Electric Cars and Chargers API
CS 493: Cloud Application Development

Matt Beachly

Welcome Page URL .... https://final-beachlym.appspot.com/

Create a User

https://final-beachlym.appspot.com/

Description

User accounts are not created via requests to the API. To create a user account, you must visit the URL above in a browser which takes you to the Welcome page. From there click "Authorize!", which directs you to where you can log in to a Google Gmail account. This allows the Electric Cars app to get your first name, last name, and user name from Google via OAuth 2.0. However, your user account is not created in the datastore until you are redirected back to the Welcome page and click the "See Your Info!" button.

Clicking the "See Your Info!" button causes the Electric Cars app to get your information and a JWT (JSON Web Token) from Google, and then display that information to you on a new page. You will need to copy your JWT token from this page in order to make POST, PUT, PATCH, and DELETE requests to the API (GET requests do NOT require a JWT token). The JWT token must be specified in requests as a "Bearer Token". Please note that the JWT token expires after a few minutes and you will need to follow the same process to get a new one, otherwise the API will verify and reject your expired token. The API also uses the JWT token to extract the User ID from the "sub" attribute, as it is a unique identifier. User name attributes are not unique and thus are not used for ID. You can copy the User ID from the "See Your Info!" page in order to make GET requests to list the cars or chargers owned by that User ID.

Clicking the "See Your Info!" button also creates a User entity in the datastore for your account. The entity contains four attributes: id (obtained from the sub attribute in the JWT token), givenName, familyName, and displayName. The ID is used to show the ownership relationship between users and cars and users and chargers. Cars and chargers both have an owner attribute that contains the ID of the user they are owned by. These relationships are defined when cars or chargers are created via a POST request containing the user's JWT token. At this point the ownership relationships are only created upon creation of cars and chargers and cannot be modified.

List all Users

List all the users, up-to-five users per page.

GET /users

Request

Request Parameters

Name	Type	Description	Required?
cursor	String	Token for which page of users to return. If not provided the first page will be returned.	No

Request Header

Name	Type	Description	Required?
Accept	String	'application/json'	Yes

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	If client accepts JSON response
Failure	406 Not Acceptable	If client does not accept JSON response

Response Examples

Success

```
Status: 200 OK
{
    "items": [
        {
            "givenName": "Matthew",
            "familyName": "Beachly",
            "displayName": "Matthew Beachly",
            "id": "108199616256893398846",
            "self": "https://final-beachlym.appspot.com/users/108199616256893398846"
        },
        {
            "givenName": "mattbeachly",
            "displayName": "mattbeachly",
            "id": "107197049100131265837",
            "self": "https://final-beachlym.appspot.com/users/107197049100131265837"
        }
    ],
    "self": "https://final-beachlym.appspot.com/users",
    "next": "None",
```

```
   "total": 2
}
```

```
Status: 406 Not Acceptable
{
   "Error": "Client must accept application/json."
}
```

Get a User

Allows you to get an existing user

GET /users/:user_id

Request

Request Parameters

Name	Type	Description	Required?
user_id	String	ID of the user	Yes

Request Header

Name	Type	Description	Required?
Accept	String	'application/json'	Yes

Request Body

None

Response

Response Body Format

JSON or HTML

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	If client accepts JSON response
Failure	404 Not Found	If no user with this user_id exists
Failure	406 Not Acceptable	If client does not accept JSON response

Response Examples

Success

```
Status: 200 OK
{
    "id": "107197049100131265837",
    "givenName": "mattbeachly",
    "displayName": "mattbeachly",
    "self": "https://final-beachlym.appspot.com/users/107197049100131265837"
}
```

Failure

```
Status: 404 Not Found
{
    "Error": "No user with this user_id exists."
}
```

Failure

```
Status: 406 Not Acceptable
{
    "Error": "Client must accept application/json."
}
```

Create a Car

Allows you to create a new car.

POST /cars

Description

Car entities have five required attributes: id, make, model, year, owner, and one optional attribute: charger. In POST requests you must only specify three attributes: make (the manufacturer of the car), model, and year (year the car was built). The ID attribute is a unique identifier that is automatically generated by the datastore. The owner attribute is your user ID and is automatically extracted from your JWT token sent with the request. The charger attribute indicates the ID of the charger the car is parked at and must be set later via a PUT or DELETE request.

