

Assignment 9: API Spec

Application URL.....	1
Account Creation URL	1
Data Model	2
List all Users	3
Create a Boat	5
Get a Boat	8
List all Boats belonging to current User	10
Edit a Boat (Update Specified Fields).....	12
Edit a Boat (Update All Editable Fields)	15
Delete a Boat.....	18
Create a Load	19
Get a Load	21
List all Loads	22
Edit a Load (Update Specified Fields).....	24
Edit a Load (Update All Editable Fields)	27
Delete a Load	30
Assign Load to Boat.....	31
Remove Load from Boat	32

Application URL

<https://a9-final-project-valdemar.uc.r.appspot.com/>

Account Creation URL

<https://a9-final-project-valdemar.uc.r.appspot.com/>

(all user-facing content except redirect to Auth0 login is displayed at the same URL)

Data Model

The app stores three kinds of entities in Datastore: Users, Boats, and Loads.

Users

Property	Data Type	Notes
id	Integer	The Datastore id of the user. Datastore automatically generates it. Only used by the database. Not editable.
user_name	String	Login email address of the user. Provided by Auth0. Not editable.
owner_id	String	Unique user identifier associated with boats created by the user. Format is "auth0 " followed by 24 characters composed of a combination of digits and lowercase letters. Provided by Auth0. Not editable.

Boats

Property	Data Type	Notes
id	Integer	The Datastore id of the boat. Datastore automatically generates it. Not editable.
owner	String	The owner_id of the User that created the boat. Assigned automatically at creation. Not editable.
name	String	Name of the boat. Must be correct date type. Allowable string length range is [1..50]. Must contain only alphanumeric characters and spaces.
type	String	Type of the boat. E.g., Sailboat, Catamaran, etc. Must be correct date type. Allowable string length range is [1..50]. Must contain only alphanumeric characters and spaces.
length	Integer	The length of the boat in feet. Must be correct date type. Allowable values are in range [5..150].
loads	List	If the boat is loaded, each element of the list is Load id. Otherwise, the list is empty. Empty at creation.

Loads

Property	Data Type	Notes
id	Integer	The Datastore id of the load. Datastore automatically generates it. Not editable.
volume	Integer	Physical volume of the load in cubic feet. Must be correct date type. Allowable values are in range [1..100].
item	String	Description of load. E.g., Furniture, Toys, etc. Must be correct date type. Allowable string length range is [1..50]. Must contain only alphanumeric characters and spaces.
creation_date	String	Date the load entry was created. Entered Manually. Must be correct date type. Allowable string length range is [6..10]. Must contain only numbers and forward slashes.
carrier	Dictionary	The Boat to which the load is assigned. Contains the Boat id and Boat name if the load is assigned to a boat. Otherwise, null. Null at creation.

List all Users

List all site users. Results are not paginated.

GET /users

Request

Path Parameters

None

Authorization Header

Required: None

Accept Header

Required: application/json OR */*

Request Body

None

Response

Response Body Format

JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Failure	406 Not Acceptable	Request Accept header does not contain application/json and is not */* (any). Boat is not created.
Success	200 OK	

(See next page for Response Examples)

Response Examples

Failure

Status: 406 Not Acceptable

```
{
  "Error": " Requested MIME type not supported; application/json only "
}
```

Success

Status: 200 OK

```
{
  "users":
  [
    {
      "id": 5631671361601536,
      "owner_id": "auth0|6484f9b0c916e6494786275a",
      "self": "http://127.0.0.1:8080/users/5631671361601536",
      "user_name": "valdemar1@cs493.com"
    },
    {
      "id": 5704568633556992,
      "owner_id": "auth0|6484f9c5ab4fa8f0bbfef665",
      "self": "http://127.0.0.1:8080/users/5704568633556992",
      "user_name": "valdemar2@cs493.com"
    }
  ]
}
```

Create a Boat

Allows an authorized user to create a new boat.

