海大機車共乘

設計文件

|  |  |
| --- | --- |
| 專案名稱 | 海大機車共乘 go |
| 撰寫日期 | 2019/11/1 ~ |
| 發展者 | 葉智竣  康致瑋 林立衡  黃炫峰 廖裕德 洪商程  韋彥安 |

**版次變更記錄**

|  |  |  |
| --- | --- | --- |
| 版次 | 變更項目 | 變更日期 |
| 0.1 | 初版 | 2019/11/1 |
| 0.2 | 新增資料設計,[流程設計](#_heading=h.2et92p0) | 2019/11/8 |
| 0.3 | 新增[系統模型與架構](#_heading=h.1fob9te),類別圖,介面... | 2019/11/15 |
| 0.4 | 修改介面需求與設計 | 2019/11/19 |
| 0.5 | 報告後依建議修正 | 2019/11/29 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

目錄

[**版次變更記錄 1**](#_heading=h.30j0zll)

[**1.**](#_heading=h.1fob9te) **系統模型與架構(System Model/System Architecture) 1**

[**2.**](#_heading=h.3znysh7) **介面需求與設計(Interface Requirement and Design) 1**

[**3.**](#_heading=h.2et92p0) **流程設計(Process Design) 1**

[**4.**](#_heading=h.tyjcwt) **使用者畫面設計(User Interface Design) 1**

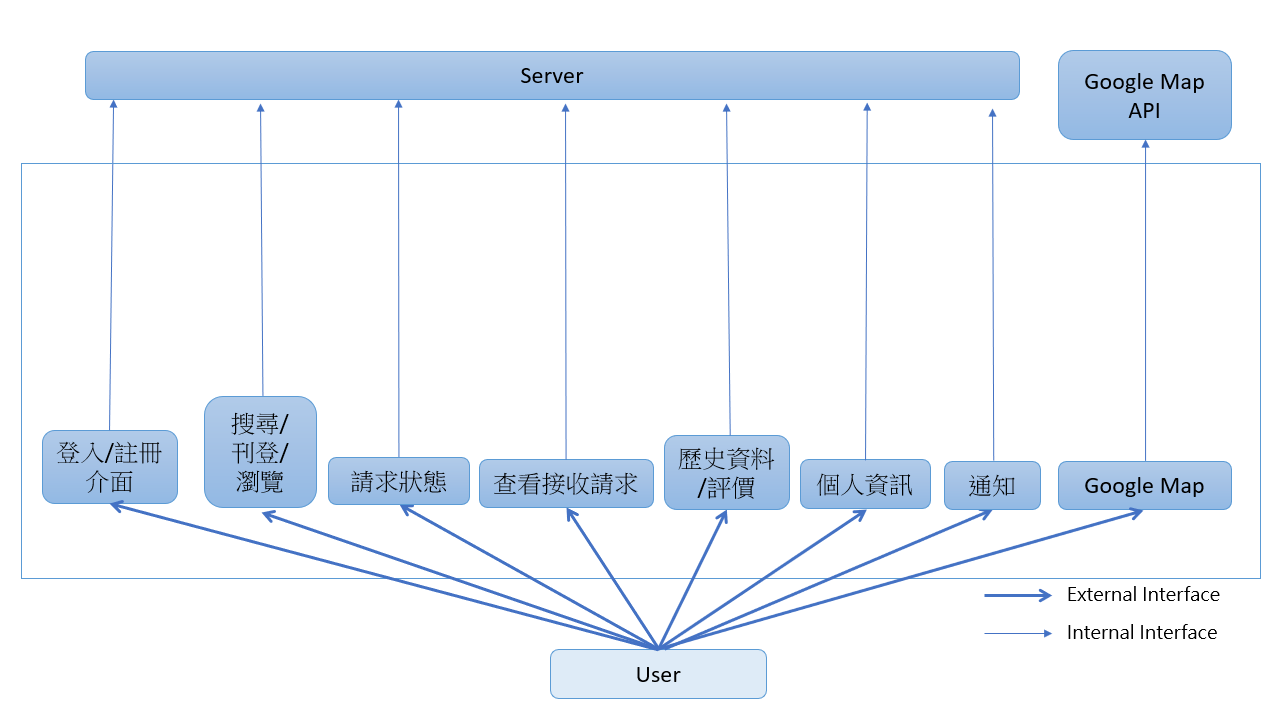
[**5.**](#_heading=h.3dy6vkm) **資料設計(Data Design) 1**

