Registration & Onboarding Flow

Registration & Onboarding Flow -- Low-Level Design

Document: LLD-REGISTRATION **Status:** Approved **Last updated:** 2026-02-21 **Author:** Standards Architect **Applies to:** Drop v1.0 (PSD2 pass-through model) **User requirements:** Minimum age 18, Norwegian residency, valid BankID (from vilkar.html)

Overview

Drop's registration is simplified by using BankID as the sole authentication provider. There is no separate registration form -- user accounts are created automatically on first BankID login. The onboarding flow then guides the user through consent collection, KYC verification, and first bank account linking.

Key principle: Progressive disclosure. Users see only what they need at each step. Heavy verification happens in the background while the user explores the app.

Complete Registration Flow

```
sequenceDiagram
    participant User
    participant App as Drop App
    participant BFF as Next.js BFF / Hono API
    participant BankID as BankID OIDC
    participant Sumsub
    participant Bank as Nordic Bank (Open Banking)
    participant DB as PostgreSQL

    Note over User,DB: Step 1 -- BankID Authentication + Auto-Registration
```

```
    User->>App: Tap "Logg inn med BankID"
    App->>BFF: GET /api/auth/bankid (or /v1/auth/bankid/initiate)
    BFF->>BFF: Generate state + nonce
    BFF->>BFF: Rate limit check (10/min per IP)
    BFF->>App: { redirectUrl }
    App->>BankID: Open BankID authorize URL

    User->>BankID: Authenticate (BankID app/code device)
    Note over User,BankID: SCA: possession (device) + knowledge (PIN)

    BankID->>App: Redirect with ?code=&state=
    App->>BFF: GET /callback?code=&state= (or POST /callback)
    BFF->>BFF: Verify state vs cookie/session
    BFF->>BankID: POST /token (exchange code)
    BankID->>BFF: { id_token, access_token }
    BFF->>BFF: Verify ID token signature (JWKS)
    BFF->>BFF: Extract pid (fodselsnummer, 11 digits)
    BFF->>BFF: Parse DOB from pid
    BFF->>BFF: Verify age >= 18

    alt Age < 18
        BFF->>App: Error: "Du ma vaere minst 18 ar"
        App->>User: Show age restriction message
    else Age >= 18
        BFF->>BFF: SHA-256 hash pid -> national_id_hash
        BFF->>DB: SELECT user WHERE national_id_hash = ?

        alt Existing user
            BFF->>DB: Create session + JWT
            BFF->>App: Set cookie / return Bearer token
            App->>User: Redirect to /dashboard
        else New user (first login)
            BFF->>DB: INSERT user (kyc_status='approved', kyc_method='bankid',
auth_provider='bankid', password_hash='EIDONLY')
            BFF->>DB: Create session + JWT
            BFF->>App: Set cookie / return Bearer token
            App->>User: Redirect to /onboarding
        end
    end
```

```
Note over User,DB: Step 2 -- Consent Collection (Onboarding Screen 1)
App->>User: Show consent checkboxes
User->>App: Accept terms + privacy + data processing
App->>BFF: POST /api/consents (type: 'terms', granted: true)
BFF->>DB: INSERT consents (terms, ip_address, granted_at)
App->>BFF: POST /api/consents (type: 'privacy', granted: true)
BFF->>DB: INSERT consents (privacy, ip_address, granted_at)
App->>User: Optional: marketing consent checkbox
Note over User,App: Marketing consent is OPTIONAL per GDPR

Note over User,DB: Step 3 -- KYC Trigger (Background)
BFF->>Sumsub: Create applicant (name, DOB from BankID)
Sumsub->>BFF: applicant_id
BFF->>DB: Store applicant_id
Sumsub->>Sumsub: PEP + sanctions screening
Sumsub->>BFF: Webhook: screening results
BFF->>DB: INSERT screening_results

Note over User,DB: Step 4 -- Bank Account Linking (Onboarding Screen 2)
App->>User: "Koble til bankkonto"
User->>App: Select bank (DNB, SpareBank1, Nordea...)
App->>BFF: POST /api/bank-accounts/link
BFF->>Bank: Open Banking: AISP consent request
Bank->>User: Authorize AISP access (SCA)
User->>Bank: Approve
Bank->>BFF: AISP access token + account list
BFF->>DB: INSERT bank_accounts (from AISP response)
BFF->>Bank: GET /accounts/{id}/balances
Bank->>BFF: { balance, currency }
BFF->>DB: UPDATE bank_accounts SET balance = ?, is_primary = 1
BFF->>App: { bankAccounts: [...] }
App->>User: Show linked account with balance

Note over User,DB: Step 5 -- Onboarding Complete
App->>User: "Velkommen til Drop!"
App->>User: Redirect to /dashboard
```

