Social. P2P Payments - Web3 Approach

Summary

This document outlines a comprehensive architecture and implementation strategy for the Social. application to enable users to connect and send gifts (monetary transfers) with integrated Web3 capabilities through Alchemy's wallet infrastructure and Meld's on/off-ramp services, as well as the flows for wallet creation, transfer funds, on and off ramping.

Summary **1**
Architecture Overview **2**
Wallets Structure and Flow **3**
Costs Breakdown **4**
Alchemy Infrastructure Costs 4
Meld On/Off-Ramp Costs 4
Gas Fees (Network-Level Costs) 4
Security and Compliance Considerations **4**
User Journey Processes **5**
Onboarding and Wallet Creation with Alchemy 5
Transfer Process: Sending USDC to Another User 6
On-Ramp Process: Adding Funds via Meld 8
Off-Ramp Process: Withdrawing Funds via Meld 10
Postponing Yield Generation: Technical, Operational, and Strategic Rationale **12**
Complexities 12

Social. Infra Cost Estimate

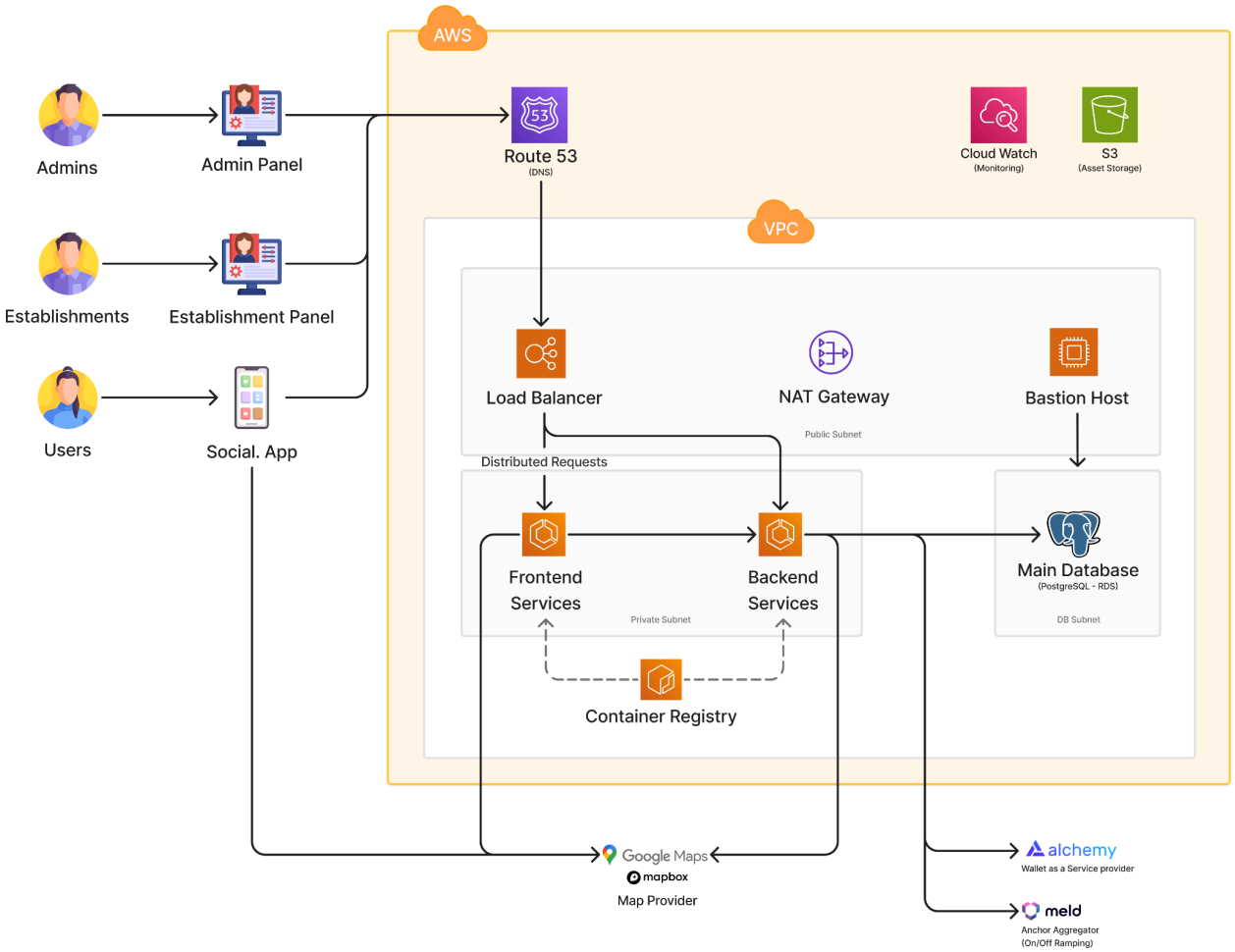
Cheesecake Labs

19/11/2025

Architecture Overview

The application follows a layered architecture spanning the client app, AWS cloud infrastructure and Web3 services:

- **Client Layer:** React Native mobile app, Admin Panel, and Establishment Panel for management operations.
- **Infrastructure Layer:** AWS Route 53 for DNS, NAT Gateway for egress connectivity, Load Balancer for request distribution across services.
- **Application Layer:** Frontend services (handling UI logic), Backend services (API and business logic), and Container Registry for deployment orchestration.
