**Documentação do Projeto:**

**Sistema NDV - NoNameCoin Distributed Validation**

**1. Introdução**

O Sistema Distribuído de Validação por Consenso NoNameCoin tem como objetivo simular o processo de validação por consenso de uma moeda digital denominada NoNameCoin. Este documento fornece uma visão detalhada da arquitetura, funcionalidades e requisitos do sistema.

**2. Escopo do Projeto**

O sistema será desenvolvido em Python e será composto por três camadas principais: Banco, Seletor e Validador. Cada camada desempenha um papel específico no processo de validação por consenso da NoNameCoin.

**3. Arquitetura do Sistema**

O Sistema Distribuído NoNameCoin será arquitetado seguindo uma abordagem de camadas, com cada camada tendo responsabilidades claras:

* **Banco**: Distribui transações e informações de contas para o Seletor.
* **Seletor**: Cadastra validadores, verifica viabilidade, seleciona validadores para transações e gerencia o consenso.
* **Validador**: Valida transações recebidas do Seletor conforme as regras do ecossistema.

**4. Tecnologias Utilizadas**

* Linguagem de Programação: Python
* Frameworks/Bibliotecas: Flask (para a criação do sistema Web Service), SQLAlchemy (para interação com banco de dados), entre outros que possam ser necessários durante o desenvolvimento.

**5. Requisitos Funcionais**

* Distribuição de transações e informações das contas pelo Banco.
* Cadastro, verificação e seleção de validadores pelo Seletor.
* Validação de transações pelo Validador.
* Definição de rotas de API para acesso ao Banco, Seletor e Validador.
* Implementação de regras de validação e seleção conforme especificado no escopo do projeto.

**6. Requisitos Não Funcionais**

* Sistema Web Service.
* Algoritmo para eleição dos validadores.
* Funcionalidade de inserção e remoção de validadores.
* Armazenamento dos passos da eleição.
* Sincronização de tempo entre Seletor e Validador.
* Tolerância a falhas no Seletor e Validador.

**7. Detalhes de Implementação**

Os detalhes de implementação incluirão a lógica de negócios, estrutura de dados, algoritmos para seleção de validadores, regras de validação de transações, entre outros aspectos técnicos do sistema.

**8. Testes**

Os testes serão realizados para garantir que todas as funcionalidades do sistema atendam aos requisitos definidos e funcionem conforme o esperado.

**9. Gerenciamento de Configuração (Git)**

O Git será utilizado para controlar o versionamento do código, seguindo um fluxo de trabalho específico que inclui a criação de branches, revisão de código e merge para a branch principal.

**10. Implantação**

Os procedimentos para implantar o sistema em um ambiente de produção serão documentados, incluindo configuração de servidores, deploy da aplicação, entre outros aspectos relacionados à infraestrutura.

**11. Considerações de Segurança**

Serão consideradas medidas de segurança durante o desenvolvimento do projeto, incluindo autenticação de usuários, proteção contra ataques, entre outros.

**12. Manutenção e Suporte**

Após a implantação, serão estabelecidos planos para manter o sistema, incluindo atualizações de software, correções de bugs e suporte aos usuários.