

108 年度氣候變遷創意實作競賽

決賽作品說明書

隊伍編號+名稱	021 進擊的農夫
作品中文名稱	農織合度-最佳化種植決策
作品英文名稱	Farming helper

參賽學校：國立台灣大學

指導老師：童慶斌

團隊成員：黃宇弘、李冠緯、蘇柏年

目錄

壹、作品摘要	3
貳、設計構想與運作說明	3
(一)問題界定.....	3
(二)預期成果.....	3
(三)APP 開發	4
參、作品材料說明	9
(一)開發裝置.....	9
(二)開發語言.....	9
(三)作物資料庫.....	9
(四)歷史氣候資料.....	9
(五)中央氣象局季長期展望報告.....	9
(六)圖片.....	10
肆、創作特點與創意說明	10
(一)短期氣候預測.....	10
(二)跨領域整合.....	10
(三)使用客群廣.....	11
伍、作品應用範圍及發展潛能	11
(一)商業模式.....	11
(二)教育應用.....	11
(三)學術應用.....	11
(四)作物生產應用.....	12
(五)發展潛能.....	12
陸、工作分配	13
柒、謝誌.....	13
捌、參考資料	13
玖、圖片來源	14

壹、作品摘要

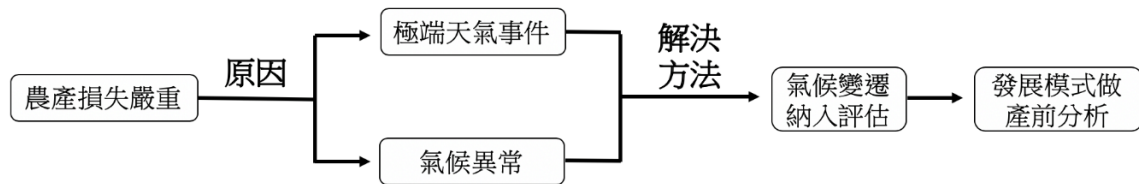
農織合度 APP 是一個將氣候變遷科學應用於農業的工具，我們希望藉此幫助使用者在氣候變遷下做出最佳的種植決策，我們藉由調整作物種植日期計算獲益最大化的種植方法，同時在此平台提供作物資料及氣候資料給使用者參考。

貳、設計構想及運作說明

(一)問題界定

現今農業耕作大多沒有考量氣候變遷，導致每當氣候異常、極端氣候事件發生時，便會造成**重大農業損失**，但氣候變遷因素真的難以預測嗎？我們認為當今在氣象學及氣候變遷科學領域的知識已逐漸成熟，若能納入這些知識，將能填補農民曆無法反應的氣候變化，並且增加未來農民面對氣候變遷影響的調適能力。現今氣候變遷科學的發展大多僅在學術領域，很少落實到大眾的生活中，我們也希望能提供一個平台讓大眾更加了解氣候變遷。

我們期望這項產品能夠**提升**農民面臨氣候變遷的不確定氣候風險時的**調適能力**，使農民在栽種管理過程中可以將**利益最大化**，促使**農作物產量增加**，也讓更多人願意投入農業生產，同時拓展大眾對於氣候變遷科學領域的認識，達到教育社會大眾的功能。



(二)預期成果

1. 有效計算氣候風險及利潤期望值

目前透過中央氣象局季長期氣候預測資訊、降尺度方法及統計方法，我們可以產生短期未來（三個月）的逐日氣候預測資料，在瞭解作物適合生長之氣候條件並建立作物資料庫後，設定一段預計種植某作物的時間，並按照這段時間內每日的各氣候因子與適合作物生長之氣候條件的差距給出一個氣候分數，將**逐日分數加總後除以整段作物生長期間最大可能總分**，我們將此值定義為單一作物生長的氣候風險；再將該氣候風險乘以單位面積收穫價格後可以得到粗略的利潤期望值。

