Assignment 5: API Spec

CS 493: Cloud Application Development

Fall 2020

Oregon State University

Last Update: Nov 2, 2020. 6:00 pm Pacific

https://hartwelg-hw5-gae.wl.r.appspot.com/

Change log

Page	Change	Date
	Initial version.	M, Oct 12
	Update for HW5	M, Nov 2

Create a Boat

Allows you to create a new boat. Only accepts application/json mimetype and only responds with application/json mimetype.

POST /boats

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	Length of the boat in feet.	Yes

Request Body Example

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

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 boat must not be created, and 400 status code must be returned. Values within the request will be validated upon arrival at endpoint. • Boat name must not exceed 15 characters and must contain only alphanumeric characters or spaces. • Boat length cannot exceed 500 feet. • Boat can be of any type (a boat in this system is defined as "anything that floats" • Boat name must be unique • A request to create or update a boat must always have the required fields listed above; no more, no less.

Failure	403 Forbidden	Boat name is not unique
Failure	415 Unsupported Media Type	Server does not support mime type sent by client

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 `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. You must not store the `self` attribute in Datastore

Success

```
Status: 201 Created

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

Failure

```
Status: 400 Bad Request
```

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

Or

```
{
"Error": "Boat length cannot exceed 500"
}
```

Or

```
{
"Error": "Extraneous field(s) in request"
}
```

Or

```
{
"Error": "Boat name is too long or contains non-alphanumeric characters"
}
```

Failure

Status: 403 Forbidden

```
{
"Error": "Boat name is not unique"
}
```

Failure

Status: 415 Unsupported Media Type

Get a Boat

Allows you to get an existing boat. Endpoint accepts text/html and application/json mimetypes and responds with only text/html or application/json mimetypes.

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

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	404 Not Found	No boat with this boat_id exists
Failure	406 Not Acceptable	Client sent unsupported mime type

Response Examples

Success

```
Status: 200 OK


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

Failure

```
Status: 404 Not Found


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

Failure

```
Status: 406 Not Acceptable
```

List all Boats

List all the boats. Endpoint accepts any mimetype, but only responds with application/json mimetype.

```
GET /boats
```

Request

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

[
{
  "id": "abc123",
  "name": "Sea Witch",
  "type": "Catamaran",
  "length": 28,
  "self": "https://<your-app>/boats/abc123"
},
{
  "id": "def456",
  "name": "Adventure",
  "type": "Sailboat",
  "length": 50,
  "self": "https://<your-app>/boats/def456"
},
{
  "id": "xyz123",
  "name": "Hocus Pocus",
  "type": "Sailboat",
  "length": 100,
  "self": "https://<your-app>/boats/xyz123"
}
]
```

Edit a Boat

Allows you to edit a boat. Endpoint only accepts application/json mimetype and will only respond with application/json mimetype.

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	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": 99
}
```

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 boat must not be updated, and 400 status code must be returned. Values within the request will be validated upon arrival at endpoint. • Boat name must not exceed 15 characters and must contain only alphanumeric characters or spaces. • Boat length cannot exceed 500 feet. • Boat can be of any type (a boat in this system is defined as "anything that floats" • Boat name must be unique • A request to create or update a boat must always have the required fields listed above; no more, no less.
Failure	404 Not Found	No boat with this boat_id exists
Failure	406 Not Acceptable	Client sent unsupported mime type
Failure	415 Unsupported Media Type	Client accepts unsupported media type

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

```
Status: 200 OK

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

Failure

```
Status: 400 Bad Request

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

```
{
"Error": "Boat length cannot exceed 500"
}
```

Or

```
{
"Error": "Extraneous field(s) in request"
}
```

Or

```
{
"Error": "Boat name is too long or contains non-alphanumeric characters"
}
```

Status: 404 Not Found

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

Status: 406 Not Acceptable

Status: 415 Unsupported Media Type

Delete a Boat

Allows you to delete a boat. Endpoint will accept any mimetype, but only respond with text/html or application/json mimetypes.

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

Response Examples

Success

Status: 204 No Content

Failure

```
Status: 404 Not Found


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

Edit a Boat

Allows you to edit a boat. Endpoint only accepts application/json mimetype and will only respond with application/json mimetype.

PUT /boats/:boat_id

Request

Request Parameters

N

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Required?
name	String	The name of the boat.	Yes
Type	String	Type of the boat	Yes
Length	Integer	Length of the boat	Yes

Request Body Example

```
{
  "name": "Odyssey III",
  "type": "Mega Yatch",
  "length": 200
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	303 See Other	Boat is updated and a link to the updated boat will be given in the Location header
Failure	400 Bad Request	If the request is missing the number attribute, the slip must not be created, and 400 status code must be returned. Values within the request will be validated upon arrival at endpoint. • Boat name must not exceed 15 characters and must contain only alphanumeric characters or spaces. • Boat length cannot exceed 500 feet. • Boat can be of any type (a boat in this system is defined as "anything that floats" • Boat name must be unique • A request to create or update a boat must always have the required fields listed above; no more, no less.
Failure	406 Not Acceptable	Client requests content type not supported by server

Failure	415 Unsupported Media Type	Client sent mime type not supported by server

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 slip. In other words, this is the URL to get this newly created slip. You must not store this attribute in Datastore

Success

Status: 303 See Other

Failure

| Status: 400 Bad Request

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

Or

{
"Error": "Boat length cannot exceed 500"
}

Or

{
"Error": "Extraneous field(s) in request"
}

Or

{
"Error": "Boat name is too long or contains non-alphanumeric characters"
} |
|---|

Status: 406 Not Acceptable

Status: 415 Unsupported Media Type

Delete all Boats

Attempt to delete all boats. Endpoint accepts any mimetype, but only responds with application/json mimetype.

DELETE /boats

Request

Request Parameters

None

Request Body

None

Response

No body

Response Body Format

Success: None

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Failure	405 Method Not Allowed	Cannot delete all boats at once

Response Examples

Failure

Status: 405 Method Not Allowed

Edit all Boats

Attempt to Edit all boats. Endpoint accepts any mimetype, but only responds with application/json mimetype.

PUT /boats

Request

Request Parameters

None

Request Body

None

Response

No body

Response Body Format

Success: None

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Failure	405 Method Not Allowed	Cannot edit all boats at once

Response Examples

Failure

Status: 405 Method Not Allowed