

LINE Bot開發指南

2021年1月18日

※此份為LINE Bot開發基本規則，未經許可禁止轉載本資料。
※各產品亦有額外對應之使用條款與規範。

LINE

Contents

- 01 關於LINE Bot
- 02 使用Webhook URL接收請求時的注意事項
- 03 發送API請求時的注意事項
- 04 LINE Login
- 05 其他相關功能

關於LINE Bot



LINE Developers

LINE Developers是用於管理開發LINE Bot以及LINE連結的應用程式與服務所需的Channel。

LINE Developers可從以下網址進入，瀏覽相關技術資訊文件。<https://developers.line.biz/>

文件



文件 <https://developers.line.biz/en/docs/>

此頁面包含使用Bot和LINE Login等功能所需的技術資訊。

控制台



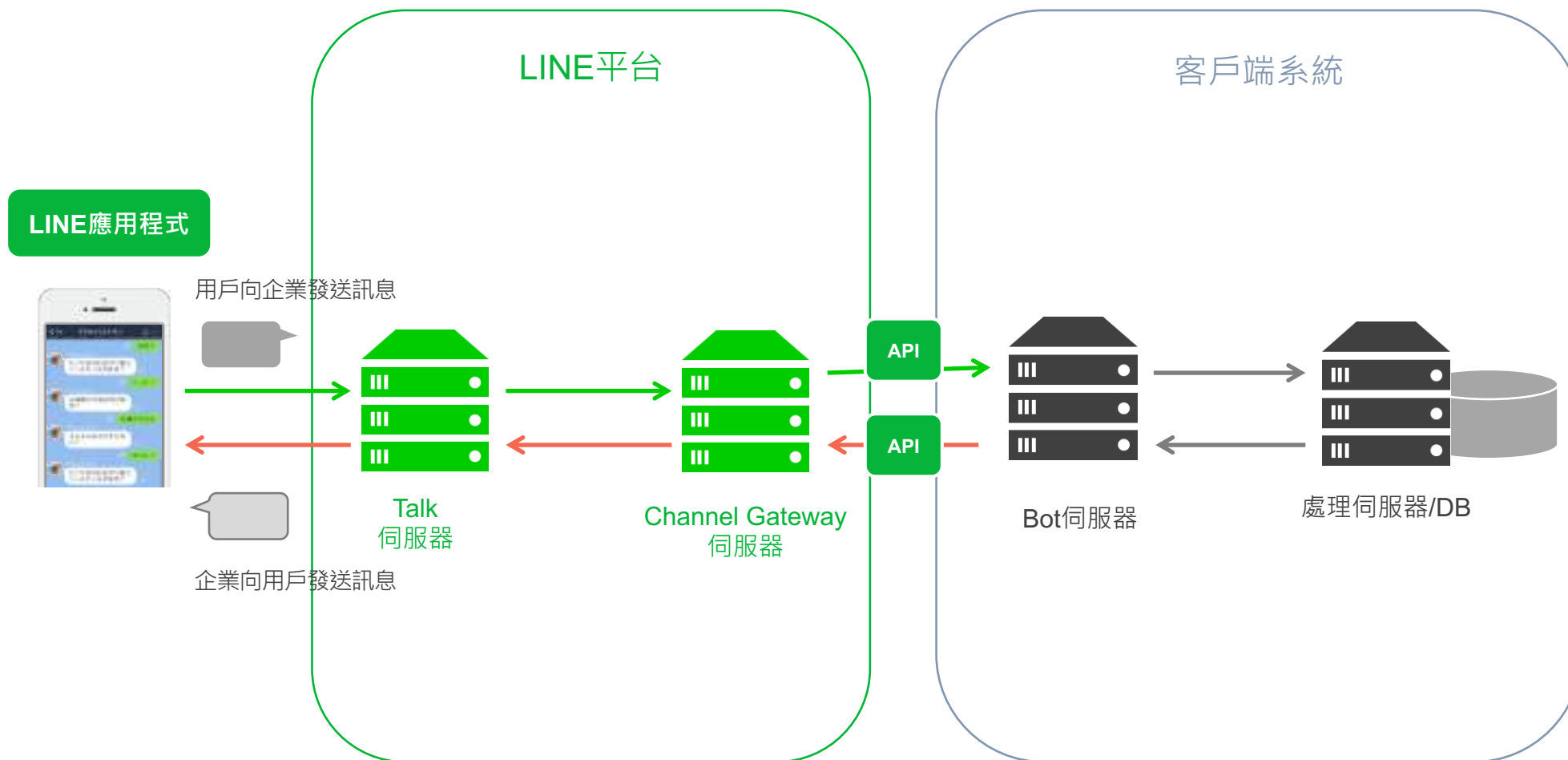
控制台 <https://developers.line.biz/console/>

此頁面用於檢查和設定Channel資訊。

在開發使用Bot和LINE Login等功能的服務時，會需要使用到Channel。

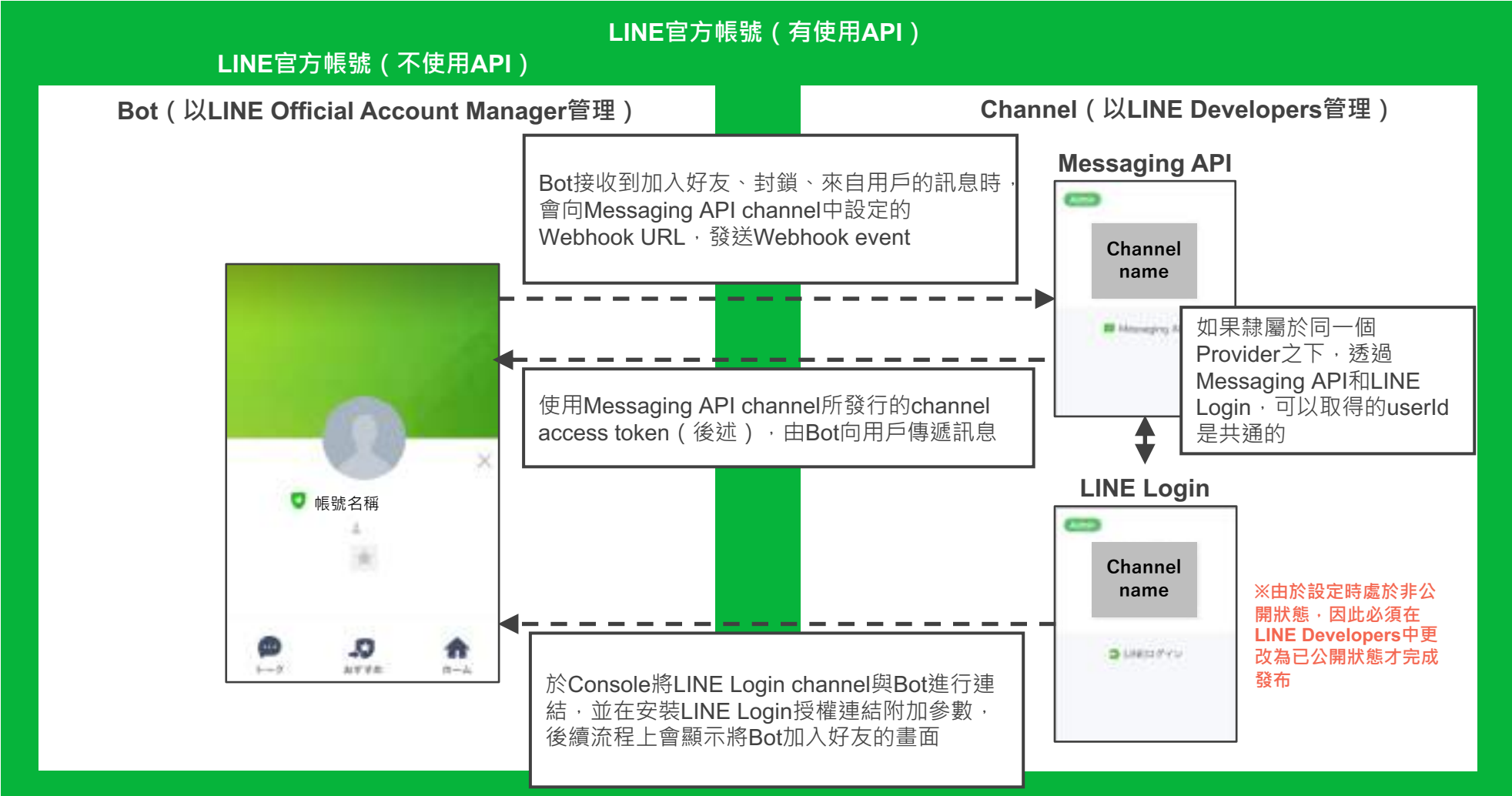
LINE Bot的機制

LINE Bot是一種透過Channel Gateway伺服器，在客戶端系統的Bot伺服器與LINE應用程式之間，發送和接收資訊的機制。使用發送和接收JSON格式數據的API，來發出請求。



關於LINE Bot與Channel之間的關聯性

操作LINE官方帳號，Bot與Channel之間的關聯性如下。



LINE Bot Glossary

Bot伺服器	為了接收LINE所發出的請求（例如加入好友、封鎖、來自用戶的訊息等），而架設於客戶端的伺服器。
Talk伺服器	LINE Platform負責發送和接收訊息的伺服器。
Channel Gateway伺服器 (以下簡稱GW伺服器)	用來將從Talk伺服器接收到的資訊發送到Bot伺服器，或將從Bot伺服器接收到的資訊發送到Talk伺服器的伺服器。使用時必須要有channel。
Request	由GW伺服器向Bot伺服器、或是由Bot伺服器向GW伺服器，發送和接收JSON格式的API資訊。 或是指發送和接收的資訊本身。
Webhook	當發生訊息交換時（例如加入好友或從用戶向Bot發送訊息等），Bot伺服器從GW伺服器接收到的請求。
Webhook event	JSON代碼中所包含的加入好友、封鎖等資訊，以及用戶向官方帳號發送的訊息統稱。 1個Webhook可能會包含多個事件。
Webhook URL	Bot伺服器的endpoint URL。觸發Webhook時，LINE平台會向Webhook URL發送Request。用戶與Bot互動時，會觸發Webhook event。
Provider	Provider 指的是提供服務的個人或公司，在LINE Developers控制台中建立。

也請參考以下的LINE平台術語。

<https://developers.line.biz/en/docs/glossary/>

開發LINE Bot的開始步驟 / 發布前的確認事項

在進行開發時，首先請執行以下步驟。

1. 申請LINE官方帳號，並使用Messaging API的channel。
2. 部署用於發布Messaging API的Bot伺服器，使用可信賴的認證機構所發行的 SSL/TLS 憑證設定Webhook URL。
3. 準備在Bot伺服器上接收事件請求（回傳狀態碼200）。
4. 將LINE Developers的 Messaging API settings的「Use webhook」開啟使用。
5. 將帳號加入好友，並確認Webhook event是否已送達Bot伺服器。

發布給用戶之前，請務必檢查以下內容。

< 遵守「LINE Bot開發指南」 >

1. 需要存取LINE Developers的所有成員，都已為可以使用的狀態。
2. 已確認在Webhook URL項目中設定了正確的URL，且請求已經送達。
3. 安裝時考慮到開發指南中記載的（接收請求時 / 發送請求時的注意事項）。

< 遵守「Terms & Conditions and Guidelines for LINE Services for Business Partners」Official Accounts & API 相關條款> (不限於以下舉例)

