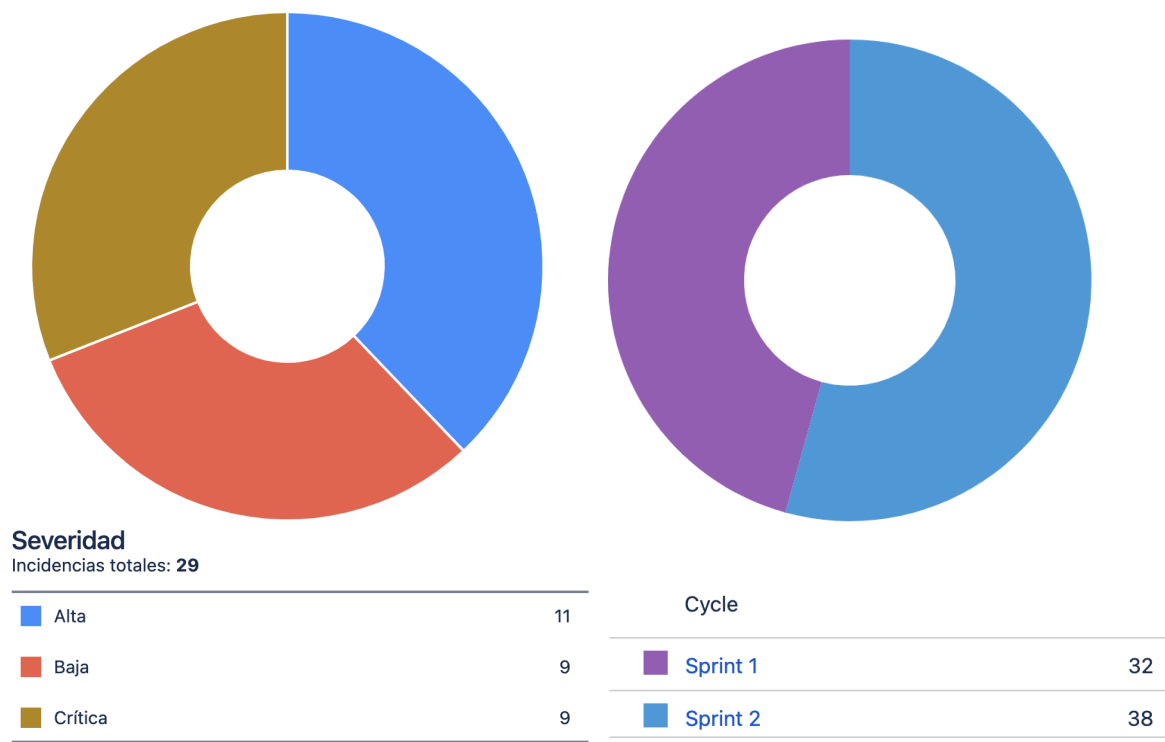
<https://fidelefflisandro.atlassian.net/jira/software/projects/MB/issues/>

Desarrollo del trabajo

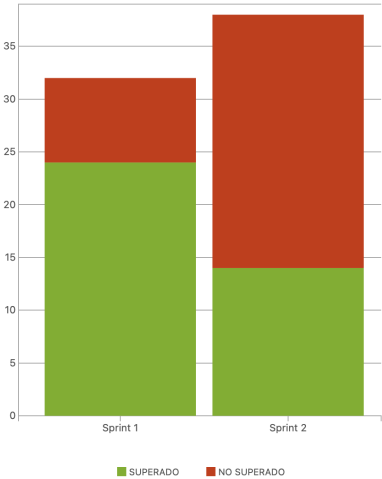
Cantidad total de incidencias



Casos de pruebas y bugs:

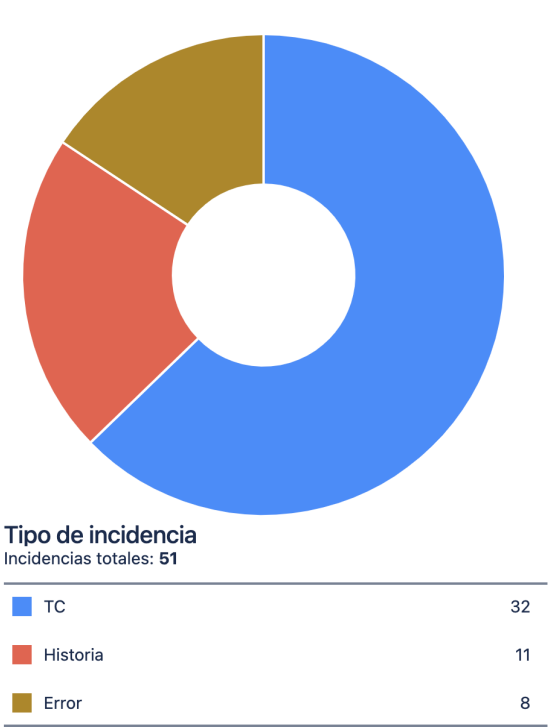


Severidad	Sprint 1	Sprint 2	T:
Alta	0	11	11
Baja	3	6	9
Crítica	5	4	9
Incidencias únicas totales:	8	21	29



Sprint 1

Durante el sprint 1 el equipo se centró en la implementación de las funcionalidades básicas de la plataforma. El objetivo principal del sprint.asegurar que todas las funcionalidades básicas se implementaron correctamente y se probaron exhaustivamente antes de continuar con las funcionalidades más avanzadas



Este gráfico puede ayudar a visualizar la relación entre las Historias de usuario y los tipos de incidencia que se encontraron durante el sprint 1.

