

Final Project: API Spec

CS 493: Cloud Application Development

Spring 2022

Oregon State University

Last Update: June 4, 2022

Change log	1
Data Model	2
Create a Boat	5
Get a Boat	8
Edit Boat Attribute	10
Replace a Boat	13
Delete a Boat	16
Get All Boats - Paginated	18
Create a Load	21
Get a Load	23
Edit Load Attribute	25
Replace a Load	28
Delete a Load	31
Get All Loads - Paginated	32
Get All Loads for a Boat	35
Add A Load to a Boat	37
Delete A Load from a Boat	39
Get All Users	40

Change log

Version	Change	Date
1.0	Initial version.	June 1, 2022

Data Model

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

Boats

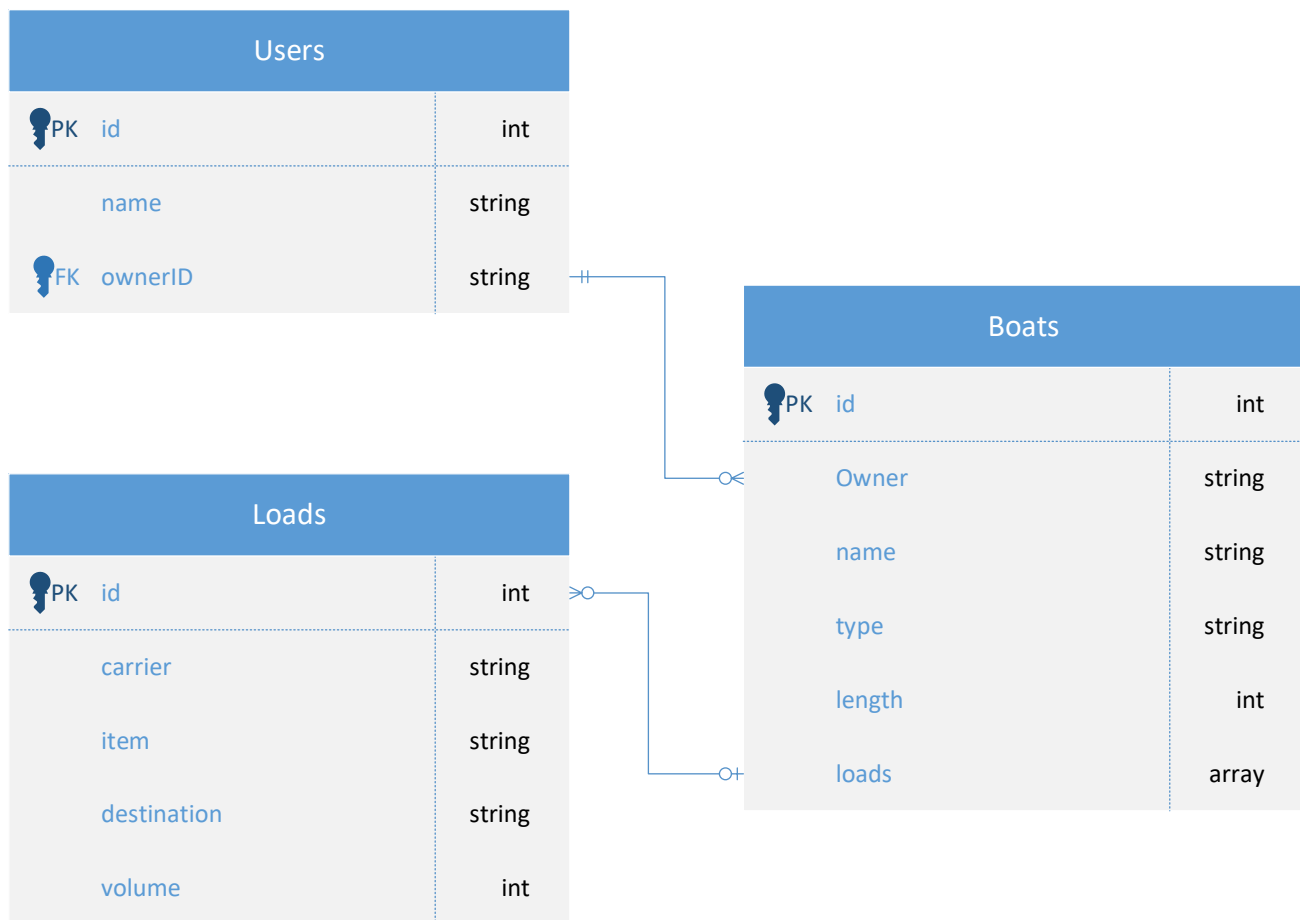
Property	Data Type	Required	Notes
id	Integer	Yes	The id of the boat. Auto-generated by Datastore
Owner	String	Yes	ownerID of user who owns boat.
name	String	Yes	Name of the boat.
type	String	Yes	Type of the boat. E.g., Sailboat, Catamaran, etc.
length	Integer	Yes	The length of the boat in feet.
Loads	Array	No	Id of loads that are on the boat. Otherwise empty

