

## Final Assignment: API Spec

Sariah Bunnell

CS 493: Cloud Application Development

Spring 2021

Oregon State University

Last Update: May 28, 2021. 10:34 am Pacific

**URL Where Deployed:** <https://bunnells493hw.wl.r.appspot.com>

**Login URL:** [https://bunnells493hw.wl.r.appspot.com/ui\\_login](https://bunnells493hw.wl.r.appspot.com/ui_login) (visit above)

Datastore Data Model .....	1
User Model and Endpoints .....	2
View all Users.....	3
Create a Pet.....	5
View a Pet .....	7
View all Pets .....	9
Edit a Pet .....	11
Delete a Pet .....	15
Create a Kennel .....	17
View a Kennel .....	19
View all Kennels .....	21
Edit a Kennel .....	23
Delete a Kennel .....	27
Add a Pet to a Kennel .....	28
Remove a Pet from a Kennel .....	30

### Datastore Data Model

Object Type	Property	Type	Required?	Valid Example
<b>pets</b>  <u>Relationships:</u> many-to-one with kennels,  protected status with users	id name type weight kennel owner	int, auto string string int int object auth0 or oauth value	Yes Yes Yes Yes Yes - null if no kennel Yes	547923 Aspen Dog 20 44587 auth0 61a073f120b 0cf1078e1cfad
<b>kennels</b>  <u>Relationship:</u> one-to-many with pets	id size inside public pets	int, auto string boolean boolean array of ids (int objects)	Yes Yes Yes Yes No - does not exist if pet has never been assigned, null if pets are removed from kennel	44587 Large True False pets:[ {"id": 547923}, {"id": 118462} ]

## User Model

Object Type	Property	Type	Required?	Valid Example
<b>users</b>	email password	address alphanumeric string	Yes Yes	gobb@gmail.com p@\$sw0rd

What is the unique identifier for a user in this data model?	The JWT contains the auth0 token as the 'sub' value which identifies the user.
If a request needs to supply a user identifier, what needs to be specified by the person making the request?	Authorization headers must include the JWT.
How does the application map a supplied JWT to the identifier of a user?	Pets have an owner attribute that holds the user auth0 token which can be compared to the session JWT.
What is the relationship between the user entity and non-user entity?	Users are the owners of the pets and they have the ability to perform CRUD operations on these protected endpoints by providing authorization.

## User Endpoints

[Allows user to create a user account or sign in to an existing one](#)

/ui_login
-----------

[Allows user to view JWT and profile data as well as auth0 value](#)

/dashboard
------------

[Allows user to end session and log out](#)

/logout
---------

## View all Users

Allows user to view all user names and unique ids [unprotected]

GET /users

### Request

Path Parameters

None

Request Body

None

### Response

Response Body Format - MIME

JSON - application/json

### Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	405 Method Not Allowed	If the request verb is not supported, 405 will be returned.
Failure	406 Not Acceptable	If the request is asking for an unsupported format response, this will be the status code.

### Response Examples

#### Success

```
Status: 200 OK
[
  {
    "name": "Sally Morgan",
    "id": auth0|61a073f120b0cf1078e1cfad
  },
  {
    "name": "Betty Frank",
    "id": auth0|83a073k249f0cf1078e1cder
  },
  {
    "name": "Hugh Donovan",
    "id": auth0|864h79190g0cf1078y1redt
  },
]
```

## Failure

Status: 405 Method Not Allowed

```
{  
  "Error": "The request verb is not supported on this URL"  
}
```

Status: 406 Not Acceptable

```
{  
  "Error": "The response content type is not supported by this API"  
}
```

## Create a Pet

Allows user to create a new pet object - [protected]

POST /pets

### Request

Path Parameters

None

Request Body

Required

Request Body Format - MIME

JSON - application/json

### Request JSON Attributes

Name	Type	Description	Required?
id	int	Automatically generated by Datastore	Yes
name	string	Name of animal	Yes
type	string	Type of animal - dog, cat, etc.	Yes
weight	int	Weight of animal in pounds	Yes
owner	token	Pet owner: 'sub' property in JWT	Yes
kennel	string	kennel id with relationship to pet	Yes

### Request Body Example

```
{
  "name": "Peaches",
  "type": "Cat",
  "weight": 12
}
```

### Response

Response Body Format - MIME

JSON - application/json

### Response Statuses

Outcome	Status Code	Notes
Success	201 Created	
Failure	401 Unauthorized	If the request header lacks valid authentication credentials, this will be returned.

