# Naga Banking

## Pioneer Partnership:
## Final Report

May 2025

# 1 - Overview

This pioneer exercise demonstrates how **conditional payments in a digital euro can be combined with on-chain escrow of regulated stablecoins to achieve atomic, cross-asset settlement** between participants in a peer-to-peer (P2P) marketplace.

The Naga Banking suite provides (i) a smart-contract–based exchange, (ii) a bank server that bridges SWIFT/DESP messages with on-chain events, and (iii) a web wallet. Together they enable payments and exchanges of stablecoins, digital Euros, and other forms of money. In the primary scenario, a seller can lock stablecoins, a buyer reserves digital euros, and both legs can be settled atomically once compliance checks pass.

The exercise confirms that **conditional payments are a natural primitive to unlock PvP (and DvP) settlement between tokenised money, including tokenised deposits and central-bank money** while keeping each leg on its native infrastructure. However, it also highlights the particular trust assumptions of externally triggered conditional payments. This programmability model focuses on **external access** with limited **internal capabilities**.

# 2 - Scope

- **Focus.** The experiment targets the *"Atomic exchange digital euro vs stablecoin"* scenario. It investigates:
  - creation and cancellation of digital euro *reservations* (hold) via DESP API;
  - escrow of ERC-20 stablecoins in the `NagaExchange` smart contract;
  - event-driven orchestration that releases both legs when pre-defined conditions are met.
- **Out-of-scope.** Regarding the current API, funding operations were not tested as they came late in the development. More advanced features were not part of the scope, such as offline use of the digital euro, tiered holding limits, and privacy-enhancing techniques.
- **Objectives.** Validate technical feasibility, measure latency, identify compliance and UX frictions, and estimate incremental cost of integrating conditional-payment APIs into existing banking systems.

# 3 - General setup of the experiment

| Layer | Component | Technology | Role |
|---|---|---|---|
| **Smart contract** | `NagaExchange.sol` `Stablecoin.sol` | Solidity 0.8.26 (Hardhat) | List offers, Escrow |
| **Bank/Fintech server** | `server/*` | Node 18, Koa, Axios | Parses SWIFT messages, screens addresses, triggers smart-contract and DESP calls |
| **Digital Euro sandbox** | DESP API | ECB cloud | Creates holdings, reservations, and payments |
| **Exchange frontend** | `frontend/*` | Vanilla JS, ethers.js, milligram.css, | Exchange interface, visual backoffice |
| **Wallet** | `NagaPay` | Vanilla JS, pico.css, ethers.js | Display Balances, allow payments and conversions |
| **Other services** | Compliance Screening | External APIs | AML/KYC |

# 4 - Use case and innovation

The primary use case is a **regulated P2P exchange that lets retail users exchange stablecoins against digital euros in a single, risk-free operation.** Pain points addressed:

- Moving value between crypto-native assets and bank money involves off-chain reconciliation, creating settlement risk and latency.
- Users lack a unified wallet that shows both forms of money.
- Solutions often lack integrated compliance mechanisms.

**Innovation aspects:**

- **Atomic PvP across infrastructures.** Conditional payment in DESP holds the euro leg until on-chain confirmation arrives.
  **Programmable compliance.** The server queries screening API before releasing funds. Further rule-based conditionality could be considered.

- **Interoperable wallet.** A single interface displays token and digital euro balances side-by-side, and allow seamless payments.
- **Hybrid reference architecture.** Implements the 'hybrid programmability' model with standardised APIs, high on-chain capabilities and limited conditionality on the digital Euro side.
- **Tokenised deposit automation.** Prototype could extend to interest-bearing tokenised deposits issued by partner banks, with automatic conversion back to digital euro at payout time.

