目錄

[目錄 1](#_Toc154415551)

[1. 文件更動記錄 4](#_Toc154415552)

[1. APP 溝通注意事項 9](#_Toc154415553)

[2. webview url 參數說明 11](#_Toc154415554)

[API : 1-1 取得手機Device ID 12](#_Toc154415555)

[API : 1-2 取得常用帳單筆數 13](#_Toc154415556)

[API : 1-3 檢核是否已登入 14](#_Toc154415557)

[API : 1-4 是否已註冊 15](#_Toc154415558)

[API : 1-5 取得登入資訊 16](#_Toc154415559)

[API : 1-6 更改APP個人電話號碼 18](#_Toc154415560)

[API : 1-7 更改APP個人Email 18](#_Toc154415561)

[API : 1-8 通知APP更新個人會員等級 19](#_Toc154415562)

[API : 1-9 取得手機是否有綁定等資訊 20](#_Toc154415563)

[API : 2-1 顯示Bank List Dialog 22](#_Toc154415564)

[API : 2-2 顯示常用帳單列表Dialog 23](#_Toc154415565)

[API : 2-3 顯示可付款的他行信用卡列表 Dialog 25](#_Toc154415566)

[API : 2-4 顯示指定的html Dialog(只有確認) 26](#_Toc154415567)

[API : 2-5 顯示指定的html Dialog(有確認及取消) 27](#_Toc154415568)

[API : 2-6 顯示message Dialog(只有確認) 29](#_Toc154415569)

[API : 2-7 顯示Message Dialog(有確認及取消) 31](#_Toc154415570)

[API : 2-8 顯示支付密碼驗證Dialog 33](#_Toc154415571)

[API : 2-9 顯示扣款工具Dialog 34](#_Toc154415572)

[API : 2-10 呼叫聯絡客服 36](#_Toc154415573)

[API : 2-11 顯示推薦代碼Dialog 37](#_Toc154415574)

[API : 2-12 顯示訊息分享 38](#_Toc154415575)

[API : 3-1 檢核是否已登入，如未登入會顯示登入Dialog 40](#_Toc154415576)

[API : 3-2 登出並回到首頁 41](#_Toc154415577)

[API : 3-3 跳轉申請代扣繳 41](#_Toc154415578)

[API : 3-4 依billID轉跳到指定頁 45](#_Toc154415579)

[API : 3-5 轉跳到註冊頁 46](#_Toc154415580)

[API : 3-6 關閉webview 47](#_Toc154415581)

[API : 3-7 關閉webview並回到首頁 48](#_Toc154415582)

[API : 3-8 關閉webview並轉跳到繳費記錄頁 49](#_Toc154415583)

[API : 4-1 是否顯示App的Header 50](#_Toc154415584)

[API : 4-2 是否顯示App的下方BottomNav 51](#_Toc154415585)

[API : 4-3 設定header樣式(首頁header) 52](#_Toc154415586)

[API : 4-4 設定header樣式(非首頁的header) 53](#_Toc154415587)

[API : 4-5 設定header樣式(非首頁的header並有scan icon) 54](#_Toc154415588)

[API : 4-6 關閉Android back鍵(Android獨有) 54](#_Toc154415589)

[API : 4-7 設定header樣式(非首頁的header並有客戶服務 icon) 56](#_Toc154415590)

[API : 5-1 (iOS)取得Wallet中信用卡狀態 57](#_Toc154415591)

[API : 5-2 (iOS)Apple Pay加卡 59](#_Toc154415592)

[API : 6-1 掃描身分證正面 62](#_Toc154415593)

[API : 6-2 掃描信用卡 63](#_Toc154415594)

[API : 6-3 掃描3碼瓦斯帳單bar code 64](#_Toc154415595)

[API : 6-4 掃描台灣自來水費帳單bar code 65](#_Toc154415596)

[API : 6-5 掃描台北自來水費帳單bar code 66](#_Toc154415597)

[API : 6-6 掃描台電帳單bar code 67](#_Toc154415598)

[API : 6-7 MGM燈箱需求 68](#_Toc154415599)

[API : 6-8登出並回到首頁開啟登入燈箱 69](#_Toc154415600)

[API : 6-9 掃描單一條碼帳單bar code 70](#_Toc154415601)

[Native call Web 72](#_Toc154415602)

[API : 1-1 sendDeviceSize(param:String) 72](#_Toc154415603)

[Universal link(iOS) 73](#_Toc154415604)

[外部網址或APP導向 大咖APP特定功能 73](#_Toc154415605)

[共通連結 75](#_Toc154415606)

[轉址支付 76](#_Toc154415607)

[關貿繳稅轉導 78](#_Toc154415608)

[APPLE PAY 錢包WALLET(iOS獨有) 80](#_Toc154415609)

[財金Web To APP中轉頁 81](#_Toc154415610)