Historias de usuario:

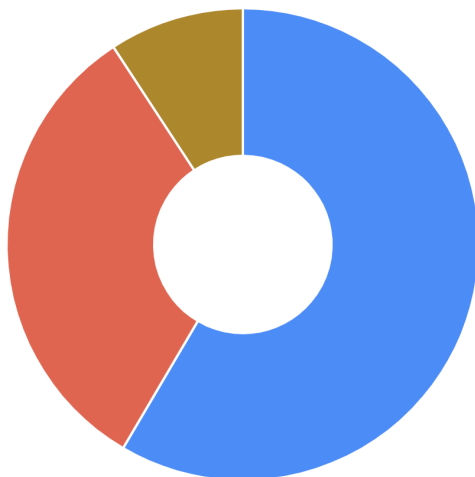
 MB-2	 MB-5	 MB-8	 MB-11
Cuentas asociadas	Tarjetas asociadas	Cerrar sesión	Realizar transacción
 MB-3	 MB-6	 MB-9	 MB-12
Entrar a una cuenta	Iniciar sesión	Crear cuentas	Solicitar préstamo
 MB-4	 MB-7	 MB-10	
Préstamos solicitados	Registrarse	Crear tarjetas	

Severidad de los bugs:

Severidad	Sprint 1
Alta	0
Baja	3
Crítica	5
Incidencias únicas totales:	8

Sprint 2

En el sprint 2 nos enfocamos en continuar mejorando la plataforma mediante la implementación de nuevas funcionalidades y mejoras en la experiencia del usuario. El objetivo principal fue asegurar que la plataforma sea más segura, estable y fácil de usar para los clientes. A continuación se presentan los detalles del trabajo realizado durante el sprint 2, así como los resultados de las pruebas realizadas y las recomendaciones.



Tipo de incidencia
Incidencias totales: 65

 TC	38
 Error	21
 Historia	6

Este gráfico puede ayudar a visualizar la relación entre las Historias de usuario y los tipos de incidencia que se encontraron durante el sprint 2. Además puede observarse en relación al sprint 1, tienen casi la misma cantidad de TC pero en este caso podemos observar casi el doble de bugs

Historias de usuario:

 MB-13	 MB-79
Comprobación de seguridad	Verificar peticiones a la API
 MB-14	 MB-87
Accesibilidad de la página	Pruebas automatizadas
 MB-15	 MB-109
Experiencia de usuario de la App	Conectividad en otros países

Severidad de los bugs:

Severidad	Sprint 2
Alta	11
Baja	6
Crítica	4
Incidencias únicas totales:	21

Seguridad

Para lo que respecta a este tema, podemos decir que es el pilar más importante de la aplicación, al considerar que un banco lo primero que debe ofrecer (históricamente) es la seguridad del dinero y transmitir confianza a sus clientes.

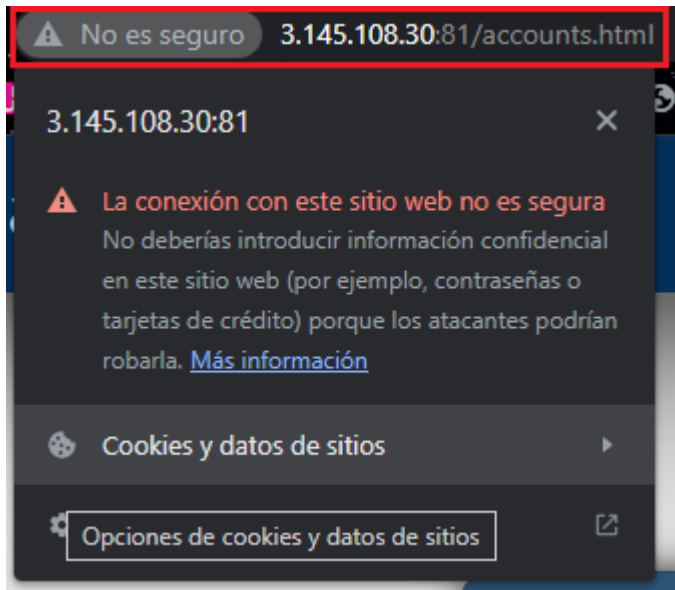
Los casos de prueba propuestos son cinco principalmente y van desde el Test Case 101 al 105:

- Validar dominio de página
- Verificar confirmación de contraseña
- Verificar alerta antes de eliminar cuenta
- Verificar longitud mínima de contraseña
- Verificar mensaje de confirmación al transferir.

Cuatro de estas cinco pruebas ejecutadas han derivado en una falla de seguridad, enumeradas como MB:107, 108, 112 y 113, lo cual corresponde a reportes de bugs, descritos de la siguiente manera:

- Falla de seguridad en dominio
- Falla de seguridad en registro
- Falla de seguridad en área cuentas
- Falla de seguridad en password

Falla de seguridad en dominio: Se refiere a que el dominio no posee certificado de seguridad (SSL) por lo cual es vulnerable ante robo de datos.



