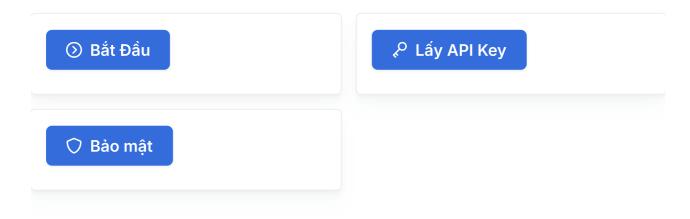
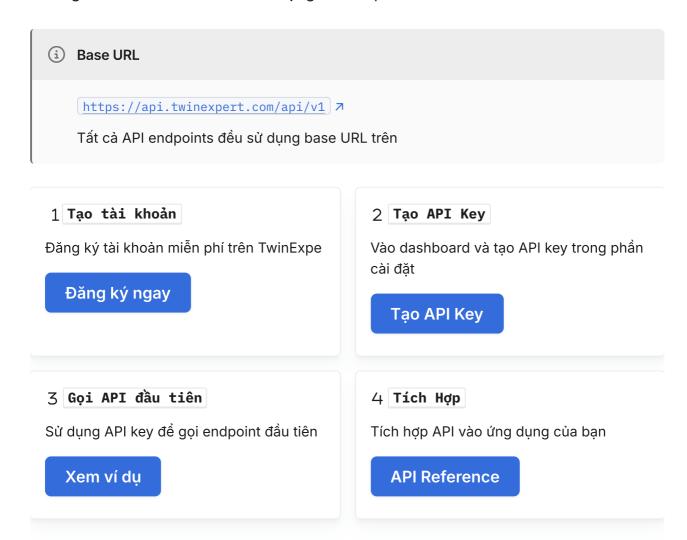
# 5. DÀNH CHO NHÀ PHÁT TRIÊN (TwinExpert API Documentation)

Tích hợp AI twins vào ứng dụng của bạn với API RESTful mạnh mẽ và dễ sử dụng.



# 5.1. Bắt đầu với API

Hướng dẫn nhanh để bắt đầu sử dụng TwinExpert API



# 5.2. Xác thực

TwinExpert API sử dụng API key để xác thực requests



#### Bảo mật API Key

API key của bạn có quyền truy cập vào dữ liệu. Hãy giữ bí mật và không chia sẻ trong code phía client.

# Cách sử dụng API Key

Thêm API key vào header Authorization của mọi request:

#### **Authorization Header**

Authorization: Bearer YOUR\_API\_KEY

#### Hoặc sử dụng X-API-Key header

X-API-Key: YOUR\_API\_KEY

#### **Post**

/api/v1/auth/api-keys

### <>Ví dụ code

Tạo API key mới bằng JWT token

#### JavaScript

```
// Tạo API key mới
const response = await
fetch('https://api.twinexpert.com/api/v1/auth/api-keys', {
  method: 'POST',
  headers: {
    'Authorization': 'Bearer YOUR_JWT_TOKEN',
    'Content-Type': 'application/json'
    },
  body: JSON.stringify({
      name: 'My App',
      scope: 'full_access'
    })
});
const data = await response.json();
console.log('API Key:', data.data.key);
```

#### **cURL**

```
# Tao API key mới
curl -X POST "https://api.twinexpert.com/api/v1/auth/api-keys" \
    -H "Authorization: Bearer YOUR_JWT_TOKEN" \
    -H "Content-Type: application/json" \
    -d '{
        "name": "My App",
        "scope": "full_access"
}'
```

# Python import requests # Tạo API key mới headers = { 'Authorization': 'Bearer YOUR\_JWT\_TOKEN', 'Content-Type': 'application/json' 3 $data = {$ 'name': 'My App', 'scope': 'full\_access' } response = requests.post( 'https://api.twinexpert.com/api/v1/auth/api-keys', headers=headers, json=data ) api\_key = response.json()['data']['key'] print(f'API Key: {api\_key}')

# Phạm vi quyền (Scopes)

```
read_only Chỉ Đọc

Xem danh sách Twins

Đọc thông tin cuộc trò chuyên

Xem lịch sử tin nhắn

200 requests/giờ
```

```
Tất cả quyền của Read_only

Tạo cuộc trò chuyện mới

Gửi tin nhắn

500 requests/giờ
```

