

🎯 目的

Raspberry PiとBME280センサーを使い、物理世界のデータ（温度・湿度・気圧）を取得する基礎を学習。計測設計の考え方でデータの信頼性確保を体験します。※右図が全10回の演習で目指すシステム全体、その中の左列のIoT・センサの部分が第1回目

📋 到達目標

I2C通信の設定からセンサー値の読み取り、Pythonによるデータ加工（欠損処理・単位変換）、CSVへの保存、そしてグラフによる可視化までの一連の流れを実装できること。

🔧 手段・技術

Raspberry Pi

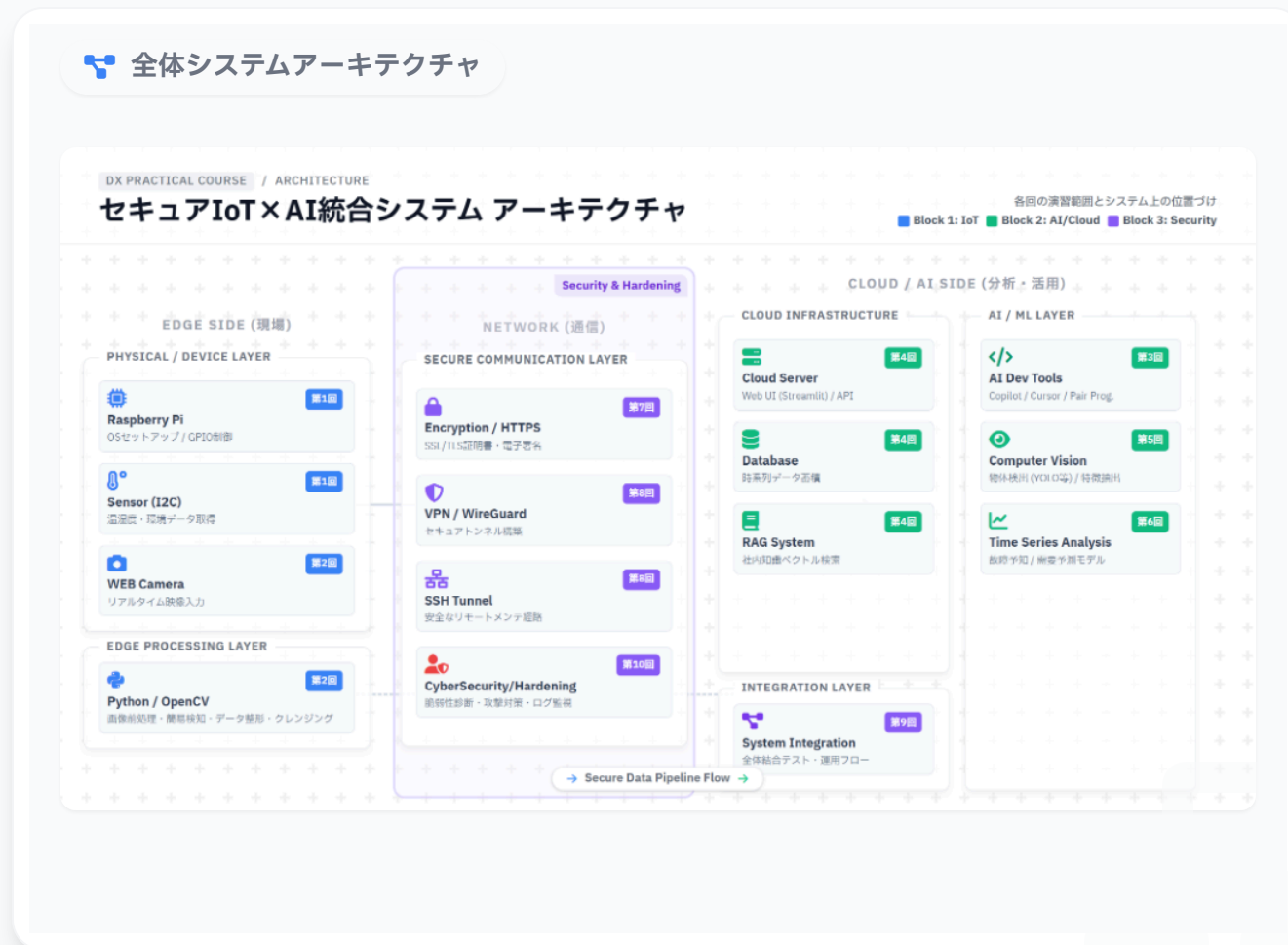
BME280

Python

smbus2

pandas

I2C / GPIO



Python・I2C・WEBカメラ・画像処理

🎯 目的

USBカメラからリアルタイム画像を取得し、OpenCVを用いた前処理・輪郭抽出・色ベースの簡易検出を行うことで、視覚的な情報処理を行うIoTパイプラインを構築します。

✅ 到達目標

フレーム取得、色空間変換 (BGR ↔ HSV)、平滑化、二値化、輪郭抽出、簡易分類を実装し、検出結果を画像上にオーバーレイ表示・保存できるようになること。

</> 手段・技術

Python OpenCV (cv2) NumPy UVC Webカメラ I2C

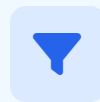
画像処理パイプライン

📷 画像処理パイプライン



カメラ入力

VideoCapture / Frame Read



前処理 (Pre-processing)

BGR to HSV GaussianBlur 二値化



検出・認識 (Detection)

findContours 面積判定 重心計算

🖥️ 画面表示

imshow / 矩形描画

💾 保存

imwrite / Log

時系列データ解析・予測

🎯 目的

センサーログ等の時系列データを対象に、ProphetやLightGBMを用いて需要予測や異常検知を実装。実運用に耐えるデータ前処理からバッチ推論までの基盤を整備する。

📌 到達目標

欠損値処理・季節性分解などの前処理を行い、特徴量を設計できる。予測モデルの学習・評価を経て、定期的に将来値を予測し可視化するバッチ処理を構築できる。

🔧 手段・技術

Prophet LightGBM pandas Airflow / cron Matplotlib

🔗 時系列解析パイプライン



🎯 目的

Difyを使用してノーコードでRAG（Retrieval-Augmented Generation）システムを構築。独自のナレッジベースを活用し、正確かつ専門的な回答を生成するAIチャットボットを短時間で実装・公開する。