Loads

Property	Data Type	Required	Notes
id	Integer	Yes	The id of the load. Auto-generated by Datastore
volume	Integer	Yes	The volume of the load.
item	String	Yes	Load item.
destination	String	Yes	Destination of load.
carrier	Integer	No	ID of the boat carrying the load.

Users

Property	Data Type	Required	Notes
id	Integer	Yes	The id of the user. Auto-generated by Datastore.
name	String	Yes	Name of the user upon account creation
ownerID	String	Yes	Unique owner ID for the user. "Sub" value of JWT upon account creation.



Users

Users entity stores the name and unique owner id of all users in the collection regardless of whether they are the owner of a boat or not. When the user creates an account through the UI (via Auth0) the name and unique owner id (value of sub from Auth0 JWT) are stored in the user entity.

Users have a relationship with the Boats entity. A user can own zero or more boats and a boat must be owned by one, and only one, owner. When a boat is created the ownerID from the Users entity is populated in the owners attribute of the Boats entity.

Users entity has no direct relationship with Loads entity.

Loads

Loads entity stores the load carrier, item, volume, and destination information for each load. Loads entity has a relationship with the Boats entity. A Load can be on at most one boat while a boat can have zero or more loads. When a load is created, the carrier will always be null until the load is assigned to a boat. When the load is assigned to a boat, the carrier attribute will be populated with

the details of the boat it was assigned to and the load information will be added to the loads array for the boat in the Boats entity.

When a load is deleted, if the load was assigned to a boat, the load id will be removed from the loads attribute for the boat in the Boats entity.

Loads has no direct relationship with Users.

Boats

Boats entity stores the name, type, length, owner, and loads attributes of a boat. The Boats entity has a relationship with both the Users entity and the Loads entity. A boat must be owned by one, and only one, owner from the Users entity. A boat can have many loads from the Loads entity but a load can only be associated with one boat at most.

Since a boat is required to have an owner, when a boat is created the request requires a valid JWT as a bearer token for authorization. Upon the successful creation of a boat, the loads array will be empty until loads are assigned and the owner attribute in the Boats entity will be populated with the value of “sub” from the JWT which becomes the unique owner ID for the boat.

All Crud operations related to the boat itself (not including any operations pertaining to boat loads e.g. assigning a load to a boat) are protected and require a valid JWT. When a boat is deleted, if the boat had loads assigned to it, the carrier attribute for the load in the Loads entity will be updated to null.

Create a Boat

Protected endpoint allowing the creation of a new boat for a user with a valid JWT.

POST /boats

Request

Authorization

Required: Yes

Type: Bearer Token

Token Value: JWT

Path Parameters

None

Request Body

Required

Request Body Format

Application/json

Response

Response Body Format

application/json

Request JSON Attributes

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

Request Body Example

```
{
  "name": "Blue Pearl",
  "type": "Pirate",
  "length": "56"
}
```

Response Statuses

Outcome	Status Code	Notes
Success	201 Created	
Failure	400 Bad Request	Invalid data
Failure	401 Unauthorized	Missing or invalid JWT
Failure	415 unsupported media type	Server only accepts application/json data.