# 5.3. API Endpoints

Danh sách đầy đủ các endpoints có sẵn

∨ GET /api/v1/twins Chi đọc

Lấy danh sách Twins có thể truy cập

#### **Parameters**

```
page integer Số trang (1-based)
```

limit integer Số items mỗi trang (max 100)

∨ GET /api/v1/twins/{id} Chỉ đọc

Lấy thông tin chi tiết twin

#### **Parameters**

id string required ID của twin

∨ POST /api/v1/twins/{id} Toàn quyền

Tạo cuộc trò chuyện mới

#### **Parameters**

twinId string required

ID của twin

title string Tiêu đề cuộc trò chuyện

✓ GET /api/v1/conversations Chi đọc

Lấy danh sách cuộc trò chuyện

#### **Parameters**

page Integer Số trang

limit integer Số items mỗi trang

twinId string Loc theo twin ID

POST /api/v1/conversations/{id}/messages Toàn quyền

Gửi tin nhắn

#### **Parameters**

id string required ID cuộc trò chuyện

content string required Nội dung tin nhắn

∨ POST /api/v1/conversations/{id}/messages/stream Toàn quyền

Gửi tin nhắn với phản hồi streaming

#### **Parameters**

id string required ID cuộc trò chuyện

content string required Nội dung tin nhắn

# 5.4. Ví dụ

Các ví dụ code thực tế để bắt đầu nhanh

# 1. Lấy danh sách Al Twins

Xem tất cả AI twins mà bạn có thể truy cập

GET /api/v1/twins

```
Thừ nghiệm API

API Key

123456

(i) Kết quà

{
    "statusCode": 401,
    "timestamp": "2025-07-23T17:06:03.675Z",
    "path": "/api/v1/twins",
    "method": "GET",
    "message": "Invalid API key format",
    "error": "Unauthorized"
}
```

# <> Ví dụ code

Lấy danh sách twins có thể truy cập

#### JavaScript

```
// Lấy danh sách twins
const response = await
fetch('https://api.twinexpert.com/api/v1/twins', {
  headers: {
    'Authorization': 'Bearer 123456',
    'Content-Type': 'application/json'
  }
});
const twins = await response.json();
console.log('Available twins:', twins.data);
```

#### Python

```
import requests

# Lấy danh sách twins
headers = {
    'Authorization': 'Bearer 123456',
    'Content-Type': 'application/json'
}

response = requests.get(
    'https://api.twinexpert.com/api/v1/twins',
    headers=headers
)

twins = response.json()['data']
print(f'Found {len(twins)} twins')
```

#### cURL

```
# Lấy danh sách twins
curl -X GET "https://api.twinexpert.com/api/v1/twins" \
   -H "Authorization: Bearer 123456" \
   -H "Content-Type: application/json"
```

# 2. Chat với Al Twin

Tạo cuộc trò chuyện và gửi tin nhắn đến AI twin

GET /api/v1/conversations

# <> Ví dụ code

Tạo cuộc trò chuyện và gửi tin nhắn đến AI twin

#### **JavaScript**

```
// Tao cuốc trò chuyên và gửi tin nhắn
async function chatWithTwin(twinId, message) {
  // 1. Tạo cuộc trò chuyện mới
  const conversationResponse = await
fetch('https://api.twinexpert.com/api/v1/conversations', {
    method: 'POST',
    headers: {
      'Authorization': 'Bearer YOUR_API_KEY',
      'Content-Type': 'application/json'
    },
    body: JSON.stringify({
     twinId: twinId,
     title: 'API Chat Session'
    })
  });
  const conversation = await conversationResponse.json();
  const conversationId = conversation.data.id;
  // 2. Gửi tin nhắn
  const messageResponse = await
fetch(`${process.env.NEXT_PUBLIC_API_URL}/api/v1/conversations/${con
versationId}/messages`, {
    method: 'POST',
    headers: {
      'Authorization': 'Bearer YOUR_API_KEY',
      'Content-Type': 'application/json'
    },
    body: JSON.stringify({
      content: message
    })
  });
  const result = await messageResponse.json();
  return result.data;
7
// Sử dụng
chatWithTwin('twin-id', 'Xin chào!').then(result => {
 console.log('Response:', result);
3);
```

