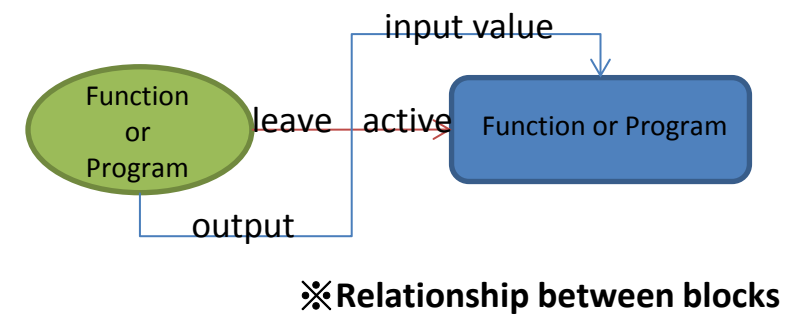
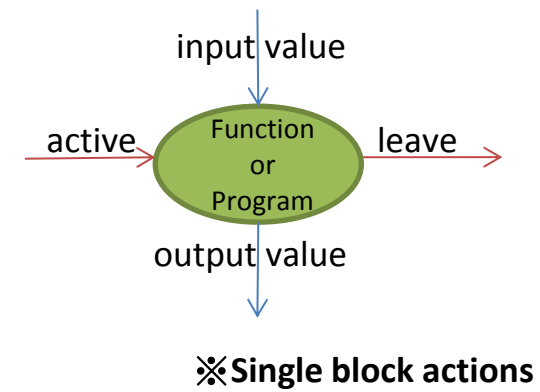
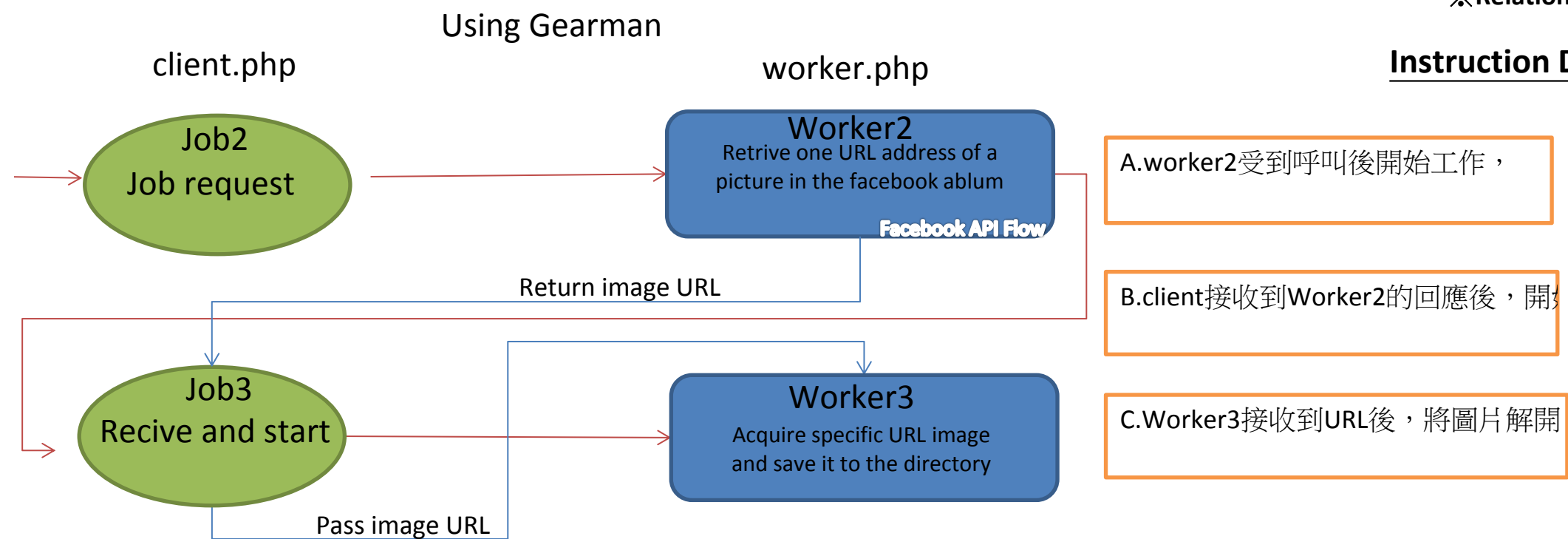


