

**Justin Tromp**

**CS 492**

**Assignment 4**

**Application URL:** <https://trompj-assignment-4.wl.r.appspot.com>

<b>Endpoint function</b>	<b>Pages</b>
Create a boat	2-3
Get a page of boats	4
Get a boat	5
Delete a boat	6
Create a load	7-8
Get a page of loads	9
Get a load	10
Delete a load	11
Add a load to a boat	12
Get loads carried by a boat	13

## CREATE A BOAT

### POST /boats

Function: Create a boat with an empty load. All newly created boats will be created with an empty load and cannot be set during creation.

#### Request

##### Request Parameters:

None

##### Request Body:

Required

##### Request Body Format:

JSON

#### Request JSON Attributes

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

#### Request Body Example

```
{
  "name": "boat name",
  "type": "boat type",
  "length": 120
}
```

#### Response

##### Response Body Format

JSON

#### Response Statuses

Outcome	Status	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, and 400 status will be returned.

#### Response Examples

##### Success

##### Status: 201 Created

```
{
  "id": 5642061785071616,
  "name": "Example name",
  "length": 120,
  "loads": [],
  "type": "Example type",
  "self": "https://trompj-assignment-4.appspot.com/boats/5642061785071616"
}
```

## Failure

Status: 400 Bad Request

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

# GET A PAGE OF BOATS

## GET /boats

Function: View all boats. Request uses pagination and returns 3 boats at a time with a link to the next page of boats.

### Request

#### Request Parameters:

None

#### Request Body:

None

### Response

#### Response Body Format

JSON

#### Response Statuses

Outcome	Status	Notes
Success	200 OK	

### Response Examples

#### Success

#### Status: 200 OK

```
{
  "boats": [
    {
      "id": 5642061785071616,
      "length": 120,
      "loads": [],
      "name": "Test_Name",
      "self": "https://trompj-assignment-4.appspot.com/boats/5642061785071616",
      "type": "Test_Type"
    },
    {
      "id": 5650452045627392,
      "length": 120,
      "loads": [],
      "name": "Test_Name",
      "self": "https://trompj-assignment-4.appspot.com/boats/5650452045627392",
      "type": "Test_Type"
    },
    {
      "id": 5675594515742720,
      "length": 120,
      "loads": [],
      "name": "Test_Name",
      "self": "https://trompj-assignment-4.appspot.com/boats/5675594515742720",
      "type": "Test_Type"
    }
  ],
  "next": "http://localhost:8080/boats?limit=3&offset=3"
}
```

## GET A BOAT

### GET /boats/:boat\_id

Function: View a boat based on provided ID.

#### Request

##### Request Parameters:

Name	Type	Description	Required?
boat_id	String	ID of the boat	Yes

#### Request Body:

None

#### Response

##### Response Body Format

JSON

#### Response Statuses

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

#### Response Examples

##### Success

##### Status: 200 OK

```
{
  "id": "5642061785071616",
  "length": 120,
  "loads": [],
  "name": "Example name",
  "self": "https://trompj-assignment-4.appspot.com/boats/5642061785071616",
  "type": "Example type"
}
```

##### Failure

##### Status: 404 Not Found

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

## DELETE A BOAT

### DELETE /boats/:boat\_id

Function: Delete a boat. If a boat is currently carrying a load, deletion of the boat causes the load entities carrier field to be emptied as it is no longer carried by the boat.

#### Request

##### Request Parameters:

None

##### Request Body:

None

#### Response

No body

#### Response Body Format

Success: No body

Failure: JSON

#### Response Statuses

Outcome	Status	Notes
Success	204 No Content	
Failure	404 Not Found	No boat with this ID exists

#### Response Examples

##### Success

Status: 204 No Content

##### Failure

Status: 404 Not Found

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

## CREATE A LOAD

### POST /loads

Function: Create a load. All loads created are created without a carrier automatically. A load may be added to a carrier through the modify load endpoint.

#### Request

##### Request Parameters:

None

##### Request Body:

Required

##### Request Body Format:

JSON

#### Request JSON Attributes

Name	Type	Description	Required?
weight	Integer	Weight of the load in kilograms.	Yes
content	String	The type of content that the load consists of.	Yes
delivery_date	Integer	The delivery date required of the load.	Yes

#### Request Body Example

```
{
  "weight": 10,
  "content": "LEGO Blocks",
  "delivery_date": "01/05/2021"
}
```

#### Response

##### Response Body Format

JSON

#### Response Statuses

Outcome	Status	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, and 400 status will be returned.

#### Response Examples

##### Success

##### Status: 201 Created

```
{
  "carrier": null,
  "content": "LEGO Blocks",
  "delivery_date": "01/05/2021",
  "id": 5649205632696320,
  "self": "https://trompj-assignment-4.appspot.com/loads/5649205632696320",
  "weight": 10
}
```

## Failure

Status: 400 Bad Request

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



## GET A PAGE OF LOADS

### GET /loads

Function: View all loads. Request uses pagination and returns 3 loads at a time (if available) with a link to the next page of boats. If there are 3 or less loads, there will be no next link.