- **Web3 Integration Layer:** Alchemy for embedded wallet creation and USDC transaction management and Meld for fiat currency on/off-ramps
- **Data Layer:** PostgreSQL RDS for transactional data, with support for blockchain event indexing and balance reconciliation.



Social. Infra Cost Estimate

Cheesecake Labs

19/11/2025

Wallets Structure and Flow

The application will leverage Alchemy as the Wallet as a Service provider to keep the levels of effort on check and use Meld's on and off ramping services for a seamless experience for the final user.

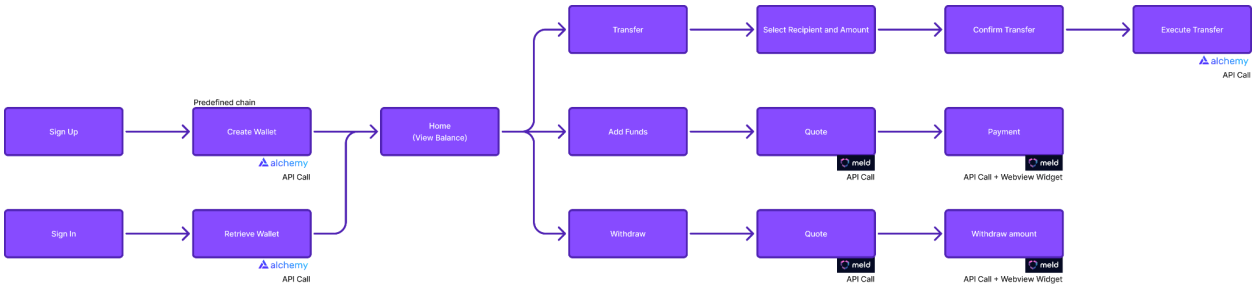
The Social. platform will have its own central wallet to account for gas sponsorship and transaction handling and will use USDC to reduce the complexity of creating and managing a new token while maintaining parity with the US Dollar.

Alchemy Embedded Wallets: User-controlled (non-custodial) wallets with social login via Google/Apple and social recovery options. Each user receives a deterministic wallet address derived from their authentication credentials.

Server Wallets: Developer-controlled wallets for backend operations (gas sponsorship, automated sweeps to lending pools, transaction monitoring).

Stablecoin: USDC (USD Coin) running on Ethereum mainnet, Polygon PoS, or Arbitrum for transaction efficiency and reduced gas costs. Users maintain USDC balances directly in their wallets.

Flow: The onboarding and transfer flows are interconnected and were designed to minimize friction for the user, assuming not all Social. users are necessarily crypto enthusiasts.