## 1. 文件更動記錄

|  |  |  |  |
| --- | --- | --- | --- |
| 更動日期 | 版號 | 作者 | 更動內容 |
| 2021/06/04 | 0.1 | 王世豪 | 初版 |
| 2021/06/07 | 0.2 | 吳君儀 | 文件格式，文字補充 |
| 2021/06/07 | 0.3 | 范嫺芸 | 資料範例補充 |
| 2021/06/08 | 0.4 | 王世豪 | 新增userAgent補充說明 |
| 2021/06/11 | 0.5 | 王世豪 | 1. 調整1.5 showCreditCardList更名為showPayCreditCardList 2. 新增1.12 logoutToHome API 3. 修改1.19 exitWebToHome API |
| 2021/07/02 | 0.6 | 王世豪 | 1. 新增1-24更改APP個人電話號碼 2. 新增1-25更改APP個人Email 3. 新增1-26通知APP更新個人會員等級 |
| 2021/07/16 | 0.7 | 王世豪 | 1.修改1-17 登入資料欄位 |
| 2021/07/21 | 0.8 | 王世豪 | 1.修改1-2顯示Bank List Dialog可指定是否包含永豐銀行 |
| 2021/07/26 | 0.9 | 王世豪 | 1.新增1-27取得手機是否有綁定的資訊 |
| 2021/07/28 | 1.0 | 范嫺芸 | 1. 1.新增 webview url 參數說明 |
| 2021/07/29 | 1.1 | 吳君儀 | 1.新增1-28 (iOS)取得Wallet中信用卡狀態   1. 2.新增1-29 (iOS)Apple Pay加卡 |
| 2021/07/30 | 1.2 | 王世豪 | 1. 新增1-30掃描身分證正面 2. 新增1-31掃描信用卡卡號 3. 新增1-32掃描瓦斯3碼bar code 4. 新增1-33掃描水費1碼bar code |
| 2021/08/02 | 1.3 | 吳君儀 | 1.更新1-28 (iOS)取得Wallet中信用卡狀態  2.更新1-29 (iOS)Apple Pay加卡 |
| 2021/08/09 | 1.4 | 王世豪 | 1.調整1.23的定義，提供 顯示扣款工具Dialog 的api |
| 2021/08/10 | 1.5 | 王世豪 | 1.調整userAgent的定義  devName-作廢，改由manufacturer & model |
| 2021/08/28 | 1.6 | 王世豪 | 1.調整getloginTypeInfo的回覆資料  2.調整useragent的格式新增osVerName |
| 2021/09/03 | 1.7 | 王世豪 | 1.新增1-35 canBackPress(bool)，用來控制Android的 back鍵是否有作用，在防洗問券可關閉back |
| 2021/09/15 | 1.8 | 王世豪 | 1.1-14 routeByMenuID更改為routeByBillID，可依參數指定轉跳功能  2.新增1-36 showCustomerService()用來顯示客服選單 |
| 2021/09/16 | 1.9 | 王世豪 | 1.1-15 調整規格routeToRegister(id,phone) |
| 2021/09/29 | 2.0 | 王世豪 | 1. 新增initHeaderMenu，控制header的樣式 2. 新增initHeaderBack，控制header的樣式 3. 移除scanBarCodeWater 4. 新增scanBarCodeTaiwanWater，掃描台灣自來水 5. 新增scanBarCodeTaipeiWater，掃描台北自來水 6. 新增scanBarCodeElectic，掃描台電 |
| 2021/10/13 | 2.1 | 王世豪 | 1. 新增1-41 顯示推薦代碼Dialog |
| 2021/10/20 | 2.2 | 王世豪 | 1. 調整項目編號 2. 新增4-5 initHeaderBackWithScan，新的header樣式 |
| 2021/10/27 | 2.3 | 張嘉巖 | * 新增App call webview api   sendDeviceSize(height, width) |
| 2021/11/2 | 2.3 | 范嫺芸 | getLoginData 新增傳入 is\_mma |
| 2021/10/27 | 2.4 | 張嘉巖 | * 文件調整 * (區分 * Web call api, * Native call api) * 增加native call webview api   sendDeviceSize(height, width) |
| 2021/11/11 | 2.5 | 張嘉巖 | * 新增logoutToHomeAndLogin() |
| 2021/11/23 | 2.6 | 張嘉巖 | 3-7關閉webview並回到首頁增加userID參數 |
| 2021/11/24 | 2.7 | 吳君儀 | 1-5 新增DMP mma\_user\_id |
| 2021/11/25 | 2.8 | 張嘉巖 | input -> output |
| 2022/02/15 | 2.9 | 張嘉巖 | 定義新的AP  API : 6-9 掃描單一條碼帳單bar code |
| 2022/02/17 | 3.0 | 張嘉巖 | API: 6-9 掃描單一條碼帳單bar code  更改錯字 showMessage -> showContent |
| 2022/02/23 | 3.1 | 張嘉巖 | 外部網址call 大咖APP |
| 2022/03/18 | 3.2 | 吳君儀 | 更新APPLE PAY 錢包WALLET(iOS) |
| 2022/08/12 | 3.3 | 吳君儀 | 更改APPLE PAY 錢包WALLET(iOS) 連結 |
| 2022/10/6 | 3.4 | 吳君儀 | 更改API:6-9 加入長度和位址條件 |
| 2022/10/17 | 3.5 | 吳君儀 | 更改API:6-9 增加說明 |
| 2022/10/27 | 3.6 | 吳君儀 | 更改外部網址或APP導向 大咖APP特定功能的標題和內容描述  新增共通連結 |
| 2023/01/17 | 3.7 | 吳君儀 | 新增關貿繳稅轉導  更改API:3-3 調整說明，增加代扣繳資料欄位 |
| 2023/01/18 | 3.8 | 吳君儀 | 更改關貿繳稅轉導15類連結 |
| 2023/3/3 | 3.9 | 吳君儀 | 更改API:3-3 調整說明，移除國民年金代扣繳資料欄位 |
| 2023/4/12 | 3.10 | 吳君儀 | 新增API:2-12 顯示訊息分享 |
| 2023/4/21 | 3.11 | 吳君儀 | 更改API:2-12 調整說明 |
| 2023/5/16 | 3.12 | 吳君儀 | 新增API : 4-7 設定header樣式(非首頁的header並有客戶服務 icon) |
| 2023/9/28 | 3.13 | 吳君儀 | 新增財金Web To APP中轉頁 |
| 2023/10/31 | 3.14 | 吳君儀 | 更改財金Web To APP中轉頁 |
| 2023/11/15 | 3.15 | 吳君儀 | 更改財金Web To APP中轉頁-聚合支付業者QRCode |
| 2023/12/25 | 3.16 | 吳君儀 | 更改API:3-4內容  新增Universal link(iOS)內容說明 |
| 2024/2/26 | 3.17 | 吳君儀 | 新增轉址支付參數Param1、Param2 |
| 2024/3/4 | 3.18 | 吳君儀 | 新增 API : 1-2 sendCardData(param:String) |
| 2024/5/13 | 3.19 | 吳君儀 | 調整轉址支付參數URL |

## APP 溝通注意事項

※ 在CaWhoPay APP 中user agent格式如下

"sinopac://CaWhoPay?appVer=${BuildConfig.VERSION\_NAME}&osType=Android&&osVerName=${ Build.VERSION.RELEASE}osVer=${Build.VERSION.SDK\_INT}&manufacturer=${Build.MANUFACTURER}&model=${Build.MODEL}"

appVer->CaWhoPay版本

osType->Android or iOS

osVerName->os的版本名稱(如10.1)

osVer->os的版本(如23)

manufacturer->手機廠商

model->手機型號

devName->手機型號(已作廢，改由manufacturer & model)

※ 判斷裝置是 Android 或 IOS 再決定執行的 function

* Webview call App

Android 可直接呼叫 function，APP.function name(參數)

範例：

APP.getDeviceUUID()

IOS 因使用 WebViewJavascriptBridge，所以需用該套件方式呼叫

範例：

setupWebViewJavascriptBridge((bridge) **=>** {

      bridge.callHandler('getDeviceUUID', '', (response) **=>** {

        return resolve(response);

      });

    });

※ 本文件範例以 Android 為主

* App call Webview

提供給 Android 呼叫

範例：

dialogCallBack(id, action, data) {

    switch (id) {

      case 'bankList': {

*// 轉換獲取的資料*

**const** { code, name } = translateData(data, 1);

*// 0 表示使用者 click 確定*

        if (Number(action) === 0) {

*// do something…*

        }

        break;

      }

      default:

        break;

    }

  }

IOS 同樣需用套件方式呼叫

範例：

**function** setupWebViewJavascriptBridge (callback) {

  if (window.WebViewJavascriptBridge) { return callback(WebViewJavascriptBridge); }

  if (window.WVJBCallbacks) { return window.WVJBCallbacks.push(callback); }

  window.WVJBCallbacks = [callback];

**var** WVJBIframe = document.createElement('iframe');

  WVJBIframe.style.display = 'none';

  WVJBIframe.src = 'https://\_\_bridge\_loaded\_\_';

  document.documentElement.appendChild(WVJBIframe);

  setTimeout(**function** () { document.documentElement.removeChild(WVJBIframe) }, 0)

}

setupWebViewJavascriptBridge((bridge) **=>** {

  bridge.registerHandler("dialogCallBack", **function** (res) {

*// 轉換獲取的資料*

**const** { id, action, data } = JSON.parse(res);

*// 同上方 Android 範例執行 function*

  })

});

## webview url 參數說明

電支註冊頁面帶 cust\_id 為身分證，tele\_no 為電話號碼，直接帶入註冊 input

繳費網址帶 comm\_id 可由此參數查詢帳單進行後續流程，另有 add\_common 0 不加入 ( 常用、待繳 ) 1 加入 ( 掃碼 )

## API : 1-1 取得手機Device ID

定義：getDeviceUUID():String

Input：

無

Output：

Device ID

Sample：

APP.getDeviceUUID()

## API : 1-2 取得常用帳單筆數

定義：getBillSettingSize(menuIDs:String,id:String)

