

# 國立臺北商業大學

資訊管理系

109' 資訊系統專案設計

## 系統手冊



組 別：第 109505 組

題 目：Food Scanner 超商食品偵查員

指導老師：張隆君 老師

組 長：10556047 周鈺祥

組 員：10556005 葉家誠    10556006 李宇平

10556014 林家弘    10556038 黃炅烱

中 華 民 國    109    年    11    月    25    日

# 目錄

<b>第 1 章 背景與動機</b> .....	1
1-1 簡介 .....	1
1-2 問題與機會 .....	2
1-3 相關系統探討 .....	4
<b>第 2 章 系統目標與預期成果</b> .....	5
2-1 系統目標 .....	5
2-2 預期成果 .....	5
<b>第 3 章 系統規格</b> .....	6
3-1 系統架構 .....	6
3-2 系統軟、硬體需求與技術平台 .....	7
3-3 開發標準與使用工具 .....	8
<b>第 4 章 專案時程與組織分工</b> .....	9
4-1 專案時程 .....	9
4-2 專案組織與分工 .....	10
<b>第 5 章 需求模型</b> .....	12
5-1 功能分解圖 (Functional decomposition diagram) .....	12
5-2 需求清單 .....	13
<b>第 6 章 程序模型</b> .....	14
6-1 資料流程圖(Data flow diagram) .....	14
6-2 程序規格書(Process specification) .....	16
<b>第 7 章 資料模型</b> .....	18
7-1 實體關聯圖(Entity relationship diagram) .....	18
7-2 資料字典(Data dictionary) .....	19
<b>第 8 章 資料庫設計</b> .....	20
8-1 資料庫關聯圖 .....	20
8-2 表格及其 Meta data .....	21
<b>第 9 章 程式規格</b> .....	22
9-1 軟體架構與程式清單 .....	22
9-2 程式規格描述 .....	24
<b>第 10 章 測試模型</b> .....	69
10-1 測試計畫 .....	69

10-2 測試個案與測試結果。 .....	69
第 11 章 操作手冊.....	71
第 12 章 使用手冊.....	75
第 13 章 感想.....	79
第 14 章 參考資料.....	84
附錄 審查評審意見之修正情形 .....	85

## 表目錄

表 1-2-1 TOWS 分析 .....	2
表 1-3-1 相關系統分析 .....	4
表 3-2-1 硬體需求 .....	7
表 3-2-2 網路需求 .....	7
表 3-3-1 系統開發工具 .....	8
表 4-2-1 分工表 .....	10
表 6-2-1 程序規格-產品查詢 .....	16
表 6-2-2 程序規格-即時語音 .....	16
表 6-2-3 程序規格-評價查詢 .....	16
表 6-2-4 程序規格-匯率查詢 .....	17
表 7-2-1 資料字典 .....	19
表 8-2-1 資料表名稱:product .....	21
表 9-1-1 程式清單 .....	22
表 9-2-1.....	24
表 9-2-2.....	25
表 9-2-3.....	27
表 9-2-4.....	28
表 9-2-5.....	30
表 9-2-6.....	31
表 9-2-7.....	33
表 9-2-8.....	35
表 9-2-9.....	36
表 9-2-10.....	37
表 9-2-11.....	38
表 9-2-12.....	40
表 9-2-13.....	46
表 9-2-14.....	48
表 9-2-15.....	66
表 9-2-16.....	67
表 10-2-1 測試功能「登入」 .....	69
表 10-2-2 測試功能「商品評價」 .....	69
表 10-2-3 測試功能「匯率查詢」 .....	69
表 10-2-4 測試功能「商品詳細」 .....	70
表 11-1 .....	71

表 11-2 .....	72
表 11-3 .....	73
表 11-4 .....	74
表 12-1 .....	75
表 12-2 .....	76
表 12-3 .....	78

## 圖目錄

圖 1-1-1.....	1
圖 3-1-1 系統架構圖 .....	6
圖 4-1-1 甘特圖 .....	9
圖 5-1-1 FDD.....	12
圖 6-1-1 外部關係圖 .....	14
圖 6-1-2 DFD 圖 1.....	15
圖 7-1-1 實體關聯圖 .....	18
圖 8-1-1 資料庫關聯圖 .....	20

# 第1章 背景與動機

## 1-1 簡介

在這個零售持續崛起的時代，儘管 2020 上半年受到疫情影響，仍然能從經濟部統計處提供的各項統計數據中發現：除了疫情嚴峻時的 3~5 月份受到嚴重衝擊，近年皆呈不斷成長的趨勢，且國內疫情稍緩的 6 月後，零售業更是迅速回到市場正常波動範圍，其中台灣零售業龍頭——超商更是逆勢成長，以台灣 7-Eleven(統一超商)為例，2020 年第一季毛利率較去年同期(2019 年第一季)提高了 0.2%。而綜觀整個台灣零售業，也能從未來流通研究所統計出的數據看到，2020 年第一季相較去年同期成長 20.1%、2020 年第二季相較去年同期成長 10%，超商的發展可謂蒸蒸日上。

近年國內連鎖式便利商店發展愈趨多元，公平交易委員會於 2020 年 8 月 7 日發布產業調查，截至去年底，五大業者營業門市合計達 11429 家店，年增 4.81%，創 14 年來最大增幅，超商密度位居全球第 2，國人平均每年去超商 130 次，每次消費金額約 82.6 元，呈現逐年遞增。

每個人在購買商品時都有很多選擇，就算只是買瓶礦泉水也有不同的牌子，不同的包裝，不同的價格、折扣方案可供選擇，如此便利性、多樣性常常讓人難以抉擇，也有人在購買後對於自己所做的決定感到後悔，再者對於國外的觀光客來說，想要看懂產品上的敘述實屬難事，即使看懂說明，也不容易在手機內輸入商品名稱做查詢。有鑒於此我們決定做一個 App，讓使用者只要掃描商品條碼(EAN-13)就能自動獲得商品的詳細資料及網路評價，並且我們在這項產品內搭載匯率轉換功能，對於國外的觀光客來說，大大省去購買商品前換算幣值的時間。

最後我們察覺到：很多在外國人在台灣買東西時，因為語言不通而看不懂產品包裝上的資訊，而不明白這是什麼樣的食物，例如：這是糖果？還是喉糖？又或者是口香糖？我們針對這方面的需求，協助消除語言隔閡，製作一款 App 讓使用者掃一掃商品條碼便迅速得到翻譯後的產品相關資訊，瞬間解決使用者煩惱，對於國外的觀光客來說，即使不會中文看不懂標示，也不用擔心買到不受歡迎或是自己不需要的商品。



▲圖 1-1- 1

## 1-2 問題與機會

有鑑於提供條碼掃描相關服務的軟體不少，我們審慎評估了內部條件、外部環境，並以 SWOT 分析為基礎，延伸制定出我們軟體的最佳發展策略。

首先是內部分析的優勢及劣勢，優勢方面我們提供與類似系統不同的**評價搜尋**，且其他兩者都因為要求某些特殊權限，在資安方面受到質疑，相較之下我們只需要使用者提供相機權限，以及檔案存取的權限就能執行。而劣勢的部分最為顯著的就是下載量及資料量，與平台上推出已久的軟體有比較大的差距。

外部分析中，市場機會為：使用者的**查詢評價**需求以及外籍人士的**語言問題**，至於威脅的部分，我們認為最主要是使用者對於資訊的敏銳度及蒐集能力大幅提升，也因此影響使用 App 協助查詢的可能。

▼表 1-2-1TOWS 分析

內部分析 外部分析	優勢 Strength	劣勢 Weakness
	1. 安全 2. 具評價搜尋功能	1. 下載量 2. 泛用性
機會 Opportunity	S O 型	W O 型
1. 民眾會想知道食物的評價 2. 外國人士需要一個能輕鬆了解產品資訊的方式	1. 專攻零售業食品 2. 提供口語化商品介紹	1. 除了四大超商自有商品，也收錄數百筆的常見產品資訊 2. 透過翻譯、匯率等附加功能提升實用性
威脅 Threat	S T 型	W T 型
1. 民眾資訊素養上升，資料取得容易，可能降低使用意願	1. 整合使用者普遍感興趣的功能及資訊，提升使用意願	1. 增加稀有性較高產品的資訊以提高民眾下載使用的意願 2. 拓展量販型超市的商品資訊，提高泛用性



綜上所述，我們需要做的就是開發一個能解決「評價搜尋」、「語音合成」、「加速產品資訊蒐集整合」需求的 App。

四個方向的策略：

SO—進攻策略分別為：「專攻零售業食品，與其他相似軟體做出區別」、「提供口語化的產品簡介，幫助使用者快速查詢該產品資訊，也讓外國人士更易於辨別產品之間的差異」

WO—弱點強化策略為：收錄超商各式常見產品，並提供其它實用副功能。

ST—差異化策略為：在資料蒐集的過程中可以發現：四大超商的產品介紹網頁，在搜尋引擎中的排名都不甚理想，影響整理資訊的速度，因此我們提供簡易的產品描述，節省使用者了解產品的時間。

WT—防守策略為：增加資料的深度及廣度、刺激下載量且提升泛用性。

### 1-3 相關系統探討

目前在 Google Play 中搜尋"條碼掃描"，大約會得到上百種搜尋結果，其中更是出現下載量破億的 App，但是進一步使用更精確地搜尋詞"超商條碼掃描"、"超商食品條碼"等，則是沒有任何搜尋結果，於是我們選擇了兩個排名靠前，且功能、性質有些相似的軟體做比較，比較結果如下圖：

▼表 1-3-1 相關系統分析

	目標客群	支援格式	主要資料	附加功能	下載量
超商食品偵查員	1.外國使用者 2. 四大超商常客	條碼 (EAN-13)	超商食品	1.商品內容即時翻譯 2.國際貨幣匯率轉換 3.一鍵語音	—
條碼掃描器	需要透過條碼至購物平台搜尋的使用者	條碼	各式物品	—	1 億+
免費 QR Code 掃描器	無特定客群	QR Code	各式物品	產生條碼 開啟手電筒	1000 萬+

現在市面上的條碼掃描，多數是屬於二維條碼的掃描，一維條碼的產品掃描系統非常稀少，而我們的「Food Scanner 超商食品偵查員」利用一維條碼做搜尋也附帶多國語言翻譯，在相關系統中我們的差異性相當的高。

## 第2章 系統目標與預期成果

### 2-1 系統目標

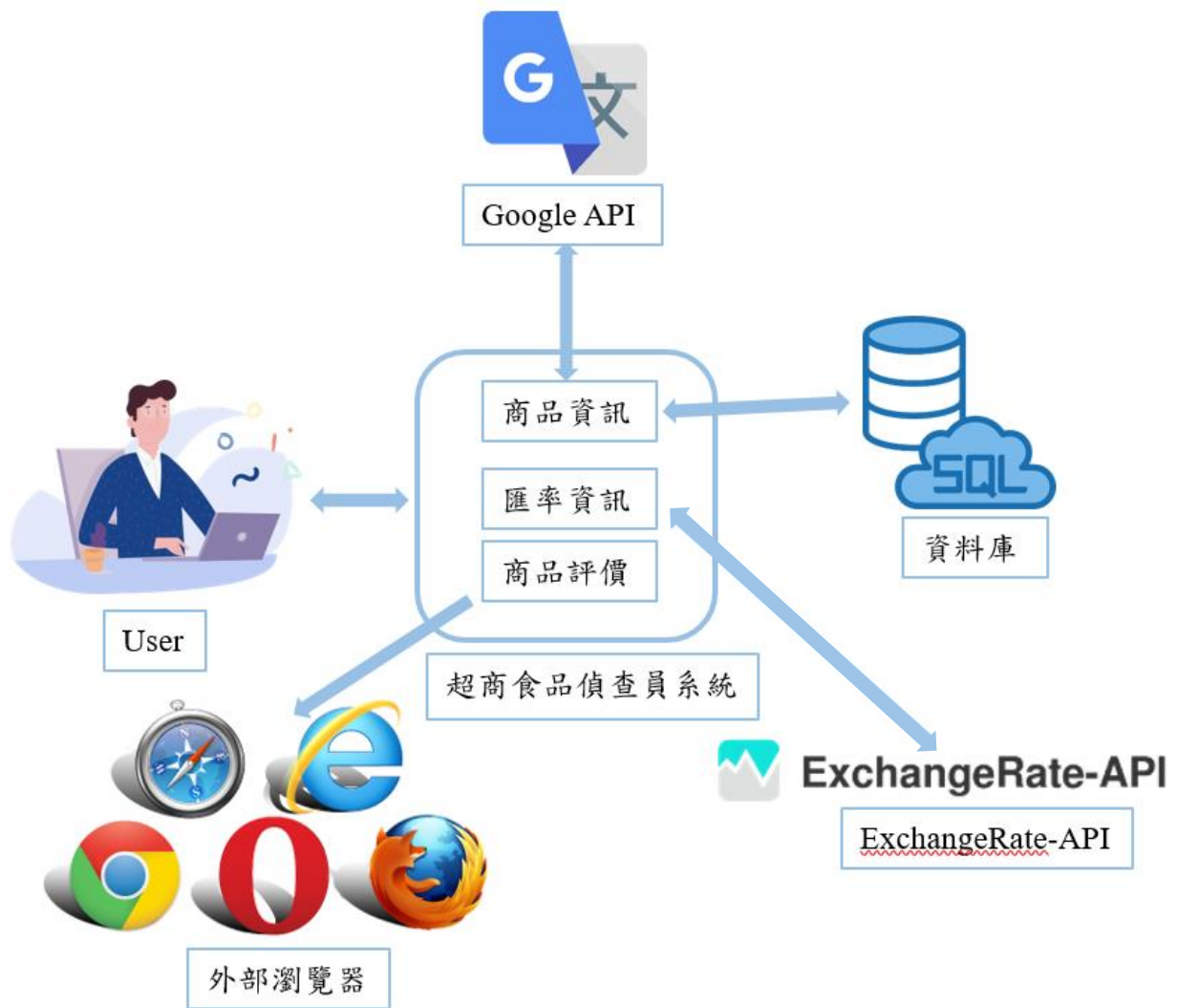
1. 提供產品各種資訊，利於使用者做決策。
2. 解決外國人士在選購商品時因語言不通的問題，EX:這是糖果?還是喉糖?又或者是曼陀珠?。
3. 解決人與人之間因害羞怕生、語言溝通問題。
4. 快速、方便的掃描，輕鬆得到產品資訊。

