

Cloud Application Development Final Project

Gregory Melsby
melsbyg@oregonstate.edu
Fall 2022

Application deployed at: <https://melsbyg-cloud-final.uw.r.appspot.com>

Table of Contents

- Create a Boat
- Get List of Owned Boats
- Get Boat by Id
- Edit Boat with PUT
- Edit Boat with PATCH
- Delete Boat
- Create a Load
- Get All Loads
- Get Load by Id
- Edit Load with PUT
- Edit Load with PATCH
- Delete Load
- Put Load on Boat
- Remove Load from Boat
- Get List of Users

Data Model

The application store three kinds of entities in Datastore: Boats, Loads and Users.

There is a one-to-many relationship of Boats to Loads: one Boat can hold many Loads.

There is a one-to-many relationship of Users to Boats: one User can own many Boats.

Boats are a protected resource, and only a Boat's owner may perform CRUD operations on the Boat.

A Boat's owner is the user that created the Boat.

Boats

Property	Description	Type	Notes
name	name of the boat	string	Required. Cannot be null. Must not include the characters “<>{}[]”. Must not be an empty string or over 30 characters in length. Must not have whitespace at the beginning or end of string.
type	type of the boat	string	Required. Cannot be null. Must not include the characters “<>{}[]”. Must not be an empty string or over 30 characters in length. Must not have whitespace at the beginning or end of string.
length	length of the boat in feet	integer	Required. Must be a positive integer.
id	id of the boat	integer	Created automatically upon creation. Do not include in POST body.
owner	user id of the boat’s owner	string	Created automatically upon creation based on creating user. Do not include in POST body.

Loads

Property	Description	Type	Notes
volume	volume of the load in cubic feet	integer	Required. Cannot be null. Must be a positive integer.
item	item in the load	string	
creation_date	date the load was created	string	Required. Cannot be null. Must not include the characters “<>{}[]”. Must not be an empty string or over 30 characters in length. Must not have whitespace at the beginning or end of string.
id	id of the load	integer	Required. Cannot be null. Must be of the form “XX/XX/XXXX” where each group of characters is a number.
carrier	id of the boat the load is on	integer	Created automatically upon creation. Do not include in POST body. Set to null upon creation, can be updated with PUT on /boats/:boatId/loads/:loadId. Do not include in POST body.

Users

Property	Description	Type	Notes
id	id of the user. “sub” property of user JWT.	string	Created upon first user login.

Property	Description	Type	Notes
email	email address of the user	string	Created upon first user login.

Create A Boat

POST /boats

Allows creation of a new boat. Boats are a protected resource, so request must contain a valid JWT.

Request

Path Parameters

None

Request Headers

Header	Notes
Authorization	must be set to a Bearer token of a registered user
Accepts	must be set to application/json

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Property	Description	Type	Notes
name	name of the boat	string	Required. Cannot be null. Must not include the characters “<>{}[]”. Must not be an empty string or over 30 characters in length. Must not have whitespace at the beginning or end of string. Required. Cannot be null. Must not include the characters “<>{}[]”. Must not be an empty string or over 30 characters in length. Must not have whitespace at the beginning or end of string. Required. Must be a positive integer.
type	type of the boat	string	
length	length of the boat in feet	integer	

Request Body Example

```
{
  "name": "Sloop John Boone",
  "type": "Sloop",
  "length": 50
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	201 Created	
Failure	400 Bad Request	If the request is missing any of the 3 required attributes, the boat will not be created.
Failure	401 Unauthorized	If the "Authorization" header is not set to a valid JWT, no boat will be created.
Failure	406 Not Acceptable	If the "Accept" header does not indicate application/json will be accepted, no boat will be created.

Response Examples

Success

Status: 201 Created

Body:

```
{
  "name": "Sloop John Boone",
  "type": "Sloop",
  "length": 50,
  "owner": "auth0|6383e893a8b2c2ec60b332c9",
  "id": "5665673409724416",
  "loads": [],
  "self": "https://melsbyg-cloud-final.uw.r.appspot.com/boats/5665673409724416"
}
```

Failure

Missing Required Property

Status: 400 Bad Request

Body:

```
{
  "Error": "Request is missing one or more required properties"
}
```

Extra Property Present

Status: 400 Bad Request

Body:
{
 "Error": "One or more properties in the request are not valid"
}

Value of 'name' is not valid

Status: 400 Bad Request

Body:
{
 "Error": "name value in request is not valid"
}

Value of 'type' is not valid

Status: 400 Bad Request

Body:
{
 "Error": "type value in request is not valid"
}

Value of 'length' is not valid

Status: 400 Bad Request

Body:
{
 "Error": "length value in request is not valid"
}

Syntax error in JSON

Status: 400 Bad Request

Body:
{
 "Error": "Unexpected string in JSON at position 13"
}

Invalid or missing JWT Bearer Token in Authorization header

Status: 401 Unauthorized

Body:
{
 "Error": "Bad Credentials"
}

Accept header does not indicate application/json is acceptable

Status: 406 Not Acceptable

Body:

```
{
  "Error": "Endpoint only can respond with application/json data"
}
```

Get List of Owned Boats

GET /boats

Gets a list of user's boats. Boats are a protected resource, so request must contain a valid JWT. Only the boats that belong to the user whose JWT is in the Authorization header as a Bearer token. Response is paginated and returns at most 5 boats per page.

Request

Path Parameters

None

Request Headers

Header	Notes
Authorization	must be set to a Bearer token of a registered user
Accepts	must be set to application/json

Query Parameters

Name	Description
cursor	Optional. Corresponds to a Datastore cursor. Helps implement pagination.

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	400 Bad Request	If the 'cursor' query parameter is not recognized by Datastore, no list of boats will be returned.
Failure	401 Unauthorized	If the "Authorization" header is not set to a valid JWT, no list of boats will be returned.
Failure	406 Not Acceptable	If the "Accept" header does not indicate application/json will be accepted, no list of boats will be returned.

Response Examples

Success

The URL indicated by "next" is how the next page of results is accessed.
If "next" is not present, there are no more results.
"count" indicates the total number of boats the user owns.

Status: 200 OK

Body:

```
{
  "boats": [
    {
      "name": "Pequod",
      "type": "Whaler",
      "length": 200,
      "owner": "auth0|6383e893a8b2c2ec60b332c9",
      "id": "5069549128908800",
      "loads": [],
      "self": "https://melsbyg-cloud-final.uw.r.appspot.com/boats/5069549128908800"
    },
    {
      "name": "Great Big Boat",
      "owner": "auth0|6383e893a8b2c2ec60b332c9",
      "type": "Schooner",
      "length": 55,
```

```

        "id": "5081054809423872",
        "loads": [],
        "self": "https://melsbyg-cloud-final.uw.r.appspot.com/boats/5081054809423872"
    },
    {
        "type": "Sloop",
        "name": "Sloop John Boone",
        "length": 50,
        "owner": "auth0|6383e893a8b2c2ec60b332c9",
        "id": "5665673409724416",
        "loads": [],
        "self": "https://melsbyg-cloud-final.uw.r.appspot.com/boats/5665673409724416"
    },
    {
        "name": "Challenger",
        "length": 5,
        "owner": "auth0|6383e893a8b2c2ec60b332c9",
        "type": "Kayak",
        "id": "5712837116690432",
        "loads": [],
        "self": "https://melsbyg-cloud-final.uw.r.appspot.com/boats/5712837116690432"
    },
    {
        "type": "Starship",
        "length": 250,
        "owner": "auth0|6383e893a8b2c2ec60b332c9",
        "name": "Enterprise",
        "id": "6205318602162176",
        "loads": [],
        "self": "https://melsbyg-cloud-final.uw.r.appspot.com/boats/6205318602162176"
    }
],
"next": "https://melsbyg-cloud-final.uw.r.appspot.com/boats?cursor=CjISLGoXenV3fm1lbHNi",
"count": 6
}

```

Failure

'cursor' query parameter not recognized by Datastore

Status: 400 Bad Request

Body:

```

{
  "Error": "Cursor in request params not recognized"
}

```

Invalid or missing JWT Bearer Token in Authorization header

Status: 401 Bad Request

Body:

```
{
  "Error": "Bad Credentials"
}
```

Accept header does not indicate application/json is acceptable

Status: 406 Not Acceptable

Body:

```
{
  "Error": "Endpoint only can respond with application/json data"
}
```

Get Boat by Id

GET /boats/:boatId

Gets representation of boat with passed-in id. Boats are a protected resource, so request must contain a valid JWT. Boat representation will be retrieved only if the boat belongs to the user whose JWT is in the Authorization header as a Bearer token.

Request

Path Parameters

Name	Description
boatId	id of the boat to be retrieved

Request Headers

Header	Notes
Authorization	must be set to a Bearer token of a registered user
Accepts	must be set to application/json

Query Parameters

None

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	If the “Authorization” header is not set to a valid JWT, no boat will be returned.
Failure	403 Forbidden	If no boat exists with the passed-in boatId or if the “Authorization” header is set to a valid JWT but the JWT does not correspond to the user who owns the boat, no boat will be returned.
Failure	406 Not Acceptable	If the “Accept” header does not indicate application/json will be accepted, no boat will be returned.

Response Examples

Success

Status: 200 OK

Body:

```
{
  "length": 50,
  "type": "Sloop",
  "owner": "auth0|6383e893a8b2c2ec60b332c9",
  "name": "Sloop John Boone",
  "loads": [],
  "id": "5673082664517632",
}
```

```
    "self": "https://melsbyg-cloud-final.uw.r.appspot.com/boats/5673082664517632"
  }
```

Failure

Invalid or missing JWT Bearer Token in Authorization header

Status: 401 Bad Request

```
Body:
{
  "Error": "Bad Credentials"
}
```

Boat with passed-in id does not exist or JWT in Authorization header corresponds to a user who does not own the boat

Status: 403 Forbidden

```
{
  "Error": "Boat does not exist or is owned by someone else"
}
```

Accept header does not indicate application/json is acceptable

Status: 406 Not Acceptable

```
Body:
{
  "Error": "Endpoint only can respond with application/json data"
}
```

Edit Boat with PUT

PUT /boats/:boatId

Edits boat with passed-in id. Boats are a protected resource, so request must contain a valid JWT. Boat will be edited only if the boat belongs to the user whose JWT is in the Authorization header as a Bearer token. Requires all required attributes in request body. Cannot change owner.

Request

Path Parameters

Name	Description
boatId	id of the boat to be edited

Request Headers

Header	Notes
Authorization	must be set to a Bearer token of a registered user
Accepts	must be set to application/json

Query Parameters

None

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Property	Description	Type	Notes
name	name of the boat	string	Required. Cannot be null. Must not include the characters “<>{}[]”. Must not be an empty string or over 30 characters in length. Must not have whitespace at the beginning or end of string.

Property	Description	Type	Notes
type	type of the boat	string	Required. Cannot be null. Must not include the characters “<>{}[]”. Must not be an empty string or over 30 characters in length. Must not have whitespace at the beginning or end of string.
length	length of the boat in feet	integer	Required. Must be a positive integer.

Request Body Example

```
{
  "name": "Sloop John B",
  "type": "Beach Boy Vessel",
  "length": 55
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	400 Bad Request	If the request is missing any of the 3 required attributes, the boat will not be edited.
Failure	401 Unauthorized	If the “Authorization” header is not set to a valid JWT, the boat will not be edited.

Outcome	Status Code	Notes
Failure	403 Forbidden	If no boat exists with the passed-in boatId or if the "Authorization" header is set to a valid JWT but the JWT does not correspond to the user who owns the boat, no boat will not be edited.
Failure	406 Not Acceptable	If the "Accept" header does not indicate application/json will be accepted, no boat will be returned.

Response Examples

Success

Status: 200 OK

Body:

```
{
  "name": "Sloop John B",
  "type": "Beach Boy Vessel",
  "length": 55,
  "owner": "auth0|6383e893a8b2c2ec60b332c9",
  "id": "5673082664517632",
  "loads": [],
  "self": "https://melsbyg-cloud-final.uw.r.appspot.com/boats/5673082664517632"
}
```

Failure

Missing Required Property

Status: 400 Bad Request

Body:

```
{
  "Error": "Request is missing one or more required properties"
}
```