2. 計算作物最佳種植決策並給予建議

使用此 APP 時使用者可以自行輸入候選作物及可耕種時間，每一種作物在不同的種植時間下都會有不同的氣候風險及利潤期望值，根據使用者輸入的時間及候選作物，我們可以利用演算法求出在農田可耕作時間內**利潤期望值加總的最大值**，以此作為**最佳耕作組合**呈現給使用者，同時我們也會呈現總利潤期望值，使用者可以

以此為基準比較不同的種植組合間的差別。

由於使用者不會只選擇一種作物，因此我們在計算時必須全盤考慮，評估哪種組合可以讓加總的期望值最大，再依據此評估結果提供農民建議。考量到農民使用時可能會想知道種植不同作物的利潤期望值差多少，因此我們會計算各種作物若要成為最大利潤者，其單價應該調升為多少，讓農民評估是否要更換作物。

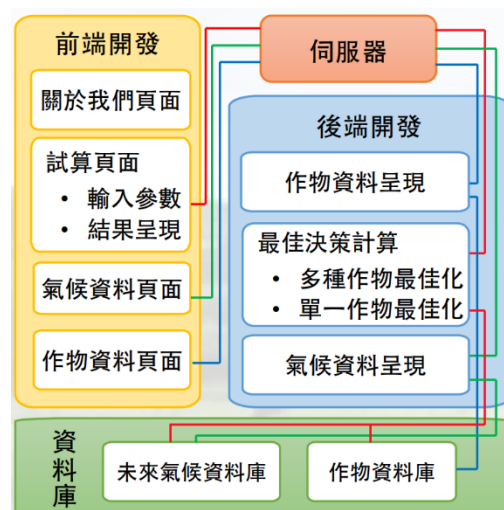
3. 提供使用者未來氣象資料及農作物栽種資訊

由於大眾對於氣候變遷了解不多，也並不太清楚未來氣候變化的趨勢為何，因此我們希望透過此平台提供我們所預測的未來氣候資料供使用者作參考；除此之外我們也希望提供整理過後的各種作物適合栽培方式及氣候因子限制給使用者，讓想投入農耕的使用者可以參考以決定要種植何種作物。

(三)APP 開發

1. 系統

在農織合度 APP 中，整個系統可分為前端開發、伺服器、後端開發及資料庫四個部分，前端為各個 APP 頁面的呈現；後端開發可分成三部分，作物資料呈現、氣候資料呈現及計算最佳種植組合；伺服器為前端與後端之間的橋樑，接收來自前端的資料或指令給後端，並且將其結果傳回前端；資料庫存放計算時所需要讀入的資訊。以下將針對後端開發與前端開發做更詳細的說明：



A. 後端開發

由於前端我們以 JavaScript 開發，為了使前後端銜接上更加方便直覺，我們選擇以 node.js[1] 為架構開發此程式的後端；另外，伺服器端我們目前使用 Express.js 框架[2]；資料方面，由於仍處於開發階段資料量還沒有很大，我們暫時沒有連接資料庫，而是把資料用 excel 表單或 JS 檔案存好。