[**6.**](#_heading=h.1t3h5sf) **類別圖設計(Class Diagram) 1**

[**7.**](#_heading=h.4d34og8) **實作技術(Implementation Languages and Platforms) 1**

[**8.**](#_heading=h.2s8eyo1) **設計議題(Design Issue) 2**

1. 系統模型與架構(System Model/System Architecture)



本系統主要分為九個模組及一個外部API，分別為登入模組、註冊模組、個人資訊模組、查詢模組、瀏覽模組、刊登模組、請求模組、評價模組、共乘模組以及Google Map API。

在註冊模組中，提供使用者註冊帳號密碼;登入模組中，使用者可以登入註冊好的帳號密碼;在個人資訊模組中，使用者可以更改自己的個人資訊;在查詢模組中，使用者可以依關鍵字找到適合的刊登資訊;在瀏覽模組中，提供所以有刊登資訊供使用者選擇;在刊登模組中，使用者可以刊登自己可以提供的資源或找到所需要的資源;在請求模組中，使用者可以對刊登資訊提出請求申請;在評價模組中，使用者可以給對象一個評價;在共乘模組，會負責已建立好的共乘的通知以及地圖資訊(Google Map API);以上都用到flask、mongoDB。

1. 介面需求與設計(Interface Requirement and Design)

|  |  |  |
| --- | --- | --- |
| 介面名稱(Interface Name) | 介面提供者(Interface Provider) | 介面使用者(Interface Consumer) |
| google map | Google Map API | user |
| 連結方式(Connection Type) | 輸入資料(Input Data) | 輸出資料(Output Data) |
| http post | url: http:[hostip]/Map  data:  {  \_id:"id"  accept\_id:"accept\_id"  \_lat:"lat"  \_lng:"lng"  } | Response:  {  other\_lat:”\_lat”  other\_lng:”\_lng”  } |
| 對應之介面描述 (Interface Description) | | |
| Google Map API 回傳車主和乘客位置。 | | |

|  |  |  |
| --- | --- | --- |
| 介面名稱(Interface Name) | 介面提供者(Interface Provider) | 介面使用者(Interface Consumer) |
| 登入 | server | user |
| 連結方式(Connection Type) | 輸入資料(Input Data) | 輸出資料(Output Data) |
| http post | url: [http://[host-ip]/](about:blank)loginAPI  data:{Account\_name : “帳號”,  \_password:”密碼”} | url\_for('homepage')  data:render\_template(url\_for('login'),fault) |
| 對應之介面描述 (Interface Description) | | |
| 輸入帳號密碼登入。 | | |

|  |  |  |
| --- | --- | --- |
| 介面名稱(Interface Name) | 介面提供者(Interface Provider) | 介面使用者(Interface Consumer) |
| 註冊介面 | server | user |
| 連結方式(Connection Type) | 輸入資料(Input Data) | 輸出資料(Output Data) |
| http post | url:  http:[hostip]/newAccount  data:  {  \_name:"name",  \_phone:"phone number", Account\_name:"account number", \_password:"password", \_email:"email"  } | Response:  render\_template('2-register.html',fault=fault) |
| 對應之介面描述 (Interface Description) | | |
| 填寫資料並傳送到後端儲存，回傳是否成功註冊。 | | |

|  |  |  |
| --- | --- | --- |
| 介面名稱(Interface Name) | 介面提供者(Interface Provider) | 介面使用者(Interface Consumer) |
| 搜尋 | server | user |
| 連結方式(Connection type) | 輸入資料(Input Data) | 輸出資料(Output Data) |
| http post | url:  http:[hostip]/search  data:  {  \_name:”name”,  \_goto:"目的地",  \_getOnTime:"搭乘時間",  \_location:”搭車地點”  } | data:{  post:”object”  } |
| 對應之介面描述 (Interface Description) | | |
| 填寫關鍵字，查詢特定刊登。 | | |

