

# Aabode.com API Documentation



## PREFACE

---

Aabode.com has recently updated its APIs that will be integrated with Channel Managers Platform.

Aabode.com

---

**COMPANY**

Aabode.com

---

**CONTACT US**

[support@aabode.com](mailto:support@aabode.com)

## CONTENTS

---

Preface .....	2
Company .....	2
Contact us .....	2
1 Getting Started.....	6
1.1 Introduction .....	6
1.2 Versions.....	6
2 Authentication .....	7
2.1 Aabode.com beta site authentication .....	7
2.2 Aabode.com production site authentication.....	7
3 Response Codes .....	8
4 REST APIs.....	9
4.1 Rental Type .....	9
4.1.1 Create a rental type .....	9
4.1.2 List all rental types .....	10
4.1.3 Retrieve a rental type.....	11
4.1.4 Update a rental type .....	11
4.2 Manager.....	12
4.2.1 Retrieve a property manager.....	12
4.3 Rental .....	13
4.3.1 Retrieve a properties list.....	13
4.4 Rental Basic.....	14
4.4.1 Create a rental basic .....	14
4.4.2 Update a rental basic .....	16
4.4.3 Retrieve a rental basic.....	17
4.5 Rooms .....	18
4.5.1 Room Details .....	18
4.5.2 Beds.....	21
4.6 Features .....	24
4.6.1 Update rental features.....	25
4.6.2 Retrieve rental features .....	38
4.7 Description .....	40

4.7.1	Update rental description .....	41
4.7.2	Retrieve rental descriptions .....	42
4.8	Pictures .....	44
4.8.1	Update Rental Pictures .....	44
4.8.2	Set Picture URL.....	47
4.8.3	Get Review Pictures .....	47
4.8.4	Retrieve rental pictures .....	48
4.9	Rates.....	49
4.9.1	Prices.....	50
4.9.2	Rate Details .....	53
4.9.3	Additional Charges in Rates .....	54
4.9.4	Promotions in Rates .....	56
4.9.5	Special Owner Rules in Rates.....	57
4.9.6	Cancellation Policy in Rates .....	62
4.9.7	Check-in instructions.....	63
4.10	Calendar .....	63
4.10.1	Retrieve allowance of instant booking .....	63
4.10.2	Retrieve availabilities .....	64
4.10.3	Retrieve lcal.....	66
4.10.4	Retrieve ical file per room.....	67
4.10.5	Retrieve ical file per rental.....	67
4.11	Location.....	68
4.11.1	Retrieve a rental location.....	68
4.11.2	Add and update a rental location .....	68
4.12	Payment .....	69
4.12.1	Add and update a rental payment.....	69
4.12.2	Retrieve a rental payment .....	69
4.13	Bundle .....	70
4.13.1	Retrieve a rental subscription plan.....	70
4.14	Search.....	71
4.14.1	Retrieve Search Rentals .....	71
4.15	Bookings.....	75
4.15.1	Retrieve a booking list.....	75

4.15.2	Create an Instant booking.....	76
4.15.3	Create Request a quote for booking.....	77
4.15.4	Cancel booking.....	78

Aabode.com

# 1 GETTING STARTED

---

## 1.1 INTRODUCTION

The purpose of Aabode.com API is to allow the property owners and managing agents to spend less time in managing their properties and their rates.

By using APIs, Aabode.com is an open source can be integrated with any channel manager to manage rental properties or rooms, update features, descriptions, images, pricing/rates and availability from a single point, making it quicker, more accurate and more productive. APIs have the capability to access certain components of a website, and therefore the delivery of information from one to the other is faster and more flexible, allowing content generated to be published automatically and is available for every connected channel.

The Aabode API is organized around REST. Our API has predictable resource-oriented URLs, accepts form-encoded request bodies, returns JSON-encoded responses, and uses standard HTTP response codes, authentication, and verbs.

## 1.2 VERSIONS

The current version is V2.X and it was released on 2020-11-30. Read our API changelog and feel free to learn more about backwards compatibility. A lot of functionalities of Aabode.com are made to the New APIs.

All requests use your account API key settings, unless you override the API version.

*Table 1.Revision History*

Date	Version	Author	Revision
01/14/2021	2.1	Oleksandr	Add an endpoint that is getting property manager id
01/29/2021	2.2	Oleksandr	<ul style="list-style-type: none"><li>•Add endpoints regarding to aabode.com bed types. Refer added 4.5.2 Beds</li><li>•Update sample response of rental pictures information including 1200x900 size image path</li><li>•Update instant booking creation parameters explanantions</li></ul>

## 2 AUTHENTICATION

The Aabode API uses API keys to authenticate requests. Your API keys carry many privileges, so be sure to keep them secure! Do not share your secret API keys in publicly accessible areas such as GitHub, client-side code, and so forth.

To get API key for Beta and Live production system of Aabode.com, the API user (e.g., Channel Manager platforms) must be registered in Aabode system (Beta and Live Production) and contact to the Aabode.com supporting team (support@aabode.com).

### 2.1 AABODE.COM BETA SITE AUTHENTICATION

The Beta system is the testing platform of Aabode.com. The request to the Beta system requires Beta API key. The request is unique according to the individual API user by using API key. The API user must be registered in Beta system. Aabode.com supporting team will provide the Beta API key after auditing the Beta API user.

In Beta system, API authentication process is following:

- Root URL: <http://beta.aabode.com>
- API URL: <http://beta.aabode.com/api/v2>
- Content-type: multipart/form-data
- The api authentication requires to the request to add an additional request header called "apikey".

For example:

```
curl GET http://beta.aabode.com/api/v2/rental/types \  
-H "content-type: multipart/form-data" \  
-H "apikey: XXXXXXXXXXXX"
```

### 2.2 AABODE.COM PRODUCTION SITE AUTHENTICATION

Aabode.com is the live production platform of Aabode. The request to the production system of Aabode requires Production API key. The request is unique according to the individual API user by using API key. The API user must be registered in Production system of Aabode. Aabode.com supporting team will provide the Production API key after auditing the Production API user.

In Production system, API authentication process is following:

- Root URL: <https://aabode.com>
- API URL: <https://aabode.com/api/v2>
- Content-type: multipart/form-data
- The api authentication requires to the request to add an additional request header called "apikey".

For example:

```
curl -X GET https://aabode.com/api/v2/rental/types \  
-H "content-type: multipart/form-data" \  
-H "apikey: XXXXXXXXXXXX"
```

### 3 RESPONSE CODES

---

Aabode uses conventional HTTP response codes to indicate the success or failure of an API request. In general: Codes in the 2xx range indicates success. Codes in the 4xx range indicate an error that failed given the information provided (e.g., a required parameter was omitted, a payment failed, etc.). Codes in the 5xx range indicate an error with Aabode's servers (these are rare).

*Table 2. HTTP status code summary*

Code	Description
<b>200 – OK</b>	Everything worked as expected.
<b>400 – Bad Request</b>	The Request was unacceptable, often due to missing a required parameter.
<b>401 – Unauthorized</b>	No Valid API key provided.
<b>403 – Forbidden</b>	The API key doesn't have permissions to perform the request.
<b>404 – Not Found</b>	The requested resource doesn't exist.
<b>500 – Internal Server Errors</b>	Something went wrong on Aabode's end.



## 4 REST APIs

### 4.1 RENTAL TYPE

Allows user to CRUD Rental Type information of the rental property.

Table 3. Rental Type Endpoints

REQUEST METHOD	ENDPOINTS
GET	/api/v2/rental/types
POST	/api/v2/rental/types
GET	/api/v2/rental/types/:id
POST	/api/v2/rental/types/:id

Table 4. Rental Type Object

Attributes	Data Type	Description
lang	int	Rental Type
inputName	String	Rental Type Name
inputDescription	String	Rental Type Description

#### 4.1.1 Create a rental type

**POST** /api/v2/rental/types

Create a Property Rental Type object.

##### URL Parameters

No parameters

##### Request Body Parameters

Parameters	Required/Optional	Description	Example
lang	Optional	Language ID for a rental type	0
inputName	Required	Rental Type Name	Test Rental Type
inputDescription	Optional	Rental Type Description	This is the test rental type.

##### Returns

Returns the rental type object if the update succeeded. Returns an error if create parameters are invalid.

##### Sample

- Request:  
**POST** http://beta.aabode.com/api/v2/rental/types
- Response:

```
{  
  "Name": [  

```

```
"Test Rental Type"  
]  
}
```

#### 4.1.2 List all rental types

**GET** /api/v2/rental/types

It returns a list of rental types for aabode.com previously created. The rental types are returned in rental type id.

##### URL parameters

No parameters

##### Request Body Parameters

No parameters

##### Returns

Array list of rental property types. Each entry in the array is a separate rental type object. If no more rental types are available, the resulting array will be empty. This request should never raise an error.

##### Sample

- Request:  
**GET** <http://beta.aabode.com/api/v2/rental/types>
- Response:

```
[  
  {  
    "name": [  
      "Apartment"  
    ]  
  },  
  {  
    "name": [  
      "House"  
    ]  
  },  
  {...}  
  ...  
]
```

#### 4.1.3 Retrieve a rental type

**GET** /api/v2/rental/types/:id

It retrieves the rental type with the given ID.

##### URL Parameters

Parameters	Required/Optional	Description	Example
id	Required	Rental Type ID	12

##### Request Body Parameters

No parameters

##### Returns

It returns a rental type if a valid rental type ID was provided, otherwise this raises an error!

##### Sample

- Request:  
`GET http://beta.aabode.com/api/v2/rental/types/12`
- Response:

```
{
  "name": [
    "Cottage"
  ]
}
```

#### 4.1.4 Update a rental type

**POST** /api/v2/rental/types/:id

Updates the specified rental types by setting the values of the parameters passed. Any parameters not provided are left unchanged.

##### URL parameters

Parameters	Required/Optional	Description	Example
id	Required	Rental Type ID	12

##### Request Body Parameters

Attributes	Required/Optional	Description	Example
lang	Optional	Language ID for a rental type	12
inputName	Required	Rental Type Name	Apartment
inputDescription	Optional	Rental Type Description	A room.

### Returns

Returns the rental type object if the update succeeded. Otherwise, it returns an error if update parameters are invalid.

### Sample

- Request:  
`POST http://beta.aabode.com/api/v2/rental/types/12`
- Response:

```
{
  "name": [
    "Cottage 1"
  ]
}
```

## 4.2 MANAGER

### 4.2.1 Retrieve a property manager

Get a property manager information of specific manager's email

#### Endpoint & method

[GET](#) /api/v2/managers

#### URL parameters

No parameters.

