數位服務個人化

資料提供者技術文件

V2.0

國家發展委員會
中華民國 109 年 3 月

版本修正紀錄

項次	版本	時間	修正內容	頁次
1	2.0	109/03/30	調整章節「貳、三、完成授權驗證等相關系統整合介接」之內容，資料提供者需配合 MyData 實作授權驗證之系統整合介接。	P5
			調整章節「陸、MyData 整合協作流程說明」之內容。	P15
			調整章節「柒、授權 API Endpoint 規格說明」之內容。	P18
			新增章節「拾、交易 Log 日誌查詢」。	P33
			調整章節「拾壹、二、系統環境主機及網址資訊」之連線資訊。	P38

目錄

壹、目的

本文件目的主要描述「資料提供者」於實作「MyData 平臺之資料提供者」時應依循的作業流程、準則、技術規格及相關注意事項。

貳、如何成為資料提供者

一、完成 **MyData** 資料提供者資格申請作業

機關單位如欲加入 MyData 成為「資料提供者」，需先完成資格申請。

內容細節請參考本文件章節「肆、資料提供者資格申請作業」。

二、完成 **MyData** 線上註冊作業

註冊作業包括：

- 身分註冊：登錄機關單位基本資料。
- 資料集註冊：登入資料集資訊。

內容細節請參考本文件章節「伍、資料提供者管理作業」。

三、完成授權驗證等相關系統整合介接

資料提供者需配合 MyData 實作授權驗證之系統整合介接。

相關之 Endpoint 規格，請參考章節「柒、授權驗證 API Endpoint 規格說明」。

四、實作資料提供介面規格

資料提供者應保有彈性並定義資料提供介面規格，但於發展實作資料提供介面規格時，應參考並依循國發會發佈之「共通性應用程式介面規範（OAS）」所提及之要點實作，使 API 具有共通性之特性，為擴大政府資訊服務效益。

資料提供介面規格準則，請參考本文件章節「捌、DP-API Endpoint 規格準則」及「玖、DP 資料打包檔案規格準則」。

參、名詞定義

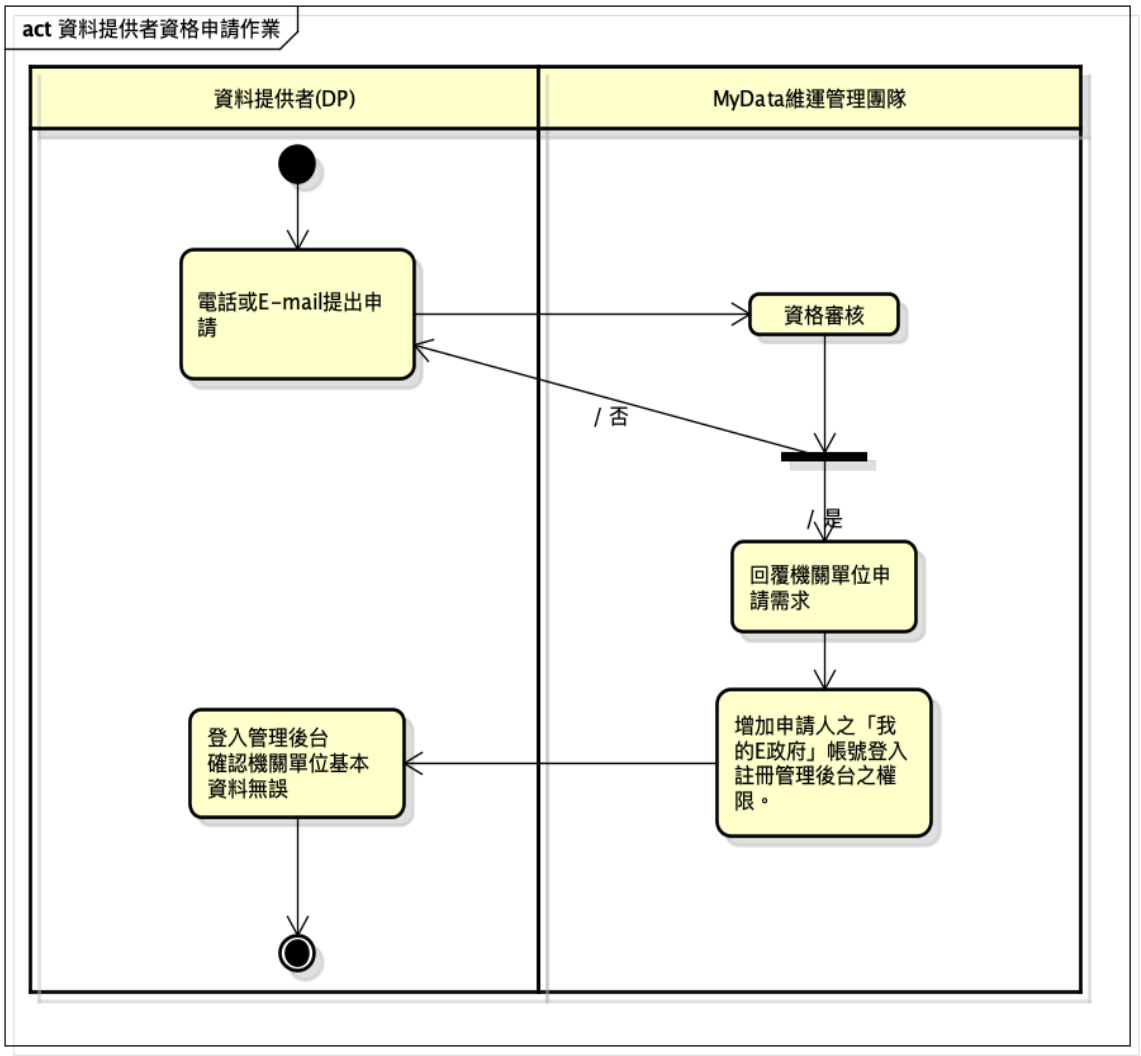
名稱	定義
OAS	共通性應用程式介面規範。
Data Provider, DP	資料提供者，存放或保管民眾個人資料之機關單位。
Service Provider, SP	服務提供者，提供民眾進行個人資料之加值服務機關單位。
Authorization Server, AS	授權管理者，執行身分驗證與授權管理機制。
Resource Owner, RO	資料擁有者/使用者，泛指用戶或民眾。
OAuth 2.0	系統授權流程規範，定義於 RFC 6749 The OAuth 2.0 Authorization Framework https://tools.ietf.org/html/rfc6749
OpenID Connect	OAuth 2.0 的補充規範，強調身分驗證流程 http://openid.net/connect/
access_token	AS 核發的授權 token

