Final Project: API Spec

CS 493: Cloud Application Development

Spring 2021

Kyeongnam Kim (kimkyeon@oregonstate.edu)

Last Update: June 6, 2021. 5:00 pm Pacific Time

Change log

Version	**Change**	**Date**
1.0	Initial version.	June 6 2021

URL

https://boat-final-315122.wn.r.appspot.com
(home and account creation/login page)

Data Model

Non-user Entity

Boat (protected)

Property Name	Type	Required	Valid value
id	-	-	The id of the boat. Datastore automatically generates it.
name	String	Yes	The name of the boat.
type	String	Yes	The type of the boat.
length	Integer	Yes	The length of the boat in feet.
loads	Array[Object]	No	The loads in the boat. The loads property will contain the ids of the loads if the boat holds a load.
owner	String	Yes	The unique id of a user that owns the boat.

Load (unprotected)

Property Name	Type	Required	Valid value
id	-	-	The id of the load. Datastore automatically generates it.
volume	Integer	Yes	The volume of the load.
content	String	Yes	The content of the load.
creation_date	String	Yes	The date of creation of the load.
carrier	Object	No	The carrier property will contain id of the boat that the load is assigned to.
public	Boolean	Yes	The visibility setting of the load.

- A boat can hold 0..many loads.
- A load can be assigned to 0..1 boat.
- A boat entity is protected and a user with valid authentication/authorization can access to data.
- A load entity is not protected and a user without authentication/authorization can access to data.

User Entity

User

Property Name	Type	Required	Valid value
id	-	-	The id of the user. Datastore automatically generates it.
name	String	Yes	The full name of the user.
email	String	Yes	The email address of the user.
sub	String	Yes	The unique id of the user generated by authentication system.

- A unique identifier for a user is 'sub', one of registered claims, extracted from id token generated by Google OAuth 2.0. This application uses this unique identifier to assign a boat to a user as a property, owner.
- Every request that needs to supply a user identifier is required to include a Bearer token as an authorization.
- A tester needs to include a Bear token as jwt1 or jwt2 in postman environment setting. Id_token shown on the page after login is matched with jwt variables.
- A tester needs to use user_id1 and user_id2 to retrieve a load that has a relationship with a boat. Unique id(sub) on the page after login is matched with user_id variables.
- Users with valid authentication/authorization can create/read/edit/delete a boat. Any users can create/read/edit/delete a load.

Create a Boat

Protected

Allows you to create a new boat.

POST /boats

Request

Content-Type	application/json
Accept	application/json

Path 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	Length of the boat in feet.	Yes

Request Body Example

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

Response

Content-Type	application/json

Response Body Format

JS

Response Statuses

Outcome	Status Code	Notes
Success	201 Created	Boat is created.
Failure	401 Unauthorized	- Missing JWT - Invalid JWT
Failure	403 Forbidden	The name of the boat is already existed.
Failure	406 Not Acceptable	Accept in request header is not application/json. Server only sends application/json.
Failure	415 Unsupported Media Type	Content-Type in request header is not application/json. Server only accepts application/json.

Response Examples

- Datastore will automatically generate an ID and store it with the entity being created. And the response body will contain this generated ID along with other attributes.

- The `self` attribute will contain the live link to the REST resource corresponding to this boat. In other words, this is the URL to get this newly created boat. This attribute doesn’t be stored in Datastore.

Success

Status: 201 Created

```
{
  "id": "abc123",
  "name": "Sea Witch",
  "type": "Catamaran",
  "length": 28,
  “owner”: {{user identifier}}
  "self": "https://<your-app>/boats/abc123"
}
```

Failure

Status: 401 Unauthorized

```
{
    "Error": "Missing JWT"
}
```

```
{
    "Error": "Invalid JWT"
}
```

Status: 403 Forbidden

```
{
    "Error": "The name of boat is already existed"
}
```

Status: 406 Not Acceptable

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

Status: 415 Unsupported Media Type

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

View a Boat

Protected

Allows you to get an existing boat that you created

GET /boats/:boat_id

Request

Accept	application/json

Path Parameters

Name	Type	Description
boat_id	String	ID of the boat

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	- Missing JWT - Invalid JWT - Unauthorized user reads the data.
Failure	403 Forbidden	The name of the boat is already existed.
Failure	404 Not Found	No boat with this boat_id exists
Failure	406 Not Acceptable	Accept in request header is not application/json. Server only sends application/json.

Response Examples

Success

Status: 200 OK

```
{
  "id": "abc123",
  "name": "Sea Witch",
  "type": "Catamaran",
  "length": 28,
  “owner”: {{user identifier}}
  “loads”:[
    {
      “id”: “zzz123”,
      “self”: “https://<your-app>/loads/zzz123”
    },
    {
      “id”: “abc123”,
      “self”: “https://<your-app>/loads/abc123”
    }
  ],
  "self": "https://<your-app>/boats/abc123"
}
```

Failure

Status: 401 Unauthorized

```
{
    "Error": "Missing JWT"
}
```

```
{
    "Error": "Invalid JWT"
}
```

```
{
    "Error": "You are not allowed to access this entity"
}
```

Status: 403 Forbidden

```
{
    "Error": "The name of boat is already existed"
}
```

Status: 406 Not Acceptable

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

Status: 404 Not Found

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

View all Boats

Protected

View all the boats that you created.

GET /boats

Request

Accept	application/json

Parameters

Name	Type	Description
cursor	String	Cursor token of the next page

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	next property will be null if there are no more pages.
Failure	401 Unauthorized	- Missing JWT - Invalid JWT

Response Examples

Success

Status: 200 OK

```
{
    "boats": [
        {
            "id": "5665673409724416",
            "name": "Hail",
            "type": "Cuddy Cabin",
            "length": 150,
            "owner": 12039480129480,
            "self": "http://<your-app>/boats/5665673409724416"
        },
        {
            "id": "5673082664517632",
            "name": "Odyssey",
            "type": "Yatch",
            "length": 99,
            "owner": 12039480129480,
            "self": "http://<your-app>/boats/5673082664517632"
        },
        {
            "id": "5707651547660288",
            "name": "Shark",
            "type": "Deck Boat",
            "length": 30,
            "owner": 12039480129480,
            "self": "http://<your-app>/boats/5707651547660288"
        }
        {
            "id": "5665673409724416",
            "name": "ddd",
            "type": "Cuddy Cabin",
            "length": 150,
            "owner": 12039480129480,
            "self": "http://<your-app>/boats/5665673409724416"
        },
        {
            "id": "5665673409724416",
            "name": "Haigggl",
            "type": "Cuddy Cabin",
            "length": 150,
            "owner": 12039480129480,
            "self": "http://<your-app>/boats/5665673409724416"
        }
    ],

    "next": http://<your-app>/boats?cursor=Ci0SJ2oSc35ib2F0LWxvYWQtMzExNjA2chELEgRCb2F0GICAgPjC4pEKDBgAIAA=
    "count": 6,
}
```

Edit a part of Boat

Protected

Allows you to edit a part of a boat you created

```
PATCH /boats/:boat_id
```

Request

Content-Type	application/json
Accept	application/json

Path Parameters

Name	**Type**	**Description**
boat_id	String	ID of the boat

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	**Type**	**Description**	**Required?**
name	String	The name of the boat.	No*
type	String	The type of the boat. E.g., Sailboat, Catamaran, etc.	No*
length	Integer	Length of the boat in feet.	No*
* The number of attributes to be modified must be between 1 and 3.			

Request Body Example

```
{
  "length": 99
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	**Status Code**	**Notes**
Success	200 OK	Boat is updated
Failure	401 Unauthorized	- Missing JWT - Invalid JWT - Unauthorized user accesses the data.
Failure	403 Forbidden	The name of the boat is already existed.
Failure	406 Not Acceptable	Accept in request header is not application/json. Server only sends application/json.
Failure	415 Unsupported Media Type	Content-Type in request header is not application/json. Server only accepts application/json.

Response Examples

Success

```
Status: 200 OK
```

```
{
  "id": "abc123",
  "name": "Sea Witch",
  "type": "Catamaran",
  "length": 99,
  “owner”: {{user identifier}},
  "self": "https://<your-app>/boats/abc123"
}
```

Failure

Status: 401 Unauthorized

```
{
    "Error": "Missing JWT"
}
```

```
{
    "Error": "Invalid JWT"
}
```

```
{
    "Error": "You are not allowed to access this entity"
}
```

Status: 403 Forbidden

```
{
    "Error": "The name of boat is already existed"
}
```

Status: 406 Not Acceptable

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

Status: 415 Unsupported Media Type

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

Edit an entire Boat

Allows you to edit all attributes of a boat you created

PUT /boats/:boat_id

Request

Content-Type	application/json
Accept	application/json

Path Parameters

Name	Type	Description
boat_id	String	ID of the boat

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	Length of the boat in feet.	Yes

Request Body Example

```
{
  "name": "Land Witch",
  "type": "Yacht",
  "length": 15
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Failure	401 Unauthorized	- Missing JWT - Invalid JWT - Unauthorized user accesses the data.
Failure	403 Forbidden	The name of the boat is already existed.
Failure	406 Not Acceptable	Accept in request header is not application/json. Server only sends application/json.
Failure	415 Unsupported Media Type	Content-Type in request header is not application/json. Server only accepts application/json.

Response Examples

Success

```
Status: 303 See Other

{
  "id": "abc123",
  "name": "Land Witch",
  "type": "Yacht",
  "length": 15,
  “owner”: {{user identifier}},
  "self": "https://<your-app>/boats/abc123"
}
```

Failure

Status: 401 Unauthorized

```
{
    "Error": "Missing JWT"
}
```

```
{
    "Error": "Invalid JWT"
}
```

```
{
    "Error": "You are not allowed to access this entity"
}
```

Status: 403 Forbidden

```
{
    "Error": "The name of boat is already existed"
}
```

Status: 406 Not Acceptable

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

Status: 415 Unsupported Media Type

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

Assign a Load to a Boat

Protected

Load is assigned to a Boat. Load will change to private.

```
PUT /boats/:boat_id/loads/:load_id
```

Request

Path Parameters

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

Request Body

None

Response

JSON

Response Body Format

Success: JSON

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	Succeeds only if a boat exists with this boat_id, a load exists with this load_id and the load is not assigned to a carrier.
Failure	401 Unauthorized	- Missing JWT - Invalid JWT - Unauthorized user accesses the data.
Failure	403 Forbidden	A load is already assigned to another boat
Failure	404 Not Found	No boat with this boat_id exists, and/or no load with this load_id exits.

Response Examples

Success

```
Status: 200 OK
{
    "boat": {
        "id": "5635008819625984",
        "name": "Odyssey",
        "type": "Yatch",
        "length": 99,
        "owner": 120938102948012,
        "loads": [
            {
                "id": "5633378543992832",
                "self": "http://<your-app>/loads/5633378543992832"
            },
            {
                "id": "5677459739508736",
                "self": "http://<your-app>/loads/5677459739508736"
            }

        ],
        "self": "http://<your-app>/boats/5635008819625984"
    },
    "load": {
        "id": "5677459739508736",
        "volume": 5,
        "content": "LEGO Blocks",
        "creation_date": "10/18/2020",
        "public": false,
        "carrier": {
            "id": "5635008819625984",
            "self": "http://<your-app>/boats/5635008819625984"
        },
        "self": "http://<your-app>/loads/5677459739508736"
    }
}
```

Failure

Status: 401 Unauthorized

```
{
    "Error": "Missing JWT"
}
```

```
{
    "Error": "Invalid JWT"
}
```

```
{
    "Error": "You are not allowed to access this entity"
}
```

Status: 403 Forbidden

```
{
"Error":  "A load is already assigned to another boat"
}
```

Status: 404 Not Found

```
{
"Error":  "Invalid Boat or Load Id"
}
```

Delete a Load from a Boat

Protected

Load is removed from a Boat's loads. Load's carrier is now empty. Load will change to public.

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

Request

Path Parameters

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

Request Body

None

Response

JSON

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 this load is in the boat's loads.
Failure	401 Unauthorized	- Missing JWT - Invalid JWT - Unauthorized user accesses the data.
Failure	403 Forbidden	Load is not in the boat.
Failure	404 Not Found	No boat with this boat_id exists, and/or no load with this load_id exits.

