

Login/Sign-up: <https://portfolio-project-369103.uw.r.appspot.com/login>

Website URL: <https://portfolio-project-369103.uw.r.appspot.com>

Portfolio Project: API Spec

CS 493: Cloud Application Development

Fall 2022

Oregon State University

Change log.....	2
Data Model	3
View All Users	5
View All Stores (Protected)	7
View a Store (Protected)	10
Create a Store (Protected)	12
Edit a Store (One or Two Attributes; Protected)	15
Edit a Store (All Attributes; Protected)	18
Edit All Stores	21
Delete a Store (Protected)	22
Delete All Stores.....	24
View All Products	24
View a Product	27
Create a Product	29
Edit a Product (One or Two Attributes)	31
Edit a Product (All Attributes)	34
Edit All Products	38
Delete a Product	39
Delete All Products.....	40

Change log

Version	Change	Date
1.0	Initial version.	November 18, 2022

Data Model

The app stores three kinds of entities in Datastore: Users, Stores, and Products.

Users

Property	Data Type	Required?	Valid Values	Notes
id	Integer	Yes	N/A (automatically generated)	ID of the user; automatically generated by Datastore
name	String	Yes	N/A (automatically generated by JWT)	Name-ID of the user; The “sub” property of a JWT

Stores

Property	Data Type	Required?	Valid Values	Notes
id	Integer	Yes	N/A (automatically generated)	ID of the store; automatically generated by Datastore
name	String	Yes	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters. The requested name must be unique and not already exist in Datastore for stores.	Name of the store, must be unique across all store names
location	String	Yes	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters.	Location of the store
size	Integer	Yes	Positive integers only; Minimum value is 1, maximum value is 2147483647.	The size of the store in square feet
owner	String	Yes	N/A (automatically set when the user signs in and creates a store)	Owner of the store
stock	Array	Yes	Initial creation: automatically set to null. The array will contain objects formed like the following: [“product_id”: 123, “product_id”: 456]	The stock of products currently in the store

Products

Property	Data Type	Required?	Valid Values	Notes
id	Integer	Yes	N/A (automatically generated)	ID of the product; automatically generated by Datastore
name	String	Yes	String only; spaces and special	Name of the product,

			characters are allowed; minimum length is 1 character, maximum length is 255 characters. The requested name must be unique and not already exist in Datastore for products.	must be unique across all product names
type	String	Yes	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters.	Type of product
description	String	No	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters.	Description of the product
stores	Array	Yes	Initial creation: automatically set to null. The array will contain store objects formed like the following: ["store_id": 789, "store_id": 1011]	A list of stores where the product is currently stocked

Relationships

- Stores will have a property named "stock" that is an array containing product IDs.
- Products will have a property named "stores" that is an array containing store IDs.
- Stores and Products have a many-to-many relationship. A store can have many products, and a product can be sold at many stores.

User Entity Details

- Users will be creating with a third-party authentication service, Auth0.
- The unique identifier of the users will be the "sub" property of their JWT, assigned to the "name" property in Datastore of the User entity.
- No endpoints will require a user ID in its URL.
- Users and Stores have a many-to-one relationship. A user can have many stores, and a store can only have one user. This relationship will be demonstrated using the "owner" property of a Store entity. The "owner" property will be assigned to the "name" property of a User entity.

View All Users (Unprotected)

Allows you to get all users

GET /users

Request

Path Parameters

None

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON or HTML.

Response Examples

Success

Status: 200 OK

```
[
  {
    "id": 123,
    "name": "auth0|028338"
  },
  {
    "id": 345,
    "name": "auth0|564656"
  },
  {
    "id": 678,
    "name": "auth0|236769"
  }
]
```

Failure

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable"
}
```

Comment

- This endpoint will ignore the request body if it is included in the request.
- The “Accept” header of the request must be set to “application/json” The setting “*/*” will not work.

View All Stores (Protected)

Allows you to get all stores that are owned by the user

GET /stores

Request

Path Parameters

None

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	This status will be returned if the user has an invalid or missing JWT.
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON or HTML.

Response Examples

Success

Status: 200 OK
<pre>{ "total_count": 7, "stores": [{ "id": 123, "name": "Lon Lon Ranch", "location": "Hyrule", "size": 5000, "stock": [], "owner": "auth0 028338", "self": "https://appspot.com/stores/123" }, { "id": 456, "name": "Kiki's Bakery", "location": "Ghibli", "size": 100, "stock": [{ "id": 1, "name": "Chocolate Chip", "price": 1.50, "stock": 100 }, { "id": 2, "name": "Vanilla", "price": 1.50, "stock": 100 }, { "id": 3, "name": "Lemon", "price": 1.50, "stock": 100 }] }] }</pre>

