

## Documento de Requisitos: NexusPay Ledger Core

### Visão geral e definições técnicas

- **Core Ledger:** Sistema de registro imutável utilizando o design pattern Append-Only.
- **Definição de Saldo:** O saldo de uma conta é conceitualmente definido como a diferença entre o somatório de créditos e o somatório de débitos registrados no ledger de transações em um determinado instante ( $t$ ). No modelo adotado, o ledger é tratado como uma estrutura **imutável e append-only**, enquanto o saldo é persistido como um **campo mutável na tabela de contas**, funcionando como um snapshot do estado financeiro da conta.
- **Modelo Contábil:** Implementação estrita de Double-Entry Bookkeeping. Toda movimentação deve ter contrapartida, garantindo que a soma das entradas da transação seja zero.s

### Requisitos Funcionais:

- **RF01 – Onboarding de Wallet:** Registro de usuários com validação de CPF, e-mail (OTP), idade ( $18 < x < 130$ ), nome completo e senha.
- **RF02 – Multi-Currency:** Suporte a múltiplas moedas (BRL, USD, EUR) isoladas por sub-contas.
- **RF03 – Transações Atômicas:** Transferências entre carteiras devem ser processadas como uma unidade de trabalho atômica.
- **RF04 – Prevenção de Overdraft:** O sistema deve rejeitar transações de débito caso o saldo resultante da conta de origem seja inferior a zero.
- **RF05 – Idempotência:** Garantir que a mesma solicitação de transação, se enviada duas vezes, não resulte em duplicidade de débito.
- **RF06 – Segurança dos Dados:** Senhas devem ser salvas no banco de dados com criptografia segura e salt para garantir unicidade nas senhas e proteção contra tabelas rainbow. Senhas de acesso (login) serão alfanuméricas, com 8 caracteres, maiúsculas e minúsculas e ao menos um caractere especial, já as senhas pin de transação serão formadas por 6 dígitos numéricos.

### Requisitos Não-Funcionais:

- **RNF01 – Consistência Forte:** Operações de leitura após escrita devem refletir o estado mais recente imediatamente. Uso de isolamento de transação no banco de dados para evitar Lost Updates.
- **RNF02 – Concorrência:** Implementação de Optimistic Locking + técnicas de idempotência com retries, para maximizar a concorrência, garantir latências menores e alta segurança nas transações.
- **RNF03 – Latência:** p99 inferior a 200ms para operações de transferência.
- **RNF04 – Imutabilidade e Integridade:** Histórico de transações protegido por Cryptographic Chaining. Cada registro de transação devem conter um Hash com os dados da transação atual + Hash da transação anterior, tornando alteração retroativa detectável.
- **RNF05 – Disponibilidade:** 99,9%, com estratégia de deploy Blue-Green para evitar downtime.
- **RNF06 – Reconciliação Assíncrona:** Um Job que, periodicamente (uma vez ao dia), soma as entradas do ledger e compara com o saldo (snapshot atual) da tabela de contas. Se houver divergência, o sistema gera um alerta crítico.
- **RNF07 – Auditoria e monitoramento:** Monitoramento em tempo real da integridade do sistema e logs detalhados.