

# คู่มือสถาปัตยกรรมและการปฏิบัติการระบบ TrendSiam

คู่มือฉบับสมบูรณ์สำหรับนักพัฒนาและผู้ดูแลระบบ

TrendSiam

เวอร์ชัน 1.0

วันที่จัดทำ: 20 สิงหาคม 2568

ระบบรวมและวิเคราะห์ข่าวแนวโน้มประเทศไทย  
Thai News Trend Aggregation & Analysis System

# สารบัญ

1. ภาพรวมสถาปัตยกรรมระบบ	หน้า 3
2. การโหลดของข้อมูล	หน้า 4
3. ความปลอดภัยของระบบ	หน้า 5
4. การใช้งาน Supabase	หน้า 6
5. คู่มือการปฏิบัติการ	หน้า 7
6. แนวทางการทดสอบ	หน้า 8
ภาคผนวก ก: บัญชีรายรับ-จ่ายไฟล์	หน้า 9
ภาคผนวก ข: รายการ API และเส้นทาง	หน้า 10+

## หมายเหตุ

- เอกสารนี้จัดทำขึ้นเพื่อใช้เป็นคู่มืออ้างอิงสำหรับนักพัฒนาและผู้ดูแลระบบ
- ข้อมูลในเอกสารอาจมีการเปลี่ยนแปลงตามการพัฒนาระบบ
- สำหรับข้อมูลล่าสุด กรุณาตรวจสอบจากเอกสารออนไลน์หรือติดต่อทีมพัฒนา

## 1. ການគົມສາປ້າຍກຮນຮະບູ

## 1.1 ຖະແນນສ່ວນ TrendSiam

ระบบ TrendSiam เป็นแพลตฟอร์มรวมและวิเคราะห์ข่าวແয়েโน้นจากแหล่งข้อมูลต่างๆ โดยเฉพาะจาก YouTube และสื่อออนไลน์ในประเทศไทย ระบบใช้เทคโนโลยี AI ในการจัดหมวดหมู่ สร้างรูป และสร้างภาพประกอบอัตโนมัติ

## 1.2 องค์ประกอบหลักของระบบ

# Frontend (Next.js)

- เว็บแอปพลิเคชันสำหรับแสดงผลข้อมูลแบบโน้ม
  - รายงาน PDF รายสัปดาห์
  - หน้าจอสำหรับผู้ดูและระบบ
  - API endpoints สำหรับการเชื่อมต่อข้อมูล

## Backend Processing (Python)

- ระบบดึงข้อมูลจาก YouTube API
  - การประมวลผลและจัดหมวดหมู่ด้วย AI
  - การสร้างสคริปภาษาไทยและอังกฤษ
  - การสร้างภาพประกอบด้วย AI
  - การคำนวณคะแนนความนิยม

## ฐานข้อมูล (Supabase)

- PostgreSQL database สำหรับเก็บข้อมูลข่าว
  - Authentication และ authorization
  - Storage สำหรับไฟล์ภาพ
  - Real-time subscriptions
  - Row Level Security (RLS) policies

### 1.3 เทคโนโลยีที่ใช้

## Frontend

- Next.js 14
  - React 18
  - TypeScript
  - Tailwind CSS

Backend

- Python 3.11+
  - OpenAI API
  - YouTube Data API
  - Supabase Python Client

- React-PDF
- Zustand (State Management)
- Pillow (Image Processing)
- BeautifulSoup4

## 1.4 การติดตั้งและใช้งาน

ระบบสามารถติดตั้งและใช้งานได้หลายรูปแบบ:

- Development: Local development server
- Staging: Vercel หรือ cloud platform อื่นๆ
- Production: Vercel + Supabase Cloud
- Self-hosted: Docker containers

## 2. การให้ผลของข้อมูล

### 2.1 การรวมข้อมูล

#### ขั้นตอนที่ 1: ดึงข้อมูลจาก YouTube

- ใช้ YouTube Data API v3 ดึงข้อมูลวิดีโอยอดนิยม
- กรองข้อมูลตามภูมิภาค (ประเทศไทย)
- ดึงข้อมูล metadata: ชื่อ, คำอธิบาย, จำนวนวิว, ไลค์, คอมเม้นต์
- บันทึกข้อมูลด้วย JSON

#### ขั้นตอนที่ 2: การประมวลผลด้วย AI

- ส่งข้อมูลไปยัง OpenAI API สำหรับการวิเคราะห์
- จัดหมวดหมู่เนื้อหา (บันเทิง, ข่าว, กีฬา, เทคโนโลยี, ฯลฯ)
- สร้างสรุปภาษาไทยและอังกฤษ
- สกัดคำสำคัญและแท็ก
- คำนวณคะแนนความเกี่ยวข้องและความนิยม

### 2.2 Pipeline การประมวลผล

YouTube API → Raw Data → AI Processing → Categorization → Summary Generation → Image Generation → Score Calculation → Database Storage → Frontend Display

รายละเอียดแต่ละขั้นตอน:

- Raw Data Collection: ดึงข้อมูลจาก API และบันทึกในรูปแบบดิบ
- Content Analysis: วิเคราะห์เนื้อหาด้วย AI เพื่อเข้าใจบริบท
- Categorization: จัดหมวดหมู่ตามประเภทเนื้อหา
- Summary Generation: สร้างสรุปที่เข้าใจง่าย
- Image Generation: สร้างภาพประกอบด้วย DALL-E
- Quality Scoring: คำนวณคะแนนความนิยมและคุณภาพ

### 2.3 โครงสร้างฐานข้อมูล

ตารางหลักในระบบ:

- news\_trends: ข้อมูลข่าวและแนวโน้มหลัก
- weekly\_report\_snapshots: ข้อมูลรายงานรายสัปดาห์

- ai\_generated\_images: ภาพที่สร้างด้วย AI

## 2.4 การอัปเดตแบบ Real-time

ระบบใช้ Supabase Real-time สำหรับการอัปเดตข้อมูลแบบกันกี

- การเจ้งเตือนเมื่อมีข้อมูลใหม่
- การอัปเดตสดติดแบบ live
- การซิงค์ข้อมูลระหว่างหน้าต่างๆ
- การเจ้งเตือนสถานะการประมวลผล

### 3. ความปลอดภัยของระบบ

#### 3.1 ภาพรวมความปลอดภัย

ระบบ TrendSiam ได้รับการออกแบบด้วยหลักการความปลอดภัยแบบ Defense in Depth โดยมีการป้องกันหลายชั้นตั้งแต่ระดับเครือข่าย แอปพลิเคชัน จนถึงฐานข้อมูล

#### 3.2 ความปลอดภัยของเว็บแอปพลิเคชัน

##### Content Security Policy (CSP)

- ป้องกันการโจมตีแบบ XSS (Cross-Site Scripting)
- จำกัดแหล่งที่มาของ script, style, และ image
- บล็อกการโหลด inline script ที่ไม่ปลอดภัย
- รายงานการละเมิดนโยบายไปยัง monitoring system