Request

Request Parameters

None

Request Authorization

Type	Description	Required?
Bearer Token	JWT token copied from "Show Your Info!" page	Yes

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Required?
make	String	The make of the car. E.g., Tesla, Kia, etc. Must be string 2 characters or longer.	Yes
model	String	The model of the car. E.g., Cybertruck, Leaf, etc. Must be string 2 characters or longer.	Yes
year	Number	Year the car was made, must be > 1800.	Yes

Request Body Example

```
{
  "make": "Anderson",
  "model": "Detroit Electric",
  "year": 1911
}
```

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

Failure	400 Bad Request	If make attribute is not a string
Failure	400 Bad Request	If make attribute has less than 2 characters
Failure	400 Bad Request	If model attribute is not a string
Failure	400 Bad Request	If model attribute has less than 2 characters
Failure	400 Bad Request	If length attribute is not a number
Failure	400 Bad Request	If length attribute is not > 1800
Failure	401 Unauthorized	If no JWT token or a bad JWT token is provided.
Failure	403 Forbidden	If ID, owner, or charger are in the request body.
Failure	405 Method Not Allowed	If a car_id is specified in URL
Failure	406 Not Acceptable	If client does not accept application/json data
Failure	415 Unsupported Media Type	If client does not provide application/json data

Response Examples

Success

```
Status: 201 Created
{
    "make": "Tesla",
    "model": "Cybertruck",
    "year": 2021,
    "owner": "108199616256893398846",
    "id": "5679790782676992",
    "self": "https://final-beachlym.appspot.com/cars/5679790782676992"
}
```

Failure

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

Failure

```
Status: 400 Bad Request
{
"Error": "Make must be >= 2 characters."
}
```

Failure

```
Status: 400 Bad Request
{
"Error": "Model must be in string format."
}
```

Failure

```
Status: 400 Bad Request
{
"Error": "Year must be > 1800."
}
```

Failure

Status: 406 Not Acceptable
{
"Error": "Client must accept application/json data."
}

Failure

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

Get a Car

Allows you to get an existing car

GET /cars/:car_id

Request

Request Parameters

Name	Type	Description	Required?
car_id	String	ID of the car	Yes

Request Header

Name	Type	Description	Required?
Accept	String	'application/json'	Yes

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	If client accepts JSON response
Failure	404 Not Found	If no car with this car_id exists
Failure	406 Not Acceptable	If client does not accept application/json data

Response Examples

Success

```
Status: 200 OK
{
    "owner": "108199616256893398846",
    "year": 1911,
    "make": "Anderson",
    "model": "Detroit Electric",
    "id": "5679790782676992",
    "self": "https://final-beachlym.appspot.com/cars/5679790782676992"
}
```

Failure

```
Status: 404 Not Found
{
"Error":  "No car with this car_id exists"
}
```

Failure

```
Status: 406 Not Acceptable
{
    "Error": "Client must accept application/json."
}
```

List all Cars

List all the cars, up-to-five cars per page.

GET /cars

Request

Request Parameters

Name	Type	Description	Required?
cursor	String	Token for which page of cars to return. If not provided the first page will be returned.	No

Request Header

Name	Type	Description	Required?
Accept	String	'application/json'	Yes

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	406 Not Acceptable	If client does not accept application/json data

Response Examples

Success

```
Status: 200 OK
{
   "items": [
     {
        "make": "Tesla",
        "model": "Cybertruck",
        "owner": "108199616256893398846",
        "year": 2021,
        "id": "5701330244993024",
        "self": "https://final-beachlym.appspot.com/cars/5701330244993024"
     },
…
   ],
   "self": "https://final-beachlym.appspot.com/cars",
   "next": "https://final-
beachlym.appspot.com/cars?cursor=CkcKGwoFbW9kZWwSEhoQRGV0cm9pdCBFbGVjdH
JpYxIkahBtfmZpbmFsLWJlYWNobHltchALEgNDYXIYgICAwNW3iwoMGAAgAA%3D%
3D",
   "total": 7
}
```