#### Request Query Params

Parameters	Required/Optional	Description	Example
Email	Required	Email address of the property manager	example@ex.com

#### Request Body Params

No parameters

### Returns

Return json data of the id for the specific property manager.

### Sample

- Request:  
`GET http://beta.aabode.com/api/v2/managers?email=oleks1203@outlook.com`
- Response:

```
{
```

```
"manager_id": 22501
}
```

## 4.3 RENTAL

Allow users to get a list of properties belong to the rental property manager.

### 4.3.1 Retrieve a properties list

Get a list of properties belong to the rental property manager.

#### Endpoints & Method

**GET** /api/v2/managers/{manager\_id}/rentals

#### URL parameters

Parameters	Required/Optional	Description	Example
<b>manager_id</b>	<b>Required</b>	Aabode ID of Rental property manager	22450

#### Request Body Parameters

No parameters.

#### Returns

It returns a list of properties belong to the rental property manager.

#### Sample

- Request:  
**GET http://beta.aabode.com/api/v2/managers/22450/rentals**
- Response

```
[
  {
    "id": 12595,
    "rentalType": {
      "name": [
        "Tree house"
      ]
    },
    "region": {
      "id": 7,
      "search": [
        "Australia - New Zealand and the South Pacific"
      ],
      "num": 1
    },
    "rentalstatus": {
      "status": "NOT_FINISHED"
    }
  }
]
```

```
},
"email": "michal.wlodarski@rentalsunited.com",
"phone": "11234565689",
"summary": [],
"address": null,
"floor": null,
"is_full_house": null,
"checkin_after": null,
"checkout_after": null,
"allow_today_booking": null,
"latitude": null,
"longitude": null,
"bookings_enabled": null,
"currency": null,
"rate_raw": null,
"source": null,
"rules": null,
"checkin_instructions": null,
"fullHouse": false,
"showLocation": false,
"bookingsEnabled": false,
"checkInAfterFormatted": "",
"checkOutAfterFormatted": "",
"todayBookingsAllowed": false
},
{...}
]
```

## 4.4 RENTAL BASIC

Table 5. Rental Basic Endpoints

REQUEST METHOD	ENDPOINTS
<b>POST</b>	/api/v2/rental
<b>PUT</b>	/api/v2/rental/<rental_id>

### 4.4.1 Create a rental basic

**POST** /api/v2/rental

Create a new property with the basic rental information.

URL parameters

No parameters

#### Request Body Parameters

Attributes	Required/Optional	Description	Example
inputType	Required	Rental Type ID of the new rental property	
inputOwnername	Optional	The Owner Name of the new rental property	
inputOwneremail	Required	The Owner Email of the new rental property	
inputWhitelabelOwneremail	Optional	The White-label Owner Email of the new rental property	
inputRentalemail	Optional	The Rental Email of the new rental property	
inputRentalphone	Optional	The Rental Phone of the new rental property	
inputSource	Optional	The Rental Source of the new rental property	
inputUrl	Optional		
inputRateRaw	Optional		

#### Returns

It returns a rental property object.

#### Sample

- Request:  
POST <http://beta.aabode.com/api/v2/rental>
- Response:

```
{
  "id": 12580,
  "rentalType": {
    "name": [
      "Tree house"
    ]
  },
  "region": {
    "id": 7,
    "search": [
      "Australia - New Zealand and the South Pacific"
    ],
    "num": 1
  },
  "rentalstatus": {
    "status": "NOT_FINISHED"
  },
}
```

```
{
  "email": "charlie@travbiz.com.au",
  "phone": "11234565689",
  "summary": [],
  "address": null,
  "floor": null,
  "is_full_house": null,
  "checkin_after": null,
  "checkout_after": null,
  "allow_today_booking": null,
  "latitude": null,
  "longitude": null,
  "bookings_enabled": null,
  "currency": null,
  "rate_raw": null,
  "source": null,
  "rules": null,
  "checkin_instructions": null,
  "fullHouse": false,
  "showLocation": false,
  "bookingsEnabled": false,
  "checkInAfterFormatted": "",
  "checkOutAfterFormatted": "",
  "todayBookingsAllowed": false
}
```

#### 4.4.2 Update a rental basic

**PUT** /api/v2/rental/<rental\_id>

Updates the specified rental basic information of the rental property by setting the values of the parameters passed. Any parameters not provided are left unchanged.

##### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	12

##### Request Body Parameters

Attributes	Required/Optional	Description	Example
inputType	Required	Rental Type ID of the new rental property	
inputOwnername	Optional	The Owner Name of the new rental property	
inputOwneremail	Required	The Owner Email of the new rental	





		property	
inputWhitelabelOwneremail	Optional	The White-label Owner Email of the new rental property	
inputRentalemail	Optional	The Rental Email of the new rental property	
inputRentalphone	Optional	The Rental Phone of the new rental property	
inputSource	Required	The Rental Source of the new rental property	
inputUrl	Optional		
inputRateRaw	Optional	The Rental Rate Raw of the new rental property	

#### Returns

The updated rental basic information of the property is returned upon success. Otherwise, this call returns an error.

#### Sample

- Request:  
`PUT http://beta.aabode.com/api/v2/rental:id`
- Response

#### 4.4.3 Retrieve a rental basic

Get rental basic information of specified rental.

#### Endpoint & Method

[GET](#) /api/v2/rentals/{rental\_id}/basic

#### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	12592

#### Request Body parameters

No parameters

#### Returns

It returns a Json Data of basic information for specified rental.

#### Sample

- Request:  
`GET http://beta.aabode.com/api/v2/rentals/12592/basic`
- Response:

```
{
```

```
{
  "rental_type": 15,
  "owner_name": "Rentals United",
  "email": "michal.wlodarski@rentalsunited.com",
  "phone": "11234565689"
}
```

## 4.5 ROOMS

### 4.5.1 Room Details

Allow user to create and update room details for the specified rental property. The request method and endpoint for creating and updating room details of each rental is below.

Table 6. Rental Room Details Endpoints

REQUEST METHOD	ENDPOINTS
POST	/api/v2/rental/<rental_id>/rooms

The definition object of each room details information is below.

Table 7. Room detail object

Attributes	Required/Optional	Data Type	Description
inputPeople	Optional	Integer	
inputRooms	Optional	Integer	The number of rooms that are available in the property.
inputIsFullHouse	Required	Boolean	How to rent the property. Options are "Rent Rooms Separately" and "Rent Whole Property".
inputRoomBeds	Optional	List<String>	The list of number of beds on each room
inputRoomNames	Optional	List<String>	Room names list
inputTypeBeds	Optional	List<List<String>>	
inputRoomBedsCapacity	Optional	List<List<String>>	

#### 4.5.1.1 Update a Rental Room Detail

POST /api/v2/rentals/<rental\_id>/rooms

Creates or updates **rental room detailed** information of the specified rental property by setting the values of the parameters passed. Any parameters not provided are left unchanged.

#### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	12

#### Request Body Parameters

- Basic rental room creation request body params

Attributes	Required/Optional	Description	Example	
			Key	value
<b>inputIsFullHouse</b>	<b>Required</b>	The method to rent property. If the inputIsFullHouse is true, the method is to rent the whole property. If not, the method is to rent the rooms separately.	inputIsFullHouse	true
<b>inputRooms</b>	Optional	The number of rooms that are available in the property.		
<b>inputRoomNames</b>	Optional		inputRoomNames[1]	Room1
			inputRoomNames[2]	Room2
<b>inputRoomBeds</b>	Optional		inputRoomBeds[1]	1
			inputRoomBeds[2]	2
<b>inputTypeBeds</b>	Optional	The room bed object id. It is unique according to the type of the bed.	inputTypeBeds[1][1]	1
			inputTypeBeds[2][1]	2
			inputTypeBeds[2][2]	3

### Returns

It returns a rental room details list of the specified rental property.

### Sample

- Request Method and Endpoint
  - POST <http://beta.aabode.com/api/v2/rental/12585/rooms>
- Request Body params and response example
  - Basic rental creation request

Request Body Params Example data	
Key	Value
inputIsFullHouse	true
inputRoomNames[1]	Room1
inputRoomNames[2]	Room2
inputRoomBeds[1]	1
inputRoomBeds[2]	2
inputTypeBeds[1][1]	1
inputTypeBeds[2][1]	2
inputTypeBeds[2][2]	3

## Response

```
{
  "is_fullhouse": true,
  "rooms": [
    {
      "rental_id": 12585,
      "count": 1,
      "name": "Room Test"
    },
    {
      "rental_id": 12585,
      "count": 2,
      "name": "Room 2"
    }
  ],
  "room_beds": [
    {
      "room_count": "17663",
      "beds": [
        {
          "bed_count": 1,
          "bed_type": 1,
          "bed_capacity": 1
        }
      ]
    },
    {
      "room_count": "17664",
      "beds": [
        {
          "bed_count": 1,
          "bed_type": 2,
          "bed_capacity": 2
        },
        {
          "bed_count": 2,
          "bed_type": 3,
          "bed_capacity": 2
        }
      ]
    }
  ]
}
```

#### 4.5.1.2 Retrieve a rental rooms information

It retrieves the room detailed information of the specified rental.

#### Endpoint & Method

[GET](#) /api/v2/rentals/{rental\_id}/rooms

#### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	

#### Request Body parameters

No parameters.

#### Returns

It returns a Json Data of room(s) information of the specified rental.

#### Sample

- Request:  
`GET http://beta.aabode.com/api/v2/rentals/12571/rooms`
- Response:

```
{
  "is_fullhouse": false,
  "rooms_info": [
    {
      "count": 1,
      "name": "Room 1"
    }
  ],
  "room_beds": [
    {
      "room_count": "17646",
      "beds": [
        {
          "bed_count": 1,
          "bed_type": 1,
          "bed_capacity": 1
        }
      ]
    }
  ]
}
```

#### 4.5.2 Beds

The types and capacity of the beds that is currently registered on aabode.com are below.

Table 8. Bed Object

Bed Type	Bed Type ID	Capacity of sleeps
Single Bed	1	1
Double Bed	2	2
Queen Bed	3	2
King Bed	4	2
King Single Bed	5	1
Super King Bed	6	2
Set of Bunks	7	Null
Sofa Bed	8	Null
Baby`s Cot	9	1

#### 4.5.2.1 Retrieve beds list

Retrieve a beds list that is registered on aabode.com.

##### Endpoint & Method

**GET** /api/v2/beds

##### URL parameters

No parameters.

##### Request Body parameters

No parameters.

##### Returns

Returns a Json Array list of beds information of aabode.com.