##### HTTP Security Headers

- X-Frame-Options: ป้องกัน clickjacking
- X-Content-Type-Options: ป้องกัน MIME type sniffing
- Referrer-Policy: ควบคุมการส่ง referrer information
- Strict-Transport-Security: บังคับใช้ HTTPS

##### Input Validation และ Sanitization

- ตรวจสอบและกำกับความสะอาดข้อมูลทุก API endpoint
- ใช้ Zod schema สำหรับ type validation
- Sanitize ข้อมูลก่อนแสดงผลใน PDF
- ป้องกัน SQL injection ด้วย parameterized queries

#### 3.3 ความปลอดภัยของ API

##### Rate Limiting

- จำกัดจำนวน request ต่อ IP address
- ป้องกันการโจมตีแบบ DDoS
- ใช้ sliding window algorithm
- แจ้งเตือนเมื่อมีการใช้งานผิดปกติ

## Authentication และ Authorization

- ใช้ Supabase Auth สำหรับการจัดการผู้ใช้
- JWT tokens สำหรับ session management
- Role-based access control (RBAC)
- Multi-factor authentication (MFA) สำหรับ admin

## 3.4 การจัดการข้อผิดพลาด

หลักการสำคัญในการจัดการข้อผิดพลาด

- ไม่เปิดเผยข้อมูลระบบภายในในข้อความ error
- Log ข้อผิดพลาดแบบละเอียดใน server-side เต็มที่
- ส่งข้อความ error ที่เป็นมิตรกับผู้ใช้
- ไม่ log ข้อมูลที่เป็นความลับ (API keys, passwords)
- ใช้ structured logging สำหรับการวิเคราะห์

## 3.5 การป้องกัน SSRF และ Path Traversal

- ตรวจสอบ URL ก่อนทำ HTTP requests
- บล็อก private IP ranges และ localhost
- Whitelist allowed domains สำหรับ external requests
- Validate file paths และป้องกัน directory traversal
- ใช้ sandbox environment สำหรับการประมวลผลไฟล์

## 4. การใช้งาน Supabase

### 4.1 ภาพรวม Supabase

Supabase เป็น Backend-as-a-Service ที่ให้บริการ PostgreSQL database, Authentication, Storage, และ Real-time subscriptions ระบบ TrendSiam ใช้ Supabase เป็นแกนหลักสำหรับการจัดเก็บและจัดการข้อมูล

### 4.2 โครงสร้างฐานข้อมูล

#### ตารางหลัก

##### news\_trends

เก็บข้อมูลข่าวและแนวโน้มหลัก รวมถึง metadata, คะแนน, และ AI analysis

##### weekly\_report\_snapshots

เก็บข้อมูล snapshot สำหรับรายงานรายสัปดาห์ ใช้สำหรับสร้าง PDF

##### ai\_generated\_images

เก็บข้อมูลภาพที่สร้างด้วย AI รวมถึง prompts และ metadata

#### Views และ Functions

- weekly\_report\_public\_v: View สำหรับข้อมูลรายงานสาธารณะ
- Various helper functions สำหรับการคำนวณและการจัดการข้อมูล

### 4.3 Row Level Security (RLS)

#### การตั้งค่า RLS policies สำหรับความปลอดภัย

- Public Read Access: อนุญาตให้ anon role อ่านข้อมูลที่เผยแพร่แล้ว
- Admin Write Access: เผา service role เท่านั้นที่เขียนข้อมูลได้
- Status-based Filtering: กรองข้อมูลตาม status (published, draft, etc.)

### 4.4 Authentication และ Authorization

#### Client-side Authentication

- ใช้ anon key สำหรับการเข้าถึงข้อมูลสาธารณะ
- JWT tokens สำหรับ authenticated users
- Session management ผ่าน Supabase Auth

## Server-side Operations

- ใช้ service role key สำหรับ backend operations
- CLI scripts และ cron jobs
- Admin operations และ data processing

## 4.5 Storage Configuration

การจัดการไฟล์และภาพ:

- ai-generated-images: Bucket สำหรับภาพที่สร้างด้วย AI
- Public Access: ภาพสามารถเข้าถึงได้โดยไม่ต้อง authentication
- CDN Integration: ใช้ Supabase CDN สำหรับการแจกจ่ายไฟล์

## 4.6 แนวทางปฏิบัติที่ดี

- แยก environment variables ระหว่าง development และ production
- ใช้ connection pooling สำหรับ high-traffic applications
- Monitor database performance และ query optimization
- Backup ข้อมูลสำรองเป็นประจำ
- ตรวจสอบ RLS policies เป็นระยะ
- ใช้ prepared statements เพื่อป้องกัน SQL injection

## 5. គ្រឿងការប្រព័ន្ធបច្ចុប្បន្ន

### 5.1 ការតิดតើងនិងការងារ

#### ការតิดតើងបែងចែង

```
# Clone repository
git clone https://github.com/your-org/trendsiam.git
cd trendsiam

# Install dependencies
cd frontend
npm install

# Setup environment
cp .env.example .env.local
# Edit .env.local with your Supabase credentials
```

#### ការរបន់បញ្ជី

```
# Development server
npm run dev

# Production build
npm run build
npm run start

# Test weekly PDF generation
curl -o weekly.pdf "http://localhost:3000/api/weekly/pdf"
```

### 5.2 ការប្រើប្រាស់ផលិតផលខ្លួន

#### ការបង្កើតផលិតផលថ្មី

```
# សម្រាប់បង្កើតផលិតផលថ្មី
python summarize_all_v2.py --20

# សម្រាប់បង្កើតផលិតផលថ្មី
python summarize_all_v2.py
```

```
# # Pipeline
python scripts/check_pipeline_health.py
```

## การสร้างรายงานรายสัปดาห์

```
# # Snapshot C+H
cd frontend
npm run snapshot:build:publish

# # Snapshot
npm run snapshot:test:selection

# # Diagnostics
curl http://localhost:3000/api/weekly/diagnostics
```

## 5.3 การตรวจสอบสุขภาพระบบ

### Health Check Endpoints

- /api/health - สถานะระบบโดยรวม
- /api/health/db - การเชื่อมต่อฐานข้อมูล
- /api/weekly/diagnostics - สถานะรายงานรายสัปดาห์
- /api/env-check - ตรวจสอบ environment variables

### การตรวจสอบข้อมูล

```
# #'*- l-!9%%H2*8
python test_pipeline_diagnostics.py

# #!- images
python scripts/verify_images.py

# #core details
python scripts/check_score_details.py
```

## 5.4 การแก้ไขปัญหาเบื้องต้น

ปัญหา: Weekly report แสดง "No snapshots available"

- ទទួលសេវា /api/weekly/diagnostics
- ចាប់ npm run snapshot:build:publish
- ទទួលសេវា environment variables

បញ្ជាក់: AI image generation ត្រូវការការពារ

- ទទួលសេវា OpenAI API key
- ទទួលសេវា quota និង rate limits
- ចាប់ python scripts/verify\_images.py