Response Examples

Success

Status: 204 No Content

Failure

Status: 401 Unauthorized

```
{
    "Error": "Missing JWT"
}
```

```
{
    "Error": "Invalid JWT"
}
```

```
{
    "Error": "You are not allowed to access this entity"
}
```

Status: 403 Forbidden

```
{
"Error":  "A load is not in the boat"
}
```

Status: 404 Not Found

```
{
"Error":  "Invalid Boat or Load Id"
}
```

Delete a Boat

Protected

Allows you to delete a boat and unload any loads that were loaded onto it. Private load will change to public.

```
DELETE /boats/:boat_id
```

Request

Path Parameters

Name	Type	Description
boat_id	String	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	- Missing JWT - Invalid JWT - Unauthorized user accesses the data.
Failure	404 Not Found	No boat with this boat_id exists.

Response Examples

Success

Status: 204 No Content

Failure

Status: 401 Unauthorized

```
{
    "Error": "Missing JWT"
}
```

```
{
    "Error": "Invalid JWT"
}
```

```
{
    "Error": "You are not allowed to access this entity"
}
```

Status: 404 Not Found

```
{
"Error":  "Invalid Boat Id"
}
```

Create a Load

Unprotected

Allows you to create a new load.

POST /loads

Request

Content-Type	application/json
Accept	application/json

Path Parameters

None

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Required?
volume	Integer	The volume of the load	Yes
content	String	The content of the load	Yes
public	Boolean	The visibility setting to public	Yes

Request Body Example

```
{
  "volume": 5,
  "content": “Lego Blocks”,
  “public”: true
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	201 Created	
Failure	406 Not Acceptable	Accept in request header is not application/json. Server only sends application/json.
Failure	415 Unsupported Media Type	Content-Type in request header is not application/json. Server only accepts application/json.

Response Examples

- Datastore will automatically generate an ID and store it with the entity being created. This value needs to be sent in the response body as shown in the example.

- The value of the attribute `self` is a live link to the REST resource corresponding to this load. In other words, this is the URL to get this newly created load.

Success

Status: 201 Created

```
{
  "id": "123abc",
  "volume": 5,
  "content": “Lego Blocks”,
  “creation_date”: “10/18/2020”,
  “public”: true,
  "self": "https://<your-app>/loads/123abc"
}
```

* creation_date is generated by server to be stored in datastore

Failure

Status: 406 Not Acceptable

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

Status: 415 Unsupported Media Type

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

View a Load

Unprotected

Allows you to get an existing load.

Load entity is unprotected. However, load that is assigned to a boat and private won’t be visible to the user.

```
GET /loads/:load_id/users/:user_id
```

Request

Accept	application/json

Path Parameters

Name	Type	Description
load_id	String	ID of the load
user_id	String	Unique ID(sub) of the current user generated by authentication system

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	Private load is not visible to public
Failure	404 Not Found	No load with this load_id exists
Failure	406 Not Acceptable	Accept in request header is not application/json. Server only sends application/json.

Response Examples

Success

Status: 200 OK

```
{
  "id": "123abc",
  "volume": 5
  "content": "Lego Blocks",
  “creation_date”: “11/20/2020”,
  “public”: true,
  “carrier”: {
     “id”: “12345”,
     “self”: "https://<your-app>/boats/12345"
  },
  "self": "https://<your-app>/loads/123abc"
}
```

Failure

Status: 401 Unauthorized

```
{
"Error":  "You are not allowed to access to this load"
}
```

Status: 404 Not Found

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

Status: 406 Not Acceptable

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

View all Loads

Unprotected

View all public loads.

GET /loads

Request

Accept	application/json

Parameters

Name	Type	Description
cursor	String	Cursor token of the next page

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	Next property will be null if there are no more pages.
Failure	406 Not Acceptable	Accept in request header is not application/json. Server only sends application/json.

Response Examples

Success

Status: 200 OK