4. 將自己公司會員資訊、顧客資料或網站內的行動等記錄結合使用時，會取得用戶的許可。
5. 用戶可以隨時檢查已收集到的資訊處理方式。（明確標示隱私權政策等）
6. 訊息內容中禁止發送需經過嚴格控管安全性的機密資訊，例如密碼、驗證碼和信用卡號等。
7. 當用戶解除與官方帳號之間的好友關係（封鎖）時，原則上開發者於24小時以內，需刪除在帳號管理上不需要用到的資訊。
（※但有可以繼續保存資訊之例外）。
8. 使用本公司提供的資訊時（例如已取得的userId等），僅限於已取得資訊的帳號管理相關操作，其他帳號不得使用。

使用Webhook URL接收請求時的注意事項



在Webhook URL中接收請求時的注意事項

A) 考量安全性的通訊環境

- 採用對應SSL /TLS的通訊環境

B) 接收到請求時，回應狀態代碼200

C) 防止來自LINE以外的未經授權請求 (Webhook Authentication)

D) 對於大規模訊息處理的考量

E) Webhook的ON/OFF設定

F) 其他注意事項

- 1個請求中含有多個事件時的處理方式
- Webhook事件新增屬性對應處理

G) 建議的請求處理步驟

H) GW伺服器 → Bot伺服器的通訊錯誤

A) 考量安全性的通訊環境

- 採用對應SSL/TLS的通訊環境

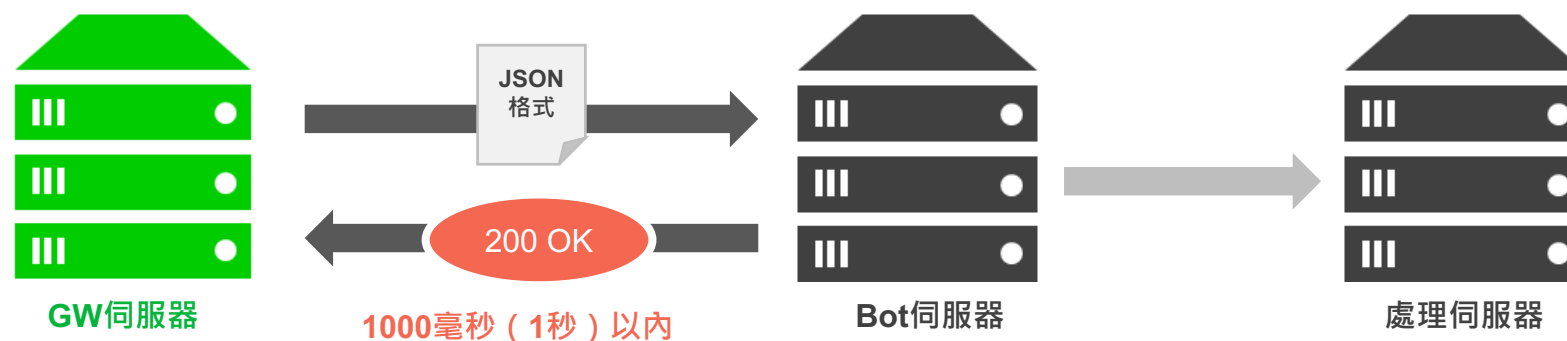
在LINE Bot中，GW伺服器 and Bot伺服器的通訊環境，必須使用對應SSL/TLS。可使用的版本請參考LINE Developers相關公告。
另外，不接受自我簽署憑證，因此**請務必使用certificate authority認可的憑證**。



B) 接收到請求時，回應狀態代碼200

在Bot伺服器上接收到請求時，首先，請盡快回傳狀態代碼200。

狀態代碼的回傳標準，必須在**1000毫秒（1秒）**以內。如果超過1秒，將自動發送警示郵件。



以下說明有關警示郵件的資訊。
欲開啟電子郵件通知，請洽詢承辦業務人員。
<https://developers.line.biz/en/docs/partner-docs/error-notification/>

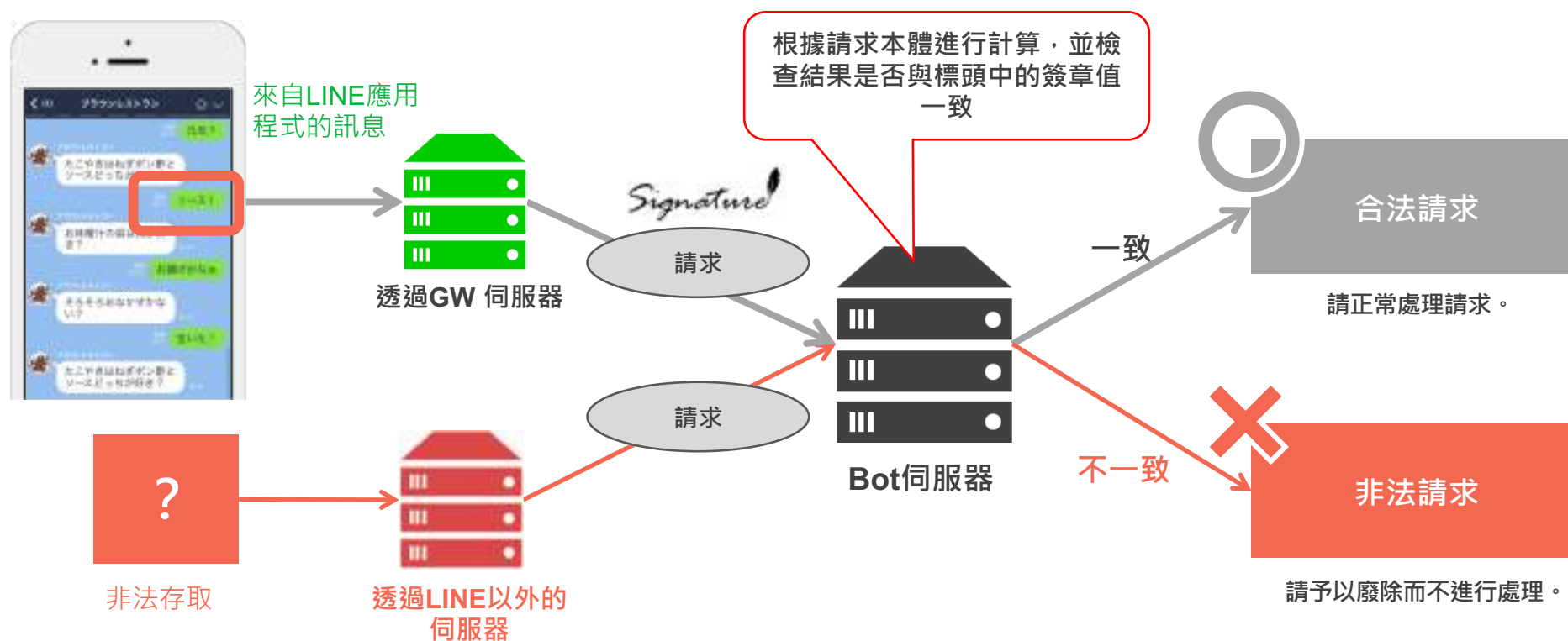
C) 防止來自LINE以外的未經授權請求 (Webhook Authentication)

Webhook請求標頭的「x-line-signature」中含數位簽章。

Bot伺服器使用指定的演算方法，計算已接收到的請求本體，並且比較結果是否與x-line-signature包含的數位簽章一致，藉此來檢查接收到的請求是否是來自LINE。

關於簽名的計算方法，請參閱<https://developers.line.biz/en/reference/messaging-api/#signature-validation>。

驗證過程中使用的channel secret，請特別小心處理。



D) 對於大規模訊息處理的考量

LINE接收到訊息操作可能於特定活動/節日會變得非常大量且集中。如果發生這種情況，可能會發生無法完全處理的請求累積在Bot伺服器上，有延遲發送回覆訊息或訊息未送達的狀況。

針對回覆訊息，請在自己使用的開發環境進行負載測試等措施，作為訊息處理的對策。

容易發生訊息集中的情況

- 才剛將帳號的「顯示搜尋結果」設定為「顯示」之後
- 才剛推出贊助貼圖之後
- 才剛透過將訊息同時發送給所有好友的方式，向用戶發送訊息之後
(特別是在活動期間，期待獲得用戶的高度回應時)
- 當出現在新聞和電視等媒體中的情形時

此外，在中午或晚間至午夜時段為較多人使用的高峰期，訊息操作可能會特別集中，敬請注意。

注意事項

- **GW伺服器並未提供負載測試環境，切勿進行壓力測試**，請自行準備。
- 如果Bot伺服器無法接收所有的請求，也可能造成在GW伺服器累積的情形。如果明顯累積大量的請求時會影響到其他的channel，LINE將不得不刪除累積的請求，敬請知悉。
- 如果是數量高達100萬人擁有大量好友的帳號時，在發布可能會引起熱烈迴響的活動等訊息後，會導致LINE平台的高度負載，因此請視情況進行調整，盡可能避免同時傳遞訊息，或是錯開傳遞時間，進行多次分批傳遞等。

也請一併查看LINE Developers上的說明。

<https://developers.line.biz/en/docs/messaging-api/development-guidelines/>

E-1) Webhook的ON/OFF設定

您可以從LINE Developers的「Messaging API settings」中，將「Use webhook」設定為On/Off。

您可以在LINE Official Account Manager 的「回應設定」中，將「加入好友的歡迎訊息」、「自動回應訊息」、「Webhook」設定為On/Off。

LINE Developers 和LINE Official Account Manager之間的Webhook設定資訊是同步的。

您可以透過設定的組合，來設定**3種樣式**。（請參閱下一頁）

※使用時，請務必利用測試帳號進行驗證與實施，確認沒有問題之後，再套用至正式帳號。

LINE Developers Messaging API設定

Webhook設定

Webhook URL ⓘ

https://l...

Verify

Edit

Use webhook ⓘ



「Use webhook」設定

當發生加入好友或從用戶發送訊息等事件時，Bot伺服器是否要設定從GW伺服器接收請求。欲接收時，請設定Webhook URL。

「加入好友的歡迎訊息」設定

設定是否要回傳已預先設定的加入好友歡迎訊息。

「自動回應訊息」設定

設定是否要回傳已預先設定的自動回應訊息。

LINE Official Account Manager 回應設定

基本設定

選擇新仕 聊天機器人
設定是否要顯示聊天機器人的名稱。
顯示 顯示
隱藏 隱藏
可選擇顯示、隱藏、不顯示及隱藏後再顯示。

加入好友的歡迎訊息 啟用
 停用
加入好友的歡迎訊息設定

進階設定

自動回應訊息 啟用
 停用
自動回應訊息設定

Webhook 啟用
 停用
Messaging API設定

E-2) 關於Webhook的ON/OFF設定選項

- ① 「**使用**」 「Webhook發送」 + 「**使用**」 「自動回應訊息」 / 「加入好友的歡迎訊息」

用於在Messaging API中**接收**請求，並且在回應中使用登錄於LINE Official Account Manager的各種設定訊息。

- ② 「**使用**」 「Webhook發送」 + 「**不使用**」 「自動回應訊息」 / 「加入好友的歡迎訊息」

在Messaging API中**接收**請求，並且透過API由Bot伺服器**回應**訊息。

- ③ 「**不使用**」 「Webhook發送」 + 「**使用**」 「自動回應訊息」 / 「加入好友的歡迎訊息」

Messaging API**不接收**請求，使用登錄於LINE Official Account Manager的各種設定訊息。

- ④ 「**不使用**」 「Webhook發送」 + 「**不使用**」 「自動回應訊息」 / 「加入好友的歡迎訊息」

Messaging API**不接收**請求，也不使用LINE Official Account Manager。不向用戶**回應**任何訊息。
此項設定是不允許的，因此請不要使用這樣的組合。

