

mTrip API

[Introduction](#)

[Access to the API](#)

[Security and authentication](#)

[Versioning](#)

[Methods](#)

[Guide](#)

[Travelers](#)

[Object format](#)

[Guide object](#)

[traveler Object](#)

[Destination object](#)

[Activity object](#)

[Transport object](#)

[Accommodation object](#)

[HTTP status and error codes](#)

1. Introduction

This document presents the mTrip API to manage your travelers, their itineraries and send travel apps. It gives an overlook of the different methods and specifies the request format.

2. Access to the API

To test your code you can use the sandbox environment:

<https://sandbox.api.mtrip.com>

Note that any guide or traveler added to the test environment can be deleted at any time.

When you are ready to push your code in production you can use:

<https://api.mtrip.com>

3. Security and authentication

All request to the API are made through https. Authentication to the API is done using HTTP Basic Auth, set your API key as the username and leave the password field empty. Note that the API keys are different for the sandbox and production environment. Also some endpoints may not be accessible depending on your account settings.

4. Versioning

The current version of the API is 1. API version will change whenever we make modifications to the API that are not backward compatible.

5. Methods

Guide

The guide object represent information about a trip: beside the basic information (name, dates...) it has 4 relationships: travelers, transports, destinations, accommodations.

The travelers relationship represents a set of traveler that will be participating in the trip. The transports relationship represents the different transport modes used during the trip (e.g. flight, boat, taxi...). The accommodations relationship represents the accommodations used during the trip. The destinations relationship represents the different destination visited by the travelers along with the different scheduled activities.

| http method | endpoint | description | data |
|-------------|-----------|--------------------|--|
| POST | /1/guides | create a new guide | a json containing the representation of a guide object |

Travelers

The traveler method is used to add new travelers to the CMS

| http method | endpoint | description | data |
|-------------|--------------|-----------------------|--|
| POST | /1/travelers | create a new traveler | a json containing the representation of a travelers object |

6. Object format

This section presents the different fields defining the objects. Each field has a type which is either a basic type (e.g. string, int...) , or an array of other objects (e.g. an array of travelers).

The date type is a string date format 'YYYY-MM-DDTHH:mm:ssZ' (ISO 8601).

The location type is an object with two fields, latitude takes values in range [-90,90] and longitude takes values in range [-180,180]

All string must be UTF-8 encoded.

Guide object

| Name | Description | Type | Optional |
|------------|--|---------|---|
| name | The name of the guide displayed to the travelers | string | NO |
| agency_id | The id of the agency in mtrip CMS. Only needed if the authenticated user manages multiple agencies. Agencies are created in the CMS and may not be available for your account. | integer | YES |
| identifier | The reference id of the guide in your system. Use to retrieve a guide after being added (e.g. to update the informations). | string | YES |
| start_date | the start date of the trip | date | YES (depending on your account settings it may be mandatory) |
| end_date | the end date of the trip | date | YES (depending on your account settings it may be mandatory) |

| | | | |
|----------------|---|---------------------------------|-----|
| description | A description of the trip that will be displayed to the travelers | string | YES |
| travelers | An array of travelers | [traveler] | NO |
| destinations | An array of destinations | [destination] | YES |
| transports | An array of transports | [transport] | YES |
| accommodations | An array of accommodations | [accommodation] | YES |

Example of a guide object

```
{
  "accommodations" : [accommodation1,...],
  "agency_id" : 345,
  "identifier" : "YTDF55",
  "destinations" : [destination1,...],
  "start_date" : "2014-09-23T18:00:00+00:00",
  "description" : "you will visit the best places in London...",
  "name" : "Trip To London",
  "end_date" : "2014-09-30T18:00:00+00:00",
  "transports" : [transport1,...],
  "travelers" : [traveler1,...]
}
```

traveler Object

| Name | Description | Type | Optional |
|-----------------|---|----------|---------------------|
| email | traveler's email | string | NO |
| first_name | traveler's first name | string | YES |
| last_name | traveler's last name | string | YES |
| language | traveler's language code (two letter ISO language code , e.g. en, fr, es...) | string | YES (en by default) |
| send_invitation | Specify if an invitation email is sent to the traveler after his account is created | boolean | YES (default false) |
| identifier | traveler reference id in your system | string | NO |
| guides | An array of existing guide identifiers | [string] | YES |

Example of traveler object json:

```
{
  "identifier" : "TFE45HGF",
  "email" : "john_smith@mtrip.com",
  "first_name" : "John",
  "language" : "en",
  "last_name" : "Smith",
  "send_invitation" : true
}
```

Destination object

| Name | Description | Type | Optional |
|------------|---|----------------------------|----------|
| name | Name of the destination | string | NO |
| start_date | the start date of the trip in the destination | date | YES |
| end_date | the end date of the trip in the destination | date | YES |
| activities | An array of activities | [activity] | YES |

