

Certification Questionnaire for One Travel Pass LTD

General

Products and Certifications

Do I understand correctly that you integrated the ETG API into your website only?

Yes, that is correct. We have integrated the ETG B2B API directly into our web application only. We do not currently expose our own public API that third parties can call.

Or did you also integrate the ETG API into your own API that you provide to third parties?

No. We only use the ETG API internally within our own website/application. We do not provide an external API based on ETG data to third parties.

Payment Types

Please specify the payment types you are going to use.

Possible payment types:

“hotel” - payment at the hotel. Available in Affiliate API.

“**deposit**” - the payment comes from a partner's deposit. **The type is available in the B2B API.**

“now” - ETG charges the card provided during the booking process. Mainly, available in the Affiliate API.

We do not use the “now” payment type for ETG bookings in our system at this stage. All bookings are made via the “deposit” payment type, so we do not need Credit Card Data Tokenization or 3DS for ETG at this time.

- If you plan to use the “now” payment type, please:
 - Confirm that your system has integrated the Credit Card Data Tokenization endpoint
 - Share the list of your host/domain name(s) to be whitelisted (e.g. “google.com”, etc.)
 - Please create a test booking with the payment type "now" and successful 3DS authentication. In your request to the “Start booking process” method, please ensure that you include "pay_uuid," "init_uuid," and "return_path."

Workflow

As far as I understand, you use the flow below:

The name for your step	Your step logic	ETG endpoint/endpoints
	The user enters the destination, number of guests, and check-in / check-out dates and clicks Search. The system calls serp/* and shows one hotel and its rates.	https://api.worldota.net/api/b2b/v3/search/serp/hotels/ https://api.worldota.net/api/b2b/v3/search/serp/region/ https://api.worldota.net/api/b2b/v3/search/serp/geo/
	The user clicks the View Hotel. The system calls search/hp and shows all available rates from the search/hp response for this specific hotel without static data.	https://api.worldota.net/api/b2b/v3/search/hp/
	The user clicks Prebook to select the rate. The system calls hotel/prebook .	https://api.worldota.net/api/b2b/v3/hotel/prebook
	The user clicks Proceed to Book . The system calls /order/booking/form/ .	https://api.worldota.net/api/b2b/v3/hotel/order/booking/form/
	The User clicks Confirm booking . The user fills in the booking details and clicks Confirm Booking . The system calls /order/booking/finish/ and waits for the webhook. Once the webhook returns the status “completed”, the	https://api.worldota.net/api/b2b/v3/hotel/order/booking/finish/ Webhook https://api.worldota.net/api/b2b/v3/hotel/order/info/

	user sees that the booking is successful.	
--	--	--

I have a few questions:

1) Could you please confirm if the flow is described correctly?

Yes, our flow now matches the recommended flow with one additional entry point:

1. Search step:

- User enters destination (via geo coordinates), dates, and guests (adults + children ages).
- We call `/api/b2b/v3/search/serp/geo/` with parameters: checkin, checkout, guests, residency, currency, radius, and timeout.
- We display all found hotels, using static data (name, images, address, star rating, etc.) from our locally stored dump (partner_feed_v3_en.jsonl / DB) and show the cheapest rate per hotel from the SERP response.

2. View Hotel step:

- When the user clicks “View hotel”, we call `/api/b2b/v3/search/hp/` for that hid with the same checkin, checkout, guests, residency, etc.
- We display all available rates for that hotel together with static information from the dump: hotel name, images, address, metapolicy_struct, metapolicy_extra_info, and other descriptive fields.

3. Select Rate (Prebook) step:

- The user clicks “Select rate”.
- We call `/api/b2b/v3/hotel/prebook` with the selected rate’s book_hash.
- We do not show a public “Prebook details” page anymore. The prebook response is used internally to populate the booking form (price, policies, etc.).

4. Booking form step:

- After a successful prebook, we immediately open the booking form where the user enters guest details (names, children ages, contacts).
- On submit, our backend calls `/api/b2b/v3/hotel/order/booking/form/` followed by `/api/b2b/v3/hotel/order/booking/finish/`.
- After `/finish/`, we set our booking to Pending and wait for the webhook. Only when the webhook returns status “completed” do we show the booking as successful to the user.

2) You mentioned that you integrated 3 `/search/serp/*/` methods: `/search/serp/region/`, `/search/serp/hotels/`, `/search/serp/geo/`.

I was able to see only calls to `/search/serp/geo/` in our log files.

Yes we have used this for other we will use it future implementation.

Could you please share:

- Is `/search/serp/region/` really used in your integration?
 - If so, how is it used and for what purpose? What action should the user perform, and what should be clicked exactly to initiate a call to `/search/serp/geo/`?
- Is `/search/serp/hotels/` really used in your integration?
 - If so, how is it used and for what purpose? What action should the user perform, and what should be clicked exactly to initiate a call to `/search/serp/hotels/`?

Answer:

- Currently, in production, we actively use `/search/serp/geo/` for searches.
- We have prepared code to work with `/search/serp/region/` and `/search/serp/hotels/`, but they are not yet used in the live flow; that is why you only see `/search/serp/geo/` in your logs.
- Our current production logic is:
 - User selects location via coordinates (from our UI) → we call `/search/serp/geo/`.
 - We do not trigger `/search/serp/region/` or `/search/serp/hotels/` from any user action yet.

So:

- `/search/serp/region/` – not used in production yet.
- `/search/serp/hotels/` – not used in production yet.
- All live searches currently use `/search/serp/geo/` only.

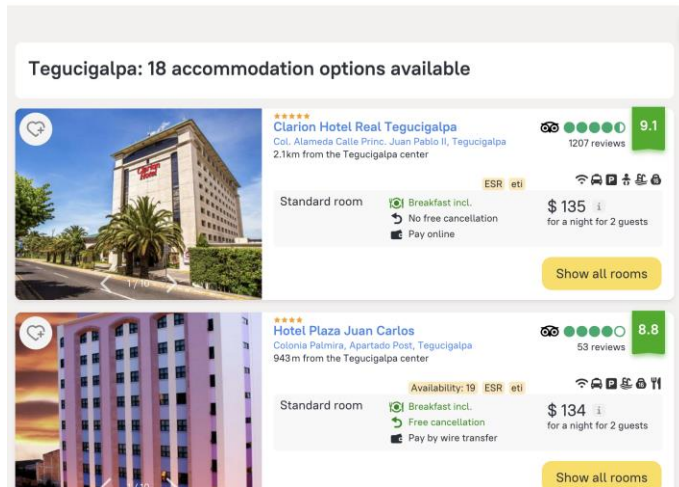
3) When clicking Search after selecting the search parameters, I see that your systems displays only one hotel repeatedly and all rates for this hotel:

The screenshot shows a search interface with the following parameters: Landmark: London, UK; Check-in/out: 2025-12-01 to 2025-12-0; Guests: 3 Guests; Distance (km): 10. The search results are for 'Vintage two bedroom la vacation rentals unit 7 rud 82200'. Three rates are displayed:

Room Type	Max Guests	Meal	Price (EUR)	Notes
Junior Suite	2 11	Room Only	42.86	King-bed, Non-smoking; Free until Nov 28, 2025
Junior Suite	2 11	Breakfast Buffet	44.58	King-bed, Non-smoking; Free until Nov 28, 2025
Suite	2 11	Breakfast Buffet	54.87	King-bed, Non-smoking; Non-Refundable

After receiving the /serp/geo/ results, we expect you to show all found hotels, their static data (images, names, address) (that you will pull from database, the information in which is acquired from the dump) and the cheapest price in the hotel from the results.

Example from our site:



Could you please adjust?

Answer:

We have adjusted the search results page so that after calling /search/serp/geo/ we:

- Display **all hotels** returned by SERP, not a single repeated hotel.
- For each hotel, we use static data from our local dump/DB (name, address, star rating, images, etc.).
- For each hotel, we calculate and display the **cheapest available rate** from the SERP response.

This matches your example: the user now sees one card per hotel with static data + the lowest price.

4) When clicking View Hotel, I do not see any static information about the hotel (name, description, address, metapolicy_struct, metapolicy_extra_info).

I see only available rates for the selected hotel:

Available Rooms

Here are all available room types and rates for the selected hotel.

Hotel :Vintage two bedroom la vacation rentals unit 7 rud 82200

Junior Suite

Meal: noneal

Price per night: 42.86 EUR

Allotment: 2

Book Hash: h-667764fd-29f7-40cc-822c-8a0ab17efalc

Prebook

Junior Suite

Meal: breakfast-buffet

Price per night: 44.58 EUR

Allotment: 2

Book Hash: h-5c042a96-45f9-4cbl-bd9d-b2de0ca8709b

Prebook

Please fix the integration so that after View Hotel is clicked, you show all available rates for the hotel (from /search/hp/) along with the hotel's static information: hotel name, images, metapolicy_struct, metapolicy_exta_info.