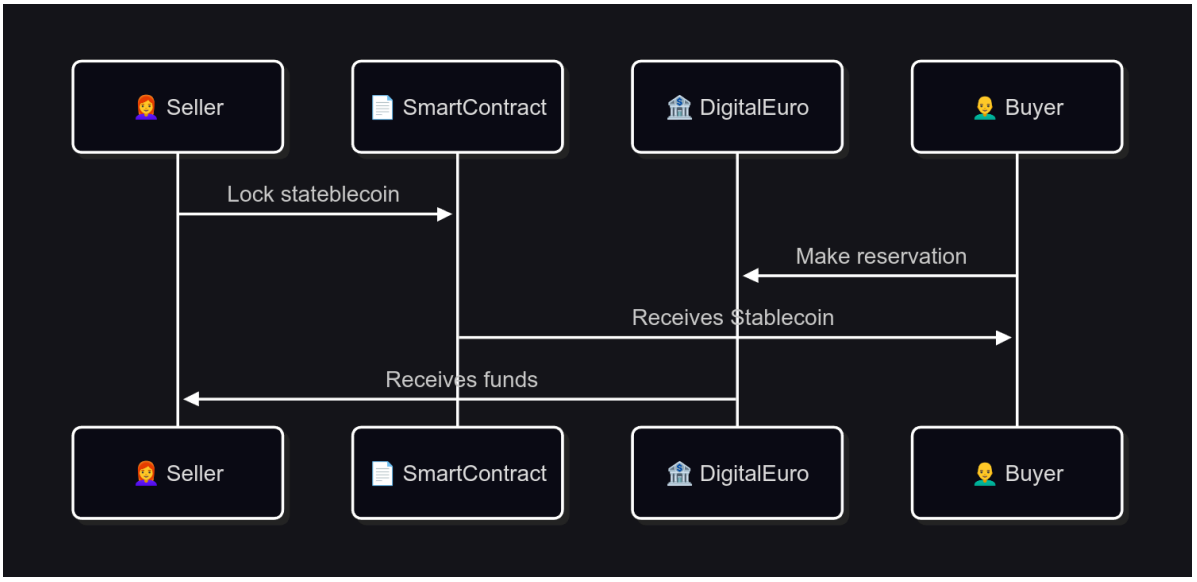
# 5 - Use-case implementation and execution

## Roles

- 👩‍🦰 **Seller** – locks stablecoin and awaits fiat leg.
- 👨‍🦲 **Buyer** – reserves digital euro and signals intent to purchase.
- 🚀 **Server** – verifies reservation & compliance and triggers settlement.

Interactions onchain are triggered directly by Buyers and Sellers using their own wallets, while interactions with DESP are triggered via the Server.