注意事項

如果您在LINE Bot帳號將「加入好友的歡迎訊息」設定為「使用」，那麼在解除封鎖時，仍然會發送加入好友時的歡迎訊息，敬請知悉。

F) 其他注意事項

請注意，如果不留意以下幾點，可能會導致系統故障。

F-1. 1個請求中含有多個事件時的處理方式

1次的Webhook請求中所包含的Webhook event，並不只限於1個。

在同一時間點發送大量訊息或是加入好友時，可能會在1次的Webhook請求中包含多個事件，請先假設會接收到這樣的請求，然後再發送。

F-2. Webhook事件新增屬性對應處理

Messaging API的功能新增或變更時，可能會被添加屬性。

為了避免將來接收到新屬性的事件時會發生問題，請確認安裝的伺服器狀態。

詳細規格請參閱<https://developers.line.biz/en/reference/messaging-api/#webhook-event-objects>。

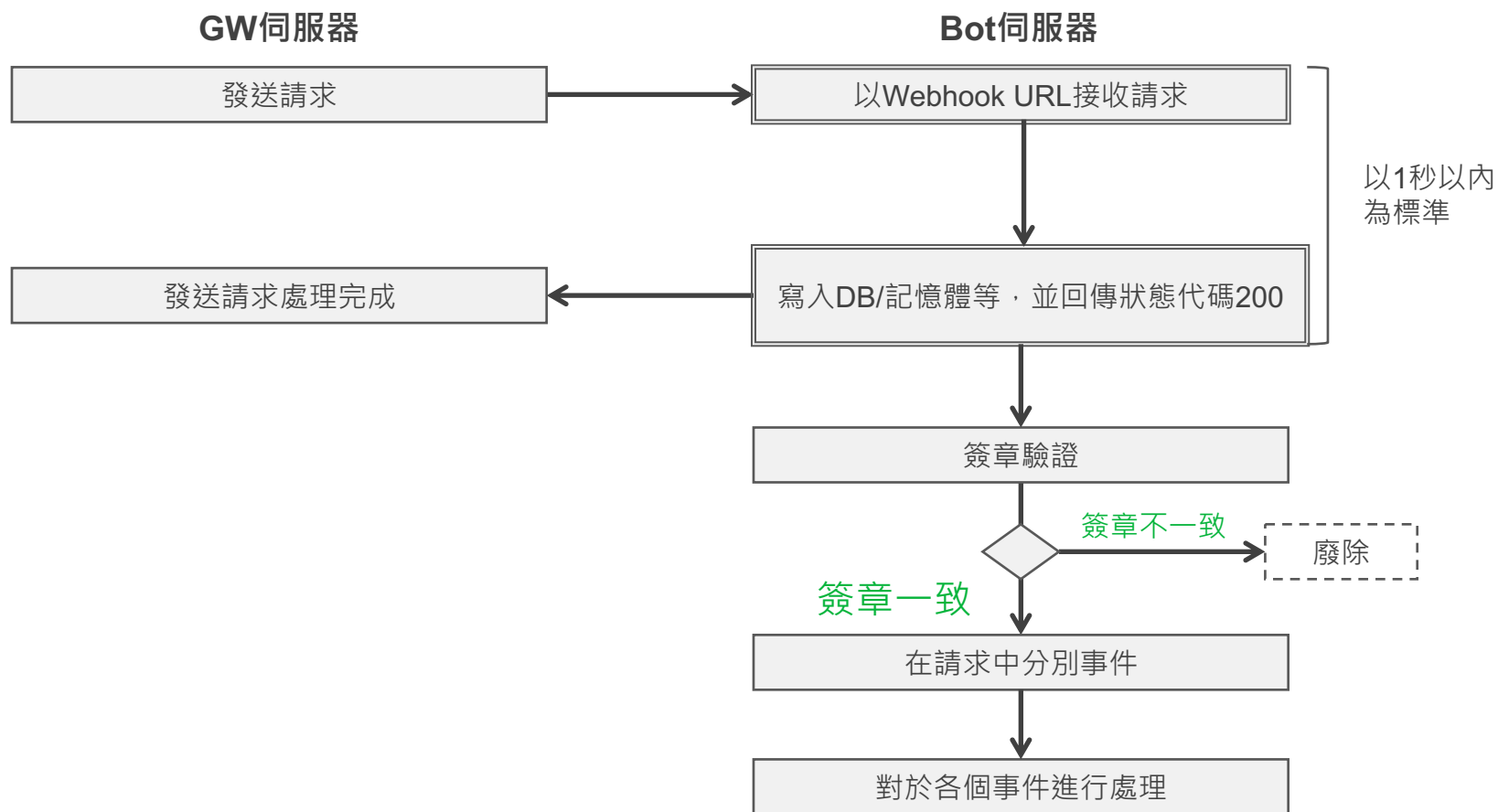
F-3. 關於請求標頭中的文字

請求標頭的欄位名稱中，大小寫的標示方式可能不經通知逕行變更。

在接收到Webhook的bot伺服器端時，請將標頭欄位名稱設定成不區分大小寫。

例如，無論是x-line-signature或X-Line-Signature的任一種方式，都需要設定為可以驗證簽章。

G) 建議的請求處理步驟



H) GW伺服器 → Bot伺服器的通訊錯誤

從GW伺服器到Bot伺服器的通訊發生錯誤時，關於通知的詳細內容，請參閱LINE Developers的文件。

<https://developers.line.biz/en/docs/partner-docs/error-notification/>

※Error notification只在Certified Provider下的channel上適用。

H-1) 關於Error notification

從GW伺服器向Bot伺服器發送請求時，如果Bot伺服器端出現任何問題，我們將透過發送電子郵件或將其顯示在LINE Developers網站上來通知您。
發生錯誤時，請視必要性進行檢查與處理。
欲開啟電子郵件通知，請洽詢承辦業務人員。

郵件收件人

- 在LINE Developers中，登錄於目標Channel基本設定中的電子郵件地址
- 在LINE Developers中，登錄於目標Channel Admin註冊的電子郵件地址

錯誤參考頁

https://developers.line.biz/console/channel/<Channel_ID>/statistics/errors/
(LINE Developers網站的 統計資訊 > 錯誤資訊 的頁面)

H-2) 通知內容範例

請參考文件。

<https://developers.line.biz/en/docs/partner-docs/error-notification/#content>

發送API請求時的注意事項



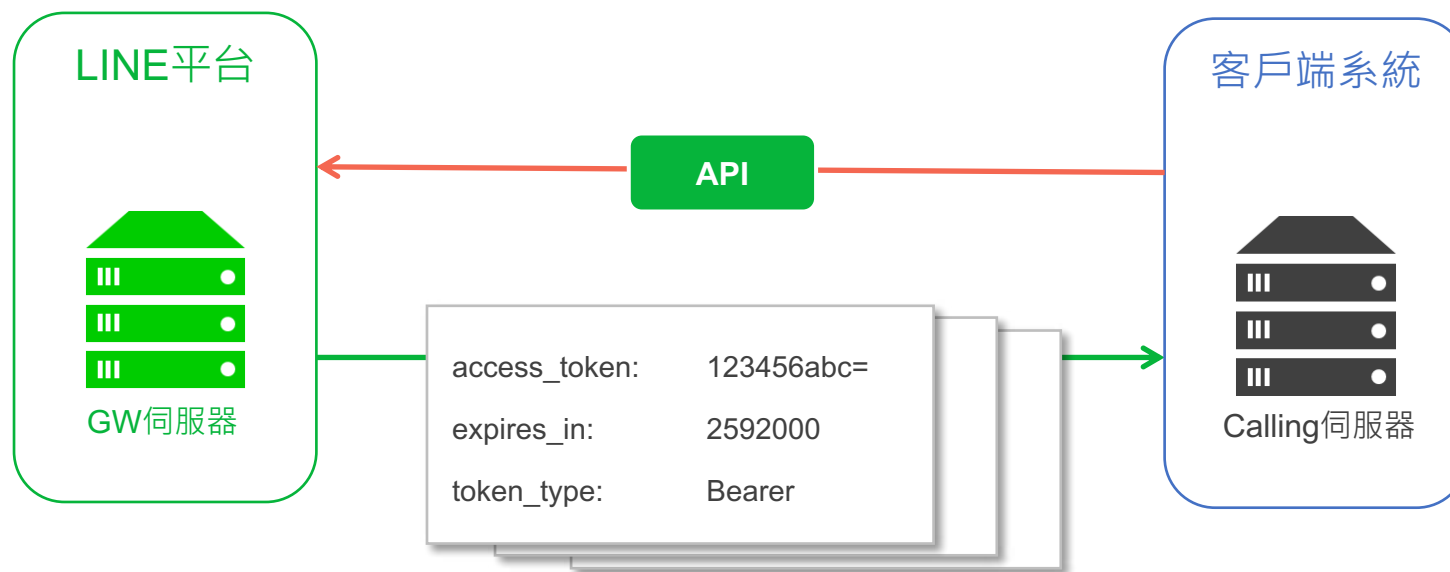
發送API請求時的注意事項

- A) Channel access token的發行
- B) Channel access token自動更新
- C) Channel access token有效上限數量
- D) 訊息發送完成後接收回應
- E) API請求重試
- F) 請求的相關限制
- G) 回應（reply）訊息和推播（push）訊息
- H) HTTPS內容的使用

A) Channel access token的發行

呼叫訊息發送API時，在請求標頭中，包含channel access token的值進行發送。GW伺服器將根據該channel access token的值，判別發出請求的channel，是否為正確的請求。

儘管可以透過LINE Developers控制台以手動方式來發行沒有時效期限的channel access token，但基於安全性的觀點，建議透過API，來發行短期30天的channel access token。



有關詳細的規格，請參閱API文件中的「Issue channel access token」項目。
<https://developers.line.biz/en/reference/messaging-api/#issue-channel-access-token-v2-1>

B) Channel access token自動更新

Channel access token有設定有效期限，超過該日期之後就無法使用。
因此，請建立能定期重新取得channel access token的系統。

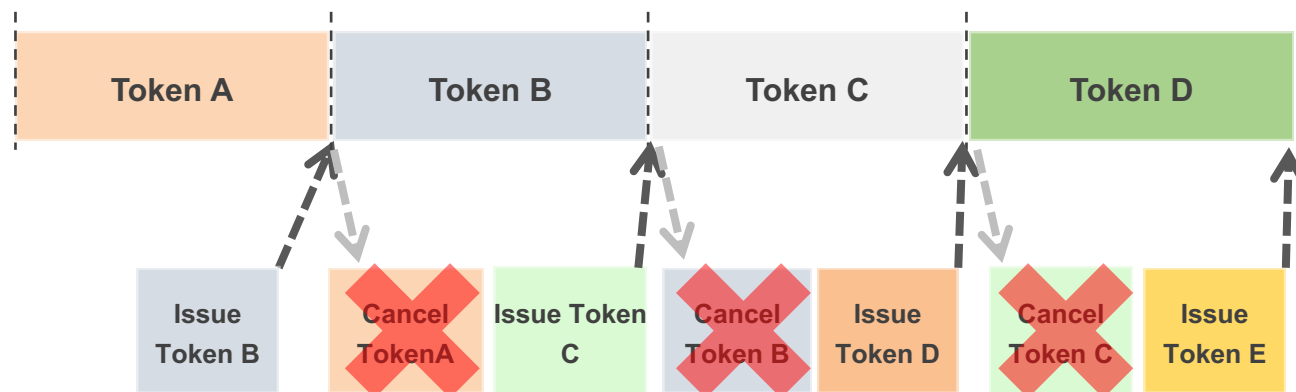
可以發行多個channel access token，但可以操作的數量有上限，因此將多個帳號的服務和伺服器連結使用時，在成功發行新的channel access token之後，建議取消（Revoke）舊的channel access token。

Calling伺服器

使用後端發行的channel access token

後端伺服器

在後端發行與取消（Revoke）channel access token



有關詳細的規格，請參閱API文件中的「Issue channel access token」項目。

<https://developers.line.biz/en/reference/messaging-api/#issue-channel-access-token-v2-1>