📋 到達目標

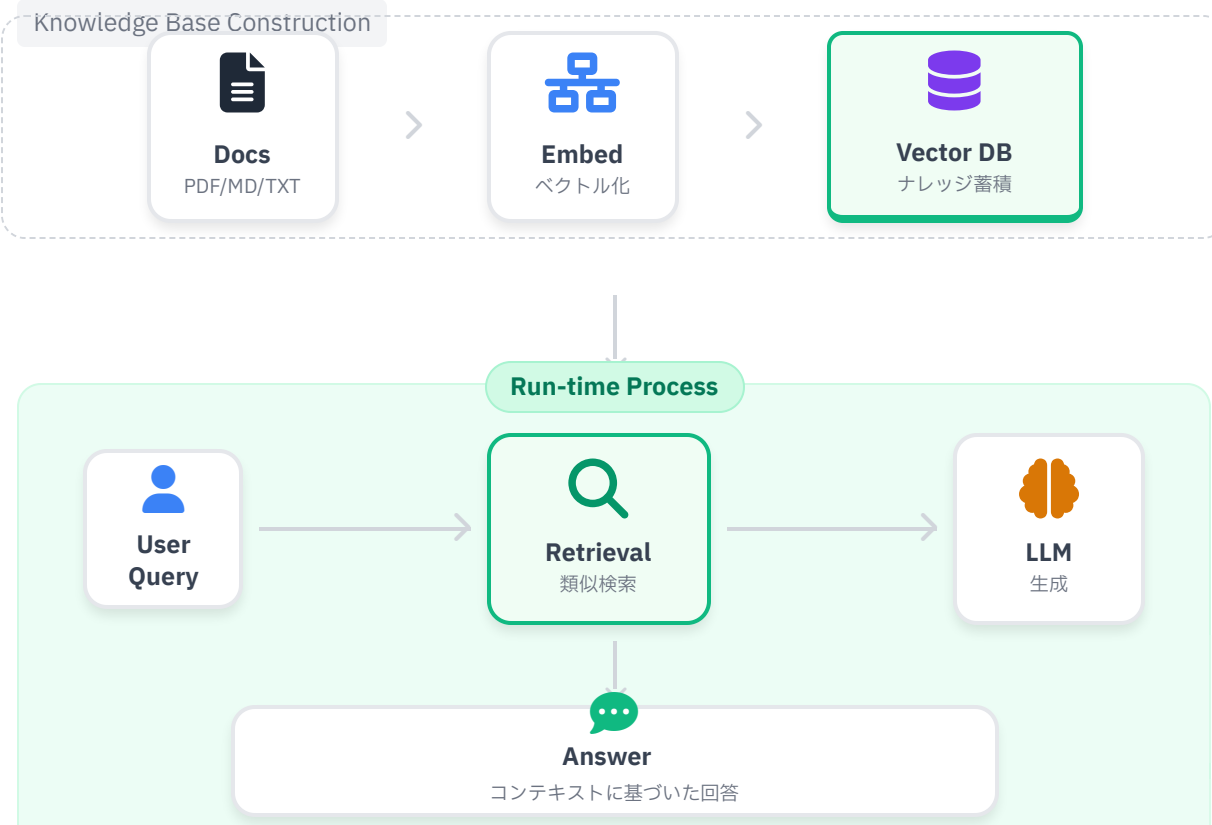
ドキュメントの分割・埋め込み設定からベクトルDBへの登録、検索パラメータの調整までを実施できる。LLMへのプロンプト設計を含めたパイプライン全体を構築し、APIまたはWebウィジェットとして公開できる。

🔧 手段・技術

Dify Vector DB LLM API Embedding Models

RAG Pipeline

🔄 RAGパイプライン構築フロー



🎯 目的

カスタムデータセットを用いてYOLOモデルを学習させ、推論APIをFastAPIで構築する。IoTデバイスからの画像入力に対し、軽量かつ低遅延で応答する「賢い目」を持つシステムを実現する。

