AI Optimization Engine API - Complete Endpoints Reference

🔐 Authentication Flow

Start here to get your access token!

1. Register User

`POST /register` *(No auth required)*

```json
{
  "email": "user@example.com",
  "password": "securepassword123",
  "full_name": "John Doe",
  "company": "Example Corp"
}
```

2. Login User

`POST /login` *(No auth required)*

```json
{
  "email": "user@example.com",
  "password": "securepassword123"
}
```

Response contains your access token!

3. Request Password Reset

`POST /password-reset` *(No auth required)*

```json
{
  "email": "user@example.com"
}
```

4. Confirm Password Reset

POST /password-reset/confirm *(No auth required)*

```
json

{
  "email": "user@example.com",
  "token": "reset_token_123",
  "new_password": "newSecurePassword456"
}
```

🚀 Core Functionality

Main AI optimization features

5. Health Check

GET /health *(No auth required)* No parameters needed

6. Root Info

GET / *(No auth required)* API information endpoint

7. Analyze Brand ⭐

POST /analyze-brand *(Requires auth)*

```
json

{
  "brand_name": "TechCorp",
  "website_url": "https://techcorp.com",
  "product_categories": ["software", "cloud services", "AI tools"],
  "content_sample": "We provide cutting-edge AI solutions for modern businesses...",
  "competitor_names": ["CompetitorA", "CompetitorB"]
}
```

8. Calculate Optimization Metrics ⭐

POST /optimization-metrics *(Requires auth)*

```
json
```

```
{
  "brand_name": "TechCorp",
  "content_sample": "Our AI-powered platform delivers exceptional results for businesses",
  "website_url": "https://techcorp.com"
}
```

9. Analyze Queries ⭐

`POST /analyze-queries` *(Requires auth)*

```json
{
  "brand_name": "TechCorp",
  "product_categories": ["AI software", "automation tools", "business intelligence"]
}
```

📊 Brand Management

Manage your brands and history

10. List Brands

`GET /brands` *(Requires auth)* No parameters needed

11. Get Brand History

`GET /brands/{brand_name}/history` *(Requires auth)*

- Path parameter: `brand_name` (e.g., "TechCorp")

📝 Server Logs Analysis

Upload and analyze server logs

12. Upload Server Log

`POST /api/v2/logs/upload` *(Requires auth)* Form data:

- `file`: Log file to upload
- `brand_id`: "brand123"
- `log_format`: "apache_combined"

- `timezone`: "UTC" (optional)

13. Get Upload Status

`GET /api/v2/logs/upload/{upload_id}/status` *(Requires auth)*

- Path parameter: `upload_id`

14. Get Log Analysis

`GET /api/v2/logs/analysis/{brand_id}` *(Requires auth)*

- Path parameter: `brand_id`
- Query parameter: `days` (default: 30)

15. Get Bot Activity

`GET /api/v2/logs/bot-activity/{brand_id}` *(Requires auth)*

- Path parameter: `brand_id`
- Query parameters: `days` (default: 7), `platform` (optional)

16. Analyze Log Sample

`POST /api/v2/logs/analyze-sample` *(Requires auth)* Query parameters:

- `log_sample`: "127.0.0.1 - - [25/Dec/2023:10:00:00 +0000] "GET /api/brands HTTP/1.1" 200 1234"
- `log_format`: "apache_combined"

👑 Admin Endpoints

Admin-only features (requires admin role)

User Management

17. List Users

`GET /api/v2/admin/users` *(Admin only)* Query parameters (all optional):

- `skip`: 0
- `limit`: 20
- `search`: "john"
- `role`: "client" or "admin"
- `status`: "active"

- `plan`: "free", "bring_your_own_key", "platform_managed", or "enterprise"

18. Toggle User Status

`POST /api/v2/admin/users/{user_id}/toggle-status` *(Admin only)*

- Path parameter: `user_id`

19. Update User Role

`PUT /api/v2/admin/users/{user_id}/role` *(Admin only)*

- Path parameter: `user_id`
- Query parameter: `new_role` ("client" or "admin")