C) Channel access token有效上限數量

每個類型的channel access token都有上限且可以同時發行。

<https://developers.line.biz/en/docs/messaging-api/channel-access-tokens/>

種類	發行上限	超過上限時的動作	Channel access token無效的條件
Short-lived channel access token ※1	30個	按照發行順序，使現有的短期channel access token無效	<ul style="list-style-type: none">• 超過有效期限• 超過發行上限• 取消(revoke API) channel access token
Long-lived channel access token ※2	1個	使現有的長期channel access token無效	<ul style="list-style-type: none">• 超過發行上限 (重新發行)• 取消(revoke API) channel access token
User-specified expiration channel access token ※3	30個	API Error 無法追加發行	<ul style="list-style-type: none">• 超過有效期限• 取消(revoke API) channel access token

※1 Short-lived channel access token (由API發行)

※2 Long-lived channel access token (由LINE Developers控制台發行)

※3 User-specified expiration channel access token channel v2.1 (由API發行)

D) 訊息發送完成後接收回應

當您發送訊息時，將收到以下回應。

成功時的回應 範例

成功時，連同狀態代碼200 OK，一併回傳空的JSON物件。

```
{
```

失敗時的回應 範例

如果發送的請求發生錯誤時，將根據錯誤狀態，回傳以下的失敗回應。

發生錯誤時，可以檢查**message**和**details**的內容，可幫助解決問題。

此外，失敗回應內的內容 (Body) 有變動可能，如有更改恕不另行通知。

```
{
  "message": "The request body has 2 error(s)",
  "details": [
    { "message": "May not be empty", "property": "messages[0].text" },
    { "message": "Must be one of the following values:
      [text, image, video, audio, location, sticker, richmessage, template, imagemap]",
      "property": "messages[1].type" } ]
  }

{
  "message": "Invalid reply token"
}
```

回應內標頭包含可識別請求唯一值Request ID 。 (X-Line-Request-Id)
建議也可以檢查該值。

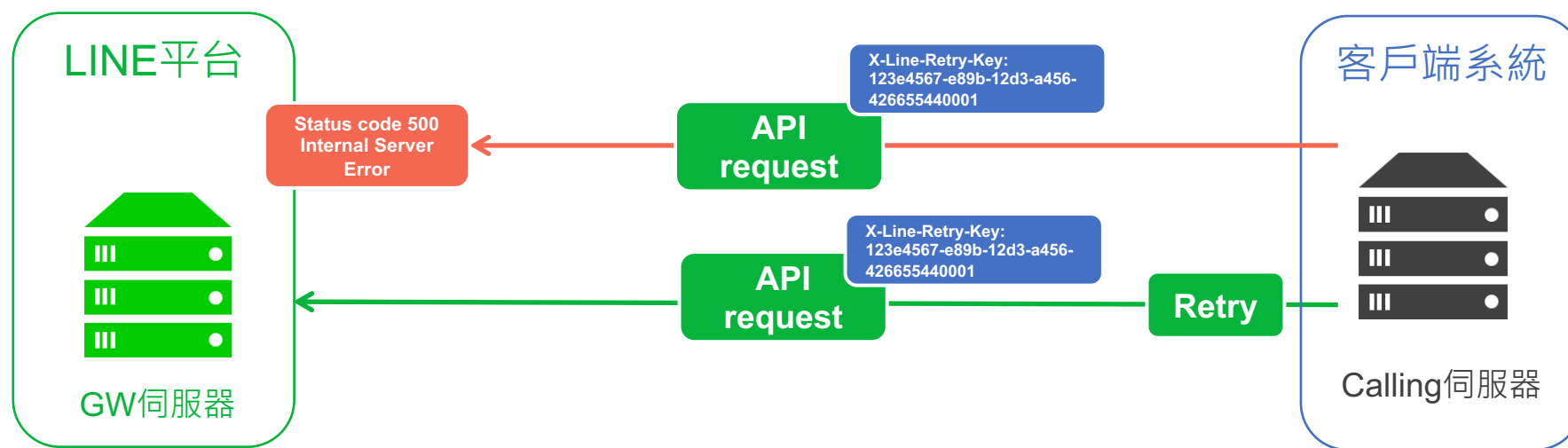
<https://developers.line.biz/en/reference/messaging-api/#response-headers>

E) API請求重試

除了Messaging API伺服器發生故障的情形之外，因網路連接狀態或其他因素，也會造成API請求無法正常被完成或無法正常獲得回應。

為了因應這種情況，建議使用X-Line-Retry-Key，可確認是否該再次發出請求，避免發送重複的訊息。

<https://developers.line.biz/en/docs/messaging-api/retrying-api-request/>



為了減少伺服器或網路故障時的負載，請使用[Exponential backoff](#)控制請求重試的間隔。

F) 請求的相關限制

LINE Bot訊息受到的限制，包括在1次的請求中可以發送的訊息長度限度，以及在一定時間內可以發出請求的次數限制。

文字訊息超過長度上限時，回傳錯誤回應

文字訊息的長度上限為**5000個字**。

如果有超過上限值的請求送達時，將回傳以下的錯誤回應。

```
{"message": "The request body has 1 error(s)",  
  "details": [{"message": "Length must be between 0 and 5000", "property": "messages[0].text"}]}
```

超出速率限制時，回傳錯誤回應

如果在一定時間內，連續發送了大量的請求時，將回傳以下的錯誤回應。

有關目前的速率限制，請參考LINE Developers的以下文件。

<https://developers.line.biz/en/reference/messaging-api/#rate-limits>

```
{"message": "The API rate limit has been exceeded. Try again later."}
```

無論正式帳號或測試帳號，**請避免發送大量測試用的請求**。執行發送負載測試時，請務必在您自己公司的環境中實施。
(以免對於其他帳號造成影響)

注意事項

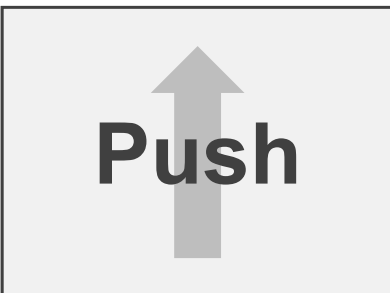
也請一併查看LINE Developers上的說明。<https://developers.line.biz/en/docs/messaging-api/development-guidelines/>

G) 回應（reply）訊息和推播（push）訊息



向用戶、群組、聊天室發送的訊息，進行**回應**的API。

在Webhook通知的事件中，針對可以回覆的事件，指定賦予的replytoken，就能將訊息回覆給事件的發送來源。



無論是否有來自用戶的訊息，都可以在任意的時間點，向**用戶、群組、聊天室發送**訊息的API。

※使用Reply時的注意事項

- 如果您在一定的時間內，未使用replytoken，將會變成**無效**。
從reply的用戶體驗觀點來看，請設定為即時回覆。
- replytoken只能使用**1次**。

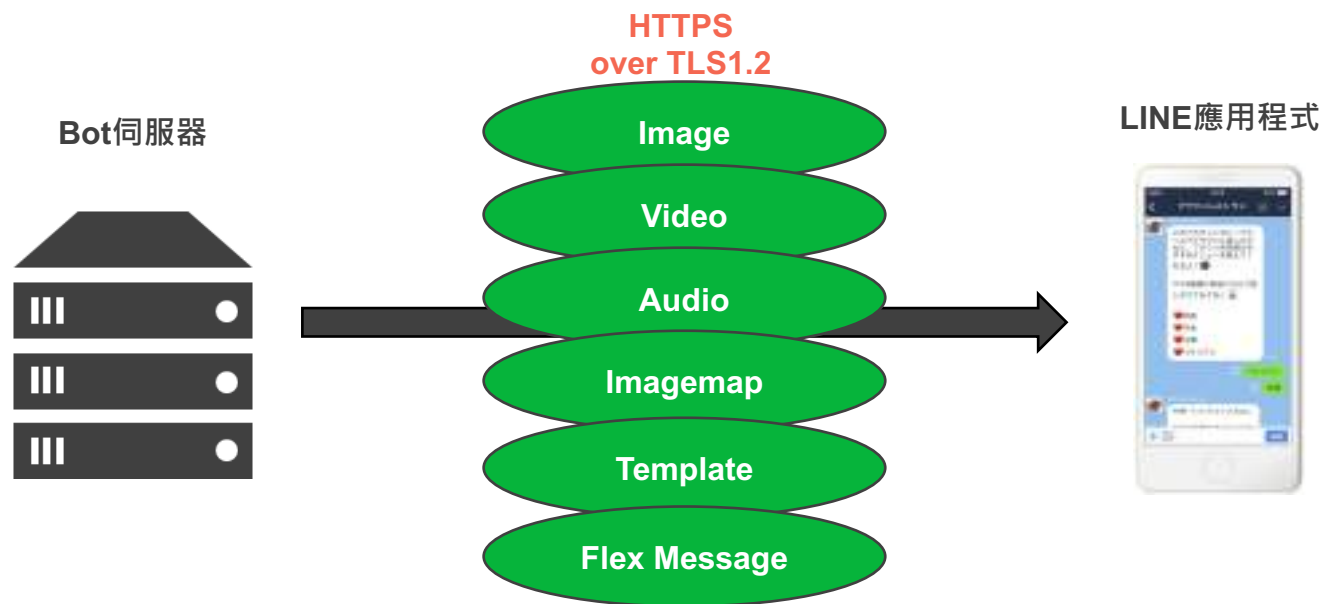
有關詳細的規格和設定方法，請參閱API參考中的Reply訊息和Push訊息項目。

Reply訊息: <https://developers.line.biz/en/reference/messaging-api/#send-reply-message>

Push訊息: <https://developers.line.biz/en/reference/messaging-api/#send-push-message>

H) HTTPS內容的使用

在LINE應用程式上發送圖片、影片、聲音等時，在該Message object內指定的URL，必須為https（ TLS 1.2以上 ）。



※範例 (Image)

```
{
  "type": "image",
  "originalContentUrl": "https://example.com/original.jpg",
  "previewImageUrl": "https://example.com/preview.jpg"
}
```

※內容符合Apple公司的iOS指南方針。

如果使用http或https（ TLS 1.2以下 ），則內容可能無法正常顯示。

LINE Login



LINE Login

在客戶Web環境和手機應用程式中，安裝LINE Login功能，就可以將用戶持有的LINE帳號資訊進行登入動作。透過用戶在LINE公司提供的登入畫面中，使用註冊在LINE中的電子郵件地址和密碼登入，就能將用戶資訊傳送給企業。將該資訊與您公司的資訊連結，就能傳送符合該用戶的特定訊息。

