# Crypto Exchange - Data Contracts and Service Interfaces

## DTO Definitions

### 1. Transfer (Internal) – Off-chain

TransferRequestDTO

• fromUserId (Long): Sender in the platform

• toUserId (Long): Receiver in the platform

• tokenSymbol (String): Token to transfer

• amount (BigDecimal): Amount transferred

TransferResponseDTO

• txId (String): Internal transaction ID

• timestamp (Instant)

### 2. Deposit (ETH/ERC-20) – On-chain

DepositRequestDTO

• userId (Long): Who is depositing

• tokenSymbol (String): Token type

• txHash (String): Blockchain transaction hash

DepositResponseDTO

• txId (String): Internal transaction ID

• status (String): pending/confirmed

• blockNumber (Long): Blockchain inclusion

• amount (BigDecimal)

### 3. Withdraw (ETH/ERC-20) – On-chain

WithdrawRequestDTO

• userId (Long): Who is withdrawing

• tokenSymbol (String): Token type

• amount (BigDecimal): Requested withdrawal

• toAddress (String): Destination blockchain address

WithdrawResponseDTO

• txId (String): Internal transaction ID

• txHash (String): Blockchain hash if sent

• status (String): pending/sent/failed

## Field Rationale

• userId, tokenSymbol: To reference user/token in DB

• amount: For balance updates and validation

• txHash, blockNumber: To track real blockchain activity

• txId: For internal transaction traceability

• status: For monitoring state across lifecycle

## Service Class & Interface Plan

Interface: TransferService

TransferResponseDTO transfer(TransferRequestDTO request);

Interface: DepositService

DepositResponseDTO processDeposit(DepositRequestDTO request);

Interface: WithdrawService

WithdrawResponseDTO requestWithdrawal(WithdrawRequestDTO request);

## Next Steps

• Review DTOs and interfaces with the dev team.

• Finalize structure before implementing logic next sprint.

• Ensure shared understanding of real vs. simulated flows.