Falla de seguridad en registro: No se solicita confirmación de contraseña a la hora de registrarse

Falla de seguridad en área cuentas: A la hora de eliminar una cuenta no hay ningún tipo de alerta o notificación previa de confirmación. Esto es crítico, y hay que mencionar que en otro caso de prueba se detectó que se pueden eliminar cuentas con dinero dentro, lo cual agrava más a este tipo de error.

Falla de seguridad en password: A la hora de realizar el registro, el campo password no solicita un mínimo de caracteres. Para el caso de prueba se introdujo un carácter y se ha notado esta falla.

A screenshot of a web form for signing up. The form has a dark blue background with a faint keyboard pattern. At the top, there are two tabs: "SIGN IN" and "SIGN UP", with "SIGN UP" being the active tab. Below the tabs are four input fields: "FIRST NAME" with the value "Pepito", "LAST NAME" with the value "Perez", "EMAIL ADDRESS" with the value "pepito@gmail.com", and "PASSWORD" which contains a single dot. The "PASSWORD" field is highlighted with a red rectangular border. At the bottom of the form is a large blue button labeled "SIGN IN".

API Testing

Introducción para el testing de apis

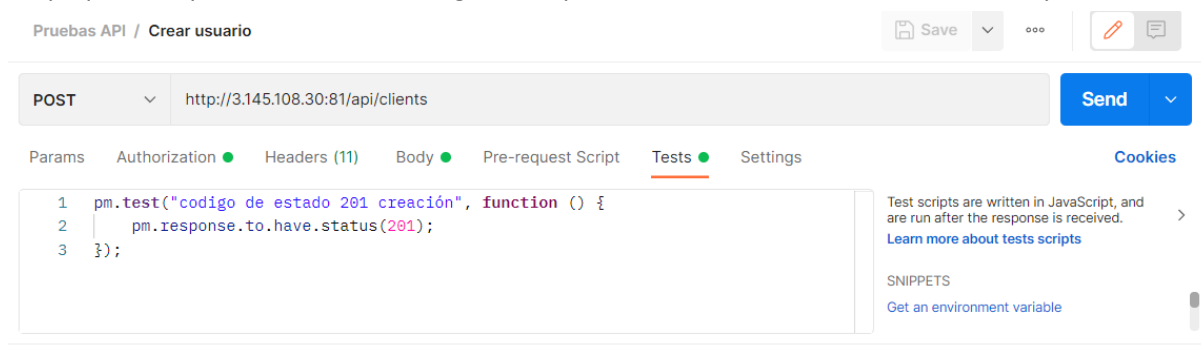
En este caso, se trata de una evaluación de las APIs de una aplicación de Homebanking de un banco ficticio llamado Mindhubbrothers mediante la herramienta Postman.

Objetivos

El objetivo de la actividad, es evaluar la calidad de las APIs en cuanto a su rendimiento, escalabilidad y capacidad para manejar un alto volumen de peticiones.

Automatización con Snippets

Para la utilización de Snippets, se tomaron pequeñas partes de código para realizar la automatización del paquete de peticiones, con el código correspondiente al estado de devolución de la petición.



Resultados obtenidos en postman

Para este caso analizamos ocho (8) endpoints para trece (13) peticiones, con los métodos GET y POST.

Tiempo de respuesta

Se realiza el tiempo de respuesta total de los paquetes durante 4 días, ininterrumpidamente. Con los siguientes resultados:

- El tiempo de respuesta del total de paquetes de peticiones es de 1,27 seg.
- Tiempo de respuesta por petición: 97ms