肆、資料提供者資格申請作業

機關單位欲使用 MyData 機制成為資料提供者角色，應先完成資格申請，步驟說明如下：

步驟項次	流程內容
1	機關單位以電話或 E-mail 聯繫管理團隊，提出 MyData 註冊管理後台使用權限申請。（聯絡資訊 Tel:02-86925588#5555, E-mail: mydata@ndc.gov.tw）
2	管理團隊回覆機關單位申請需求，增加機關單位申請人之「我的 E 政府」帳號登入註冊管理後台之權限。
3	機關單位申請人以「我的 E 政府」帳號登入註冊管理後台並確認機關單位基本資料無誤後，使用資料提供者管理功能項目。
4	完成。機關單位已成為資料提供者，可使用後台資料集註冊相關功

	能。



機關單位以電話（號碼）、電子郵件（信箱）聯繫 MyData 維運團隊申辦註冊管理後台機關帳號，並於申辦時提供「機關單位名稱」及「機關單位地址」、「聯絡人姓名」、「聯絡電話」、「電子郵件信箱」與申請人之「我的 E 政府註冊帳號」，由 MyData 維運人員協助完成帳號註冊作業。完成機關單位註冊後，MyData 維運人員將透過註冊時機關單位提供之「聯絡電話」及「電子郵件信箱」通知機關單位聯絡人。

伍、資料提供者管理作業

一、基本資料編輯

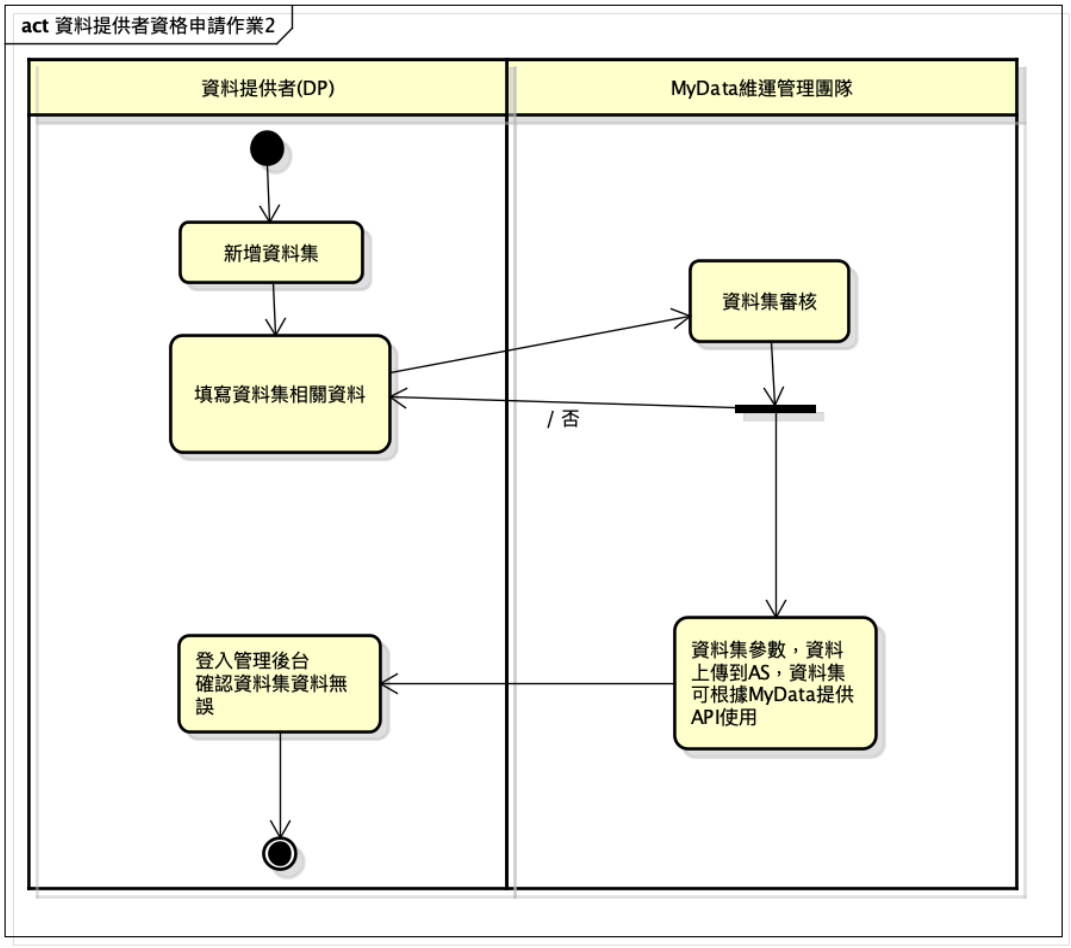
機關單位登入管理平臺後，點選「機關單位管理」功能項目，可自行編輯機關單位基本資料，包含「聯絡人姓名」、「聯絡電話」、「聯絡 E-mail」、「E 政府帳號」（管理後台登入使用）。於此功能頁面中，可瀏覽目前機關單位已建立之資料集與加值服務項目。

