

Notification Management

Author: Josh S. Sakweli, Backend Lead Team

Last Updated: 2025-10-26

Version: v1.0

Base URL: `https://api.nextgate.com/api/v1`

Short Description

The Notification Management API provides a comprehensive in-app notification system that allows users to receive, manage, and interact with notifications across different services and shops. This API supports creating notifications, marking them as read, filtering by various criteria, and managing notification lifecycle with full pagination support.

Hints

- **Authentication Required:** All endpoints under `/notifications` require Bearer token authentication except the `/notifications/in-app` endpoint
- **Pagination:** List endpoints support pagination with default `page=1` and `size=20`
- **Page Numbering:** API uses 1-based page numbering (page 1 is the first page)
- **Rate Limiting:** Standard rate limits apply (100 requests per minute per user)
- **Timezones:** All timestamps are returned in ISO 8601 format with local server time
- **Notification Types:** Supported types include `INFO`, `WARNING`, `ERROR`, `SUCCESS`
- **Priority Levels:** Supported priorities include `LOW`, `MEDIUM`, `HIGH`, `URGENT`
- **Service Types:** Examples include `ORDER`, `PAYMENT`, `SHIPPING`, `PRODUCT`, `ACCOUNT`

Standard Response Format

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

Success Response Structure

```
{  
  "success": true,
```

```
"httpStatus": "OK",
"message": "Operation completed successfully",
"action_time": "2025-10-26T10:30:45",
"data": {
  // Actual response data goes here
}
}
```

Error Response Structure

```
{
  "success": false,
  "httpStatus": "BAD_REQUEST",
  "message": "Error description",
  "action_time": "2025-10-26T10:30:45",
  "data": "Error description"
}
```

Standard Response Fields

Field	Type	Description
success	boolean	Always true for successful operations, false for errors
httpStatus	string	HTTP status name (OK, BAD_REQUEST, NOT_FOUND, etc.)
message	string	Human-readable message describing the operation result
action_time	string	ISO 8601 timestamp of when the response was generated
data	object/string	Response payload for success, error details for failures

HTTP Method Badge Standards

For better visual clarity, all endpoints use colored badges for HTTP methods with the following standard colors:

- GET** - Green (Safe, read-only operations)

- **POST** - Blue (Create new resources)
- **PUT** - Yellow (Update/replace entire resource)
- **DELETE** - Red (Remove resources)

Endpoints

1. Get My Notifications

Purpose: Retrieve all notifications for the authenticated user with pagination support

Endpoint: **GET** `{base_url}/notifications/me`

Access Level: Protected (Requires Bearer Token Authentication)

Authentication: Bearer Token

Request Headers

Header	Type	Required	Description
Authorization	string	Yes	Bearer token for authentication

Query Parameters

Parameter	Type	Required	Description	Validation	Default
page	integer	No	Page number (1-based)	Must be positive integer	1
size	integer	No	Number of items per page	Must be between 1-100	20

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Notifications retrieved successfully",
  "action_time": "2025-10-26T10:30:45",
  "data": {
    "notifications": [
      {
```

```

    "id": "123e4567-e89b-12d3-a456-426614174000",
    "userId": "987fcdeb-51a2-43d7-9c4e-123456789abc",
    "shopId": "456e7890-e89b-12d3-a456-426614174001",
    "serviceId": "ORD-2024-001",
    "serviceType": "ORDER",
    "title": "Order Confirmed",
    "message": "Your order #ORD-2024-001 has been confirmed",
    "type": "SUCCESS",
    "priority": "MEDIUM",
    "isRead": false,
    "data": {
      "orderId": "ORD-2024-001",
      "amount": 150.00,
      "currency": "USD"
    },
    "createdAt": "2025-10-26T09:15:30",
    "readAt": null
  }
],
"currentPage": 1,
"pageSize": 20,
"totalElements": 45,
"totalPages": 3,
"hasNext": true,
"hasPrevious": false,
"isFirst": true,
"isLast": false
}
}

```

Success Response Fields

Field	Description
notifications	Array of notification objects
notifications[].id	Unique identifier for the notification
notifications[].userId	ID of the user who received the notification
notifications[].shopId	ID of the associated shop (nullable)
notifications[].serviceId	Identifier for the service that generated the notification
notifications[].serviceType	Type of service (ORDER, PAYMENT, SHIPPING, etc.)

Field	Description
notifications[].title	Notification title
notifications[].message	Notification message content
notifications[].type	Notification type (INFO, WARNING, ERROR, SUCCESS)
notifications[].priority	Priority level (LOW, MEDIUM, HIGH, URGENT)
notifications[].isRead	Whether the notification has been read
notifications[].data	Additional metadata as JSON object
notifications[].createdAt	Timestamp when notification was created
notifications[].readAt	Timestamp when notification was read (null if unread)
currentPage	Current page number (1-based)
pageSize	Number of items per page
totalElements	Total number of notifications
totalPages	Total number of pages
hasNext	Whether there is a next page
hasPrevious	Whether there is a previous page
isFirst	Whether this is the first page
isLast	Whether this is the last page

Error Response JSON Sample

