Final Project: API Spec

CS 493: Cloud Application Development

Oregon State University - Fall 2019

Alexander Drath (dratha)

URL: https://final-project-dratha.appspot.com

How to obtain JWT:

Step 1: Go to https://final-project-dratha.appspot.com/

Step 2: Click on here on the page

Step 3: Select the email account you wish to use

Step 4: Response will appear as such:

Your user_id is: 109851309851526290975

Your JWT is:

eyJhbGciOiJSUzI1NiIsImtpZCI6IjViNWRkOWJlNDBiNWUxY2YxMjFlMzU3M2M4ZTQ5ZjEyNTI3MTgzZDMiLCJ0eXAiO
iJKV1QifQ.eyJpc3MiOiJhY2NvdW50cy5nb29nbGUuY29tIiwiYXpwIjoiNDAxNTA1MDMxNTg0LWZnZZ2E2dHJtbzJydml1a2
U4MWdlNTF1NWhubjVhNzJiLmFwcHMuZ29vZ2xldXNlcmNvbnRlbnQuY29tIiwiYXVkIjoiNDAxNTA1MDMxNTg0LWZn
Z2E2dHJtbzJydml1a2U4MWdlNTF1NWhubjVhNzJiLmFwcHMuZ29vZ2xldXNlcmNvbnRlbnQuY29tIiwic3ViIjoiMTA5ODU
xMzA5ODUxNTI2MjkwOTc1IiwiaGQiOiJvcmVnb25zdGF0ZS5lZHUiLCJlbWFpbCI6ImRyYXRoYUBvcmVnb25zdGF0ZS5l
ZHUiLCJlbWFpbF92ZXJpZmllZCI6dHJ1ZSwiYXRfaGFzaCI6Ikp2WFlsajZSQ2g2M1ItZXZFWXp4NEEiLCJpYXQiOjE1Nz
U5NDIzNDUsImV4cCI6MTU3NTk0NTk0NX0.cvc060O29A2T6YDL-
DdgVjQ1HnfBasXRmHDOp4Z2tNehOndmsj8oux5h62KtYtfukFNdO93UA9W2--ZP_7ZzTSlV44o2MUdkVuBjC-
lbYx98FC8pGm9GJzOt2DmC4AEdor9OUskQ15CfoblH6cCJkpXKmWVyKnK-o9dYx2MB5-
EGU17mfHLOpz_7Cz8CSvcBV9fU1fqb1yFaFiATZezlF0NGnTehmi-AKqn5hUm6CptwK6FdnyV8H4lFI391-
rk9A02jCGcJ8gp1Srncah69S8HEACREmSvJrdp2scTJScoqweSsG7glyIvNMtjiNNCZoD2RjKPbZIOGM5LFYCHaKA

*__Disclaimer__: There is an issue where sometimes the state is not equal in request and response (oauthlib.oauth2.rfc6749.errors.MismatchingStateError: (mismatching_state) CSRF Warning! State not equal in request and response."). This generates a 500 error that Professor Chaudhry and I have not been able to fix. If you get a 500 error, please repeat the steps above again to generate the JWT.
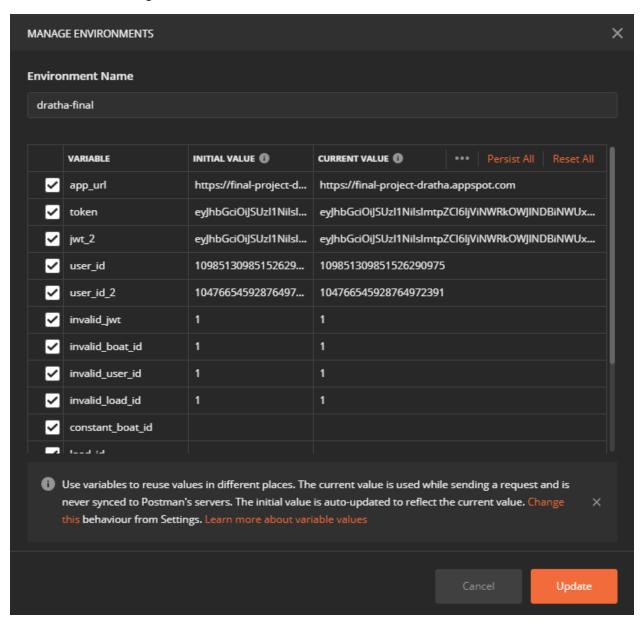
Environmental variables need to be set to:

There are 4 environment variables that need to be set. You will need to get 2 JWTs from two separate email accounts in order to set the required variables.

1) **Token** will be the first JWT you generate for one email account. The JWT will appear on the screen when when you go through the steps to obtain a JWT.
2) **user_id** will be the "sub" value of the first JWT account. The user_id will appear above your JWT when the JWT is generated and can be copy and pasted into the variable.
3) **jwt_2** will be the second JWT you will generate from another email account.

4) **user_id_2** will be the "sub" value for the second JWT account.
5) **invalid_jwt**, **invalid_boat_id**, **invalid_user_id**, and **invalid_load_id** should all be set to 1 already but if for some reason they are not, please enter 1 into them.

