

Marina REST API

CS 493: Cloud Application Development

URLs	2
Data Model	2
Get Users.....	3
Create a Boat	4
Get Boats.....	6
Get a Boat	8
Delete a Boat.....	10
Create a Load	12
Get Loads	14
Get a Load	16
Delete a Load	18
Load Assigned to Boat.....	20
Load Removed from Boat	22

URLs

API: <https://kp-marina-api.uw.r.appspot.com>

Account creation and login/logout: <https://kp-marina-api.uw.r.appspot.com>

Data Model

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

Users

Property	Data Type	Required?	Notes/Requirements
sub_id	String	Yes	A unique identifier that ties the user to a JWT. This value is automatically generated when a user is authenticated.

Boats

Property	Data Type	Required?	Notes/Requirements
id	Integer	Yes	The id of the boat. Datastore automatically generates it. Don't add it yourself as a property of the entity.
name	String	Yes	Name of the boat. Name must be a string 2 – 40 characters long and can only contain alphanumeric and space characters.
type	String	Yes	Type of the boat. E.g., Sailboat, Catamaran, etc. Type must be a string 2 – 20 characters long and can only contain alphanumeric and space characters.
length	Number	Yes	The length of the boat in feet. Length must be a number between 1 – 9999, inclusive.
owner	JSON Object	Yes	The sub_id of the user who created the boat. This is automatically handled using the JWT linked to the user.
loads	JSON Array	No	Each object in array contains a load's id. Empty array if no loads on boat.

Loads

Property	Data Type	Required?	Notes/Requirements
id	Integer	Yes	The id of the load. Datastore automatically generates it. Don't add it yourself as a property of the entity.
volume	Integer	Yes	The volume of the load in cubic feet. Volume must be a number between 1 – 9999, inclusive.
item	String	Yes	Describes the items contained in the load. Must be a string 2 – 20 characters long and can only contain alphanumeric and space characters.
weight	Integer	Yes	The weight of the load in pounds. Weight must be a number between 1 – 9999, inclusive.
boat	JSON Object	No	If the load is on a boat, then the property contains the carrying boat's id. Otherwise, boat is null.

Get Users

Allows you to get all existing users.

GET /users

Accept Header

application/json

Request

Path Parameters

None

Request Body

None

Response

Content Type

application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	406 Not Acceptable	Requested MIME type not supported by endpoint

Response Examples

Success

Status: 200 OK

Content-Type: application/json

```
{
  "users": [
    {"sub_id": "auth0|456", "self": "https://appspot.com/users/auth0|456"},
    {"sub_id": "auth0|001", "self": "https://appspot.com/users/auth0|001"}
  ]
}
```

Failure

Status: 406 Not Acceptable

Content-Type: application/json

```
{
  "Error": "Requested MIME type not supported by endpoint"
}
```

Create a Boat

Allows you to create a new boat.

POST /boats

Accept Header

application/json

Request

Path Parameters

None

Request Body

Required

Content Type(s)

application/json

Request JSON Properties

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

Request Body Example

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

Response

Content Type(s)

application/json

Response Statuses

Outcome	Status Code	Notes
Success	201 Created	
Failure	400 Bad Request	The boat is not created if: <ul style="list-style-type: none">• The request is missing any of the 3 required properties• The request has extraneous properties• Property requirements are not followed
Failure	401 Unauthorized	Invalid JWT

Response Examples

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

Success

Status: 201 Created
Content-Type: application/json

```
{
  "id": 123456
  "name": "Sea Witch",
  "type": "Catamaran",
  "length": 28,
  "loads": [],
  "owner": {"id": "auth0|456", "self": "https://appspot.com/users/auth0|456"}
  "self": "https://appspot.com/boats/123456"
}
```

Failure

Status: 400 Bad Request
Content-Type: application/json

```
{
  "Error": "The request object does not fulfill the property requirements"
}
```

Status: 401 Unauthorized
Content-Type: application/json

```
{
  "Error": "The request has an invalid JWT"
}
```

Get Boats

Lists five boats per request, using cursor pagination, of all existing boats owned by user corresponding to request's associated JWT.

GET /boats

Accept Header

application/json

Request

Path Parameters

None

Request Body

None

Response

Content Type(s)

application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	Invalid JWT
Failure	406 Not Acceptable	Requested MIME type not supported by endpoint

Response Examples

- Self and next properties are automatically generated upon a request and are not stored in Datastore.