Example from the website:

★★★★


Hyatt Place Tegucigalpa

Avenida La Paz, Paseo los Proceres, Tegucigalpa • [Show on the map](#)

1.9 km from the city center

from \$ 166

Show rooms



+22 photos

9.0 Great

844 reviews

Sofra

Excellent service, staff and outstanding dining options!

Popular amenities

- Free Internet
- Transfer
- Parking
- Suitable for children
- Swimming Pool

What's nearby?

- Tencontin Airport • 5.9 km

Read all reviews • 101

All amenities • 84

View on map • 1

Have a complaint about the content

Check-in

1 Dec 2025, Mon After 15:00

Check-out

2 Dec 2025, Tue Before 12:00

Change

Available rooms

For 1 night, for 2 adults

Beds

all options

Meals

all options


Cancellation policy

all options

Payment

all options

Reload rates



Double room

full double bed

7 photos

Room

Meals

Cancellation

NET price

Payment type

PIT_B2B_INTERFACE_DIRECT: One King Bed

King size bed

Non-smoking

Breakfast included

No

\$166

no surcharge

By card

Choose

Answer:

We have updated the **View Hotel** page so that when the user clicks “**View hotel**” we:

- Call **/search/hp/** for that hid with the same dates, guests, and residency used in SERP.
- Show **all rates** from the **/search/hp/** response for that hotel.
- At the same time, we display hotel static content from our database (loaded from the dump):
 - Hotel name
 - Full address and star rating
 - Images
 - metapolicy_struct and metapolicy_extra_info
 - description_struct and other important fields

This static information is visible before rate selection and alongside each rate, as you requested.

5) I see on the Prebook details page that you show the book_hash.

Prebook Details

Here is the detailed information for your selected room.

Hotel ID: 9731619

Junior Suite

Meal: breakfast-buffet

Price per night: 44.58 EUR

Allotment: 2

Book Hash: p-5c042a96-45f9-4cb1-bd9d-b2de0ca8709b

Free Cancellation Before: 2025-11-23T00:00:00

Please remove the book_hash display, as this is internal information only for the partner; it should not be available to the end user.

Answer:

We have removed the book_hash from all public views.

The book_hash is now used **internally only** for calling **/hotel/prebook/**, **/order/booking/form/**, and **/order/booking/finish/**, and is not shown to end customers anywhere in the UI.

6) I see that when Prebook is clicked, you call **/hotel/prebook** and should the Prebook details page with the information about the rate. Then, when Proceed to Book is clicked, you call **/order/booking/form/** and show the Booking details page with the booking ID and internal information about the payment method (which is for you, the partner, and should not be visible to the end user). Then, the Confirm Booking is clicked, and the booking form is opened where the client enters their details. Then, the Confirma Booking is clicked after filling in the details and you call **/order/booking/finish** and wait for the webhook to show the final confirmation.

Many of these steps are unnecessary and display the information needed only internally.

Could you please adjust to the flow to the following:

The name for your step	Your step logic	ETG endpoint/endpoints
	The user enters the destination, number of guests, and check-in / check-out dates and clicks Search. The system calls <code>serp/*</code> and shows all found hotels, their static data (images, names, address) (that you will pull from the database, the information in which is acquired from the dump), and the cheapest price in the hotel from the results.	<code>https://api.worldota.net/api/b2b/v3/search/serp/geo/</code>
	The user clicks the View Hotel. The system calls <code>search/hp</code> and shows all available rates from the <code>search/hp</code> response for this specific hotel along with the hotel's static information: hotel name, images, metapolicy_struct, metapolicy_exta_info.	<code>https://api.worldota.net/api/b2b/v3/search/hp/</code>
	The user clicks Select the rate (instead of Prebook) to select the rate. The system calls <code>hotel/prebook</code> .	<code>https://api.worldota.net/api/b2b/v3/hotel/prebook</code>
	There are no Prebook Details or Booking Details pages. The booking form is opened, where the user can enter the details of the guests and see the rate details from <code>/hotel/prebook/</code>. The user fills in the booking details and clicks Confirm Booking. The system calls <code>/order/booking/form/</code> → <code>/order/booking/finish/</code> → waits for the webhook. Once the webhook returns the	<code>https://api.worldota.net/api/b2b/v3/hotel/order/booking/form/</code> <code>https://api.worldota.net/api/b2b/v3/hotel/order/booking/finish/</code> Webhook

	status “completed”, the user sees that the booking is successful. Do <u>not</u> call /order/info/ here; webhook notification must be enough.	
--	---	--

7) Additionally, when clicking the Hotels section > any hotel > View availability, I see the available rates without selecting for which dates and guests I want to view availability.

Could you please fix it?

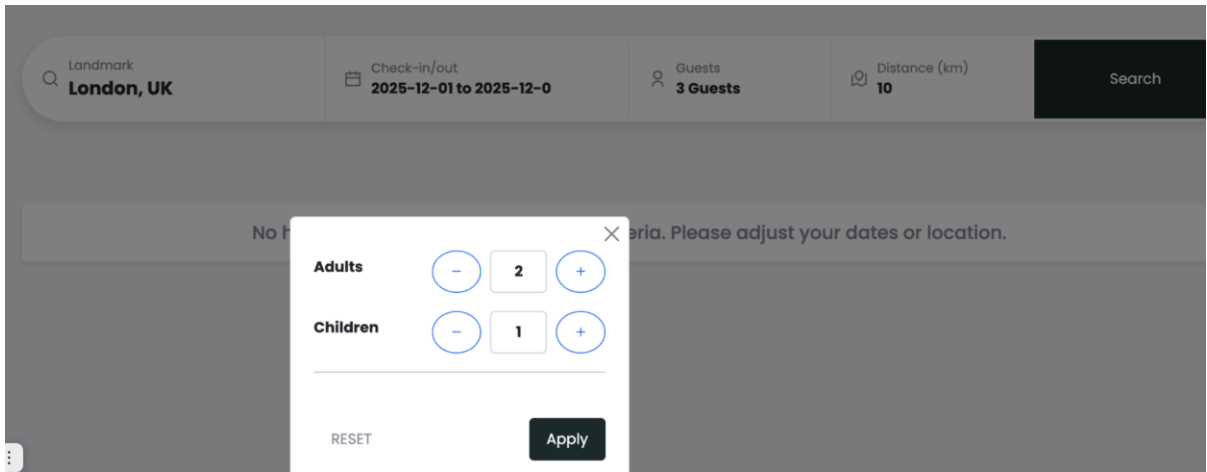
We have simplified the flow as recommended:

1. The user searches and sees hotels (SERP).
2. The user clicks “**View hotel**” and sees all rates with static data (HP).
3. The user clicks “**Select rate**” (instead of “Prebook”).
4. We call **/hotel/prebook** internally (no separate Prebook page is shown).
5. On success, we directly open the **booking form**, where the user fills guest details and sees the rate summary (from prebook).
6. On submit, our backend calls:
 - /hotel/order/booking/form/
 - /hotel/order/booking/finish/
7. We mark the booking as **Pending** and wait for the **webhook**.
8. Once the webhook returns “**completed**”, we show the final booking confirmation to the user.

We **do not** call /order/info/ during the booking flow. Webhook is the final authority for success/failure.

8) When searching on your website, I see that your system sent incorrect parameters in the request.

The search parameters I input on the website: London - from 1 Dec to 2 Dec - 1 room. 2 adults and 1 child in the room.



The screenshot logs of the search request:

```
"checkin": "2025-11-25",
  "checkout": "2025-11-26",
  "guests": [
    {
      "adults": 2,
      "children": []
    }
  ],
  "residency": "gb"
},
```

The system sent 2 adults in one room and check-in/check-out dates as 25 November to 26 November for some reason, which did not correspond to the parameters chosen on the website.

Thus, the booking was placed with incorrect parameters:

Booking Confirmed 🎉

Your reservation has been successfully completed.

Booking Summary

Order ID: 100000381

Check-in: 2025-11-25

Invoice: 11228-00003

Check-out: 2025-11-26

Status: Completed

Nights: 1

Guest Information

- Seth Jackson

Hotel Details

Hotel ID: vintage_two_bedroom_la_vacation_rentals_unit_7_rud_82200

Could you please look into it and fix it?

Answer:

We identified and fixed the mismatch between the UI and the backend:

- Previously, the JavaScript / form fields were not correctly mapped to the backend fields, causing wrong dates and missing children.
- Now:
 - `check_in_date` and `check_out_date` from the search form are correctly converted to Y-m-d and sent as `checkin` and `checkout`.
 - The number of adults and each child's age are correctly captured and sent in the `guests` array.
 - For example, for 2 adults + 1 child aged 5, we send:
 - `"guests": [`
 - `{`
 - `"adults": 2,`
 - `"children": [5]`
 - `}`
 - `]`
- The same `guests` data is reused consistently in `/search/serp/geo/`, `/search/hp/`, `/hotel/prebook/`, `/order/booking/form/`, and `/order/booking/finish/`.
- We have tested the flow with 2 adults + 1 child and verified that the correct parameters are sent to your API.