Social. Infra Cost Estimate

Cheesecake Labs

19/11/2025

Costs Breakdown

Alchemy Infrastructure Costs

Free Tier: 30M Compute Units (CUs) monthly at no cost. The average transaction consumes 25-50 CUs. At 50 CUs/transaction, this supports ~600k transactions monthly.

Meld On/Off-Ramp Costs

Meld does not charge a platform fee; instead, its revenue model is embedded in the spread negotiated with upstream ramp providers. For Social., this manifests as a 1-3% transaction fee passed through to end users. Meld's Smart Routing optimizes for lowest fees + highest conversion.

Gas Fees (Network-Level Costs)

Transactions execute on Polygon for minimal gas. Users pay network fees directly (not billed by Alchemy or Meld). However, if Social. sponsors gas, these become operational costs.
Gas Cost per Transaction: Polygon: $0.01-0.10 per transaction

Security and Compliance Considerations

Wallet Security:

- Use Alchemy's embedded wallets with social recovery to eliminate seed phrase management burden
- Store private keys in encrypted hardware security modules (HSMs) for server wallets

KYC/AML Compliance:

- Meld handles initial KYC via its upstream ramp providers, but our backend must track transaction limits and flag suspicious patterns
- Implement transaction monitoring: flag transfers exceeding thresholds, multiple transactions within minutes, off-hours activity for review

Smart Contract Risk:

- Move Yield generation strategies for the next phase to reduce risks associated with investing third party owned funds

Data Privacy:

- Store user wallet addresses and transaction history in encrypted database columns

- Comply with GDPR/CCPA: implement data deletion and export workflows

Social. Infra Cost Estimate

Cheesecake Labs

19/11/2025

User Journey Processes

Onboarding and Wallet Creation with Alchemy

The onboarding process will be designed to eliminate all blockchain complexity from the user's perspective. Users experience a frictionless signup flow identical to traditional fintech apps, with the wallet creation happening transparently in the background.

Step 1: App Launch and Initial Screen
The user opens Social. for the first time. They're presented with a clean welcome screen showing two options: "Sign up with Google" or "Sign up with Apple." There's no mention of cryptocurrency, wallets, or blockchain, just a simple choice between social authentication providers.

Step 2: Social Authentication
The user taps "Sign up with Google" (or Apple). The app redirects them to Google's authentication flow, where they log in with their existing Google account. This uses standard OAuth 2.0 authentication, the same mechanism used by Gmail, Spotify, and most modern apps.

Behind the scenes, the app sends the authentication token to Social. backend, which verifies it with Google's servers. Once verified, the backend knows this user's unique identity (their Google account ID).

Step 3: Automatic Wallet Creation
The moment authentication succeeds, Alchemy automatically creates a deterministic smart account wallet for the user. This wallet address is cryptographically derived from the user's Google account identity, meaning:

- The wallet address is always the same (deterministic)
- No private keys or seed phrases are generated or shown to the user
- The wallet can only be accessed by someone authenticated to that specific Google account
- If the user logs in from a different device using the same Google account, they see the same wallet and balance

The wallet creation happens instantly on Alchemy's infrastructure and no blockchain transaction is required at this stage.

Step 4: Backend Wallet State Registration
Social. backend stores the wallet address in the database, associating it with the user's profile. It also subscribes to Alchemy webhooks for this wallet address, which means any incoming or outgoing USDC transfers will trigger real-time notifications to the backend. This enables instant balance updates across all the user's devices.

Social. Infra Cost Estimate

Cheesecake Labs

19/11/2025

Step 5: Dashboard Display
The user is now logged in and sees their dashboard. The screen displays their wallet balance: $0.00 USDC (since they haven't received or deposited funds yet)

Quick actions suggestion - three buttons:

- "Add Funds" (opens on-ramp via Meld)
- "Send" (opens contact picker for transfers)
- "Withdraw" (opens off-ramp via Meld)

Transaction History: Empty list, ready to populate as they transact

The user is now fully onboarded with a live blockchain wallet, but they've experienced zero friction. They never managed private keys, seed phrases, or interacted with blockchain terminology.