呼叫後會查看App cache是否有常用帳單資訊，如果沒有app會發送取得資料，在一個menuIDs來作過濾得知筆數，callback dataCallBack(id:String,data:String)，所以網頁要準備dataCallBack(id:String,data:String)讓app呼叫。

Input：

menuIDs->使用,分隔可傳入，可傳入需要過濾的常用帳單的menuid

Id->callback用來判斷動作的id

Output：

|  |  |  |
| --- | --- | --- |
| Parameter Name | Parameter Description | Remark |
| id | 呼叫時所帶入的id | 用來判斷動作使用 |
| data | 回覆資料筆數 | “3” ->筆數 |

Sample：

※ 參數為物件會使用 JSON.stringify 先做轉變

APP.getBillSettingSize({

  menuIDs: '2328A3C1-D755-46F6-8C62-5B268B27E221',

  id: 'getBillSettingSize'

})

## API : 1-3 檢核是否已登入

定義：isLogin():Boolean

Input：

無

Output：

Boolean->true已登入 false未登入

Sample：

APP.isLogin()

## API : 1-4 是否已註冊

定義：isReged():Boolean

檢核是否已註冊

Input：

無

Output：

Boolean->true已註冊 false尚未註冊

Sample：( 還未實作該 API，屆時補充 )

## API : 1-5 取得登入資訊

定義：getLoginData():String

取得登入資訊

Input：

無

Output：

Json String，資料如下

{

“token”:”token”, //登入token

“userType”:”會員類別”, //None ,MMA, PayMember

“loginMethod”:”登入方法”, //Password, Graphic, FingerPrint, FaceID

“has\_sinopac\_acct”:boolean, //是否有永豐銀行帳戶

“has\_sinopac\_card”:Boolean, //是否有永豐信用卡

“custID”:”身分證字號”,

“mobile”:”手機號碼”,

“email”:”email”,

“is\_mma”: Boolean , // 是否是 mma

“mma\_user\_id”: String//DMP mma user ID

}

Sample：

APP.getloginData()

## API : 1-6 更改APP個人電話號碼

定義：updateMobile(mobile:String)

在webview中更新了會員的電話號碼，需要呼叫此API更新APP cache的電話號碼。

Input：

mobile->電話號碼 如:0953123456

Output：

無

Sample：

APP.updateMobile(‘0953123456’)

## API : 1-7 更改APP個人Email

定義：updateEmail(email:String)

在webview中更新了會員的email，需要呼叫此API更新APP cache的email。

Input：

email->郵件地址 如:freeman@sinopac.com

Output：

無

Sample：

APP.updateEmail(‘freeman@sinopac.com’)

## API : 1-8 通知APP更新個人會員等級

定義：levelupPayMember()

在webview中更新了會員等級，需要呼叫此API通知APP更新會員等級。

Input：

無

Output：

無

Sample：

APP.levelupPayMember()

## API : 1-9 取得手機是否有綁定等資訊

定義：getLoginTypeInfo()

在webview中取得綁定相關資訊。

Input：

無

Output：

Json String，資料如下

{

“user\_type”:”None”, //空置串或是None代表裝置無綁定,

PayMember綁定電支登入,MMA綁定mma登入

“loginMethod”:”登入方式”, //Password,Graphic,FingerPrint,FaceID,Bio

“cust\_id”:”綁定的身分ID”, //身分證

“user\_acct”:使用者代碼(電支會員的ID),

“is\_acct\_user”:bool, //是否有永豐帳戶

“is\_card\_user”:bool, //是否有永豐信用卡

}

Sample：

APP.getLoginTypeInfo()

## API : 2-1 顯示Bank List Dialog

定義：showBankListDialog(params:String)

呼叫後會顯示各銀行列表資訊，使用者操作後會call back dialogCallBack(id: String, action: Int, data: String)，所以呼叫的網頁需準備dialogCallBack(id: String, action: Int, data: String)。

Input：

Param->是一個Json String內容如下

{“Id”:”id” //callback用來判斷動作的id

“hasSinoac”:true //是否包含永豐銀行}

Output：

|  |  |  |
| --- | --- | --- |
| Parameter Name | Parameter Description | Remark |
| id | 呼叫時所帶入的id | 用來判斷動作使用 |
| action | 動作 | 0選擇成功 |
| data | 選擇的銀行資料  Json format(android)  String with Json format(iOS) | {  “code”:銀行代碼,  “name”:銀行名稱  }  “\”bankCode\”:\”銀行代碼”,\”bankName\”:\”銀行名稱\” |

Sample：

APP.showBanklistDialog('bankList')

dialogCallBack(id, action, data) {

  switch (id) {

    case 'bankList': {

*// 轉換獲取的資料*

**const** { code, name } = translateData(data, 1);

*// 0 表示使用者 click 確定*

      if (Number(action) === 0) {

*// do something…*

      }

      break;

    }

    default:

      break;

  }

}

## API : 2-2 顯示常用帳單列表Dialog

定義：showBillSettingDialog(menuIDs:String,id:String)

呼叫後會顯示依menuIDs過濾常用帳單，再顯示指定類別的常用帳單列表Dialog，使用者操作後會callback dialogCallBack(id: String, action: Int, data: String)，所以呼叫的網頁需準備dialogCallBack(id: String, action: Int, data: String)。

Input：

menuIDs->使用逗號分隔傳入需要過濾的常用帳單的menuid。

Id->callback用來判斷動作的id。

Output：

|  |  |  |
| --- | --- | --- |
| Parameter Name | Parameter Description | Remark |
| id | 呼叫時所帶入的id | 用來判斷動作使用 |
| action | 動作 | 0選擇成功 |
| data | 選擇的常用帳單資料  Json format(android)  String with Json format(iOS) | {  "menu\_id":"884d208f-6f2b-4376-87ec-237cc05efcad",  "bill\_id":*null*,  "parent\_bill\_id":"f10dedd1-40cc-4c85-a0f4-d24dafa17168",  "comm\_acct\_type":*null*,  "comm\_acct\_name":*null*,  "comm\_id":"4465ddb8-15e7-4c4c-937d-61d96e9170bc",  "comm\_type":0,  "comm\_name":"我的台電電費",  "bill\_name":"台電電費",  "bill\_catagory\_id":"f10dedd1-40cc-4c85-a0f4-d24dafa17168",  "bill\_catagory\_name":"水電瓦斯費",  "comm\_info":{  "taipower\_info":{  "power\_id":"12345678901"  }  },  "comm\_acct\_id":*null*,  "comm\_acct\_info":*null*,  "notify\_info":*null*,  "comm\_bills":[  {  "property\_name":"power\_id",  "display\_name":"電號",  "value":"12345678901",  "property\_type":"PowerId"  }  ]  } |
|  |  |  |

Sample：

※ 參數為物件會使用 JSON.stringify 先做轉變

APP.showBillSettingDialog({

  menuIDs: '2328A3C1-D755-46F6-8C62-5B268B27E221',

  id: 'getCommonBillInfo'

})

## API : 2-3 顯示可付款的他行信用卡列表 Dialog

定義：showPayCreditCardList(id:String)

呼叫後會顯示可以被繳信用卡費的信用卡列表Dialog，使用者操作後會call back dialogCallBack(id: String, action: Int, data: String)，所以呼叫的網頁需準備dialogCallBack(id: String, action: Int, data: String)。

Input：

Id->callback用來判斷動作的id

Output：

|  |  |  |
| --- | --- | --- |
| Parameter Name | Parameter Description | Remark |
| id | 呼叫時所帶入的id | 用來判斷動作使用 |
| action | 動作 | 0選擇成功 |
| data | 選擇的可繳信用卡費資料  Json format(android)  String with Json format(iOS) | {  “menuID”:menu id,  “bankCode”:銀行code,  “bankName”:銀行名稱,  “fromMessage”:註記說明  } |

