

## Transacciones y Parámetros

### **MCWNT100 - Creación de Cuenta**

Descripción: Transacción para la creación de una cuenta y su titular.

#### **Parámetros de entrada:**

- holder.name (String, obligatorio, longitud: 100)
- holder.rfc (String, obligatorio, longitud: 13)
- holder.account.accountNumber (Long, obligatorio, longitud: 10)
- holder.account.balance (Double, obligatorio, longitud: 11)
- holder.account.accountCard (Long, obligatorio, longitud: 16)
- holder.lastName (String, opcional, longitud: 100)
- holder.age (Long, opcional, longitud: 3)
- holder.curp (String, opcional, longitud: 18)

#### **Parámetros de salida:**

- holder.name (String, opcional, longitud: 100)
- holder.age (Long, opcional, longitud: 3)
- holder.lastName (String, opcional, longitud: 100)
- holder.curp (String, opcional, longitud: 18)
- holder.rfc (String, opcional, longitud: 13)
- holder.account.accountNumber (Long, opcional, longitud: 10)
- holder.account.accountNip (Long, opcional, longitud: 4)
- holder.account.balance (Double, opcional, longitud: 11)
- holder.account.accountCard (Long, opcional, longitud: 16)
- holder.account.status (Boolean, opcional)
- holder.holderType (String, opcional, longitud: 50)

## **MCWNT101 - Consulta de Cuenta**

Descripción: Transacción que obtiene los datos de un titular a partir del número de cuenta y NIP.

### **Parámetros de entrada:**

- account.accountNumber (Long, obligatorio, longitud: 10)
- account.accountNip (Long, obligatorio, longitud: 4)

### **Parámetros de salida:**

- holder.name (String, opcional, longitud: 100)
- holder.age (Long, opcional, longitud: 3)
- holder.lastName (String, opcional, longitud: 100)
- holder.curp (String, opcional, longitud: 18)
- holder.rfc (String, opcional, longitud: 13)
- holder.account.accountNumber (Long, opcional, longitud: 10)
- holder.account.accountNip (Long, opcional, longitud: 4)
- holder.account.balance (Double, opcional, longitud: 11)
- holder.account.accountCard (Long, opcional, longitud: 16)
- holder.account.status (Boolean, opcional)
- holder.holderType (String, opcional, longitud: 50)

## **MCWNT102 - Consulta de Todas las Cuentas**

Descripción: Transacción que recupera todas las cuentas activas / inactivas.

### **Parámetros de entrada:**

- Ninguno

### **Parámetros de salida:**

- List<accounts> Obligatorio con objetos HolderOutDTO, incluyendo:
  - name (String, Opcional, 100)
  - age (Long, Opcional, 3)
  - lastName (String, Opcional, 100)
  - curp (String, Opcional, 18)
  - rfc (String, Opcional, 13),
  - holderType (String, Opcional, 50)
  - account.accountNumber (Long, Opcional, 10)
  - account.accountNip (Long, Opcional, 4)
  - account.balance (Double, Opcional, 11)
  - account.accountCard (Long, Opcional, 16)
  - account.status (Boolean, Opcional)

## **MCWNT103 - Actualización de Cuenta**

Descripción: Transacción para actualizar los datos de una cuenta y su titular.

### **Parámetros de entrada:**

- holder.account.accountNumber (Long, obligatorio, longitud: 10)
- holder.rfc (String, opcional, longitud: 13 para física, 12 para moral)
- holder.account.balance (Double, opcional, longitud: 11)
- holder.account.accountCard (Long, opcional, longitud: 16)
- holder.account.accountNip (Long, opcional, longitud: 4)
- holder.name (String, opcional, longitud: 100)
- holder.lastName (String, opcional, longitud: 100)
- holder.age (Long, opcional, longitud: 3)
- holder.curp (String, opcional, longitud: 18)

### **Parámetros de salida:**

- holder.name (String, obligatorio, longitud: 100)
- holder.age (Long, opcional, longitud: 3)
- holder.lastName (String, opcional, longitud: 100)
- holder.curp (String, opcional, longitud: 18)
- holder.rfc (String, obligatorio, longitud: 13)
- holder.account.accountNumber (Long, obligatorio, longitud: 10)
- holder.account.accountNip (Long, obligatorio, longitud: 4)
- holder.account.balance (Double, obligatorio, longitud: 11)
- holder.account.accountCard (Long, obligatorio, longitud: 16)
- holder.holderType (String, opcional, longitud: 50).