Onboarding States

```
stateDiagram-v2
    [*] --> bankid_redirect : User taps "Logg inn med BankID"

    bankid_redirect --> bankid_auth : BankID authorize page
    bankid_auth --> age_check : BankID callback received

    age_check --> rejected_underage : Age < 18
    age_check --> user_lookup : Age >= 18

    rejected_underage --> [*]

    user_lookup --> existing_user : national_id_hash found
    user_lookup --> new_user : national_id_hash not found

    existing_user --> dashboard : Session created, redirect

    new_user --> user_created : Auto-register from BankID data
    user_created --> consent_collection : Redirect to /onboarding

    consent_collection --> consents_granted : Terms + privacy accepted
    consents_granted --> kyc_background : Sumsub screening starts

    kyc_background --> bank_linking : Screening runs in background
    bank_linking --> bank_consent : User selects bank
    bank_consent --> bank_authorized : AISP consent granted
    bank_authorized --> account_linked : Balance fetched

    account_linked --> onboarding_complete : First bank account linked
    onboarding_complete --> dashboard : Redirect to /dashboard

    state kyc_background {
        [*] --> sumsub_pending
        sumsub_pending --> screening
        screening --> kyc_approved : All clear
        screening --> kyc_review : Match found
        kyc_review --> kyc_approved : Cleared by compliance
        kyc_review --> kyc_rejected : Confirmed risk
    }

    dashboard --> [*]
```

Age Verification (18+)

Norwegian fodselsnummer (11-digit personal identification number) encodes the date of birth:

Digits	Meaning	Example
1-2	Day of birth (DD)	`15`
3-4	Month of birth (MM)	`03`
5-6	Year of birth (YY)	`95`
7-9	Individual number	`123`
10-11	Check digits	`45`

Century determination (from individual number, digits 7-9):

- 000-499: born 1900-1999
- 500-749: born 1854-1899 (historical) or 2000-2039
- 750-899: born 1854-1899 (historical)
- 900-999: born 1940-1999

Verification logic:

```
1. Extract DD, MM, YY from pid[0..5]
2. Determine century from pid[6..8]
3. Construct full birthdate
4. Calculate age = today - birthdate
5. If age < 18: reject with "Du ma vaere minst 18 ar for a bruke Drop"
```

Source: vilkar.html section 3 -- "Du ma vaere minst 18 ar og bosatt i Norge for a bruke Drop."

Norwegian Residency Check

BankID issuance inherently confirms Norwegian residency:

- BankID is issued only by Norwegian banks to their customers
- Norwegian bank accounts require Norwegian national ID (fodselsnummer or D-number)
- D-numbers are issued to foreign nationals with legitimate ties to Norway

Additional signals:

- Phone number prefix: `+47` (Norwegian)
- BankID issuer: Norwegian bank (from ID token `amr` claim)

Signal	Check	Enforcement
BankID ownership	Implicit -- only Norwegian residents have BankID	At authentication
Phone number	`+47` prefix	Optional validation at profile update
Postal address	Norwegian postal code	At bank account linking (from AISP data)

Consent Collection

Required Consents

Per GDPR Article 6(1)(a) and Article 7, explicit consent must be collected before processing personal data. The following consents are collected during onboarding:

Consent Type	Required	Legal Basis	Description	Withdrawable
`terms`	**Yes** (mandatory)	Contract (Art. 6(1)(b))	Terms of service acceptance	Account deletion required
`privacy`	**Yes** (mandatory)	Consent (Art. 6(1)(a))	Privacy policy acknowledgment	Account deletion required
`data_processing`	**Yes** (mandatory)	Consent (Art. 6(1)(a))	PSD2 AISP/PISP data processing consent	Revokes bank access
`marketing`	No (optional)	Consent (Art. 6(1)(a))	Marketing communications	Yes, at any time
`cookies_analytics`	No (optional)	Consent (Art. 6(1)(a))	Analytics cookies	Yes, at any time
`cookies_marketing`	No (optional)	Consent (Art. 6(1)(a))	Marketing/tracking cookies	Yes, at any time

Consent Checklist

#	Consent	Checkbox Text (Norwegian)	Default	Validation
1	Terms of service	"Jeg godtar Drop sine brukervilkar"	Unchecked	Must be checked to proceed

#	Consent	Checkbox Text (Norwegian)	Default	Validation
2	Privacy policy	"Jeg har lest og godtar personvernerklaringen"	Unchecked	Must be checked to proceed
3	PSD2 data access	"Jeg godtar at Drop leser kontoinformasjon og initierer betalinger via Open Banking"	Unchecked	Must be checked to proceed
4	Marketing	"Jeg onsker a motta nyheter og tilbud fra Drop"	Unchecked	Optional

