

Informe Técnico Funcional: Prototipo de Conciliación Bancaria

Fecha: 25 de Noviembre de 2025

Equipo: TheGitGuardians

Integrantes:

- *Líder:* Juan Morales, 30916541
- *Sub-Líder:* Ismael Chávez, 31169498
- *Sub-Líder:* Omaira Rodríguez, 15351394
- *Integrante:* Estefany Gil, 31368020
- *Integrante:* Herly Canizales, 31663564
- *Integrante:* José Escalona, 31041717
- *Integrante:* Gabriel León, 28454113
- *Integrante:* Heibert Ramírez, 31986852
- *Integrante:* Leoneiber Mendoza, 31121996

Contexto: Gestión Financiera / Auditoría

Alcance: Descripción del flujo de navegación, lógica de interfaz, validaciones y algoritmos de reporte para la aplicación web de conciliación bancaria.

1. Introducción y Arquitectura General

El sistema es una aplicación web desarrollada en TypeScript para la gestión integral de movimientos bancarios y su conciliación contra estados de cuenta externos.

Se basa en una arquitectura de diseño **MVC (Modelo-Vista-Controlador)** que garantiza una clara separación de responsabilidades:

- **Modelos:** Gestionan la lógica de negocio, cálculos de saldo y comunicación con la base de datos (Cl_dcytdb).
- **Vistas:** Gestionan la interfaz de usuario, formularios y tablas de visualización.
- **Controlador:** Orquesta la comunicación entre los datos y la interfaz.

La interfaz mantiene una identidad visual uniforme y moderna:

- **Encabezado Institucional:** Barra superior con el título del sistema y navegación global.
- **Diseño de Tarjetas y Tablas:** Uso de contenedores para el Dashboard y tablas interactivas para el listado de movimientos, optimizado para visualización responsive.
- **Navegación Clara:** Botones de acción explícitos con iconografía SVG (Ojo, Lápiz, Basurera) y alertas visuales.

2. Descripción Detallada de Módulos y Pantallas

2.1. Pantalla Principal (Dashboard / Home)

Descripción: Es el punto de entrada de la aplicación. Su objetivo es proporcionar acceso rápido a las operaciones financieras y ofrecer una visión inmediata de la liquidez.

Elementos y Funcionalidad Lógica:

1. **Botones de Navegación Principal (Accesos Directos):**
 - **[Abono]:** Redirige al formulario de movimiento configurado como "Ingreso".
 - **[Cargo]:** Redirige al formulario de movimiento configurado como "Egreso".
 - **[Conciliar]:** Redirige al módulo de carga de archivos JSON para la conciliación.
 - **[Ajustes]:** Acceso a la gestión de Categorías.
 - **[Ver Movimientos]:** Acceso a la tabla completa con filtros.
2. **Sección de Estadísticas (Saldo en Tiempo Real):**
 - **Saldo Total:** Muestra el cálculo dinámico (Σ Abonos - Σ Cargos).
 - Este número no es estático; debe ser el resultado del método SaldoActual() invocado desde el modelo Cl_mBanco al momento de cargar la pantalla.
 - **Formato:** Debe mostrarse formateado en Bolívares (Bs.) o la moneda configurada.

CONCILIACION BANCARIA

Registrar Movimiento

Abono

Cargo

Operaciones

Ajustes

Conciliar

Saldo Total

22700.00 Bs

Ver Movimientos

Designed by TheGitGuardians

2.2. Módulo de Movimientos

2.2.1. Pantalla: Registrar / Editar Movimiento (Formulario)

Descripción: Formulario único reutilizable para Abonos y Cargos.

Campos y Validaciones:

1. **Campo Fecha - Hora:**
 - **Formato:** Obligatorio. Debe estar en formato **DD-MM-AAAA HH:MM** (26-11-2025 23:33).
 - **Lógica:** No puede ser una fecha posterior a la actual (fechaOK).
2. **Campo Referencia:**
 - **Obligatorio:** No puede estar vacío.
 - **Longitud:** Debe tener exactamente 13 caracteres (referenciaOK).
 - **Unicidad:** El sistema debe validar que la referencia no exista ya en la base de datos (en modo Inserción).
3. **Campo Descripción:**
 - **Restricción:** El texto no puede ser mayor a 50 caracteres. (descripcionOK).
4. **Campo Monto:**
 - **Restricción:** Debe ser un valor numérico mayor a 0 (montoOK).
5. **Campo Categoría:**
 - Select dinámico llenado desde la entidad Cl_mCategoría.

