Protocol POC Requirements – StreamVault

- The protocol shall allow users to deposit USDC or any whitelisted SPL token to initiate a subscription

- The protocol shall create a unique on-chain escrow vault for each user-provider subscription pair

- The protocol shall allow funds to be streamed linearly over time or unlocked via milestone achievements

- The protocol shall allow providers to claim unlocked funds from the escrow vault

- The protocol shall allow users to cancel active subscriptions and reclaim unused funds

- The protocol shall charge a small protocol fee on each deposit (e.g., 50 basis points), sent to a separate fee vault

- The protocol shall support optional integration with guardian/arbiter accounts for dispute resolution

- The protocol shall maintain all stream metadata on-chain, including start time, duration, and milestone hash map

- The protocol shall expose a claim, cancel, initialize, deposit, and close instruction set for full vault lifecycle management

- The protocol shall enable integration with off-chain bots or platforms (e.g., Discord/Twitter) for access gating or payout automation.

StreamVaultProgram

BotInterface — Solana OnChainProgram — ClockworkThread

User
- initialize_vault → initialize_vault
- deposit_funds → deposit_funds
- cancel_subscription → cancel_subscription

Provider
- withdraw_unlocked_funds → withdraw_unlocked_funds

Anyone
- close_vault → close_vault

User Side

User's Wallet (ATA)

Call: initialize_vault

Call: deposit_funds: Transfer USDC

Program

initialize_vault: Creates VaultState PDA and VaultTokenAccount PDA

deposit_funds: Escrow USDC/SPL to VaultTokenAccount PDA

Create → VaultState (PDA)

Create → VaultTokenAccount (PDA)

Deposit funds → VaultTokenAccount (PDA)

```mermaid
flowchart TD
    A[User calls 'initialize_vault'] --> B[Creates VaultState & VaultTokenAccount PDAs]
    B --> C[User calls 'deposit_funds']
    C --> D[Funds (USDC/SPL) moved to VaultTokenAccount]
    D --> E{Has time/milestone unlocked funds?}
    D --> F{Does User cancel?}
    E -->|Yes| G[Provider calls 'withdraw_unlocked_funds']
    F -->|Yes| H[User calls 'cancel_subscription']
    G --> I[Unlocked funds moved to Provider's ATA]
    H --> J[Unused funds returned to User's ATA]
    I --> K{Vault fully drained?}
    J --> K
    K -->|Yes| L[Anyone calls 'close_vault']
```

```mermaid
Anyone
  | call: close_vault
  v
Clockwork Automation (Optional) - Triggers milestone unlocks & close
  : Optionally triggers unlock/closure
  v
Bot Signing (Optional) - Signs/calls for user/provider
  : Signs/calls for users/providers
  v
Close Vault - Closes vault, reclaims rent
  | close accounts        | close accounts
  v                       v
VaultState (PDA)          VaultTokenAccount (PDA)
```

User

User's Wallet (ATA)

Call: cancel_subscription → Refund unused funds

Program

cancel_subscription\nUser cancels early

Update state, set cancelled → VaultState (PDA)

Sweep remaining → VaultTokenAccount (PDA)