機關單位登入管理平臺後，點選「資料提供者管理」功能項目，若尚未同意「資料提供者合作契約」內容，需先同意契約內容成為資料提供者身分後，即可開通資料提供者管理相關功能使用權限（資料集新增、編輯與管理功能）。

二、資料集註冊

步驟項次	流程內容

1	資料提供者使用 MyData 註冊管理後台「新增資料集」功能。
2	完成新增資料集頁面中相關欄位內容，並提交審核。
3	經維運管理團隊確認相關內容符合 MyData 規範後，通知機關單位（申請人）審核通過並於註冊管理後台的資料集清單中上架資訊；若審核未通過，將回覆機關單位（申請人）修正建議，機關單位依修正建議調整後可重新提交審核申請。
4	服務提供者於管理後台中的「可運用資料集」功能中，閱覽審核通過上架之資料集。
5	待服務提供者依資料提供者說明之介接方式完成資料介接作業。



新增資料集	
基本資料欄位	機關單位名稱、申請人、聯絡電話、聯絡 Email、資料集類別、資料集名稱、資料集說明、授權取用資料安全等級、資料更新時間、資料集欄位、資料集存取範圍、resource_id、介接 API 文件、資料檔解析規則文件。
選填欄位	無
功能動作	上傳 API 文件、上傳資料檔解析規則文件、取消新增、提交審核。
系統產生欄位	申請日期、resource_id、resource_secret(審核通過後才會顯示)。

欄位介面示意：

三、資料集管理

顯示已註冊、申請中之資料資源項目清單，並提供關鍵字查詢與新增、修改、刪除功能。刪除功能僅限資料集服務狀態為停用者，服務狀態為啟用之資料集應先完成停用申請流程。

修改資料集	
基本資料欄位	機關單位名稱、申請人、聯絡電話、聯絡 Email、資料集類別、資料集名稱、資料集說明、授權取用資料安全等級、資料更新時間、資料集欄位、資料集存取範圍、resource_id、resource_secret(審核通過後才會顯示)、介接 API 文件、資料檔解析規則文件。
選填欄位	介接 API 文件。
功能動作	啟用／停用資料集、上傳介接 API 文件、上傳資料檔解析規則文件、取消修改、提交審核。

欄位介面示意：

四、資料集運用情形

資料提供者需檢視各啟用之資料集現行運用之情形時，可使用「被運用資料集」功能項目，將以清單顯示介接資料集之服務提供者與相對應服務名稱，並提供依資料集名稱篩選及服務提供者、註冊服務關鍵字搜尋功能。