បញ្ជាក់: Database connection error

- ទទួលសេវា Supabase credentials
- ទទួលសេវា network connectivity
- ទទួលសេវា RLS policies

## 6. แนวทางการทดสอบ

### 6.1 การรวมการทดสอบ

ระบบ TrendSiam มีการทดสอบหลายระดับเพื่อให้มั่นใจในคุณภาพและความเสถียร ตั้งแต่ unit tests, integration tests, จนถึง end-to-end tests

### 6.2 การทดสอบ Frontend

#### การทดสอบพื้นฐาน

```
# Type checking
npm run type-check

# Linting
npm run lint

# Build test
npm run build

# Security scan
npm run scan:build:secrets
```

#### การทดสอบ Weekly Report System

```
# *- 2# 1 |-!9%
npm run snapshot:test:count

# *- 2#@%7- |-!9%
npm run snapshot:test:selection

# $10
npm run snapshot:test
```

#### การทดสอบ PDF Generation

```
# PDF overlap fix
npm run test:pdf-overlap

# *#Weekly PDF
curl -o test-weekly.pdf "http://localhost:3000/api/weekly/pdf"
```

```
# #P#F 5H*I2
# @ 4 D %LA%0 #' *- 'H2D!H!5 I- '2! 1 I-
```

## 6.3 การทดสอบ Backend

### การทดสอบ Data Pipeline

```
# *- 2# 6 I-!9%
python test_pipeline_diagnostics.py

# AI processing
python test_ai_image_generation.py

# Categorization
python test_categories_pipeline.py

# Keyword extraction
python -m pytest tests/test_keyword_extraction.py
```

### การทดสอบ Security

```
# Crypto security
python -m pytest tests/test_crypto_security.py

# Image URL protection
python -m pytest tests/test_image_url_protection.py

# Security audit
python security_audit.py
```

## 6.4 การทดสอบแบบ Integration

### End-to-End Testing

```
# E2E fixes
python validate_e2e_fixes.py

# Acceptance
python acceptance_tests.py
```

```
# compatibility
python test_compat_scores.py
```

## การทดสอบ API Endpoints

- GET /api/health - ควรได้ status 200
- GET /api/weekly/diagnostics - ตรวจสอบข้อมูล snapshot
- GET /api/weekly/pdf - ควรได้ไฟล์ PDF
- POST /api/admin/revalidate - ต้องมี secret header

## 6.5 การทดสอบประสิทธิภาพ

แนวทางการทดสอบประสิทธิภาพ:

- Load testing สำหรับ API endpoints
- Database query performance monitoring
- PDF generation time measurement
- Memory usage profiling
- Rate limiting effectiveness
- CDN cache hit rates

## 6.6 Checklist การทดสอบก่อน Deploy

Type checking ผ่าน (npm run type-check)  
Build สำเร็จ (npm run build)  
Security scan ผ่าน (npm run scan:build:secrets)  
Weekly report ทำงานได้ (npm run snapshot:test:selection)  
PDF generation ทำงานได้  
Backend tests ผ่าน  
API endpoints ตอบสนองปกติ  
Database connection ใช้งานได้  
Environment variables ครบถ้วน

# ภาคผนวก ก: บัญชีรายชื่อไฟล์

บัญชีรายชื่อไฟล์ทั้งหมดในระบบ TrendSiam โดยเฉพาะไฟล์ Python ที่เป็นส่วนสำคัญ ของการประมวลผลข้อมูลและ AI processing

## ก.1 ไฟล์ Python หลัก

### Root Directory

#### acceptance\_tests.py

ชุดทดสอบการยอมรับสำหรับตรวจสอบฟังก์ชันหลัก

ฟังก์ชันหลัก: test\_a\_force\_refresh, test\_b\_verify\_api\_parity, test\_c\_manual\_ui\_check, test\_d\_dry\_run\_safety, run\_all\_acceptance\_tests...

การใช้งาน: Test suite, Database operations

ขนาดไฟล์: 10.4 KB | เส้นทาง: acceptance\_tests.py

#### ai\_image\_generator.py

การประมวลผลภาพและ AI image generation สำหรับ ai\_image\_generator

ฟังก์ชันหลัก: get\_precise\_score, sanitize\_prompt\_text, generate\_safe\_fallback\_prompt, generate\_ai\_images\_for\_top3\_news

การใช้งาน: CLI script

ขนาดไฟล์: 45.4 KB | เส้นทาง: ai\_image\_generator.py

#### ai\_image\_generator\_v2.py

ระบบสร้างภาพประกอบด้วย AI สำหรับข่าวและเนื้อห

ฟังก์ชันหลัก: generate\_story\_id, parse\_publish\_time, get\_precise\_score, validate\_image\_file, sanitize\_prompt\_text...

การใช้งาน: CLI script

ขนาดไฟล์: 25.6 KB | เส้นทาง: ai\_image\_generator\_v2.py

#### ai\_image\_supabase\_generator.py

การประมวลผลภาพและ AI image generation สำหรับ ai\_image\_supabase\_generator

การใช้งาน: Database operations

ขนาดไฟล์: 8.1 KB | เส้นทาง: ai\_image\_supabase\_generator.py

#### app.py

TrendSiam - Thai Daily News Summary Web App

A simple Streamlit application that displays summarized Thai trending news from YouTube videos with AI-generated Thai summaries. (ໃນຊາ Python สำหรับ app)

ฟังก์ชันหลัก: apply\_dynamic\_theme, apply\_radio\_theme\_css, is\_dev\_mode, validate\_youtube\_video\_id, create\_secure\_youtube\_url...

การใช้งาน: CLI script

ขนาดไฟล์: 218.5 KB | เส้นทาง: app.py

## archive\app\_original.py

TrendSiam - Thai Daily News Summary Web App

A simple Streamlit application that displays summarized Thai trending news from YouTube videos with AI-generated Thai summaries. (ในดู Python สำหรับ archive\app\_original)

ฟังก์ชันหลัก: apply\_dynamic\_theme, apply\_radio\_theme\_css, is\_dev\_mode, validate\_youtube\_video\_id, create\_secure\_youtube\_url...

การใช้งาน: CLI script

ขนาดไฟล์: 212 KB | เส้นทาง: archive\app\_original.py

## clean\_trending\_data.py

เครื่องมือทำความสะอาดและปรับปรุงข้อมูลเบนโน้

ฟังก์ชันหลัก: is\_fake\_video\_id, is\_fake\_channel, is\_fake\_title, has\_suspicious\_view\_count, is\_fake\_entry...