```
{
  "success": false,
  "httpStatus": "UNAUTHORIZED",
  "message": "Token has expired",
  "action_time": "2025-10-26T10:30:45",
  "data": "Token has expired"
}
```

2. Get Unread Notifications

Purpose: Retrieve only unread notifications for the authenticated user

Endpoint: **GET** `{base_url}/notifications/unread`

Access Level: Protected (Requires Bearer Token Authentication)

Authentication: Bearer Token

Request Headers

Header	Type	Required	Description
Authorization	string	Yes	Bearer token for authentication

Query Parameters

Parameter	Type	Required	Description	Validation	Default
page	integer	No	Page number (1-based)	Must be positive integer	1
size	integer	No	Number of items per page	Must be between 1-100	20

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Notifications retrieved successfully",
  "action_time": "2025-10-26T10:30:45",
  "data": {
    "notifications": [
      {
        "id": "223e4567-e89b-12d3-a456-426614174002",
        "userId": "987fcdeb-51a2-43d7-9c4e-123456789abc",
        "shopId": null,
        "serviceId": "PAY-2024-050",
        "serviceType": "PAYMENT",
        "title": "Payment Pending",
        "message": "Your payment for order #ORD-2024-001 is pending",
        "type": "WARNING",
        "priority": "HIGH",
        "isRead": false,
        "data": {
          "paymentId": "PAY-2024-050",
          "amount": 150.00
        }
      },
      {
        "createdAt": "2025-10-26T10:15:30",
```

```
        "readAt": null
      }
    ],
    "currentPage": 1,
    "pageSize": 20,
    "totalElements": 12,
    "totalPages": 1,
    "hasNext": false,
    "hasPrevious": false,
    "isFirst": true,
    "isLast": true
  }
}
```

3. Get Unread Count

Purpose: Get the total count of unread notifications for the authenticated user

Endpoint: GET `{base_url}/notifications/unread-count`

Access Level: Protected (Requires Bearer Token Authentication)

Authentication: Bearer Token

Request Headers

Header	Type	Required	Description
Authorization	string	Yes	Bearer token for authentication

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Unread count retrieved successfully",
  "action_time": "2025-10-26T10:30:45",
  "data": {
    "unreadCount": 12
  }
}
```

Success Response Fields

Field	Description
unreadCount	Total number of unread notifications

4. Get Notification Summary

Purpose: Get a summary of all notifications including total, unread, and read counts

Endpoint: **GET** `{base_url}/notifications/summary`

Access Level: Protected (Requires Bearer Token Authentication)

Authentication: Bearer Token

Request Headers

Header	Type	Required	Description
Authorization	string	Yes	Bearer token for authentication

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Notification summary retrieved successfully",
  "action_time": "2025-10-26T10:30:45",
  "data": {
    "total": 45,
    "unread": 12,
    "read": 33
  }
}
```

Success Response Fields

Field	Description
total	Total number of notifications
unread	Number of unread notifications
read	Number of read notifications

5. Get Notifications by Shop

Purpose: Retrieve notifications filtered by a specific shop

Endpoint: GET `{base_url}/notifications/shop/{shopId}`

Access Level: Protected (Requires Bearer Token Authentication)

