

臺北市立大安高級工業職業學校專題實作及創意競賽
「專題組」作品說明書



群別：電機與電子群

作品名稱：The Elder Protector

關鍵詞：跌倒檢測、資料傳輸、年長者安全

目錄

目錄	I
圖目錄.....	III
表目錄.....	IV
【THE ELDER PROTECTOR】	1
壹、摘要	1
貳、研究動機.....	1
參、主題與課程之相關性或教學單元之說明	2
一、硬體製作.....	2
(一)外觀設計與3D列印	2
二、程式撰寫.....	2
(一) Arduino.....	2
(二) GoogleScript.....	3
三、3D繪圖	3
(一)Blender	3
肆、研究步驟.....	4
一、研究流程.....	4
二、使用材料.....	5
三、使用軟體與服務.....	6
四、使用機具.....	7
伍、研究結果.....	8
一、硬體結構.....	8
二、軟體通訊與動作流程.....	9
(一)上傳端.....	10
(二)伺服器端.....	11
(三)發送端.....	11
三、整體結果.....	12
(一)動作說明	12
(二)成果展示	13
陸、討論	14
一、影片記錄與即時影像.....	14
二、聯繫醫療單位.....	14

三、 更合適的材質.....	14
柒、 結論	15
捌、 參考資料.....	16
書面資料.....	16
網路資料.....	16
玖、 附錄	17
一、 作品分工表.....	17
二、 競賽日誌.....	17

圖目錄

圖 1	一些國家老年人口比例	1
圖 2	警用密錄器	1
圖 3	Inventor 的介面	2
圖 4	3D 列印參數調整圖	2
圖 5	Arduino	2
圖 6	GoogleScript	3
圖 7	Blender 介面	3
圖 8	時程	4
圖 9	ESP32	5
圖 10	ESP32-cam	5
圖 11	MPU6050	5
圖 12	鋰電池	6
圖 13	Neo M8n GPS	6
圖 14	結構爆炸圖	8
圖 15	產品實體	8
圖 16	通訊流程圖	9
圖 17	ESP32 中的程式	10
圖 18	ESP32-cam 的程式碼	10
圖 19	Google 表格	11
圖 20	雲端資料夾	11
圖 21	GoogleScript 程式碼	11
圖 22	動作說明流程	12
圖 23	Line 端的訊息	13
圖 24	附加訊息	13

表目錄

表 1	ESP-32.....	5
表 2	ESP-32-cam.....	5
表 3	陀螺儀	5
表 4	鋰電池	6
表 5	GPS.....	6

【The Elder Protector】

壹、摘要

本專題以年長者的安全作為核心概念，設計出一款具有跌倒檢測判斷與資料發送的警報盒，使用陀螺儀進行跌倒的偵測、ESP32-CAM 拍攝影像，透過 ESP32 傳輸與 GoogleScript 的整合技術，搭配 Linebot 的訊息傳送功能，幫助使用者在遇到危機時，可立即與家屬作聯繫，在危機當下即時做出應對措施。

為了避免家屬在收到訊息的當下無法立即處理，我們在此專題中運用蜂鳴器及 LED 燈處理警報的功能，在跌倒的當下，產生燈光與警鈴聲響，讓經過的路人也可協助處理。

貳、研究動機

如圖 1 所示，預估到 2060 年，各國的老年人口比例將逐年提高，其中台灣更是達到 40%。如果老年人的安全未能得到妥善照顧和監控，這將真的是一場災難。跌倒事件經常發生，而且相當危險，採取快速的應對措施至關重要。如果救助延誤，可能會導致危及生命的問題，甚至更嚴重的後果。

為了解決這個問題，我們提出了一個解決方案。設計一款類似執法機關使用的錄音設備(如圖 2)，這些設備通常由警察掛在胸前，因此體積小巧，我們在此基礎上加入一些專為老年人設計的功能，例如當使用者跌倒時，能夠發送帶有現場照片和位置信息的緊急警報。透過這款設備，老年人的安全可以得到有效監控。

