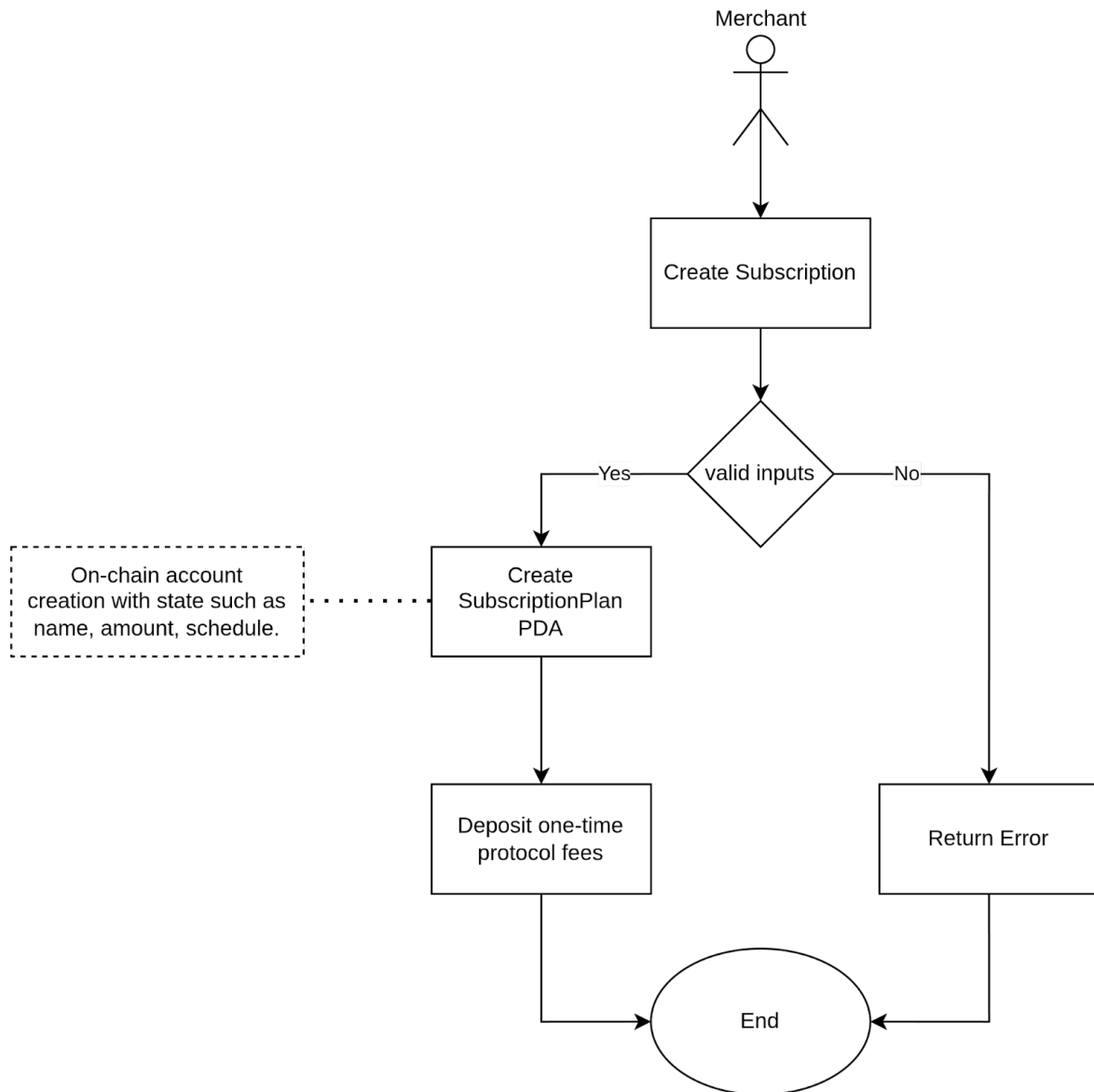


## Protocol POC Requirements

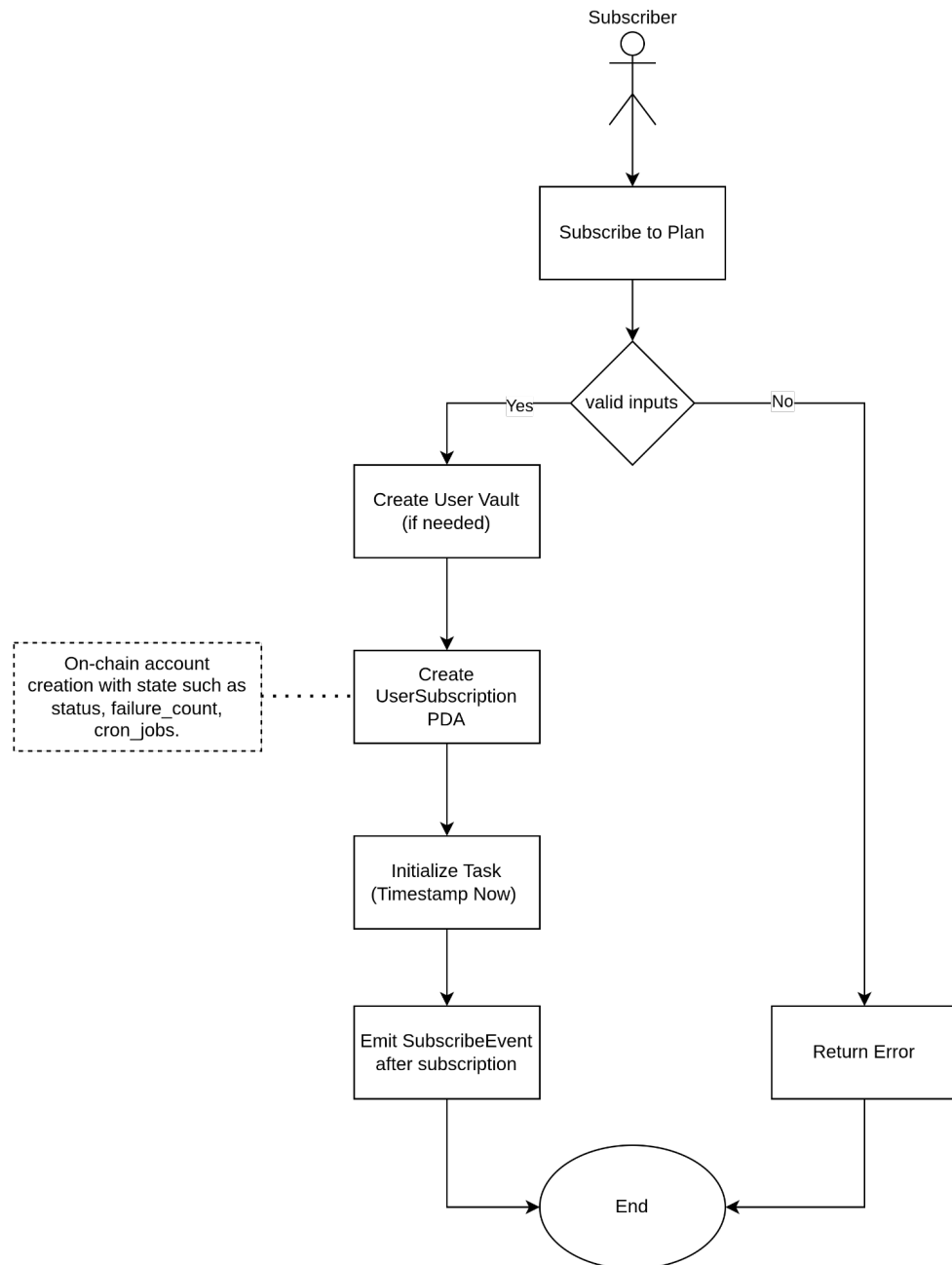
- The protocol shall allow a Merchant to define and initialize a SubscriptionPlan
- The Protocol shall require the Subscriber to perform a single, initial authorization step, granting the Protocol the right to spend tokens directly from their vault.
- The Protocol shall guarantee that the Subscriber retains the right to revoke this payment authorization at any time, instantly and permanently halting future deductions.
- The Protocol shall use a dedicated Automation Service to monitor all Subscription Contracts and act as the trusted clock for billing.
- The Protocol shall ensure this transfer occurs without any required signature or interaction from the Subscriber at the time of deduction.