### 2-2 預期成果

1. 使用者滿意程度達到評價 4 顆星以上。
2. 增加更多語種，提高實用度。
3. 除了四大超商，提供服務範圍擴展至二線零售商(美聯社、頂好、TTL、IMEI...等)及常見批發零售賣場(家樂福、好事多、大潤發...等)
4. 增加多種條碼類型格式。

## 第3章 系統規格

### 3-1 系統架構



▲圖 3-1-1 系統架構圖

系統分為商品資訊、商品評價、匯率資訊三大功能

1. 商品資訊:  
提供多**語言翻譯**，讓使用者方便了解產品價格、成分等詳細資料，並利用語音將多國語言播放給使用者。
2. 商品評價:  
給予使用者跳轉至外部瀏覽器上，快速獲取對應商品的**相關評價**。
3. 匯率資訊:  
方便使用者做即時的**匯率轉換**。

## 3-2 系統軟、硬體需求與技術平台

▼表 3-2-1 硬體需求

硬體需求	
作業系統	Android 系統的智慧型裝置
版本	Android 9.0.0 以上(含)版本

▼表 3-2-2 網路需求

網路需求	
方式	3G/4G 、5G 行動網路、無線網路

### 3-3 開發標準與使用工具

▼表 3-3-1 系統開發工具

專題管理工具	GitHub GitHub Desktop
圖表工具	GanttProject 2.8 Draw.io
開發工具	Android Studio 4.0.1 Visual Studio Code
資料庫工具	Microsoft Office Excel 2016 Navicat 15 for PostgreSQL Heroku
文件製作工具	Microsoft Office Word 2016 Microsoft Office Excel 2016 Adobe XD 24.4.22.1 Adobe Illustator CS6
簡報製作工具	Microsoft Office PowerPoint 2016

# 第4章 專案時程與組織分工

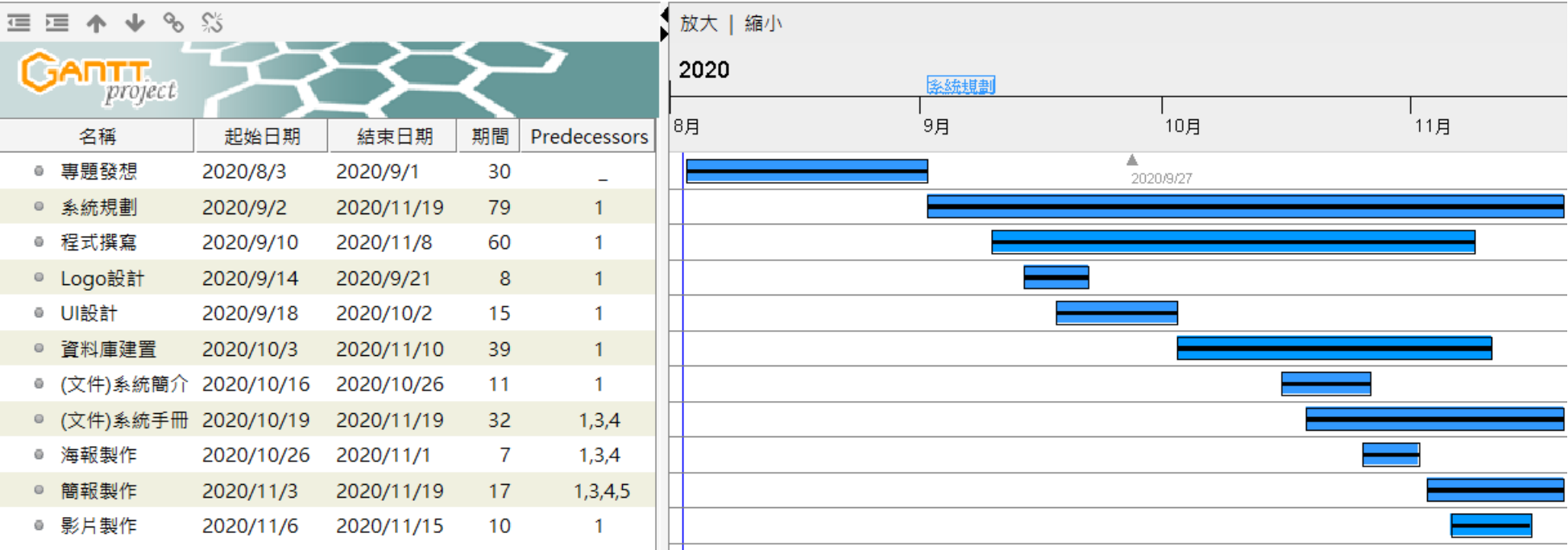
## 4-1 專案時程

圖例:

預計 

實際 

6



▲圖 4-1- 1 甘特圖

## 4-2 專案組織與分工

● 主要負責人 ○ 協助負責人

▼表 4-2-1 分工表

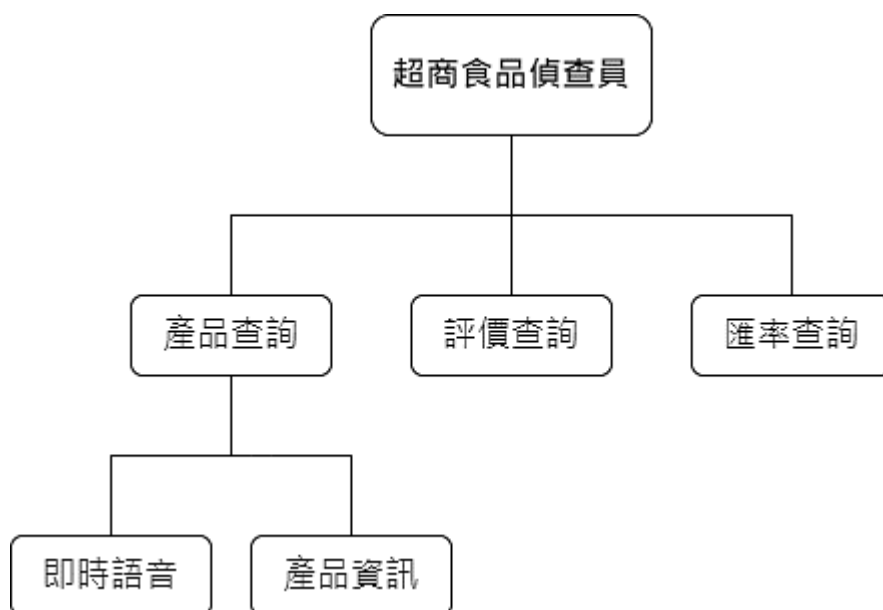
		10556047 周鈺祥	10556005 葉家誠	10556006 李宇平	10556014 林家弘	10556038 黃炅烱
企 劃	資料蒐集	○	○	●		
	訂定主題	●		○		
	系統設計		○			●
	可行性分析			●	○	
文 件	內文撰寫	○	○	●	●	○
	統整	○	●		●	
美 術 設 計	UI / UX 設計		○		●	
	海報設計			○	●	
	影片設計	●		○	○	
	影片剪輯		○	●		
	簡報製作	○	○		●	
	LOGO 設計			○	●	



資料庫	資料表設計	○	○			●
	建置資料庫	○	●			
	資料庫維護		●	○	○	
程式設計	一維條碼掃描	○	○			●
	文字翻譯	●		○		○
	匯率轉換	●				○
	即時語音	●				○
	資料庫連結	○	○			●
	評價搜尋	●				○
	前端介面建置	○	○		○	●
	後端網頁建置	○				●
	程式整合	●		○		●
	系統測試		●	○	○	
	程式維護	●	○			●

## 第5章 需求模型

### 5-1 功能分解圖（Functional decomposition diagram）



▲圖 5-1- 1 FDD

## 5-2 需求清單

### 功能性需求:

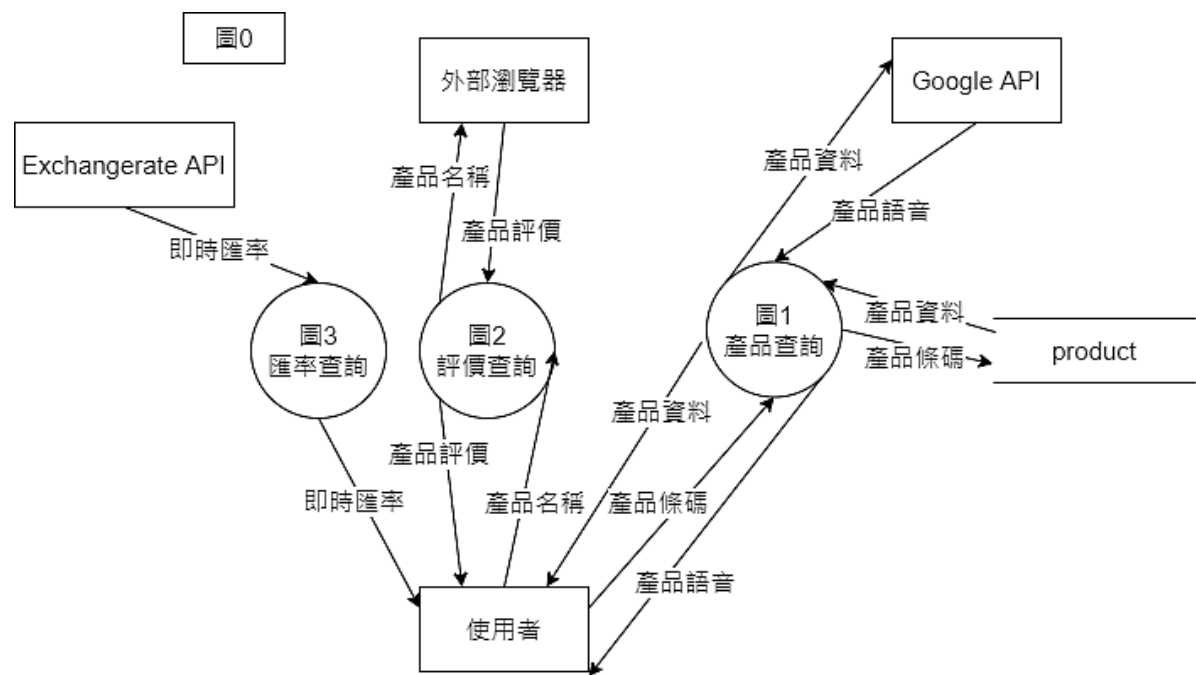
1. 使用者能利用條碼從系統查詢產品資訊
2. 系統能為使用者提供網路上的產品評價
3. 系統能為使用者提供即時國際匯率
4. 友善設計的使用者介面
5. 提供多種語音系統

### 非功能性需求:

1. 支援多平台設備
2. 支援多語言
3. 讀取速度快
4. 軟體占用資料空間小
5. 符合資訊安全考量
6. 高度發展品質
7. 高度軟體可移植性
8. 免註冊

