

## 目錄

- (1) 個人簡歷
- (2) 專題報告
- (3) 氣機實習專案

## 個人簡歷

姓名	林建誠	
畢業學校	國立臺北科技大學(109 - 112)	
畢業科系	電機工程系	
學業成績	總平均分數：93.21 (GPA：4 / 4)	
成績排名	班排：1 / 48 (2.1%)	系排：1 / 143 (0.7%)
技能	<ul style="list-style-type: none"><li>● 軟體：C / C++ / C#(WinForm) 、Python 、 kotlin(Android App) 、Dart</li><li>● 韌體：ESP32</li><li>● 硬體：表面貼焊(SMT)</li></ul>	
企業實習	氣機科技 <b>軟韌工程師實習</b> (111.1 ~ 112.9)	
實務專題	App 智慧鏡頭偵測黃疸貧血狀況 <b>通過北醫人體研究處(IRB)計畫申請</b> <b>入圍創新創業激勵計畫(FITI)初選</b>	
實習專案	<ul style="list-style-type: none"><li>● C# WinForm 檢貨系統(具前後端及資料庫系統)</li><li>● BLE 應用 Android App (藍芽通訊與韌體結合)</li><li>● 韌體(ESP32)開發</li><li>● 敏捷式專案管理(Agile Project Management)</li></ul>	

國立臺北科技大學  
電機工程系  
一一二年度專題實務

App 智慧鏡頭偵測黃疸貧血狀況

組員

電機四丙 林建誠 109830051

電機四甲 曾柏揚 109301143

指導教授：張正春副教授

中華民國 112 年 9 月 28 號

# 目錄

## 第一章、 前言

## 第二章、 研究問題與目的

- (1) 影像裁切與色溫問題
- (2) 預測黃疸貧血
- (3) 在手機 App 運行 AI 模型

## 第三章、 研究流程與方法

- (1) 解決影像裁切與色溫的問題
- (2) 預測黃疸貧血演算法
- (3) 解決在手機 App 運行 AI 模型

## 第四章、 成果展現

## 第五章、 參考文獻

## 壹、前言

貧血全球普遍影響 20 億人，通常以血紅素(Hb)低於平均水平定義，導致氧氣運輸受損。早期貧血常無症狀，需侵入性測試，多年來，醫師經常通過觀察病人的眼睛結膜來粗略評估貧血的情況。本研究試圖以非侵入 AI 影像演算法檢測貧血，提供早期幫助。此外，黃疸常伴隨肝疾，以眼鞏膜變黃表現。本研究也試圖以非侵入 AI 演算法檢測黃疸，預測黃疸程度，為醫護提供參考數據，早期發現處理黃疸貧血，改善病人生活。

## 貳、研究問題與目的

本專題為透過手機鏡頭偵測黃疸貧血，參與 IRB 計畫(人體試驗委員會)與雙和醫院的盧柏文醫師合作，收集雙和醫院與新國民醫院在臨床上疑似有貧血或者是黃疸之患者，經抽血驗證血色素與血容積和膽紅素的數值之後，與手機拍攝的眼部色澤運用 AI 做進一步的比對。以下為本研究需要處理的問題。

### (1) 影像裁切與色溫問題：

運用手機採集患者眼部影像，並透過 AI 模型辨識眼睛進行裁切，取得眼瞼與眼鞏膜(眼白)的影像訊息，以便進行預測數據的步驟。且拍攝者所處環境、或使用的手機廠牌不同，拍攝的影像色溫不同，會使往後的預測不準確，為解決此問題，本研究運用白平衡演算法，減少每組影像差異。

### (2) 預測黃疸貧血：

取得眼瞼和眼白的影像信息後，下一步為開發用於分辨是否存在黃疸和貧血的演算法或人工智慧模型。為此，本研究參與 IRB 計畫並收集約 300 名患者數據，包括患者的眼睛影像信息以及相應的侵入性檢驗結果，特別是黃疸和貧血數據。本研究計劃分析這些數據並進行線性回歸分析，以建立預測模型。這個模型將根據眼睛的影像特徵預測是否存在黃疸或貧血。

