



DRIVANIA

Chauffeurs

API USAGE AND INTEGRATION MANUAL



TABLE OF CONTENTS

1. INTRODUCTION.....	1
1.1 VERSION HISTORY.	1
1.2 OBJECTIVES.	2
2. PRELIMINARY CONSIDERATIONS.....	2
2.1. REGISTRATION AS INTEGRATOR	2
2.2. API CATALOG	2
2.3. INTEGRATION PHASES	2
3. API OVERVIEW.....	3
3.1. LOGIN	3
3.2. QUOTE REQUEST	3
3.3. CREATE SERVICE.	4
3.4. SPECIAL REQUEST.....	5
3.5. CONFIRM SPECIAL REQUEST.	5
3.6. GET SERVICE DETAILS.	5
3.7. GET DRIVER POSITION.	6
3.8. GET SERVICE INVOICE.	7
3.9. UPDATE SERVICE.....	7
3.10. CANCEL SERVICE.	8
3.11. GET SERVICES LIST.	9
3.12. GET FBO FROM AIRPORT	9
3.13. CLIENTS	9
3.14. CALLBACKS	10

1. INTRODUCTION.

1.1 VERSION HISTORY.

Version	Date	Description
1.0	03/07/25	API integration document

1.2 OBJECTIVES.

This document is intended as a user and integration guide for the API provided by DRIVANIA, enabling access to the booking tool.

2. PRELIMINARY CONSIDERATIONS.

2.1. REGISTRATION AS INTEGRATOR

The term “integrator” refers to DRIVANIA clients who interact with the company via the API, rather than through the platform’s web portal.

2.2. API CATALOG

DRIVANIA provides two environments: integration (sandbox) and production. Access to the integration environment is granted first, allowing the integrator to test and connect the API with their own system. Once the tests defined by DRIVANIA are successfully completed in this phase, access to the production environment is granted, enabling full interaction with DRIVANIA through the API.

API: <https://app.swaggerhub.com/apis-docs/Drivania-Chauffeurs/drivania-public-booking-api>

2.3. INTEGRATION PHASES

The API integration process is structured into five main phases, designed to ensure an efficient, controlled implementation aligned with DRIVANIA’s technical standards.

PHASE 1: DELIVERY OF DOCUMENTATION AND CREDENTIALS

DRIVANIA will provide the integrator with the technical API documentation along with the credentials required to access the integration (sandbox) environment.

PHASE 2: PROJECT PLANNING

The integrator and DRIVANIA will jointly agree on a detailed timeline covering both the integration and testing phases. This schedule will serve as a reference to ensure a smooth and well-organized implementation process.

During the integration phase, a weekly meeting will be held to review progress and address any issues. These meetings should take place on the same day and at the same time each week.

The estimated duration of the full integration process, including testing, is approximately **one month**.

PHASE 3: INTEGRATION DEVELOPMENT

At this stage, the integrator will implement the API within their own system.

Technical support from DRIVANIA will be provided primarily via email. If necessary, a Microsoft Teams chat channel can be made available for additional support, subject to availability.

Technical assistance hours are Monday to Friday, from 06:00 to 15:00 (UTC).

PHASE 4: VALIDATION AND TESTING

Once the integrator has completed their implementation and run internal tests, a joint End-to-End testing session will be scheduled.

DRIVANIA will provide a Test Plan, which must be fully completed during a single session. If any issues are identified, the entire Test Plan will need to be repeated in a new session.

Once DRIVANIA validates that the integration meets all functional and technical requirements, the process may proceed to the final phase.

PHASE 5: GOING LIVE (PRODUCTION DEVELOPMENT)

A date will be agreed for migration to the production environment. DRIVANIA will deliver the corresponding production credentials and a final validation session will be conducted.

Once this final validation is approved, the integration will be considered live and operational.

Important: From this point on, testing in the production environment is strictly prohibited. All operations performed will be considered real and will take immediate effect.

3. API OVERVIEW.

This section provides a summary of the available API methods along with a brief explanation of their purpose. Before diving into the details, it is essential to understand a few key concepts that will help in using and interpreting the API operations more effectively.