Failure	405 Method Not Allowed	If the request verb is not supported, 405 will be returned.
Failure	406 Not Acceptable	If the request is asking for an unsupported format response, this will be the status code.

#### Response Examples

- Datastore will automatically generate an ID and store it with the entity being created. The app will send back this value in the response body.
- The **self** attribute will contain the live link to the REST resource corresponding to this pet. This URL will not be stored in Datastore but will be returned in the response body.
- The **kennel** attribute will contain the kennel id with a relationship to this pet. This will be null if no relationship exists, and is automatically applied in the API.
- The **owner** attribute is the owner of the pet (aka user id) and is handled by the API with data from the JWT header. The app will send back this value in the response body.

#### Success

Status: 201 Created

```
{
  "id": 123,
  "name": "Peaches",
  "type": "Cat",
  "weight": 12,
  "self": "https://appspot.com/pets/123",
  "owner": auth0|61a073f120b0cf1078e1cfad
  "kennel": null
}
```

#### Failure

Status: 401 Unauthorized

```
{
  "Error": "Incorrect authentication credentials"
}
```

Status: 405 Method Not Allowed

```
{
  "Error": "The request verb is not supported on this URL"
}
```

Status: 406 Not Acceptable

```
{
  "Error": "The response content type is not supported by this API"
}
```

## View a Pet

Allows user to view a pet object - [protected]

GET /pets/:pet\_id

### Request

#### Path Parameters

Name	Type	Description
pet_id	int	Identifier of the pet

### Request Body

None

### Request Body Format - MIME

JSON - application/json

### Response

#### Response Body Format

JSON

#### Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	If the request header lacks valid authentication credentials, this will be returned.
Failure	405 Method Not Allowed	If the request verb is unsupported, 405 will be returned.
Failure	406 Not Acceptable	If the request is asking for an unsupported format response, this will be the status code.

### Response Examples

#### Success

Status: 200 OK

```
{
  "id": 123,
  "name": "Peaches",
  "type": "Cat",
```

```
"kennel": 5678,  
"weight": 12,  
"self": "https://appspot.com/pets/123",  
"owner": auth0|61a073f120b0cf1078e1cfad  
"kennel": null  
}
```

#### Failure

Status: 401 Unauthorized

```
{  
  "Error": "Incorrect authentication credentials"  
}
```

Status: 405 Method Not Allowed

```
{  
  "Error": "The request verb is not supported on this URL"  
}
```

Status: 406 Not Acceptable

```
{  
  "Error": "The response content type is not supported by this API"  
}
```



## View all Pets

This allows the user to view the list of pet objects that he or she owns - paginated to 5 pets per page with the next attribute containing the next page url. [protected]

GET /pets

### Request

#### Path Parameters

None

#### Request Body

None

### Response

#### Response Body Format - MIME

JSON - application/json

#### Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	If the request header lacks valid authentication credentials, this will be returned.
Failure	405 Method Not Allowed	If the request verb is not supported, 405 will be returned.
Failure	406 Not Acceptable	If the request is asking for an unsupported format response, this will be the status code.

#### Response Examples

- The **total** attribute is the entire collection of pet objects, whether by this owner or not. It is handled by the API and sent back in the response body.

#### Success

```
Status: 200 OK
[
  {
    "id": 123,
    "name": "Peaches",
    "type": "Cat",
    "kennel": 5678,
    "weight": 12,
```

```
"self": "https://appspot.com/pets/123",
"owner": auth0|61a073f120b0cf1078e1cfad
},
{
  "id": 864,
  "name": "Aspen",
  "type": "Dog",
  "kennel": 4386,
  "weight": 46,
  "self": "https://appspot.com/pets/864",
  "owner": auth0|61a073f120b0cf1078e1cfad
},
{
  "id": 952,
  "name": "Bijou",
  "type": "Bird",
  "kennel": 7135,
  "weight": 4,
  "self": "https://appspot.com/pets/952",
  "owner": auth0|61a073f120b0cf1078e1cfad
},
"total": 3
]
```

#### Failure

Status: 401 Unauthorized

```
{
  "Error": "Incorrect authentication credentials"
}
```

Status: 405 Method Not Allowed

```
{
  "Error": "The request verb is not supported on this URL"
}
```

Status: 406 Not Acceptable

```
{
  "Error": "The response content type is not supported by this API"
}
```

## Edit a Pet

Allows user to change attributes of a pet except for id - [protected]

PATCH /pets/:pet\_id

### Request

#### Path Parameters

Name	Type	Description
pet_id	int	Identifier of the pet

### Request Body

any 1 or 2 together are required

```
{
  "name": "Sammy",
  "type": "Rabbit"
}
```

### Request Body Format - MIME Types

JSON - application/json

#### Request JSON Attributes

Name	Type	Description	Required?
id	int	Previously generated, does not change	Yes
name	string	Name of the pet	Possibly
type	string	Type of pet	Possibly
weight	int	Weight of pet in pounds	Possibly

### Response

#### Response Body Format - MIME Types

JSON - application/json

#### Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	If the request header lacks valid authentication credentials, this will be returned.

