

Final Project: Data Model and API Specification

CS 493: Cloud Application Development

Fall 2022

Steven Tieu

Oregon State University

Application URLs	1
Data Model	2
View all Users	5
Create a Boat for a User	6
View/get a Boat for a User	8
View/get all Boats for a User	10
Delete a Boat for a User	13
Fully edit/update a Boat for a User	15
Partially edit/update a Boat for a User	18
Create a Load	21
View/get a Load	23
View/get all Loads	25
Delete a Load	27
Fully update/edit a Load	28
Partially update/edit a Load	30
Assign a Load to a Boat	32
Remove a Load from a Boat	34

Application URLs

URL where application is deployed on GCP: <https://tieus-final.wl.r.appspot.com/>

URL for account creation/login: <https://tieus-final.wl.r.appspot.com/>

Data Model

This API has three entities: users, boats, and loads.

Users

Property	Data Type	Notes	Required?	Valid Values Example
id	Integer	Id of the User. Automatically generated by Datastore.	No (generated by Datastore)	67310298320
name	String	Name of the User. Represented by the User's JSON Web Token(JWT)'s "name" property value.	No (obtained from User's JWT's "name" property value)	"email@email.com"
unique_id	String	The unique ID of the User. This is represented by the "sub" property value obtained from a User's JSON Web Token(JWT).	No (obtained from User's JWT's "sub" property value)	"auth0 89017a0490b9010"

Boats

Property	Data Type	Notes	Required?	Valid Values Example
id	Integer	Id of the boat. Automatically generated by Datastore.	Yes (Required in the path parameters for certain endpoints such as GET /boats/:boat_id and PUT /boats/:boat_id/loads/:load_id More details in the API specification.)	75894378597
name	String	Name of the boat.	Yes (Required in request body for certain endpoints such as POST /boats and PUT /boats/:boat_id. More details in the API specification)	"Greenhorn"
type	String	The type/model of the boat.	Yes (Required in request body for certain endpoints such as POST /boats and PUT /boats/:boat_id. More details in the API specification)	"Warship"
length	Integer	The length of the boat in feet.	Yes (Required in request body for certain endpoints such as POST /boats and PUT /boats/:boat_id. More details in the API specification)	1000

owner	String	The owner of the boat. Automatically generated from a User's JWT upon creation of a boat.	No	"auth0 89017a0490b9010"
loads	Array of Objects	The loads carried by the boat. "Loads" objects contain the id(integer) and self-URL(string) of the loads as properties.	No	Example 1: [] Example 2: [{ "id": 9148390388, "self": https://tieus-final.wl.r.appspot.com/loads/481984309890 }, { "id": 9148390388, "self": https://tieus-final.wl.r.appspot.com/loads/48129890898 }]
self	String	The self-URL linked directly to the subject boat.	No (automatically generated by server)	" https://tieus-final.wl.r.appspot.com/boats/48129890898 "

Loads

Property	Data Type	Notes	Required?	Valid Values Example
id	Integer	Id of the load. Automatically generated by Datastore.	Yes (Required in the path parameters for certain endpoints such as GET /loads/:load_id and PUT /boats/:boat_id/loads/:load_id More details in the API specification.)	849083209489
volume	Integer	The volume of the load.	Yes (Required in request body for certain endpoints such as POST /loads and PUT /loads. More details in the API specification)	2000

carrier	Object	The boat carrying the load. "Carrier" objects contain the id(integer), name(string), and self-URL(string) of the boat as properties. If there is no carrier, the carrier is <i>null</i> .	No	Example 1: null Example 2: { "id": 519039012, "name": "Yokozuna", "self": "https://tieus-final.wl.r.appspot.com/boats/519039012" }
item	String	The item of the load. E.g. "Hot dogs".	Yes (Required in request body for certain endpoints such as POST /loads and PUT /loads/:load_id. More details in the API specification)	"Hot dogs"
creation_date	String	The date when the load was created. E.g. "12/25/22".	Yes (Required in request body for certain endpoints such as POST /loads and PUT /loads/:load_id. More details in the API specification)	"12/22/22"
self	String	The self-URL linked directly to the subject load.	No (automatically generated by server)	"https://tieus-final.wl.r.appspot.com/loads/5123390242"

