

Shop Subscription

Author: Josh S. Sakweli, Backend Lead Team

Last Updated: 2026-06-04

Version: v1.0

Base URL: `api/v1/e-commerce/shops`

Short Description: The Shop Subscription API allows users to subscribe and unsubscribe from shops on the Nexgate marketplace. Subscriptions drive personalised product discovery — subscribed shops receive a ranking boost in the Trending and For You feeds, and a dedicated "from your shops" signal in the relevance formula. Shop owners can view who has subscribed to their shop.

Hints:

- The action is called **Subscribe** — not Follow. This is intentional to distinguish from the social module's user→user **Follow** feature. Users follow people, they subscribe to shops.
- `POST /{shopId}/subscribe` is a **toggle** — calling it once subscribes, calling it again unsubscribes. No separate unsubscribe endpoint is needed.
- Subscription status (`isSubscribed`) is automatically included in `ShopResponse` for all shop detail endpoints. Authenticated users see their real status; anonymous users always receive `false`.
- Subscriptions directly feed into the Marketplace `FOR_YOU` and `TRENDING` personalisation — see `marketplace_api_doc.md` for formula details.
- Pages are **1-based** (`page=1` returns the first page).

Standard Response Format

All API responses follow a consistent structure using the Globe Response Builder pattern:

Success Response Structure

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Operation completed successfully",
  "action_time": "2026-06-04T10:30:45",
```

```
"data": { }  
}
```

Error Response Structure

```
{  
  "success": false,  
  "httpStatus": "BAD_REQUEST",  
  "message": "Error description",  
  "action_time": "2026-06-04T10:30:45",  
  "data": "Error description"  
}
```

Standard Response Fields

Field	Type	Description
success	boolean	true for success, false for errors
httpStatus	string	HTTP status name (OK, BAD_REQUEST, NOT_FOUND, etc.)
message	string	Human-readable operation result
action_time	string	ISO 8601 timestamp of response generation
data	object	Response payload for success, error detail for failures

HTTP Method Badge Standards

- GET — GET Green
- POST — POST Blue

Subscription State Reference

SUBSCRIPTION STATE FLOW

```

|
| User not subscribed
|   |
|   ▼ POST /{shopId}/subscribe
| User subscribed →
|   |
|   | subscriberCount + 1
|   | subscribed: true
|   |
|   ▼ POST /{shopId}/subscribe (toggle again)
| User not subscribed ←
|   |
|   | subscriberCount - 1
|   | subscribed: false
|
| Side effects:
| - Shop's subscriberCount is updated atomically
| - Marketplace FOR_YOU feed recalculates on next request
| - Marketplace TRENDING feed gives +0.25 boost to subscribed shops

```

How Subscriptions Affect the Marketplace

```

| SUBSCRIPTION → MARKETPLACE SIGNALS
|
| TRENDING feed:
|   personalizedScore = trendingScore + 0.25
|   (applied to every product from a subscribed shop)
|
| FOR YOU feed:
|   relevanceScore = categoryMatch × 0.40
|                   + favShopBoost × 0.40 ← subscription signal
|                   + trendingScore × 0.20
|   favShopBoost = 1.0 if subscribed, 0.0 otherwise
|
| Result: products from subscribed shops surface near the top of

```

| both feeds while still allowing genuinely trending products to
| compete (soft priority, not guaranteed top position). |

Endpoints

1. Toggle Subscribe / Unsubscribe

Purpose: Subscribes the authenticated user to a shop if they are not already subscribed, or unsubscribes them if they are. Returns the new subscription state and the updated subscriber count.

Endpoint: **POST** `api/v1/e-commerce/shops/{shopId}/subscribe`

Access Level: Protected (authentication required)

Authentication: Bearer Token (required)

Request Headers:

Header	Type	Required	Description
<code>Authorization</code>	string	Yes	<code>Bearer <token></code>

Path Parameters:

Parameter	Type	Required	Description	Validation
<code>shopId</code>	UUID	Yes	The shop to subscribe or unsubscribe from	Must be a valid UUID of an active, non-deleted shop

Success Response JSON Sample — Subscribe:

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Subscribed to shop successfully",
  "action_time": "2026-06-04T10:30:45",
  "data": {
    "subscribed": true,
```

```
    "subscriberCount": 1284
  }
}
```

Success Response JSON Sample — Unsubscribe (same endpoint, called again):