```

        "product_id": 100,
        "self": "https://appspot.com/products/100"
    },
    {
        "product_id": 101,
        "self": "https://appspot.com/products/101"
    },
    {
        "product_id": 102,
        "self": "https://appspot.com/products/102"
    }
],
"owner": "auth0|028338",
"self": "https://appspot.com/stores/456"
},
{
    "id": 789,
    "name": "Howl's Castle",
    "location": "Ghibli",
    "size": 5500,
    "stock": [],
    "owner": "auth0|028338",
    "self": "https://appspot.com/stores/789"
},
{
    "id": 1011,
    "name": "Lon Lon Ranch II",
    "location": "Hyrule",
    "size": 5000,
    "stock": [],
    "owner": "auth0|028338",
    "self": "https://appspot.com/stores/1011"
},
{
    "id": 1213,
    "name": "Lon Lon Ranch III",
    "location": "Hyrule",
    "size": 5000,
    "stock": [],
    "owner": "auth0|028338",
    "self": "https://appspot.com/stores/1213"
},
],
"next": "https://appspot.com/stores?cursor=urlfornextresults"
}

```


Failure

Status: 401 Unauthorized

```
{  
  "Error": "Unauthorized"  
}
```

Failure

Status: 406 Not Acceptable

```
{  
  "Error": "Not Acceptable"  
}
```

Comment

- This endpoint will ignore the request body if it is included in the request.
- The "Accept" header of the request must be set to "application/json" The setting "*/*" will not work.
- This endpoint implements pagination and will display 5 entities at a time, with a total number of stores being displayed on the first results set.

View a Store (Protected)

Allows user to get an existing store

GET /stores/:store_id

Request

Path Parameters

Name	Description
store_id	ID of the store

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	401 Unauthorized	This status will be returned if the user has an invalid or missing JWT.
Failure	403 Forbidden	This status code will be returned if the store_id is owned by someone else.
Failure	404 Not Found	No store with this store_id exists.
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON or HTML.

Response Examples

Success

Status: 200 OK

```
{
  "id": 123,
  "name": "Lon Lon Ranch",
  "location": "Hyrule",
  "size": 5000,
  "stock": [],
  "owner": "auth0|028338",
  "self": "https://appspot.com/stores/123"
}
```

Failure

Status: 401 Unauthorized

```
{
  "Error": "Unauthorized"
}
```

Failure

Status: 403 Forbidden

```
{  
  "Error": "This store is owned by another user"  
}
```

Failure

Status: 404 Not Found

```
{  
  "Error": "No store with this store_id exists"  
}
```

Failure

Status: 406 Not Acceptable

```
{  
  "Error": "Not Acceptable"  
}
```

Comment

- This endpoint will ignore the request body if it is included in the request.
- The "Accept" header of the request must be set to "application/json" or "text/html" The setting "*/*" will not work.

Create a Store (Protected)

Allows the user to create a new store.

POST /stores

Request

Path Parameters

None

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?	Restrictions
name	The name of the store.	Yes	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters. The requested name must be unique and not already exist in Datastore for stores.
location	The location of the store.	Yes	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters.
size	Size of the store in square feet.	Yes	Positive integers only; Minimum value is 1, maximum value is 2147483647.

Request Body Example

```
{
  "name": "Lon Lon Ranch",
  "location": "Hyrule",
  "size": 5000
}
```

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 store will not be created, and 400 status code will be returned. This status code will also be returned if there are any invalid inputs for the attributes or if there are any extraneous attributes.
Failure	401 Unauthorized	This status will be returned if the user has an invalid or missing JWT.

Failure	403 Forbidden	This status code will be returned if the name of the store already exists in Datastore/is not unique.
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON.

Response Examples

Success

Status: 201 Created

```
{
  "id": 123,
  "name": "Lon Lon Ranch",
  "location": "Hyrule",
  "size": 5000,
  "stock": [],
  "owner": "auth0|028338",
  "self": "https://appspot.com/stores/123"
}
```

Failure

Status: 400 Bad Request

```
{
  "Error": "The request object is missing at least one of the required attributes or contains an invalid input for an attribute"
}
```

Failure

Status: 401 Unauthorized

```
{
  "Error": "Unauthorized"
}
```

Failure

Status: 403 Forbidden

```
{
  "Error": "The requested name attribute is already assigned to another store"
}
```

Failure

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable"
}
```

Comment

- Capitalization does not matter for names in terms of uniqueness. For example, the name "Odyssey" is unique from "odyssey"

- The “Accept” header of the request must be set to “application/json” The setting “*/*” will not work.

