https://gemini.google.com/u/1/app/3332d7c5163a3dae

\* \* \* \* \*

### OpenAPI 3.1 Spec (for Module 4a - Accommodations)

YAML

```

openapi: 3.1.0

info:

title: Via di Francesco - Module 4a: Accommodations API

version: v1.0

description: API endpoints for managing and retrieving accommodation-specific data, including types, amenities, room configurations, payment methods, meal services, and reviews.

contact:

name: Platform Engineering Team

url: https://viafrancesco.example.com

email: api-dev@viafrancesco.example.com

servers:

- url: https://<YOUR\_PROJECT\_REF>.supabase.co/rest/v1 # Replace <YOUR\_PROJECT\_REF>

description: Supabase Production API

tags:

- name: Accommodations

description: Operations related to accommodation listings and their detailed attributes.

- name: AccommodationReviews

description: Operations related to user reviews for accommodations.

- name: AccommodationMeta

description: Read-only endpoints for accommodation-related master data (types, amenities, etc.).

components:

securitySchemes:

bearerAuth: # As defined in Auth & Security Blueprint

type: http

scheme: bearer

bearerFormat: JWT

description: Supabase JWT token for authenticated users.

apiKey: # As defined in Auth & Security Blueprint

type: apiKey

in: header

name: apikey

description: Supabase anonymous or service key.

schemas:

Error: # As defined in Auth & Security Blueprint [cite: 45]

type: object

required:

- code

- message

properties:

code:

type: string

description: Error code (PostgREST or application-specific).

example: "PGRST116"

message:

type: string

description: User-friendly error message.

example: "The requested resource was not found."

details:

type: string

nullable: true

description: Specific technical details or validation errors.

example: "No rows found for Waypoint ID 99999."

hint:

type: string

nullable: true

description: Suggestion on how to fix the error.

example: "Check the Waypoint ID and try again."

Pagination: # Global design standard [cite: 332]

type: object

properties:

total\_items:

type: integer

example: 150

total\_pages:

type: integer

example: 8

current\_page:

type: integer

example: 1

per\_page:

type: integer

example: 20

description: Pagination metadata for list responses.

# Localized Master Data Schemas (reflecting structure from v\_...\_localized views)

AccommodationTypeLocalized: # Based on v\_accommodation\_types\_localized [cite: 451]

type: object

description: Defines a type of accommodation, including all translations.

properties:

id:

type: integer

readOnly: true

code:

type: string

description: Unique, machine-readable code.

label: # Primary English label from base table [cite: 451]

type: string

description: Primary reference language (English) label.

description: # Primary English description from base table [cite: 451]

type: string

nullable: true

description: Primary reference language (English) description.

icon\_identifier:

type: string

nullable: true

sort\_order:

type: integer

is\_active:

type: boolean

# Audit columns from master table could be included if needed by API client

# created\_at, updated\_at, created\_by\_profile\_id, updated\_by\_profile\_id

all\_translations: # JSONB object from the view [cite: 451]

type: object

additionalProperties:

type: object

properties:

label: { type: string }

description: { type: string, nullable: true }

description: "All available translations for label and description, keyed by language code."

BookingStatusLocalized: # Based on v\_booking\_statuses\_localized [cite: 496]

type: object

description: Defines a booking availability status, including all translations.

properties:

id:

type: integer

readOnly: true

code:

type: string

label: # Primary English label

type: string

description: # Primary English description

type: string

nullable: true

is\_positive\_status:

type: boolean

sort\_order:

type: integer

is\_active:

type: boolean

all\_translations: # JSONB object from the view

type: object

additionalProperties:

type: object

properties:

label: { type: string }

description: { type: string, nullable: true }

description: "All available translations for label and description."

AmenityLocalized: # Based on v\_amenities\_localized [cite: 656]

type: object

description: Defines an amenity that can be offered, including all translations.

properties:

id:

type: integer

readOnly: true

amenity\_code:

type: string

name: # Primary English name from base table [cite: 656]

type: string

description: # Primary English description from base table [cite: 656]

type: string

nullable: true

icon\_identifier:

type: string

nullable: true

amenity\_category: # Primary English category from base table [cite: 656]

type: string

