

Transaction History Flow

Flow: Transaction History

Document: LLD-003 **Version:** 1.0 **Date:** 2026-02-21 **Author:** Frontend Architect (AI Agent)
Status: Draft **Scope:** Transaction list rendering, filtering, pagination, transaction detail view, receipt download, and dispute initiation

1. Overview

The transaction history view provides users with a chronological list of all their financial transactions (remittances and QR payments). The list supports filtering by type and status, date-based grouping, infinite scroll pagination, and drill-down to transaction detail with receipt download.

Key capabilities:

- Paginated transaction list (20 per page, max 50)
 - Filter by type: All, Remittance, QR Payment
 - Filter by status: Processing, Completed, Failed
 - Date grouping: I dag, I gar, Denne uken, Eldre
 - Transaction detail view with exchange rate info
 - Receipt download (JSON receipt)
 - Complaint/dispute initiation via `/complaints`
-

2. Transaction List Load + Filter + Detail View

2.1 Sequence Diagram

```
sequenceDiagram
    actor User
```

participant App as Drop App
(//transactions)
participant API as Drop API
participant DB as Database

User->>App: Navigate to /transactions
App->>App: useAuth() – verify authenticated

App->>API: GET /api/transactions?page=1&limit=20
API->>API: JWT verification (cookie/Bearer)
API->>DB: SELECT transactions WHERE user_id = ?
ORDER BY created_at DESC LIMIT 20
API-->>App: { data: [...], pagination: { page: 1, limit: 20, total: N } }

App->>App: groupByDate(transactions)
-> "I dag", "I gar", "Denne uken", "Eldre"
App->>App: Render grouped transaction list

User->>App: Tap filter "Overforinger"
App->>API: GET /api/transactions?type=remittance&limit=50
API->>DB: SELECT WHERE type = 'remittance'
API-->>App: { data: [...filtered...] }
App->>App: Re-group and render

User->>App: Scroll to bottom (infinite scroll)
App->>API: GET /api/transactions?page=2&limit=20&type=remittance
API-->>App: { data: [...more...], pagination: { page: 2 } }
App->>App: Append to list, re-group

User->>App: Tap transaction row
App->>API: GET /api/transactions/{id}
API->>DB: SELECT transaction with exchange_rate info
API-->>App: { data: { ...fullDetails } }
App->>App: Show transaction detail modal/page

User->>App: Tap "Last ned kvittering"
App->>API: GET /api/transactions/{id}/receipt
API-->>App: { data: { receipt } }
App->>App: Download/display receipt

User->>App: Tap "Klag" (dispute)
App->>App: Navigate to /complaints
with transaction context

3. Transaction List Components

3.1 Component Diagram

```
graph TD
  subgraph "Transaction History Page"
    PageHeader["PageHeader<br/>Back button + 'Transaksjonshistorikk' title"]
    FilterTabs["FilterTabs<br/>(Tabs component)"]
    TransactionList["TransactionList<br/>(scrollable area)"]
    LoadMore["LoadMore Trigger<br/>(infinite scroll sentinel)"]
  end

  subgraph "Filter Tabs"
    TabAll["Alle<br/>(all types)"]
    TabRemittance["Overforinger<br/>(type=remittance)"]
    TabQR["QR-betalinger<br/>(type=qr_payment)"]
  end

  subgraph "Transaction List Items"
    DateGroup["DateGroupHeader<br/>('I DAG', 'I GAR', etc.)"]
    TransactionCard["TransactionCard"]
  end

  subgraph "Transaction Card"
    TxIcon["TypeIcon<br/>(ArrowUpRight, ScanLine, Clock)"]
    TxInfo["TxInfo<br/>(name, type label)"]
    TxAmount["TxAmount<br/>(amount + status badge)"]
  end

  subgraph "Transaction Detail"
    DetailHeader["DetailHeader<br/>(back + title)"]
    DetailSummary["DetailSummary<br/>(type, status, date)"]
    AmountBreakdown["AmountBreakdown<br/>(send, receive, rate, fee)"]
    RecipientInfo["RecipientInfo<br/>(name, country, bank)"]
    ReceiptButton["ReceiptButton<br/>('Last ned kvittering')"]
    DisputeButton["DisputeButton<br/>('Klag')"]
  end
```

```

FilterTabs --> TabAll
FilterTabs --> TabRemittance
FilterTabs --> TabQR

TransactionList --> DateGroup
DateGroup --> TransactionCard
TransactionCard --> TxIcon
TransactionCard --> TxInfo
TransactionCard --> TxAmount

TransactionCard -->|tap| DetailHeader
DetailHeader --> DetailSummary
DetailSummary --> AmountBreakdown
AmountBreakdown --> RecipientInfo
RecipientInfo --> ReceiptButton
RecipientInfo --> DisputeButton