Extra Property Present

Status: 400 Bad Request

Body:
{
 "Error": "One or more properties in the request are not valid"
}

Value of 'name' is not valid

Status: 400 Bad Request

Body:
{
 "Error": "name value in request is not valid"
}

Value of 'type' is not valid

Status: 400 Bad Request

Body:
{
 "Error": "type value in request is not valid"
}

Value of 'length' is not valid

Status: 400 Bad Request

Body:
{
 "Error": "length value in request is not valid"
}

Syntax error in JSON

Status: 400 Bad Request

Body:
{
 "Error": "Unexpected string in JSON at position 13"
}

Invalid or missing JWT Bearer Token in Authorization header

Status: 401 Unauthorized

Body:
{
 "Error": "Bad Credentials"
}

Boat with passed-in id does not exist or JWT in Authorization header corresponds to a user who does not own the boat

Status: 403 Forbidden

```
{
  "Error": "Boat does not exist or is owned by someone else"
}
```

Accept header does not indicate application/json is acceptable

Status: 406 Not Acceptable

Body:

```
{
  "Error": "Endpoint only can respond with application/json data"
}
```

Edit Boat with PATCH

PATCH /boats/:boatId

Edits boat with passed-in id. Boats are a protected resource, so request must contain a valid JWT. Boat will be edited only if the boat belongs to the user whose JWT is in the Authorization header as a Bearer token. Not all properties are required in request body. Included properties will update on the boat and other properties will remain the same. Cannot change owner.

Request

Path Parameters

Name	Description
boatId	id of the boat to be edited

Request Headers

Header	Notes
Authorization	must be set to a Bearer token of a registered user
Accepts	must be set to application/json

Query Parameters

None

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Property	Description	Type	Notes
name	name of the boat	string	If included, cannot be null. Must not include the characters "<>{}[]". Must not be an empty string or over 30 characters in length. Must not have whitespace at the beginning or end of string.
type	type of the boat	string	If included, cannot be null. Must not include the characters "<>{}[]". Must not be an empty string or over 30 characters in length. Must not have whitespace at the beginning or end of string.
length	length of the boat in feet	integer	If included, must be a positive integer.

Request Body Example

```
{  
  "length": 60  
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	400 Bad Request	If the properties in the request are not valid, the boat will not be edited.
Failure	401 Unauthorized	If the “Authorization” header is not set to a valid JWT, the boat will not be edited.
Failure	403 Forbidden	If no boat exists with the passed-in boatId or if the “Authorization” header is set to a valid JWT but the JWT does not correspond to the user who owns the boat, no boat will not be edited.
Failure	406 Not Acceptable	If the “Accept” header does not indicate application/json will be accepted, no boat will be returned.

Response Examples

Success

Status: 200 OK

Body:

```
{
  "name": "Sloop John B",
  "owner": "auth0|6383e893a8b2c2ec60b332c9",
  "type": "Beach Boy Vessel",
  "length": 60,
  "id": "5673082664517632",
  "loads": [],
```

```
    "self": "https://melsbyg-cloud-final.uw.r.appspot.com/boats/5673082664517632"
  }
```

Failure

Extra Property Present

Status: 400 Bad Request

Body:

```
{
  "Error": "One or more properties in the request are not valid"
}
```

Value of 'name' is not valid

Status: 400 Bad Request

Body:

```
{
  "Error": "name value in request is not valid"
}
```

Value of 'type' is not valid

Status: 400 Bad Request

Body:

```
{
  "Error": "type value in request is not valid"
}
```

Value of 'length' is not valid

Status: 400 Bad Request

Body:

```
{
  "Error": "length value in request is not valid"
}
```

Syntax error in JSON

Status: 400 Bad Request

Body:

```
{
  "Error": "Unexpected string in JSON at position 13"
}
```

Invalid or missing JWT Bearer Token in Authorization header

Status: 401 Unauthorized

Body:

```
{
  "Error": "Bad Credentials"
}
```

Boat with passed-in id does not exist or JWT in Authorization header corresponds to a user who does not own the boat

Status: 403 Forbidden

```
{
  "Error": "Boat does not exist or is owned by someone else"
}
```

Accept header does not indicate application/json is acceptable

Status: 406 Not Acceptable

Body:

```
{
  "Error": "Endpoint only can respond with application/json data"
}
```

Delete Boat

DELETE /boats/:boatId

Deletes boat with passed-in id. Boats are a protected resource, so request must contain a valid JWT. Boat will be deleted only if the boat belongs to the user whose JWT is in the Authorization header as a Bearer token.

Request

Path Parameters

Name	Description
boatId	id of the boat to be deleted

Request Headers

Header	Notes
Authorization	must be set to a Bearer token of a registered user

Query Parameters

None

Request Body

None

Response

Response Body Format

Success: No body

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	401 Unauthorized	If the “Authorization” header is not set to a valid JWT, the boat will not be deleted.
Failure	403 Forbidden	If no boat exists with the passed-in boatId or if the “Authorization” header is set to a valid JWT but the JWT does not correspond to the user who owns the boat, no boat will not be deleted.

Response Examples

Success

Status: 204 No Content

Failure

Invalid or missing JWT Bearer Token in Authorization header

Status: 401 Unauthorized

Body:

{

```
    "Error": "Bad Credentials"
}
```

Boat with passed-in id does not exist or JWT in Authorization header corresponds to a user who does not own the boat

Status: 403 Forbidden

```
{
  "Error": "Boat does not exist or is owned by someone else"
}
```

Create A Load

POST /loads

Allows creation of a new load.

Request

Path Parameters

None

Request Headers

Header	Notes
Accepts	must be set to application/json

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Property	Description	Type	Notes
volume	volume of the load in cubic feet	integer	Required. Cannot be null. Must be a positive integer.

Property	Description	Type	Notes
item	item in the load	string	Required. Cannot be null. Must not include the characters "<>{}[]". Must not be an empty string or over 30 characters in length. Must not have whitespace at the beginning or end of string. Required. Cannot be null. Must be of the form "XX/XX/XXXX" where each group of characters is a number.
creation_date	date the load was created	string	

Request Body Example

```
{
  "volume": 50,
  "item": "Roses",
  "creation_date": "10/20/2022"
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	201 Created	
Failure	400 Bad Request	If the request is missing any of the 3 required attributes, the boat will not be created.

Outcome	Status Code	Notes
Failure	406 Not Acceptable	If the “Accept” header does not indicate application/json will be accepted, no boat will be created.

Response Examples

Success

Status: 201 Created

Body:

```
{
  "volume": 50,
  "item": "Roses",
  "creation_date": "10/20/2022",
  "carrier": null,
  "id": "5702893864747008",
  "self": "https://melsbyg-cloud-final.uw.r.appspot.com/loads/5702893864747008"
}
```

Failure

Missing Required Property

Status: 400 Bad Request

Body:

```
{
  "Error": "Request is missing one or more required properties"
}
```

Extra Property Present

Status: 400 Bad Request

Body:

```
{
  "Error": "One or more properties in the request are not valid"
}
```

Value of ‘volume’ is not valid

Status: 400 Bad Request

Body:

```
{
  "Error": "volume value in request is not valid"
}
```

Value of 'item' is not valid

Status: 400 Bad Request

```
Body:
{
  "Error": "item value in request is not valid"
}
```

Value of 'creation_date' is not valid

Status: 400 Bad Request

```
Body:
{
  "Error": "creation_date value in request is not valid"
}
```

Syntax error in JSON

Status: 400 Bad Request

```
Body:
{
  "Error": "Unexpected string in JSON at position 13"
}
```

Accept header does not indicate application/json is acceptable

Status: 406 Not Acceptable

```
Body:
{
  "Error": "Endpoint only can respond with application/json data"
}
```

Get All Loads

GET /loads

Gets a list of all loads. Response is paginated and returns at most 5 boats per page.

Request

Path Parameters

None

Request Headers

Header	Notes
Accepts	must be set to application/json

Query Parameters

Name	Description
cursor	Optional. Corresponds to a Datastore cursor. Helps implement pagination.

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	400 Bad Request	If the 'cursor' query parameter is not recognized by Datastore, no list of loads will be returned.
Failure	406 Not Acceptable	If the "Accept" header does not indicate application/json will be accepted, no list of loads will be returned.

Response Examples

Success

The URL indicated by “next” is how the next page of results is accessed.

If “next” is not present, there are no more results.

“count” indicates the total number of loads.