Please refer to the image below to make sure it looks the same.

	VARIABLE	INITIAL VALUE ⓘ	CURRENT VALUE ⓘ	•••	Persist All	Reset All
☑	app_url	https://final-project-d...	https://final-project-dratha.appspot.com			
☑	token	eyJhbGciOiJSUzl1Nilsl...	eyJhbGciOiJSUzl1NilsImtpZCI6IjViNWRkOWJINDBiNWUx...			
☑	jwt_2	eyJhbGciOiJSUzl1Nilsl...	eyJhbGciOiJSUzl1NilsImtpZCI6IjViNWRkOWJINDBiNWUx...			
☑	user_id	10985130985152629...	109851309851526290975			
☑	user_id_2	10476654592876497...	104766545928764972391			
☑	invalid_jwt	1	1			
☑	invalid_boat_id	1	1			
☑	invalid_user_id	1	1			
☑	invalid_load_id	1	1			
☑	constant_boat_id					
☑	load_id					

ⓘ Use variables to reuse values in different places. The current value is used while sending a request and is never synced to Postman's servers. The initial value is auto-updated to reflect the current value. Change this behaviour from Settings. Learn more about variable values ✕

Cancel   Update

List all Boats

Allows you to get an existing boat

GET /boats

Request

Request Parameters

None

Request Body

None

Response

Response Body Format

JSON (application/json)

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	406 Not Found	The request type is neither application/json or */*

Response Examples

Success

./boats	Status: 200
	{
	    "boats": [
	        {
	            "type": "Yatch",
	            "loads": [],
	            "length": 99,
	            "name": "Odyssey",
	            "id": "5740703418155008",
	            "self": "http://\<your app>.appspot.com/boats/5740703418155008"
	        }
	    ]
	}
	
	Or
	{
	    "boats": [
	        {
	            "type": "Yatch",
	            "loads": [],
	            "length": 99,
	            "name": "Odyssey",
	            "id": "5684099876388864",
	            "self": "http://\<your-app>.appspot.com/boats/5684099876388864"
	        },
	        {
	            "type": "Yatch",

```
                    "loads": [],
                    "length": 99,
                    "name": "Odyssey",
                    "id": "5738600293466112",
                    "self": "http:// ://<your-
app>.appspot.com/boats/5738600293466112"
            },
…
…
            {
                    "type": "Yatch",
                    "loads": [],
                    "length": 99,
                    "name": "Odyssey",
                    "id": "5740703418155008",
                    "self": "http:// ://<your-
app>.appspot.com/boats/5740703418155008"
            }
        ],
        "next": "http:// ://<your-
app>.appspot.com/boats?cursor=Ci4SKGoSc353NC1ub2RlanMtZHJhdG
hhchILEgVCb2F0cxiAgIDguqSZCgwYACAA"
}
```

Failure

./boats
Accept: application/xml
	Status: 406
	{
	"Error": "The request mimetype should be application/json
	}

Create a Boat

Allows you to create a new boat.

POST /boats

Request

Request Parameters

None

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Accepted Input	Required?

name	String	The name of the boat.	A-Z, a-z, 0-9, _ . Cannot have two __ in a row and cannot start with a _ Min # of characters = 3, Max # of characters = 25.	Yes
type	String	The type of the boat. E.g., Sailboat, Catamaran, etc.	A-Z, a-z, 0-9, _ . Cannot have two __ in a row and cannot start with a _ Min # of characters = 3, Max # of characters = 25.	Yes
length	Integer	Length of the boat in feet.	Min = 5 Max = 1000000	Yes

Request Body Example

```
{
  "name": "Odyssey",
  "type": "Yacht",
  "length": 99
}
```

Response

Response Body Format

JSON

Content-Location: http://<your-app>/boats/:boat_id

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 must not be created, and 400 status code must be returned. If the required attributes contain improper user input characters.
Failure	403 Forbidden	Same name as another boat
Failure	405 Method Not Allowed	PUT, PATCH, DELETE request on root url
Failure	406 Not Acceptable	Request a response type besides 'application/json'

Response Examples

- Datastore will automatically generate an ID and store it with the entity being created.
- The `self` attribute will contain the live link to the REST resource corresponding to this boat.