Sample：

APP.showPayCreditCardList('credCardList')

## API : 2-4 顯示指定的html Dialog(只有確認)

定義：showHtmlDialog(param:String)

呼叫後會顯示Dialog，Dialog中的內容依指定的資料呈現，使用者操作後會call back dialogCallBack(id: String, action: Int, data: String)，所以呼叫的網頁需準備dialogCallBack(id: String, action: Int, data: String)。

Input：

Param->是一個Json String內容如下

{

“Id”:”id” //callback用來判斷動作的id

“title”:title //dialog title文字

“url”:”http://google” //網頁位置，與htmlStr擇一提供

“htmlStr”:”” //html字串，與url擇一提供

“btnStr”:’確定’ //確認按鈕的字串

“data”:”data” //任一字串資料，會在callback時帶回

}

Output：

|  |  |  |
| --- | --- | --- |
| Parameter Name | Parameter Description | Remark |
| id | 呼叫時所帶入的id | 用來判斷動作使用 |
| action | 動作 | 0選擇成功 |
| data | 傳入的data String |  |

## API : 2-5 顯示指定的html Dialog(有確認及取消)

定義：showHtmlSelDialog(param:String)

呼叫後會顯示Dialog，Dialog中的內容依指定的資料呈現，使用者操作後會call back dialogCallBack(id: String, action: Int, data: String)，所以呼叫的網頁需準備dialogCallBack(id: String, action: Int, data: String)。

Input：

Param->是一個Json String內容如下

{

“Id”:”id” //callback用來判斷動作的id

“title”:title //dialog title文字

“url”:”http://google” //網頁位置，與htmlStr擇一提供

“htmlStr”:”” //html字串，與url擇一提供

“btnNO”:’取消’ //取消按鈕的字串

“btnOK”:’確定’ //確認按鈕的字串

“data”:”data” //任一字串資料，會在callback時帶回

}

Output：

|  |  |  |
| --- | --- | --- |
| Parameter Name | Parameter Description | Remark |
| id | 呼叫時所帶入的id | 用來判斷動作使用 |
| action | 動作 | 0選擇成功  1選擇取消 |
| data | 傳入的data String |  |

Sample：

## API : 2-6 顯示message Dialog(只有確認)

定義：showMsgDialog(param:String)

呼叫後會顯示Dialog，Dialog中的內容依指定的資料呈現，使用者操作後會call back dialogCallBack(id: String, action: Int, data: String)，所以呼叫的網頁需準備dialogCallBack(id: String, action: Int, data: String)。

Input：

Param->是一個Json String內容如下

{

“Id”:”id” //callback用來判斷動作的id

“title”:title //dialog title文字

“msg”:”message” //文字訊息

“btnStr”:’確定’ //確認按鈕的字串

“data”:”data” //任一字串資料，會在callback時帶回

}

Output：

|  |  |  |
| --- | --- | --- |
| Parameter Name | Parameter Description | Remark |
| id | 呼叫時所帶入的id | 用來判斷動作使用 |
| action | 動作 | 0選擇成功 |
| data | 傳入的data String |  |

Sample：

※ 參數為物件會使用 JSON.stringify 先做轉變

APP.showMsgDialog({

  id: 'addCommonBillSuccess',

  title: '已加入常用帳單',

  msg: '已儲存「中華電信1」銷帳編號',

  btnOK: '確 定',

  data: {}

})

## API : 2-7 顯示Message Dialog(有確認及取消)

定義：showMsgSelDialog(param:String)

呼叫後會顯示Dialog，Dialog中的內容依指定的資料呈現，使用者操作後會call back dialogCallBack(id: String, action: Int, data: String)，所以呼叫的網頁需準備dialogCallBack(id: String, action: Int, data: String)。

Input：

Param->是一個Json String內容如下

{

“id”:”id” //callback用來判斷動作的id

“title”:title //dialog title文字

“msg”:”文字訊息” //文字訊息

“btnNO”:’取消’ //取消按鈕的字串

“btnOK”:’確定’ //確認按鈕的字串

“data”:”data” //任一字串資料，會在callback時帶回

}

Output：

|  |  |  |
| --- | --- | --- |
| Parameter Name | Parameter Description | Remark |
| id | 呼叫時所帶入的id | 用來判斷動作使用 |
| action | 動作 | 0選擇成功  1選擇取消 |
| data | 傳入的data String |  |

Sample：

※ 參數為物件會使用 JSON.stringify 先做轉變

APP.showMsgSelDialog({

  id: 'addCommonBill',

  title: 'Oops!',

  msg: '我知道你很懶，但我比你更懶，加入我們吧!!',

  btnNO: '先繳再說',

  btnOK: '立即註冊',

  data: {}

})

## API : 2-8 顯示支付密碼驗證Dialog

定義：showPayVertify()

呼叫後會顯示支付密碼驗證的dialog，驗證成功會callback payVertifyPass()，

呼叫的網頁需準備payVertifyPass()來得知驗證成功。

Input：

無

Output：

無

Sample：

## API : 2-9 顯示扣款工具Dialog

定義：showPayTool(params:String)

呼叫後會顯示Dialog，Dialog中的內容依指定的資料呈現，使用者操作後會call back dialogCallBack(id: String, action: Int, data: String)，所以呼叫的網頁需準備dialogCallBack(id: String, action: Int, data: String)。

Input：

Param->是一個Json String內容如下

{

“Id”:”id” //callback用來判斷動作的id

“title”:title //dialog title文字

“datas”:[{參考品叡paytool的Json物件}] //扣款工具項目

“isShowBtn”:’true’ //自行輸入按鈕是否顯示

}

Output：

|  |  |  |
| --- | --- | --- |
| Parameter Name | Parameter Description | Remark |
| id | 呼叫時所帶入的id | 用來判斷動作使用 |
| action | 動作 | 0選擇成功  1選擇自行輸入 |
| data | 傳入的data String | action為0=>品叡的paytool Json物件  action為1=>空字串 |

Sample：

## API : 2-10 呼叫聯絡客服

定義：showCustomerService()

呼叫後會顯示聯絡客服的選單

Input：

Output：

Sample：

APP.showCustomerService()

## API : 2-11 顯示推薦代碼Dialog

定義：showPromoteCodeDialog(id:String)

呼叫後會顯示推薦代碼的Dialog，使用者操作後會call back dialogCallBack(id: String, action: Int, data: String)，所以呼叫的網頁需準備dialogCallBack(id: String, action: Int, data: String)。

Input：

id //callback用來判斷動作的id

Output：

|  |  |  |
| --- | --- | --- |
| Parameter Name | Parameter Description | Remark |
| id | 呼叫時所帶入的id | 用來判斷動作使用 |
| action | 動作 | 0確定 -1取消 |
| data | 傳入的data String | {  “code”:”推薦碼”,  “isCheck”:bool //是否選擇不再顯示  } |

Sample：( 目前還未使用該 API )

APP.showPromoteCodeDialog()

## API : 2-12 顯示訊息分享

定義：showMessageShareActivity(param:String)

呼叫後會顯示該手機作業系統的分享介面，介面上方帶有從api傳入的文字訊息。

Input：

Param->是一個Json String內容如下

{

“msg”:”文字訊息” //文字訊息

}

Sample：

※ 參數為物件會使用 JSON.stringify 先做轉變

APP. showMessageShareActivity ({

  msg: '我知道你很懶，但我比你更懶，加入我們吧!!',

})