List all Cars by Owner

List all the cars owned by a user

GET /users/:user_id/cars

Request

Request Parameters

Name	Type	Description	Required?
user_id	String	ID of the user	Yes

Request Header

Name	Type	Description	Required?
Accept	String	'application/json'	Yes

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	404 Not Found	If no user with this user_id exists
Failure	406 Not Acceptable	If client does not accept application/json data

Response Examples

Success

```
Status: 200 OK
{
   "items": [
     {
        "owner": "107197049100131265837",
        "year": 2021,
        "make": "Tesla",
        "model": "Cybertruck",
        "id": "5634263223369728",
        "self": "https://final-beachlym.appspot.com/cars/5634263223369728"
     },
…
   ],
   "self": "https://final-beachlym.appspot.com/users/107197049100131265837/cars",
   "total": 5
}
```

Patch a Car

Allows you to edit a car.

PATCH /cars/:car_id

Request

Request Parameters

Name	Type	Description	Required?
car_id	String	ID of the car	Yes

Request Header

Name	Type	Description	Required?
Accept	String	'application/json'	Yes

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Required?
make	String	The make of the car. E.g., Tesla, Kia, etc. Must be string 2 characters or longer.	No
model	String	The model of the car. E.g., Cybertruck, Leaf, etc. Must be string 2 characters or longer.	No
year	Number	Year the car was made, must be > 1800.	No

Request Body Example

```
{
   "year": 1910
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	400 Bad Request	If make attribute is not a string
Failure	400 Bad Request	If make attribute has less than 2 characters
Failure	400 Bad Request	If model attribute is not a string
Failure	400 Bad Request	If model attribute has less than 2 characters
Failure	400 Bad Request	If length attribute is not a number
Failure	400 Bad Request	If length attribute is not > 1800
Failure	401 Unauthorized	If no JWT token or a bad JWT token is provided.
Failure	403 Forbidden	If request redefines the id, owner, or charger attribute
Failure	403 Forbidden	If ID in JWT does not match owner
Failure	404 Not Found	If no car with this car_id exists

Failure	405 Method Not Allowed	If no car_id provided in URL

Response Examples

Success

Status: 200 OK
{
   "year": 1910,
   "make": "Anderson",
   "model": "Detroit Electric",
   "owner": "107197049100131265837",
   "id": 5711673465765888,
   "self": "https://final-beachlym.appspot.com/cars/5711673465765888"
}

Failure

Status: 400 Bad Request
{
"Error": "Make must be in string format."
}

Failure

Status: 403 Forbidden
{
"Error": "The ID cannot be updated"
}

Failure

Status: 404 Not Found
{
"Error": "No car with this car_id exists"
}

Failure

Status: 405 Method Not Allowed
{
"Error": "Must specify car_id to be put."
}

Put a Car

Allows you to edit a car.

PUT /cars/:car_id

Request

Request Parameters

Name	Type	Description	Required?
car_id	String	ID of the car	Yes

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Required?
make	String	The make of the car. Must be string 2 characters or longer.	Yes
model	String	The model of the car. Must be string 2 characters or longer.	Yes
year	Number	Year the car was made, must be > 1800.	Yes

Request Body Example

```
{
  "make": "Anderson",
  "model": "Detroit Electric",
  "year": 1911
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	303 See Other	URL to newly created car is returned
Failure	400 Bad Request	If any of the three required attributes are missing
Failure	400 Bad Request	If make attribute is not a string
Failure	400 Bad Request	If make attribute has less than 2 characters
Failure	400 Bad Request	If model attribute is not a string
Failure	400 Bad Request	If model attribute has less than 2 characters
Failure	400 Bad Request	If length attribute is not a number
Failure	400 Bad Request	If length attribute is not > 1800
Failure	401 Unauthorized	If no JWT token or a bad JWT token is provided.
Failure	403 Forbidden	If request redefines the id, owner, or charger attribute
Failure	404 Not Found	If no car with this car_id exists
Failure	405 Method Not Allowed	If no car_id provided in URL