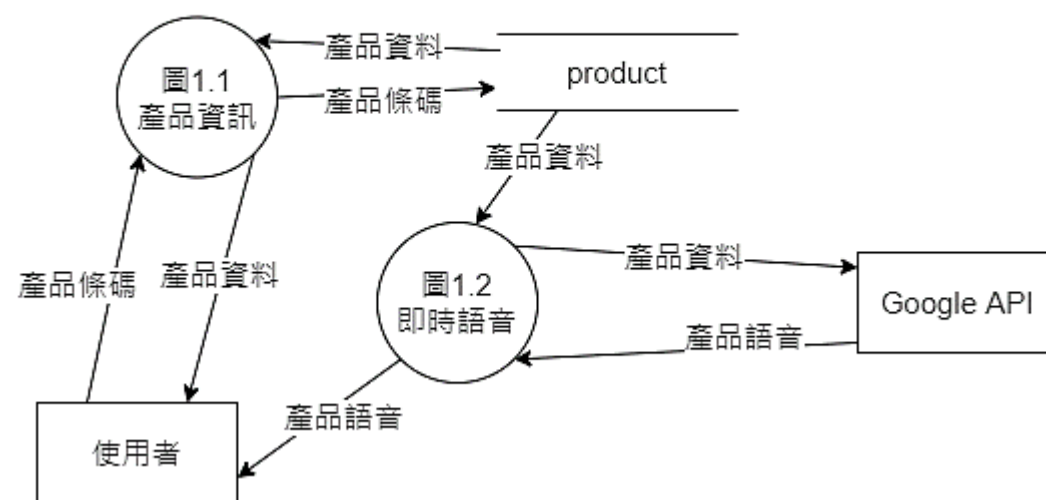
第6章 程序模型

6-1 資料流程圖(Data flow diagram)



▲圖 6-1- 1 外部關係圖

圖1



▲圖 6-1- 2 DFD 圖 1

## 6-2 程序規格書(Process specification)

▼表 6-2-1 程序規格-產品查詢

編號	1.1	動作名稱	產品查詢
動作說明	使用者開啟 App 後會先要求相機權限，以及檔案存取權限，使用者掃描超商產品條碼，利用條碼序號找出資料庫中產品資訊		
傳入值	產品條碼		
傳出值	產品資料		

▼表 6-2-2 程序規格-即時語音

編號	1.2	動作名稱	即時語音
動作說明	搜尋產品資料後可以利用多國語音播放產品資訊		
傳入值	產品資料		
傳出值	產品語音		

▼表 6-2-3 程序規格-評價查詢

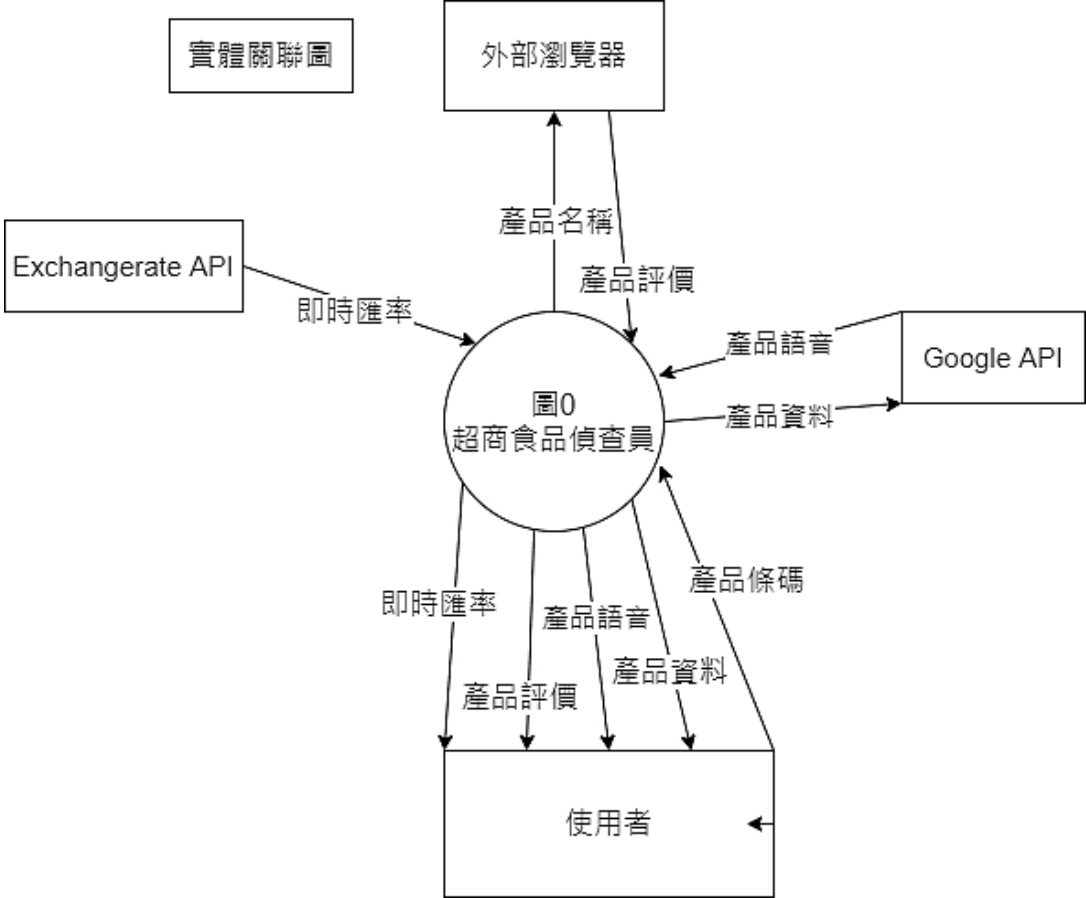
編號	2	動作名稱	評價查詢
動作說明	利用產品名稱自動在瀏覽器上搜尋產品評價給使用者。		
傳入值	產品名稱		
傳出值	產品評價		

▼表 6-2-4 程序規格-匯率查詢

編號	3	動作名稱	匯率查詢
動作說明	利用爬蟲工具從網路上抓取即時國際匯率供使用者計算金額。		
傳入值	即時匯率		
傳出值	即時匯率		

第7章 資料模型

7-1 實體關聯圖(Entity relationship diagram)



▲圖 7-1- 1 實體關聯圖



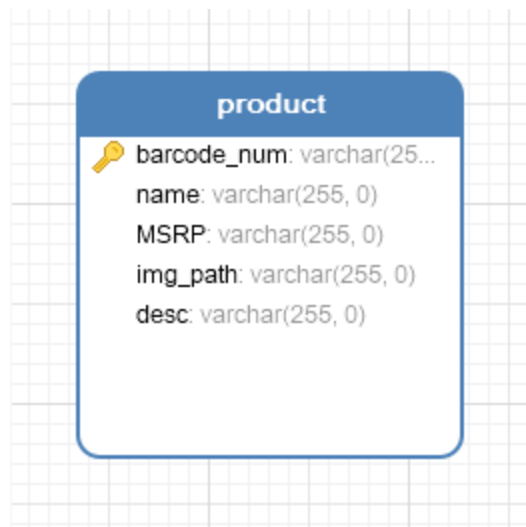
## 7-2 資料字典(Data dictionary)

▼表 7-2-1 資料字典

資料表編號	資料表名稱	資料表名稱(中文)	頁碼	圖表編號
T01	product	產品	21	表 8-2-1

## 第8章 資料庫設計

### 8-1 資料庫關聯圖



▲圖 8-1- 1 資料庫關聯圖

## 8-2 表格及其 Meta data

▼表 8-2-1 資料表名稱:product

	資料名稱	資料型態	資料長度
	barcode_num	varchar	255
	name	varchar	255
	MSRP	varchar	255
	img_path	varchar	255
	desc	varchar	255

### ● product 資料表欄位名稱說明

- barcode\_num：產品一維條碼用於資料比對。
- name：產品名稱讓程式利用產品名稱自動到網路搜尋評價。
- MSRP：產品價格供使用者比對產品價格，與匯率計算。
- img\_path：產品圖片供使用者比對產品樣貌。
- desc：產品詳細供使用者了解產品更細節資訊。

## 第9章 程式規格

### 9-1 軟體架構與程式清單

▼表 9-1-1 程式清單

資料夾名稱: /		
編號	檔案名稱	功能
1-1	package.json	匯入網頁相關套件
1-2	app.js	增加相關路徑名稱
資料夾名稱: routes		
編號	檔案名稱	功能
2-1	index.js	跳轉至首頁
2-2	product_query.js	條碼比對
2-3	product_query_form.js	跳轉至搜尋結果頁
資料夾名稱: routes/utility		
編號	檔案名稱	功能
3-1	asyncDB.js	連結至資料庫
3-2	product.js	條碼比對之 SQL 語法
資料夾名稱: views		
編號	檔案名稱	功能
4-1	error.ejs	錯誤訊息
4-2	index.ejs	產品搜尋頁面(首頁)
4-3	product_query.ejs	產品結果

資料夾名稱:app/java/com.example.a109505project		
編號	檔案名稱	功能
5-1	google_translationAPI	Google API
5-2	MainActivity	起始跳轉頁
5-3	Scanning	掃描頁
5-4	ShowActivity	呈現畫面
資料夾名稱:app/res		
編號	檔案名稱	功能
6-1	AndroidManifest.xml	App 基本設定
資料夾名稱:Gradle Scripts		
編號	檔案名稱	功能
7-1	build.gradle	App API 及套件添加

## 9-2 程式規格描述

▼表 9-2-1

編號	1-1	程式名稱	package.json
目的	匯入網頁相關套件		
部分程式碼			
<pre>{   "name": "getdata",   "version": "0.0.0",   "private": true,   "scripts": {     "start": "node ./bin/www"   },   "dependencies": {     "cookie-parser": "~1.4.4",     "debug": "~2.6.9",     "ejs": "~2.6.1",     "express": "~4.16.1",     "http-errors": "~1.6.3",     "moment": "^2.29.1",     "morgan": "~1.9.1",     "pg": "^8.4.1"   } }</pre>			

▼ 表 9-2-2

編號	1-2	程式名稱	app.js
目的	增加相關路徑名稱		
部分程式碼			
<pre>var createError = require('http-errors'); var express = require('express'); var path = require('path'); var cookieParser = require('cookie-parser'); var logger = require('morgan');  var indexRouter = require('./routes/index'); var usersRouter = require('./routes/users');  var app = express();  //----- // 增加引用模組 //----- var product_query_form = require('./routes/product_query_form'); var product_query = require('./routes/product_query'); //-----  // view engine setup app.set('views', path.join(__dirname, 'views')); app.set('view engine', 'ejs');  app.use(logger('dev')); app.use(express.json()); app.use(express.urlencoded({ extended: false })); app.use(cookieParser()); app.use(express.static(path.join(__dirname, 'public')));  app.use('/', indexRouter); app.use('/users', usersRouter);  //----- // 設定模組使用方式 //----- app.use('/product/query/form', product_query_form); app.use('/product/query', product_query); //-----</pre>			

```
// catch 404 and forward to error handler
app.use(function(req, res, next) {
  next(createError(404));
});

// error handler
app.use(function(err, req, res, next) {
  // set locals, only providing error in development
  res.locals.message = err.message;
  res.locals.error = req.app.get('env') === 'development' ? err : {};

  // render the error page
  res.status(err.status || 500);
  res.render('error');
});

module.exports = app;
```



▼表 9-2-3

編號	2-1	程式名稱	index.js
目的	跳轉至首頁		
部分程式碼			
<pre>var express = require('express'); var router = express.Router();  /* GET home page. */ router.get('/', function(req, res, next) {   res.render('index', { title: '?????' }); });  module.exports = router;</pre>			

▼ 表 9-2-4

編號	2-2	程式名稱	product_query.js
目的	條碼比對		
部分程式碼			
<pre>const { json } = require('express'); var express = require('express'); var router = express.Router();  //增加引用函式 var moment = require('moment'); const product = require('./utility/product'); var obj = {   barcode_num : "",   name : "",   MSRP : "",   img_path : 'https://upload.cc/i1/2020/11/13/yLM7rY.png',   desc:" };  //接收 GET 請求 router.get('/', function(req, res, next) {   var barcode_num = req.query.barcode_num; //取出參數    product.query(barcode_num).then(data =&gt; {     if (data==null){       res.render('error'); //導向錯誤頁面     }else if(data== -1){       product.query2(barcode_num).then(data =&gt; {         if (data==null){           res.render('error'); //導向錯誤頁面         }else if(data== -1){           res.send(JSON.stringify(obj)); //導向找不到頁面         }else{           res.send(JSON.stringify(data) ); //將資料傳給顯示頁面         }       })     }else{       res.send(JSON.stringify(data) ); //將資料傳給顯示頁面     }   }) });</pre>			

```
module.exports = router;
```

▼表 9-2-5

編號	2-3	程式名稱	product_query_form.js
目的	跳轉至搜尋結果頁		
部分程式碼			
<pre>var express = require('express'); var router = express.Router();  //接收 GET 請求 router.get('/', function(req, res, next) {   res.render('product_query_form'); });  module.exports = router;</pre>			