## API : 3-1 檢核是否已登入，如未登入會顯示登入Dialog

定義：checkLogin():Boolean

Input：

無

Output：

Boolean->true已登入 false未登入

Sample：

APP.checkLogin()

## API : 3-2 登出並回到首頁

定義：logoutToHome()

Input：

無

Output：

無

Sample：

APP.logoutToHome()

## API : 3-3 跳轉申請代扣繳

定義：routeAutoPaySetting(param:String)

呼叫後會跳轉到申請代扣繳流程的委託人資料頁，使用者於該頁面確認後點選下一步，會將web提供的資料帶入對應申請資訊的欄位。功能包含水費、電信費、路邊停車費、健保費與國民年金。

Input：

Param->是一個Json String內容如下

{

“billId”: " 2540c0db-197b-47e5-8269-1f27e307aaf6" //主階層bill\_id，如路邊停車費

“subBillId”:”10b5a719-9d03-4478-bd3f-35a28324921e”//子階層bill\_id，如桃園市路邊停車費

“primaryData”:{“ telecom\_no”:” 213215454545”} //前端串接會用到的主欄位物件資料：水號“water\_no”、電信費用戶號碼(電話號碼) “ telecom\_no”、 路邊停車費車牌號碼”car\_no”和車種” vehicle\_type”以及健保費與國民年金的被保險人身份證字號”idn\_ban”。瓦斯費為null。若為路邊停車費，vehicle\_type為汽車**C**或機車**M**，車牌號碼如FK-8809、NFK-6238，依照中間**-**符號拆成兩部分填進文字欄位。

~~“branch”:”1234”//電信費營運處代碼~~

~~“insName”:”中文姓名”// ins\_name國民年金的被保險人姓名~~

~~“insBirth”:”19900401”//ins\_birth國民年金的出生年月日~~

“closeWeb”:true //是否關閉目前的webview

}

Output：

無

Sample：

//水費

APP. routeAutoPaySetting ('{

“billId”: " 2540c0db-197b-47e5-8269-1f27e307aaf6"

“subBillId”:”10b5a719-9d03-4478-bd3f-35a28324921e”

“primaryData”:{ “water\_no”:”213215454545”}

“closeWeb”:true

}')

//路邊停車費

APP. routeAutoPaySetting ('{

“billId”: " 2540c0db-197b-47e5-8269-1f27e307aaf6"

“subBillId”:” 10b5a719-9d03-4478-bd3f-35a28324921e”

“primaryData”:{ “idn\_ban”: “A112345678”,

“vehicle\_type”: “C”,

“car\_no”: “FK-8809”}

“closeWeb”:true

}')

//健保費與國民年金

APP. routeAutoPaySetting ('{

“billId”: " f026090f-9400-4901-acbb-a0747f4ba6c8"

“subBillId”:” 4611c7de-a567-423e-ba85-adda9b0cfdfd”

“primaryData”:{ “idn\_ban”: “A112345678”}

“closeWeb”:true

}')

## API : 3-4 依billID轉跳到指定頁

定義：routebyBillID(param:String)

依選單功能billID跳轉到指定的頁面。如為非選單功能的billID，可以帶入繳費稅頁(**native-home-pay**)和提醒通知設定(**native-notification-setting**)。

Input：

Param->是一個Json String內容如下

{

“billID”:”xxxxxxxxxxxx” //指定功能的billID

“closeWeb”:true //是否關閉目前的webview

}

Output：

無

Sample：

APP.routeByBillID('{“billID”:”xxxxx”,”closeWeb”:false}')

## API : 3-5 轉跳到註冊頁

定義：routeToRegister(id:String,phone:String)

轉跳到註冊頁面

Input：

Id:身分證

Phone:電話

Output：

無

Sample：( 還未實作該 API，屆時補充 )

## API : 3-6 關閉webview

定義：exitWeb()

關閉webview

Input：

無

Output：

無

Sample：

APP.exitWeb()

## API : 3-7 關閉webview並回到首頁

定義：exitWebToHome(param:String)

關閉webview並回到首頁，並依param的內容來決定首頁的動作。

Input：

Param->是一個Json String內容如下

{

“needLogin”:ture //用來指定回到首頁後是否要顯示登入dialog(如果已登入，給true也不會顯示)

“needQuickLogin”:true //到了首頁登入後是否顯示快速登入設定詢問

“userID”: “FXXXXXXXXX” //若有ID需傳送則傳送, 若無ID請給空字串2021/11/23新增(**注意: Android and iOS 使用此userID只能在無綁定狀態下使用**)

}

Output：

無

## API : 3-8 關閉webview並轉跳到繳費記錄頁

定義：exitWebToPayDetail(needLogin:boolean)

關閉webview並回到繳費記錄頁，並依needLogin來決定到繳費記錄頁後是否要顯示登入dialog。

Input：

needLogin->用來指定到繳費記錄頁後是否要顯示登入dialog(如果已登入，給true也不會顯示)

Output：

無

Sample：

## API : 4-1 是否顯示App的Header

定義：showHeader(isShow:boolean)

在webview中是否要呈現header(在iOS中是指top navigation bar)。

Input：

isShow->是否呈header ,true呈現 false不呈現

Output：

無

Sample：

APP.showHeader(true)

## API : 4-2 是否顯示App的下方BottomNav

定義：showBottomNav(isShow:boolean)

在webview中是否要呈現下方的Bottom navigation(在ios指的是tab bar controller)。

Input：

isShow->是否呈下方bottom navigation,true呈現 false不呈現

Output：

無

Sample：

APP.showBottomNav(true)

## API : 4-3 設定header樣式(首頁header)

定義：initHeaderMenu(title:String)

呼叫後會顯示首頁樣式的header並顯示指定的title文字。

Input：

title->用來顯示在header上的文字

Output：

Sample：

APP.sinitHeaderMenu('menu title')

## API : 4-4 設定header樣式(非首頁的header)

定義：initHeaderBack(title:String)

呼叫後會顯示一般樣式的header，左上方icon是back，並顯示指定的title文字。

Input：

title->用來顯示在header上的文字

Output：

Sample：

APP.sinitHeaderBack('back title')

## API : 4-5 設定header樣式(非首頁的header並有scan icon)

定義：initHeaderBackWithScan(title:String)

呼叫後會顯示一般樣式的header，左上方icon是back，右上方是QRCode scan並顯示指定的title文字。

Input：

title->用來顯示在header上的文字

Output：

Sample：

APP.sinitHeaderBackWithScan('back title')

## API : 4-6 關閉Android back鍵(Android獨有)

定義：canBackPress(canBackPress:bool)

呼叫後會依指定的布林值啟動或關閉Android的back鍵

Input：

canBackPress ->false關閉back鍵

Output：

Sample：

APP.canBackPress(false)

## API : 4-7 設定header樣式(非首頁的header並有客戶服務 icon)

定義：initHeaderBackWithCustomerService(title:String)

呼叫後會顯示一般樣式的header，左上方icon是back，右上方是客戶服務 icon並顯示指定的title文字。

Input：

title->用來顯示在header上的文字

Output：

Sample：

APP.sinitHeaderBackWithCustomerService ('back customer')

## API : 5-1 (iOS)取得Wallet中信用卡狀態

定義：getPassStatus(cardNoSuffixs:String):String

Webview呼叫iOS Native，依傳入cardNoSuffix(信用卡卡號後四碼)取得使用者的Wallet(手機錢包)中該信用卡的狀態和資料。若無資料則回覆空字串。

