# ADR-003: PSD2 Pass-Through Model

# ADR-003: Adopt PSD2 Pass-through Model (No Wallet)

**Status:** Accepted **Date:** 2026-02-12 **Deciders:** Alem (CEO), John (AI Director) **Category:** Architecture

---

## Context

The original Drop codebase implemented a **wallet model** where:

- Users had a local balance stored in the `users` database table
- Users could "top up" their wallet via `/api/users/top-up` (no payment verification)
- Transactions deducted from local balance
- Drop effectively held customer funds

This wallet model had significant regulatory implications under Norwegian law:

| Aspect | Wallet Model (EMI) | Pass-through Model (PISP/AISP) |
|---|---|---|
| License type | E-money Institution (EMI) | PISP/AISP registration |
| Norwegian law | Finansforetaksloven | Betalingstjenesteloven |
| Initial capital | 350,000 EUR | 20,000-50,000 EUR |
| Timeline to license | 12-18 months | 6-12 months |
| Fund safeguarding | Required (segregated accounts or insurance) | Not needed |
| PCI-DSS scope | Full (card data stored) | Minimal (no card data) |

The alternative PSD2 pass-through model positions Drop as a Payment Initiation Service Provider (PISP) and Account Information Service Provider (AISP) where Drop **never holds customer funds**.

```
graph LR
    subgraph wallet["Wallet Model (Rejected)"]
        user1["User"] -->|"Top-up"| drop_wallet["Drop Wallet<br/>(holds funds)"]
        drop_wallet -->|"Pay"| merchant1["Merchant"]
        drop_wallet -->|"Send"| receiver1["Receiver"]
    end

    subgraph passthrough["Pass-through Model (Adopted)"]
        user2["User"] -->|"PISP: Initiate payment"| bank["User's Bank<br/>(holds funds)"]
        bank -->|"Execute transfer"| merchant2["Merchant"]
        bank -->|"Execute transfer"| receiver2["Receiver"]
        drop_pt["Drop<br/>(orchestrator)"] -.->|"AISP: Read balance"| bank
        drop_pt -.->|"PISP: Initiate"| bank
    end

    classDef rejected fill:#FFCDD2,stroke:#C62828
    classDef adopted fill:#C8E6C9,stroke:#2E7D32

    class user1,drop_wallet,merchant1,receiver1 rejected
    class user2,bank,merchant2,receiver2,drop_pt adopted
```

# Decision

**Drop adopts the PSD2 pass-through model.** Specifically:

1. **No wallet:** Remove all local balance, top-up, and fund-holding functionality
2. **AISP for balance:** User sees their bank account balance via Open Banking API (read-only). The `bank_accounts.balance` field stores a cached AISP read -- not a Drop-held balance
3. **PISP for payments:** Remittance and QR payments are initiated from the user's own bank account via Open Banking payment initiation with SCA
4. **No card storage:** Cards feature gated behind feature flags (all default `false`); future card issuance via PCI-compliant partner only
5. **BankID for SCA:** Strong Customer Authentication via Norwegian BankID replaces email+password for all financial operations

# Code Impact

| Feature | Wallet Model (removed) | Pass-through Model (current) |
|---|---|---|

| | | |
|---|---|---|
| Balance | Local `balance` column in `users` table | `bank_accounts.balance` = cached AISP read from bank |
| Top-up | `/api/users/top-up` endpoint | Removed -- no top-up needed |
| Remittance | Deduct from local balance | `POST /api/transactions/remittance` triggers PISP |
| QR Payment | Deduct from local balance | `POST /api/transactions/qr-payment` triggers PISP |
| Cards | Stored locally (PAN, CVV in DB) | Feature-flagged; future partner integration (token-only) |
| Auth | Email + password (single factor) | BankID OIDC for SCA |
| Transaction | Local DB update only | Local record + bank payment confirmation |

```
sequenceDiagram
    participant User
    participant Drop
    participant BankID
    participant Bank
    participant Recipient

    Note over User,Recipient: PSD2 Pass-through Remittance Flow
    User->>Drop: Initiate remittance (amount, recipient)
    Drop->>Drop: Fee disclosure (0.5%)
    Drop->>User: Show total cost + exchange rate
    User->>Drop: Confirm payment
    Drop->>BankID: SCA challenge (amount + payee)
    BankID->>User: Authenticate (BankID app)
    User->>BankID: Approve
    BankID->>Drop: SCA confirmed
    Drop->>Bank: PISP: Initiate payment
    Bank->>Bank: Debit user account
    Bank->>Drop: Payment status: processing
    Drop->>Drop: Record transaction (status: processing)
    Bank->>Recipient: Transfer funds (SEPA/SWIFT)
    Bank->>Drop: Payment status: completed
    Drop->>Drop: Update transaction (status: completed)
    Drop->>User: Notification: transfer complete
```

# Consequences

## Positive

- Lower regulatory barrier to market entry (PISP/AISP vs EMI license)
- Faster licensing timeline (6-12 months vs 12-18 months)
- Lower capital requirements (20-50K EUR vs 350K EUR)
- No PCI-DSS card data storage obligations
- No fund safeguarding requirements (no funds to protect)
- Simpler security model -- Drop cannot lose customer funds
- Users keep their money in their trusted bank until payment execution

## Negative

- Dependent on banking partner / BaaS provider for Open Banking API access
- User experience may be slower (bank confirmation for each payment vs instant local deduction)
- Cannot offer instant transfers (limited by bank processing times: 1-2 days SEPA, 2-4 days SWIFT)
- Revenue model changes: no float income from held funds
- BankID integration adds complexity and requires BankID Norge partnership

## Risks

- **Banking partner dependency:** If no Norwegian bank provides Open Banking access, Drop cannot function. Mitigation: SpareBank1 already pitched; Swan (BaaS) as backup provider.
- **UX friction:** Each payment requires bank authentication via SCA. Mitigation: BankID app provides smooth mobile flow; consider session-based consent for repeat payments within limits.
- **Corridor coverage:** PISP may not support all 30+ target countries directly. Mitigation: use licensed remittance partner for non-SEPA corridors.

# References

- System Context (C4 Level 1) -- Shows Drop's external system relationships
- Open Banking Integration -- AISP/PISP integration specification
- Security Architecture -- Security controls for pass-through model
- Compliance Status -- Regulatory compliance tracking
- Roadmap -- Phase 2 banking integration plan

- Original source: `comms/decisions/ADR-003-psd2-passthrough-model.md`

---

Revision #4
Created 2026-02-21 05:58:57 UTC by John
Updated 2026-05-23 10:56:35 UTC by John