```
{
    "loads": [
        {
            "id": "5629654941564928",
            "volume": 8,
            "content": "Water",
            "creation_date": "2/26/2021",
            "public": true,
            "carrier": {
                "id": "5714080711049216",
                "self": "http://<your-app>/boats/5714080711049216"
            },
            "self": "http://<your-app>/loads/5629654941564928"
        },
        {
            "id": "5645760532054016",
            "volume": 7,
            "content": "Apples",
            "creation_date": "3/26/2021",
            "public": true,
            "carrier": {
                "id": "5714080711049216",
                "self": "http://<your-app>/boats/5714080711049216"
            },
            "self": "http://<your-app>/loads/5645760532054016"
        },
        {
            "id": "5677459739508736",
            "volume": 6,
            "content": "Bananas",
            "creation_date": "4/26/2021",
            "public": true,
            "carrier": {
                "id": "5714080711049216",
                "self": "http://<your-app>/boats/5714080711049216"
            },
            "self": "http://<your-app>/loads/5677459739508736"
        },
        {
            "id": "5677459739508736111",
            "volume": 6,
            "content": "Bananas",
            "creation_date": "4/26/2021",
            "public": true,
            "carrier": {
                "id": "5714080711049216",
                "self": "http://<your-app>/boats/5714080711049216"
            },
            "self": "http://<your-app>/loads/5677459739508736111"
        },
        {
            "id": "5677459739508736222",
            "volume": 6,
            "content": "Bananas",
            "creation_date": "4/26/2021",
            "public": true,
            "carrier": {
                "id": "5714080711049216",
                "self": "http://<your-app>/boats/5714080711049216"
            },
            "self": "http://<your-app>/loads/5677459739508736222"
        }
    ],
    "count": 6
    "next": "http://<your-
app>/loads?cursor=Ci0SJ2oSc35ib2F0LWxvYWQtMzExNjA2chELEgRMb2FkGICAgNjp84oKDBgAIAA="
}
```

Status: 406 Not Acceptable

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

Edit a part of Load

Unprotected

Allows you to edit a part of a load

```
PATCH /loads/:load_id
```

Request

Content-Type	application/json
Accept	application/json

Path Parameters

Name	Type	Description
load_id	String	ID of the load

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Required?
volume	Integer	The volume of the load	No*
content	String	The content of the load	No*
public	Boolean	The visibility setting to public	No*
* The number of attributes to be modified must be between 1 and 3.			

Request Body Example

```
{
  "volume": 2
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	Load is updated
Failure	406 Not Acceptable	Accept in request header is not application/json. Server only sends application/json.
Failure	415 Unsupported Media Type	Content-Type in request header is not application/json. Server only accepts application/json.

Response Examples

Success

Status: 200 OK

```
{
    "id": "5879447253155840",
    "volume": "2",
    "content": "Apples",
    "creation_date": "06/07/2021",
    "public": true,
    "self": "https://<<your-app>>/loads/5879447253155840"
}
```

Failure

```
Status: 406 Not Acceptable
{
    "Error": "Server only sends application/json data"
}
```

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

Edit an entire Load

Unprotected

Allows you to edit all attributes of a load you created

PUT /loads/:load_id

Request

Content-Type	application/json
Accept	application/json

Path Parameters

Name	Type	Description
load_id	String	ID of the load

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Required?
volume	Integer	The volume of the load	Yes
content	String	The content of the load	Yes
public	Boolean	The visibility setting to public	Yes

Request Body Example

```
{
    "volume": 20,
    "content": "Strawberries",
    "public": true
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	Load is updated
Failure	406 Not Acceptable	Accept in request header is not application/json. Server only sends application/json.
Failure	415 Unsupported Media Type	Content-Type in request header is not application/json. Server only accepts application/json.

Response Examples

Success

```
Status: 200 OK
{
    "id": "5879447253155840",
    "volume": 20,
    "content": "Strawberries",
    "creation_date": "06/07/2021",
    "public": true,
    "self": "https://<<your-app>>/loads/5879447253155840"
}
```

Failure

Status: 406 Not Acceptable

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

Status: 415 Unsupported Media Type

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

Delete a Load

Unprotected

Allows you to delete a load. A boat that was carrying it remove the load.

```
DELETE /loads/:load_id
```

Request

Path Parameters

Name	Type	Description
load_id	String	ID of the load

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":  "Invalid Load Id"
}
```

View all Users

Unprotected

View all users who logged in to the application

```
GET /users
```

Request

Parameters

None

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	

Response Examples

Success

```
Status: 200 OK
{
    "users": [
        {
            "name": "Kyeongnam Kim",
            "email": "kimkyeon@oregonstate.edu",
            "sub": "112990147843569814603"
        },
        {
            "name": "Kyeongnam Kim",
            "email": "doublekn88@gmail.com",
            "sub": "106785807497461593019"
        }
    ],
    "count": 2
}
```