## Create Subscription Plan



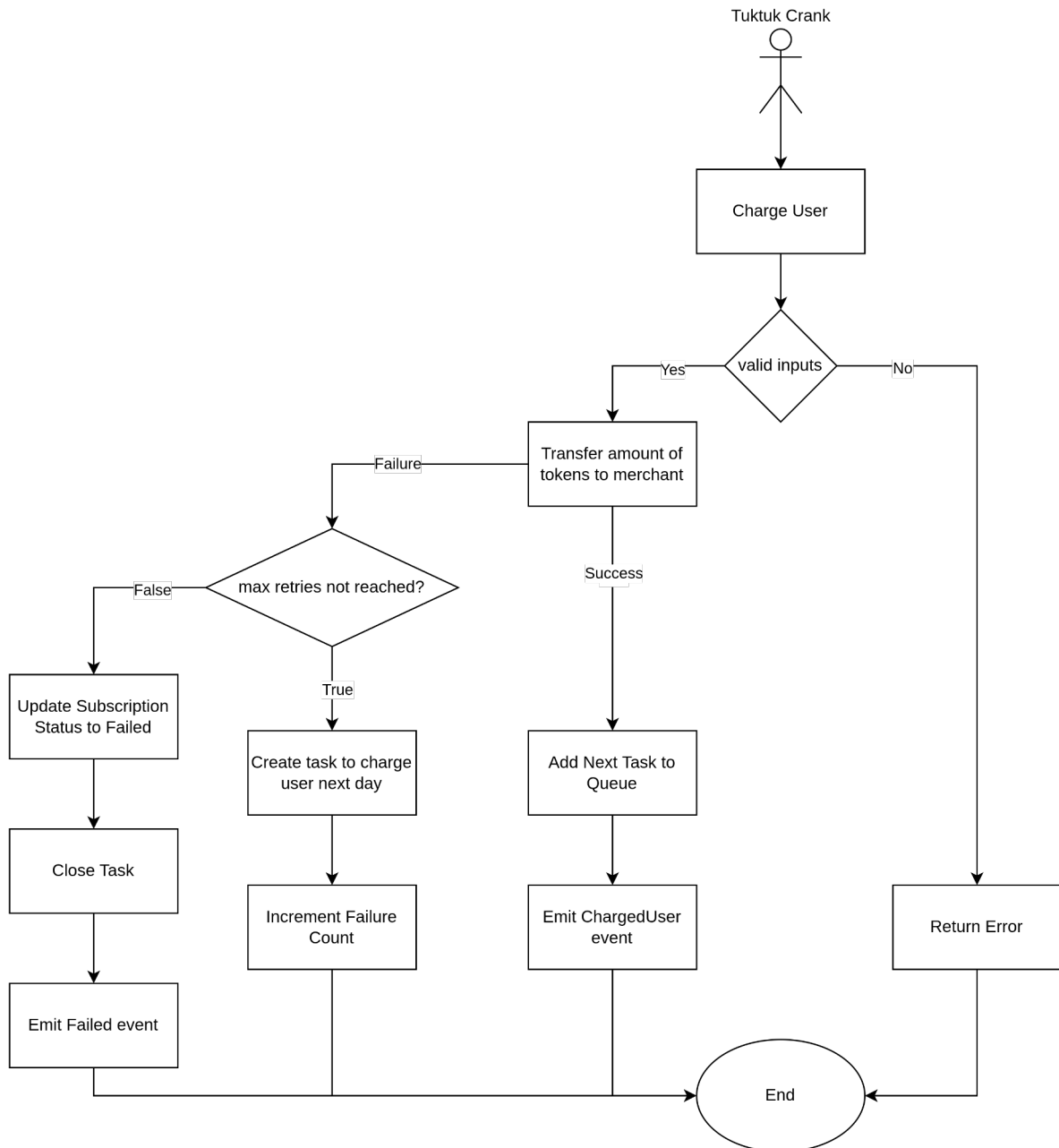
- The Merchant must successfully create a new, unique Subscription Schema (a specific SubscriptionPlan PDA).
- The Schema must store the name, amount, mint and schedule for the plan.
- **SubscriptionPlan PDA:** seeds: [b"plan", merchant.key.as\_ref(), hash(name.as\_bytes()).as\_ref()]
- **SubscriptionPlan Schema:** merchant, mint, amount, active, max\_failure\_count, name, schedule