手機版登入畫面



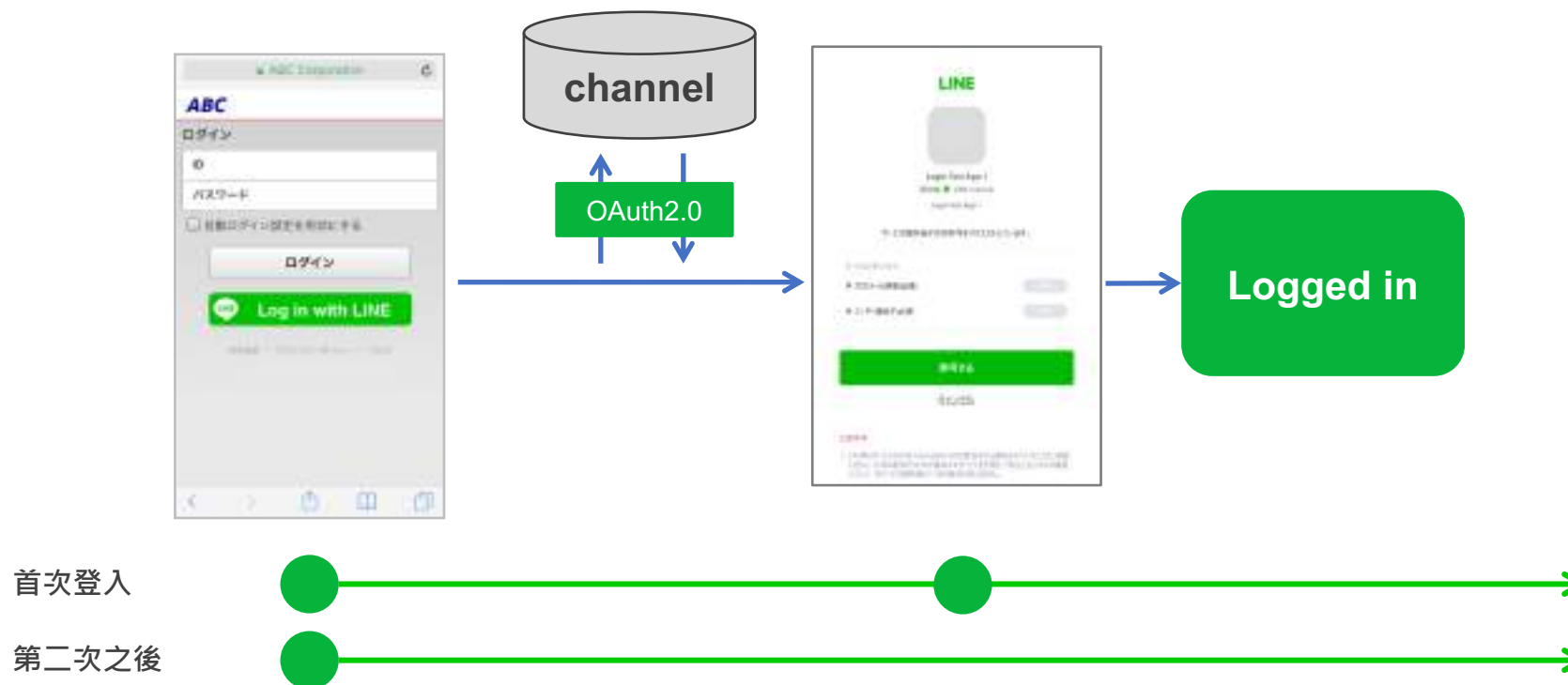
PC版登入畫面



有關詳細的規格和設定方法，請參閱LINE Developers的以下項目。
<https://developers.line.biz/en/docs/line-login/integrate-line-login/>

LINE Login authentication

LINE登入的驗證和授權過程，係基於OAuth 2.0和OpenID® Connect協議。
瀏覽器中的轉換路徑如下。



注意事項

在應用程式內的瀏覽器中，會自動登入，但在外部瀏覽器中，則有透過單一登入(SSO)畫面進行登入、自動登入、瀏覽器顯示登入對話框中輸入註冊在LINE中的電子郵件地址和密碼進行登入等方式。

關於自動登入的動作條件，請參閱LINE Developers的以下項目。

<https://developers.line.biz/en/docs/line-login/integrate-line-login/#authentication-process>

(1) Callback URL的設定

在LINE Developers的「LINE Login」頁面中，可以設定Callback URL。
作為在驗證後，註冊為重新導向的頁面URL。
註冊多個URL時，請使用換行方式進行註冊。

Basic settings LINE Login LIFF Roles

LINE Login settings

Web app ☒

Callback URL ?

https://sample.com/callback
https://sample.com/auth

必須要有https
redirect_uri預設於此
多個URL註冊時需換行

- ✓ Enter no more than 400 lines
- ✓ Use no more than 1000 characters in each line
- ✓ Enter a valid URL on each line

Update Cancel

有關詳細的規格和設定方法，請參閱LINE Developers的以下項目。
<https://developers.line.biz/en/docs/line-login/integrate-line-login/#setting-callback-url>

(2) 驗證與授權

要求LINE登入授權的URL如下。

URL : <https://access.line.me/oauth2/v2.1/authorize>

當用戶點擊依此規則生成的URL時，用戶將以對應的驗證方式登入 LINE，例如輸入電子郵件地址/密碼。

驗證方法的說明

<https://developers.line.biz/en/docs/line-login/integrate-line-login/#authentication-process>

部份參數的概要

redirect_uri：LINE驗證完成後重新導向的頁面。

必須與LINE Developers控制台中預設的Callback URL一致。

scope：指定要求用戶許可的權限。（可以指定多項）

bot_prompt：顯示「將Bot加入好友的選項」同意畫面。

如果省略，則不會顯示選項。

（※請一併參閱後面所述的「自動加入好友功能」）

驗證URL範例

https://access.line.me/oauth2/v2.1/authorize?response_type=code&client_id=1234567890&redirect_uri=https%3A%2F%2Fsample.com%2Fauth&state=12345abcde&scope=openid%20profile%20email&nonce=09876xyz&prompt=consent&bot_prompt=normal

有關呼叫URL時的參數詳細資訊，請參閱LINE Developers中的以下項目。

<https://developers.line.biz/en/docs/line-login/integrate-line-login/#making-an-authorization-request>

(3) 重新導向

完成登入且用戶授予存取權限後，在賦予以下查詢參數的狀態下，重新導向至Callback URL。

重新導向URL範例

```
https://sample.com/auth?code=dibBcll12nWH8KPRZCyQ&state=12345abcde&friendship_status_changed=true
```

參數概要

friendship_status_changed：

如果透過登入，將官方帳號加成好友或解除封鎖時，為true

如果透過登入，但未加成好友或未解除封鎖時，為false

如果在登入時，已經是好友狀態，為false

如果在登入時，未指定bot_prompt參數，無顯示「將Bot加入好友的選項」同意畫面，則此參數將不會出現在回傳的結果裡。

有關Callback URL回傳時的詳細資訊，請參閱LINE Developers的以下項目。
<https://developers.line.biz/en/docs/line-login/integrate-line-login/#receiving-the-authorization-code>

(4) 取得access token API

使用重新導向時的參數中所指定的code值，發出以下API請求，以取得存取權杖。

HTTP Method	POST
Endpoint URL	https://api.line.me/oauth2/v2.1/token
Request header	Content-Type: application/x-www-form-urlencoded

如果API請求成功，將在回應中回傳以下資訊。

回應範例

```
{
  "access_token": "bNI4YEFPI/hjFWhTqexp4MuEw5YPs...",
  "token_type": "Bearer",
  "expires_in": 2591977,
  "refresh_token": "8iFFRdyxNVNLWYeteMMJ",
  "scope": "profile"
  "id_token": "do5d2DokjdF2w35jVe...."
}
```

有關API規格的詳細資訊，請參閱LINE Developers中的以下項目。

<https://developers.line.biz/en/docs/line-login/integrate-line-login/#get-access-token>

(5) 取得ID token

ID token是一種可以透過LINE Login而取得驗證結果的token。
其規格符合OpenID Connect的標準規格。

當上述驗證URL的scope參數中包含「openid」時，將回傳ID token，作為取得access token API的回應。

取得access token API的回應範例

```
{
  "access_token": "bNI4YEFPI/hjFWWhT.....qexp4MuEw5YPs",
  "token_type": "Bearer",
  "expires_in": 2591977,
  "refresh_token": "8iFFRdyxNVNLWYeteMMJ",
  "scope": "profile",
  "id_token": "bWFpbCI6ICJqYW5lZ.....G9lQGV4YW1wbGUuY29t"
}
```

有關ID token處理的詳細資訊，請參閱LINE Developers中的以下項目。
<https://developers.line.biz/en/docs/line-login/integrate-line-login/#verify-id-token>

注意事項

- 為了保護用戶，避免受到第三方的未經授權濫用，請以state、nonce、簽章等方式進行驗證。
- ID token中包含的用戶資訊，是用戶透過LINE Login而完成驗證時的值。
並未隨時保持最新的資訊。

