

Disclaimer

This isn't a 100% RESTful API. After reading through some of the Piazza posts, it looks like it better to have a logical API rather than be 110% restful.

For example:

I have routes such as [GET] `http://35.169.224.183:3111/boats/docked` or [GET] `http://35.169.224.183:3111/boats/at_sea` to get the logical results of the boats.

Also for linking the URLs of boats / slips. I instead included the specific id that the ship or slip is occupying. Not the full URL to the item. To query, simply make a get request with the ID.

Also.. Some of these routes are /boats vs /boat I know it's bad, its super bowl sunday though give me a break :D

Boat Routes

Route: /boats

Method: GET

Description:

- Get all boats in the database.
- Pass in query parameters to refine results by specific properties

Parameters(OPTIONAL):

- Query: type:<string> | length:<string> | name:<string> | at_sea: <boolean> | docked_at: <number>

Example: `http://35.169.224.183:3111/boats` |
`http://35.169.224.183:3111/boats?length=15`

Route: /boats/{boat_id}

Method: GET

Description:

Get singular boat from the database based on the DB _id.

Parameters(Required):

- Route: boat_id: mongoDB id

Example: `http://35.169.224.183:3111/boats/5a7753d70db47b72f9d1f035`

Route: /boats/at_sea

Method: GET

Description:

Get all boats that are currently at sea in the DB

Parameters(NONE)

Example: http://35.169.224.183:3111/boats/at_sea

Route: /boats/docked

Method: GET

Description:

Get all boats that are currently docked in the marina

Parameters(NONE)

Example: http://35.169.224.183:3111/boats/at_sea

Route: /boats

Method: PUT

Description:

Not allowed, because you can't replace an entire boat in real life.

Parameters(NONE)

Returns

` Status: 403`

Route: /boats/{boat_id}

Method: PUT

Description:

Not allowed, because you can't replace an entire boat in real life. Only included for consistency

Parameters(Required)

MongoDB ID

Returns

` Status: 403`

Route: /boat/{boat_id}

Method: DELETE

Description:

- Delete a boat from the DB with the exact mongoDB ID
- If the boat is currently docked, it sets the slip that it was in to empty.
This can be seen in the Postman test called 'DELETE Boat DOCKED'

Parameters(Required):

ROUTE: boat_id : mongoDB _id string

Returns

- ```
{
 ok: 1,
 n: 1
}
```

**Route:** /boat

**Method:** POST

### Description:

- Create a boat in the marina, defaults to be set at sea

### Parameters(Required)

- ```
{
  "type": "catamaran", // string
  "length": 5, // number
  "name": "Brandon" //string
}
```

Returns

- ```
{
 "_id": "5a7753c00db47b72f9d1f034",
 "type": "catamaran",
 "length": 5,
 "name": "Brandon",
 "at_sea": "true",
 "docked_at": ""
}
```

---

**Route:** /boat/{boat\_id}/depart

**Method:** Patch

**Description:**

- Departs the boat from either the marina, or returns an error saying that the boat is already at sea
- Tests in Postman called Depart Boat & Depart Boat - AT SEA

**Parameters(Required):**

- boat\_id: mongoDB \_id string

**Returns either**

- ```
{
  "status": 503,
  "Message": "Boat already at sea."
}
```
- ```
{
 "slip": {
 "_id": "5a7761d10db47b73d3bb8e4b",
 "number": 31,
 "current_boat": "",
 "arrival_date": "",
 "departure_history": []
 },
 "boat": {
 "_id": "5a7761d10db47b73d3bb8e4a",
 "type": "catamaran",
 "length": 25,
 "name": "Brandon",
 "at_sea": "true",
 "docked_at": ""
 }
}
```

---

**Route:** /boat/{boat\_id}/arrival

**Method:** Patch

**Description:**

- docks the boat from the sea, or returns an error saying that the boat is already docked
- Tests in Postman called Park / arrive & park / arrive - At SEA