Edit a Store (One or Two Attributes; Protected)

Allows you to edit one or two attributes of an existing store.

PATCH /stores/:store_id

Request

Path Parameters

Name	Description
store_id	ID of the store

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?	Restrictions
name	The name of the store.	No	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters. The requested name must be unique and not already exist in Datastore for stores.
location	The location of the store.	No	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters.
size	Size of the store in square feet.	No	Positive integers only; Minimum value is 1, maximum value is 2147483647.

Request Body Example

```
{
  "name": "Armor Land"
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	400 Bad Request	This status code will be returned if there are any invalid inputs for the attributes, if all three attributes of a store are included in the request body, if no attributes are in the request body, or if there are any extraneous attributes.
Failure	401 Unauthorized	This status will be returned if the user has an invalid or missing JWT.

Failure	403 Forbidden	This status code will be returned if the name of the store already exists in Datastore/is not unique or is owned by another user.
Failure	404 Not Found	No store with this store_id exists
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON.

Response Examples

Success

Status: 200 OK

```
{
  "id": 123,
  "name": "Armor Land",
  "location": "Hyrule",
  "size": 5000,
  "stock": [],
  "owner": "auth0|028338",
  "self": "https://appspot.com/stores/123"
}
```

Failure

Status: 400 Bad Request

```
{
  "Error": "The request object does not contain at least one attribute, contains all attributes of the entity, or contains an invalid input for an attribute"
}
```

Failure

Status: 401 Unauthorized

```
{
  "Error": "Unauthorized"
}
```

Failure

Status: 403 Forbidden

```
{
  "Error": "The requested name attribute already assigned to another store or is owned by another user"
}
```

Failure

Status: 404 Not Found

```
{
  "Error": "No store with this store_id exists"
}
```


Failure

Status: 406 Not Acceptable

```
{  
  "Error": "Not Acceptable"  
}
```

Comment

- Capitalization does not matter for names in terms of uniqueness. For example, the name "Odyssey" is unique from "odyssey"
- The "Accept" header of the request must be set to "application/json" The setting "*/*" will not work.

Edit a Store (All Attributes; Protected)

Allows you to edit all attributes of an existing store.

PUT /stores/:store_id

Request

Request

Path Parameters

Name	Description
store_id	ID of the store

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?	Restrictions
name	The name of the store.	Yes	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters. The requested name must be unique and not already exist in Datastore for stores.
location	The location of the store.	Yes	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters.
size	Size of the store in square feet.	Yes	Positive integers only; Minimum value is 1, maximum value is 2147483647.

Request Body Example

```
{
  "name": "Cookies Inc.",
  "location": "Wonderland",
  "size": 350
}
```

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 store will not be edited, and a 400 status code will be returned. This status code will also be returned if there are any invalid inputs for the attributes or if there are any

		extraneous attributes.
Failure	401 Unauthorized	This status will be returned if the user has an invalid or missing JWT.
Failure	403 Forbidden	This status code will be returned if the name of the store already exists in Datastore/is not unique or is owned by another user.
Failure	404 Not Found	No store with this store_id exists
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON.

Response Examples

Success

Status: 200 OK

```
{
  "id": 123,
  "name": "Cookies Inc.",
  "location": "Wonderland",
  "size": 350,
  "stock": [],
  "owner": "auth0|028338",
  "self": "https://appspot.com/stores/123"
}
```

Failure

Status: 400 Bad Request

```
{
  "Error": "The request object is missing at least one of the required attributes or contains an invalid input for an attribute"
}
```

Failure

Status: 401 Unauthorized

```
{
  "Error": "Unauthorized"
}
```

Failure

Status: 403 Forbidden

```
{
  "Error": "The requested name attribute already assigned to another store or is owned by another user"
}
```

Failure

Status: 404 Not Found

```
{
```

```
"Error": "No store with this store_id exists"
}
```

Failure

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable"
}
```

Comment

- Capitalization does not matter for names in terms of uniqueness. For example, the name “Odyssey” is unique from “odyssey”
- The “Accept” header of the request must be set to “application/json” The setting “*/*” will not work.