POST /boats

Request

Path Parameters

None

Authorization Header

Required: Bearer Token

Content-Type Header

Required: application/json

Accept Header

Required: application/json OR */*

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?
name	The name of the boat.	Yes
type	The type of the boat. E.g., Sailboat, Catamaran, etc.	Yes
length	Length of the boat in feet.	Yes

Request Body Example

```
{
  "name": "Sea Witch",
  "type": "Catamaran",
  "length": 28
}
```

(See next page for Response details)

Response

Response Body Format

JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Failure	401 Unauthorized	Missing, invalid, unauthorized, or expired JWT Access Token. Boat is not created.
Failure	415 Unsupported Media Type	Any request Content-Type header other than application/json. Boat is not created.
Failure	406 Not Acceptable	Request Accept header does not contain application/json and is not */* (any). Boat is not created.
Failure	400 Bad Request	<p>If the request</p> <ul style="list-style-type: none">• JSON syntax is invalid• is missing any of the 3 required attributes• includes any other attribute• includes a required attribute that fails data type, length, or content validation* <p>the boat is not created.</p> <p>* See Notes column of Boats table in Data Model section for validation details.</p>
Success	201 Created	

(See next page for Response Examples)

Response Examples

- Datastore will automatically generate an ID and store it with the entity being created. A newly created boat's `loads` attribute is an empty list. The `self` attribute is generated on-the-fly.

Failure

```
Status: 401 Unauthorized
{
  "code": "no auth header",
  "description": "Authorization header is missing"
}
```

Failure

```
Status: 415 Unsupported Media Type
{
  "Error": "Request body MIME type not supported; application/json only"
}
```

Failure

```
Status: 406 Not Acceptable
{
  "Error": "Requested MIME type not supported; application/json only"
}
```

Failure

```
Status: 400 Bad Request
{
  "Error": "The request object contains at least one prohibited attribute value;
type must contain only letters, numbers, and/or spaces "
}
```

Success

```
Status: 201 Created
{
  "id": 123,
  "owner": "auth0|6484f9b0c916e6494786275a",
  "name": "Sea Witch",
  "type": "Catamaran",
  "length": 28,
  "loads": [],
  "self": "https://a9-final-project-valdemar.uc.r.appspot.com/boats/123"
}
```

Get a Boat

Allows an authorized user to view an existing boat they created. Cannot be used to view other user's boats.

GET /boats/:boat_id

Request

Path Parameters

Name	Description
boat_id	ID of the boat

Request Body

None

Authorization Header

Required: Bearer Token

Accept Header

Required: application/json OR */*

Response

Response Body Format

JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Failure	401 Unauthorized	Missing or invalid JWT Access Token
Failure	406 Not Acceptable	Request Accept header does not contain application/json or text/html and is not */* (any).
Failure	404 Not Found	No boat with this boat_id exists.
Failure	403 Forbidden	Boat is not assigned to current user.
Success	200 OK	

(See next page for Response Examples)

Response Examples

Failure

```
Status: 401 Unauthorized
{
  "code": "no auth header",
  "description": "Authorization header is missing"
}
```

Failure

```
Status: 406 Not Acceptable
{
  "Error": "Requested MIME type not supported; application/json or text/html only"
}
```

Failure

```
Status: 404 Not Found
{
  "Error": "No boat with this boat_id exists"
}
```

Failure

```
Status: 403 Forbidden
{
  "Error": "User not authorized to view this boat"
}
```

Success

```
Status: 200 OK
{
  "id": 123,
  "owner": "auth0|6484f9b0c916e6494786275a",
  "name": "Sea Witch",
  "type": "Catamaran",
  "length": 28,
  "loads": [],
  "self": "https://a9-final-project-valdemar.uc.r.appspot.com/boats/123"
}
```

List all Boats belonging to current User

List all an authorized user's boats. Results are paginated using offsets and limits with 5 results per page. Page "next" link requires authorization.

