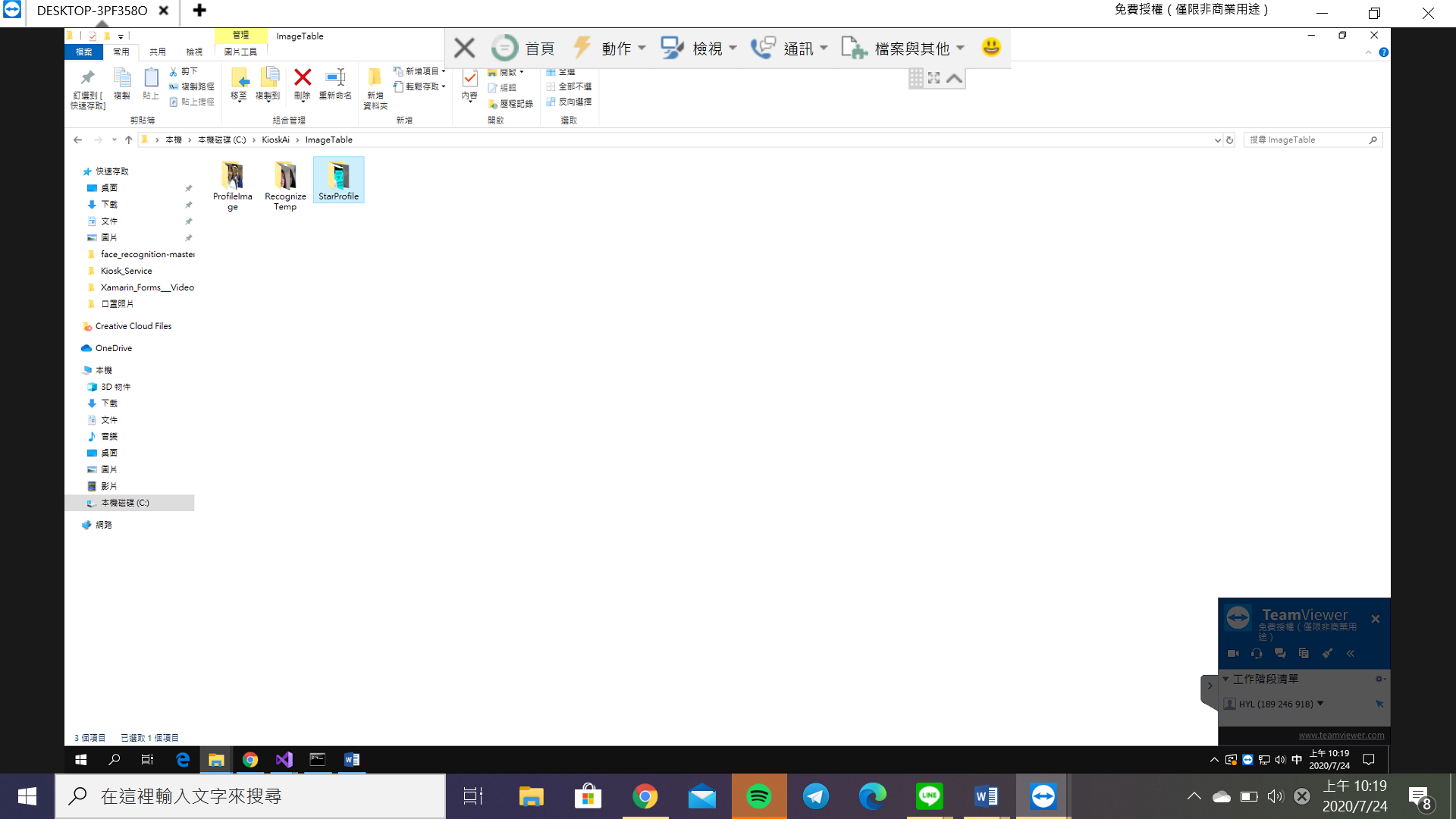
* **Server端(IP:** [120.126.151.103](http://120.126.151.103/kioskservice/help)**)：**

1. Kiosk\_Service資料夾

* 建置 : ASP.NET MVC架構，有將其上架到iis
* 講解 : 主要的接受KIOSK、手機等客戶端的請求，但AI的部分是放在另一個資料夾(django\_service)，本專案透過http去access AI的請求。

所以要改變server提供給KIOSK或APP的API才會需要改到這個檔案

* 連接資料庫(MS SQL SERVER)，只有儲存使用者的基本資訊 : APP註冊所填的資料，使用者的照片和明星照片以及AI模型都沒有存在資料庫裡。
* 照片位置的儲存路徑 : C:\KioskAi\ImageTable



ProfileImage : 人臉訓練的照片(app上傳做訓練的照片)

RecognizeTemp : 人臉辨識的照片 (KIOSK拍照做辨識的照片)

StarProfile : 明星臉偵測的照片

* Controller資料夾
* kioskController : 提供的api和http互相傳遞的程式邏輯撰寫
* Common資料夾
* CommonClass : 串接資料庫所有功能撰寫(查詢、新增…)
* ServiceTool\_Rugal : 主要是命名提供的API的function ，包含參數的回傳
* SrvModel\_Rugal : 參數的格式設定
* Models資料夾
* KioskModel : 資料庫連接的參數回傳設定

沒提及到的其他檔案就一些mvc架構的東西。

本服務所提供的API總覽 : <http://120.126.151.103/kioskservice/help>

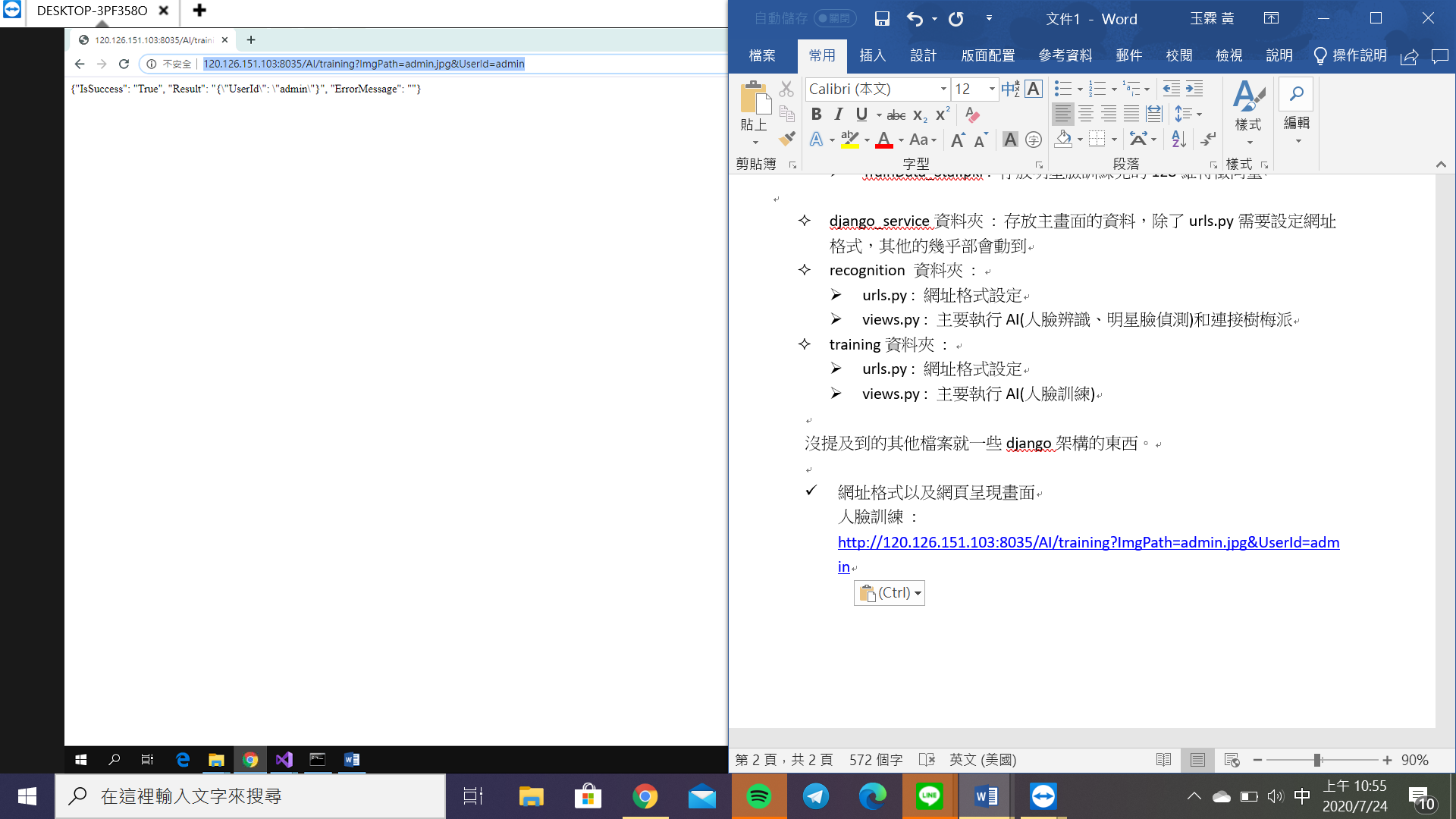
1. django\_service

* 建置 : django架構，有上架到IIS
* 講解 : 關於AI的部分都放在這裡，此外和樹梅派串接的也有
* TrainData.pkl : 存放人臉訓練完的128維特徵向量
* TrainData\_Star.pkl : 存放明星臉訓練完的128維特徵向量
* django\_service資料夾 : 存放主畫面的資料，除了urls.py需要設定網址格式，其他的幾乎部會動到
* recognition 資料夾 :
* urls.py : 網址格式設定
* views.py : 主要執行AI(人臉辨識、明星臉偵測)和連接樹梅派
* training資料夾 :
* urls.py : 網址格式設定
* views.py : 主要執行AI(人臉訓練)

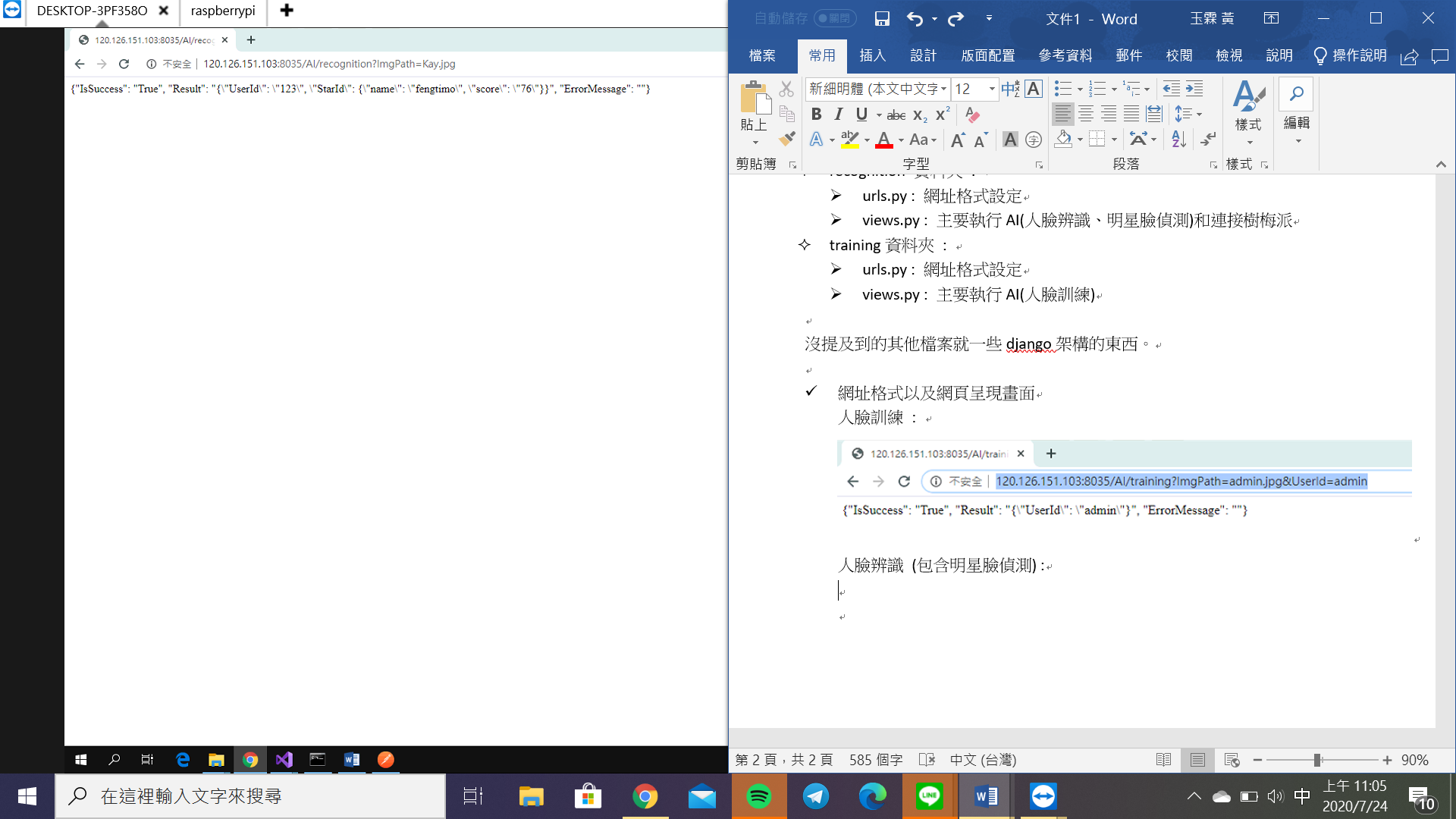
沒提及到的其他檔案就一些django架構的東西。

* 網址格式以及網頁呈現畫面

人臉訓練 :



人臉辨識 (包含明星臉偵測) :



* 手機app端(KioskPhoneApp\_new 資料夾)
* 建置 : xamarin架構，可以寫一個軟體就能讓android、ios、uwp使用
* KioskPhoneApp資料夾

1. Pages資料夾

* MainPage.xml : 主畫面
* Register.xml : 註冊畫面
* Upload.xml : 照片上傳做人臉訓練

1. NetCommon資料夾(與server連結的)

* SrvModel\_Ting .cs : 參數設定
* ServiceTool\_Ting.cs : 與server連結的function撰寫

沒提及到的其他檔案就一些xamarin架構的東西。

* KIOSK端(videoPlayer資料夾)
* 建置 : xamarin架構
* 點進這些路徑後 : Xamarin\_Forms\_\_\_Video\_Player\_Demos\VideoPlayerDemos\VideoPlayerDemos
* HomePage.xaml : 主畫面
* ResultSuccess.xaml : 辨識成功畫面
* ResultFail.xaml : 辨識失敗畫面
* NetConnect資料夾 : 負責與server連接，和app相同名字的檔案一模一樣

沒提及到的其他檔案就一些xamarin架構的東西，有一些我們去參考別人的檔案雖然沒再用，但不敢刪除害怕出錯。

* 操作流程

樹莓派 : 執行test2.py的檔案

KIOSK (1)按下按鈕會呈現全螢幕的相機畫面

(2)隨便觸控螢幕上各個地方，進行拍照的動作

(3)等待結果

手機app (1) 登入會員

(2) 以拍照或從相簿挑選要訓練的照片

需注意的流程

(1)每一次人臉辨識，都要重新執行樹莓派的test2.py的檔案

(2)app要做人臉訓練時，也要執行test2.py

(3)test2.py執行似乎超過1分半就會有問題不行用，所以要控制時間

總結:遇到人臉辨識和人臉訓練結果出不來的話，就重新執行test2.py即可~

* Teamviewer遠端操控ID

Server: 1462920185

Pi: 1591711487

Kiosk:1452854400

密碼都是jizzbear