Edit All Stores

The functionality to edit all stores at once is not supported.

PUT /stores

Request

Path Parameters

None

Request Body

None

Response

No body

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Failure	405 Method Not Allowed	The functionality to edit all stores at once is not supported.

Response Examples

Failure

Status: 405 Method Not Allowed

```
{
  "Error": "The functionality to edit all entities at once is not supported"
}
```

Comment

- This endpoint will ignore the request body if it is included in the request.

Delete a Store (Protected)

Allows you to delete a store.

DELETE /stores/:store_id

Request

Path Parameters

Name	Description
store_id	ID of the store

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	This status will be returned if the user has an invalid or missing JWT.
Failure	403 Forbidden	This status code will be returned if the store_id is owned by someone else.
Failure	404 Not Found	No store with this store_id exists
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON.

Response Examples

Success

Status: 204 No Content

Failure

Status: 401 Unauthorized

```
{
  "Error": "Unauthorized"
}
```

Failure

Status: 403 Forbidden

```
{
  "Error": "This store is owned by another user"
}
```

Failure

Status: 404 Not Found

```
{  
  "Error": "No store with this store_id exists"  
}
```

Failure

Status: 406 Not Acceptable

```
{  
  "Error": "Not Acceptable"  
}
```

Comment

- This endpoint will ignore the request body if it is included in the request.
- The “Accept” header of the request must be set to “application/json” The setting “*/*” will not work.

Delete All Stores

The functionality to delete all stores is not supported.

DELETE /stores

Request

Path Parameters

None

Request Body

None

Response

No body

Response Body Format

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Failure	405 Method Not Allowed	The functionality to delete all stores is not supported.

Response Examples

Failure

Status: 405 Method Not Allowed

<pre>{ "Error": "The functionality to delete all entities at once is not supported" }</pre>

Comment

- This endpoint will ignore the request body if it is included in the request.

View All Products

Allows you to get all products

GET /products

Request

Path Parameters

None

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON or HTML.

Response Examples

Success

Status: 200 OK

```
{
  "total_count": 3,
  "products": [
    {
      "id": 123,
      "name": "Health Potion",
      "type": "Potion",
      "description": "An item used for healing. Tastes like strawberries!",
      "stores": [],
      "self": "https://appspot.com/products/123"
    },
    {
      "id": 456,
      "name": "Magic Potion",
      "type": "Potion",
      "description": "An item used for restoring magic. Tastes like blueberries!",
      "stores": [],
      "self": "https://appspot.com/products/456"
    },
    {

```

```
{
  "id": 789,
  "name": "Baguette",
  "type": "Bread",
  "description": "",
  "stores": [
    {
      "store_id": 1000,
      "self": "https://appspot.com/stores/1000"
    }
  ],
  "self": "https://appspot.com/products/789"
}
```

Failure

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable"
}
```

Comment

- This endpoint will ignore the request body if it is included in the request.
- The "Accept" header of the request must be set to "application/json" The setting "*/*" will not work.
- This endpoint implements pagination and will display 5 entities at a time, with a total number of products being displayed on the first results set.

View a Product (Unprotected)

Allows user to get an existing product

GET /products/:product_id

Request

Path Parameters

Name	Description
product_id	ID of the product

Request Body

None

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	
Failure	404 Not Found	No product with this product_id exists.
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON or HTML.

Response Examples

Success

Status: 200 OK

```
{
  "id": 123,
  "name": "Health Potion",
  "type": "Potion",
  "description": "An item used for healing. Tastes like strawberries!",
  "stores": [],
  "self": "https://appspot.com/products/123"
}
```

Failure

Status: 404 Not Found

```
{
  "Error": "No product with this product_id exists"
}
```

Failure

Status: 406 Not Acceptable

```
{
```

```
"Error": "Not Acceptable"  
}
```

Comment

- This endpoint will ignore the request body if it is included in the request.
- The "Accept" header of the request must be set to "application/json" or "text/html" The setting "*" will not work.

Create a Product (Unprotected)

Allows user to create a new product.

POST /products

Request

Path Parameters

None

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?	Restrictions
name	The name of the product.	Yes	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters. The requested name must be unique and not already exist in Datastore for products.
type	The type of product.	Yes	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters.
description	The description of the product.	No	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters. If no description is in the request body, is null/undefined, or is an empty string, this attribute will be set to "" automatically.

Request Body Example

