

API Specification Doc

(Labb 2 Webbprog)

| Version | Date | Author | Description |
|---------|--------|--------------|--------------------------------|
| 1.0 | 230325 | Jens Eresund | Initial draft |
| 1.1 | 310325 | | Amended Customer specification |
| 1.2 | 010425 | | Amended Product specification |
| 1.3 | 010425 | | Amended Order specification |
| 1.4 | 230525 | | Amended Order specification |
| 1.5 | 060625 | | Added Auth specification |

Index

Endpoints

Login

Customers

Request Endpoints

Parameters

Request body

Responses

Products

Request Endpoints

Parameters

Request body

Response

Orders

Request Endpoints

Parameters

Request body

Response

Glossary

Conventions

Status Codes

Endpoints

Login

Authenticate the user with the system and obtain the auth_token

Request Endpoints

| Method | URL | Description |
|--------|------------------------|------------------------------|
| POST | api/auth/register | Register as a new user |
| POST | api/auth/login | Login |
| GET | api/auth/users | Get all users |
| GET | api/auth/users/{email} | Get user with {email} |
| DELETE | api/auth/users/{email} | Remove the user with {email} |

Parameters

| Type | Name | Values |
|----------|-----------------------|-------------------------------------|
| Header | Authorization | Auth-token provided from api/login/ |
| Path | email * | string |
| Property | email * password * | string |
| | role | number [0 - 2] |
| | customerId | integer > 0 |

* required

Authorization

HTTP Authorization header containing the auth-token provided from api/login/.

email

The unique email address of the specific user.

password

The desired or valid password of the specific user.

role

The authorization role of the user: 0 (none), 1 (Admin), 2 (User).

customerId

The unique id number of the users customer profile.

Request body

| Method | Body |
|------------------------------------|---|
| POST /register /login | <pre>{ "email": "john.doe@email.se", "password": "%&123aBc" }</pre> |
| GET (Auth required) | none |
| DELETE (Auth required) | none |

Responses

| Method | Status | Response meaning/Body |
|------------------------------|--------|---|
| POST /register | 201 | "Registration successful." |
| | 409 | "There is already an account associated with this email address." |
| POST /login | 200 | Response will be the generated JWT token to be provided in the Authorization header. |
| GET /users | 200 | Response will be an array of users. (Pagination to be implemented) <pre>{ "users": [{ (as shown in GET /users/{email}) }, ...] }</pre> |
| GET /users/{email} | 200 | Response will be a user object. <pre>{ "email": "john.doe@email.se", "role": 0, "customerId": 1 (only provided if the user has a customer profile) }</pre> |
| DELETE | 204 | none |
| Any | 400 | Request cannot be processed for some reason. |
| | 401 | If client tries to access restricted content, such as customers or orders with insufficient privileges. |
| | 404 | The resource does not exist on the server. |
| | 500 | Something went wrong in the server. Authenticated administrator will receive more detailed error messages. |

Customers

Access customers information. Access to any other than self requires elevated privileges.

Request Endpoints

| Method | URL | Description |
|--------|---------------------------|-------------------------------|
| GET | api/customers/ | Get all customers |
| POST | api/customers/ | Create a new customer |
| GET | api/customers/{id} | Get customer with {id} |
| PATCH | api/customers/{id} | Change the customer with {id} |
| DELETE | api/customers/{id} | Remove the customer with {id} |
| GET | api/customers/{id}/orders | Get the customers orders |

Parameters

| Type | Name | Values |
|----------|--|-------------------------------------|
| Header | Authorization * | Auth-token provided from api/login/ |
| Path | id * | integer > 0 |
| Property | firstname * lastname * email * phone address | string |

* required

Authorization

HTTP Authorization header containing the auth-token provided from api/login/.

id

The unique id number of the specific customer.

firstname

The customers given name.

lastname

The customers surname.

email

The customers email address.

phone

The customers phone number.

address

The customers billing address.

