**Boats and Slips API Design**

All response bodies are to be sent as ‘application/json’, which is to say the body is to be written in JSON format.

**IMPORTANT NOTE:**

When running the POSTMAN test runner collection, make sure that there is a 100ms delay between each API call. This is to deal with eventual consistency for the data on the server.

# Get all boats

**GET** /boats

## Response

Status: 200 OK

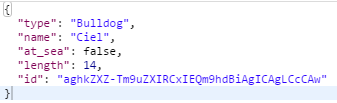


# Get a single boat

**GET** /boats/{boatId}

## Responses

Status: 200 Ok



Status: 404 Not Found

Occurs when an invalid boat ID is provided

# Creating a boat

**POST** /boats

## Body Request Data

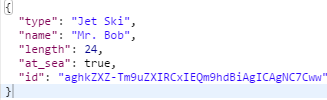
name (string) – required

length (number) – required

type (string) – required

## Responses

Status: 201 Created



Status: 400 Bad Request

Occurs when one or more of the following is true:

* One or more of the required fields is not provided
* One or more of the required fields uses the wrong data type

## Notes

All created boats start at sea and are given an automatically generated ID

# Modifying a Boat

**PATCH** /boats/{boatId}

## Body Request Data

name (string) – optional

type (string) – optional

length (number) – optional

## Responses

Status: 204 No Content

Status: 400 Bad Request

Occurs when one or more of the following is true:

* No request body is provided
* One or more of the parameters uses the wrong data type

## Notes

Other boat properties such as at\_sea can only be modified through other actions.

# Replacing a Boat

**PUT** /boats/{boatId}

## Body Request Data

name (string) – required

type (string) – required

length (number) – required

## Responses

Status: 204 No Content

Status: 400 Bad Request

Occurs when one or more of the following occurs:

* One or more of the required fields is not provided
* One or more of the parameters uses the incorrect data type

## Notes

Replacing a boat that is docked at a slip does not change the boat ID or any of the slip’s properties. This is because we assume the boat has been “upgraded”.

# Deleting a Boat

**DELETE** /boats/{boatId}

## Responses

Status: 204 No Content

Status: 400 Bad Request

Occurs when an invalid boat ID is provided

## Notes

If the deleted boat was docked at a slip, the following will happen to the previously occupied slip:

* The departure history will be updated an entry of a blank departure date and the correct boat ID
* The current\_boat property will be set to null
* The arrival\_date will be set to null

# Get all Slips

**GET** /slips

## Response

Status: 200 Ok



# Get a single slip

**GET** /slips/{slipId}

## Responses

Status: 200 Ok



Status: 404 Not Found

Occurs when the supplied slip ID is invalid

# Creating a slip

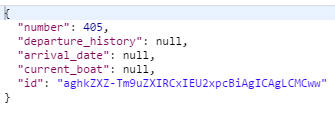
**POST** /slip

## Body Request Data

number (number) – required, must be unique

## Responses

Status: 201 Created



Status: 403 Forbidden

Occurs when a number was passed in the body request data that is already in use

Status: 400 Bad Request

Occurs when the body request data is not provided or the number is not a number data type

# Modifying a Slip

**PATCH** /slips/{slipId}

## Body Request Data

number (number) – optional

arrival\_date (string) – optional, only modified if a boat is currently docked

## Responses

Status: 204 No Content

Status: 403 Forbidden

Occurs when the number supplied is already taken

Status: 400 Bad Request

Occurs when one or more of the following is true:

* No request body is passed
* The slip ID is invalid
* One or more of the parameters uses an incorrect data type

## Notes

All other properties of the slip can only be modified through other actions.

# Replacing a Slip

**PUT** /slips/{slipId}

## Body Request Data

number (number) – required

## Responses

Status: 204 No Content

Status: 400 Bad Request

Occurs when one or more of the following is true:

* A number is not provided
* The number provided is already taken
* The number provided uses an incorrect data type

## Notes

When replacing a slip, if there was a boat docked at the slip, that boat becomes docked at the new slip. This also erases the departure history of the slip.

# Deleting a Slip

**DELETE** /slips/{slipId}

## Responses

Status: 204 No Content

Status: 400 Bad Request

Occurs when an invalid slip ID is provided

## Notes

If there was a boat previously docked at this slip, the boat will be put out to sea.

# Docking a boat at a slip

**PUT** /slip/{slipID}/boat/{boatId}

## Body Request Data

arrival\_date (string) - required

## Responses

Status: 204 No Content

Status: 403 Forbidden

Occurs if one or more of the following is true:

* The boat is already at sea
* The slip is already occupied

Status: 400 Bad Request

Occurs if one or more of the following is true:

* The arrival\_date is not provided in the request body or uses the incorrect data type
* Either the boat ID or slip ID is invalid

# Sending a boat to sea

**PUT** /boats/{boatId}/depart

## Body Request Data

departure\_date (string) – required

## Responses

Status: 204 No Content

Adds the boat ID and departure date to the slip’s departure history

Status: 403 Forbidden

Occurs if the boat is already at sea

Status: 400 Bad Request

Occurs if one or more of the following is true:

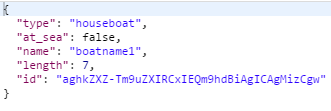
* departure\_date is not provided in the request body
* departure\_date is not a string
* Boat ID provided is not valid

# Look up docked boat information from a slip

**GET** /slip/{slipId}/boat

## Responses

Status: 200 Ok



Status: 404 Not Found

Occurs if the slip does not have a boat that is not occupied

Status: 400 Bad Request

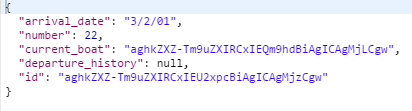
Occurs if the slip ID is invalid

# Look up which slip a boat is docked at

**GET** /boat/{boatId}/slip

## Responses

Status: 200 Ok



Status: 404 Not Found

Occurs if the boat is not docked at a slip

Status: 400 Bad Request

Occurs if the supplied boat ID is invalid