การใช้งาน: CLI script

ขนาดไฟล์: 8 KB | เส้นทาง: clean\_trending\_data.py

## cleanup\_backup\_20250728\_200913\fix\_dev\_password.py

Script to fix the developer password security vulnerability in app.py (ในดู Python สำหรับ cleanup\_backup\_20250728\_200913\fix\_dev\_password)

ฟังก์ชันหลัก: fix\_dev\_password

การใช้งาน: CLI script

ขนาดไฟล์: 1.9 KB | เส้นทาง: cleanup\_backup\_20250728\_200913\fix\_dev\_password.py

## config\constants.py

การตั้งค่าและกำหนดค่าสำหรับ config\constants

ฟังก์ชันหลัก: get\_developer\_password, get\_feature\_flag

ขนาดไฟล์: 6 KB | เส้นทาง: config\constants.py

## config\_openai.py

การตั้งค่าและกำหนดค่าสำหรับ config\_openai

การใช้งาน: CLI script

ขนาดไฟล์: 2.5 KB | เส้นทาง: config\_openai.py

## core\api\_security.py

API endpoints และการจัดการ requests สำหรับ core\api\_security

ฟังก์ชันหลัก: rate\_limit, secure\_api\_call

ขนาดไฟล์: 14.7 KB | เส้นทาง: core\api\_security.py

## core\config.py

การตั้งค่าและกำหนดค่าสำหรับ core\config

ฟังก์ชันหลัก: get\_config, reload\_config, get\_openai\_config

## การใช้งาน: Database operations

ขนาดไฟล์: 7.3 KB | เส้นทาง: core\config.py

### core\constants.py

Constants and configuration values for TrendSiam application.

This file centralizes all magic numbers and hardcoded values to improve maintainability. (ໃນດາ Python ສໍາຮັບ core\constants)

ขนาดไฟล์: 3.3 KB | เส้นทาง: core\constants.py

### core\crypto\_security.py

Cryptography Security Wrapper for TrendSiam

This module provides secure cryptographic operations and prevents usage of vulnerable APIs. All cryptographic operations should go through this module.

Security measures:

- Blocks RSA PKCS1v15 padding for decryption (forces OAEP)
- Validates PKCS#12 certificate/key matching
- Provides secure defaults for all operations (ໃນດາ Python ສໍາຮັບ core\crypto\_security)

ຟັງກີບຫຼັກ: detect\_insecure\_crypto\_usage

## การใช้งาน: CLI script

ขนาดไฟล์: 10.1 KB | เส้นทาง: core\crypto\_security.py

### core\env\_config.py

การຕັ້ງຄ່າແລະກຳນົດຄ່າສໍາຮັບ core\env\_config

ຟັງກີບຫຼັກ: load\_config, get\_config, create\_template

## การใช้งาน: CLI script, Database operations

ขนาดไฟล์: 12.7 KB | เส้นทาง: core\env\_config.py

### core\\_\_init\_\_.py

TrendSiam Core Security Module

This package provides comprehensive security, privacy, and compliance management for the TrendSiam application including:

- Secure configuration management with environment variable protection
- Input validation and sanitization with XSS/injection prevention
- Comprehensive logging with credential sanitization
- Secure subprocess execution with command injection prevention
- API security with rate limiting and request validation
- Legal compliance for YouTube, OpenAI, and DALL-E APIs
- Privacy protection and GDPR compliance
- Ethical AI usage validation

Security Features:

Environment variable protection

API key validation and secure storage

Input sanitization and validation

Command injection prevention  
Rate limiting and abuse prevention  
Legal compliance checking  
Privacy-compliant data handling  
Comprehensive security logging  
Error handling with information disclosure prevention (ໃນຊາ Python ສໍາຮັບ core\\_\_init\_\_)

ຟັງກ່ອນຫຼັກ: initialize\_security, run\_security\_audit

ຂາດໄຟສ: 8.5 KB | ເສັນຖາງ: core\\_\_init\_\_.py

---

## core\legal\_compliance.py

Legal Compliance Module for TrendSiam

This module ensures compliance with Terms of Service, privacy regulations, and responsible data usage for all integrated platforms and services. (ໃນຊາ Python ສໍາຮັບ core\legal\_compliance)

ຟັງກ່ອນຫຼັກ: get\_compliance\_manager, ensure\_youtube\_compliance, ensure\_openai\_compliance, ensure\_dalle\_compliance

ຂາດໄຟສ: 17.1 KB | ເສັນຖາງ: core\legal\_compliance.py

---

## core\logging\_config.py

ກາຣຕັ້ງຄ່າແລະກໍາເນດຄ່າສໍາຮັບ core\logging\_config

ຟັງກ່ອນຫຼັກ: performance\_monitor, error\_handler, create\_module\_logger, setup\_logging, get\_app\_logger

ຂາດໄຟສ: 11.5 KB | ເສັນຖາງ: core\logging\_config.py

---

## core\secure\_subprocess.py

Secure Subprocess Wrapper for TrendSiam

This module provides secure subprocess execution with input validation, command injection prevention, and comprehensive logging. (ໃນຊາ Python ສໍາຮັບ core\secure\_subprocess)

ຟັງກ່ອນຫຼັກ: secure\_subprocess\_run, secure\_ytddlp\_call

ຂາດໄຟສ: 11.1 KB | ເສັນຖາງ: core\secure\_subprocess.py

---

## core\storage\_config.py

ກາຣຕັ້ງຄ່າແລະກໍາເນດຄ່າສໍາຮັບ core\storage\_config

ຟັງກ່ອນຫຼັກ: get\_storage\_config, get\_bucket\_name, get\_image\_public\_url, validate\_storage

ກາຣໃໝ່ງານ: CLI script, Database operations

ຂາດໄຟສ: 4.7 KB | ເສັນຖາງ: core\storage\_config.py

---

## `core\validators.py`

Input Validation and Security Module for TrendSiam

This module provides comprehensive input validation, sanitization, and security checks for all external data sources. (ໃນຊາ Python ສໍາຮັບ core\validators)

ຟັງກ່ຽວຂ້ອງ: validate\_news\_data, safe\_get\_env\_var

ຂາດໄຟຣ: 13.5 KB | ເສັ້ນທາງ: core\validators.py

---

## `debug_second_story_image.py`

ການປະໜວລົດກາພແລະ AI image generation ສໍາຮັບ debug\_second\_story\_image

ຟັງກ່ຽວຂ້ອງ: check\_supabase\_data, check\_frontend\_api, main

ການໃໝ່ງານ: CLI script, Database operations

ຂາດໄຟຣ: 6.5 KB | ເສັ້ນທາງ: debug\_second\_story\_image.py

---

## `diagnose_supabase.py`

Diagnostic script to test Supabase connection and identify issues (ໃນຊາ Python ສໍາຮັບ diagnose\_supabase)

ການໃໝ່ງານ: Database operations

ຂາດໄຟຣ: 7.1 KB | ເສັ້ນທາງ: diagnose\_supabase.py

---

## `fix_ai_images_for_existing.py`

ການປະໜວລົດກາພແລະ AI image generation ສໍາຮັບ fix\_ai\_images\_for\_existing

ຟັງກ່ຽວຂ້ອງ: generate\_images\_for\_top3

ການໃໝ່ງານ: CLI script, Database operations

ຂາດໄຟຣ: 3.9 KB | ເສັ້ນທາງ: fix\_ai\_images\_for\_existing.py

---

## `fix_env_DANGEROUS_DELETE_ME.py`

ໃນຊາ Python: fix\_env\_DANGEROUS\_DELETE\_ME

ຂາດໄຟຣ: 139 B | ເສັ້ນທາງ: fix\_env\_DANGEROUS\_DELETE\_ME.py

---

## `force regenerate_images.py`

ການປະໜວລົດກາພແລະ AI image generation ສໍາຮັບ force regenerate\_images

ຟັງກ່ຽວຂ້ອງ: force regenerate\_images, main

ການໃໝ່ງານ: CLI script

ຂາດໄຟຣ: 4.5 KB | ເສັ້ນທາງ: force regenerate\_images.py

---

## `generate_html_pdf.py`

TrendSiam HTML Report Generator with WeasyPrint

Purpose: Generate professional PDF reports using HTML/CSS template and WeasyPrint (ໃນຊາ Python ສໍາຮັບ generate\_html\_pdf)

ຟັງກ່ຽວຂ້ອງ: safe\_int, safe\_float, parse\_view\_count, get\_precise\_score, get\_score\_class...