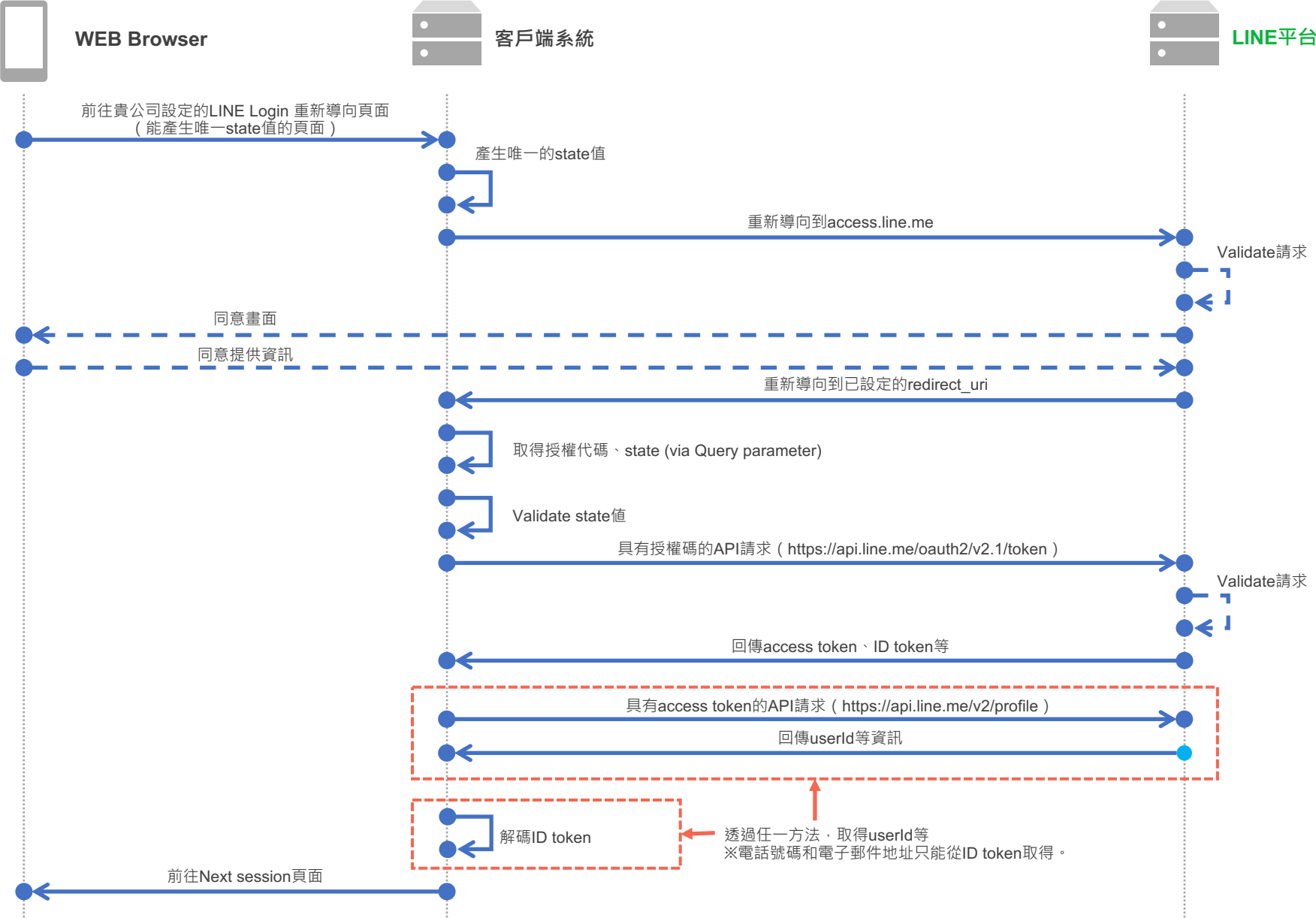
(6) 取得用戶的個人資料

您可以透過使用access token以進行取得個人資料的API，來取得包括用戶ID (userId) 在內的個人資料資訊。

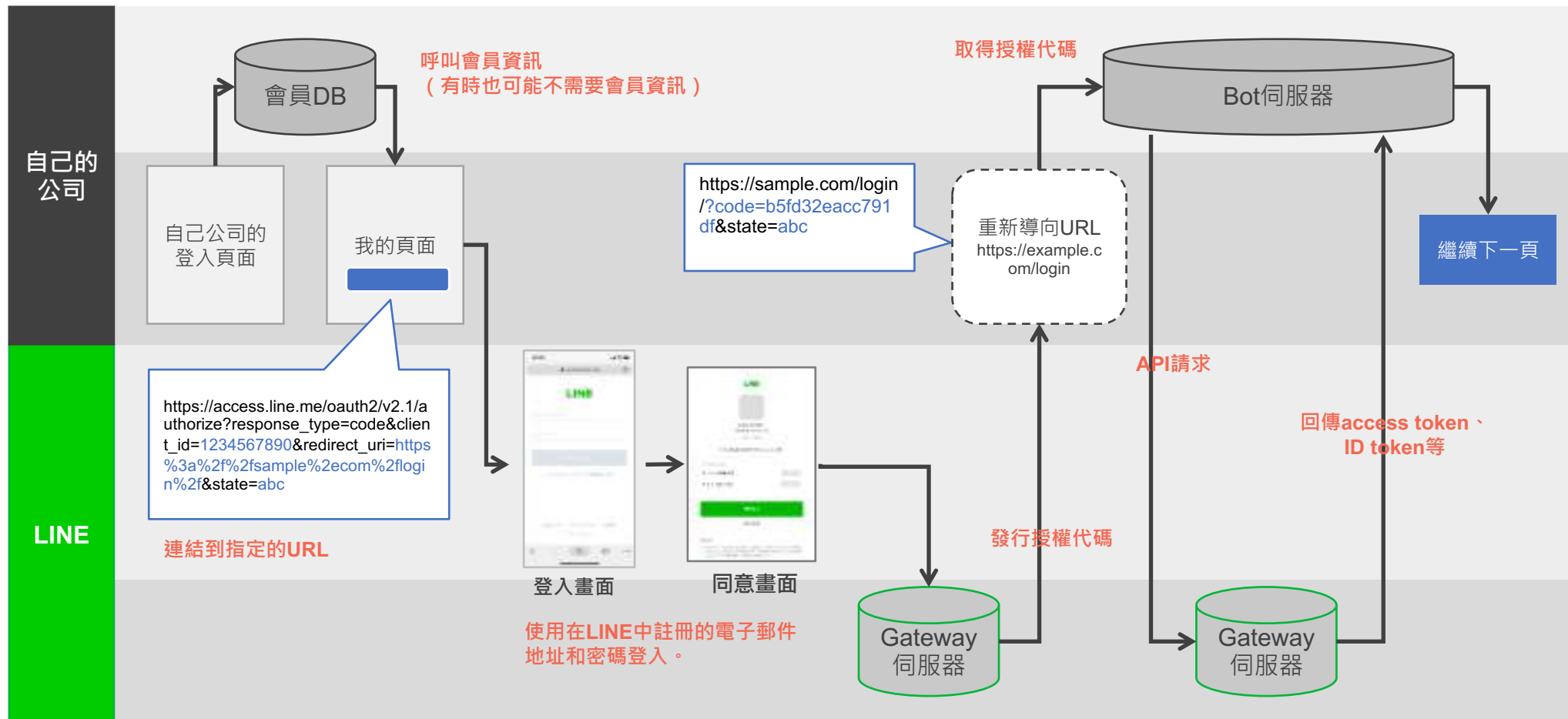
HTTP Method	GET
Endpoint URL	https://api.line.me/v2/profile
Required request header	Authorization: Bearer ACCESS TOKEN

有關API規格的詳細資訊，請參閱LINE Developers中的以下項目。
<https://developers.line.biz/en/reference/line-login/#get-user-profile>

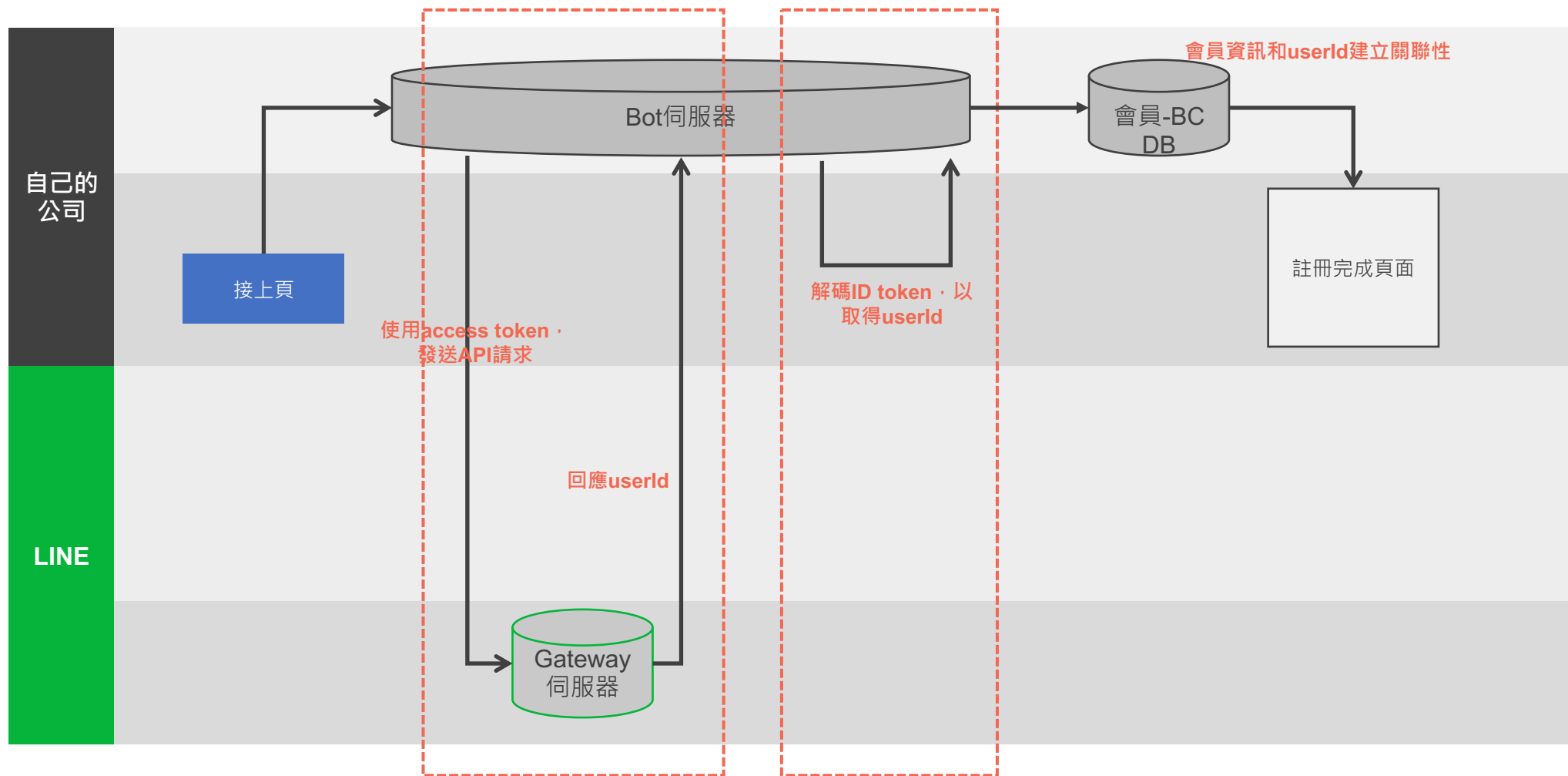
LINE登入處理流程



透過LINE登入，建立帳號關聯性的機制（1/2）



透過LINE登入，建立帳號關聯性的機制（2/2）



透過任一方法，取得userId等
※電話號碼和電子郵件地址只能從ID token取得。

自動加入好友功能（1）

在LINE登入時，您可以使用LINE提供的「將Bot加入好友選項」，自動加入好友。

- 透過指定驗證URL的參數，來使用加入好友的功能。
- 即使已封鎖的用戶，也可以解除封鎖。

有關詳細資訊，請參閱以下文件。

<https://developers.line.biz/en/docs/line-login/link-a-bot/>

【必要的設定】

- 要啟用將Bot加成好友的選項時，必須執行「將Bot連結到LINE Login channel」的作業。

【啟用選項的要件】

- 具有被授予LINE Login使用權限（「WEB」或「NATIVE_APP」）的channel。
- 至少存在一個以上的Bot，Bot的Messaging API channel和LINE Login channel是隸屬同Provider。
- 進行設定的用戶，具有在**LINE Official Account Manager**關聯的官方帳號**Admin**權限。
- 進行設定的用戶，具有在**LINE Developers**中的 **Login channel Admin**權限。

【注意事項】

- 登入時選擇加成好友則和官方帳號關係將被更新。
- 原則上在選擇官方帳號時，除了該channel的正式官方帳號以外，禁止與其餘官方帳號建立關聯性。
- 由於更新會立即反映，因此為了避免誤設到其他帳號（測試帳號等），請在操作時特別小心。
- 如果LINE Login channel是建立在Certified Provider之下，則預設值將會是已勾選加成好友（解除封鎖）的狀態。

自動加入好友功能（2）

當您建置可從公司網站整合LINE Login，可讓先前不是LINE官方帳號好友的用戶，透過在channel授權畫面下方顯示「將Bot加入好友的選項」，達到在用戶登入的同時，使該用戶將公司經營的官方帳號加成好友。



自動加入好友功能 (3)

在LINE Login時顯示「將Bot加入好友的選項」前提，
需於登入驗證URL中，授予「**bot_prompt**」的參數。

驗證URL範例

```
https://access.line.me/oauth2/v2.1/authorize?  
response_type=code&client_id=1234567890&redirect_uri={RedirectURL}&state={State}  
&scope=openid%20profile&bot_prompt=normal
```

如果未包含該參數，則不會顯示「將Bot加入好友的選項」。

有關參數規格的詳細資訊，請參閱LINE Developers中的以下項目。
<https://developers.line.biz/en/docs/line-login/link-a-bot/#displaying-the-option-to-add-your-line-official-account-as-a-friend>

自動加入好友功能（4）

登入完成後，用戶將被重新導向到`redirect_uri`指定的URL。
URL追加`friendly_status_changed`參數，顯示是否已透過登入動作與官方帳號加入好友。

重新導向URL範例

```
https://sample.com/auth?code=dibBcII12nWH8KPRZCyQ&state=12345abcde&friendship_status_changed=true
```

如果您在登入時，未看到將Bot加成好友的選項，則不會出現此參數。
(例：已同意過同意畫面，或者未指定`bot_prompt`參數時等情形)

有關參數規格的詳細資訊，請參閱LINE Developers中的以下項目。

https://developers.line.biz/en/docs/line-login/link-a-bot/#use-friendship_status_changed

好友狀態檢查API

可以檢查用戶和官方帳號是否為好友的API。

您可以使用LINE Login所取得的access token，執行「好友狀態檢查API」，來確定該用戶與官方帳號是否已成好友。

HTTP Method	GET
Endpoint URL	https://api.line.me/friendship/v1/status
Required request header	Authorization: Bearer ACCESS TOKEN

