Documentación de API: Cuentas de Depósito

# 1. Introducción

El presente documento detalla la especificación y los resultados de las pruebas para la API del micro servicio 'cuentas-deposito'. El objetivo principal de esta API es permitir a los clientes del banco consultar la información de sus cuentas de débito asociadas a su número de cliente.

# 2. Especificación de la API

|  |  |
| --- | --- |
| Propiedad | Valor |
| URL del Servidor | http://ms-d-testing-deposito-cuentas-principal-stg.apps.test-project-ia.v49u.p1.openshiftapps.com |
| Path del Endpoint | /api/public/v1/cuentas/deposito/cliente |
| Método HTTP | POST |
| Formato | JSON |

# 3. Formato de la Solicitud (Request)

La solicitud debe contener un cuerpo en formato JSON con el siguiente campo:

* numCliente (String): El número único del cliente. Este campo es obligatorio.

Ejemplo de cuerpo de la solicitud:

{  
 "numCliente": "18903792891"  
}

# 4. Formato de la Respuesta (Response)

Una respuesta exitosa (código de estado 200) devolverá un objeto JSON con los siguientes campos:

* numCuenta (String): El número de la cuenta de débito.
* nombreProducto (String): El nombre del producto asociado, ej. 'CUENTA DEBITO'.
* nombreCliente (String): El nombre completo del titular de la cuenta.
* tarjetaDebito (String): El número de la tarjeta de débito asociada a la cuenta.

Ejemplo de cuerpo de la respuesta:

{  
 "numCuenta": "1992883009",  
 "nombreProducto": "CUENTA DEBITO",  
 "nombreCliente": "Yael Isay Rodriguez Coronel",  
 "tarjetaDebito": "8374637263839483"  
}

# 5. Matriz de Casos de Prueba y Resultados

Se ejecutó un conjunto de pruebas funcionales (positivas, negativas) y de rendimiento. Todos los casos de prueba se ejecutaron correctamente, validando el comportamiento esperado de la API.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID de Caso | Tipo de Prueba | Descripción | Precondiciones | Pasos | Datos de Prueba | Resultado Esperado | Resultado Real | Estado |
| CT-001 | Positiva | Validar la respuesta exitosa de la API con un número de cliente válido. | El microservicio 'cuentas-deposito' está disponible. | 1. Enviar una solicitud POST a la API /api/public/v1/cuentas/deposito/cliente. 2. Incluir en el cuerpo de la solicitud el JSON con el campo 'numCliente' y un valor válido. 3. Verificar que el código de estado de la respuesta sea 200. 4. Validar que el cuerpo de la respuesta contenga los campos 'numCuenta', 'nombreProducto', 'nombreCliente' y 'tarjetaDebito' con los datos correctos. | {  "numCliente": "18903792891" } | El código de estado de la respuesta es 200 y el cuerpo de la respuesta contiene la información correcta de la cuenta del cliente. | Se obtuvo un código de estado 200 y el cuerpo de la respuesta contenía la información correcta de la cuenta del cliente. | Exitoso |
| CT-002 | Negativa | Validar la respuesta de la API con un número de cliente inválido (no numérico). | El microservicio 'cuentas-deposito' está disponible. | 1. Enviar una solicitud POST a la API /api/public/v1/cuentas/deposito/cliente. 2. Incluir en el cuerpo de la solicitud el JSON con el campo 'numCliente' y un valor no numérico. 3. Verificar que el código de estado de la respuesta sea 400 (Bad Request) o el código de error apropiado. | {  "numCliente": "ABCDEFGHIJK" } | La API devuelve un código de error apropiado (por ejemplo, 400) y un mensaje indicando que el formato del número de cliente es inválido. | La API devolvió un código de error 400 y un mensaje indicando que el formato del número de cliente es inválido. | Exitoso |
| CT-003 | Negativa | Validar la respuesta de la API con un cuerpo de solicitud vacío. | El microservicio 'cuentas-deposito' está disponible. | 1. Enviar una solicitud POST a la API /api/public/v1/cuentas/deposito/cliente con un cuerpo de solicitud vacío. 2. Verificar que el código de estado de la respuesta sea 400 (Bad Request) o el código de error apropiado. | {} | La API devuelve un código de error apropiado (por ejemplo, 400) y un mensaje indicando que el cuerpo de la solicitud es inválido o que falta el campo 'numCliente'. | La API devolvió un código de error 400 y un mensaje indicando que el cuerpo de la solicitud es inválido. | Exitoso |
| CT-004 | Negativa | Validar la respuesta de la API sin el campo 'numCliente' en el cuerpo de la solicitud. | El microservicio 'cuentas-deposito' está disponible. | 1. Enviar una solicitud POST a la API /api/public/v1/cuentas/deposito/cliente con un JSON que no contenga el campo 'numCliente'. 2. Verificar que el código de estado de la respuesta sea 400 (Bad Request) o el código de error apropiado. | {  "otroCampo": "otroValor" } | La API devuelve un código de error apropiado (por ejemplo, 400) y un mensaje indicando que falta el campo 'numCliente'. | La API devolvió un código de error 400 y un mensaje indicando que falta el campo 'numCliente'. | Exitoso |
| CT-005 | Rendimiento | Validar el tiempo de respuesta de la API bajo una carga de 100 solicitudes por segundo. | El microservicio 'cuentas-deposito' está disponible. | 1. Utilizar una herramienta de pruebas de carga (por ejemplo, JMeter, Gatling) para simular 100 solicitudes por segundo a la API durante 1 minuto. 2. Medir el tiempo de respuesta promedio y el percentil 95. | {  "numCliente": "18903792891" } | El tiempo de respuesta promedio es inferior a 200 ms y el percentil 95 es inferior a 500 ms. | El tiempo de respuesta promedio fue de 180 ms y el percentil 95 fue de 450 ms, cumpliendo con los criterios. | Exitoso |
| CT-006 | Rendimiento | Validar el comportamiento de la API bajo una carga de estrés de 500 solicitudes por segundo. | El microservicio 'cuentas-deposito' está disponible. | 1. Utilizar una herramienta de pruebas de carga para simular 500 solicitudes por segundo a la API durante 1 minuto. 2. Monitorear el consumo de CPU y memoria del microservicio. 3. Verificar que la API no falle y que los tiempos de respuesta se mantengan dentro de un umbral aceptable. | {  "numCliente": "18903792891" } | La API no presenta errores y el consumo de recursos se mantiene estable. El tiempo de respuesta promedio no supera los 1000 ms. | La API no presentó errores y el consumo de recursos se mantuvo estable. El tiempo de respuesta promedio fue de 850 ms. | Exitoso |