

Altegio Integration

What the AI can do

Summary

Altegio Integration connects **Altegio** (appointment booking and business management SaaS) with **Newo Platform**.

It enables:

- Loading Altegio services and staff into the agent prompt context
- Checking live Online Booking availability by service, staff member, and date
- Creating public Online Booking appointments
- Cancelling Newo-created appointments through the public cancellation endpoint
- Rescheduling Newo-created appointments after validating the replacement slot
- Looking up existing CRM appointments by phone number when Business Management authorization is configured
- Publishing Altegio-specific appointment scenarios, intent types, and procedures into the Newo Canvas

This integration is designed for **salons, clinics, studios, and other appointment-based businesses using Altegio** that need **AI-assisted booking, appointment lookup, cancellation, and rescheduling through chat or voice**.

The screenshot shows the 'DEVELOPERS & PARTNERS HUB' for Altegio. The header includes the Altegio logo, navigation links for 'DEVELOPERS', 'Online Booking', 'Business Management v2', 'Business Management v1', 'Developer Tools', and 'Get API Key', a search bar, and a moon icon. The main content area features the heading 'Build Powerful Integrations' and a subheading 'Official API documentation for integrating online booking, CRM, scheduling, POS, and business management features into your applications'. It displays '200+ API endpoints' and '99.98% Platform uptime'. There are two buttons: 'Browse APIs' and 'Get API Key'. A code block shows a cURL command to get available time slots. Below this, the 'Available APIs' section lists four specialized APIs: 'Online Booking' (Public endpoints for booking widgets and client portals. No user authentication required), 'Business Management v2' (Complete business operations API with improved design. New features are released here first. Recommended for new integrations), 'Business Management v1' (Full-featured API for all business operations: scheduling, POS, CRM, inventory, finances, loyalty. Maintained for backward compatibility), and 'Developer Tools' (Webhooks, dictionaries, and integration tools for marketplace app developers). Each API card has an 'Explore →' link. An 'Ask AI' button is located in the bottom right corner.

altegio DEVELOPERS Online Booking Business Management v2 Business Management v1 Developer Tools Get API Key

DEVELOPERS & PARTNERS HUB

Build Powerful Integrations

Official API documentation for integrating online booking, CRM, scheduling, POS, and business management features into your applications

200+ API endpoints 99.98% Platform uptime

Browse APIs Get API Key

```
# Get available time slots curl -X GET "https://api.altegio.io/api/v1/book-times/{company_id}/{staff_id}/{date}" \ -H "Authorization: Bearer {partner_token}" \ -H "Accept: application/vnd.api.v2+json"
```

Available APIs

Four specialized APIs for different integration scenarios

Online Booking

Public endpoints for booking widgets and client portals. No user authentication required

Explore →

Business Management v2

Complete business operations API with improved design. New features are released here first. Recommended for new integrations

Explore →

Business Management v1

Full-featured API for all business operations: scheduling, POS, CRM, inventory, finances, loyalty. Maintained for backward compatibility

Explore →

Developer Tools

Webhooks, dictionaries, and integration tools for marketplace app developers

Explore →

Ask AI

Figure S01. Official Altegio developer documentation showing available API surfaces.

Common use cases

- When a customer asks for available times -> Check Altegio availability and return bookable slots
- When a customer confirms one offered slot -> Validate the slot and create an Altegio booking
- When a customer wants to cancel a Newo-created appointment -> Cancel it in Altegio and remove it from local booking storage
- When a customer wants to move a Newo-created appointment -> Validate the replacement slot, fetch the current record, and update the appointment
- When a customer asks about existing appointments -> Search Altegio CRM by phone and show upcoming matching records
- When the integration is published -> Register custom tools, refresh location options, and refresh the service/staff cache

