

# IoT 技術を活用した中学生向け教材の開発

宮崎大学 工学部教育研究支援技術センター

○長友 耀希

## 1. はじめに

宮崎大学工学部では、JST 女子中高生の理系進路選択支援プログラムを開催している。本プログラムにおいて、2024 年 11 月 21 日に日南学園中学校の学生 38 名を対象に IoT システムについて学ぶ講座が開講され、Raspberry Pi とセンサーを用いたシステムの作成・体験を通じて IoT 技術の基礎を学習する内容の講座を実施した。その講座において、Raspberry Pi とセンサーを用いたシステムの作成・体験を通じた IoT 技術の基礎を学習する内容の教材を開発したのでその内容を報告する。

キーワード：Raspberry Pi, IoT 教育, リアルタイムデータ可視化, 環境データ

## 2. システム概要

作成したシステムの概要について説明する。本講座では、身近にある環境データが IoT 機器を用いることで簡単に計測・可視化できることを体験してもらうために簡易的なシステムを構築した。センサーは、温湿度センサーと距離センサーを用い、GUI を Python にて作成した。以下にその構成を記す。

### 2.1 使用機器

使用した機器（図 1）は以下のとおりである。

- ・Raspberry Pi 400
- ・マウス
- ・モニター
- ・温度センサー（DHT22）
- ・距離センサー（VL53L0X）
- ・抵抗付き LED
- ・ブレッドボード



図 1 使用機器一覧

### 2.2 システム説明

ユーザは、Raspberry Pi 上のターミナルにて Python プログラムを実行することで、システムの起動を行う。システム画面を図 2、3 に示す。センサー別に大きく 2 つの領域に分けることができ、各センサー値、グラフ、閾値設定を表示することができる。また、各センサーの閾値を超えるとブレッドボード上の LED ランプが点灯する設定にした。

各センサーの挙動として、例えば温度センサーの場合は、センサーを手で覆う等を行うことで、リアルタイムに温度や湿度の上昇を確認することができる。また、距離センサーを使用したシステムでは、センサーの受光部と障害物との距離を前後させることで、距離の測定を体験することができる。



図2 温度センサー画面

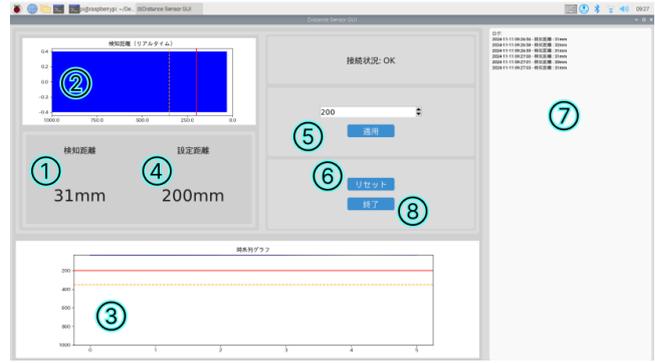


図3 距離センサー画面

図上の番号については以下のとおりである。

- |                   |                 |
|-------------------|-----------------|
| ① センサー値の表示        | ⑤ 閾値設定用のスピンボックス |
| ② センサー値のリアルタイムグラフ | ⑥ 時系列グラフのリセット   |
| ③ センサー値の時系列グラフ    | ⑦ センサー値取得等のログ表示 |
| ④ LED点灯用の閾値表示     | ⑧ アプリケーション終了ボタン |

## 2.3 実装

作成したプログラムは、データをリアルタイムで取得し、可視化するための GUI ツールである。Python の customtkinter ライブラリを用いた GUI 設計と、matplotlib によるデータのグラフ表示機能を搭載している。また、センサーの閾値を設定し、検知できるように設計されている。

### 2.3.1 使用ライブラリ

- ・ G U I : customtkinter, tkinter
- ・ データ処理 : numpy
- ・ グラフ描画 : matplotlib, japanize\_matplotlib
- ・ 時間管理 : datetime, time

