Design rationale and justification

Our schema centers on Transactions as the canonical records extracted from MOMO SMS messages. The Messages owner gets all his messages organized into 4 entities(The Users, Transactions, Transaction_categories and System_Logs). Users are modelled in a single Users table to avoid duplication and here the owner gets to see all persons (considered as Users) whom he sent or received money from, also considering those organisation like banks or MTN he gets transactions from and as well as the messages that follow it. Transactions_categories separates each transaction in its own related category like Money transfer, Payments, Utilities, Deposit, and others. System_logs records ETL/parsing events and links to transactions for traceability and debugging.

I intentionally used DECIMAL (10,0) for Amount and NewBalance because Rwandan francs are whole units (no cents). Numeric integrity is enforced with CHECK constraints on non-negativity. CategoryType is an ENUM that provides a controlled vocabulary for common transaction types, allowing fast filtering and reducing inconsistent free-text values. Messages in categories preserves the original text used for parsing rules and enables re-parsing when formats change.

Foreign keys enforce referential integrity: transactions must reference an existing user and category, and system logs reference transactions. Indexing choices (primary keys and the implicit index on Foreign Keys) optimize joins typical in reporting and audits. This design favors auditability, analytical queries, and iterative refinement of parsing logic while staying normalized and performant for moderate message volumes.

Data dictionary (table-by-table; column / type / description)

Users

- **Id:** INT AUTO_INCREMENT — Primary key. Unique user identifier.
- **Fullnames:** VARCHAR(255) — Full name.
- **PhoneNumber:** VARCHAR(10) UNIQUE — Phone number, national format.

Transactions_categories

- **Id:** INT AUTO_INCREMENT — Primary key.
- **UserId:** INT — FK → Users(Id). Owner of category (who created or uses it).
- **CategoryName:** VARCHAR(255) UNIQUE — Name of category (unique).

- **Messages:** VARCHAR(255) — Representative message snippet from XML used to classify transactions.

Transactions

- **Id:** INT AUTO_INCREMENT — Primary key.
- **UserId:** INT — FK → Users(Id). Owner/actor of the transaction.
- **CategoryId:** INT — FK → Transactions_categories(Id).
- **Amount:** DECIMAL(10,0) CHECK (Amount >= 0) — Transaction amount (RWF whole units).
- **Date:** DATETIME — When transaction occurred.
- **Fee:** INT CHECK (Fee >= 0) — Fee amount (RWF).
- **NewBalance:** DECIMAL(10,0) CHECK (NewBalance >= 0) — Balance after transaction.
- **Currency:** VARCHAR(10) DEFAULT 'RWF' — Currency code.
- **CategoryType:** ENUM(...) — High-level transaction type.
- **ServiceCenter:** VARCHAR(255) — Optional vendor/service center.
- **Status:** INT — Optional status code.
- **Locked:** INT — Optional boolean flag (0/1).

System_logs

- **LogId:** INT AUTO_INCREMENT — Primary key.
- **TransactionId:** INT — FK → Transactions(Id).
- **LogMessage:** VARCHAR(255) — Short message / title.
- **Issue:** ENUM('ERROR', 'WARNING', 'INFO') — Severity.
- **Description:** TEXT — Full description.
- **CreatedAt:** TIMESTAMP DEFAULT CURRENT_TIMESTAMP — When entry created.