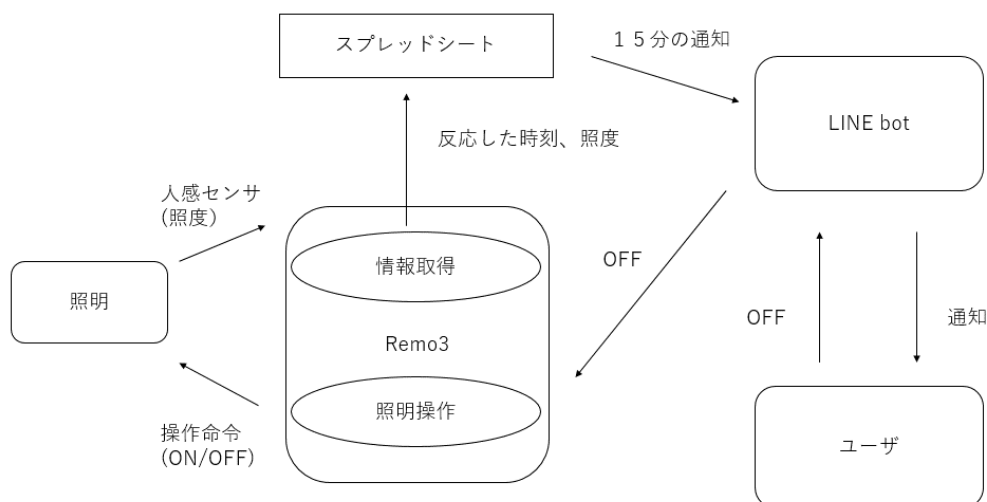


設計書

設計内容の概要

- ・ Remo3 を用いて 1 分ごとに人感センサーの情報を取得し、取得した時刻をスプレッドシートに記録しておく。
- ・ 人感センサーが反応した場合以下のように Remo3、照明、line を操作する。line での通知の文はあらかじめ決められた文章を送る。
- ・ 照明がついておらず、人感センサーが反応した場合 Remo 3 を通じて照明をつける。
- ・ Remo3 の人感センサーが反応し、照明がついた際、line 上で照明をつけたという通知を送る。
- ・ 照明をつけたと同時にスプレッドシート上でタイマーを作動させる。15 分経過したときに line 上で照明を消すかどうかを聞き、はいと返信がくると照明を消し、いいえと返信がくるとそのままにしておく。また、なにも反応がない場合照明はつけたままにしておく。

データフロー図



必要なモジュール

大まかに分けると以下の 3 つである。

- スプレッドシート管理用プログラム
- line 用プログラム（照明の点灯状況、15 分後に照明を消すかの確認を行う）
- 照明操作用プログラム（ユーザー設定、稼働状況、人感センサーの反応状況応じて操作する）

以下、一つ一つのモジュールについて

1.remoo()

メイン関数で、Nature Remo デバイスからデータを取得し、それをスプレッドシートに書き込み、照明を制御。

2.getNatureRemoData()

Nature Remo デバイスからセンサーデータを取得する関数。

3.getLastData()

スプレッドシートのログシートから最終行のデータを取得する関数。

4.addToSpreadSheet(data, row)

取得したデータを指定した行にスプレッドシートに書き込む関数。

5.controlLightBasedOnSpreadsheet()

スプレッドシートのデータを基に照明を制御する関数。

6.controlLightOn()

照明をオンにする関数。

7.controlLightOff()

照明をオフにする関数。

8.doPost(e)

LINE の Webhook に対応するための関数。ユーザーからのメッセージに基づいて照明を制御。

9.sendMessage(message)

LINE にメッセージを送信する関数。

10.resetTimer()

15 分後に確認メッセージを送信するためのタイマーをリセットする関数。

11.clearExistingTriggers()

既存のトリガーをクリアする関数。

12.sendConfirmationMessage()

15 分後に実行される関数で、ユーザーに照明を消灯するかどうかを確認するメッセージを

送信。