Success

Status: 200 OK

Content-Type: application/json

```
{
  "boats": [
    {
      "id": 123456
      "name": "Sea Witch"
      "type": "Catamaran"
      "length": 28
      "owner": { "id": "auth0|456", "self": "https://appspot.com/users/auth0|456" }
      "loads": []
      "self": "https://appspot.com/boats/123456"
    },
    {
      "id": 987654
      "name": "SS Fun"
    }
  ]
}
```

```
{
  "type": "Yacht"
  "length": 100
  "owner": {"id": "auth0|001", "self": "https://appspot.com/users/auth0|001" }
  "loads": [
    {"id": 101, "self": "https://appspot.com/loads/101"},
    {"id": 202, "self": "https://appspot.com/loads/202"}
  ]
  "self": "https://appspot.com/boats/987654"
}
],
"total": 2
}
```

Failure

Status: 401 Unauthorized
Content-Type: application/json

```
{
  "Error": "The request has an invalid JWT"
}
```

Status: 406 Not Acceptable
Content-Type: application/json

```
{
  "Error": "Requested MIME type not supported by endpoint"
}
```

Get a Boat

Allows you to get an existing boat.

GET /boats/:boat_id

Accept Header

application/json

Request

Path Parameters

Name	Description
boat_id	ID of the boat

Request Body

None

Response

Content Type(s)

application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	Invalid JWT
Failure	403 Forbidden	User does not own boat with this boat_id
Failure	404 Not Found	No boat with this boat_id exists
Failure	406 Not Acceptable	Requested MIME type not supported by endpoint

Response Examples

- Self property is automatically generated upon a request and is not stored in Datastore.

Success

Status: 200 OK

Content-Type: application/json

```
{
  "id": 123456,
  "name": "Sea Witch",
  "type": "Catamaran",
  "length": 28,
  "owner": {"id": "auth0|456", "self": "https://appspot.com/users/auth0|456"},
  "loads": []
  "self": "https://appspot.com/boats/123456"
}
```


Failure

Status: 401 Unauthorized
Content-Type: application/json

```
{  
  "Error": "The request has an invalid JWT"  
}
```

Status: 403 Forbidden
Content-Type: application/json

```
{  
  "Error": "User does not own boat with this boat_id"  
}
```

Status: 404 Not Found
Content-Type: application/json

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

Status: 406 Not Acceptable
Content-Type: application/json

```
{  
  "Error": "Requested MIME type not supported by endpoint"  
}
```

Delete a Boat

Allows you to delete a boat. Note that if there are any loads on the boat, deleting the boat unloads them all.

DELETE /boats/:boat_id

Accept Header

application/json

Request

Path Parameters

Name	Description
boat_id	ID of the boat

Request Body

None

Response

Content Type(s)

Success: No body

Failure: application/json

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	401 Unauthorized	Invalid JWT
Failure	403 Forbidden	User does not own boat with this boat_id
Failure	404 Not Found	No boat with this boat_id exists
Failure	405 Method Not Allowed	No boat_id included in URL. Response's Allow header set to allowed methods for root boat URL, i.e. POST

Response Examples

Success

Status: 204 No Content

Failure

Status: 401 Unauthorized

Content-Type: application/json

```
{
  "Error": "The request has an invalid JWT"
}
```

Status: 403 Forbidden
Content-Type: application/json

```
{  
  "Error": "User does not own boat with this boat_id"  
}
```

Status: 404 Not Found
Content-Type: application/json

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

Status: 405 Method Not Allowed
Content-Type: application/json
Allow: GET, POST

```
{  
  "Error": "Invalid method"  
}
```

Create a Load

Allows you to create a new load.

POST /loads

Request

Path Parameters

None

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Property	Notes	Required?
volume	The volume of the load.	Yes
item	Describes the items contained in the load.	Yes
weight	The weight of the load.	Yes

Request Body Example

```
{
  "volume": 15,
  "item": "LEGO Blocks",
  "weight": 1000
}
```

Response

Response Body Format

application/json

Response Statuses

Outcome	Status Code	Notes
Success	201 Created	
Failure	400 Bad Request	The load is not created if: <ul style="list-style-type: none">• The request is missing any of the 3 required properties• The request has extraneous properties• Property requirements are not followed
Failure	401 Unauthorized	Invalid JWT

Response Examples

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

Success

Status: 201 Created

Content-Type: application/json

```
{
  "id": 101,
  "volume": 15
  "item": "LEGO Blocks",
  "weight": 1000,
  "boat": null
  "self": "https://appspot.com/loads/101"
}
```

Failure

Status: 400 Bad Request

Content-Type: application/json

```
{
  "Error": "The request object does not fulfill the property requirements"
}
```

Status: 401 Unauthorized

Content-Type: application/json

```
{
  "Error": "The request has an invalid JWT"
}
```

Get Loads

Lists five loads per request, using cursor pagination, of all existing loads.