### (3) 在手機 App 運行 AI 模型：

為將此預測系統運行於手機上，本研究需開發 App，並具備以下功能：

#### a. 照片拍攝與裁切：

用戶可使用手機相機拍攝眼睛。這些影像將用於後續分析。

#### b. 預測功能：

App 將使用 AI 模型對眼睛影像進行分析，預測是否存在黃疸和貧血跡象。預測結果將在用戶的手機上顯示。

#### c. 數據上傳：

預測結果將被上傳到雲端伺服器，以便持續收集和分析數

據，用於改進預測模型的準確性。

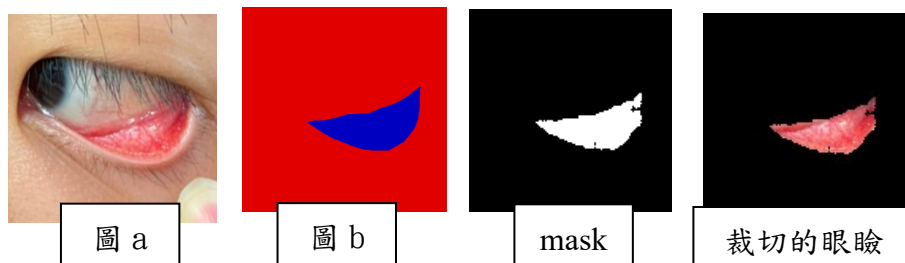
## 參、 研究流程與方法

透過 IRB 計畫(人體試驗委員會)與雙和醫院的盧柏文醫師合作，約取得 300 名的受試者資料，分別拍攝其左右眼的影像以及上半身影像(以確認環境光源)，經過嚴格的資料篩選(剔除沒拍攝到眼瞼或眼白的部分，因眼瞼為判斷貧血與否的依據，而眼白為判斷黃疸與否的依據)，本研究最終擁有約 1000 張適合的影像，並將資料分成 800 張的訓練集以及 200 的驗證集。以下是實驗步驟設計。

### (1) 解決影像裁切與色溫的問題：

#### a. AI 照片裁切(Unet)：

本研究使用 Unet 模型進行影像裁切，因 Unet 模型在少量數據下也能表現出色，這對於有限的資料數量非常適合。使用 800 張訓練集進行訓練(影像兩張一組，一張原圖(圖 a)，一張標記出想辨識的區域(圖 b)，影像大小為 256\*256，因為資料集不多，EPOCHS 選擇 20，VAL\_SUBSPLIT 選擇 5，以免 overfitting)，分別訓練標記眼瞼與眼白部分。本研究稱 Unet 模型輸出影像為 mask(只有 0 和 1 的二元影像)，mask 大小與原圖相同，與原圖做影像相乘後即可裁切出眼瞼和眼白。



#### b. 白平衡模型：

研讀各種文獻後，選擇「Deep White-Balance Editing」此篇論文的白平衡模型(以 PyTorch 架構)作為本研究白平衡的基準，除了原本論文提供的資料集(同影像兩張一組，一張經過白平衡，一張給予不同色溫)，本研究另外加入 800 張的眼睛訓練集影像，可更符合使用場景。

### (2) 預測黃疸貧血演算法：

#### a. 貧血預測演算法：

貧血判別是基於眼瞼顏色，偏紅則血色素濃度較高，貧血機率較低，而偏粉或白則為血色素濃度較低，貧血機率較高，所以本研究透過眼瞼顏色與血色素回歸預測，判別血色素濃度，具體方法如下：

