



MyPlay iOS SDK

介面規格書

Doc Prepared by	主旨：My Play iOS 介面規格書			第 1 頁，共 13 頁
	Date:2012.11.26	Rev	File	
		1.1	TWM -iOS SDK.doc	



修訂履歷(Modification History)

Date	Author	Version	Description
2012/11/26	Henry Dai	1.1	新增註冊 Method
2012/11/19	Henry Dai	1.0	新增 Method
2012/11/13	Henry Dai Weil Chang	0.9	初版

Doc Prepared by	主旨：My Play iOS 介面規格書			第 2 頁，共 13 頁	
	Date:2012.11.26	Rev	File		
		1.1	TWM -iOS SDK.doc		



目錄

1.	目的.....	4
2.	範圍.....	4
3.	SDK 的 API 使用說明.....	4
3.1	事前準備	4
2.2	安裝 SDK	5
3.3	登入	5
3.4	IN-APP PURCHASE	7
3.5	SDK REFERENCE.....	9

Doc Prepared by	主旨：My Play iOS 介面規格書			第 3 頁，共 13 頁
	Date:2012.11.26	Rev	File	
		1.1	TWM -iOS SDK.doc	

1. 目的

MyPlay 服務為台灣大哥大推出的線上遊戲平台，提供 Android/iOS 裝置的用戶，下載相關的程式和遊戲。用戶可以使用 MyPlayApps 進行遊戲的瀏覽、購買、下載與意見回饋等等活動。此文件描述遊戲平台系統與手機遊戲 APP 溝通的 SDK，此 SDK 將交付與手機開發商以重新包裝 TWM 專屬的遊戲 APP。其功能有會員認證，金流介接的功能等。

對遊戲軟體開發者來說，My Play iOS SDK 服務幫助 iOS App 開發人員使用 MyPlay 平台金流與會員系統。

2. 範圍

以下分項描述目前 SDK 與遠端服務介面所提供的功能項目簡介：

- 登入 SDK

玩家下載並進入 App 後，需先進行登入動作。需提供登入的 SDK，以利 App 開發商將包進遊戲中，進行認證登入。

- 金流-消費紀錄 SDK

手持 iOS 系統裝置的玩家可以透過本平台進行付費 SDK 以及付費後將消費紀錄記錄於系統的 SDK。

3.SDK 的 API 使用說明

3.1 事前準備

1. 下載 MyPlay Framework for iOS

2. 提交 App 之 Bundle ID 做上架之用。Bundle ID 設定原則為: *com.twm.遊戲名稱縮寫(英文)*

注意：此 SDK 僅適用於 iOS 5.0 或以上版本

Doc Prepared by	主旨：My Play iOS 介面規格書			第 4 頁，共 13 頁	
	Date:2012.11.26	Rev	File		
		1.1	TWM -iOS SDK.doc		



3.2 安裝 SDK

1. 將 MyPlayFramework.framework 加入至專案 Libraries 中並設定為 Require
2. 加入 StoreKit.framework 至專案 Libraries 中並設定為 Require
3. 將壓縮檔中的 MPCONFIG.plist 檔案放置到專案中。此檔案為設定必要檔案，請勿擅自修改內容。

3.3 登入

1. Import MyPlayFramework 中的 MPFramework.h，引入 MyPlay Framework。
2. 於初始頁面 ViewController 的 header 檔中，設定其需實作 UIWebViewDelegate Protocol。
3. 宣告並初始一 CGRect 個體，提供登入畫面區塊設定使用。iPhone App 建議使用為手機全版畫面。
4. 於同一區塊中，宣告並初始一 UIWebView 個體，並將上步驟宣告之 CGRect 設定帶入其初始值。
5. 初始一 MPLogin 個體並指定給步驟二宣告之變數，初始方式如下：
initWithWebView: (UIWebView) forState: (NSString*)*
Webview 帶入步驟四初始之 WebView，forState 則帶入註冊或是登入狀態，以連接到不同的頁面。其中相關參數為：
 - MPUserLogin: 使用者登入
 - MPUserSignUp: 使用者註冊
6. 設定步驟 5 之 WebView 的 delegate 為 self。
7. 將步驟 5 之 WebView 加入到本身之 Subview。
8. 於初始頁面 ViewController 的 m 檔中，實作以下方法：
(void) webViewDidStartLoad
(void) webViewDidFinishLoad
(BOOL) webview: (UIWebView)webview shouldStartLoadWithRequest: navigationType*
(void) webview: (UIWebView)webview didFailLoadWithError:*
9. 於上步驟實作之方法中填入步驟二宣告之 MPLogin 相對應方法：
 - *webViewDidStartLoad*
 - *webViewDidFinishLoad*
 - *shouldStartLoadWithRequest: (NSURLRequest*) navigationType: (UIWebViewNavigationType)*
 - *didFailLoadWithError: (NSError)*