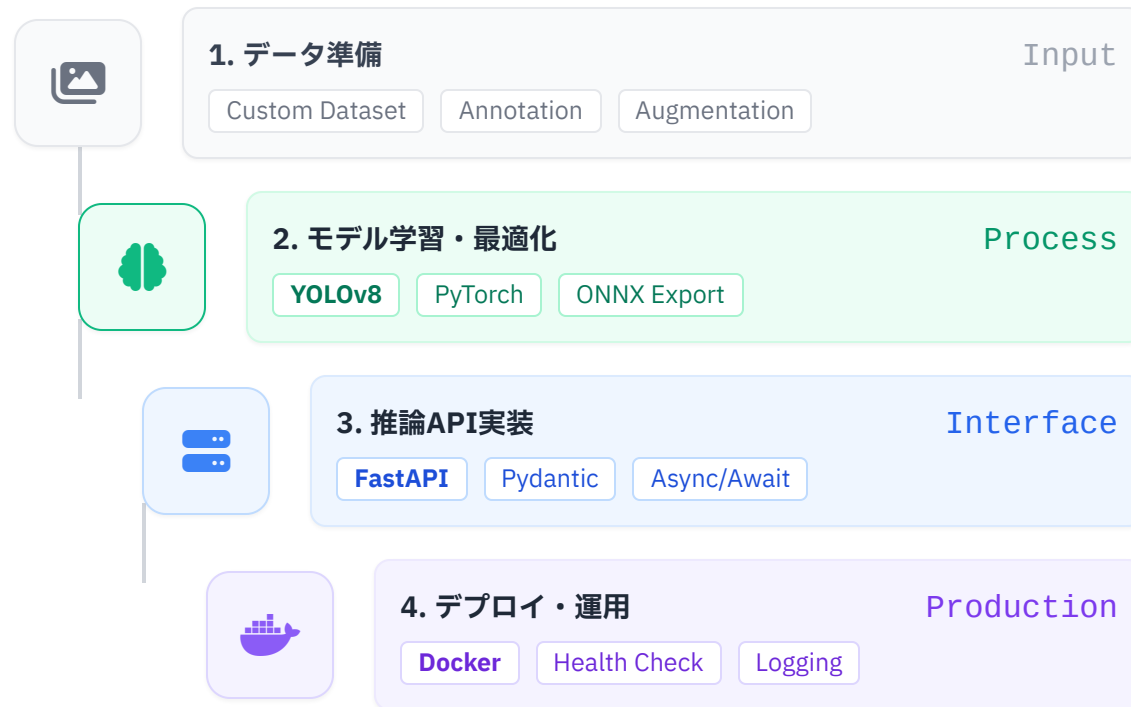
📋 到達目標

アノテーション済みデータからYOLOを学習し、評価指標（mAP）を確認。ONNX等への最適化を経て、FastAPIでエンドポイントを実装し、Dockerコンテナとしてデプロイできる。

🔧 手段・技術

YOLOv8 PyTorch FastAPI ONNX/量子化 Docker

📁 MLパイプライン & 技術スタック



AI開発支援ツール・ペアプログラミング

🎯 目的

VSCoideとCLINEを用いてAIペアプログラミングを実践。要件整理からコード生成、リファクタリング、レビュー、デバッグまでの一連の開発プロセスをAIと共に効率化する手法を習得する。

📋 到達目標

プロンプトデザインパターンを活用し、AIに適切なコンテキストを与えて設計補助やテスト生成を依頼できる。生成コードの根拠を確認し、Git管理下で安全に適用できるスキルを身につける。

🔧 手段・技術

VSCoide CLINE Claude 3.5 Sonnet Git

Prompt Engineering

🎵 AI協働開発フロー



🎯 目的

近代暗号の基礎と証明書の信頼モデル（Chain of Trust）を理解し、証明書発行からNginxでのHTTPS化、さらに自動更新までの一連のセキュア通信基盤を実装する。