Description of the relationship between the non-Users entities:

A Load can be loaded onto a Boat. When this happens, the Load is added to the list representing the Boat's "loads" property value. In the "loads" property list, each Load carried by the Boat is represented by an object containing the Load's "id" and "self" properties. Additionally, the Load's "carrier" property value becomes an object representing the Boat. The object contains the Boat's "id", "name", and "self" properties.

Description of how the Users entity is modeled:

The Users entity is modeled with the properties "id", "name", and "unique_id". The "id" property value is generated by Datastore automatically. The "name" property value is taken from the "name" property value in a User's JSON Web Token(JWT). The "unique_id" property is the unique identifier for a User in the data model. The "unique_id" property value is taken from the "sub" property value of the User's JWT. In order for someone to make a request for requiring a unique User identifier, the person needs to supply a JWT in the "Authorization" header of the request as a Bearer Token. The format for the "Authorization" header is a string as follows: "Bearer {{YOUR_JWT_HERE}}". The relationship between the user entity and the non-user entity is between Users and Boats. Therefore, a person needs to supply a User's JWT in the "Authorization" header of requests towards the /boats and /boats/:boat_id endpoints(as they are protected) in order for the requests to succeed. Users can create, view, edit, or delete Boats with a valid JWT. When a Boat is created, the "sub" property value of the User's JWT (which is also the User's "unique_id" property value) is used to represent the value of the Boat's "owner" property value. For requests to the /boats/:boat_id endpoint, the "sub" property value of the User's JWT is compared with the "owner" property value of the Boat. If they are not equal, the request is rejected.

View all Users

Displays all User entities.

GET /users

Request

Path Parameters

None

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header

No

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	406 Not Acceptable	If the request does not accept a JSON-formatted response, then return 406 Not Acceptable.

Response Examples

Success

Status: 200 OK

```
{
  "users": [
    {
      "name": "mannyp@kixotic.com",
      "unique_id": "auth0|793480adfk89d9a09931asf9"
    },
    {
      "name": "mickeym@cosaxu.com",
      "unique_id": "auth0|894832fdsk14e3ru0u089089"
    }
  ]
}
```

Failure

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable. Must accept JSON."
}
```

Create a Boat for a User

Creates a new boat for a User corresponding to a supplied JWT.

POST /boats

Request

Path Parameters

None

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header
Yes

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?
name	Name of the boat as a string	Yes
type	Type/model of the boat as a string	Yes
length	Length of the boat in feet as an integer	Yes

Request Body Example

