

- [API Requirements](#)
  - [API Endpoints](#)
    - [Main Endpoints Implemented](#)
    - [Products Endpoints Requirements](#)
    - [Products Endpoints Implemented](#)
    - [Users Endpoints Requirements](#)
    - [Users Endpoints Implemented](#)
    - [Orders Endpoints Requirements](#)
    - [Orders Endpoints Implemented](#)
  - [Data Shapes](#)
    - [Product](#)
    - [User](#)
    - [Orders](#)

# API Requirements

---

The company stakeholders want to create an online storefront to showcase their great product ideas. Users need to be able to browse an index of all products, see the specifics of a single product, and add products to an order that they can view in a cart page. You have been tasked with building the API that will support this application, and your coworker is building the frontend.

These are the notes from a meeting with the frontend developer that describe what endpoints the API needs to supply, as well as data shapes the frontend and backend have agreed meet the requirements of the application.

## API Endpoints

---

### Main Endpoints Implemented

- Index

Endpoint	Route	Method	Header	Body	Description
Index	/	GET		--	Welcome message

### Products Endpoints Requirements

- Index
- Show
- Create [token required]
- [OPTIONAL] Top 5 most popular products
- [OPTIONAL] Products by category (args: product category)

Products Endpoints Implemented

Endpoint	Route	Method	Header	Body	Description
Index	/products/	GET	access-token	--	Show all products
Show	/products/:product_ID	GET	access-token	--	Show one product
Create	/products/	POST	access-token	{ "name": "productName", "price": productPrice, "category": "productCategory" }	Create new product
Update	/products/:product_ID	PUT	access-token	{ "name": "productName", "price": productPrice, "category": "productCategory" }	Update product
Delete	/products/:product_ID	DELETE	access-token	--	Delete product
Top 5	/products/top	GET	access-token	--	Show top 5 most popular products

Endpoint	Route	Method	Header	Body	Description
Categories	/products/:category	GET	access-token	--	Show products that belong to the category

access-token is token provided when you logged in.

### Users Endpoints Requirements

- Index [token required]
- Show [token required]
- Create N[token required]

### Users Endpoints Implemented

Endpoint	Route	Method	Header	Body	Description
Index	/users/	GET	access-token	--	Show all users
Show	/users/:user_ID	GET	access-token	--	Show one user
Create	/users/	POST		{           "username": "username",           "firstName": "firstName",           "lastName": "lastName",           "password": "password"         }	Create new user

Endpoint	Route	Method	Header	Body	Description
Update	/users/:user_ID	PUT	access-token	{ "username": "username", "firstName": "firstName", "lastName": "lastName", "password": "password" }	Update user
Delete	/users/:user_ID	DELETE	access-token	--	Delete user
Authentication	/users/auth	POST		{ "username": "username", "password": "password" }	Auth user with username and password to get the token

access-token is token provided when you logged in.

## Orders Endpoints Requirements

- Current Order by user (args: user id)[token required]
- [OPTIONAL] Completed Orders by user (args: user id)[token required]

## Orders Endpoints Implemented

Endpoint	Route	Method	Header	Body	Description
Index	/orders/	GET	access-token	--	Show all Orders
Show	/orders/:order_ID	GET	access-token	--	Show one order

Endpoint	Route	Method	Header	Body	Description
Create	/orders/	POST	access-token	{ "user_id": "user_id", "orderStatus": "orderStatus" }	Create new Order
Update	/orders/:order_ID	PUT	access-token	{ "user_id": "user_id", "orderStatus": "orderStatus" }	Update order
Delete	/orders/:order_ID	DELETE	access-token	--	Delete order
Add product	/orders/add/:order_ID	POST	access-token	{ "product_id": "product_id", "qty": "qty" }	Add product from current order
Update qty	/orders/update/:order_ID	POST	access-token	{ "product_id": "product_id", "qty": "qty" }	Update qty product from current order
Remove product from current order	/orders/remove/:order_ID	POST	access-token	{ "product_id": "product_id", }	Remove product from current order
Complete Order	/orders/:order_ID	PATCH	access-token	{ "orderStatus": "Completed", }	Finish order and change status

Endpoint	Route	Method	Header	Body	Description
Current order by user	/orders/current/:user_ID	POST	access-token	--	Show current order that belongs to user
Completed orders by user	/orders/completed/:user_ID	GET	access-token	--	Show all completed orders that belongs to user

access-token is token provided when you logged in.

## Data Shapes

Next data shapes its about how data will going to be stored in Data Base so next information it's data shapes requirements from Front-End Engineer and below them it's the EER diagram.Data shapes it about how data will going to be stored in DataBase so following information it's data shapes requirements, from Front-End Engineer and bellow its the final diagram.

### Product

- id
- name
- price
- [OPTIONAL] category

### User

- id
- firstName
- lastName
- password

### Orders

- id

- id of each product in the order
- quantity of each product in the order
- user\_id
- status of order (active or complete)

EER Diagram:

```
erDiagram
    Users {
        uuid id PK
        character_varying username
        character_varying firstName
        character_varying lastName
        text password
    }

    Orders {
        uuid id PK
        uuid user_id FK
        character_varying orderStatus
    }

    Products {
        uuid id PK
        character_varying name
        double_precision price
        character_varying category
    }

    ProductsOnOrders {
        uuid product_id PK
        uuid order_id PK
        integer quantity
    }

    Orders }o--|| Users : "user_id"
    ProductsOnOrders }o--|| Orders : "order_id"
    ProductsOnOrders }o--|| Products : "product_id"
```