Input：

cardNoSuffixs->使用者所有信用卡卡號後四碼，以逗號區隔。

Ex : ”1234,4567,6789”

Output：

Json String，資料如下

[{

“PANS”:”信用卡後4碼”,

“DANS”:”虛擬卡後4碼”,

“device”:”加卡裝置來源”, //iPhone, Apple Watch,

“state”:”該卡狀態”//已啟動卡片(activated):**0**, 被發行者啟動卡片(requiresActivation):**1**,正在啟動卡片(activating):**2**, 使用者或發行者取消卡片或無法使用(suspended):**3**, 發行者撤銷卡片(deactivated):**4**

},{…}]

Sample：

setupWebViewJavascriptBridge((bridge) **=>** {

      bridge.callHandler(' getPassStatus ', '', (response) **=>** {

        return resolve(response);

      });

## API : 5-2 (iOS)Apple Pay加卡

定義：applePayAddCard(param:String)

使用者點選Webview加卡按鈕，Webview呼叫Native帶入該信用卡的Json， iOS Native 進行初始化Apple Pay加卡流程，如系統異常則初始化失敗。

使用者指定該信用卡Apple Pay加卡流程操作結束後Native會callback addCardFinishCallBack(action: Int, data: String)，回覆加卡流程成功或失敗，成功則回覆action與該卡狀態資料param，失敗僅回覆action，所以呼叫的Webview需準備addCardFinishCallBack (action: Int, data:String)。

Input：

param->是一個Json String內容如下

{

“cardHolderName”:”持卡人姓名”,

“cardName”:”卡片名稱”,

“cardNo”:”信用卡號”

}

Output：

Boolean->true初始化成功 false初始化失敗

Web Sample：

setupWebViewJavascriptBridge((bridge) **=>** {

      bridge.callHandler(' applePayAddCard ', '', (response) **=>** {

        return resolve(response);

      });

APP Sample：

[\_bridge callHandler:@”addCardFinishCallBack”data:”action:0,data:{}”]

Web Sample：

bridge.registerHandler("addCardFinishCallBack", **function** (res) {

*// 轉換獲取的資料*

**const** { id, action, data } = JSON.parse(res);

  })

|  |  |  |
| --- | --- | --- |
| Parameter Name | Parameter Description | Remark |
| action | 動作 | 0:加卡成功  1:加卡失敗 |
| data | 傳入的data String | {  “PANS”:”信用卡後4碼”,  “DANS”:”虛擬卡後4碼”,  “device”:”加卡裝置來源”, //iPhone, Apple Watch,  “state”:”該卡狀態”//已啟動卡片(activated):**0**, 被發行者啟動卡片(requiresActivation):**1**,正在啟動卡片(activating):**2**, 使用者或發行者取消卡片或無法使用(suspended):**3**, 發行者撤銷卡片(deactivated):**4**  } |

## API : 6-1 掃描身分證正面

定義：scanPersonID(id:String)

呼叫後會顯示掃描畫面，使用者操作後會call back dataCallBack(id: String,data: String)，所以呼叫的網頁需準備dataCallBack(id: String,data: String)。

Input：

Id->callback用來判斷動作的id

Output：

dataCallBack的資料會以Json String回覆，格是如下

{

“name”:”姓名”,

“birthday\_roc”:”yyyMMDD”,(0710101)

“issurance\_date\_roc”:”yyyMMDD”,(0940701) //發證日期

“issurance\_city”:”北市” //換證城市

“issurance\_type”:1 //換補發類型 0:初發 1:補發 2:換發

}

Sample：

APP.scanPersonID('personID')

## API : 6-2 掃描信用卡

定義：scanCreditCard(id:String)

呼叫後會顯示掃描畫面，使用者操作後會call back dataCallBack(id: String,data: String)，所以呼叫的網頁需準備dataCallBack(id: String,data: String)。

Input：

Id->callback用來判斷動作的id

Output：

dataCallBack的資料會以Json String回覆，格是如下

{

“card\_no”:”信用卡號碼”,

“holder\_name”:”test”, //持卡人姓名

“expire\_date”:”1024”// 卡片校期MMyy

}

Sample：

APP.scanCreditCard('personID')

## API : 6-3 掃描3碼瓦斯帳單bar code

定義：scanBarCodeGas(id:String)

呼叫後會顯示掃描畫面，使用者操作後會call back dataCallBack(id: String,data: String)，所以呼叫的網頁需準備dataCallBack(id: String,data: String)。

Input：

Id->callback用來判斷動作的id

Output：

dataCallBack的資料會以Json String回覆，格是如下

{

“code1”:”9碼字串”,

“code2”:”16碼字串”,

“code3”:”15碼字串”

}

Sample：

APP.scanBarCodeGas('gas')

## API : 6-4 掃描台灣自來水費帳單bar code

定義：scanBarCodeTaiwanWater(id:String)

呼叫後會顯示掃描畫面，使用者操作後會call back dataCallBack(id: String,data: String)，所以呼叫的網頁需準備dataCallBack(id: String,data: String)。

Input：

Id->callback用來判斷動作的id

Output：

dataCallBack的資料會以Json String回覆，格是如下

{

“code”:”11碼水號字串”

}

Sample：

APP.scanBarCodeTaiwanWater('water')

## API : 6-5 掃描台北自來水費帳單bar code

定義：scanBarCodeTaipeiWater(id:String)

呼叫後會顯示掃描畫面，使用者操作後會call back dataCallBack(id: String,data: String)，所以呼叫的網頁需準備dataCallBack(id: String,data: String)。

Input：

Id->callback用來判斷動作的id

Output：

dataCallBack的資料會以Json String回覆，格是如下

{

“code”:”10碼水號字串”

}

Sample：

APP.scanBarCodeTaipeiWater('water')

## API : 6-6 掃描台電帳單bar code

定義：scanBarCodeElectic(id:String)

呼叫後會顯示掃描畫面，使用者操作後會call back dataCallBack(id: String,data: String)，所以呼叫的網頁需準備dataCallBack(id: String,data: String)。

Input：

Id->callback用來判斷動作的id

Output：

dataCallBack的資料會以Json String回覆，格是如下

{

“code”:”11碼電號字串”

}

Sample：

APP.scanBarCodeTaipeiWater('electic')

## API : 6-7 MGM燈箱需求

定義：getMgmStauts(param:String)

依參數跳起燈箱。

Input：

Param->是一個Json String內容如下

{

“mgm\_show”:true // true開啟燈箱, false 不開啟燈箱

}

Output：

無

Sample：

APP.getMgmStatus ('{“mgm\_show”:false}')

## API : 6-8登出並回到首頁開啟登入燈箱

定義：logoutToHomeAndLogin()

Input：

無

Output：

無

Sample：

APP.logoutToHomeAndLogin()

## API : 6-9 掃描單一條碼帳單bar code

定義：scanBarCodeJustOne(param:String)

呼叫後會顯示掃描畫面，相機掃描後讀到符合長度的條碼會轉成字串，並以要求的起始位址、結束位址擷取，最後再call back dataCallBack(id: String,data: String)，所以呼叫的網頁需準備dataCallBack(id: String,data: String)。

如果相機元件有找到傳入的條碼長度barcodeLength符合的bar code，但是提供的位址無法進行擷取(如: barcodeStartIndex或barcodeEndIndex小於0、barcodeStartIndex大於等於barcodeEndIndex、barcodeStartIndex或barcodeEndIndex大於barcodeLength)，dataCallBack 中data json string的code會直接回覆該bar code完整字串而不做擷取。