我們設定此程式主要有三大功能，分別是**種植最佳化決策**、**未來氣候資料呈現**，及**作物資料呈現**，因此我們將後端程式開發分成這三個部分分別進行。

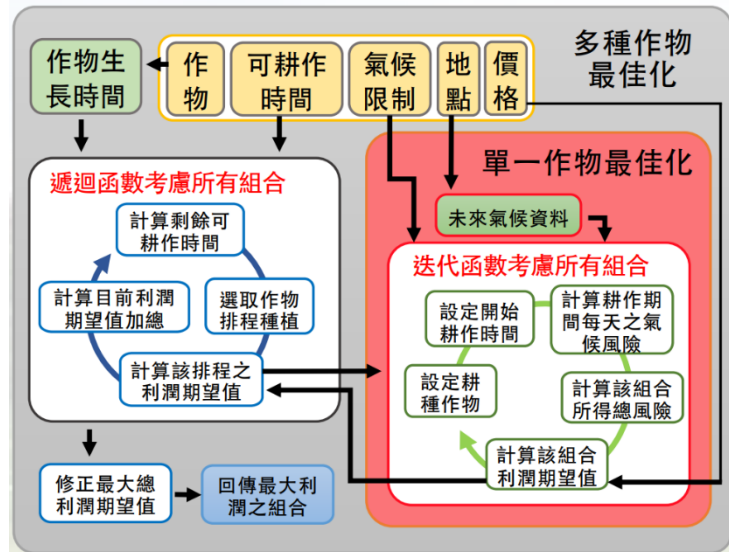
a. 種植最佳化決策：

我們將作物種植決策最佳化的過程分成兩個部分，首先計算單一作物在不同時間下的氣候得分，接著再用演算法找出這段時間內什麼樣的種植組合是利潤期望值最大的。

在計算單一作物的氣候得分方面，我們將作物生長階段中每種氣候條件每天的值分成幾個區間，並給予權重，此權重大小視作物對該種氣候條件的調適能力而定，比對未來每日每種氣候條件落在哪一個區間內，該日該種氣候條件便會得到對應的分數。舉例而言，菠菜在發芽階段(前 15 天)的適合溫度為 15 到 20 度之間，若超過 35 度或低於 4 度則發芽困難，我們定義若溫度介於 15 到 20 度之間則

權重為 2 分；若介在 4 到 15 度或 20 到 35 度間則權重為 1 分；其餘權重為 0 分，若該日溫度為 23.5 度則其在這天溫度方面得 1 分。我們將整個生長階段中的所有分數加總並且除以最大可能總分，作為該作物在此段種植時間下的氣候風險，接著再乘以該項作物的單價，便可得到我們定義的利潤期望值。

計算完每種作物對所有可能種植時間的利潤期望值後，我們接著要找可以讓總利潤期望值最大的種植組合，目前由於未來氣候資料限制的關係，我們最長只能模擬三個月，因此可以直接計算**全域最佳解**而不會花太多時間，計算時我們用遞迴的方法，當栽種好第一種作物之後，針對目前僅剩的時間再進行第二次種植的排



列，依此類推找出第二次種植的作物、第三次種植的作物…，窮舉完後我們會更改第一次種植之作物，依同樣方法去類推。我們會有記憶體不斷記錄分數最高的種植組合及其分數，直到比完所有組合，便代表我們已經找到總利潤期望值最大的種植組合。右圖為決定最佳種植決策組合背後的演算流程。

得到最佳種植組合之後，我們除了計算利潤期望值最大組合之外，亦去計算當最大利潤組合之外的作物售價調整為多少元時，最佳栽種之組合會變動，也就是各作物的替換價格。

b. 未來氣候資料呈現

在取得中央氣象局季長期展望報告及歷史氣候資料後，我們將過去 30 年來台北測站的歷史氣候資料格式轉成每一日一筆，也就是說每一個月總共會有 30 筆歷史資料，



而 30 年換算之後同一個月則有約 900 筆資料，由於氣象局季長期展望報告是以月為單位，因此我們假設同一月份的資料有相同統計特性，再利用此資料產生**經驗分布函數(ECDF)**，得出一個月份有一條回歸曲線，而後根據季長期展望報告中的機率，利用蒙地卡羅方法[3]模擬出未來 3 個月(4-6 月)每日的氣候資料，除了呈現亦用來評估最佳化的種植決策。

c. 作物資料呈現

作物資料方面我們參考書籍將各個作物的栽種資訊整理好後以 JS 檔來儲存，伺服器端會讀取作物資料將之傳至前端呈現給使用者，讓使用者能夠參考此栽種資訊去調整其適合的作物栽種方式。

B. 前端(Client)開發

前端設計我們採用 React-Native[4] 架構開發，其優點是自由度高，**可動態生成畫面**，且較不會受限於手機的作業系統，目前我們先以 iOS 系統呈現為主。我們設計四個分頁，分別是“關於我們”、“試算”、“未來氣候”及“作物資料”，以下將各別說明我們如何呈現：