nullable: true

is\_common\_pilgrim\_need:

type: boolean

sort\_order:

type: integer

is\_active:

type: boolean

all\_translations: # JSONB object from the view [cite: 656]

type: object

additionalProperties:

type: object

properties:

name: { type: string }

description: { type: string, nullable: true }

amenity\_category: { type: string, nullable: true }

description: "All available translations for name, description, and amenity\_category."

RoomTypeLocalized: # Based on v\_room\_types\_localized [cite: 689]

type: object

description: Defines a type of room in an accommodation, including all translations.

properties:

id:

type: integer

readOnly: true

room\_type\_code:

type: string

name: # Primary English name

type: string

description: # Primary English description

type: string

nullable: true

typical\_capacity\_persons:

type: integer

is\_dorm\_style:

type: boolean

is\_private\_occupancy:

type: boolean

has\_private\_bathroom\_option:

type: boolean

is\_entire\_unit\_rental:

type: boolean

sort\_order:

type: integer

is\_active:

type: boolean

all\_translations: # JSONB object from the view

type: object

additionalProperties:

type: object

properties:

name: { type: string }

description: { type: string, nullable: true }

description: "All available translations for name and description."

PaymentMethodLocalized: # Based on v\_payment\_methods\_localized [cite: 623]

type: object

description: Defines an accepted payment method, including all translations.

properties:

id:

type: integer

readOnly: true

code:

type: string

label: # Primary English label

type: string

description: # Primary English description

type: string

nullable: true

icon\_identifier:

type: string

nullable: true

category: # Primary English category

type: string

nullable: true

sort\_order:

type: integer

is\_active:

type: boolean

all\_translations: # JSONB object from the view

type: object

additionalProperties:

type: object

properties:

label: { type: string }

description: { type: string, nullable: true }

category: { type: string, nullable: true }

description: "All available translations for label, description, and category."

MealServiceLocalized: # Based on v\_meal\_services\_localized [cite: 720]

type: object

description: Defines a meal service offered, including all translations.

properties:

id:

type: integer

readOnly: true

code:

type: string

name: # Primary English name

type: string

description: # Primary English description

type: string

nullable: true

icon\_identifier:

type: string

nullable: true

sort\_order:

type: integer

is\_active:

type: boolean

all\_translations: # JSONB object from the view

type: object

additionalProperties:

type: object

properties:

name: { type: string }

description: { type: string, nullable: true }

description: "All available translations for name and description."

# Schemas for Accommodation Endpoints

AccommodationListSummary: # For GET /accommodations list view [cite: 409]

type: object

properties:

waypoint\_id:

type: integer

format: int64

waypoint\_name:

type: string

description: Waypoint name, translated based on 'lang' query parameter.

waypoint\_slug:

type: string

latitude:

type: number

format: double

longitude:

type: number

format: double

elevation\_meters:

type: integer

nullable: true

town\_name:

type: string

nullable: true

description: Town name, translated based on 'lang' query parameter.

primary\_image\_thumbnail\_url:

type: string

format: url

nullable: true

accommodation\_type: # Embedded simplified master data

type: object

properties:

code:

type: string

label:

type: string

description: Accommodation type label, translated.

booking\_availability\_status:

type: object

properties:

code:

type: string

label:

type: string

description: Booking status label, translated.

is\_positive\_status:

type: boolean

capacity\_summary: # From accommodations\_capacity\_summary\_view

type: object

properties:

calculated\_total\_beds:

type: integer

brief\_description:

type: string

nullable: true

description: Brief description from waypoint, translated.

highlight\_amenity\_icons:

type: array

items:

type: string

description: Example list of key amenity icon identifiers.

AccommodationDetail: # For GET /accommodations/{waypoint\_id} [cite: 418]

type: object

properties:

waypoint\_id:

type: integer

format: int64

waypoint\_name:

type: string

description: Waypoint name, translated.

waypoint\_slug:

type: string

# ... other waypoint fields (geom, primary\_image full object etc.) from API Spec [cite: 419]

description: # From waypoint table

type: string

nullable: true

description: Waypoint description, translated.

accommodation\_type: # From v\_accommodation\_types\_localized