Response Examples

Success

```
Status: 201
{
  "id": "5071211717459968",
  "name": "Blue Pearl",
  "type": "Pirate",
  "length": "56",
  "owner": "auth0|627ac7f493f62100671c205f",
  "loads": [],
  "self": "http://cs493-final-leggetb.uw.r.appspot.com/boats/5071211717459968"
}
```

Failure

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

```
Status: 400 Bad Request
{
  "Error": "Only numbers are accepted for boat length"
}
```

```
Status: 400 Bad Request
{
  "Error": "Only accepted attributes are name, type, and length"
}
```

```
Status: 400 Bad Request
{
  "Error": "Only string type acceptable for boat name and boat type"
}
```

```
Status: 401 Unauthorized
{
  "Error": "Unauthorized"
}
```

```
Status: 415 Unsupported Media Type
{
```

```
"Error": "Server only accepts application/json data."  
}
```

Comment

- Load will always be empty upon boat creation.

Get a Boat

Protected endpoint for the retrieval of an existing boat that belongs to a user with a valid JWT.

GET /boats/:boat_id

Request

Authorization

Required: Yes

Type: Bearer Token

Token Value: JWT

Path Parameters

Name	Description
boat_id	ID of the boat

Request Body

None

Response

Response Body Format

Success: No body

Failure: application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	Request has missing or invalid JWT
Failure	403 Forbidden	Boat does not belong to user
Failure	404 Not Found	A boat with this boat id does not exist
Failure	406 Not Acceptable	Invalid media type

Response Examples

Success

```
Status: 200 OK
{
  "id": "5644004762845184",
  "name": "Blue Pearl",
  "type": "Pirate",
  "length": "56",
```



```
"owner": "auth0|627ac7f493f62100671c205f",  
"loads": [],  
"self": "http://cs493-final-leggetb.uw.r.appspot.com/boats/5644004762845184"  
}
```

Failure

Status: 401 Unauthorized

```
{  
  "Error": "Unauthorized"  
}
```

Status: 403 Forbidden

```
{  
  "Error": "Forbidden"  
}
```

Status: 404 Not Found

```
{  
  "Error": "No boat with this boat_id exists"  
}
```

Status: 406 Not Acceptable

```
{  
  "Error": "Media Type Not Acceptable"  
}
```

Edit Boat Attribute

Protected endpoint for updating any attribute of an existing boat belonging to the user of a valid JWT while the other attributes remain unchanged.

PATCH /boats/:boat_id

Request

Authorization

Required: Yes

Type: Bearer Token

Token Value: JWT

Path Parameters

Name	Description
boat_id	ID of the boat

Request Body

Required

Request Body Format

application/json

Request JSON Attributes

Name	Description	Data Type Constraints	Required?
name	The name of the boat.	String	No
type	The type of the boat. E.g., Sailboat, Catamaran, etc.	String	No
length	Length of the boat in feet.	Integer	No

Comments

Available attributes to update: name, type, length. ID, owner, and loads attribute is not modifiable.
A minimum of one attribute must be specified for update.

Request Body Example

```
{  
  "name": "Sea Siren"  
}
```

Response

Response Body Format

application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	400 Bad Request	Invalid data
Failure	401 Unauthorized	Request has missing or invalid JWT
Failure	403 Forbidden	Boat does not belong to user
Failure	404 Not Found	No boat with this boat_id exists
Failure	415 unsupported media type	Server only accepts application/json data.

Response Examples

Success

Status: 200 OK

```
{
  "id": "5644004762845184",
  "name": "Sea Siren",
  "type": "Pirate",
  "length": "56",
  "owner": "auth0|627ac7f493f62100671c205f",
  "loads": [],
  "self": "http://cs493-final-leggetb.uw.r.appspot.com/boats/5644004762845184"
}
```

Failure

Status: 400 Bad Request

```
{
  "Error": "A minimum of one value must be specified for update"
}
```

Status: 400 Bad Request

```
{
  "Error": "Request cannot contain a non-modifiable value"
}
```

Status: 400 Bad Request

```
{
  "Error": "Only numbers are accepted for boat length"
}
```

```
}
```

Status: 400 Bad Request

```
{  
  "Error": "Only string type acceptable for boat name and boat type"  
}
```

Status: 401 Unauthorized

```
{  
  "Error": "Unauthorized"  
}
```

Status: 403 Forbidden

```
{  
  "Error": "Forbidden"  
}
```

Status: 404 Not Found

```
{  
  "Error": "No boat with this boat_id exists"  
}
```

Status: 415 Unsupported Media Type

```
{  
  "Error": "Server only accepts application/json data."  
}
```

Replace a Boat

Protected endpoint where all modifiable attributes of an existing boat that belong to a user with a valid JWT are updated.