讀取裁切後只有眼瞼的影像，並分成 R、G、B 三個通道，光學中當 R、G、B 三個色彩的數值相同時會呈現白光(但光強度偏低時影像可能呈現灰白色)，當 R 通道數值相對於 G、B 通道數值高時，則偏紅色，反之，當 R 通道數值與 G、B 通道數值差不多時，則偏灰或粉色。由此可知 R 通道數值與 G、B 通道數值差異會與血色素正相關。透過 Python，本研究計算出一張眼瞼影像中各個像素點的  $C_{RmGB}$

$$C_{RmGB} = C_R - (0.5 * C_G + 0.5 * C_B)$$

$C_R$ : R 通道的數值,  $C_G$ : G 通道的數值,  $C_B$ : B 通道的數值

去除影像反光(c 部分會說明如何去除)像素點後的  $C_{RmGB}$ ，取其中位數  $Md_{RmGB}$ 。將各組影像與其  $Md_{RmGB}$  做多項式回歸，幕次為三，此多項式做為「眼瞼影像與血紅素關係式」，並將驗證集套入關係式中，得到相對應的血紅素值，低於本研究設定的閾值 (12gm/dL) 則判斷為貧血，與透過侵入式檢驗判斷的貧血做比較，正確率為七成以上。

- b. 黃疸的判斷是基於眼白的顏色，本研究將裁切後影像進行色彩分析。因為專題時間關係，尚未尋找到準確率高於七成的分析方式。
- c. 影像可能存在反光問題影響「眼瞼影像與血紅素關係式」的準確性，本研究使用 HSV 色彩空間篩選飽和度 (S，當飽和度過低時，色彩偏灰) 和明度 (V，當明度過高時，色彩偏亮)，當飽和度過低且明度過高時，影像上表現為白色反光，並使準確性降低。所以將飽和度過低且明度偏高的像素點去除，不列入黃疸貧血判斷，以提高預測的準確性(約可以提高 5~10% 的準確率)。

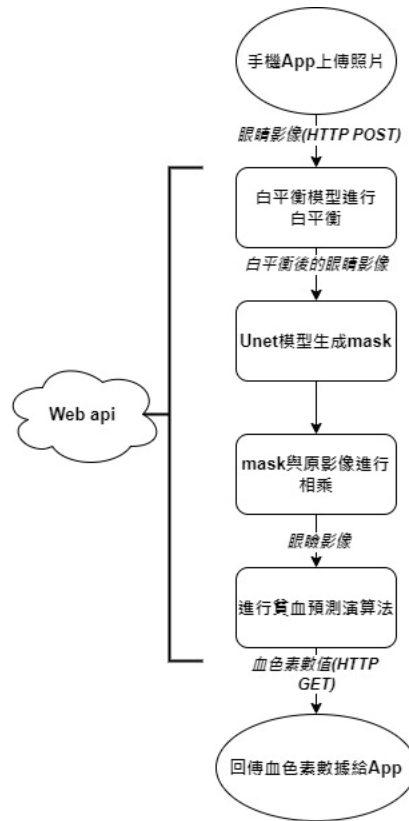
### (3) 解決在手機 App 運行 AI 模型：

使白平衡模型和 Unet 模型在手機 App 上運行，本研究透過 Python Flask 在 AWS(Amazon Web Services)上架設輕量級的後端 web API，再透過手機 App 串接 API，達成在手機運行 AI 模型的要求，具體方式如下：

#### a. API 建構：

**HTTP POST：**透過 HTTP POST 上傳手機 APP 端影像，經過白平衡、Unet 模型以及貧血預測演算法，將會得到相對應的血色素濃度。

**HTTP GET**：return 的數值即是相對應血紅素值，回傳數值到手機後，最後由 App 判斷是否低於閾值。以下是架構圖。



API 架構圖

b. API 上線：

使用 AWS(Amazon Web Services)中 EC2(雲端虛擬機)建立執行個體，且需選擇較大記憶體空間的類型(太小導致很多 Python 套件無法安裝而無法運行 AI 模型)，本研究使用 t3.large 類型。且建立時需設定安全群組，使手機 API 可以連線，否則手機 API 無法成功執行。最後開啟執行個體，上傳模型及執行前段 API 程式碼即完成 API 架構與上線。