Success

```{ "name": "Odyssey", "type": "Yacht", "length": 99 }```	Status: 201 Created ```{ "name": "Odyssey", "type": "Yacht", "loads": [], "length": 99```

	"id": 5741031244955648
	"self": http://<your-app>/boats/5741031244955648
	}
	Content-Location: http://<your-app>/boats/5741031244955648

{ "name": "Odyssey1", "type": "Yacht" }	Status: 400 Bad Request
	{
	"Error": "The request object is missing at least one of the required attributes"
	}

{ "name": 100, "type": "Yacht", "length": 99 }	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{ "name": "Odyssey1", "type": true, "length": 99 }	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{ "name": "Odyssey1", "type": "Yacht", "length": "2.2" }	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{ "name": "Od", "type": "Yacht", "length": 99 }	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{ "name": "GreaterThan25Characte rsLong", "type": "Yacht", "length": 99 }	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{ "name": "Odyssey1", "type": " GreaterThan25Character sLong ", "length": 99 }	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{ "name": "Odyssey1", "type": "Ya",	Status: 400 Bad Request {

"length": 99 }	"Error": "The request object contains inappropriate attribut es or attribute input" }

{
 "name": "Odyssey1",
 "type": "Yacht",
 "length": 4
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{
 "name": "Odyssey1",
 "type": "Yacht",
 "length": 1000001
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{
 "name": "Odyssey",
 "type": "Yacht",
 "length": 99
 "id": 1
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{
 "name": "Odyssey",
 "type": "Yacht",
 "length": 99
}
	Status: 400 Bad Request
	{
	"The request mimetype should be application/json"
	}

{
 "name": "Odyssey",
 "type": "Yacht",
 "length": 99
}
	Status: 403
	{
	"Error": "The requested boat name is already in use"
	}

{
 "name": "Odyssey",
 "type": "Yacht",
 "length": 99
}
	Status: 406
	{
	"Error": "Accept header must be application/json"
	}

Get a Boat

Allows you to get an existing boat

GET /boats/:boat_id

Request

Request Parameters

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

Request Body

None

Response

Response Body Format

JSON (application/json)

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	404 Not Found	No boat with this boat_id exists
Failure	406 Not Found	The request type is neither application/json or text/html

Response Examples

Success

./boats/:boat_id
Accept: application/json
	Status: 200 OK
	{
	    "type": "Yatch",
	    "loads": [],
	    "length": 99,
	    "name": "Odyssey",
	    "id": "5740703418155008",
	    "self": "http://<your app>.appspot.com/boats/5740703418155008"
	}

Failure

./boats/invalid_boat_id	Status: 404
	{
	"Error": "No boat with this boat_id exists"
	}

./boats/:boat_id
Accept: application/xml
	Status: 406
	{
	"Error": "The request mimetype should be application/json or text/html"
	}

Edit a Boat

Allows you to edit a boat.

PUT /boats/:boat_id

Request

Request Parameters

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

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Accepted Input	Required?
name	String	The name of the boat.	A-Z, a-z, 0-9, _ . Cannot have two __ in a row and cannot start with a _ Min # of characters = 3, Max # of characters = 25.	Yes
type	String	The type of the boat. E.g., Sailboat, Catamaran, etc.	A-Z, a-z, 0-9, _ . Cannot have two __ in a row and cannot start with a _ Min # of characters = 3, Max # of characters = 25.	Yes
length	Integer	Length of the boat in feet.	Min = 5 Max = 1000000	Yes

Request Body Example

```
{
  "name": "Sea Witch",
  "type": "Catamaran",
  "length": 99
}
```

Response

Response Body Format

JSON

Location: http://<your-app>/boats/:boat_id

Response Statuses

Outcome	Status Code	Notes
Success	200 Success	If request is successful, the Location will update to the url you can find the updated boat on.

Failure	400 Bad Request	If the request is missing any of the 3 required attributes, the boat must not be created, and 400 status code must be returned. If the required attributes contain improper user input characters.
Failure	403 Forbidden	Same name as another boat
Failure	405 Method Not Allowed	GET, PUT, PATCH, DELETE request on root url
Failure	406 Not Acceptable	Request a response type besides 'application/json'

Response Examples

- The `self` attribute will contain the live link to the REST resource corresponding to this boat. You must not store this attribute in Datastore.

Success

``
{
  "name": Batman,
  "type": "Yacht",
  "length": 99
}
``
	Status: 200 Success
	``
	{
	    "type": "Yacht",
	    "loads": [],
	    "length": 99,
	    "name": "Batman",
	    "id": "5740703418155008",
	    "self": "http://<your
	app>.appspot.com/boats/5740703418155008"
	}
	``

Failure

``
{
  "name": "Odyssey1",
  "type": "Yacht"
}
``
	Status: 400 Bad Request
	``
	{
	"Error": "The request object is missing at least one of the
	required attributes"
	}
	``

``
{
  "name": "Odyssey1",
}
``
	Status: 400 Bad Request
	``
	{
	"Error": "The request object is missing at least one of the
	required attributes"
	}
	``

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

``
{
  "name": 100,
  "type": "Yacht",
  "length": 99
}
``
	Status: 400 Bad Request
	``
	{
	"Error": "The request object contains inappropriate attribut
	es or attribute input"
	}
	``

``
{
  "name": "Odyssey1",
  "type": true,
``
	Status: 400 Bad Request
	``
	{
	"Error": "The request object contains inappropriate attribut
	es or attribute input"
	``

"length": 99 }	}

{
  "name": "Odyssey1",
  "type": "Yacht",
  "length": "2.2"
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{
  "name": "Od",
  "type": "Yacht",
  "length": 99
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{
  "name":
"GreaterThan25Charact ersLong",
  "type": "Yacht",
  "length": 99
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{
  "name": "Odyssey1",
  "type": "
GreaterThan25Characte rsLong ",
  "length": 99
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{
  "name": "Odyssey1",
  "type": "Ya",
  "length": 99
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{
  "name": "Odyssey1",
  "type": "Yacht",
  "length": 4
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{
  "name": "Odyssey1",
  "type": "Yacht",
  "length": 1000001
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

{
  "name": "Odyssey",
  "type": "Yacht",
  "length": 99
  "id": 1
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut es or attribute input"
	}

}	

{
  "name": "Odyssey",
  "type": "Yacht",
  "length": 99
}
	Status: 400 Bad Request
	`{`
	`"The request mimetype should be application/json"`
	`}`

{
  "name": "Odyssey",
  "type": "Yacht",
  "length": 99
}
	Status: 403
	`{`
	`"Error": "The requested boat name is already in use"`
	`}`

./:invalid_boat_id
{
  "name": "Odyssey",
  "type": "Yacht",
  "length": 99
}
	Status: 404
	
	`{`
	`"Error": "No boat with this boat_id exists"`
	`}`

PUT /boats
{
  "name": "Odyssey",
  "type": "Yacht",
  "length": 99
}
	Status: 405
	
	`{`
	`"Error": "Method not recognized"`
	`}`

GET /boats
{
  "name": "Odyssey",
  "type": "Yacht",
  "length": 99
}
	Status: 405
	
	`{`
	`"Error": "Method not recognized"`
	`}`

PATCH /boats
{
  "name": "Odyssey",
  "type": "Yacht",
  "length": 99
}
	Status: 405
	
	`{`
	`"Error": "Method not recognized"`
	`}`

DELETE /boats
{
  "name": "Odyssey",
  "type": "Yacht",
  "length": 99
}
	Status: 405
	
	`{`
	`"Error": "Method not recognized"`
	`}`

{
  "name": "Odyssey",
  "type": "Yacht",
  "length": 99
}
	Status: 406
	`{`
	`"Error": "Accept header must be application/json"`
	`}`

Valid JWT but not user entity created	Status: 401
	{
	"Error": "Not a valid user, please register"
	}

User2 tries to interact with boat that was created by user1	Status: 403
	{
	"Error": "You are not the owner of this boat"
	}

Update a Boat Attribute

Allows you to edit a boat.

PATCH /boats/:boat_id

Request

Request Parameters

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

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Accepted Input	Required?
name	String	The name of the boat.	A-Z, a-z, 0-9, _ . Cannot have two __ in a row and cannot start with a _ Min # of characters = 3, Max # of characters = 25.	Yes
type	String	The type of the boat. E.g., Sailboat, Catamaran, etc.	A-Z, a-z, 0-9, _ . Cannot have two __ in a row and cannot start with a _ Min # of characters = 3, Max # of characters = 25.	Yes
length	Integer	Length of the boat in feet.	Min = 5 Max = 1000000	Yes

Request Body Example

```
{
   "name": "Sea Witch",
   "type": "Catamaran",
   "length": 99
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	If request is successful, a JSON response will be returned.
Failure	400 Bad Request	If the request is missing any of the 3 required attributes, the boat must not be created, and 400 status code must be returned.

		If the required attributes contain improper user input characters.
Failure	403 Forbidden	Same name as another boat
Failure	406 Not Acceptable	Request a response type besides 'application/json'

Response Examples

- The `self` attribute will contain the live link to the REST resource corresponding to this boat. You must not store this attribute in Datastore.

Success

``
{
  "name": "Odyssey",
  "type": "Yacht",
  "length": 99
}
``
	Status: 200 OK
	``
	{
	    "type": "Yacht",
	    "loads": [],
	    "length": 99,
	    "name": "Odyssey",
	    "id": "5740703418155008",
	    "self": "http://<your
	app>.appspot.com/boats/5740703418155008"
	}
	``

``
{
  "name": "Odyssey1",
  "type": "Freighter"
}
``
	Status: 200 OK
	``
	{
	    "type": "Freighter",
	    "loads": [],
	    "length": 99,
	    "name": "Odyssey1",
	    "id": "5740703418155008",
	    "self": "http://<your
	app>.appspot.com/boats/5740703418155008"
	}
	``

``
{
  "name": "Odyssey1",
}
``
	Status: 200 OK
	``
	{
	    "type": "Yacht",
	    "loads": [],
	    "length": 99,
	    "name": "Odyssey1",
	    "id": "5740703418155008",
	    "self": "http://<your
	app>.appspot.com/boats/5740703418155008"
	}
	``

Failure

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

``
{
  "name": 100,
  "type": "Yacht",
  "length": 99
}
``
	Status: 400 Bad Request
	``
	{
	"Error": "The request object contains inappropriate attributes or attribute input"
	}
	``

{
  "name": "Odyssey1",
  "type": true,
  "length": 99
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut
	es or attribute input"
	}

{
  "name": "Odyssey1",
  "type": "Yacht",
  "length": "2.2"
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut
	es or attribute input"
	}

{
  "name": "Od",
  "type": "Yacht",
  "length": 99
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut
	es or attribute input"
	}

{
  "name":
"GreaterThan25Charact
ersLong",
  "type": "Yacht",
  "length": 99
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut
	es or attribute input"
	}

{
  "name": "Odyssey1",
  "type": "
GreaterThan25Characte
rsLong ",
  "length": 99
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut
	es or attribute input"
	}

{
  "name": "Odyssey1",
  "type": "Ya",
  "length": 99
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut
	es or attribute input"
	}

{
  "name": "Odyssey1",
  "type": "Yacht",
  "length": 4
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut
	es or attribute input"
	}

{
  "name": "Odyssey1",
  "type": "Yacht",
  "length": 1000001
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attribut
	es or attribute input"
	}

{ "name": "Odyssey",	Status: 400 Bad Request {

"type": "Yacht", "length": 99 "id": 1 }	"Error": "The request object contains inappropriate attribut es or attribute input" }

{
"name": "Odyssey",
"type": "Yacht",
"length": 99
}
	Status: 400 Bad Request
	{
	"The request mimetype should be application/json"
	}

{
"name": "Odyssey",
"type": "Yacht",
"length": 99
}
	Status: 403
	{
	"Error": "The requested boat name is already in use"
	}

./:invalid_boat_id
{
"name": "Odyssey",
"type": "Yacht",
"length": 99
}
	Status: 404
	
	{
	"Error": "No boat with this boat_id exists"
	}

{
"name": "Odyssey",
"type": "Yacht",
"length": 99
}
	Status: 406
	{
	"Error": "Accept header must be application/json"
	}

Valid JWT but not user entity created	Status: 401
	{
	"Error": "Not a valid user, please register"
	}

User2 tries to interact with boat that was created by user1	Status: 403
	{
	"Error": "You are not the owner of this boat"
	}

Delete a Boat

Allows you to delete a boat. Note that if the boat is currently in a slip, deleting the boat makes the slip empty.

DELETE /boats/:boat_id

Request

Request Parameters

None

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 boat with this boat_id exists
Failure	403	Not the owner of the boat
Failure	401	Not a valid user

Response Examples

Success

./boats/:boat_id	Status: 204 No Content

Failure

./:invalid_boat_id
{
  "name": "Odyssey",
  "type": "Yacht",
  "length": 99
}
	Status: 404
	
	{
	"Error": "No boat with this boat_id exists"
	}

Valid JWT but not user entity created	Status: 401
	{
	"Error": "Not a valid user, please register"
	}

User2 tries to interact with boat that was created by user1	Status: 403
	{
	"Error": "You are not the owner of this boat"
	}

List all Loads

Allows you to get an existing boat

GET /loads

Request

Request Parameters

None

Request Body

None

Response

Response Body Format

JSON (application/json)

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	406 Not Found	The request type is neither application/json or */*

Response Examples

Success

./loads	Status: 200
	{
	    "loads": [
	        {
	            "carrier": null,
	            "content": "LEGO_Blocks",
	            "weight": 5,
	            "delivery_date": "2020-01-01",
	            "id": "5761068945113088",
	            "self": "http://w4-nodejs-dratha.appspot.com/loads/5761068945113088"
	        }
	    ]
	}
	
	Or
	{
	    "loads": [
	        {
	            "carrier": null,
	            "content": "LEGO_Blocks",
	            "weight": 5,
	            "delivery_date": "2020-01-01",
	            "id": "5761068945113088",
	            "self": "http://w4-nodejs-dratha.appspot.com/loads/5761068945113088"
	        },
	        {
	            "carrier": null,
	            "content": "LEGO_Blocks",
	            "weight": 5,

```
                    "delivery_date": "2020-01-01",
                    "id": "5761068945113088",
                    "self": "http://w4-nodejs-
dratha.appspot.com/loads/5761068945113088"
                },
…
…
            {
                "carrier": null,
                "content": "LEGO_Blocks",
                "weight": 5,
                "delivery_date": "2020-01-01",
                "id": "5761068945113088",
                "self": "http://w4-nodejs-
dratha.appspot.com/loads/5761068945113088"
            }
        ],
        "next": "http:// ://<your-
app>.appspot.com/loads?cursor=Ci4SKGoSc353NC1ub2RlanMtZHJhdG
hhchILEgVCb2F0cxiAgIDguqSZCgwYACAA"
}
```

Failure

./loads
Accept: application/xml
	Status: 406
	{
	"Error": "The request mimetype should be application/json
	}

Create a Load

Allows you to create a new boat.

POST /loads

Request

Request Parameters

None

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Accepted Input	Required?
name	String	The name of the boat.	A-Z, a-z, 0-9, _ . Cannot have two __ in a row and cannot start with a _	Yes

			Min # of characters = 3, Max # of characters = 25.	
delivery_date	String	The delivery date of the load.	"XXXX-XX-XX" Only numbers and – and must be in that format	Yes
weight	Integer	Weight of the load in lbs.	Min = 5 Max = 1000000	Yes

Request Body Example

```
{
  "contents": "LEGO_Blocks",
  "delivery_date": "2020-01-01",
  "weight": 5
}
```

Response

Response Body Format

JSON

Content-Location: http://<your-app>/boats/:boat_id

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 must not be created, and 400 status code must be returned. If the required attributes contain improper user input characters.
Failure	403 Forbidden	Same name as another boat
Failure	405 Method Not Allowed	PUT, PATCH, DELETE request on root url
Failure	406 Not Acceptable	Request a response type besides 'application/json'

Response Examples

- Datastore will automatically generate an ID and store it with the entity being created.
- The `self` attribute will contain the live link to the REST resource corresponding to this boat.

Success

```{ "contents": "LEGO_Blocks", "delivery_date": "2020-01-01", "weight": 5 }```	Status: 201 Created ```{ "carrier": null, "content": "LEGO_Blocks", "weight": 5, "delivery_date": "2020-01-01", "id": "5761068945113088", "self": "http://w4-nodejs-dratha.appspot.com/loads/5761068945113088" }```

Failure

{	Status: 400 Bad Request

"contents": "LEGO_Blocks",
  "weight": 5
}
	{
	"Error": "The request object is missing at least one of the required attributes"
	}

{
  "contents": "LEGO_Blocks",
  "delivery_date": "01-01-2020",
  "weight": 5
}
	Status: 400 Bad Request
	{
	"Error": "The request object contains inappropriate attributes or attribute input"
	}

Valid JWT but not user entity created	Status: 401
	{
	"Error": "Not a valid user, please register"
	}

Invalid_jwt	Status: 401
	{
	"Error": "Missing or Invalid JWT"
	}

Get a Load

Allows you to get an existing boat

GET /loads/:load_id

Request

Request Parameters

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

Request Body

None

Response

Response Body Format

JSON (application/json)

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	404 Not Found	No load with this load_id exists
Failure	406 Not Found	The request type is neither application/json or */*

Response Examples

Success

./loads/:load_id
Accept: application/json
	Status: 200 OK
	{
	    "carrier": null,
	    "content": "LEGO_Blocks",
	    "weight": 5,
	    "delivery_date": "2020-01-01",
	    "id": "5761068945113088",
	    "self": "http://w4-nodejs-
	dratha.appspot.com/loads/5761068945113088"
	    }

Failure

./loads/invalid_load_id	Status: 404
	{
	"Error": "No boat with this load_id exists"
	}

./loads/:load_id
Accept: application/xml
	Status: 406
	{
	"Error": "The request mimetype should be application/json or text/html"
	}

Edit a Load

Allows you to edit an existing load

PUT /loads/:load_id

Request

Request Parameters

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

Request Body

None

Response

Response Body Format

JSON (application/json)

Request JSON Attributes

Name	Type	Description	Accepted Input	Required?
name	String	The name of the boat.	A-Z, a-z, 0-9, _ . Cannot have two __ in a row and cannot start with a _ Min # of characters = 3, Max # of characters = 25.	Yes
delivery_date	String	The delivery date of the load.	"XXXX-XX-XX" Only numbers and – and must be in that format	Yes
weight	Integer	Weight of the load in lbs.	Min = 5 Max = 1000000	Yes

Request Body Example

```
{
  "contents": "LEGO_Blocks",
  "delivery_date": "2020-01-01",
  "weight": 5
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 Success	
Failure	400 Bad Request	If the request is missing any of the 3 required attributes, the boat must not be created, and 400 status code must be returned.

		If the required attributes contain improper user input characters.
Failure	401 Unauthorized	Not a user
Failure	405 Method Not Allowed	PUT, PATCH, DELETE request on root url
Failure	406 Not Acceptable	Request a response type besides 'application/json'

Response Examples

- The `self` attribute will contain the live link to the REST resource corresponding to this boat. You must not store this attribute in Datastore.

Success

``
{
   "contents":
"LEGO_Blocks",
   "delivery_date":
"2020-01-01",
   "weight": 5
}
``
	Status: 200 Success
	``
	{
	            "carrier": null,
	            "content": "LEGO_Blocks",
	            "weight": 5,
	            "delivery_date": "2020-01-01",
	            "id": "5761068945113088",
	            "self": "http://w4-nodejs-
	dratha.appspot.com/loads/5761068945113088"
	            }
	``

Failure

``
{
   "contents":
"LEGO_Blocks",
   "weight": 5
}
``
	Status: 400 Bad Request
	``
	{
	"Error": "The request object is missing at least one of the
	required attributes"
	}
	``

``
{
   "contents":
"LEGO_Blocks",
   "delivery_date":
"01-01-2020",
   "weight": 5
}
``
	Status: 400 Bad Request
	``
	{
	"Error": "The request object contains inappropriate attribute
	s or attribute input"
	}
	``

``
./:invalid_load_id
{
   "contents":
"LEGO_Blocks",
   "delivery_date":
"2020-01-01",
   "weight": 5
}
``
	Status: 404
	
	``
	{
	"Error": "No boat with this load_id exists"
	}
	``

``
PUT /loads
{
   "contents":
"LEGO_Blocks",
   "delivery_date":
"2020-01-01",
   "weight": 5
}
``
	Status: 405
	
	``
	{
	"Error": "Method not recognized"
	}
	``

``
{
   "contents":
"LEGO_Blocks",
``
	Status: 405
	
	``
	{
	``

"delivery_date": "2020-01-01", "weight": 5 }	"Error": "Method not recognized" }

Accept:
 application/xml
 {
   "contents": "LEGO_Blocks",
   "delivery_date": "2020-01-01",
   "weight": 5
 }
	Status: 406
	 {
	 "Error": "Accept header must be application/json"
	 }

Valid JWT but not user entity created	Status: 401
	 {
	 "Error": "Not a valid user, please register"
	 }

Update a Load Attribute

Allows you to edit a boat.

PATCH /loads/:load_id

Request

Request Parameters

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

Request Body

None

Response

Response Body Format

JSON (application/json)

Request JSON Attributes

Name	Type	Description	Accepted Input	Required?
name	String	The name of the boat.	A-Z, a-z, 0-9, _ . Cannot have two __ in a row and cannot start with a _ Min # of characters = 3, Max # of characters = 25.	Yes
type	String	The type of the boat. E.g., Sailboat, Catamaran, etc.	A-Z, a-z, 0-9, _ . Cannot have two __ in a row and cannot start with a _ Min # of characters = 3, Max # of characters = 25.	Yes
length	Integer	Length of the boat in feet.	Min = 5 Max = 1000000	Yes

Request Body Example

```
{
  "name": "Sea Witch",
  "type": "Catamaran",
  "length": 99
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	If request is successful, a JSON response will be returned.
Failure	400 Bad Request	If the request is missing any of the 3 required attributes, the boat must not be created, and 400 status code must be returned.

		If the required attributes contain improper user input characters.
Failure	401 Unauthorized	Not a user
Failure	406 Not Acceptable	Request a response type besides 'application/json'

Response Examples

- The `self` attribute will contain the live link to the REST resource corresponding to this boat. You must not store this attribute in Datastore.

Success

<table>
<tr>
<td>

```
{
   "contents":
"LEGO_Blocks",
   "delivery_date":
"2020-01-01",
   "weight": 5
}
```

</td>
<td>

Status: 200 Success
```
{
             "carrier": null,
             "content": "LEGO_Blocks",
             "weight": 5,
             "delivery_date": "2020-01-01",
             "id": "5761068945113088",
             "self": "http://w4-nodejs-
dratha.appspot.com/loads/5761068945113088"
          }
```

</td>
</tr>
<tr>
<td>

```
{
   "contents":
"LEGO_STACKS ",
   "delivery_date":
"2020-07-01"
}
```

</td>
<td>

Status: 200 Success
```
{
             "carrier": null,
             "content": "LEGO_STACKS",
             "weight": 5,
             "delivery_date": "2020-07-01",
             "id": "5761068945113088",
             "self": "http://w4-nodejs-
dratha.appspot.com/loads/5761068945113088"
          }
```

</td>
</tr>
<tr>
<td>

```
{
   "contents":
"LEGO_Blocks"
}
```

</td>
<td>

Status: 200 Success
```
{
             "carrier": null,
             "content": "LEGO_Blocks",
             "weight": 5,
             "delivery_date": "2020-07-01",
             "id": "5761068945113088",
             "self": "http://w4-nodejs-
dratha.appspot.com/loads/5761068945113088"
          }
```

</td>
</tr>
</table>

Failure

<table>
<tr>
<td>

```
{
   "contents":
"LEGO_Blocks",
   "delivery_date":
"01-01-2020",
   "weight": 5
}
```

</td>
<td>

Status: 400 Bad Request
```
{
"Error": "The request object contains inappropriate attribute
s or attribute input"
}
```

</td>
</tr>
<tr>
<td>

```
./:invalid_load_id
{
   "contents":
"LEGO_Blocks",
   "delivery_date":
"2020-01-01",
```

</td>
<td>

Status: 404

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

</td>
</tr>
</table>

`    "weight": 5` `}`	

PUT /loads
`{`
`   "contents":`
`"LEGO_Blocks",`
`   "delivery_date":`
`"2020-01-01",`
`   "weight": 5`
`}`
	Status: 405
	
	`{`
	`"Error": "Method not recognized"`
	`}`

`{`
`   "contents":`
`"LEGO_Blocks",`
`   "delivery_date":`
`"2020-01-01",`
`   "weight": 5`
`}`
	Status: 405
	
	`{`
	`"Error": "Method not recognized"`
	`}`

`Accept:`
`application/xml`
`{`
`   "contents":`
`"LEGO_Blocks",`
`   "delivery_date":`
`"2020-01-01",`
`   "weight": 5`
`}`
	Status: 406
	`{`
	`"Error": "Accept header must be application/json"`
	`}`

Valid JWT but not user
entity created
	Status: 401
	`{`
	`"Error": "Not a valid user, please register"`
	`}`

Delete a Boat

Allows you to delete a boat. Note that if the boat is currently in a slip, deleting the boat makes the slip empty.

DELETE /loads/:load_id

Request

Request Parameters

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

Request Body

None

Response

Response Body Format

JSON (application/json)

Request JSON Attributes

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	404 Not Found	No load with this load_id exists
Failure	403	Not the owner of the boat
Failure	401	Not a valid user

Response Examples

Success

./loads/:load_id	Status: 204 No Content

Failure

./:invalid_load_id	Status: 404
	{
	"Error": "No boat with this load_id exists"
	}

Valid JWT but not user entity created	Status: 401
	{
	"Error": "Not a valid user, please register"
	}

User2 tries to delete a load that is on a boat that was created by user1	Status: 403
	{
	"Error": "You are not the owner of this boat"
	}

Load a Load Attribute onto a Boat

Allows you to add a load onto a boat.

PATCH /loads/:load_id/boats/:boat_id

Request

Request Parameters

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

Request Body

None

Response

Response Body Format

JSON (application/json)

Request JSON Attributes

Request Body Example

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	If request is successful, a JSON response will be returned.
Failure	400 Bad Request	If the request is missing any of the 3 required attributes, the boat must not be created, and 400 status code must be returned. If the required attributes contain improper user input characters.
Failure	401 Unauthorized	Not a user
Failure	406 Not Acceptable	Request a response type besides 'application/json'

Response Examples

- The `self` attribute will contain the live link to the REST resource corresponding to this boat. You must not store this attribute in Datastore.

Success

load_id = 5756147248136192
boat_id = 5691035543928832
	Status: 200 Success
	`{`
	`"contents": "LEGO_Blocks",`
	`"carrier": {`
	`  "id": 5691035543928832,`
	`  "name": "Hades",`
	`  "self": `https://<your-app>.appspot.com/boats/5691035543928832
	`},`

	"weight": 5, "delivery_date": "2020-01-01", "self": "https://<your-app>.appspot.com/loads/5756147248136192"}

Valid JWT but not user entity created	Status: 401
	{
	"Error": "Not a valid user, please register"
	}

No JWT	Status: 401
	{
	"Error": "Missing or Invalid JWT"
	}

User2 tries to add the load to a boat that was created by user1	Status: 403
	{
	"Error": "You are not the owner of this boat"
	}

Try to add a load that is already on another boat	Status: 403
	{
	"Error": "Load currently has a carrier"
	}

/loads/:invalid_load_id/ boats/boat_id	Status: 404
	
	{
	"Error": "No boat with this load_id exists"
	}

/loads/:load_id/boats/:invalid_boat_id	Status: 404
	
	{
	"Error": "No boat with this boat_id exists"
	}

{
  "contents": "LEGO_Blocks",
  "delivery_date": "2020-01-01",
  "weight": 5
}
	Status: 405
	
	{
	"Error": "Method not recognized"
	}

Accept: application/xml	Status: 406
	{
	"Error": "Accept header must be application/json"
	}

Remove a Load Attribute from a Boat

Allows you to remove a load from a boat.

PATCH /loads/:load_id/boats/:boat_id

Request

Request Parameters

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

Request Body

None

Response

Response Body Format

JSON (application/json)

Request JSON Attributes

Request Body Example

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	404 Not Found	No load with this load_id exists
Failure	403	Not the owner of the boat
Failure	401	Not a valid user

Response Examples

Success

./loads/:load_id/boats/:boat_id	Status: 204 No Content

Failure

./loads/:invalid_load_id/boats/:boat_id	Status: 404
	{
	"Error": "No boat with this load_id exists"
	}

./loads/:load_id/boats/:invalid_boat_id	Status: 404
	{
	"Error": "No boat with this boat_id exists"
	}

Valid JWT but not user entity created	Status: 401
	{
	"Error": "Not a valid user, please register"
	}

No JWT	Status: 401

	{ "Error": "Missing or Invalid JWT" }

User2 tries to add the load to a boat that was created by user1	Status: 403
	{
	"Error": "You are not the owner of this boat"
	}

Try to remove a load that is not loaded onto a boat already	Status: 404
	{
	"Error": "The load is not loaded on a boat"
	}

Try to remove a load that is not loaded onto this boat_id already	Status: 404
	{
	"Error": "The load_id selected is not on this boat_id"
	}

List all the Users Boats

Allows you to get an existing boat

GET /users/:user_id/boats

Request

Request Parameters

None

Request Body

None

Response

Response Body Format

JSON (application/json)

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	
Failure	406 Not Found	The request type is neither application/json or */*

Response Examples

Success

./users/:user_id/boats	Status: 200
	{
	  "type": "Yatch",
	  "loads": [],
	  "length": 99,
	  "name": "Odyssey",
	  "id": "5684099876388864",

```
        "self": "http://<your-
app>.appspot.com/boats/5684099876388864"
        },
        {
            "type": "Yatch",
            "loads": [],
            "length": 99,
            "name": "Odyssey",
            "id": "5738600293466112",
            "self": "http:// ://<your-
app>.appspot.com/boats/5738600293466112"
        },
…
…
        {
            "type": "Yatch",
            "loads": [],
            "length": 99,
            "name": "Odyssey",
            "id": "5740703418155008",
            "self": "http:// ://<your-
app>.appspot.com/boats/5740703418155008"
        }
```

Failure

./boats
Accept: application/xml
	Status: 406
	{
	"Error": "The request mimetype should be application/json
	}

Valid JWT but not user entity created	Status: 401
	{
	"Error": "Not a valid user, please register"
	}

No JWT	Status: 401
	{
	"Error": "Missing or Invalid JWT"
	}

Trying the user_id trying to access match the JWT but the JWT is a valid user	Status: 401
	{
	"Error": "This is not your user_id"
	}