- a. “關於我們”頁面：此頁面主要介紹我們團隊的基本資料，另外還有該如何使用此 APP，包括每個按鈕的功能、數據解釋等等。
- b. “試算”頁面：此頁面在計算最佳化種植決策，使用者需要自行輸入開始、結束時間、耕作地點、候選作物，及各候選作物所要考慮的氣候因子，這些參數都是此頁面中的 state [5]，使用者改變數值的同時這些參數也會透過 setState 函數被寫入檔案中，輸入完畢後點擊“確定”便可以將輸入值送至伺服器端運算，待計算完成後，伺服器端會回傳結果，經過整理呈現在頁面下方。
- c. “未來氣候”頁面：當切到此頁面時，程式會自動連上伺服器取得氣候並且刷新畫面，這樣的好處是未來若要更新資料只需從後端更新便可，使用者不需要另外更新手機程式。在此我們用表格的方式把逐日的未來氣候預測資訊呈現出來讓使用者查看。
- d. “作物資料”頁面：和“未來氣候”頁面相同，程式會自動連上伺服器抓取作物資料，其中包含作物圖片、種植需注意事項、生長條件等等，以列點的方式呈現。

使用 React-Native 架構的特點是頁面的每個功能都可以獨立做成元件(React component[6])，具有相同特性部分可以以元件為模板動態產生而不用寫死，達到**有系統的整理程式**，階層式結構不僅讓程式跑起來更有效率，也提高了共同開發的方便性；另外，React 架構下 render 函式[7]會自動隨時監聽各個 state 及畫面上的元件，當數值有改變時會立即進行部分的刷新，達到最有效率的動態畫面生成，比起傳統網頁或傳統 APP 要針對每個元件寫許多監聽及更新函數要方便許多。

開發 React 專案時，我們將許多重要資訊設成上層頁面的 state，與 state 相關的邏輯判斷則寫成 call back function[8]設為此頁面的 class property [9]，接著再利用 this.props 的方法將這些 state 資料或 call back function 傳到每個元件去執行，這樣的好處是所有與 state 相關的邏輯判斷與指令都是寫在主要檔案裡，較不會混亂也不會有互相衝突的情形發生。

連接伺服器我們使用 JavaScript 內建 fetch 函式中的 'POST' 方法[10]，使前端與後端能夠互相溝通，由於目前測試階段伺服器端仍沒有固定 IP 提供服務，

使用前使用者要先輸入 IP 位址才可連接伺服器。未來此程式開發成熟時定會有一個固定 IP 提供服務，屆時便可以省去這個步驟。

2. APP 介面呈現

A. 試算畫面

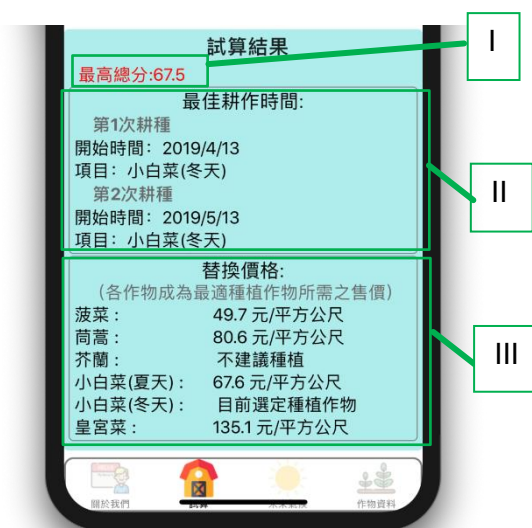
a. 自行輸入值

- I. **日期**:輸入農地可以耕種的時間。
- II. **地點**:選擇農地位置。
- III. **作物選擇及作物價格**:從資料庫中已有的作物選擇想要考慮進行種植評估的作物，目前作物資料庫中有菠菜、茼蒿、芥藍菜、夏天種小白菜、冬天種小白菜、皇宮菜、葉萵苣、結球萵苣、莧菜等 9 項，農民可以根據喜好選擇想要評估的作物，及預定的作物價格(單位面積多少元)。未來期望可以陸續增加新種作物，使整個程式更加適用。
- IV. **氣候因子**:每種作物有四種氣候因子，分別為雨量、溫度、風速、日照，由農民選擇其認為種植過程中作物需要考慮的氣候因子。
- V. **確定鈕**:當全部的輸入都結束後，即可按下確定鈕讓 APP 試算。



b. 計算後的結果

- I. **最高總分**:此總分代表此氣候風險下的最大利潤期望值。
- II. **最佳耕作時間**:此呈現的是最終產生最大利潤的組合，在這段農地可種植的時間內可以種幾輪，每一輪中有開始的時間以及種植的作物。
- III. **替換價格**:由此可知當其他作物調整其價格至多少元時，其種植決策可以考慮種此作物以獲得最大利潤。“目前選定種植作物”為最大利潤之組合，“不建議種植”原因為價格不斷上升組合依舊不變。



