```yaml
swagger: "2.0"
info:
  title: Listo Backend
  description: "
    設計為 Listo 前後端使用的 API 規格。\n
    API 旁邊有鎖頭符號的都要帶 api key 來查詢(需要登入)。
  "
  version: "1.0.0"
  #termsOfService: "http://swagger.io/terms/"
  #contact:
  #  email: "apiteam@swagger.io"
  #license:
  #  name: "Apache 2.0"
  #  url: "http://www.apache.org/licenses/LICENSE-2.0.html"
# host: "example.io"
# basePath: "/"

tags:
- name: "Common"
  description: "通用類別，大部分是未登入的使用者也會使用到的。"
- name: "Auth"
  description: "關於使用者登入、權限相關。"
- name: "User"
  description: "操作或讀寫使用者所擁有的資訊。"
- name: "Map"
  description: "探索地圖時的相關操作。"

schemes:
- "https"
- "http"

paths:
  #========================================================
  # Common
  #========================================================
  /common/get_recommand_lists/:
    get:
      tags:
```

```yaml
- "Common"
summary: "取得精選清單列表。"
description: ""
operationId: "getRecommandLists"
consumes:
- "application/json"
produces:
- "application/json"
responses:
    "200":
        description: "執行成功，回傳結果"
        schema:
            allOf:
            - $ref: "#/definitions/APIMessage"
            - properties:
                data:
                    type: "object"
                    properties:
                        lists:
                            type: "array"
                            items:
                                $ref: "#/definitions/List"
                            description: "精選清單的列表"
/common/get_hot_tags/:
    get:
        tags:
        - "Common"
        summary: "取得熱門標籤。"
        description: ""
        operationId: "getHotTags"
        consumes:
        - "application/json"
        produces:
        - "application/json"
        parameters:
        - name: "count"
          in: "query"
          type: "integer"
```

```yaml
          default: 20
          description: "抓取的標籤數量，預設值 20，最大 50"
        - name: "page"
          in: "query"
          type: "integer"
          default: 0
          description: "抓取第幾頁，預設值 0，Ex:count=10, page=2，就是抓第
11~20 個標籤"
      responses:
        "200":
          description: "執行成功，回傳結果"
          schema:
            allOf:
            - $ref: "#/definitions/APIMessage"
            - properties:
                data:
                  type: "object"
                  properties:
                    lists:
                      type: "array"
                      items:
                        $ref: "#/definitions/Tag"
                      description: "指定範圍熱門標籤的列表"
  /common/get_list/:
    post:
      tags:
      - "Common"
      summary: "取得清單中的地點。"
      description: ""
      operationId: "getList"
      consumes:
      - "application/json"
      produces:
      - "application/json"
      parameters:
      - name: "body"
        in: "body"
        schema:
```

```yaml
            type: "object"
            properties:
              list_id:
                type: "integer"
                description: "欲讀取之清單的 id"
              filter:
                type: "array"
                items:
                  type: "integer"
                default: []
                example: [1, 2]
                description: "用於篩選的標籤，tag_id 的陣列"
      responses:
        "200":
          description: "執行成功，回傳結果"
          schema:
            allOf:
            - $ref: "#/definitions/APIMessage"
            - properties:
                data:
                  type: "object"
                  properties:
                    info:
                      allOf:
                      - $ref: "#/definitions/List"
                      - description: "清單的資訊"
                    places:
                      type: "array"
                      items:
                        $ref: "#/definitions/Place"
                      description: "符合篩選條件的地點列表"
                    tags:
                      type: "array"
                      items:
                        $ref: "#/definitions/Tag"
                      description: "涵蓋所有顯示的地點之標籤列表"
  /common/search_tags/:
    get:
```

```yaml
      tags:
      - "Common"
      summary: "取得給定搜尋參數相應的 tag。"
      description: ""
      operationId: "searchTags"
      consumes:
      - "application/json"
      produces:
      - "application/json"
      parameters:
      - name: "text"
        in: "query"
        type: "string"
        required: true
        description: "欲用來搜尋的文字"
      responses:
        "200":
          description: "執行成功，回傳結果"
          schema:
            allOf:
            - $ref: "#/definitions/APIMessage"
            - properties:
                data:
                  type: "object"
