AutoCreatorX Spartan Commander Manual

# 📦 System Overview

AutoCreatorX is a fully autonomous, resilient, and offline-first AI-driven content empire framework. It manages trend analysis, content creation, monetization, distribution, and system-level self-healing under a tactical Spartan dashboard interface.

# 🧠 Core Modules

1. Trend Analyzer – Detects viral or emerging content topics across platforms.

2. Content Generator – Creates scripts, captions, hooks using local or external models.

3. Media Factory – Produces videos, thumbnails, memes, and more.

4. Uploader Suite – Connects to APIs for YouTube, TikTok, Instagram, etc.

5. Analytics Engine – Feedback loop for optimizing future content.

6. Revenue Intelligence – Tracks income, suggests high-CPM niches.

7. AI Resilience Layer – Watchman AI, Self-Healing, Cron-based recovery.

# 🛡️ Security & Defense Systems

• Honeypot Detection System

• Shadow IP Rotator (Disabled by default)

• Sentiment Deflector

• Anti-Prompt Injection & Override Logging

# ⚙️ Configuration Files

• spartan\_ai\_config.yaml – Controls fallback, retries, mode, agents

• event\_map.json – Maps system events to auto-responses

• current\_metrics.json – Tracks empire KPIs

• agent\_scorecard.json – Live agent performance data

# 📊 Spartan Dashboard Features

• Live Pulse Graph

• Critical Event Alerts (Flash/Sound/Haptic)

• Commander HUD + Tactical Briefings

• Logs Viewer + Auto Maintenance Switch

• Satellite Expansion Panel (Blog/Podcast engines)

# 🧾 How to Use

• Launch: `python server.py`

• Manual Override: Use `commander\_controls.py` or Spartan Dashboard buttons

• Restart Modules: Use heartbeat monitor or Watchman AI

• View Logs: Run `logs\_monitor\_panel.py` or enable live logs tab

# 🚨 Emergency & Troubleshooting

1. If models are missing – run diagnostics via `spartan\_diagnostics.py`

2. If memory overload – Watchman will notify; override using dashboard

3. If agent fails repeatedly – Scorecard will disable it automatically

4. If any module crashes – Self-Heal Cron will restart and report

# 🔧 Deployment & Recovery

• Backup Entire System: Zip entire project directory manually or via deployment bundle.

• Restore: Extract and run diagnostics to re-init all modules.

# 📚 Appendices

• Requirements.txt – Dependencies

• Full Logs stored under /logs/

• Templates under /templates/ and /assets/

• Static sites under /satellites/