五、查詢所有服務列表

提供查詢 MyData 專區已註冊服務提供者與服務清單，並提供服務連結（服務提供者服務申辦頁面）。

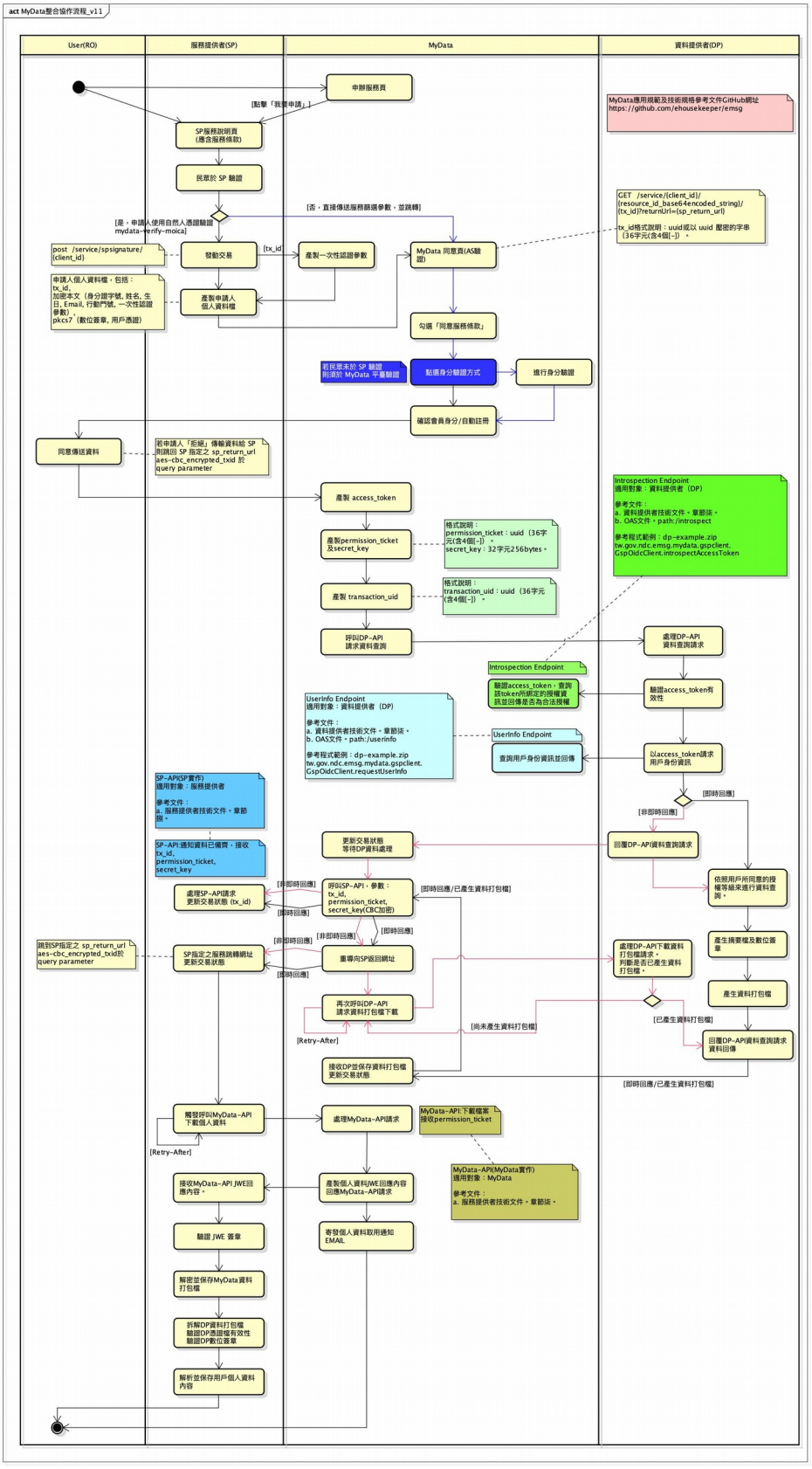
陸、**MyData** 整合協作流程說明

MyData 平臺提供多種身分驗證方式，包括：自然人憑證、健保卡...等，MyData 平臺使用民眾用戶的個人身分證字號+生日做為用戶歸戶的依據，民眾用戶則需同意授權 MyData 平臺，可以自資料提供者取得用戶自己的個人資料。MyData 平臺會將用戶同意授權後產生的 access_token 傳遞給資料提供者，資料提供者可透過 Introspection Endpoint, UserInfo Endpoint 向 MyData 平臺發出 access_token 驗證請求，以檢核 access_token 之有效性。

本文件主要對象為提供資料提供者參考，為避免混淆，僅著重描述資料提供者何時請求呼叫以下 API：

- Introspection Endpoint
- UserInfo Endpoint

一、MyData 整合協作流程說明



註：流程說明圖檔案可至下述 **Github** 連結下載、瀏覽。

https://github.com/ehousekeeper/emsg/blob/master/MyData 服務說明、應用規範與技術文件/MyData 整合協作流程_V2.0.jpg

二、應用範圍

（一）檢核用戶同意授權的有效性

當 MyData 平臺向資料提供者發出資源存取請求時，會一併傳遞民眾用戶同意授權的憑據 access_token 給資料提供者，資料提供者需檢核這個 access_token 是否有效，若有效，才接續以 access_token 取得用戶資訊。

access_token 範例：
mydata::fd05257048b303d2bad3cc4aa6313290eb80654e554054ea8e2fc88ff516b15d

資料提供者可利用 MyData Introspection Endpoint 來檢核 access_token 是否為合法有效，請參閱章節「柒、二、Introspection Endpoint」。

（二）取得用戶資訊

當資料提供者已經確認 access_token 為合法有效後，接下來可以呼叫 MyData UserInfo Endpoint 來取得用戶資訊。取得用戶資訊的主要目的是為了識別用戶身分，資料提供者以身分證字號做為識別用戶的主要依據。

資料提供者取得用戶資訊的方法，請參閱章節「柒、三、UserInfo Endpoint」。

柒、授權 API Endpoint 規格說明

一、系統環境與 API Endpoint

所有的 API endpoint 皆以 RESTful Service 方式提供介面，且皆基於 TLS v1.2 提供加密傳輸管道。

目前請使用原 GSP 的網址，預計 6 月份改用 MyData 網址（會再行更新文件）。

請參考章節「拾貳、二、系統環境主機及網址資訊」。

二、Introspection Endpoint

Introspection Endpoint 主要功能，是讓資料提供者可利用於檢核 MyData access_token 的有效性。

（一）Introspection 請求

Introspection Endpoint 支援 HTTP POST 呼叫，參數的傳遞方式為 application/x-www-form-urlencoded，同時需以 HTTP Basic authentication 方式帶入身分驗證資訊。

Credential 的形式為將 resource_id 及 resource_secret 以冒號（:）合併為一個字串後（如：resource_id:resource_secret）再以 Base64 編碼，並將編碼後的字符串放置於 http header 中。

如：
Authorization: Basic czZCaGRSa3F0MzpnWDFmQmF0M2JW

請求網址示意：

```
POST /connect/introspect
HTTP/1.1 TLS 1.2
Content-Type: application/x-www-form-urlencoded
Authorization:Basic {credential}

token={access_token}
```

參數/欄位說明：

參數/欄位	說明
credential	resource_id 及 resource_secret 以冒號:合併為一個字串後再以 Base64 編碼後的字符串。
access_token	代表用戶同意授權之 token。

