

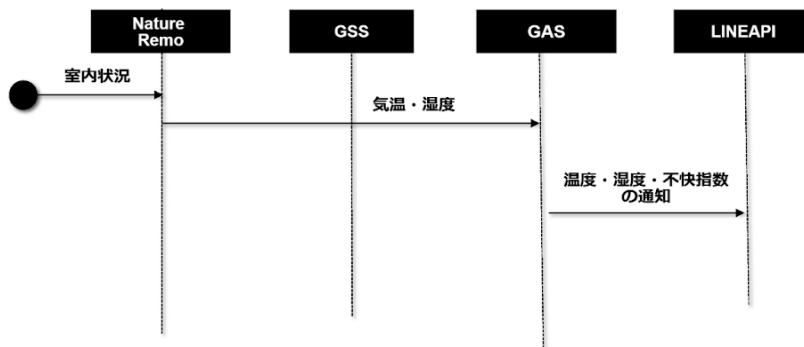
設計書

1.実装する機能

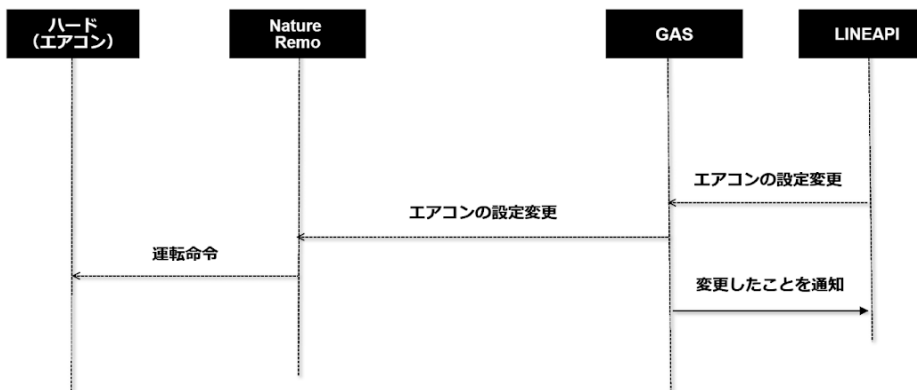
- 1.1時間毎に室内の不快指数を計測し、不快指数80以上なら不快指数をLINEに通知
- 2.リッチメニューからエアコンのON・OFF、温度、運転モードの変更を操作
- 3.リッチメニューから温度、湿度、不快指数を確認
- 4.リッチメニューからエアコンの設定を確認
- 5.リッチメニューからエアコンのON・OFFのタイマーを設定
- 6.リッチメニューから不快指数によって自動でエアコンを付けるモードに変更

2.機能のシーケンス図

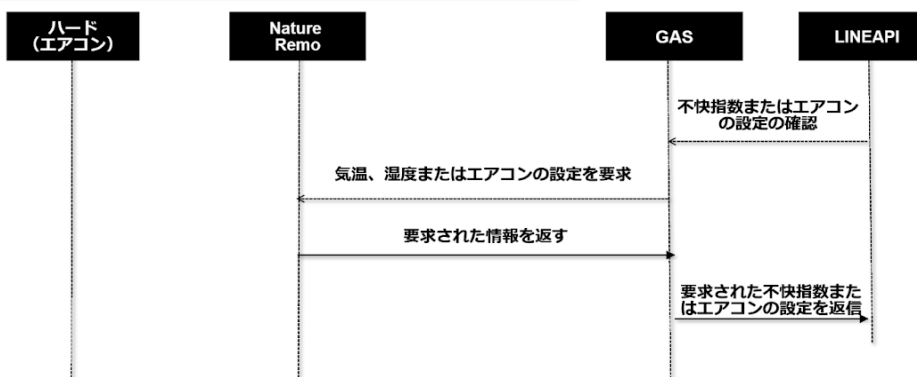
シーケンス図(不快指数の通知)



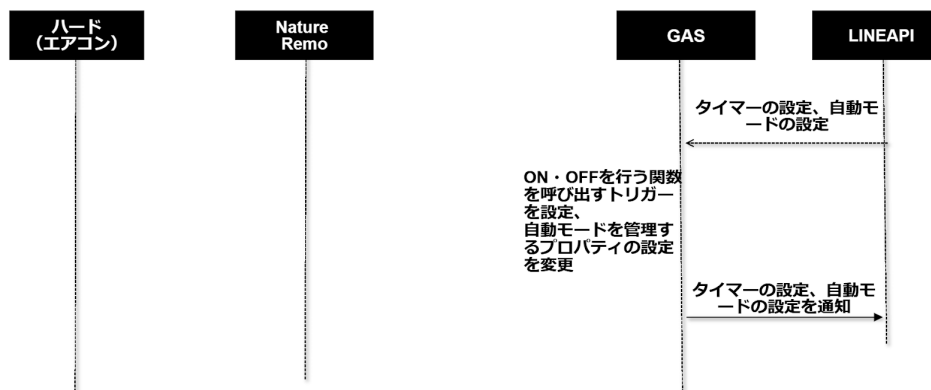
シーケンス図(エアコンのON・OFF、温度、運転モードの変更)