Response Examples

Status: 303 See Other
{
See Other.
}

Status: 400 Bad Request
{
"Error": "Model must be in string format."
}

Status: 403 Forbidden
{
"Error": "The ID cannot be updated"
}

Status: 403 Forbidden
{
"Error": "User sub ID in JWT token does not match car owner."
}

Status: 404 Not Found
{
"Error": "No car with this car_id exists"
}

Status: 405 Method Not Allowed
{
"Error": "Must specify car_id to be patched."
}

Delete a Car

Allows you to delete a car.

```
DELETE /cars/:car_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	401 Unauthorized	If no JWT token or a bad JWT token is provided.
Failure	403 Forbidden	User sub ID in JWT token does not match car owner
Failure	404 Not Found	No car with this car_id exists
Failure	405 Method Not Acceptable	If car_id not provided in URL

Response Examples

Success

```
Status: 204 No Content
```

Failure

```
Status: 403 Forbidden
{
"Error":  "User sub ID in JWT token does not match car owner."
}
```

Failure

```
Status: 404 Not Found
{
"Error":  "No car with this car_id exists"
}
```

Failure

```
Status: 405 Method Not Allowed
{
"Error":  "Must specify car_id to be deleted."
}
```

Create a Charger

Allows you to create a new charger.

POST /chargers

Description

Charger entities have five required attributes: id, address, connector, output, owner, and one optional attribute: car. In POST requests you must only specify three attributes: address (where the charger is located), connector, and output (power in kW). The ID attribute is a unique identifier that is automatically generated by the datastore. The owner attribute is your user ID and is automatically extracted from your JWT token sent with the request. The car attribute indicates the ID of the car parked at the charger and must be set later via a PUT or DELETE request.

Request

Request Parameters

None

Request Authorization

Type	Description	Required?
Bearer Token	JWT token copied from "Show Your Info!" page	Yes

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Required?
address	String	The address of the charger. E.g. 2350 Electric Ave. Must be string 2 characters or longer.	Yes
model	String	The connector type. E.g. Type 2, CCS, CHAdeMO, etc. Must be string 2 characters or longer.	Yes
output	Number	The power output of the charger in kW. Must be > 0.	Yes

Request Body Example

```
{
  "address": "123 Fake St.",
  "connector": "Type 2",
  "output": 43
}
```

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

Failure	400 Bad Request	If address attribute is not a string
Failure	400 Bad Request	If address attribute has less than 2 characters
Failure	400 Bad Request	If connector attribute is not a string
Failure	400 Bad Request	If connector attribute has less than 2 characters
Failure	400 Bad Request	If output attribute is not a number
Failure	400 Bad Request	If output attribute is not > 0
Failure	401 Unauthorized	If no JWT token or a bad JWT token is provided.
Failure	403 Forbidden	If ID, owner, or car are in the request body.
Failure	405 Method Not Allowed	If a car_id is specified in URL
Failure	406 Not Acceptable	If client does not accept application/json data
Failure	415 Unsupported Media Type	If client does not provide application/json data

Response Examples

Success

```
Status: 201 Created
{
    "address": "123 Fake St.",
    "connector": "Type 2",
    "output": 43,
    "owner": "107197049100131265837",
    "id": "5715999101812736",
    "self": "https://final-beachlym.appspot.com/chargers/5715999101812736"
}
```

Failure

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

Failure

```
Status: 403 Forbidden
{
    "Error": "Can't send car, must be set via /chargers/:charger_id/car/:car_id."
}
```

Failure

```
Status: 406 Not Acceptable
{
"Error": "Client must accept application/json data."
}
```

Failure

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

Get a Charger

Allows you to get an existing charger

GET /chargers/:charger_id

Request Parameters

Name	Type	Description	Required?
charger_id	String	ID of the charger	Yes

Request Header

Name	Type	Description	Required?
Accept	String	'application/json'	Yes

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	If client accepts JSON response
Failure	404 Not Found	If no charger with this charger_id exists
Failure	406 Not Acceptable	If client does not accept application/json data

Response Examples

Success

```
Status: 200 OK
{
    "connector": "CHAdeMO",
    "owner": "107197049100131265837",
    "output": 62.5,
    "address": "456 Lame Ln.",
    "id": "5715999101812736",
    "self": "https://final-beachlym.appspot.com/chargers/5715999101812736",
    "car": "None"
}
```

Failure

