

# Visitor Management System - Postman Collection

This directory contains a complete Postman collection for testing and interacting with the Visitor Management System API.

## Contents

- `vms-api-collection.json` - The complete API collection
- `vms-environment.json` - Environment variables for the collection

## Setup Instructions

### 1. Import the Collection and Environment

1. Open Postman
2. Click on “Import” in the top left corner
3. Select both files:
  - `vms-api-collection.json`
  - `vms-environment.json`
4. Click “Import”

### 2. Select the Environment

1. After importing, select the “Visitor Management System Environment” from the environment dropdown in the top right corner
2. Verify the `baseUrl` is set correctly (default is `http://localhost:3000` )

### 3. Authentication Flow

Before using the API endpoints that require authentication, you need to log in:

1. Go to the “Authentication” folder in the collection
2. First run the “Get CSRF Token” request
  - This will automatically set the `csrfToken` environment variable
3. Then run the “Login” request with your credentials
  - This will automatically set the `authToken` environment variable

4. You can verify your authentication by running the “Auth Status” request

## 4. API Testing Workflow

The collection is organized into logical folders for different aspects of the system:

1. **Authentication** - Login, logout, and session management
2. **Dashboard** - Dashboard data endpoints
3. **Visitors** - Visitor management
4. **Visits** - Visit management and status updates
5. **Hosts** - Host user management
6. **Departments** - Department management
7. **Analytics** - Data analytics endpoints
8. **System** - System-level endpoints like health checks

## 5. Variable Management

The collection uses several environment variables to store values between requests:

- `baseUrl` - The base URL of the API
- `csrfToken` - The CSRF token for authentication
- `authToken` - The authentication token
- `visitorId` - ID of a created visitor
- `hostId` - ID of a created host
- `departmentId` - ID of a created department
- `visitId` - ID of a created visit

These variables are automatically set when you create resources and then used in subsequent requests.

## Testing Sequence

For a complete testing flow, follow this sequence:

1. Authentication (login)
2. Create a department (store departmentId)
3. Create a host (store hostId)
4. Create a visitor (store visitorId)
5. Create a visit (store visitId)

6. Test status updates on the visit
7. Test analytics endpoints
8. Test dashboard endpoints

## Notes

- Some endpoints require SuperAdmin privileges
- The collection includes test scripts that automatically set environment variables
- Pre-request scripts are used where necessary to prepare request data
- All requests include detailed descriptions of their purpose and required parameters