Status: 200 OK

Body:

```
{
  "loads": [
    {
      "carrier": null,
      "item": "Water Bottles",
      "creation_date": "09/20/2001",
      "volume": 80,
      "id": "4802063531769856",
      "self": "https://melsbyg-cloud-final.uw.r.appspot.com/loads/4802063531769856"
    },
    {
      "carrier": null,
      "volume": 70,
      "item": "Bananas",
      "creation_date": "09/20/2000",
      "id": "5075408403824640",
      "self": "https://melsbyg-cloud-final.uw.r.appspot.com/loads/5075408403824640"
    },
    {
      "item": "Roses",
      "carrier": null,
      "creation_date": "10/21/2022",
      "volume": 40,
      "id": "5638358357245952",
      "self": "https://melsbyg-cloud-final.uw.r.appspot.com/loads/5638358357245952"
    },
    {
      "volume": 60,
      "carrier": null,
      "creation_date": "10/20/2000",
      "item": "Phones",
      "id": "5679095853613056",
      "self": "https://melsbyg-cloud-final.uw.r.appspot.com/loads/5679095853613056"
    },
    {
      "creation_date": "10/20/2022",
```

```

        "carrier": {
          "id": "6263382969679872",
          "self": "https://melsbyg-cloud-final.uw.r.appspot.com/boats/6263382969679872",
        },
        "volume": 50,
        "item": "Roses",
        "id": "5702893864747008",
        "self": "https://melsbyg-cloud-final.uw.r.appspot.com/loads/5702893864747008"
      }
    ],
    "next": "https://melsbyg-cloud-final.uw.r.appspot.com/loads?cursor=CjISLGoXenV3fm1lbHNie",
    "count": 6
  }
}

```

Failure

‘cursor’ query parameter not recognized by Datastore

Status: 400 Bad Request

```

Body:
{
  "Error": "Cursor in request params not recognized"
}

```

Accept header does not indicate application/json is acceptable

Status: 406 Not Acceptable

```

Body:
{
  "Error": "Endpoint only can respond with application/json data"
}

```

Get Load by Id

GET /loads/:loadId

Gets representation of load with passed-in id.

Request

Path Parameters

Name	Description
loadId	id of the load to be retrieved

Request Headers

Header	Notes
Accepts	must be set to application/json

Query Parameters

None

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	404 Not Found	If no load exists with the passed-in loadId, no load will be returned.
Failure	406 Not Acceptable	If the “Accept” header does not indicate application/json will be accepted, no load will be returned.

Response Examples

Success

Status: 200 OK

Body:

```
{
  "carrier": {
    "id": "5680529164730368",
    "self": "https://melsbyg-cloud-final.uw.r.appspot.com/boats/5680529164730368"
  },
  "volume": 50,
```

```
    "creation_date": "10/20/2022",
    "item": "Roses",
    "id": "5714489739575296",
    "self": "https://melsbyg-cloud-final.uw.r.appspot.com/loads/5714489739575296"
}
```

Failure

Load with passed-in id does not exist

Status: 404 Not Found

```
{
  "Error": "Load does not exist"
}
```

Accept header does not indicate application/json is acceptable

Status: 406 Not Acceptable

Body:

```
{
  "Error": "Endpoint only can respond with application/json data"
}
```

Edit Load with PUT

PUT /loads/:loadId

Edits load with passed-in id. Requires all required attributes in request body.
Cannot change carrier.

Request

Path Parameters

Name	Description
loadId	id of the load to be edited

Request Headers

Header	Notes
Accepts	must be set to application/json

Query Parameters

None

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Property	Description	Type	Notes
volume	volume of the load in cubic feet	integer	Required. Cannot be null. Must be a positive integer. Required. Cannot be null. Must not include the characters “<>{}[]”. Must not be an empty string or over 30 characters in length. Must not have whitespace at the beginning or end of string.
item	item in the load	string	
creation_date	date the load was created	string	Required. Cannot be null. Must be of the form “XX/XX/XXXX” where each group of characters is a number.

Request Body Example

```
{
  "volume": 30,
  "item": "Wine",
  "creation_date": "09/09/1999"
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	400 Bad Request	If the request is missing any of the 3 required attributes, the load will not be edited.
Failure	404 Not Found	If no load exists with the passed-in loadId, no load will be edited.
Failure	406 Not Acceptable	If the "Accept" header does not indicate application/json will be accepted, no boat will be returned.

Response Examples

Success

Status: 200 OK

Body:

```
{
  "volume": 30,
  "item": "Wine",
  "creation_date": "09/09/1999",
  "carrier": {
    "id": "5680529164730368",
    "self": "https://melsbyg-cloud-final.uw.r.appspot.com/boats/5680529164730368"
  },
  "id": "5714489739575296",
  "self": "https://melsbyg-cloud-final.uw.r.appspot.com/loads/5714489739575296"
}
```

Failure

Missing Required Property

Status: 400 Bad Request

Body:

```
{  
  "Error": "Request is missing one or more required properties"  
}
```

Extra Property Present

Status: 400 Bad Request

Body:

```
{  
  "Error": "One or more properties in the request are not valid"  
}
```

Value of 'volume' is not valid

Status: 400 Bad Request

Body:

```
{  
  "Error": "volume value in request is not valid"  
}
```

Value of 'item' is not valid

Status: 400 Bad Request

Body:

```
{  
  "Error": "item value in request is not valid"  
}
```

Value of 'creation_date' is not valid

Status: 400 Bad Request

Body:

```
{  
  "Error": "creation_date value in request is not valid"  
}
```

Syntax error in JSON

Status: 400 Bad Request

Body:

```
{  
  "Error": "Unexpected string in JSON at position 13"  
}
```

Load with passed-in id does not exist

Status: 404 Not Found

```
{
  "Error": "Load does not exist"
}
```

Accept header does not indicate application/json is acceptable

Status: 406 Not Acceptable

Body:

```
{
  "Error": "Endpoint only can respond with application/json data"
}
```

Edit Load with PATCH

PATCH /loads/:loadId

Edits load with passed-in id. Not all properties are required in request body. Included properties will update on the load and other properties will remain the same. Cannot change carrier.

Request

Path Parameters

Name	Description
loadId	id of the load to be edited

Request Headers

Header	Notes
Accepts	must be set to application/json

Query Parameters

None

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Property	Description	Type	Notes
volume	volume of the load in cubic feet	integer	If included, cannot be null. Must be a positive integer.
item	item in the load	string	If included, cannot be null. Must not include the characters "<>{}[]". Must not be an empty string or over 30 characters in length. Must not have whitespace at the beginning or end of string.
creation_date	date the load was created	string	If included, cannot be null. Must be of the form "XX/XX/XXXX" where each group of characters is a number.

Request Body Example

```
{
  "volume": 40,
  "item": "Flowers"
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	400 Bad Request	If the properties in the request are not valid, the load will not be edited.
Failure	404 Not Found	If no load exists with the passed-in loadId, no load will be edited.
Failure	406 Not Acceptable	If the “Accept” header does not indicate application/json will be accepted, no boat will be returned.

Response Examples

Success

Status: 200 OK

Body:

```
{
  "item": "Flowers",
  "creation_date": "10/20/2022",
  "carrier": {
    "id": "5680529164730368",
    "self": "https://melsbyg-cloud-final.uw.r.appspot.com/boats/5680529164730368"
  },
  "volume": 40,
  "id": "5714489739575296",
  "self": "https://melsbyg-cloud-final.uw.r.appspot.com/loads/5714489739575296"
}
```

Failure

Extra Property Present

Status: 400 Bad Request

Body:

```
{
  "Error": "One or more properties in the request are not valid"
}
```

Value of 'volume' is not valid

Status: 400 Bad Request

Body:

```
{  
  "Error": "volume value in request is not valid"  
}
```

Value of 'item' is not valid

Status: 400 Bad Request

Body:

```
{  
  "Error": "item value in request is not valid"  
}
```

Value of 'creation_date' is not valid

Status: 400 Bad Request

Body:

```
{  
  "Error": "creation_date value in request is not valid"  
}
```

Syntax error in JSON

Status: 400 Bad Request

Body:

```
{  
  "Error": "Unexpected string in JSON at position 13"  
}
```

Load with passed-in id does not exist

Status: 404 Not Found

```
{  
  "Error": "Load does not exist"  
}
```

Accept header does not indicate application/json is acceptable

Status: 406 Not Acceptable

Body:

```
{  
  "Error": "Endpoint only can respond with application/json data"  
}
```

Delete Load

DELETE /loads/:loadId

Deletes load with passed-in id.

Request

Path Parameters

Name	Description
loadId	id of the load to be deleted

Request Headers

None

Query Parameters

None

Request Body

None

Response

Response Body Format

Success: No body

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	404 Not Found	If no load exists with the passed-in loadId, no load will not be deleted.