##### Sample

- Request:  
`GET http://beta.aabode.com/api/v2/beds`
- Response:

```
[
  {
    "id": 1,
    "publicName": "Single Bed",
    "editName": "Single Bed",
    "defaultCapacity": 1
  },
  {
    "id": 2,
    "publicName": "Double Bed",
    "editName": "Double Bed",
    "defaultCapacity": 2
  },
]
```

```
{
  "id": 3,
  "publicName": "Queen Bed",
  "editName": "Queen Bed",
  "defaultCapacity": 2
},
{
  "id": 4,
  "publicName": "King Bed",
  "editName": "King Bed",
  "defaultCapacity": 2
},
{
  "id": 5,
  "publicName": "King Single Bed",
  "editName": "King Single Bed",
  "defaultCapacity": 1
},
{
  "id": 6,
  "publicName": "Super King Bed",
  "editName": "Super King Bed",
  "defaultCapacity": 2
},
{
  "id": 7,
  "publicName": "Set of Bunks",
  "editName": "Set of Bunks"
},
{
  "id": 8,
  "publicName": "Sofa Bed",
  "editName": "Sofa Bed"
},
{
  "id": 9,
  "publicName": "Baby's Cot",
  "editName": "Baby's Cot",
  "defaultCapacity": 1
}
]
```

#### 4.5.2.2 Retrieve a bed

Retrieve a bed information by specific bed type id.

### Endpoint & Method

**GET** /api/v2/beds/{bed\_id}

### URL parameters

Parameters	Required/Optional	Description	Example
<b>bed_id</b>	<b>Required</b>	Bed Type ID	2

### Request Body parameters

No parameters.

### Returns

Returns a Json data of a bed information of the specified bed type.

### Sample

- Request:  
`GET http://beta.aabode.com/api/v2/beds/2`
- Response:

```
{
  "id": 2,
  "publicName": "Double Bed",
  "editName": "Double Bed",
  "defaultCapacity": 2
}
```

## 4.6 FEATURES

Allow users to create and update features of the specified rental property. The request method and endpoint for creating and updating features of the rental property is below.

Table 9. Rental Features Endpoints

REQUEST METHOD	ENDPOINTS
<b>POST</b>	/api/v2/rental/<rental_id>/features

The definition object of the rental property feature information is below.

Table 10. Rental Feature Object

Attributes	Data Type	Description
<b>inputFeature</b>	List<Integer>	
<b>inputFeaturePrice</b>	List<Integer>	
<b>inputFeatureIsMain</b>	List<Integer>	
<b>inputFeatureIsMainSwitchOn</b>	List<Integer>	
<b>inputFeatureDistance</b>	List<Integer>	
<b>inputFeatureValue</b>	List<Integer>	
<b>inputFeatureRoom</b>	List<List<Integer>>	



#### 4.6.1 Update rental features

Creates or updates rental features of the new or existing specified property by setting the values of the parameters passed. Any parameters not provided are left unchanged.

##### Endpoint & method

**PUT** /api/v2/rentals/{rental\_id}/features

##### URL parameters

Parameters	Required/Optional	Description	Example
<b>id</b>	<b>Required</b>	Rental Property ID	12

##### Request Body Parameters

- Save feature with Property facilities.

Attributes	Required/Optional	Description	Example	
			Key	value
<b>inputFeature</b>	<b>Required</b>	The method to rent property. If the inputIsFullHouse is true, the method is to rent the whole property. If not, the method is to rent the rooms separately.	inputFeature[3]	1

- Saving main feature

Attributes	Required/Optional	Description	Example	
			Key	value
<b>inputFeature</b>	<b>Required</b>	The method to rent property. If the inputIsFullHouse is true, the method is to rent the whole property. If not, the method is to rent the rooms separately.	inputFeature[3]	1
<b>inputFeaturesMain</b>	Optional		inputFeaturesMain[4]	1

- Saving main bounds

Attributes	Required/Optional	Description	Example	
			Key	value
<b>inputFeature</b>	<b>Required</b>	The method to rent property. If the inputIsFullHouse is true, the method is to rent the whole property. If not, the method is to rent the rooms separately.	inputFeature[15]	1
<b>inputFeatures</b>	Optional		inputFeaturesMain[15]	

<b>Main</b>				
<b>inputFeatureDistance</b>	Optional		inputFeatureDistance[15]	

- Saving distance feature

Attributes	Required/Optional	Description	Example	
			Key	value
<b>inputFeature</b>	<b>Required</b>	The method to rent property. If the inputIsFullHouse is true, the method is to rent the whole property. If not, the method is to rent the rooms separately.	inputFeature[15]	1
<b>inputFeatureDistance</b>	Optional		inputFeatureDistance[15]	

- Saving price feature

Attributes	Required/Optional	Description	Example	
			Key	value
<b>inputFeature</b>	<b>Required</b>	The method to rent property. If the inputIsFullHouse is true, the method is to rent the whole property. If not, the method is to rent the rooms separately.	inputFeature[26]	1
<b>inputFeaturePrice</b>	Optional		inputFeaturePrice[26]	

- Saving number feature

Attributes	Required/Optional	Description	Example	
			Key	value
<b>inputFeature</b>	<b>Required</b>	The method to rent property. If the inputIsFullHouse is true, the method is to rent the whole property. If not, the method is to rent the rooms separately.	inputFeature[23]	2

- Saving value feature

Attributes	Required/Optional	Description	Example	
			Key	value
<b>inputFeature</b>	<b>Required</b>	The method to rent property. If the inputIsFullHouse is true,	inputFeature[25]	1



		the method is to rent the whole property. If not, the method is to rent the rooms separately.		
inputFeatureValue	Required		inputFeatureValue[25]	123

- Saving room feature

Attributes	Required /Optional	Description	Example	
			Key	value
inputFeature	Required	The method to rent property. If the inputIsFullHouse is true, the method is to rent the whole property. If not, the method is to rent the rooms separately.	inputFeature[25]	1
inputFeatureRoom	Required		inputFeatureRoom[1][14]	1

### Returns

It returns a rental features list.

### Sample

- Request Method and Endpoint
  - `POST http://beta.aabode.com/api/v2/rental/12585/features`
- Request Body params and response example:
  - Saving feature with property facilities

Request Body Params Example data	
Key	Value
inputFeature[3]	1
Response	
<pre>{   "rental_features": [     {       "feature": {         "feature_id": 3,         "feature_name": "Cot",         "feature_text": ""       }     }   ] }</pre>	



```
"price_processed_online": false,
"feature_type": {
  "feature_type_id": 1,
  "feature_type_name": "Property Facilities",
  "feature_type_text": "What facilities are available for your tenants to use?"
},
"feature_price": {
  "feature_price_id": 1,
  "feature_price_name": "No extra charges"
}
},
"distance": -1,
"number": 1,
"price": -1,
"value": -1,
"is_main": false
}
],
"room_features": [
  {
    "room_id": "17663",
    "room_features": []
  },
  {
    "room_id": "17664",
    "room_features": []
  }
]
}
```

- Saving main feature

**Request Body Params Example data**

Key	Value
inputFeature[4]	1
inputFeatureIsMain[4]	1
Response	
<pre>{   "rental_features": [     {       "feature": {         "feature_id": 4,         "feature_name": "Air conditioning",         "feature_text": "",         "price_processed_online": false,         "feature_type": {           "feature_type_id": 1,           "feature_type_name": "Property Facilities",           "feature_type_text": "What facilities are available for your tenants to use?"         },         "feature_price": {           "feature_price_id": 1,           "feature_price_name": "No extra charges"         }       },       "distance": -1,       "number": 1,       "price": -1,       "value": -1,       "is_main": true     }   ],   "room_features": [     {       "room_id": "17663",       "room_features": []     }   ] }</pre>	

```

    },
    {
      "room_id": "17664",
      "room_features": []
    }
  ]
}

```

- Saving main bounds

Request Body Params Example data	
Key	Value
inputFeature[15]	1
inputFeatureIsMain[15]	1
inputFeatureDistance[15]	500
Response	
<pre> {   "rental_features": [     {       "feature": {         "feature_id": 15,         "feature_name": "Pet friendly",         "feature_text": "",         "price_processed_online": false,         "feature_type": {           "feature_type_id": 1,           "feature_type_name": "Property Facilities",           "feature_type_text": "What facilities are available for your tenants to use?"         },         "feature_price": {           "feature_price_id": 1,           "feature_price_name": "No extra charges"         }       },       "distance": 500, </pre>	

```

    "number": 1,
    "price": -1,
    "value": -1,
    "is_main": true
  }
],
"room_features": [
  {
    "room_id": "17663",
    "room_features": []
  },
  {
    "room_id": "17664",
    "room_features": []
  }
]
}

```

- Saving distance feature

Request Body Params Example data	
Key	Value
inputFeature[15]	1
inputFeatureDistance[15]	500
Response	
<pre> {   "rental_features": [     {       "feature": {         "feature_id": 15,         "feature_name": "Pet friendly",         "feature_text": "",         "price_processed_online": false,         "feature_type": { </pre>	

```

    "feature_type_id": 1,
    "feature_type_name": "Property Facilities",
    "feature_type_text": "What facilities are available for your tenants to use?"
  },
  "feature_price": {
    "feature_price_id": 1,
    "feature_price_name": "No extra charges"
  }
},
"distance": 500,
"number": 1,
"price": -1,
"value": -1,
"is_main": false
}
],
"room_features": [
  {
    "room_id": "17663",
    "room_features": []
  },
  {
    "room_id": "17664",
    "room_features": []
  }
]
}

```

- Saving price feature

Request Body Params Example data	
Key	Value
inputFeature[26]	1





inputFeaturePrice[26]	123
Response	
<pre>{   "rental_features": [     {       "feature": {         "feature_id": 26,         "feature_name": "Cooking pots and pans",         "feature_text": "",         "price_processed_online": false,         "feature_type": {           "feature_type_id": 3,           "feature_type_name": "Kitchen Facilities",           "feature_type_text": "What equipment do you have available in the kitchen?"         },         "feature_price": {           "feature_price_id": 1,           "feature_price_name": "No extra charges"         }       },       "distance": -1,       "number": 1,       "price": 123,       "value": -1,       "is_main": false     }   ],   "room_features": [     {       "room_id": "17663",       "room_features": []     }   ] }</pre>	

```

    },
    {
      "room_id": "17664",
      "room_features": []
    }
  ]
}