ການໃໝ່ງານ: CLI script

ຂາດໄຟຣ: 22.5 KB | ເສັ້ນທາງ: generate\_html\_pdf.py

---

## `legacy_adapter.py`

Legacy Adapter for TrendSiam V2 Compatibility

This module provides adapters to reuse legacy popularity scoring and category classification logic in the V2 ingestion system, ensuring 100% compatibility. (ໃນດາ Python ສໍາເກີນ `legacy_adapter`)

ขนาดໄຟສ: 10.1 KB | ເສັ້ນທາງ: `legacy_adapter.py`

---

## `mark_json_legacy.py`

SECTION H - JSON Legacy Marking Script

This script marks JSON data files as legacy by adding `.legacy` suffix while preserving functionality for emergency use. (ໃນດາ Python ສໍາເກີນ `mark_json_legacy`)

ຟັງກີ່ຫັນຫຼັກ: `mark_json_files_as_legacy`

ການໃຊ້ຈານ: **CLI script**

ขนาดໄຟສ: 2.2 KB | ເສັ້ນທາງ: `mark_json_legacy.py`

---

## `popularity_scorer.py`

ຮະບບຄຳນວນຄະແນນຄວາມນິຍມແລກຈັດອັນດັບ

ຟັງກີ່ຫັນຫຼັກ: `score_video_popularity`, `add_popularity_scores`, `main`

ການໃຊ້ຈານ: **CLI script**

ขนาดໄຟສ: 37.3 KB | ເສັ້ນທາງ: `popularity_scorer.py`

---

## `scripts\check_aux_fields_pipeline.py`

Check auxiliary fields in the pipeline

Verify which fields are being computed and written (ໃນດາ Python ສໍາເກີນ `scripts\check_aux_fields_pipeline`)

ຟັງກີ່ຫັນຫຼັກ: `check_pipeline_fields`

ການໃຊ້ຈານ: **CLI script, Database operations**

ขนาดໄຟສ: 2.2 KB | ເສັ້ນທາງ: `scripts\check_aux_fields_pipeline.py`

---

## `scripts\check_pipeline_health.py`

SECTION G - Pipeline Health Check Script

Reads `system_meta('news_last_updated')`, prints age in minutes, and top 1 story.

Exit non-zero if age > 180 minutes in production. (ໃນດາ Python ສໍາເກີນ `scripts\check_pipeline_health`)

ຟັງກີ່ຫັນຫຼັກ: `get_supabase_client`, `get_last_updated`, `get_top_story`, `calculate_age_minutes`, `main`

ການໃຊ້ຈານ: **Database operations**

ขนาดໄຟສ: 3.8 KB | ເສັ້ນທາງ: `scripts\check_pipeline_health.py`

---

## `scripts\check_score_details.py`

Check `score_details` column and data in the database (ໃນດາ Python ສໍາເກີນ `scripts\check_score_details`)

ການໃຊ້ຈານ: **Database operations**

ขนาดໄຟສ: 2.5 KB | ເສັ້ນທາງ: `scripts\check_score_details.py`

## `scripts\find_unused_files.py`

Find Unused Files in TrendSiam Repository

This script identifies potentially unused files that can be safely removed.

It generates a dry-run report before any deletion. (ໃບດາ Python ສໍາເລັບ scripts\find\_unused\_files)

ຟັງກີ່ຂັ້ນຫຼັກ: main

ການໃຊ້ຈານ: CLI script, Database operations

ຂາດໄຟຟ: 16.3 KB | ເສັ້ນທາງ: scripts\find\_unused\_files.py

---

## `scripts\fix_single_video_enrichment.py`

Fix missing auxiliary fields for a single video

Specifically handles BNK48 and other edge cases (ໃບດາ Python ສໍາເລັບ scripts\fix\_single\_video\_enrichment)

ຟັງກີ່ຂັ້ນຫຼັກ: extract\_keywords\_multilingual, calculate\_growth\_rate\_safe, extract\_platform\_mentions, generate\_score\_details, generate\_ai\_opinion...

ການໃຊ້ຈານ: CLI script, Database operations

ຂາດໄຟຟ: 10.6 KB | ເສັ້ນທາງ: scripts\fix\_single\_video\_enrichment.py

---

## `scripts\security_audit_comprehensive.py`

Comprehensive Security Audit for TrendSiam Repository

This script performs a thorough security audit including:

1. Secret detection
2. Dependency vulnerabilities
3. Code security patterns
4. Configuration issues (ໃບດາ Python ສໍາເລັບ scripts\security\_audit\_comprehensive)

ຟັງກີ່ຂັ້ນຫຼັກ: main

ການໃຊ້ຈານ: CLI script, Database operations

ຂາດໄຟຟ: 18.4 KB | ເສັ້ນທາງ: scripts\security\_audit\_comprehensive.py

---

## `scripts\test_bnk48_fix.py`

ໄວ້ກົດສອບສໍາເລັບ scripts\bnk48\_fix

ຟັງກີ່ຂັ້ນຫຼັກ: extract\_keywords\_old, extract\_keywords\_new, detect\_platforms\_old, detect\_platforms\_new, generate\_opinion\_old...

ການໃຊ້ຈານ: CLI script, Test suite

ຂາດໄຟຟ: 6.4 KB | ເສັ້ນທາງ: scripts\test\_bnk48\_fix.py

---

## `scripts\verify_image_setup.py`

ກາປະນວລພລກພາບແລະ AI image generation ສໍາເລັບ scripts\verify\_image\_setup

ຟັງກີ່ຂັ້ນຫຼັກ: verify\_storage\_setup, verify\_database\_state, main

ການໃຊ້ຈານ: CLI script, Database operations

ຂາດໄຟຟ: 6.6 KB | ເສັ້ນທາງ: scripts\verify\_image\_setup.py

---

## `scripts\verify_images.py`

ກາປະນວລພລກພາບແລະ AI image generation ສໍາເລັບ scripts\verify\_images

ฟังก์ชันหลัก: verify\_supabase\_connection, get\_recent\_items\_without\_images, verify\_image\_url, check\_existing\_images, regenerate\_missing\_images...

