

SMBifyRank – A to Z Technical Refactor Guide

Purpose: This document is designed to be handed directly to an AI coding agent or developer to refactor, stabilize, and professionalize the SMBifyRank SaaS platform without a full rewrite.

1. Executive Summary

SMBifyRank is an AI-powered CMS and static website generator for local businesses. The current system is unstable because HTML is being treated as the source of truth. This guide explains how to move to a data-first, CMS-grade architecture.

2. Core Problem Diagnosis

The main issue is direct HTML editing and regeneration conflicts. AI and users both modify HTML, causing overwrites, instability, and broken state management.

3. Correct Mental Model

AI generates structured JSON. Users edit structured JSON. HTML is generated only at publish time. JSON stored in the database is the single source of truth.

4. Target Architecture

Layers:

- UI Layer (Next.js Editor)
- Content Layer (JSON stored in DB)
- AI Orchestration Layer
- Static Rendering Layer
- Hosting Layer (Netlify/Vercel)

5. Page Content JSON Schema

Each page must be stored as structured data.

```
{ pagId, template, sections:[ { type, data } ] }
```

6. Editor Refactor Rules

Editor must never manipulate DOM HTML directly.

Editor loads JSON → edits fields → saves JSON → previews render.

7. AI Integration Rules

AI must only return JSON.

AI is not allowed to output HTML.

AI can regenerate individual sections only.

8. User vs AI Conflict Prevention

Once a section is edited manually, it must be locked from AI overwrite unless the user explicitly resets it.

9. Publishing Pipeline

Publishing is a stateless process:
JSON → Renderer → Static HTML → Deploy.
Publishing must not modify stored content.

10. Blog System Rules

Blogs follow the same JSON model as pages.
Markdown or rich-text stored as data.
Rendered to static HTML at publish time.

11. Database Responsibility

Database stores:
• Users
• Sites
• Pages (JSON)
• Blog posts (JSON)
• Publish history

12. Deployment Strategy

Main SaaS runs on Vercel.
Generated sites deploy to Netlify or Vercel static projects.
Custom domains mapped via platform APIs.

13. What NOT To Do

Do not:
• Store raw HTML as primary data
• Let AI directly write templates
• Mix editor state with publish logic

14. Refactor Execution Order

Step 1: Introduce JSON content layer
Step 2: Refactor editor to JSON-based
Step 3: Restrict AI to JSON output
Step 4: Clean publish pipeline

15. Final Outcome

After refactor, SMBifyRank becomes a stable, scalable, multi-tenant AI CMS suitable for agencies, SaaS resale, and long-term growth.