# Subscribe to a Plan



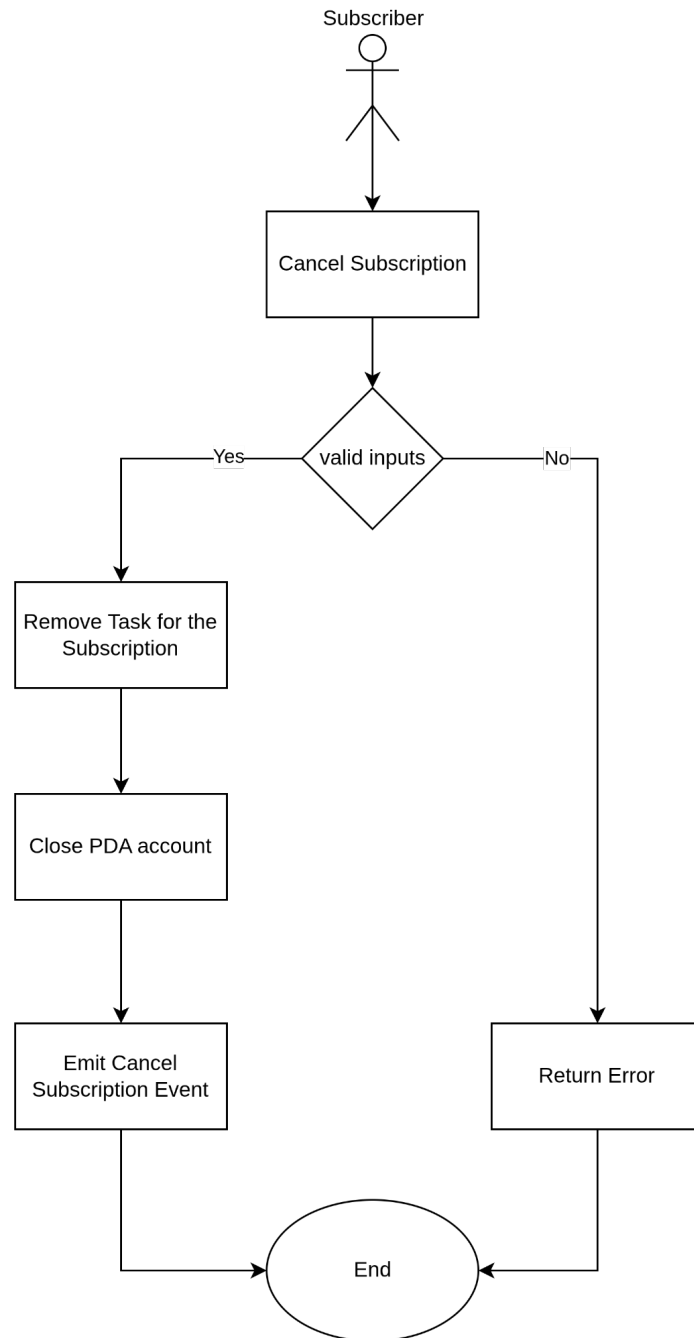
- The Subscriber must successfully execute the protocol's subscribe instruction, creating a unique UserSubscription PDA and linking it to the chosen plan.
- The Subscriber's dApp experience must clearly prompt them to create a vault with authority to the Protocol's Service Authority (PDA).
- **UserSubscription PDA: Seeds:** [SUBSCRIPTION\_SEED, subscriber.key.as\_ref(), subscription\_plan.key().as\_ref()]
- **UserSubscription Schema:** Subscriber, Plan, Status, Next Job