                  properties:
                    tags:
                      type: "array"
                      items:
                        $ref: "#/definitions/Tag"
                      description: "符合給定搜尋字串的標籤列表"


#=======================================================
# Auth
#=======================================================
/auth/register/:
  post:
    tags:
    - "Auth"
```

```yaml
        summary: "讓使用者透過信箱與密碼進行註冊。"
        description: ""
        operationId: "register"
        consumes:
        - "application/json"
        produces:
        - "application/json"
        parameters:
        - name: "body"
          in: "body"
          schema:
            type: "object"
            properties:
              email:
                type: "string"
                description: "使用者用於登入的 e-mail"
              username:
                type: "string"
                description: "使用者的名稱"
              password:
                type: "string"
                description: "使用者的密碼"
          required: true
        responses:
          "200":
            description: "執行成功，回傳結果"
            schema:
              $ref: "#/definitions/APIMessage"
/auth/login/:
  post:
    tags:
    - "Auth"
    summary: "讓使用者登入並給予 api_key。"
    description: ""
    operationId: "login"
    consumes:
    - "application/json"
    produces:
```

```yaml
    - "application/json"
    parameters:
    - name: "body"
      in: "body"
      schema:
        type: "object"
        properties:
          email:
            type: "string"
            description: "使用者用於登入的 e-mail"
          password:
            type: "string"
            description: "使用者的密碼"
      required: true
    responses:
      "200":
        description: "執行成功，回傳結果"
        schema:
          allOf:
          - $ref: "#/definitions/APIMessage"
          - properties:
              data:
                type: "object"
                properties:
                  username:
                    type: "string"
                    description: "使用者的名稱"
                  api_key:
                    type: "string"
                    description: "使用者的驗證用  api_key"
/auth/logout/:
  post:
    tags:
    - "Auth"
    summary: "讓使用者登出。"
    description: ""
    operationId: "logout"
    consumes:
```

```yaml
    - "application/json"
    produces:
    - "application/json"
    security:
    - api_key: []
    parameters:
    - name: "body"
      in: "body"
      schema:
          type: "object"
          properties:
            email:
                type: "string"
                description: "使用者用於登入的 e-mail"
            username:
                type: "string"
                description: "使用者的名稱"
            password:
                type: "string"
                description: "使用者的密碼"
      required: true
    responses:
      "200":
        description: "執行成功，回傳結果"
        schema:
            $ref: "#/definitions/APIMessage"
#=====================================================
# User
#=====================================================
/user/get_user_lists/:
  post:
    tags:
    - "User"
    summary: "取得使用者建立的清單列表。"
    description: ""
    operationId: "get_user_lists"
    consumes:
    - "application/json"
```

```yaml
    produces:
    - "application/json"
    security:
    - api_key: []
    parameters:
    - name: "body"
      in: "body"
      schema:
        type: "object"
        properties:
          filter:
            type: "array"
            items:
              type: "integer"
            example: [1,2]
            description: "用於篩選的標籤，tag_id 的陣列"
            default: []
    responses:
      "200":
        description: "執行成功，回傳結果"
        schema:
          allOf:
          - $ref: "#/definitions/APIMessage"
          - properties:
              data:
                type: "object"
                properties:
                  lists:
                    type: array
                    items:
                      $ref: "#/definitions/List"
                    description: "使用者的清單列表"
                  tags:
                    type: array
                    items:
                      $ref: "#/definitions/Tag"
                    description: "使用者的清單中所有的 tag 之聯集"
/user/create_list/:
```

```yaml
post:
  tags:
  - "User"
  summary: "建立清單。"
  description: ""
  operationId: "create_list"
  consumes:
  - "application/json"
  produces:
  - "application/json"
  security:
  - api_key: []
  parameters:
  - name: "body"
    in: "body"
    schema:
      type: "object"
      properties:
        name:
          type: "string"
