# ScrapingIDE

## 名词解释

爬虫规则配置文件

## 设计目的

网络爬虫是根据人工配置的规则来抓取内容的，无法实现全自动化。当前的做法是通过Chrome DevTools来分析页面结构，然后手工完成规则的配置文件的编写。而配置文件通常需要为每个网站定制，编写工作量大，而且后期需要维护，效率低下。

ScrapingIDE的设计初衷是为了简化爬虫规则的编写工作，提升端到端的爬虫规则的编写效率。ScrapingIDE提供向导式的工具界面，简化规则的编写工作。

## 使用限制

ScrapingIDE无法实现的傻瓜式的全自动配置文件编写工作，使用ScrapingIDE必须了解基本的HTML/CSS概念，ScrapingIDE主要目的是为了提升效率，而无法简化爬虫的规则。

## 架构

由于ScrapingIDE需要代替Chrome DevTools来分析页面，目前比较常见的Web分析工具主要是DevTools以及Firefox的Firebug。综合考虑，使用的跨平台的Firebug Lite，Firebug Lite使用JavaScript编写，并且使用BSD开源协议，符合我们使用的功能要求，可以方便的修改源代码。主体架构是围绕Firebug Lite展开的。

ScrapingIDE是基于Chrome 应用扩展实现的，Chrome应用扩展和Chrome插件类似，但是功能更加完善，权限控制更加严格。主要通过HTML/CSS/JavaScript实现。

## 代码结构

## 后续思路

1. 保存的缺省是项目配置文件，文件不可见

2. 下面的区域做成配置项，可视化的，项目工程文件本身不可见

3. 最后可以导出供爬虫使用的配置文件。