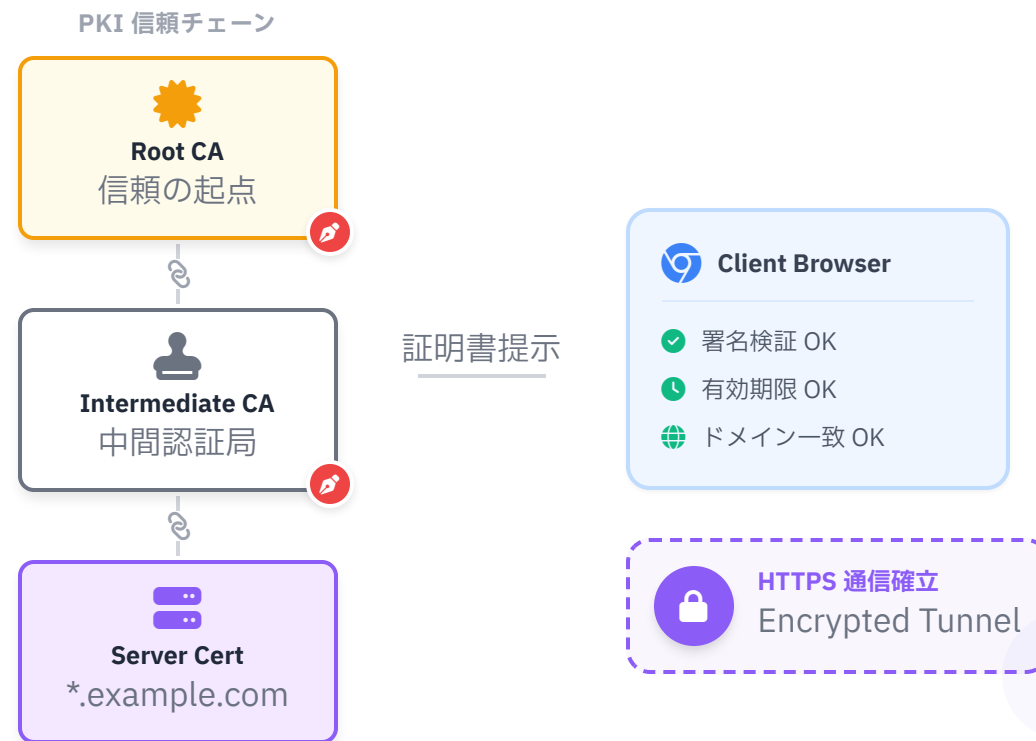
📋 到達目標

OpenSSLでのCSR・鍵生成、自己署名証明書の作成手順、およびLet's Encryptを用いた本番運用（ACME連携・自動更新）。NginxのTLS設定と暗号スイートの選定ができる。

🔧 手段・技術

OpenSSL Let's Encrypt Nginx Certbot TLS 1.2/1.3

🛡️ 信頼チェーンとHTTPS通信



SSHトンネル・VPN・WireGuard

🎯 目的

セキュアな遠隔通信基盤を整備し、SSHトンネル（ポートフォワードリング）やWireGuardを用いたVPNを構築する。さらにWiresharkでパケットを解析し、暗号化の効果を検証する。

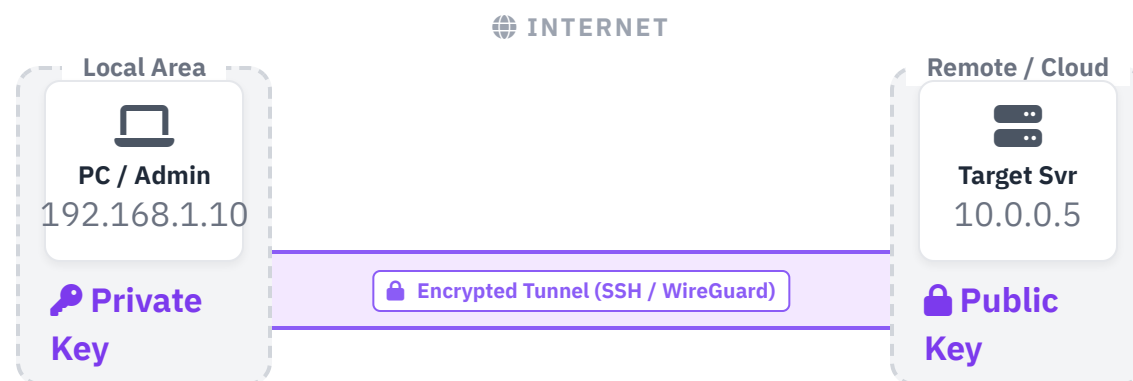
📋 到達目標

公開鍵認証によるSSH設定、WireGuardによるL3 VPN構築とルーティング設定ができる。Wiresharkを用いたパケットキャプチャで、平文と暗号文の違いを視覚的に確認できる。

🔧 手段・技術

WireGuard OpenSSH Wireshark iptables/UFW TCP/IP

🛠️ セキュアトンネルとパケット解析



🔍 Wireshark Capture: wg0

Time: 14:23:05

No.	Protocol	Info
124	WireGuard	Handshake Initiation, Sender: 0xA1B2...
125	WireGuard	Handshake Response, Sender: 0xC3D4...
126	UDP	[Encrypted Payload] Length: 148 bytes