```

- Saving number feature

Request Body Params Example data	
Key	Value
inputFeature[23]	2
Response	
<pre> [   {     "feature": {       "id": 23,       "featureType": {         "name": [           "Stunning Features"         ]       },       "featurePrice": {         "name": [           "No extra charges"         ]       },       "name": [         "Outdoor eating area"       ],       "priceProcessedOnline": false     },     "number": 2, </pre>	

```

    "distance": -1,
    "value": -1,
    "price": -1,
    "is_main": null
  }
]

```

- Saving value feature

Request Body Params Example data	
Key	Value
inputFeature[25]	1
inputFeatureValue[25]	123
Response	
<pre> {   "rental_features": [     {       "feature": {         "feature_id": 25,         "feature_name": "Kitchen crockery",         "feature_text": "",         "price_processed_online": false,         "feature_type": {           "feature_type_id": 3,           "feature_type_name": "Kitchen Facilities",           "feature_type_text": "What equipment do you have available in the kitchen?"         },         "feature_price": {           "feature_price_id": 1,           "feature_price_name": "No extra charges"         }       },       "distance": -1, </pre>	



```
"number": 1,
"price": -1,
"value": 123,
"is_main": false
}
],
"room_features": [
{
"room_id": "17663",
"room_features": []
},
{
"room_id": "17664",
"room_features": []
}
]
}
```

- Saving room feature

Request Body Params Example data	
Key	Value
inputFeature[25]	1
inputFeatureRoom[1][14]	1
Response	
{	
"rental_features": [	
{	
"feature": {	
"feature_id": 25,	
"feature_name": "Kitchen crockery",	
"feature_text": "",	
"price_processed_online": false,	
"feature_type": {	



```
"feature_type_id": 3,
"feature_type_name": "Kitchen Facilities",
"feature_type_text": "What equipment do you have available in the kitchen?"
},
"feature_price": {
  "feature_price_id": 1,
  "feature_price_name": "No extra charges"
},
"distance": -1,
"number": 1,
"price": -1,
"value": -1,
"is_main": false
},
"room_features": [
  {
    "room_id": "17663",
    "room_features": [
      {
        "feature_id": 14,
        "feature": {
          "feature_name": "BBQ facilities",
          "feature_text": "",
          "feature_type": {
            "feature_type_id": 1,
            "feature_type_name": "Property Facilities",
            "feature_type_text": "What facilities are available for your tenants to use?"
          }
        }
      }
    ]
  }
]
```

```
    },
    "feature_price": {
      "feature_price_id": 1,
      "feature_price_name": "No extra charges"
    },
    "price_processed_online": false
  },
  "distance": -1,
  "number": 1,
  "value": -1,
  "is_main": false
}
]
},
{
  "room_id": "17664",
  "room_features": []
}
]
}
```

#### 4.6.2 Retrieve rental features

Retrieve rental features information of specified rental property.

##### Endpoint & method

[GET](#) /api/v2/rentals/{rental\_id}/features

##### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	12571

##### Request Body parameters

No parameters.

##### Returns

It returns Json Data of rental features of the specified rental property by rental ID.

### Sample

- Request:  
GET <http://beta.aabode.com/api/v2/rentals/12571/features>
- Response:

```
{
  "rental_features": [
    {
      "feature": {
        "feature_id": -2,
        "feature_name": "Cleaning Fee",
        "feature_text": "",
        "price_processed_online": true,
        "feature_type": {
          "feature_type_id": -1,
          "feature_type_name": "Pricing",
          "feature_type_text": ""
        },
        "feature_price": {
          "feature_price_id": 3,
          "feature_price_name": "per stay"
        }
      },
      "distance": -1,
      "number": 1,
      "price": 50,
      "value": -1,
      "is_main": false
    },
    {
      "feature": {
        "feature_id": -1,
        "feature_name": "Bond",
        "feature_text": "",
        "price_processed_online": true,
        "feature_type": {
          "feature_type_id": -1,
          "feature_type_name": "Pricing",
          "feature_type_text": ""
        },
        "feature_price": {
          "feature_price_id": 3,
          "feature_price_name": "per stay"
        }
      }
    }
  ]
}
```

```

    "distance": -1,
    "number": 1,
    "price": 300,
    "value": -1,
    "is_main": false
  },
  {
    "feature": {
      "feature_id": 91,
      "feature_name": "No smoking ",
      "feature_text": "",
      "price_processed_online": false,
      "feature_type": {
        "feature_type_id": 1,
        "feature_type_name": "Property Facilities",
        "feature_type_text": "What facilities are available for your tenants to use?"
      },
      "feature_price": {
        "feature_price_id": 1,
        "feature_price_name": "No extra charges"
      }
    },
    "distance": -1,
    "number": 1,
    "price": -1,
    "value": -1,
    "is_main": false
  }
],
"rooms_features": [
  {
    "room_id": "17646",
    "room_features": []
  }
]
}

```

## 4.7 DESCRIPTION

Allow user to create and update descriptions for the specified rental property. The request method and endpoint for creating and updating descriptions of the rental property is below.

Table 11. Rental Description Endpoints

REQUEST METHOD	ENDPOINTS
----------------	-----------



<b>POST</b>	<code>/api/v2/rental/&lt;rental_id&gt;/descriptions</code>
-------------	--

Table 12. Rental Description Text Object

Attributes	Data Type	Description
<b>langPosition</b>	Integer	The language ID of description text.
<b>inputSummary</b>	String	Description summary of the rental property.
<b>inputTexts</b>	List<String>	Description text of the rental property.

#### 4.7.1 Update rental description

Creates or updates rental descriptions of the new or existing specified property by setting the values of the parameters passed. Any parameters not provided are left unchanged.

##### Endpoint & method

**POST** `/api/v2/rental/<rental_id>/descriptions`

##### URL parameters

Parameters	Required/Optional	Description	Example
<b>Id</b>	<b>Required</b>	Rental Property ID	12

##### Request Body Parameters

- Save summary and descriptions

Attributes	Required/Optional	Description	Example	
			Key	value
<b>langPosition</b>	<b>Required</b>	The method to rent property. If the inputIsFullHouse is true, the method is to rent the whole property. If not, the method is to rent the rooms separately.	langPosition	0
<b>inputSummary</b>	<b>Required</b>	Description Summary of the rental property.	inputSummary	Summary
<b>inputTexts</b>	<b>Required</b>	Description text of the rental property.	inputTexts[1]	Accommodation
			inputTexts[2]	What to see

##### Returns

Returns rental description text objects list.

##### Sample

- Request Method and Endpoint
  - POST** `http://beta.aabode.com/api/v2/rental/12585/descriptions`
- Request Body params and response example:
  - Save summary and descriptions

Request Body Params Example data	
Key	Value
langPosition	0
inputSummary	Summary
inputTexts[1]	Accommodation
inputTexts[2]	What to see
Response	
<pre>[   {     "rental_id": 12585,     "text_type": {       "text_type_id": 1,       "text_type_name": "Accommodation",       "text_type_type": "Describe your property. E.g.: What are the main features of your ur property? Why will a guest enjoy their stay? What's great about your property?"     },     "text": "Accommodation"   },   {     "rental_id": 12585,     "text_type": {       "text_type_id": 2,       "text_type_name": "Nearby attractions",       "text_type_type": "What is there to see and do close to your property?"     },     "text": "What to see"   } ]</pre>	

#### 4.7.2 Retrieve rental descriptions

Retrieve rental descriptions of the specified rental property.

Endpoint & method

**GET** /api/v2/rentals/{rental\_id}/descriptions

#### URL parameters

Parameters	Required/Optional	Description	Example
<b>rental_id</b>	<b>Required</b>	Rental Property ID	12571

#### Request Body parameters

No parameters.

#### Returns

Return descriptions of the specified rental property.

#### Sample

- Request:  
`GET http://beta.aabode.com/api/v2/rentals/12571/descriptions`
- Response:

```
{
  "accommodation_name": "Coogee Melody Street 16",
  "descriptions": [
    {
      "rental_id": 10310,
      "text_type": {
        "text_type_id": 1,
        "text_type_name": "Accommodation",
        "text_type_text": "Describe your property. E.g.: What are the main features of your property? Why will a guest enjoy their stay? What's great about your property?"
      },
      "text": "Just a 10-minute walk from the Coogee beachfront this four bedroom home offers a relaxed sanctuary for your beach break complete with free wifi, laundry facilities, linen and towels.\r\n\r\nIf you're travelling with kids there's table tennis that's sure to excite plus a safe backyard that'll keep four-legged friends tail waggingly happy!\r\n\r\nStart the day with coffee along Coogee Bay Road at your pick of caf (we recommend Tropicana or Courtyard Caf) before an invigorating coastal walk or dip in the sparkling ocean.\r\n\r\nEnjoy a sundowner at the Coogee Pavilions popular rooftop bar and come evening cook up a BBQ and dine alfresco with your co-travellers on your private deck.\r\n\r\nVenture into the city or shopping paradise Westfield Bondi via buses on Carrington Road just around the corner.\r\n\r\n***Rates quoted above may not be valid for \"peak\" or other \"special events\" periods. Please make a booking request via Aabode.com to receive a quote*** "
    },
    {
      "rental_id": 10310,
      "text_type": {
        "text_type_id": 2,
```

```

    "text_type_name": "Nearby attractions",
    "text_type_text": "What is there to see an do close to your property?"
  },
  "text": ""
}
]
}

```

## 4.8 PICTURES

Allow user to update pictures information for the specified rental property. The request method and endpoint for creating and updating pictures information of the rental property is below.

Table 13. Rental pictures information Endpoints

REQUEST METHOD	ENDPOINTS
POST	/api/v2/rental/<rental_id>/pictures

Table 14. Rental picture information Object

Attributes	Data Type	Description
<b>langPosition</b>	Integer	The language ID of description text.
<b>inputPictureType</b>	List<String>	The picture type list of the specified rental property
<b>inputPublished</b>	List<Integer>	The list of picture IDs that are published from the specified rental property.
<b>inputMain</b>	Integer	The ID of featured picture image of the specified rental property. You can select your favorite image to be used as feature image. They will be seen by guests when they search for a property.
<b>inputDeleted</b>	List<Integer>	The list of picture IDs that are deleted from the specified rental property.

### 4.8.1 Update Rental Pictures

**POST** /api/v2/rental/<rental\_id>/pictures

Update existing pictures information of the specified rental property.

#### URL parameters

Parameters	Required/Optional	Description	Example
<b>rental_id</b>	Required	Rental Property ID	12

#### Request Body parameters

- Update feature image

Attributes	Required /Optional	Description	Example	
			Key	value

<b>langPosition</b>	<b>Required</b>	The language ID of the picture information	langPosition	0
<b>inputMain</b>	<b>Required</b>	The ID of featured picture image of the specified rental property.	inputMain	12593

- 
- Update deleted image information
- Update picture category from rental to room
- Update picture category from room to rental
- Update picture category from room1 to room2
- Update picture category from feature to room feature
- Update picture category from room feature to feature

### Returns

It returns a list of the pictures for the specified rental property.