▼ 表 9-2-6

編號	3-1	程式名稱	asyncDB.js
目的	連結至資料庫		
部分程式碼			
<pre>'use strict';  //----- // 引用資料庫模組 //----- const {Client} = require('pg');  //----- // 自己的資料庫連結位址 //----- process.env['NODE_TLS_REJECT_UNAUTHORIZED'] = 0  var pgConn = 'postgres://mtbkslzevtlwep:d87c6427b300c6f901234fe4ac472ef4def14fefe7e05fc200 41768cf3729c03@ec2-3-214-46-194.compute- 1.amazonaws.com:5432/d1roeimk3gkona';  //產生可同步執行 sql 的函式 function query(sql, value=null) {   return new Promise((resolve, reject) =&gt; {     //產生資料庫連線物件     var client = new Client({       connectionString: pgConn,       ssl: true     })      //連結資料庫     client.connect();      //執行並回覆結果     client.query(sql, value, (err, results) =&gt; {       if (err){         reject(err);       }else{         resolve(results);       }     })   }) }</pre>			

```
        //關閉連線
        client.end();
    });
});
}

//匯出
module.exports = query;
```

▼ 表 9-2-7

編號	3-2	程式名稱	product.js
目的	條碼比對之 SQL 語法		
部分程式碼			
<pre>'use strict';  //引用操作資料庫的物件 const sql = require('./asyncDB');  //----- //執行資料庫動作的函式-取出單一商品 //----- var query = async function(barcode_num){   var result={};   await sql('SELECT * FROM product WHERE "barcode_num" = \$1', [barcode_num])     .then((data) =&gt; {       if(data.rows.length &gt; 0){         result = data.rows[0];       }else{         result = -1;       }     }, (error) =&gt; {       result = null;     });    return result; }  var query2 = async function(barcode_num){   var result={}   var s;   s = barcode_num.substr(0,8) + '%';   await sql('SELECT * FROM product WHERE "barcode_num" LIKE \$1' , [s])     .then((data) =&gt; {       if(data.rows.length &gt; 0){         result = data.rows[0];       }else{         result = -1;       }     }, (error) =&gt; {       result = null;     }); }</pre>			

```
    });  
  
    return result;  
}  
  
//匯出  
module.exports = {query,query2};
```



▼表 9-2-8

編號	4-1	程式名稱	error.ejs
目的	錯誤訊息		
部分程式碼			
<pre>&lt;h1&gt;&lt;%= message %&gt;&lt;/h1&gt; &lt;h2&gt;&lt;%= error.status %&gt;&lt;/h2&gt; &lt;pre&gt;&lt;%= error.stack %&gt;&lt;/pre&gt;</pre>			

▼表 9-2-9

編號	4-2	程式名稱	index.ejs
目的	產品搜尋頁面(首頁)		
部分程式碼			
<pre>&lt;!DOCTYPE html&gt; &lt;html&gt;   &lt;head&gt;     &lt;title&gt;&lt;%= title %&gt;&lt;/title&gt;     &lt;link rel='stylesheet' href='/stylesheets/style.css' /&gt;   &lt;/head&gt;   &lt;body&gt;     &lt;h2&gt;產品查詢&lt;/h2&gt;     &lt;form action = "/product/query" method = "get"&gt;       &lt;div class="form"&gt;         &lt;span class="name"&gt;產品編號: &lt;/span&gt;         &lt;span          class="value"&gt;&lt;input          type="text" name="barcode_num"&gt;&lt;/span&gt;         &lt;br/&gt;         &lt;span class="name"&gt;&lt;/span&gt;         &lt;span  class="value"&gt;&lt;input type="submit" value="查詢" &gt;&lt;/span&gt;       &lt;/div&gt;     &lt;/form&gt;   &lt;/body&gt; &lt;/html&gt;</pre>			

▼表 9-2-10

編號	4-3	程式名稱	product_query.ejs
目的	產品結果		
部分程式碼			
<pre>&lt;!DOCTYPE html&gt; &lt;html&gt;   &lt;head&gt;     &lt;link rel='stylesheet' href='/stylesheets/style.css' /&gt;   &lt;/head&gt;   &lt;body&gt;     &lt;table&gt;       &lt;tr&gt;         &lt;td&gt;&lt;%= item.barcode_num %&gt;&lt;/td&gt;       &lt;/tr&gt;       &lt;tr&gt;         &lt;td&gt;&lt;%= item.name %&gt;&lt;/td&gt;       &lt;/tr&gt;     &lt;/table&gt;   &lt;/body&gt; &lt;/html&gt;</pre>			

▼ 表 9-2-11

編號	5-1	程式名稱	google_translationAPI
目的	Google API		
部分程式碼			
<pre>package com.example.a109505project;  import com.loopj.android.http.AsyncHttpClient; import com.loopj.android.http.AsyncHttpResponseHandler;  import java.net.URLEncoder; //===== ===== /**連上 google translationAPI*/ //===== =====  public class google_translationAPI {     private static final String BASE_URL = "https://translation.googleapis.com/language/translate/v2?";     private static final String KEY = "AIzaSyC_yw7nAD4ipwys- nMSRz_PSYVIKIJFs9I";     private static AsyncHttpClient client = new AsyncHttpClient();      public static void post(String transText,String sourceLang, String destLang, AsyncHttpResponseHandler responseHandler) {         client.get(getAbsoluteUrl(transText, sourceLang, destLang), responseHandler);     }      private static String makeKeyChunk(String key) {         return "key=" + KEY;     }      private static String makeTransChunk(String transText) {         String encodedText = URLEncoder.encode(transText);         return "&amp;q=" + encodedText;     }      private static String langSource(String langSource) {         return "&amp;source=" + langSource;     }      private static String langDest(String langDest) {         return "&amp;target=" + langDest;     } }</pre>			

```

    }

    private static String getAbsoluteUrl(String transText, String sourceLang, String
destLang) {
        String apiUrl = BASE_URL + makeKeyChunk(KEY) +
makeTransChunk(transText) + langSource(sourceLang) + langDest(destLang);
        return apiUrl;
    }
}
//=====
=====

```

▼ 表 9-2-12

編號	5-2	程式名稱	起始跳轉頁
目的	起始跳轉頁		
部分程式碼			
<pre>package com.example.a109505project;  import androidx.appcompat.app.AppCompatActivity; import androidx.core.app.ActivityCompat;  import android.Manifest; import android.content.Intent; import android.content.pm.PackageManager; import android.os.Bundle; import android.os.CountDownTimer; import android.text.Layout; import android.util.DisplayMetrics; import android.util.Log; import android.view.KeyEvent; import android.view.View; import android.widget.AdapterView; import android.widget.ImageView; import android.widget.LinearLayout; import android.widget.TextView; import android.widget.Toast;  import com.loopj.android.http.JsonHttpResponseHandler;  import org.json.JSONArray; import org.json.JSONException; import org.json.JSONObject; import org.w3c.dom.Text; import java.util.Locale;  import cz.msebera.android.httpclient.Header;  public class MainActivity extends AppCompatActivity {  //===== =====  //宣告物件</pre>			

```

//=====
=====

private TextView goshow;
private LinearLayout touchlayout,logolayout;
private ImageView logo;

//=====
=====

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

//=====
=====

    //物件的對應 ID

//=====
=====

    goshow = (TextView)findViewById(R.id.goshow);
    touchlayout = (LinearLayout)findViewById(R.id.touchlayout);
    logolayout = (LinearLayout) findViewById(R.id.logolayout);
    logo = (ImageView)findViewById(R.id.logo);

//=====
=====

    //主要執行部分

//=====
=====

    goshow.setText("點擊螢幕以繼續");
    getPC();

//=====
=====

    //取得螢幕解析度及螢幕尺寸(寬跟高)

//=====
=====

    DisplayMetrics dm = new DisplayMetrics();
    getWindowManager().getDefaultDisplay().getMetrics(dm);
    int WidthPixels =dm.widthPixels;

```

```

int HeightPixels =dm.heightPixels;

//=====
=====

//設定物件的大小縮放

//=====
=====

logolayout.setMinimumWidth(900*WidthPixels/1080);
logolayout.setMinimumHeight(900*WidthPixels/1080);
logo.setMaxHeight(900*WidthPixels/1080);
logo.setMaxWidth(900*WidthPixels/1080);

//=====
=====

//偵測手機語言

//=====
=====

String language = Locale.getDefault().getLanguage();

//=====
=====

//點擊螢幕以繼續使用

//=====
=====

touchlayout.setOnClickListener(new View.OnClickListener() {
    String language = Locale.getDefault().getLanguage();
    //=====
    @Override
    public void onClick(View v) {
        boolean TF = ActivityCompat.checkSelfPermission(MainActivity.this,
Manifest.permission.CAMERA) ==
            PackageManager.PERMISSION_GRANTED;
        if(TF) {
            delay();
        }else{
            goshow.setText("請重啟 app,給予相機權限後,以繼續使用");
        }
    }
});

```



```

//=====
=====
    //執行翻譯的動作

//=====
=====

    translate();

//=====
=====

}

//=====
=====

    //檢查是否授權相機權限，如未授權將向使用者索取權限

//=====
=====

private void getPC(){
    if(ActivityCompat.checkSelfPermission(this, Manifest.permission.CAMERA)
        != PackageManager.PERMISSION_GRANTED){
        ActivityCompat.requestPermissions(this,new
String[]{Manifest.permission.CAMERA},1);
    }
}

//=====
=====

    //延遲的設定

//=====
=====

private void delay(){
    new CountDownTimer(2000, 1000) {
        public void onTick(long millisUntilFinished) {
            goshow.setText("尚餘" + millisUntilFinished/1000 + "秒後繼續");
            translate();
        }
        public void onFinish() {
            Intent intent = new Intent();
            intent.setClass(MainActivity.this, Scanning.class);
            startActivity(intent);
        }
    }.start();
}

```

```

    }
    }.start();
}

//=====
//此活動離開畫面時，關閉此活動

//=====

public void onStop() {
    super.onStop();
    finish();
}

//=====
//翻譯

//=====

public void translate(){
    String language = Locale.getDefault().getLanguage();
    String translationString = goshow.getText().toString();
    google_translationAPI.post(translationString, "zh", language, new
JsonHttpServletResponseHandler(){
    @Override
    public void onSuccess(int statusCode, Header[] headers, JSONObject
response) {
        try {
            JSONObject serverResp = new JSONObject(response.toString());
            JSONObject jsonObject = serverResp.getJSONObject("data");
            JSONArray transObject = jsonObject.getJSONArray("translations");
            JSONObject transObject2 = transObject.getJSONObject(0);
            goshow.setText(transObject2.getString("translatedText"));
        } catch (JSONException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
    });
}

```

```

//=====
=====
//使用手機本身的返回鍵，來離開應用程式

//=====
=====

private long exitTime = 0;
@Override
public boolean onKeyDown(int keyCode, KeyEvent event) {
    if(keyCode == KeyEvent.KEYCODE_BACK && event.getAction() ==
KeyEvent.ACTION_DOWN){
        if((System.currentTimeMillis()-exitTime) > 2000){
            Toast.makeText(getApplicationContext(),
MainActivity.this.getResources().getString(R.string.toast),
Toast.LENGTH_SHORT).show();
            exitTime = System.currentTimeMillis();
        } else {
            finish();
            System.exit(0);
        }
        return true;
    }
    return super.onKeyDown(keyCode, event);
}

//=====
=====
}

```

▼ 表 9-2-13

編號	5-3	程式名稱	Scanning
目的	掃描頁		
部分程式碼			
<pre>package com.example.a109505project;  import androidx.appcompat.app.AppCompatActivity; import androidx.core.app.ActivityCompat;  import android.Manifest; import android.content.Intent; import android.content.pm.PackageManager; import android.os.Bundle; import android.util.SparseArray; import android.view.KeyEvent; import android.view.SurfaceHolder; import android.view.SurfaceView; import android.widget.Toast;  import com.google.android.gms.vision.CameraSource; import com.google.android.gms.vision.Detector; import com.google.android.gms.vision.barcode.Barcode; import com.google.android.gms.vision.barcode.BarcodeDetector;  import java.io.IOException;  public class Scanning extends AppCompatActivity {      private SurfaceView surfaceView;     CameraSource cameraSource;     BarcodeDetector barcodeDetector;      @Override     protected void onCreate(Bundle savedInstanceState) {         super.onCreate(savedInstanceState);         setContentView(R.layout.activity_scanning);          //=====         surfaceView = (SurfaceView)findViewById(R.id.surfaceView);         //選擇條碼掃描的類型         barcodeDetector = new BarcodeDetector.Builder(this)             .setBarcodeFormats(Barcode.EAN_13).build();</pre>			

```

//設定相機大小
cameraSource=new CameraSource.Builder(this,barcodeDetector)
    .setRequestedPreviewSize(100,100).build();
//讓相機能夠對焦
cameraSource = new CameraSource.Builder(this,barcodeDetector)
    .setAutoFocusEnabled(setAutoFocusEnabled(true)).build();

//=====
=====

surfaceView.getHolder().addCallback(new SurfaceHolder.Callback(){
    @Override
    public void surfaceCreated(SurfaceHolder surfaceHolder) {
        if(ActivityCompat.checkSelfPermission(getApplicationContext(),
Manifest.permission.CAMERA)
            != PackageManager.PERMISSION_GRANTED)
            return;
        try{
            cameraSource.start(surfaceHolder);
        }catch (IOException e){
            e.printStackTrace();
        }
    }
    @Override
    public void surfaceChanged(SurfaceHolder holder, int format, int width, int
height) {

    }
    @Override
    public void surfaceDestroyed(SurfaceHolder holder) {
        cameraSource.stop();
    }
});
barcodeDetector.setProcessor(new Detector.Processor<Barcode>(){
    @Override
    public void release() {

    }
    @Override
    public void receiveDetections(Detector.Detections<Barcode> detections) {
        final SparseArray<Barcode> barCodes=detections.getDetectedItems();
        if(barCodes.size()!=0){
            //將掃到的條碼號轉換成文字
            String number = barCodes.valueAt(0).displayValue;