#### Request

##### Request Parameters:

None

##### Request Body:

None

#### Response

##### Response Body Format

JSON

##### Response Statuses

Outcome	Status	Notes
Success	200 OK	

##### Response Examples

###### Success

###### Status: 200 OK

```
{
  "loads": [
    {
      "carrier": null,
      "content": "LEGO Blocks",
      "delivery_date": "01/05/2021",
      "id": 5649205632696320,
      "self": "https://trompj-assignment-4.appspot.com/loads/5649205632696320",
      "weight": 10
    },
    {
      "carrier": null,
      "content": "LEGO Blocks",
      "delivery_date": "01/05/2021",
      "id": 5674591867371520,
      "self": "https://trompj-assignment-4.appspot.com/loads/5674591867371520",
      "weight": 10
    },
    {
      "carrier": null,
      "content": "LEGO Blocks",
      "delivery_date": "01/05/2021",
      "id": 5715110588841984,
      "self": "https://trompj-assignment-4.appspot.com/loads/5715110588841984",
      "weight": 10
    }
  ],
  "next": "http://localhost:8080/loads?limit=3&offset=3"
}
```

## GET A LOAD

### GET /loads/:load\_id

Function: View a load based on provided ID.

#### Request

##### Request Parameters:

Name	Type	Description	Required?
load_id	String	ID of the load	Yes

#### Request Body:

None

#### Response

##### Response Body Format

JSON

#### Response Statuses

Outcome	Status	Notes
Success	200 OK	
Failure	404 Not Found	No load with this ID exists.

#### Response Examples

##### Success

##### Status: 200 OK

```
{
  "carrier": null,
  "content": "LEGO Blocks",
  "delivery_date": "01/05/2021",
  "id": "5649205632696320",
  "self": "https://trompj-assignment-4.appspot.com/loads/5649205632696320",
  "weight": 10
}
```

##### Failure

##### Status: 404 Not Found

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

## DELETE A LOAD

### DELETE /loads/:load\_id

Function: Delete a load. If a load is currently being carried by a boat, the load will be removed from the boat prior to deletion.

#### Request

##### Request Parameters:

None

##### Request Body:

None

#### Response

No body

#### Response Body Format

Success: No body

Failure: JSON

#### Response Statuses

Outcome	Status	Notes
Success	204 No Content	
Failure	404 Not Found	No load with this ID exists.

#### Response Examples

##### Success

Status: 204 No Content

##### Failure

Status: 404 Not Found

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

## ADD A LOAD TO A BOAT

### PUT /boats/:boat\_id/loads/:load\_id

Function: Assign a load to a boat. The load with the load\_id provided will be added to the boat with the boat\_id provided if the load is not already assigned to a boat.

#### Request

##### Request Parameters:

Name	Type	Description	Required?
boat_id	String	ID of the boat to add a load to	Yes
load_id	String	ID of the load to be added	Yes

#### Request Body:

None

#### Response

No body

#### Response Body Format

Success: No body

Failure: JSON

#### Response Statuses

Outcome	Status	Notes
Success	204 No Content	
Failure	403 Forbidden	If the load is already assigned to a boat, then a 403 is returned.
Failure	404 Not Found	If the load_id or boat_id are not associated with an entity, then return 404.

#### Response Examples

##### Success

Status: 204 No Content

##### Failure

Status: 403 Bad Request

```
{
  "Error": "The load is already assigned to a boat"
}
```

Status: 404 Not Found

```
{
  "Error": "The specified boat and/or load don't exist"
}
```

## GET LOADS CARRIED BY A BOAT

### GET /boats/:boat\_id/loads

Function: View all loads on a boat

#### Request

##### Request Parameters:

Name	Type	Description	Required?
boat_id	String	ID of the boat to get loads from	Yes

#### Request Body:

None

#### Response

##### Response Body Format

JSON

#### Response Statuses

Outcome	Status	Notes
Success	200 OK	
Failure	404 Not Found	No boat with this ID exists.

#### Response Examples

##### Success

##### Status: 200 OK

```
[
  {
    "content": "LEGO Blocks",
    "delivery_date": "01/05/2021",
    "id": "5681150794137600",
    "self": "https://trompj-assignment-4.appspot.com/loads/5681150794137600",
    "weight": 10
  },
  {
    "content": "LEGO Blocks",
    "delivery_date": "01/05/2021",
    "id": "5655374346584064",
    "self": "https://trompj-assignment-4.appspot.com/loads/5655374346584064",
    "weight": 10
  }
]
```

##### Failure

##### Status: 404 Not Found

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