          description: "清單的名稱"
        description:
          type: "string"
          description: "清單的說明"
        coverImageURL:
          type: "string"
          description: "清單的封面圖片網址"
        privacy:
          type: "integer"
          description: "開放權限的設定值"
        places:
          type: "array"
          items:
            type: "integer"
          example: [1,2]
          description: "欲加入清單的 place_id 陣列"
  responses:
    "200":
```

```yaml
              description: "執行成功，回傳結果"
              schema:
                allOf:
                - $ref: "#/definitions/APIMessage"
                - properties:
                    data:
                      type: "object"
                      properties:
                        id:
                          type: integer
                          description: "新建立清單的 list_id"
/user/add_list_places/:
  post:
    tags:
    - "User"
    summary: "新增地點到清單。"
    description: ""
    operationId: "add_list_places"
    consumes:
    - "application/json"
    produces:
    - "application/json"
    security:
    - api_key: []
    parameters:
    - name: "body"
      in: "body"
      schema:
        type: "object"
        properties:
          list_id:
            type: "integer"
            description: "欲修改的 list_id"
          places:
            type: "array"
            items:
              type: "integer"
            example: [1,2]
```

```yaml
                    description: "欲加入清單的  place_id  陣列"
          responses:
            "200":
              description: "執行成功，回傳結果"
              schema:
                allOf:
                - $ref: "#/definitions/APIMessage"
  /user/remove_list_places/:
    post:
      tags:
      - "User"
      summary: "從清單移除地點。"
      description: ""
      operationId: "remove_list_places"
      consumes:
      - "application/json"
      produces:
      - "application/json"
      security:
      - api_key: []
      parameters:
      - name: "body"
        in: "body"
        schema:
          type: "object"
          properties:
            list_id:
              type: "integer"
              description: "欲修改的 list_id"
            places:
              type: "array"
              items:
                type: "integer"
              example: [1,2]
              description: "欲刪除的  place_id  陣列"
      responses:
        "200":
          description: "執行成功，回傳結果"
```

```yaml
            schema:
                allOf:
                - $ref: "#/definitions/APIMessage"
/user/edit_list/:
    post:
        tags:
        - "User"
        summary: "編輯清單名稱。"
        description: ""
        operationId: "edit_list"
        consumes:
        - "application/json"
        produces:
        - "application/json"
        security:
        - api_key: []
        parameters:
        - name: "body"
          in: "body"
          schema:
              type: "object"
              properties:
                name:
                    type: "string"
                    description: "清單的名稱"
                    default: ""
                description:
                    type: "string"
                    description: "清單的說明"
                    default: ""
                coverImageURL:
                    type: "string"
                    description: "清單的封面圖片網址"
                    default: ""
                privacy:
                    type: "integer"
                    description: "開放權限的設定值"
                    default: ""
```

```yaml
      responses:
        "200":
          description: "執行成功，回傳結果"
          schema:
            allOf:
            - $ref: "#/definitions/APIMessage"
  /user/modify_place_tag/:
    post:
      tags:
      - "User"
      summary: "編輯使用者對地點下的標籤。如果 gmap_id 尚未被紀錄，則
需讀取地圖資訊以新建地點。"
      description: ""
      operationId: "modify_place_list"
      consumes:
      - "application/json"
      produces:
      - "application/json"
      security:
      - api_key: []
      parameters:
      - name: "body"
        in: "body"
        schema:
          type: "object"
          properties:
            gmap_id:
              type: "integer"
              description: "欲修改標籤的地點由 GoogleAPI 定義之 placeid"
            add:
              type: array
              items:
                type: integer
              description: "新增的 tag_id 陣列"
            remove:
              type: array
              items:
                type: integer
```

```yaml
                    description: "要移除的 tag_id 陣列"
                newTags:
                    type: array
                    items:
                        type: string
                    description: "使用者新增的 tag 名稱"
        responses:
            "200":
                description: "執行成功，回傳結果"
                schema:
                    allOf:
                    - $ref: "#/definitions/APIMessage"
```

```
#=====================================================
# Map
#=====================================================
```

```yaml
/map/get_marks/:
    post:
        tags:
        - "Map"
        summary: "取得指定範圍內的所有地點。"
        description: ""
        operationId: "get_mark"
        consumes:
        - "application/json"
        produces:
        - "application/json"
        parameters:
        - name: "body"
            in: "body"
            schema:
                type: "object"
                properties:
                    from:
                        $ref: "#/definitions/Location"
                    to:
                        $ref: "#/definitions/Location"
        responses:
            "200":
```

```yaml
                    description: "執行成功，回傳結果"
                    schema:
                      allOf:
                      - $ref: "#/definitions/APIMessage"
                      - properties:
                            data:
                              type: object
                              properties:
                                marks:
                                  type: array
                                  items:
                                    $ref: "#/definitions/Mark"
                                  description: "要顯示在地圖上的圖標點列表"
/map/get_place_info/:
  post:
    tags:
    - "Map"
    summary: "獲得地圖上地點的資訊。"
    description: ""
    operationId: "get_place_info"
    consumes:
    - "application/json"
    produces:
    - "application/json"
    parameters:
    - name: "body"
      in: "body"
      schema:
        type: "object"
        properties:
          gmap_id:
            type: integer
            description: "GoogleAPI 定義的 placeid"
    responses:
      "200":
        description: "執行成功，回傳結果"
        schema:
          allOf:
```

```yaml
              - $ref: "#/definitions/APIMessage"
              - properties:
                  data:
                    type: object
                    properties:
                      name:
                        type: string
                        description: "地點的名稱(由 GoogleAPI 提供)"
                      phone:
                        type: string
                        description: "地點的電話(由 GoogleAPI 提供)"
                      address:
                        type: string
                        description: "地點的地址(由 GoogleAPI 提供)"
                      type:
                        type: string
                        description: "地點的類型(由 GoogleAPI 提供)"
/map/get_place_tag/:
  post:
    tags:
    - "Map"
    summary: "獲得使用者已對此地點上的 tag。"
    description: ""
    operationId: "get_place_tag"
    consumes:
    - "application/json"
    produces:
    - "application/json"
    security:
    - api_key: []
    parameters:
    - name: "body"
      in: "body"
      schema:
        type: "object"
        properties:
          gmap_id:
            type: integer
```

```yaml
                    description: "GoogleAPI 定義的 placeid"
        responses:
          "200":
            description: "執行成功，回傳結果"
            schema:
              allOf:
              - $ref: "#/definitions/APIMessage"
              - properties:
                  data:
                    type: object
                    properties:
                      tags:
                        type: array
                        items:
                          $ref: "#/definitions/Tag"
                        description: "使用者已對此地點上的 tag。"
/map/search_tag/:
  post:
    tags:
    - "Map"
    summary: "根據使用者輸入的字串搜尋相關 tag。"
    description: ""
    operationId: "search_tag"
    consumes:
    - "application/json"
    produces:
    - "application/json"
    security:
    - api_key: []
    parameters:
    - name: "body"
      in: "body"
      schema:
        type: "object"
        properties:
          gmap_id:
            type: integer
            description: "GoogleAPI 定義的 placeid"
```

```yaml
            type:
                type: string
                description: "地點的類型(由 GoogleAPI 提供)"
            text:
                type: string
                description: "使用者輸入的字串"
    responses:
        "200":
            description: "執行成功，回傳結果"
            schema:
                allOf:
                - $ref: "#/definitions/APIMessage"
                - properties:
                    data:
                        type: object
                        properties:
                            tags:
                                type: array
                                items:
                                    $ref: "#/definitions/Tag"
                                description: "搜尋所得之 tag 結果"
/map/candidate_tag/:
    post:
        tags:
        - "Map"
        summary: "根據地點搜尋相關 tag。"
        description: ""
        operationId: "candidate_tag"
        consumes:
        - "application/json"
        produces:
        - "application/json"
        security:
        - api_key: []
        parameters:
        - name: "body"
            in: "body"
            schema:
```

```
                type: "object"
                properties:
                    gmap_id:
                        type: integer
                        description: "GoogleAPI 定義的 placeid"
                    type:
                        type: string
                        description: "地點的類型(由 GoogleAPI 提供)"
                    limit:
                        type: integer
                        default: 20
                        description: "回傳數量的最大上限"
            responses:
                "200":
                    description: "執行成功，回傳結果"
                    schema:
                        allOf:
                        - $ref: "#/definitions/APIMessage"
                        - properties:
                            data:
                                type: object
                                properties:
                                    tags:
                                        type: array
                                        items:
                                            $ref: "#/definitions/Tag"
                                        description: "搜尋所得之 tag 結果"


securityDefinitions:
    api_key:
        type: "apiKey"
        name: "api_key"
        in: "header"
definitions:
    APIMessage:
        type: "object"
        properties:
```

```yaml
      status:
        type: "integer"
        format: "int32"
        description: "API 執行的狀態，1 為正常，其餘錯誤代碼將各自定義。"
        example: 1
      msg:
        type: "string"
        description: "後端發生錯誤或其它需要傳送給前端的訊息。"
Tag:
  type: "object"
  properties:
    id:
      type: "integer"
      format: "int32"
      description: "在資料庫的唯一 ID"
    type:
      type: "integer"
      description: "分類用的代碼"
      format: "int32"
    name:
      type: "string"
      description: "用於顯示的名稱"
Place:
  type: "object"
  properties:
    id:
      type: "integer"
      format: "int32"
      description: "在資料庫的唯一 ID"
    gmap_id:
      type: "integer"
      format: "int32"
      description: "GoogleAPI 對於這個地點的 placeid"
    name:
      type: "string"
      description: "地點的名稱(由 GoogleAPI 提供)"
    phone:
      type: "string"
```

```yaml
        description: "地點的電話(由 GoogleAPI 提供)"
      address:
        type: "string"
        description: "地點的地址(由 GoogleAPI 提供)"
      type:
        type: "string"
        description: "地點的類型(由 GoogleAPI 提供)"
      location:
        $ref: "#/definitions/Location"

Location:
  type: "object"
  description: "包裝經緯度的物件"
  properties:
    lon:
      type: "number"
      format: "float"
      description: "經度的數值(由 GoogleAPI 提供)"
    lat:
      type: "number"
      format: "float"
      description: "緯度的數值(由 GoogleAPI 提供)"
List:
  type: "object"
  properties:
    id:
      type: "integer"
      format: "int32"
      description: "在資料庫的唯一 ID"
    user_id:
      type: "integer"
      format: "int32"
      description: "建立此清單的使用者 ID"
    privacy:
      type: "integer"
      format: "int32"
      enum:
      - 0
```

```yaml
        - 1
        - 2
        description: "清單開放權限設定。0:公開, 1:非公開, 2:私人"
      name:
        type: "string"
        description: "清單的名稱"
      description:
        type: "string"
        description: "清單的說明"
      coverImageURL:
        type: "string"
        description: "清單的封面圖片網址"
      createdTime:
        type: "integer"
        format: "int64"
        description: "建立時間(UNIX time，單位毫秒)"
      updatedTime:
        type: "integer"
        format: "int64"
        description: "最後更新時間(UNIX time，單位毫秒)"
  Mark:
    type: "object"
    properties:
      gmap_id:
        type: "integer"
        format: "int32"
        description: "GoogleAPI 對於這個地點的 placeid"
      location:
        $ref: "#/definitions/Location"
externalDocs:
  description: "Find out more about Swagger"
  url: "http://swagger.io"
# Added by API Auto Mocking Plugin
host: virtserver.swaggerhub.com
basePath: /garyuu/Listo-Backend/1.0.0
```