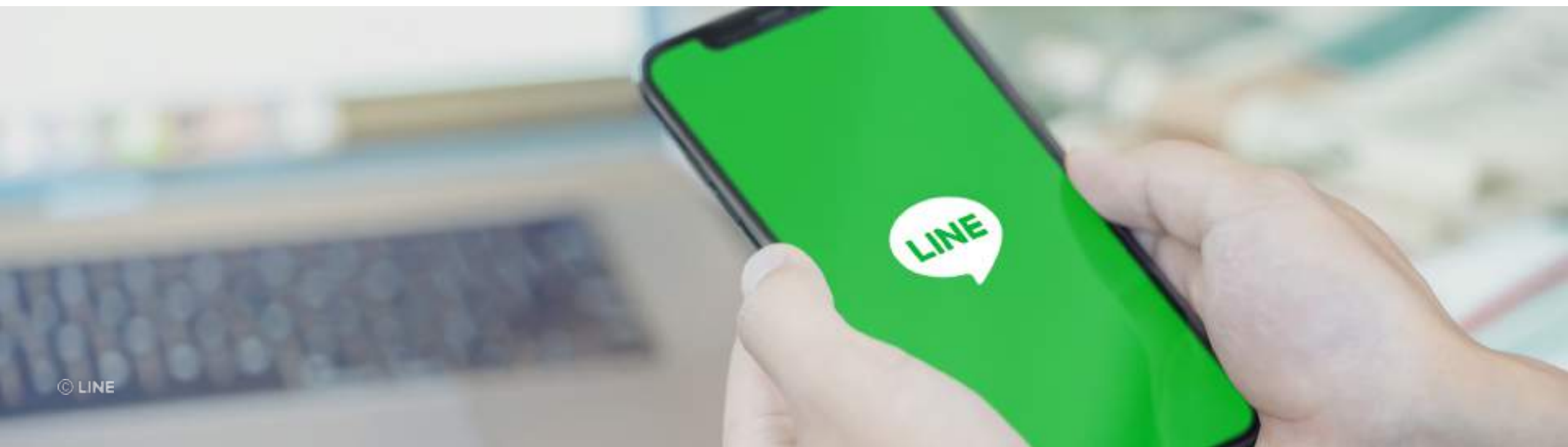
※ Requires an access token with the profile scope.

回應

```
{  
  "friendFlag":true  
}
```

有關API規格的詳細資訊，請參閱LINE Developers中的以下項目。
<https://developers.line.biz/en/reference/line-login/#get-friendship-status>

LINE Login – 補充



關於防止以state不法使用的對策

「state」參數是用於防止跨站請求偽造 (Cross-site request forgery) 的唯一性屬性，避免非法冒充第三方，進行付款等濫用手法的情形。產生授權的URL中保有state參數，並在完成用戶的身份驗證和授權後，確認重新導向URL的state值，對比是否為最初生成的state屬性值一致，以防止未經授權的使用。

利用state的驗證範例



※強烈建議，state參數設定的值，不應為「固定值」或「容易猜測的值」。

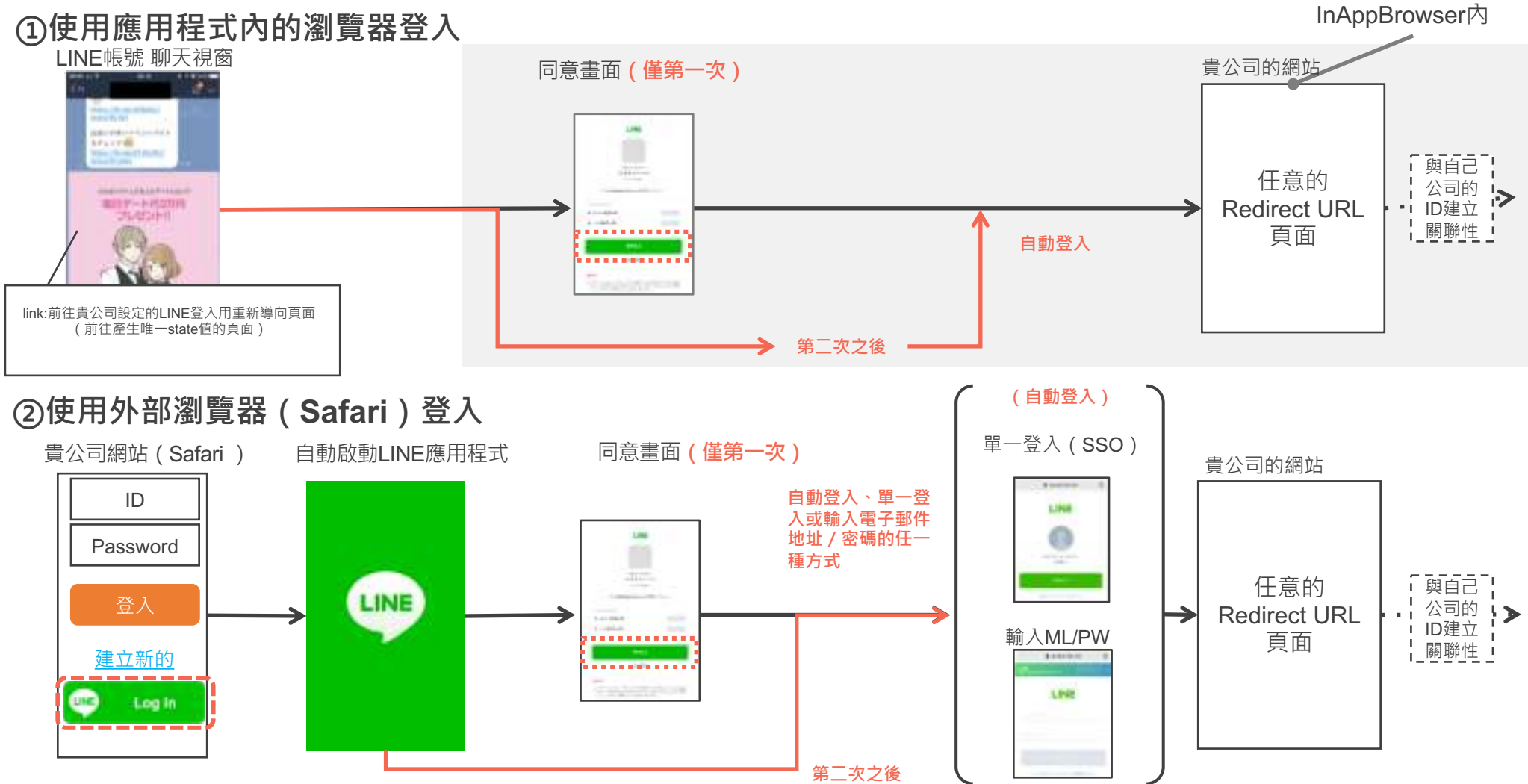
依照不同OS進行LINE登入的用戶流程範例（ iOS ）

以下是建議的UX，實際上也請考慮到企業端的規格等。

自動登入是指：在沒有輸入LINE的電子郵件地址 / 密碼等用戶操作之下，使其登入LINE的方式。

單一登入功能（SSO）是指：點擊確認畫面上的登入按鈕進行登入。只要曾經從 Web app 登入到 LINE，將會在 Cookie 中儲存 access.line.me 的網域名稱。只要 Cookie 尚未過期，當您在相同的瀏覽器登入時，就會顯示 SSO 畫面，

※由於自動登入和SSO，是由LINE應用程式端處理，因此貴公司只需要對應正常的LINE登入系統即可。



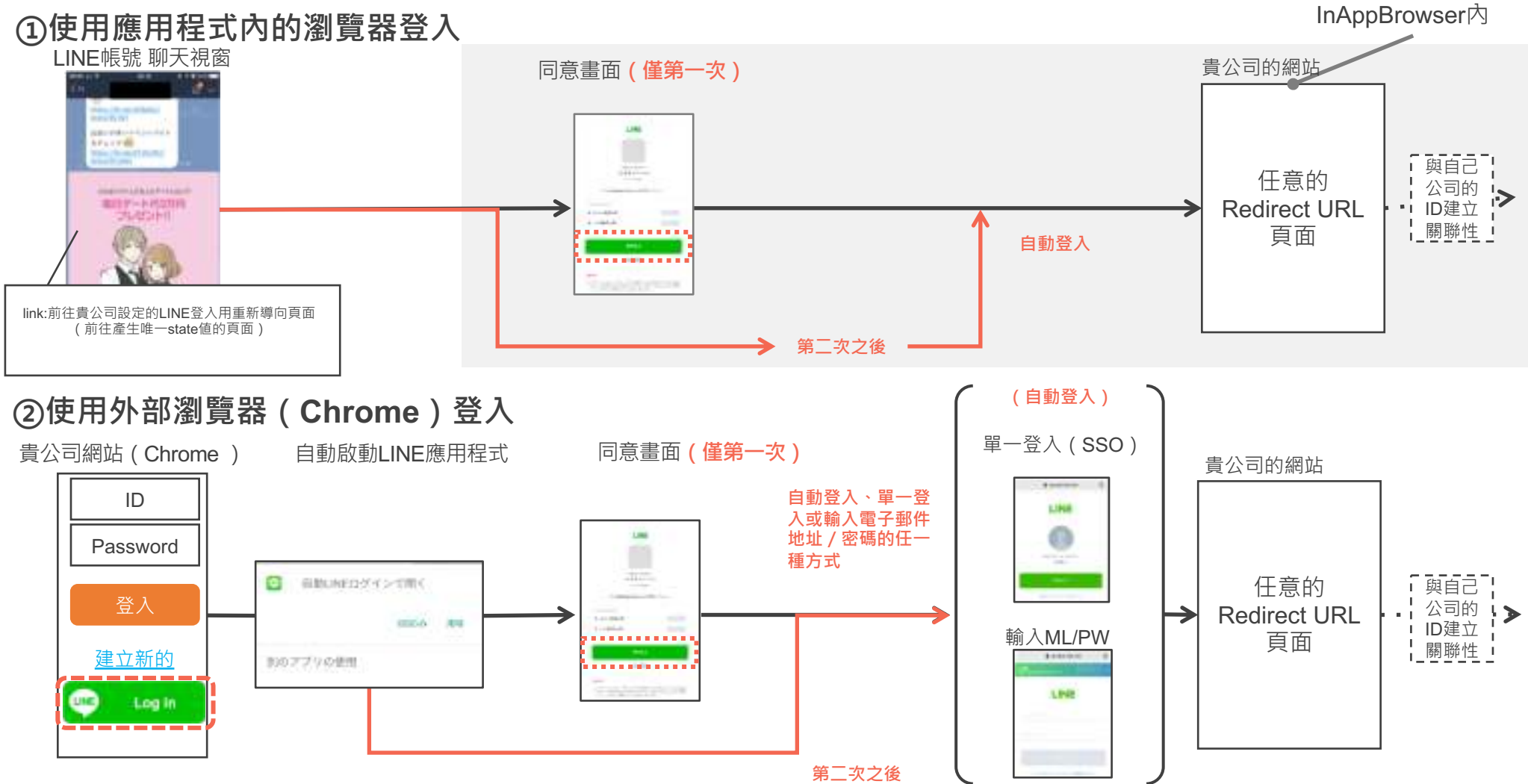
依照不同OS進行LINE登入的用戶流程範例（Android）

以下是建議的UX，實際上也請考慮到企業端的規格等。

自動登入是指：在沒有輸入LINE的電子郵件地址 / 密碼等用戶操作之下，使其登入LINE的方式。

單一登入功能（SSO）是指：點擊確認畫面上的登入按鈕進行登入。只要曾經從 Web app 登入到 LINE，將會在 Cookie 中儲存 access.line.me 的網域名稱。只要 Cookie 尚未過期，當您在相同的瀏覽器登入時，就會顯示 SSO 畫面，