|  |  |  |
| --- | --- | --- |
| 介面名稱(Interface Name) | 介面提供者(Interface Provider) | 介面使用者(Interface Consumer) |
| 刊登 | server | user |
| 連結方式(Connection Type) | 輸入資料(Input Data) | 輸出資料(Output Data) |
| http post | url:  http:[hostip]/pasPost  http:[hostip]/driPost  data:  {  \_goto:"目的地",  \_getOnTime:"搭乘時間",  \_location:”搭車地點”  \_note:”備註”  \_haveraincoat:”bool”  \_haveHat:”bool”  } | Response:  redirect(url\_for('allPost')) |
| 對應之介面描述 (Interface Description) | | |
| 填寫表格發出刊登。 | | |

|  |  |  |
| --- | --- | --- |
| 介面名稱(Interface Name) | 介面提供者(Interface Provider) | 介面使用者(Interface Consumer) |
| 瀏覽 | server | user |
| 連結方式(Connection Type) | 輸入資料(Input Data) | 輸出資料(Output Data) |
| http post | url:  http:[hostip]/postBoard  data:  {  post\_type:"身分",  } | data:{  post:”object”  } |
| 對應之介面描述 (Interface Description) | | |
| 查看所有刊登資訊。 | | |

|  |  |  |
| --- | --- | --- |
| 介面名稱(Interface Name) | 介面提供者(Interface Provider) | 介面使用者(Interface Consumer) |
| 請求狀態 | server | user |
| 連結方式(Connection Type) | 輸入資料(Input Data) | 輸出資料(Output Data) |
| http post | url:  http:[hostip]/passengerResponse  http:[hostip]/driverResponse  data:  {“”} | data:{  request:”object”  } |
| 對應之介面描述 (Interface Description) | | |
| 查看所發出的請求狀態 | | |

|  |  |  |
| --- | --- | --- |
| 介面名稱(Interface Name) | 介面提供者(Interface Provider) | 介面使用者(Interface Consumer) |
| 查看接收請求 | server | user |
| 連結方式(Connection Type) | 輸入資料(Input Data) | 輸出資料(Output Data) |
| http post | url: http:[hostip]/  data:  {  ""  } | response:  {  request”object”  } |
| 對應之介面描述 (Interface Description) | | |
| 查看收到的請求。 | | |

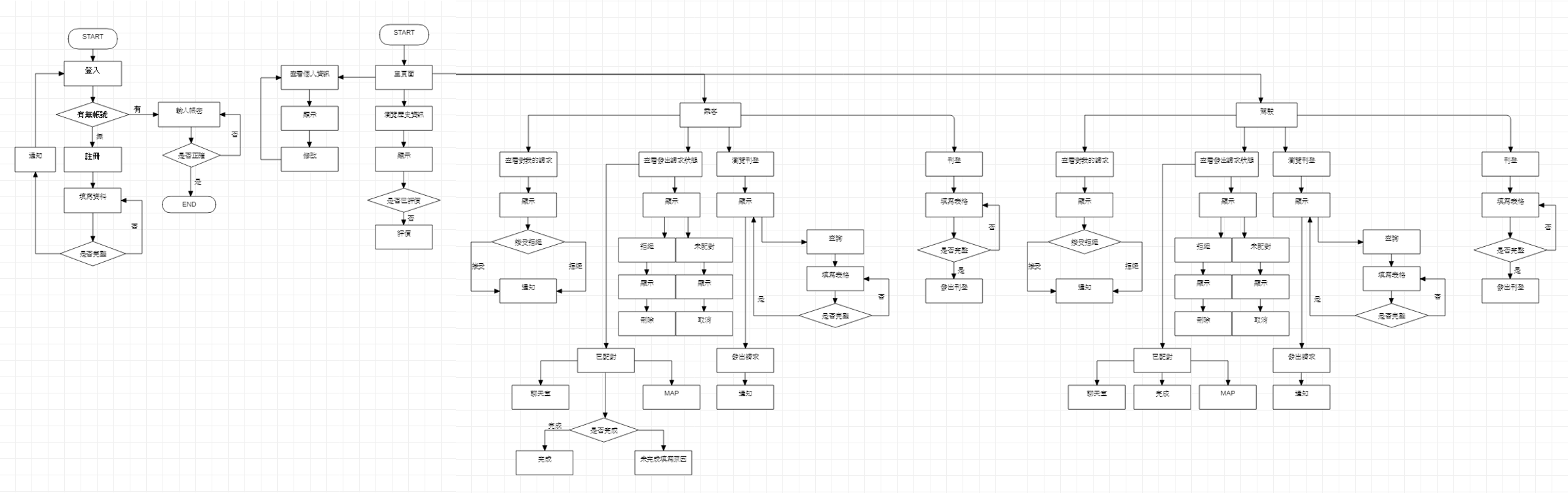
|  |  |  |
| --- | --- | --- |
| 介面名稱(Interface Name) | 介面提供者(Interface Provider) | 介面使用者(Interface Consumer) |
| 歷史資料 | server | user |
| 連結方式(Connection Type) | 輸入資料(Input Data) | 輸出資料(Output Data) |
| http post | url:  http:[hostip]/getHistory  date:  {“"} | {driverName:"name", passengerName:"name",  Location :"post\_location", Goto:"pos\_goto",date:"date",getonTime:"pos\_getOnTime", driver\_id:"\_id", passenger\_id:"id", user\_id:"\_id", notice:"post\_notice",  sendRating:”bool”,  receiveRating:”bool”,  post\_id:”postid”  } |
| 對應之介面描述 (Interface Description) | | |
| 查看所有已配對完成或未完成的行程 | | |

