**Spring API Documentation for Ecommerce Application**

This documentation provides an overview of the Spring API endpoints for an ecommerce application, including user registration and login functionality. It covers the different endpoints, their purposes, and the required request parameters and response formats.

Base URL

The base URL for accessing the ecommerce API is http://localhost:8888/health

Authentication

To access protected endpoints, you need to include an authentication token in the request headers. After successful user registration or login, the API will return an authentication token that should be included in subsequent requests.

Request Headers

Authorization: Bearer <authentication\_token>

Endpoints

User Registration

Register a new user

Endpoint: POST

/users/register

Description: Registers a new user with the ecommerce application.

Request Body:

{

"username": "john\_doe",

"email": "john@example.com",

"password": "password123"

}

Response:

201 Created on success with no response body.

400 Bad Request if the request body is invalid.

409 Conflict if the username or email is already taken.

User Login

Authenticate a user

Endpoint: POST /users/login

Description: Authenticates a user and returns an authentication token.

Request Body:

{

"username": "john\_doe",

"password": "password123"

}

Response:

200 OK on success

400 Bad Request if the request body is invalid.

401 Unauthorized if the username or password is incorrect.

Product Endpoints

Retrieve all products

Endpoint: GET /products

Description: Retrieves a list of all products available in the ecommerce application.

Response:

200 OK on success with the following response body:

[

{

"id": 1,

"name": "Product 1",

"price": 19.99,

"description": "Product 1 description"

},

{

"id": 2,

"name": "Product 2",

"price": 29.99,

"description": "Product 2 description"

}

]

401 Unauthorized if the authentication token is missing or invalid.

Retrieve a product by ID

Endpoint: GET /products/{id}

Description: Retrieves a specific product by its ID.

Response:

200 OK

{

"id": 1,

"name": "Product 1",

"price": 19.99,

"description": "Product 1 description"

}

401 Unauthorized if the authentication token is missing or invalid.

404 Not Found if the product with the specified ID does not exist.

Cart Endpoints

Add a product to the cart

Endpoint: POST /cart/items

Description: Adds a product to the user's shopping cart.

Request Body:

{

"productId": 1,

"quantity": 2

}

Response:

200 OK on success with no response body.

400 Bad Request if the request body is invalid.

401 Unauthorized if the authentication token is missing or invalid.

404 Not Found if the product with the specified ID does not exist.

Retrieve the user's cart

Endpoint: GET /cart

Description: Retrieves the contents of the user's shopping cart.

Response:

200 OK on success with the following response body:

{

"items": [

{

"id": 1,

"product": {

"id": 1,

"name": "Product 1",

"price": 19.99,

"description": "Product 1 description"

},

"quantity": 2

},

{

"id": 2,

"product": {

"id": 2,

"name": "Product 2",

"price": 29.99,

"description": "Product 2 description"

},

"quantity": 1

}

],

"totalPrice": 69.97

}

401 Unauthorized if the authentication token is missing or invalid.

Remove a product from the cart

Endpoint: DELETE /cart/items/{id}

Description: Removes a product from the user's shopping cart.

Response:

200 OK on success with no response body.

401 Unauthorized if the authentication token is missing or invalid.

404 Not Found if the cart item with the specified ID does not exist.