```

```

        barcodeDetector.release();//關閉相機
        //將掃描到的條碼號打包後傳送至 ShowActivity
        Intent intent = new Intent();
        intent.setClass(Scanning.this , ShowActivity.class);
        Bundle bundle =new Bundle();
        bundle.putString("number" , number);
        intent.putExtras(bundle);
        startActivity(intent);

//=====
=====

    }
}
});
}

private boolean setAutoFocusEnabled(boolean b) {return true;}

public void onStop() {
    super.onStop();
    finish();
}

private long exitTime = 0;
@Override
public boolean onKeyDown(int keyCode, KeyEvent event) {
    if(keyCode == KeyEvent.KEYCODE_BACK && event.getAction() ==
KeyEvent.ACTION_DOWN){
        if((System.currentTimeMillis()-exitTime) > 2000){
            Toast.makeText(getApplicationContext(),
Scanning.this.getResources().getString(R.string.toast),
Toast.LENGTH_SHORT).show();
            exitTime = System.currentTimeMillis();
        } else {
            finish();
            System.exit(0);
        }
        return true;
    }
    return super.onKeyDown(keyCode, event);
}
}
}

```

▼ 表 9-2-14

編號	5-4	程式名稱	ShowActivity
目的	呈現畫面		
部分程式碼			
<pre>package com.example.a109505project;  import androidx.appcompat.app.AppCompatActivity;  import android.annotation.SuppressLint; import android.app.AlertDialog; import android.app.SearchManager; import android.content.DialogInterface; import android.content.Intent; import android.graphics.Bitmap; import android.graphics.BitmapFactory; import android.graphics.Color; import android.graphics.Typeface; import android.os.AsyncTask; import android.os.Bundle; import android.os.Looper; import android.util.DisplayMetrics; import android.util.Log; import android.view.KeyEvent; import android.view.View; import android.widget.Button; import android.widget.EditText; import android.widget.ImageButton; import android.widget.ImageView; import android.widget.LinearLayout; import android.widget.Spinner; import android.widget.TextView; import android.widget.Toast;  import com.google.android.gms.vision.text.Line; import com.loopj.android.http.JsonHttpResponseHandler;  import org.jetbrains.annotations.NotNull; import org.json.JSONArray; import org.json.JSONException; import org.json.JSONObject;  import java.io.IOException; import java.io.InputStream; import java.util.List;</pre>			

```

import java.util.Locale;

import cz.msebera.android.httpclient.Header;
import okhttp3.Call;
import okhttp3.Callback;
import okhttp3.OkHttpClient;
import okhttp3.Request;
import okhttp3.Response;
import okhttp3.logging.HttpLoggingInterceptor;

import android.content.res.Configuration;
import android.content.res.Resources;
import android.speech.tts.TextToSpeech;

public class ShowActivity extends AppCompatActivity {

//=====
//=====

    //物件宣告

//=====
//=====

    private TextView
tv_barcodenum,tv_name,tv_convert,comment,tv_show,tv_code_num,tv_pro,tv_pric
e;
    private ImageButton rescan,exchange;
    private LinearLayout
imglayout,showlayout,txtlayout,moneylayout,namelayou,numlayout,morelayout,
    rescanlayout,exchangelayout,commentlayout;
    private ImageView imgshow;
    private EditText et_twd;
    private Button bt_convert,more;
    private ImageView img_sound=null;
    private Spinner sp_currency;
    private static final int REQ_TTS_STATUS_CHECK = 0;
    private static final String TAG = "TTS Demo";
    private TextToSpeech mTts;
    String txt = "",desc,product;
    List<String> keysList;
    int out = 0;

//=====
//=====

```



@Override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_show);
```

```
//=====
```

```
//物件 ID 設定
```

```
//=====
```

```
comment = (TextView) findViewById(R.id.comment);  
rescan = (ImageButton) findViewById(R.id.rescan);  
tv_barcodenum = (TextView) findViewById(R.id.tv_barcodenum);  
imglayout = (LinearLayout) findViewById(R.id.imglayout);  
imgshow = (ImageView) findViewById(R.id.imgshow);  
exchange = (ImageButton) findViewById(R.id.exchange);  
tv_name = (TextView) findViewById(R.id.tv_name);  
tv_convert = (TextView) findViewById(R.id.tv_convert);  
more = (Button) findViewById(R.id.more);  
showlayout = (LinearLayout) findViewById(R.id.showlayout);  
textlayout = (LinearLayout) findViewById(R.id.textlayout);  
numlayout = (LinearLayout) findViewById(R.id.numlayout);  
namelayout = (LinearLayout) findViewById(R.id.namelayout);  
moneylayout = (LinearLayout) findViewById(R.id.moneylayout);  
tv_code_num = (TextView) findViewById(R.id.tv_code_num);  
morelayout = (LinearLayout) findViewById(R.id.morelayout);  
rescanlayout = (LinearLayout) findViewById(R.id.rescanlayout);  
exchangelayout = (LinearLayout) findViewById(R.id.exchangelayout);  
commentlayout = (LinearLayout) findViewById(R.id.commentlayout);  
tv_pro = (TextView) findViewById(R.id.tv_pro);  
tv_price = (TextView) findViewById(R.id.tv_price);
```

```
//=====
```

```
//主程式
```

```
//=====
```

```
//=====
```

```
more.setText("more ...");
```

```

//=====
=====

//抓取掃描對應的資料與圖片

//=====
=====

final Bundle bundle = this.getIntent().getExtras();
final String number = bundle.getString("number");
final      String      getdata      =      "https://project109505-
1.herokuapp.com/product/query?barcode_num=" + number;
final String uri = getdata;

//=====
=====

//取得螢幕解析度及螢幕尺寸(寬跟高)

//=====
=====

DisplayMetrics dm = new DisplayMetrics();
getWindowManager().getDefaultDisplay().getMetrics(dm);
int WidthPixels =dm.widthPixels;
int HeightPixels =dm.heightPixels;

//=====
=====

//設定物件的大小縮放

//=====
=====

tv_price.setShadowLayer(10F,11F,5F, Color.BLACK);
tv_pro.setShadowLayer(10F,11F,5F, Color.BLACK);
tv_code_num.setShadowLayer(10f,11f,5f,Color.BLACK);
tv_name.setShadowLayer(10f,11f,5f,Color.BLACK);
tv_barcodenum.setShadowLayer(10f,11f,5f,Color.BLACK);
tv_convert.setShadowLayer(10f,11f,5f,Color.BLACK);
tv_code_num.setWidth(100*WidthPixels/1080);
tv_code_num.setHeight(60*HeightPixels/2214);
tv_barcodenum.setHeight(60*HeightPixels/2214);
tv_barcodenum.setWidth(190*WidthPixels/1080);
tv_pro.setWidth(100*WidthPixels/1080);
tv_pro.setHeight(60*HeightPixels/2214);
tv_price.setWidth(100*WidthPixels/1080);

```

```

tv_price.setHeight(60*HeightPixels/2214);
tv_name.setHeight(60*HeightPixels/2214);
tv_name.setWidth(190*WidthPixels/1080);
tv_convert.setHeight(60*HeightPixels/2214);
tv_convert.setWidth(190*WidthPixels/1080);
showlayout.setMinimumWidth(1200*WidthPixels/1080);
imglayout.setMinimumHeight(800*HeightPixels/2214);
imglayout.setMinimumWidth(740*WidthPixels/1080);
imgshow.setMaxHeight(800*HeightPixels/2214);
imgshow.setMaxWidth(600*WidthPixels/1080);
rescan.setMaxWidth(100*WidthPixels/1080);
rescan.setMaxHeight(60*HeightPixels/2214);
comment.setMaxHeight(60*HeightPixels/2214);
comment.setMaxWidth(100*WidthPixels/1080);
textlayout.setPadding(0,50*HeightPixels/2214,0,0);
numlayout.setPadding(0,20*HeightPixels/2214,0,0);
namelayout.setPadding(0,20*HeightPixels/2214,0,0);
moneylayout.setPadding(0,10*HeightPixels/2214,0,0);
morelayout.setPadding(0,100,0,0);
more.setPadding(0,0,30,0);
exchangelayout.setPadding(0,0,0,30*HeightPixels/2214);
rescanlayout.setPadding(0,0,100*WidthPixels/1080,30*HeightPixels/2214);
commentlayout.setPadding(100*WidthPixels/1080,0,0,30*HeightPixels/2214);

```

```

//=====
=====

```

```

//重新扫描

```

```

//=====
=====

```

```

rescan.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        out = 1;
        Intent intent = new Intent();
        intent.setClass(ShowActivity.this,Scanning.class);
        startActivity(intent);
    }
});

```

```

//=====
=====

```

```

//=====
=====

//=====
=====

//利用 OKHttp 去網頁抓取對應的資料結果(以 JSON 型態呈顯)

//=====
=====

OkHttpClient client = new OkHttpClient().newBuilder()
    .addInterceptor(new
HttpLoggingInterceptor().setLevel(HttpLoggingInterceptor.Level.BASIC))
    .build();
/**設置傳送需求*/
Request request = new Request.Builder().url(uri).build();
/**設置回傳*/
Call call = client.newCall(request);
call.enqueue(new Callback() {
    @Override
    public void onFailure(@NotNull Call call, @NotNull IOException e) {
        tv_barcodenum.setText(number);

tv_name.setText(ShowActivity.this.getResources().getString(R.string.no_informatio
n));
    }
    @Override
    public void onResponse(@NotNull Call call, @NotNull Response response)
throws IOException {
        final String responseData = response.body().string();
        /**取得回傳*/
        ShowActivity.this.runOnUiThread(new Runnable() {
            @SuppressWarnings("StaticFieldLeak")
            @Override
            public void run() {

//=====
=====

                String [] i = null;
                String data = responseData;
                i = data.split("\\");
                data =i[15];
                String [] url = null;

```

```

        url = data.split("\\\\");
        data = url[0];
        txt = i[19];
        product=i[7];
        if(i[3].equals("")) {
            tv_barcodenum.setText(number);

tv_name.setText(ShowActivity.this.getResources().getString(R.string.no_informatio
n));

tv_convert.setText(ShowActivity.this.getResources().getString(R.string.no_informat
ion));

        comment.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

Toast.makeText(ShowActivity.this,ShowActivity.this.getResources().getString(R.stri
ng.no_information) , Toast.LENGTH_SHORT).show();
            }
        });
        more.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

Toast.makeText(ShowActivity.this,ShowActivity.this.getResources().getString(R.stri
ng.no_information) , Toast.LENGTH_SHORT).show();
            }
        });
    }else{
        String s =tv_code_num.getText().toString();
        tv_barcodenum.setText(s + i[3]);
        tv_code_num.setVisibility(View.GONE);
        s =tv_pro.getText().toString();
        tv_pro.setVisibility(View.GONE);
        tv_name.setText(s + i[7]);
        s =tv_price.getText().toString();
        tv_price.setVisibility(View.GONE);
        tv_convert.setText(s + i[11]);

//=====
=====

        /**搜尋產品評價*/

```

```

//=====
=====
        comment.setOnClickListener(new View.OnClickListener() {
            String                                pro_comment=product+"
"+comment.getText().toString();
            public void onClick(View v) {
                Intent intent = new Intent(Intent.ACTION_WEB_SEARCH);
                intent.putExtra(SearchManager.QUERY,pro_comment);
                startActivity(intent);
            }
        });

//=====
=====
        translate();
        more.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                /**設定 AlertDialog.Builder*/

//=====
=====
                AlertDialog.Builder        alertDialog        =        new
AlertDialog.Builder(ShowActivity.this);

        alertDialog.setTitle(ShowActivity.this.getResources().getString(R.string.more_title))
        ;
                alertDialog.setMessage(desc);
                View view = getLayoutInflater().inflate(R.layout.more_layout,
null);
                alertDialog.setView(view);
                alertDialog.setIcon(R.drawable.more);
                alertDialog.setCancelable(false);