PUT /boats/:boat_id

Request

Authorization

Required: Yes

Type: Bearer Token

Token Value: JWT

Path Parameters

Name	Description
boat_id	ID of the boat

Request Body

Required

Request Body Format

application/json

Request JSON Attributes

Name	Description	Data Type Constraints	Required?
name	The name of the boat.	String	No
type	The type of the boat. E.g., Sailboat, Catamaran, etc.	String	No
length	Length of the boat in feet.	Integer	No

Comments

Available attributes to update: name, type, length. ID, owner, and loads attribute is not modifiable.
A minimum of one attribute must be specified for update.

Request Body Example

```
{
  "name": "Sea Tugger",
  "type": "Tug",
  "length": 15
}
```

Response

Response Body Format

application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	400 Bad Request	Invalid data
Failure	401 Unauthorized	Request has missing or invalid JWT
Failure	403 Forbidden	Boat does not belong to user
Failure	404 Not Found	No boat with this boat_id exists
Failure	415 unsupported media type	Server only accepts application/json data.

Response Examples

Success

```
Status: 200

{
  "id": "5644004762845184",
  "name": "Sea Tugger",
  "type": "Tug",
  "length": "15",
  "owner": "auth0|627ac7f493f62100671c205f",
  "loads": [],
  "self": "http://cs493-final-leggetb.uw.r.appspot.com/boats/5644004762845184"
}
```

Failure

```
Status: 400 Bad Request

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

```
Status: 400 Bad Request

{
  "Error": "Request cannot contain a non-modifiable value"
}
```

```
Status: 400 Bad Request

{
  "Error": "Only numbers are accepted for boat length"
}
```

Status: 400 Bad Request

```
{  
  "Error": "Only string type acceptable for boat name and boat type"  
}
```

Status: 401 Unauthorized

```
{  
  "Error": "Unauthorized"  
}
```

Status: 403 Forbidden

```
{  
  "Error": "Forbidden"  
}
```

Status: 404 Not Found

```
{  
  "Error": "No boat with this boat_id exists"  
}
```

Status: 415 Unsupported Media Type

```
{  
  "Error": "Server only accepts application/json data."  
}
```

Delete a Boat

Protected endpoint for deleting a boat belonging to a user with a valid JWT. Note that if the boat currently has a load, deleting the boat makes the carrier value in load to become empty.

DELETE /boats/:boat_id

Request

Authorization

Required: Yes

Type: Bearer Token

Token Value: JWT

Path Parameters

Name	Description
boat_id	ID of the boat

Request Body

None

Response

Response Body Format

Success: No body

Failure: application/json

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	401 Unauthorized	Request has missing or invalid JWT
Failure	403 Forbidden	Boat does not belong to user
Failure	404 Not Found	A boat with this boat id does not exist

Response Examples

Success

Status: 204 No Content

Failure

Status: 401 Unauthorized

<pre>{ "Error": "Unauthorized" }</pre>
--

Status: 403 Forbidden

```
{  
  "Error": "Forbidden"
```

Status: 404 Not Found

```
{  
"Error": "No boat with this boat_id exists"
```

Get All Boats - Paginated

Protected endpoint that returns all boats related to the user associated with the valid JWT provided in the request. Uses a cursor to get all boats paginated returning 5 boats per page with a next link at the bottom, if there are more results, directing to the next 5 results.

GET /boats

Request

Authorization

Required: Yes

Type: Bearer Token

Token Value: JWT

Path Parameters

None

Request Body

None

Response

Response Body Format

application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	Missing or invalid JWT
Failure	406 Not Acceptable	Invalid media type

Response Examples

Success

Status: 200 OK

```
{
  "boats": [{
    "type": "Yacht",
    "length": 18,
    "name": "401K",
    "loads": [],
```