B. 介紹畫面

此部分為介紹 APP 之頁面，除了介紹我們的團隊，包含我們的背景、為何想要做這種最佳化決策的 APP 的動機之外，也附有各分頁主要功能的說明，另外也利用圖與文字說明 APP 的操作方式、結果表單中各項代表的意義為何，讓使用者輕易了解如何使用此 APP。



C. 未來氣候資料畫面

因採用短期氣候預測的方式，故目前所產生的未來氣候資訊以 4-6 月的資料呈現，在我們可以取得的資料當中，溫度、降雨、日照、風速為影響作物生長的氣候因子，因而只呈現四種氣候因子的資料。左圖為分別每一天的溫度、雨量、日照、風速。



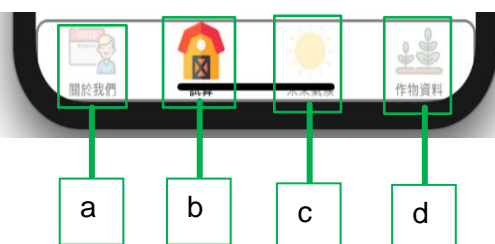
D. 作物栽種資訊畫面

主要以用土、播種、培育、收穫四大要項去呈現各個作物的簡略種植的注意事項，而目前的作物有菠菜、茼蒿、芥藍、夏天種小白菜、冬天種小白菜、皇宮菜、葉萵苣、結球萵苣、莧菜這幾種的說明，供作種植參考，右圖為此介面的呈現。

E. 分頁按鈕

這些按鈕分別為介紹畫面、試算畫面、未來氣候資料畫面及作物資料畫面：

- 介紹畫面**: APP 使用說明。
- 試算畫面**: 根據農民的輸入值計算此農地最佳種植組合的頁面。
- 未來氣候資料畫面**: 短期氣候預測的未來氣候資料。
- 作物資料畫面**: 各個作物的簡略種植方式。



參、作品材料說明

(一)開發裝置

我們選擇開發手機 APP 而非網頁的主因為在科技進步的時代下，手機幾乎是人手一台，人們花在手機上的時間比花在電腦上的時間多上許多，此外相對於電腦，用手機查詢較為方便。民眾可輕鬆藉由下載此 APP 以獲得服務。

(二)開發語言

在開發 APP 的過程中，我們選擇用 Javascript 作為我們開發 APP 的程式語言，選擇此語言的原因為這是目前最主流的網頁開發程式語言，且其中的 React native 廣泛被應用於手機 APP 開發。此部分主要可分為前端及後端：

1. 前端:功能為 APP 使用介面呈現，我們選擇用 Javascript 中的 React native 套件去開發，其優勢為能跨 Android 和 IOS 兩個平台，但此次我們先以 IOS 系統呈現為主。
2. 後端:為背後演算法的部分，使用 Javascript 中 node.js 套件去撰寫最佳化種植決策的計算，讓前後端銜接更加順暢，開發起來更為簡易。

(三)作物資料庫

菠菜、茼蒿、芥藍菜、夏天種小白菜、冬天種小白菜、皇宮菜、葉萵苣、結球萵苣及莧菜這些作物都是查詢自園藝編輯組所編著《有機蔬菜栽培指南》裡的資料，栽培資訊部份其每種作物內容主要分為用土、播種、培育方法與收穫來作敘述，而作物生長期間其可容忍的溫度、雨量、日照、風速範圍則尚有參考網路上的資訊。