Key UX Design Note: The app never shows the underlying wallet address to casual users. For tech-savvy users, there may be an optional "Wallet Details" section accessible from settings, showing the wallet address, network (Polygon), and USDC contract address. This maintains simplicity for mainstream users while providing transparency for crypto-aware users.

Transfer Process: Sending USDC to Another User

Once users have USDC in their wallet (via on-ramp), they can send money to other users with a frictionless peer-to-peer transfer experience.

Step 1: Initiating a Transfer
The user taps the "Send" button on their dashboard. A contact picker UI opens showing:

- List of their friends already using the app
- Search bar to find users by username
- Recent recipients (users they've sent to before)

This contact selection is entirely within the app with no blockchain addresses, just human-readable usernames.

Step 2: Selecting Recipient and Amount
The user selects a recipient from the list. The app displays a transfer form with:

- Recipient name displayed with their profile picture
- Amount Input: A text field where user enters the amount to be transferred (may be implemented as predefined amounts/gifts)
- Attached message to personalize the transfer

The user taps "Continue" or "Next".

Step 3: Transaction Review
The app displays a summary screen. This is the final confirmation step. The user reviews and taps "Confirm Send".

Step 4: Transaction Submission
Once confirmed, the app collects the recipient's wallet address (Alchemy wallet, which Social. backend already knows). The backend constructs a USDC transfer transaction using the transfer function.

The backend then submits this transaction through Alchemy's gasless orchestration service. If we're sponsoring gas fees, the transaction is wrapped in a "user operation" with a paymaster address, meaning the recipient wallet address doesn't need to hold funds to pay gas and the app backend covers it.

The transaction is signed by the sender's wallet (this signing happens in the app, with Alchemy handling the cryptographic details).

Step 5: On-Chain Settlement
The signed transaction is broadcast to the blockchain. Validators include it in the next block. The USDC smart contract executes, transferring the amount from the sender's address to the recipient's address. This typically takes 5-20 seconds on Polygon, depending on network congestion.

Step 6: Real-Time Balance Updates
The backend's webhook receives a notification that the transfer was confirmed on-chain. It immediately:

- Deducts the amount from the sender's balance cache
- Adds the amount to João's balance cache
- Records the transaction in the database with timestamp, recipient, amount, transaction hash, and block number

The sender's app displays a success message.

Step 7: Recipient Notification
The recipient is notified of the received transaction.

Both users' transaction histories are permanently recorded, creating an auditable payment trail.

Key UX Features:

- No wallet addresses required: Users only know their contact, not their blockchain address
- Instant feedback: Transfer confirms within 5-20 seconds, with real-time balance updates

Social. Infra Cost Estimate

Cheesecake Labs

19/11/2025

- Gasless from user perspective
- Social context: Transfers show who sent/received, not random wallet addresses

Social. Infra Cost Estimate

Cheesecake Labs

19/11/2025

On-Ramp Process: Adding Funds via Meld

Users start with $0 USDC. To fund their wallet, they use Meld's integrated on-ramp service to convert fiat currency into USDC.

Step 1: Initiating On-Ramp
The user taps the "Add Funds" button on their dashboard and Meld's widget or SDK loads directly within the app (embedded in a webview or iframe on web).

Step 2: Meld Widget Initialization
Meld's interface appears within the app, showing:

- You're sending: Currency selector (predefined)
- You'll receive: USDC (pre-filled, not editable)
- Amount input: User enters amount in fiat currency

Payment method selector shows a dropdown showing available options like:

- "Credit/Debit Card"
- "Bank Transfer"
- "Apple Pay" / "Google Pay"

The user selects their desired payment method and enters the amount. Meld's Smart Routing backend immediately evaluates this request against all 15+ upstream ramp providers and picks the one offering (unless previously set):

- Lowest fee
- Highest conversion rate
- Fastest settlement
- Highest KYC approval rate

Step 3: Amount and Fee Review
Meld displays:

- You Send: the amount to be sent
- Fee: varies by provider and payment method
- You Receive: amount to be received
- Estimated Time: "2-5 minutes" (for card payments) or "1-3 business days" (for bank transfers)

The user reviews and taps "Continue" or "Proceed to Payment".