Example of destination object json:

```
{
  "activities" : [activity1,...],
  "start_date" : "2014-09-23T18:00:00+00:00",
  "name" : "London",
  "end_date" : "2014-09-30T18:00:00+00:00"
}
```

Activity object

| Name | Description | Type | Optional |
|-------------------|---|--|--|
| name | Name of the activity | string | YES (this can be omitted if you only want to present an informative text in the "info" variable) |
| date | Date of activity | date | NO |
| period | The id of the period of the day when the activity occurs | enum{morning,afternoon,lunch,evening,dinner,day} | YES (will be computed from the date if not present) |
| type | The type id of the activity | enum{meal,coffee_break,cultural_visit,working_session,conference,not_classified} | YES |
| comments | Information about the activity displayed to the user | string | YES |
| location | Geo-location of the activity | location | YES |
| duration | Duration of the activity (in minutes) | int | YES |
| address | Address of the activity | string | YES |
| associated_places | An array of place ids (in mTrip CMS) associated with the visit. | [int] | YES |
| picture_url | A URL to an image (format must be .png or .jpeg) | string | YES |

Example of activity object json:

```
{
  "address" : "Riverside Bldg, County Hall, Westminster Bridge Rd, London SE1 7PB, United Kingdom",
  "associated_places" : [
    2342345,
    2352352,
    5654675
  ],
  "date" : "2014-09-24T09:10:00+00:00",
  "duration" : 60,
  "comments" : "This activity...",
  "location" : {
    "latitude" : 51.50344,
    "longitude" : -0.119476
  },
  "name" : "The London Eye Ultimate 4D Experience",
  "period" : "morning",
  "picture_url" : "http://upload.wikimedia.org/wikipedia/commons/d/d6/London-Eye-2009.JPG",
  "position" : 0,
  "type" : cultural_visit
}
```


Transport object

| Name | Description | Type | Optional |
|-----------------------|--|---|----------|
| transport_type | The transportation mode type | enum{Boat Bus Car Ferry Plane Shuttle Taxi Train Tram} | NO |
| reservation_reference | The booking number for the transportation | string | YES |
| airline_company | The company operating the transport | string | YES |
| transport_number | The number identifying the transport, e.g. the flight number for a flight, the train number for a train... | string | YES |
| departure_city | The city from where the transport leave | string | NO |
| arrival_city | The city to where the transport goes | string | NO |
| departure_airport | The departure airport of a flight or the departure station for other transport mode | string | YES |
| arrival_airport | The arrival airport of a flight or the arrival station for other transport mode | string | YES |
| departure_terminal | The departure terminal, only for flights | string | YES |
| arrival_terminal | The arrival terminal, only for flights | string | YES |
| departure_date | Departure date | date | NO |
| arrival_date | Arrival date | date | NO |
| comments | Information about the transport mode displayed to the traveler | string | YES |

Example of transport object json:

```
{
  "arrival_airport" : "LHR",
  "arrival_city" : "London",
  "arrival_date" : "2014-09-30T18:00:00+00:00",
  "arrival_terminal" : "G4",
  "reservation_reference" : "ERD353WFR",
  "airline_company" : "Air Canada",
  "departure_airport" : "YUL",
  "departure_city" : "Montreal",
  "departure_date" : "2014-09-23T18:00:00+00:00",
  "departure_terminal" : "A45",
  "comments" : "A taxi will be waiting for you...",
  "transport_number" : "AC 564",
  "transporttype" : "flight"
}
```

Accommodation object

| Name | Description | Type | Optional |
|--------------|---|----------|---|
| name | Name of the accommodation | string | NO (will have no effect if "mtrip_id" is present) |
| from_date | Check-in date | date | NO |
| to_date | Check-out date | date | NO |
| info | Information about the accommodation displayed to the user | string | YES |
| location | Geo-location of the accommodation | location | YES |
| mtrip_id | Accommodation id in mtrip CMS | int | YES |
| address | Address of the accommodation | string | YES (will have no effect if "mtrip_id" is present) |
| picture_url | A URL to an image in png or jpeg format | string | YES (will have no effect if "mtrip_id" is present) |
| type_of_room | The type of room booked | string | YES |
| city | Name of the city where the accommodation is located | string | NO if "mtrip_id" is not present |

Example of accommodation object json:

```
{
  "address" : "5-9 Norfolk Square, Paddington",
  "booking_number" : "ERD353WFR",
  "city" : "London",
  "from_date" : "2014-09-24T16:00:00+00:00",
  "info" : "...",
  "location" : {
    "latitude" : 51.50344,
    "longitude" : -0.119476
  },
  "mtrip_id" : 456789,
  "name" : "Cardiff Hotel",
  "to_date" : "2014-09-30T11:00:00+00:00",
  "type_of_room" : "double"
}
```

7. HTTP status and error codes

We use standard HTTP status 2XX for success and 4XX when something bad happened.

In particular 401 means that your API key is invalid and 422 suggest that the json data you sent are invalid.

If the request fails we sent a json object containing information about the cause of the failure that are useful for debugging.