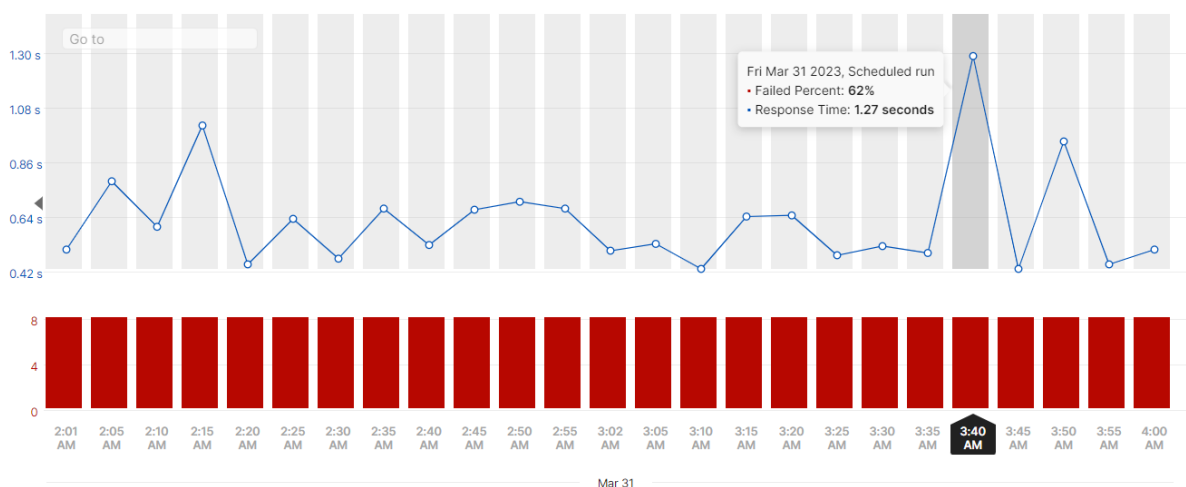
Test Results		Console Log	
✖	✖	POST Crear usuario http://3.145.108.30:81/api/clients	403 Forbidden 132 ms 19 B
✔	✖	FAIL codigo de estado 201 creación	
✖	✔	POST Iniciar Sesión http://3.145.108.30:81/api/login	200 OK 222 ms
	✔	PASS Status code is 200	
	✔	GET Obtener lista de usuarios http://3.145.108.30:81/api/clients	200 OK 98 ms 119.0 kB
	✔	PASS Código de Estado 200	
	✔	GET Obtener datos de usuario http://3.145.108.30:81/api/clients/1	200 OK 52 ms 735 B
	✔	PASS Código de Estado 200	
	✔	GET Obtener datos del usuario actual http://3.145.108.30:81/api/clients/current	200 OK 88 ms 249 B
	✔	PASS Código de Estado 200	
	✖	POST Crear cuenta nueva http://3.145.108.30:81/api/clients/current/accounts	400 Bad Request 66 ms 119 B
	✖	FAIL codigo de estado 201 creación	
	✖	GET Obtener lista de cuentas http://3.145.108.30:81/api/clients/current/accounts	405 Method Not Allowed 52 ms 126 B
	✖	FAIL Código de Estado 200	
	✖	POST Crear tarjeta http://3.145.108.30:81/api/clients/current/cards	400 Bad Request 23 ms 116 B
	✖	FAIL codigo de estado 201 creación	
	✖	GET Obtener lista de tarjetas http://3.145.108.30:81/api/clients/current/cards	405 Method Not Allowed 117 ms 123 B
	✖	FAIL Código de Estado 200	
	✖	POST Crear transacción http://3.145.108.30:81/api/transactions	400 Bad Request 177 ms 107 B
	✖	FAIL codigo de estado 201 creación	
	✖	POST Crear préstamo nuevo http://3.145.108.30:81/api/loans	404 Not Found 49 ms 98 B
	✖	GET Obtener lista de préstamos http://3.145.108.30:81/api/loans	404 Not Found 93 ms 98 B
	✖	FAIL Código de Estado 200	
	✔	POST Cerrar Sesión http://3.145.108.30:81/api/logout	200 OK 103 ms
	✔	PASS Código de Estado 200	

Escalabilidad:

El sistema permite sostener el sistema, con cantidades moderadas de peticiones cada 5 minutos de manera ininterrumpida. Por lo que el sistema no sufre cambios ni caídas bruscas durante ese lapso de tiempo.

Errores y fallos:

Con un 62% de promedio de errores obtenidos de las apis, las cuales no cumplen con lo solicitado por las peticiones.



Conclusiones y recomendaciones de APIs

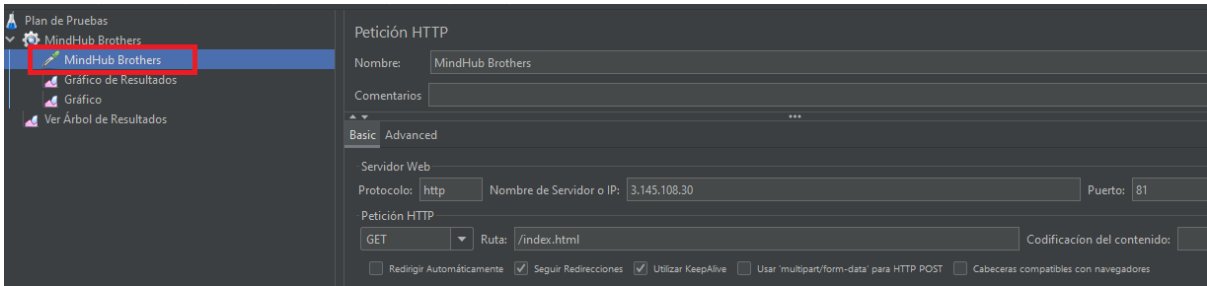
En general, se considera que una aplicación de Home Banking debe responder en menos de 2 segundos para proporcionar una buena experiencia al usuario. Por lo que se puede decir que, en tiempos de respuesta del servidor la aplicación cumple con el tiempo admisible para una aplicación de homebanking. Pero, la API no funciona correctamente ya que devuelve errores en las peticiones por lo que se deben corregir con el equipo de desarrollo pertinente.

Performance de API Testing

Para poder medir el performance de la página se utilizó JMeter.

Solo se realizaron pruebas con el método GET en cuanto al ingreso a la página, titulada cómo “index”.