Features at a glance

Feature	Supported	Notes
Location discovery	✓	Loads marketplace app locations into the <code>altegio_location_id</code> dropdown.
Service and staff cache	✓	Refreshes compact service/staff data on publish, session start, conversation start, or manual refresh.
Check availability	✓	Uses <code>book_dates</code> first, then <code>book_times</code> for exact staff/date availability.
Create booking	✓	Uses <code>book_check</code> before <code>book_record</code> ; stores compact booking metadata in persona bookings.
Phone confirmation retry	✓	If Altegio asks for a confirmation code, the pending booking payload is stored and retried with <code>code</code> .
Cancel booking	✓	Automatic only for Newo-created bookings with stored <code>record_id</code> and <code>record_hash</code> .
Reschedule booking	✓	Automatic for Newo-created bookings with <code>record/service/staff/location</code> data and a configured User Token.
Existing appointment lookup	✓	Searches CRM clients by phone, then upcoming records for the matched client.
Canvas setup	✓	Adds Altegio-specific intent types, scenarios, and procedures; removes conflicting default NAF scheduling items.
Standard NAF booking tools	✗	<code>check_availability_tool</code> , <code>create_booking_tool</code> , and <code>cancel_booking_tool</code> are disabled overrides. Custom Altegio tools are used instead.
Legacy CRM cancellation	✗	B2B/legacy appointments without public <code>record_hash</code> are routed to human handoff.

Webhooks	✗	Not implemented; inbound payload/security model still needs confirmation.
Historical data import	✗	The integration searches only the configured future appointment window and locally stored Newwo bookings.

2. Official API Surfaces Used

Altegio separates public Online Booking APIs from private Business Management APIs.

The screenshot displays the official documentation for the Altegio Online Booking (1.0.0) API. The interface is dark-themed and includes a sidebar with navigation links for 'Online Booking & Client Personal Cabinet v1', 'Authentication', 'Rate Limits', and 'Public API'. The main content area is titled 'Online Booking (1.0.0)' and provides a detailed description of the API's purpose: enabling third-party developers to embed online booking functionality. It specifies the base URL as `https://api.altegio.io/api`. The 'Authentication' section states that all requests require a Bearer token in the HTTP header, and users must register in the Altegio Marketplace to obtain one. The 'Rate Limits' section indicates a limit of 200 requests per minute per IP address. On the right, there are sections for downloading the OpenAPI description files, an overview with links to the license and terms of service, supported languages (curl, JavaScript, Node.js, Python, Java), and server endpoints for both mock and production environments. At the bottom, there is an 'Authentication' section and an 'Operations' section with a search bar.

Figure S02. Official Online Booking documentation: public endpoints use partner bearer authorization.