Doc Prepared by	主旨：My Play iOS 介面規格書			第 5 頁，共 13 頁
	Date:2012.11.26	Rev	File	
		1.1	TWM -iOS SDK.doc	

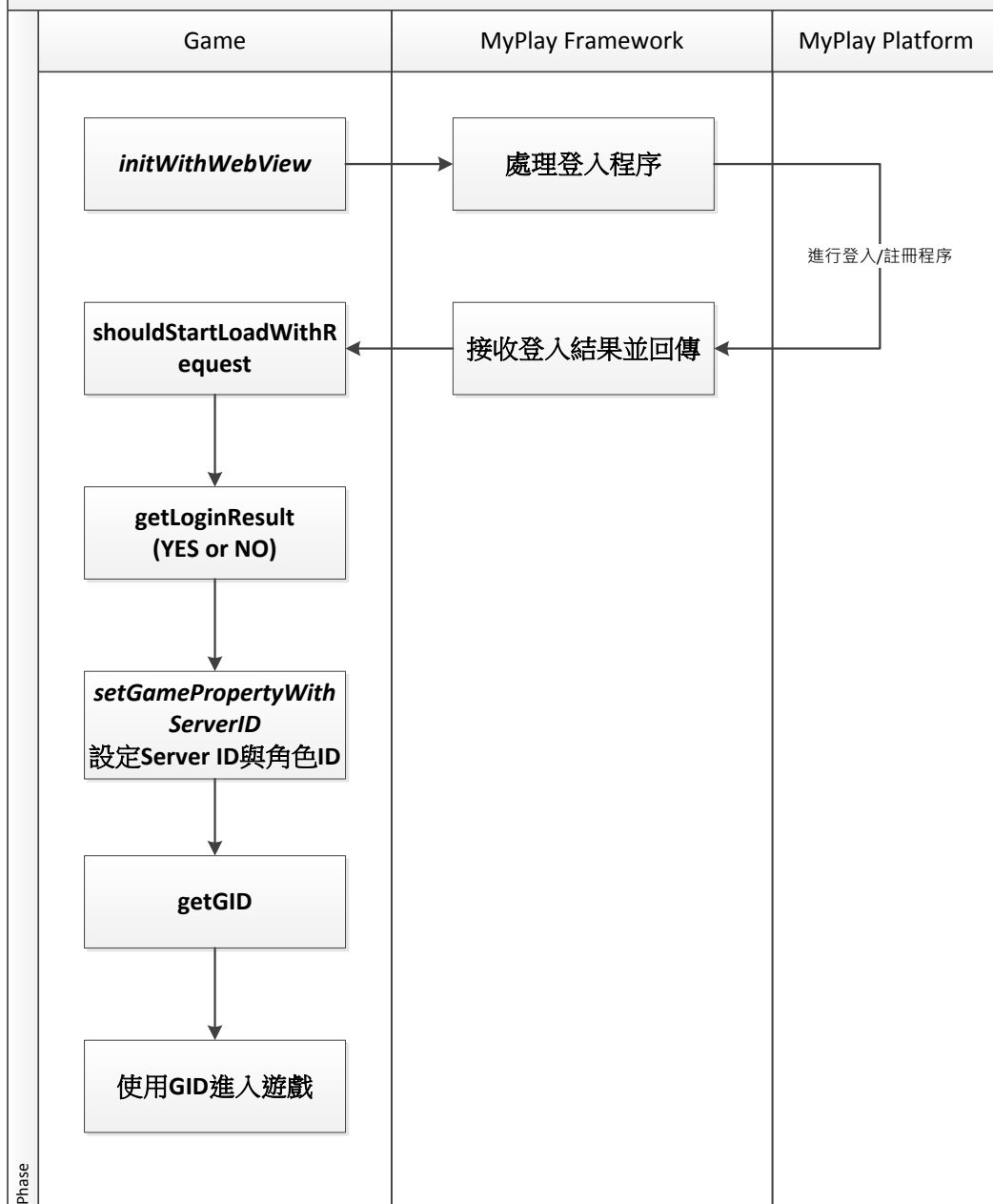
10. 於 *shouldStartLoadWithRequest* 中呼叫 *getLoginResult* 方法取得登入結果並設計登入後行為。
11. 呼叫 *setGamePropertyWithServerID:(NSString*) Role:(NSString*)* 並傳入伺服器 ID 與使用者角色 ID。
12. 使用 *getGID* 取得玩家 GID 進行遊戲登入。

登入運作流程：

Doc Prepared by	主旨：My Play iOS 介面規格書			第 6 頁，共 13 頁	
	Date:2012.11.26	Rev	File		
		1.1	TWM -iOS SDK.doc		



iOS 登入流程



3.4 In-App Purchase

1. Import *MyPlayFramework* 中的 *MPFramework.h*，引入 MyPlay Framework
2. 於欲實作購買之頁面中，header 檔設定其實作 *SKProductsRequestDelegate* 與 *MPPurchaseResult*。
3. 於購滿頁面的 m 檔中，宣告一全域之 *MPPurchaseItem* 之變數(public or private)，並且設定好

Doc Prepared by	主旨：My Play iOS 介面規格書			第 7 頁，共 13 頁
	Date:2012.11.26	Rev	File	
		1.1	TWM -iOS SDK.doc	



變數之 **property** 與 **synthesize**。

4. 於此程式進入點呼叫 **MPPurchaseItem** 內之 **setMPPurchaseResultGetter: (id<MPPurchaseResult>)** 將 **self** 帶入其參數。
5. 於欲實作購買功能之方法中，初始化 **MGPurchaseItem** 物件，並呼叫以下方法：
-(NSString*) sendPurchaseRequestWithDelegate: (id< SKProductsRequestDelegate >)

若購買物品有分屬不同狀況時，請使用以下方法：

-(NSString*) sendPurchaseRequestWithDelegate: (id< SKProductsRequestDelegate >) inSituation: (NSString*)

- **delegate**: 傳入實作 **SKProductsRequestDelegate** 之物件
- **inSituation**: 傳入設定好之狀況名稱 (Optional)

6. 於 **m** 檔中，實作 **SKProductsRequestDelegate** 中方法：
-(void) productRequest: (SKProductRequest*) didReceiveResponse: (SKProductResponse*)

於以上方法中設定一 **Array** 接收由 **MGPurchaseItem** 內之方法傳回內容：

-(NSArray*) productRequest: (SKProductRequest*) didReceiveResponse: (SKProductResponse*)

此方法用於接收產品清單資訊，傳回之 **Array** 內含一產品清單 **Array** 相對應的內容為：

- **MPIItemTitle**: 標題
- **MPIItemDescription**: 產品說明
- **MPIItemPrice**: 金額
- **MPIItemProductID**: 產品 ID

並同時實作以下方法：

-(void) getProductStatus: (NSString*) withError: (NSString*)

此方法用於接收取得產品清單之例外狀況，如連線逾時等等，並做出相對應之動作。其中 **Status** 傳回之內容為：

- **MPProductStatusSuccess**: 取得產品清單成功。請注意，此方法被呼叫時會自動導入 **productRequest** 取得產品內容，除非有特殊狀況，建議不實作此條件之任何內容。
 - **MPProductStatusTimeOut**: 連線逾時
 - **MPProductStatusEmpty**: 空白產品清單
 - **MPProductStatusError**: 發生錯誤，若需要可由 **Error** 變數取得相關之錯誤訊息
- 請依照以上狀況分別實作不同之回應方式。