```
{
  "name": "Health Potion",
  "type": "Potion",
  "description": "An item used for healing. Tastes like strawberries!"
}
```

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 2 required attributes, the product will not be created, and 400 status code will be returned. This status code will also be returned if

		there are any invalid inputs for the attributes or if there are any extraneous attributes.
Failure	403 Forbidden	This status code will be returned if the name of the store already exists in Datastore/is not unique.
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON.

Response Examples

Success

Status: 201 Created

```
{
  "id": 123,
  "name": "Health Potion",
  "type": "Potion",
  "description": "An item used for healing. Tastes like strawberries!",
  "stores": [],
  "self": "https://appspot.com/products/123"
}
```

Failure

Status: 400 Bad Request

```
{
  "Error": "The request object is missing at least one of the required attributes or contains an invalid input for an attribute"
}
```

Failure

Status: 403 Forbidden

```
{
  "Error": "The requested name attribute already assigned to another product"
}
```

Failure

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable"
}
```

Comment

- Capitalization does not matter for names in terms of uniqueness. For example, the name "Odyssey" is unique from "odyssey"
- The "Accept" header of the request must be set to "application/json" The setting "*/*" will not work.

Edit a Product (One or Two Attributes; Unprotected)

Allows you to edit one or two attributes of an existing product.

PATCH /products/:product_id

Request

Path Parameters

Name	Description
product_id	ID of the product

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?	Restrictions
name	The name of the product.	No	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters. The requested name must be unique and not already exist in Datastore for products.
type	The type of product.	No	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters.
description	The description of the product.	No	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters. If the description is left out of the request body or is null/undefined, the description will remain unchanged. If the description is set to an empty string, the description will update to an empty string.

Request Body Example

```
{
  "name": "Hearty Omelet",
  "description": ""
}
```

Response

Response Body Format

JSON

Response Statuses

Outcome	Status Code	Notes
Success	200 OK	

Failure	400 Bad Request	This status code will be returned if there are any invalid inputs for the attributes, if both the “name” and “type” entities are included, if no attributes are in the request body, or if there are any extraneous attributes.
Failure	403 Forbidden	This status code will be returned if the name of the product already exists in Datastore/is not unique.
Failure	404 Not Found	No product with this product_id exists
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON.

Response Examples

Success

Status: 200 OK

```
{
  "id": 123,
  "name": "Hearty Omelet",
  "type": "Potion",
  "description": "",
  "stores": [],
  "self": "https://appspot.com/products/123"
}
```

Failure

Status: 400 Bad Request

```
{
  "Error": "The request object does not contain at least one attribute, contains all attributes of the entity, or contains an invalid input for an attribute"
}
```

Failure

Status: 403 Forbidden

```
{
  "Error": "The requested name attribute already assigned to another product"
}
```

Failure

Status: 404 Not Found

```
{
  "Error": "No product with this product_id exists"
}
```

Failure

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable"
}
```


Comment

- Capitalization does not matter for names in terms of uniqueness. For example, the name “Odyssey” is unique from “odyssey”
- The “Accept” header of the request must be set to “application/json” The setting “*/*” will not work.

Edit a Product (All Attributes; Unprotected)

Allows you to edit all attributes of an existing product.

PUT /products/:product_id

Request

Request

Path Parameters

Name	Description
product_id	ID of the product

Request Body

Required

Request Body Format

JSON

Request JSON Attributes

Name	Description	Required?	Restrictions
name	The name of the product.	Yes	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters. The requested name must be unique and not already exist in Datastore for products.
type	The type of product.	Yes	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters.
description	The description of the product.	No	String only; spaces and special characters are allowed; minimum length is 1 character, maximum length is 255 characters. If the description is left out of the request body or is null/undefined, the description will remain unchanged. If the description is set to an empty string, the description will update to an empty string.

Request Body Example

```
{
  "name": "Hearty Omelet",
  "type": "Egg Dish",
  "description": "A delicious omelet. Restores health immediately upon consumption."
}
```

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 2 required attributes, the store will not be edited, and a 400 status code will be returned. This status code will also be returned if there are any invalid inputs for the attributes or if there are any extraneous attributes.
Failure	403 Forbidden	This status code will be returned if the name of the product already exists in Datastore/is not unique
Failure	404 Not Found	No product with this product_id exists
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON.

Response Examples

Success

Status: 200 OK
<pre>{ "id": 123, "name": "Hearty Omelet", "type": "Egg Dish", "description": "A delicious omelet. Restores health immediately upon consumption.", "stores": [], "self": "https://appspot.com/products/123" }</pre>

Failure

Status: 400 Bad Request
<pre>{ "Error": "The request object is missing at least one of the required attributes or contains an invalid input for an attribute" }</pre>

Failure

Status: 403 Forbidden
<pre>{ "Error": "The requested name attribute already assigned to another product" }</pre>

Failure

Status: 404 Not Found
<pre>{ "Error": "No product with this product_id exists" }</pre>

Failure

Status: 406 Not Acceptable