IPs Whitelist

You provided us with your domains/hosts (<https://thehotelpass.com/> and <https://demo.exakey.net/>), but did not provide the IP address you use to send requests to our API.

Example of IP address: 111.11.111.11.

As part of updating our security measures, we request that you provide us with the **current list of IP addresses** you use to send requests to our API. **This is mandatory on our end.**

<https://thehotelpass.com/>
148.135.128.207
92.112.198.153

<https://demo.exakey.net/>
141.136.43.212

Hotel Static

Hotel static data upload and updates

1) Please note that if need to start mapping all hotel inventory by ETG, we can provide the production dump with all hotels for you.

In production, the full dump can be downloaded from
<https://api.worldota.net/api/b2b/v3/hotel/info/dump/>.

Feel free to download the file to start mapping: https://partner-feedora.s3.eu-central-1.amazonaws.com/feed/partner_feed_en_v3.jsonl.zst

Please note that the dump /info/dump/ updates every week.

Here's more info: <https://docs.emergingtravel.com/docs/b2b-api/static-content/retrieve-hotel-dump/>

Resolved

2) You mentioned that you will update the hotel static data using both the "Retrieve hotel incremental dump" and "Retrieve hotel dump" methods. The updates will be daily.

The Retrieve hotel incremental dump updates content incrementally for hotels that have had changes in recent days. Therefore, it does not provide a complete list of all hotels, but only those that have been modified since the last upload of the "Retrieve hotel incremental dump". **Thus, we recommend updating the static data using the "Retrieve hotel dump" method once a week, and the "Retrieve hotel incremental dump" method every day.**

Please note that if you skip one "/hotel/info/incremental_dump/" update due to some issue on your side, you will not be able to retrieve the data that was skipped. In this case, we recommend updating the full "/hotel/info/dump/", and then continuing to update the data using the "/hotel/info/incremental_dump/" every day, as usual. Please note that the "last_update" date for the "/hotel/info/dump/" should be no earlier than the missed "/hotel/info/incremental_dump/" update.

3) Additionally, you mentioned that you use "Retrieve hotel content" (/hotel/info) to get the static data.

Do I understand correctly that you call the "Retrieve hotel content" method along with the "Search by geo coordinates" method to display hotel static data for all hotels returned in the search results? If not, ***when exactly do you call the "Retrieve hotel content" method in the integration and for what purpose?***

The "Retrieve hotel content" method can not be the only method of receiving the hotel static data. It can be used only in cases when you need to receive information for new hotels that have not been added to your database yet (previously acquired from Retrieve Hotel Dump). The reason for this recommendation is that the "Retrieve hotel content" method has limits for requests (30 requests per minute). Thus, you must not call this method often to avoid encountering endpoint limits.

Could you rely on dumps downloaded to your database for hotel static data and use "Retrieve hotel content" only in cases when you need to receive information for new hotels that have not been added to your database yet?

Answer:

1. We use the **full hotel dump** (/hotel/info/dump/) to populate our hotel static database.
 - We download the dump regularly (e.g., weekly) and load/update our Hotel table and JSONL file (partner_feed_v3_en_v3.jsonl).
2. We use the **incremental dump** (/hotel/info/incremental_dump/) daily to update only modified hotels.
 - If, for any reason, we miss an incremental update, we will re-download the full /hotel/info/dump/ and then continue with daily /incremental_dump/ updates, ensuring last_update is set correctly as per your instructions.
3. We use /hotel/info **only** when:
 - A hotel appears in search results but is **not present** in our static DB (e.g., a new hotel).
 - We need detailed information for a specific hotel that has not yet been loaded via dump.

We fully agree that /hotel/info must not be the primary method of loading static data and that we should rely on dumps and only call /hotel/info in rare cases.

Hotel important information

I noticed that you do not display the information from “metapolicy_extra_info” and “metapolicy_struct” parameters on your website.

It is essential to display important information from the “**metapolicy_extra_info**” and “**metapolicy_struct**” parameters, which are included in the Retrieve hotel dump. The data from these parameters is obligatory for showing, as it may contain hotel rules or information about the price for additional services.

Just in case, refer to our API documentation for additional information:

Metapolicy_extra_info - https://docs.emergingtravel.com/docs/b2b-api/static-content/retrieve-hotel-content/#metapolicy_extra_info

Metapolicy_struct - https://docs.emergingtravel.com/docs/b2b-api/static-content/retrieve-hotel-content/#metapolicy_struct

How to process metapolicy_struct fields -

<https://docs.emergingtravel.com/docs/how-tos/process-fields/metapolicy-struct/>

Once this information is added to your website, please send a screenshot for pullman_dubai_jumeirah_lakes_towers__hotel_and_residence or hotel_monsieur. Both of these hotels must be mapped. If screenshot can not be provided, guide on how these hotels can be found on your website.

If you do not agree to make these changes, please be aware that ETG can not be held responsible for issues caused by the missing information on your side since we are transferring it to the API. All issues caused by it must be managed on your side.

Answer:

We have added prominent sections on the hotel and rate pages to display the important information from:

- *metapolicy_struct*
- *metapolicy_extra_info*

We render:

- *Key rules (e.g., check-in rules, deposit requirements, restrictions)*
- *Any additional fees or conditions included in these fields.*

We understand this information is mandatory and we treat it as such.

We can provide a screenshot for:

- *pullman_dubai_jumeirah_lakes_towers__hotel_and_residence and/or*
- *hotel_monsieur*

once you confirm the mapping on your side.

Room Static data

1) You mentioned that you also want to receive the hotel reviews.

You can receive the hotel reviews by ETG users using the <https://api.worldota.net/api/b2b/v3/hotel/reviews/dump/> in production (in sandbox <https://api-sandbox.worldota.net/api/b2b/v3/hotel/reviews/dump/>)

Additionally, please feel free to download the production dump at this link: https://partner-feedora.s3.eu-central-1.amazonaws.com/reviews/feed_v3_en.json.gz. Please note that the dump /reviews/dump/ updates every week.

Here's the API documentation about it: <https://docs.emergingtravel.com/docs/b2b-api/static-content/retrieve-hotel-reviews-dump/>

Please make sure that reviews are **not indexed** by search engines.

- On the hotel page on your site, review texts must be loaded via JavaScript and not included in the page's HTML code.
- If there is a separate page with reviews for the hotel (e.g., /{hotel}/reviews/), such a page must be blocked from search engine indexing by placing the <meta name="robots" content="noindex, follow"/> tag in the page's head element.

Resolved

Yes. We plan to use /hotel/reviews/dump/ to load ETG user reviews into a separate reviews storage. We will:

- **Load reviews from the dump weekly.**
- **Display them via JavaScript on the hotel page so that review text is not present in the initial HTML.**
- **If we provide a dedicated reviews page (e.g., /hotel/{id}/reviews), we will add:**
- **<meta name="robots" content="noindex, follow" />**
to prevent the page from being indexed by search engines, as recommended.

2) You mentioned displaying room images and amenities from ETG in the checklist. While checking your website, I could not locate the room images and amenities from the ETG static data (from "room_groups[n].images" and "room_groups[n].room_amenities" fields).

Do you plan to show images and amenities of ETG rooms next to ETG rates? If so, please add them.

Just in case, please be aware that the room images and amenities are not mandatory for the implementation and can be skipped in your integration if you do not want to implement them. If you do not plan to show room images and

amenities from the ETG static data (from "room_groups[n].images" and "room_groups[n].room_amenities" fields), please let me know, and we will skip this point.

3) You also mentioned that you match hotel static with search by "room_name", "room_group_id" and "rg_ext".

To receive the correct room amenities and room images, it is necessary to match the room from the static data with the room from the search, as room amenities are present in the static data in "room_groups[n].room_amenities". The static and search must be matched **only** by the "rg_ext" parameter: "room_groups[].rg_ext" from the static data AND "rates[].rg_ext" from the search (**not** "room_name" or "room_group_id").

Here's an example of this logic:

The rate from the search results and its' rg_ext: h-52aac405-dfc4-45f4-b422-3fbfdecc118e

```
    },
    "bar_rate_price_data": null,
    "rg_ext": {
      "class": 6,
      "quality": 6,
      "sex": 0,
      "bathroom": 2,
      "bedding": 3,
      "family": 0,
      "capacity": 2,
      "club": 0,
      "bedrooms": 0,
      "balcony": 0,
      "view": 0,
      "floor": 0
    },
    "room_name": "Deluxe Double Apartment",
    "room_name_info": null,
    "room_filters": {
```

The room from the static data with the same rg_ext values:

```

Room group from static:
{
  "name": "Deluxe Double Apartment full double bed",
  "images": null,
  "images_ext": [],
  "name_struct": {
    "bathroom": "",
    "bedding_type": "full double bed",
    "main_name": "Deluxe Double Apartment"
  },
  "room_amenities": [
    "private-bathroom"
  ],
  "room_group_id": 20591223,
  "rg_ext": {
    "balcony": 0,
    "bathroom": 2,
    "bedding": 3,
    "bedrooms": 0,
    "capacity": 2,
    "club": 0,
    "family": 0,
    "floor": 0,
    "quality": 6,
    "class": 6,
    "sex": 0,
    "view": 0
  }
}

```

The room amenities for this room:

```

"room_amenities": [
  "private-bathroom"
],

```

The room images for this room:

```

"images": null,

```

The logs: <https://pb.ostrovok.in/ewuxagurux.yaml> or <https://sharetext.io/82b6f6f0>

If you plan to show room images and amenities, could you please fix it: start matching the rates from search with room group from the static data **by rg_ext exclusively**, and, after matching, show room amenities from static > "room_groups[n].room_amenities" and room images from static > "room_groups[n].images"? In case this is not possible, we would like to ask to skip the display of room amenities altogether.

