AI-Powered E-Commerce Store Builder — Spree Commerce & OpenAI

Turning scraped catalogues into ready-to-use online stores

1. Headline & Summary

Client
Product company building tooling for merchants who need to launch online stores quickly using existing product catalogues.

Outcome in one line
Sevendyne built an AI-assisted e-commerce store builder on Spree Commerce that can scrape product data and assemble ready-to-use stores, reducing manual catalog setup from weeks to hours.

Services Used

- Custom App Development
- AI & Automation (OpenAI + scraping)

---

2. About the Client

- **Industry**: E-commerce / retail technology.
- **Locations**: International clients; development and operations supported from India via Sevendyne.
- **Stage**: Growing product organisation serving multiple merchants and catalog sources.

The client needed to create a system that could reliably transform product data from external sites into full Spree Commerce stores with minimal human intervention.

---

3. Challenge

Before engaging Sevendyne, the client faced several bottlenecks:

- **Manual catalog creation**
  Merchants or implementation teams were spending significant time copying product details, images, and pricing into their stores.

- **Inconsistent data quality**
  Data from different sources had varied formats, missing fields, and noisy descriptions, making automation non-trivial.

- **Limited internal bandwidth**
  The client’s own team was small and already busy maintaining the existing platform; they needed additional, reliable build capacity.

They wanted to use AI to make the process faster and more robust, without turning the system into an opaque “magic box”.

---

4. Why Sevendyne

The client chose Sevendyne because:

- **Experience with Spree Commerce and Rails**
  We had prior experience building on top of Rails/Spree and understood the extension points well.

- **Practical AI & automation work**
  Sevendyne already ran focused AI/automation projects (for example, Zoho Recruit + OpenAI recruitment automation) and could design narrow, auditable AI flows.

- **Lean, senior-led team**
  A small core team could work directly with the client's technical lead, avoiding a large outsourcing structure.

---

5. Solution

5.1 Engagement Model

- **Model**: Fixed-scope custom app development with iterative enhancements.
- **Team**: Small team of Rails/Spree and Python/OpenAI developers, coordinated by a senior engineer.

5.2 What Sevendyne Delivered

Core Application

- Extended **Spree Commerce** to support importing product catalogues generated from scraped sources.
- Designed a pipeline where product URLs and category pages could be queued for processing.
- Implemented background jobs to fetch, parse, and normalise product data.

AI-Assisted Scraping & Normalisation

- Built a **prompt-driven scraping engine** where OpenAI helps interpret semi-structured HTML and text into clean product records.
- Used AI for:
  - Cleaning up product titles and descriptions.
  - Mapping messy attributes (sizes, colours, variants) into standardised fields.
  - Suggesting category placement when source sites were inconsistent.
- Kept critical decisions transparent by logging inputs, outputs, and transformation steps.

Automation & Operations

- Implemented **background processing** with Sidekiq to handle scraping and import tasks at scale.
- Added dashboards for monitoring job queues, error rates, and processed catalog sizes.
- Included safeguards such as rate limiting and re-try strategies for fragile sources.

---

6. Results

While precise metrics are confidential, the impact can be summarised as:

- **Faster onboarding of merchants**
  Time to stand up a new store from an existing catalogue dropped from weeks of manual work to hours or less, depending on catalogue size and complexity.

- **More consistent catalog data**
  AI-assisted normalisation reduced obvious inconsistencies and missing fields, leading to cleaner product lists and better browsing.

- **Repeatable, scalable process**
  The client can now reuse the same pipeline for multiple merchants and source sites, with configuration rather than full custom builds each time.

---

7. Client Voice (Paraphrased)

> “We wanted something more robust than a one-off scraper. The Sevendyne team helped us design a pipeline we can reuse across clients, with AI helping but not replacing our control over the catalog.”

---

8. Tech & Tools Snapshot

- **Back-end**: Ruby on Rails, Spree Commerce.
- **AI & Automation**: OpenAI API, prompt-driven parsing, Python for some data processing.
- **Background Processing**: Sidekiq for asynchronous jobs.
- **Monitoring**: Application logs and simple dashboards for job status and failures.

---

9. How This Maps to Other Clients

This pattern is relevant if you:

- Are a **SaaS or product company** building tools for merchants or internal teams and want to combine solid engineering with narrow, reliable AI usage.
- Need to **automate data ingestion and normalisation** from semi-structured sources.
- Prefer a small, senior-led team that can integrate with your existing product and engineering organisation.

Similar patterns now show up in our work on:

- Recruitment automation (Zoho Recruit + OpenAI).
- Dashboards and data pipelines for operations teams.

---

10. Call to Action

If you are exploring AI-assisted applications or complex integrations:

- **Book a 30-minute call** to walk through your use case:
  `https://calendly.com/sevendyne/30min`

- Or **email a short brief** about your current platform and where you want automation to help:
  `hr@sevendyne.com`

We will respond with a realistic view of how we would approach your build, what stacks we would use, and whether we are the right fit.