The screenshot shows a web form titled 'CONCILIACION BANCARIA'. It has a section 'Registrar Movimiento' with a sub-section 'Registrar Abono'. The form contains the following fields: 'Fecha y hora:' with a text input, 'Referencia:' with a text input, 'Categoría:' with a dropdown menu showing 'Seleccione la categoría', 'Descripción:' with a text input, and 'Monto:' with a text input. At the bottom, there are two buttons: 'Registrar' and 'Cancelar'.

Regla de Integridad en Edición:

Al editar un monto, el sistema debe realizar una operación de dos pasos: primero revertir el saldo del monto "viejo" y luego aplicar el saldo del monto "nuevo" para evitar inconsistencias contables.

2.3. Módulo de Categorías

2.3.1. Pantalla: Gestión de Categorías(Listado/Tabla)

Descripción: Permite administrar las etiquetas (Categorías)

asignadas a movimientos bancarios (Abonos/Cargos).

Funcionalidades Clave:

1. **Listado:** Tabla con Nombre de Categoría.
2. **Creación:** Botón **[Registrar]**.

The screenshot shows a web form titled 'Gestión de Categorías'. It has a section 'Nombre de la Categoría:' with a text input. Below this are two buttons: 'Registrar' and 'Regresar'. Below the buttons is a table titled 'Categorías Registradas'. The table has two columns: 'NOMBRE' and 'ACCIONES'. The table contains two rows: 'Ingresos' and 'Deudas'. Each row has two action buttons: a blue button with a pencil icon and a red button with a trash icon.

Nombre obligatorio y único.

3. Acciones:

- **Editar:** Modificar nombre. Los movimientos asociados actualizan la referencia automáticamente.
- **Eliminar:** Solo si no tiene movimientos asociados (contador = 0). Si tiene movimientos, se bloquea y se exige reasignar o eliminar dependencias.

2.4. Módulo de Tabla de Movimientos

2.4.1. Pantalla: Gestión de Movimientos (Listado/Tabla)

Descripción: Muestra la bitácora completa de transacciones registradas con capacidades de filtrado.

Elementos y Reglas:

1. Filtros de Cabecera:

- Permite filtrar por: Texto (Referencia/Descripción),
- Categoría, Tipo (Abono/Cargo) y Rango de Fechas.

2. Listado de Registros:

- Columnas: Fecha, Referencia, Descripción, Categoría,
- Monto (Verde para Abonos, Rojo para Cargos).

3. Opciones por Ítem (Iconos):

- **[Icono Ojo (Ver)]:** Muestra el detalle completo del movimiento en modo solo lectura.
- **[Icono Lápiz (Editar)]:** Redirige al formulario precargando los datos del movimiento.
- **[Icono Basurera (Eliminar)]:**
 - **Validación Crítica:** Solicita confirmación mediante SweetAlert.
 - **Regla de Negocio:** Al eliminar, el sistema debe "revertir" el impacto en el saldo (restar si era abono, sumar si era cargo) antes de borrar el registro físico.

Tabla de movimientos

Regresar

Buscar:

Referencia o descripción...

Categoría:

Todas

Tipo:

Todos

Desde:






















dd/mm/aaaa

Hasta:

dd/mm/aaaa

Filtrar

Limpiar

FECHA HORA	CATEGORÍA	MONTO	ACCIONES
20-11-2025 10:00	Ingresos	20000	  
20-11-2025 11:00	Transporte	50	  
20-11-2025 17:00	Transporte	50	  
20-11-2025 08:00	Alimentos	450	  
21-11-2025 11:00	Estudios	250	  
26-11-2025 10:30	Ingresos	4500	  
26-11-2025 08:15	Internet	1000	  

2.4.2. Pantalla: Detalles de Movimientos

Descripción: Muestra la información completa de un movimiento bancario sin permitir edición.

Claves:

- **Solo Lectura:** Campos (Fecha, Referencia, Descripción, Monto, Categoría) inmodificables.
- **Monto:** Resaltado (Bs.) con color según tipo (Verde: Ingreso, Rojo: Egreso).
- **Acción:** Botón **[Regresar]** al listado.
- **Auditoría (Opcional):** Incluir creador y fechas.
- **Acceso:** Desde el **[Icono Ojo]** del listado, carga usando `buscarMovimientoPorReferencia`.

Detalle del Movimiento

Fecha y Hora:

20-11-2025 10:00

Referencia:

00000000000001

Categoría:

Ingresos

Descripción:

Pago Pendiente

Monto:

20000

Tipo:

Abono

Regresar

2.5. Módulo de Conciliación Bancaria

2.5.1. Pantalla: Gestión de Conciliados (Tabla)

Descripción: Módulo especializado para auditar los registros internos contra el banco.

