

Architecture Decision Records (ADRs)

ADR 1: Transaction Manager

Status: Accepted

Decision: Opt for a Transaction Manager as a central orchestration service to coordinate the different steps and interactions between microservices involved in transaction processing.

Context: A transaction process in a banking system can involve multiple steps and services. Effective coordination is necessary to ensure transaction integrity and a consistent user experience.

Considerations: For - Centralizes transaction management, reduces complexity for clients, and facilitates maintenance. Against - Creates a single point of failure and can become a bottleneck if not properly scaled.

Consequences: The Transaction Manager must be designed to handle high load and be highly available to minimize the risks of failure.

ADR 2: Transaction Validator

Status: Accepted

Decision: Use a Transaction Validator to verify the legitimacy and feasibility of transactions before processing.

Context: Validation is a crucial step to prevent fraud and ensure solvency of transactions.

Considerations: For - Ensures an additional layer of security and prevents inappropriate transactions. Against - Can introduce delays if the validation rules are too complex or the service is overloaded.

Consequences: The Transaction Validator must be optimized for high performance and integrate dynamic validation rules to adapt to different transaction requirements.

ADR 3: Currency Exchange

Status: Accepted

Decision: Develop a Currency Exchange service capable of calculating the necessary currency conversions for simulations and the validation of transactions.

Context: Multi-currency transactions are common in global banking operations and require precise conversions based on current exchange rates.

Considerations: For - Allows accurate simulations for clients and ensures correct conversions during validations. Against - Depends on real-time exchange rates that can vary rapidly.

Consequences: The Currency Exchange must have access to updated exchange rate data and integrate logic to handle situations where the stock exchange is closed, including the application of fees.

ADR 4: Transaction Processing

Status: Accepted

Decision: Establish a Transaction Processing service to execute account updates following transaction validation and to initiate transactions with the stock exchange through the Trader.

Context: After validation, transactions need to be processed in a way that reflects changes in user accounts.

Considerations: For - Centralizes the processing of transactions after validation to ensure data integrity. Against - Requires close synchronization with other services and effective error management.

Consequences: The Transaction Processing must be highly reliable and capable of handling concurrent transactions consistently.

ADR 5: Trader

Status: Accepted

Decision: Implement a dedicated Trader service to perform transactions on the stock exchange on behalf of the bank.

Context: Some transactions require direct interaction with the stock exchange for the buying and selling of currencies.

Considerations: For - Allows the bank to respond agilely to market conditions and execute currency transactions in a timely manner. Against - Increases system complexity and introduces market risks.

Consequences: The Trader must be designed to operate reliably during stock exchange hours and appropriately manage transactions when the exchange is closed.