```
{
  "name": "Why Not?",
  "type": "Pacifica Oasis",
  "length": 40
}
```

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 three required attributes, then the boat will not be created, and 400 status code will return. If the request has an extraneous/invalid attribute, then the boat will not be created, and 400 status code will return.

Failure	401 Unauthorized	<p>If the request is missing an authorization JWT, then boat is not created and 401 status code will be returned.</p> <p>If the request supplies an authorization JWT but is invalid, then the boat is not created and 401 status code will be returned.</p>
Failure	406 Not Acceptable	If the request does not accept a JSON-formatted response, then the Boat is not created and status code 406 Not Acceptable is returned.

Response Examples

- Datastore will automatically generate an ID and store it with the entity being created.

Success

<p>Status: 201 Created</p> <pre>{ "id": 431890418900918, "name": "Leviathan", "type": "Oasis", "length": 1000 "loads": [], "owner": "auth0 89017a0490b9010", "self": "https://tieus-final.wl.r.appspot.com/boats/431890418900918" }</pre>

Failure

<p>Status: 400 Bad Request</p> <pre>{ "Error": "The request object is missing at least one of the required attributes or includes an extraneous/invalid attribute" }</pre>
<p>Status: 401 Unauthorized</p> <pre>{ "code": "no auth header", "description": "Authorization header is missing" }</pre>
<p>Status: 401 Unauthorized</p> <pre>{ "code": "invalid_header", "description": "Invalid header. Use an RS256 signed JWT Access Token" }</pre>
<p>Status: 406 Not Acceptable</p> <pre>{ "Error": "Not Acceptable. Must accept JSON." }</pre>

View/get a Boat for a User

View/get an existing Boat for a User corresponding to a supplied JWT.

GET /boats/:boat_id

Request

Path Parameters

Name	Description
boat_id	The ID of the boat.

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header
Yes

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	If the request is missing an authorization JWT, then the requested Boat is not displayed and 401 status code will be returned. If the request supplies an authorization JWT but is invalid, then the requested Boat is not displayed and 401 status code will be returned.
Failure	403 Forbidden	If the request supplies an authorization JWT but the “sub” property value of the authorization JWT does not match the “owner” property value of the requested Boat entity, the requested Boat is not displayed and 403 status code will be returned.
Failure	404 Not Found	No boat with this boat_id exists
Failure	406 Not Acceptable	If the request does not accept a JSON-formatted response, then the requested Boat is not displayed and 406 Not Acceptable status code is returned.

Response Examples

Success

Status: 200 OK

```
{
  "id": 431890418900918,
  "name": "Leviathan",
```



```

"type": "Oasis",
"length": 1000
"loads": [
  {
    "id": 9148390388,
    "self": https://tieus-final.wl.r.appspot.com/loads/481984309890
  },
  {
    "id": 9148390388,
    "self": https://tieus-final.wl.r.appspot.com/loads/48129890898
  }
],
"owner": "auth0|89017a0490b9010",
"self": "https://tieus-final.wl.r.appspot.com/boats/431890418900918"
}

```

Failure

Status: 401 Unauthorized

```

{
  "code": "no auth header",
  "description": "Authorization header is missing"
}

```

Status: 401 Unauthorized

```

{
  "code": "invalid_header",
  "description": "Invalid header. Use an RS256 signed JWT Access Token"
}

```

Status: 403 Forbidden

```

{
  "Error": "Viewing this boat is not allowed for this user"
}

```

Status: 404 Not Found

```

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

```

Status: 406 Not Acceptable

```

{
  "Error": "Not Acceptable. Must accept JSON."
}

```

View/get all Boats for a User

View/get all existing boats for a User corresponding to a supplied JWT.

GET /boats

Request

Path Parameters

None

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header
Yes

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	If the request is missing an authorization JWT, then the requested Boats are not displayed and 401 status code will be returned. If the request supplies an authorization JWT but is invalid, then the requested Boats are not displayed and 401 status code will be returned.
Failure	406 Not Acceptable	If the request does not accept a JSON-formatted response, then the requested Boat is not displayed and 406 Not Acceptable status code is returned.

Response Examples

Success

Status: 200 OK

```
{
  "boats": [
    {
      "id": 5079418695319552,
      "length": 200,
      "loads": [],
      "name": "Black Shadow",
      "owner": "auth0|638a8745a442109d49bd43e4",
      "self": "https://tieus-final.wl.r.appspot.com/boats/5079418695319552",
      "type": "Stealth Ship"
    },
    {
```

