

GET /v2/charter (Multi?Mode Charter Loader)

Charter: Get Charter(s) Details

رتراچ کی دناوتی م، ی دورو هب هتسب .تس اه رتراچ تاغالطای بایزاب ی لصا هتسه ،تنی و پدنا نی راهچ دناوتی م `action` رادقم هب هتسب نی چمه و ،هی آ تروص هب ار رتراچ نی دنچ ای ،دنا درگرب ار دجاو Endpoint نی **all / list / base / calculations / capacity** :دنا درگرب ار اه داد زا فل تخم حطس .تساتی د نی گنس رایسب یزاسی نغ لماش هک دن کی م هدا فتسا `CharterResource` هدیچی پ سالک

Request Overview

URL: `/v2/charter`

Method: GET

Controller: CharterController@indexCharter

Middleware: authWithJwt

Behavior Summary

- دن کی م دول ار رتراچ کی طوقف → دشاب ددع کی `id` رگا
- دن کی م دول ار رتراچ نی دنچ → دشاب هی آ `id` رگا
- دنوشی م هداوخ `charters` لودج زا اهدروکر ی مامت
- دوشی م لی دبت `new CharterResource()` اب یی اهن هداد

Query Parameters

Field	Type	Required	Description
id	integer array	yes	دنچ ای رتراچ کی هسانش رتراچ

Field	Type	Required	Description
action	string	optional	نویسی عملیات خروجی دکمه: all list base calculations capacity
item_id	integer	optional	ردیف action = capacity

Single Charter Mode

```
if (is_numeric($request->id)) {  
    $charter = DB::table('charters')->where('id', $request->id)->first();  
}
```

دوستان تفریحی:

```
{  
    "status": false,  
    "time": 1710000000,  
    "message": "item not found!"  
}
```

Multiple Charter Mode

```
$charters = DB::table('charters')->whereIn('id', $request->id)->get();
```

دوستان تفریحی map در هر مورد:

```
$charters->map(fn($item) => new CharterResource($item))
```

CharterResource Overview

تعارف شما، اماک یا راه‌های دسترسی، لیست دریا، لوئیسم گریزب رایسب سالک نی
تسا رگید توافقتم هداد عون اهه و، اه‌گژی و، ورزرتی فرط، اه‌متی، نی‌ناوق، تاب‌ساح، روتارپا

Modes (action)

- **all:** در اوم همه ل م اش . دوم ن یرت ل م اک
- **list:** یر گت س ی ل ی ا ر ب ک ب س ه خ س ن
- **base:** ا ه م ت ی آ + ن ی ن ا و ق + ه ی ا پ ت ا ع ا ل ط ا
- **calculations:** ی ل ا م ت ا ب س ا ح م و ا ه م ت ی آ ط ق ف
- **capacity:** (یر ا ی ت خ ا item_id ا ب) و ر ز ر ی ا ه ت ی ف ر ط ط ق ف

Shared Data Loaders (Resource Level)

- ر و ت ا ر پ ا : ل و د ج operators
- ل و د ج : ا ه م ت ی آ charter_items
- ل و د ج : ت ا ب س ا ح م {table}_calculation
- د ا د ر ت س ا ن ی ن ا و ق : charter_refund_rules
- ی م و م ع ن ی ن ا و ق : charter_public_rules
- و ر ز ر ت ی ف ر ط : ReservationController::capacityItemCharter()
- ی ل ک ت ا ع ا ل ط ا : getInformation()
- ل و د ج : ه د ش ی د ن ب ن ا م ز ن ش ک ی ف ی ت و ن scheduled_notifications

sortItems() Logic

enrich ا ر ا ت ی د م ت ی آ method/submethod ه ب ه ت س ب و ت س ا ه ش خ ب ن یرت ن ی گ ن س ز ا ی ک ی ع ب ا ت ن ی ا د ن ک ی م :

- method = route (aircraft, train, bus, ship)
- method = accommodation
- method = service
- method = custom

ل م اش ت ا ی ئ ج :

- ا یر ل ک ن ی ل airline, airports, aircraft
- ت م ر ف → ه ق ی ق د ی و ر ز ا duration ه ب س ا ح م
- ا ه ر ه ش و a supplier, ا ه ت ک ر ش پ آ ک و ل
- و company و vehicle ن د ر ک normalize

sortCalculations() Logic

ل م اش ت م س ق ن ی ا :

- ل و د ج : ا ه ت ا ی ل ا م د و ل charter_taxes
- ت ا م د خ د و ل : charter_services + charter_service_categories

- type و subtype ر ب ین تبم تاب ساجم
- یارب mapping rule:
 - commission / markup / citizenship
 - charter_financial_handling لودج زا
- financial.base = اهت می ق
- financial.staircases
- financial.taxes
- financial.commissions
- financial.markups
- financial.citizenship

getInformation() Logic

دی لک تحت دنوش می شک Redis رد ری دا قم . دن کی م دی لوت تو اف تم blocks و title → type ه ب هت س ب

```
charter:{id}:information:title
```

- tour → origin, destination, start, end
- route → origin, destination, start
- accommodation → destination, start, end (+ hotel title)
- service → از ی و/ه می ب ل ث م س ی ورس تا ع ا ل ط ا /CIP

Response Structure (Single Charter, action=all)

```
{
  "status": true,
  "time": 1710000000,
  "data": {
    "id": 123,
    "serial": 10000 + serial,
    "title": { fa, en },
    "type": "...",
    "subtype": "...",
    "capacity": ...,
    "buy": ...,
    "deadline": ...,
    "information": {...},
    "reservation": {...},
    "calculations": [...],
    "rules": { refund: [...], public: [...] },
    "operator": {...},
    "items": [...],
  }
}
```

```
"details": {...},
"plan": ...,
"slug": ...,
"hub": ...,
"schedule": {...} | false,
"status": ...
}
}
```

Error Response

```
{
  "status": false,
  "time": 1710000000,
  "message": "item not found!"
}
```

Flowchart

Validate JWT

↓

Check id type (numeric | array)

↓

Load charter(s)

↓

If empty → return 'item not found'

↓

For each → new CharterResource()

↓

Inside Resource → action switch

↓

Load items, rules, calculations, information

↓

SortItems() + SortCalculations()

↓

Return Final JSON

Revision #1

Created 2025-12-03 18:36:37 UTC by Alireza

Updated 2025-12-03 18:37:25 UTC by Alireza