透過 Web API 可將使用者的資料再訓練 Unet、白平衡模型以及修正預測黃疸貧血演算法，產生更準確的預測，也可及時提供最新版本的 AI 與演算法。

(4) 手機 APP 設計：

使用 Flutter 架構與 Dart 語言開發手機 App，主要功能有拍攝眼部照片、紀錄侵入式檢查結果、紀錄黃疸貧血預測值(計畫透過 SQL 將數值儲存)、上傳拍攝照片到 AWS(Amazon Web Services)的 Web API 並回傳黃疸貧血預測值。而連接 API 方式為 Flutter 函式庫中的 Dio，使



用 HTTP POST 上傳手機拍攝影像到 API 與 HTTP GET 回傳相對應血色素濃度，最後經由閾值(貧血為血色素低於 12gm/dL)判斷是否貧血與黃疸。

#### 肆、 成果展現

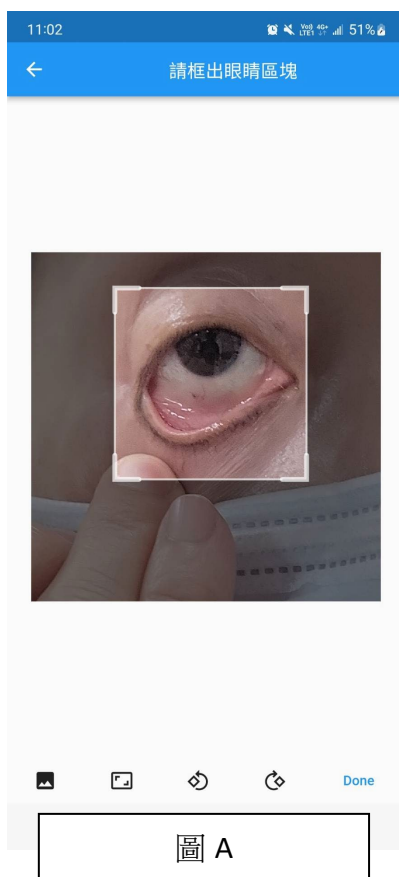
功能展示(手機 App 截圖)

##### 圖 A 功能介紹：

使用者透過相機拍攝影像，也可匯入原相簿影像，並使用可縮放的正方形格子裁切，取得眼睛部分，此影像大小可能不一，透過 Resize 將影像大小設定為 256\*256，HTTP POST 上 AWS 虛擬機並進行預測貧血演算法處理。

##### 圖 B 功能介紹：

HTTP GET 取得 AWS 虛擬機預測數值，並呈現於使用者，判斷是否數值異常。



#### 伍、 參考文獻

(1) 論文 Deep White-Balance Editing

[https://openaccess.thecvf.com/content\\_CVPR\\_2020/papers/Afifi\\_Deep\\_White-Balance\\_Editing\\_CVPR\\_2020\\_paper.pdf](https://openaccess.thecvf.com/content_CVPR_2020/papers/Afifi_Deep_White-Balance_Editing_CVPR_2020_paper.pdf)

- (2) Robert G. Mannino, Smartphone app for non-invasive detection of anemia using only patient-sourced photos, Journal of Nature Communication. 2018
- (3) Angelopoulou, E. Understanding the color of human skin. in Human vision and electronic imaging VI. 2001. International Society for Optics and Photonics
- (4) Spinelli, M.G.N., et al., Reliability and validity of palmar and conjunctival pallor for anemia detection purposes. Revista de saude publica, 2003. 37(4) : p. 404-408.
- (5) Sheth, T.N., et al., The relation of conjunctival pallor to the presence of anemia. Journal of general internal medicine, 1997. 12(2) : p. 102-106.