KEY CONCEPTS

Quotation

A quotation refers to a set of vehicle options that DRIVANIA offers for a specific service request. Quotations are temporary and come with an expiration date. It is important to note that a quotation does not imply any commitment on the part of DRIVANIA.

Service

The receipt of a *service_reference* in the response confirms that the service has been successfully registered in DRIVANIA's system. If the service was created from a previously issued quotation, it is considered automatically confirmed.

Special Request

This refers to a real service request that cannot be quoted automatically. In such cases, the DRIVANIA team must manually review the request to generate a quotation. Once available, the integrator will be notified via callback and can proceed to confirm the service.

Clients

Agencies may register clients in the system in order to request services on their behalf.

3.1. LOGIN

POST */login*

This endpoint allows you to obtain an authentication key required to access the other API methods.

The response includes an access token, which is valid for a limited time as specified by the *ttl* (time-to-live) field. This field indicates the token's expiration time.

3.2. QUOTE REQUEST

POST */quote-requests*

This endpoint allows you to request a quotation from DRIVANIA for a specific service.

The request must include the following information:

- Type of service
- Pickup location
- Drop-off location
- Date and time of the service
- Number of passengers

Each request generates a unique service identifier (*service_id*). If DRIVANIA is able to offer quotations for the requested service, the response will include an array of quotations (*quotes*), each containing an array of vehicles (*vehicles*). Each vehicle is identified by a *vehicle_id* and includes detailed information such as:

- Vehicle type and category
- Price
- Cancellation policy
- Other relevant attributes

If it is not possible to return an automatic quotation, the *vehicles* array will be null, and the *unavailable_reason* field will explain why the service could not be quoted.

Depending on the commercial agreement between the integrator and DRIVANIA, it may be possible to request special services. In these cases, even though a *service_id* is generated, DRIVANIA will not be able to provide an immediate quotation and the request will be handled as a Special Request.

3.3. CREATE SERVICE.

POST */services*

This endpoint is used to confirm a previously quoted service, providing the additional information required for its execution.

The request must include:

- Main passenger details
- Flight information
- Number of child seats and boosters
- Client reference (for internal tracking)
- Contact configuration for service notifications

Once the service has been successfully created, the response will include:

- DRIVANIA's unique service reference (*service_reference*)
- The status confirmed

From this point on, the service is considered confirmed, and the *service_reference* will be required for any further operations related to this service.

SERVICE NOTIFICATIONS

It is possible to configure contacts to receive updates regarding the status of the service, via email or SMS. Each contact may have one or more assigned permissions, which determine the type of notifications they will receive. The available permissions are:

- **chauffeur_details**

Email: Includes vehicle and driver information.

SMS: Sent on the day of the service and includes the driver's name and phone number, the scheduled time, and DRIVANIA's 24/7 support line.

- **chauffeur_on_the_way**
Notifies that the driver is en route to the pickup location.
- **chauffeur_on_location**
Indicates that the driver has arrived at the pickup location and is waiting for the passenger.
- **passenger_on_board**
Confirms that the passenger is onboard. If geolocation is enabled for the service, the estimated time of arrival (ETA) at the destination is also included.
- **vehicle_eta_to_destination**
This notification is only sent if, 20 minutes before the end of the service (based on the ETA provided in `passenger_on_board`), a significant change in arrival time is detected. The updated ETA is included. This notification is not sent for hourly (disposition) services.
- **passenger_dropoff**
Confirms that the passenger has arrived at the destination.

3.4. SPECIAL REQUEST.

POST */special-requests*

This endpoint is used to request special services for which DRIVANIA cannot provide an automatic quotation.

These requests must be manually reviewed and quoted by DRIVANIA's operations team.

To initiate the process, the `service_id` corresponding to the desired service must be submitted. Once the request has been processed, DRIVANIA will notify the integrator via callback with the available quotation, allowing the service to be confirmed afterwards.

3.5. CONFIRM SPECIAL REQUEST.

POST */special-requests/{service_reference}/confirm*

This endpoint is used to confirm a special service that has been manually quoted by the DRIVANIA team.

Once the quotation is received via callback, the integrator can confirm the service using this method.