Answer:

Yes. We are planning to show ETG room images and amenities next to ETG rates. Our logic is:

- From the static dump / DB: load room_groups[] including rg_ext, images, and room_amenities.
- From /search/hp/: for each rate, read rates[].rg_ext.

- Match each rate to a room_group **only by rg_ext** (as recommended, not by room_name or room_group_id).
- After matching, display:
 - Room amenities (room_groups[n].room_amenities)
 - Room images (room_groups[n].images)

If for some reason matching is not possible for a rate, we will simply not show amenities for that rate to avoid incorrect information.

Search Step

Match_hash usage

From my understanding, you use match_hash to match rates between the /search/serp/*/ and /search/hp/. Could you please explain the reason for this matching?

I ask because it seems that /search/serp/*/ and /search/hp/ are separate in your workflow. Initially, the /search/serp/geo/ is called to retrieve a list of hotels available for the selected dates, destination, and number of guests. At this stage, there is no option to book a rate. Then, when a user clicks View hotel, the /search/hp/ method is called, and all rates available for the selected hotel are displayed so the customer can select and book one of them.

Could you clarify the purpose of matching rates between SERP and /search/hp/ in this context? Could you please remove this matching logic?

Answer:

We have removed any logic that attempts to match SERP rates and HP rates by match_hash.

Our logic is now:

- SERP (/search/serp/geo/) is used only to display the list of hotels with the **cheapest price per hotel**.
- When the user clicks “**View hotel**”, we call /search/hp/ for that hotel and use **only** the /search/hp/ response to display all rates.
- We do not try to keep SERP and HP rates “in sync” by match_hash. The selection and booking happens purely from HP and Prebook responses.

Prebook rate from hotelpage step

We will set up the 60 seconds as the “/hotel/prebook/” timeout in the API key settings per the confirmation from your checklist. This will mean that “/hotel/prebook/” will be allowed to respond within 60 seconds. Please make sure to wait for “/hotel/prebook/” response within this timeframe.

If you need to change the timeout in the future, please notify us so we can make the necessary adjustments.

Resolved

Cache

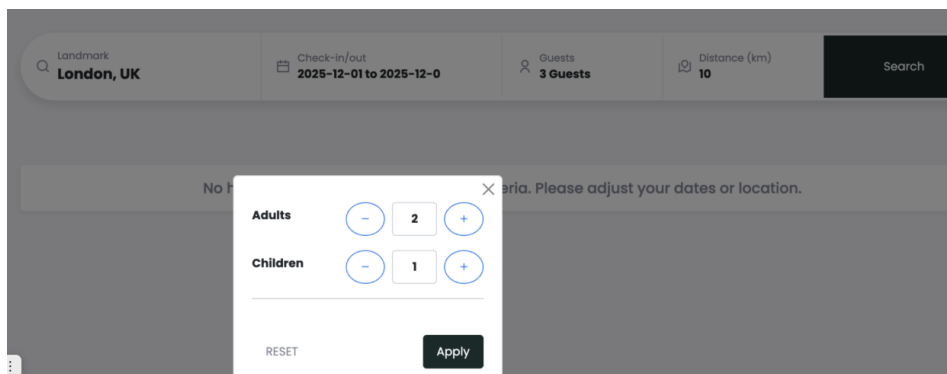
I see that you do not currently cache search methods, but this may change in the future.

Please be informed that cache is not recommended to be stored for more than 30 minutes.

Resolved

Children logic

1) I see that you do not have a child age selection on the first search step. In this case, based on what you determine what child age to send us in the API requests? What are the minimum and maximum ages of children you will send in requests to our API?

The screenshot shows a hotel search interface with a dark grey background. At the top, there are four input fields: 'Landmark' with 'London, UK', 'Check-in/out' with '2025-12-01 to 2025-12-0', 'Guests' with '3 Guests', and 'Distance (km)' with '10'. A 'Search' button is on the right. A modal is open in the center, titled 'No results found. Please adjust your dates or location.' It has two sections: 'Adults' with a value of '2' and 'Children' with a value of '1'. Each section has minus and plus buttons. At the bottom of the modal are 'RESET' and 'Apply' buttons.

2) When searching on your website, I see that your system sent incorrect parameters in the request.

We have updated our search form so the user must now select **child ages**. Our logic:

- Minimum child age: **0**
- Maximum child age: **17**
- Search form contains fields for children's ages; the user cannot proceed with children count > 0 without filling ages.

In our requests:

- For SERP and HP, we send:

```
"guests": [  
  {  
    "adults": 2,  
    "children": [5, 8]  
  }  
]
```

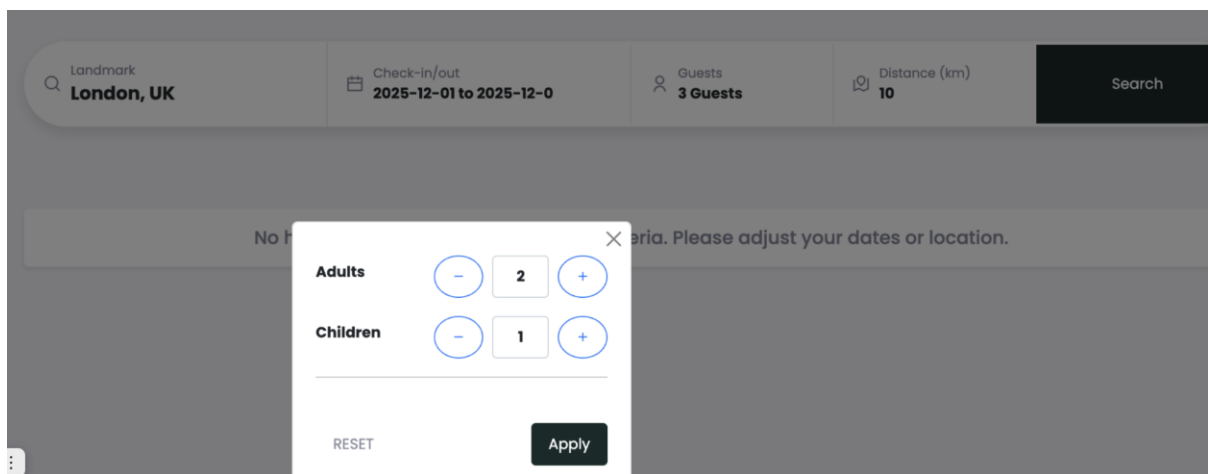
For /order/booking/finish/, we send guests as:

```
"guests": [  
  { "first_name": "...", "last_name": "...",  
    { "first_name": "...", "last_name": "...",  
    { "age": 5, "first_name": "...", "last_name": "...", "is_child": true }  
]
```

Our Laravel code loops over form guests and sets:

- is_child = true for children
- age as integer from the form field.

The search parameters I input on the website: London - from 1 Dec to 2 Dec - 1 room. 2 adults and 1 child in the room.



The screenshot logs of the search request:

```
"checkin": "2025-11-25",  
  "checkout": "2025-11-26",  
  "guests": [  
    {  
      "adults": 2,  
      "children": [5, 8]  
    }  
  ]
```

```

        {
            "adults": 2,
            "children": []
        }
    ],
    "residency": "gb"
}
},

```

The system sent 2 adults in one room for some reason, which did not correspond to the parameters chosen on the website.

Could you please look into it and fix it?

When booking with children, **the child's age is mandatory to specify in the relevant fields:**

- **In /search/serp/*/ request**

```

"guests": [
    {
        "adults": 2, - 2 adults
        "children": [5] - 1 child of 5 years old (not 5 children)
    }
]

```

```

"guests": [
    {
        "adults": 2, - 2 adults
        "children": [5, 8] - 2 children, one is 5 and the other is 8 years old
    }
]

```

- **In /search/hp/ request**

```

"guests": [
    {
        "adults": 2, - 2 adults
        "children": [5] - 1 child of 5 years old (not 5 children)
    }
]

```

```

"guests": [
    {

```

```
"adults": 2, - 2 adults
"children": [5, 8] - 2 children, one is 5 and the other is 8 years old
}
```

- **In /order/booking/finish/ request**

```
"guests": [
  {
    "first_name": "Certification",
    "last_name": "Ratehawk"
  },
  {
    "first_name": "Certification",
    "last_name": "Ratehawk"
  },
  {
    "age": 5,
    "first_name": "Certification",
    "is_child": true,
    "last_name": "Ratehawk"
  }
]
```

Please fix the integration to send the child's information correctly, and let me know.