```
Status: 404 Not Found
{
"Error":  "No charger with this charger_id exists"
}
```

Failure

```
Status: 406 Not Acceptable
{
    "Error": "Client must accept application/json."
}
```

List all Chargers

List all the chargers, up-to-five chargers per page.

GET /chargers

Request

Request Parameters

Name	Type	Description	Required?
cursor	String	Token for which page of chargers to return. If not provided the first page will be returned.	No

Request Header

Name	Type	Description	Required?
Accept	String	'application/json'	Yes

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	406 Not Acceptable	If client does not accept application/json data

Response Examples

Success

```
Status: 200 OK
{
   "items": [
     {
       "connector": "Type 2",
       "owner": "107197049100131265837",
       "output": 43,
       "address": "123 Fake St.",
       "id": "5665833682468864",
       "self": "https://final-beachlym.appspot.com/chargers/5665833682468864"
     },
     {
       "address": "456 Lame Ln.",
       "connector": "CHAdeMO",
       "owner": "107197049100131265837",
       "output": 62.5,
       "id": "5667370274127872",
       "self": "https://final-beachlym.appspot.com/chargers/5667370274127872"
     }
   ],
   "self": "https://final-beachlym.appspot.com/chargers",
```

```
    "next": "None",
    "total": 2
}
```

Failure
```
Status: 406 Not Acceptable
{
    "Error": "Client must accept application/json."
}
```

List all Chargers by Owner

List all the chargers owned by a user

GET /users/:user_id/chargers

Request

Request Parameters

Name	Type	Description	Required?
user_id	String	ID of the user	Yes

Request Header

Name	Type	Description	Required?
Accept	String	'application/json'	Yes

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	404 Not Found	If no user with this user_id exists
Failure	406 Not Acceptable	If client does not accept application/json data

Response Examples

Success

```
Status: 200 OK
{
   "items": [
      {
         "output": 43,
         "address": "123 Fake St.",
         "connector": "Type 2",
         "owner": "107197049100131265837",
         "id": "5665833682468864",
         "self": "https://final-beachlym.appspot.com/chargers/5665833682468864"
      },
      {
         "connector": "CHAdeMO",
         "owner": "107197049100131265837",
         "output": 62.5,
         "address": "456 Lame Ln.",
         "id": "5667370274127872",
         "self": "https://final-beachlym.appspot.com/chargers/5667370274127872"
      }
   ],
   "self": "https://final-beachlym.appspot.com/users/107197049100131265837/chargers",
```

```
    "total": 2
}
```

Failure

```
Status: 404 Not Found
{
    "Error": " No user with this user_id exists."
}
```

Failure

```
Status: 406 Not Acceptable
{
    "Error": "Client must accept application/json."
}
```

Patch a Charger

Allows you to edit a charger.

PATCH /chargers/:charger_id

Request

Request Parameters

Name	Type	Description	Required?
charger_id	String	ID of the charger	Yes

Request Header

Name	Type	Description	Required?
Accept	String	'application/json'	Yes

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Required?
address	String	The address of the charger. E.g. 2350 Electric Ave. Must be string 2 characters or longer.	No
model	String	The connector type. E.g. Type 2, CCS, CHAdeMO, etc. Must be string 2 characters or longer.	No
output	Number	The power output of the charger in kW. Must be > 0.	No

Request Body Example

```
{
   "output": 100
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	400 Bad Request	If address attribute is not a string
Failure	400 Bad Request	If address attribute has less than 2 characters
Failure	400 Bad Request	If connector attribute is not a string
Failure	400 Bad Request	If connector attribute has less than 2 characters
Failure	400 Bad Request	If output attribute is not a number
Failure	400 Bad Request	If output attribute is not > 0
Failure	401 Unauthorized	If no JWT token or a bad JWT token is provided.
Failure	403 Forbidden	If request redefines the id, owner, or car attribute
Failure	403 Forbidden	If ID in JWT does not match owner
Failure	404 Not Found	If no charger with this charger_id exists

