

■ PHRMA Production App

Complete API Documentation

API Version	v2
Generated	February 17, 2026
Status	Active
Endpoints	30+

Table of Contents

1. Authentication & Authorization

2. User Management

3. Medicine Store

4. Admin Store Management

5. Items Management

6. Location Services

7. Unit Management

8. GST Management

9. Orders

10. Payments

11. Notifications

12. Response Formats

■ Authentication & Authorization

Authentication Methods

Method	Description	Usage
JWT Token	Bearer token in Authorization header or cookie	Authorization: Bearer {token}
Gateway Mode	Headers from API Gateway	X-User-ID, X-User-Role, X-User-Email

Required Headers

Header	Type	Description
Content-Type	application/json	Required for POST/PUT requests
Authorization	Bearer {token}	For authenticated endpoints
X-User-ID	String (UUID)	Optional (Gateway mode)
X-User-Role	admin user manager	Optional (Gateway mode)

User Roles & Permissions

Role	Permissions	Endpoints Access
admin	Full access, verification, store management	/admin/*, /units/*, /gst/*
user	Regular user, can manage own store	/medicine-store/*, /items/*, /orders/*
manager	Store manager, manages items and orders	/items/*, /orders/*, /payments/*

User Management Endpoints

POST /api/v2/user/login

Authenticate user with email and password. Returns JWT token for subsequent requests.

```
Request Body: - email (String, Required): User email address - password (String, Required): User password (minimum 8 characters) - fcmToken (String, Optional): Firebase Cloud Messaging token for push notifications
```

Request Example:

```
{"email": "user@example.com", "password": "securePassword123", "fcmToken": "eO...K1"}
```

Success Response (200 OK):

```
{"status": 200, "message": "Login Successful", "data": {"user": {"_id": "507f1f77bcf86cd799439011", "name": "John Doe", "email": "user@example.com", "phone": "+919876543210", "role": "user", "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9"}}
```

Requirements:

- ✓ Email must be valid format
- ✓ Password must be at least 8 characters
- ✓ User must be registered in the system
- ✓ Account must not be suspended

POST /api/v2/user/logout

Logout user by invalidating the JWT token. Clears authentication cookie.

Authentication Required: Bearer Token or Valid JWT Cookie

POST /api/v2/user/forgot/password

Request password reset. Sends OTP to registered email address.

```
Request Body: email (String, Required)
```

POST /api/v2/user/verifyOtp

Verify OTP sent to email for password reset.

```
Request Body: email (String, Required), otp (String, Required)
```

POST /api/v2/user/ResetPassword

Reset user password after OTP verification. Internal service only.

Requires: verifyInternalService Middleware

■ Medicine Store Endpoints

POST /api/v2/medicine-store/register

Register a new medicine store with complete verification including GST, pharmacy license, and address validation.

Field	Type	Required	Validation
userNmae	String	Yes	Name of store owner
email	String	Yes	Valid email format
phone	String	Yes	10-11 digit Indian phone
storeName	String	Yes	Official store name
storeType	String	Yes	retail wholesale chain
GSTNumber	String	Yes	Valid 15-digit GST number
pharmacyLicence	String	Yes	Valid pharmacy license for state
address	String	Yes	Complete address
city	String	Yes	Must match pincode
state	String	Yes	Must match pincode
pincode	String	Yes	6-digit Indian pincode

Request Example:

```
{"userNmae": "Dr. Rajesh Kumar", "email": "rajesh.med@example.com", "phone": "9876543210", "storeName": "Apollo Pharmacy", "storeType": "retail", "GSTNumber": "27AABCU9603R1Z0", "pharmacyLicence": "PL2024000123", "address": "Plot 123, Medical Complex", "city": "Bangalore", "state": "Karnataka", "pincode": "560034"}
```

Validation Requirements:

- ✓ GST format validation
- ✓ Phone: 10-11 digits, valid Indian format
- ✓ Pharmacy License valid for state
- ✓ Pincode must match city and state
- ✓ Email must be unique
- ✓ Store enters 'pending' verification status

GET /api/v2/admin/stores/pending

Get all stores awaiting verification. Admin only.

GET /api/v2/admin/stores/:storeId

Get detailed information about a specific store for verification review.

PUT /api/v2/admin/stores/:storeId/status

Update store verification status (approve, reject, suspend).

GET /api/v2/admin/stores/stats/verification

Get verification statistics and summary.

Status Update Body:

```
{"action": "approve", "adminRemarks": "All documents verified successfully"} Valid actions: approve|reject|suspend
```

Requirements:

- ✓ User must have admin role
- ✓ For reject/suspend, adminRemarks is mandatory
- ✓ Once approved, store can add items and process orders

■ Items Management

POST /api/v2/items/add

Create new regular item. Requires authentication and file uploads.

POST /api/v2/items/premium

Create premium/featured item with enhanced visibility.

PUT /api/v2/items/update/:itemId

Update existing item including images.

DELETE /api/v2/items/delete/:itemId

Delete single item (admin only).

DELETE /api/v2/items/

Delete all items (admin only - destructive operation).

Item Requirements:

- ✓ User must be authenticated
- ✓ Must have associated medicine store
- ✓ Store must be verified/approved
- ✓ Images: JPEG/PNG, max 5MB each
- ✓ Price > 0, Quantity ≥ 0

■■ Location Services

GET /api/v2/location/states

Get list of all Indian states and union territories.