### Sample

- Request Method and Endpoint
- **POST <http://beta.aabode.com/api/v2/rental/12593/pictures>**
- Request Body params and response example:
  - Update feature image information

Request Body Params Example data	
Key	Value
langPosition	0
inputSummary	Summary
inputTexts[1]	Accommodation
inputTexts[2]	What to see
Response	
<pre>[   {     "rental_id": 12585,     "text_type": {       "text_type_id": 1,       "text_type_name": "Accommodation",       "text_type_type": "Describe your property. E.g.: What are the main features of your       property? Why will a guest enjoy their stay? What's great about your property?"     }   } ]</pre>	



```
    },
    "text": "Accommodation"
  },
  {
    "rental_id": 12585,
    "text_type": {
      "text_type_id": 2,
      "text_type_name": "Nearby attractions",
      "text_type_type": "What is there to see and do close to your property?"
    },
    "text": "What to see"
  }
]
```

- Update published image information

Request Body Params Example data	
Key	Value
langPosition	0
inputPictureType[0]	
inputPublished[0]	
Response	
<pre>[   {     "rental_id": 12585,     "text_type": {       "text_type_id": 1,       "text_type_name": "Accommodation",       "text_type_type": "Describe your property. E.g.: What are the main features of your property? Why will a guest enjoy their stay? What's great about your property?"     },     "text": "Accommodation"   }, ]</pre>	

```
{
  "rental_id": 12585,
  "text_type": {
    "text_type_id": 2,
    "text_type_name": "Nearby attractions",
    "text_type_type": "What is there to see and do close to your property?"
  },
  "text": "What to see"
}
```

- Update deleted image information
- Update picture category from rental to room
- Update picture category from room to rental
- Update picture category from room1 to room2
- Update picture category from feature to room feature
- Update picture category from room feature to feature

#### 4.8.2 Set Picture URL

##### URL parameters

##### Request Body parameters

##### Returns

##### Sample

- Request
- Response

#### 4.8.3 Get Review Pictures

##### Delete Review Pictures

##### URL parameters

##### Request Body parameters

##### Returns

##### Sample

- Request
- Response

#### 4.8.4 Retrieve rental pictures

It retrieves rental pictures information of the rental property with a given rental property ID. Rental pictures information of the rental property are published, unpublished and deleted pictures of the specified rental property.

##### Endpoint & method

**GET** /api/v2/rentals/{rental\_id}/pictures

##### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	12593

##### Request Body Parameters

No parameters

##### Returns

It returns a list of pictures information of the rental property.

##### Sample

- Request Method and Endpoint:  
GET <http://beta.aabode.com/api/v2/rentals/12593/pictures>
- Request Body params and response example:

```
{
  "pictures": [
    {
      "id": 140793,
      "text": [],
      "main": false,
      "published": false,
      "failed": false,
      "success": true,
      "deleted": false,
      "pathSearch": "/images/140/140793.350.png",
      "path300": "/images/140/140793.300.png",
      "path58": "/images/140/140793.58.png",
      "pathViewSmall": "/images/140/140793.127.png",
      "pathViewLarge": "/images/140/140793.780.png",
      "pathViewRU": "/images/140/140793.1200.png",
      "pictureType": {
        "pictureCategory": "RENTAL",
        "label": "Accommodation",
        "id": "rental_12593"
      }
    }
  ]
}
```



```

    },
    {
      "id": 140794,
      "text": [],
      "main": true,
      "published": false,
      "failed": false,
      "success": true,
      "deleted": false,
      "pathSearch": "/images/140/140794.350.png",
      "path300": "/images/140/140794.300.png",
      "path58": "/images/140/140794.58.png",
      "pathViewSmall": "/images/140/140794.127.png",
      "pathViewLarge": "/images/140/140794.780.png",
      "pathViewRU": "/images/140/140794.1200.png",
      "pictureType": {
        "pictureCategory": "RENTAL",
        "label": "Accommodation",
        "id": "rental_12593"
      }
    }
  ]
}

```

## 4.9 RATES

“Rates” is the price list of the specified rental property.

Table 15. Rental Rates Endpoints

REQUEST METHOD	ENDPOINTS
<b>PUT</b>	/api/v2/rental/:id/rate/prices

Table 16. Rental Price Object

Attributes	Data Type	Description
<b>id</b>	Long	Rental Price Object ID
<b>from</b>	Date	Date where the price starts. The format follows the ISO 8601 and it must be yyyy-MM-dd. The year for the price ranges <b>from</b> and <b>to</b> must be the same year for every request.
<b>to</b>	Date	Date where the price finishes. The format follows the ISO 8601 and it must be yyyy-MM-dd. The year for the price ranges <b>from</b> and <b>to</b> must be the same year for every request.
<b>room_id</b>	Long	When adding an accommodation in aabode.com you can



		choose to rent the whole accommodation or to rent the different rooms.
<b>supplier_price_night</b>		The price amount for every night
<b>supplier_price_weekend</b>		Different prices per night for working nights (Sunday to Thursday) and for weekend nights (Friday and Saturday).
<b>currency</b>		The currency for the accommodation.
<b>min_stay</b>		The minimum stay for the given date ranges.

#### 4.9.1 Prices

##### 4.9.1.1 Add and update rental rates

Aabode.com allows a seamless process to manage rates/pricing for the rental properties. The system manages the rates/pricing on a yearly basis, and allows enough granularities to set the rates/pricing for each calendar day.

If you add new rate/price for a given date range that already contains rates/pricing, those rates/pricing will be overridden. For example:

- We add a basic nightly price from 01/01/2020 to 31/12/2020 where **supplier\_price\_night** and **supplier\_price\_weekend** is 200 AUD.
- For the month of the March the accommodation has a different price. Therefore, we execute a new request where from 01/03/2020 to 31/03/2020 where **supplier\_price\_night** and **supplier\_price\_weekend** is 300 AUD.
- The system will store three sets of prices:
  - From 01/01/2020 to 28/02/2020, nightly price 200
  - From 01/03/2020 to 31/03/2020, nightly price 300
  - From 01/04/2020 to 31/12/2020, nightly price 200

#### Endpoint & method

**POST** /api/v2/rental/{rental\_id}/rate/prices

#### URL parameters

Parameters	Required/Optional	Description	Example
<b>id</b>	<b>Required</b>	Rental Property ID	380

#### Request Body Parameters

Attributes	Required/Optional	Description	Example
<b>from</b>	<b>Required</b>	Date where the price starts.	2020-11-20
<b>to</b>	<b>Required</b>	Date where the price finishes.	2020-11-25
<b>room_id</b>	Optional	When adding an accommodation in aabode.com you can choose to rent the whole accommodation or to rent the different rooms. The room_id identifies the room within the accommodation.	3
<b>supplier_price_night</b>	<b>Required</b>	Sets the price amount for every night.	200



supplier_price_weekend	Required	It is common to set different prices per night for working nights (Sunday to Thursday) and for weekend nights (Friday and Saturday). If the date ranges from to includes weekend nights, you can specify a different price.	250
currency	Required	The currency for the accommodation. Every accommodation supports one single currency. The accepted currencies are: AUD, EUR, NZD, USD, THB and IDR.	AUD
min_stay	Optional	The minimum stay for the given date ranges. It defaults to 1.	2

### Returns

It returns a rental rate price object.

### Sample

- Request:  
PUT /api/v2/rental/:id/rate/prices
- Response:

```
{
  "price_id": 2,
  "room_id": null,
  "rental_id": 380,
  "supplier_price_night": 100,
  "supplier_price_weekend": 200,
  "currency": "AUD",
  "date_from": "2015-12-31",
  "date_to": "2016-01-30",
  "min_stay": 1
}
```

#### 4.9.1.2 Delete a Rental Rate

**DELETE** /api/v2/rental/<rental\_id>/rate/prices/<price\_id>

It allows you to delete a rate/price range by specified rental property ID.

### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	380
price_id	Required	Rental Rate/Price ID of the specified Rental property	2

### Request Body Parameters

No parameters

### Returns

Returns the price object of the specified rental property that has been removed from the system.

### Sample

- Request:  
DELETE /api/v2/rental/380/rate/prices/2
- Response:

```
{
  "price_id": 2,
  "room_id": null,
  "rental_id": 380,
  "supplier_price_night": 100,
  "supplier_price_weekend": 200,
  "currency": "AUD",
  "date_from": "2015-12-31",
  "date_to": "2016-01-30",
  "min_stay": 1
}
```

#### 4.9.1.3 Retrieve rate prices

It retrieves rental rates prices list of the rental property.

### Endpoint & method

GET /api/v2/rentals/<rental\_id>/rate/prices

### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	12547

### Request Body parameters

No parameters

### Returns

It returns the price list of the rental property.

### Sample

- Request:
  - GET http://beta.aabode.com/api/v2/rentals/12547/rate/prices

- Response:

```
{
  "prices": [
    {
      "priceId": 55955,
      "supplierPriceNight": 2900,
      "supplierPriceWeekend": 2900,
      "dateFrom": "2016-01-01",
      "dateTo": "2016-12-31",
      "minStay": 1,
      "currency": "AUD"
    }
  ],
  "missingPriceRange": [],
  "roomNoPrices": [
    {
      "roomId": 17619,
      "roomName": "Room 1",
      "count": 1
    }
  ]
}
```

## 4.9.2 Rate Details

### 4.9.2.1 Add and update rate details

**PUT** /api/v2/rental/<rental\_id>/rate/details

Allows you to add or update rate details information of the rental property by the specified given ID.

#### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	380

#### Request Body parameters

Attributes	Required/Optional	Description	Example
checkIn	Required		
checkout	Required		
samedayBookings	Required		1

#### Returns

Returns rate details object of the specified rental property by the given ID.

### Sample

- Request:
- Response:

#### 4.9.2.2 Retrieve rate details

Retrieve rental rate details information of the specified rental property by the given ID.

### Endpoint & method

[GET](#) /api/v2/rentals/{rental\_id}/rate/details

### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	380

### Request Body Parameters

No parameters

### Returns

It returns a rate details information object.