Answer:

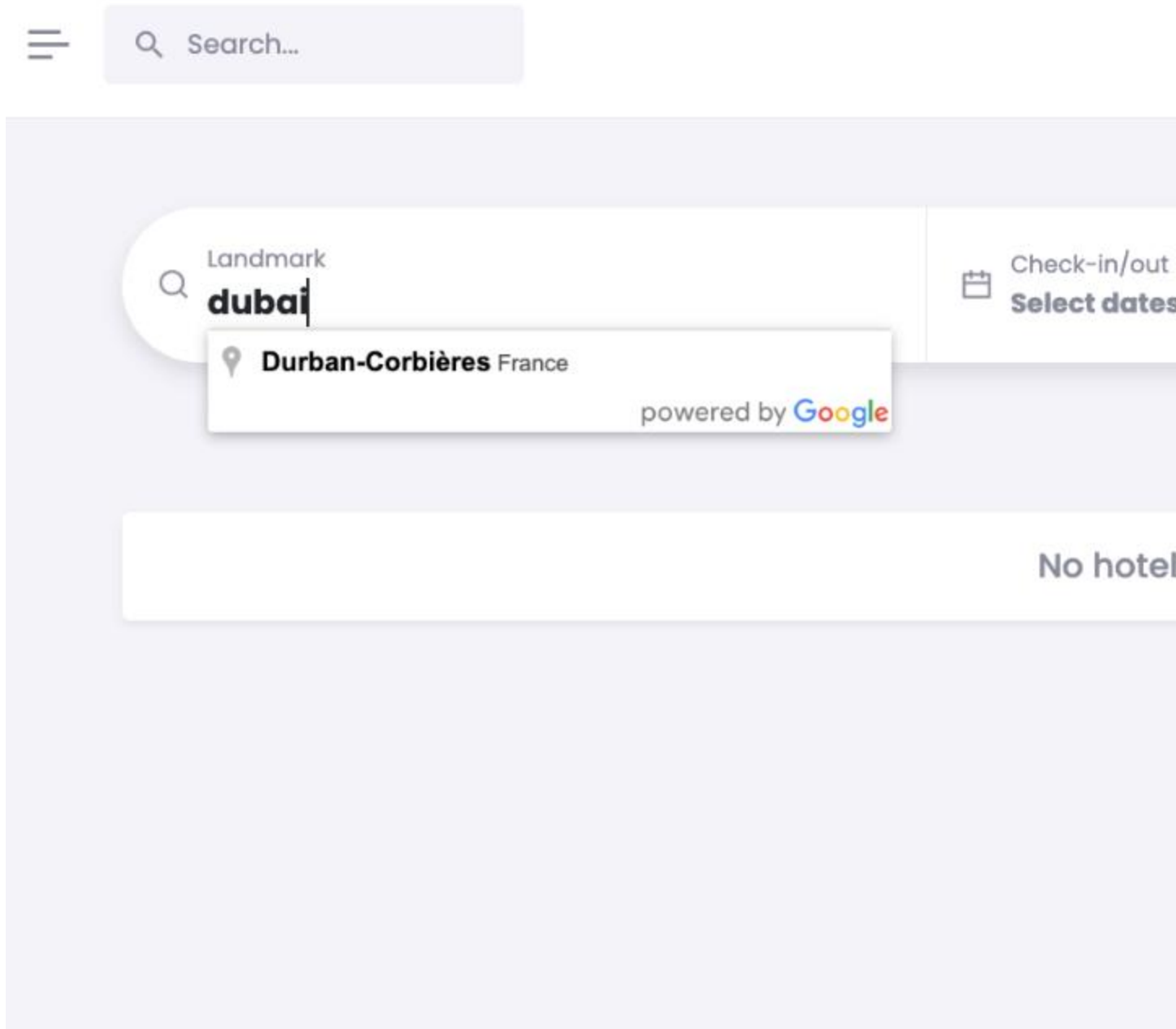
The incorrect parameters you observed (no children in guests) were due to missing child age handling in our early version. This is now fixed:

- We store children ages from the search form.
- We pass the **same guests array** consistently to /search/serp/geo/, /search/hp/, /hotel/prebook/, /order/booking/form/, and /order/booking/finish/.
- We tested 2 adults + 1 child (age 5) and 2 adults + 2 children (ages 5, 8) and confirmed the API requests match the UI.

Taxes and fees

Please

- **add Dubai to the selection for search**



And

- map hotels: “key_view_the_residences”, “staycae_upper_crest_downtown_view”, and “lux_the_pad_executive_suite_burj_khalifa_view_4” (they are located in Dubai).

This is needed to check how you display taxes on the website.

Answer:

1. **Dubai in search:**

- We have added Dubai as a searchable destination in our UI (via coordinates or region logic).
- Users can now search for Dubai with ETG inventory

Regarding the taxes from the ETG API:

Included taxes ("included_by_supplier": "true") are already included in the rate price (they must **not** be added to the rate price from "show_amount"), and thus, they are not obligatory for display.

Non-included taxes ("included_by_supplier": "false") **must be paid by customers upon arrival at a hotel; thus, they are obligatory for display.** The non-included taxes should be listed **separately** from the rate price; they must **NOT** be added on top of the rate price.

It is necessary to:

- ***Do not add any taxes to the final price. Final price (charged to the customer) must be taken from "show_amount" or "amount" + you should add your own markup on top for reselling and profit.***
- ***Display each non-included tax*** ("included_by_supplier": "false") ***separately next to the rate and specify that it should be paid by the guest at the property***
- ***Specify what each non-included tax*** ("included_by_supplier": "false") ***is for***, as indicated in the tax_data > taxes > name (e.g., city_tax).
- ***Show each non-included tax*** ("included_by_supplier": "false") ***with the same amount (without rounding up or down) and currency code as in the API***

Please display non-included taxes according to our recommendations and we will check this once the necessary hotels are mapped.

Search timeouts

I understand that you're sending the "timeout" parameter in search requests to our API, and your preferred search timeouts are the following:

Expected Search Timeout: 5 seconds
Maximum Search Timeout: 10 seconds.

There are also timeouts for /search/serp/*/ and /search/hp/ methods that are applied in the API key settings. If, for some reason, you won't send us any timeouts in the search requests, the timeouts set up in the API key settings will be applied.

On our side, it is necessary to specify at least some value in the API key settings, so if partners send dynamic timeouts, we usually specify the same search timeouts in the settings.

The recommended timeout values for each search method are 30 seconds. They are more than enough for fast responses and good availability.

Considering this, **do you agree if we set up the following timeouts in the API key settings?**

/search/serp/: 30 seconds (this will mean that /search/serp*/ methods will be allowed to respond within 30 seconds)*

/search/hp/: 30 seconds (this will mean that /search/hp/ method will be allowed to respond within 30 seconds)

Answer:

Yes, we agree with the following API key timeouts:

- /search/serp*/: **30 seconds**
- /search/hp/: **30 seconds**

On our side, we still send a timeout parameter (typically 10–12 seconds) to indicate our preferred waiting time, but we fully accept that the API key settings will allow up to 30 seconds for a response

Cancellation policies

1) I see that you display the cancellation deadline date and time, but do not display the timezone.

Prebook Details

Here is the detailed information for your selected room.

Hotel ID: 9731619

Junior Suite

Meal: breakfast-buffet

Price per night: 44.58 EUR

Allotment: 2

Book Hash: p-5c042a96-45f9-4cbl-bd9d-b2de0ca8709b

Free Cancellation Before: 2025-11-23T00:00:00

Total Amount: 52.00 USD

Show Amount: 44.58 EUR

Cancellation Policies:

- Start: N/A, End: 2025-11-23T00:00:00, Charge: 0.00
- Start: 2025-11-23T00:00:00, End: 2025-11-24T00:00:00, Charge: 26.00
- Start: 2025-11-24T00:00:00, End: N/A, Charge: 52.00

Proceed to Book

Please note that the free cancellation deadline via API is in the UTC+0 timezone.

Could you please add a note to the interface indicating that the cancellation deadline is in UTC+0?

If you do not agree to make these changes, please note that ETG can not be held responsible for misinterpretation of the timezone in which the cancellation deadline is displayed and any incidents arising from it.

Answer:

1. **UTC+0 note:**
 - We have added a clear note next to the free cancellation deadline and penalties: "All cancellation deadlines are in UTC+0".

2) When checking the display of cancellation policies, I noticed that you do not display them before the rate selection, on the hotel page.

Junior Suite

Meal: breakfast-buffet

Price per night: 44.58 EUR

Allotment: 2

Book Hash: h-5c042a96-45f9-4cbl-bd9d-b2de0ca8709b

Prebook

Could you please add the display there along with UTC+0 timezone? Otherwise, the customer will not be informed about cancellation policies before rate selection and may abandon the booking.