#### Python

```
import requests
class TwinExpertAPI:
    def __init__(self, api_key):
        self.api_key = api_key
        self.base_url = 'https://api.twinexpert.com/api/v1'
        self.headers = {
            'Authorization': f'Bearer {api_key}',
            'Content-Type': 'application/json'
        3
    def create_conversation(self, twin_id, title="API Chat"):
        response = requests.post(
            f'{self.base_url}/conversations',
            headers=self.headers,
            json={'twinId': twin_id, 'title': title}
        return response.json()['data']
    def send_message(self, conversation_id, message):
        response = requests.post(
f'{self.base_url}/conversations/{conversation_id}/messages',
            headers=self.headers,
            json={'content': message}
        return response.json()['data']
    def chat_with_twin(self, twin_id, message):
        # Tạo cuộc trò chuyện
        conversation = self.create_conversation(twin_id)
        # Gửi tin nhắn
        result = self.send_message(conversation['id'], message)
        return result
# Sử dụng
api = TwinExpertAPI('YOUR_API_KEY')
response = api.chat_with_twin('twin-id', 'Xin chào!')
print(response)
```

cURL

```
# 1. Tao cuộc trò chuyện mới
curl -X POST "https://api.twinexpert.com/api/v1/conversations" \
    -H "Authorization: Bearer YOUR_API_KEY" \
    -H "Content-Type: application/json" \
    -d '{
        "twinId": "twin-id",
        "title": "API Chat Session"
}'

# 2. Gửi tin nhắn (với conversation ID từ response trên)
curl -X POST
"https://api.twinexpert.com/api/v1/conversations/CONVERSATION_ID/mes
sages" \
    -H "Authorization: Bearer YOUR_API_KEY" \
    -H "Content-Type: application/json" \
    -d '{
        "content": "Xin chào!"
}'
```

### 2. Streaming Chat

Nhận phản hồi real-time bằng Server-Sent Events

#### **ॐ** Streaming Response

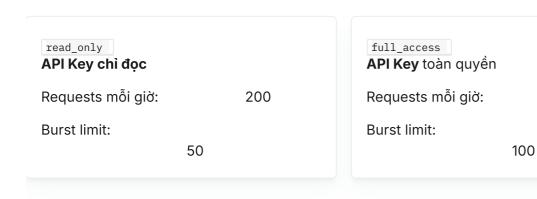
Endpoint streaming trả về Server-Sent Events với các event types:

- progress: Thông tin tiến trình
- delta: Từng phần của response
- complete: Hoàn thành message
- error: Lỗi xảy ra

# 5.5. Rate Limiting

Giới hạn tần suất gọi API để đảm bảo chất lượng dịch vụ

### **Rate Limiting**



# **Response Headers**

Mọi response đều bao gồm các headers về rate limiting:

X-RateLimit-Limit Tổng số requests được phép trong window
 X-RateLimit-Remaining Số requests còn lại
 X-RateLimit-Reset Timestamp khi counter reset
 X-RateLimit-Window Thời gian window (giây)

500

# 5.6. Xử lý lỗi

Hiểu các mã lỗi và cách xử lý chúng

# Định dạng lỗi

Tất cả lỗi API đều trả về theo format chuẩn:

```
"success": false,
"message": "Mô tả lỗi cho người dùng",
"error": "ERROR_CODE",
"statusCode": 400,
"meta": {
    "timestamp": "2024-01-15T10:30:00Z",
    "requestId": "req_abc123"
},
"details": {
    // Thông tin chi tiết lỗi (nếu có)
}
```

# Mã lỗi phổ biến

### √ 400 Bad Request

Request không hợp lệ hoặc thiếu parameters

#### √ 401 Unauthorized

API key không hợp lệ hoặc đã hết hạn

### √ 403 Forbidden

API key không có quyền truy cập resource này

### √ 404 Not Found

Resource không tồn tại

### √ 429 Too Many Requests

Vượt quá rate limit

# √ 500 Internal Server Error

Lỗi server, vui lòng thử lại sau