$ref: '#/components/schemas/AccommodationTypeLocalized'

booking\_status: # From v\_booking\_statuses\_localized

$ref: '#/components/schemas/BookingStatusLocalized'

# Direct fields from 'accommodations' table, translated

host\_name\_or\_organization:

type: string

nullable: true

description: Translated. [cite: 354]

contact\_person:

type: string

nullable: true

description: Translated.

phone\_number:

type: string

nullable: true

email\_address:

type: string

format: email

nullable: true

website\_url:

type: string

format: url

nullable: true

booking\_url\_primary:

type: string

format: url

nullable: true

booking\_notes:

type: string

nullable: true

description: Translated. [cite: 354]

check\_in\_time\_start:

type: string

format: time

nullable: true

check\_out\_time\_latest:

type: string

format: time

nullable: true

# ... other accommodations table fields (opening\_months etc.)

# Linked entities

amenities: # Array of linked amenities with their details [cite: 356]

type: array

items:

type: object

properties:

amenity\_details: # from v\_amenities\_localized

$ref: '#/components/schemas/AmenityLocalized'

notes\_on\_amenity: # from accommodation\_amenities, translated

type: string

nullable: true

room\_configurations: # Array [cite: 357]

type: array

items:

type: object

properties:

room\_type\_details: # from v\_room\_types\_localized

$ref: '#/components/schemas/RoomTypeLocalized'

count\_of\_this\_room\_type:

type: integer

price\_amount:

type: number

format: float

nullable: true

price\_currency\_code:

type: string

price\_is\_per\_person:

type: boolean

price\_notes: # translated

type: string

nullable: true

room\_specific\_notes: # translated

type: string

nullable: true

payment\_methods: # Array [cite: 359]

type: array

items:

type: object

properties:

payment\_method\_details: # from v\_payment\_methods\_localized

$ref: '#/components/schemas/PaymentMethodLocalized'

notes\_on\_method: # translated

type: string

nullable: true

meal\_services: # Array [cite: 360]

type: array

items:

type: object

properties:

meal\_service\_details: # from v\_meal\_services\_localized

$ref: '#/components/schemas/MealServiceLocalized'

price\_amount:

type: number

format: float

nullable: true

price\_currency\_code:

type: string

price\_is\_per\_person:

type: boolean

availability\_and\_timing\_notes: # translated

type: string

nullable: true

reviews: # Array [cite: 361]

type: array

items:

$ref: '#/components/schemas/AccommodationReviewResponse'

capacity\_summary: # From view [cite: 355]

type: object

properties:

calculated\_total\_beds:

type: integer

calculated\_total\_rooms:

type: integer

AccommodationReviewResponse: # For GET .../reviews array [cite: 362, 1477]

type: object

properties:

id:

type: integer

format: int64

review\_title: # Translated

type: string

nullable: true

review\_body: # Translated

type: string

language\_code: # Original language of review

type: string

stay\_date:

type: string

format: date

nullable: true

overall\_vote:

type: string # from vote\_type\_enum

nullable: true

created\_at:

type: string

format: date-time

author: # Simplified public author info

type: object

properties:

# profile\_id: { type: string, format: uuid } # May not be public

display\_name:

type: string

nullable: true # From profiles.public\_display\_name

avatar\_url:

type: string

format: url

nullable: true # From profiles.public\_avatar\_media\_id

AccommodationReviewSubmission: # For POST /reviews [cite: 1477]

type: object

required:

- accommodation\_waypoint\_id

- review\_body

- language\_code

properties:

accommodation\_waypoint\_id:

type: integer

format: int64

review\_title:

type: string

maxLength: 255

nullable: true

review\_body:

type: string

minLength: 1

maxLength: 2000

language\_code: # Explicitly set by client for the submitted text

type: string

example: "en"

stay\_date:

type: string

format: date

nullable: true

overall\_vote:

type: string # vote\_type\_enum (e.g., "up", "down")

nullable: true

AccommodationReviewUpdate: # For user PUT /reviews/{id} [cite: 1529]

type: object

properties:

review\_title:

type: string

maxLength: 255

nullable: true

review\_body:

type: string

minLength: 1

maxLength: 2000

language\_code: # If user can correct submission language

type: string

stay\_date:

type: string

format: date

nullable: true

overall\_vote:

type: string # vote\_type\_enum

nullable: true

ReviewModerationUpdate: # For PATCH /admin/reviews/{id}/status [cite: 1477]

type: object

required:

- moderation\_status

properties:

moderation\_status:

type: string # content\_moderation\_status\_enum (e.g. "approved\_visible")

moderation\_notes\_internal:

type: string

nullable: true

security: # Global security definition

- bearerAuth: []

- apiKey: []

paths:

/accommodations:

get:

tags:

- Accommodations

summary: List accommodations

description: Retrieves a paginated list of published accommodations, with filtering options.

operationId: listAccommodations

security:

- apiKey: [] # Public access, only API key needed

parameters:

- name: lang

in: query

required: false

schema: { type: string, example: "it" }

description: Language code for localized content. [cite: 326]

- name: page

in: query

required: false

schema: { type: integer, default: 1, minimum: 1 }

description: Page number for pagination. [cite: 331]

- name: per\_page

in: query

required: false

schema: { type: integer, default: 20, minimum: 1, maximum: 100 }

description: Number of items per page. [cite: 331]

- name: accommodation\_type\_codes

in: query

required: false

schema: { type: string }

description: Comma-separated list of accommodation\_types\_master.code values. [cite: 337]

- name: amenity\_codes

in: query

required: false

schema: { type: string }

description: Comma-separated list of amenities\_master.amenity\_code values (logical AND). [cite: 338]

- name: min\_beds

in: query

required: false

schema: { type: integer, minimum: 1 }

description: Minimum total beds required. # relies on accommodations\_capacity\_summary\_view [cite: 338]

responses:

'200':

description: A paginated list of accommodation summaries.

content:

application/json:

schema:

type: object

properties:

data:

type: array

items:

$ref: '#/components/schemas/AccommodationListSummary'

pagination:

$ref: '#/components/schemas/Pagination'

default:

description: Unexpected error.

content:

application/json:

schema:

$ref: '#/components/schemas/Error'

/accommodations/{waypoint\_id}:

get:

tags:

- Accommodations

summary: Get accommodation details

description: Retrieves comprehensive details for a single accommodation by its waypoint\_id.

operationId: getAccommodationByWaypointId

security:

- apiKey: [] # Public access

parameters:

- name: waypoint\_id

in: path

required: true

schema: { type: integer, format: int64 }

description: The waypoint ID of the accommodation. [cite: 415]

- name: lang

in: query

required: false

schema: { type: string, example: "it" }

description: Language code for localized content. [cite: 349]

- name: reviews\_limit

in: query

required: false

schema: { type: integer, default: 3, minimum: 0 }

description: Number of recent reviews to include. [cite: 350]

responses:

'200':

description: Detailed information about the accommodation.

content:

application/json:

schema:

$ref: '#/components/schemas/AccommodationDetail'

# This response is complex, leveraging a DB function (e.g. get\_public\_accommodation\_details) and localized views is key. [cite: 368, 434]

'404': # [cite: 323]

description: Accommodation not found.

content:

application/json:

schema:

$ref: '#/components/schemas/Error'

default:

description: Unexpected error.

content:

application/json:

schema:

$ref: '#/components/schemas/Error'

/reviews: # For accommodation\_reviews

post:

tags:

- AccommodationReviews

summary: Submit a new review for an accommodation

description: Allows an authenticated pilgrim to submit a review for an accommodation. [cite: 1474]

operationId: submitAccommodationReview

security:

- bearerAuth: [] # Requires pilgrim role [cite: 1528]

- apiKey: []

requestBody:

required: true

content:

application/json:

schema:

$ref: '#/components/schemas/AccommodationReviewSubmission'

responses:

'201': # [cite: 320]

description: Review successfully submitted and pending moderation.

content:

application/json:

schema:

$ref: '#/components/schemas/AccommodationReviewResponse' # Returns the created review, with moderation\_status set

'400': { $ref: '#/components/responses/BadRequest' } # [cite: 321]

'401': { $ref: '#/components/responses/Unauthorized' } # [cite: 322]

'403': { $ref: '#/components/responses/Forbidden' } # [cite: 322]

default: { $ref: '#/components/responses/DefaultError' }

/reviews/{review\_id}:

put: # Exemplar WRITE endpoint for user self-management

tags:

- AccommodationReviews

summary: Update an existing review by its author

description: Allows an authenticated user to update their own review, typically if it is still pending approval. [cite: 1529]

operationId: updateMyAccommodationReview

security:

- bearerAuth: [] # Requires ownership & specific RLS conditions

- apiKey: []

parameters:

- name: review\_id

in: path

required: true

schema: { type: integer, format: int64 }

description: The ID of the review to update.

requestBody:

required: true

content:

application/json:

schema:

$ref: '#/components/schemas/AccommodationReviewUpdate'

responses:

'200':

description: Review successfully updated.

content:

application/json:

schema:

$ref: '#/components/schemas/AccommodationReviewResponse'

'403': { $ref: '#/components/responses/Forbidden' }

'404': { $ref: '#/components/responses/NotFound' }

default: { $ref: '#/components/responses/DefaultError' }

# Meta endpoints for master data (using localized views)

/meta/accommodation-types:

get:

tags:

- AccommodationMeta

summary: List accommodation types

description: Retrieves all active accommodation types with localized information.

operationId: listMetaAccommodationTypes

security:

- apiKey: [] # Public

parameters:

- name: lang

in: query

required: false

schema: { type: string, example: "it" }

description: Preferred language. The 'all\_translations' object will always be provided by the view.

responses:

'200':

description: A list of accommodation types.

content:

application/json:

schema:

type: array

items:

$ref: '#/components/schemas/AccommodationTypeLocalized' # Based on v\_accommodation\_types\_localized

# This endpoint should query public.v\_accommodation\_types\_localized view [cite: 448]

default: { $ref: '#/components/responses/DefaultError' }

/meta/booking-statuses:

get:

tags:

- AccommodationMeta

summary: List booking statuses

operationId: listMetaBookingStatuses

security: { apiKey: [] }

parameters: [{ name: lang, in: query, required: false, schema: { type: string } }]

responses:

'200':

description: A list of booking statuses.

content: { application/json: { schema: { type: array, items: { $ref: '#/components/schemas/BookingStatusLocalized' } } } }

# Queries public.v\_booking\_statuses\_localized view [cite: 494]

default: { $ref: '#/components/responses/DefaultError' }

/meta/amenities:

get:

tags:

- AccommodationMeta

summary: List amenities

operationId: listMetaAmenities

security: { apiKey: [] }

parameters: [{ name: lang, in: query, required: false, schema: { type: string } }]

responses:

'200':

description: A list of amenities.

content: { application/json: { schema: { type: array, items: { $ref: '#/components/schemas/AmenityLocalized' } } } }

# Queries public.v\_amenities\_localized view [cite: 654]

default: { $ref: '#/components/responses/DefaultError' }

/meta/room-types:

get:

tags:

- AccommodationMeta

summary: List room types

operationId: listMetaRoomTypes

security: { apiKey: [] }

parameters: [{ name: lang, in: query, required: false, schema: { type: string } }]

responses:

'200':

description: A list of room types.

content: { application/json: { schema: { type: array, items: { $ref: '#/components/schemas/RoomTypeLocalized' } } } }

# Queries public.v\_room\_types\_localized view [cite: 687]

default: { $ref: '#/components/responses/DefaultError' }

/meta/payment-methods:

get:

tags:

- AccommodationMeta

summary: List payment methods

operationId: listMetaPaymentMethods

security: { apiKey: [] }

parameters: [{ name: lang, in: query, required: false, schema: { type: string } }]

responses:

'200':

description: A list of payment methods.

content: { application/json: { schema: { type: array, items: { $ref: '#/components/schemas/PaymentMethodLocalized' } } } }

# Queries public.v\_payment\_methods\_localized view [cite: 621]

default: { $ref: '#/components/responses/DefaultError' }

/meta/meal-services:

get:

tags:

- AccommodationMeta

summary: List meal services

operationId: listMetaMealServices

security: { apiKey: [] }

parameters: [{ name: lang, in: query, required: false, schema: { type: string } }]

responses:

'200':

description: A list of meal services.

content: { application/json: { schema: { type: array, items: { $ref: '#/components/schemas/MealServiceLocalized' } } } }

# Queries public.v\_meal\_services\_localized view [cite: 718]

default: { $ref: '#/components/responses/DefaultError' }

components: # Re-declaring to ensure it's at the root, as per strict OpenAPI structure

responses:

NotFound:

description: The specified resource was not found.

content:

application/json:

schema:

$ref: '#/components/schemas/Error'

Unauthorized:

description: Authentication information is missing or invalid.

content:

application/json:

schema:

$ref: '#/components/schemas/Error'

Forbidden:

description: Authenticated user does not have permission for this action.

content:

application/json:

schema:

$ref: '#/components/schemas/Error'

BadRequest:

description: The request was malformed or contained invalid parameters.

content:

application/json:

schema:

$ref: '#/components/schemas/Error'

DefaultError: # For unhandled or generic errors

description: Unexpected error.

content:

application/json:

schema:

$ref: '#/components/schemas/Error'