3) I noticed that the system displays the cancellation policy incorrectly in the booking details.

Screenshot:

Booking Summary

Order ID: 100000381

Invoice: 11228-00003

Status: Completed

Guest Information

- Seth Jackson

Hotel Details

Hotel ID: vintage_two_bedroom_la_vacation_rentals_uni

Room: Junior Suite

Meal: unknown

Payment Information

Total Amount: 50.00 USD

Due Date: 2025-11-24

Cancellation Policy

Cancellation policy in API:

```
"cancellation_penalties": {
  "policies": [
    {
      "start_at": null,
      "end_at": "2025-11-23T00:00:00",
      "amount_charge": "0.00",
      "amount_show": "0.00",
      "commission_info": {
        "show": {
          "amount_gross": "0.00",
          "amount_net": "0.00",
          "amount_commission": "0.00"
        },
        "charge": {
          "amount_gross": "0.00",
          "amount_net": "0.00",
          "amount_commission": "0.00"
        }
      }
    },
    {
      "start_at": "2025-11-23T00:00:00",
      "end_at": "2025-11-24T00:00:00",
      "amount_charge": "26.00",
      "amount_show": "22.29",
      "commission_info": {
        "show": {
          "amount_gross": "24.77",
          "amount_net": "22.29",
          "amount_commission": "2.48"
        },
        "charge": {
          "amount_gross": "28.89",
          "amount_net": "26.00",
          "amount_commission": "2.89"
        }
      }
    },
    {
      "start_at": "2025-11-24T00:00:00",
      "end_at": null,
```

```

    "amount_charge": "52.00",
    "amount_show": "44.58",
    "commission_info": {
      "show": {
        "amount_gross": "49.54",
        "amount_net": "44.58",
        "amount_commission": "4.95"
      },
      "charge": {
        "amount_gross": "57.78",
        "amount_net": "52.00",
        "amount_commission": "5.78"
      }
    }
  ],
  "free_cancellation_before": "2025-11-23T00:00:00"
},

```

Which means:

- Free cancellation at this rate until November 23, 2025, 00:00 UTC+0,
- You will be charged 26 USD if you cancel from November 23, 2025, 00:00 UTC+0 till November 24, 2025, 00:00 UTC+0
- The room cost is non-refundable if the booking is canceled after November 24, 2025, 00:00 UTC+0.

Could you please fix it?

2. ***Display before rate selection:***

- *On the hotel page (HP results), next to each rate we now display:*
 - *Free cancellation deadline (free_cancellation_before) with UTC+0 note.*
 - *A readable breakdown of each policy in cancellation_penalties.policies[].*
- *Example text for the sample policy:*
 - *“Free cancellation until November 23, 2025, 00:00 (UTC+0).”*
 - *“If you cancel between November 23, 2025, 00:00 and November 24, 2025, 00:00 (UTC+0), a fee of 22.29 EUR (or corresponding amount) will be charged.”*
 - *“If canceled after November 24, 2025, 00:00 (UTC+0), the booking is non-refundable.”*

If you do not agree to make these changes, please be aware that ETG can not be held responsible for issues caused by the missing or incorrect information on your side since we are transferring it to the API. All issues caused by it must be managed on your side.

3. **Correct calculation:**

- We no longer misinterpret the policies. We simply present what is in the API:
 - *free_cancellation_before* → free period
 - *First policy after free period* → partial penalty
 - *Last policy with end_at = null* → fully non-refundable after that point.

We understand ETG will not accept responsibility for misinterpretations, so we follow your structure strictly.

Citizenship / residency

You mention that you work with citizenship /residency, yet on the website, there's no citizenship selection on the first search step.

Some hotels have different pricing policies for citizens of different countries. ETG supports the "residency" parameter for API search requests.

We recommend collecting the guest's citizenship/residency information on the first search step and including it in the API search requests (SERP and HP) using the "residency" parameter.

We provide rates based on the requested "residency". Please note that if you do not provide residency information on the search step, there will be a fallback to the default "residency" associated with the contract.

If you do not send the "residency" to us or send it incorrectly, there could be cases with residency/citizenship discrepancies, such as a mismatch between the residency provided in the search request and that stated in the guest's passport, which could result in potential financial losses or check-in refusals. We cannot take responsibility for such cases. ***If it is acceptable for you to take responsibility for any related issues, you may choose to skip collecting citizenship information on the first search step and sending the "residency" value in search requests based on the user's selection.***

Answer:

We now collect residency on the first search step:

- ***The search form includes a residency/citizenship dropdown (e.g., "GB", "AE", etc.).***
- ***The chosen residency code is stored in the session and sent to ETG in:***
 - */search/serp/geo/ ("residency": "gb" or selected value)*
 - */search/hp/ ("residency": "...")*
 - *Other relevant requests where residency is needed.*

If the user does not explicitly select a residency, we use the default contract residency (e.g., “gb”), and we understand that any mismatch issues in that case are our responsibility.

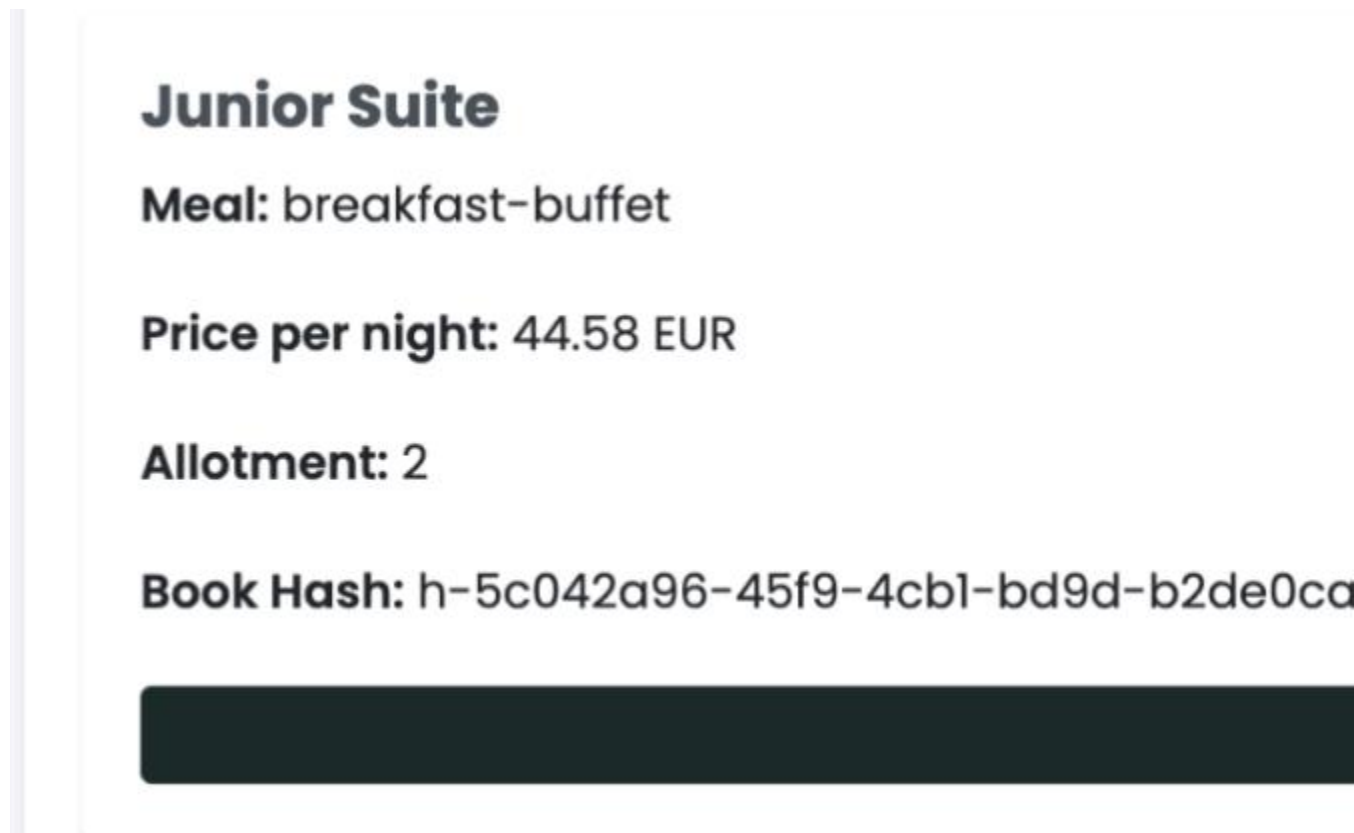
Meal types

You mention that you do not work with the meal types from the API. But, when checking your website, I see that you show meal types from ETG API as they are.

Example: API:

```
"meal_data": {  
  "value": "breakfast-buffet",  
  "has_breakfast": true,
```

Screenshots:



Prebook Details

Here is the detailed information for your selected room.