## **MCWNT104 - Eliminación de Cuenta**

Descripción: Transacción para eliminar lógicamente una cuenta por su número.

### **Parámetros de entrada:**

- accountNumber (Long, obligatorio, longitud: 10)

### **Parámetros de salida:**

- deleted (Boolean, obligatorio)

## **MCWNT104 (V2) - Activación de Cuenta**

Descripción: Transacción para reactivar una cuenta eliminada.

### **Parámetros de entrada:**

- accountNumber (Long, obligatorio, longitud: 10)
- accountNip (Long, obligatorio, longitud de 4)

### **Parámetros de salida:**

activated (Boolean, obligatorio)

---

## **Reglas de Negocio Detalladas para el Módulo de Cuentas - MCWN**

---

---

---

---

---

---

---

---

---

---

### **MCWNT100 - Crear Cuenta**

#### **Validaciones Generales:**

- Los campos obligatorios deben definirse con base en el modelo de datos de la base de datos (BD).
- No se debe asumir que ya existe una estructura predeterminada: los desarrolladores deberán construir la estructura de entrada en función de los campos obligatorios definidos en la BD.
- Excepciones a validar directamente desde el flujo de backend:
  - clientType: se obtiene exclusivamente del header client-document.
  - account.accountNip: **no se recibe**; se genera automáticamente como un número aleatorio de 4 dígitos entre 1000 y 9999.
  - account.status e id: no se requieren como parte de la entrada, ya que son campos internos o calculados.
- Si falta algún campo obligatorio (según definición del modelo): MCWN01415034.

#### **Validación de tipo de persona (obtenido desde header client-document):**

- El valor debe ser "0" (persona física) o "1" (persona moral).
  - Si es diferente: MCWN01415030.

#### **Si tipo es Persona Física (clientType == 0):**

- Campos requeridos: name, lastName, age, curp, rfc.
  - Si falta alguno: MCWN01415034.
- Validación de formato:

- curp debe contener exactamente 18 caracteres: MCWN01415031.
- rfc debe contener exactamente 13 caracteres: MCWN01415032.

**Si tipo es Persona Moral (clientType == 1):**

- Campo requerido: rfc.
  - Si no se recibe: MCWN01415034.
- Validación de formato:
  - rfc debe contener exactamente 12 caracteres: MCWN01415033.

**Validación de Balance:**

- Se debe consultar el valor máximo permitido desde la propiedad parametrizada MAX.BALANCE (por ejemplo, 9000).
- Si el valor de balance supera el límite: MCWN01415048.

**Validaciones de existencia:**

- Antes de crear la cuenta, se debe verificar si ya existe:
  - Una cuenta activa con el mismo número: MCWN01415038.
  - Una cuenta activa con la misma tarjeta: MCWN01415039.
  - Una cuenta inactiva con el mismo número: MCWN01415042.
  - Una cuenta inactiva con la misma tarjeta: MCWN01415043.

**Procesamiento:**

- Se genera un NIP aleatorio de 4 dígitos dentro del rango 1000 a 9999 utilizando una función de tipo Math.random().
- Se realiza la inserción en la base de datos.
  - Si la operación de inserción falla (no se inserta ninguna fila):  
MCWN01415040.
- Una vez insertada la cuenta, se debe recuperar la cuenta recién creada desde la BD y retornar esa información, **no se debe retornar la información recibida en la entrada.**
- Se calcula el valor legible holderType a partir del clientType:

- Si clientType == 0 → holderType = "FISICO".
- Si clientType == 1 → holderType = "MORAL".

**Nota:** El equipo debe analizar si esta transacción requiere rollback en caso de error. Esta decisión dependerá de si otros procesos posteriores dependen del éxito de la creación de la cuenta.

---

## **MCWNT101 - Consulta de Cuenta por Número y NIP**

### **Validaciones Generales:**

- 1. Se debe recibir accountNumber y accountNip. Ambos son obligatorios.**
  - Si alguno falta: MCWN01415034

### **Validaciones de existencia:**

- 2. Se debe buscar la cuenta con el número proporcionado.**
  - Si no existe: MCWN01415035, Severity: RESOURCE\_NOT\_FOUND\_ENR
- 3. Validar si la cuenta está inactiva:**
  - Si lo está: MCWN01415044

### **Validación de NIP:**

- 4. El NIP proporcionado (accountNip) debe coincidir exactamente con el almacenado en la cuenta.**
  - Si no coincide: MCWN01415036

### **Procesamiento:**

- 5. Si las validaciones son exitosas, se devuelve toda la información de la cuenta desde BD.**
- 6. Se asigna holderType según el clientType:**

- 0 → "FISICO"
- 1 → "MORAL"

**Nota:** El equipo debe analizar si esta transacción requiere rollback ante fallos en la validación del NIP o estado de cuenta.