```
{  
  "Error": "Not Acceptable"  
}
```

Comment

- Capitalization does not matter for names in terms of uniqueness. For example, the name "Odyssey" is unique from "odyssey"
- The "Accept" header of the request must be set to "application/json" The setting "*/*" will not work.

Edit All Products

The functionality to edit all products is not supported.

PUT /products

Request

Path Parameters

None

Request Body

None

Response

No body

Response Body Format

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Failure	405 Method Not Allowed	The functionality to edit all products is not supported.

Response Examples

Failure

Status: 405 Method Not Allowed

```
{  
  "Error": "The functionality to edit all entities at once is not supported"  
}
```

Comment

- This endpoint will ignore the request body if it is included in the request.

Delete a Product (Unprotected)

Allows you to delete a product.

DELETE /products/:product_id

Request

Path Parameters

Name	Description
product_id	ID of the product

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 product with this product_id exists
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON.

Response Examples

Success

Status: 204 No Content

Failure

Status: 404 Not Found

```
{
  "Error": "No product with this product_id exists"
}
```

Failure

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable"
}
```

Comment

- This endpoint will ignore the request body if it is included in the request.
- The "Accept" header of the request must be set to "application/json" The setting "*"/*" will not work.

Delete All Products

The functionality to delete all products is not supported.

DELETE /products

Request

Path Parameters

None

Request Body

None

Response

No body

Response Body Format

Failure: JSON

Response Statuses

Outcome	Status Code	Notes
Failure	405 Method Not Allowed	The functionality to delete all products is not supported.

Response Examples

Failure

Status: 405 Method Not Allowed

<pre>{ "Error": "The functionality to delete all entities at once is not supported" }</pre>

Comment

- This endpoint will ignore the request body if it is included in the request.

Adding a Product to a Store (Protected)

Allows user to add a product to a store

PATCH /stores/:store_id/products/:product_id

Request

Path Parameters

Name	Description
store_id	ID of the store
product_id	ID of the product

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	This status will be returned if the user has an invalid or missing JWT.
Failure	403 Forbidden	This status code will be returned if the store_id is owned by someone else, or this product is already stocked at the store.
Failure	404 Not Found	The specified store and/or product does not exist.
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON or HTML.

Response Examples

Success

Status: 204 No Content

Failure

Status: 401 Unauthorized

```
{
  "Error": "Unauthorized"
}
```

Failure

Status: 403 Forbidden

```
{
  "Error": "This store is owned by another user or the product is already stocked at this store"
}
```

Failure

Status: 404 Not Found

```
{
  "Error": "No store with this store_id exists or no product with this product_id exists"
}
```

Failure

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable"
}
```

Comment

- This endpoint will ignore the request body if it is included in the request.
- The "Accept" header of the request must be set to "application/json" or "text/html" The setting "*/*" will not work.

Removing a Product from a Store (Protected)

Allows user to remove a product from a store

DELETE /stores/:store_id/products/:product_id

Request

Path Parameters

Name	Description
store_id	ID of the store
product_id	ID of the product

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	This status will be returned if the user has an invalid or missing JWT.
Failure	403 Forbidden	This status code will be returned if the store_id is owned by someone else, or this product is not at the store.
Failure	404 Not Found	The specified store and/or product does not exist.
Failure	406 Not Acceptable	This status code will be returned if a request is sent with the Accept header set to a MIME type that is not JSON or HTML.

Response Examples

Success

Status: 204 No Content

Failure

Status: 401 Unauthorized

```
{
  "Error": "Unauthorized"
}
```

Failure

Status: 403 Forbidden

```
{
  "Error": "This store is owned by another user or the product is not at this store"
}
```

Failure

Status: 404 Not Found

```
{
  "Error": "No store with this store_id exists or no product with this product_id exists"
}
```

Failure

Status: 406 Not Acceptable

```
{
  "Error": "Not Acceptable"
}
```

Comment

- This endpoint will ignore the request body if it is included in the request.
- The "Accept" header of the request must be set to "application/json" or "text/html" The setting "*/*" will not work.