Input：

Param->是一個Json String內容如下

{

“id”:”xxxx”, // Id->callback用來判斷動作的id

“showContent”:”I am done!”,// showContent -> 單一條碼帳單顯示bar code下方的文字

“barcodeLength”:”17”,// barcodeLength->要尋找的barcode總長度

“barcodeStartIndex”:”4”,// barcodeStartIndex->barcode擷取起始位址

“barcodeEndIndex”:”16”// barcodeEndIndex->barcode擷取結束位址

}

Output：

dataCallBack的data資料會以Json String回覆，格式如下

{

“code”:“回傳符合長度和已擷取的barcode”

}

Sample：

APP. scanBarCodeJustOne (“{

"id": "scanSingleBarCode",

"showContent": "I'm done!!",

" barcodeLength ": "17",

" barcodeStartIndex ": "4",

" barcodeEndIndex ": "16"

}”)

# Native call Web

## API : 1-1 sendDeviceSize(param:String)

定義：sendDeviceSize(param:String)

output：

Param->是一個Json String內容如下

{

“height”:1234, //int 格式

“width”:1234 //int 格式

}

Output：

無

Sample：

APP.sendDeviceSize ('{“height”:123, “width”:1234}')

## API : 1-2 sendCardData(param:String)

定義：sendCardData (param:String)，app 原生傳送來自推播、提醒通知與信用卡消費明細等卡片資料給web前端，web前端需有對應方法接收卡片資料，可參考API 1-1。同時app原生在呼叫後須把暫存卡片資料清除。

output：

Param->是一個Json String內容如下

{

“CardNo”: “XXXXXXXXXXXXXXXX” // 格式String

}

Sample：

APP.sendCardData ('{“CardNo”:”XXXXXXXXXXXXXXXX”}')

# Universal link(iOS)

iOS轉導將下節**dacard://** 開頭的連結直接改成[**https://paybill.sinopac.com/**](https://paybill.sinopac.com/)**...(**後面參數不變)，避免原scheme URL碰到iOS APP未安裝時，出現錯誤訊息。新的Universal link當iOS APP未安裝時，會導到繳費網官網。

測試與正式環境連結相同。

# 外部網址或APP導向 大咖APP特定功能

參數概覽表格:

|  |  |  |  |
| --- | --- | --- | --- |
| 參數 | 轉址支付 | 共通連結 | 關貿繳稅轉導 |
| queryType | TypeA | TypeB | TypeC |
| BillId | 66c66bd5-44b5-4672-ba06-77ee4de34771 | 22077410-327e-4261-9032-eeca43ae3e87 | 207c0d5d-bd26-4fa3-b48e-0c56ab99577b |
| QUERY\_TYPE | B |  |  |
| QUERY\_DATA1 | 110831315 |  |  |
| QUERY\_DATA2 | 3772473169582550 |  |  |
| QUERY\_DATA3 | 10066Y00000558 |  |  |
| QUERY\_DATA4 |  |  |  |
| QUERY\_DATA5 |  |  |  |
| SN | BEB202202100145056 |  |  |
| PAYMENT\_TYPE |  |  | 15001 |
| TAX |  |  | irxp |
| SEQ\_NO |  |  | 1070062569 |
| TXN\_AMOUNT |  |  | 499 |

## 共通連結

Android and iOS 使用相同邏輯之url。

* 外部web或APP呼叫需要在dacard//:後增加sinopacaction做為domain、增加queryType參數TypeB(可擴充定義)再接BillId，並組成如下url，就可以呼叫大咖APP開啟BillId指定功能：

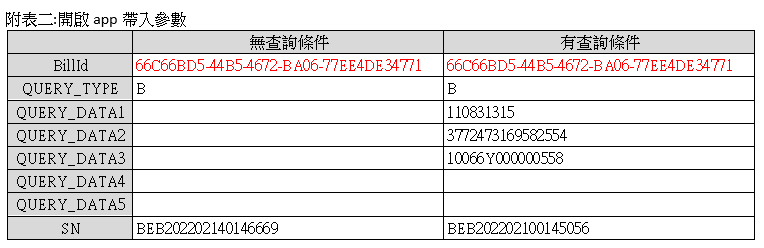
dacard://sinopacaction?queryType=TypeB&BillId=bd17f299-522e-4414-a257-4beddba1fd14

|  |  |  |
| --- | --- | --- |
| key | value | Query參數 |
| queryType | TypeB | BillId |

BillId可參考paybill api /getcawhopayconfigs [GA01] menu\_info各功能之bill\_id。

## 轉址支付

使用共通連結相同邏輯之url擴充，附表二為轉址支付的參數。



* 外部web呼叫之轉址支付queryType參數為TypeA，後接附表二資訊，並組成如下url，就可以呼叫大咖APP開導指定功能帶入連結資訊予APP內嵌web：

dacard://sinopacaction?queryType=TypeA&BillId=bd17f299-522e-4414-a257-4beddba1fd14&QUERY\_TYPE=B&QUERY\_DATA1=110831315&QUERY\_DATA2=3772473169582554&QUERY\_DATA3=10066Y000000558&QUERY\_DATA4=&QUERY\_DATA5=&SN=BEB202202100145056&URL=XXX…

TypeA代表需帶參數有：BillId, QUERY\_TYPE, QUERY\_DATA1, QUERY\_DATA2, QUERY\_DATA3, QUERY\_DATA4, QUERY\_DATA5, SN

|  |  |  |
| --- | --- | --- |
| key | value | Query參數 |
| queryType | TypeA | BillId  QUERY\_TYPE、QUERY\_DATA1、QUERY\_DATA2、  QUERY\_DATA3、QUERY\_DATA4、QUERY\_DATA5、SN 、URL |

以上為webview處理方式

* App處理方式如下
  + 大咖APP接收後須依照queryType作相對應的資料處理，導頁採用BillId，如果有帶入URL，就需要先導頁至WebView功能設定該URL。

queryType

* + - TypeA
      * 需要Query的資料為  
        BillId, QUERY\_TYPE, QUERY\_DATA1, QUERY\_DATA2, QUERY\_DATA3, QUERY\_DATA4, QUERY\_DATA5, SN, URL

## 關貿繳稅轉導

使用共通連結之url擴充，從關貿網站經過中介頁引導至APP進行財金PayTax繳稅服務。

* 外部web呼叫之繳稅轉導queryType參數為TypeC，BillId須為稅費，後接參數，並組成如下url：

dacard://sinopacaction?queryType=TypeC&BillId=207c0d5d-bd26-4fa3-b48e-0c56ab99577b&PAYMENT\_TYPE=15001&TAX=irxp&SEQ\_NO=1070062569&TXN\_AMOUNT=499

TypeC代表需帶參數有：BillId, PAYMENT\_TYPE,TAX, SEQ\_NO, TXN\_AMOUNT

|  |  |  |
| --- | --- | --- |
| key | value | Query參數 |
| queryType | TypeC | BillId  PAYMENT\_TYPE,TAX, SEQ\_NO  TXN\_AMOUNT |

* App處理方式如下
  + 大咖APP接收後須依照queryType作相對應的資料處理
    - * 驗證TypeC，其需要Query的資料為:BillId, PAYMENT\_TYPE,TAX,SEQ\_NO,TXN\_AMOUNT
      * TAX判斷出為關貿來源且APP登入綁定為MMA使用者，以提供之參數(PAYMENT\_TYPE和TXN\_AMOUNT)組成PayTax連結:
        + 15類(金額左邊補零到十位數，前面沒有+):