```

    "owner": "auth0|627ac7f493f62100671c205f",
    "id": "5250870199451648",
    "self": "http://cs493-final-leggetb.uw.r.appspot.com/boats/5250870199451648"
  }, {
    "name": "Lucifer",
    "type": "Yacht",
    "owner": "auth0|627ac7f493f62100671c205f",
    "loads": ["4731398703284224", "6629167181332480", "5857298610126848", "5294348656705536"],
    "length": 15,
    "id": "5503267274489856",
    "self": "http://cs493-final-leggetb.uw.r.appspot.com/boats/5503267274489856"
  }, {
    "loads": [],
    "type": "Sailing Boat",
    "length": 13,
    "owner": "auth0|627ac7f493f62100671c205f",
    "name": "Calypso",
    "id": "5524935921369088",
    "self": "http://cs493-final-leggetb.uw.r.appspot.com/boats/5524935921369088"
  }, {
    "length": 18,
    "owner": "auth0|627ac7f493f62100671c205f",
    "type": "Yacht",
    "name": "401K",
    "loads": [],
    "id": "5813820152872960",
    "self": "http://cs493-final-leggetb.uw.r.appspot.com/boats/5813820152872960"
  }, {
    "name": "Calypso",
    "owner": "auth0|627ac7f493f62100671c205f",
    "type": "Sailing Boat",
    "length": 13,
    "loads": [],

```

```
"id": "6087885874790400",
"self": "http://cs493-final-leggetb.uw.r.appspot.com/boats/6087885874790400"
}],
"next": "http://cs493-final-
leggetb.uw.r.appspot.com/boats?cursor=CjMSLWoXenV3fmNzNDkzLWZpbmFsLWxlZ2dldGJyEgsSBUJvYXRzG
ICAgJjmnOgKDBgAlAA=",
"Total_Items": 6
}
```

Failure

Status: 401 Unauthorized

```
{
  Error: "Unauthorized"
}
```

Status: 406 Not Acceptable

```
{
  Error: "Media Type Not Acceptable"
}
```

Create a Load

Unprotected endpoint for creating a new load.

POST /load

Request

Authorization

None

Path Parameters

None

Request Body

Required

Request Body Format

Application/json

Response

Response Body Format

application/json

Request JSON Attributes

Name	Description	Data Type Constraints	Required?
Item	Contents of the load	String	Yes
Volume	Volume of load	Integer	Yes
Destination	Destination of load	string	Yes

Request Body Example

```
{
  "volume": 5,
  "item": "LEGO Blocks",
  "destination": "Madrid"
}
```

Response Statuses

Outcome	Status Code	Notes
Success	201 Created	
Failure	400 Bad Request	Invalid data
Failure	415 unsupported media type	Server only accepts application/json data.

Response Examples

Success

```
{
  "id": "6264990059200512",
  "volume": 5,
  "item": "LEGO Blocks",
  "destination": "Madrid",
  "carrier": null,
  "self": "http://cs493-final-leggetb.uw.r.appspot.com/loads/6264990059200512"
}
```

Failure

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

```
Status: 400 Bad Request
{
  "Error": "Only numbers are accepted for boat length"
}
```

```
Status: 400 Bad Request
{
  "Error": "Only accepted attributes are item, volume, and created date"
}
```

```
Status: 400 Bad Request
{
  "Error": "Only string type acceptable for load item and created date"
}
```

```
Status: 415 Unsupported Media Type
{
  "Error": "Server only accepts application/json data."
}
```

Comment

- Carrier will always be empty upon load creation.

Get a Load

Unprotected endpoint for retrieving an existing load.

GET /loads/:load_id

Request

Authorization

None

Path Parameters

Name	Description
load_id	ID of the load

Request Body

None

Response

Response Body Format

Success: No body

Failure: application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	404 Not Found	A load with this load id does not exist
Failure	406 Not Acceptable	Invalid media type

Response Examples

Success

```
Status: 200 OK
{
  "id": "6264990059200512",
  "volume": 5,
  "item": "LEGO Blocks",
  "destination": "Madrid",
  "carrier": null,
  "self": "http://cs493-final-leggetb.uw.r.appspot.com/loads/6264990059200512"
}
```

Failure

Status: 404 Not Found

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

Status: 406 Not Acceptable

```
{  
  "Error": "Media Type Not Acceptable"  
}
```


Edit Load Attribute

Unprotected endpoint allows for updating any attributes of an existing load while the other attributes remain unchanged.

PATCH /loads/:load_id

Request

Authorization

None

Path Parameters

Name	Description
load_id	ID of the load

Request Body

Required

Request Body Format

application/json

Request JSON Attributes

Name	Description	Data Type Constraints	Required?
item	Load contents	String	No
Volume	Volume of load (tons)	integer	No
Destination	Destination of load	string	No

Comments

Available attributes to update: item, volume, and destination. ID attribute is not modifiable. Carrier is not modifiable through this endpoint.

A minimum of one attribute must be specified for update.

Request Body Example