```

4. Filter Options

4.1 Filter Options Table

Filter	API Parameter	Value	Label (Norwegian)	Default
All transactions	(none)	—	Alle	Yes
Remittances only	<code>type</code>	<code>remittance</code>	Overforinger	No
QR payments only	<code>type</code>	<code>qr_payment</code>	QR-betalinger	No
Processing status	<code>status</code>	<code>processing</code>	Behandles	No
Completed status	<code>status</code>	<code>completed</code>	Fulfort	No
Failed status	<code>status</code>	<code>failed</code>	Mislykket	No

4.2 Pagination Parameters

Parameter	Type	Default	Min	Max	Description
<code>page</code>	int	1	1	—	Page number
<code>limit</code>	int	20	1	50	Items per page

5. Transaction Status Display Mapping

Status	Label (Norwegian)	Color	Icon	Background
<code>completed</code>	Fulfort	<code>#0B6E35</code> (green)	ArrowUpRight / ScanLine	<code>#F8FAFC</code>
<code>processing</code>	Behandles	<code>#D4A017</code> (gold)	Clock	<code>#FEF3C7</code>
<code>failed</code>	Mislykket	<code>#EF4444</code> (red)	X	<code>#FEF2F2</code>

5.1 Transaction Type Icons

Type	Icon	Color	Description
<code>remittance</code> (sent)	ArrowUpRight	<code>#0B6E35</code>	Outgoing international transfer
<code>qr_payment</code>	ScanLine	<code>#0B6E35</code>	QR code payment to merchant
<code>remittance</code> (processing)	Clock	<code>#D4A017</code>	Transfer in progress

5.2 Amount Display

Condition	Format	Color	Example
Outgoing (sent)	<code>-{amount} kr</code>	<code>#0F172A</code> (dark)	-2 000 kr
Incoming (received)	<code>+{amount} kr</code>	<code>#10B981</code> (green)	+5 000 kr
Processing	<code>-{amount} kr</code>	<code>#0F172A</code> (dark)	-3 000 kr

6. Date Grouping Logic

The `groupByDate()` function categorizes transactions into temporal groups:

Group	Label	Condition
Today	I DAG	<code>createdAt</code> is today
Yesterday	I GAR	<code>createdAt</code> is yesterday
This Week	DENNE UKEN	<code>createdAt</code> is within current week (Mon-Sun)

Group	Label	Condition
Older	Date string (e.g., "12. OKT")	All older transactions, grouped by date

7. Transaction Detail View

7.1 Remittance Detail Fields

Field	Source	Example
Transaction ID	<code>data.id</code>	tx_rem_a1b2c3d4e5f6g7h8
Type	<code>data.type</code>	Overforing
Status	<code>data.status</code>	Fullfort
Send Amount	<code>data.sendAmount</code> + <code>data.sendCurrency</code>	2 000 NOK
Receive Amount	<code>data.receiveAmount</code> + <code>data.receiveCurrency</code>	23 400 RSD
Exchange Rate	<code>data.exchangeRate</code>	1 NOK = 11.70 RSD
Fee	<code>data.fee</code>	10.00 NOK (0.5%)
Total Cost	<code>data.total</code>	2 010 NOK
Recipient	<code>data.recipientName</code>	Mama Jasmina
Destination	<code>data.recipientCountry</code>	Serbia
Created	<code>data.createdAt</code>	21. feb 2026 kl. 14:32
Completed	<code>data.completedAt</code>	21. feb 2026 kl. 14:35