Java Code Example：

```
List< NameValuePair> pairList = new ArrayList< > ();
pairList.add(new BasicNameValuePair("token", accessToken));
CloseableHttpClient httpClient = HttpClientBuilder.create().build();
HttpPost post = new HttpPost(config.getIntrospectionEndpoint());

post.setEntity(new StringEntity(
        URLEncodedUtils.format(pairList, "UTF-8")));

post.addHeader(
        "Accept",
        "application/json");

post.addHeader(
  "Content-Type",
  "application/x-www-form-urlencoded; charset= UTF-8");
post.addHeader(
        "Authorization",
        "Basic "+ basicAuthenticationSchema(resourceId,resourceSecret));

IntrospectEntity introspectEntity = null;
HttpResponse response = httpClient.execute(post);

private String basicAuthenticationSchema(
```

```
        String resourceId,
        String resourceSecret) {
        StringBuilder sb =  new StringBuilder();
        sb.append(resourceId)
          .append(":")
          .append(resourceSecret);
        try {
            return encoder
                    .encodeToString(sb.toString()
                    .getBytes());
        } catch (Exception e) {
            e.printStackTrace();
            return "";
        }
}
```

（二）Introspection 請求 - 回覆成功

回覆請求示意：

```
HTTP/1.1 TLS 1.2 200 OK
Content-Type: application/json
Cache-Control: no-store
Pragma: no-cache

{
  "active": ""
}
```

參數/欄位說明：回應欄位數目未來可能因需求而增加，接收時請考量擴充性。

參數/欄位	說明
active	必要。 指示是否所呈現的 access_token 當前處於活動狀態。

（三）Introspection 請求 - 回覆失敗

回覆失敗網址示意：

```
HTTP/1.1 TLS 1.2 400 Bad Request
Content-Type: application/json
Cache-Control: no-store
Pragma: no-cache

{
  "error": "invalid_request"
}
```

http header：

參數/欄位	說明
Content-Type	application/json
Cache-Control	no-store
Pragma	no-cache

參數/欄位說明：

參數/欄位	說明
error	必要。錯誤代碼。 invalid_request：缺少必要參數。 invalid_client：client 身分驗證失敗。 invalid_grant：非合法授權或已過期。 unauthorized_client：未授權的 client。
error_description	非必要。錯誤描述。
error_uri	非必要。錯誤描述頁面網址。

三、UserInfo Endpoint

（一）UserInfo 請求

使用 HTTP GET 方法來進行請求，並採用 Bearer token 進行身分驗證。

請求網址示意：

```
GET /connect/userinfo
HTTP/1.1 TLS 1.2
Authorization: Bearer {access_token}
```

http header：

參數/欄位	說明
access_token	用戶同意授權 token。

Java Code Example：

```
CloseableHttpClient httpClient = HttpClientBuilder.create().build();
HttpGet get = new HttpGet(config.getUserinfoEndpoint());
get.addHeader("Content-Type", "application/json");
get.addHeader("Authorization", "Bearer "+ accessToken);

HttpResponse response = httpClient.execute(get);
UserInfoEntity userInfo = null;
if(response.getStatusLine().getStatusCode() == HttpStatus.SC_OK) {
  String responseString =
  EntityUtils.toString(response.getEntity(), "UTF-8");
  if(StringUtils.isNotEmpty(responseString)) {
     ObjectMapper om = new ObjectMapper();
     userInfo = om.readValue(
       responseString,UserInfoEntity.class);
  }
}else {
  response.getStatusLine().getStatusCode());
}
```

（二）回覆 UserInfo 請求成功

UserInfo 實際回傳的欄位若少於規範已列示的欄位，以不出現該欄位為原則，而非將該欄位值付予 null 或是空字串。

回覆請求示意：

```
HTTP/1.1 TLS 1.2 200 OK
Content-Type: application/json

{
        "sub": "",
        "cn": " 王小明 ",
        "uid": " 身分證字號 ",
        "uid_verified": "True|False 身分證字號是否已驗證 ",
        "birthdate": "1973/07/14",
        "gender": "M|F 性別 ",
        "email": "janedoe@example.com",
        "account": "MyData 帳號 "
}
```

參數/欄位說明：

參數/欄位	說明
sub	必要。用來代表帳戶的唯一識別值。
cn	非必要。中文姓名。
uid	必要。身分證字號。
uid_verified	非必要。身分證字號是否已驗證。
birthdate	必要。生日，格式為 YYYY-MM-DD 八碼。
gender	非必要。性別 male/female。
email	非必要。電子郵件。

（三）回覆 UserInfo 請求失敗

回覆請求示意：

```
HTTP/1.1 TLS 1.2 401 Unauthorized
WWW-Authenticate: error= "invalid_token",
error_description= "The access token expired"
```

參數/欄位說明：

參數/欄位	說明
error	錯誤代碼。 invalid_request： 缺少必要參數、提供了不支援的參數、提供了錯誤的參數值、同樣的參數出現多次、使用一種以上的方法來出示 access_token（如放在 header 裡又放在 form 裡）、或是其他無法解讀 request 的情況。 invalid_token： access_token 過期、被收回授權、無法解讀、或其他 access_token 不合法的情況。
error_description	錯誤說明。

捌、**DP-API Endpoint** 規格準則

一、系統環境與條件

API endpoint 以 RESTful Service 方式提供介面，且皆基於 TLS v1.2 以上提供加密傳輸管道。

二、DP-API 請求及回覆規格說明

（一）DP-API 請求（由 MyData 發動請求）

```
POST /mydata-dp/{resource}
HTTP/1.1 TLS 1.2
Content-Type: {content_type}
Authorization: Bearer {access_token}
transaction_uid: {transaction_uuidv4_string}
{custom_param1_key}: {custom_param1_value}
{custom_param2_key}: {custom_param2_value}
```