Failure	403 Forbidden	If the request has 3 required attributes or the attributes are improperly named, 403 will be returned.
Failure	405 Method Not Allowed	If the request verb is unsupported, 405 will be returned.
Failure	406 Not Acceptable	If the request is asking for an unsupported format response, this will be the status code.

## Response Examples

### Success

Status: 200 OK

```
{
  "id": 123,
  "name": "Sammy",
  "type": "Rabbit",
  "weight": 12,
  "self": "https://appspot.com/pets/123",
  "owner": "auth0|61a073f120b0cf1078e1cfad",
  "kennel": null
}
```

### Failure

Status: 401 Unauthorized

```
{
  "Error": "Incorrect authentication credentials"
}
```

Status: 403 Forbidden

```
{
  "Error": "Request attributes are missing or incorrect"
}
```

Status: 405 Method Not Allowed

```
{
  "Error": "The request verb is not supported on this URL"
}
```

Status: 406 Not Acceptable

```
{
  "Error": "The response content type is not supported by this API"
}
```

---

PUT /pets/:pet\_id

## Request

### Path Parameters

Name	Type	Description
pet_id	int	Identifier of the boat

### Request Body

[must have all attributes](#)

```
{
  "name": "Sammy",
  "type": "Rabbit",
  "weight": 9
}
```

### Request Body Format - MIME Types

[JSON - application/json](#)

### Request JSON Attributes

Name	Type	Description	Required?
id	int	Previously generated, does not change	Yes
name	string	Name of the pet	Yes
type	string	Type of pet	Yes
weight	int	Weight of pet in pounds	Yes

## Response

### Response Body Format - MIME Types

[JSON - application/json](#)

### Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	If the request header lacks valid authentication

		credentials, this will be returned.
Failure	405 Method Not Allowed	If the request verb is unsupported, 405 will be returned.
Failure	406 Not Acceptable	If the request is asking for an unsupported format response, this will be the status code.

## Response Examples

### Success

Status: 200 OK

```
{
  "id": 123,
  "name": "Sammy",
  "type": "Rabbit",
  "weight": 9,
  "self": "https://appspot.com/pets/123",
  "owner": "auth0|61a073f120b0cf1078e1cfad",
  "kennel": null
},
```

### Failure

Status: 401 Unauthorized

```
{
  "Error": "Incorrect authentication credentials"
}
```

Status: 405 Method Not Allowed

```
{
  "Error": "The request verb is not supported on this URL"
}
```

Status: 406 Not Acceptable

```
{
  "Error": "The response content type is not supported by this API"
}
```

## Delete a Pet

Allows a user to delete a pet object [protected]. Deletion of a pet will change the assigned kennels back to empty.

```
DELETE /pets/:pet_id
```

### Request

#### Path Parameters

Name	Type	Description
pet_id	int	Identifier of the pet

### Request Body

None

### Response

#### Response Body Format

Success: No body

Failure: JSON

#### Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	401 Unauthorized	If the request header lacks valid authentication credentials, this will be returned.
Failure	405 Method Not Allowed	If the request verb is unsupported, 405 will be returned.

### Response Examples

#### Success

```
Status: 204 No Content
```

#### Failure

```
Status: 401 Unauthorized
```

```
{  
  "Error": "Incorrect authentication credentials"  
}
```

Status: 405 Method Not Allowed

```
{  
  "Error": "The request verb is not supported on this URL"  
}
```



## Create a Kennel

Allows user to create a new kennel object

POST /kennels

Request

Path Parameters

None

Request Body

Required

Request Body Format - MIME

JSON - application/json

Request JSON Attributes

Name	Type	Description	Required?
id	int	Automatically generated by Datastore	Yes
size	string	Container size - small, medium, large	Yes
inside	bool	False if location is outside, else true	Yes
public	bool	False if pet door must be kept closed, otherwise the pets can mingle in the public area and the kennel door is open	Yes

Request Body Example

```
{
  "size": "large",
  "inside": true,
  "public": true
}
```

Response

Response Body Format - MIME

JSON - application/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 kennel will not be created, and a 400 status code will be returned.
Failure	405 Method Not Allowed	If the request verb is not supported, 405 will be returned.
Failure	406 Not Acceptable	If the request is asking for an unsupported format response, this will be the status code.

