

HV-Com——一个全程使用 Vditor 的评论系统

作者: [lixiang810](#)

原文链接: <https://ld246.com/article/1642593438775>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

HV-Com

WIP - 功能和结构可能迅速变化, 且兼容性不被保证

[GitHub 地址](https://ld246.com/forward?goto=https%3A%2F%2Fgithub.com%2Flixiang8102FHV-Com)

简介

HV-Com 是一个使用 TypeScript 编写的评论系统。它使用 [Vditor](https://ld246.com/forward?goto=https%3A%2F%2Fgithub.com%2FVanessa219%2Fvditor) 作为 markdown 编辑器, [lute](https://ld246.com/forward?goto=https%3A%2F%2Fgithub.com%2F88250%2Flute) 为 markdown 引擎。

特性

HV-Com 的 HV 有很多个意思:

-

- HeaVy - 这套评论系统的规模不小, 对前后端都将有不小的性能开销。gzip 下 `React` + `HV-Com` + `Vditor` + `lute` = `45k` + `42kB` + `110kB` + `315kB` = `51kB`, 其中 `Vditor` 暂未 minify, 体积偏大, 将在它最新的 release 中修复

以上述缺陷为代价, HV-Com 有如下优势:

-

- Have Vditor - HV-Com 使用 Vditor 作为 markdown 编辑器与渲染器, 做到从编辑到查看的面“所见即所得”。
- High Value - HV-Com 拥有不亚于论坛评论区的功能, 在这里可以进行高价值的公共讨论。事实上, HV-Com 离论坛只差一个发帖功能。

它还拥有这些特性:

-

- 使用 markdown, 拥有从编辑到查看的全过程所见即所得。
- 支持 Vditor 可编辑的所有格式, 包括:

-

- LaTeX 公式
- mermaid.js 流程图
- abc.js 乐谱
- 代码高亮
- 详见 [Vditor readme](https://ld246.com/forward?goto=https%3A%2F%2Fgithub.com%2FVanessa219%2Fvditor)

-

- 可自行部署后端, 也可托管。
- 支持 PostgreSQL、MySQL、SQLite 三大 SQL 数据库, 未来可能支持 [OrbitDB](https://ld246.com/forward?goto=https%3A%2F%2Fgithub.com%2Forbitdb%2Forbit-db), 成为去中心化评论系统。
- 具备良好的可扩展性:

-

- 全程以 React + TypeScript 编写
- 使用 jest 进行自动化测试以保证稳定
- 使用 OpenAPI 规范对后端接口进行定义与检查

-

- 支持 JSX 语法, 可作为 React 组件引入。
- 自由软件。以 AGPLv3 开放所有源代码。

如何开发

```
<pre> <code class="language-bash highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl">sudo npm install -g yarn
</span> </span> <span class="highlight-line"> <span class="highlight-cl">git clone https://g
thub.com/lixiang810/HV-Com
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class="high
ight-nb">cd</span> HV-Com
</span> </span> <span class="highlight-line"> <span class="highlight-cl">yarn
</span> </span> <span class="highlight-line"> <span class="highlight-cl">yarn dev
</span> </span> </code> </pre>
```

<h2 id="---背景">:world map: 背景</h2>

<p>Vditor 作为所见即所得的 markdown 编辑器而闻名，又有 <code>Vditor.preview()</code> 法可对 markdown 进行前端渲染，这让据此实现一个评论系统成为可能。与此同时，现有的评论系却无一使用 vditor 作为其编辑器，绝大多数评论系统的富文本编辑器都不支持所见即所得，为普通用户的使用带来了门槛。</p>

<p>我的博客曾希望使用 <code>valine</code>，但它是闭源的。与此同时，它还有不小的安全风险，更何况它还要求使用国内需要绑定身份证的 LeanCloud。</p>

<p>我转而使用 gitalk。这个过程见于我的博。但随后 GitHub 在中国大陆的访问受限，而且“注册 GitHub 账号”对评论者也是一个门槛这会降低评论积极性。</p>

<p>此后我选择了 Isso 和 HashOver，其中 HashOver 的各类功能令我印象深刻，如果能对它的前进行二次开发以加入 vditor，我将会继续使用 HashOver。然而，HashOver 的前端代码事实上已过陈旧，需要使用现代前端技术进行重写。</p>

<p>waline 对我而言有不小吸引力，但它:</p>

不完全是 TypeScript。

markdown 只能分屏渲染，无法进行所见即所得的编辑，支持的格式也不够多。

评论区只能按时间倒序查看，且不能分页。

用了 vue，而我的技术栈在 React。

使用 GPLv2 作为协议。开源有余，自由不足。

<p>正好 React 也缺评论系统，而我此前编写 n2rv 时又正好积累了相关经验和代码，于是决定完全重写一个。</p>