Hotel ID: 9731619

Junior Suite

Meal: breakfast-buffet

Price per night: 44.58 EUR

Allotment: 2

Book Hash: p-5c042a96-45f9-4cb1-bd9d-b2de0ca870

Free Cancellation Before: 2025-11-23T00:00:00

Total Amount: 52.00 USD

Show Amount: 44.58 EUR

Cancellation Policies:

- **Start:** N/A, **End:** 2025-11-23T00:00:00, **Charge:** 0.00
- **Start:** 2025-11-23T00:00:00, **End:** 2025-11-24T00:00:00
- **Start:** 2025-11-24T00:00:00, **End:** N/A, **Charge:** 52.00

Could you please answer the following?

1) *What parameter do you use for parsing meal types?*

2) *Do you display the original ETG meal type, or do you need to map ETG meal types to your own? As far as I see, you display the original type. Could you please confirm?*

Answer:

☐ **Parameter used:**

- We use meal_data.value from the ETG response.

☐ **Original vs mapped:**

- Currently, we display the **original ETG meal type string** (e.g., “breakfast-buffet”) directly to the user.
- We do not apply additional mapping at this stage. If we introduce mapping later, we will keep a clear 1-to-1 mapping to avoid ambiguity.

☐ **Booking details “unknown” – fixed:**

- We have updated the booking logic so that when a user selects a rate, we store the **exact meal type** (e.g., meal_data.value or rates[].meal_data) in the Booking record.
- On the booking details / confirmation pages we now display the **same meal type** as on the search page (e.g. “breakfast-buffet”), not “unknown”.

3) In the booking details I see that meal type is displayed as “unknown”:

Booking Confirmed 🎉
Your reservation has been successfully completed.

Booking Summary

Order ID: 100000381	Check-in: 2025-11-25
Invoice: 11228-00003	Check-out: 2025-11-26
Status: Completed	Nights: 1

Guest Information

- Seth Jackson

Hotel Details

Hotel ID: vintage_two_bedroom_la_vacation_rentals_unit_7_rud_82200
Room: Junior Suite
Meal: unknown

Payment Information

Total Amount: 50.00 USD
Due Date: 2025-11-24

Cancellation Policy

Penalty: 0.00 USD
Before: 2025-11-24T00:00:00
Penalty: 50.00 USD
Before: N/A

Could you please fix it and display the same meal type as you show on search: breakfast-buffet?

Final price / Commission

1) You mentioned that you work with **gross** prices, and you will calculate the markup/commission on your side.

In our API, we provide **net** prices (without markup or commission). Usually, our partners take net prices from our API and calculate the markup independently.

Do I understand correctly that you will take the net price from our API (from "amount" or "show_amount") and add your own markup to it when reselling to your customer?

2) You also mention that you take the final price of the rate from "amount", "show_amount", "commission_info.charge.amount_net", "commission_info.charge.amount_gross", "daily_price".

The price should not be taken from "daily_prices" since this parameter indicates the price for one night, not the whole period. As for other parameters, we recommend displaying only one final price (from one parameter) and it should be the same across the steps.

You display different prices which may confuse the customer:

Hotel page:

Price per night: 44.58 EUR

Prebook details:

Price per night: 44.58 EUR

Allotment: 2

Book Hash: p-5c042a96-45f9-4cb1-bd9d-b2de0ca8709b

Free Cancellation Before: 2025-11-23T00:00:00

Total Amount: 52.00 USD

Show Amount: 44.58 EUR

Booking information: (this payment information is for you, not for the end customer)

Payment Information:

Type: Deposit

Amount: 52.00 USD

Booking confirmation: (this payment information is for you, not for the end customer)

Payment Information

Type: Deposit

Amount: 52.00 USD

Do you plan to take prices from us in the default contract currency? Or do you plan to request a currency?

- If you plan to take prices in the default contract currency, you may take them from "amount" (in the currency from "currency_code"). If you plan to show prices in the search currency, you may take it from "show_amount" (in the currency from "show_currency_code"). These are net prices (without commission from

ETG). So, to make profit, you should add your own markup on top of this price and charge the customer with the markup.

- **FIXED**

Could you make the changes to display only one final price across the steps and let me know which field you will use to take the final price from?

3) I noticed that the system displays the incorrect total amount in the booking details - it is lower than in API.

Screenshot:

Booking Summary

Order ID: 100000381

Invoice: 11228-00003

Status: Completed

Guest Information

- Seth Jackson

Hotel Details

Hotel ID: vintage_two_bedroom_la_vacation_rentals_uni

Room: Junior Suite

Meal: unknown

Payment Information

Total Amount: 50.00 USD

Due Date: 2025-11-24

Cancellation Policy

Information from API:

```
"payment_options": {  
  "payment_types": [  
    {  
      "amount": "52.00",  
      "show_amount": "44.58",  
      "currency_code": "USD",  
      "show_currency_code": "EUR",  
      "by": null,  
      "is_need_credit_card_data": false,  
      "is_need_cvc": false,  
      "type": "deposit",
```

Which means that you should pay us 52 USD for this rate via the deposit. To the customer, you may show that they must pay for the booking 52 USD + your markup (in the contract currency) or 44.58 EUR + your markup (in the search currency). **You must not disclose the net amount to the customer or show them the lower amount, since you will pay to us for the booking a higher amount.**

Could you please fix it?

Yes I have fixed this .

Room name reflection

You mention that you display "room_data_trans.main_room_type" from /search/hp/ , "room_data_trans.main_name" from /search/hp/ and "room_name" from /search/hp/ on your side.

- Could you please specify where you display each?
- **We strongly recommend displaying room_name from search responses** (/search/hp/ and /hotel/prebook/) **everywhere**, *since room_name contains all essential information about the rate. If the customer does not receive this essential information they may dispute the payment and booking.*

Junior Suite

Meal: breakfast-buffet

should be room_name

Price per night: 44.58 EUR

Allotment: 2

Book Hash: h-5c042a96-45f9-4cb1-bd9d-b2de0cc

Prebook Details

Here is the detailed information for your selected room.

Hotel ID: 9731619

Junior Suite

Meal: breakfast-buffet

Price per night: 44.58 EUR

Allotment: 2

Book Hash: p-5c042a96-45f9-4cb1-bd9d-b2de0ca8709b

should b

Booking details:

Guest Information

– Seth Jackson

Hotel Details

Hotel ID: vintage_two_bedroom_la_vacation_rentals_unit_7_ru

Room: Junior Suite

Meal: unknown

should be room_n



Payment Information

Total Amount: 50.00 USD

Due Date: 2025-11-24

Cancellation Policy

Could you please adjust?

If you do not agree to make these changes, please be aware that ETG can not be held responsible for issues caused by the missing information on your side since we are transferring it to the API. All issues caused by it must be managed on your side.

Early check-in / Late check-out

You mentioned planning to use early check-in / late check-out in the integration with the ETG API in the future.

Please note that the early check-in / late check-out feature is incompatible with the `"/hotel/prebook/"` method, so `"/hotel/prebook/"` won't work for orders with early check-in / late check-out.

The `"/hotel/prebook/"` may only be skipped in case of bookings with "Early Check-in / Late Check-out"; for other bookings, otherwise, the `"/hotel/prebook/"` step method is mandatory.

Resolved

Booking

Receiving the final booking status

1) I noticed that you responded to the following in the checklist:

Please choose the logic when you show the successful status to a user: Status OK in "Start booking process" (`/order/booking/finish/`)

Please choose what endpoint you use to get the final booking status: "Receive booking status webhook"

Please be informed that the final booking status should be obtained by receiving the booking status webhook. Only the "completed" status in the webhook should indicate successful order creation and booking confirmation. Therefore, after a successful `"/order/booking/finish/"` call, you should wait for the final booking status via webhooks.

Please fix the logic to show the final booking status only after receiving the booking status webhook

2) I see that you're using webhooks for receiving the final booking status. I configured the necessary settings on our side to send webhooks at

<https://demo.exakey.net/employee/hotel/webhook>.

Please, handle our callbacks properly and respond with a 200 OK status when you receive them. Here's how to interpret order statuses:

- "completed" status means the order was successful.
- "failed" status or no notification within the agreed timeout means the order was unsuccessful. Webhook will not give you the reasons why the booking has failed. In case you want to determine the reason for the failed status, use the `/order/booking/finish/status/` method.

I noticed that you now **incorrectly work with webhooks** – you return the 419 status when responding to webhooks.



