

# National Eye API Documentation

## Table of Contents

1. [Overview](#)
  2. [Base URL](#)
  3. [Authentication](#)
  4. [API Endpoints](#)
    - [Authentication Endpoints](#)
    - [Reports Endpoints](#)
    - [Admin Endpoints](#)
    - [Employee Endpoints](#)
    - [Citizen Endpoints](#)
- 

## Overview

The National Eye API is a comprehensive system for managing reports, users, and administrative functions. It provides endpoints for authentication, report management, and various administrative operations.

## Base URL

### Development Environment:

```
http://127.0.0.1:8000/api
```

### Production Environment:

```
https://ne.doctoria.app/testing/public/api
```

## Authentication

The API uses Bearer token authentication for protected endpoints. After successful login or OTP verification, you'll receive a token that should be included in the Authorization header:

```
Authorization: Bearer {your_token_here}
```

---

# API Endpoints

## Authentication Endpoints

### 1. Send OTP

**Endpoint:** POST /send-otp

**Description:** Sends an OTP to the provided phone number for user verification.

**Request Body:**

```
json
{
  "national_id": "12345678901234",
  "phone": "01157861819"
}
```

**Response:** OTP sent to the specified phone number.

---

### 2. Verify OTP

**Endpoint:** POST /verify-otp

**Description:** Verifies the OTP sent to the user's phone number.

**Request Body:**

```
json
{
  "phone": "01157861819",
  "otp": "2023"
}
```

**Response:** Returns authentication token upon successful verification.

---

### 3. Employee Login

**Endpoint:** POST /employee/login

**Description:** Authenticates employees using email and password.

**Request Body:**

json

```
{  
  "email": "ahmed@example.com",  
  "password": "123456"  
}
```

**Response:** Returns authentication token and employee details.

---

## Reports Endpoints

### 1. Get All Reports

**Endpoint:** `GET /all-reports`

**Description:** Retrieves all reports in the system.

**Authentication:** Not required

**Response:** List of all reports with basic information.

---

### 2. Get Report Data

**Endpoint:** `GET /reports/{id}`

**Description:** Retrieves detailed information for a specific report.

**Parameters:**

- `id` (path parameter): Report ID

**Authentication:** Not required

**Example:** `GET /reports/1`

---

## Admin Endpoints

All admin endpoints require Bearer token authentication with admin privileges.

### Report Types Management

Get Report Types

**Endpoint:** `GET /admin/report-types`

**Description:** Retrieves all report types.

**Authentication:** Required (Admin)

---

### Create Report Type

**Endpoint:** `POST /admin/report-types`

**Description:** Creates a new report type.

**Authentication:** Required (Admin)

**Request Body:**

```
json
{
  "name": "جريمة قتل",
  "notes": "Test"
}
```

---

### Show Report Type

**Endpoint:** `GET /admin/report-types/{id}`

**Description:** Retrieves a specific report type.

**Authentication:** Required (Admin)

**Parameters:**

- `id` (path parameter): Report type ID
- 

### Update Report Type

**Endpoint:** `PUT /admin/report-types/{id}`

**Description:** Updates an existing report type.

**Authentication:** Required (Admin)

**Parameters:**

- `id` (path parameter): Report type ID

## Request Body:

json

```
{
  "name": "حفرة في الطريق",
  "notes": "الابلاغ عن حفرة في الطريق"
}
```

---

## Delete Report Type

**Endpoint:** DELETE /admin/report-types/{id}

**Description:** Deletes a report type.

**Authentication:** Required (Admin)

### Parameters:

- `id` (path parameter): Report type ID
- 

## Procedures Management

### Get Procedures

**Endpoint:** GET /admin/procedures

**Description:** Retrieves all procedures.

**Authentication:** Required (Admin)

---

### Create Procedure

**Endpoint:** POST /admin/procedures

**Description:** Creates a new procedure.

**Authentication:** Required (Admin)

## Request Body:

json

```
{  
  "name": "Test",  
  "description": "Test"  
}
```

---

## Show Procedure

**Endpoint:** `GET /admin/procedures/{id}`

**Description:** Retrieves a specific procedure.

**Authentication:** Required (Admin)

### Parameters:

- `id` (path parameter): Procedure ID
- 

## Update Procedure

**Endpoint:** `PUT /admin/procedures/{id}`

**Description:** Updates an existing procedure.

**Authentication:** Required (Admin)

### Parameters:

- `id` (path parameter): Procedure ID

### Request Body:

json

```
{  
  "name": "Test3",  
  "description": "Test3"  
}
```

---

## Delete Procedure

**Endpoint:** `DELETE /admin/procedures/{id}`

**Description:** Deletes a procedure.

**Authentication:** Required (Admin)

**Parameters:**

- `{id}` (path parameter): Procedure ID
- 

## Employee Management

Get All Employees

**Endpoint:** `GET /admin/employees`

**Description:** Retrieves all employees.

**Authentication:** Required (Admin)

---

Add Employee

**Endpoint:** `POST /admin/employees`

**Description:** Creates a new employee account.

**Authentication:** Required (Admin)

**Request Body:**

```
json
{
  "name": "Ahmed Hassan",
  "email": "ahmed@example.com",
  "password": "123456",
  "is_admin": false
}
```

---

Update Employee

**Endpoint:** `PUT /admin/employees/{id}`

**Description:** Updates an existing employee.

**Authentication:** Required (Admin)

**Parameters:**

- `id` (path parameter): Employee ID

**Request Body:**

json

```
{  
  "name": "Mohamed Mostafa",  
  "email": "mohamed@example.com",  
  "password": "12345678",  
  "is_admin": false  
}
```

**Note:** Password field is optional during updates.

---

### Delete Employee

**Endpoint:** `DELETE /admin/employees/{id}`

**Description:** Deletes an employee account.

**Authentication:** Required (Admin)

**Parameters:**

- `id` (path parameter): Employee ID
- 

## Employee Endpoints

All employee endpoints require Bearer token authentication with employee privileges.

### Report Procedure Management

#### Get Report Procedures

**Endpoint:** `GET /employee/reports/{report_id}/procedures`

**Description:** Retrieves all procedures associated with a specific report.

**Authentication:** Required (Employee)

**Parameters:**

- `report_id` (path parameter): Report ID



---

## Create Report Procedure

**Endpoint:** `POST /employee/report/{report_id}/procedures`

**Description:** Associates a procedure with a report.

**Authentication:** Required (Employee)

### Parameters:

- `report_id` (path parameter): Report ID

### Request Body:

json

```
{  
  "procedure_id": 1,  
  "notes": "تحويل الي الجهة المختصة"  
}
```

---

## Update Report Procedure

**Endpoint:** `PUT /employee/report/{report_id}/procedures/{procedure_id}`

**Description:** Updates notes for a report procedure.

**Authentication:** Required (Employee)

### Parameters:

- `report_id` (path parameter): Report ID
- `procedure_id` (path parameter): Procedure ID

### Request Body:

json

```
{  
  "notes": "تم التدخل و الحل حالا"  
}
```

---

## Delete Report Procedure

**Endpoint:** `DELETE /employee/report/{report_id}/procedures/{procedure_id}`

**Description:** Removes a procedure from a report.

**Authentication:** Required (Employee)

**Parameters:**

- `report_id` (path parameter): Report ID
  - `procedure_id` (path parameter): Procedure ID
- 

## Citizen Endpoints

All citizen endpoints require Bearer token authentication.

### User Reports Management

Create User Report

**Endpoint:** `POST /user/reports`

**Description:** Allows citizens to submit new reports.

**Authentication:** Required (Citizen)

**Content-Type:** `multipart/form-data`

**Form Data:**

- `description` (text): Report description (e.g., "هناك جريمة سرقة")
  - `ll` (text): Latitude coordinate (e.g., "20.589")
  - `lg` (text): Longitude coordinate (e.g., "20.589")
  - `location` (text): Address or location description (e.g., "ش المعز لدين الله الحسين 16")
  - `report_type_id` (text): ID of the report type (e.g., "1")
  - `file` (file): Optional file attachment (image, document, etc.)
  - `report_no` (text): Report number (e.g., "REP-202569")
- 

Get User Reports

**Endpoint:** `GET /user/reports`

**Description:** Retrieves all reports submitted by the authenticated citizen.

**Authentication:** Required (Citizen)

**Response:** List of reports created by the current user.

---

## Error Handling

The API returns appropriate HTTP status codes:

- `200 OK`: Successful request
- `201 Created`: Resource created successfully
- `400 Bad Request`: Invalid request parameters
- `401 Unauthorized`: Authentication required or invalid token
- `403 Forbidden`: Insufficient permissions
- `404 Not Found`: Resource not found
- `500 Internal Server Error`: Server error

## Rate Limiting

Please implement appropriate rate limiting on the client side to avoid overwhelming the server.

## Support

For API support and questions, please contact the development team.

---

*Last Updated: May 27, 2025*