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Unsubscribed from shop successfully",
  "action_time": "2026-06-04T10:30:45",
  "data": {
    "subscribed": false,
    "subscriberCount": 1283
  }
}
```

Success Response Fields:

Field	Type	Description
<code>subscribed</code>	boolean	Current subscription state after the toggle — <code>true</code> if now subscribed, <code>false</code> if now unsubscribed
<code>subscriberCount</code>	long	Updated total number of subscribers for this shop

Error Response JSON Samples:

Unauthorized (401):

```
{
  "success": false,
  "httpStatus": "UNAUTHORIZED",
  "message": "Token has expired",
  "action_time": "2026-06-04T10:30:45",
  "data": "Token has expired"
}
```

Shop Not Found (404):

```
{
  "success": false,
```

```
"httpStatus": "NOT_FOUND",
"message": "Shop not found",
"action_time": "2026-06-04T10:30:45",
"data": "Shop not found"
}
```

Standard Error Types:

- `401 UNAUTHORIZED`: Token missing, expired, or invalid
- `404 NOT_FOUND`: Shop does not exist or has been deleted
- `500 INTERNAL_SERVER_ERROR`: Unexpected server failure

2. My Subscriptions

Purpose: Returns a paginated list of all shops the authenticated user is currently subscribed to, ordered by most recently subscribed first. Each entry includes shop details and the current subscriber count.

Endpoint: `GET` `api/v1/e-commerce/shops/my-subscriptions`

Access Level: Protected (authentication required)

Authentication: Bearer Token (required)

Request Headers:

Header	Type	Required	Description
<code>Authorization</code>	string	Yes	<code>Bearer <token></code>

Query Parameters:

Parameter	Type	Required	Description	Default
<code>page</code>	integer	No	Page number (1-based)	<code>1</code>
<code>size</code>	integer	No	Items per page	<code>10</code>

Success Response JSON Sample:

```
{
  "success": true,
  "httpStatus": "OK",
}
```

```

"message": "My subscriptions retrieved successfully",
"action_time": "2026-06-04T10:30:45",
"data": {
  "content": [
    {
      "subscriptionId": "9b1deb4d-3b7d-4bad-9bdd-2b0d7b3dcb6d",
      "shopId": "7cb3a812-1234-4abc-b3fc-9d84f55bce12",
      "shopName": "TechStore Tanzania",
      "shopSlug": "techstore-tanzania",
      "logoUrl": "https://cdn.nexgate.com/shops/techstore-logo.jpg",
      "bannerUrl": "https://cdn.nexgate.com/shops/techstore-banner.jpg",
      "status": "ACTIVE",
      "isVerified": true,
      "verificationBadge": "GOLD",
      "trustScore": 4.80,
      "subscriberCount": 1284,
      "subscribedAt": "2026-05-20T14:32:00"
    }
  ],
  "currentPage": 1,
  "pageSize": 10,
  "totalElements": 7,
  "totalPages": 1,
  "hasNext": false,
  "hasPrevious": false
}
}

```

Success Response Fields:

Field	Description
<code>content[].subscriptionId</code>	Unique identifier of this subscription record
<code>content[].shopId</code>	Shop's unique identifier
<code>content[].shopName</code>	Shop display name
<code>content[].shopSlug</code>	Shop URL slug — use for navigating to the shop page
<code>content[].logoUrl</code>	Shop logo image URL
<code>content[].bannerUrl</code>	Shop banner image URL
<code>content[].status</code>	Shop status — <code>ACTIVE</code> , <code>TEMPORARILY_OFFLINE</code> , etc.
<code>content[].isVerified</code>	Whether the shop has passed platform verification

Field	Description
<code>content[].verificationBadge</code>	Verification badge level — <code>BRONZE</code> , <code>SILVER</code> , <code>GOLD</code>
<code>content[].trustScore</code>	Shop trust rating <code>0.00–5.00</code>
<code>content[].subscriberCount</code>	Total subscribers this shop currently has
<code>content[].subscribedAt</code>	ISO 8601 timestamp of when the user subscribed
<code>currentPage</code>	Current page number (1-based)
<code>totalElements</code>	Total shops the user is subscribed to
<code>hasNext</code> / <code>hasPrevious</code>	Pagination navigation flags

Error Response JSON Samples:

Unauthorized (401):

