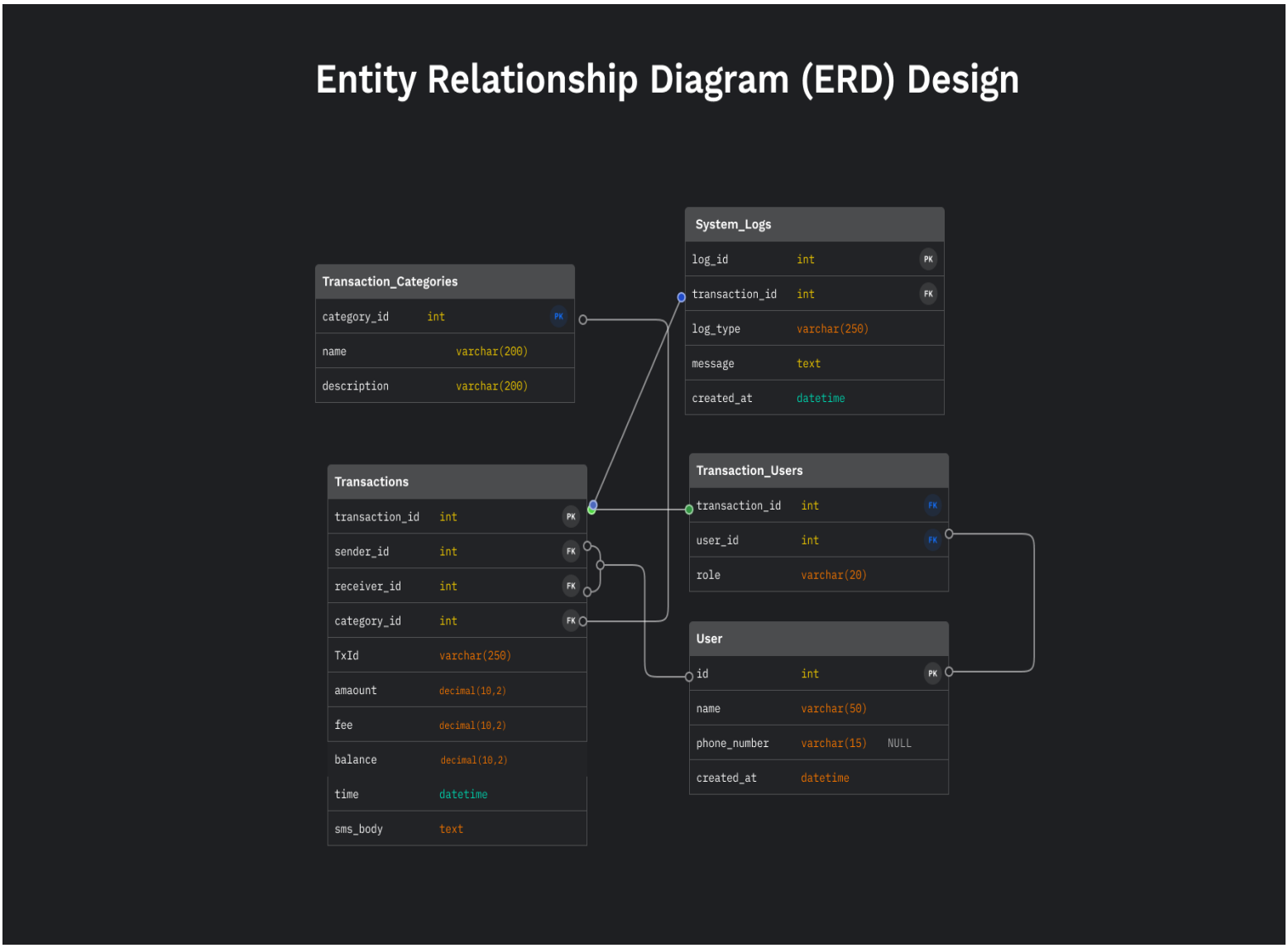
Entity Relationship Diagram (ERD) Design & Justification (Documentation)

Down below is a relational database design designed after careful analysis of xml data given:



 # ERD Design Justification

The Entity Relationship Diagram (ERD), we designed was to reshape mobile money (MoMo) transactions extracted from SMS data given in xml file. The design captures core entities and relationships necessary for robust financial data processing.

The **Transactions** entity is central, storing details such as transaction amount, fee, balance, timestamp as time in our table's field, and raw SMS body. It links directly to **Transaction_categories**, which classifies transactions into types such as transfers, deposits, and payments. This separation supports flexibility in categorization and reporting.

The **User** entity manages customer details, including names and phone numbers, while sender and receiver roles in a transaction are represented using foreign keys. To support more complex associations where multiple users may be linked to a single transaction, the **Transaction_users** junction table resolves the many-to-many relationship and specifies user roles (e.g., sender, receiver, agent).

To ensure accountability, **System_logs** records processing metadata such as log type, messages, and timestamps. This entity provides auditability and helps track errors or processing activities

Generally, ERD is assured of integrity of information, scalability and trace-ability, similar to the actual MoMo SMS data structure. The design can do both transactional and analytical applications and thus it can be implemented in a financial system..