(四)歷史氣候資料

因我們目前所設定的耕作地點為台大公館校區，因此我們從台大永續發展研究室取得台北測站(測站編碼:466920，位於中正區公園路 64 號)過去 30 年來的歷史逐時氣候資料，接著將其整理為逐日氣候資料並儲存。此資料會根據不同耕作地點而需要不同的測站氣候資料，因此若未來可選擇的地點越多，資料庫會越加龐大。

(五)中央氣象局季長期展望報告

中央氣象局在每個月底會發布一則季長期展望報告，其中包含未來三個月的每月氣候可能的狀態與機率，舉例而言，如右圖所示，三月所發布的季長期展望報告中，四月雨量預測低於歷史正常值的機率為 20%；接近為 50%；多於為 30%。我們將以此為參考產生未來三個月的逐日氣候資料。

4月：預測上旬仍有受北方冷空氣南下影響的可能，降雨以中部以北及東半部為主。預測平均氣溫各地以「低於」氣候正常值的機會最小；雨量預測，各地少於、接近、多於氣候正常值的機率為20%、50%、30%。

5月：氣候上而言，臺灣西半部雨量約於5月中、下旬有較顯著的增加，當鋒面在臺灣附近徘徊時易伴有雷雨，偶有大雨或豪雨發生。預測平均氣溫各地以「低於」氣候正常值的機會最小；雨量預測，各地少於、接近、多於氣候正常值的機率為30%、50%、20%。

6月：氣候上，6月上旬的鋒面系統較活躍，常造成豪大雨；下旬鋒面北抬後，逐漸進入夏季季節。預測平均氣溫各地以「低於」氣候正常值的機會最小；雨量預測，南部少於、接近、多於氣候正常值的機率為30%、50%、20%，其他地區為20%、50%、30%。

(六)圖片

1. 作物圖片：



圖一 菠菜



圖二 芥藍



圖三 小白菜



圖四 皇宮菜



圖五 茼蒿



圖六 莧菜



圖七 葉莧菜



圖八 結球莧菜

2. 首頁照片(背景):來源為自行拍攝



圖九

3. 按鈕的 icon:



圖十

肆、創作特點與創意說明

(一)使用短期氣候預測

未來的氣候預測一般是模擬十年甚至是二十年後等時間尺度較大的氣候，但是這種趨勢上的預測對於農民的影響往往是看不見效果的，因此我們使用中央氣象局季長期預測降尺度到逐日資料來獲得最佳種植決策。我們的產品為氣候變遷科學領域用於短期的結果，與一般溫度、雨量趨勢上的研究有所差異。

(二)氣候預測+作物生產+最佳化+程式開發:跨領域整合

我們將氣候變遷領域的氣候預測結果應於作物生產，並且將此兩者結合起來運用於最佳化及程式開發上，使得我們有辦法讓作物在最適合的氣候條件下生長，也藉此希望可以對我們的服務對象—農業生產，產生更好的成果預測及意見提供。

(三)使用客群廣:降低人民與氣候變遷科學之間的科技門檻

我們了解並不是每個人都知道如何用氣候變遷的知識去預測未來的氣候，故 APP 可以讓許多人在了解有限的情況下，依然可以利用氣候變遷研究產生的結果幫助自己。不只是農民，如教育界甚至可以將我們的 APP 納入於各國中推動的食農計畫中進行，讓國中學生們在學習如何種植蔬菜給自己吃之外，也了解氣候變遷可以如何為種植日期及產量預測作出建議。不需要額外深奧的背景知識，也因此這 APP 適用於任何年齡層的人。