การใช้งาน: CLI script, Database operations

ขนาดไฟล์: 8.9 KB | เส้นทาง: scripts\verify\_images.py

## scripts\verify\_image\_urls.py

การประมวลผลภาพและ AI image generation สำหรับ scripts\verify\_image\_urls

ฟังก์ชันหลัก: verify\_image\_urls

การใช้งาน: CLI script, Database operations

ขนาดไฟล์: 4.2 KB | เส้นทาง: scripts\verify\_image\_urls.py

## scripts\verify\_score\_details\_pipeline.py

Verify score details are being generated and written correctly (ในดู Python สำหรับ scripts\verify\_score\_details\_pipeline)

การใช้งาน: Database operations

ขนาดไฟล์: 3.1 KB | เส้นทาง: scripts\verify\_score\_details\_pipeline.py

## security\_audit.py

เครื่องมือตรวจสอบความปลอดภัยของระบบ

ฟังก์ชันหลัก: audit\_frontend\_security, audit\_environment\_usage, print\_audit\_results

การใช้งาน: CLI script, Database operations

ขนาดไฟล์: 7.7 KB | เส้นทาง: security\_audit.py

## setup\_backend\_env.py

Helper script to set up the backend .env file with proper Supabase credentials (ในดู Python สำหรับ setup\_backend\_env)

การใช้งาน: Database operations

ขนาดไฟล์: 3.5 KB | เส้นทาง: setup\_backend\_env.py

## setup\_environment.py

scrriptตั้งค่าสภาพแวดล้อมและ dependencies

ฟังก์ชันหลัก: create\_env\_file, test\_with\_fallback, check\_existing\_env, main

การใช้งาน: CLI script, Test suite, Database operations

ขนาดไฟล์: 5.3 KB | เส้นทาง: setup\_environment.py

## summarize\_all.py

TrendSiam News Ingestion Pipeline with Idempotency and Image Persistence

This script implements a robust news ingestion pipeline with the following guarantees:

A) Idempotency without losing history:

- Two-layer model: stories (canonical) + snapshots (per-run/day)
- Never destroys historical data
- Re-runs create/update snapshots, preserving history

- Atomic writes and non-destructive DB upserts only

#### B) Image persistence and regeneration policy (Top-3 focus):

- Never deletes/overwrites valid existing images
- Generates new images only when missing/invalid
- Stable story\_id-based image mapping
- Top-3 image validation with retry logic

#### C) Ordering and alignment:

- Deterministic Top-3 using popularity\_score desc, publish\_time desc, story\_id
- story\_id and rank included in all outputs for UI alignment
- Images never reordered independently of stories

#### D) UX/UI and caching safety:

- Frontend JSON includes story\_id, rank, image\_status, data\_version
- Placeholder handling for pending images
- Cache busting with data\_version timestamp

#### E) Reliability and logging:

- Structured logging for all operations
- Exit codes: 0 (success), 5 (partial), others (errors)
- Configurable retry logic with exponential backoff
- Dry-run mode for testing (ໃນງາ Python សໍາຮັບ summarize\_all)

ຟັງກີ່ບັນຫຼັກ: get\_precise\_score, generate\_story\_id, parse\_publish\_time, validate\_image\_file, determine\_top3\_ordering...

ການໃຊ້ຈ່ານ: [CLI script](#), [Database operations](#)

ຂາດໄຟຣີ: 85.7 KB | ເສັນຕາງ: summarize\_all.py

---

## summarize\_all\_v2.py

ສະເອົາໂປ່ຕໍ່ຫັກສໍາຮັບດຶງຂ້ອມມູລແລະສ໌ຮ້າງສຽງຂ່າວຈາກ YouTube API ວິຊ້ອມການປະໜວລົດ AI

ຟັງກີ່ບັນຫຼັກ: generate\_story\_id, parse\_publish\_time, get\_precise\_score, determine\_top3\_ordering, validate\_image\_file...

ການໃຊ້ຈ່ານ: [CLI script](#), [Database operations](#)

ຂາດໄຟຣີ: 87.5 KB | ເສັນຕາງ: summarize\_all\_v2.py

---

## summarize\_all\_v3\_supabase\_only.py

ເວັບສັນທິໃຊ້ Supabase ເປັນຫັກສໍາຮັບການຈັດເກີບແລະປະໜວລົດຂ້ອມ

ຟັງກີ່ບັນຫຼັກ: parse\_args, main

ການໃຊ້ຈ່ານ: [Database operations](#)

ຂາດໄຟຣີ: 22 KB | ເສັນຕາງ: summarize\_all\_v3\_supabase\_only.py

---

## summarizer.py

YouTube Video Summarizer

This script provides functionality to generate Thai-language summaries of YouTube videos using OpenAI's GPT-3.5-turbo model. It processes video metadata and creates concise, news-style summaries suitable for reporting.

Features:

- Thai-language summary generation
- News-style reporting format
- Error handling and graceful degradation
- Configurable OpenAI API settings
- Legal compliance (uses only public metadata)
- Robust retry logic with exponential backoff
- Rate limit handling for API stability (ໃນຊາ Python ສໍາເຮັບ summarizer)

ຟັງກີ່ຫັນຫຼັກ: get\_summarizer, summarize\_video\_info, summarize\_thai\_video, summarize\_english\_video, main

ການໃຊ້ຈານ: CLI script

ຂາດໄຟຟ: 26.6 KB | ເສັ້ນທາງ: summarizer.py

---

### test\_ai\_image\_generation.py

ໄວ້ລົດສອບສໍາເຮັບ ai\_image\_generation

ຟັງກີ່ຫັນຫຼັກ: test\_supabase\_connection, test\_ai\_image\_generator, test\_supabase\_storage, test\_supabase\_ai\_generator, test\_ai\_images\_table...

ການໃຊ້ຈານ: CLI script, Test suite, Database operations

ຂາດໄຟຟ: 6.5 KB | ເສັ້ນທາງ: test\_ai\_image\_generation.py

---

### test\_ai\_only\_images.py

ໄວ້ລົດສອບສໍາເຮັບ ai\_only\_images

ຟັງກີ່ຫັນຫຼັກ: test\_backend\_pipeline, test\_frontend\_api, test\_frontend\_code, test\_database\_views, main

ການໃຊ້ຈານ: CLI script, Test suite, Database operations

ຂາດໄຟຟ: 9.4 KB | ເສັ້ນທາງ: test\_ai\_only\_images.py

---

### test\_categories\_pipeline.py

ໄວ້ລົດສອບສໍາເຮັບ categories\_pipeline

ຟັງກີ່ຫັນຫຼັກ: load\_sample\_data, classify\_with\_legacy, test\_golden\_set, analyze\_unknown\_rate, print\_category\_distribution...