Consent Storage

Each consent is stored in the `consents` table:

Column	Value	Purpose
`id`	`con_<hex16>`	Unique consent record ID
`user_id`	`usr_<hex16>`	References the user
`consent_type`	`terms`, `privacy`, `marketing`, etc.	Type of consent
`granted`	`1` (true) or `0` (false)	Current consent state
`granted_at`	ISO timestamp	When consent was granted
`withdrawn_at`	ISO timestamp or NULL	When consent was withdrawn
`ip_address`	Client IP	Proof of consent action (GDPR Art. 7(1))

Consent API

Endpoint	Method	Purpose
`GET /api/consents`	GET	List all user consents
`POST /api/consents`	POST	Grant or withdraw consent

Consent withdrawal flow:

1. User navigates to Settings > Privacy
2. Toggles marketing consent off
3. `POST /api/consents` with `{ consentType: "marketing", granted: false }`
4. Record updated: `granted = 0`, `withdrawn_at = now()`

5. User can re-grant at any time

First Bank Account Linking

After consent collection, the user links their first bank account via AISP:

```
sequenceDiagram
    participant User
    participant App as Drop App
    participant BFF as Drop API
    participant Bank as Nordic Bank

    User->>App: Select bank from list (DNB, SpareBank1, etc.)
    App->>BFF: POST /api/bank-accounts/link { bankId: "dnb" }

    BFF->>Bank: Open Banking: GET /authorize (AISP scope)
    Bank->>User: Show consent screen (SCA required)
    Note over User,Bank: "Drop vil lese kontosaldo og<br/>transaksjonshistorikk"

    User->>Bank: Approve (BankID in banking app)
    Bank->>BFF: Authorization code
    BFF->>Bank: POST /token (exchange code)
    Bank->>BFF: AISP access token (90-day validity)

    BFF->>Bank: GET /accounts (list user accounts)
    Bank->>BFF: [{ accountId, iban, name, type }]

    BFF->>Bank: GET /accounts/{id}/balances
    Bank->>BFF: { balance: 45230.00, currency: "NOK" }

    BFF->>BFF: INSERT bank_accounts
    BFF->>BFF: Set first account as is_primary = 1

    BFF->>App: { bankAccounts: [{ bankName: "DNB", balance: 45230, isPrimary: true }] }
    App->>User: "DNB koblet! Saldo: 45 230 kr"
```

Progressive Disclosure UX

The onboarding flow uses progressive disclosure to minimize upfront friction:

Step	Screen	User Action	Background Action
1	BankID login	Tap "Logg inn med BankID"	Auto-create account from BankID data
2	Consent	Check 3 mandatory boxes + optional marketing	Record consents with IP + timestamp
3	Link bank	Select bank, approve AISP	Sumsub KYC screening runs in parallel
4	Dashboard	Explore the app	KYC result webhook updates user status

Time to first value: ~2 minutes (BankID auth + consents + bank linking)

Deferred actions (not required during onboarding):

- Add remittance recipients (done when user first sends money)
- Merchant registration (done from Settings if user is a business)
- Notification preferences (defaults: push enabled, email enabled)
- Profile completion (BankID provides name and DOB; address is optional)

Error Handling

Error	Screen	User Message	Action
BankID timeout	Login	"BankID-tidsavbrudd. Prov igjen."	Retry button
Age < 18	Login	"Du ma vaere minst 18 ar for a bruke Drop."	No retry -- explain policy
BankID state mismatch	Login	"Noe gikk galt. Prov igjen."	Redirect to login
Consent not accepted	Onboarding	Cannot proceed without mandatory consents	Highlight unchecked boxes
Bank linking failed	Onboarding	"Kunne ikke koble banken. Prov igjen."	Retry or skip (link later)
Sumsub KYC rejected	Background	"Identitetsbekreftelse mislyktes. Kontakt oss."	Account limited until resolved

---

Cross-References

- KYC/AML Flow -- Detailed KYC verification and AML monitoring
- Login Authentication Flow -- BankID login implementation details
- Bank Account Linking Flow -- AISP integration details
- Authentication System -- JWT, sessions, BankID OIDC
- BankID OIDC Integration -- BankID technical specification
- API Reference -- Consent and auth endpoints
- Database Schema -- users, consents, bank_accounts tables
- ADR-007: BankID OIDC Auth -- Authentication provider decision
- ADR-003: PSD2 Pass-through -- Pass-through model context

Revision #5
Created 2026-02-21 05:58:51 UTC by John
Updated 2026-06-28 20:01:35 UTC by John