Estos son los parámetros se configuraron las pruebas:



Protocolo: Http
Nombre de Servidor o IP: 3.145.108.30
Puerto: 81

Número de hilos: 200000
Periodo de subida: 0
Contador del bucle: 1

Gráfico											
Nombre: Gráfico											
Comentarios											
Escribir todos los datos a Archivo											
Nombre de archivo <input type="text"/> Navegar... Log/Mostrar sólo: <input type="checkbox"/> Escribir en Log Sólo Errores <input type="checkbox"/> Éxitos <input type="checkbox"/> Co											
Etiqueta	# Muestras	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimien...	Kb/sec
MindHub ...	69516	17934	21264	22002	66096	99131	0	101380	95,11%	685,4/sec	1755,09
Total	69516	17934	21264	22002	66096	99131	0	101380	95,11%	685,4/sec	1755,09

Automation

Para las pruebas de automatización se tomaron algunos de los flujos principales, exceptuando el registro. Las mismas se llevaron a cabo en la extensión de Selenium dentro del navegador.

Los casos de prueba propuestos incluían lo siguiente:

1. Verificación de Inicio de sesión
2. Verificar creación de caja de ahorro
3. Verificación creación de Cuenta Corriente
4. Verificar solicitud de préstamo Personal

Tres de las cuatro pruebas, se superaron con éxito, mientras que la verificación del préstamo personal falló en la última etapa de ejecución (dando así un 94,73% de completado)

1-

Project: MindHub Brothers*

Tests ▾	+	▶▶▶	🔍	🕒	▼
Search tests...					
Run current test Ctrl+R index.html					
✓ Inicio_Sesion*	Command	Target	Value		
	1 ✓ open	http://3.145.108.30:81/index.html			
	2 ✓ set window size	1007x720			
	3 ✓ click	id=user			
	4 ✓ type	id=user	admin		
	5 ✓ click	id=password			
	6 ✓ type	id=password	admin		
	7 ✓ click	css=group:nth-child(3) > .button			

2-

Project: MindHub Brothers*

Tests ▾	+	▶▶▶	🔍	🕒	▼
Search tests... Q					
http://3.145.108.30:81/index.html					
✓ Caja_Ahorro*	Command	Target	Value		
✓ Inicio_Sesion*	1 ✓ open	http://3.145.108.30:81/index.html			
	2 ✓ set window size	1007x720			
	3 ✓ click	id=user			
	4 ✓ type	id=user	admin		
	5 ✓ click	id=password			
	6 ✓ type	id=password	admin		
	7 ✓ click	css=group:nth-child(3) > .button			
	8 ✓ click	css=createAccBtn			
	9 ✓ click	css=btn:nth-child(2)			

3-

Selenium IDE - MindHub Brothers*

Project: MindHub Brothers*

Tests ▾ +

Search tests... 🔍

http://3.145.108.30:81/index.html

	Command	Target	Value
1	✓ open	http://3.145.108.30:81/index.html	
2	✓ set window size	1007x720	
3	✓ click	id=user	
4	✓ type	id=user	admin
5	✓ type	id=password	admin
6	✓ click	css=group:nth-child(3) > .button	
7	✓ click	css= .createAccBtn	
8	✓ click	css= .btn:nth-child(3)	

4- El problema observado es que el botón “Apply” queda deshabilitado.