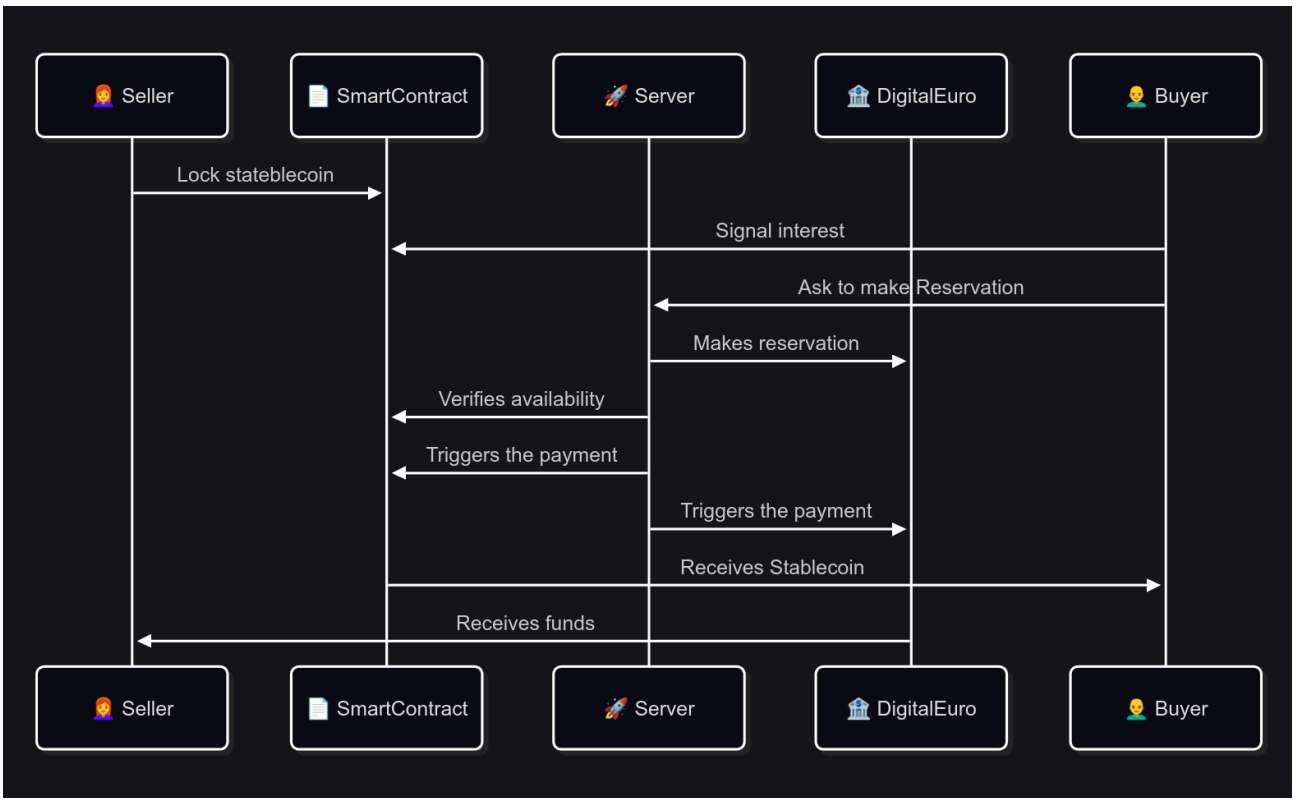
*Figure 1. Simplified representation of an atomic exchange*



