IoT 物聯網期末專題報告

R09942113 張凱傑

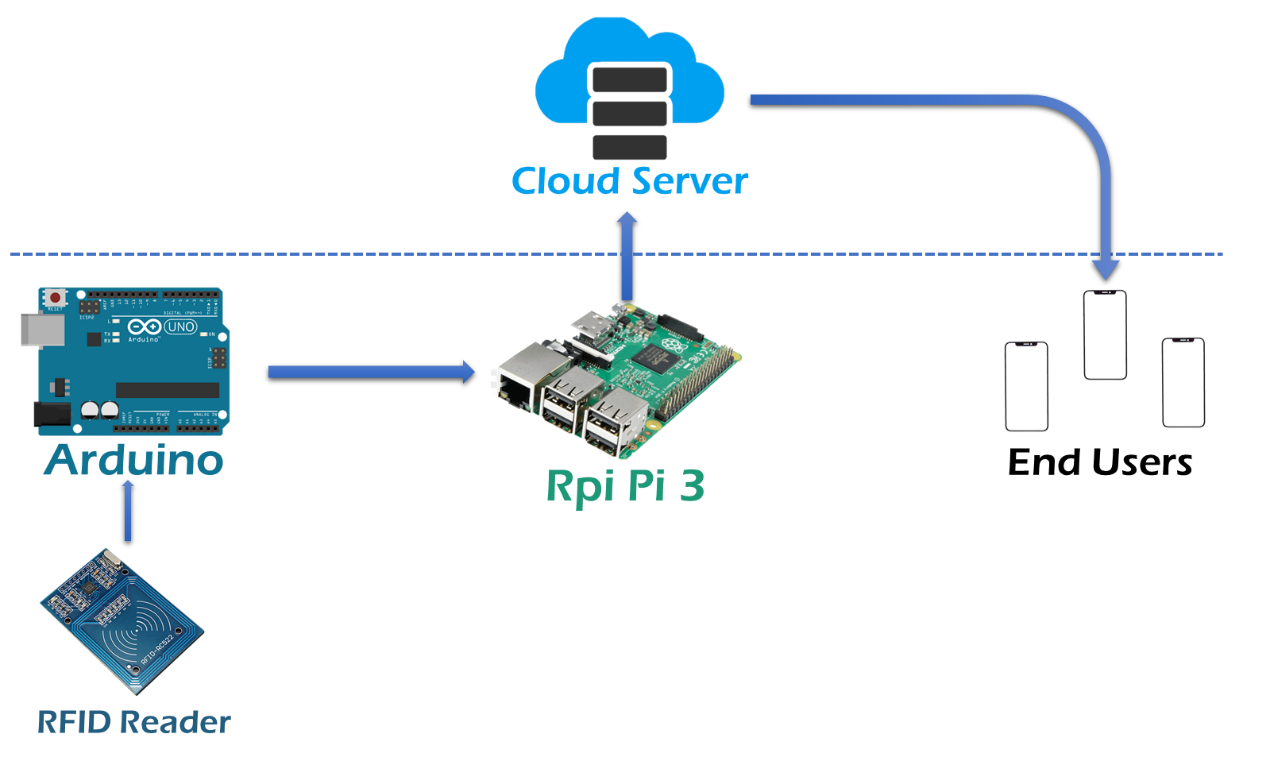
D09942016 梁梓峰

1. **動機與目的**

開發豬隻適用之RFID耳標，將RFID感應技術與國內畜牧設備結合，開發相關軟體將畜牧設備所蒐集之資訊彙整及應用。

本次專題與農科院合作，針對這次專題農科院所長表示，較多的重心放在RFID的建置上，與RFID廠商進行討論後，原擬專題內容為豬隻RFID耳標的規格化，主要在RF reader上對天線與耳標的偵測需求進行建議。後來發現題目制定上與原課程核心目標有些出入，因而在討論過程中，找到一相關議題，豬場對豬隻的管理應用很多，檢測豬隻在農場的分布及其習性、家族基因等等議題，檢測目的即為減少豬隻損失，提早發現異狀進而降低成本，因此我們利用RFID耳標與reader偵測豬隻是否長時間未飲食為主題，透過雲端伺服器在檢測飲食異常時，使用Line的推撥訊息到養豬戶隨身攜帶的手機。

1. **材料與系統架構**



**圖一、系統架構圖**

1. **演算法原理**

紀錄RFID更新時間，若超過警告設定值，則推撥至Line用戶。

1. **核心函式(code)**

Rpi with rf-reader

from re import match

from datetime import date, datetime, timedelta

import time, psycopg2, os

from portdetect import \*

def update\_table\_row(DATABASE\_URL, uid, nowTime):

    conn = psycopg2.connect(DATABASE\_URL, sslmode='require')

    cursor = conn.cursor() # 初始化執行指令的工具

    # nowTime = datetime.datetime.now()

    cursor.execute("UPDATE pig\_care SET LastEatTime = %s WHERE rfid = %s;", (str(nowTime), uid))

    print("Updated 1 row of data")

    conn.commit()

    cursor.close()

    conn.close()

if \_\_name\_\_ == '\_\_main\_\_':

    DATABASE\_URL = os.popen('heroku config:get DATABASE\_URL -a kj-iot').read()[:-1]

    port\_opened = serial\_ports()

    s = serial.Serial(port\_opened, 9600, timeout = 0.1)

    try:

        while True:

            msg = s.readline().decode('utf-8')

            tmp = re.match(uid\_pattern, msg)

            if tmp:

                uid = "".join(str(tmp.group(1)).split(" "))

                uid = str(uid.split("\r")[0])

                nowTime = datetime.now()

                print('uid =',uid, 'nowTime',nowTime)

                update\_table\_row(DATABASE\_URL, uid, nowTime)

    except KeyboardInterrupt:

        s.close()

        print("Program Closed")

Server - Heroku

from flask import Flask, request, abort

from linebot import (

    LineBotApi, WebhookHandler

)

from linebot.exceptions import (

    InvalidSignatureError

)

from linebot.models import \*

#======這裡是呼叫的檔案內容=====

from message import \*

from new import \*

from Function import \*

#======這裡是呼叫的檔案內容=====

#======python的函數庫==========

import tempfile, os, re

import datetime

import time

import psycopg2

from urllib.parse import urlparse

#======python的函數庫==========

app = Flask(\_\_name\_\_)

static\_tmp\_path = os.path.join(os.path.dirname(\_\_file\_\_), 'static', 'tmp')

# Channel Access Token

line\_bot\_api = LineBotApi('A/lZF7guXu0wk/Vryza9qVhKO1CxPSQyLdsLlrEtmf/vnMYuX5bABO4JWWcDcPQ17vOPTZC/jgOV788AG71YP1IvlWiI++11UN5VljRLMgJBOQGU8CrmCjciD67WlPQ+g8GTY4Vc6j4tX/Y5sIzNegdB04t89/1O/w1cDnyilFU=')

# Channel Secret

handler = WebhookHandler('e755ec351e78012112aa037adbeaa097')

# DATABASE URL

DATABASE\_URL = os.popen('heroku config:get DATABASE\_URL -a kj-iot2').read()[:-1]

# User ID

# 主動推撥 (每月500則)

line\_bot\_api.push\_message('U70d4875c22240836332561cc9485826c', TextSendMessage(text='App start!!'))

@app.route("/callback", methods=['POST'])

def callback():

    signature = request.headers['X-Line-Signature']

    # get request body as text

    body = request.get\_data(as\_text=True)

    app.logger.info("Request body: " + body)

    try:

    # handler.handle(body, signature)

        while 1:

            url = urlparse.urlparse(os.environ.get('DATABASE\_URL'))

            db = "dbname=%s user=%s password=%s host=%s " % (url.path[1:], url.username, url.password, url.hostname)

            conn = psycopg2.connect(db)

            # conn = psycopg2.connect(DATABASE\_URL, sslmode='require')

            cursor = conn.cursor() # 初始化執行指令的工具

            cursor.execute("SELECT \* FROM pig\_care;")

            rows = cursor.fetchall()

            conn.commit()

            cursor.close()

            conn.close()

            nowTime = datetime.datetime.now()

            for row in rows:

                rfid = row[1]

                past\_time = datetime.datetime.strptime(row[2], "%Y-%m-%d %H:%M:%S.%f")

                duration = (nowTime - past\_time).seconds

                text\_info = 'ID：「'+str(rfid)+'」| Abnormal diet about '+ str(duration) + 'seconds!'

                if duration > 5:

                    line\_bot\_api.push\_message('U70d4875c22240836332561cc9485826c', TextSendMessage(text=text\_info))

            time.sleep(30)

    except InvalidSignatureError:

        abort(400)

    return 'OK'

# 主程式

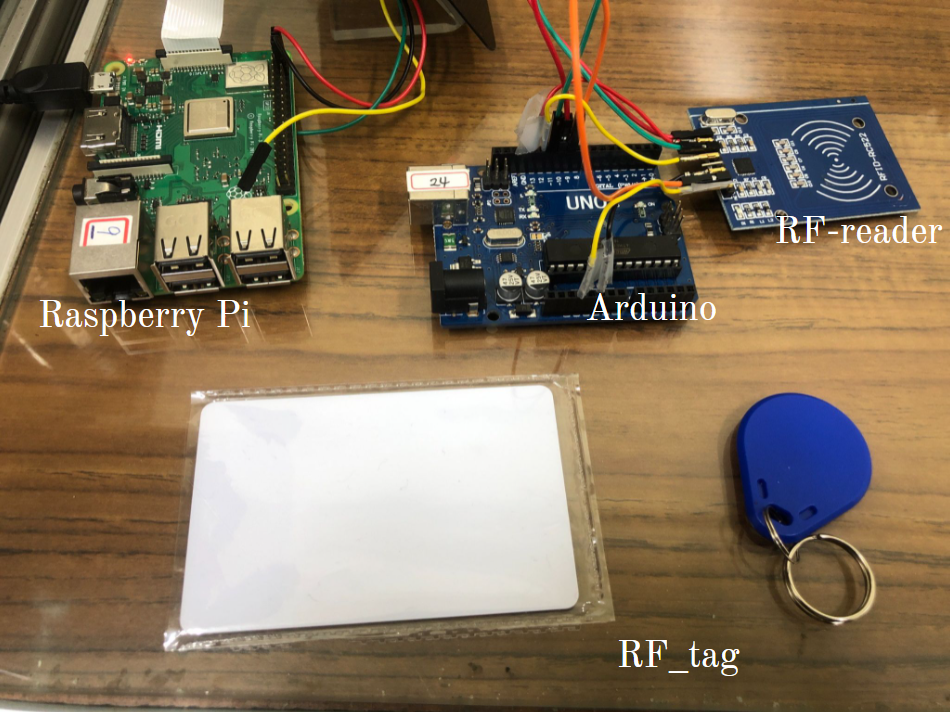
import os

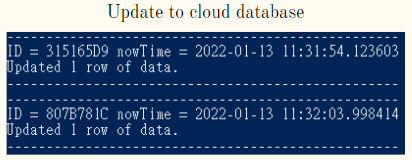
if \_\_name\_\_ == "\_\_main\_\_":

    port = int(os.environ.get('PORT', 5000))

    app.run(host='0.0.0.0', port=port)

1. **結果 –** demo







1. **結論與未來展望**

此次專題，透過實地參訪了解環境對產品的影響，及多方討論來解決問題。豬隻上的RFID主要會因同伴互咬脫落或走路晃動造成Reader偵測失敗，為提升豬隻活動偵測及分析，此次專題方案提供給農科院，對豬隻健康管理有較適用的解決方案，未來可擴充多樣功能至雲端伺服器，隨時地將資訊傳給養豬戶，獲得更即時且有效率的資訊，也可透過分析資料庫數據，系統式統計提供養豬場改善養殖方式，對豬場豬隻管理成本及品質維護提升。