Subscription Management

20. List Subscriptions

`GET /api/v2/admin/subscriptions` *(Admin only)* Query parameters (all optional):

- `status`: "active"
- `plan`: "free", "bring_your_own_key", "platform_managed", or "enterprise"
- `skip`: 0
- `limit`: 20

21. Update Subscription

`PUT /api/v2/admin/subscriptions/{subscription_id}` *(Admin only)*

- Path parameter: `subscription_id`

```json
{
  "plan": "enterprise",
  "status": "active",
  "credits_remaining": 1000
}
```

System Metrics

22. Get System Overview

`GET /api/v2/admin/metrics/overview` *(Admin only)* No parameters needed

23. Get API Usage Metrics

`GET /api/v2/admin/metrics/api-usage` *(Admin only)* Query parameters (optional):

- `days`: 7 (max: 90)
- `provider`: "openai", "anthropic", "google", or "perplexity"

Error Management

24. Get Error Logs

`GET /api/v2/admin/errors` *(Admin only)* Query parameters (all optional):

- `severity`: "error"
- `category`: "authentication"
- `unresolved_only`: true
- `skip`: 0
- `limit`: 50 (max: 200)

25. Resolve Error

`PUT /api/v2/admin/errors/{error_id}/resolve` *(Admin only)*

- Path parameter: `error_id`
- Query parameter: `resolution_notes` (required)

Activity & Improvements

26. Get Admin Activity Logs

`GET /api/v2/admin/activity-logs` *(Admin only)* Query parameters (all optional):

- `admin_id`: "admin123"
- `action`: "user_status_change"
- `days`: 7 (max: 90)

27. Get User Improvements

`GET /api/v2/admin/improvements` *(Admin only)* Query parameters (all optional):

- `status`: "pending"
- `category`: "feature_request"
- `skip`: 0

- `limit`: 20

28. Update Improvement Status

`PUT /api/v2/admin/improvements/{improvement_id}` *(Admin only)*

- Path parameter: `improvement_id`
- Query parameters:
  - `status`: "in_progress" (required)
  - `priority`: "high" (optional)
  - `admin_notes`: "Reviewing feasibility" (optional)

🔑 Authentication in Swagger UI

Step-by-Step:

1. **Click "Authorize"** button (🔒 icon) at top of Swagger UI
2. **First time**: Use `POST /register` then `POST /login`
3. **Copy the access_token** from login response
4. **In Authorization dialog**: Enter `Bearer YOUR_TOKEN_HERE`
5. **Click "Authorize"** then **"Close"**
6. **Now all 🔒 endpoints will work!**

Quick Test Sequence:

1. `GET /health` ✅ (no auth)
2. `POST /register` ✅ (create account)
3. `POST /login` ✅ (get token)
4. Authorize with token 🔑
5. `POST /analyze-brand` ⭐ (main feature)
6. `GET /brands` 📊 (see results)

📋 Testing Tips

Essential Endpoints to Test:

- **Core**: `/health`, `/analyze-brand`, `/optimization-metrics`
- **Auth**: `/register`, `/login`

- **Data**: `/brands`, `/brands/{name}/history`

Sample Brand for Testing:

```json
{
  "brand_name": "EcoTech Solutions",
  "website_url": "https://ecotech.com",
  "product_categories": ["renewable energy", "smart home", "sustainability"],
  "content_sample": "Leading provider of innovative eco-friendly technology solutions for homes and businesses.",
  "competitor_names": ["GreenTech Corp", "Sustainable Systems"]
}
```

Admin Testing:

- Need admin role to access `/api/v2/admin/*` endpoints
- Start with `/api/v2/admin/metrics/overview` for system status

🎯 Priority Order for Testing:

Basic Flow:

1. Health Check
2. Register/Login
3. Analyze Brand
4. List Brands

Advanced Flow: 5. Optimization Metrics 6. Query Analysis
7. Brand History 8. Log Analysis (if you have logs)

Admin Flow: 9. System Overview 10. User Management 11. Error Logs