|  |  |  |
| --- | --- | --- |
| 介面名稱(Interface Name) | 介面提供者(Interface Provider) | 介面使用者(Interface Consumer) |
| 評價 | server | user |
| 連結方式(Connection Type) | 輸入資料(Input Data) | 輸出資料(Output Data) |
| http post | url:  http:[hostip]/getRequests  data:  {post\_id"postid",  rate\_range:"幾顆星"  ,rate\_note:"備註"} | url\_for('homepage') |
| 對應之介面描述 (Interface Description) | | |
| 可對另一方填寫評價 | | |

|  |  |  |
| --- | --- | --- |
| 介面名稱(Interface Name) | 介面提供者(Interface Provider) | 介面使用者(Interface Consumer) |
| 個人資訊 | server | user |
| 連結方式(Connection Type) | 輸入資料(Input Data) | 輸出資料(Output Data) |
| http post | url:  http:[hostip]/getinfo  date:  {""} | data:{userinfo:”object”} |
| 對應之介面描述 (Interface Description) | | |
| 查看個人資訊 | | |

|  |  |  |
| --- | --- | --- |
| 介面名稱(Interface Name) | 介面提供者(Interface Provider) | 介面使用者(Interface Consumer) |
| 通知 | server | user |
| 連結方式(Connection Type) | 輸入資料(Input Data) | 輸出資料(Output Data) |
| http post | url:  http:[hostip]/getNotification  date:  {“"} | {  “\_target”:”目標objectid”  “\_type”:”post,requ,rate,system”  “\_msg”:”顯示訊息”  "\_msgTime":”date”  } |
| 對應之介面描述 (Interface Description) | | |
| 根據發生事件發出通知。 | | |

1. 流程設計(Process Design)



<https://drive.google.com/file/d/1iW-vZEM64CDKoXXjriXBi9U0wDE5ubTD/view?usp=sharing>

1. 使用者畫面設計(User Interface Design)

<https://docs.google.com/presentation/d/18coF0JPybIE4v4aPaxP08ZpvHvQQd19d/edit#slide=id.p1>

1. 資料設計(Data Design)