GET /boats

Request

Path Parameters

None

Authorization Header

Required: Bearer Token

Accept Header

Required: application/json OR */*

Request Body

None

Response

Response Body Format

JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Failure	401 Unauthorized	Missing or invalid JWT Access Token
Failure	406 Not Acceptable	Request Accept header does not contain application/json and is not */* (any). Boat is not created.
Success	200 OK	

(See next page for Response Examples)

Response Examples

Failure

```
Status: 401 Unauthorized
{
  "code": "no auth header",
  "description": "Authorization header is missing"
}
```

Failure

```
Status: 406 Not Acceptable
{
  "Error": "Requested MIME type not supported; application/json only "
}
```

Success

```
Status: 200 OK
{
  "boats":
  [
    {
      "id": 123,
      "owner": "auth0|6484f9b0c916e6494786275a",
      "name": "Sea Witch",
      "type": "Catamaran",
      "length": 28
      "loads": [],
      "self": https://a9-final-project-valdemar.uc.r.appspot.com/boats/123
    },
    {
      "id": 456,
      "owner": "auth0|6484f9b0c916e6494786275a",
      "name": "Adventure",
      "type": "Sailboat",
      "length": 50,
      "loads": [],
      "self": https://a9-final-project-valdemar.uc.r.appspot.com/boats/456
    },
    {
      <boat data as above>
    },
    {
      <boat data>
    },
    {
      <boat data>
    }
  ],
  "next": "https://a9-final-project-valdemar.uc.r.appspot.com/boats?limit=5&offset=5"
}
```

Edit a Boat (Update Specified Fields)

Allows an authorized user to edit only specified fields of an existing boat they created. Cannot be used to edit other user's boats. Does not support editing loads. See Edit a Load section for that functionality.

PATCH /boats/:boat_id

Request

Path Parameters

Name	Description
boat_id	ID of the boat

Authorization Header

Required: Bearer Token

Content-Type Header

Required: application/json

Accept Header

Required: application/json OR */*

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?
name	The name of the boat.	If no other attributes in request body
type	The type of the boat. E.g., Sailboat, Catamaran, etc.	If no other attributes in request body
length	Length of the boat in feet.	If no other attributes in request body

Request Body Example

```
{  
  "name": "Sea Witch II"  
}
```

(See next page for Response details)

Response

Response Body Format

JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Failure	401 Unauthorized	Missing or invalid JWT Access Token
Failure	415 Unsupported Media Type	Any request Content-Type header other than application/json. Boat is not edited.
Failure	406 Not Acceptable	Request Accept header does not contain application/json and is not */* (any). Boat is not edited.
Failure	400 Bad Request	<p>If the request</p> <ul style="list-style-type: none">• JSON syntax is invalid• includes an id or owner attribute• does not include a name, type, or length attribute• includes any other attribute• includes a valid attribute that fails data type, length, or content validation* <p>the boat is not edited.</p> <p>* See Notes column of Boats table in Data Model section for validation details.</p>
Failure	404 Not Found	No boat with this boat_id exists.
Failure	403 Forbidden	Boat is not assigned to current user. Boat is not edited.
Success	200 OK	

(See next page for Response Examples)

Response Examples

Failure

```
Status: 401 Unauthorized
{
  "code": "no auth header",
  "description": "Authorization header is missing"
}
```

Failure

```
Status: 415 Unsupported Media Type
{
  "Error": "Request body MIME type not supported; application/json only"
}
```

Failure

```
Status: 406 Not Acceptable
{
  "Error": "Requested MIME type not supported; application/json only"
}
```

Failure

```
Status: 400 Bad Request
{
  "Error": "The request object contains a prohibited attribute; cannot edit id"
}
```

Failure

```
Status: 404 Not Found
{
  "Error": "No boat with this boat_id exists"
}
```

Failure

```
Status: 403 Forbidden
{
  "Error": "User not authorized to edit this boat"
}
```

Success

```
Status: 200 OK
{
  "id": 123,
  "name": "Sea Witch II",
  "type": "Catamaran",
  "length": 99
  "loads": [],
  "self": "https://a9-final-project-valdemar.uc.r.appspot.com/boats/123"
}
```

Edit a Boat (Update All Editable Fields)

Allows an authorized user to edit all fields of an existing boat they created. Requires all attributes.

