

Product Requirements Document (PRD)

项目名称: Hacker News Chinese

版本: v0.1 (MVP)

状态: Draft

技术栈: React + FastAPI + Supabase

1. 项目背景与目标 (Background & Objectives)

1.1 背景

Hacker News (HN) 是全球高质量技术信息的聚集地，但存在两个主要痛点：

- 语言门槛:** 全英文内容阻碍了中文母语者的快速阅读。
- 信息过载:** 每天产生大量内容，用户难以在点击链接前判断文章价值，且难以快速提取核心观点。

1.2 目标

构建一个 **AI 驱动的技术资讯聚合平台**，通过自动化流程获取 HN 热门文章，利用 LLM 进行中文翻译与深度总结，并提供基于文章内容的问答互动功能，帮助用户高效获取技术资讯。

1.3 核心价值 (Value Proposition)

- 无障碍阅读:** 中文标题与摘要。
- 高效获取:** AI 提炼的核心观点与优缺点分析。
- 深度交互:** 针对特定文章的 RAG (检索增强生成) 问答。

2. 用户流程 (User Flow)

- 浏览列表:** 用户进入首页，浏览“当前热门”文章列表（显示中文标题 + 简短摘要 + 热度）。
- 查看详情:** 用户点击感兴趣的文章，查看 AI 生成的详细分析报告（背景、核心观点、技术细节）。
- 原文跳转:** 用户可点击链接跳转至英文原文。
- AI 对话:** 用户对文章内容有疑问，在侧边/底部对话框提问，AI 基于文章正文进行回答。

3. 功能需求 (Functional Requirements)

• 3.1 后端与数据处理 (Backend - FastAPI)

ID	功能模块	详细描述	优先级
B1	定时任务调度	系统每隔固定时间（如 30 分钟）自动触发数据更新流程。	P0
B2	HN 数据获取	调用 HN API 获取 Top Stories，筛选 Score > 100 或 Top 30 的文章。	P0
B3	网页内容爬取	根据 HN 原文 URL 爬取正文内容。需处理基础的反爬策略，提取主要文本（Main Content Extraction）。	P0
B4	LLM 处理管道	调用 LLM API 对爬取的文本进行： 1. 翻译标题。 2. 生成一句话摘要。 3. 生成结构化分析 (JSON)。	P0
B5	数据持久化	将原始元数据、爬取的正文、LLM 生成的结果存入 Supabase 数据库。需处理去重逻辑。	P0
B6	文章列表 API	提供 GET /api/articles 接口，支持分页，按时间或热度排序。	P0
B7	对话 API	提供 POST /api/chat 接口，接收 article_id 和 message，检索对应的 raw_content 作为 Context，流式返回 LLM 回答。	P0

• 3.2 前端界面--暂定 (Frontend - React)

ID	页面/组件	详细描述	优先级
F1	首页信息流	卡片式布局，展示：中文标题、一句话摘要、HN 分数、发布时间、原文链接。	P0
F2	文章详情页	展示 AI 生成的完整报告（使用 Markdown 渲染）。包括：背景、核心要点列表、潜在影响等。	P0
F3	加载状态	在数据加载或 AI 回复时展示 Skeleton Screen 或 Loading Spinner。	P1
F4	对话组件	类似 ChatGPT 的聊天界面，集成在详情页中。支持用户输入、发送、接收流式打字机效果的回复。	P0

• 3.3 AI 提示词策略 (Prompt Strategy)

- **Summarizer Agent:** 输出需强制为 JSON 格式，以便前端结构化展示。
 - 字段示例: title_cn, summary, key_points (Array), tech_stack_mentioned.
- **Chat Agent:** System Prompt 需包含：“你是一个技术助手，请仅依据以下提供的文章内容回答用户问题...”

4. 数据模型 (Database Schema - Supabase)

核心表结构设计 (`public.articles`):

```
create table public.articles (  
  id bigint generated by default as identity primary key,  
  hn_id integer unique not null,           -- HN 原始 ID  
  original_title text not null,            -- 英文原标题  
  original_url text,                       -- 原文链接  
  score integer,                           -- HN 热度分数  
  posted_at timestamptz,                   -- 发布时间  
  raw_content text,                        -- 爬取的正文 (用于 RAG)  
  translated_title text,                   -- 中文标题  
  summary text,                           -- 一句话摘要  
  detailed_analysis jsonb,                 -- 结构化分析 (JSON)  
  created_at timestamptz default now()  
);  
  
-- 索引  
create index articles_score_idx on public.articles (score desc);  
create index articles_posted_at_idx on public.articles (posted_at desc);
```