# 氣機科技有限公司實習專案

## 壹、 檢貨系統(獨立完成)

### a. 簡介

檢貨系統為 C# WinForm 開發的 Windows GUI 桌面應用程式，透過 Google Cloud API 提供的 Google sheet API，讀取雲端 Google sheet 上固定格式的產品庫存、訂單資訊，也可以匯入本地端出貨需求，並顯示於檢貨系統，提供使用者檢貨，檢貨完成後，更新雲端 Google sheet 的庫存與訂單資訊並產生出貨單。

### b. 學習到或具備的技能

- (1) 深入了解 C# WinForm，且因為此專案更加了解物件導向。
- (2) Google Cloud API，因為此專案更加了解 API 怎麼運作。
- (3) 敏捷(Agile)開發。
- (4) 更加了解資料結構，並實際運用。
- (5) 學習寫一個大家看得懂的程式碼，使用註解。

### c. 參考照片

使用者: lynn

網址: <https://docs.google.com/s>

請選擇操作系統

自系統 外系統

OrderPicking

姓名: 75 訂單編號: 國內 國外 貨運: 新竹物流

國家: 運送方式:

商品	數量	庫存	已完成
001噴氣源寶			
002盒系通寶			
003送系充寶便重組 (噴氣源寶 + 盒系通寶, 2人)			
003送系充寶便重組 (噴氣源寶 + 盒系通寶 X 2, 3人)			
003送系充寶便重組 (噴氣源寶 + 盒系通寶 X 4, 5人)			
004樹藤晶系-佛			
004樹藤晶系-觀音			
004樹藤晶系-生命之花5cm			
004樹藤晶系-十字架			
004樹藤晶系-SAM			
004樹藤晶系-乾坤			

全部單號 未完成單號 已完成單號

貨品讀取完畢，請「匯入訂單」

自動換下一筆

Aqive 氣機科技



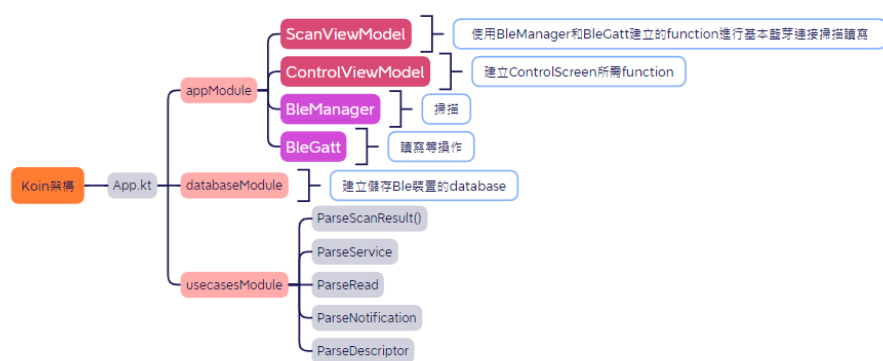
## 貳、 BLE 藍芽 APP(獨立完成)

### a. 簡介

使用 Android Studio 搭配 Jetpack Compose，藉由 kotlin 語言搭配 koin 架構，開發可以連接 BLE(低功耗藍芽)的 APP，並以 GUI 介面操做 BLE Characteristics write、BLE Characteristics read 和 BLE Characteristics notify，用來操控裝置的 RGB LED，以及隨時接收溫濕度氣壓感測器的數據。

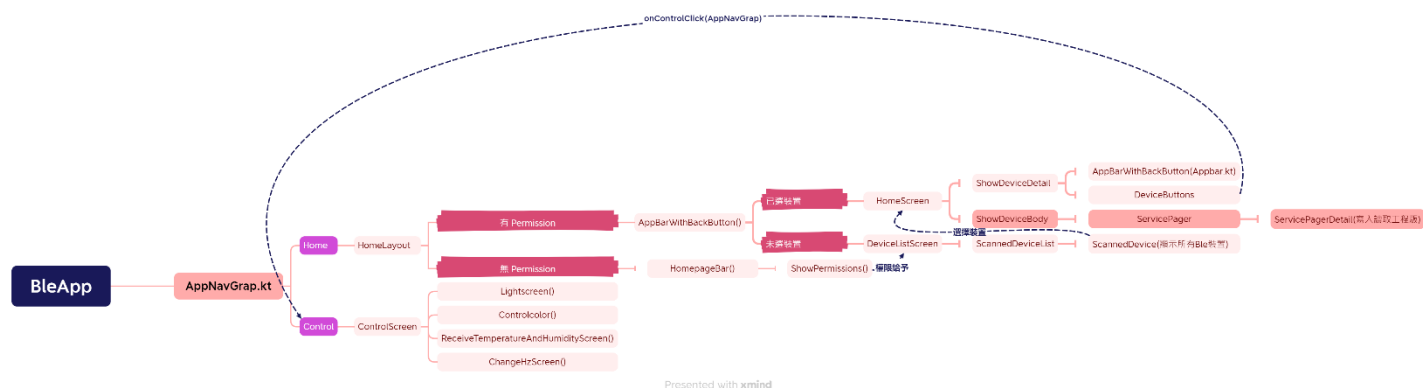
### b. 學習到或具備的技能

- (1) Jetpack Compose 架構 UI 介面。
- (2) Koin 架構。

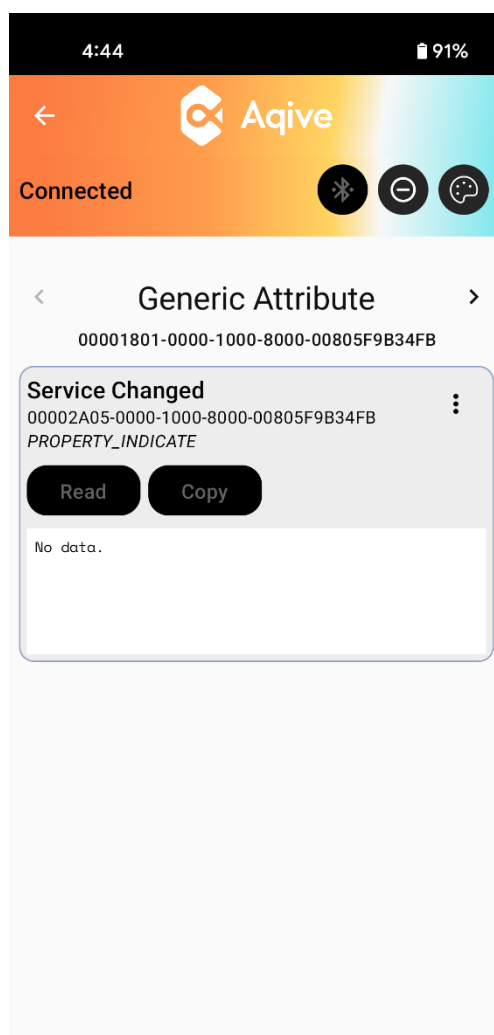


- (3) BLE 低功耗藍芽傳輸方式 GATT(Profile、Service & Characteristic)。
- (4) UI 和 UX 設計。
- (5) 藍芽通訊協定

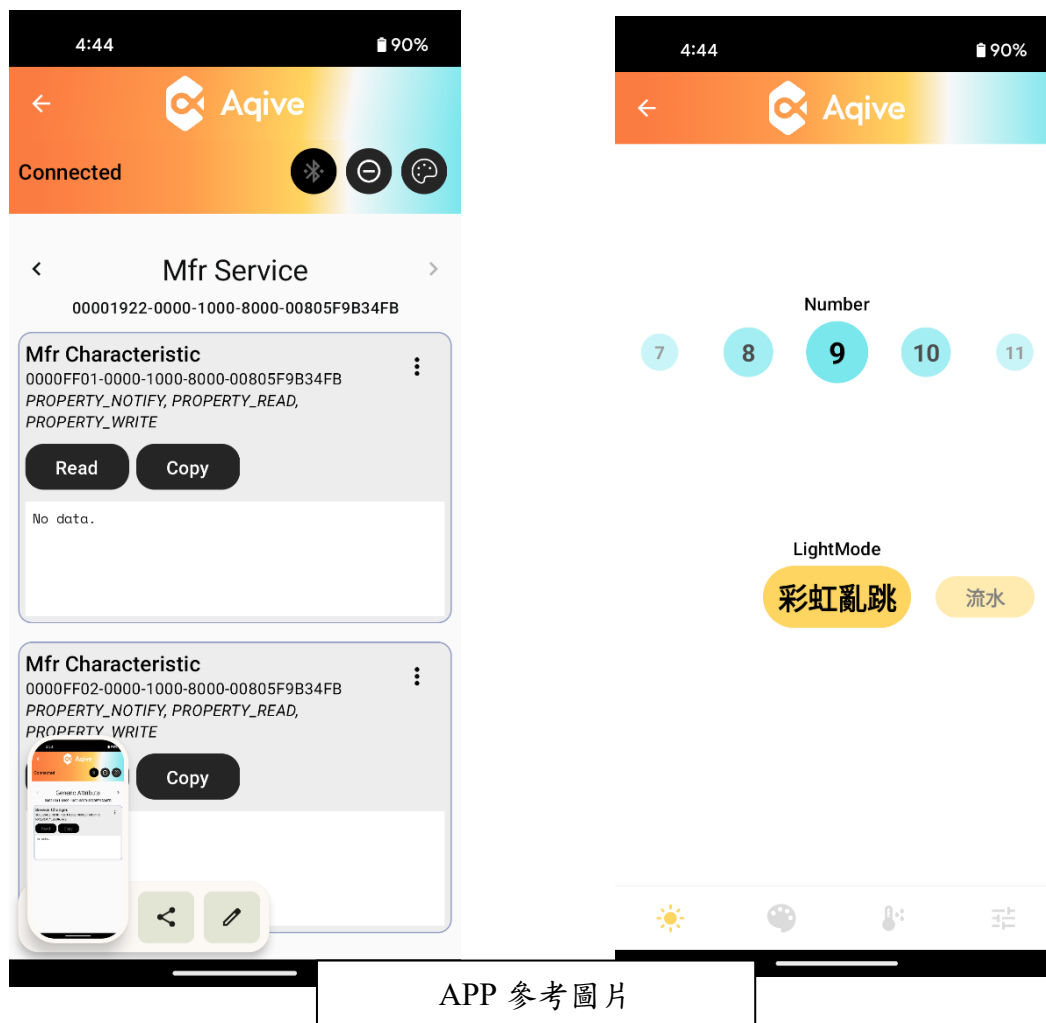
### c. 參考圖片



## BLE 藍芽 APP 架構



APP 參考圖片



APP 參考圖片

## 參、 ESP32 韌體開發(獨立完成)

### a. 簡介

使用 esp-idf 開發 ESP32 MCU，從最基本的 GPIO、GPIO interrupt、esp\_timer、PWM、DAC(數位類比轉換器)與 ADC(類比數位轉換器)等，還有進階的 I2C 溝通其他 IC(Audio CODEC，溫濕度氣壓感測器)、I2S 傳輸音訊、BLE 低功耗藍芽(connect、Characteristics read、Characteristics write、Characteristics notify)、BT 經典藍芽(連接手機，接收手機音訊)以及 RGB 燈效變化(流水燈，呼吸燈以及顏色和亮度變化)。

### b. 學習到或具備的技能

(1) C 語言運用



- (2) 音訊通訊
- (3) 藍芽通訊協定
- (4) I2C、I2S 通訊
- (5) 基本電路知識與如何硬體電路除錯

c. 參考圖片

