

Dokumentacja API ReactShop

Wprowadzenie

Dokumentacja opisuje sposób korzystania z API ReactShop. Każdy endpoint jest szczegółowo opisany wraz z przykładami żądań i odpowiedzi.

1 Rejestracja użytkownika

- **Endpoint:** POST /api/users/signup
- **Opis:** Rejestruje nowego użytkownika w systemie.
- **Nagłówki:**

```
1 Content-Type: application/json
2
```

- **Parametry (body):**

```
1 {
2   "username": "string", // Wymagane
3   "password": "string", // Wymagane
4   "email": "string",    // Wymagane
5   "isAdmin": "boolean", // Opcjonalne, domylnie false
6   "permissions": ["string"] // Opcjonalne, domylnie ["read"]
7 }
8
```

- **Przykładowa odpowiedź (201 Created):**

```
1 {
2   "message": "User registered successfully.",
3   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
4   "isAdmin": false
5 }
6
```

2 Logowanie użytkownika

- **Endpoint:** POST /api/users/login
- **Opis:** Loguje użytkownika do systemu i zwraca token JWT.
- **Nagłówki:**

```
1 Content-Type: application/json
2
```

- **Parametry (body):**

```
1 {
2   "username": "string", // Wymagane
3   "password": "string"  // Wymagane
4 }
5
```

- Przykładowa odpowiedź (200 OK):

```

1 {
2   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
3   "isAdmin": false
4 }
5

```

3 Chroniona trasa

- Endpoint: GET /api/protected
- Opis: Zwraca dane użytkownika po weryfikacji tokenu JWT.
- Nagłówki:

```

1 Authorization: Bearer <token>
2

```

- Przykładowa odpowiedź (200 OK):

```

1 {
2   "message": "This is a protected route.",
3   "user": {
4     "_id": "65a1b2c3d4e5f6g7h8i9j0",
5     "username": "testuser",
6     "email": "test@example.com",
7     "isAdmin": false,
8     "permissions": ["read"]
9   }
10 }
11

```

4 Pobierz koszyk użytkownika

- Endpoint: GET /api/users/:username/cart
- Opis: Zwraca zawartość koszyka dla danego użytkownika.
- Nagłówki:

```

1 Authorization: Bearer <token>
2

```

- Przykładowa odpowiedź (200 OK):

```

1 []
2

```

5 Aktualizuj koszyk użytkownika

- Endpoint: POST /api/users/:username/cart
- Opis: Aktualizuje zawartość koszyka dla danego użytkownika.
- Nagłówki:

```

1 Authorization: Bearer <token>
2 Content-Type: application/json
3

```

- Parametry (body):

```

1 {
2   "cart": [
3     {
4       "itemId": "string",
5       "quantity": "number"
6     }
7   ]
8 }
9

```

- **Przykładowa odpowiedź (200 OK):**

```

1 {
2   "message": "Cart updated successfully"
3 }
4

```

6 Pobierz historie zakupów użytkownika

- **Endpoint:** GET /api/users/:username/purchases
- **Opis:** Zwraca historie zakupów dla danego użytkownika.
- **Nagłówki:**

```

1 Authorization: Bearer <token>
2

```

- **Przykładowa odpowiedź (200 OK):**

```

1 []
2

```

7 Pobierz liste produktów

- **Endpoint:** GET /api/items
- **Opis:** Zwraca liste wszystkich produktów.
- **Przykładowa odpowiedź (200 OK):**

```

1 [
2   {
3     "_id": "65a1b2c3d4e5f6g7h8i9j0",
4     "title": "Product 1",
5     "price": 100,
6     "quantity": 10
7   }
8 ]
9

```

8 Kup produkt

- **Endpoint:** POST /api/items/buy
- **Opis:** Realizuje zakup produktu i aktualizuje stan magazynowy.
- **Nagłówki:**

```

1 Authorization: Bearer <token>
2 Content-Type: application/json
3

```

- **Parametry (body):**

```

1 {
2   "title": "string", // Wymagane
3   "quantity": "number", // Wymagane
4   "username": "string" // Wymagane
5 }
6

```

- **Przykładowa odpowiedź (200 OK):**

```

1 {
2   "message": "Purchase successful"
3 }
4

```

9 Pobierz opinie o produkcie

- **Endpoint:** GET /api/items/:title/opinion
- **Opis:** Zwraca liste opinii dla danego produktu.
- **Nagłówki:**

```

1 Authorization: Bearer <token>
2

```

- **Przykładowa odpowiedź (200 OK):**

```

1 []
2

```

10 Dodaj opinie o produkcie

- **Endpoint:** POST /api/items/:title/opinion
- **Opis:** Dodaje opinie do danego produktu.
- **Nagłówki:**

```

1 Authorization: Bearer <token>
2 Content-Type: application/json
3

```

- **Parametry (body):**

```

1 {
2   "author": "string", // Wymagane
3   "content": "string", // Wymagane
4   "rating": "number" // Wymagane (1-5)
5 }
6

```

- **Przykładowa odpowiedź (201 Created):**

```

1 {
2   "message": "Opinion added successfully."
3 }
4

```

11 Usuń opinie o produkcie

- **Endpoint:** DELETE /api/items/:title/opinion
- **Opis:** Usuwa opinie danego użytkownika o produkcie.
- **Nagłówki:**

```
1 Authorization: Bearer <token>  
2 Content-Type: application/json  
3
```

- **Parametry (body):**

```
1 {  
2   "author": "string" // Wymagane  
3 }  
4
```

- **Przykładowa odpowiedź (200 OK):**

```
1 {  
2   "message": "Opinion deleted successfully."  
3 }  
4
```