```
"id": 5147403464212480,
"length": 200,
"loads": [],
"name": "Black Shadow",
"owner": "auth0|638a8745a442109d49bd43e4",
"self": "https://tieus-final.wl.r.appspot.com/boats/5147403464212480",
"type": "Stealth Ship"
},
{
  "id": 5151539786153984,
  "length": 200,
  "loads": [],
  "name": "Black Shadow",
  "owner": "auth0|638a8745a442109d49bd43e4",
  "self": "https://tieus-final.wl.r.appspot.com/boats/5151539786153984",
  "type": "Stealth Ship"
},
{
  "id": 5631671361601536,
  "length": 200,
  "loads": [],
  "name": "Black Shadow",
  "owner": "auth0|638a8745a442109d49bd43e4",
  "self": "https://tieus-final.wl.r.appspot.com/boats/5631671361601536",
  "type": "Stealth Ship"
},
{
  "id": 5636645067948032,
  "length": 200,
  "loads": [],
  "name": "Black Shadow",
  "owner": "auth0|638a8745a442109d49bd43e4",
  "self": "https://tieus-final.wl.r.appspot.com/boats/5636645067948032",
  "type": "Stealth Ship"
}
],
"next": "https://tieus-final.wl.r.appspot.com/boats?limit=5&offset=5",
"total_number_of_boats": 8
}
```

Failure

Status: 401 Unauthorized

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

Status: 401 Unauthorized

```
{  
  "code": "invalid_header",  
  "description": "Invalid header. Use an RS256 signed JWT Access Token"  
}
```

Status: 406 Not Acceptable

```
{  
  "Error": "Not Acceptable. Must accept JSON."  
}
```

Delete a Boat for a User

Deletes a Boat for a User corresponding to a supplied JWT. If a load is currently assigned to a boat, the boat is removed as the load's carrier.

DELETE /boats/:boat_id

Request

Path Parameters

Name	Description
boat_id	The ID of the boat

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header
Yes

Request Body

None

Response

No body

Response Body Format

Success: No body

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	401 Unauthorized	If the request is missing an authorization JWT, then the requested Boat is not deleted and 401 status code will be returned. If the request supplies an authorization JWT but is invalid, then the requested Boat is not deleted displayed and 401 status code will be returned.
Failure	403 Forbidden	If the request supplies an authorization JWT but the "sub" property value of the authorization JWT does not match the "owner" property value of the requested Boat entity, the requested Boat is not deleted and 403 status code will be returned.
Failure	404 Not Found	No boat with this boat_id exists

Response Examples

Success

Status: 204 No Content

Failure

Status: 401 Unauthorized

<pre>{ "code": "no auth header", "description": "Authorization header is missing" }</pre>
<p>Status: 401 Unauthorized</p> <pre>{ "code": "invalid_header", "description": "Invalid header. Use an RS256 signed JWT Access Token" }</pre>
<p>Status: 403 Forbidden</p> <pre>{ 'Error': "Deleting this boat is not allowed for this user" }</pre>
<p>Status: 404 Not Found</p> <pre>{ "Error": "No boat with this boat_id exists" }</pre>

Fully edit/update a Boat for a User

Fully edit/update a Boat for a User corresponding to a supplied JWT.

PUT /boats/:boat_id

Request

Path Parameters

None

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header
Yes

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?
name	Name of the boat as a string	Yes
type	Type/model of the boat as a string	Yes
length	Length of the boat in feet as an integer	Yes

Request Body Example

```
{
  "name": "Leviathan",
  "type": "Oasis",
  "length": 1000
}
```

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 three required attributes, then the boat is not edited/updated, and 400 status code will return. If the request has an extraneous/invalid attribute, then the boat is not edited/updated, and 400 status code will return.

Failure	401 Unauthorized	<p>If the request is missing an authorization JWT, then boat is not edited/updated and 401 status code will be returned.</p> <p>If the request supplies an authorization JWT but is invalid, then the boat is not edited/updated and 401 status code will be returned.</p>
Failure	403 Forbidden	If the request supplies an authorization JWT but the “sub” property value of the authorization JWT does not match the “owner” property value of the requested Boat entity, the requested Boat is edited/updated and 403 status code will be returned.
Failure	406 Not Acceptable	If the request does not accept a JSON-formatted response, then the Boat is not edited/updated and status code 406 Not Acceptable is returned.

Response Examples

- Datastore will automatically generate an ID and store it with the entity being created.

Success

Status: 200 OK