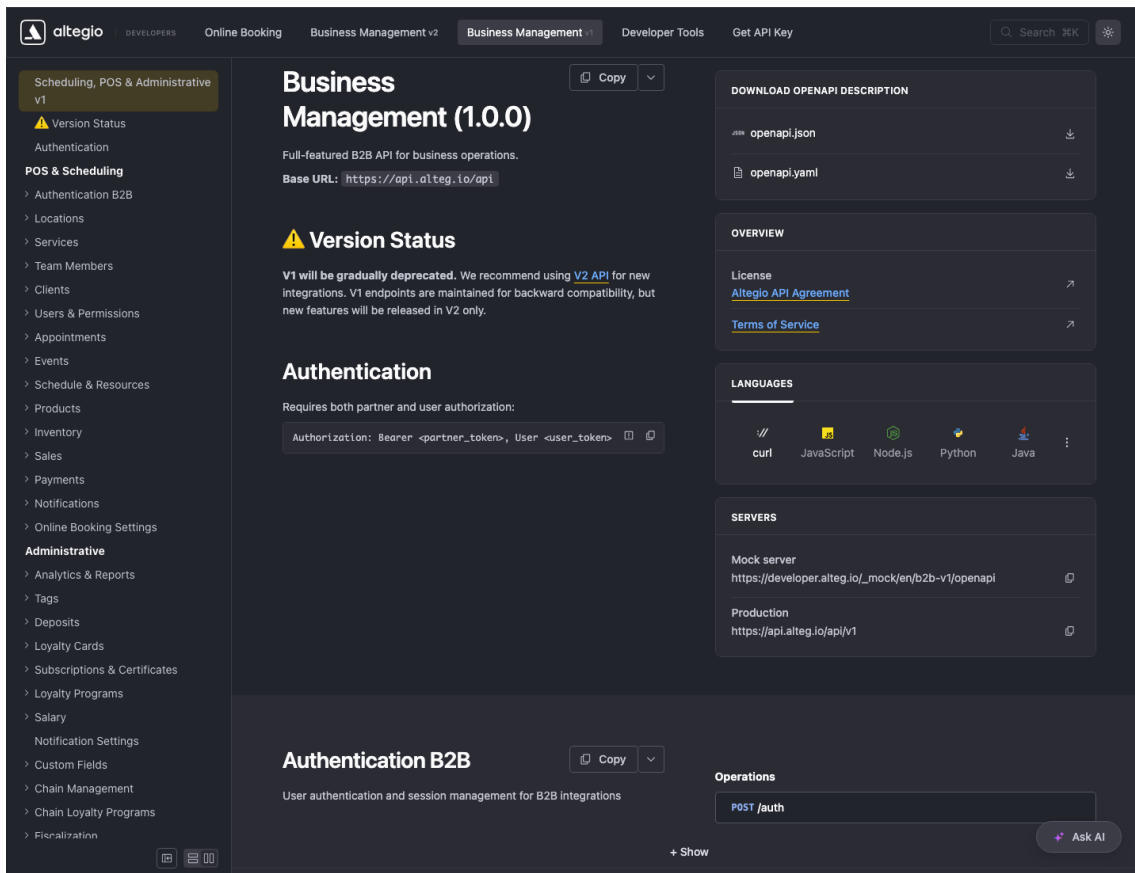


Figure S03. Official Business Management v1 documentation: B2B endpoints require partner plus user authorization.

Official references:

- [Altegio Developer Hub](#)
- [Online Booking API](#)
- [Business Management v1 API](#)
- [Business Management v2 API](#)

Before You Start

Before installation:

- Active **Altegio** account with marketplace/developer API access
- **Partner Token** from the Altegio marketplace application
- Numeric **Marketplace Application ID** used for location discovery
- Connected Altegio location selected in the generated Location dropdown
- **User Token** if using existing appointment lookup or automatic rescheduling
- User Token permissions for clients and appointment records in the selected location

Public Online Booking calls use:

Authorization: Bearer <partner_token>

Business Management calls use:

```
Authorization: Bearer <partner_token>, User <user_token>
```

The integration does not use username/password credentials. Configure `altegio_user_token` directly.

Setup

Create the Newo project

If this project already exists in **Newo Builder**, skip this subsection and continue with the integration-specific settings below.

1. In **Newo Builder**, open the projects list and click **Create Project** (or **Create New Project** from the top-right menu).

 Create New Project menu in Newo Builder

2. Fill **IDN** and **Title**. The exact names do not matter; use any clear names your team will recognize.
3. In **Registry**, choose the release channel:
 - **staging** — the newest module fixes appear here first. Use it when you need the latest fix, but expect possible unfinished changes.
 - **production** — the final stable version for live projects.
4. In **Module**, select the module for this integration.

 Create Project form showing IDN, Title, Registry, and Module fields 5. Leave **Module version** on **Latest version** unless support tells you to pin a specific version, then click **Create**.

4.1 Step 1 — Prepare Altegio Credentials

1. Open the Altegio developer or marketplace application settings.
2. Copy the **Partner Token**.
3. Copy the numeric **Application ID**. Use the number, for example `1856`, not the text application slug.
4. In **API Access**, copy the **User Token** if existing appointment lookup or reschedule should be enabled.
5. Confirm the application has access rights for clients and appointment records in the target location.