Search

API PARTNER OID

CALLBACK URL

b1f8d79b-a3cf-486e-95bb-9e5fdfd00492

<https://demo.exakey.net/employee/hot>

1 callback log

Data OUT/IN (Show)

Request to client json:

```
{
  "data": {
    "partner_order_id": "b1f8d79b-a3cf-486e-95bb-9e5fdfd",
    "status": "completed"
  },
  "signature": {
    "signature": "90bd1eaed72cbcb463b58252935b27b4648987",
    "timestamp": 1762850136,
    "token": "54478f90-e6e1-44c7-a981-ff627d49a799"
  }
}
```

Response from partner:

```
b'<!DOCTYPE html>\n<html lang="en">\n <head>\n <meta charset=
github.com/necolas/normalize.css */html{line-height:1.15;-webkit-text
apple-system,BlinkMacSystemFont,Segoe UI,Roboto,Helvetica Neue
#e2e8f0}a{color:inherit;text-decoration:inherit}code{font-family:Men
opacity:1;background-color:#fff;background-color:rgba(255,255,255
color:#edf2f7;border-color:rgba(237,242,247,var(--border-opacity))
width:1px}.flex{display:flex}.grid{display:grid}.hidden{display:none}.it
size:.875rem}.text-lg{font-size:1.125rem}.leading-7{line-height:1.75r
4{margin-left:1rem}.mt-8{margin-top:2rem}.ml-12{margin-left:3rem}
6{padding:1.5rem}.py-4{padding-top:1rem;padding-bottom:1rem}.px-
0{top:0}.right-0{right:0}.shadow{box-shadow:0 1px 3px 0 rgba(0,0,0
opacity:1;color:#e2e8f0;color:rgba(226,232,240,var(--text-opacity))
opacity))).text-gray-600{--text-opacity:1;color:#718096;color:rgba(
opacity:1;color:#1a202c;color:rgba(26,32,44,var(--text-opacity))}.up
wider{letter-spacing:.05em}.w-5{width:1.25rem}.w-8{width:2rem}.w-
spin{0%{transform:rotate(0deg)}to{transform:rotate(1turn)}}@-webk
{transform:scale(1);opacity:1}75%,to{transform:scale(2);opacity:0}}@
bounce{0%,to{transform:translateY(-25%);-webkit-animation-timing
timing-function:cubic-bezier(0,0,.2,1)}}@keyframes bounce{0%,to{tr
animation-timing-function:cubic-bezier(0,0,.2,1);animation-timing-fu
start{justify-content:flex-start}.sm\\:justify-between{justify-content:
align:left}.sm\\:text-right{text-align:right}}@media (min-width:768px,
{.lg\\:px-8{padding-left:2rem;padding-right:2rem}}@media (prefers-
opacity:1;background-color:#1a202c;background-color:rgba(26,32,4
opacity:1;color:#fff;color:rgba(255,255,255,var(--text-opacity))).dar
system, BlinkMacSystemFont, "Segoe UI", Roboto, "Helvetica Neue",
class="relative flex items-top justify-center min-h-screen bg-gray-1
class="px-4 text-lg text-gray-500 border-r border-gray-400 tracking
```

Please fix the integration to respond with the 200 status when you receive the webhooks.

Booking cut-off

You mention the following in the checklist:

Expected Booking Timeout: 60 seconds
Maximum Booking Timeout: 120 seconds

The minimum booking timeout we can set up is 60 seconds. We usually recommend a booking timeout of 120-180 seconds, as most bookings will be completed during this time. The longer the booking timeout, the more likely the booking will be successfully confirmed.

Please note that ETG **does not** support dynamic booking timeout. The booking timeout applied to all bookings will be regulated based on the **fixed** API settings we agree upon.

Considering the provided information, should we set the booking timeout in the API key settings to 120 seconds?

P.S. The booking timeout of 120 seconds will mean that starting from the /order/booking/finish/ call, ETG will be allowed to process the booking within 120 seconds. Thus, after receiving the "ok" status, errors "unknown", "timeout", or 5xx status codes in the /order/booking/finish/ response, you need to wait for the webhook within the agreed booking timeout (120 seconds, if we agree on this timeout). No notification within the agreed timeout means the order was unsuccessful.

Answer:

Yes, we agree to set the **booking timeout to 120 seconds** in the API key settings.

- From /order/booking/finish/ onward, we will wait up to 120 seconds for the webhook.
- If we receive completed within 120 seconds → booking is successful.
- If we receive failed or no webhook within 120 seconds → booking is unsuccessful.

Errors and Status Processing Logic

I have reviewed the provided table of errors and statuses processing and identified the following issues, comparing it against the recommended flow:

1. **/order/booking/finish/ - Status "ok", 5xx status code, Error "timeout", Error "unknown":**

Current:

Status "ok" - status on frontend: Booking or request completed successfully; logic on backed: Store API response in DB.- Display hotel info or booking confirmation. - Continue normal flow to next step (e.g., booking summary or payment page).

5xx status codes - status on frontend: Server temporarily unavailable. Please try again..; logic on backend: Log error details, retry request once after short delay, and if still failing - return standardized error response to frontend.

Error "timeout" - status on frontend: Request taking too long, please try again.; logic on backend: SImplement retry mechanism (e.g. 2 retries with exponential backoff). If still fails, mark request as failed and notify user.

Error "unknown" - status on frontend: Unexpected error occurred.; logic on backed: Log full payload and response for debugging; mark booking as failed with error_type = "unknown" for admin inspection.

Recommended: The recommended flow states that upon receiving "ok" status, 5xx status code, or errors "timeout", "unknown" from `/order/booking/finish/`, the booking should be with the status "Pending", and the system should **wait for webhook notification within the agreed timeout (which we will agree upon in booking cut-off section)**. It is incorrect to show the booking as successful or failed, or repeat the same request.

2. `Webhook`

Since **you use the webhooks** and do **not** use `/order/booking/finish/status`, the logic should be the following:

Recommended

- Status "completed": Status on frontend: **Success**; Logic on backend: **Stop calling any API methods and stop waiting for further webhooks. Update booking status to successful.**
- Status "failed": Status on frontend: **Failed**; Logic on backend: **Stop waiting for further webhooks and show the booking as failed**

Please review these questions/discrepancies and implement the necessary fixes to align your booking status handling process with the recommended flow. **Once the fixes are implemented, please let me know and confirm what exact changes have been made.**

Just in case, I'm attaching the full recommendation table below, marking the changes to be made in bold and green.

Additionally, please review the entire table and confirm that it aligns with the current system behavior.

Endpoint: <https://api.worldota.net/api/b2b/v3/hotel/order/booking/finish/>

ETG API	Status	How you must process
Status "ok"	Pending	Wait for the webhook notification within the agreed booking timeout (which we will agree upon in booking cut-off section)
5xx status code	Pending	Wait for the webhook notification within the agreed booking timeout (which we will agree upon in booking cut-off section)
Error "timeout"	Pending	Wait for the webhook notification within the agreed booking timeout (which we will agree upon in booking cut-off section)
Error "unknown"	Pending	Wait for the webhook notification within the agreed booking timeout (which we will agree upon in booking cut-off section)
Error "booking_form_expired"	Failed	Stop waiting for the webhook and show the booking as failed
Error "order_not_found"	Failed	Stop waiting for the webhook and show the booking as failed
Error "insufficient_b2b_balance"	Failed	Stop waiting for the webhook and show the booking as failed

Webhook

ETG webhook	Your status (that you show to the end-user)	How you process it and interact with our API (if you wait for webhook / if you call an API method (which method is called, if the method is called several times, with what frequency, for how long) / or if you stop calling any API methods and stop waiting for webhooks)
-------------	---	--

Status "completed"	Success	Stop calling any API methods and stop waiting for further webhooks. Update booking status to successful.
Status "failed"	Failed	Stop waiting for further webhook and show the booking as failed

Confirmation e-mails

You mentioned sending us the guests' personal email addresses in the /order/booking/finish/ requests.

For the "user" object in the /order/booking/finish/ request, the provided email address in the "email" field will receive the order confirmation letters.

As you use the B2B model, it is important that these confirmation letters, which will include net prices, are not sent to the end customer. Thus, please make sure to send the corporate email address of the partner in the user's "email" field; typically, this would be an email address with the partner's business domain. The email address should be different for different partners working through your platform.

Once you make the changes, please let me know.

Post-Booking

Retrieve bookings

As far as I understand, you are calling the "/order/info/" method during the booking process to show booking details. You call the "/order/info/" method immediately after the receiving webhook with the booking status.

The "/order/info/" method should be used to get booking information after the booking process is completed. It should not be called during the booking process. Due to internal synchronization processes, we cannot guarantee that the "/order/info/" method will return a response immediately after the order is confirmed.

- **What booking status will you display to the user if the “/order/info/” endpoint returns a blank response, while the webhook previously returned the “completed” status?**

I want to note that the final booking status should be obtained through the webhook. Only the "completed" status in the webhook should indicate successful order creation and booking confirmation.

- **Please fix the logic on your end and let me know.**

We have adjusted our logic:

Answer:

- **During** the booking process, we rely exclusively on /order/booking/finish/ + webhook.
- We do **not** call /order/info/ immediately after receiving the webhook.
- The final booking status for the user is based solely on the webhook status:
 - "completed" → success
 - "failed" or no webhook within timeout → failure
- We use /order/info/ only **after** the booking process is completed, for:
 - Internal admin tools (back office).
 - Post-booking synchronization if required.

If /order/info/ returns a blank response but the webhook is "completed", we still treat the booking as successful because the webhook is the source of truth, as you advise.