```
{
  "item": "Board Games"
}
```

Response

Response Body Format

application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	

Failure	400 Bad Request	Invalid data
Failure	404 Not Found	No boat with this boat_id exists
Failure	415 unsupported media type	Server only accepts application/json data.

Response Examples

Success

```
Status: 200 OK
{
  "id": 6617661500817408,
  "item": "Board Games",
  "volume": 5,
  "destination": "san Juan, PR",
  "carrier": null,
  "self": "http://cs493-final-leggetb.uw.r.appspot.com/loads/6617661500817408"
}
```

Failure

```
Status: 400 Bad Request
{
  "Error": "A minimum of one value must be specified for update"
}
```

```
Status: 400 Bad Request
{
  "Error": "Request cannot contain a non-modifiable value"
}
```

```
Status: 400 Bad Request
{
  "Error": "Only numbers are accepted for volume"
}
```

```
Status: 400 Bad Request
{
  "Error": "Only string type acceptable for load item and destination"
}
```

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

```
Status: 415 Unsupported Media Type
{
```

```
"Error": "Server only accepts application/json data."  
}
```

Replace a Load

Unprotected endpoint for all modifiable attributes of an existing load to be updated.

PUT /loads/:load_id

Request

Authorization

None

Path Parameters

Name	Description
load_id	ID of the load

Request Body

Required

Request Body Format

application/json

Request JSON Attributes

Name	Description	Data Type Constraints	Required?
item	Load contents	String	No
volume	Volume of load (tons)	integer	No
destination	Destination of load	string	No

Comments

Available attributes to update: item, volume, destination. ID and carrier are not modifiable.

A minimum of one attribute must be specified for update.

Request Body Example

```
{
  "volume": 5,
  "item": "LEGO Blocks",
  "destination": "San Juan, PR"
}
```

Response

Response Body Format

application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	

Failure	400 Bad Request	Invalid data
Failure	404 Not Found	No load with this load_id exists
Failure	415 unsupported media type	Server only accepts application/json data.

Response Examples

Success

```
Status: 200
{
  "id": 6617661500817408,
  "item": "LEGO Blocks",
  "volume": 5,
  "destination": "San Juan, PR",
  "carrier": null,
  "self": "http://cs493-final-leggetb.uw.r.appspot.com/loads/6617661500817408"
}
```

Failure

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

```
Status: 400 Bad Request
{
  "Error": "Request cannot contain a non-modifiable value"
}
```

```
Status: 400 Bad Request
{
  "Error": "Only numbers are accepted for load volume"
}
```

```
Status: 400 Bad Request
{
  "Error": "Only string type acceptable for load item and destination"
}
```

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

```
Status: 415 Unsupported Media Type
{
```

```
"Error": "Server only accepts application/json data."  
}
```

Delete a Load

Unprotected endpoint allows you to delete a boat. Note that if the load is on a boat, deleting the load removes the load from the boat.

DELETE /loads/:load_id

Request

Authorization

None

Path Parameters

Name	Description
load_id	ID of the load

Request Body

None

Response

Response Body Format

Success: No body

Failure: application/json

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	404 Not Found	A boat with this boat id does not exist

Response Examples

Success

Status: 204 No Content

Failure

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

Get All Loads - Paginated

Unprotected endpoint returns all loads. Uses a cursor to get all loads paginated returning 5 loads per page with a next link at the bottom, if there are more results, directing to the next 5 results.

GET /loads

Request

Authorization

None

Path Parameters

None

Request Body

None

Response

Response Body Format

application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	406 Not Acceptable	Invalid media type

Response Examples

Success

Status: 200 OK

```
{
  "loads": [{
    "destination": "San Juan, PR",
    "carrier": "6650835828211712",
    "volume": 5,
    "item": "LEGO Blocks",
    "id": "4610142247059456",
    "self": "http://cs493-final-leggetb.uw.r.appspot.com/loads/4610142247059456"
  }, {
    "volume": 4,
```