參數說明：

參數	說明
resource	用來識別資料集的字符串，DP 自行決定即可。
content_type	application/zip 代表請求回覆「資料打包 zip 檔」
access_token	代表用戶同意授權的 token
transaction_uid	交易鍵值。用於讓 DP 方便識別資料查詢請求為同一次交易。 交易鍵值有效期為，始自第一次發動 DP-API 請求後，到 MyData 取得 DP 回覆資料檔後或 DP 回覆請求失敗或查無資料後止。 格式為 UUID v4 字符串。
custom_param1_key custom_param2_key	用戶填入的自訂參數鍵。 若需多個自訂參數，則分別帶入，不限於 2 個。例：carNo
custom_param1_value custom_param2_value	用戶填入的自訂參數值。 若需多個自訂參數，則分別帶入，不限於 2 個。例：1234-QQ 每多一組自訂參數即帶入一組 key: value。 例：carNo: 1234-QQ

部份的 DP 查詢資料之必要條件，除了用戶身分證字號外（利用 UserInfo Endpoint 取得），尚需帶入其它條件值（用戶自行於 MyData 網頁上填入的必要查詢條件）。為了避免條件值外漏於 GET request url 中，MyData 呼叫 DP-API 時，會將查詢條件值放置於 HTTP Header 中。

預設是以 POST 呼叫 DP-API，若 DP 希望以 GET 呼叫請求，則直接比照 POST 的做法，並備註於 OAS 文件即可。

```
HTTP/1.1 TLS 1.2 200 OK
Content-Type: {content_type}
Content-Disposition: attachment; filename={filename}
Content-Transfer-Encoding: binary
Accept-Ranges: bytes
```

（二）DP-API 請求 – 回覆成功，即時回應

依據請求的 content_type 回應該格式的檔案。filename 中註明檔案名稱。

參數說明：

參數	說明
content_type	application/zip 代表回覆「資料打包 zip 檔」
filename	檔案名稱

（三）DP-API 請求 – 回覆成功，等候處理

```
HTTP/1.1 TLS 1.2 429 Too Many Requests
Content-Type: {content_type}
Retry-After: {delay_seconds}
```

若 DP-API 不能即時回應請求，則以 HTTP 429 回應。

參數說明：

參數	說明
content_type	application/zip 代表請求回覆「資料打包 zip 檔」

delay_seconds	下次發動請求前需等待的秒數。

（四）DP-API 請求 – 回覆成功，查無資料

依據請求的 content_type 回應該格式的檔案。filename 中註明檔案名稱。

參數說明：

參數	說明
content_type	application/zip 代表回覆「資料打包 zip 檔」
filename	檔案名稱

依據請 pdf 設計格式與標準應如正常回復 pdf 檔並於內文標明「查無資料」；json 檔內容請請回覆：

{“code”：“204”,”text”:” 查無資料”}

（五）DP-API 請求 – 回覆失敗

```
HTTP/1.1 TLS 1.2 401 Unauthorized
Content-Type: application/json
```

DP 以 HTTP 狀態碼來表示回覆請求失敗的狀況。

HTTP 狀態碼	說明
400	缺少必要參數或參數驗證失敗。
401	access_token 檢核失敗。
403	拒絕存取。
504	無法完成傳送個人資料檔案。

三、**DP-API Heartbeat** 機制說明

DP-API Heartbeat 機制的設計目的，是為了令 MyData 平台確認 DP-API 仍有效存在。目前 MyData 平台預設以每一小時呼叫一次的頻率來確認 DP-API Endpoint 的狀態。

（一）MyData 發出 heartbeat 請求

```
GET /mydata-dp/{resource}?heartbeat= true
HTTP/1.1 TLS 1.2
```

（二）DP 回覆 heartbeat 請求成功

```
HTTP/1.1 TLS 1.2 200 OK
```

DP-API 回應 HTTP 200 即代表 API Endpoint 仍有效存在。

（三）DP 回覆 heartbeat 請求失敗

DP-API 回應 HTTP 狀態碼為非 200 時，MyData 平台即視為 heartbeat 回應失敗。

當 DP-API 發生連線逾時，MyData 平台即視為 heartbeat 回應失敗。

玖、**DP** 資料打包檔案規格準則

一、**DP** 資料打包檔案規格說明

為了使 MyData 的應用更加廣泛，同一份資料可能提供多種格式，須包括至少一種機器可讀的格式，如：json，以及至少一種易於人讀的格式，如：pdf，並須以民眾身分證字號加密 pdf 檔。

其中，PDF 檔之內容須符合正式文件之規範，包括：資料提供者之機關 LOGO、浮水印等可證明文件出處之標誌，以示該機關提供文件之公正性，示意如下圖：

此外，資料檔也須包括保證資料未經竄改的數位簽章檔，及為了方便服務提供者進行驗簽的憑證檔。

為了簡化 MyData 平臺與資料提供者之間傳遞資料的機制，MyData 平臺希望資料提供者可以將同一份資料的多個檔案打包成為一個壓縮 zip 檔案後再傳遞給 MyData 平臺，因此 MyData 平臺規範了用戶個人資料檔案的打包規則，同時也使服務提供者在處理來自不同資料提供者的檔案時，可以有一致性的處理規則及做法。

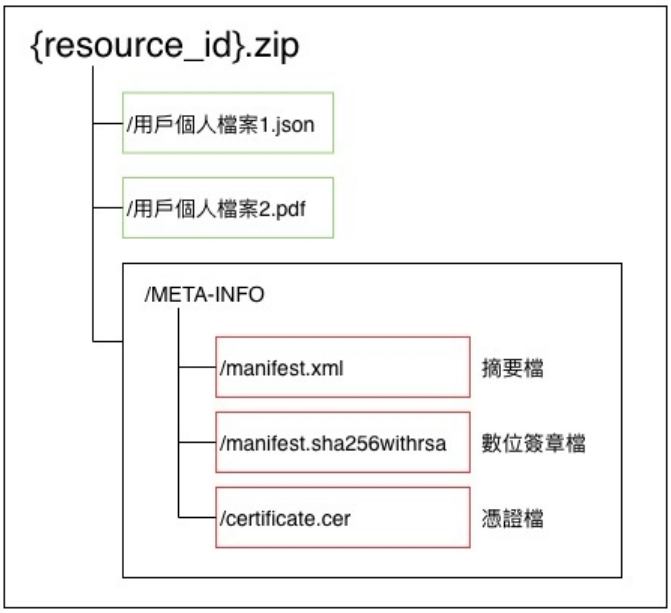
（一）DP 資料打包檔案規格要點如下：

1. DP 資料打包檔格式為壓縮 zip 檔，因 MyData 會提供線上預覽功能，此 zip 檔請勿設定密碼。
2. zip 檔的檔名非限定但建議為 {resource_id}.zip。resource_id 是變數，代表資料集的識別代碼。
3. zip 檔中可以包含多個個人資料檔案，依資料提供者規範需包含的檔案用途、數量、檔名、副檔名等。
4. 將數位簽章置於 META-INFO 子目錄。

承上，META-INFO 子目錄中包含檔案：

- manifest.xml 摘要檔
- manifest.sha256withrsa 數位簽章檔
- certificate.cer 憑證檔

（二）DP 資料打包檔案目錄結構示意如下：



（三）META-INFO 目錄及內含檔案說明：

META-INFO 目錄下放置摘要檔、數位簽章檔及憑證檔。

manifest.xml：

針對各別的資料檔案，以 SHA256 演算法，演算出的數位指紋（摘要值）後載明於 manifest.xml 檔案中。

內容格式示意如下：

```
<?xml version="1.0" encoding="UTF-8"?>
<files>
 <file>
  <filename> 用戶個人資料檔案 1.json</filename>
  <digest>{digest value}</digest>
 </file>
 <file>
  <filename> 用戶個人資料檔案 2.pdf</filename>
  <digest>{digest value}</digest>
 </file>
</files>
```

manifest.sha256withrsa：

以 SHA256 演算出 manifest.xml 的數位指紋後，以 DP 的 RSA 私鑰進行加密演算後所得的二進位內容，以副檔名 sha256withrsa 來示意所使用的演算機制為 SHA256withRSA。

建議 DP 使用長度至少 2048bits 的私鑰，向 CA 申請數位簽章憑證時，須申請支援 SHA256 的憑證。

certificate.cer：

憑證檔。PEM 格式的憑證資訊。PEM 格式的檔案是 ASCII (base64) 檔案，內容包含前置及後置文字，如下示意：

```
-----BEGIN CERTIFICATE-----
MIID/
zCCAuegAwIBAgIJAMhtYm3fde9AMA0GCSqGSIb3DQEBCwUAMIGVMQswCQY
D
-----END CERTIFICATE-----
```

二、SP 驗證 DP 資料打包沒有被竄改的方法說明

（一）驗證憑證檔的有效性

服務提供者 SP 向簽發憑證的 CA 驗證憑證有效性。建議 DP 向 GCA 政府憑證管理中心來申請數位簽章用的憑證。

GCA 支援兩種驗證憑證有效性的方法，包括：CRL 及 OCSP。

（二）憑證檔中取出 DP 公鑰

DP 夾帶的憑證檔為 PEM 格式，SP 須從憑證檔中取出 DP 公鑰，用於後續驗證數位簽章檔案 manifest.sha256withrsa。

（三）驗證 manifest.xml 沒有被竄改

manifest.xml 檔案中載明了各別資料檔案的數位指紋（摘要值）。因此先驗證 manifest.xml 沒有被竄改，即代表 manifest.xml 檔中所載明的摘要值沒有被竄改。

manifest.sha256withrsa 是針對 manifest.xml 所做出來的數位簽章檔，SP 須以 DP 的公鑰對 manifest.sha256withrsa 進行解密後，可得到正確的 manifest.xml 的摘要值。

SP 對 manifest.xml 進行 SHA256 演算後，比較前後兩者摘要值是否相符，若相符則代表 manifest.xml 沒有被竄改。

（四）驗證各別的資料檔案沒有被竄改

SP 讀取 manifest.xml 內容後，得到各別資料檔案的正確的摘要值，再針對各別資料檔進行 SHA256 演算後，比較前後兩者摘要值是否相符，若相符則代表該資料檔案沒有被竄改。

拾、交易 Log 日誌查詢

建立 DP、MyData、SP 之間的交易勾稽機制。

說明如下：

1. 各角色勾稽必要參數說明如下：
   - DP：transaction_uid, resource_id, 交易事件代碼, 日誌產生時間, 請求來源 IP。
   - MyData ：transaction_uid, client_id, resource_id, tx_id, 交易事件代碼, 身分證字號/統一編號, 日誌產生時間, 請求來源 IP。
   - SP：client_id, resource_id, tx_id, 交易事件代碼, 身分證字號/統一編號, 日誌產生時間, 請求來源 IP。
2. 交易日誌產生時機，說明如下。

#	事件代碼	事件時機	DP	MyData	SP

1	110	民眾在 SP 做自然人憑證驗證			V
2	120	SP 請求一次性驗證參數		V	V
3	130	將壓密過的民眾的個人資料與簽章憑證傳給 MyData		V	V
4	140	SP 跳轉至 MyData 同意頁		V	V
5	150	MyData 向內政部 API 驗民眾憑證與數位簽章		V	
6	160	MyData 呼叫 ICS API		V	
7	170	MyData 呼叫生日 API		V	
8	180	民眾於 MyData 頁面完成身分驗證		V	
9	190	自動註冊帳號		V	
10	200	發送手機認證簡訊		V	
11	210	完成手機認證		V	
12	220	發送 email 認證信		V	
13	230	完成 email 認證		V	
14	240	民眾同意傳輸資料給 SP		V	
15	250	MyData 請求 DP 資料集	V	V	
16	260	DP 呼叫 Introspection API	V	V	
17	270	DP 呼叫 UserInfo API	V	V	
18	280	MyData 取得 DP 資料集	V	V	
19	290	MyData 呼叫 SP-API 通知取資料		V	V
20	300	MyData 跳轉回 SP		V	V
21	310	SP 呼叫 MyData-API 取個人資料		V	V
22	320	民眾臨櫃申辦，MyData 發送資料條碼驗證碼給民眾		V	
23	330	臨櫃人員輸入資料條碼驗證碼		V	
24	340	MyData 發送資料取用通知簡		V	

		訊/信（轉存、服務應用、條碼取用）			
25	350	MyData 刪除個人資料檔案		V	
26	360	SP 刪除個人資料檔案			V

一、DP 請求交易日誌

```
POST /log/dp
HTTP/1.1 TLS 1.2
Content-Type: application/json

requestBody:
{
  ""resource_id": "API.xxxxxxxx",
  "stime": "yyyy-mm-dd",
  "etime": "yyyy-mm-dd",
  "transaction_uid": [ “” , “” ],
  "event": [ “” , “” ],
}

responeBody:
{
  ""resource_id": "API.xxxxxxxx",
  “data” : [
     {
        "transaction_uid":"",
         "ctime":"yyyy-MM-dd hh24:MI:SS",
        "event":"",
        "ip":"",
     }
  ]
}
```

參數說明：

參數	說明

resource_id	資料集鍵值。
stime	查詢起始時間。以 tx_id 的產生時間為依據。
etime	查詢結束時間。以 tx_id 的產生時間為依據。
ctime	交易日誌產生時間。
transaction_uid	交易鍵值。用於讓 DP 方便識別資料查詢請求為同一次交易。
event	事件代碼。非必填。 第三層過濾條件，查詢結果會滿足 stime, etime, transaction_uid, event 的條件交集結果。
ip	該事件的請求來源 IP。

二、失敗回應

```
HTTP/1.1 TLS 1.2 403 Forbidden
Content-Type: application/json
```

HTTP 狀態碼	說明
400	參數格式或內容不正確，或是缺少必要參數。
401	權限錯誤。不允許此 IP 連線。
403	拒絕存取。參數（resource_id）不存在。

拾壹、**DP** 資料檔解析規則說明文件撰寫原則

一、目的

為了令 **SP** 能順利解析 **DP** 所提供的資料檔內容，針對機器可讀的文件格式，請 **DP** 提供明確的解析規則，以利 **SP** 運用。

二、資料檔解析規則說明文件檔案格式及命名原則

格式採用 **PDF** 並將資料檔解析規則命名為「**XXX_MyData** 介接資料檔案規格書**.**pdf」。另外關於資料範例部分，請依照資料檔實際格式提供範例檔案，例：example.json。

三、資料檔解析規則說明文件撰寫原則

文件格式請參考「MyData 介接資料檔案規格書(範例).odt」，下載網址如下：
https://github.com/ehousekeeper/emsg/blob/master/MyData 介接資料檔案規格書/MyData 介接資料檔案規格書(範例).odt

拾貳、**DP** 與 **MyData** 測試流程說明

一、測試流程

測試機關申請流程如下：

步驟項次	流程內容
1	機關單位以 E-mail 聯繫管理團隊，提出 MyData 管理後台使用權限申請。 （聯絡資訊 Tel:02-86925588#5555, E-mail:mydata@ndc.gov.tw）
2	管理團隊回覆機關單位申請需求，增加機關單位申請人之「我的 E 政府」帳號登入管理後台之權限（不限於公務帳號）。
3	機關單位申請人以「我的 E 政府」帳號登入管理後台並確認機關單位基本資料無誤後，使用服務提供者管理功能項目。
4	機關單位成為資料提供者，使用相關功能測試作業。

二、系統環境主機及網址資訊

提供主機及網址資訊如下，以利 DP 進行防火牆設定及測試工作。

項目	IP 或網址
正式機負載平衡 IP	117.56.91.59

正式機 AP1 IP	117.56.91.72
正式機 AP2 IP	117.56.91.73
正式機 AP3 IP	117.56.91.245
正式機後臺登入網址	https://mydata.nat.gov.tw/mydata-backend
正式機前臺首頁網址	https://mydata.nat.gov.tw
正式機 Introspection 網址（現況）	https://login.cp.gov.tw/v1/connect/introspect
正式機 UserInfo 網址（現況）	https://login.cp.gov.tw/v1/connect/userinfo
正式機 Introspection 網址（6 月份實施）	https://mydata.nat.gov.tw/connect/introspect
正式機 UserInfo 網址（6 月份實施）	https://mydata.nat.gov.tw/connect/userinfo
測試機 IP	117.56.91.143
測試機後臺登入網址	https://mydatadev.nat.gov.tw/mydata-backend
測試機前臺首頁網址	https://mydatadev.nat.gov.tw/mydata
測試機 Introspection 網址	與正式機 Introspection 網址相同。
測試機 UserInfo 網址	與正式機 UserInfo 網址相同。