### 2.3.2 システム構成

本アプリケーションは以下のコンポーネント（図4）で構成されている。

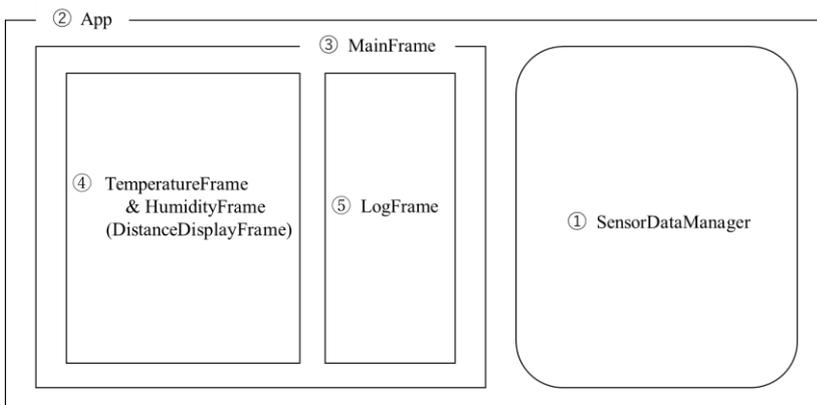


図4 システムの構成

### ① SensorDataManager

- ・ センサーのデータを管理するクラス
- ・ `update_data ()` メソッドでセンサーデータを取得
- ・ 更新間隔は 1 秒 (`update-interval=1000ms`) に設定

### ② App

- ・ `customtkinter.CTk` を継承したメインアプリケーション
- ・ `SensorDataManager` を利用し、定期的にセンサーデータを更新
- ・ `schedule_update ()` により 1 秒ごとにデータを取得
- ・ `MainFrame` に UI を組み込む

### ③ MainFrame

- ・ このフレーム以下を GUI として表示
- ・ `SensorFrame` と `LogFrame` をレイアウト
- ・ 温湿度センサーの場合には、`SensorFrame` に `TemperatureFrame` と `HumidityFrame` を配置
- ・ 距離センサーの場合には、`SensorFrame` に `DistanceFrame` を配置
- ・ `LogFrame` でセンサーデータの履歴を表示

### ④ TemperatureFrame & HumidityFrame (DistanceDisplayFrame)

- ・ 温湿度センサーの場合には、温度と湿度、距離センサーの場合には距離の表示を管理
- ・ それぞれ表示用の `DisplayFrame` を内包
- ・ 各データをリアルタイムで更新し、取得した値を可視化した棒グラフと、時系列グラフを表示
- ・ 各データについて閾値を変更する機能を実装
- ・ 閾値の変更に伴ってグラフの閾値表示を更新

### ⑤ LogFrame

- ・ センサーの取得データをログとして記録

## 2.3.3 実装上の工夫

- ・ `matplotlib` を用いたリアルタイムデータの可視化
- ・ 閾値を設定し、グラフ上に警告ライン (`axhline`) を表示
- ・ 例外処理 (`try-except`) を導入し、アプリケーションの安定動作を確保
- ・ `customtkinter` による、柔軟な UI デザインの作成

## 2.3.4 開発した教材の改善点

今回作成したシステムでは、一時的にログを表示することはできるが、保存機能は実装しておらず、システム再起動後にリセットされる。今回は教材開発という観点から、ログを保存する機能は省略して実装したが、温度や湿度などの環境を測定するモニタリングシステム等を実際に運用する場合は、ログのデータベース保存機能を追加することが望ましい。

また、今回のシステムでは閾値を超えた際に LED の点灯を行ったが、通知システム等を組み込むことにより、より利便性を高めることができる。

軽量化のために、GUI には Python フレームワークを使用して実装を行ったが、web アプリケーション等によって実装することにより、さらに柔軟なシステム構築を行うことが可能である。

### 3. 実際の対応

本講座前に、システムの作成、講義資料の準備を行った。

作成したシステムは、中学生全員に一人ずつ体験してもらうため、作成したシステムを SD カードごとコピーし、中学生全員分と予備用の合計 40 台作成した。Raspberry Pi やセンサー等も同様に、40 台分セットアップを行った（図 5）。

講座の対象が中学生であったため、プログラミング等の難しい部分は省き、簡単な Linux コマンドによる Python スクリプトの起動、システム、センサー等の操作を行ってもらい、実際目の前でセンサーにより測定された値が画面に出力される過程を体験してもらった。

中学生が操作するための説明用資料に関しては、PowerPoint でスライドを作成し、各種操作内容や手順について説明を行い、手順に沿って体験してもらった。また中学生が作業している中での質問への回答や逐次不具合等の対応も行った（図 6）。



図 5 セットアップ



図 6 講座の様子

### 4. まとめ

中学生たちは、実際にセンサーを操作することで、IoT 技術に対する理解を深めることができたと思われる。また、リアルタイムでデータが変化する様子を体験することで、IoT 技術への興味関心を高めることができたと考える。

一方で、教材開発のための大規模な環境構築やトラブル対応には多くの時間と労力を要し、今後の効率化が課題となる。

本講座の取り組みを通じ、IoT 教育の可能性を広げる貴重な経験となった。今後も、より多くの学生が技術に触れられる機会を提供し、教育の充実を図っていくことが重要であると考えます。