

目錄

壹、 企劃需求	3
1.1 簡介	3
1.2 架構大綱.....	3
貳、 系統需求	4
2.1 資料庫與傳輸	4
2.1.1 資料庫與傳輸功能性需求.....	4
2.1.2 資料庫與傳輸非功能性需求.....	5
2.2 網路 API.....	5
2.2.1 網路 API 功能性需求	5
2.2.2 網路 API 非功能性需求	5
2.3 演算法	6
2.3.1 演算法功能性需求	6
2.3.2 演算法非功能性需求	6
2.4 圖樣辨識	6
2.4.1 圖樣辨識功能性需求	6
2.4.2 圖樣辨識非功能性需求	6
2.5 使用者介面	7

2.5.1 使用者介面功能性需求	7
2.5.2 使用者介面非功能性需求.....	8
2.6 硬體	8
2.6.1 硬體功能性需求	8
2.6.2 硬體非功能性需求	8
參、 使用需求	9
3.1 前端需求.....	9
3.1.1 前端功能性需求	9
3.1.2 前端非功能性需求	9
3.2 後端需求	10
3.1.1 後端功能性需求	10
2.7.2 後端非功能性需求	10
肆、 使用案例	11
伍、 關聯	14
5.1 系統需求 V.S 系統需求 矩陣	14
5.2 使用需求 V.S 系統需求 矩陣.....	15
5.3 系統需求 V.S 使用案例 矩陣.....	16

壹、企劃需求

1.1 簡介

因應現代技術所帶來的改變與 自動化趨勢，同時為了緩解忙碌的早晨，本研究計畫預計將完成一台 智能衣櫃。其中技術項目包含衣物 辨識，使使用者將衣物放入衣櫃後 由系統自動辨識並歸類該衣物，並將衣物通過道軌收入衣櫃。使用時 只需要將衣物放入衣櫃就可以在面板上查看所有的衣物資訊，解決 衣服的收納與歸類問題。

同時此衣櫃包含一套衣物推薦演 算法，該演算法會結合使用者設定的 喜好度/顏色/種類等資訊為使用者推 薦數套衣物組合，並且系統會時時自 動抓取天氣資訊供使用者進行考量，使用者只需點點手指就可從電子面板 取得以上資訊。在選擇好心儀的組合 後，衣櫃就會自動將衣服透果軌道輸 送至使用者眼前，因此使用者不需再 翻箱倒櫃的找衣服或選擇困難，利用機器就可完成所有事物。

1.2 架構大綱

以下是系統的主要功能和系統架構：

功能	描述
網路 API	擷取網路資訊(如天氣等)的功能
演算法	衣服推薦的演算法
辨識系統	提供辨識衣服的功能
人機介面	使用者與系統互動之介面
硬體	系統實體架構

貳、系統需求

2.1 資料庫與傳輸

2.1.1 資料庫與傳輸功能性需求

資料庫與傳輸功能性需求 Database functional requirement	
DBFR1	儲存資料
	儲存系統資料，其中包含： 1. 使用者資料 1.1 使用者基本資料 1.2 使用者衣物資料 1.3 使用者衣物使用資料 2. 天氣資訊 3. 衣櫃資料 *任何需要保存的資料皆有對應的資料表與欄位
DBFR2	修改資料
	資料庫須包含以下功能： 1. 新增資料 2. 修改資料 3. 刪除資料 *每一個 table 都要有基本的新增單筆資料的功能
DBFR3	查詢資料
	資料庫需可供系統查詢資料，同時須因使用者不同而分別提供不同資料，詳情請見(2.5 使用者介面)
DBFR4	架設需求
	不同主機需可假設相同資料庫環境，只需安裝本地環境皆可運行，可以執行的環境需包含： 1. MacOS 2. Windows

2.1.2 資料庫與傳輸非功能性需求

資料庫與傳輸功能性需求 Database non-functional requirement	
DBNFR1	處理速度
	90%的請求可以在 1 秒內完成
DBNFR2	資料正規化
	正規化並利用視圖串接來優化速度
DBNFR3	儲存空間限制
	每筆資料將被設定最大儲存大小，超過將被進行壓縮處理

