

Test Plan: Drop — Fintech Payment App

Test Plan: Drop — Fintech Payment App

“ **Project:** Drop — Remittance + QR Payments **Version:** 1.0 **Date:** 2026-02-23
Author: John (AI Director) **Status:** Approved **Reviewers:** Alem Bašić (CEO)

Document History

Version	Date	Author	Changes
0.1	2026-02-23	John	Initial test plan — all MVP modules

1. Test Objectives

This test plan covers testing for **Drop MVP + Phase 0.5 Security Hardening** (v0.5.0).

Primary objectives:

1. Verify that all authentication and onboarding flows (registration, OTP, PIN, login) work correctly for Norwegian residents (age \geq 18, phone +47)
2. Verify that remittance transactions apply correct 0.5% fee across all 6 NOK corridors with mock BaaS
3. Verify that QR payments apply correct 1% merchant fee with mock BaaS
4. Confirm the pass-through model invariant: Drop NEVER stores user balances or full card data
5. Confirm Phase 0.5 security hardening: bcrypt 12 rounds, persistent rate limiting, CSRF, security headers, audit logging

6. Validate performance under expected load (40+ concurrent users; target 200 for Phase 1)

Out of scope for this plan: BankID SCA (Phase 2), real BaaS payments (Phase 2), real Sumsud KYC (Phase 2), Cards feature (Phase 3), mobile native app (Phase 2).

2. Features Under Test

Feature / Story	Priority	Test Types	Owner
User Registration — 3-step (FR-001)	Critical	Unit, Integration, E2E	Builder + Validator
User Login (FR-002)	Critical	Unit, Integration, E2E	Builder + Validator
Remittance Transaction (FR-020)	Critical	Unit, Integration, E2E	Builder + Validator
Exchange Rates API (FR-021)	High	Integration	Builder + Validator
QR Payment — Consumer (FR-030)	Critical	Unit, Integration, E2E	Builder + Validator
Merchant Registration + QR (FR-031)	High	Unit, Integration	Builder + Validator
Rate Limiting (NFR-SEC05)	Critical	Integration	Builder + Validator
Input Validation / Security (NFR-SEC06)	Critical	Unit, E2E (input-chaos)	Builder + Validator
DB Compliance — No Balance/CVV (NF-AC-020/021)	Critical	Integration (db.test.ts)	Builder + Validator
bcrypt Hashing (NFR-SEC02)	Critical	Unit (auth.test.ts)	Builder + Validator
Performance Benchmarks (NFR-P01..P06)	High	Performance (api-benchmarks)	Builder + Validator
Feature Flags (FR-090)	Medium	Unit (feature-flags.test.ts)	Builder + Validator

3. Scope

In Scope

- Authentication module: registration, OTP verification, PIN setup, login, logout, `/api/auth/me`

- Remittance module: `POST /api/transactions/remittance`, `GET /api/transactions`, exchange rates
- QR payments module: `POST /api/transactions/qr-payment`, `POST /api/merchants`, `GET /api/merchants/me`
- Security middleware: rate limiting, CSRF, JWT validation, security headers
- Database compliance: schema assertions (no balance, no card_number, no cvv), FK constraints, transaction type enum
- Performance benchmarks: bcrypt timing, DB query latency, concurrent rate limit check throughput
- Regression testing of all 26 API routes
- Input validation: XSS, SQL injection, boundary ages, Unicode names, long passwords

Out of Scope

Item	Justification
BankID SCA integration	Phase 2 — not yet implemented
Real BaaS PISP/AISP payments	Phase 2 — mock mode only in MVP
Real Sumsb KYC webhooks	Phase 2 — auto-approved in MVP
Cards feature	Phase 3 — feature-flagged OFF
Mobile native app	Phase 2 — web only in MVP
Load testing > 200 concurrent users	Phase 1 migration to PostgreSQL required first