Authentication: Bearer Token

Request Headers

Header	Type	Required	Description
Authorization	string	Yes	Bearer token for authentication

Path Parameters

Parameter	Type	Required	Description	Validation
shopId	UUID	Yes	Unique identifier of the shop	Must be valid UUID format

Query Parameters

Parameter	Type	Required	Description	Validation	Default
page	integer	No	Page number (1-based)	Must be positive integer	1
size	integer	No	Number of items per page	Must be between 1-100	20

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Notifications retrieved successfully",
  "action_time": "2025-10-26T10:30:45",
  "data": {
    "notifications": [
      {
        "id": "323e4567-e89b-12d3-a456-426614174003",
        "userId": "987fcdeb-51a2-43d7-9c4e-123456789abc",
```

```
"shopId": "456e7890-e89b-12d3-a456-426614174001",
"serviceId": "PROD-2024-100",
"serviceType": "PRODUCT",
"title": "New Product Available",
"message": "A new product has been added to your shop",
"type": "INFO",
"priority": "LOW",
"isRead": true,
"data": {
  "productId": "PROD-2024-100",
  "productName": "Sample Product"
},
"createdAt": "2025-10-25T14:20:00",
"readAt": "2025-10-25T15:30:00"
}
],
"currentPage": 1,
"pageSize": 20,
"totalElements": 8,
"totalPages": 1,
"hasNext": false,
"hasPrevious": false,
"isFirst": true,
"isLast": true
}
}
```

6. Get Notifications by Service Type

Purpose: Retrieve notifications filtered by service type

Endpoint: GET `{base_url}/notifications/service/{serviceType}`

Access Level: Protected (Requires Bearer Token Authentication)

Authentication: Bearer Token

Request Headers

Header	Type	Required	Description
--------	------	----------	-------------

Authorization	string	Yes	Bearer token for authentication
---------------	--------	-----	---------------------------------

Path Parameters

Parameter	Type	Required	Description	Validation
serviceType	string	Yes	Type of service (ORDER, PAYMENT, SHIPPING, PRODUCT, ACCOUNT)	Must not be empty

Query Parameters

Parameter	Type	Required	Description	Validation	Default
page	integer	No	Page number (1-based)	Must be positive integer	1
size	integer	No	Number of items per page	Must be between 1-100	20

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Notifications retrieved successfully",
  "action_time": "2025-10-26T10:30:45",
  "data": {
    "notifications": [
      {
        "id": "423e4567-e89b-12d3-a456-426614174004",
        "userId": "987fcdeb-51a2-43d7-9c4e-123456789abc",
        "shopId": "456e7890-e89b-12d3-a456-426614174001",
        "serviceId": "ORD-2024-002",
        "serviceType": "ORDER",
        "title": "Order Shipped",
        "message": "Your order #ORD-2024-002 has been shipped",
        "type": "SUCCESS",
        "priority": "MEDIUM",
        "isRead": false,
        "data": {
          "orderId": "ORD-2024-002",
          "trackingNumber": "TRK123456789"
        }
      }
    ]
  }
}
```

```
    },
    "createdAt": "2025-10-26T08:45:00",
    "readAt": null
  }
],
"currentPage": 1,
"pageSize": 20,
"totalElements": 15,
"totalPages": 1,
"hasNext": false,
"hasPrevious": false,
"isFirst": true,
"isLast": true
}
}
```

7. Get Notification by ID

Purpose: Retrieve a single notification by its unique identifier

Endpoint: **GET** `{base_url}/notifications/{notificationId}`

Access Level: Protected (Requires Bearer Token Authentication)

Authentication: Bearer Token

Request Headers

Header	Type	Required	Description
Authorization	string	Yes	Bearer token for authentication

Path Parameters

Parameter	Type	Required	Description	Validation
notificationId	UUID	Yes	Unique identifier of the notification	Must be valid UUID format