Cannot be used to edit other user's boats. Does not support editing loads. See Edit a Load section for that functionality.

PUT /boats/:boat_id

Request

Path Parameters

Name	Description
boat_id	ID of the boat

Authorization Header

Required: Bearer Token

Content-Type Header

Required: application/json

Accept Header

Required: application/json OR */*

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?
name	The name of the boat.	Yes
type	The type of the boat. E.g., Sailboat, Catamaran, etc.	Yes
length	Length of the boat in feet.	Yes

Request Body Example

```
{
  "name": "Sea Witch III",
  "type": "Catamaran",
  "length": 99
}
```

(See next page for Response details)

Response

Response Body Format

JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Failure	401 Unauthorized	Missing or invalid JWT Access Token
Failure	415 Unsupported Media Type	Any request Content-Type header other than application/json. Boat is not edited.
Failure	406 Not Acceptable	Request Accept header does not contain application/json and is not */* (any). Boat is not edited.
Failure	400 Bad Request	<p>If the request</p> <ul style="list-style-type: none">• JSON syntax is invalid• includes an id or owner attribute• is missing any of the 3 required attributes• includes any other attribute• includes a required attribute that fails data type, length, or content validation * <p>the boat is not edited. * See Notes column of Boats table in Data Model section for validation details.</p>
Failure	404 Not Found	No boat with this boat_id exists.
Failure	403 Forbidden	Boat is not assigned to current user. Boat is not edited.
Success	200 OK	

(See next page for Response Examples)

Response Examples

Failure

```
Status: 401 Unauthorized
{
  "code": "no auth header",
  "description": "Authorization header is missing"
}
```

Failure

```
Status: 415 Unsupported Media Type
{
  "Error": "Request body MIME type not supported; application/json only"
}
```

Failure

```
Status: 406 Not Acceptable
{
  "Error": "Requested MIME type not supported; application/json only"
}
```

Failure

```
Status: 400 Bad Request
{
  "Error": "The request object contains at least one invalid attribute; color is not a boat attribute"
}
```

Failure

```
Status: 404 Not Found
{
  "Error": "No boat with this boat_id exists"
}
```

Failure

```
Status: 403 Forbidden
{
  "Error": "User not authorized to edit this boat"
}
```

Success

```
Status: 200 OK
{
  "id": 123,
  "owner": "auth0|6484f9b0c916e6494786275a",
  "name": "Sea Witch III",
  "type": "Sailboat",
  "length": 78
  "loads": [],
  "self": "https://a9-final-project-valdemar.uc.r.appspot.com/boats/123"
}
```

Delete a Boat

Allows an authorized user to delete a boat they created.

DELETE /boats/:boat_id

Request

Path Parameters

Name	Description
boat_id	ID of the boat

Authorization Header

Required: Bearer Token

Request Body

None

Response

No body

Response Body Format

Success: No body

Failure: JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Failure	404 Not Found	No boat with this boat_id exists.
Failure	403 Forbidden	Boat is not assigned to current user. Boat is not deleted.
Success	204 No Content	

Response Examples

Failure

```
Status: 401 Unauthorized
{
  "code": "no auth header",
  "description": "Authorization header is missing"
}
```

Failure

```
Status: 404 Not Found
{
  "Error": "No boat with this boat_id exists"
}
```

Failure

```
Status: 403 Forbidden
{
  "Error": "User not authorized to delete this boat"
}
```

Success

```
Status: 204 No Content
```

Create a Load

Allows any user to create a new load. No authentication or authorization required.