Failure	405 Method Not Allowed	If no charger_id provided in URL

Success

```
Status: 200 OK
{
    "connector": "CCS",
    "output": 80,
    "address": "123 Fake St.",
    "owner": "107197049100131265837",
    "id": 5665833682468864,
    "self": "https://final-beachlym.appspot.com/chargers/5665833682468864"
}
```

Failure

```
Status: 400 Bad Request
{
"Error": "Address must be in string format."
}
```

Failure

```
Status: 403 Forbidden
{
"Error": "The ID cannot be updated"
}
```

Failure

```
Status: 404 Not Found
{
"Error": "No charger with this charger_id exists"
}
```

Failure

```
Status: 405 Method Not Allowed
{
"Error": "Must specify charger_id to be put."
}
```

Put a Charger

Allows you to edit a charger.

PUT /chargers/:charger_id

Request

Request Parameters

Name	Type	Description	Required?
charger_id	String	ID of the charger	Yes

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Type	Description	Required?
address	String	The address of the charger. E.g. 2350 Electric Ave. Must be string 2 characters or longer.	Yes
model	String	The connector type. E.g. Type 2, CCS, CHAdeMO, etc. Must be string 2 characters or longer.	Yes
output	Number	The power output of the charger in kW. Must be > 0.	Yes

Request Body Example

```
{
  "address": "456 Lame Ln.",
  "connector": "CHAdeMO",
  "output": 62.5
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	303 See Other	URL to newly created car is returned
Failure	400 Bad Request	If any of the three required attributes are missing
Failure	400 Bad Request	If address attribute is not a string
Failure	400 Bad Request	If address attribute has less than 2 characters
Failure	400 Bad Request	If connector attribute is not a string
Failure	400 Bad Request	If connector attribute has less than 2 characters
Failure	400 Bad Request	If output attribute is not a number
Failure	400 Bad Request	If output attribute is not > 0
Failure	401 Unauthorized	If no JWT token or a bad JWT token is provided.
Failure	403 Forbidden	If request redefines the id, owner, or car attribute
Failure	404 Not Found	If no charger with this charger_id exists
Failure	405 Method Not Allowed	If no charger_id provided in URL

Response Examples

Status: 303 See Other
{
See Other.
}

Status: 400 Bad Request
{
"Error": "Model must be in string format."
}

Status: 403 Forbidden
{
"Error": "The ID cannot be updated"
}

Status: 403 Forbidden
{
"Error": "User sub ID in JWT token does not match charger owner."
}

Status: 404 Not Found
{
"Error": "No charger with this charger_id exists"
}

Status: 405 Method Not Allowed
{
"Error": "Must specify charger_id to be patched."
}

Delete a Charger

Allows you to delete a charger.

DELETE /chargers/:charger_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	401 Unauthorized	If no JWT token or a bad JWT token is provided.
Failure	403 Forbidden	User sub ID in JWT token does not match charger owner
Failure	404 Not Found	No charger with this charger_id exists
Failure	405 Method Not Acceptable	If charger_id not provided in URL

Response Examples

Success

Status: 204 No Content

Failure

```
Status: 403 Forbidden
{
"Error":  "User sub ID in JWT token does not match charger owner."
}
```

Failure

```
Status: 404 Not Found
{
"Error":  "No car with this charger_id exists"
}
```

Failure

```
Status: 405 Method Not Allowed
{
"Error":  "Must specify charger_id to be deleted."
}
```

Put a Car at a Charger

Allows you to park a car at a charger.

Description

Only one car can be parked at a charger at a time, and likewise, a car can only be parked at one charger at a time. A JWT token must be provided that matches the owner of the car, but it does not need to match the owner of the charger. This is because you can only move (drive) cars that you own, but you may use chargers owned by someone else if another car is not already using them. If an address of a charger can support charging more than one car then there should be multiple charger records for that address (address is not unique). At this point the API does not check if the charger's connector type supports the electric car parked at it.