---

## **MCWNT102 - Consulta de Todas las Cuentas**

### **Validaciones:**

- **No requiere parámetros de entrada.**

### **Procesamiento:**

1. **Se consultan todas las cuentas activas desde la BD.**
2. **A cada cuenta se le asigna holderType según su clientType:**
  - **0 → "FISICO"**
  - **1 → "MORAL"**

**Nota: Esta transacción no requiere rollback.**

---

## **MCWNT103 - Actualización de Cuenta**

### **Validaciones Generales:**

1. El campo accountNumber es obligatorio.
  - Si no se recibe: MCWN01415034.
2. Se debe verificar que la cuenta exista:
  - Si no existe: MCWN01415035, Severity: RESOURCE\_NOT\_FOUND\_ENR.
3. Validar si la cuenta ya está inactiva:
  - Si es así: MCWN01415037.

### **Validación de tipo de persona (client-document del header):**

4. Solo se aceptan valores "0" o "1".

- Si el valor es inválido: MCWN01415030.

### **Si nuevo tipo es Persona Física:**

5. Se deben validar los campos lastName, age, curp, rfc:

- Si no se reciben y tampoco existen previamente en la BD: MCWN01415034.
- 6. Validar longitud de curp: debe ser de 18 caracteres: MCWN01415031.
- 7. Validar longitud de rfc: debe ser de 13 caracteres: MCWN01415032.

### **Si nuevo tipo es Persona Moral:**

8. No deben enviarse campos propios de persona física (curp, age, lastName).

- Si alguno se recibe: MCWN01415045 (uno por cada campo).
- 9. El sistema debe limpiar automáticamente estos campos con los siguientes valores:
  - curp = "NO\_APlica", age = 0, lastName = "NO\_APlica".
- 10. Validar que rfc tenga exactamente 12 caracteres: MCWN01415033.

### **Cambio de tipo de persona:**

- Si se detecta cambio entre persona física y moral (o viceversa):
  - Aplicar reglas y validaciones completas del nuevo tipo.
  - Limpiar o reemplazar todos los campos que ya no apliquen al nuevo tipo.

**Validación de tarjeta:**

11. Si se modifica el accountCard, validar que no exista en otra cuenta activa:

- Si existe: MCWN01415039.

**Procesamiento:**

12. Solo se deben actualizar los campos que hayan sido modificados.

13. Si la operación no afecta ninguna fila en BD:

- Error: MCWN01415041.

14. Se debe recuperar la cuenta actualizada desde BD y retornar su estado final.

15. Asignar nuevamente holderType conforme al nuevo clientType:

- Si clientType == 0 → holderType = "FISICO".
- Si clientType == 1 → holderType = "MORAL".

**Nota:** El equipo debe analizar si esta transacción requiere rollback si la operación de actualización falla.

---

## **MCWNT10401 - Eliminación Lógica de Cuenta**

### **Validaciones Generales:**

- 1. accountNumber es obligatorio.**
  - **Si no se recibe: MCWN01415034**

### **Validaciones de existencia:**

- 2. Verificar si la cuenta existe en BD.**

- **Si no existe: MCWN01415035, Severity: RESOURCE\_NOT\_FOUND\_ENR**
- 3. Validar si la cuenta ya está inactiva:**

- **Si lo está: MCWN01415037**

### **Procesamiento:**

- 4. Se actualiza el estado de la cuenta (status = 0).**

- **Si la operación falla: MCWN01415041**

**Nota: El equipo debe analizar si esta transacción requiere rollback si la cuenta no pudo eliminarse correctamente.**

---

## **MCWNT10402 - Reactivación de Cuenta**

### **Validaciones Generales:**

- 1. accountNumber es obligatorio.**
  - **Si no se recibe: MCWN01415034**

### **Validaciones de existencia:**

- 2. Verificar si la cuenta existe en BD.**
  - **Si no existe: MCWN01415035, Severity: RESOURCE\_NOT\_FOUND\_ENR**
- 3. Validar si la cuenta ya está activa:**
  - **Si lo está: MCWN01415047**

### **Procesamiento:**

- 4. Se actualiza el estado de la cuenta (status = 1).**
  - **Si la operación falla: MCWN01415046**

**Nota: El equipo debe analizar si esta transacción requiere rollback si no se logró reactivar correctamente la cuenta.**