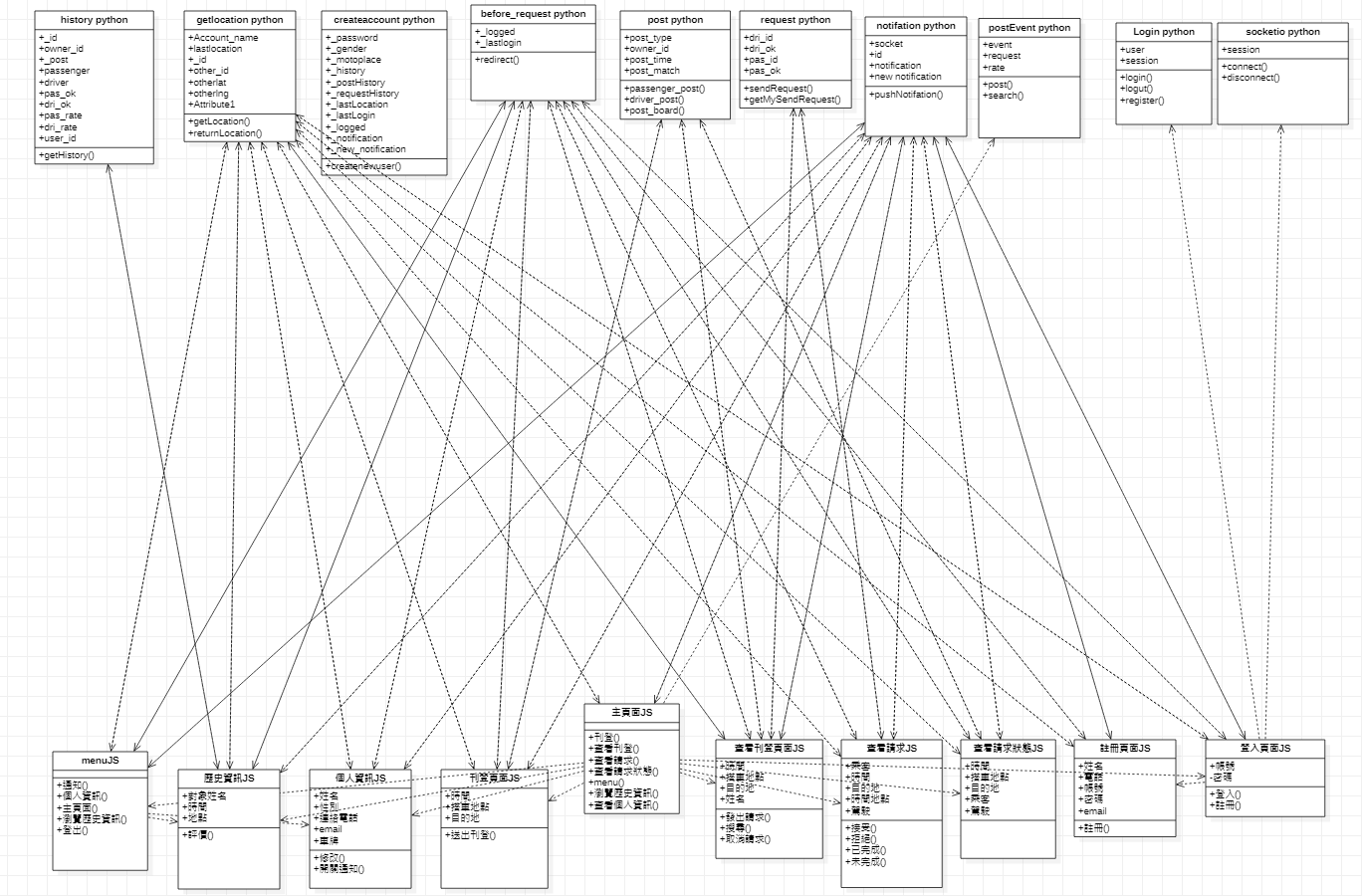








1. 類別圖設計(Class Diagram)



|  |  |
| --- | --- |
| Class Name | 註冊頁面 JS |
| Description | 使用者註冊帳戶資料 |
| Relationship with other classes | 登入頁面，getlocation，notification，before\_request |

|  |  |
| --- | --- |
| Class Name | 登入頁面JS |
| Description | 使用者填寫帳密登入 |
| Relationship with other classes | 主頁面，註冊頁面，socketio，login  ，notification，before\_request，getlocation |

|  |  |
| --- | --- |
| Class Name | 主頁面JS |
| Description | 連到各頁面 |
| Relationship with other classes | 乘客主頁面，駕駛主頁面，瀏覽歷史資訊頁面，查看個人資訊，getlocation，notification，postEvent |

|  |  |
| --- | --- |
| Class Name | Login |
| Description | 檢查登入是否正確 |
| Relationship with other classes | userInfo，pymongo,flask-bcrypt,session |

|  |  |
| --- | --- |
| Class Name | menu |
| Description | 右上角的menu，點開裡面有:  通知、個人資訊、主頁面、瀏覽歷史資訊、設置，按下後可連至該頁面 |
| Relationship with other classes | 通知，個人資訊，主頁面，瀏覽歷史資訊，設置，getlocation，before\_request，notification |

|  |  |
| --- | --- |
| Class Name | 刊登頁面JS |
| Description | 根據自身的需求填寫表格，並刊登出去供別人查看 |
| Relationship with other classes | 主頁面，getlocation，before\_request，post，notification |

|  |  |
| --- | --- |
| Class Name | 查看刊登頁面JS |
| Description | 搜尋後會列出符合條件的刊登資訊 |
| Relationship with other classes | 主頁面，request，getlocation，notification，post，before\_request |