Step 4: Payment Method Authentication and KYC
Depending on the payment method and the user's KYC tier:

For Credit/Debit Card:

1. The user enters card number, expiry, CVV (or uses saved card from previous transactions)
2. Meld's payment processor (Stripe, Payment Express, or the selected upstream ramp's processor) validates the card
3. If card is valid and user's account hasn't exceeded KYC limits, payment proceeds immediately
4. User may be asked to complete 3D Secure verification (SCA challenge) for additional security (this shows a popup or SMS verification)

For Bank Transfer (ACH, SEPA, or local equivalent):

1. Meld collects bank account details or initiates open banking authentication (user logs into their bank app to authorize the transfer)
2. User authorizes the debit from their bank account
3. Settlement time is longer: 1-3 business days (bank processing time)

For KYC-Gated Transactions:
If this is the user's first transaction or if the amount exceeds their KYC tier limit:

1. Meld prompts user to verify identity: "Please verify your identity to continue"
2. User provides name, date of birth, ID photo, and proof of address
3. Meld's KYC provider (typically Jumio, Fractal, Onfido, or similar) performs automated verification: document scanning, liveness check, address verification
4. Approval typically takes 30 seconds to 5 minutes (automated); if flagged, manual review adds 24-48 hours
5. Upon approval, payment proceeds; if declined, user is notified with reason

Step 5: Payment Processing
The chosen upstream ramp provider processes the payment. For card payments, this is near-instantaneous. For bank transfers, it's queued for overnight or next-business-day processing.

Step 6: USDC Arrival in Wallet
Once the upstream ramp provider has received and verified the fiat payment, they mint USDC and send it to the user's wallet address (which Meld knows from Social. backend).

The backend's webhook receives notification from Meld: "Transaction completed". The backend updates the user's balance cache and records the transaction.

Key On-Ramp UX Features:
Minimal friction: No leaving the app, no account creation with external provider (Meld handles this)
Smart routing: Best rate automatically selected without user action
Multiple payment methods: Users choose what's most convenient for them

Social. Infra Cost Estimate

Cheesecake Labs

19/11/2025

KYC handled smoothly: Verification happens once, then higher limits apply to future transactions
Real-time balance: Balance updates immediately upon USDC arrival
Transaction history: Permanent record of what was purchased and when

Social. Infra Cost Estimate

Cheesecake Labs

19/11/2025

Off-Ramp Process: Withdrawing Funds via Meld

When users want to convert USDC back to fiat currency, they use Meld's off-ramp service.

Step 1: Initiating Off-Ramp
The user taps the "Withdraw" button on their dashboard and Meld's widget loads again, this time in off-ramp mode.

Step 2: Meld Off-Ramp Widget Configuration
Meld's interface displays:

- You're sending: USDC (pre-filled, not editable)
- You'll receive: Currency selector (user's location detected)
- Amount input: User enters "50" (all available USDC, or partial amount)

Receiving method selector: Dropdown showing available options:

- "Bank Transfer" (to user's bank account)
- "Debit Card" (direct card top-up)

The user selects the desired method and enters the amount.

Step 3: Amount and Fee Review
Meld displays:

- You Send: amount in USDC
- You'll Receive: amount in user’s fiat currency
- Fees: varies by provider and destination
- Net Amount
- Estimated Time: varies by method and region

The user reviews and taps "Continue".

Step 4: Bank Account and KYC Verification
Meld collects the destination details and performs KYC/Compliance verification.
If the user's withdrawal amount exceeds their KYC tier, Meld may request additional verification

Meld's compliance system checks against sanctions lists, AML (Anti-Money Laundering) rules, and transaction patterns

If all checks pass, the user taps "Confirm Withdrawal".

