

Danh sách các api sử dụng

URL: <http://localhost:3000>

1. Gửi Dữ Liệu Cảm Biến

Mô tả	Nhận dữ liệu từ các cảm biến (Flame Sensor, MQ2, MQ135) và gửi lên máy chủ để xử lý.
URL	/api/v1/sensors/data
Phương thức	POST
Tham số yêu cầu	device_id (string): ID của thiết bị IoT. flame_sensor (boolean): True nếu phát hiện cháy. mq2_gas_level (integer): Nồng độ khí gas từ cảm biến MQ2. mq135_air_quality (integer): Chỉ số chất lượng không khí từ MQ135. timestamp (string): Thời gian (ISO 8601).
Ví dụ yêu cầu	<pre>{ "device_id": "esp8266_001", "flame_sensor": true, "mq2_gas_level": 500, "mq135_air_quality": 300, "timestamp": "2024-10-13T14:00:00Z" }</pre>
Phản hồi	<pre>{ "code": 200, "data": [], "status": "success", "message": "Dữ liệu đã được nhận và đang xử lý.", "error": "" }</pre>
Mã phản hồi	200 OK : Dữ liệu đã được nhận. 400 Bad Request : Dữ liệu yêu cầu không hợp lệ.

2. Lưu Dữ Liệu Vào Cơ Sở Dữ Liệu

Mô tả	Lưu trữ dữ liệu cảm biến vào cơ sở dữ liệu.
URL	/api/v1/data/save
Phương thức	POST
Tham số yêu cầu	device_id (string), flame_sensor

	(boolean), mq2_gas_level (integer), mq135_air_quality (integer), timestamp (string).
Ví dụ yêu cầu	{ "device_id": "esp8266_001", "flame_sensor": true, "mq2_gas_level": 500, "mq135_air_quality": 300, "timestamp": "2024-10-13T14:00:00Z" }
Phản hồi	{ "code": 200, "data": {}, "status": "success", "message": "Dữ liệu đã được lưu thành công.", "error": "" }
Mã phản hồi	200 OK: Dữ liệu đã lưu thành công. 500 Internal Server Error: Lỗi khi lưu trữ dữ liệu.

3. Gửi Thông Báo Đến Người Dùng

Mô tả	Gửi thông báo cảnh báo đến người dùng qua ứng dụng Mobile hoặc Web.
URL	/api/v1/notifications/send
Phương thức	POST
Tham số yêu cầu	user_id (string): ID người dùng. message (string): Nội dung thông báo. timestamp (string): Thời gian (ISO 8601).
Ví dụ yêu cầu	{ "user_id": "user_001", "message": "Cảnh báo! Phát hiện cháy.", "timestamp": "2024-10-13T14:05:00Z" }
Phản hồi	{ "code": 200, "data": [], "status": "success", "message": "Thông báo đã được gửi tới người dùng.", "error": "" }

Mã phản hồi	200 OK: Thông báo đã gửi thành công. 400 Bad Request: Dữ liệu yêu cầu không hợp lệ.
-------------	--

4. Gọi Lực Lượng Cứu Hỏa

Mô tả	Gửi yêu cầu tự động gọi lực lượng cứu hỏa khi phát hiện cháy nghiêm trọng.
URL	/api/v1/emergency/call
Phương thức	POST
Tham số yêu cầu	location (string): Địa điểm xảy ra sự cố. incident_details (string): Chi tiết sự cố. timestamp (string): Thời gian (ISO 8601).
Ví dụ yêu cầu	{ "location": "123 ABC Street", "incident_details": "Phát hiện cháy lớn.", "timestamp": "2024-10-13T14:10:00Z" }
Phản hồi	{ "code": 200, "data": [], "status": "success", "message": "Lực lượng cứu hỏa đã được thông báo.", "error": "" }
Mã phản hồi	200 OK: Lực lượng cứu hỏa đã được thông báo. 500 Internal Server Error: Lỗi hệ thống.

5. Thông Báo Cho Người Thân

Mô tả	Gửi thông báo cho người thân khi phát hiện sự cố liên quan đến người dùng.
URL	/api/v1/notifications/family
Phương thức	POST
Tham số yêu cầu	user_id (string): ID người dùng. family_member_id (string): ID người thân. message (string): Nội dung thông báo. timestamp (string): Thời gian (ISO 8601).
Ví dụ yêu cầu	{ "user_id": "user_001", "family_member_id": "family_001", "message": "Người thân của bạn đang gặp nguy" }

	hiếm.", "timestamp": "2024-10-13T14:12:00Z" }
Phản hồi	{ "code": 200, "data": [], "status": "success", "message": "Thông báo đã được gửi đến người thân.", "error": "" }
Mã phản hồi	200 OK: Thông báo đã gửi thành công. 400 Bad Request: Tham số yêu cầu không hợp lệ.

6. Lấy Lịch Sử Cảnh Báo

Mô tả	Truy xuất lịch sử các thông báo cảnh báo đã được gửi cho người dùng.
URL	/api/v1/history
Phương thức	GET
Tham số yêu cầu	{}
Ví dụ yêu cầu	/api/v1/history? user_id=user_001&start_date=2024-10-13&end_date=2024-10-13
Phản hồi	{ "code": 404, "data": [], "status": "fail", "message": "Không tìm thấy lịch sử.", "error": "" }
Mã phản hồi	200 OK: Lịch sử đã được truy xuất thành công. 404 Not Found: Không tìm thấy lịch sử.