## Process

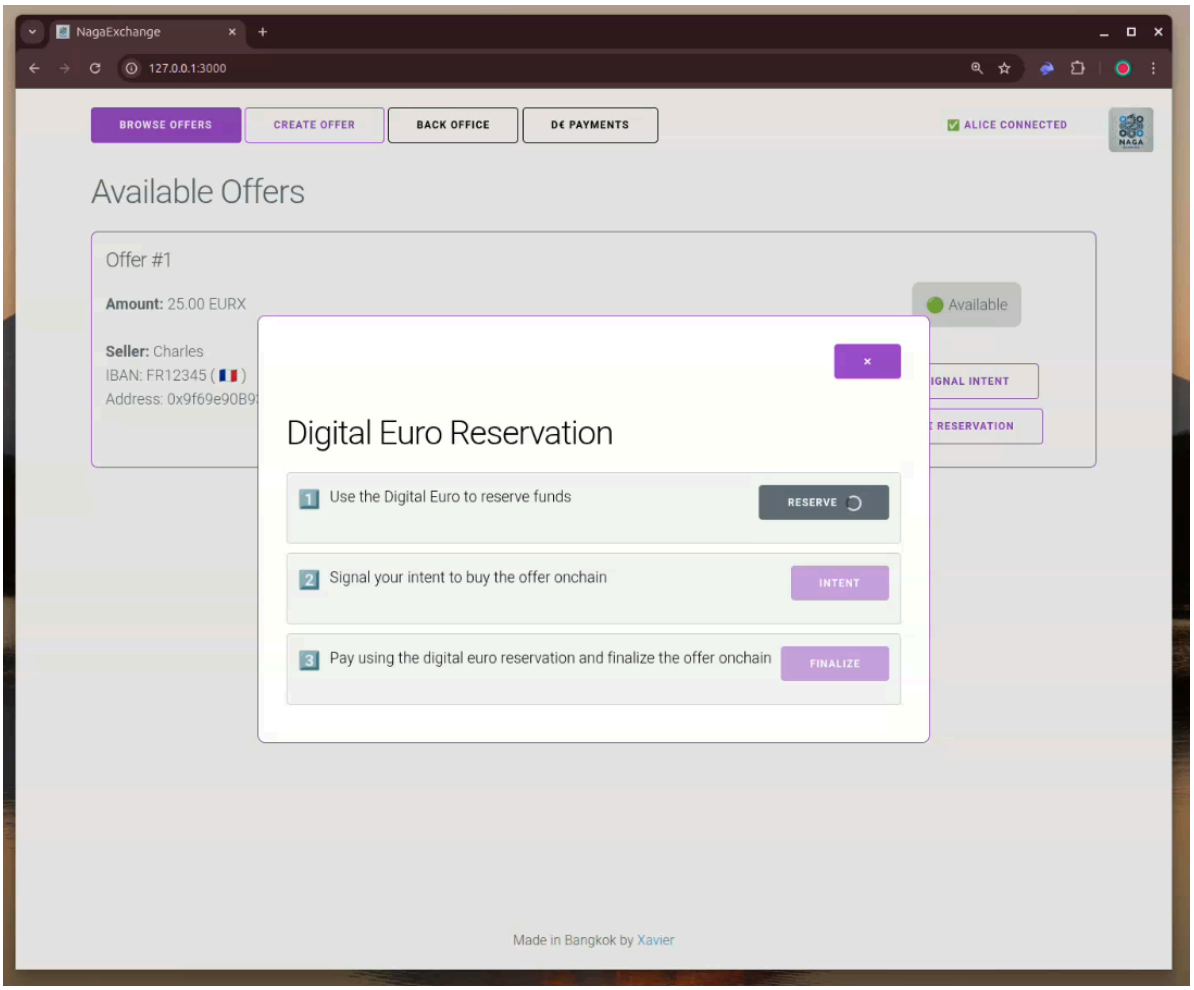
1. **Offer creation.** Seller calls `makeOffer(amount)` stablecoins are transferred to escrow and an `OfferMade` event is emitted.
2. **Reservation by buyer.** Buyer requests `createReservation` via the fintech server that interacts with the DESP (`interact.js`). It places a one-hour hold on digital euro funds.
3. **Compliance & Match.** The server compares reservation amount, offer ID, and IBAN; screens buyer address via Circle.
4. **Atomic release.** On approval, the server invokes `unlockFunds(offerID)` onchain **and** `createPaymentFromReservation` in DESP within the same process tick. Both operations are idempotent.

*Figure 2. Detailed workflow*



## Interfaces

*Figure 3. User interface for the exchange*




The interface comprises the P2P exchange and its backoffice, and an independent wallet.