### Sample

- Request:  
`GET http://beta.aabode.com/api/v2/rentals/10090/rate/details`
- Response:

```
{
  "check_in_after": "15:00",
  "check_out_before": "11:00",
  "sameday_bookings": true
}
```

## 4.9.3 Additional Charges in Rates

### 4.9.3.1 Add and update rental additional charges

[PUT](#) /api/v2/rental/<rental\_id>/rate/additional\_charges

### URL parameters

### Request Body parameters

### Returns

### Sample

- Request

- Response

#### 4.9.3.2 Retrieve rental additional charges

Retrieve rental additional charges information of the specified rental.

##### Endpoint & method

[GET](#) /api/v2/rentals/{rental\_id}/rate/features

##### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	10090

##### Request Body parameters

No parameters

##### Returns

It return a json data of the additional charge information for the specified rental property.

##### Sample

- Request:  
`GET http://beta.aabode.com/api/v2/rentals/10090/rate/features`
- Response:

```
{
  "priceFeatures": [
    {
      "id": -1,
      "name": "Bond",
      "priceProcessedOnline": true,
      "featureType": "Pricing",
      "featurePrice": "per stay"
    },
    {
      "id": -2,
      "name": "Cleaning Fee",
      "priceProcessedOnline": true,
      "featureType": "Pricing",
      "featurePrice": "per stay"
    }
  ],
  "bondFeature": {
    "feature": "Bond",
    "price": 750,
```

```
"featureUnit": "per stay"
},
"cleaningFeeFeature": {
  "feature": "Cleaning Fee",
  "price": 225,
  "featureUnit": "per stay"
}
}
```

#### 4.9.4 Promotions in Rates

##### 4.9.4.1 Add and update rate promotions

PUT /api/v2/rental/<rental\_id>/rate/promotions

##### URL parameters

##### Request Body parameters

##### Returns

##### Sample

- Request:
- Response:

##### 4.9.4.2 Retrieve rental rate promotions

It retrieves rate promotions information of the specified rental property.

##### Endpoint & method

GET /api/v2/rentals/{rental\_id}/rate/promotions

##### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	12586

##### Request Body parameters

No parameters.

##### Returns

Return a list of the promotion information of the specified rental property.

##### Sample

- Request:  
GET http://beta.aabode.com/api/v2/rentals/12586/rate/promotions
- Response:

```
{
  "promotions": [
```



```
{
  "date_from": "2021-01-13",
  "date_to": "2021-01-20",
  "valid_until": "2021-01-14",
  "enabled": "2021-01-11",
  "discount": 5,
  "created": "2021-01-11 14:47:03.697179"
}
```

#### 4.9.5 Special Owner Rules in Rates

##### 4.9.5.1 Add and update rental rate special owner rules

PUT /api/v2/rental/<rental\_id>/rate/special\_rules

URL parameters

Request Body parameters

Returns

Sample

- Request:
- Response:

##### 4.9.5.2 Retrieve rental rate special owner rules

Retrieve rental rate special owner rules for the specified rental property.

Endpoint & method

GET /api/v2/rentals/{rental\_id}/rate/rules

URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	10090

Request Body parameters

No parameters.

Returns

Return json data of the special owner rules for the specified rental property.

Sample

- Request:

GET <http://beta.aabode.com/api/v2/rentals/10090/rate/rules>

- Response:

```
{  
  "specialOwnerRules": "****STRICT NO PARTY POLICY****\r\n\r\nGENERAL INFORMATION and  
HOUSE RULES\r\n\r\nPlease read carefully before making your booking. We may, at any time, a  
nd at our sole discretion, modify these Terms and Conditions, including our Privacy Policy, with o  
r without notice to the User. Any such modification will be effective immediately upon public po  
sting. Your continued use of our Service and this Site following any such modification constitutes  
your acceptance of these modified Terms.\r\n\r\nBookings:\r\nA booking will not be confirmed  
until we have received a deposit or a confirmation from your booking channel. All transactions  
will be done in Australian dollars or Euros.\r\n\r\nGuest numbers:\r\nSee property information  
for maximum guest numbers, a full guest list must be provided and not to exceed the maximum  
number of guests allowed.\r\n\r\nCleaning:\r\nPlease note individual propertys cleaning ch  
arges upon booking.\r\n\r\nParty Policy: Zero Tolerance\r\nNOTE: MaisonNets follows best pra  
ctice policies to ensure the properties we represent for short-  
term accommodation are safe, secure and protected and not damaged. As these properties are  
not in commercial areas Government regulators and Councils also have strict rules and requirem  
ents for holiday letting and will prosecute those who book properties for holiday/short term wh  
o dont abide by the short term holiday letting regulations. Reasons such as: noise, disturbance,  
parties and overcrowding etc... MaisonNets has fair yet strict guidelines also. We require an und  
ertaking from guests and clients who book for short term accommodation only not to have pa  
rties, events or gatherings and give an assurance that the property rented is for the use of regi  
stered guests and for the use of those people. A day/evening visit/luncheon/dinner by a few  
friends is fine if confirmed with the MaisonNets team in advance.\r\n\r\nParty/Disturbance Proc  
edures: In the event an unsolicited event/party ensues at a MaisonNets managed property, then  
the procedure is as follows:\r\n\r\nOur Concierges will phone first and if the issue is not resolve  
d attend immediately (at your cost) and attempt to resolve the concern. If a resolution is not for  
thcoming they will request immediate eviction as per this agreement. If this is met with resistan  
ce the Concierges are instructed to call Private Security (at your cost for a minimum of 4 hrs for  
2x Security Personal). Our Private Security team will secure the premises, safeguard from damag  
e and remain indefinitely or until the property is secure (this is at the Security/Caretakers discret  
ion). The Police will also be called and if Councils are alerted further fines to you may arise. By ag  
reeing to these Terms and Conditions below you agree to this procedure and all costs being as s  
et out in clause 13. As explained above, this costly, zero tolerance approach is necessary to prot  
ect our homes.\r\n\r\nWe are not unreasonable and encourage guests to talk to us if they are c  
oncerned communication can solve anything. We know that 99.9% of our guests are respectful  
and responsible people who simply wish to unwind in comfort and enjoy their holiday this is wh  
at we strive to provide. Please, if in doubt, talk to us as these measures are not designed to prev  
ent our guests from the normal, relaxing use of a holiday home only to prevent those who belie  
ve having a party is acceptable.\r\n\r\nWelcome:\r\nWe welcome you to this property where g  
uests can relax and unwind in comfort and coziness. MaisonNets strives to ensure that each and  
every guest enjoys an experience beyond their expectations; if you have any special requests ple  
ase let us know in contacting us. We ask that you please read and abide by the terms and conditi  
ons (especially our strictly enforced no party policy) We ask guests to show the utmost care and
```

responsibility as it is expected that the property and furnishings be returned in the exact same condition at the end of a stay as at the beginning. Before booking please understand it is a requirement that any damage and/or accidents, unintentional or otherwise, are a responsibility of the guest and that the guest will meet these costs, small or large. Given its definition we do accept that accidents may occur but any accident is the responsibility of the guest. Of course MaisonNets will always go above and beyond to repair/replace by the most cost effective means in order to restore the property to the same condition as it was before the guest took occupancy. On arrival our caretaker will always show you through the property and can be on hand for your exit therefore if there are any concerns guests can raise them on the spot. It is important that guests choose a property to suit their expectations whilst knowing ours, it is our expectation that our properties are solely used for quiet holiday purposes and that guests understand their responsibilities before booking. Of course if there are any questions, please ask and we will be more than happy to explain to ensure you experience your perfect getaway. We thank you and look forward to welcoming you soon.

**MAISONNETS TERMS AND CONDITIONS:**

**1. Reservations/Bookings:** The booking will be confirmed on receipt of full cleared payment. The person who has made the booking must be the same person changing or cancelling the booking and coordinating payment we accept EFT, cheque and credit cards. All transactions will be done in Australian dollars or Euros. To help cover the costs of running MaisonNets and for direct bookings only we charge guests a service fee every time a reservation is confirmed directly by us: the amount of this service fee is 10% or otherwise indicated on your invoice.

**2. Payment Method and Bond:** EFT/Cheque/Visa/Mastercard/Amex/Diners. Credit card payment will incur a 3.05% service fee. All rent monies received will be deposited into our account and held until contract termination. If a security bond is processed as a Pre-Authorization on the guests credit card it will be held by MaisonNets until expiration of the stay. Your bond will be forfeited on breach of this agreement, in addition to damages rectification, or will be kept after vacating the property until such point that the property is inspected and cleared of damage. In the event that repairs are to be made the bond will be kept indefinitely until such point as repairs can be assessed and made. The bond will be kept until the damage is resolved, at which point the money will either be in full or part refunded to the guest or in full or part paid to the owner of the property for the damage caused. Further payments may be required if there is loss or damage exceeding this amount. Agreeing to the Terms and Conditions of this agreement gives MaisonNets permission to deposit these monies into our account and, at the conclusion of this contract, direct monies to be paid to the owner of the property, less commissions, paid to MaisonNets and agents where applicable. Any bond claim will also be subject to a \$50 per hour administration charge.

**3. Guest Payment Terms & Cancellations:** The guest pays a non-refundable booking deposit equal to 25 % of the total booking amount once you accept the booking request, and pays the remaining balance 28 days prior to arrival (2 - 28 days). The guest pays 100% of the booking amount if their booking request date is within this period. If the guest cancels their booking within 28 days of arrival (2 - 28 days), MaisonNets keeps 100% of the total booking amount. If the guest cancels their booking before this, MaisonNets keeps the booking deposit amount (as per our 'Guest payment terms' settings above). Month to month tenancy can be ended at any time, for any reason, by either side. The only requirement is that the person who wants to end the arrangement must give proper