ການໃຊ້ຈານ: Test suite

ຂາດໄຟຟ: 8.8 KB | ເສັ້ນທາງ: test\_categories\_pipeline.py

---

### test\_compat\_scores.py

ໄວ້ລົດສອບສໍາເຮັບ compat\_scores

ຟັງກີ່ຫັນຫຼັກ: load\_sample\_data, score\_with\_legacy, score\_with\_v2, extract\_score, compare\_scores...

ການໃຊ້ຈານ: Test suite

ຂາດໄຟຟ: 7.2 KB | ເສັ້ນທາງ: test\_compat\_scores.py

---

### test\_idempotency\_cases.py

ໄວ້ລົດສອບສໍາເຮັບ idempotency\_cases

ຟັງກີ່ຫັນຫຼັກ: create\_mock\_video\_data, create\_mock\_image, test\_case\_1\_incomplete\_third\_image, test\_case\_2\_images\_disappear, test\_frontend\_json\_format...

ການໃຊ້ຈານ: CLI script, Test suite

ຂາດໄຟຟ: 20.4 KB | ເສັ້ນທາງ: test\_idempotency\_cases.py

## [tests\image\test\\_image\\_fallback\\_fix.py](#)

ไฟล์ทดสอบสำหรับ image\_fallback\_fix

ฟังก์ชันหลัก: test\_frontend\_data, main

การใช้งาน: CLI script, Test suite

ขนาดไฟล์: 4.3 KB | เส้นทาง: tests\image\test\_image\_fallback\_fix.py

---

## [tests\image\test\\_image\\_prompt\\_presence.py](#)

ไฟล์ทดสอบสำหรับ image\_prompt\_presence

ฟังก์ชันหลัก: load\_sample\_data, validate\_image\_prompts, print\_results, main

การใช้งาน: Test suite

ขนาดไฟล์: 5.7 KB | เส้นทาง: tests\image\test\_image\_prompt\_presence.py

---

## [tests\json\test\\_json\\_contract.py](#)

ไฟล์ทดสอบสำหรับ json\_contract

ฟังก์ชันหลัก: load\_json\_data, validate\_json\_contract, print\_results, main

การใช้งาน: Test suite

ขนาดไฟล์: 5.9 KB | เส้นทาง: tests\json\test\_json\_contract.py

---

## [tests\no\\_mock\test\\_no\\_mock\\_output.py](#)

ไฟล์ทดสอบสำหรับ no\_mock\_output

ฟังก์ชันหลัก: scan\_json\_file, scan\_item\_for\_mock\_content, validate\_summary\_fields, check\_required\_fields, print\_violations...

การใช้งาน: Test suite

ขนาดไฟล์: 10.5 KB | เส้นทาง: tests\no\_mock\test\_no\_mock\_output.py

---

## [tests\pipeline\test\\_pipeline\\_diagnostics.py](#)

การทดสอบและตรวจสอบสุขภาพของ data pipeline

ฟังก์ชันหลัก: main

การใช้งาน: Test suite, Database operations

ขนาดไฟล์: 14.9 KB | เส้นทาง: tests\pipeline\test\_pipeline\_diagnostics.py

---

## [tests\pipeline\test\\_enrichment\\_persist.py](#)

ไฟล์ทดสอบสำหรับ tests\pipeline\enrichment.persist

การใช้งาน: Test suite, Database operations

ขนาดไฟล์: 6.1 KB | เส้นทาง: tests\pipeline\test\_enrichment\_persist.py

---

## [tests\test\\_bnk48\\_enrichment.py](#)

ไฟล์ทดสอบสำหรับ tests\bnk48\_enrichment

การใช้งาน: Test suite

ขนาดไฟล์: 5.1 KB | เส้นทาง: tests\test\_bnk48\_enrichment.py

---

## [tests\test\\_build\\_score\\_details.py](#)

ไฟล์ทดสอบสำหรับ tests\build\_score\_details

ฟังก์ชันหลัก: test\_build\_score\_details

การใช้งาน: Test suite

ขนาดไฟล์: 3.7 KB | เส้นทาง: tests\test\_build\_score\_details.py

---

### tests\test\_crypto\_security.py

ไฟล์ทดสอบสำหรับ tests\crypto\_security

การใช้งาน: CLI script, Test suite

ขนาดไฟล์: 8.5 KB | เส้นทาง: tests\test\_crypto\_security.py

---

### tests\test\_images\_pipeline\_backend.py

ไฟล์ทดสอบสำหรับ tests\images\_pipeline\_backend

การใช้งาน: Test suite, Database operations

ขนาดไฟล์: 7.9 KB | เส้นทาง: tests\test\_images\_pipeline\_backend.py

---

### tests\test\_image\_url\_protection.py

ไฟล์ทดสอบสำหรับ tests\image\_url\_protection

การใช้งาน: Test suite, Database operations

ขนาดไฟล์: 9.8 KB | เส้นทาง: tests\test\_image\_url\_protection.py

---

### tests\test\_keyword\_extraction.py

ไฟล์ทดสอบสำหรับ tests\keyword\_extraction

ฟังก์ชันหลัก: test\_basic\_extraction, test\_hashtag\_extraction, test\_multilingual\_keywords, test\_featuring\_artist\_extraction, test\_stopwords\_filtering...

การใช้งาน: Test suite

ขนาดไฟล์: 8.7 KB | เส้นทาง: tests\test\_keyword\_extraction.py

---

### tests\test\_multilingual\_enrichment.py

ไฟล์ทดสอบสำหรับ tests\multilingual\_enrichment

ฟังก์ชันหลัก: test\_keyword\_extraction, test\_platform\_detection, test\_bnk48\_enrichment

การใช้งาน: CLI script, Test suite

ขนาดไฟล์: 5.4 KB | เส้นทาง: tests\test\_multilingual\_enrichment.py

---

### tests\test\_score\_details\_deterministic.py

ไฟล์ทดสอบสำหรับ tests\score\_details\_deterministic

ฟังก์ชันหลัก: test\_build\_score\_details\_deterministic

การใช้งาน: Test suite

ขนาดไฟล์: 5.5 KB | เส้นทาง: tests\test\_score\_details\_deterministic.py

---

### test\_summaries\_presence.py

ไฟล์ทดสอบสำหรับ summaries\_presence

ฟังก์ชันหลัก: load\_sample\_data, validate\_summaries, print\_results, main

## การใช้งาน: Test suite

ขนาดไฟล์: 7 KB | เส้นทาง: test\_summaries\_presence.py

### update\_video\_data.py

TrendSiam Video Data Updater

=====

Safely updates video view counts and categories from YouTube Data API while maintaining full compatibility with existing TrendSiam system and security policies.

