With an emphasis on traceability, query efficiency, and future extension, the database is built to dependablely store MoMo SMS transaction data. Users, Transactions, Transaction\_Categories, and System\_Logs are the fundamental entities. All participants (senders and receivers) are stored in Users; Transactions refers to a single user record in either the sender or receiver role. Financial fields (amount, currency), timestamps, status, raw XML snippets (for auditing or debugging), and foreign keys for users are all contained in transactions, the system’s main object.

In order to provide flexible categorization (a single transaction can map to numerous categories or sub-types), Transaction\_Categories provides canonical kinds (such as airtime, payment, transfer, and fee) and is connected to Transactions via a junction table transaction\_category\_map. Data lineage and dead-letter tracking are made possible by System\_Logs, which logs ETL processing events and faults along with connections to Transactions where appropriate. Foreign keys are used to enforce referential integrity, and the right indexes (transaction\_time, sender\_id, receiver\_id, and status) facilitate lookup and analytics. Data types employ VARCHAR for text, DECIMAL for monetary data, and DATETIME for timestamps.

Consistent states are guaranteed via ENUM (for status) and CHECK restrictions. The architecture maintains a normalized but practical transaction table: the cleaned fields are stored for quick queries, and the raw XML is kept in a TEXT/JSON column for forensic purposes. This framework is easy to expand (e.g., add wallets, merchants, or multi-currency support) and supports standard analytics tasks (volume by category, total value by day, failed transaction audit).