**文档修改历史**

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **时间** | **修改内容** | **修改人** |
| **V1.0.0** | 2017-12-07 | 新建文档 | 高炳炳 |
| **V1.0.1** | 2017-12-08 | 补充插件对比 | 高炳炳 |
| **V2.0.0** | 2018-07-09 | 修改插件对比模块内容，补充新插件,名称解释等。 | 高炳炳 |

**在线公式编辑器组件研究**

# 目标

查找目前流行的稳定的**在线公式编辑组件，**选出一款合适的，供开发使用。

# 思路

先查阅相关内容资料，之后进行筛选，对比，在之后进行部署，测试，最终选出合适的一款组件。

# 要点

在线公式编辑，易操作，用户对公式的输入与编辑简单可行。

# 名词解释

**latex** : 是一种基于ΤΕΧ的排版系统 , 利用这种格式， 可以充分发挥由TeX所提供的强大功能，latex公式则是有特殊符号（$或其它）开头结尾的符号组合，不同符号组合代表数学中的不同特殊符号，通过特殊的识别机制，可以在页面上展示出对应的实际符号。

  **MathJax** : MathJax是一个JavaScript引擎，用来显示网络上的数学公式, 它可以工作于所有流行的浏览器上。可以对 latex 公式代码识别，进行实际渲染。

# 插件介绍

1. **ueditor**

 UEditor 是由百度「FEX前端研发团队」开发的所见即所得富文本web编辑器，具有轻量，可定制，注重用户体验等特点，开源基于MIT协议，允许自由使用和修改代码。



1. **Mathquil**

mathquill能满足大多数数学公式的输入和渲染，使用数学符号的 LaTeX 表示方法



1. **JMEditor**

JMEditor（JavaScript Math Editor）是基于CKEditor、jQuery、mathquill等组件开发的，轻量级、开放源代码、所见即所得、无任何插件的在线公式编辑器。CKEditor与jQuery应用的十分广泛，不再过多介绍。需要指出的是mathquill，使用html+css+js实现公式编辑的效果，他把dom的力量发挥到了极致。



1. **MathquillBasedEditor**

是一款所见即所得的jquery**数学公式编辑器**插件。该数学公式编辑器依赖于mathquill，可以通过点击图标来直接生成乘方和开方等数学公式。

 

1. **基于latex和MathJax 编写的编辑器**



如上图，是知乎的公式编辑器，点击选择常用的公式，代码选择区和预览区实时更新，用户只需要修改替换掉其中的字母即可实时预览插入。

通过查看其功能的实现，对比我们目前使用的也是latex代码，能否模拟实现一个公式编辑器？如下：是基于 latex代码 和mathJax渲染实现的公式编辑demo。用户选择常用公式，插入，修改替换对应位置字母即可。



# 插件对比

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **插件** | **支持功能** | **Github活跃度** | **文档是否齐全** | **兼容性** | **评价** |
| **ueditor** | 公式编辑，可直接选择。 | Star：3207Fork：1344Issues：未开放 | 齐全，百度已停止维护。 | IE8不支持 | 相对其他，可取。但对IE8的支持是个问题。 |
| **Mathquil** | 公式编辑，不支持选择。 | Star：1185Fork：361Issues：213 | 有官方网站，提供文档， | 兼容IE8 | 兼容性很好，但是在公式的输入不能直接选择，需要用户自己输入各符号对应代码。 |
| **JMEditor** | 公式编辑，可直接选择。 | 官方未维护。 | 有官方网站，但没有系统的文档。 | 兼容IE8,不支持谷歌浏览器的https协议。 | 没有系统文档，且https不兼容问题较大。 |
| **MathquillBasedEditor** | 公式编辑，可直接选择。 | Star：10Fork：5Issues：1 | 有简单的使用说明文档 | IE8不支持 | 基于Mathquil封装，功能较少，复杂公式没有。 |
| **自制公式编辑器** | 公式编辑，可直接选择。 |  | 后续可以完善相关设计文档。 | 兼容IE 8 | 基于目前使用技术自制，兼容性较好，但是对用户相对没有做到完全傻瓜。 |

注：1. **Star** 为用户的收藏量，**Fork**为用户克隆量，**Issues**为问题。

2. 兼容性从**ie8**开始考虑，是因为目前我们只做**ie8及以上**的兼容。

 从上述对比可以看出，目前可以选择的有 **ueditor 和 自制编辑器，**但前者目前已不维护，后续我们维护可能是个问题， 而自制编辑器虽然对于用户没有完全傻瓜，但是兼容性等方面相对较好，可考虑使用（若后续使用，demo可进一步完善封装，且梳理相关文档等）。