Loans

Loan Type

Personal ▾

Max Amount Loan \$100000

Fee %15

Payments

6 ▾

Account Destiny

VIN-16955695 ▾

Amount

15444

Apply

Project: MindHub Brothers*

Executing ▾

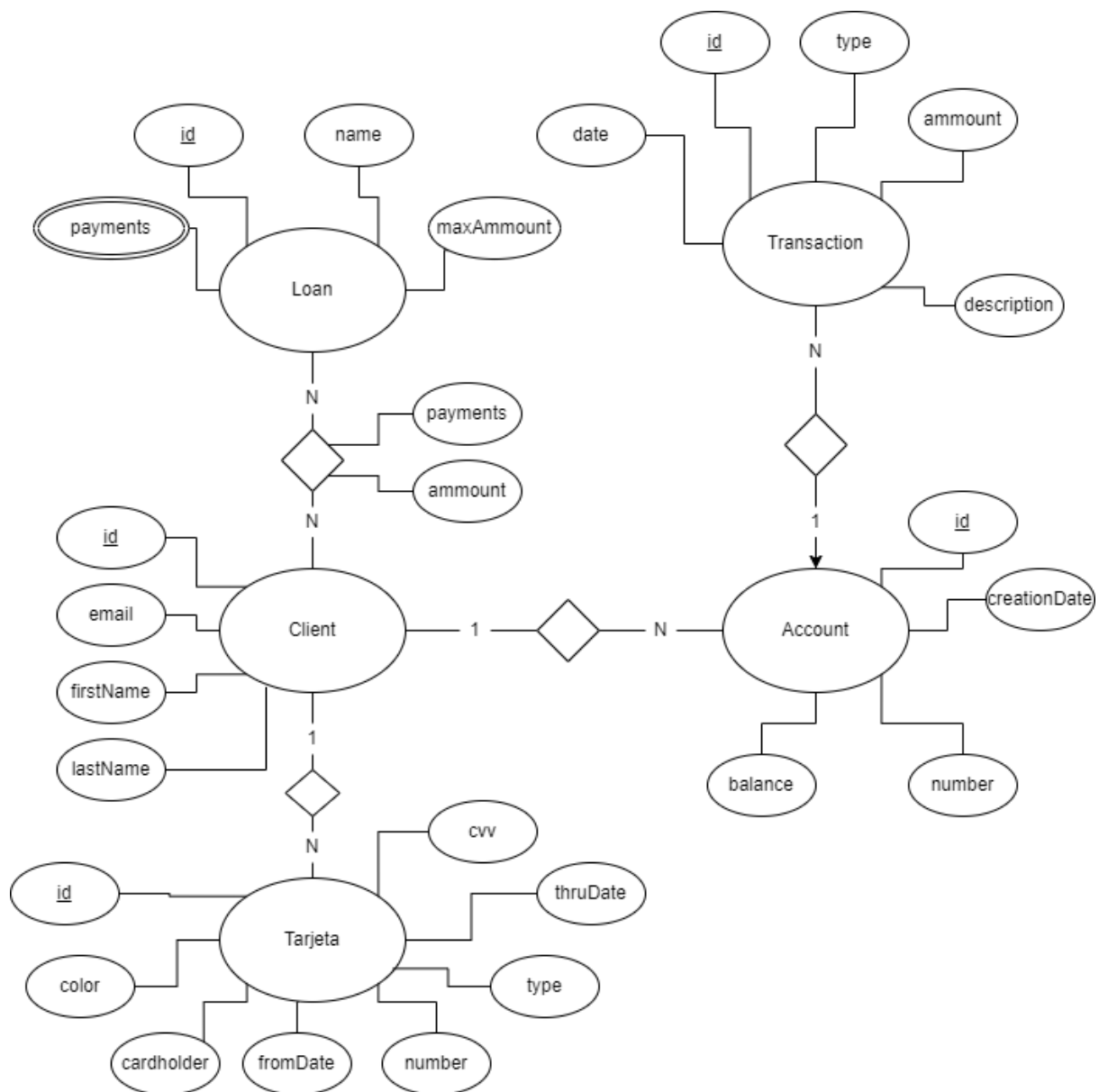
Run current test Ctrl+R

index.html

	Command	Target	Value
11	✓ select	id=payments	label=6
12	✓ click	id=account	
13	✓ select	id=account	label=VIN-16955695
14	✓ click	id=amount	
15	✓ type	id=amount	15500
16	✓ click	css= .btn:nth-child(12)	
17	✓ type	id=amount	15444
18	✓ click	css= .btn:nth-child(12)	
19	✗ click	css= .btn:nth-child(2)	