GET /loads

Request

Path Parameters

None

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	Invalid JWT
Failure	406 Not Acceptable	Requested MIME type not supported by endpoint

Response Examples

Success

Status: 200 OK

Content-Type: application/json

```
{
  "loads": [
    {
      "id": 101,
      "volume": 15,
      "boat": {"id": 987654, "self": "https://appspot.com/boats/987654"},
      "item": "LEGO Blocks",
      "weight": 1000,
      "self": "https://appspot.com/loads/101"
    },
    {
      "id": 202,
      "volume": 10,
      "boat": {"id": 987654, "self": "https://appspot.com/boats/987654"},
      "item": "Teddy Bears",
      "weight": 20,
      "self": "https://appspot.com/loads/202"
    },
    {
      "id": 303,
      "volume": 100,
      "boat": null,
      "item": "Bananas",

```

```
        "weight": 300
        "self": "https://appspot.com/loads/303"
      },
    ],
    "total": 3
  }
}
```

Failure

Status: 401 Unauthorized
Content-Type: application/json

```
{
  "Error": "The request has an invalid JWT"
}
```

Status: 406 Not Acceptable
Content-Type: application/json

```
{
  "Error": "Requested MIME type not supported by endpoint"
}
```

Get a Load

Allows you to get an existing load.

GET /loads/:load_id

Accept Header

application/json

Request

Path Parameters

Name	Description
load_id	ID of the load

Request Body

None

Response

Content Type(s)

application/json

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	Invalid JWT
Failure	404 Not Found	No load with this load_id exists
Failure	406 Not Acceptable	Requested MIME type not supported by endpoint

Response Examples

- Self property is automatically generated upon a request and is not stored in Datastore.

Success

Status: 200 OK

Content-Type: application/json

```
{
  "id": 101,
  "volume": 15,
  "boat": {"id": 987654, "self": "https://appspot.com/boats/987654"},
  "item": "LEGO Blocks",
  "weight": 1000
  "self": "https://appspot.com/loads/101"
}
```

Failure

Status: 401 Unauthorized

Content-Type: application/json

```
{
  "Error": "The request has an invalid JWT"
}
```


Status: 404 Not Found

Content-Type: application/json

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

Status: 406 Not Acceptable

Content-Type: application/json

```
{  
  "Error": "Requested MIME type not supported by endpoint"  
}
```

Delete a Load

Allows you to delete a load. If the load being deleted is on a boat, that boat's loads are updated accordingly.

DELETE /loads/:load_id

Request

Path Parameters

Name	Description
load_id	ID of the load

Request Body

None

Response

No body

Response Body Format

Success: No body

Failure: application/json

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	401 Unauthorized	Invalid JWT
Failure	404 Not Found	No load with this load_id exists
Failure	405 Method Not Allowed	No load_id included in URL. Response's Allow header set to allowed methods for root boat URL, i.e. POST

Response Examples

Success

Status: 204 No Content

Failure

Status: 401 Unauthorized
Content-Type: application/json

```
{  
  "Error": "The request has an invalid JWT"  
}
```

Status: 404 Not Found
Content-Type: application/json

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

Status: 405 Method Not Allowed
Content-Type: application/json
Allow: GET, POST

```
{  
  "Error": "Invalid method"  
}
```

Load Assigned to Boat

A load is assigned to a boat.

PUT /boats/:boat_id/loads/:load_id

Request

Path Parameters

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

Request Body

None

Note: Set Content-Length to 0 in your request when calling to this endpoint.

Response

No body

Response Body Format

Success: No body

Failure: application/json

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	401 Unauthorized	Invalid JWT
Failure	403 Forbidden	User does not own boat with this boat_id
Failure	404 Not Found	No boat with this boat_id exists and/or no load with this load_id exists, or the load is already on another boat.

Response Examples

Success

Status: 204 No Content

Failure

Status: 401 Unauthorized
Content-Type: application/json

```
{
  "Error": "The request has an invalid JWT"
}
```

Status: 403 Forbidden
Content-Type: application/json

```
{
  "Error": "User does not own boat with this boat_id"
}
```

Status: 404 Not Found

Content-Type: application/json

```
{  
  "Error": "The specified boat and/or load does not exist. The load may also already  
be on another boat"  
}
```

Load Removed from Boat

A load is removed from a boat.

DELETE /boats/:boat_id/loads/:load_id

Request

Path Parameters

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

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	Invalid JWT
Failure	403 Forbidden	User does not own boat with this boat_id
Failure	404 Not Found	No load with this load_id is on this boat with this boat_id. This could be because no boat with this boat_id exists, or because no load with this load_id exists, or even if both boat_id and load_id are valid, the load with this load_id is not on this boat with this boat_id.

Response Examples

Success

Status: 204 No Content

Failure

Status: 401 Unauthorized

Content-Type: application/json

```
{
  "Error": "The request has an invalid JWT"
}
```

Status: 403 Forbidden

Content-Type: application/json

```
{  
  "Error": "User does not own boat with this boat_id"  
}
```

Status: 404 Not Found

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