**Parameters(Required):**

- boat\_id: mongoDB \_id string

**Returns either**

- ```
{
  "slip": {
    "_id": "5a77304c8649ddbe4445cc58",
    "number": 5,
    "current_boat": "5a77304c8649ddbe4445cc57",
    "arrival_date": "04-02-2018",
    "departure_history": []
  },
  "boat": {
    "_id": "5a77304c8649ddbe4445cc57",
    "type": "catamaran",
    "length": 25,
    "name": "Brandon",
    "at_sea": "false",
    "docked_at": 5
  }
}
```
- ```
{
 "status": 503,
 "Message": "Boat already docked."
}
```

---

**Route:** /boat/{boat\_id}

**Method:** Patch

**Description:**

- Route used to just change either length, name, or type of boat
- Postman tests: PATCH - Boat with ID & BOGUS PATCH - Boat with ID

**Parameters(Required):**

```
- ROUTE - boat_id: mongoDB _id string
- BODY - {
 "type": ["sail" | "power" | "dingy" | "catamaran" | "row"],
 "length": 30, // number
 "name": "Kelli" // string
}
```

### Returns either

- ```
{
  "_id": "5a77636b0db47b73d3bb8e4c",
  "type": "sail",
  "length": 30,
  "name": "Kelli",
  "at_sea": "true",
  "docked_at": ""
}
```
- ```
{
 "Status": 503,
 "Message": "Bad Request"
}
```

---

## Slip Routes

### Route: /slips

**Method:** GET

#### Description:

- ```
- Get all slips in the database.
- Pass in query parameters to refine results by specific properties
```

Parameters(OPTIONAL):

- Query: number:<number> | current_boat:<string> | arrival_date:<string>

Example: <http://35.169.224.183:3111/slipss> |
<http://35.169.224.183:3111/slips?number=3>

Route: /slips/slip_number

Method: GET

Description:

- ```
- Get single slip in DB by human readable number.
```

### Parameters(Required):

- Route - number : number

Example: `http://35.169.224.183:3111/slipss | http://35.169.224.183:3111/slips?number=3`

---

**Route:** `/slip`

**Method:** PUT

### Description:

Not allowed, because you can't replace a port in a marina.

### Parameters(NONE)

### Returns

Status: 403

---

**Route:** `/slip/{slip_id}`

**Method:** PUT

### Description:

Not allowed, because you can't replace an entire port in real life. Only included for consistency

### Parameters(Required)

Marina number

### Returns

Status: 403

---

**Route:** `/slip/{slip_id}`

**Method:** DELETE

### Description:

- Delete a slip from the DB with the human readable marina number  
- If there is a boat is currently docked, it sets the boat to be at sea This can be seen in the Postman test called 'Slip - SLIP OCCUPIED'

### Parameters(Required):

ROUTE: slip\_id : number

### Returns either..

- ```
{
  "ok": 1,
  "n": 1
}
```
- ```
{
 "boat": {
 "ok": 1,
 "n": 1
 },
 "slip": {
 "ok": 1,
 "n": 1
 }
}
```

---

### Route: /slip

Method: POST

### Description:

- Create a slip in the marina

### Parameters(None)

### Returns

- ```
{
  "_id": "5a7769f90db47b73d3bb8e53",
  "number": 34,
  "current_boat": "",
  "arrival_date": "",
  "departure_history": []
}
```

Route: /slip

Method: PATCH

Description:

- Not allowed, as the fields are all unique it isn't allowed to alter any of them
- To alter the boats at the slip instead have the boat depart from the slip, then dock another ship

Parameters(NONE)

Returns

Status: 403

Route: /slip/{slip_id}

Method: PATCH

Description:

- Not allowed, as the fields are all unique it isn't allowed to alter any of them
- To alter the boats at the slip instead have the boat depart from the slip, then dock another ship
- Only included for consistency

Parameters(Required)

Marina number

Returns

Status: 403