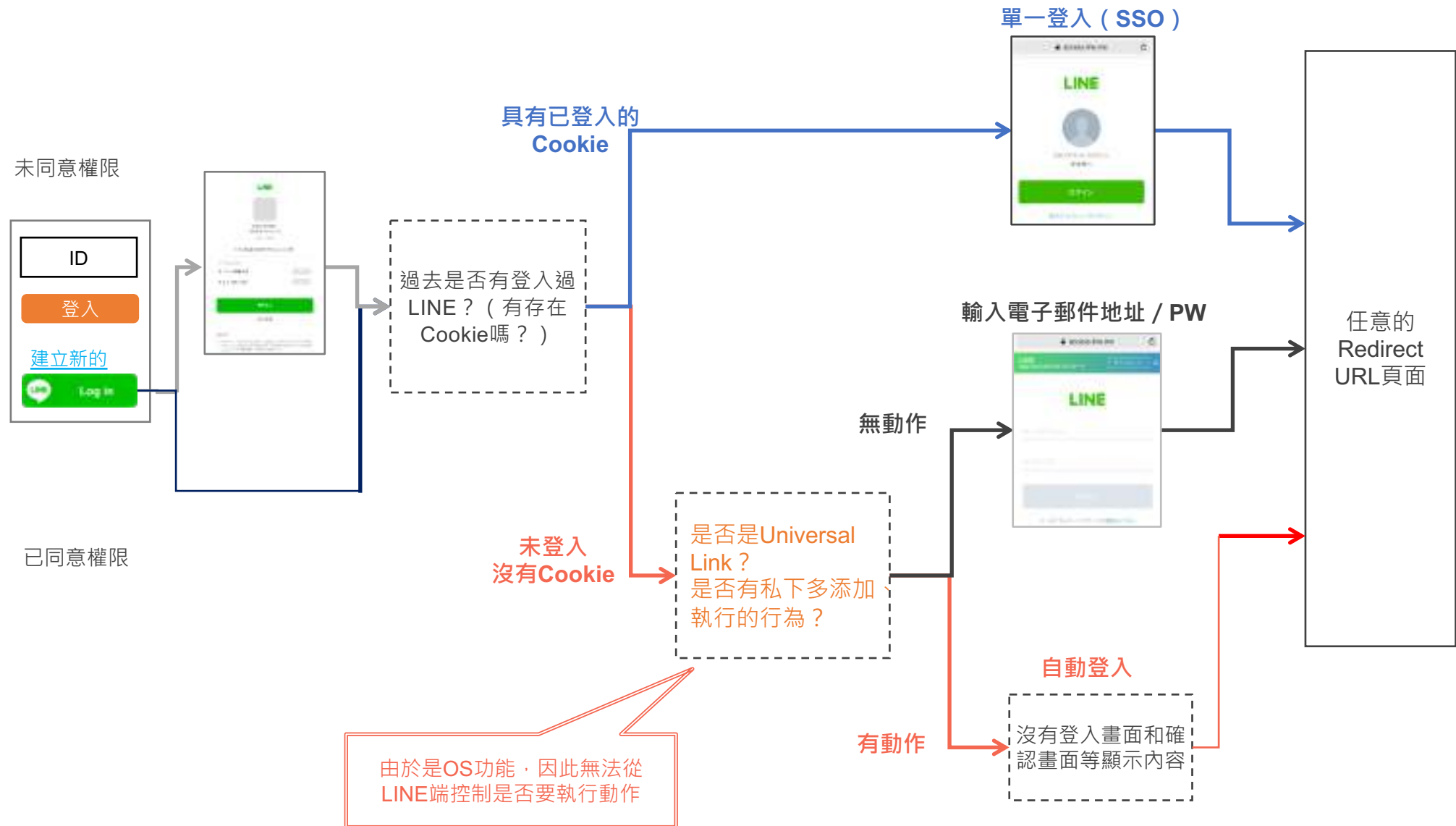
※由於自動登入和SSO，是由LINE應用程式端處理，因此貴公司只需要對應正常的LINE登入系統即可。



外部瀏覽器的登入流程（示意圖）

有關登入LINE時的驗證方法詳細資訊，請參閱以下的「驗證流程」。

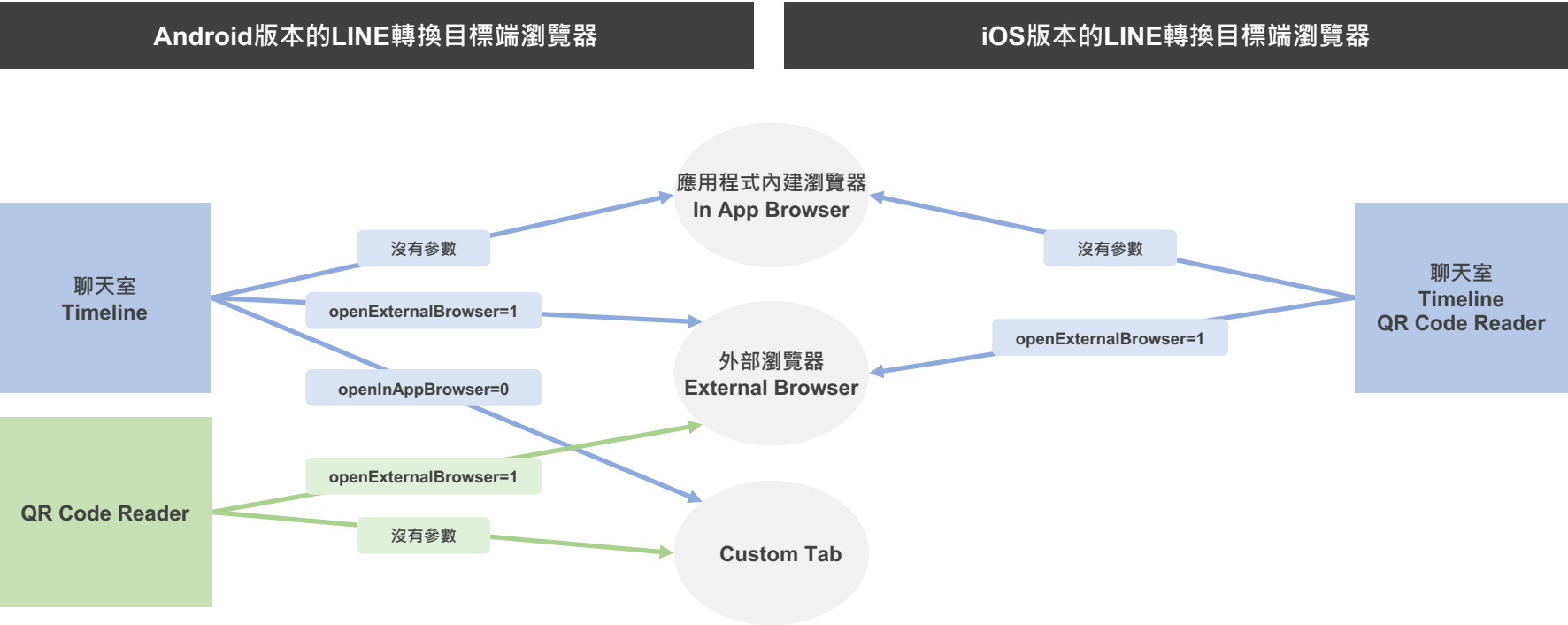
<https://developers.line.biz/en/docs/line-login/integrate-line-login/#authentication-process>



關於轉換目標端瀏覽器的設定方法

透過設定以下查詢參數，可以選擇轉換目標端的瀏覽器。
※請注意，Android版本的LINE和iOS版本的LINE，規格不同。

瀏覽器	設定方法
應用程式內建瀏覽器 In App Browser	沒有參數 ※但是，利用Android版本LINE的QR Code Reader讀取QR Code時，則會到轉換Custom Tab。
外部瀏覽器 External Browser	在URL查詢參數中設定openExternalBrowser=1 (或 true)
Custom Tab	在URL查詢參數中設定openInAppBrowser=0 (或 false) ※但是，僅適用於Android版本的LINE (8.15.0以上)。



其他相關功能



LINE URL scheme

LINE提供了幾種能夠更加方便使用LINE Bot的LINE URL scheme。

有關詳細資訊，請參閱LINE Developers的文件。<https://developers.line.biz/en/docs/line-login/using-line-url-scheme/>

A) https://line.me/R/oaMessage/{LINE ID}?{TEXT_MESSAGE}

點擊連結開啟官方帳號的聊天畫面，指定的文字訊息將出現在輸入欄內。如果該官方帳號尚未被加入好友，則聊天視窗會顯示加好友畫面。如果要填寫的訊息包含多字節時，請使用UTF-8進行URL編碼。

範例：<https://line.me/R/oaMessage/@linejpen/?%e3%81%93%e3%82%93%e3%81%ab%e3%81%a1%e3%81%af>

B) <https://line.me/R/ti/p/LINE ID>

點擊連結，以進入到任何官方帳號的加入好友畫面。如果從PC開啟頁面，將顯示QRCode。

範例：<https://line.me/R/ti/p/@linenews>

C) https://line.me/R/msg/text/{TEXT_MESSAGE} https://line.me/R/share?text={text_message}

開啟分享畫面，分享指定文字的訊息。如果要填寫的訊息包含多字節時，請使用UTF-8進行URL編碼。

範例：<https://line.me/R/msg/text/?%e3%81%93%e3%82%93%e3%81%ab%e3%81%a1%e3%81%af>

D) <https://line.me/R/nv/location/>

在與用戶聊天畫面上，將開啟分享位置資訊的畫面。此LINE URLscheme不支援1-1和Bot聊天以外的形式或LIFF Apps。

LIFF (LINE Front-end Framework)

LIFF (LINE Front-end Framework) 是由LINE 提供的web app platform。

在LINE platform啟動的web apps統稱LIFF apps，可從LINE platform取得 user ID和 chat ID（在聊天室中打開時）及使用各種功能所需的token。透過使用這些功能，可以開發與用戶資訊連結的功能以及代替用戶將訊息發送到聊天室的應用程式。

可以使用以下的URL scheme，開啟LIFF畫面。

<https://liff.line.me/{liffid}>

有關詳細的規格和設定方法，請參閱LINE Developers的以下項目。

<https://developers.line.biz/en/docs/liff/>



權限管理

在開始LINE Login服務之前（非公開狀態），執行測試時，可先透過LINE Developers channel權限管理確認對應人員。

<https://developers.line.biz/en/docs/line-developers-console/managing-roles/>

設定權限

可以對於該channel進行編輯 / 查看 / 測試的權限。

- Admin：可以執行與該channel有關的所有操作。
- Member：可以查看該channel的名稱和圖示等基本資訊，以及統計資訊。
- Tester：可以查看該channel的名稱。另外，可以在非公開狀態的channel上進行測試。
※具有管理員權限的用戶，將自動成為測試人員，並且可以在終端上進行測試。



貼圖的使用方法

在LINE Bot中，是透過ID資訊進行發送和接收，Bot無法知道貼圖本身是什麼。貼圖是根據每個提供商的權利來提供，因此請根據以下規則使用。

從官方帳號向用戶發送貼圖時

官方帳號可以發送以下2種類型的貼圖。

A. 官方預設貼圖

請從此處確認ID對應表。[點擊此處查看對應表](#)

B. 由企業提供，並且自己公司擁有授權的貼圖

發送此類貼圖時，本公司需要將其貼圖設定為白名單才可進行發送。
有關申請方法與相關遵守規範，請洽詢承辦業務人員。



關於Emoji的發送和接收

建議使用的Emoji有以下2種: Unicode emoji和 LINE emoji。

有關詳細資訊，請參閱LINE Developers的文件。 <https://developers.line.biz/en/reference/messaging-api/#text-message>

LINE emoji

Sendable LINE emoji list:

https://d.line-scdn.net/r/devcenter/sendable_line_emoji_list.pdf

Emoji ID List

 Emoji ID

Product ID: 5ac1bfd5040ab15980c9b435

 001	 002	 003	 004	 005
 008	 009	 010	 011	 012
 015	 016	 017	 018	 019
 022	 023	 024	 025	 026
 029	 030	 031	 032	 033

THANK YOU