Bases de Datos

Diagrama ER original



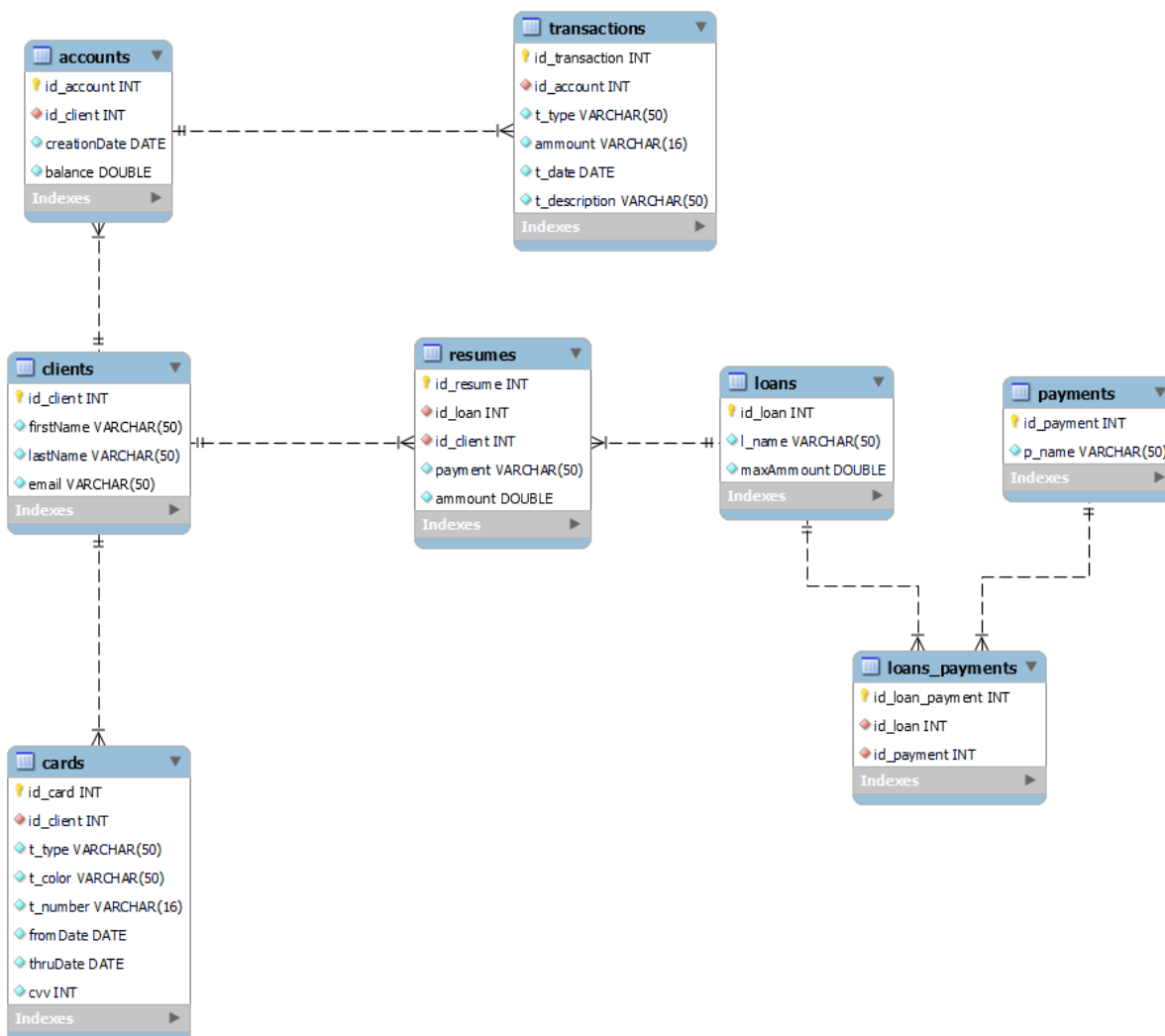
Defectos

1. Cardinalidad N:N : esta clase de relaciones no existe como tal en una base de datos.
2. Valores multivaluados: los campos de este tipo guardan varios datos distintos para una misma entidad lo que genera redundancia a la hora de consultar datos. Esto puede generar una demora en el tiempo de respuesta de las transacciones que se puedan generar.

Soluciones

1. Cardinalidad N:N : para solucionar este defecto se debe generar una nueva tabla que contenga las llaves primarias de las tablas de la relación como llaves foráneas. Esto hará posible el acceso a los datos desde ambas partes de la relación evitando la redundancia de los datos.
2. Valores multivaluados: la solución es generar una nueva tabla que contenga los datos de cada valor del campo y su id, y relacionarla con la tabla a la que estaba vinculada.

BBDD Propuesta



Cambios

Analizando el diagrama ER original y los posibles defectos de su implementación, proponemos nuestra BBDD implementada en MySQL donde aplicamos las soluciones vistas anteriormente:

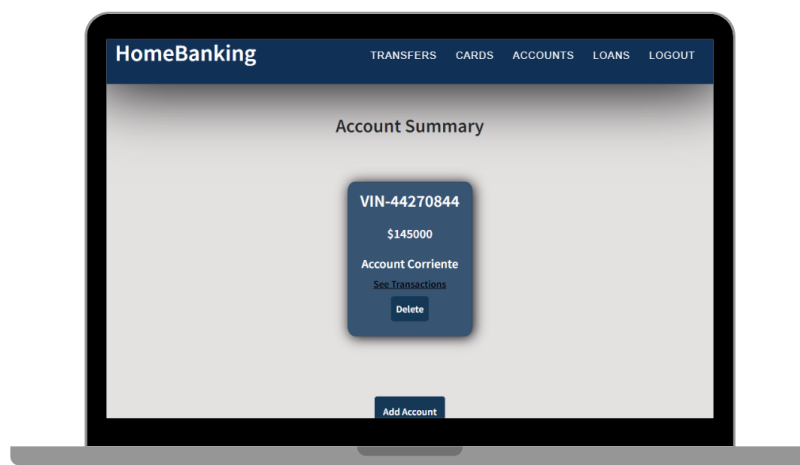
1. Para las relaciones N:N : proponemos la creación de las tablas “resumes”. La misma contendrá la llave primaria de las respectivas tablas de la relación como llave foránea, lo que facilitará el acceso a los datos y evitará la redundancia de los mismos.

2. Para los valores multivaluados: proponemos la creación de la tabla “loans_payments”. Esta guardará la relación N:N que resultará de la descomposición del campo multivaluado. Por otro lado, los distintos valores del campo multivaluado “payments” se verán reflejados en una nueva tabla “payments” con su propio ID.

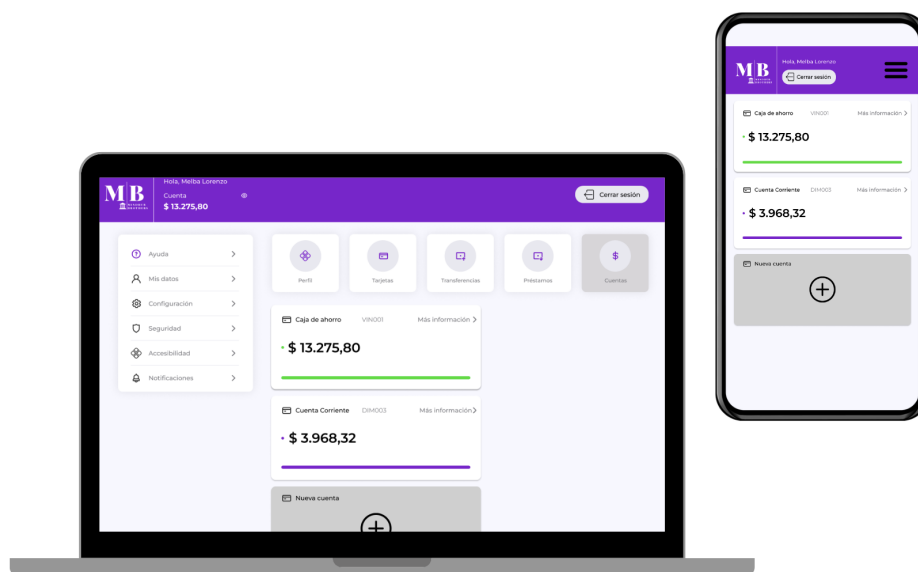
Estos cambios propuestos evitarán la redundancia de los datos, la confusión y facilitarán el acceso a los mismos, además de reducir los tiempos de respuesta de las transacciones.