Facebook API sequence of saving my pic

※Please read the Instruction Diagram for the logic of diagrams

1.System working flow



Instruction Diagram

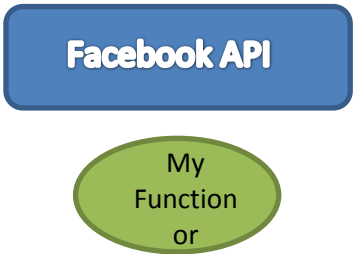
A.worker2受到呼叫後開始工作，

B.client接收到Worker2的回應後，開始

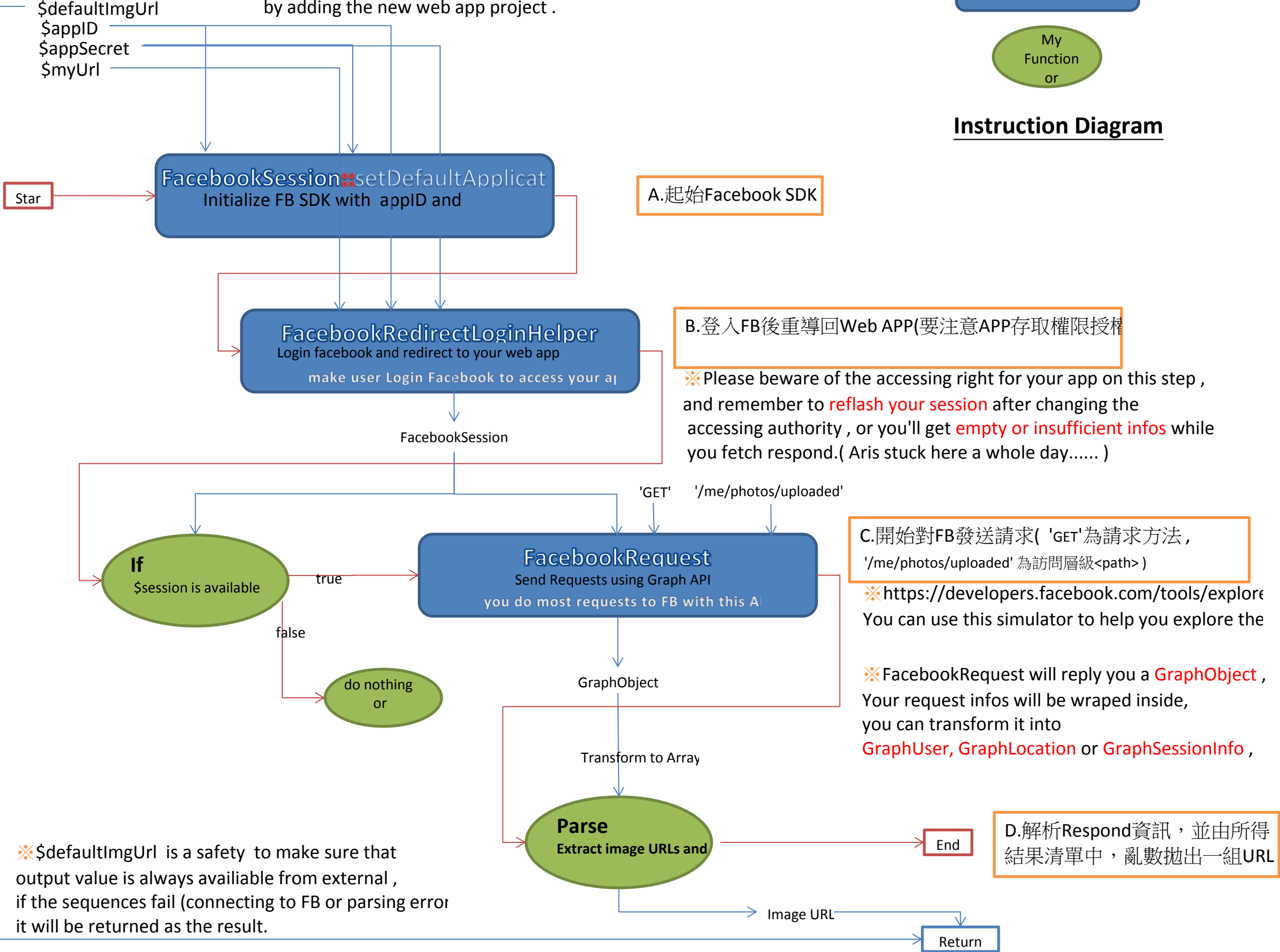
C.Worker3接收到URL後，將圖片解開

2.Facebook API Flow

✂Add your facebook account to facebook developers before you getting start, and you'll get **appId** and **appSecret** from the official website by adding the new web app project .