# Charge User



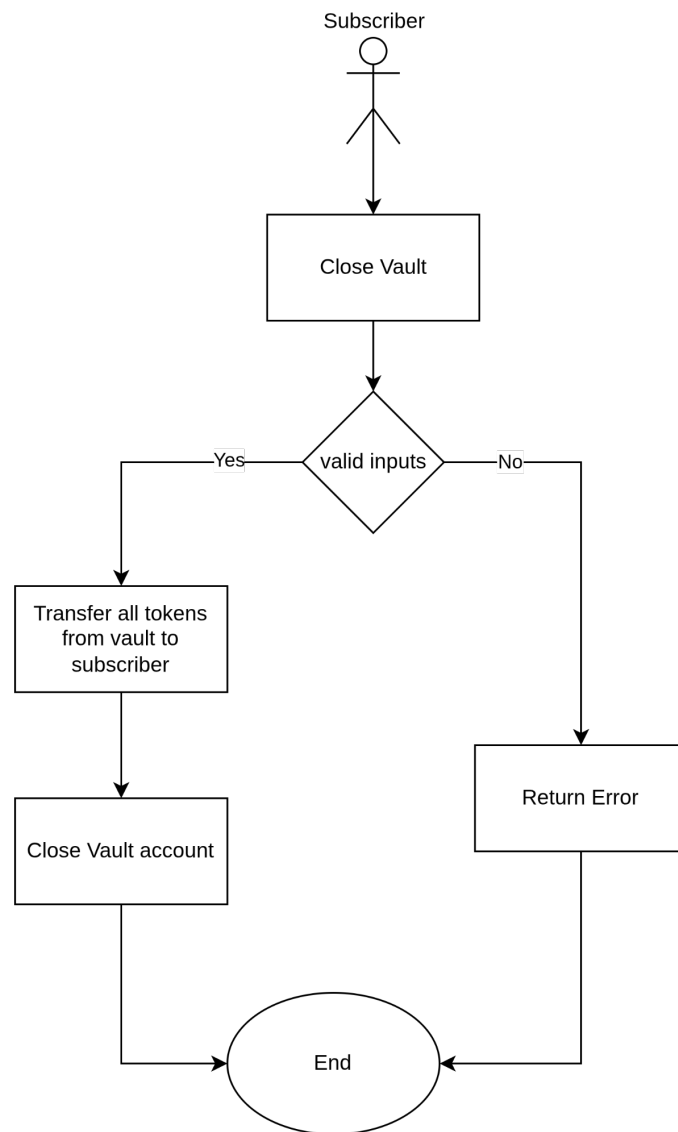
- The Automation Crank (Tuktuk) must successfully call the charge\_user instruction.
- The protocol must successfully add charge tasks to the queue, and allow it only when the time is on or after the due date using Tuktuk.
- Upon successful payment, the protocol must instantly update the subscription record by advancing the next\_payment\_date by the required frequency.

# Cancel Subscription



- The Subscriber must be able to call a `cancel_subscription` instruction, instantly marking the subscription record as inactive.
- The Instruction must remove the Tasks from the Task Queues and Close the PDA account.

## Close Vault



- The Subscriber must be able to close his vault and claim the remaining balances from the protocol

# End to End Overview