UX/UI

Versión actual



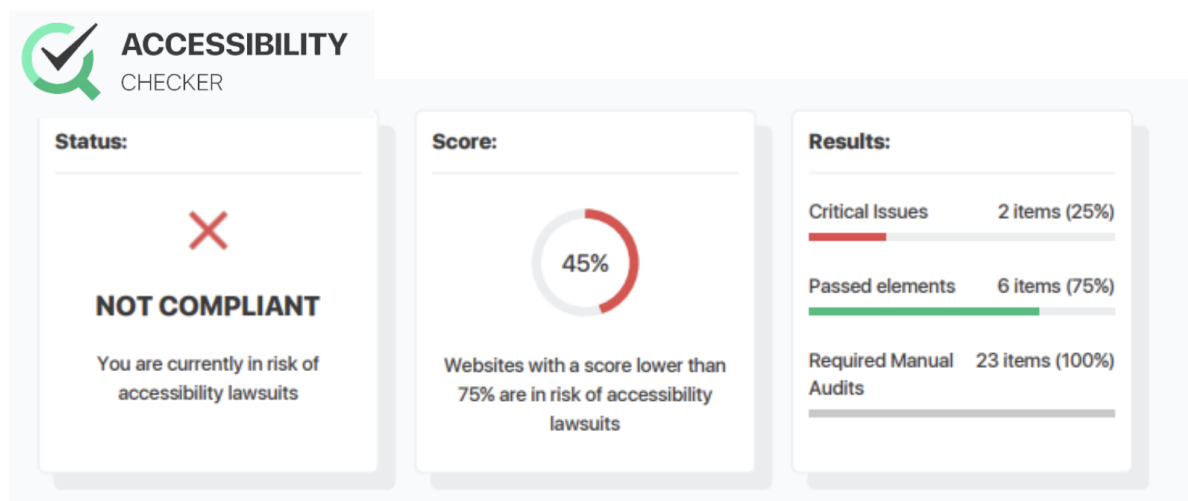
Propuesta de Mockup



A modo de mejora, se propone intentar conseguir una interfaz más amigable e intuitiva para el usuario, tomando como referencia la oferta de otras entidades que se dedican al mismo rubro y poseen aplicaciones web similares a la desarrollada.

Además, se sugiere la adición de menús y elementos que permitan al usuario la posibilidad de personalizar su experiencia al utilizar el programa. Como puede observarse en el mockup diseñado, sería interesante agregar opciones tales como: Perfil, Configuraciones, Accesibilidad, Seguridad, Ayuda, Mis datos y configuración de Notificaciones.

Accesibilidad



Hemos realizado pruebas en Accessibility Checker y en la herramienta Talkback para validar la accesibilidad de la página y recibimos información un tanto desalentadora en ese sentido. Se observaron elementos que no deberían encontrarse en ese sitio para una correcta utilización de herramientas de accesibilidad y se evidenció un contraste muy bajo en el texto de la aplicación -lo que limita la visibilidad por parte de personas con discapacidades visuales-.

Conclusión

Luego de haber realizado un exhaustivo análisis sobre el producto estudiado, podemos inferir que las funcionalidades principales del mismo se encuentran conservadas y la aplicación web cumple con la finalidad para la cual fue desarrollada.

Como se desprende del informe presentado, un 75% de las pruebas realizadas en el Sprint 1 han sido superadas correctamente. Por su parte, únicamente un 44,74% de las pruebas llevadas adelante en el Sprint 2 han resultado satisfactorias. En función de lo mencionado anteriormente, llegamos a la conclusión de que los mayores defectos y fallas se encuentran en lo que atañe a las cuestiones no funcionales del programa.

La seguridad del sistema se encuentra comprometida, ya que el acceso a datos sensibles de los usuarios es irrestricto para terceros, se carece de certificado de seguridad en la página (SSL), se encuentran vulnerabilidades en el registro de los datos al crear una cuenta y la base de datos está mal implementada.

En cuanto a la accesibilidad, se evidencian problemas serios y limitaciones para que las personas con discapacidad puedan hacer uso del programa, ya que la aplicación web no cumple con los estándares de la aplicación Accessibility y no reconoce elementos indispensables para el funcionamiento de herramientas tales como Talkback.

Haciendo referencia a la experiencia de los usuarios, podemos observar que, en relación con otros Home Bankings de entidades bancarias, no se respetaron muchos elementos de la página que contribuyen a satisfacer las necesidades de usabilidad de las personas que interactúan con el programa. Por destacar las más importantes, se evidencia la falta de menús desplegables con opciones de Ayuda, Configuraciones, Perfil, etc.

A modo de conclusión general, resulta menester comunicar que el programa no se encuentra en condiciones de salir a producción si no se analizan y resuelven las problemáticas planteadas en el presente informe.