Request body

| Method | Body |
|--------|--|
| GET | none |
| POST | <pre>{ "firstname": "John", "lastname": "Doe", "email": "john.doe@email.se", "phone": "+4610234567", "address": "Street 123, 456 78 City, Country" }</pre> |
| PATCH | <p>Request is case sensitive, all properties use CamelCase.</p> <pre>{ "LastName": "Smith", "Address" : "OtherStreet 321, 654 87 City, Country", ... }</pre> |
| DELETE | none |

Responses

| Method | Status | Response meaning/Body |
|--------|--------|---|
| GET | 200 | <p>Response will be an array of customers. (Pagination to be implemented)</p> <pre>{ "customers": [{ (as shown in GET {id}) }, ...] }</pre> |
| | 204 | There are no customers in the database. |
| POST | 201 | <p>Response will be the created customer object along with the location header.</p> <pre>{ "id": 1, "firstname": "John", "lastname": "Doe", "email": "john.doe@email.se", "phone": "+4610234567", "address": "Street 123, 456 78 City, Country" }</pre> |

| Method | Status | Response meaning/Body |
|---------------------------|--------|--|
| GET {id} | 200 | Response will be a customer object. <pre>{ "id": 1, "firstname": "John", "lastname": "Doe", "email": "john.doe@email.se", "phone": "+4610234567", "address": "Street 123, 456 78 City, Country", "orders": [{ "id": 1, "customerid": 1, "status": 0 }, ...] }</pre> |
| PATCH | 200 | Response will be the modified customer object. <pre>{ "id": 1, "firstname": "John", "lastname": "Doe", "email": "john.doe@newemail.se", "phone": "+4610234567", "address": "OtherStreet 321, 654 87 City, Country" }</pre> |
| DELETE | 204 | none |
| GET {id}/orders | 200 | Response will be an array of order objects. (See Orders) <pre>{ "orders": [{ Order (See Orders) }, ...] }</pre> |
| | 204 | Customer has no orders. |
| Any | 400 | Request cannot be processed for some reason. |
| | 401 | If client tries to access restricted content, such as customers or orders with insufficient privileges. |
| | 404 | The resource does not exist on the server. |
| | 500 | Something went wrong in the server. Authenticated administrator will receive more detailed error messages. |

Products

Access products information. Access to orders the product is part of requires elevated privileges.

Request Endpoints

| Method | URL | Description |
|--------|--------------------------|-------------------------|
| GET | api/products/ | Get all products |
| POST | api/products/ | Create a new product |
| GET | api/products/{id} | Get product with {id} |
| PATCH | api/products/{id} | Change the product |
| DELETE | api/products/{id} | Remove the product |
| GET | api/products/{id}/orders | Get the products orders |

Parameters

| Type | Name | Values |
|----------|---------------------------------------|-------------------------------------|
| Header | Authorization | Auth-token provided from api/login/ |
| Path | id * | integer > 0 |
| Property | name * description * category * | string |
| | price * stock * | number |
| | status * | number [0 - 1] |

* required

Authorization

HTTP Authorization header containing the auth-token provided from `api/login`

id

The unique id number of the specific product.

name

The products full name.

description

A description of the product.

category

A comma separated string of the category/ies of the product i.e: "`cat1,cat2,...`"

price

The products price.

stock

The number of items available for sale.

status

The availability of the product: `0` (Active), `1` (Discontinued).

Request body

| Method | Body |
|---------------------------|---|
| GET | none |
| POST (Auth required) | <pre>{ "name": "Itemname", "description": "A very good product.", "category": "Categoryname", "price": 9.99, "stock": 100 }</pre> |
| PATCH (Auth required) | <p>Request is case sensitive, all properties use CamelCase.</p> <pre>{ "Description": "A very good product.", "Stock": 99, ... }</pre> |
| DELETE (Auth required) | none |

Response

| Method | Status | Response meaning/Body |
|-------------|--------|--|
| GET | 200 | Response will be an array of products. (Pagination to be implemented) <pre>{ "products": [{ (as shown in GET {id}) }, ...] }</pre> |
| POST | 201 | Response will be the created product object along with the location header. <pre>{ "id": 1, "name": "Itemname", "description": "A very good product.", "category": "Categoryname", "price": 9.99, "status": 0, "stock": 100, }</pre> |
| GET {id} | 200 | Response will be a product object. <pre>{ "id": 1, "name": "Itemname", "description": "A very good product.", "category": "Categoryname", "price": 9.99, "status": 0, "stock": 100 }</pre> |
| PATCH | 200 | Response will be the modified product object. <pre>{ "id": 1, "name": "Newname", "description": "Very new description.", "category": "Newcategoryname,Addedcategory", "price": 13.99, "status": 0, "stock": 99 }</pre> |

| Method | Status | Response meaning/Body |
|--------------------|--------|--|
| DELETE | 204 | none |
| GET {id}/orders | 200 | <p>Response will be a an array of objects with order details for the product.</p> <pre> { "orders": [{ "path": "api/orders/7", "count": 1, "price": 8.99, "orderId": 1, }, ...] } </pre> |
| All | 400 | Request cannot be processed for some reason. |
| | 401 | If client tries to access restricted content, such as customers or orders with insufficient privileges. |
| | 404 | The resource does not exist on the server. |
| | 500 | Something went wrong in the server. Authenticated administrator will receive more detailed error messages. |

Orders

Access orders information. Requires elevated privileges, unless the orders are owned by the user.