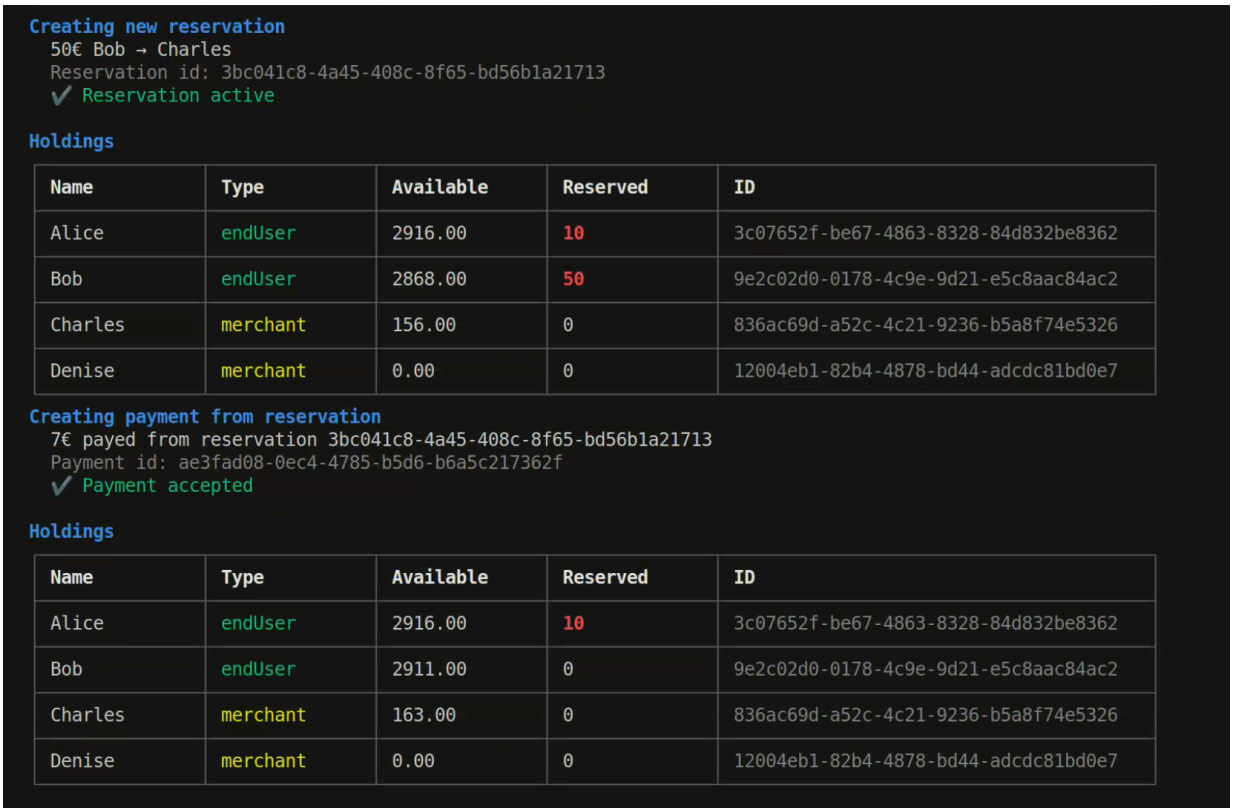
## Performance

Performance tests were conducted in a development environment that is not reflective of a production environment.

| Metric | Value |
|---|---|
| Avg. time from reservation to finality | 7.2 s (±1.1 s) |
| Failed settlements | 0 / 48 scenarios |

Regarding the API, light load experiments were conducted on the DESP API v1. The API resulted in 0 failures over 500 tests. The 95th-percentile latency sits at 320-390 ms. A slight increase (<10%) has been measured over the period March-May 2025.

*Figure 4. Command line interface for server side operations*




# 6 - Challenges, costs and recommendations

The documentation provided a clear explanation of the APIs. However, some useful features appeared late in the development process.

Key challenges are less technological but regulatory. What would be the requirements for using DESP features? Could there be light pure technological actors, with limited responsibilities as they would not have access to the funds directly?

Involved costs represent about 30 000€. It is not possible to draw the costs of a large-scale experiment. The compliance part is likely to be more important than the technology part.

## 7 - Expectations and possible implementation of a digital euro

Several elements are to be tested more extensively:

- Refactoring the server for the v2 API
- Testing funding mechanisms from various sources
- Complete compliance integrations with a full e-money/banking backend
- Offer a full wallet, with user focused UI/UX
- Large performance tests
- Other forms of tokenised asset integrations
- Confidentiality with fhEVM operations

## 8 - Demo

- **Video walk-through** of the offer UI and settlement flow (MP4, 45s). NagaBankingAndDigitalEuro.mp4

- **Open-source repository** `https://github.com/Xalava/NagaBanking`

## 9 - Key takeaways and lessons learned

- **Conditional payments usage.** Even with limited internal capabilities, conditional payments work as a bridge primitive between CBDC and tokenised forms of money.
- **Standardized API facilitates integration.** Wallets could be developed in record time with clean APIs for payments integrations.
- **Compliance and regulatory implications remain unclear.** As the technological barrier is lowered with such a solution, it could open new business models. However, the regulatory requirements for such businesses are unclear.

---

*Contact:* nagabanking@xvlv.io

*Prepared by:* Xavier Lavayssière