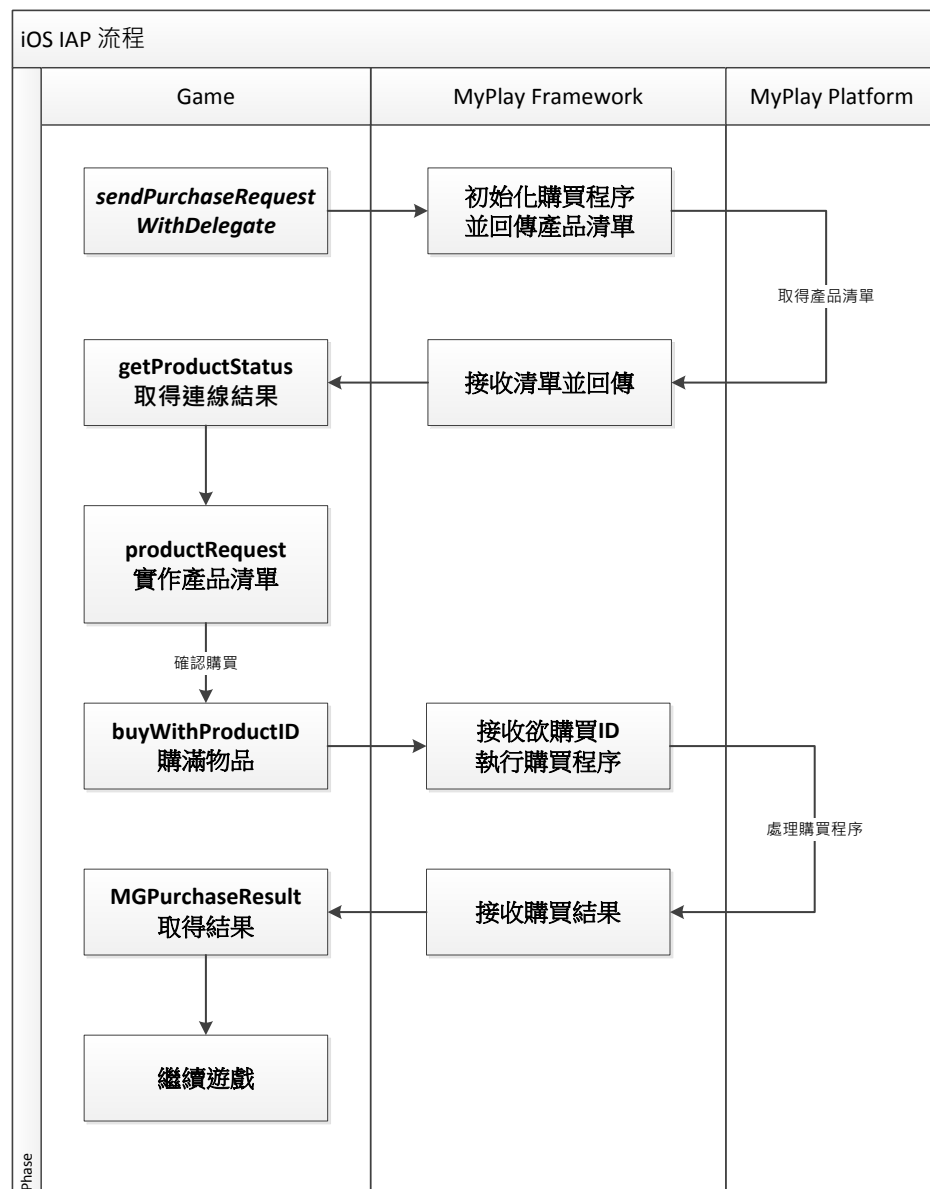
7. 實作 **MGPurchaseResult** 內方法：
 - **purchaseCompleted**
在此方法內實作購買成功時行為。
 - **purchaseFailed**

Doc Prepared by	主旨：My Play iOS 介面規格書			第 8 頁，共 13 頁
	Date:2012.11.26	Rev	File	
		1.1	TWM -iOS SDK.doc	



在此方法內實作購買失敗時行為，可使用 `err` 回傳錯誤訊息。

IAP 運作流程：



3.5 SDK Reference

MPLogin	
Methods:	
(NSString*) getGID	
方法說明:	
取得玩家的 GID。	

Doc Prepared by	主旨：My Play iOS 介面規格書			第 9 頁，共 13 頁
	Date:2012.11.26	Rev	File	
		1.1	TWM -iOS SDK.doc	



傳入參數:

無

回傳:

回傳玩家的 GID。

(id) initWithWebView: (UIWebView*) forState: (NSString*)

方法說明:

初始化 MPLogin。

傳入參數:

使用於登入介面之用的 WebView 及相關狀態參數。

其中相關參數為:

- MPUserLogin: 使用者登入
- MPUserSignUp: 使用者註冊

回傳:

MPLogin 實體

(void) setGamePropertyWithServerID:(NSString*) Role: (NSString*)

方法說明:

設定 ServerID 與 RoleID。

傳入參數:

ServerID 與 RoleID。

回傳:

無

(void) webViewDidStartLoad

方法說明:

網頁開始讀取之運算處理。

傳入參數:

無

回傳:

無

(void) webViewDidFinishLoad

方法說明:

網頁結束讀取之運算處理。

傳入參數:

無

回傳:

無

Doc Prepared by	主旨：My Play iOS 介面規格書			第 10 頁，共 13 頁
	Date:2012.11.26	Rev	File	
		1.1	TWM -iOS SDK.doc	



(BOOL) shouldStartLoadWithRequest: navigationType

方法說明:

網頁動作讀取與相對應之運算處理。

傳入參數:

實作 `UIWebViewDelegate` 時，同方法帶入之參數

回傳:

型態 `Boolean`，決定是否執行網頁動作

(void) didFailLoadWithError: (NSError*)

方法說明:

網頁結束錯誤之處理。

傳入參數:

實作 `UIWebViewDelegate` 時，同方法帶入之參數

回傳:

無

(BOOL) getLoginResult

方法說明:

取得登入結果。

傳入參數:

無

回傳:

型態 `Boolean`，判斷是否登入成功。

MPPurchaseItem

Methods:

(void) setMPPurchaseResultGetter: (id<MPPurchaseResult>)

方法說明:

設定 In-App Purchase 之結果接收代理。

傳入參數:

`MPPurchaseResult` 之實作實體

回傳:

無

(BOOL) sendPurchaseRequestWithDelegate: (id< SKProductsRequestDelegate >)

方法說明:

傳送取得產品清單要求。

傳入參數:

Doc Prepared by	主旨：My Play iOS 介面規格書			第 11 頁，共 13 頁
	Date:2012.11.26	Rev	File	
		1.1	TWM -iOS SDK.doc	



SKProductsRequestDelegate 之實作實體

回傳:

Boolean, 是否開通 In-App Purchase

**(BOOL)sendPurchaseRequestWithGameID: (id< SKProductsRequestDelegate >)
inSituation: (NSString*)**

方法說明:

傳送取得產品清單要求。

傳入參數:

SKProductsRequestDelegate 之實作實體與可購買物品之情況(Optional)

回傳:

Boolean, 是否開通 In-App Purchase

**(NSArray*) productRequest: (SKProductRequest*) didReceiveResponse:
(SKProductResponse*)**

方法說明:

取得產品清單。

傳入參數:

實作 SKProductsRequestDelegate 時，同方法帶入之參數

回傳:

NSArray 表示回傳之產品清單，Array 內容為一 NSArray，陣列內容為:

- MPlItemTitle: 標題
- MPlItemDescription: 產品說明
- MPlItemPrice: 金額
- MPlItemProductID: 產品 ID

(void) buyWithProductID: (NSString*)

方法說明:

購買指定產品。

傳入參數:

欲購買產品之 Product ID

回傳:

無。

MPPurchaseResult

Required Protocol:

(void) purchaseCompleted

Doc Prepared by	主旨：My Play iOS 介面規格書			第 12 頁，共 13 頁
	Date:2012.11.26	Rev	File	
		1.1	TWM -iOS SDK.doc	



方法說明:

購買成功 Protocol。

使用參數:

無

(void) purchaseFailed: (NSError*) err

方法說明:

購買失敗 Protocol。

使用參數:

無

- (void) getProductStatus: (NSString*) status withError: (NSString*) description

方法說明:

取得產品清單狀況。

使用參數:

Status 表取得清單狀況，詳情如下:

- **MPProductStatusSuccess:** 取得產品清單成功
- **MPProductStatusTimeOut:** 連線逾時
- **MPProductStatusEmpty:** 空白產品清單
- **MPProductStatusError:** 發生錯誤，若需要可由 Error 變數取得相關之錯誤訊息

Doc Prepared by	主旨：My Play iOS 介面規格書			第 13 頁，共 13 頁
	Date:2012.11.26	Rev	File	
		1.1	TWM -iOS SDK.doc	