

Database Design Justification

The ETL workflow specified in our system architecture and the MoMo SMS XML structure are both reflected in our database schema.

All transactional information is contained in the raw messages that are provided by the XML input.

We divide raw storage, parsed transactions, and processing logs into distinct entities to maintain data integrity and facilitate analytics.

Every original SMS, complete with metadata like protocol, address, date, and message body, is kept in the messages table exactly as it was received.

Every record is guaranteed to be auditable and reprocessable in the event that the parsing logic changes thanks to this raw layer.

From these messages, our ETL pipeline parses business details (amount, currency, sender, receiver, and timestamps) into the transactions table, creating a clean and query-friendly dataset.

We introduce a categories lookup table and a transaction_category_link junction table to resolve this many-to-many relationship because a single transaction can belong to multiple categories (for instance, a deposit that is also a promotional transfer).

Without duplicating data, this design allows for flexible analytics, such as filtering by promotional activity or payment type.

Errors and dead-letter events from the ETL process are recorded in the system_logs table to facilitate debugging and ensure traceability.

Effective joins for dashboards and reports are made possible by foreign keys and indexes, which also maintain referential integrity.

Overall, this schema offers a strong basis for storing, querying, and analyzing MoMo transaction data by striking a balance between normalization, scalability, and operational transparency.