```
{
  "id": 431890418900918,
  "name": "Leviathan",
  "type": "Oasis",
  "length": 1000
  "loads": [],
  "owner": "auth0|89017a0490b9010",
  "self": "https://tieus-final.wl.r.appspot.com/boats/431890418900918"
}
```

Failure

Status: 400 Bad Request

```
{
  "Error": "The request object is missing at least one of the required attributes
or includes an extraneous/invalid attribute"
}
```

Status: 401 Unauthorized

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

Status: 401 Unauthorized

```
{
  "code": "invalid_header",
```


<pre>"description": "Invalid header. Use an RS256 signed JWT Access Token" }</pre>
<pre>Status: 403 Forbidden { "Error": "Editing/updating this boat is not allowed for this user" }</pre>
<pre>Status: 406 Not Acceptable { "Error": "Not Acceptable. Must accept JSON." }</pre>

Partially edit/update a Boat for a User

Partially edit/update a Boat for a User corresponding to a supplied JWT.

PATCH /boats/:boat_id

Request

Path Parameters

None

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header
Yes

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?
name	Name of the boat as a string	Yes/No*
type	Type/model of the boat as a string	Yes/No*
length	Length of the boat in feet as an integer	Yes/No*

*Having at least one of the above attributes is required in the request body

Request Body Examples

<pre>{ "name": "Leviathan", "type": "Oasis", "length": 1000 }</pre>
<pre>{ "type": "Oasis", "length": 1000 }</pre>
<pre>{ "length": 1000 }</pre>

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	

Failure	400 Bad Request	<p>If the request does not have at least one valid attribute, then the boat is not edited/updated, and 400 status code will return.</p> <p>If the request has an extraneous/invalid attribute, then the boat is not edited/updated, and 400 status code will return.</p>
Failure	401 Unauthorized	<p>If the request is missing an authorization JWT, then boat is not edited/updated and 401 status code will be returned.</p> <p>If the request supplies an authorization JWT but is invalid, then the boat is not edited/updated and 401 status code will be returned.</p>
Failure	403 Forbidden	<p>If the request supplies an authorization JWT but the “sub” property value of the authorization JWT does not match the “owner” property value of the requested Boat entity, the requested Boat is edited/updated and 403 status code will be returned.</p>
Failure	406 Not Acceptable	<p>If the request does not accept a JSON-formatted response, then the Boat is not edited/updated and status code 406 Not Acceptable is returned.</p>

Response Examples

- Datastore will automatically generate an ID and store it with the entity being created.

Success

Status: 200 OK

```
{
  "id": 431890418900918,
  "name": "Leviathan",
  "type": "Oasis",
  "length": 1000
  "loads": [],
  "owner": "auth0|89017a0490b9010",
  "self": "https://tieus-final.wl.r.appspot.com/boats/431890418900918"
}
```

Failure

Status: 400 Bad Request

```
{
  "Error": "The request object is missing a valid attribute or includes an extraneous/invalid attribute"
}
```

Status: 401 Unauthorized

<pre>{ "code": "no auth header", "description": "Authorization header is missing" }</pre>
<p>Status: 401 Unauthorized</p> <pre>{ "code": "invalid_header", "description": "Invalid header. Use an RS256 signed JWT Access Token" }</pre>
<p>Status: 403 Forbidden</p> <pre>{ "Error": "Editing/updating this boat is not allowed for this user" }</pre>
<p>Status: 406 Not Acceptable</p> <pre>{ "Error": "Not Acceptable. Must accept JSON." }</pre>

Create a Load

Creates a new load.

POST /loads

Request

Path Parameters

None

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header

No

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?
volume	The volume of the load as an integer.	Yes
item	The item of the load as a string. E.g. "Hot dogs".	Yes
creation_date	The date the load was created. E.g. "12/25/22".	Yes

Request Body Example