POST /loads

Request

Path Parameters

None

Authorization Header

Required: None

Content-Type Header

Required: application/json

Accept Header

Required: application/json OR */*

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?
volume	Physical volume of the load expressed as an integer.	Yes
item	Description of the load. E.g., Furniture, Toys, etc.	Yes
creation_date	Date of load entry creating. Manually assigned.	Yes

Request Body Example

```
{
  "volume": "12",
  "item": "Books",
  "creation_date": "5/1/2023"
}
```

(See next page for Response details)

Response

Response Body Format

JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Failure	400 Bad Request	If the request is missing any of the 3 required attributes, the load is not created, and 400 status code is returned. Attribute values are not validated.
Success	201 Created	

Response Examples

- Datastore will automatically generate an ID and store it with the entity being created. A newly created load's `carrier` attribute is null. The `self` attribute is generated on-the-fly.

Failure

```
Status: 400 Bad Request
{
  "Error": "The request object is missing at least one of the required attributes"
}
```

Success

```
Status: 201 Created
{
  "id": 987,
  "volume": 12,
  "item": "Books",
  "creation_date": "5/1/2023",
  "carrier": null,
  "self": "https://a9-final-project-valdemar.uc.r.appspot.com/loads/987"
}
```

Get a Load

Allows user to get an existing load. The `carrier self` attribute is generated on-the-fly. No authentication or authorization required.

GET /loads/:load_id

Request

Path Parameters

Name	Description
load_id	ID of the load

Request Body

None

Authorization Header

Required: None

Accept Header

Required: application/json OR */*

Response

Response Body Format

JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Failure	404 Not Found	No load with this load_id exists
Success	200 OK	

Response Examples

Failure

```
Status: 404 Not Found
{
  "Error": "No boat with this boat_id exists"
}
```

Success

```
Status: 200 OK
{
  "id": 987,
  "volume": 12,
  "item": "Books",
  "creation_date": "5/1/2023",
  "carrier":
  {
    "id": 123
    "self": https://a9-final-project-valdemar.uc.r.appspot.com/boats/123
  },
  "self": https://a9-final-project-valdemar.uc.r.appspot.com/loads/987
}
```

List all Loads

List all the loads. Results are paginated using offsets and limits with 5 results per page.

GET /loads

Request

Path Parameters

Name	Description
load_id	ID of the load

Request Body

None

(See next page for Response details)

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	

Response Examples

Success

Status: 200 OK

```
{
  "loads":
  [
    {
      "id": 987,
      "volume": 12,
      "item": "Books",
      "creation_date": "5/1/2023",
      "carrier":
      {
        "id": 123
        "self": https://a9-final-project-valdemar.uc.r.appspot.com/boats/123
      },
      "self": https://a9-final-project-valdemar.uc.r.appspot.com/loads/987
    },
    {
      "id": 986,
      "volume": 24,
      "item": "Shoes",
      "creation_date": "4/30/2023",
      "carrier":
      {
        "id": 123
        "self": https://a9-final-project-valdemar.uc.r.appspot.com/boats/123
      },
      "self": https://a9-final-project-valdemar.uc.r.appspot.com/loads/986
    },
    {
      "id": 980,
      "volume": 32,
      "item": "Basketballs",
      "creation_date": "4/28/2023",
      "carrier": null,
      "self": https://a9-final-project-valdemar.uc.r.appspot.com/loads/980
    }
  ],
  "next": "https://a9-final-project-valdemar.uc.r.appspot.com/loads?limit=3&offset=3"
}
```

Edit a Load (Update Specified Fields)

Allows a user to edit only specified fields of an existing load. Cannot be used to edit loads that are assigned to another user's boat.