4. Test Schedule & Milestones

Milestone	Date	Responsible
Test plan approved	2026-02-23	John (AI Director)
Test environment ready (staging)	Before Phase 0.5 release	John (DevOps)
Test data seeded	Before E2E run	Builder agent
Unit + integration tests complete	Per PR (CI automated)	Builder agent
Playwright E2E authoring complete	Before Phase 0.5 release	Builder agent
Regression testing complete (all 26 routes)	Before Phase 0.5 release	Validator agent
Performance benchmarks run	Before Phase 0.5 release	Builder agent
UAT start (CEO walkthrough)	TBD — before Phase 1 launch	John
UAT sign-off	TBD	Alem Bašić (CEO)
Go/no-go decision	Before Phase 1 launch	Alem Bašić (CEO)

Milestone	Date	Responsible
Production release	Phase 1 (BaaS partner confirmed)	John (AI Director)

5. Resource Allocation

Resource	Role	Testing Activities	Availability
Builder Agent (Claude Sonnet)	Developer / QA	Unit + integration + E2E authoring	Per task
Validator Agent (Claude Sonnet, read-only)	QA Lead	Code review + test verification	Per task
John (AI Director)	Tech Lead	Test strategy, UAT coordination	Continuous
Alem Bašić (CEO)	Product Owner / UAT	CEO UAT walkthrough	TBD

6. Entry Criteria

Testing may begin when:

- Feature development is code-complete (all tickets in "Ready for QA")
- Unit tests passing ($\geq 100\%$ pass rate on unit + integration suite)
- Build artifact deployed to staging (<https://drop-staging.fly.dev/>)
- Staging environment is stable (health checks passing)
- Test data is seeded (`npm run db:seed`)
- Previous known blocking bugs resolved (Mission Control backlog reviewed)

7. Exit Criteria

Testing is complete when:

- All 14 test files execute cleanly
- $\geq 100\%$ of unit + integration tests pass
- All Critical and High test cases in AC-001-AC-092 pass
- Code coverage $\geq 80\%$ overall; 100% for auth + transaction paths

- All Playwright E2E tests passing on staging (user-flows, full-flows, input-chaos)
- Performance benchmarks meeting NFR-P01..P06 targets (api-benchmarks.test.ts green)
- DB compliance tests passing (db.test.ts: no balance, no card_number/cvv columns)
- UAT sign-off obtained from Alem Bašić (CEO) — or conditional approval documented
- Security audit score \geq 80/100 (post Phase 0.5 hardening)

Exceptional circumstances: If exit criteria cannot be met, a documented risk acceptance from Alem Bašić (CEO) is required.

8. Test Strategy Summary Per Type

Type	Approach	Tool	Owner	Gate
Unit	White-box — bcrypt, JWT, fee calc, validators	Vitest	Builder	Blocks merge
Integration	Real SQLite test DB — 26 API routes, DB schema	Vitest	Builder	Blocks merge
E2E	Critical journeys on staging — 3 Playwright projects	Playwright	Builder	Blocks release
Regression	All 26 routes via api-endpoints.test.ts	Vitest	Builder	Blocks merge
Performance	api-benchmarks.test.ts — bcrypt timing, query latency	Vitest bench	Builder	Warning → release
Security	<code>npm audit</code> + validation.test.ts + middleware.test.ts	Vitest + GitHub Actions	Builder	Blocks merge
DB compliance	db.test.ts — schema assertions	Vitest	Builder	Blocks merge
UAT	CEO business scenario walkthrough	Manual	Alem Bašić	Blocks Phase 1 launch

9. Test Environment Requirements

Environment	Purpose	URL	Access Needed
-------------	---------	-----	---------------

Local dev	Unit/integration	<code>http://localhost:3000</code>	Builder agent
Staging (Fly.io, Stockholm)	E2E, regression, UAT	<code>https://drop-staging.fly.dev/</code>	Team + Alem
Performance	Benchmarks	Local (api-benchmarks.test.ts)	Builder agent

Environment requirements:

- Staging must have `NEXT_PUBLIC_SERVICE_MODE=mock` (no real BaaS)
- Staging SQLite DB seeded with synthetic test data (no real PII)
- Monitoring enabled (Fly.io metrics)