セキュアIoT×AI活用システム統合

🎯 目的

IoTデバイス、エッジ、クラウド、AI推論、UI画面を疎結合アーキテクチャで統合し、エンドツーエンド（E2E）で連携する一貫したシステムを完成させる。

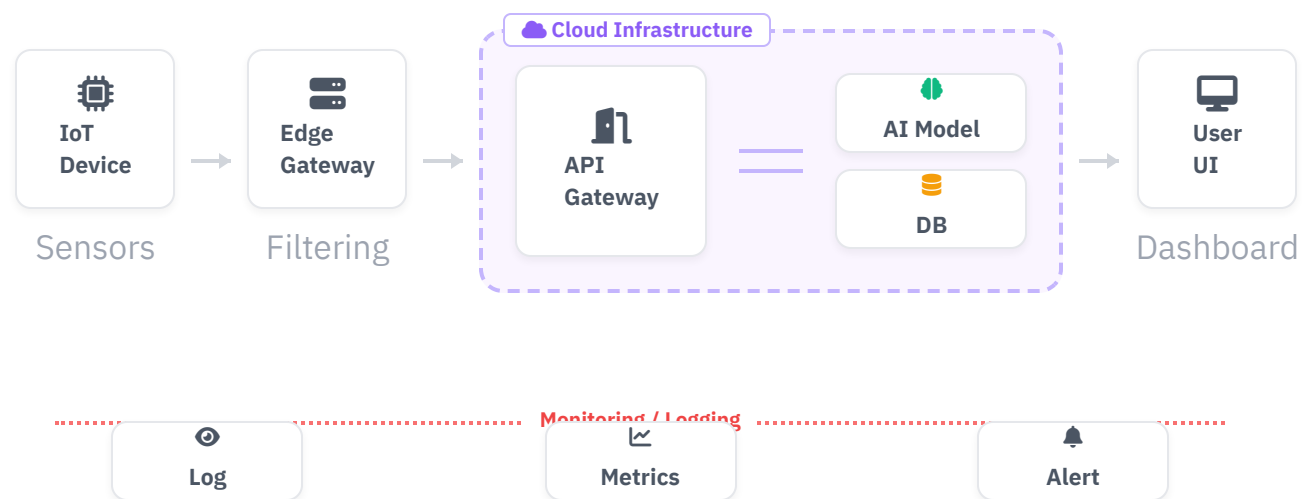
📋 到達目標

各マイクロサービスをAPIで接続し、全体の動作確認を行う。
 Docker Composeによる構成管理、負荷テストの実施、
 Prometheus/Grafanaを用いた監視設定ができる。

🔧 手段・技術

API Gateway MQTT Docker FastAPI Grafana

📦 疎結合システムアーキテクチャ



サイバーセキュリティ・ハードニング

🎯 目的

ペネトレーションテスト（侵入テスト）の手法を用いて自システムの脆弱性を能動的に発見し、ハードニング（堅牢化）対策を適用してシステム全体のセキュリティレベルを強化する。

📋 到達目標

NmapやMetasploitを用いた脆弱性診断を実施できる。また、Purple Team演習（攻撃と防御の協働）を通じて、発見された弱点に対するファイアウォール設定や設定変更などの防御策を実践できる。

🔧 手段・技術

Kali Linux

Nmap

Metasploit

Purple Team演習

UFW / iptables

Hardening

🛡️ サイバーキルチェーンと対策

PHASE 1
偵察 (Recon)
情報収集・OSINT

PHASE 2
スキャン (Scan)
ポート/脆弱性検知

PHASE 3
攻撃 (Exploit)
Metasploit実行

PHASE 4
侵入 (Intrusion)
シェル奪取・権限昇格



COUNTERMEASURE 対策・ハードニング

- ポート閉鎖・FW設定
- パッチ適用
- ログ監視強化

🔄 PDCAサイクルで継続的改善