```
PUT /chargers/:charger_id/car/:car_id
```

Request

Request Parameters

Name	Type	Description	Required?
charger_id	String	ID of the charger	Yes
car_id	String	ID of the car	Yes

Request Body

None

Response

Response Body Format

Success: No body
Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	Car is assigned to charger
Failure	401 Unauthorized	If no JWT token or a bad JWT token is provided
Failure	403 Forbidden	If user ID in JWT token does not match car owner
Failure	403 Forbidden	If this car is already parked at another charger
Failure	403 Forbidden	If this charger already has a car parked at it
Failure	404 Not Found	If no car with this car_id exists
Failure	404 Not Found	If no charger with this charger_id exists

Response Examples

Failure

```
Status: 403 Forbidden
{
    "Error": "This car is already parked at a charger"
}
```

Failure

```
Status: 403 Forbidden
{
"Error":  "User sub ID in JWT token does not match car owner."
}
```

Failure

```
Status: 404 Not Found
{
"Error": "The specified car doesn't exist."
}
```

Failure

```
Status: 404 Not Found
{
"Error": "The specified charger doesn't exist."
}
```

Get Car at Charger

Allows you to get the car assigned to a charger

GET /chargers/:charger_id/car/

Request

Request Parameters

Name	Type	Description	Required?
charger_id	String	ID of the charger	Yes

Request Header

Name	Type	Description	Required?
Accept	String	'application/json'	Yes

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	If client accepts JSON response
Failure	404 Not Found	If no charger with this charger_id exists
Failure	406 Not Acceptable	If client does not accept application/json data

Response Examples

Success

```
Status: 200 OK
[
   {
      "id": "5711673465765888",
      "year": 1911,
      "make": "Anderson",
      "owner": "107197049100131265837",
      "charger": "5665833682468864",
      "model": "Detroit Electric",
      "self": "https://final-beachlym.appspot.com/cars/5711673465765888"
   }
]
```

Failure

```
Status: 404 Not Found
{
"Error":  "No charger with this charger_id exists"
}
```

Failure

```
Status: 406 Not Acceptable
{
```

```
    "Error": "Client must accept application/json."
}
```

Get Charger for Car

Allows you to get the car assigned to a charger

GET /cars/:car_id/charger/

Request

Request Parameters

Name	Type	Description	Required?
car_id	String	ID of the car	Yes

Request Header

Name	Type	Description	Required?
Accept	String	'application/json'	Yes

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	If client accepts JSON response
Failure	404 Not Found	If no car with this car_id exists
Failure	406 Not Acceptable	If client does not accept application/json data

Response Examples

Success

```
Status: 200 OK
{
    "address": "456 Lame Ln.",
    "connector": "CHAdeMO",
    "owner": "107197049100131265837",
    "output": 62.5,
    "id": "5665833682468864",
    "self": "https://final-beachlym.appspot.com/chargers/5665833682468864"
}
```

Failure

```
Status: 404 Not Found
{
"Error":  "No car with this car_id exists"
}
```

Failure

```
Status: 406 Not Acceptable
{
    "Error": "Client must accept application/json."
}
```

Delete Car from Charger

Allows you remove a car from a charger

Description

A JWT token must be provided that matches the owner of the car, but it does not need to match the owner of the charger. This is because you can only move (drive) cars that you own, but you may use chargers owned by someone else if another car is not already using them.

```
DELETE /chargers/:charger_id/car/:car_id
```

Request

Request Parameters

Name	Type	Description	Required?
charger_id	String	ID of the charger	Yes
car_id	String	ID of the car	Yes

Request Body

None

Response

Response Body Format

Success: No body
Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	Car is unassigned from charger
Failure	401 Unauthorized	If no JWT token or a bad JWT token is provided
Failure	403 Forbidden	If user ID in JWT token does not match car owner
Failure	403 Forbidden	If this car is not parked at this charger
Failure	404 Not Found	If no car with this car_id exists
Failure	404 Not Found	If no charger with this charger_id exists

Response Examples

Failure

```
Status: 403 Forbidden
{
    "Error": "This car is not parked at this charger"
}
```

Failure

```
Status: 403 Forbidden
{
"Error":  "User sub ID in JWT token does not match car owner."
}
```

Failure

```
Status: 404 Not Found
{
"Error": "The specified car doesn't exist."
}
```

Failure

```
Status: 404 Not Found
{
"Error": "The specified charger doesn't exist."
}
```