Step 5: USDC Transfer to Meld's Wallet
Once confirmed, Social. backend initiates an on-chain USDC transfer from the user's wallet to Meld's collecting wallet address. This transaction:

- Deducts the amount in USDC from the user's wallet

- Sends the amount in USDC to Meld's contract address
- Takes 5-20 seconds to confirm on-chain
- Costs minimal gas (typically <$0.01 on Polygon)

The backend's webhook receives confirmation that the on-chain transfer succeeded.

Step 6: Meld Processes USDC and Initiates Fiat Payout
Meld receives the amount in USDC, verifies it arrived on-chain, and then:

- Routes to best off-ramp provider: Meld's aggregation logic picks the best off-ramp provider for this destination country and bank
- Initiates fiat transfer: Meld instructs the chosen provider to deposit to the user's selected method
- Batch settlement: Typically, Meld batches multiple user withdrawals and settles with the upstream provider once daily, reducing transaction costs

Step 7: Fiat Arrival and Confirmation
Within 2-5 business days (typical bank transfer time, faster on card), the user's bank account receives the fiat deposit.

Simultaneously, Social. backend receives a webhook from Meld: "Withdrawal complete".

Key Off-Ramp UX Features:

Minimal user action: Two clicks to initiate, then Meld handles complexity
Transparent fees: All fees shown upfront before confirmation
KYC efficient: Verification happens once, then higher limits apply
Bank integration: Direct fiat settlement to user's existing bank account (no middleman wallets)
Audit trail: Blockchain transaction hash and bank transfer reference both recorded for transparency and dispute resolution
Real-time balance: USDC balance updates immediately upon on-chain transfer (before fiat arrives)

Social. Infra Cost Estimate

Cheesecake Labs

19/11/2025

Postponing Yield Generation: Technical, Operational, and Strategic Rationale

Postponing the DeFi yield generation feature to Phase 2 represents a decision to de-risk the MVP, accelerate time-to-market, and ensure product-market fit validation before introducing the complexities of on-chain yield infrastructure.

Complexities

Smart Contract Interaction and State Management
Integrating yield generation requires managing a complex, multi-layer smart contract ecosystem. Such protocols consist of modular contracts handling different concerns: the Pool contract (core lending logic), interest rate strategy contracts (dynamic rate calculation), risk management contracts, and access control layers. Each interaction requires precise sequencing and gas optimization.

State Management Challenge
When a deposit is made using USDC to a lending pool, the wallets will receive tokens that represent ownership of underlying assets plus accrued interest. The protocol maintains a virtual balance layer internally, separating recorded balances from actual token transfers. Your backend must reconcile three distinct state representations: (1) the user's USDC balance, (2) the user's representation token balance held in Alchemy's smart account, and (3) the interest accrued since last update. Discrepancies between these layers, caused by blockchain reorganizations, failed transactions, or indexing delays, require manual reconciliation logic.

Smart Contract Audit Requirements and Liability
While lending protocols such as Aave V3 itself are extensively audited, any custom integration layer we build on top of them (custom withdrawal logic, automated rebalancing, liquidation detection) requires professional security audit. Typical costs: $15,000-50,000 per audit from reputable firms (Trail of Bits, Quantstamp, OpenZeppelin). Timeline: 2-4 weeks post-development, with mandatory fixes before mainnet deployment.

For MVP without custom contracts (only using the lending protocol public interface), the audit burden is lower, but you still inherit risk if your backend incorrectly calculates yields or mismanages user funds.

Regulatory and Compliance Complexity
Offering yield on deposits introduces regulatory ambiguity. In several jurisdictions. In the US, SEC has not provided clear guidance, but yield products may be classified as unregistered securities

Social. legal counsel must review the yield product before launch, which adds weeks to timeline and potential ongoing compliance costs ($5,000-20,000+ annually for ongoing legal review).