```

    "carrier": null,
    "destination": "San Juan, PR",
    "item": "Toys",
    "id": "4901538329788416",
    "self": "http://cs493-final-leggetb.uw.r.appspot.com/loads/4901538329788416"
  }, {
    "volume": 5,
    "carrier": "6220526812921856",
    "destination": "San Juan, PR",
    "item": "LEGO Blocks",
    "id": "4969395222740992",
    "self": "http://cs493-final-leggetb.uw.r.appspot.com/loads/4969395222740992"
  }, {
    "item": "Food",
    "carrier": "4891617223770112",
    "volume": 3,
    "destination": "San Juan, PR",
    "id": "5012873679994880",
    "self": "http://cs493-final-leggetb.uw.r.appspot.com/loads/5012873679994880"
  }, {
    "item": "Stuffed Animals",
    "volume": 10,
    "destination": "San Juan, PR",
    "carrier": "4891617223770112",
    "id": "5183013306499072",
    "self": "http://cs493-final-leggetb.uw.r.appspot.com/loads/5183013306499072"
  }
],
"next": "http://cs493-final-leggetb.uw.r.appspot.com/loads?cursor=CjMSLWoXenV3fmNzNDkzLWZpbmFsLWxlZ2dldGJyEgsSBUxvYWRzGICAgJjFvZzJDbGAlAA=",
"Total_Items": 11
}

```

Failure

Status: 406 Not Acceptable

{

Error: "Media Type Not Acceptable"

}

Get all Loads for a Boat

Unprotected endpoints for retrieving all loads assigned to an existing boat

GET /boats/:boat_id/loads

Request

Authorization

None

Path Parameters

Name	Description
boat_id	ID of the boat

Request Body

None

Response

Response Body Format

Application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	404 Not Found	No boat with this boat_id exists

Response Examples

Success

Status: 200 OK

```
{
  "loads": [{
    "carrier": "4721851393638400",
    "volume": 3,
    "destination": "San Juan, PR",
    "item": "Food",
    "id": "4576967919665152",
    "self": "http://cs493-final-leggetb.uw.r.appspot.com/req.baseUrl/4576967919665152"
  },
  {
    "item": "Toys",
    "volume": 4,
```

```
"destination": "San Juan, PR",  
"carrier": "4721851393638400",  
"id": "4998352664002560",  
"self": "http://cs493-final-leggetb.uw.r.appspot.com/req.baseUrl/4998352664002560"  
}  
]  
}
```

Failure

```
Status: 406 Not Acceptable  
{  
  Error: "Media Type Not Acceptable"  
}
```

Add a Load to a Boat

Unprotected endpoint allows for adding a Load to a boat.

PUT /boats/:boat_id/loads/load_id

Request

Authorization

None

Path Parameters

Name	Description
boat_id	ID of the boat
load_id	ID of load

Request Body

Required

Request Body Format

Application/json

Response

Response Body Format

application/json

Response Statuses

Outcome	Status Code	Notes
Success	204 OK	
Failure	403 Bad Request	The load is already on another boat
Failure	404 Not Found	The boat and/or load does not exist

Response Examples

Success

Status: 204 No Content

Failure

Status: 403 Bad Request

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

Status: 404 Not Found

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

Delete a Load from a Boat

Unprotected endpoint allows for deleting a load from a boat. When a load is deleted from a boat, the carrier value in loads will become empty.

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

Request

Authorization

None

Path Parameters

Name	Description
boat_id	ID of the boat
load_id	ID of load

Request Body

None

Response

Response Body Format

Failure: No body

Success: application/json

Response Statuses

Outcome	Status Code	Notes
Success	204 OK	
Failure	404 Not Found	No boat with this boat_id is loaded with the load with this load_id

Response Examples

Success

```
Status: 204 No Content
```

Failure

```
Status: 404 Not Found
```

```
{
  "Error": " No boat with this boat_id is loaded with the load with this load_id"
}
```

Get All Users

Unprotected endpoint that returns all users regardless of whether they are associated with a boat.

GET /users

Request

Authorization

None

Path Parameters

None

Request Body

None

Response

Response Body Format

application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	

Response Examples

Success

Status: 200 OK

```
[
  {
    "uniqueID": "auth0|629c5c2acd1f83006facbf2",
    "name": "newuser",
    "id": "4715270597771264"
  },
  {
    "uniqueID": "auth0|62817e4d5b321c006fe57755",
    "name": "test",
    "id": "4872136191639552"
  }
]
```