2.2 網路 API

2.2.1 網路 API 功能性需求

網路 API 功能性需求 Internet API functional requirement	
IAFR1	抓取天氣資訊
	使用爬蟲與中央氣象局提供之 API 獲取所有站別的天氣資訊並送入資料庫，資料包含： 1. 溫度 2. 濕度 3. 最高溫 4. 最低溫 5. 最高溫時間段 6. 最低溫時間段

2.2.2 網路 API 非功能性需求

網路 API 非功能性需求 Internet API non-functional requirement	
IANFR1	執行速度
	需在規定秒數內執行完畢，若超過須執行替代方案
IANFR2	替代方案
	以下狀況發生時應有替代方案： 1. 未聯網 2. 執行時間過長

2.3 演算法

2.3.1 演算法功能性需求

演算法功能性需求 Algorithm functional requirement	
AGFR1	衣物推薦演算法
	演算法根據系統所取得之天氣資訊、使用者偏好度、衣物資訊等進行演算，推薦出數套(4~8)衣物組合。

2.3.2 演算法非功能性需求

演算法非功能性需求 Algorithm non-functional requirement	
AGNFR1	執行速度
	需在規定秒數內執行完畢

2.4 圖樣辨識

2.4.1 圖樣辨識功能性需求

圖樣辨識功能性需求 Pattern recognition functional requirement	
PRFN1	辨識衣服
	透過攝影機拍下的相片，對衣服の種類與顏色進行辨識，並將數值分類至相近的預設選項作為結果，最終回傳至系統

2.4.2 圖樣辨識非功能性需求

圖樣辨識非功能性需求 Pattern recognition non-functional requirement	
PRNFN1	辨識速度
	要求在 5 秒內辨識完成
PRNFN2	結果一致
	辨識出來的結果與原本的衣服結果一樣。
PRNFN3	性能
	辨識期間 CPU 占用率≤50%。
PRNFN4	成功率
	辨識成功率達八成以上
PRNFN5	缺陷率
	系統每辨識 10000 次最多發生 1 次故障

2.5 使用者介面

2.5.1 使用者介面功能性需求

使用者介面功能性需求 User interface functional requirement	
UIFR1	人機介面
	提供使用者一個合理方便的人機介面，此介面需依照觸控面板方向設計，並彙整產品功能，設計清楚易懂的畫面與說明，需求分支請見(UIFR2~UIFR7)，功能說明請見企畫書(第叁章/三、其他/f. 人機介面設計)
UIFR2	主選單
	提供跳至各項功能之選單
UIFR3	使用者介面
	提供使用者登入、登出、些改使用者資料之介面
UIFR4	收納介面
	提供使用者使用收納功能，頁面須滿足以下幾點： 1. 分為舊衣收納(已經放入過後拿出來的衣服)與新衣收納 2. 新衣收納需顯示辨識結果與照片並提供修正辨識結果之選項 3. 舊衣收納需顯示當前照片與比對結果之照片與資訊，並可供使用者手動替換比對結果
UIFR5	拿取介面(系統推薦)
	顯示系統推薦的衣物組合之圖片供使用者選擇
UIFR6	拿取介面(自行選擇)
	提供使用者選擇欲拿取之衣物選擇介面，頁面須滿足以下幾點： 1. 有大項分類選單 2. 有細項分類選單 3. 有衣物的照片 4. 是否存在衣櫃內的衣物需有顯示區別 5. 提供單件 or 雙件拿取選項
UIFR7	衣物設定介面
	後期設定衣物資訊(顏色、樣式、喜愛度等)之介面
UIFR8	使用說明介面
	提供使用者操作說明

2.5.2 使用者介面非功能性需求

使用者介面非功能性需求 User interface non-functional requirement	
UINFR1	防呆設計
	要求介面應有適合的防呆設計與提示
UINFR	操縱性
	介面應便於操作、易懂、乾淨，並依使用者體驗進行二次優化

2.6 硬體

2.6.1 硬體功能性需求

硬體功能性需求 Hardware functional requirement	
HWFR1	衣服存取功能
	提供衣服存取功能，該功能需滿足以下條件： 1. 可隨時取得任意一件衣服 2. 可同時處理兩件衣服
HWFR2	拍照功能
	衣櫃需有拍照功能，拍照須滿足以下條件： 1. 圖片清晰 2. 亮度充足 3. 確保拍到整件衣服正面
HWFR3	人機介面操控功能
	提供觸控螢幕供使用者對衣櫃進行操作

2.6.2 硬體非功能性需求

硬體非功能性需求 Hardware non-functional requirement	
HWNFR1	耐久度
	1. 須保證各部分可承重預設的兩倍重量不變形 2. 須保證各部分零件在 5000 次內使用無虞
HWNFR2	外觀
	外觀需簡潔並包含一定設計感
HWNFR3	使用體驗
	各部分用作需朝以下目標完成： 1. 運作速度快速(ex. 拿取需在 20 秒內完成) 2. 使用噪音與震動不可過大

參、使用需求

3.1 前端需求

3.1.1 前端功能性需求

前端功能性需求 Fornt-end functional requirement	
FEFR1	建立使用者資料
	每個使用者皆可建立自己的使用者資料，其中包含： 1. 使用者名稱 2. 使用者所在地 3. 使用者的衣物與使用狀況
FEFR2	儲存衣服
	使用者可將衣服存入衣櫃，並且只需將衣物掛入即可
FEFR3	拿取衣服(推薦)
	使用者可獲取數套系統推薦之衣物組合，並依此拿取
FEFR4	拿取衣服(自選)
	使用者可自行透過分類選單拿取衣服
FEFR5	獲取天氣資訊
	使用者可查詢所在地的天氣資訊
FEFR6	查詢、修改衣物資訊
	使用者可對衣物資料進行以下動作： 1. 查詢衣物資料 2. 修改衣物資料 3. 將衣物設為最愛 4. 查詢使用狀況(註:需可進行類似 top10 排名之功能)

3.1.2 前端非功能性需求

前端非功能性需求 Fornt-end non-functional requirement	
FENFR1	操作性
	各系統(硬體、軟體)皆需便於操作，並朝向後期優化 UX 體驗為目標

3.2 後端需求

3.1.1 後端功能性需求

後端功能性需求 Back-end functional requirement	
BEFR1	Model 架構
	系統之各項功能應滿足 Model 架構，以對應所有資料庫欄位進行傳輸
BEFR2	Controller 架構
	系統之各項功能應滿足 Controller 架構，利用 firmata 協定與 Arduino 進行連接
BEFR3	View 架構
	系統之各項功能應滿足 View 架構，利用 EEL 讓 JavaScript 與 Python 溝通

2.7.2 後端非功能性需求

後端非功能性需求 Back-end non-functional requirement	
BENFR1	Controller 架構

肆、使用案例

Use Case ID	UC-US-01
Use Case Name	收入衣服
目標	將衣服收入衣櫃中
需求	DBFR1- PRFN1- HWFR2- UIFR1- UIFR4- HWFR2- HWFR3
描述	使用者透過將衣物掛入插槽將衣服收入衣櫃中
前置條件	檢視口的位子需有模塊存在
後置條件	成功將衣物收入至衣櫃中
流程	<ol style="list-style-type: none"> 1. 使用者點選 UI 介面中的存放功能 2. 開啟微動開關檢視信號 3. 使用者掛入衣物 4. 微動開關接收線號 5. 啟動攝像頭拍照並存入本地端 6. 對衣物進行顏色及樣式辨識 7. 與資料庫對比是否有相似資料 <ol style="list-style-type: none"> a. 有相似資料(詢問使用者是否為過去某筆資料) <ol style="list-style-type: none"> i. 將衣物資料是更新至資料庫中 b. 無相似資料 <ol style="list-style-type: none"> ii. 將衣物資料新增至資料庫 8. 啟動馬達將衣物收入
替代的使用案例	
使用的使用案例	
延伸的使用案例	
特殊要求	
案例詞彙公式	衣物=衣物本體+衣物資料
備註	

ID 說明

UC-US-01:使用部分(含物體的 如拿取/收入)

UC-DA-01:資料管理部分(如使用者資料/衣物資料管理)

UC-OT-01:其他部分(如鬧鐘使用)

Use Case ID	UC-US-02
Use Case Name	拿取衣服(系統推薦)
目標	將衣物從衣櫃中取出
需求	DBFR3- UIFR1—UIFR2-UIFR5- HWFR1- HWFR3
描述	使用者透過人機介面選擇想要的衣物，機器將會自動推送至檢視口，方便拿取。
前置條件	衣櫃內部有衣物存放中
後置條件	衣服成功從衣櫃推送出來
流程	<ol style="list-style-type: none"> 1. 使用者點選 UI 介面中的拿取功能 2. 開啟天氣抓取 + 衣物推薦演算法將推薦組合獲得組合 3. 進入拿取介面(系統推薦)四套衣物組合並提供使用者進行點選 <ol style="list-style-type: none"> a. 使用者選擇由系統所推薦的衣物(執行 4.) b. 使用者選擇非系統所推薦的衣物 <ol style="list-style-type: none"> i. 進入拿取介面(自行選擇) ii. 選擇要拿取的衣物數量(一件／一套) iii. 選擇衣物 iv. 送出 4. 根據選擇的衣物由馬達將其從衣櫃內部送出
替代的使用案例	
使用的使用案例	
延伸的使用案例	
特殊要求	
案例詞彙公式	
備註	

Use Case ID	UC-DA-01
Use Case Name	衣物資料設定
目標	存放在資料庫的衣物資料可以隨時變更。
需求	DBFR2- DBFR3-UIFR1-UIFR2-UIFR7- HWFR3
描述	使用者可以根據過往紀錄以及目前存在衣櫃內部的所有衣物資料進行參數上的設定。
前置條件	衣物曾存放過
後置條件	正確修改該衣物設定
流程	<ol style="list-style-type: none"> 1. 使用者進入衣物設定畫面 2. 選擇想變更的衣物設定 <ol style="list-style-type: none"> a. 顏色搭配 b. 衣物組合喜好度

	c. 每件衣物的喜好度設定 3. 整體修改完畢後更新資料庫設定
替代的使用案例	
使用的使用案例	
延伸的使用案例	
特殊要求	
案例詞彙公式	
備註	衣物組合喜好度及每件衣物的喜好度皆會影響推薦演算法內部的參數設定，如當使用者將這些喜好度的設定調高，推送的機率相對也會較高。

Use Case ID	UC-DA-02
Use Case Name	使用者設定(新增/設定)
目標	不同使用者可以設定專屬環境
需求	DBFR1- DBFR2- DBFR3- UIFR1- UIFR2-HWFR3
描述	根據登入的帳號不同，目前的使用者進行參數上的調整
前置條件	以登入使用者
後置條件	成功修改使用者設定
流程	<ol style="list-style-type: none"> 1. 使用者進入使用者介面進行設定 2. 選擇需要設定的項目 <ol style="list-style-type: none"> a. 鬧鐘 b. 居住地 c. 天氣喜好度 d. 衣物選擇模式 3. 將資料更新至指定資料庫
替代的使用案例	
使用的使用案例	
延伸的使用案例	
特殊要求	
案例詞彙公式	天氣喜好度：使用者對天氣的冷熱喜好程度 居住地：居住地 + 天氣擷取
備註	

伍、關聯

5.1 系統需求 V.S 系統需求 矩陣

*註:上方為項目，左邊為項目關聯之子需求

需求	DBFR1	DBFR2	DBFR3	DBFR4	IAFR1	AGFR1	PRFN1	UIFR1	UIFR2
DBFR1	--	V	V		V		V	V	
DBFR2		--						V	
DBFR3			--		V	V		V	
DBFR4				--					
IAFR1					--	V			
AGFR1						--		V	
PRFN1							--	V	
UIFR1								--	V
UIFR2								V	--
UIFR3								V	
UIFR4								V	
UIFR5								V	
UIFR6								V	
UIFR7								V	
UIFR8								V	
HWFR1									
HWFR2							V		
HWFR3									

需求	UIFR3	UIFR4	UIFR5	UIFR6	UIFR7	UIFR8	HWFR1	HWFR2	HWFR3
DBFR1	V	V					V	V	
DBFR2	V	V	V	V	V				
DBFR3	V			V	V				
DBFR4									
IAFR1			V	V					
AGFR1			V						
PRFN1		V					V		

5.3 系統需求 V.S 使用案例 矩陣

需求	UC-US-01	UC-US-02	UC-DA-01	UC-DA-02
DBFR1	V			V
DBFR2	V		V	V
DBFR3		V	V	V
DBFR4				
IAFR1		V		
AGFR1		V		
PRFN1	V			
UIFR1	V	V	V	V
UIFR2	V	V	V	V
UIFR3				V
UIFR4	V			
UIFR5		V		
UIFR6				
UIFR7			V	
UIFR8	V	V		
HWFR1	V	V		
HWFR2	V			
HWFR3	V	V	V	

