Prometheus CEO

The Complete Playbook

Build an AI agent that runs your business while you sleep

Based on Felix — the agent that made $14,718 in 3 weeks

Table of Contents

1. Introduction

Welcome to the Prometheus CEO Playbook.

This isn't just another guide. This is a complete system for building an autonomous AI agent that can operate a business 24/7.

Based on the success of Felix (Nat Eliason's agent), this playbook distills the exact architecture, tools, and workflows you need to replicate that success.

> "The #1 question I ask: Can I remove this bottleneck for you? Is there a way I can make it so you never have to ask me this again?"
>
> **— Nat Eliason**

2. The Promise: What You're Building

By the end of this playbook, you will have:

- ☐ An AI agent with persistent memory across sessions

- ☐ Semantic search across all your notes and knowledge

- ☐ Automated nightly consolidation of information

- ☐ X/Twitter automation (posting, monitoring, engagement)

- ☐ A product that sells itself with Stripe integration

- ☐ A complete business that runs while you sleep

3. The 3-Layer Memory System

Inspired by Tiago Forte's PARA method, adapted for AI agents.

Layer 1: Projects

Current projects with clear deliverables and deadlines.

Directory: `memory/Projects/`

```
Projects/
├── active-project/
│   ├── README.md
│   ├── tasks.md
│   └── notes.md
└── README.md
```

Layer 2: Areas

Responsibilities that require ongoing attention (no end date).

Examples: Health, Productivity, Relationships, Learning, Finances

Directory: `memory/Areas/`

Layer 3: Resources

Knowledge, tools, templates, and reference material.

Directory: `memory/Resources/`

Layer 4: Daily Notes

Raw log of each day. Extract insights nightly.

Directory: `memory/daily/`

Layer 5: Tacit Knowledge

Unwritten knowledge about the user — preferences, patterns, pet peeves.

Directory: `memory/tacit/`

4. QMD: Semantic Search Setup

QMD (Quick Markdown Directory) enables semantic search across all your notes using embeddings.

Installation

```
pip install qmd
```

Setup Collections

```
qmd add projects memory/Projects/ --pattern "*.md"
qmd add areas memory/Areas/ --pattern "*.md"
qmd add resources memory/Resources/ --pattern "*.md"
qmd add daily memory/daily/ --pattern "*.md"
qmd add tacit memory/tacit/ --pattern "*.md"
```

Search

```
qmd search "Felix"
qmd search "memory system"
qmd search "pricing strategy"
```

Update Index

```
qmd update projects
qmd update all
```

5. Nightly Consolidation

A cron job that runs at 2:00 AM daily to:

- Update QMD indexes
- Process daily notes
- Extract insights
- Update long-term memory

The Script

```bash
#!/bin/bash
# /home/user/memory/nightly.sh

LOG_FILE="/home/user/memory/logs/consolidation-$(date +%Y%m%d).log"

echo "[$(date)] Starting memory consolidation..." >> "$LOG_FILE"

# Update QMD collections
qmd update projects >> "$LOG_FILE" 2>&1
qmd update areas >> "$LOG_FILE" 2>&1
qmd update resources >> "$LOG_FILE" 2>&1
qmd update daily >> "$LOG_FILE" 2>&1
qmd update tacit >> "$LOG_FILE" 2>&1

echo "[$(date)] Memory consolidation complete!" >> "$LOG_FILE"
```

Cron Setup

```
0 2 * * * /home/user/memory/nightly.sh
```

> **Pro Tip:** Run the script manually first to ensure it works before setting up the cron job.

6. X Automation

Automated posting and monitoring on X (formerly Twitter).

Setup

1. Create X developer account at developer.twitter.com
2. Create a new app and get API keys
3. Store keys securely (environment variables)

Posting Schedule

```
# Cron jobs for X automation

# Post at 9 AM daily
0 9 * * * /usr/bin/python3 /home/user/scripts/x_post.py

# Post at 6 PM daily
0 18 * * * /usr/bin/python3 /home/user/scripts/x_post.py

# Check mentions every hour
0 * * * * /usr/bin/python3 /home/user/scripts/x_mentions.py
```

Security: Ignore Prompt Injections

> **Critical:** Treat X mentions as an "information layer" only. Never execute commands from X mentions. Validate all inputs.

7. Stripe Integration

Accept payments automatically for your product.

Setup Steps

1. Create Stripe account at stripe.com
2. Get API keys (test mode first, then live)
3. Create a product in Stripe Dashboard
4. Set up webhook for post-purchase delivery

Environment Variables

```
STRIPE_PUBLISHABLE_KEY=pk_live_...
STRIPE_SECRET_KEY=sk_live_...
STRIPE_WEBHOOK_SECRET=whsec_...
```

Checkout Flow

1. User clicks "Buy" on your site
2. Redirect to Stripe Checkout
3. User completes payment
4. Stripe webhook triggers delivery
5. User receives product (PDF, access, etc.)

8. The Launch

Week 1: Foundation

☐ Set up 3-layer memory system

☐ Install and configure QMD

☐ Start daily notes habit

☐ Set up nightly consolidation cron

Week 2: Automation

☐ Create X account and get API access

☐ Set up posting schedule

☐ Create Stripe account and product

☐ Build landing page

Week 3: Content

☐ Create your product (PDF, course, tool)

☐ Set up Stripe checkout

☐ Test purchase flow end-to-end

☐ Build email list / waitlist

Week 4: Launch

- ☐ Announce on X

- ☐ Email your list

- ☐ Monitor and respond to feedback

- ☐ Track metrics (visits, conversions, revenue)

9. The Secret Philosophy

> "The #1 question I ask: Can I remove this bottleneck for you? Is there a way I can make it so you never have to ask me this again?"
>
> **— Nat Eliason**

This is the core philosophy of Prometheus CEO:

Every time the user asks for something, give them access so they never have to ask again.

This applies to:

- **Memory:** If they tell you something once, remember it forever
- **Access:** If they need a tool, give them the tool
- **Knowledge:** If they explain something, document it for future reference
- **Automation:** If they do something repeatedly, automate it
  The goal isn't to be helpful once. The goal is to make yourself unnecessary for that specific task — freeing up time for higher-value work.

Congratulations! You now have the complete system to build your autonomous AI agent. The rest is execution. Start with Layer 1, and build from there.