Flujo y Lógica:

1. **Carga de Datos:** Usuario sube un archivo .json con el estado de cuenta bancario.
2. **Algoritmo de Comparación (realizarConciliacion):**
 - El sistema itera sobre el JSON y compara contra la base de datos interna.
 - **Criterio de Match:** Coincidencia exacta de Referencia Y Monto.

CONCILIACION BANCARIA

Conciliación Bancaria

Cargar Estado de Cuenta (JSON):

Seleccionar archivo

estado_cuenta_banco.json

Conciliar

Regresar

Resultados de Conciliación

FECHA HORA	CATEGORÍA	MONTO	ESTADO	ACCIONES
20-11-2025 11:00	Transporte	50	Conciliado	
20-11-2025 17:00	Transporte	50	Conciliado	
20-11-2025 08:00	Alimentos	450	Conciliado	
26-11-2025 08:15	Internet	1500	No Conciliado	Conciliar
26-11-2025 10:30	Ingresos	4500	Conciliado	

3. Tabla de Resultados:

- **Estado "Conciliado" (Verde):** El movimiento existe en ambos lados.
- **Estado "No Conciliado" (Rojo):** El movimiento existe en el JSON pero no en la base de datos interna.

4. Acción Correctiva:

- Para los ítems rojos, se habilita un botón [**Conciliar**].
 - **Automatización:** Al hacer clic, abre el formulario de "Agregar Movimiento" con **todos los datos precargados** desde el JSON (Fecha, Ref, Monto, Categoría).
 - Al guardar, la tabla de conciliación se actualiza automáticamente a estado "Verde".
-

3. Reportes que se plantean

Se tienen las siguientes estructuras de datos básicas para los retornos:

- **Objeto Movimiento:** { id, fecha, referencia, descripcion, monto, categoría, tipo }
- **Objeto Conciliación:** { movimiento, estado: "Conciliado" | "Faltante" }

3. a. Métodos de Reporte Por Movimiento

3.a.1. Buscar movimiento por referencia

- **Nombre del método:** buscarMovimientoPorReferencia
- **Descripción general:** Localiza y devuelve la información completa de una transacción específica basada en su identificador bancario.
- **Parámetros de entrada:**
 - referencia (String): Cadena de 13 caracteres.
- **Retorno:** Un Objeto Movimiento si se encuentra; null si no existe.
- **Consideraciones:** Búsqueda exacta. Fundamental para evitar duplicidad de registros.

3.a.2. Filtrar movimientos por rango de fecha

- **Nombre del método:** filtrarMovimientosPorFecha
- **Descripción general:** Devuelve un subconjunto de movimientos ocurridos dentro de un periodo específico.
- **Parámetros de entrada:**
 - fechaInicio (Date/String).
 - fechaFin (Date/String).
- **Retorno:** Un Array de Objetos Movimiento.
- **Consideraciones:** Debe incluir los movimientos ocurridos exactamente en los límites del rango (inclusive).

3.a.3. Listar movimientos por categoría

- **Nombre del método:** listarMovimientosPorCategoria
 - **Descripción general:** Agrupa y devuelve las transacciones asociadas a un concepto presupuestario específico.
 - **Parámetros de entrada:**
 - nombreCategoria (String).
 - **Retorno:** Un Array de Objetos Movimiento.
 - **Consideraciones:** Útil para análisis de gastos específicos.
-

3. b. Métodos de Reporte Por Banco (Globales)

3.b.1. Calcular saldo total actual

- **Nombre del método:** SaldoActual (implementado en Cl_mBanco)
- **Descripción general:** Calcula la liquidez real de la cuenta procesando todo el historial de transacciones.
- **Parámetros de entrada:** Ninguno.
- **Retorno:** Un valor Number (Float).
- **Consideraciones:**
 - Fórmula: $(\sum \text{Monto donde tipo='Abono'}) - (\sum \text{Monto donde tipo='Cargo'})$.
 - Debe recalcular cada vez que se agrega, edita o elimina un movimiento.

3.b.2. Generar reporte de conciliación (Comparativo)

- **Nombre del método:** generarReporteConciliacion
- **Descripción general:** Ejecuta el cruce masivo de información entre los datos del sistema y un conjunto de datos externo (JSON).
- **Parámetros de entrada:**
 - datosBancoExterno (Array de Objetos JSON).
- **Retorno:** Un Array de Objetos Conciliación indicando el estatus de cada transacción.
- **Consideraciones:** Este proceso es intensivo. Identifica discrepancias financieras y audita la integridad de los datos locales frente a la realidad bancaria.