

Arquitetura PLC Manager

Tecnologias:

- Python 3: Linguagem
- CustomTkinter 5.2.2: Interface gráfica
- opcua 0.98.0: Cliente OPC UA
- psycopg2-binary 2.9.0: Driver PostgreSQL
- requests 2.31.0: Cliente HTTP
- PyJWT 2.8.0: Decodificação de tokens
- python-dotenv 1.0.0: Variáveis de ambiente
- cryptography 41.0.0: Criptografia

Arquitetura:

- Aplicação desktop (GUI com CustomTkinter)
- Comunicação OPC UA com CLPs
- Integração com backend via REST API
- Persistência local em PostgreSQL

ESTRUTURA PRINCIPAL:

— main.py	# Aplicação principal
— container_manager.py	# Gerenciador de containers
— clp_container.py	# Container CLP individual
— opcua_connection.py	# Conexão OPC UA
— api_service.py	# Integração com backend
— auth_service.py	# Autenticação
— database.py	# Gerenciamento de banco local
— session_manager.py	# Gerenciamento de sessão
— command_processor.py	# Processamento de comandos
— login_window.py	# Janela de login
└— logger_config.py	# Configuração de logs

FUNCIONALIDADES PRINCIPAIS:

- Interface gráfica para gerenciamento de múltiplos CLPs
- Conexão OPC UA com CLPs S7-1200
- Sincronização de containers com backend (SmartCaldas)

- Monitoramento contínuo de variáveis
 - Descoberta automática de variáveis OPC UA
 - Leitura/escrita de variáveis em tempo real
 - Histórico de valores no banco local
 - Autenticação com backend (JWT)
 - Seleção de filial (branch)
 - Teste de conectividade OPC UA
-