Request Endpoints

| Method | URL | Description |
|--------|--------------------------|---|
| GET | api/orders/ | Get all orders |
| POST | api/orders/ | Create a new order |
| GET | api/orders/{id} | Get order with {id} |
| PATCH | api/orders/{id} | Change the order status |
| DELETE | api/orders/{id} | Remove the order |
| GET | api/orders/{id}/products | Get the products in the order |
| PATCH | api/orders/{id}/products | Set the product(s) amounts in the order |
| PUT | api/orders/{id}/products | Replace all products in the order |

Parameters

| Type | Name | Values |
|----------|-----------------|---|
| Header | Authorization * | Auth-token provided from api/login/ |
| Path | id | integer > 0 |
| | pid | |
| Property | customerid * | |
| | status * | number [0 - 3] |
| | products * | array[n][2] where each row is [productid, count] [[1, 3], ...] |

* required

Authorization

HTTP Authorization header containing the auth-token provided from api/login/.

id

The unique id number of the specific order.

pid

The unique id number of a product in the order.

customerid

The unique id number of the owner of the order.

status

The status of the order: 0 (New), 1 (Processing), 2 (Shipped), 3 (Delivered).

products

A list of product-ids and the number of them in the order. The list with full product-info is accessed at [GET api/orders/{id}/products].

Request body

| Method | Body |
|-------------------------|--|
| GET | none |
| POST | <pre>{ "customerid": 1, "products": [[1, 3], ...] }</pre> |
| PATCH /{id} | <pre>{ "status": 1 }</pre> |
| PATCH /{id}/products | Sets the specified product ids to the specified amounts in the order. <pre>{ "products": [[2, 1], ...] }</pre> |
| PUT /{id}/products | Replace all products in the order with those specified. <pre>{ "products": [[1, 1], ...] }</pre> |
| DELETE | none |

Response

| Method | Status | Response meaning/Body |
|----------------|--------|--|
| GET | 200 | Response will be an array of orders. (Pagination to be implemented) <pre>{ "orders": [{ (as shown in GET {id}) }, ...] }</pre> |
| POST | 201 | Response will be the created order object along with the location header. <pre>{ "id": 1, "customerid": 1, "status": 0, "products": [[1, 3], ...] }</pre> |
| | 409 | The customer already has an active cart which will is returned along with a "X-Conflict-Message" header. |
| GET /{id} | 200 | Response will be an order object. <pre>{ "id": 1, "customerid": 1, "status": 0, "products": [[1, 3], ...] }</pre> |
| PATCH /{id} | 200 | Response will be the modified order object. <pre>{ "id": 1, "customerid": 1, "status": 1, "products": [[1, 3], ...] }</pre> |
| DELETE | 204 | none |

| Method | Status | Response meaning/Body |
|--------------------------------|--------|---|
| GET /{id}/products | 200 | Response will be a an array of wrapper objects. <pre>{ "products": [{ "path": "api/products/1", "count": 3, "price": 8.99, "product": { Product (See Products) }, ... }] }</pre> |
| PATCH /{id}/products | 200 | Response will be the resulting product array of the order. <pre>{ "products": [[1, 3], [2, 1], ...] }</pre> |
| PUT /{id}/products | | |
| | 403 | Order content cannot be changed after it has been processed. |
| All | 400 | Request cannot be processed for some reason. |
| | 401 | An unauthenticated client tries to access restricted content, such as customers or orders. Insufficient privileges. |
| | 404 | The resource does not exist on the server. |
| | 500 | Something went wrong in the server. Authenticated administrator will receive more detailed error messages. |

Glossary

Conventions

- **Client** - Client application.
- **Status** - HTTP status code of response.
- All the possible responses are listed under 'Responses' for each method. Only one of them is issued per request server.
- All response are in JSON format.

Status Codes

All status codes are standard HTTP status codes. The below ones are used in this API.

2XX - Success of some kind

4XX - Error occurred in client's part

5XX - Error occurred in server's part

| Status Code | Description |
|-------------|---------------------------------------|
| 200 | OK |
| 201 | Created |
| 204 | No Content |
| 400 | Bad request |
| 401 | Authentication failure/Not Authorized |
| 403 | Forbidden |
| 404 | Resource not found |
| 409 | Conflict |
| 500 | Internal Server Error |
| 503 | Service Unavailable |