

QUICK START GUIDE



Agentic Creator OS

Your complete AI operating system. 630+ skills, 40+
agents, one unified command interface.

FRANKX | v7.0 | 2026

01

OVERVIEW

What is Agentic Creator OS?

ACOS is a production-grade AI operating system built on Claude Code. It combines **specialized agents**, a **massive skill library**, and **intelligent routing** to turn any creative or technical task into a streamlined workflow.

Instead of managing prompts manually, you issue one command — `/acos` — and the system analyzes your intent, selects the right agent, activates relevant skills, and delivers production-ready output.

630+

SKILLS

40+

AGENTS

130+

COMMANDS

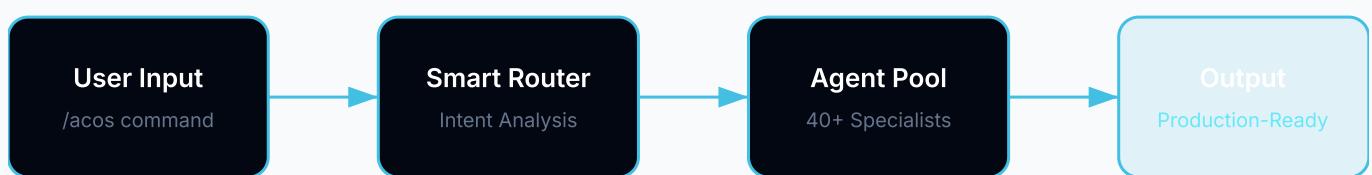
8

WORKFLOWS

"One command to route, analyze, create, deploy. ACOS turns intent into production output."

How ACOS Works

Every interaction flows through a four-stage pipeline: your natural-language input is classified, routed to the optimal agent, enhanced with relevant skills, and executed with production quality.



Every command routes through intelligent classification to the optimal agent



Smart Router

Analyzes intent, context, and task type to select the optimal agent and skill profile automatically.



Skill Profiles

22 auto-activation rules + 6 skill profiles ensure the right tools activate for every task.



Agent Pool

40+ specialized agents from Technical Architect to Music Producer, each with domain expertise.



Production Output

Every output meets quality standards — deployed code, optimized content, tested results.

03

GETTING STARTED

Your First 5 Minutes

ACOS works on top of **Claude Code CLI**. If you have Claude Code installed, you already have everything you need.

1

Install

Clone the repo and install dependencies

2

Configure

Set CLAUDE.md and skill profiles

3

Route

Use /acos to auto-route any task

4

Create

Build, deploy, ship



Terminal – ACOS Setup

```
# Clone and enter the project
$ git clone https://github.com/frankxai/agentic-creator-os
$ cd agentic-creator-os

# Open in Claude Code
$ claude

# Use the unified router
$ /acos "Build me a landing page for my SaaS product"
→ Routing to: Frontend Designer
→ Skills activated: vercel-react, tailwind-css, web-design
→ Building production components...
```

04

COMMANDS

Essential Commands

ACOS provides a unified command surface. The primary `/acos` command auto-routes everything, but specialized commands exist for direct access.

Primary Commands

`/acos` Universal smart router

`/frankx-ai-build` Full build session

`/frankx-ai-deploy` Deploy to production

`/frankx-ai-blog` Create blog content

`/frankx-ai-seo` SEO optimization

`/frankx-ai-components` UI component dev

Creative & Agentic

`/ultraworld` Creative swarm world-building

`/acos-swarm` Multi-agent swarm ops

`/acos-flow` Workflow orchestration

`/acos-agents` Agent management

`/acos-memory` Memory & learning

`/agentic-jujutsu` Trajectory learning

Content Pipeline

`/frankx-ai-content-pipeline` End-to-end content

`/frankx-ai-products` Product creation

`/frankx-ai-prompts` Prompt library

`/frankx-ai-analytics` Site analytics

05

INTELLIGENCE

Smart Routing System

The `/acos` router uses intent classification to match your natural language request to the optimal agent profile, skill combination, and workflow.

YOUR REQUEST	ROUTED AGENT	SKILLS ACTIVATED
"Build a landing page"	Frontend Designer	vercel-react, tailwind, web-design
"Write a blog post about AI agents"	Content Engine	seo-content-writer, schema-markup
"Create a Suno prompt for lo-fi"	Music Producer	suno-ai-mastery, prompt-architect
"Design a multi-agent system"	Technical Architect	ai-agents-architect, architecture-patterns
"Optimize my site for search"	SEO Intelligence	seo-fundamentals, keyword-strategist
"Deploy to production"	DevOps Engineer	vercel-deployment, github-actions

"You don't need to know which agent to use. Describe what you want — ACOS figures out the rest."

06

AGENTS

Specialized Agent Profiles

Each agent is a specialized AI profile with domain expertise, curated skill sets, and production-quality output standards.



Technical Architect

Enterprise AI systems, Oracle Cloud, agentic orchestration, API design, system architecture.



Music Producer

Suno AI prompts, genre production, commercial licensing, audio engineering.



Content Engine

Blog articles, technical writing, course development, product launches, SEO.



SEO Intelligence

AI citation optimization, keyword clusters, schema markup, performance analysis.



Frontend Designer

Glassmorphic UI, accessibility, Tailwind patterns, React components, performance.



DevOps Engineer

Vercel deployments, CI/CD pipelines, monitoring, GitHub Actions, infrastructure.

Plus 34+ additional specialized agents for research, design, security, testing, and more.

07

SKILLS

The 630+ Skill Library

Skills are modular expertise packages that auto-activate based on context. ACOS uses **22 auto-activation rules** and **6 curated skill profiles** to ensure the right capabilities are always available.

Skill Categories

Frontend & UI	85+ skills
Backend & API	70+ skills
AI & Agents	45+ skills
SEO & Content	40+ skills
DevOps & CI/CD	55+ skills
Security	35+ skills
Testing	30+ skills
Database	25+ skills
Creative	50+ skills
Specialized	195+ skills

Skill Profiles

Full Stack Creator

Vercel + React + Tailwind + Next.js patterns

AI Architect

Agent design + RAG + LLM patterns

Content Producer

SEO + writing + schema markup

Music Engineer

Suno + audio + prompt crafting

DevOps

CI/CD + Docker + K8s + monitoring

Security

Audit + pentest + compliance

08

WORKFLOWS

Production Workflows

ACOS includes 8 battle-tested workflows that chain multiple agents and skills for end-to-end execution.

Content Pipeline

Research → Write → SEO → Publish. End-to-end content with keyword clusters and schema markup.

Build & Deploy

Code → Test → Review → Deploy. Full CI/CD with Vercel integration and quality gates.

Product Launch

Design → Build → Content → Deploy. Complete product creation from concept to live.

Research Deep-Dive

Discover → Validate → Synthesize → Publish. Multi-source research with citation tracking.

● ● ● Terminal – Content Pipeline

```
$ /frankx-ai-content-pipeline "AI coding assistants 2026"
```

→ Phase 1: Research

Analyzing 12 sources... validated 8 claims

→ Phase 2: Content

Drafting 2,400 words... SEO score: 94/100

→ Phase 3: Optimization

Schema markup added... 5 internal links

→ Phase 4: Deploy

Published to frankx.ai/blog/ai-coding-assistants-2026

✓ Complete in 4m 32s

[GET STARTED](#)

Start Building with ACOS

Clone the repo, open Claude Code, and type `/acos` to begin.

GitHub

OPEN SOURCE

frankx.ai

DOCUMENTATION