//=====
=====
                /**設定物件*/

//=====
=====
                img_sound=                                (ImageView)
view.findViewById(R.id.img_sound);

```

```

        img_sound.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                /**朗讀輸入框裡的內容*/

mTts.speak(desc,TextToSpeech.QUEUE_FLUSH,null,null);

Toast.makeText(ShowActivity.this,ShowActivity.this.getResources().getString(R.string.playing) , Toast.LENGTH_SHORT).show();
            }
        });
        /**檢查 TTS 資料是否已經安裝並且可用*/
        Intent checkIntent = new Intent();

checkIntent.setAction(TextToSpeech.Engine.ACTION_CHECK_TTS_DATA);
        startActivityForResult(checkIntent,
REQ_TTS_STATUS_CHECK);

//=====
=====

//=====
=====

AlertDialog.setPositiveButton(ShowActivity.this.getResources().getString(R.string.close), new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                onPause();
            }
        });
        alertDialog.show();

    }
});
}

        new DownloadImageTask((ImageView)
findViewById(R.id.imgshow)).execute(data);

    }
});

```

```

    }
    });

//=====
=====
    /**轉換匯率*/

//=====
=====
    exchange.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {

            /**設定 AlertDialog.Builder*/

//=====
=====
            AlertDialog.Builder builder = new AlertDialog.Builder(ShowActivity.this);
builder.setTitle(ShowActivity.this.getResources().getString(R.string.currency_title));
            builder.setIcon(R.drawable.exchange_money);
            View view = getLayoutInflater().inflate(R.layout.exchange_money, null);
            builder.setView(view);
            builder.setCancelable(false);

//=====
=====
            /**設定物件*/

//=====
=====
            bt_convert = (Button) view.findViewById(R.id.bt_convert);
            et_twd = (EditText) view.findViewById(R.id.et_twd);
            sp_currency = (Spinner) view.findViewById(R.id.sp_currency);
            tv_show = (TextView) view.findViewById(R.id.tv_show);

//=====
=====
            /**轉換匯率*/

//=====
=====

```



```

String get_lan=Locale.getDefault().getLanguage();
int list;
switch(get_lan) {
    case "zh":
        list=2;
        break;
    case "in":
        list=6;
        break;
    case "ja":
        list=7;
        break;
    case "ko":
        list=8;
        break;
    case "th":
        list=11;
        break;
    case "en":
        list=12;
        break;
    default:
        list=0;
}
sp_currency.setSelection(list, true);
bt_convert.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(!et_twd.getText().toString().isEmpty())
        {
            String focus = sp_currency.getSelectedItem().toString();
            double Vlaue = Double.valueOf(et_twd.getText().toString());

```

```

Toast.makeText(ShowActivity.this,ShowActivity.this.getResources().getString(R.string.wait) , Toast.LENGTH_SHORT).show();

```

```

        try {
            convertCurrency(focus, Vlaue);
        } catch (IOException e) {
            e.printStackTrace();
            Toast.makeText(ShowActivity.this, e.getMessage(),
Toast.LENGTH_SHORT).show();
        }
    }
}

```

```

        else
        {

Toast.makeText(ShowActivity.this,ShowActivity.this.getResources().getString(R.string.enter_value) , Toast.LENGTH_SHORT).show();

        }
    }
});

//=====
=====

builder.setPositiveButton(ShowActivity.this.getResources().getString(R.string.close)
, new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int which) {

    }
});

    builder.show();
}
});

}

//=====
=====

    /**匯率網址*/

//=====
=====

    public void convertCurrency(final String toCurr, final double euroVlaue) throws
IOException {

        String url = "https://v6.exchangerate-
api.com/v6/e4c277e6fda1bc30d5fe3f7d/latest/TWD";
        OkHttpClient client = new OkHttpClient();
        Request request = new Request.Builder()
            .url(url)
            .header("Content-Type", "application/json")

```

```

        .build();

        client.newCall(request).enqueue(new Callback() {
            @Override
            public void onResponse(@NotNull Call call, @NotNull Response response)
throws IOException {
                final String mMessage = response.body().string();
                ShowActivity.this.runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
                        try {
                            JSONObject obj = new JSONObject(mMessage);
                            JSONObject b = obj.getJSONObject("conversion_rates");
                            String val = b.getString(toCurr);
                            double output = euroVlaue*Double.valueOf(val);
                            String money = String .format("%.2f",output);
                            tv_show.setText(money);
                        } catch (JSONException e) {
                            e.printStackTrace();
                        }
                    }
                });
            }
            @Override
            public void onFailure(@NotNull Call call, @NotNull IOException e) {
                String mMessage = e.getMessage().toString();
                Log.w("failure Response", mMessage);
                Looper.prepare();
                Toast.makeText(ShowActivity.this, mMessage,
Toast.LENGTH_SHORT).show();
                Looper.loop();
            }
        });

//=====
=====

    }

    public void onStop() {
        super.onStop();
        if(out == 1) {
            out = 0;
            finish();
        }
    }

```

```

}

private long exitTime = 0;
@Override
public boolean onKeyDown(int keyCode, KeyEvent event) {
    if(keyCode == KeyEvent.KEYCODE_BACK && event.getAction() ==
KeyEvent.ACTION_DOWN){
        if((System.currentTimeMillis()-exitTime) > 2000){
            Toast.makeText(getApplicationContext(),
ShowActivity.this.getResources().getString(R.string.toast),
Toast.LENGTH_SHORT).show();
            exitTime = System.currentTimeMillis();
        } else {
            finish();
            System.exit(0);
        }
        return true;
    }
    return super.onKeyDown(keyCode, event);
}

private class DownloadImageTask extends AsyncTask<String, Void, Bitmap> {
    ImageView bmImage;

    public DownloadImageTask(ImageView bmImage) {
        this.bmImage = bmImage;
    }

    protected Bitmap doInBackground(String... urls) {
        String urldisplay = urls[0];
        Bitmap mIcon11 = null;
        try {
            InputStream in = new java.net.URL(urldisplay).openStream();
            mIcon11 = BitmapFactory.decodeStream(in);
        } catch (Exception e) {
            Log.e("Error", e.getMessage());
            e.printStackTrace();
        }
        return mIcon11;
    }

    protected void onPostExecute(Bitmap result) {
        bmImage.setImageBitmap(result);
    }
}

```

```

    }

//=====
//=====

    //翻譯

//=====
//=====

    public void translate(){
        String language = Locale.getDefault().getLanguage();
        if(language=="zh"){language="zh-Hant";}
        String translationString = txt;
        google_translationAPI.post(translationString,      "zh",      language,      new
JsonHttpResponderHandler()){
            @Override
            public void onSuccess(int statusCode, Header[] headers, JSONObject
response) {
                try {
                    JSONObject serverResp = new JSONObject(response.toString());
                    JSONObject jsonObject = serverResp.getJSONObject("data");
                    JSONArray transObject = jsonObject.getJSONArray("translations");
                    JSONObject transObject2 = transObject.getJSONObject(0);
                    desc=transObject2.getString("translatedText");
                } catch (JSONException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
            }
        });
    }

//=====
//=====

    //實現 TTS 初始化介面

//=====
//=====

    public void onInit(int status) {
        // TODO Auto-generated method stub
        /**TTS Engine 初始化完成*/
        if(status == TextToSpeech.SUCCESS)

```

```

    {
        int result = mTts.setLanguage(Locale.US);
        /*設定發音語言*/
        if(result == TextToSpeech.LANG_MISSING_DATA || result ==
TextToSpeech.LANG_NOT_SUPPORTED)
            /*判斷語言是否可用*/
            {
                Log.v(TAG, "Language is not available");
                img_sound.setEnabled(false);
            }
        else
        {
            mTts.speak("This is an example of speech synthesis.",
TextToSpeech.QUEUE_ADD, null);
            img_sound.setEnabled(true);
        }
    }

}

//=====
//=====

//回傳語音語言

//=====
//=====

protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    if(requestCode == REQ_TTS_STATUS_CHECK)
    {
        if (resultCode==TextToSpeech.Engine.CHECK_VOICE_DATA_PASS) {
            /*這個返回結果表明 TTS Engine 可以用*/
            {
                /* Set up the Text To Speech engine.*/
                TextToSpeech.OnInitListener listener =
                    new TextToSpeech.OnInitListener() {
                        @Override
                        public void onInit(final int status) {
                            if (status == TextToSpeech.SUCCESS) {
                                Log.d("OnInitListener", "hello");
                                //取得手機系統出廠語系和國家
                                Resources res = ShowActivity.this.getResources();
                                Configuration conf = res.getConfiguration();
                                String language = conf.locale.getLanguage();

```

```

        Log.e("conf",language);

        mTts.setLanguage(new Locale(language,""));
    } else {
        Log.d("OnInitListener", "Error starting the text to speech
engine.");
    }
}
};

mTts = new TextToSpeech(this(getApplicationContext(), listener);
Log.v(TAG, "TTS Engine is installed!");

    }
}
}
super.onActivityResult( requestCode, resultCode, data);
}
@Override
protected void onPause() {
    /* TODO Auto-generated method stub */
    super.onPause();
    if(mTts != null)
        /*activity 暫停時也停止 TTS*/
        {
            mTts.stop();
        }
}

@Override
protected void onDestroy() {
    /*TODO Auto-generated method stub */
    super.onDestroy();
    /*釋放 TTS 的資源*/
}

//=====
=====
}

```

▼ 表 9-2-15

編號	6-1	程式名稱	AndroidManifest.xml
目的	App 基本設定		
部分程式碼			
<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;manifest xmlns:android="http://schemas.android.com/apk/res/android"     package="com.example.a109505project"&gt;      &lt;uses-permission android:name="android.permission.CAMERA" /&gt;     &lt;uses-permission android:name="android.permission.INTERNET" /&gt;     &lt;uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/&gt;     &lt;application         android:label="Food Scanner"         android:allowBackup="true"         android:icon="@drawable/logo2"         android:roundIcon="@drawable/rlogo"         android:supportsRtl="true"         android:theme="@style/AppTheme"&gt;         &lt;activity android:name=".ShowActivity"&gt;&lt;/activity&gt;         &lt;activity android:name=".Scanning" /&gt;         &lt;activity android:name=".MainActivity"&gt;             &lt;intent-filter&gt;                 &lt;action android:name="android.intent.action.MAIN" /&gt;                 &lt;category android:name="android.intent.category.LAUNCHER" /&gt;             &lt;/intent-filter&gt;         &lt;/activity&gt;     &lt;/application&gt; &lt;/manifest&gt;</pre>			



▼ 表 9-2-16

編號	7-1	程式名稱	build.gradle
目的	App API 及套件添加		
部分程式碼			
<pre>apply plugin: 'com.android.application'  android {     compileSdkVersion 29     buildToolsVersion '29.0.3'      defaultConfig {         applicationId "com.example.a109505project"         minSdkVersion 21         targetSdkVersion 29         versionCode 1         versionName "1.0"         testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"     }      buildTypes {         release {             minifyEnabled false             proguardFiles    getDefaultProguardFile('proguard-android- optimize.txt'), 'proguard-rules.pro'         }     }     sourceSets {         main {             assets {                 srcDirs 'src/main/assets', 'src/main/res/assets/'             }         }     } }  dependencies {     implementation fileTree(dir: 'libs', include: ['*.jar'])      implementation 'androidx.appcompat:appcompat:1.2.0'     implementation 'androidx.constraintlayout:constraintlayout:2.0.2'</pre>			

