📘 API Documentation Template

# 📌 Project Overview

Project Name: Your API Project Name

Base URL: https://api.example.com/v1

Auth Type: Bearer Token / API Key / None

Content-Type: application/json

# 🔐 Authentication

## 🔑 Login

Endpoint: POST /auth/login

Description: Authenticate user and return JWT token.

Request Body:

{  
 "email": "user@example.com",  
 "password": "string"  
}

Response:

{  
 "token": "jwt-token-here",  
 "user": {  
 "id": "123",  
 "name": "John Doe"  
 }  
}

Status Codes:

- 200 OK – Login successful

- 401 Unauthorized – Invalid credentials

# 📂 API Endpoints

## 🧾 1. Get All Users

Method: GET

URL: /users

Auth Required: ✅

Response:

{  
 "data": [  
 {  
 "id": "1",  
 "name": "Jane Doe",  
 "email": "jane@example.com"  
 }  
 ],  
 "pagination": {  
 "page": 1,  
 "limit": 10,  
 "totalPages": 5  
 }  
}

## 👤 2. Get Single User

Method: GET

URL: /users/:id

Auth Required: ✅

Response:

{  
 "id": "1",  
 "name": "Jane Doe",  
 "email": "jane@example.com"  
}

## ✍️ 3. Create User

Method: POST

URL: /users

Auth Required: ✅

Request Body:

{  
 "name": "Jane Doe",  
 "email": "jane@example.com",  
 "password": "secret"  
}

Response:

{  
 "message": "User created successfully",  
 "userId": "1"  
}

## 🗑️ 4. Delete User

Method: DELETE

URL: /users/:id

Auth Required: ✅

Response:

{  
 "message": "User deleted successfully"  
}

# 📌 Error Format

All error responses follow the same format:

{  
 "error": true,  
 "message": "Detailed error message",  
 "code": 400  
}

# 📈 Rate Limiting

Limit Type: Requests/minute  
Value: 60  
Exceed Response: 429 Too Many Requests

# 📎 Appendix

- 📄 Status Codes Reference

- 🔐 JWT Token Format

- 📁 Environment Configs

# 🧪 Example Environment (Postman)

Environment: Production  
Base URL: https://api.example.com/v1  
Token: Bearer YOUR\_JWT\_TOKEN