

# 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
<b>Main Video</b>	The primary scene file (PPV, Full Scene)
<b>Metadata</b>	JSON file with tags, intensity scores, timeline
<b>Trailer</b>	30-60s highlight clip
<b>FYP Clips</b>	15s vertical "stop-the-scroll" shorts
<b>Thumbnail</b>	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
<b>Source Browser</b>	Tree view of z:\ with filtering
<b>Content Queue</b>	List of files to process
<b>Organization Preview</b>	Shows proposed folder structure before execution
<b>Progress Dashboard</b>	Real-time logs, progress bars, estimated time
<b>Settings</b>	RunPod URLs, API keys, output paths

## Integration Points

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

## System Requirements

Component	Spec
<b>OS</b>	Windows 10/11
<b>Python</b>	3.10+
<b>Storage</b>	Access to z:\ (S3/DigitalOcean Spaces)
<b>GPU</b>	Optional for local YOLO (thumbnail framing)
<b>Cloud</b>	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
<b>GUI</b>	PySide6 (Qt6)
<b>Styling</b>	QSS (Qt Style Sheets) with astropunk theme
<b>Video Processing</b>	MoviePy, FFmpeg, OpenCV
<b>AI (Local)</b>	YAMNet, YOLOv8n (optional)
<b>AI (Cloud)</b>	RunPod (Qwen2.5-VL / Moondream)
<b>Config</b>	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