# Build a crypto wallet and custody stack

A wallet and custody architecture covering onboarding, key management, transaction policy, qualified custody, compliance, fiat connectivity and network operations.

LAYER 1 / REQUIRED

## Embedded or application wallets

Create wallet experiences and authenticate users.

- ☐ Custodial model
- ☐ Recovery and export
- ☐ Supported networks

Privy — Embedded wallets under Stripe
Dynamic — Wallet and authentication
Crossmint — Wallet infrastructure
Dfns — Wallet infrastructure

LAYER 2 / REQUIRED

## Key management and policy

Control keys, approvals and transaction policy.

- ☐ MPC or HSM model
- ☐ Quorum and policy controls
- ☐ Recovery and audit

Turnkey — Key and wallet infrastructure
Fordefi — MPC wallet controls
Fireblocks — Wallet and transfer infrastructure
Utila — Institutional wallet operations

LAYER 3 / OPTIONAL

## Regulated or institutional custody

Use external custody where legal or risk requirements demand it.

- ☐ Legal entity and permissions
- ☐ Asset and network support
- ☐ Segregation and insurance

BitGo — Custody and wallets
Anchorage Digital — Institutional custody
Zodia Custody — Institutional custody
Komainu — Institutional custody
Copper — Custody and settlement

LAYER 4 / REQUIRED

## Blockchain compliance and Travel Rule

Screen addresses and exchange counterparty data.

- ☐ Chain coverage
- ☐ Risk methodology
- ☐ Travel Rule interoperability

Chainalysis — Blockchain analytics
TRM Labs — Blockchain intelligence
Elliptic — Crypto AML
Notabene — Travel Rule

LAYER 5 / OPTIONAL

## Fiat connectivity and liquidity

Fund, convert and settle wallet balances.

- ☐ On/off-ramp licences
- ☐ Liquidity and settlement
- ☐ Bank eligibility

Zero Hash — Embedded digital assets
BVNK — Stablecoin and fiat infrastructure
MoonPay — On/off-ramp
Banking Circle — Fiat accounts and settlement
Mercuryo — Embedded fiat-to-crypto purchase and sale flows

## Questions to ask providers

- ☐ Which legal entity contracts for each service and in which jurisdictions?
- ☐ Which responsibilities remain with the product operator rather than the provider?
- ☐ What are the implementation, approval, testing and migration timelines?
- ☐ How are incidents, exits, data portability and business continuity handled?
- ☐ Who can recover, freeze or move assets under each failure scenario?