Features:

- Fetches latest view counts, likes, comments from YouTube API
- Re-classifies categories using existing TrendSiam logic
- Defensive programming to prevent data loss
- Backup and rollback functionality
- Rate limiting and API quota management
- Full validation and error handling

Requirements:

- YOUTUBE\_API\_KEY in .env file
- Existing thailand\_trending\_summary.json file
- All TrendSiam dependencies installed

Usage:

python update\_video\_data.py [--dry-run] [--backup-only] [--restore]

Security:

- Validates all API responses
- Creates automatic backups before changes
- Maintains data integrity with extensive validation
- Respects YouTube API terms of service
- Implements rate limiting and error recovery (ໃນດາ Python ສໍາຮັບ update\_video\_data)

ຟັງກ້ອບເຂົ້າກັບ: main

## การใช้งาน: CLI script

ขนาดไฟล์: 23.3 KB | เส้นทาง: update\_video\_data.py

### utils\\_\_init\_\_.py

Utility functions for the TrendSiam pipeline (ໃນດາ Python ສໍາຮັບ utils\\_\_init\_\_)

ขนาดไฟล์: 287 B | เส้นทาง: utils\\_\_init\_\_.py

### utils\keyword\_extractor.py

Advanced keyword extraction for multilingual content (ໃນດາ Python ສໍາຮັບ utils\keyword\_extractor)

ขนาดไฟล์: 14 KB | เส้นทาง: utils\keyword\_extractor.py

### utils\safe.py

Safe utilities for protecting existing data during pipeline operations (ໃນດາ Python ສໍາຮັບ utils\safe)

ຟັງກ້ອບເຂົ້າກັບ: truthy\_url, set\_if\_truthy, get\_safe\_url

ขนาดไฟล์: 1012 B | เส้นทาง: utils\safe.py

---

### [validate\\_e2e\\_fixes.py](#)

Validate E2E fixes are working correctly (ໂນດາ Python ສໍາຮັບ validate\_e2e\_fixes)

ກາຣໃຊ້ຈານ: Database operations

ຂນາດໄຟສ: 3.5 KB | ເສັ້ນທາງ: validate\_e2e\_fixes.py

---

### [validate\\_fixes.py](#)

[DEPRECATED] Use validate\_e2e\_fixes.py instead

Validate that the pipeline 'DB' 'API' 'Frontend flow is working correctly (ໂນດາ Python ສໍາຮັບ validate\_fixes)

ກາຣໃຊ້ຈານ: Database operations

ຂນາດໄຟສ: 4 KB | ເສັ້ນທາງ: validate\_fixes.py

---

### [youtube\\_api\\_fetcher.py](#)

ຕັວດຶງຂໍ້ອມູລຈາກ YouTube Data API

ຟັງກີບຫັນຫຼັກ: main

ກາຣໃຊ້ຈານ: CLI script

ຂນາດໄຟສ: 11.6 KB | ເສັ້ນທາງ: youtube\_api\_fetcher.py

---

### [youtube\\_fetcher.py](#)

ກາຣດຶງຂໍ້ອມູລແລະ web scraping ສໍາຮັບ youtube\_fetcher

ຟັງກີບຫັນຫຼັກ: main

ກາຣໃຊ້ຈານ: CLI script

ຂນາດໄຟສ: 17.6 KB | ເສັ້ນທາງ: youtube\_fetcher.py

---

## [Frontend/src/app](#)

### [Frontend/src/app/weekly-report](#)

### [Frontend/src/lib/weekly](#)

### [Frontend/src/app/api/weekly/pdf](#)

## ກ.2 ສຄຣິປົກສໍາຄັກ

## สคริปต์หลักสำหรับการดึงข้อมูล

- summarize\_all\_v2.py - สคริปต์หลักสำหรับดึงข้อมูลจาก YouTube
- youtube\_api\_fetcher.py - ตัวดึงข้อมูล API เดพา
- ai\_image\_generator\_v2.py - สร้างภาพประกอบด้วย AI

## สคริปต์ทดสอบและตรวจสอบ:

- test\_pipeline\_diagnostics.py - ตรวจสอบสุขภาพ pipeline
- acceptance\_tests.py - ทดสอบการยอมรับ
- security\_audit.py - ตรวจสอบความปลอดภัย

## สคริปต์ตั้งค่าและบำรุงรักษา

- setup\_environment.py - ตั้งค่าสภาพแวดล้อม
- clean\_trending\_data.py - ทำความสะอาดข้อมูล
- popularity\_scorer.py - คำนวณคะแนนความนิยม

# ภาคผนวก ข: รายการ API และเส้นทาง

## ข.1 API Endpoints

GET /api/health

ตรวจสอบสถานะระบบโดยรวม

การยืนยันตัวตน: ไม่ต้อง ผลลัพธ์: JSON status object

GET /api/health/db

ตรวจสอบการเชื่อมต่อฐานข้อมูล

การยืนยันตัวตน: ไม่ต้อง ผลลัพธ์: Database connection status

GET /api/weekly/diagnostics

ข้อมูลการวินิจฉัยระบบรายงานรายสัปดาห์

การยืนยันตัวตน: ไม่ต้อง ผลลัพธ์: Diagnostic information

GET /api/weekly/pdf

สร้างและดาวน์โหลดรายงาน PDF รายสัปดาห์

การยืนยันตัวตน: ไม่ต้อง ผลลัพธ์: PDF file

GET /api/weekly/data

ข้อมูลรายงานรายสัปดาห์ในรูปแบบ JSON

การยืนยันตัวตน: ไม่ต้อง ผลลัพธ์: Weekly snapshot data

POST /api/admin/revalidate

บังคับให้ระบบโหลดข้อมูลใหม่

การยืนยันตัวตน: x-revalidate-secret header ผลลัพธ์: Revalidation result

GET /api/env-check

ตรวจสอบการตั้งค่า environment variables

การยืนยันตัวตน: ไม่ต้อง ผลลัพธ์: Environment status

## ข.2 หน้าเว็บไซต์

/ Public

หน้าแรก - แสดงข้อมูลข่าวสารในล่าสุด

/weekly-report Public

/dev-dashboard Development

แดชบอร์ดสำหรับนักพัฒนา

/legal Public

ข้อมูลทางกฎหมาย

/privacy Public

นโยบายความเป็นส่วนตัว

/terms Public

ข้อกำหนดการใช้งาน

## ข.3 คำสั่ง CLI สำคัญ

```
# Frontend Commands
npm run dev          # Development server
npm run build         # Production build
npm run snapshot:build:publish # *#12A%&@ "A #H
npm run snapshot:test:selection # *- 2#@%7- I-!9%

# Backend Commands
python summarize_all_v2.py --20  # 620-!92% 2#
python test_pipeline_diagnostics.py # Pipeline
python security_audit.py      # #'*- '2! %- 1"

# Health Checks
curl http://localhost:3000/api/health
curl http://localhost:3000/api/weekly/diagnostics
```

## ข.4 Environment Variables สำคัญ

ตัวแปรสภาพแวดล้อมที่จำเป็นสำหรับการทำงานของระบบ

```
NEXT_PUBLIC_SUPABASE_URL      # URL -Supabase project
NEXT_PUBLIC_SUPABASE_ANON_KEY # Anon key *34#1ent
SUPABASE_SERVICE_ROLE_KEY     # Service role *34#1ver
OPENAI_API_KEY                # API key *34#1OpenAI
YOUTUBE_API_KEY               # API key *34#1Tube
REVALIDATE_SECRET             # Secret *34#1malidation
```