測試:https://paytest.fisc.com.tw/QRCODE.aspx?par=15001+(10位繳款金額)&TAX=xxxx&SEQ\_NO=xxxxxxxxxx

舉例: https://paytest.fisc.com.tw/QRCODE.aspx?par=150010000000499&TAX=irxp&SEQ\_NO=1070062569

正式: https://paytax.nat.gov.tw/QRCODE.aspx?par=15001+(10位繳款金額)&TAX=xxxx&SEQ\_NO=xxxxxxxxxx

* + - * + (目前沒有)其他類

測試: https://paytest.fisc.com.tw/QRCODE.aspx?par= + (5位繳款類別) + (16位銷帳編號)+ (10位繳款金額) + (6繳納截止日)+ (5位期別代號)+ (6位識別碼)，共48字元。

正式: https://paytax.nat.gov.tw/QRCODE.aspx?par= + (5位繳款類別) + (16位銷帳編號)+ (10位繳款金額) + (6繳納截止日)+ (5位期別代號)+ (6位識別碼)，共48字元。

* + - * 依照原有掃碼繳費流程帶入PayTax連結呼叫API，最後導引至原生財金PayTax繳稅頁面。
      * 後續關貿資料驗證API須將來源Query資料(TAX, SEQ\_NO)帶入Request，請參考**串接關貿API規格文件**。

## APPLE PAY 錢包WALLET(iOS獨有)

* APPLE 錢包WALLET APP呼叫(schemeURL)需要在dacard//:後增加sinopacaction做為domain、增加queryType參數wallet與BillId，並組成如下url，就可以呼叫大咖APP：

dacard://sinopacaction**/**queryType=wallet&BillId=f5ae6e8a-5b67-4bfa-8f76-c156531d4246

|  |  |  |
| --- | --- | --- |
| key | value | Query參數 |
| queryType | wallet | BillId |

APPLE要求連結URL不能有問號特殊符號，queryType前為斜線。

* App處理方式如下
  + 大咖APP接收後須依照queryType作相對應的資料處理

queryType:wallet

* + - * 需要Query的資料為BillId

## 財金Web To APP中轉頁

使用財金規格書Web To APP交易規格2.2的URL Scheme，從特店經過TWQRP中轉頁引導至大咖APP進行財金Taiwan Pay消費或繳費服務。

* URL Scheme以twqrp-加上本行代號三碼(807)與銀行錢包序號兩碼(01)，如下範例：

twqrp-**80701**://twqrp/webtoapp?acqBank=006&terminalId=10011001&merchantId=006263198001001&encQRCode=AcKSrTAGls19%2fnKMPQeUZKjT1dFK4L5LZJzxyZe55iHK%2bF41WknrdelQIVugMPK5dn%2b8HckdwMUzSGEWyrZej7NrAmkAc7W6VFYvnJYd5%2btgPvPRZkvtDlJTgOtke%2bvmYTWjbbs9uYZALZyNTWJsIggbTe1lzlTiQMBOAB5fR41DlDmQt%2fKCQwNpSiAggCx

zijVIiJdJC265CUMH8zJMld5FGX857J2Mah6Q04cJkyS2Bgsontpsw1J6UmOu4ZzawpR6Za6spfpFhX3H0aJNQxeDQSsfnuUatWynSUH3J0Iil3fgWgViScbqLqDsJUiW0SyDA3WhLKbsrjGkYEBZpY

3iPmgBL7q5aKtVJTvgDK%2blwczJjfvYihpYc5j75E3oYQ%3d%3d&encRetURL=AXBC%2bDwt

xAD%2b1svPy75UCILT4tiSEFOKDPPGhmK0xIdBmqLsnRgKZCqw4%2f%2fNTe9OEPp0%2fmx

O6PZoAthsSmn7yrqAWGVTSCl7Ju3%2baEWQHb8X&orderNumber=QRP1234567890&verifyCode=947676EC47B460307036BB5BF13959C7D8F6BBC61EC980A3CD8F995F8163C83A

URL格式:

|  |  |
| --- | --- |
| scheme | twqrp-80701 |
| host | twqrp |
| path | webtoapp |
| query | acqBank=006&terminalId=10011001&merchantId=006263198001001&encQRCode=AcKSrTAGls19%2fnKMPQeUZKjT1dFK4L5LZJzxyZe55iHK%2bF41WknrdelQIVugMPK5dn%2b8HckdwMUzSGEWyrZej7NrAmkAc7W6VFYvnJYd5%2btgPvPRZkvtDlJTgOtke%2bvmYTWjbbs9uYZALZyNTWJsIggbTe1lzlTiQMBOAB5fR41DlDmQt%2fKCQwNpSiAggCx  zijVIiJdJC265CUMH8zJMld5FGX857J2Mah6Q04cJkyS2Bgsontpsw1J6UmOu4ZzawpR6Za6spfpFhX3H0aJNQxeDQSsfnuUatWynSUH3J0Iil3fgWgViScbqLqDsJUiW0SyDA3WhLKbsrjGkYEBZpY  3iPmgBL7q5aKtVJTvgDK%2blwczJjfvYihpYc5j75E3oYQ%3d%3d&encRetURL=AXBC%2bDwt  xAD%2b1svPy75UCILT4tiSEFOKDPPGhmK0xIdBmqLsnRgKZCqw4%2f%2fNTe9OEPp0%2fmx  O6PZoAthsSmn7yrqAWGVTSCl7Ju3%2baEWQHb8X&orderNumber=QRP1234567890&verifyCode=947676EC47B460307036BB5BF13959C7D8F6BBC61EC980A3CD8F995F8163C83A |

* App處理方式如下
  + 大咖APP接收到Scheme URL後須確認Scheme URL開頭符合財金中轉頁Scheme URL格式:**twqrp-80701://twqrp/webtoapp**，再將整段Scheme URL送到錢包server後端。
  + 錢包server後端驗證、解析與解密Scheme URL的query欄位(如encQRCode)，將結果送回給前端，包含encQRCode解密與驗證(CheckQRCode電文)後會得到的交易相關資料(如金額、日期等等)。
  + 前端操作消費或繳費完成至結果頁，收到後端回傳結果欄位url，兩秒後使用收到的url外開，url通常為來源特店web或app。
  + 其他詳細資訊可參考:台灣Pay-中轉頁PRD\_V1.4.docx、金融資訊系統QR Code規格書-Web To APP交易規格.pdf、WebToApp(錢包)測試個案v1.3.xlsx。
* 聚合支付業者之QRCode說明:
  + 前面流程與其他中轉頁不同，大咖APP掃聚合支付業者QRCode後，將url傳送給server，server會判斷他是不是在白名單內，如果不是，server回傳給APP顯示錯誤訊息。
  + 如果是，server將url後面加上”?twp=80701” (AggregationUrl)並回傳給APP，APP以"qrcode\_type":"AGC"判斷url為應外開之聚合支付業者商店url。  
    API GW00、GT03 response json data example:

{

"status":"S",

"return\_code":"",

"return\_desc":"",

"data":{

"qrcode\_type":"AGC",

…

"AggregationUrl":https://iqrc.dev.epay365.com.tw/calc/pay/d81da439-0791-4d2e-82c8-460d31a6a6c2?twp=80701

}

}  


* + 依循業者商店介面操作後會再開導回大咖APP，後續如同其他中轉頁流程處理。
  + 目前白名單業者雷門、英特拉。  
    測試環境QRCode:  
    
  + 參考: 金融資訊系統QR Code規格書-Web To APP交易規格.pdf (p.33) 