legal notice ahead of time to the other side. The notice must be in writing, and it must give \"on e full rental period\" of notification.\r\n\r\n4. House Manager:\r\nThe person signing this confi rmation will be designated as the house manager and will be nominated as the primary contact. The house manager must be 18 years of age or older, warrants that he/she is authorized to agre e to this agreement on behalf of all occupants and takes full responsibility to ensure that all occu pants (and any guests or invitees of such occupants) understand and agree to this agreement. A mobile phone number must be provided and be contactable when occupying the property 24/7 for security purposes.\r\n\r\n5. Property Usage:\r\nThe property is to be used for short-term residential accommodation (holiday letting) only. The property may not be used for any co mmercial purpose, wedding, party, gathering or any other function unless agreed upon prior to making the booking. Also see clause 13 below.\r\n\r\n6. Check-In, Check-Out & Keys:\r\nCheck-in is from 3pm and check-out is 11am. An additional \$40 will be charged to your total if a check-in happens after 8pm. Our property Guest Concierge will meet you at the property at your agree d time from 3pm onwards and hand over keys. We require your arrival time as the Concierge do es not live on site. If we cannot meet you at the property, we will organize a way to hand you ov er the keys (local business, neighbor, keybox, etc). Early check-in or late check-out will be accommodated when possible and charged at half of the daily rate on the day of che ck-in/check-out. A \$100 deduction will be charged with each Concierge callout for key related issues. Additio nal charges might apply if we need to replace the keys.\r\n\r\nPlease contact the caretaker the day before your arrival to confirm your arrival time. If you wish an early arrival it is best to conta ct the Concierge a day before your booking, early arrivals cannot be granted outside of this (as w e might have a booking before you). Same policy for late check-out, the Concierge will happily grant a late check-out unless we have an arrival on your departure day and the property needs to be cleaned. Addi tional charges might apply for early check-in and late check-out. An earlier check and late check is not available during the peak summer season.\r\n\r\n7. T elephone & Internet:\r\nVaries from property to property, please refer to property details or as k MaisonNets Guest Concierge. We cannot guarantee phone or Internet coverage at any propert y.\r\n\r\n8. Parking:\r\nRefer to property or ask MaisonNets.\r\n\r\n9. Housekeeping & Maint enance:\r\nThere is a cleaning charge for each property, this includes all general cleaning but no t extraordinary cleaning if the home is left extra-ordinarily unclean. Extra services can be arranged for an additional fee. On arrival and in the unli kely event the property requires further housekeeping or if there is a minor maintenance issue, please inform our Guest Concierge immediately. They will coordinate the appropriate person to fix the problem as soon as practically possible.\r\n\r\n10. Garbage Removal & Cleaning:\r\nThe occupants are required to leave the property in the same clean, neat and tidy condition, as it w as when they arrived. This includes, without limitation, washing up, fridge cleaned out, BBQ clean ed and rubbish to be placed in appropriate bins provided on vacating the property. If additional extra-ordinary cleaning is required the cost will be charged against your security bond. In an attempt t o keep rates low we only have provision for rubbish for the household bins, any additional rubbi sh can either be removed by the guest (our recommendation) or, if additional extra-ordinary rubbish is left behind, the Guest Concierge will organise for our contractor to collect an

d dispose, at the guests expense, any excess rubbish at a cost of \$45 per bag. If you happen to be a guest during rubbish evenings it is a guests responsibility to ensure the bins are available for council collection, any questions please ask the Guest Concierge.

**11. Smoking & Pets:**  
There is strictly no smoking inside the property and we ask also you to leave your pets at home as they are not permitted unless you book a pet friendly property.

**12. Loss, Damage or Injury:**  
MaisonNets, the owner and their respective employees and agents, take no responsibility whatsoever for loss or damage of any property, or personal injury or death, of any occupant, guest or invitee of any such occupant and they each exclude all liability to the maximum extent permitted by law. The occupants acknowledge the environment in which the property is located and must take extreme care including, without limitation, observing safety signage, fire and safety directions and procedures and exercise caution whilst inside and when venturing outside the property. Occupants must report breakages immediately (or as soon as practical) and are responsible for loss or damage of property belonging to them, or their guests, and for the damage or loss of property belonging to the property during their stay. Your security bond will be kept for up to 7 working days after your vacating the property for this purpose. We recommend travel/health insurance as security for the event of accident, sickness or cancellation caused by unforeseen circumstances such as transport problems or illness. Please be aware of any surrounding natural bush land and there may be dangerous snakes, spiders and other animals and bugs including ticks from time to time at some properties. If you are sensitive, please insure you use an insect repellent and wear covered shoes if you are bush walking.

**13. Noise, Parties & Behaviour:**  
The house is for registered occupants only (as indicated above). Parties are not permitted under any circumstances and there is to be no disturbing noise between 10pm to 8am with no excessive noise at any other time. A \$100 deduction will be charged with each Concierge callout, a Private Security callout involves 2x Private Security personal for a minimum of 4 hours each. If your behaviour or that of your guests causes damage, danger or annoyance we reserve the right to ask you to vacate the property immediately. Please note functions of any kind are not permitted and the invitation of additional guests beyond the homes capacity will enact our Event Policy. If you breach this agreement, you will forfeit your entire security bond in addition to any damages, cleaning and Concierge and security callouts. Any breach of contract will forfeit your bond and/or event surcharge will be deducted from your security bond.

Please note: Our policy here must be extremely strict due to our responsibilities to our properties. If guests damage/break chattels or invite additional guests then we have little recourse as we must protect our properties. We are not unreasonable by any means; therefore if you think your ideal weekend would meet expectations, not damage the house or cause any neighbourly grievances please talk to us first.

**14. Guest Concierge:**  
Our Guest Concierge will be of assistance for check-ins and is happy to offer advice at this time and by phone when required, however we kindly ask you to respect her/his time during the weekends and evenings unless it is an emergency. Other than as provided for in this agreement, the property is not serviced. If additional services are requested of the Concierge, please contact MaisonNets. The Guest Concierge may assist in additional services as shown on our website at an extra charge.

**15. House rules:**  
The property contains house rules (see MaisonNets folder in the house). Please read and observe these rules which are deemed to be incorporated in this agreement.

**16. Please Note:**  
A breach or indicated breach of any of the above conditions will permit MaisonNets to cancel o

```

r amend a booking, refuse access or occupation or immediately terminate a tenancy. Refunds will not be paid on terminated tenancies and penalties may apply in the event of parties/events. \r\n\r\nWe look forward to your visit and hope that you will have a memorable experience staying with us."
}

```

#### 4.9.6 Cancellation Policy in Rates

##### 4.9.6.1 Add and update rental rate cancellation policy

###### URL parameters

###### Request Body parameters

###### Returns

###### Sample

- Request:
- Response:

##### 4.9.6.2 Retrieve rental rate cancellation policy

Retrieve rental rate cancellation policies for the specified rental property.

###### Endpoint & method

**GET** /api/v2/rentals/{rental\_id}/rate/cps

###### URL parameters

Parameters	Required/Optional	Description	Example
<b>rental_id</b>	<b>Required</b>	Rental Property ID	10090

###### Request Body parameters

No parameters.

###### Returns

It return a json data of the cancellation policies list of the specified rental property.

###### Sample

- Request:  
**GET** http://beta.aabode.com/api/v2/rentals/10090/rate/cps
- Response:

```

{
  "cancellationPolicy": [
    {
      "id": 18349,

```

```
{
  "leadingDays": 5,
  "percent": 100
}
```

#### 4.9.7 Check-in instructions

##### 4.9.7.1 Retrieve check-in instructions

Retrieve check-in instructions of the specified rental property.

##### Endpoint & method

[GET](#) /api/v2/rentals/{rental\_id}/rate/instructions

##### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	10090

##### Request Body parameters

No parameters

##### Returns

It return a json data of the check-in instruction for the specified rental property.

##### Sample

- Request:
  - GET <http://beta.aabode.com/api/v2/rentals/10090/rate/instructions>
- Response:

```
{
  "checkInInstructions": "Our property Guest Concierge will meet you at the property at your agreed time from 3pm onwards and hand over keys. "
}
```

#### 4.10 CALENDAR

##### 4.10.1 Retrieve allowance of instant booking

Retrieve allowance information of instant booking enabled for the specified rental property.

##### Endpoint & method

[GET](#) /api/v2/rentals/{rental\_id}/calendar/instant

##### URL parameters

Parameters	Required/Optional	Description	Example
------------	-------------------	-------------	---------



[illegible]



[illegible]

```
[
  {
    "id": 11798,
    "rental_id": 10090,
    "room_id": -1,
    "name": "Coogee B Alexander ",
    "url": "https://www.airbnb.com/calendar/ical/23903575.ics?s=ee714ffc7b57fcee bd6e61a ec1d3a537",
    "modified_by": "pAWat3WD",
    "md5": "439c29abd69eac3c2b2eef9b29c4c5be",
    "failed": 1560061545318,
    "success": 1562567341878,
    "enabled": null,
    "created": 1555541197568,
    "deleted": null,
    "modified": 1562565724471
  }
]
```

#### 4.10.4 Retrieve ical file per room

Retrieve a ical file for the specified room.

##### Endpoint & method

[GET](#) /api/v2/rentals/room/{room\_id}/calendar/ical\_export

##### URL parameters

Parameters	Required/Optional	Description	Example
room_id	Required	room ID	9108

##### Request Body parameters

No parameters

##### Returns

Return a ical file for the specified room.

##### Sample

- Request:  
`GET http://beta.aabode.com/api/v2/rentals/room/9108/calendar/ical_export`
- Response:

```
BEGIN:VCALENDAR
PRODID:-//Aabode//Room 9108//EN
VERSION:2.0
CALSCALE:GREGORIAN
END:VCALENDAR
```

#### 4.10.5 Retrieve ical file per rental

Retrieve exporting ical file for the specified rental property.

##### Endpoint & method

[GET](#) /api/v2/rentals/{rental\_id}/calendar/ical\_export

##### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	10090

##### Request Body parameters

No parameters

##### Returns

Return ical file for the specified rental property.

##### Sample

- Request

**GET** [http://beta.aabode.com/api/v2/rentals/10090/calendar/ical\\_export](http://beta.aabode.com/api/v2/rentals/10090/calendar/ical_export)

- Response

```
BEGIN:VCALENDAR
PRODID:-//Aabode//Rental 10090//EN
VERSION:2.0
CALSCALE:GREGORIAN
END:VCALENDAR
```

## 4.11 LOCATION

### 4.11.1 Retrieve a rental location

Retrieve a rental location of the specified rental.

#### Endpoint & method

**GET** [/api/v2/rentals/{rental\\_id}/location](http://beta.aabode.com/api/v2/rentals/{rental_id}/location)

#### URL parameters

Parameters	Required/Optional	Description	Example
<b>rental_id</b>	<b>Required</b>	Rental Property ID	10090

#### Request Body parameters

No parameters

#### Returns

It return a json data of location information for the specified rental property.

#### Sample

- Request:

**GET** <http://beta.aabode.com/api/v2/rentals/10090/location>

- Response:

```
{
  "Location": "Coogee, Sydney, Australia",
  "Address": "23 Alexander St",
  "Public Location": "1",
  "Floor": " Unit 4",
  "Longitude": "151.2548167",
  "Latitude": "-33.9260438"
}
```

### 4.11.2 Add and update a rental location

#### URL parameters

#### Request Body parameters

## Returns

### Sample

- Request:
- Response:

## 4.12 PAYMENT

### 4.12.1 Add and update a rental payment

#### URL parameters

#### Request Body parameters

## Returns

### Sample

- Request:
- Response:

### 4.12.2 Retrieve a rental payment

Retrieve payment information of the rental property manager.