GET /api/v2/location/cities/:state

Get list of cities for a given state.

GET /api/v2/location/pincode/:pincode

Get city and state information for a given pincode.

```
Example: /api/v2/location/pincode/560034 Response: {"status": 200, "data": {"pincode": "560034", "city": "Bangalore", "state": "Karnataka"}}
```

Note: Public endpoints, no authentication required.

Unit Management

Parent Units (e.g., Kilogram, Liter)

POST /api/v2/units/add-parent-units

Create new parent unit (base measurement unit).

GET /api/v2/units/get-parent-units

Get all parent units (admin only).

PUT /api/v2/units/update-parent-units/:id

Update parent unit (admin only).

DELETE /api/v2/units/delete-parent-units/:id

Delete parent unit (admin only).

Child Units (e.g., 100ml, 500g)

POST /api/v2/units/add-child-units

Create child unit (derived from parent unit).

GET /api/v2/units/get-child-units

Get all child units (admin only).

PUT /api/v2/units/update-child-units/:id

Update child unit (admin only).

DELETE /api/v2/units/delete-child-units/:id

Delete child unit (admin only).

```
Child Unit Example: {"childUnitName": "100 Milliliters", "childUnitSymbol": "100ml", "parentUnitId": "507f...", "conversionFactor": 0.1}
```

GST Rate Management

POST /api/v2/gst/add

Add new GST rate for product category.

PUT /api/v2/gst/update/:gstId

Update existing GST rate.

DELETE /api/v2/gst/delete/:gstId

Delete GST rate.

GET /api/v2/gst/

Get all GST rates.

```
Add GST Example: {"productCategory": "Medicines", "gstRate": 5, "description": "GST rate for medicines and pharmaceuticals"} Valid rates in India: 0%, 5%, 12%, 18%, 28%
```

Orders

POST /api/v2/orders

Create new order with automatic notification to user.

Request Body: - userId (String, Required): User ID - items (Array, Required): Order items with quantity and price - totalAmount (Number, Required): Order total - deliveryAddress (String, Optional): Delivery address

```
{"userId": "507f...", "items": [{"itemId": "507f...", "quantity": 2, "price": 299}], "totalAmount": 598, "deliveryAddress": "123 Main St, Bangalore"}
```

PATCH /api/v2/orders/:orderId/status

Update order status and notify user.

Status Flow: pending → processing → shipped → delivered → cancelled Example: {"status": "shipped", "userId": "507f..."}
[{"status": "pending"}, {"status": "processing"}, {"status": "shipped"}, {"status": "delivered"}, {"status": "cancelled"}]

■ Payments

POST /api/v2/payments

Process payment and send confirmation notification.

```
Request Body: - userId (String, Required) - orderId (String, Required) - amount (Number, Required) - paymentMethod  
(String, Required): credit_card|debit_card|upi|net_banking
```

POST /api/v2/payments/:paymentId/refund

Process refund. Reflects in 5-7 business days.

```
{"userId": "507f...", "amount": 598, "reason": "Order cancelled by customer"}
```

POST /api/v2/payments/failed

Handle failed payment and notify user.

■ Notifications

GET /api/v2/notification-service/health

Check notification service health (public).

POST /api/v2/notification-service/send-to-user

Send to authenticated user (requires auth).

POST /api/v2/notification-service/send-to-users

Send to multiple users by userIds array.

POST /api/v2/notification-service/send-bulk

Send bulk notifications (up to 1000 users).

```
Notification Body: {"title": "Order Confirmed", "body": "Your order #ORD123 has been confirmed", "data": {"orderId": "ORD123", "screen": "OrderDetails"}}
```

Notification Requirements:

- ✓ Title under 100 characters
- ✓ Body should be brief and meaningful
- ✓ Data object optional with custom properties
- ✓ Bulk: up to 1000 userIds per request

Standard Response Formats

Success Response (2xx)

```
{"status": 200, "message": "Operation completed successfully", "data": {...}}
```

Error Response (4xx/5xx)

```
{"status": 400, "message": "Error description", "error": {"code": "ERROR_CODE", "details": "..."}, "data": {...}}
```

HTTP Status Codes

Code	Meaning	Scenarios
200	OK	Successful GET/PATCH
201	Created	Successful POST creating resource
400	Bad Request	Validation errors
401	Unauthorized	Missing/invalid token
403	Forbidden	Insufficient permissions
404	Not Found	Resource not found
409	Conflict	Duplicate resource
500	Server Error	Internal server error

Pagination & Rate Limiting

Parameter/Header	Type	Description
skip	Number	Records to skip (default: 0)
limit	Number	Records per page (default: 10, max: 100)
X-RateLimit-Limit	Number	1000 requests per hour
X-RateLimit-Remaining	Number	Remaining requests in window
X-RateLimit-Reset	Timestamp	When limit resets

Key Features:

- ✓ Complete REST API with JWT Authentication
- ✓ Role-based Access Control (Admin, User, Manager)
 - ✓ Comprehensive Input Validation
- ✓ Automatic Notifications for Orders & Payments
- ✓ File Upload Support with Image Processing
- ✓ Error Handling with Descriptive Messages
 - ✓ Pagination & Rate Limiting
- ✓ Gateway Mode Support for Microservices