```
{
  "volume": 100
  "item": "Hot dogs"
  "creation_date": "4/20/20"
}
```

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 load will not be created. Then, a 400 status code will be sent.
Failure	406 Not Acceptable	If the request does not accept a JSON-formatted response, then the Load is not created and status code 406 Not Acceptable is returned.

Response Examples

- Datastore will automatically generate an ID and store it with the load being created.
- The value of the attribute "carrier" is an object representing the boat currently holding the load. If there is no carrier holding the load, the value of "carrier" should be null.

Success

Status: 201 Created

```
{
  "id": 4856008086126592,
  "carrier": null,
  "creation_date": "10/18/2021",
  "item": "LEGO Blocks",
  "self": "https://tieus-final.wl.r.appspot.com/loads/4856008086126592",
  "volume": 5
}
```

Failure

Status: 400 Bad Request

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

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable. Must accept JSON."
}
```

View/get a Load

View/get an existing load.

GET /loads/:load_id

Request

Path Parameters

Name	Description
load_id	The ID of the load

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header
No

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	404 Not Found	No load with this load_id exists
Failure	406 Not Acceptable	If the request does not accept a JSON-formatted response, then the Load is not displayed and status code 406 Not Acceptable is returned.

Response Examples

Success

Status: 200 OK
<pre>{ "carrier": null, "creation_date": "03/11/10", "id": 4865928487501824, "item": "GameBoys", "self": "https://tieus-final.wl.r.appspot.com/loads/4865928487501824", "volume": 555 }</pre>

Failure

Status: 404 Not Found
<pre>{ "Error": "No load with this load_id exists" }</pre>
Status: 406 Not Acceptable

```
{  
  "Error": "Not Acceptable. Must accept JSON."  
}
```


View/get all Loads

View/get all existing Loads.

GET /loads

Request

Path Parameters

None

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header

No

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	406 Not Acceptable	If the request does not accept a JSON-formatted response, then the loads are not displayed and status code 406 Not Acceptable is returned.

Response Examples

Success

Status: 200 OK

```
{
  "loads": [
    {
      "carrier": null,
      "creation_date": "03/11/10",
      "id": 4865928487501824,
      "item": "GameBoys",
      "self": "https://tieus-final.wl.r.appspot.com/loads/4865928487501824",
      "volume": 555
    },
    {
      "carrier": null,
      "creation_date": "12/12/20",
      "id": 5083538508480512,
      "item": "Hats",
      "self": "https://tieus-final.wl.r.appspot.com/loads/5083538508480512",
      "volume": 1000
    },
  ],
}
```

```
{
  "carrier": null,
  "creation_date": "12/12/20",
  "id": 5428878440923136,
  "item": "Hats",
  "self": "https://tieus-final.wl.r.appspot.com/loads/5428878440923136",
  "volume": 1000
},
{
  "carrier": null,
  "creation_date": "12/12/20",
  "id": 5635008819625984,
  "item": "Hats",
  "self": "https://tieus-final.wl.r.appspot.com/loads/5635008819625984",
  "volume": 1000
},
{
  "carrier": null,
  "creation_date": "12/12/20",
  "id": 5677459739508736,
  "item": "Hats",
  "self": "https://tieus-final.wl.r.appspot.com/loads/5677459739508736",
  "volume": 1000
}
],
"next": "https://tieus-final.wl.r.appspot.com/loads?limit=5&offset=5",
"total_number_of_loads": 6
}
```

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable. Must accept JSON."
}
```

Delete a Load

Deletes a Load. If the load being deleted has a carrier, the load is removed from that carrier's loads list.

DELETE /loads/:load_id

Request

Path Parameters

Name	Description
load_id	The ID of the load

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header
No

Request Body

None

Response

No body

Response Body Format

Success: No body

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	404 Not Found	No load with this load_id exists

Response Examples

Success

Status: 204 No Content

Failure

Status: 404 Not Found

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

Fully update/edit a Load

Fully updates/edits a Load.

PUT /loads/:load_id

Request

Path Parameters

None

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header

No

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?
volume	The volume of the load as an integer.	Yes
item	The item of the load as a string. E.g. "Hot dogs".	Yes
creation_date	The date the load was created. E.g. "12/25/22".	Yes

Request Body Example