PATCH /loads/:load_id

Request

Path Parameters

Name	Description
boat_id	ID of the boat

Authorization Header

Allowed: Bearer Token (required for loads assigned to boats)

Content-Type Header

Required: application/json

Accept Header

Required: application/json OR */*

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?
volume	Physical volume of the load expressed as an integer.	If no other attributes in request body
item	Description of the load. E.g., Furniture, Toys, etc.	If no other attributes in request body
creation_date	Date of load entry creating. Manually assigned.	If no other attributes in request body

Request Body Example

```
{
  "volume": 10
}
```

(See next page for Response details)

Response

Response Body Format

JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Failure	415 Unsupported Media Type	Any request Content-Type header other than application/json. Load is not edited.
Failure	406 Not Acceptable	Request Accept header does not contain application/json and is not */* (any). Load is not edited.
Failure	400 Bad Request	If the request <ul style="list-style-type: none">• JSON syntax is invalid• includes an id attribute• does not include a volume, item, or creation_date attribute• includes any other attribute the load is not edited. Attribute values are not validated.
Failure	404 Not Found	No load with this load_id exists.
Failure	403 Forbidden	Load is assigned to a boat belonging to another user. The load is not edited.
Success	200 OK	

(See next page for Response Examples)

Response Examples

Failure

Status: 415 Unsupported Media Type

```
{
  "Error": "Request body MIME type not supported; application/json only"
}
```

Failure

Status: 406 Not Acceptable

```
{
  "Error": "Requested MIME type not supported; application/json only"
}
```

Failure

Status: 400 Bad Request

```
{
  "Error": "The request object contains a prohibited attribute; cannot edit id"
}
```

Failure

Status: 404 Not Found

```
{
  "Error": "No load with this load_id exists"
}
```

Failure

Status: 403 Forbidden

```
{
  "Error": "Load is assigned to a boat belonging to another user; current user not
authorized to edit this load."
}
```

Success

Status: 200 OK

```
{
  "id": 987,
  "volume": 10,
  "item": "Books",
  "creation_date": "5/1/2023",
  "carrier":
  {
    "id": 123
    "self": https://a9-final-project-valdemar.uc.r.appspot.com/boats/123
  },
  "self": https://a9-final-project-valdemar.uc.r.appspot.com/loads/987
}
```

Edit a Load (Update All Editable Fields)

Allows a user to edit all editable fields of an existing load. Cannot be used to edit loads that are assigned to another user's boat.

PATCH /loads/:load_id

Request

Path Parameters

Name	Description
boat_id	ID of the boat

Authorization Header

Allowed: Bearer Token (required for loads assigned to boats)

Content-Type Header

Required: application/json

Accept Header

Required: application/json OR */*

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?
volume	Physical volume of the load expressed as an integer.	Yes
item	Description of the load. E.g., Furniture, Toys, etc.	Yes
creation_date	Date of load entry creating. Manually assigned.	Yes

Request Body Example

```
{
  "volume": "14",
  "item": "Basketballs",
  "creation_date": "6/1/2023"
}
```

(See next page for Response details)

Response

Response Body Format

JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Failure	415 Unsupported Media Type	Any request Content-Type header other than application/json. Load is not edited.
Failure	406 Not Acceptable	Request Accept header does not contain application/json and is not */* (any). Load is not edited.
Failure	400 Bad Request	If the request <ul style="list-style-type: none">• JSON syntax is invalid• includes an id attribute• does not include a volume, item, and creation_date attribute• includes any other attribute the load is not edited. Attribute values are not validated.
Failure	404 Not Found	No load with this load_id exists.
Failure	403 Forbidden	Load is assigned to a boat belonging to another user. The load is not edited.
Success	200 OK	

(See next page for Response Examples)

Response Examples

Failure

Status: 415 Unsupported Media Type

```
{
  "Error": "Request body MIME type not supported; application/json only"
}
```

Failure

Status: 406 Not Acceptable

```
{
  "Error": "Requested MIME type not supported; application/json only"
}
```

Failure

Status: 400 Bad Request

```
{
  "Error": "The request object contains a prohibited attribute; cannot edit id"
}
```