|  |  |
| --- | --- |
| Class Name | 查看請求JS |
| Description | 查看別人對我發出的請求，並可做出回應 |
| Relationship with other classes | request，getlocation，notification，post |

|  |  |
| --- | --- |
| Class Name | 個人資訊JS |
| Description | 顯示或修改個人資訊，顯示評價 |
| Relationship with other classes | getlocation，notification，before\_request |

|  |  |
| --- | --- |
| Class Name | 歷史資訊JS |
| Description | 查看配對完成的資訊 |
| Relationship with other classes | getlocation，notification，before\_request，主畫面，history |

|  |  |
| --- | --- |
| Class Name | 查看請求狀態JS |
| Description | 查看我對別人發出的請求，檢查是否被回應 |
| Relationship with other classes | getlocation，notification，before\_request，notification，主畫面 |

|  |  |
| --- | --- |
| Class Name | notification python |
| Description | 傳送通知 |
| Relationship with other classes | 全部JS頁面 |

|  |  |
| --- | --- |
| Class Name | postEvent python |
| Description | 儲存刊登的清單資訊到資料庫 |
| Relationship with other classes | 無 |

|  |  |
| --- | --- |
| Class Name | socket.io python |
| Description | 每次連線時都需要使用 |
| Relationship with other classes | 無 |

|  |  |
| --- | --- |
| Class Name | request python |
| Description | 發出請求到資料庫，獲取資料查看發給自己的請求 |
| Relationship with other classes | 查看刊登頁面，查看請求 |

|  |  |
| --- | --- |
| Class Name | post python |
| Description | 乘客/駕駛刊登請求到後端資料庫，獲取資料使乘客/駕駛能查看刊登請求 |
| Relationship with other classes | 查看刊登，查看請求，刊登 |

|  |  |
| --- | --- |
| Class Name | before\_request python |
| Description | 要求後端做任何事前，必須先確認是否登入 |
| Relationship with other classes | 全部JS頁面 |

|  |  |
| --- | --- |
| Class Name | create account python |
| Description | 註冊新用戶到資料庫 |
| Relationship with other classes | 註冊頁面(dependency) |

|  |  |
| --- | --- |
| Class Name | get location python |
| Description | 配對成功後，在所有頁面存取用戶位置資訊 |
| Relationship with other classes | 全部頁面 |

|  |  |
| --- | --- |
| Class Name | history python |
| Description | 從後端獲取已配對成功的資訊 |
| Relationship with other classes | 歷史資訊 |

1. 實作方案(Implementation Languages and Platforms)

前端:HTML+CSS+JavaScript+jQuery

後端:Python Flask, MongoDB

1. 設計議題(Design Issue)
2. 使用者的請求狀態與邀請訊息是否不要分乘客跟駕駛的頁面?

可能解決方案:

方案一-分乘客和駕駛:

使用者方便了解現在的身分是駕駛或乘客，比較不會混淆

方案二-不分乘客和駕駛:

單純的顯示各個訊息，由使用者自己判斷

最終選擇方案:方案二

原因: 使用者可能同時擁有兩個身分，使用方案二的話可以不用特別去做切換身分的 動作，就可以一次瀏覽所有訊息，而對於只有一個身分的使用者，因為不會搞混身分，直接顯示的訊息對使用者理應是非常直觀的。

1. 是否要在相似功能的頁面做切換身分的功能?

可能解決方案:

方案一-不做切換身分的功能:

對於只有單一身分的使用者而言不會有影響

方案二-做切換身分的功能:

對於具有雙重身分的使用者較方便

最終選擇方案:方案二

原因: 考慮到具有雙重身分的使用者的可能性，多了切換身分的功能對使用者的使用體驗上有助益

3. 通知應如何實現?

可能解決方案:

方案一-只用alert呈現:

方便簡單好寫

方案二-在右上menu會有一個閃爍的通知顯示，有一個頁面專門顯示所有通知

比較方便管理，可以查看所有通知

最終選擇方案:方案二

原因:alert太過簡單，不方便查詢所有通知，方案二可以了解所有歷史通知、動作

https://ntoumotogo.kangs.idv.tw

<https://docs.google.com/document/d/1toeL4ijfJUcxLoGRXt7vm6gc4yKF9dyE/edit#>