#### Response Examples

- Datastore will automatically generate an ID and store it with the entity being created. The app will send back this value in the response body.
- The **self** attribute will contain the link corresponding to this kennel. This is the URL of the newly created kennel and will not be stored in Datastore but will be returned in the response body.
- The **pets** attribute will contain any pet ids assigned to this kennel. This is not required, and will be nonexistent if newly created or null if empty of pets.

#### Success

Status: 201 Created

```
{
  "id": 5678,
  "size": "large",
  "inside": true,
  "public": true,
  "self": "https://appspot.com/kennels/5678"
}
```

#### Failure

Status: 400 Bad Request

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

Status: 405 Method Not Allowed

```
{
  "Error": "The request verb is not supported on this URL"
}
```

Status: 406 Not Acceptable

```
{
  "Error": "The response content type is not supported by this API"
}
```

## View a Kennel

Allows user to view a kennel object

GET /kennels/:kennel\_id

### Request

#### Path Parameters

Name	Type	Description
kennel_id	int	Identifier of the kennel

### Request Body

None

### Request Body Format - MIME

JSON - application/json

### Response

### Response Body Format - MIME

JSON - application/json

#### Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	405 Method Not Allowed	If the request verb is not supported, 405 will be returned.
Failure	406 Not Acceptable	If the request is asking for an unsupported format response, this will be the status code.

### Response Examples

#### Success

Status: 200 OK

```
{
  "id": 5678,
  "size": "large",
  "inside": true,
  "public": true,
  "pets": [
    {
```

```
    "id": 1234,
  },
  {
    "id": 4582,
  }
]
"self": "https://appspot.com/kennels/5678"
}
```

#### Failure

Status: 405 Method Not Allowed
{ "Error": "The request verb is not supported on this URL" }
Status: 406 Not Acceptable
{ "Error": "The response content type is not supported by this API" }

## View all Kennels

Allows user to view entire list of kennel objects - paginated to 5 kennels per page and next attribute to url of next page

GET /kennels

### Request

#### Path Parameters

None

#### Request Body

None

### Response

#### Response Body Format - MIME

JSON - [application/json](#)

#### Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	405 Method Not Allowed	If the request verb is not supported, 405 will be returned.
Failure	406 Not Acceptable	If the request is asking for an unsupported format response, this will be the status code.

#### Response Examples

- The **total** attribute is the number of kennel objects. It is handled by the API and sent back in the response body.

#### Success

```
Status: 200 OK
[
  {
    "id": 5678,
    "size": "large",
    "inside": true,
    "public": true,
    "pets": [
      {
        "id": 1234,
      },
    ],
  },
]
```

```
{
  "id": 4582,
}
]
"self": "https://appspot.com/kennels/5678"
}
{
  "id": 5984,
  "size": "medium",
  "inside": false,
  "public": true,
  "pets": [{
    "id": 5643,
  }],
  "self": "https://appspot.com/kennels/5984"
},
"total": 2
]
```

Failure

Status: 405 Method Not Allowed

```
{
  "Error": "The request verb is not supported on this URL"
}
```

Status: 406 Not Acceptable

```
{
  "Error": "The response content type is not supported by this API"
}
```

## Edit a Kennel

Allows user to change attributes of a kennel

PATCH /kennels/:kennel\_id

### Request

#### Path Parameters

Name	Type	Description
kennel_id	int	Identifier of the kennel

### Request Body

any 1 or 2 together are required

```
{
  "inside": false,
  "public": false
}
```

### Request Body Format - MIME Types

JSON - [application/json](#)

#### Request JSON Attributes

Name	Type	Description	Required?
id	int	Previously generated, does not change	Yes
size	string	Container size - small, medium, large	Possibly
inside	bool	False if location is outside, else true	Possibly
public	bool	False if pet door must be kept closed, otherwise the pets can mingle in the public area and the kennel door is open	Possibly

### Response

### Response Body Format - MIME Types

JSON - [application/json](#)

### Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	403 Forbidden	If the request has 3 required attributes or the attributes are improperly named, 403 will be returned.
Failure	404 Not Found	No kennel with this kennel_id exists
Failure	405 Method Not Allowed	If the request verb is unsupported, 405 will be returned.