7. Kiểm Tra Trạng Thái Hệ Thống IoT

Mô tả	Kiểm Tra Trạng Thái Hệ Thống IoT
URL	/api/v1/iot/status
Phương thức	GET
Tham số yêu cầu	{}
Ví dụ yêu cầu	{ "status": "success",

	<pre>"iot_status": { "flame_sensor": "active", "mq2_sensor": "active", "mq135_sensor": "active", "buzzer": "inactive" } }</pre>
Phản hồi	<pre>{ "code": 200, "data": { "flame_sensor": "active", "mq2_sensor": "active", "mq135_sensor": "active" }, "status": "success", "message": "Trạng thái IoT đã được truy xuất thành công.", "error": "" }</pre>
Mã phản hồi	<p>200 OK: Trạng thái IoT đã được truy xuất thành công.</p> <p>500 Internal Server Error: Lỗi hệ thống.</p>

8. Lấy dữ liệu hướng dẫn phòng cháy chữa cháy và tin tức cảnh báo từ hệ thống.

Mô tả	Lấy dữ liệu hướng dẫn phòng cháy chữa cháy và tin tức cảnh báo từ hệ thống.
URL	/api/v1/guides_and_news
Phương thức	GET
Tham số yêu cầu	<p>category (string): Loại dữ liệu cần lấy (ví dụ: "guide" hoặc "news").</p> <p>limit (integer): Số lượng mục cần lấy, mặc định là</p>
Ví dụ yêu cầu	/api/v1/guides_and_news?category=guide&limit=5
Phản hồi	<pre>{ "code": 200, "data": [{ "id": 1, "title": "Cách sử dụng bình chữa cháy", "type": "video", "url": "https://example.com/video1", "content": null }] }</pre>

	<pre> }, { "id": 2, "title": "Cách thoát hiểm khi có cháy", "type": "article", "url": null, "content": "https://example.com/article1" }], "status": "success", "message": "Dữ liệu đã được truy xuất thành công.", "error": "" }</pre>
Mã phản hồi	200 OK: Dữ liệu đã được truy xuất thành công. 404 Not Found: Không có dữ liệu phù hợp.

9. Thêm dữ liệu về hướng dẫn phòng cháy chữa cháy và tin tức cảnh báo

Mô tả	Thêm một hoặc nhiều mục dữ liệu về hướng dẫn phòng cháy chữa cháy và tin tức cảnh báo vào hệ thống.
URL	/api/v1/guides_and_news/add
Phương thức	POST
Tham số yêu cầu	<p>Một mảng chứa các đối tượng với các trường sau:</p> <ul style="list-style-type: none"> <code>title</code> (string): Tiêu đề của hướng dẫn hoặc tin tức. <code>type</code> (string): Loại dữ liệu (ví dụ: "video", "article"). <code>url</code> (string): URL của tài nguyên (nếu có). <code>content</code> (string): Nội dung bài viết (nếu có). <code>category</code> (string): Phân loại dữ liệu (ví dụ: "guide" hoặc "news").
Ví dụ yêu cầu	<pre>[{ "title": "Cách sử dụng bình chữa cháy", "type": "video", "url": "https://example.com/video1", "content": null, "category": "guide" }]</pre>

	<pre> }, { "title": "Cách thoát hiểm khi có cháy", "type": "article", "url": null, "content": "Hướng dẫn cách thoát hiểm khi có cháy", "category": "guide" }] </pre>
Phản hồi	<pre> { "code": 200, "data": [], "status": "success", "message": "Dữ liệu đã được thêm thành công.", "error": "" } </pre>
Mã phản hồi	<p>200 OK: Dữ liệu đã được thêm thành công.</p> <ul style="list-style-type: none"> • 400 Bad Request: Dữ liệu không hợp lệ. • 500 Internal Server Error: Lỗi hệ thống khi thêm dữ liệu.

10. Lưu trạng thái thiết bị IoT

Mô tả	Lưu trạng thái của một hoặc nhiều thiết bị IoT (cảm biến, báo động, v.v.) vào cơ sở dữ liệu.
URL	/api/v1/iot/status/save
Phương thức	POST
Tham số yêu cầu	<p>Tham số yêu cầu:</p> <ul style="list-style-type: none"> • Một mảng chứa các đối tượng với các trường sau: <ul style="list-style-type: none"> • device_name (string): Tên của thiết bị IoT (ví dụ: "flame_sensor", "mq2_sensor", "buzzer"). • status (string): Trạng thái của thiết bị (ví dụ: "active", "inactive", "error").

	<ul style="list-style-type: none"> <code>timestamp</code> (string): Thời gian ghi nhận trạng thái (ISO 8601).
Ví dụ yêu cầu	<pre>[{ "device_name": "flame_sensor", "status": "active", "timestamp": "2024-10-14T10:00:00Z" }, { "device_name": "mq2_sensor", "status": "active", "timestamp": "2024-10-14T10:00:00Z" }, { "device_name": "mq135_sensor", "status": "active", "timestamp": "2024-10-14T10:00:00Z" }]</pre>
Phản hồi	<pre>{ "code": 200, "data": [], "status": "success", "message": "Trạng thái của các thiết bị đã được lưu thành công.", "error": "" }</pre>
Mã phản hồi	<p>200 OK: Trạng thái thiết bị đã được lưu thành công.</p> <ul style="list-style-type: none"> 400 Bad Request: Dữ liệu không hợp lệ. 500 Internal Server Error: Lỗi hệ thống khi lưu trạng thái.