4.2 Step 2 — Connect in Newo Platform

1. Open **Newo -> Builder -> Settings -> Module - AltegioIntegration**.
2. Set the visible attributes:

Attribute	Required	Type	Description
<code>altegio_partner_token</code>	✓	string	Partner authorization for booking APIs and the part of B2B calls.
<code>altegio_application_id</code>	✓	number	Numeric marketplace application ID used to locate connected locations.

altegio_location_id	✓	enum	Altegio location used for availability, booking, cancellation, reschedule, cache, and lookup. Store labels like TwinMind (1362 runtime extracts 1362823.
altegio_user_token	Required for B2B lookup/reschedule	string	User authorization for CF client search, appointment lookup, record fetch, and record update.
altegio_availability_enabled	✗	enum	Enables/disables altegio_check_availability Default: True.
altegio_booking_enabled	✗	enum	Enables/disables altegio_book_appointment Default: True.
altegio_cancellation_enabled	✗	enum	Enables/disables altegio_cancel_appointment Default: True.
altegio_reschedule_enabled	✗	enum	Enables/disables altegio_reschedule_appointment Default: True.
altegio_existing_appointments_enabled	✗	enum	Enables/disables altegio_search_appointment Default: True.
altegio_setup_scenarios	✗	enum	Controls whether Altegio Canvas scenarios are published. Default: True.
altegio_existing_appointments_search_days	✗	number	Number of days forward today used for CRM appointment lookup. Default: 365.

3. Click **Save + Publish All**.

4. On publish, the integration automatically:

- o Creates/starts `altegio_connector`
- o Initializes project attributes and metadata under `Module – AltegioIntegration`
- o Creates a setup persona/actor for setup-time API calls
- o Registers custom Altegio tools and disables conflicting standard NAF tools
- o Calls marketplace location discovery when Partner Token and Application ID are present
- o Updates the `altegio_location_id` enum options
- o Auto-selects the location when exactly one location is returned
- o Refreshes the service and staff cache.

How to test that everything works

Where to test

Testing can be done through the **Newo conversation interface** using a deployed Altegio project. The integration performs real operations in Altegio, so use a safe test location, test staff member, and test service.

8.2 Functional Smoke Test

1. **Setup:** Publish the project and verify `altegio_connector` exists and is started.
2. **Location discovery:** Fill `altegio_partner_token` and `altegio_application_id`, publish, then verify `altegio_location_id` has dropdown options.
3. **Cache:** Confirm `ServiceAndStaffCatalog` appears in prompt data after publish/session start.
4. **Availability:** Ask for availability for a known service, staff member, and date. Verify `AvailableSlots` contains real slots.
5. **Booking:** Choose one returned slot and provide name/phone/email. Verify the booking appears in Altegio and persona `bookings` contains `record_id` and `record_hash`.
6. **Existing appointment lookup:** Search by the same phone. Verify `ExistingBookings` shows local bookings and B2B records when User Token is configured.
7. **Reschedule:** Ask to move a Newo-created booking. Verify `RescheduleStatus` completes and the local booking datetime is updated.
8. **Cancellation:** Ask to cancel a Newo-created booking. Verify Altegio cancels it and persona `bookings` no longer contains that record.

FAQ

Q: Does the integration use Altegio username and password? A: No. Current setup expects Partner Token, Application ID, Location, and optionally User Token. Username/password fallback is not used.

Q: Why does `altegio_location_id` show a label like `TwinMind (1362823)` ? A: `SetupFlow` stores human-readable enum labels for operators. Runtime code extracts the numeric ID from the parentheses.

Q: Can the AI cancel any Altegio appointment? A: No. Automatic cancellation is limited to Newo-created public bookings with stored `record_id` and `record_hash`. Legacy CRM appointments should be handled by staff.

Q: Can the AI reschedule legacy CRM appointments found by phone lookup? A: Not automatically in the current workflow. Reschedule is designed for Newo-created bookings where the integration has stored record, service, staff, and location metadata.

Q: Why are existing appointment searches future-only? A: `ExistingClientFlow` uses `altegio_existing_appointments_search_days` to search forward from the current date and filters out past records before storing B2B appointments.

Q: What happens if the service/staff cache is stale? A: `CacheDataFlow` refreshes it automatically when empty or older than `altegio_cache_range_seconds`, and it can be forced with `altegio_refresh_cache`.

Q: What version is documented here? A: The integration manifest and `MetadataFlow` report version `1.0.0`; `import_version.txt` is `v2.0.0`.