```
{
  "success": false,
  "httpStatus": "UNAUTHORIZED",
  "message": "Token has expired",
  "action_time": "2026-06-04T10:30:45",
  "data": "Token has expired"
}
```

Standard Error Types:

- `401 UNAUTHORIZED`: Token missing, expired, or invalid
- `500 INTERNAL_SERVER_ERROR`: Unexpected server failure

3. Shop Subscribers (Owner Only)

Purpose: Returns a paginated list of users who have subscribed to a specific shop, ordered by most recently subscribed first. Only the shop owner can access this endpoint — calling it for a shop you do not own returns a 403 error.

Endpoint: `GET` `api/v1/e-commerce/shops/{shopId}/subscribers`

Access Level: `🔒` Protected (shop owner only)

Authentication: Bearer Token (required — must be the owner of the shop)

Request Headers:

Header	Type	Required	Description
Authorization	string	Yes	Bearer <token>

Path Parameters:

Parameter	Type	Required	Description	Validation
shopId	UUID	Yes	The shop whose subscriber list to retrieve	Caller must own this shop

Query Parameters:

Parameter	Type	Required	Description	Default
page	integer	No	Page number (1-based)	1
size	integer	No	Items per page	10

Success Response JSON Sample:

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Subscribers retrieved successfully",
  "action_time": "2026-06-04T10:30:45",
  "data": {
    "content": [
      {
        "userId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "fullName": "Amina Hassan",
        "userName": "amina.h",
        "avatarUrl": "https://cdn.nexgate.com/avatars/amina-h.jpg",
        "subscribedAt": "2026-06-01T09:45:00"
      },
      {
        "userId": "1cb2e847-3f1a-4bcd-a3fc-9e84f22bce99",
        "fullName": "John Mwangi",
        "userName": "jmwangi",
        "avatarUrl": null,
        "subscribedAt": "2026-05-29T16:12:00"
      }
    ]
  }
}
```

```

    "currentPage": 1,
    "pageSize": 10,
    "totalElements": 1284,
    "totalPages": 129,
    "hasNext": true,
    "hasPrevious": false
  }
}

```

Success Response Fields:

Field	Description
<code>content[].userId</code>	Subscriber's unique account identifier
<code>content[].fullName</code>	Subscriber's first + last name
<code>content[].userName</code>	Subscriber's public username
<code>content[].avatarUrl</code>	Subscriber's profile picture URL — <code>null</code> if no avatar set
<code>content[].subscribedAt</code>	ISO 8601 timestamp of when this user subscribed
<code>currentPage</code>	Current page number (1-based)
<code>totalElements</code>	Total subscriber count for this shop
<code>hasNext</code> / <code>hasPrevious</code>	Pagination navigation flags

Error Response JSON Samples:

Unauthorized (401):

```

{
  "success": false,
  "httpStatus": "UNAUTHORIZED",
  "message": "Token has expired",
  "action_time": "2026-06-04T10:30:45",
  "data": "Token has expired"
}

```

Forbidden — Not the shop owner (403):

```

{
  "success": false,
  "httpStatus": "FORBIDDEN",
  "message": "You do not own this shop",
}

```

```
"action_time": "2026-06-04T10:30:45",
"data": "You do not own this shop"
}
```

Shop Not Found (404):

```
{
  "success": false,
  "httpStatus": "NOT_FOUND",
  "message": "Shop not found",
  "action_time": "2026-06-04T10:30:45",
  "data": "Shop not found"
}
```

Standard Error Types:

- `401 UNAUTHORIZED`: Token missing, expired, or invalid
- `403 FORBIDDEN`: Authenticated user does not own this shop
- `404 NOT_FOUND`: Shop does not exist or has been deleted
- `500 INTERNAL_SERVER_ERROR`: Unexpected server failure

isSubscribed & subscriberCount in ShopResponse

Subscription state is automatically embedded in the standard `ShopResponse` returned by all shop detail endpoints — no separate call needed.

```
{
  "shopId": "7cb3a812-...",
  "shopName": "TechStore Tanzania",
  "isVerified": true,
  "trustScore": 4.80,
  "isSubscribed": true,
  "subscriberCount": 1284,
  "productCount": 47
}
```

Field	Type	Description
-------	------	-------------

<code>isSubscribed</code>	boolean	<code>true</code> if the requesting authenticated user is subscribed to this shop. Always <code>false</code> for anonymous users
<code>subscriberCount</code>	long	Total number of subscribers this shop currently has
<code>productCount</code>	long	Number of active (published) products in this shop

Quick Reference — All Subscription Endpoints

#	Endpoint	Method	Auth	Who can call
1	<code>api/v1/e-commerce/shops/{shopId}/subscribe</code>	POST	Required	Any authenticated user
2	<code>api/v1/e-commerce/shops/my-subscriptions</code>	GET	Required	Any authenticated user
3	<code>api/v1/e-commerce/shops/{shopId}/subscribers</code>	GET	Required	Shop owner only

Revision #1

Created 4 June 2026 09:56:30 by Admin Qbit

Updated 4 June 2026 09:56:50 by Admin Qbit