```
testImplementation 'junit:junit:4.12'
androidTestImplementation 'androidx.test.ext:junit:1.1.2'
androidTestImplementation 'androidx.test.espresso:espresso-
core:3.3.0'

implementation 'com.google.android.gms:play-services-
vision:16.2.0'
implementation 'com.squareup.okhttp3:okhttp:4.7.2'
implementation 'com.squareup.okhttp3:okhttp-
urlconnection:4.7.2'
implementation 'com.squareup.okhttp3:logging-interceptor:4.7.2'

implementation 'com.loopj.android:android-async-http:1.4.9'
}
```

## 第10章 測試模型

### 10-1 測試計畫

以使用者角度使用此應用程式：

1. 商品資訊：商品是否顯示成功且正確。
2. 商品評價：是否成功跳轉並自動搜尋。
3. 匯率查詢：是否成功抓取即時匯率並轉換成功。
4. 產品詳細：商品資訊是否顯示成功且可以播放語音

### 10-2 測試個案與測試結果。

▼表 10-2-1 測試功能「登入」

功能編號	1	功能名稱	商品資訊
測試流程	1. 進入 App 2. 掃描超商商品條碼		
預期成果	讀取成功，顯示掃描商品資訊		
執行結果	讀取成功，顯示掃描商品資訊		

▼表 10-2-2 測試功能「商品評價」

功能編號	2	功能名稱	商品評價
測試流程	1. 點擊評價		
預期成果	跳轉至外部瀏覽器並自動搜尋對應商品評價		
執行結果	跳轉至外部瀏覽器並自動搜尋對應商品評價		

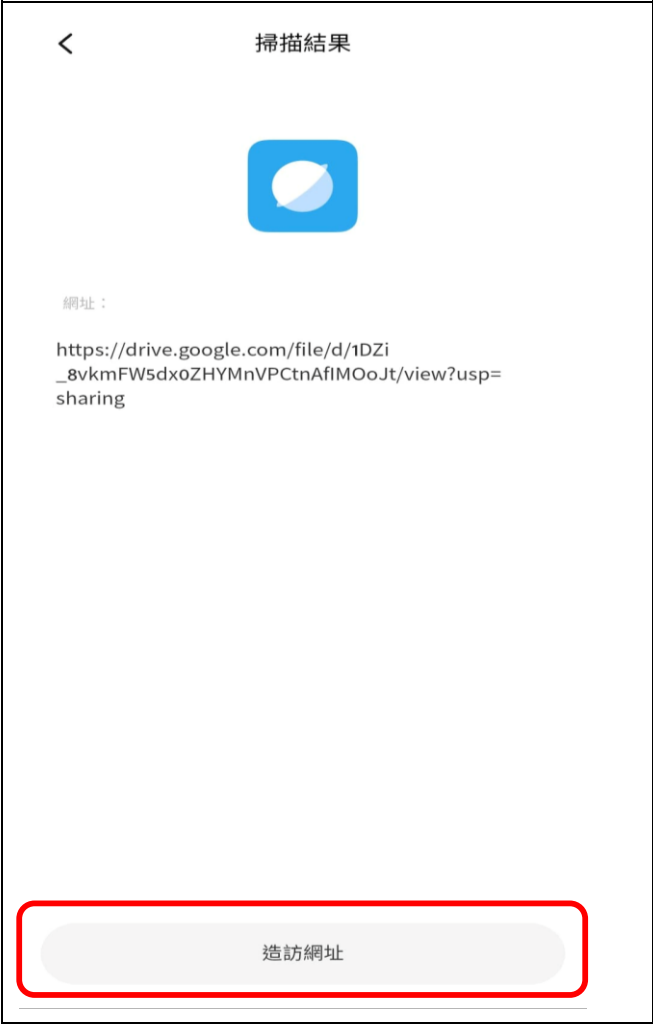
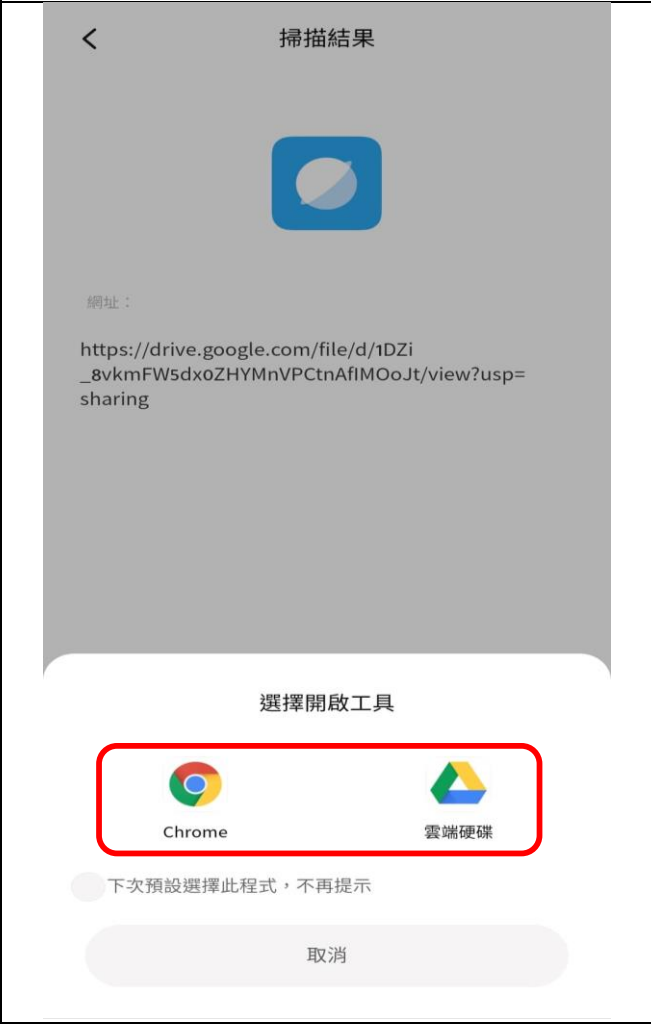

▼表 10-2-3 測試功能「匯率查詢」

功能編號	3	功能名稱	匯率查詢
測試流程	1. 點選匯率圖示 2. 輸入金額 3. 選擇轉換的幣種 4. 點選轉換鍵		
預期成果	顯示轉換後正確金額		
執行結果	顯示轉換後正確金額		

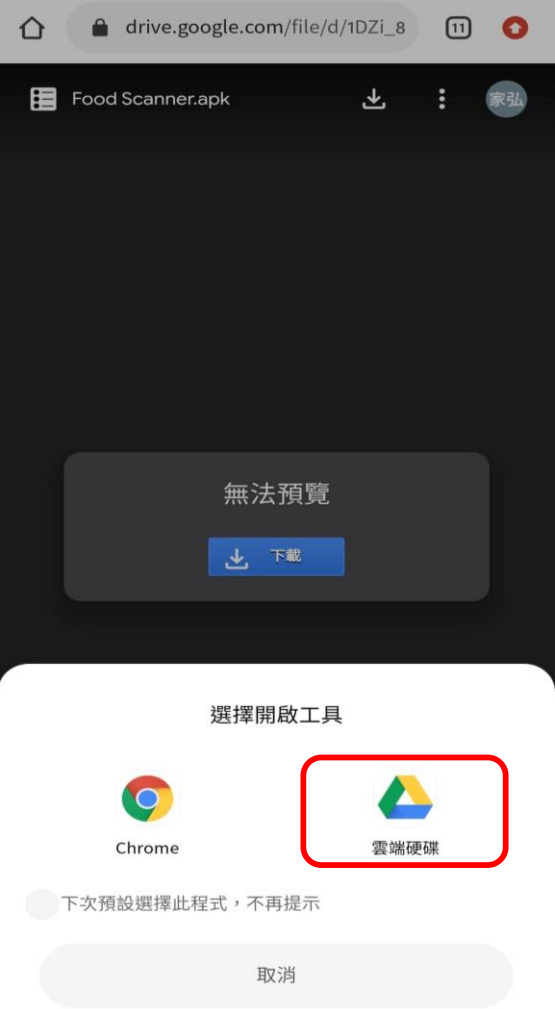
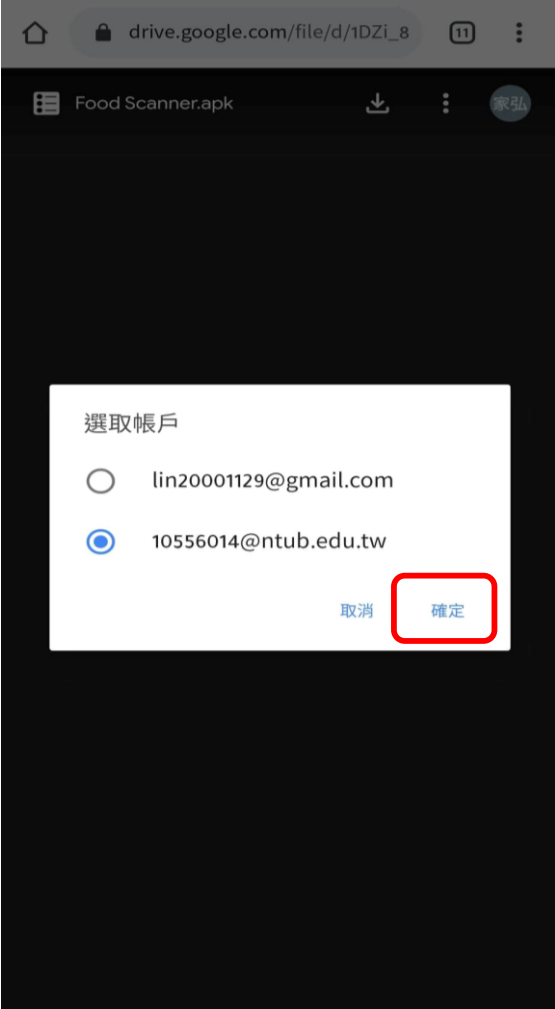

▼表 10-2-4 測試功能「商品詳細」

功能編號	4	功能名稱	商品詳細
測試流程	1. 進入 App 2. 掃描超商商品條碼 3. 點擊 MORE		
預期成果	顯示正確商品詳細，且播放語音		
執行結果	顯示正確商品詳細，且播放語音		

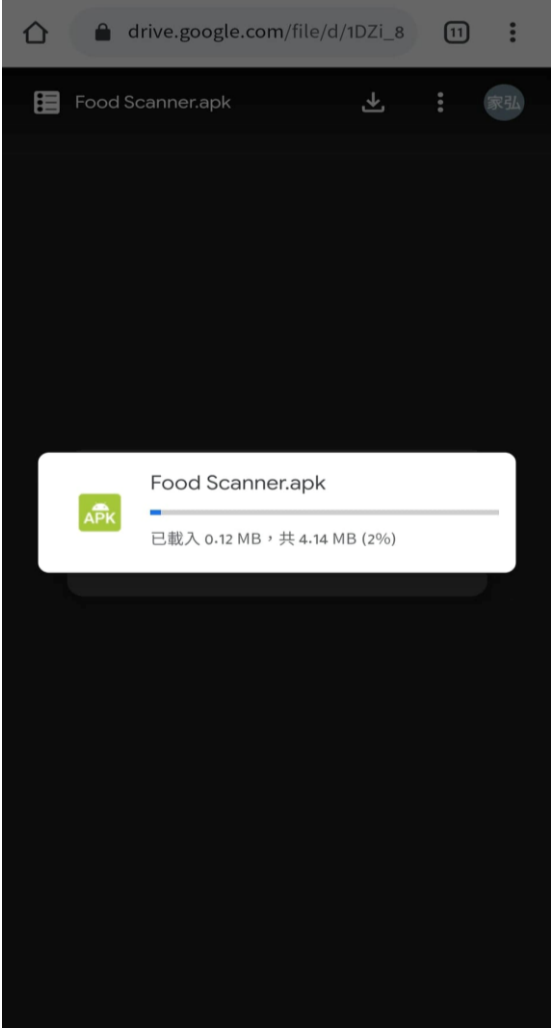
▼表 11-1

步驟 1	步驟 2	步驟 3
造訪掃描 QRCode 得到的網址	選擇 Google/雲端硬碟，開啟連結	點擊下載
		

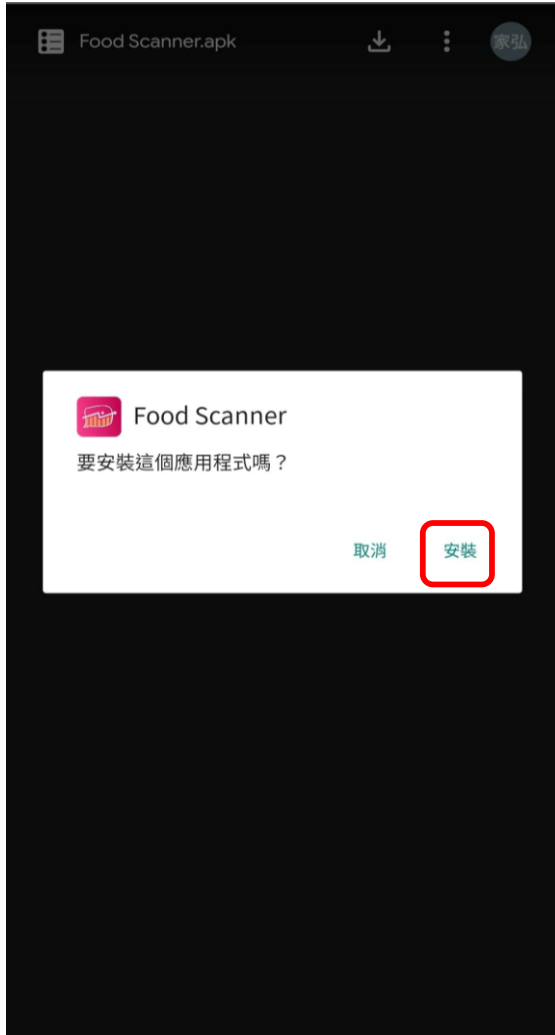
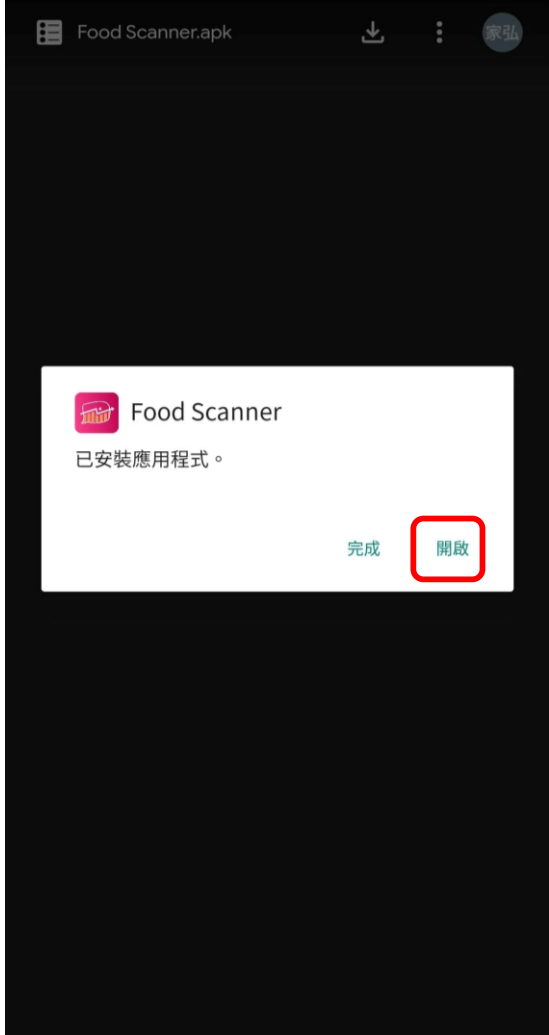
▼表 11-2

步驟 4	步驟 5	步驟 6
選擇雲端硬碟，開啟連結	選擇 Google 帳戶	點擊套件安裝程式
		

▼表 11-3

步驟 7	步驟 8	步驟 9
等待下載	點擊設定	開啟允許這個來源的應用程式
 <p>Food Scanner.apk 已載入 0.12 MB，共 4.14 MB (2%)</p>	 <p>雲端硬碟 為了安全起見，你的手機禁止安裝這個來源提供的不明應用程式。 取消 設定</p>	 <p>安裝未知應用</p> <p>允許這個來源的應用程式</p> <p>來歷不明的應用程式可能會損害你的手機和個人資料。安裝這個來源提供的應用程式即表示你同意，如因使用這些應用程式導致你的手機受損或資料遺失，須自行負責。</p>

▼表 11-4

步驟 10	步驟 11
點擊安裝	點擊開啟
 <p>Food Scanner.apk</p> <p>Food Scanner</p> <p>要安裝這個應用程式嗎？</p> <p>取消 安裝</p>	 <p>Food Scanner.apk</p> <p>Food Scanner</p> <p>已安裝應用程式。</p> <p>完成 開啟</p>



第12章 使用手冊

▼表 12-1

<div>查詢商品資訊</div> <div>步驟一</div> <div>點擊著陸頁</div> <div></div>	<div>步驟二</div> <div>掃描商品</div> <div></div>
<div>步驟三</div> <div>點擊 MORE 鍵</div> <div></div>	<div>最終結果</div> <div>商品詳細資訊</div> <div></div>

▼表 12-2

匯率換算	步驟二
步驟一 點擊匯率鍵	輸入欲轉換價錢
	
步驟三 選擇幣種	步驟四 點擊轉換
	

最終結果
------

轉換後價格



▼表 12-3

查詢商品評價
步驟一
點擊評價


最終結果
外部瀏覽器之商品評價


## 第13章 感想

10556047 周鈺祥

We do it! 我們做到了！

經過了這一年半的專題終於要宣告結束了，我們一路上經過許多波折，一開始運用影像辨識開發面相占卜，當初我們接觸這個主題也是相當沒有概念，不斷的在網路上搜尋資料，尋找有關面相學、相片分割、影像辨識的各種資訊，從第一次評鑑時我就已經在學習臉部辨識程式，但一路摸索到第二次評鑑，還是只有基礎偵測到臉部，直到二評評審的建議，認為技術含量過高，我們手握的資源、能力難以完成，也面臨在二評結束後，不得不重新思考專題方向。

一年半之中也發生過各種大小事，造成我們團隊之間溝通落差，或是做出來的結果跟原先計畫不同，引起些許紛爭，傷了和氣，還好我們彼此間還是願意冷靜思考該如何解決，也讓我學習到：專題製作不僅僅是對系統設計能力的考驗，也要磨練彼此團隊合作的精神，才能夠達成共識，一同完成專題研究。

儘管過程跌宕起伏，我們依然克服萬難，在短短幾個月中，做出一個為外食族、外國人士打造的食物搜尋利器，快速方便且實用的功能，也令我十分喜歡，希望這項 App 之後能幫助更多人。

最後，謝謝隆君老師及組員們一路上陪伴與支持，大家辛苦了！

經歷了一年半，我們的專題報告終於要畫下句點。在這漫長的時間裡，雖然有過不少的爭執與磨合，也換過題目，時間的壓力壓得我們喘不過氣，但這些困難也讓我們學習到很多，讓我理解了一個專案的從無到有需要運用哪些技術，在團隊中需要如何溝通與合作，我也特別想感謝我的隊友們，若沒有他們的協助，相信我們無法完成這次的專題研究。

我也想感謝我們的專題指導老師——張隆君老師，若是沒有他一路支持著我們的各種想法，我們不可能如此順利，不管遇到甚麼瓶頸，我們都挺過來了，相信這些經驗會成為我們未來的養分，滋潤著我們成長，這些老師所傳授給我們的技術與想法，我一定會學以致用將這些技術帶往我未來的生活與工作。

生命是一張弓，那弓弦是夢想。——羅曼·羅蘭

成就夢想所需的，並不僅僅是需要豐富的想像力與創意，同時也是需要團隊合作，最重要的是踏出第一步的勇氣。一直以來，我對於自己的能力抱持著疑惑，導致我難以在第一時間做出最佳的判斷，也沒有勇氣邁向夢想的第一步。在這次專題的過程中，我也認知到了團隊中的每一個人皆是這個專案的一個齒輪，缺少了任何一個，整個程序便無法進行，我也對自己的能力更有自信，相信現在的我，能夠面對自己的夢想，為自己的未來塗上鮮艷的色彩。

醴泉無源、芝草無根、人貴自勉-《會稽典錄》

這一年半的專題讓我學到了很多為人處世，使我深刻理解了一個專案從發想到實現，中間的過程與艱辛之處，有許多東西都必須要我們自發去學習，自己去理解，不像以前的每一門課，有問題都可以尋求老師的協助。

我們在第二次評鑑後，小組經歷了一次重大決定:更換題目，這個與眾不同的經驗，讓我深刻的領悟到很多東西，五個人說可以是沒有用的，要大家都同意且執行才能做到，很多事情不是我想像中的容易執行，往往要做到某個程度才會發現困難之處，我也非常感謝我們的組員每個人都盡力發揮自身所長，也非常努力的想把這個專題做好，即使我們因為更換題目充滿壓力，我們依然如期完成。

我也要感謝所有幫助、教導我們的老師，尤其是我們的專題老師張隆君，他在線上開會時提出很多意見，在我們題目發想階段與我們討論很多，也提供給我們最初的題目，只可惜我們因為許多原因無法實行到最後，但我還是非常謝謝老師，讓我理解到選題目的重要性。

這一年半的專題研究，可謂一波三折，最初我們在指導老師的提議下，選擇製作與影像辨識相關的主題，因為是沒有接觸過的陌生領域，我們瀏覽且蒐集了數以百計的資料、認真的研究影像辨識是如何使用 OpenCV 等軟體運作，並提出了完善的開發目標、應用方向、商業模式...等，第一次評鑑時的三位評審委員，對我們「將影像辨識結合 AI 機器人」的發想讚譽有加，也讓組員們備感信心。

隨著第一次評鑑的結束，我們在評審的肯定及建議下，開始踏入程式撰寫、軟體開發的部分，我們非常了解:這是一個困難的題目，一個軟體開發需要不同領域的人合作，而恰巧我們組正是每個人擁有不一樣的專長，所以我們也在資源匱乏的時間壓力的情況下簿案盈几、剖決如流，轉眼間也迎來第二次評鑑。

儘管困難重重，我們依然做足了準備，第二次的評鑑委員卻不太滿意，提出了與第一次評鑑委員相反的評論，質疑我們的專題發想，並且認為進度落後，要求我們更換方向。這樣的結果使我們組員間出現很大的分歧，有人認為應該聽從意見變更方向，也有人認為應該堅持到底不輕言放棄，最後歷經三個月與老師的協調討論，我們決定放手一搏，選擇屬於我們的路:更換題目。

謝謝老師同意成為指導老師；謝謝老師的栽培；謝謝老師百忙之中與我們線上開會；謝謝老師的各種意見；謝謝老師訓練我們獨立思考；謝謝老師激發我們的創造力；謝謝老師讓我們了解運算思維；謝謝老師讓我們有學習「#面對問題、#處理問題、#解決問題的機會」；最後，謝謝老師的翻轉教育。

「天行健，君子以自強不息」。



經過了長達了這一年半的專題研究，讓我自己深刻的學習、得到了許多不少以前從來都沒有過的特別的經驗，與組員們互助學習&&認真的研究，製作了一個只專屬於我們這組自己的特別的 Application：「Food Scanner—超商食品偵查員」。

在這個長達一年半的途中，我們遇到了很多真的很困難的事，以及很多的不一樣的想法衝突，雖然我們的組員彼此之間都有過鬧的很不愉快的時候的那些經驗，不過但是這也讓我們更加的了解彼此之間的不一樣的想法。

我們的專題老師：運算思維實踐家—[Angus Chang](#)，他也提供了很多很多不一樣的意見&&他的不一樣的想法，讓我們五個組員有了更多更多的更不一樣的想法。雖然但是在第二次的系統的評鑑之後，我們換了我們的專題題目，但我們大家也還是積極的討論更換的題目及更多的功能，我真的覺得很開心能夠跟這四個我的組員一起努力的合力完成這一個屬於我們的 Application：「Food Scanner—超商食品偵查員」系統，所以我也非常非常的感謝他們這四位我的組員的協助以及我們的專題老師：Angus Chang—運算思維實踐家，認真&&非常細心&&非常努力的指導！