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Notification retrieved successfully",
  "action_time": "2025-10-26T10:30:45",
  "data": {
    "id": "123e4567-e89b-12d3-a456-426614174000",
    "userId": "987fcdeb-51a2-43d7-9c4e-123456789abc",
    "shopId": "456e7890-e89b-12d3-a456-426614174001",
    "serviceId": "ORD-2024-001",
    "serviceType": "ORDER",
    "title": "Order Confirmed",
    "message": "Your order #ORD-2024-001 has been confirmed",
    "type": "SUCCESS",
    "priority": "MEDIUM",
    "isRead": false,
    "data": {
      "orderId": "ORD-2024-001",
      "amount": 150.00,
      "currency": "USD"
    },
    "createdAt": "2025-10-26T09:15:30",
    "readAt": null
  }
}
```

Error Response JSON Sample

```
{
  "success": false,
  "httpStatus": "NOT_FOUND",
  "message": "Notification not found or access denied",
  "action_time": "2025-10-26T10:30:45",
  "data": "Notification not found or access denied"
}
```

8. Mark Notification as Read

Purpose: Mark a single notification as read

Endpoint: **PUT** `{base_url}/notifications/{notificationId}/read`

Access Level: Protected (Requires Bearer Token Authentication)

Authentication: Bearer Token

Request Headers

Header	Type	Required	Description
Authorization	string	Yes	Bearer token for authentication

Path Parameters

Parameter	Type	Required	Description	Validation
notificationId	UUID	Yes	Unique identifier of the notification	Must be valid UUID format

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Notification marked as read",
  "action_time": "2025-10-26T10:30:45",
  "data": null
}
```

Error Response JSON Sample

```
{
  "success": false,
  "httpStatus": "NOT_FOUND",
  "message": "Notification not found or access denied",
  "action_time": "2025-10-26T10:30:45",
  "data": "Notification not found or access denied"
}
```

9. Mark Multiple Notifications as Read

Purpose: Mark multiple notifications as read in a single request

Endpoint: **PUT** `{base_url}/notifications/read`

Access Level: Protected (Requires Bearer Token Authentication)

Authentication: Bearer Token

Request Headers

Header	Type	Required	Description
Authorization	string	Yes	Bearer token for authentication
Content-Type	string	Yes	Must be application/json

Request JSON Sample

```
{
  "notificationIds": [
    "123e4567-e89b-12d3-a456-426614174000",
    "223e4567-e89b-12d3-a456-426614174002",
    "323e4567-e89b-12d3-a456-426614174003"
  ]
}
```

Request Body Parameters

Parameter	Type	Required	Description	Validation
notificationIds	array	Yes	Array of notification IDs to mark as read	Must not be empty, each element must be valid UUID

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "3 notification(s) marked as read",
  "action_time": "2025-10-26T10:30:45",
  "data": null
}
```

Error Response JSON Sample

```
{
  "success": false,
  "httpStatus": "BAD_REQUEST",
  "message": "Some notifications not found or access denied",
  "action_time": "2025-10-26T10:30:45",
  "data": "Some notifications not found or access denied"
}
```

10. Mark All Notifications as Read

Purpose: Mark all unread notifications for the authenticated user as read

Endpoint: **PUT** `{base_url}/notifications/read-all`

Access Level: Protected (Requires Bearer Token Authentication)

Authentication: Bearer Token

Request Headers

Header	Type	Required	Description
Authorization	string	Yes	Bearer token for authentication

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "All notifications marked as read",
  "action_time": "2025-10-26T10:30:45",
  "data": null
}
```

11. Delete Notification(DON'T USE)

Purpose: Delete a single notification permanently

Endpoint: **DELETE** `{base_url}/notifications/{notificationId}`

Access Level: Protected (Requires Bearer Token Authentication)

Authentication: Bearer Token

Request Headers

Header	Type	Required	Description
Authorization	string	Yes	Bearer token for authentication

Path Parameters

Parameter	Type	Required	Description	Validation
notificationId	UUID	Yes	Unique identifier of the notification	Must be valid UUID format

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Notification deleted successfully",
  "action_time": "2025-10-26T10:30:45",
  "data": null
}
```

Error Response JSON Sample