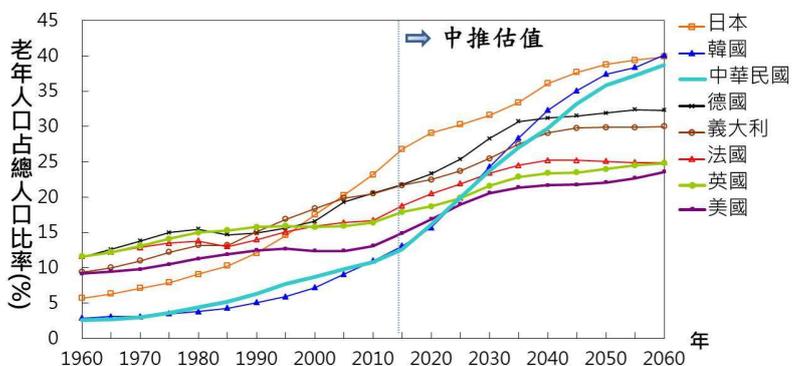


圖 1 一些國家的老年人口比例



圖 2 警用密錄器

參、主題與課程之相關性或教學單元之說明

一、硬體製作

(一) 外觀設計與 3D 列印

我們利用高二彈性課程時所學的繪圖軟體 Inventor 繪製外殼的外觀(如圖 3)，並配合 3D 列印技術(如圖 4)打造外殼的主要結構。

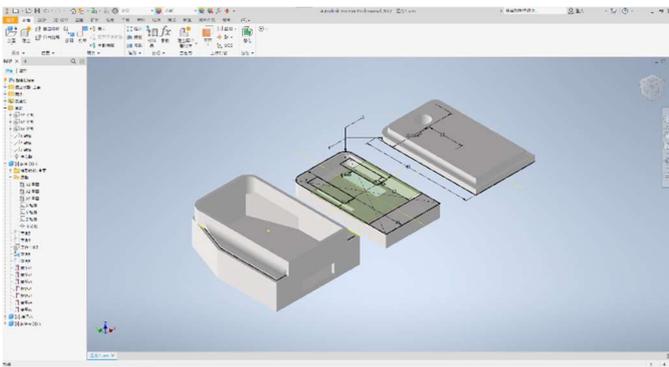


圖 3 Inventor 的繪畫介面

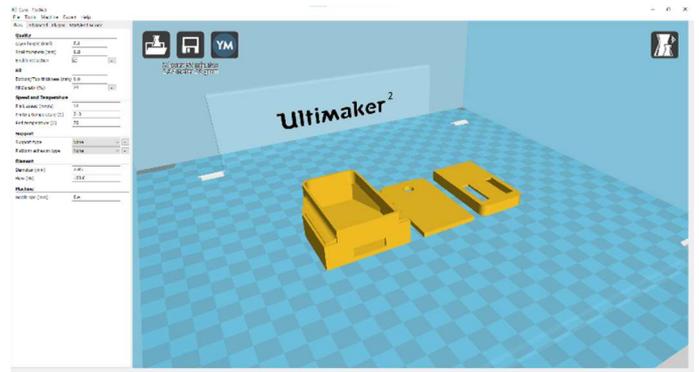


圖 4 3D 列印參數調整

二、程式撰寫

(一) Arduino

我們利用高二智慧居家監控實習中學到的 Arduino (如圖 5)，作為核心晶片 ESP32 和 ESP32-CAM 的驅動程式，使其能連接網路並完成上傳圖片及位置資訊的動作。

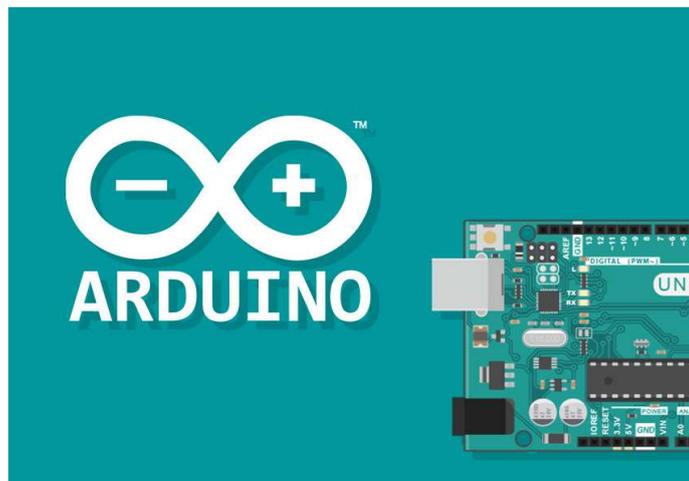


圖 5 Arduino

(二) GoogleScript

我們利用課餘時間自學 GoogleScript(如圖 6) 的語法功能及應用、與 Linebot 的通訊方式等，負責處理 arduino 端上傳的圖片與定位資料等功能。



圖 6 GoogleScript

三、3D 繪圖

(一)Blender

我們使用 Blender 來繪製我們專題的 3D 動畫(如圖 7)，其免費安裝，動畫品質高，不會有卡頓等問題，也有許多輔助插件可供使用。



圖 7 Blender 的介面

肆、研究步驟

一、研究流程

在六月底訂下專題題目後，組員們開始卯足全力查找相關網路資料，並在七月初發現了以目前的實力不足以支撐我們製作，並開始思考以及學習，八月正式著手基礎程式編寫。九月初做第一次把 ESP32CAM 和雲端連結並傳送照片。九月中旬遇到了問題因為 CAM 運算能力不足且傳送資料不穩定所以進程式優化。在十月時開始硬體設計，並優化軟體系統和硬體接線。到了十一月尾聲機殼也被設計完成，並使用 3D 列印技術做出第一版的機殼。其餘時間皆花在程式優化的收集和機殼外觀的微調，且在隔年一月初成功完成，最後測試了陀螺儀穩定性和動作影片拍攝，整個 The Elder Proctor 研究到此告一個段落。其大致時程如下圖 8。



圖 8 時程表

二、使用材料

(一) ESP-32

我們使用該晶片作為本專題的核心晶片，進行功能整合及控制使用者與家屬間的通訊方式。其外觀如圖 9，規格如表 1



圖 9 ESP-32

表 1 ESP-32

ESP-32	
尺寸	50 *23 *11.0 mm
工作電壓	2.7 V ~ 3.6 V
工作電流	80 mA

(二) ESP-32-cam

ESP-32-cam 是一款具拍照功能的晶片，我們使用此元件進行跌倒時的影像拍攝。其外觀如圖 10 規格如表 2。

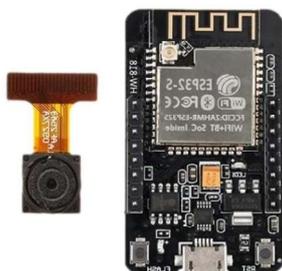


圖 10 ESP-32-cam

表 2 ESP-32-cam

ESP-32-cam	
尺寸	27*40.5*4.5mm
工作電壓	3.3 ~5V
工作電流	6.7 ~310mA

(三) 陀螺儀

我們選用陀螺儀 MPU6050 檢測加速度的變化，用於跌倒的判定。其外觀如圖 11 規格如表 3。

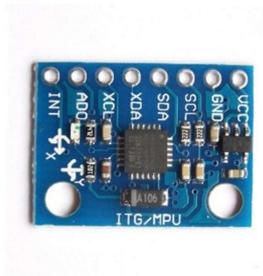


圖 11 陀螺儀

表 3 陀螺儀

MPU6050	
電壓	3~5V
陀螺儀範圍	±250 500 1000 2000 °/s
加速度範圍	±2 ±4 ±8 ±16g

(四) 鋰電池

我們選用鋰電池作為整個專題的供電來源。其外觀如圖 12 規格如 4。



圖 12 鋰電池

表 4 鋰電池

鋰聚合物電池	
電壓	3.7V
容量	1600mAh

(五) GPS 模組與天線

我們選用此元件作為定位的工具，藉由精準的定位功能讓家屬知道使用者的確切位置。其外觀如圖 13 規格如 5。



圖 13 GPS

表 5 GPS

Neo M8n GPS	
電壓	3~5V
尺寸	25*35mm

三、使用軟體與服務

(一) Autodesk Inventer

一款能輕鬆入門的工業繪圖軟體，擔任這次專題的硬體設計擔當，我們用它來建模和繪製草圖，是硬體製作的根本。

(二) Line Developer

構建 Linebot 處，負責發送緊急訊息。

(三) Ultimaker Cura

3D 列印的編譯程式，負責調整 3D 列印時的各項參數。

(四) Google Script

資料處理中樞，處理圖片與 GPS 資訊，傳送至 Linebot。

(五) Arduino IDE

Arduino 語法的開發環境，能配合多種晶片進行編譯和多樣的程式庫可供使用。我們用其對 ESP-32 晶片進行程式的編譯。

(六) Blender

我們使用 blender 製作 3D 動畫，其功能多元加上動畫精緻，很多細節都有處理好，且完全免費，因此我們使用 blender 製作 3D 動畫。

四、使用機具

(一) 3D 列印機

我們採用多種型號的 3D 列印機製作硬體外觀，這種方法具有極高的自由度，且無需使用傳統的模板，正是我們所青睞的特點。

(二) 鑽孔機

負責加工專題外殼，以提供天線放置的位置。

伍、研究結果

一、硬體結構

以 PLA 材料為主體，做成一個裝載電子元件的六邊形容器，內部構造如圖 14，實體如圖 15。

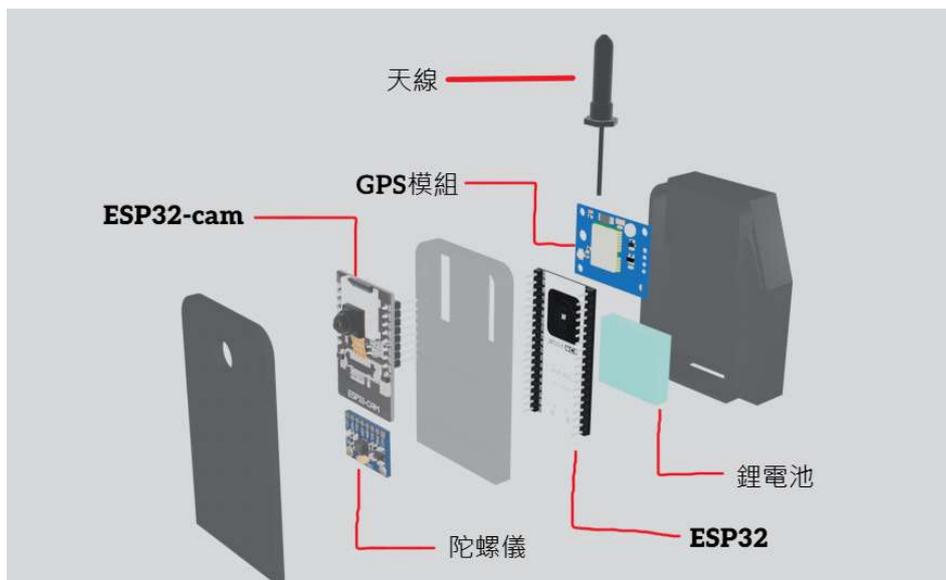


圖 14 結構爆炸圖



圖 15 產品實體

二、軟體通訊與動作流程

通訊方面，我們選擇使用網路通訊，原因有兩點：第一，我們使用雲端硬碟作為伺服器處理數據，網路相較於藍芽，傳輸比較穩定；第二，藍芽發送數據需要經過與藍芽連結之設備處理數據才能上傳至雲端，而 wifi 可直接上傳至雲端，故我們選擇使用網路。

連接網路後，我們使用 http 協議向雲端發送請求，將圖片與定位傳送雲端，因為 http 協議支援多種資料類型，如圖片、文件等，資料也有保密性。通訊過程中我們使用了兩種程式語言和一種協議來相互配合，從 Arduino 的類 C 語言，透過 http 協定傳送數據到雲端，最後使用 GoogleScript 語言連接並呼叫 Line Messaging API 從雲端抓取數據傳送到用戶端上，架構大致如下圖 16。

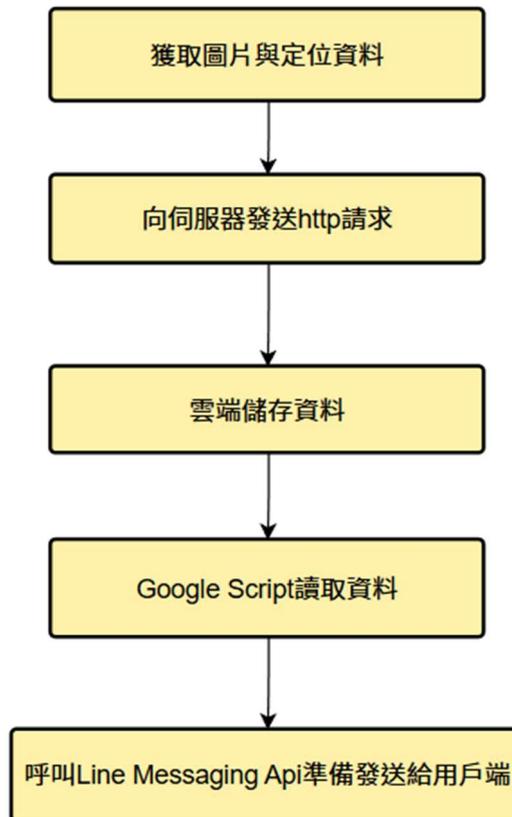


圖 16 通訊流程圖

(一) 上傳端

檢測到跌倒情況後，ESP32 記錄 GPS 數據、ESP32-cam 拍攝圖片，把這兩種資料透過 http 協定傳送到伺服器，如圖 17、圖 18 分別為將位置資訊、圖片傳送到雲端的程式。

```
HTTPClient http;
String url="https://script.google.com/macros/s/"+GOOGLE_SCRIPT_ID+"/exec?"+params;
//Serial.print(url);
Serial.println("Posting GPS data to Google Sheet");
//-----
//starts posting data to google sheet
http.begin(url.c_str());
http.setFollowRedirects(HTTPC_STRICT_FOLLOW_REDIRECTS);
int httpCode = http.GET();
Serial.print("HTTP Status Code: ");
Serial.println(httpCode);
//-----
//getting response from google sheet
String payload;
if (httpCode > 0) {
  payload = http.getString();
  Serial.println("Payload: "+payload);
}
//-----
http.end();
```

圖 17 ESP32 中的程式

```
Serial.println("Sending image to Google Drive.");
Serial.println("Size: " + String(fb->len) + "byte");

String url = "/macros/s/" + myDeploymentID + "/exec?folder=" + myMainFolderName;

client.println("POST " + url + " HTTP/1.1");
client.println("Host: " + String(host));
client.println("Transfer-Encoding: chunked");
client.println();

int fbLen = fb->len;
char *input = (char *)fb->buf;
int chunkSize = 3 * 1000; //--> must be multiple of 3.
int chunkBase64Size = base64_enc_len(chunkSize);
char output[chunkBase64Size + 1];
```

圖 18 ESP32-cam 中的程式碼

(二) 伺服器端

伺服器端分成兩邊，一邊為放置位置資訊的 Google 試算表，其中包含日期、經緯度等(如圖 19)，其中 Current Data 為是否為及時數據的標準；另一邊為檔案資料夾，圖片就會存放在資料夾裡(如圖 20)。有了放資料的地方就能讓 API 讀取了。

	A	B	C	D	E
1	Timestamp	Latitude	Longitude	GPS Time	Current Data
2	2025/1/8	[REDACTED]	[REDACTED]	16:32	1
3	2025/1/9	[REDACTED]	[REDACTED]	21:56	1
4	2025/1/9	[REDACTED]	[REDACTED]	22:30	0
5	2025/1/11	[REDACTED]	[REDACTED]	14:22	1
6	2025/1/12	[REDACTED]	[REDACTED]	18:07	1
7					
8					

圖 19 放置位置資訊的 Google 表格



圖 20 檔案資料夾

(三) 發送端

資料上傳完畢後，執行如圖 21 的 Google Script 程式碼與 Line Messaging API 做連接，使其能夠讀取資料夾得圖片以及表格中的位置資訊，整合到這些數據後，透過 Line 發送資料給用戶。

```
for (var i = 0; i < messages.length; i += 5) {
  var payload = {
    "to": userId,
    "messages": messages.slice(i, i + 5)
  };

  var options = {
    "method": "post",
    "headers": {
      "Content-Type": "application/json",
      "Authorization": "Bearer " + accessToken
    },
    "payload": JSON.stringify(payload)
  };

  UrlFetchApp.fetch("https://api.line.me/v2/bot/message/push", options);
  Logger.log("Sent images batch: " + (i / 5 + 1));
}
```

圖 21 Google Script 程式碼

三、整體結果

(一) 動作說明

平常為待機模式，無動作，若陀螺儀檢測到使用者跌倒時，ESP32-cam 會拍攝現場圖片；ESP32 會從 GPS 獲取定位資訊，兩者將資料上傳到雲端後，雲端中的程式碼會呼叫 line Messaging API 讀取並整理訊息後發送到用戶端。整體動作流程如圖 22。

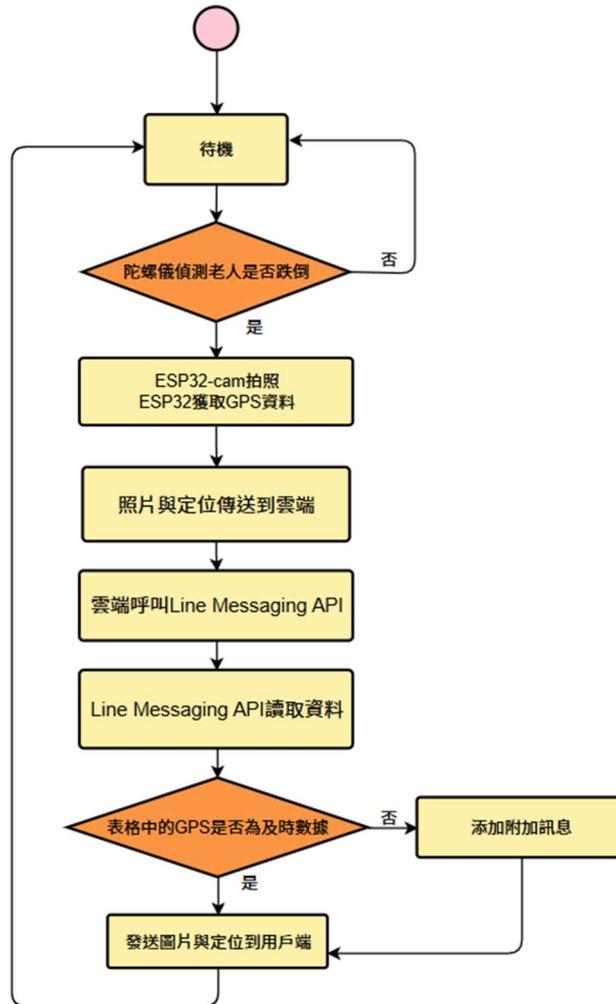


圖 22 動作流程圖

(二) 成果展示

如圖 23 為用戶端看到的訊息，其中包含圖片與位置資訊，而我們也有想到如果使用者在室內跌倒之情況，室內會使 GPS 無法收到即時的定位資訊，對此我們想了解解決辦法，若使用者在 GPS 無法及時更新的狀況下，我們採用附加訊息提醒該定位不是即時的數據(如圖 24)。



圖 23 Line 端的訊息



圖 24 含附加訊息的位置資訊

陸、討論

一、影片記錄與即時影像

目前我們的專題僅做到在事發當下拍攝圖片，但圖像較難記錄下完整的過程，因此我們希望能用影片來取代圖像，透過影片可清楚了解配戴者發生危險的原因。

除了影片外，我們認為傳送即時影像也是可以新增的功能，影片主要是當記錄保存而即時影像則能讓配戴者的家屬更加理解現場狀況。

為了實現上述功能，我們希望能使用一些容量更充裕的元件，例如使用 SD 卡，提供更大的儲存空間，以及伺服器，讓用戶端不論在何地皆能查看即時影像。

二、聯繫醫療單位

在遇到更為嚴重的傷害時，聯絡家屬的同時，更重要的是與醫療單位的聯繫，為了提升救援效率，自動發送通知至醫護端是能夠優化本專題的功能。

為此，我們希望能想出其他的通訊方法，包括如何與醫療單位做聯繫以及事故嚴重程度的判別，未來會在裝置外殼上新增一個按鈕，讓使用者或旁人視當下的情況決定是否有通知醫護人員的必要，同時也能解決誤報的問題。

三、更合適的材質

在有限的資源下，我們使用 3D 列印 PLA 作為外殼的材質，但其材質較堅硬，因此受到衝擊時也較容易碎裂，影響產品的使用壽命甚至是損壞，另外堅硬材質對年長者來說衍伸出了潛在風險，例如設備碎裂後，其碎屑可能刮傷人，或是萬一老人跌倒壓到外殼時，會壓迫到老人的身體造成內傷。

因此未來我們想使用像是矽膠或是 TPU 等柔性高的材料，解決設備易碎問題，耐用性也高，保護內部元件的同時也能讓年長者配戴起來更舒適、更安全。

柒、結論

本專題以「檢測跌倒」為主題，旨在解決高齡化社會中老人安全與照護的問題。透過整合硬體製作（3D 繪圖與列印）和程式撰寫（Arduino），開發出一款能進行跌倒判斷並發出警報及訊息，進而即時提供長者協助的工具。

硬體製作涵蓋 3D 列印的外殼之技術，並透過 Arduino 控制晶片、陀螺儀的判斷及 Linebot 的設置來實現各項功能。

主要功能為加速度檢測晶片 MPU6050 進行測量，並透過 ESP-32-CAM 晶片自創的拍照技術，拍攝事發當下的照片，經過 Linebot，傳到用戶的手機。

研究時，我們進行了硬體製作、軟體開發和測試，包括連結 Linebot、設計外殼等。此外，透過了解跌倒時的瞬間加速度變化，使其對於跌倒的判定更加準確，大幅減少誤發警訊的情形，提升了檢測的精準度。

未來會朝向，傳送即時影像、聯繫醫療單位、探索更小巧的產品以及更合適的材質。這些方向將進一步優化產品性能，提高使用上的便利性，符合專題初衷，解決高齡化延伸的問題。

這個專題結合了社會議題和科技，成功地開發了一款具有保護長者功能的工具，為家屬提供更方便、安心照護長輩的方法，同時展望未來有進一步的發展和改進空間。

捌、 參考資料

書面資料

- 1.江秉潤, 陳飛亨 (2015/03/31)。3D 設計 for 3D 列印超入門：Blender 快速建模全書。基峰。
- 2.陳明熒 (2020/11/30)。Arduino 實作入門與專題應用。博碩文化。
- 3.清水曉生 (2023/05/25)。圖解電子電路：元件功能構成 x 設計手法 x 實做流程，實現未來科技的電子製作應用實務書。易博士。
- 4.姚家琦 (2019/02/25)。Autodesk Inventor 電腦繪圖與輔助設計。基峰。
- 5.張宗彥 (2022/12/20)。一本精通 - LINE BOT + Python + Google Dialogflow 完整掌握 LINE BOT 的開發技巧 打造全方位 AI 機器人。深智。
- 6.安守中 (2018/12/01)。GPS 定位原理及應用。全華圖書。

網路資料

1. Cubie(2022/2/22)。ESP32-CAM 開發板 (一)：簡介與燒錄程式。
<https://swf.com.tw/?p=1723>
2. Cian(2021/10/21)。淺談 Base64 與應用實例分享。
<https://blog.errorbaker.tw/posts/cian/base64-qr-code/>
3. Ahmad Logs(2021/10/25)。ESP32 GPS Data Logger with Google Sheets。
<https://www.youtube.com/watch?v=wQGqydQUwRw&list=LL&index=5>
4. Ben L. Collins(2021/4/16)。Google Apps Script: A Beginner's Guide。
<https://www.benlcollins.com/apps-script/google-apps-script-beginner-guide/>
5. Apifox(2024/7/5)。什么是 HTTP POST 请求？初学者指南与示范。
。 https://blog.csdn.net/m0_71808387/article/details/140204629
6. Ceiling Tsai(2019/9/30)。Arduino 筆記(71)：MPU-6050 (GY-521) 三軸陀螺儀+三軸加速計感測模組。
<https://atceiling.blogspot.com/2019/09/arduino57mpu-6050-gy-521.html>
7. Andreas Harth(2023/7/22)。你了解 Line Bot 嗎？一步步深入探索這個聊天機器人的奧秘。
<https://www.pintech.com.tw/tw/article/321/what-is-line-bot>

玖、附錄

一、作品分工表

參賽學生	工作任務
A	海報製作、PPT 製作
B	英文逐字稿、報告
C	機構設計、硬體組裝、影片製作
D	程式撰寫、影片製作、說明書、3D 動畫

二、競賽日誌

年	月	日	進度	紀錄	工作分配
113	5	26	資料蒐集	地點：教室 器材：手機、電腦 時數：3 小時	同學 A：資料查詢 同學 B：資料查詢 同學 C：資料查詢 同學 D：資料查詢
113	6	6	討論主題	地點：教室 器材：手機、筆電、紙 時數：3 小時	同學 A：討論主題 同學 B：討論主題 同學 C：討論主題 同學 D：討論主題
113	6	20	結構討論 功能討論	地點：教室 器材：手機、筆電、紙 時數：3 小時	同學 A：功能討論 同學 B：功能討論 同學 C：結構討論 同學 D：結構討論
113	7	2	材料購買	地點：網路商場、光華商場 時數：3 小時	同學 A：材料購買 同學 B：材料購買 同學 C：材料購買 同學 D：材料購買
113	7	25	討論專題細節	地點：科辦 器材：手機、紙 時數：1 小時	同學 A：功能討論 同學 B：功能討論 同學 C：結構討論 同學 D：結構討論
113	8	2	結構設計	地點：實習工廠 器材：手機、筆電 時間：3 小時	同學 A：簡報討論 同學 B：簡報製作 同學 C：結構討論 同學 D：程式撰寫

			簡報製作		
113	9	12	初步校正 簡報製作	地點：實習工廠 器材：手機、筆電 時間：6 小時	同學 A：簡報製作 同學 B：簡報製作 同學 C：結構討論 同學 D：程式撰寫
113	9	18	3D 建模 簡報製作	地點：實習工廠 器材：手機、筆電、3D 列印機 時間：6 小時	同學 A：3D 建模 同學 B：簡報製作 同學 C：3D 列印 同學 D：程式撰寫
113	9	23	軟體測試 3D 建模	地點：實習工廠 器材：手機、筆電、3D 列印機 時間：7 小時	同學 A：3D 建模 同學 B：簡報製作 同學 C：3D 列印 同學 D：程式撰寫
113	9	25	模組開發 硬體製作	地點：實習工廠 器材：手機、筆電 時間：7 小時	同學 A：3D 建模 同學 B：簡報製作 同學 C：電路設計 同學 D：程式撰寫
113	9	26	報告製作	地點：實習工廠 器材：手機、筆電 時間：6 小時	同學 A：3D 建模 同學 B：簡報製作 同學 C：硬體整合 同學 D：程式撰寫
113	9	27	機構修改	地點：實習工廠 器材：手機、筆電 時間：7 小時	同學 A：大數據收集 同學 B：大數據收集 同學 C：硬體設計 同學 D：程式撰寫
113	10	28	3D 建模	地點：實習工廠 器材：手機、筆電、3D 列印機 時間：7 小時	同學 A：3D 建模 同學 B：大數據收集 同學 C：硬體整合 同學 D：大數據收集
113	11	16	動畫製作	地點：實習工廠 器材：手機、筆電、3D 列印機 時間：7 小時	同學 A：3D 建模 同學 B：動畫製作 同學 C：硬體整合 同學 D：動畫製作

			3D 建模		
113	12	23	海報製作	地點：實習工廠 器材：手機、筆電 時間：7 小時	同學 A：3D 建模 同學 B：海報製作 同學 C：硬體微調 同學 D：程式撰寫
114	1	1	簡報製作	地點：實習工廠 器材：手機、筆電、 時間：7 小時	同學 A：硬體整合 同學 B：簡報製作 同學 C：硬體整合 同學 D：軟體微調
114	1	3	影片構思	地點：實習工廠 器材：手機、筆電 時間：7 小時	同學 A：影片構思 同學 B：影片構思 同學 C：影片構思 同學 D：影片構思
114	1	4	影片拍攝	地點：實習工廠 器材：手機、筆電 時間：7 小時	同學 A：影片拍攝 同學 B：影片拍攝 同學 C：影片拍攝 同學 D：影片拍攝
114	1	7	影片拍攝、剪輯 最終測試	地點：實習工廠 器材：手機、筆電 時間：7 小時	同學 A：影片拍攝 同學 B：影片拍攝、剪輯 同學 C：書面報告製作 同學 D：影片拍攝、剪輯
114	1	9	說明書製作 最終測試	地點：實習工廠 器材：手機、筆電 時間：7 小時	同學 A：簡報修改 同學 B：報告練習 同學 C：最終測試 同學 D：最終測試、說明書