#### Endpoint & method

[GET](#) /api/v2/rentals/{rental\_id}/payment

#### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	10090

#### Request Body parameters

No parameters

## Returns

Return a json data of the payment information of the rental property manager.

### Sample

- Request:  
`GET http://beta.aabode.com/api/v2/rentals/10090/payment`
- Response:

```
{
  "BIC_SWIFT_code": "HKBAAU2S",
  "BSB_num": "032032",
  "accountNum": "420049"
}
```

## 4.13 BUNDLE

### 4.13.1 Retrieve a rental subscription plan

Retrieve a subscription plan information of the specified rental property.

#### Endpoint & method

**GET** /api/v2/rentals/{rental\_id}/bundle

#### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	10090

#### Request Body parameters

No parameters

#### Returns

Return a json data of the bundle information for the specified rental property.

#### Sample

- Request:  
**GET http://beta.aabode.com/api/v2/rentals/10090/bundle**
- Response:

```
{
  "name": "Free",
  "supplierPrice": 0,
  "supplierCurrency": "AUD",
  "buyPrice": 0,
  "buyCurrency": "AUD",
  "netPrice": 0,
  "netCurrency": "AUD",
  "grossPrice": 0,
  "grossCurrency": "AUD",
  "discountSupplierPrice": 0,
  "discountSupplierCurrency": "AUD",
  "discountBuyPrice": 0,
  "discountBuyCurrency": "AUD",
  "discountNetPrice": 0,
  "discountNetCurrency": "AUD",
  "discountGrossPrice": 0,
  "discountGrossCurrency": "AUD",
  "gstPrice": 0,
  "gstCurrency": "AUD"
}
```

#### 4.14 SEARCH

It allows user to be capable for search accommodation. This is not required for the Channel Managers, but it will be required for many other types of clients.

Table 17. Rental Search Endpoint

REQUEST METHOD	ENDPOINTS
GET	/api/v2/rental/search

Table 18. Rental Search Object

Attributes	Data Type	Description
from	String	
to	String	
results	JsonObject List	
exactresults	JsonObject List	
approxresults	JsonObject List	
page	Integer	
exact_region		
approx_region		
groupfeatures	JsonObject List	
grouprentaltypes	JsonObject List	
wishlists	Integer List	

##### 4.14.1 Retrieve Search Rentals

[GET /api/v2/rental/search](#)

It retrieves the information of rentals with the specified conditions by setting query parameter values.

URL parameters

No parameters

Query parameters

Parameters	Required/Optional	Data Type	Description	Example
country	Optional	String	Country tag raw data that allows you to search rentals in.	
region	Optional	String	Region tag data that allows you to search rentals in.	
from	Optional	Date	Date that is available to do check-in. The format follows the ISO 8601 and it	2020-11-20

			must be yyyy-MM-dd.	
<b>to</b>	Optional	Date	Date that is available to do check-out. The format follows the ISO 8601 and it must be yyyy-MM-dd.	2020-11-29
<b>adults</b>	Optional	Integer	The number of guests.	
<b>f</b>	Optional	Integer List	The features ID list of rentals.	
<b>t</b>	Optional	Integer List	The rental property type ID list.	
<b>pfrom</b>	Optional	Integer	First price value of Average Price per night.	
<b>pto</b>	Optional	Integer	Last price value of Average Price per night.	
<b>p</b>	Required, default=1	Integer	The page number of searching rentals.	
<b>s</b>	Required, default=50	Integer	The page size of searching rentals.	
<b>o</b>	Optional, default=relevance	String	The sorting type of searching rentals. available values: price-desc, price-asc	

#### Request Body parameters

No parameters

#### Returns

Returns search rental object with specified conditions.

#### Sample

- Request:  
GET <http://beta.aabode.com/api/v2/rental/search?from=2020-11-20&to=2020-11-29&s=2&p=1>
- Response:

```
{
  "from": "2020-11-20",
  "to": "2020-11-29",
  "results": [
    {
      "rentalid": 2516,
      "score": 1.0036697,
      "img": "/images/103/103828.350.png",
      "min_price": 238,
      "min_buy_price": 220,
      "max_buy_price": 300,
      "min_discount_price": 238,
      "min_discount_buy_price": 220,
      "summary": "Berry Creek Bush Retreat",
      "region": "Darwin, Northern Territory, Australia",

```





```
"tag_country": "australia",
"tag_region": "darwin",
"rentaltype": "HOUSE",
"isrentrooms": "PROPERTY FOR RENT",
"bookings_enabled": true,
"room_number": "1 Room",
"bathroom_number": "1 Bathroom",
"sleeps": 6,
"daydiff": 9,
"lat": "-12.7318008",
"lng": "131.09155099999998",
"features": [
  {
    "text": "20/11/20 to 29/11/20",
    "type": "SUCCESS"
  },
  {
    "text": "BBQ facilities",
    "type": "NONE"
  },
  {
    "text": "Pet friendly",
    "type": "NONE"
  },
  {
    "text": "Fireplace",
    "type": "NONE"
  },
  {
    "text": "Outdoor eating area",
    "type": "NONE"
  },
  {
    "text": "Bushwalking trails",
    "type": "NONE"
  }
],
"total_price": 1980,
"total_discounted_price": 1980,
"discount_validity": -1
},
{...}
],
"exactresults": 3219,
```



```
"approxresults": 9,
"page": 1,
"groupfeatures": [
  {
    "-1": 0
  },
  {
    "-2": 0
  },
  {
    "1": 1634
  },
  {
    "10": 84
  },
  {
    "11": 398
  },
  {...}
],
"grouprentaltypes": [
  {
    "-1": 17
  },
  {
    "1": 874
  },
  {
    "10": 10
  },
  {
    "11": 24
  },
  {...}
],
"wishlists": [
  1191,
  7614,
  7939,
  5642,
  11161
]
}
```

## 4.15 BOOKINGS

### 4.15.1 Retrieve a booking list

It retrieves a booking list information belong to the specified rental property.

#### Endpoint & method

[GET](#) /api/v2/book/{rental\_id}/bookings

#### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	12571

#### Query parameters

No parameters

#### Request Body parameters

No parameters

#### Returns

Returns booking object list by given rental property ID.

#### Sample

- Request:  
`GET http://beta.aabode.com/api/v2/book/12571/bookings`
- Response:

```
[
  {
    "id": 10206,
    "rental_id": 12571,
    "user_id": -2,
    "rental_user_id": -2,
    "basketstatus_id": 3,
    "name": "Test",
    "status_name": "Confirmed and Paid",
    "type_name": "Bundle",
    "user_name": "charlie",
    "checkin": null,
    "checkout": null,
    "created": 1602816515799,
    "checkInFormatted": null,
    "checkOutFormatted": null,
    "createdRelative": "2 months ago"
  },
]
```

```
{
  "id": 10233,
  "rental_id": 12571,
  "user_id": -2,
  "rental_user_id": -2,
  "basketstatus_id": 8,
  "name": "Test",
  "status_name": "Payment Failed",
  "type_name": "Rental",
  "user_name": "charlie",
  "checkin": 1608076800000,
  "checkout": 1608163200000,
  "created": 1606759341581,
  "checkInFormatted": "16/12/2020",
  "checkOutFormatted": "17/12/2020",
  "createdRelative": "10 hours ago"
},
{...},
]
```

#### 4.15.2 Create an Instant booking

Create an instant booking.

##### Endpoint & method

**POST** /api/v2/book/{rental\_id}/instant

##### URL parameters

Parameters	Required/Optional	Description	Example
rental_id	Required	Rental Property ID	12571

##### Request Body parameters

Parameters	Required/Optional	Description	Data Type	Example
inputPaymentType	Required	Booking payment type (support only credit card payment)	String	cc
inputIsDeposit	Optional	Deposit Payment for the booking (0-Full payment, 1-Deposit)	Integer	0
inputCC	Required	Credit Card number	String	5123456789012346
inputCVV	Required	CVV number (security code)	String	123
inputExpiration	Required	Expiration Date	String	05/21
From	Required	Booking start date	Date (YYYY-mm-dd)	2020-12-24
To	Required	Booking end date	Date (YYYY-mm-dd)	2020-12-25

<b>inputName</b>	<b>Required</b>	Booking guest name	String	RU
<b>inputEmail</b>	<b>Required</b>	Booking guest email	String	test@test.com
<b>inputPhone</b>	Optional	Booking guest phone	String	04000001
<b>inputTermsAnd Conditions</b>	<b>Required</b>	Agree Aabode terms and conditions or not (1-agree, 0-not)	String	1
<b>inputMessage</b>	Optional	Booking message	String	This is a test
<b>roomId</b>	Optional	Booking room ID If roomId is defined, it means booking a room. If not, that is booking a property	Long	

#### Return

Return a json data of the booking information for the specified rental property.

#### Sample

- Request:  
`POST http://beta.aabode.com/api/v2/book/12547/instant`
- Response:

### 4.15.3 Create Request a quote for booking

Create a quote request for booking.

#### Endpoint & method

**POST** /api/v2/book/{rental\_id}/request

#### URL parameters

Parameters	Required/Optional	Description	Example
<b>rental_id</b>	<b>Required</b>	Rental Property ID	12571

#### Request Body parameters

Parameters	Required/Optional	Description	Example
<b>from</b>	<b>Required</b>	Booking start date	
<b>to</b>	<b>Required</b>	Booking end date	
<b>inputName</b>	<b>Required</b>	Booking guest name	
<b>inputEmail</b>	<b>Required</b>	Booking guest email	
<b>inputPhone</b>	Optional	Booking guest phone	
<b>inputMessage</b>	<b>Required</b>	Agree Aabode terms and conditions or not	
<b>inputAction</b>			
<b>inputAdults</b>	Optional	A number of adults to book the property	
<b>roomId</b>	Optional	Booking room ID	

#### Return

It returns a json data of the booking information for the booking request of the specified rental property.

#### Sample

- Request:  
`POST http://beta.aabode.com/api/v2/book/12547/request`
- Response:

#### 4.15.4 Cancel booking

Cancel the booking (instant, on\_request).

#### Endpoint & method

**POST** /api/v2/book/{bookingId}/cancel

#### URL parameters

Parameters	Required/Optional	Description	Example
<b>bookingId</b>	<b>Required</b>	Booking ID	12571

#### Request body parameters

No parameters

#### Returns

Return a json data of the cancelled booking information.

#### Sample

- Request:  
`POST http://beta.aabode.com/api/v2/book/12547/cancel`
- Response: