**IoT Gateway:**

https://thenewstack.io/tutorial-prototyping-a-sensor-node-and-iot-gateway-with-arduino-and-raspberry-pi-part-1/

https://thenewstack.io/tutorial-configuring-a-sensor-node-and-iot-gateway-to-collect-and-visualize-data-part-2/

**Xbee:**

**Xbee與Zigbee基礎知識:**

https://swf.com.tw/?s=Xbee

https://yuanchieh.wordpress.com/category/%e7%8e%a9%e7%8e%a9mcu/xbee%e7%b3%bb%e5%88%97/

**Arduino對 PC 1 對1通訊**

https://qiita.com/s\_fujii/items/9804eaf3599139164aa3

**官方Guide:**

http://ftp1.digi.com/support/documentation/XBee%20ZigBee%20Migration%20Guide.pdf

**Xbee api mode相關測試(使用PIC):**

http://zattouka.net/GarageHouse/micon/XBee/XBee5.htm

**AT command一覽:**

http://zattouka.net/GarageHouse/micon/XBee/ATcommands.pdf

**從network構成到api mode，sleep mode都有詳細解說的網站(含arduino程式碼):**

https://www.aihara.co.jp/~junt/electronic-circuit/XBee.html

**相關書籍:**

Robert Faludi - Building Wireless Sensor Networks with ZigBee, XBee, Arduino, and Processing – 2010

**Arduino:**

**說明:**源始碼位於ArduinoEndDevice資料夾中

**arduino Xbee library:**

https://github.com/andrewrapp/xbee-arduino

**MH-Z19B接線與程式碼:**

https://sites.google.com/site/wenyumaker/03-mh-z19er-yang-hua-tan/01-mh-z19-arduino-oled

**SHT-31接線與程式碼:**

http://arduinolearning.com/code/arduino-sht31-module.php

**Android app:**

**說明:**源始碼位於MoniterApp資料夾中，為android studio專案格式，啟動後會到webserver載入end devices列表並繪製畫面layout，再以一定的時間間隔回webserver拿回更新的資料，另外透過firebase service在接到訊息時會發出通知到手機畫面上。

**MPAndroidChart官方文件:**

https://weeklycoding.com/mpandroidchart-documentation/

**MPAndroidChart相關實例:**

https://blog.csdn.net/jinmie0193/article/details/90239935#%E4%B8%80%E3%80%81MpAndroidChart%E7%9A%84%E5%9F%BA%E6%9C%AC%E4%BD%BF%E7%94%A8

https://blog.csdn.net/HAndroidevelopcker/article/details/77967471

**Fragment(用來製作分頁):**

https://qiita.com/Reyurnible/items/dffd70144da213e1208b

**Viewpager(Fragment的容器，可以切換當前顯示的fragment):**

https://developer.android.com/training/animation/screen-slide?hl=ja

**在app中實裝firebase功能:**

https://firebase.google.com/docs/android/setup?authuser=0

https://litotom.com/2016/06/26/firebase-android-send-2/

**透過firebase送訊息:**

https://firebase.google.com/docs/cloud-messaging/send-message?authuser=0

**從app傳送通知給手機:**

https://developer.android.com/training/notify-user/build-notification

**Transmitter:**

**Receive port:** **/dev/ttyUSB0 fixed in code**

**To: entry.showdata.nctu.me fixed in code**

**說明:** 在raspberry pi上設定了當連上網路才完成開機，並在開機完成後自動執行程序(用rc.local達成)從/dev/ttyUSB0上之xbee接收訊息。源始碼位於Transmitter資料夾中，其中tran\_v2.py是主程式，aes.py提供加密的功能。

**設定raspberry pi**

http://blog.itist.tw/2017/03/headless-wifi-setup-without-monitor-keyboard-or-cables-for-raspberry-pi-zero-w-and-3-model-b.html

**用usb連接xbee:**

https://www.digi.com/support/forum/68264/help-required-xbee-connected-raspi3-using-explorer-ch340g

**python多線程:**

http://yoyzhou.github.io/blog/2013/02/28/python-threads-synchronization-locks/

**對稱加密:**

https://github.com/ijl20/python\_java\_crypto/

**Entry:**

**Receive port:6000 fixed in code**

**To: showdata.nctu.me fixed in code**

源始碼位於AcceptEntry資料夾中且為一netbeans ide project格式(dist資料夾會包含編譯後的程式)，在使用前請先確定目前資料夾有包含正確的

sslclienttrust, sslentrykeys檔案，使用方式有一種



1. 使用命令列

java -jar AcceptEntry.jar

另請注意本程式使用了ssl機制中的憑證是有時效的，如發現過期請新簽發更換

若從監測端收到了一對稱解密出現異常的訊息，則直接捨棄之。

**java自訂信任機制(包含keystore, truuussstore說明):**

https://www.ibm.com/developerworks/cn/java/j-lo-ssltls/index.html

https://blog.csdn.net/pzysoft/article/details/62888780

**linux系統安裝jdk8:**

https://blog.csdn.net/zbj18314469395/article/details/86064849

**對稱加密:**

https://github.com/ijl20/python\_java\_crypto/

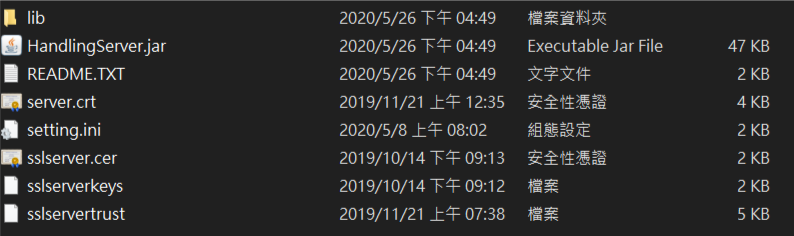
**HandingServer:**

**Receive port:5000 fixed in code**

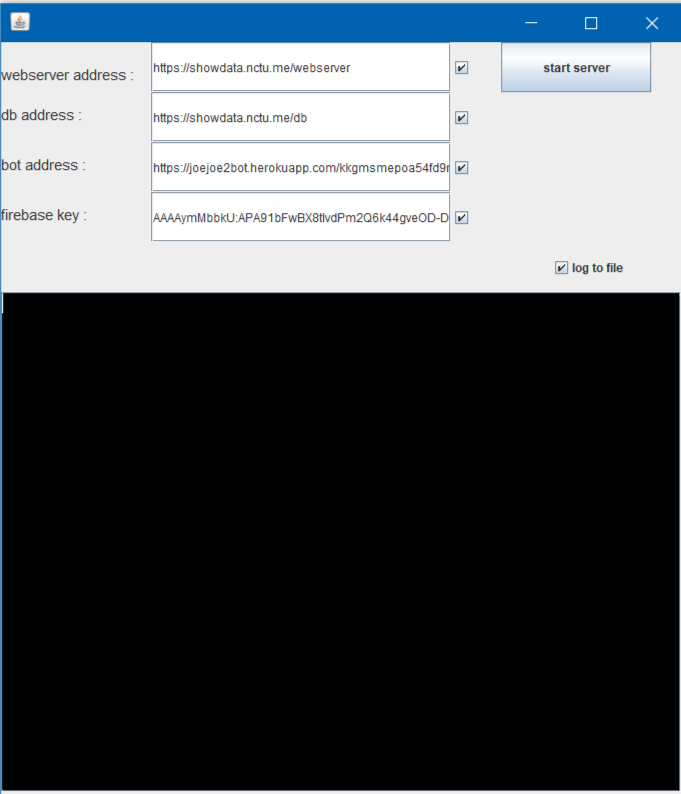
**To:** **webserver、db、bot、firebase can be changed**

**說明:**源始碼位於HandingServer資料夾中且為一netbeans ide project格式(dist資料夾會包含編譯後的程式)，在使用前請先確定目前資料夾有包含正確的

setting.ini(格式見本資料夾的範例), sslservertrust, sslserverkeys檔案，使用方式有兩種



1. 雙擊HandingServer.jar啟動圖形介面，可以設定webserver、db、bot address、firebase key的位置和啟用與否或是是否要生成log檔，按下start/stop server即可開啟或停止



1. 使用命令列

java -jar HandingServer.jar -c -w webserver\_addr -d db\_addr -b bot\_addr -f firebase\_key 不啟動圖形介面，由後方參數設定各項位置，不設定者則視為不啟用該功能，並會生成log檔，或是由java -jar HandingServer.jar -g 啟動圖形介面，則同1.

另請注意本程式使用了ssl機制中的憑證是有時效的，如發現過期請新簽發更換

此外setting.ini紀錄了所有end devices，任何修改將在下次start server時生效，若從監測端收到了一未經登記之end device的訊息(或不合法的訊息格式、非信任憑證)，則不會對其進行處理並會log 警語。

**java自訂信任機制(包含keystore, truuussstore說明):**

https://www.ibm.com/developerworks/cn/java/j-lo-ssltls/index.html

https://blog.csdn.net/pzysoft/article/details/62888780

**linux系統安裝jdk8:**

https://blog.csdn.net/zbj18314469395/article/details/86064849

**域名與架站:**

若要自行架站需要實體ip和網域名，若主機使用無線或分享網路還要考慮通訊埠轉發，可以參考以下文章進行操作

**固定IP:**

https://wuangus.cc/domain/

**免費網域:**

https://nctu.me/

**Websever & db:**

**Receive port:80/443 fixed**

**說明:**源始碼位於webserver和db資料夾中，為php格式配合mysqli，本方案可由Appserv軟體包一次安裝完成，之後在於安裝路徑之AppServ/www/下配置檔案即可映射到網站上(例如: 本機名myhost.ncku，AppServ/www/index.html可由myhost.ncku/index.html瀏覽)，要啟動server的話請依序執行Apache Start、MySQL Start(以win10為例)，database管理可透過http://localhost/phpmyadmin/index.php進行。

其中組織結構大略為

webserver/

clear\_devices\_cfg.php 負責清除end diveces列表

testjson.php負責讀取最新資料

get\_devices\_cfg.php負責傳回end diveces列表

commit.php 負責儲存資料

set\_devices\_cfg.php負責設定end diveces列表

monitor.html 監控面板主頁

db/

clear\_devices\_cfg.php 負責清除end diveces列表

get\_data.php負責讀取資料

get\_devices\_cfg.php負責傳回end diveces列表

save\_to\_db.php 負責儲存資料

set\_devices\_cfg.php負責設定end diveces列表

**openssl、配置憑證和https:**

https://joe01032002.pixnet.net/blog/post/92665237

**網頁與vue.js的使用:**

https://cn.vuejs.org/v2/guide/transitions.html#%E5%88%97%E8%A1%A8%E7%9A%84%E8%BF%9B%E5%85%A5-%E7%A6%BB%E5%BC%80%E8%BF%87%E6%B8%A1

**plotly.js:**

https://plotly.com/javascript/

**line bot:**

**說明:**本服務建立在我以前做過的專案的擴充，請參考https://github.com/joejoe2/LineBot，基本上由flask建立路徑映射配合gunicorn執行微服務，整個過程都是在docker container中進行，目前託管於heroku雲端上的免費方案。

**基本介紹:**

https://github.com/line/line-bot-sdk-python

https://xiaosean.github.io/chatbot/2018-04-19-LineChatbot\_usage/

**利用gunicorn:**

https://juejin.im/post/5ce8cab8e51d4577523f22f8

**簡單運作範例:**

arduno + sensor + xbee 🡪 xbee + raspberry pi 🡪 entry (entry.showdata.nctu.me:6000) 🡪

handling server (showdata.nctu.me:5000)

🡪 webserver(showdata.nctu.me/webserver/xxx), db (showdata.nctu.me/db/xxx), bot, firebase

最後才到使用者端