## Response Examples

### Success

Status: 200 OK

```
{
  "id": 5678,
  "size": "large",
  "inside": false,
  "public": false,
  "pets": [
    {
      "id": 1234,
    },
    {
      "id": 4582,
    }
  ]
}
"self": "https://appspot.com/kennels/5678"
```

### Failure

Status: 403 Forbidden

```
{
  "Error": "Request attributes are missing or incorrect"
}
```

Status: 404 Not Found

```
{
  "Error": "No kennel with this kennel_id exists"
}
```

Status: 405 Method Not Allowed



```
{
  "Error": "The request verb is not supported on this URL"
}
```

PUT /kennels/:kennel\_id

## Request

### Path Parameters

Name	Type	Description
kennel_id	int	Identifier of the kennel

### Request Body

[Required - must have all attributes](#)

```
{
  "size": "small",
  "inside": false,
  "public": false,
}
```

### Request Body Format - MIME Types

[JSON - application/json](#)

### Request JSON Attributes

Name	Type	Description	Required?
id	int	Previously generated, does not change	Yes
size	string	Container size - small, medium, large	Yes
inside	bool	False if location is outside, else true	Yes
public	bool	False if pet door must be kept closed, otherwise the pets can mingle in the public area and the kennel door is open	Yes

## Response

## Response Body Format - MIME Types

[JSON - application/json](#)

### Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	404 Not Found	No kennel with this kennel_id exists
Failure	405 Method Not Allowed	If the request verb is unsupported, 405 will be returned.

### Response Examples

#### Success

Status: 200 OK

```
{
  "id": 5678,
  "size": "small",
  "inside": false,
  "public": false,
  "pets": [
    {
      "id": 1234,
    },
    {
      "id": 4582,
    }
  ]
  "self": "https://appspot.com/kennels/5678"
},
```

#### Failure

Status: 404 Not Found

```
{
  "Error": "No kennel with this kennel_id exists"
}
```

Status: 405 Method Not Allowed

```
{
  "Error": "The request verb is not supported on this URL"
}
```

## Delete a Kennel

Allows a user to delete a kennel object from the Datastore list. This will update any pet kennel attributes to null.

```
DELETE /kennels/:kennel_id
```

### Request

#### Path Parameters

Name	Type	Description
kennel_id	int	Identifier of the kennel

### Request Body

None

### Response

#### Response Body Format

Success: No body

Failure: JSON

#### Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	
Failure	404 Not Found	No kennel with this kennel_id exists

### Response Examples

#### Success

```
Status: 204 No Content
```

#### Failure

```
Status: 404 Not Found
```

```
{
  "Error": "No kennel with this kennel_id exists"
}
```

### Add a Pet to a Kennel

Allows a user to assign a pet to a kennel. A pet cannot be assigned to another kennel if not removed from the first kennel it is assigned to.

PUT /kennels/:kennel\_id/pets/:pet\_id

#### Request

#### Path Parameters

Name	Type	Description
kennel_id	int	Identifier of the kennel
pet_id	int	Identifier of the pet

#### Request Body

None

#### Response

#### Response Body Format

Success: No body

Failure: JSON

#### Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	Succeeds only if a kennel exists with this id, a pet exists with pet id, and the pet and kennel are unassigned.
Failure	403 Forbidden	This pet is already assigned to another kennel. It must be removed first.
Failure	404 Not Found	No kennel with this kennel_id exists, and/or no pet with this pet_id exists

#### Response Examples

##### Success

Status: 204 No Content

##### Failure

Status: 403 Forbidden

{

<pre>"Error": "This pet is already assigned to another kennel" }</pre>
<pre>Status: 404 Not Found  {   "Error": "No kennel with this kennel_id exists, and/or no pet with this pet_id exists" }</pre>

## Remove a Pet from a Kennel

Allows a user to unassign a pet. The kennel and pet will still exist as unassigned.

```
DELETE /kennels/:kennel_id/pets/:pet_id
```

### Request

#### Path Parameters

Name	Type	Description
kennel_id	int	Identifier of the kennel
pet_id	int	Identifier of the pet

### Request Body

None

### Response

#### Response Body Format

Success: No body

Failure: JSON

#### Response Statuses

Outcome	Status Code	Notes
Success	204 No Content	Succeeds only if the pet and kennel are appropriately assigned.
Failure	404 Not Found	This pet is not in this kennel

### Response Examples

#### Success

```
Status: 204 No Content
```

#### Failure

```
Status: 404 Not Found

{
  "Error": "This pet is not in this kennel"
}
```