To do so, the request must include the service reference (`service_reference`) along with the complete service details, following the same structure required in the standard service creation endpoint (POST */services*), as described in section 3.3.

3.6. GET SERVICE DETAILS.

GET */services/{service_reference}*

This endpoint is used to retrieve detailed information about a service—from the moment it is quoted or requested until its completion. It provides visibility into the status of both the service and the vehicle, as well as driver and vehicle details once they become available, including pickup and drop-off points, dates, flight numbers, and more.

- The response includes:
- Service and vehicle status
- Driver and vehicle information (once assigned)
- Pickup and drop-off locations

- Dates and times
- Flight numbers
- Passenger details
- Cancellation policy
- Selling price
- And other less critical data

SERVICE STATUS

Throughout the lifecycle of a service, the following statuses may be received

- **Requested**
DRIVANIA was unable to generate an automatic quotation and has initiated a Special Request. This status may also appear after a modification that requires manual intervention. The service will remain in this state until it is either set back to Confirmed or resolved through client interaction.
- **Quoted**
The service has been successfully quoted and is ready to be confirmed by the integrator
- **Confirmed**
The service has been confirmed successfully.
- **Cancelled**
The service has been cancelled.
- **In service**
The service has started. This status indicates that the driver is en route to the pickup location, although the passenger may not yet be on board.
- **Invoicing**
The service has been completed, but the invoice is not yet available.
- **Completed**
The service has been finalized and the invoice has been issued.

VEHICLE STATUS

During service execution, the following vehicle statuses may be received:

1. **On my way**
The driver is en route to the pickup location.
2. **Chauffeur on location**
The driver has arrived at the pickup location and is waiting for the passenger.
3. **Passenger on board**
The passenger is in the vehicle and the trip has begun.
4. **Passenger dropoff**
The passenger has arrived at the destination.

3.7. GET DRIVER POSITION.

GET */services/{service_reference}/driver-position*

This endpoint provides the real-time geographic location of the driver assigned to a service.

Important: This method can only be used when the service status is In service, meaning the driver has already started the trip toward the pickup location.

3.8. GET SERVICE INVOICE.

GET */services/{service_reference}/invoice/{invoice_number}*

This endpoint allows you to download the PDF invoice corresponding to a service, once it has reached the Completed status.

To perform the download, you must have the invoice number (*invoice_number*), which can be retrieved beforehand by using the GET Method */services/{service_reference}* endpoint.

3.9. UPDATE SERVICE.

Primary method: PUT */services/{service_reference}*

Additional method (if applicable): POST */services/{service_reference}/changes/{change_id}*

Updating a service may require one or two calls, depending on the type of modification being made.

Direct update

You must always start with a call to the PUT */services/{service_reference}* endpoint.

If the submitted changes do not affect the cost of the service and no comments are included, the modification will be confirmed automatically.

Update requiring a new quotation

If the changes result in a price variation, the response will include a new quotation along with a change identifier (*change_id*).

In this case, the modification must be confirmed using the following endpoint:

POST */services/{service_reference}/changes/{change_id}*

Changes that require a new quotation

- Type of service
- Pickup location
- Drop-off location
- Date and time of the service
- Number of passengers (when exceeding the capacity of the confirmed vehicle)
- Vehicle type (*vehicle_type*)
- Changes that include comments.

If the update includes comments, they must be manually reviewed by the DRIVANIA team before the change can be confirmed.

Vehicle type – only change

If the only desired modification is the vehicle type, you must submit the *vehicle_type* field.

Even if no other changes require a new quotation, the API will return a new offer based on the requested vehicle type—unless it matches the one already confirmed.

Changes not supported via API

Certain modifications cannot be processed automatically. In these cases, the API will return a message indicating that the integrator must contact DRIVANIA directly to handle the request. These exceptions are described in the next section.

3.9.1 UPDATE LIMITATIONS.

There are certain situations in which changes cannot be processed automatically through the API. These limitations vary depending on the type of service and the nature of the requested modification.

NORMAL SERVICES

For normal services, most changes can be handled online. However, if an attempt is made to apply a drastic modification—for example, changing a service originally requested in Barcelona to a new location such as Paris—the API will reject the request. In such cases, it is necessary to contact DRIVANIA to handle the update manually.