伍、作品應用範圍及發展潛能

(一) 商業模式圖

<p>關鍵夥伴</p> <ul style="list-style-type: none"> ● 資訊工程師 ● 行政院農委會 ● 各地區農會 ● 氣候變遷永續發展研究室 	<p>關鍵活動</p> <ul style="list-style-type: none"> ● 設計 APP ● 政策推動 <p>關鍵資源</p> <ul style="list-style-type: none"> ● 歷史氣候資料 ● 植物生長資料 ● 季長期展望報告 	<p>價值主張</p> <p>利用此 app 在氣候變遷的狀況下選擇最大利潤之種植組合以獲取最大利益</p>	<p>顧客關係</p> <p>銷售 APP 提供客戶使用</p> <p>通路(廣告)</p> <ul style="list-style-type: none"> ● 客服服務中心 ● 文章粉專宣傳 ● APP 商店 	<p>目標客層</p> <ul style="list-style-type: none"> ● 台灣農民 ● 台灣國中小學生 ● 對氣候變遷有興趣之民眾
<p>成本結構</p> <ul style="list-style-type: none"> ● 廣告宣傳費用 ● APP 研發費用 ● 工程師薪資 		<p>收入來源</p> <ul style="list-style-type: none"> ● 購買 APP 之盈餘 ● 進階功能的部分收費 ● 贊助費用 ● 廣告 		

(二) 教育應用

可結合國中小的食農計畫，讓學生在種菜之餘也能藉由 APP 學習氣候變遷相關知識，也讓國中小學生可看見在氣候變遷下如何應用科技的力量去幫助農業生產，進而對氣候變遷學科產生好奇心，增加未來可能的專業人才去幫助減緩氣候變遷所帶來的影響，而使氣候變遷學科的未來更具有發展潛力也增加未來民眾面對氣候變遷的應對能力。

(三) 學術應用

透過 APP 可以在學術上尋找未來氣候與作物生長模式之間的關係，當學者們嘗試對未來氣候和作物生長模式做出更精準的預測的同時，這個研究也會幫助我們的 APP 進化。隨著日積月累的技术提升，農民將可以更精準的使用氣候變遷學科與農藝學科合併產生的跨領域知識，進而降低生產背後的風險，增加利潤而有辦法投資更多資源在這方面的研究上，引發產業與學術互利的良性循環。

(四) 作物生產應用

農民透過輸入自己想要栽種的農作物、想要考量的氣候因子、售價、農地可利用時間，我們可以提供給農民考量短期氣候預測之下最有利的栽種時間排序方式，讓農民輕鬆得知什麼時候種什麼擁有最大的利益。此外我們也提供給農民其他作物的替換價格而讓農民在選擇上有更多空間。

(五) 發展潛能

1. 自行訂定各種作物種植區間及利潤比較的功能建立

由於目前的種植區間及替換價格皆由程式自行運算後跑出，我們期望可以設計一個功能，讓農民自行輸入想要在什麼時候種什麼作物，然後由系統算出該種植決策下的利潤。農民可以拿自己決策的總利潤與最佳化後算出的總利潤作比較，而選擇捨棄或採取自己的決策。此一功能提供農民更多自由選擇的空間且目前該功能的後端已開發完畢，有賴於時間因素而無法將之與前端做連接，有待未來可以迅速建立。

2. 考慮市場供需的作物價格

目前價格為使用者自行輸入，無法直接反映市場價格，未來納入市場經濟供需研究後，可以讓使用者輸入耕作面積並在氣候條件下計算預估產量，便可以計算出預估市場價格，這樣的機制使用客群愈多市場價格便能預測的愈準，若能與各地區農會合作，最後將能真正解決農作物價格波動大的問題，達到農產供需平衡、價格穩定。雖然這樣的願景看似遙不可及，我們認為是非常有潛力的發展方向，將能大大降低農民的耕作風險！

3. 農耕地點更多選擇

根據不同的農耕地點，我們必須取得該地區附近的歷史氣候資料並且進行空間降尺度才能精準預測該地區的未來氣候，目前的模擬我們使用資料最完整的台北測站(測站編碼:466920，位於中正區公園路64號)的氣候資料，並且降尺度到台大公館校區，未來若要真的發展成適用於任何地點將需要更多氣象測站且更完整的歷史氣候資料，我們相信未來資訊爆炸的時代是做得到的，最終這項產品可以適用於全台灣甚至其他國家，作出個別化的服務。