7.2 QR Payment Detail Fields

Field	Source	Example
Transaction ID	<code>data.id</code>	tx_qr_a1b2c3d4e5f6g7h8
Type	<code>data.type</code>	QR-betaling
Status	<code>data.status</code>	Fullfort
Amount	<code>data.amount</code>	129 NOK
Fee	<code>data.fee</code>	1.29 NOK (1%)
Merchant	<code>data.merchantName</code>	Ahmetov Kebab

Field	Source	Example
Source Account	<code>data.fromAccount</code>	DNB
Created	<code>data.createdAt</code>	21. feb 2026 kl. 12:15

7.3 Receipt Endpoint

`GET /api/transactions/{id}/receipt` returns a structured receipt:

```
{
  "data": {
    "transactionId": "tx_rem_1",
    "date": "2026-02-21T14:32:00.000Z",
    "type": "remittance",
    "amount": 2000,
    "currency": "NOK",
    "fee": 10,
    "exchangeRate": 11.7,
    "receiveAmount": 23400,
    "receiveCurrency": "RSD",
    "recipient": { "name": "Mama Jasmina", "country": "RS" },
    "reference": "tx_rem_1",
    "status": "completed",
    "completedAt": "2026-02-21T14:35:00.000Z"
  }
}
```

8. Dispute Initiation

Users can initiate a dispute from the transaction detail view by navigating to the complaints page:

Step	Action
1	User taps "Klag" on transaction detail
2	Navigate to <code>/complaints</code> with pre-filled category = "transaction"
3	User fills subject and description
4	<code>POST /api/complaints { category: "transaction", subject, description }</code>

Step	Action
5	Confirmation: "Vi behandler klagen din innen 15 virkedager" (Finansavtaleloven 3-53)

9. Platform Differences

Feature	Web (/transactions)	Mobile (history.js)
Filter tabs	Tabs component (Alle, Overforinger, QR-betalinger)	Custom buttons (Alle, Sendering, QR)
Pagination	Infinite scroll with limit=50	FlatList with pull-to-refresh
Transaction detail	Inline expansion or modal	Separate screen (not yet implemented)
Receipt download	API call + browser download	Not implemented
Dispute link	Navigate to /complaints	Not implemented
Date grouping	groupByDate() utility	Same pattern
Refresh	Re-fetch on filter change	Pull-to-refresh via RefreshControl

10. Accessibility Considerations (WCAG 2.1 AA)

Requirement	Implementation
List semantics	Transaction list uses semantic list markup
Filter announcement	Active filter tab announced to screen readers
Amount polarity	Positive/negative amounts use text prefix (+/-) in addition to color
Status indication	Status uses both color AND text label (not color alone)
Tap targets	Transaction cards are full-width, min 48px height
Loading state	Skeleton placeholders shown during initial load
Empty state	"Ingen transaksjoner" message when list is empty
Keyboard nav	Tab cycles through filters, then transaction items

11. Cross-References

- **Transaction list API:** `GET /api/transactions` — See [API Reference](#)
- **Transaction detail API:** `GET /api/transactions/{id}` — See [API Reference](#)
- **Receipt API:** `GET /api/transactions/{id}/receipt` — See [API Reference](#)
- **Complaints API:** `POST /api/complaints` — See [API Reference](#)
- **Transaction schema:** `transactions` table — See [Database Schema](#)
- **Component overview:** See [component-overview.md](#)
- **Figma transaction history:** `mockups/figma-make-export/src/app/screens/TransactionHistory.tsx`
- **Web page:** `src/drop-app/src/app/transactions/page.tsx` — See [PAGES.md](#)
- **Mobile screen:** `src/drop-mobile/app/history.js` — See [MOBILE-APP.md](#)
- **QR payment flow:** See [flow-qr-payment.md](#)

Revision #4

Created 2026-02-21 05:58:54 UTC by John

Updated 2026-05-23 10:52:04 UTC by John