シーケンス図(不快指数、エアコンの設定の確認)

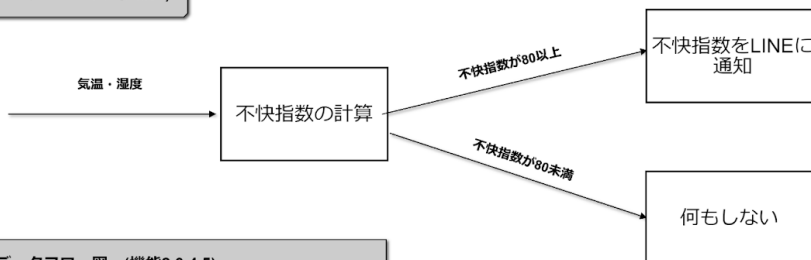


シーケンス図(不快指数、エアコンの設定の確認)

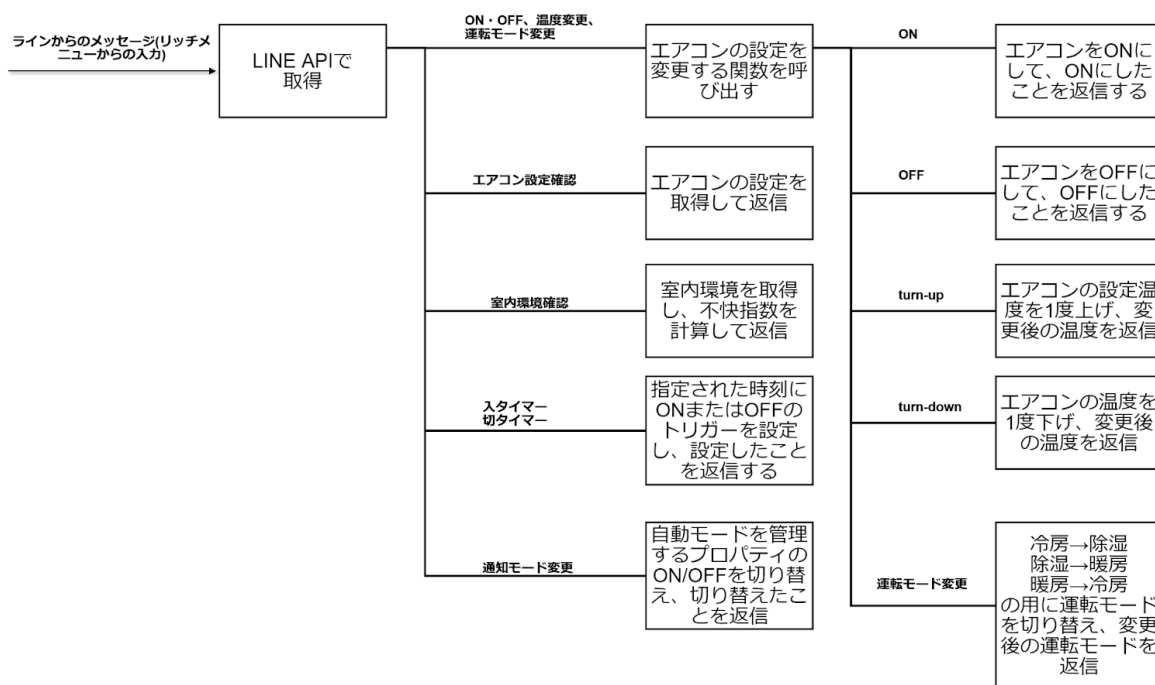


3.各機能のデータフロー図

データフロー図 (機能1)



データフロー図 (機能2,3,4,5)



