

Digital Transaction Ledger & Client CRM Blueprint (Updated)

This blueprint incorporates the requested enhancements: including **Canadian Dollars (CAD)** and a robust **Edit History** for auditability.

1. Currency Expansion: Canadian Dollar (CAD)

The existing currency fields must be expanded to officially support CAD in both sendCurrency and receiveCurrency fields.

Field Name	Data Type	Description
sendCurrency	String	EUR, USD, GBP, CAD (Source currency)
receiveCurrency	String	IRR, USD, EUR, CAD , etc. (Target currency)

This ensures that the application UI and database structure are ready to handle Canadian Dollar transactions, whether as cash exchange or bank transfer source/target.

2. Auditable Transaction Edit History

This is the most critical addition, ensuring that while you can correct mistakes, the original, unedited data is preserved for compliance and audit purposes.

A. New Data Structure Fields for Audit

The Transactions Collection schema will be updated with three new fields to manage the audit trail of any edits.

Field Name	Data Type	Description
isEdited	Boolean	NEW. Set to true if the transaction has ever been modified. Default is false.
editHistory	Array of Objects	NEW. Stores a record of every previous version of the document before an edit occurred. CRITICAL

		for auditability.
lastEditedBy	String	NEW. Stores the loggedByUserId of the last person who made an edit.

B. The editHistory Object Structure

When a user clicks the "Edit" button and saves a change, the current transaction object (before the changes are saved) is pushed into the editHistory array.

Each object within the editHistory array will look like this:

Edit History Object Field	Data Type	Description
timestamp	Timestamp	Date and time the edit occurred.
editorUserId	String	loggedByUserId of the employee who made the correction.
originalDocument	Map (Object)	A full copy of the transaction document BEFORE the edit. This includes the original sendAmount, rateApplied, feeCharged, etc.
editReason	String	(Optional but Recommended) A mandatory text field in the UI explaining <i>why</i> the transaction was edited (e.g., "Client provided wrong send amount").

C. User Flow for Editing

1. **"Edit" Button Visibility:** An "Edit" button is visible on the detailed Transaction History view.
2. **Edit Action:** When the user clicks "Edit," a modal opens allowing modifications to fields

like `sendAmount`, `receiveAmount`, `userNotes`, and requires an **editReason**.

3. Saving Changes (The Atomic Operation):

- The system creates the `editHistory` object (capturing the current transaction data, timestamp, editor ID, and reason).
- The system updates the transaction document:
 - Sets `isEdited`: `true`.
 - Pushes the new audit object into the `editHistory` array.
 - Updates the modified data fields (`sendAmount`, `receiveAmount`, etc.).
 - Updates `lastEditedBy` and `date` (to reflect the edit time).

4. Displaying the Audit Trail (The "Previous" Touch):

- If `isEdited` is `true`, a clear visual indicator (e.g., an **"EDITED"** label) appears next to the transaction.
- Tapping or clicking this label (or a dedicated "View History" button) shows a chronologically ordered list of the previous document versions stored in the `editHistory` array.