

API Documentation – Sabhyasha E-commerce Backend

This documentation covers all available endpoints for the Sabhyasha E-commerce backend built using Django, DRF, JWT, and MySQL.

Authentication

All protected routes require an access token in the header:

```
Authorization: Bearer <access_token>
```

Accounts APIs (3)

1. Register

POST /accounts/register/

Registers a new user.

```
{
  "email": "user@example.com",
  "username": "user123",
  "password": "testpass123",
  "is_buyer": true,
  "is_seller": false
}
```

2. Login

POST /accounts/login/

Returns access and refresh tokens.

```
{
  "email": "user@example.com",
  "password": "testpass123"
}
```

3. Refresh

POST /accounts/refresh/

Refreshes the access token.

```
{
  "refresh": "<your_refresh_token>"
}
```



Products APIs (2)

1. List Products

GET /products/

Public endpoint with optional `search` and `ordering=price`.

2. Add Product

POST /products/add/

```
{
  "name": "Cotton Kurta",
  "description": "Soft cotton material",
  "price": "799.00",
  "stock": 20
}
```



Requires seller authentication.

Orders APIs (5)

1. Create Order

POST `/orders/create/`

```
{
  "status": "PENDING",
  "items": [
    {
      "product": 1,
      "quantity": 2,
      "price": "799.00"
    }
  ]
}
```

 Requires buyer authentication.

2. List Orders

GET `/orders/`

Buyers see their orders.

Sellers see orders that contain their products.

3. Order Detail

GET `/orders/{id}/`

4. Update Order Status (Planned)

PUT or PATCH `/orders/{id}/status/`

```
{  
  "status": "SHIPPED"  
}
```

Role-Based Access Summary

Endpoint	Buyer	Seller	Guest
/accounts/register/	✓	✓	✓
/accounts/login/	✓	✓	✓
/products/	✓	✓	✓
/products/add/	✗	✓	✗
/orders/create/	✓	✗	✗
/orders/	✓	✓	✗
/orders/{id}/	✓	✓	✗
/orders/{id}/status/	✓	✓	✗