4.エラー処理

remo.gs内の関数→リクエスト制限が起きたら、**"Too Many Requests"**という例外を発生させる

messageController() → 例外が "Too Many Requests" なら "少し時間を置いてから操作してね！"、それ以外なら "エラーが起こってるよ！" をメッセージにして送信

recordSensorData() → 例外が "Too Many Requests" なら一分後に再度実行する

5. 作成する関数

必要なモジュールとその内容

remo.gs	getNatureRemoData(endpoint)	endpoint(devicesかappliances)を引数に取り、remoやエアコンから情報をJSON形式で返す関数
	postNatureRemoData(endpoint, payload)	endpointと操作内容を引数に取り、エアコン等を操作する関数
aircon.gs	getApplianceID()	設定されたアクセストークンで操作できるエアコンの情報の一覧を返す関数
	controlAircon(control)	引数に応じてON・OFF、運転モード変更、設定温度変更の操作を行い操作に応じた返信用のメッセージを返す関数
	setTrigger(datetime, func)	日時と動作(ONまたはOFF)を受け取り、日時に動作を行うトリガーを設定し、返信用のメッセージを返す関数
	getSettings()	現在のエアコンの設定を返す関数
	checkSettings()	現在のエアコンの設定を返信用メッセージにして返す関数
	on(reset)	引数resetが"reset"なら27℃それ以外なら、直前の設定でエアコンをONにする関数
	off()	エアコンをOFFにする関数
	turnUp()	エアコンの設定温度を1℃上げる関数 30℃以上には設定できないようになっている
	turnDown()	エアコンの設定温度を1℃下げる関数 16℃以下には設定できないようになっている
	changeMode()	エアコンの運転モードを変更する関数 (冷房→除湿→暖房→冷房の順で変更する)
sensordata.gs	recordSensorData()	室内環境を記録し不快指数が80を超えたら、気温、湿度、不快指数を引数に指定し、gas2lineを呼び出す関数
	setSensorData(data, row, di)	remoで取得した室内環境、書き込む行、不快指数を受け取り、スプレッドシートに書き込む関数
	checkSensorData()	室内環境を取得し、気温、湿度、不快指数を知らせる返信用メッセージを返す関数
sheet.gs	getSheet(name)	スプレッドシートの名前を引数に取り、プロパティで正しいIDと引数に指定された名前のシートを開く関数
	getLastData(name)	引数になっている名前を持つスプレッドシートの最終行を取得する関数
discomfort.gs	discomfort(te, hu)	温度と湿度を引数に取り、不快指数を返す関数
line2gas.gs	doPost(e)	LINEからメッセージが送信された際に呼び出され、リッチメニューからのメッセージをmessageController()に送る関数
	messageController(event, replyToken)	doPost()で受信したポストバックイベントをメッセージの内容に応じて適切な関数を呼び出し、返信を行う関数
	replyLine(LineMessageObject, replyToken)	messageController()内で呼び出され、指定されたメッセージオブジェクトをLINEに返信する関数
gas2line.gs	gas2line(te, hu, di)	温度、湿度、不快指数を引数に取り、LINEに通知する関数
richmenueSettings.gs	richmenuSetUp()	newRichmenu1()、newRichmenu2()を呼び出し、リッチメニューを登録する関数
	getRichmenus()	現在作成されているリッチメニューの一覧を取得する関数
	getAliases()	現在作成されているエイリアスの一覧を取得する関数
	deleteRichmenu(id)	リッチメニューをIDで指定して削除する関数
	deleteOldRichmenu(name)	リッチメニューの名前を指定し、名前が一致したリッチメニューを一括削除する関数
richmenueLeft.gs	newRichmenu1()	左側のリッチメニューを作成し、画像の紐付け、エイリアスの設定し、デフォルトのリッチメニューに設定する関数
	makeRichmenu1()	左側のリッチメニューを作成し、リッチメニューIDを返す関数
	makeAlias1()	左側のリッチメニューIDを受け取り、リッチメニューエイリアスを新規作成する関数
	setAlias1()	左側のリッチメニューIDを受け取り、リッチメニューエイリアスを更新する関数
	setImage1()	左側のリッチメニューIDを受け取り、左側のリッチメニューに画像を紐付けする関数
	setRichmenu()	リッチメニューIDを受け取り、受け取ったリッチメニューをリッチメニューをデフォルトに設定する関数
richmenueRight.gs	newRichmenu2()	右側のリッチメニューを作成し、画像の紐付け、エイリアスの設定する関数
	makeRichmenu2()	右側のリッチメニューを作成し、リッチメニューIDを返す関数
	makeAlias2()	右側のリッチメニューIDを受け取り、リッチメニューエイリアスを新規作成する関数
	setAlias2()	右側のリッチメニューIDを受け取り、リッチメニューエイリアスを更新する関数
	setImage2()	右側のリッチメニューIDを受け取り、右側のリッチメニューに画像を紐付けする関数
property.gs	setToken()	アクセストークンやスプレッドシートのID等をプロパティで設定する関数
	changeNoticeMode()	自動ON・OFFのプロパティ設定を変更し、変更したことを通知する返信用メッセージを返す関数