```

\* \* \* \* \*

### Quick-Start Examples

1\. List Accommodations with Filters (Public)

Bash

```

curl -X GET\

"https://<YOUR\_PROJECT\_REF>.supabase.co/rest/v1/accommodations?lang=it&per\_page=1&accommodation\_type\_codes=b\_and\_b&amenity\_codes=WIFI\_INTERNET"\

-H "apikey: <SUPABASE\_ANON\_KEY>"\

-H "Content-Type: application/json"

```

Sample JSON Response:

JSON

```

{

"data": [

{

"waypoint\_id": 123,

"waypoint\_name": "B&B Sogni D'oro", // Italian, due to lang=it

"waypoint\_slug": "b-and-b-sogni-d-oro",

"accommodation\_type": {

"code": "b\_and\_b",

"label": "Bed & Breakfast" // Italian

},

"booking\_availability\_status": {

"code": "open\_bookings\_available",

"label": "Aperto - Prenotazioni Disponibili", // Italian

"is\_positive\_status": true

}

// ... other simplified fields ...

}

],

"pagination": {

"total\_items": 1,

"total\_pages": 1,

"current\_page": 1,

"per\_page": 1

}

}

```

2\. Get Accommodation Details with Localized Master Data (Public)

Bash

```

curl -X GET\

"https://<YOUR\_PROJECT\_REF>.supabase.co/rest/v1/accommodations/123?lang=en&reviews\_limit=1"\

-H "apikey: <SUPABASE\_ANON\_KEY>"\

-H "Content-Type: application/json"

```

Sample JSON Response (Snippet):

JSON

```

{

"waypoint\_id": 123,

"waypoint\_name": "B&B Sweet Dreams", // English (lang=en)

"accommodation\_details": {

"host\_name\_or\_organization": "Mario Rossi", // English

"booking\_notes": "Call for seasonal discounts.", // English

"accommodation\_type": { // Data from v\_accommodation\_types\_localized or DB function

"id": 1, // Example ID

"code": "b\_and\_b",

"label": "Bed & Breakfast", // English label from view's base field

"description": "Accommodation in a private home offering breakfast.", // English description

"icon\_identifier": "icon-b-and-b",

"sort\_order": 20,

"is\_active": true,

"all\_translations": { // The full translation object from the view

"en": {"label": "Bed & Breakfast", "description": "Accommodation in a private home offering breakfast."},

"it": {"label": "Bed & Breakfast", "description": "Alloggio in casa privata che offre la colazione."}

}

}

// ... other details ...

},

"amenities": [

{

"amenity\_details": { // From v\_amenities\_localized

"id": 5, // Example ID

"amenity\_code": "WIFI\_INTERNET",

"name": "Wi-Fi Internet", // English name from view's base field

"description": "Access to wireless internet.",

"icon\_identifier": "wifi",

"amenity\_category": "Connectivity",

"is\_common\_pilgrim\_need": true,

"sort\_order": 10,

"is\_active": true,

"all\_translations": {

"en": {"name": "Wi-Fi Internet", "description": "Access to wireless internet.", "amenity\_category": "Connectivity"},

"it": {"name": "Internet Wi-Fi", "description": "Accesso a internet wireless.", "amenity\_category": "Connettività"}

}

},

"notes\_on\_amenity": "Available in common areas." // English, from accommodation\_amenities

}

]

// ... other sections like room\_configurations, reviews ...

}

```

3\. Submit an Accommodation Review (Authenticated Pilgrim)

Bash

```

curl -X POST\

"https://<YOUR\_PROJECT\_REF>.supabase.co/rest/v1/reviews"\

-H "apikey: <SUPABASE\_ANON\_KEY>"\

-H "Authorization: Bearer <PILGRIM\_JWT\_ACCESS\_TOKEN>"\

-H "Content-Type: application/json"\

-H "Prefer: return=representation"\

-d '{

"accommodation\_waypoint\_id": 123,

"review\_title": "A wonderful refuge",

"review\_body": "The hosts were incredibly kind and the food was amazing. A true pilgrim spirit here!",

"language\_code": "en",

"stay\_date": "2025-04-10",

"overall\_vote": "up"

}'

```

Sample JSON Response (201 Created):

JSON

```

{

"id": 11,

"accommodation\_waypoint\_id": 123,

"profile\_id": "<pilgrim\_user\_uuid\_from\_token>",

"review\_title": "A wonderful refuge", // English, as submitted with language\_code "en"

"review\_body": "The hosts were incredibly kind and the food was amazing. A true pilgrim spirit here!",

"language\_code": "en",

"stay\_date": "2025-04-10",

"overall\_vote": "up",

"moderation\_status": "pending\_approval",

"is\_publicly\_visible": false,

"created\_at": "2025-05-18T20:30:00Z"

// ... other fields from AccommodationReviewResponse schema

}

```

\* \* \* \* \*

### Implementation Notes

- SQL Views/Indexes for Performance:

- The localized views (e.g., `v\_accommodation\_types\_localized`, `v\_amenities\_localized` ) are fundamental for the `/meta/\*` endpoints and for populating linked master data within the `GET /accommodations/{waypoint\_id}` response. Their performance relies on a robust composite index on `public.translations (table\_identifier, row\_foreign\_key, language\_code, column\_identifier)`.

- The `public.accommodations\_capacity\_summary\_view` is essential for the `min\_beds` filter and displaying capacity.

- A GIN index on `accommodations.opening\_months` is necessary for the `opening\_month` filter.

- All foreign keys involved in joins must be indexed.

- Endpoints Needing Special Attention:

- `GET /accommodations/{waypoint\_id}`: This endpoint aggregates significant data. A PostgreSQL function (e.g., `get\_public\_accommodation\_details(p\_waypoint\_id BIGINT, p\_lang TEXT) RETURNS JSONB`) is strongly recommended to encapsulate join logic, handle translations (utilizing the localized views or direct joins), and structure the JSON output efficiently. This will simplify the API layer and allow database-level optimizations. Caching for this endpoint is advisable.

- `GET /accommodations`: Requires pagination. Caching common filter sets would improve user experience.

- All `/meta/\*` endpoints: Good candidates for application-level or CDN caching as master data changes infrequently. If the underlying localized views face performance issues with very large translation sets, converting them to materialized views could be considered in a future iteration.

- Schema Assessment:

- 🟢 Schema OK. The database schema for Module 4a, including the master tables with direct English storage and linked translations, junction tables, `accommodations`, and `accommodation\_reviews`, effectively supports the API design. The RLS policies defined in individual table specs provide the necessary security controls.

- Localization Strategy: The API will accept a `lang` parameter.

- For direct translatable fields on entities (e.g., `accommodations.booking\_notes`, `accommodation\_reviews.review\_body`), the API backend (preferably via a DB function) will fetch the translation from `public.translations` if `lang` is provided and a translation exists, otherwise defaulting to the English text from the base column. The OpenAPI schema shows a single field (e.g., `booking\_notes`) which would contain the resolved text.

- For linked master data in detailed responses (e.g., the `accommodation\_type` object within `AccommodationDetail`), the API can leverage the corresponding localized view (e.g., `v\_accommodation\_types\_localized`). The response can include the `label` (already resolved to the requested `lang` or English by the view/function logic if the view is queried with a language filter) and the full `all\_translations` JSONB object, giving the client flexibility.

- Security Implementation: RLS policies in the database, using helper functions like `public.has\_role\_on\_profile(auth.uid(), 'role\_code')` (as per Auth & Security Blueprint ), will enforce access control. The API endpoints will rely on Supabase Auth for JWT validation and role information.