Failure

Status: 404 Not Found

```
{
  "Error": "No load with this load_id exists"
}
```

Failure

Status: 403 Forbidden

```
{
  "Error": "Load is assigned to a boat belonging to another user; current user not
authorized to edit this load."
}
```

Success

Status: 200 OK

```
{
  "id": 987,
  "volume": 14,
  "item": "Basketballs",
  "creation_date": "6/1/2023",
  "carrier":
  {
    "id": 123
    "self": https://a9-final-project-valdemar.uc.r.appspot.com/boats/123
  },
  "self": https://a9-final-project-valdemar.uc.r.appspot.com/loads/987
}
```

Delete a Load

Allows a user to delete a load. User may not delete a load assigned to a boat belonging to another user. Note that if the load is in a boat assigned to the current user, deleting the load removes the entry from the boat's loads list.

DELETE /loads/:load_id

Request

Path Parameters

Name	Description
load_id	ID of the load

Request Body

None

Authorization Header

Allowed: Bearer Token (required for loads assigned to boats)

Accept Header

Required: application/json OR */*

Response

No body

Response Body Format

Success: No body

Failure: JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Failure	404 Not Found	No load with this load_id exists
Failure	403 Forbidden	Load assigned to a boat belonging to another user. Load is not deleted.
Success	204 No Content	

Response Examples

Failure

```
Status: 404 Not Found
{
  "Error": "No load with this load_id exists"
}
```

Failure

```
Status: 403 Forbidden
{
  "Error": "Load assigned to a boat belonging to another user; current user not
authorized to delete load."
}
```

Success

```
Status: 204 No Content
```

Assign Load to Boat

Load is placed in a boat owned by the authorized user. This updates both the load's `carrier` attribute and the boat's `loads` list. If a load is assigned to another boat it cannot be reassigned before it is removed. See Remove Load from Boat section.

PUT /boats/:boat_id/loads/:load_id

Request

Path Parameters

Name	Description
boat_id	ID of the boat
load_id	ID of the load

Request Body

None

Authorization Header

Allowed: Bearer Token (required for loads assigned to boats)

Accept Header

Required: application/json OR */*

Response

No body

Response Body Format

Success: No body

Failure: JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Success	204 No Content	Succeeds only if a boat exists with this boat_id and a load exists with this load_id.
Failure	403 Forbidden	The load is already assigned to another boat.
Failure	404 Not Found	The specified boat and/or load does not exist

Response Examples

Failure

```
Status: 404 Not Found
{
  "Error": "The specified boat and/or load does not exist"
}
```

Failure

```
Status: 403 Forbidden
{
  "Error": "The load is already loaded on another boat"
}
```

Success

```
Status: 204 No Content
```

Remove Load from Boat

The load is removed from the boat owned by the authorized user. The load's `carrier` attribute is nulled and the load is removed from the boat's `loads` list.

```
DELETE /boats/:boat_id/loads/:load_id
```

Request

Path Parameters

Name	Description
<code>boat_id</code>	ID of the boat
<code>load_id</code>	ID of the load

Request Body

None

Authorization Header

Required: Bearer Token

Accept Header

Required: `application/json` OR `*/*`

Response

No body

Response Body Format

Success: No body

Failure: JSON

Response Statuses (priority order)

Outcome	Status Code	Notes
Failure	404 Not Found	No load with this <code>load_id</code> is in the boat with this <code>boat_id</code> .
Failure	403 Forbidden	Load assigned to a boat belonging to another user. Load is not deleted.
Success	204 No Content	

Response Examples

Failure

```
Status: 404 Not Found
{
  "Error": "No boat with this boat_id is loaded with the load with this load_id"
}
```

Failure

```
Status: 403 Forbidden
{
  "Error": "Boat belongs to another user; current user not authorized to add or remove loads."
}
```

Success

```
Status: 204 No Content
```