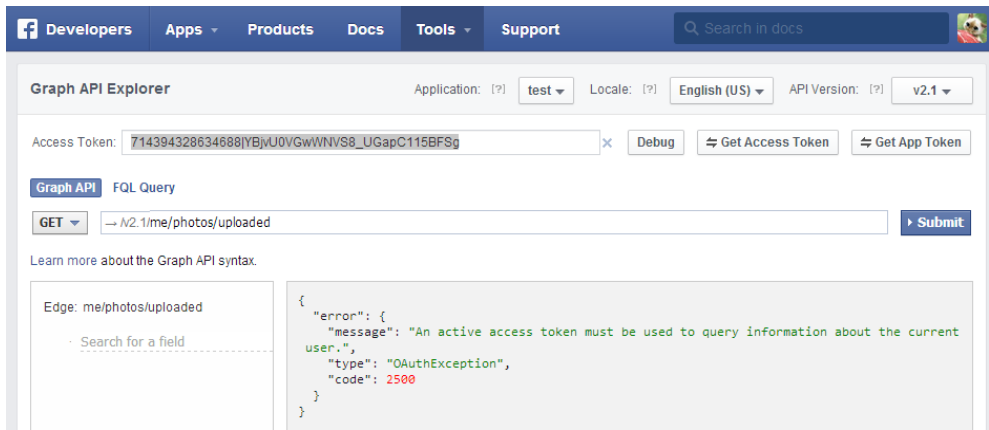


Instruction Diagram



3.Pros & Cons

1.Facebook SDK有版本的分別，舊版已無法運作(4.0之前的版本)，API的使用方式也完全不同，最好先看看官方的資訊跟教學，這點在獵捕範例的時候要留意(別只看範例不看官方文件!!)，不然容易費時打轉，



<Graph API >
新版SDK導入Graph API，Facebook上的記錄活動以及連結動作大部分都可依靠它來串接，它整合Facebook上的層級概念，可以說它是參數版的FB頁面結構，其回傳格式皆為JS參數結構與頁面的內容配置大致相同，對於資料處理十分有幫助，比較容易釐清資料層此外，官方提供預覽工具，以利在開發時檢視所得的資料結構，十分便利，
當做Request的時候，須指定訪問層級(Path)，層級需查找Graph API root nodes列表，但實測發現，有許多層級縱使開了存取權限，仍然無法造訪，API會直接丟出Exception若是對層級結構不熟，又沒有對回傳做例外處理，容易直接掛掉。
官方給的參照並沒有明確說明層級無法造訪的原因跟錯誤回饋，這點在撰寫時會是個困難雖然可以在explorer中可以先簡覽所得內容，但須注意TOKEN認證的項目。

<https://developers.facebook.com/docs/facebook-login/access-tokens>

<TOKEN>
APP使用者登入時(在Facebook API Flow的B動)，會得到一組認證的TOKEN，這組TOKEN包含FB存取權限的認證，對帳戶做存取的動作皆需要經由使用者權限授權才會有效。
在Graph API Explorer中可以看到虛擬的TOKEN值，該值是用來模擬FB APP的權限，以便在Explorer中模擬存取值的結果，在CODE中，該值在登入時便會被賦予，並記錄在Session若是訪問未經授權的層級，將會拿到空的資料包如<右圖所示>，可藉由Get Access Token

因此TOKEN必須在權限被更動時一併被更新;筆者第一次在撰寫時，便遺漏了這點，第一次在使用者權限賦予的選項中並沒有開啟相關權限，而是在第二次測試時才賦予，然而瀏覽頁面的TOKEN並沒有被更新，因此測試過程一直無法撈取到相關資訊，此時，可設法更