```
{
  "success": false,
  "httpStatus": "NOT_FOUND",
  "message": "Notification not found or access denied",
  "action_time": "2025-10-26T10:30:45",
  "data": "Notification not found or access denied"
}
```

12. Batch Delete Notifications(DON'T USE)

Purpose: Delete multiple notifications in a single request

Endpoint: **DELETE** `{base_url}/notifications/batch`

Access Level: Protected (Requires Bearer Token Authentication)

Authentication: Bearer Token

Request Headers

Header	Type	Required	Description
Authorization	string	Yes	Bearer token for authentication
Content-Type	string	Yes	Must be application/json

Request JSON Sample

```
{
  "notificationIds": [
    "123e4567-e89b-12d3-a456-426614174000",
    "223e4567-e89b-12d3-a456-426614174002",
    "323e4567-e89b-12d3-a456-426614174003"
  ]
}
```

Request Body Parameters

Parameter	Type	Required	Description	Validation
notificationIds	array	Yes	Array of notification IDs to delete	Must not be empty, each element must be valid UUID

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "3 notification(s) deleted successfully",
  "action_time": "2025-10-26T10:30:45",
  "data": null
}
```

Error Response JSON Sample

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

```
"httpStatus": "BAD_REQUEST",
"message": "Some notifications not found or access denied",
"action_time": "2025-10-26T10:30:45",
"data": "Some notifications not found or access denied"
}
```

13. Delete All Read Notifications(DON'T USE)

Purpose: Delete all read notifications for the authenticated user

Endpoint: **DELETE** `{base_url}/notifications/read`

Access Level: Protected (Requires Bearer Token Authentication)

Authentication: Bearer Token

Request Headers

Header	Type	Required	Description
Authorization	string	Yes	Bearer token for authentication

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "15 read notification(s) deleted successfully",
  "action_time": "2025-10-26T10:30:45",
  "data": 15
}
```

Success Response Fields

Field	Description
data	Number of notifications that were deleted

14. Create In-App Notification(DON'T USE)

Purpose: Create a new in-app notification (Internal service-to-service endpoint)

Endpoint: POST `{base_url}/notifications/in-app`

Access Level: Protected (Requires API Key or Service Authentication)

Authentication: API Key or Service Token

Request Headers

Header	Type	Required	Description
X-API-Key	string	Yes	API key for service authentication
Content-Type	string	Yes	Must be application/json

Request JSON Sample

```
{
  "userId": "987fcdeb-51a2-43d7-9c4e-123456789abc",
  "shopId": "456e7890-e89b-12d3-a456-426614174001",
  "serviceId": "ORD-2024-001",
  "serviceType": "ORDER",
  "title": "Order Confirmed",
  "message": "Your order #ORD-2024-001 has been confirmed and is being processed",
  "type": "SUCCESS",
  "priority": "MEDIUM",
  "data": {
    "orderId": "ORD-2024-001",
    "amount": 150.00,
    "currency": "USD",
    "items": 3
  }
}
```

Request Body Parameters

Parameter	Type	Required	Description	Validation
userId	UUID	Yes	ID of the user to receive the notification	Must be valid UUID
shopId	UUID	No	ID of the associated shop	Must be valid UUID if provided

Parameter	Type	Required	Description	Validation
serviceId	string	Yes	Identifier for the service that generated the notification	Max length 100 characters
serviceType	string	Yes	Type of service (ORDER, PAYMENT, SHIPPING, PRODUCT, ACCOUNT)	Max length 50 characters
title	string	Yes	Notification title	Max length 255 characters
message	string	Yes	Notification message content	Max length 1000 characters
type	string	Yes	Notification type (INFO, WARNING, ERROR, SUCCESS)	Max length 50 characters
priority	string	Yes	Priority level (LOW, MEDIUM, HIGH, URGENT)	Max length 20 characters
data	object	No	Additional metadata as JSON object	Any valid JSON object

Success Response JSON Sample

```
{
  "success": true,
  "httpStatus": "OK",
  "message": "Notification saved successfully",
  "action_time": "2025-10-26T10:30:45",
  "data": "123e4567-e89b-12d3-a456-426614174000"
}
```

Success Response Fields

Field	Description
data	UUID of the created notification

Error Response JSON Sample