```
{
  "volume": 100
  "item": "Hot dogs"
  "creation_date": "4/20/20"
}
```

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 is not be edited/updated. Then, a 400 status code will be sent.
Failure	404 Not Found	No load with this load_id exists
Failure	406 Not Acceptable	If the request does not accept a JSON-formatted response, then the Load is not edited/updated and status code 406 Not Acceptable is returned.

Response Examples

Success

Status: 200 OK

```
{
  "id": 4856008086126592,
  "carrier": null,
  "creation_date": "10/18/2021",
  "item": "LEGO Blocks",
  "self": "https://tieus-final.wl.r.appspot.com/loads/4856008086126592",
  "volume": 5
}
```

Failure

Status: 400 Bad Request

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

Status: 404 Not Found

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

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable. Must accept JSON."
}
```

Partially update/edit a Load

Partially updates/edits a Load.

PATCH /loads/:load_id

Request

Path Parameters

None

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header

No

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?
volume	The volume of the load as an integer.	Yes/No*
item	The item of the load as a string. E.g. "Hot dogs".	Yes/No*
creation_date	The date the load was created. E.g. "12/25/22".	Yes/No*

*Having at least one of the above attributes in the request body is required

Request Body Examples

<pre>{ "volume": 100 "item": "Hot dogs" "creation_date": "4/20/20" }</pre>
<pre>{ "volume": 100 "item": "Hot dogs" }</pre>
<pre>{ "volume": 100 }</pre>

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	400 Bad Request	If the request does have at least one valid attribute in it's body, then the load is not be edited/updated. Then, a 400 status code will be sent.

Failure	404 Not Found	No load with this load_id exists
Failure	406 Not Acceptable	If the request does not accept a JSON-formatted response, then the Load is not edited/updated and status code 406 Not Acceptable is returned.

Response Examples

Success

Status: 200 OK

```
{
  "id": 4856008086126592,
  "carrier": null,
  "creation_date": "10/18/2021",
  "item": "LEGO Blocks",
  "self": "https://tieus-final.wl.r.appspot.com/loads/4856008086126592",
  "volume": 5
}
```

Failure

Status: 400 Bad Request

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

Status: 404 Not Found

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

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable. Must accept JSON."
}
```

Assign a Load to a Boat

Assigns a Load to a Boat.

PUT /boats/:boat_id/loads/:load_id

Request

Path Parameters

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

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header
No

Request Body

None

Response

No body

Response Body Format

Success: No body

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	Succeeds only if a boat exists with the given boat_id, a load exists with given load_id and the load is not being carried by any boat.
Failure	403 Forbidden	The load is already loaded on another boat.
Failure	404 Not Found	The specified boat and/or load does not exist

Response Examples

Success

Status: 204 No Content

Failure

Status: 403 Forbidden

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

Status: 404 Not Found

```
{
```



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

Comment

- A load cannot be assigned to more than one boat.

Remove a Load from a Boat

Removes a load from a boat.

DELETE /boats/:boat_id/loads/:load_id

Request

Path Parameters

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

Protected Endpoint requiring a JSON Web Token(JWT) as Bearer Token in Authorization Header
No

Request Body

None

Response

No body

Response Body Format

Success: No body

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	Succeeds only if a boat exists with this boat_id, a load exists with this load_id and the load is on the boat.
Failure	404 Not Found	No boat with this boat_id is loaded with the load with this load_id. This could be because no boat with the given boat_id exists, no load with the given load_id exists, or even if both boat_id and slip_id are valid, the boat with this given boat_id is not loaded with the load with the given 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"
}
```