Response Examples

Success

Status: 204 No Content

Failure

Load with passed-in id does not exist

Status: 404 Not Found

```
{
  "Error": "Load does not exist"
}
```

Put Load on Boat

PUT /boats/:boatId/load/:loadId

Places load with passed-in loadId on boat with passed-in loadId. Boats are a protected resource, so request must contain a valid JWT. Load will be placed on boat only if the boat belongs to the user whose JWT is in the Authorization header as a Bearer token and the load is not already on another boat. Updates “carrier” property of load.

Request

Path Parameters

Name	Description
loadId	id of the load to be placed on the boat
boatId	id of the boat the load is to be placed on

Request Headers

Header	Notes
Authorization	must be set to a Bearer token of a registered user

Query Parameters

None

Request Body

None

Response

Response Body Format

Success: No body

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	401 Unauthorized	If the “Authorization” header is not set to a valid JWT, the load will not be placed on the boat.
Failure	403 Forbidden	If no boat exists with the passed-in boatId, if the “Authorization” header is set to a valid JWT but the JWT does not correspond to the user who owns the boat, or if the load is already on a boat, the load will not be placed on the boat.
Failure	404 Not Found	If no load exists with the passed-in loadId, the load will not be placed on the boat.

Response Examples

Success

Status: 204 No Content

Failure

Invalid or missing JWT Bearer Token in Authorization header

Status: 401 Unauthorized

Body:

```
{
  "Error": "Bad Credentials"
```

```
}
```

Boat with passed-in id does not exist, JWT in Authorization header corresponds to a user who does not own the boat, or the load is already on a boat

Status: 403 Forbidden

```
{
```

```
  "Error": "The boat is owned by someone else or does not exist, or the load is already l
```

```
}
```

Load with passed-in id does not exist.

Status: 404 Not Found

Body:

```
{
```

```
  "Error": "The load does not exist"
```

```
}
```

Remove Load from Boat

DELETE /boats/:boatId/load/:loadId

Removes load with passed-in loadId on boat with passed-in loadId. Boats are a protected resource, so request must contain a valid JWT. Load will be removed from boat only if the boat belongs to the user whose JWT is in the Authorization header as a Bearer token and the load is on the boat. Updates “carrier” property of load to null.

Request

Path Parameters

Name	Description
loadId	id of the load to be removed from the boat
boatId	id of the boat the load is to be removed from

Request Headers

Header	Notes
Authorization	must be set to a Bearer token of a registered user

Query Parameters

None

Request Body

None

Response

Response Body Format

Success: No body

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	401 Unauthorized	If the “Authorization” header is not set to a valid JWT, the load will not be removed from the boat.
Failure	403 Forbidden	If no boat exists with the passed-in boatId, if the “Authorization” header is set to a valid JWT or if the JWT does not correspond to the user who owns the boat, the load will not be removed from the boat.
Failure	404 Not Found	If no load exists with the passed-in loadId or if the load is not on the boat with the passed-in boatId, no load will not be removed from the boat.

Response Examples

Success

Status: 204 No Content

Failure

Invalid or missing JWT Bearer Token in Authorization header

Status: 401 Unauthorized

Body:

```
{
  "Error": "Bad Credentials"
}
```

Boat with passed-in id does not exist, JWT in Authorization header corresponds to a user who does not own the boat, or the load is already on a boat

Status: 403 Forbidden

```
{
  "Error": "The boat is owned by someone else or does not exist"
}
```

Load with passed-in id does not exist.

Status: 404 Not Found

Body:

```
{
  "Error": "The load does not exist"
}
```

Get List of Users

GET /users

Gets a list of the application's users.

Request

Path Parameters

None

Request Headers

Header	Notes
Accepts	must be set to application/json

Query Parameters

None

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	406 Not Acceptable	If the “Accept” header does not indicate application/json will be accepted, no list of users will be returned.

Response Examples

Success

Status: 200 OK

Body:

```
[
  {
    "email": "wallace@cheese.com",
    "id": "auth0|6383e893a8b2c2ec60b332c9"
  },
  {
    "email": "melsbyg@oregonstate.edu",
    "id": "google-oauth2|101200093123048132943"
  }
]
```

Failure

Accept header does not indicate application/json is acceptable

Status: 406 Not Acceptable

Body:

```
{  
  "Error": "Endpoint only can respond with application/json data"  
}
```