```
package 10556038;

public class Experience{
    public static void main(String[] args) {
        if(不能幫自己的表現評分) {
            System.out.println("黃炘烱:00000000 分");//0 分
        }else{
            System.out.println("黃炘烱:01100100 分");//100 分
        }
    }
}
```

## 第14章 參考資料

網站資料：

經濟部統計處

[https://www.moea.gov.tw/MNS/dos/bulletin/Bulletin.aspx?kind=8&html=1&menu\\_id=6727](https://www.moea.gov.tw/MNS/dos/bulletin/Bulletin.aspx?kind=8&html=1&menu_id=6727)

統一超商

[https://www.ir-cloud.com/taiwan/2912/irwebsite\\_c/index.php?mod=monthlysales&id=2020](https://www.ir-cloud.com/taiwan/2912/irwebsite_c/index.php?mod=monthlysales&id=2020)

[https://www.ir-cloud.com/taiwan/2912/irwebsite\\_c/index.php?mod=monthlysales&year=2019](https://www.ir-cloud.com/taiwan/2912/irwebsite_c/index.php?mod=monthlysales&year=2019)

未來流通研究所

<https://www.mirai.com.tw/>

公平交易委員會

<https://www.ftc.gov.tw/>

Google API

<https://console.developers.google.com/?hl=zh-tw>

Exchangerate-API

<https://www.exchangerate-api.com/>

i18n 多國語言

<https://medium.com/%E5%B7%A5%E7%A8%8B%E5%B8%AB%E6%B1%82%E7%94%9F%E6%8C%87%E5%8D%97-software-engineer-survival-guide/android%E5%A4%9A%E8%AA%9E%E7%B3%BB%E5%9C%A8%E5%9C%B0%E5%8C%96-i18n-%E6%94%AF%E6%8F%B47-0-%E7%B9%81%E4%B8%AD-3ac0fc6ffdbb>

## 附錄 審查評審意見之修正情形

▼ 表 14 評審意見表

評審建議事項	修正情形
1. 占卜的本質為何?如何與使用者的需求 meet，宜深入研究	綜合以上建議事項，以及評審楊老師的建議:「核心系統的難度甚高，在剩餘時間考量下可能無法如期完成」，因此希望我們能夠再三考慮是否延續此主題。在我們與專題老師討論後，決定更換主題，故無以上建議事項之修正情形。
2. 專題是想以人臉辨識取代專業占卜師?否則單純"娛樂"是否能夠讓 App 永續經營?宜考量	
3. 結合機器人是有趣的嘗試，但必須重視占卜的本質，在於"專業與信任"如何說服使用者此系統的專業度?	