10. Test Data Requirements

Data Category	Volume	Creation Method	Responsible
Test consumer accounts	3 (fresh, KYC-approved, KYC-pending)	<code>npm run db:seed</code>	Builder agent
Test merchant accounts	2 (registered, unregistered)	<code>npm run db:seed</code>	Builder agent
Test recipients (for remittance)	3	<code>npm run db:seed</code>	Builder agent
Edge case data (under-18, duplicate email, max amounts)	Defined per test	Vitest fixtures	Builder agent

Data cleanup: All test data removed after test run via Vitest `afterEach` teardown. Staging DB reset between major test runs.

11. Risk-Based Test Prioritization

Risk Area	Likelihood	Impact	Priority	Mitigation
Pass-through model violation (Drop stores balance)	Low	Critical	P1	db.test.ts always asserts no balance column
Authentication bypass	Low	Critical	P1	Full auth.test.ts suite + middleware.test.ts
Fee calculation error (wrong percentage)	Medium	Critical	P1	Unit tests for 0.5% and 1% fee calculations

Risk Area	Likelihood	Impact	Priority	Mitigation
Double-spend race condition	Low	Critical	P1	Transaction lock integration test
Rate limiter reset on server restart	Medium (was a bug)	High	P2	middleware.test.ts with persistent limiter
BaaS mock mode leaking to production config	Low	High	P2	CI check for <code>NEXT_PUBLIC_SERVICE_MODE</code> env var
SQLite concurrent write limit reached	High (at ~200 users)	Medium	P3	Phase 1: PostgreSQL migration

12. Dependencies & Assumptions

Dependencies:

- Staging environment provisioned and accessible at <https://drop-staging.fly.dev/>
- Mock BaaS and Mock Sumsb configured in staging environment variables
- Playwright installed in CI (`npx playwright install`)

Assumptions:

- Feature requirements will not change during the testing phase without John (AI Director) review
- All Builder agent PRs include tests alongside code
- Validator agent reviews test files before merge
- BaaS partnership not confirmed — mock mode accepted for MVP/staging

13. Defect Management Process

Bug tracker: Mission Control tasks + Slack #drop-bugs on alai-talk.slack.com **Severity levels:**

Severity	Definition	Resolution SLA
Critical	Financial invariant broken; auth bypass; data loss	Fix before release — no exceptions
High	Major feature broken; security finding; no workaround	Fix before release
Medium	Feature degraded; mock/workaround exists	Fix in next sprint
Low	Minor issue, cosmetic	Backlog

Bug lifecycle: Open → Assigned (Mission Control) → In Progress → Fixed → Verified by Validator → Closed
Triage cadence: On each PR/commit (CI-driven); daily for active test phase

14. Test Deliverables

Deliverable	Format	Due Date	Owner
Test plan (this document)	Markdown	2026-02-23	John (AI Director)
Test strategy	test-strategy.md	2026-02-23	John
Test cases (automated)	Vitest + Playwright test files	Per sprint	Builder agent
Test execution results	Vitest + Playwright CI reports	Per PR	CI
Performance test report	api-benchmarks.test.ts output	Per release	Builder agent
UAT sign-off	uat-signoff.md	Before Phase 1	Alem Bašić
Test summary report	Markdown (per release)	Per release	Validator agent

Related Documents

- [Test Strategy](#)
 - [Test Case Template](#)
 - [E2E Test Plan](#)
 - [Performance Test Plan](#)
 - [Definition of Done](#)
 - [UAT Sign-off](#)
 - [Testing Guide](#)
 - [Test Inventory](#)
-

Approval

Role	Name	Date	Signature
Author	John (AI Director)	2026-02-23	Approved (AI)

Role	Name	Date	Signature
QA Lead	Validator Agent	2026-02-23	Approved (AI)
AI Director (John)	John	2026-02-23	Approved
CEO (Alem)	Alem Bašić	TBD	

Revision #7

Created 2026-02-23 12:06:01 UTC by John

Updated 2026-05-31 20:03:26 UTC by John