



# Content Organizer Backend

**Project Code:** `08_Utility_Datatagger_Organizer`

**Goal:** All-in-one tool to scan, organize, tag, and package adult content for a future tube site.



## Problem Statement

Content is scattered across an S3 bucket ( `Z:\` via DigitalOcean Spaces) with inconsistent naming. The goal is to restructure every piece of content into **self-contained folders** that include:

Asset	Description
Main Video	The primary scene file (PPV, Full Scene)
Metadata	JSON file with tags, intensity scores, timeline
Trailer	30-60s highlight clip
FYP Clips	15s vertical "stop-the-scroll" shorts
Thumbnail	Auto-generated eye-catching preview image

This structure will power a tube site that pulls directly from these organized folders.



## Folder Structure (Output)

```
Z:\Organized\
├─ 2026-01-24_Morning Glory - Scene 1\
│   ├── PPV_2026-01-24_Morning Glory - Scene 1.mp4    # Main video
│   ├── TRAILER_2026-01-24_Morning Glory - Scene 1.mp4
│   ├── FYP_2026-01-24_Morning Glory - Scene 1_01.mp4
│   ├── FYP_2026-01-24_Morning Glory - Scene 1_02.mp4
│   ├── THUMB_2026-01-24_Morning Glory - Scene 1.jpg
│   └── META_2026-01-24_Morning Glory - Scene 1.json
```

## Naming Convention

`TYPE_YYYY-MM-DD_Scene Name - Scene Number[_##].ext`

- **TYPE:** PPV , TRAILER , FYP , THUMB , META , TWITTER
- **Scene Number:** Sequential (Scene 1, Scene 2, etc.)
- **##:** Optional clip number for FYP (01, 02, etc.)



## Core Features

### 1. File Scanner & Discovery

- Recursively scan `Z:\` for video files ( `.mp4` , `.mov` , `.mkv` )
- Build inventory with file metadata (size, duration, resolution)

- Group files by heuristics (date, filename similarity)

**2. Metadata Generator (via [07 Metadata Collector](#))**

- Run VLM analysis on each video (Qwen2.5-VL via RunPod)
- Output: Tags, intensity scores, timeline, subject count

**3. Trailer & FYP Generator (via [FYP Generator](#))**

- Auto-generate 1-3 FYP clips (15s, vertical 9:16)
- Auto-generate 1 trailer (30-60s, square or landscape)
- Uses YAMNet audio + Moondream VLM for best moment detection

**4. Thumbnail Generator (NEW)**

- Extract the "most eye-catching" frame using VLM scoring
- Match source video dimensions (e.g., 1920x1080 → 1920x1080 thumb)
- One thumbnail per video

**5. Organizer Engine**

- Create destination folder structure
- Move/copy files with new naming convention
- Support rollback if operation fails

---

 **User Interface**

**Framework:** PySide6 (Qt for Python)  
**Style:** Apple-clean minimalism + Astropunk accents (neon highlights, dark mode, cosmic gradients)

**Main Panels**

Panel	Purpose
Source Browser	Tree view of z:\ with filtering
Content Queue	List of files to process
Organization Preview	Shows proposed folder structure before execution
Progress Dashboard	Real-time logs, progress bars, estimated time
Settings	RunPod URLs, API keys, output paths

---

 **Integration Points**

Tool	Integration Method
07_Metadata_Collector	Import main.py:process_video() as library
FYP_Generator	Import clipper.py + semantic_analyzer.py
RunPod	Shared API configuration for VLM calls

---

## System Requirements

Component	Spec
OS	Windows 10/11
Python	3.10+
Storage	Access to z:\ (S3/DigitalOcean Spaces)
GPU	Optional for local YOLO (thumbnail framing)
Cloud	RunPod GPU instance for VLM (Qwen2.5-VL or Moondream)

## Verification Plan

### Automated Tests

- Unit tests for naming convention parser/generator
- Unit tests for file grouping logic
- Integration test: End-to-end pipeline on a sample video

### Manual Verification

1. **Scan Test:** Load Z:\ and verify file discovery
2. **Preview Test:** Queue a video, verify proposed folder structure
3. **Full Pipeline:** Process one video → verify all assets created in correct structure
4. **GUI Smoke Test:** Verify all panels render without crash

## Proposed Tech Stack

Layer	Technology
GUI	PySide6 (Qt6)
Styling	QSS (Qt Style Sheets) with astropunk theme
Video Processing	MoviePy, FFmpeg, OpenCV
AI (Local)	YAMNet, YOLOv8n (optional)
AI (Cloud)	RunPod (Qwen2.5-VL / Moondream)
Config	config.ini OR settings.json

## UI Inspiration

### Astropunk + Apple Clean:

- Dark charcoal background (#1A1A2E)
- Neon accent colors (Electric Purple #7B2CBF, Cosmic Teal #00D9FF)
- Soft rounded corners, subtle glow effects
- SF Pro or Inter font family

- Minimalist icons with gradient fills