4. 未來氣候資料更加精準

我們目前未來氣候資料的產生仰賴於歷史氣候資料及氣象局的季長期氣候展望報告，若要使這項預測更精準，除了歷史氣候資料需要更完整外(第3點也有提及)，也需要氣象局季長期展望的不斷優化，另外，產生未來氣候資料的方法也是可以再優化的，我們期待未來這些技術的提升可以使氣候預測愈來愈精準。

5. 使用客群

目前的鎖定客群皆為小農，期望在其他的領域知識逐漸納入、APP的功能趨

完善後，可以將客群從小農逐漸擴大到大農戶、產銷班、縣市政府、中央政府，使用客群增加不僅能讓產品決策更準確更能促進農業發展。

6. 農民參與作物資料庫的建立

因目前所納入的作物種類有限，我們期盼未來可以開發自行創造作物的功能，讓農民可以種出在我們資料庫中沒有納入的作物。農民設定其栽種期間及各個區間的氣候因子限制後，系統會建立新的作物模式檔案並將之納入計算。此一功能亦能解決在作物種類多樣化的情況下建立資料庫的困難，透過審核機制，我們的作物種類將會愈來愈多，面對種植決策也就有更多組合去比較，也能因此找到更大利潤的種植組合。

陸、工作分配

黃宇弘	未來氣候預測資料整理、前端開發、海報製作
李冠緯	後端開發、成果報告書製作
蘇柏年	後端開發、作物資料整理、成果影片製作

柒、謝誌

這次的成果很感謝各位教授、指導老師、學長們提供有如短期氣候預測資訊、農作物栽種及品種選擇注意要點，未來氣候資料生成方法、程式開發等相當重要的資訊，由於您的努力及幫助，我們有辦法於短時間內做出成品，並且避免了許多不必要的討論。很感謝您們犧牲自己的時間及對我們的付出，故在此文後再次致謝。

捌、參考資料

- [1]. <https://nodejs.org/en/>
- [2]. <https://expressjs.com>
- [3]. <https://zh.wikipedia.org/wiki/蒙地卡羅方法>
- [4]. <https://facebook.github.io/react-native/>
- [5]. <https://facebook.github.io/react-native/docs/state>
- [6]. https://reactjs.org/docs/react-component.html?utm_source=caibaojian.com
- [7]. <https://www.npmjs.com/package/react-native-render-html>
- [8]. <https://medium.com/@thejasonfile/callback-functions-in-react-e822ebede766>
- [9]. <https://michalzalecki.com/react-components-and-class-properties/>
- [10]. https://developer.mozilla.org/zh-TW/docs/Web/API/Fetch_API/Using_Fetch
- [11]. <https://developer.mozilla.org/zh-TW/>
- [12]. <https://ric2kl.github.io>
- [13]. <https://www.w3schools.com>
- [14]. <https://www.npmjs.com>
- [15]. <https://stackoverflow.com>
- [16]. 園藝編輯組(2005)。有機蔬菜栽培指南。台南市:文國書局

玖、圖片來源

圖一：

<https://kknews.cc/food/l2vobgz.html>

圖二：

<https://www.twfood.cc/vege/LK8/%E8%8A%A5%E8%97%8D%E8%8F%9C-%E6%B0%B4%E8%80%95>

圖三：

<https://www.twfood.cc/vege/LB11/%E5%B0%8F%E7%99%BD%E8%8F%9C-%E5%9C%9F%E7%99%BD%E8%8F%9C>

圖四：

<http://jona6677.pixnet.net/blog/post/223279990-%E7%9A%87%E5%AE%AE%E8%8F%9C>

圖五：

https://kids.coa.gov.tw/view.php?func=kids_school&category=A01&id=78&season=S4

圖六：

<https://www.twfood.cc/vege/LM2/%E8%8E%A7%E8%8F%9C-%E7%99%BD%E8%8E%A7%E8%8F%9C>

圖七：

<https://blog.xuite.net/hsu123.tw/wretch/511127106-%E8%91%89%E8%90%B5%E8%8B%A3%E6%A0%BD%E5%9F%B9%E7%AE%A1%E7%90%86>

圖八：

<https://www.owlting.com/market/items/10430>

圖十：

<https://www.flaticon.com/>