```
{
  "success": false,
  "httpStatus": "UNPROCESSABLE_ENTITY",
  "message": "Validation failed",
}
```

```
"action_time": "2025-10-26T10:30:45",
"data": {
  "userId": "must not be null",
  "title": "must not be blank",
  "message": "must not be blank"
}
}
```

Standard Error Types

Application-Level Exceptions (400-499)

- **400 BAD_REQUEST:** General invalid request data, random exceptions, or item already exists
- **401 UNAUTHORIZED:** Authentication issues (empty, invalid, expired, or malformed tokens)
- **403 FORBIDDEN:** Access denied, permission issues, verification failures, expired invitations
- **404 NOT_FOUND:** Requested resource does not exist
- **422 UNPROCESSABLE_ENTITY:** Validation errors with detailed field information
- **429 TOO_MANY_REQUESTS:** Rate limit exceeded

Server-Level Exceptions (500+)

- **500 INTERNAL_SERVER_ERROR:** Unexpected server errors

Error Response Examples

Bad Request - General (400)

```
{
  "success": false,
  "httpStatus": "BAD_REQUEST",
  "message": "User ID cannot be null",
  "action_time": "2025-10-26T10:30:45",
}
```

```
"data": "User ID cannot be null"
}
```

Unauthorized - Token Issues (401)

```
{
  "success": false,
  "httpStatus": "UNAUTHORIZED",
  "message": "Token has expired",
  "action_time": "2025-10-26T10:30:45",
  "data": "Token has expired"
}
```

Forbidden - Access Denied (403)

```
{
  "success": false,
  "httpStatus": "FORBIDDEN",
  "message": "Access denied: Insufficient permissions",
  "action_time": "2025-10-26T10:30:45",
  "data": "Access denied: Insufficient permissions"
}
```

Not Found (404)

```
{
  "success": false,
  "httpStatus": "NOT_FOUND",
  "message": "Notification not found or access denied",
  "action_time": "2025-10-26T10:30:45",
  "data": "Notification not found or access denied"
}
```

Validation Error (422)

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

```
"httpStatus": "UNPROCESSABLE_ENTITY",
"message": "Validation failed",
"action_time": "2025-10-26T10:30:45",
"data": {
  "userId": "must not be null",
  "title": "must not be blank",
  "serviceType": "must not be blank",
  "priority": "must not be blank"
}
}
```

Quick Reference Guide

Common HTTP Status Codes

- **200 OK:** Successful GET/PUT request
- **201 Created:** Successful POST request
- **204 No Content:** Successful DELETE request
- **400 Bad Request:** Invalid request data
- **401 Unauthorized:** Authentication required/failed
- **403 Forbidden:** Insufficient permissions
- **404 Not Found:** Resource not found
- **422 Unprocessable Entity:** Validation errors
- **429 Too Many Requests:** Rate limit exceeded
- **500 Internal Server Error:** Server error

Authentication Types

- **Bearer Token:** Include `Authorization: Bearer your_token` in headers (for user endpoints)
- **API Key:** Include `X-API-Key: your_key` in headers (for service-to-service endpoints)

Data Format Standards

- **Dates:** Use ISO 8601 format (2025-10-26T14:30:00)
- **IDs:** All IDs use UUID format
- **Pagination:** Uses 1-based page numbering
- **Boolean Fields:** Use `true` or `false` (not 1/0)

Notification Types

- **INFO:** General informational notifications
- **WARNING:** Warning or caution notifications
- **ERROR:** Error or failure notifications
- **SUCCESS:** Success or confirmation notifications

Priority Levels

- **LOW:** Low priority, can be checked later
- **MEDIUM:** Normal priority
- **HIGH:** High priority, needs attention soon
- **URGENT:** Urgent priority, requires immediate attention

Service Types

- **ORDER:** Order-related notifications
- **PAYMENT:** Payment-related notifications
- **SHIPPING:** Shipping and delivery notifications
- **PRODUCT:** Product-related notifications
- **ACCOUNT:** Account and profile notifications

Revision #4

Created 18 October 2025 20:44:11 by Admin Qbit

Updated 11 December 2025 08:21:53 by Admin Qbit