SPECIAL REQUEST SERVICES

In the case of special request services, limitations depend on the quotation status.

Before manual quotation.

As long as the DRIVANIA team has not yet provided a quotation, changes can be made without restrictions.

After manual quotation.

Once DRIVANIA has processed the request and provided a quotation, the following limitations apply:

- **Change of pickup location**
The new pickup point must not be more than 100 km from the original location. If this distance is exceeded, the modification must be handled manually.
- **Increase in number of passengers**
If the assigned vehicle is a Sedan, Minivan, SUV, or Eco Sedan, and the new passenger count exceeds the vehicle's capacity, the change will be rejected and must be handled offline. For all other vehicle types, any increase in the number of passengers must be processed manually, without exception.

3.10. CANCEL SERVICE.

Cancelling a service may require one or two API calls, depending on whether the cancellation incurs any associated charges.

CANCELLATION WITHOUT CHARGES

If the service is cancelled with sufficient notice and no fees apply, a single call to the following endpoint is enough:

DELETE /services/{service_reference}

The cancellation will be processed immediately, and no additional confirmation will be required.

CANCELLATION WITH CHARGES

If the cancellation involves costs (e.g., penalties of 50% or 100%), the call to the DELETE method will return:

- A cancellation identifier (cancellation_id)
- The cancellation fee amount (cancellation_fee)
- Unconfirmed status.

In this case, the cancellation must be explicitly confirmed using the following endpoint:

PUT /services/{service_reference}/cancellations/{cancellation_id}

CANCELLATION POLICY

Information on whether a cancellation incurs charges is available in the `cancellation_policy` field, which is included in the response of the `GET /services/{service_reference}` endpoint.

3.11. GET SERVICES LIST.

`GET /service-references`

This endpoint retrieves the full list of services associated with the authenticated client.

Optional filters can be applied by date and service status, making it easy to query active, past, or upcoming services based on the integrator's needs.

3.12. GET FBO FROM AIRPORT

`GET /fbos?icao={ICAO}`

This endpoint returns the full list of available FBOs at a specific airport, identified by its ICAO code.

3.13. CLIENTS

DRIVANIA allows agencies to manage multiple end clients under a single account, enabling personalized service requests for each of them

CLIENT_CODE USAGE

In the quotation request method (Quote Request), a `client_code` parameter provided by DRIVANIA can be included. This code uniquely identifies the end client for whom the service is being requested.

Important: Once a service is requested with a `client_code`, it becomes exclusively linked to that client and cannot be modified later.

CLIENT MANAGEMENT VIA API

Client management is handled through two endpoints:

- Client creation
- Client search by name

Both methods return the corresponding `client_code`, which is unique and permanent for each client within the DRIVANIA system.

It is recommended that the integrator maintains a persistent relationship between the `client_code` and the client data in their own system, as this identifier will be required for future service requests.

3.13.1. CREATE CLIENT

`POST /clients`

This endpoint allows the creation of new clients associated with the authenticated agency. The following data must be provided:

- Client name (required)
- Country (required)
- Address (optional)
- Postal code (optional)
- VAT number (optional, except in specific cases)

Note: If the specified country is **Spain**, the VAT number field is mandatory.

Once the client is successfully created, the response will include a `client_code`, which uniquely identifies the client within the DRIVANIA system. This code will be required for submitting service requests on behalf of the client, as described in section **3.13**.

3.13.2. SEARCH CLIENTS

GET `/clients?name={name}`

This endpoint allows searching for clients registered under the authenticated agency, using the client's name as the search criterion.

- A minimum of 3 characters is required for the search
- The response returns an array of all clients whose names partially or fully match the input.
- Each result includes the `client_code`, which is required to request services on behalf of the client.

3.14. CALLBACKS

DRIVANIA offers an automatic notification system (callback) that provides real-time updates on service status changes—both before and during execution.

To use this system, the integrator must provide a data reception URL for each environment (integration and production).

Note: During Phase 3 of the integration process, all technical details regarding the structure and content of the messages sent through the callback system will be provided.