Easy-bootstrap V3 规划书

前言：

为了提高WEB项目的开发效率，决定把前端页面中的通用部分，封装成JQuery插件，最终的JQuery插件集就成了Easy-bootstrap v1.0 。从多个项目中的实际使用情况来看，达到了提高项目开发效率的初期目标，基本满足了项目开发需求。目前Esay-bootstrap最新版本为版本v2.0。

在之前的版本中，由于当时项目周期紧，框架存在一些问题。首先，框架中不合理特性，例如通过使用在标签上加class初始化组件的方式，容易造成多次初始化的问题；然后，框架体系结构不合理，各个组件的关系是并列的，根本没有层级关系，这种情况导致了框架中出现大量的重复代码；其次，组件的特性单一，部分组件需要添加例如拖拽，缩放的特性；再其次，老版本JS代码与HTML代码耦合度过高，修改组件风格难度过高。最后，之前版本，文档残缺，更新不及时，在新版本中，需要出具规范、标准的API文档。

# 项目规划

## 收集组件功能变动

1. Input输入框，不再封装成JQuery插件，只提供Input样式，需要Input 验证器。
2. Window组件，需要添加拖拽、缩放功能，需要模态窗口。

## 分析设计框架体系结构

Class类为框架类的基类，提供了JS对象继承的规范接口extend，Widgets为所有JQuery插件类的祖先类，提供了options参数处理。Grid为表格类父类。

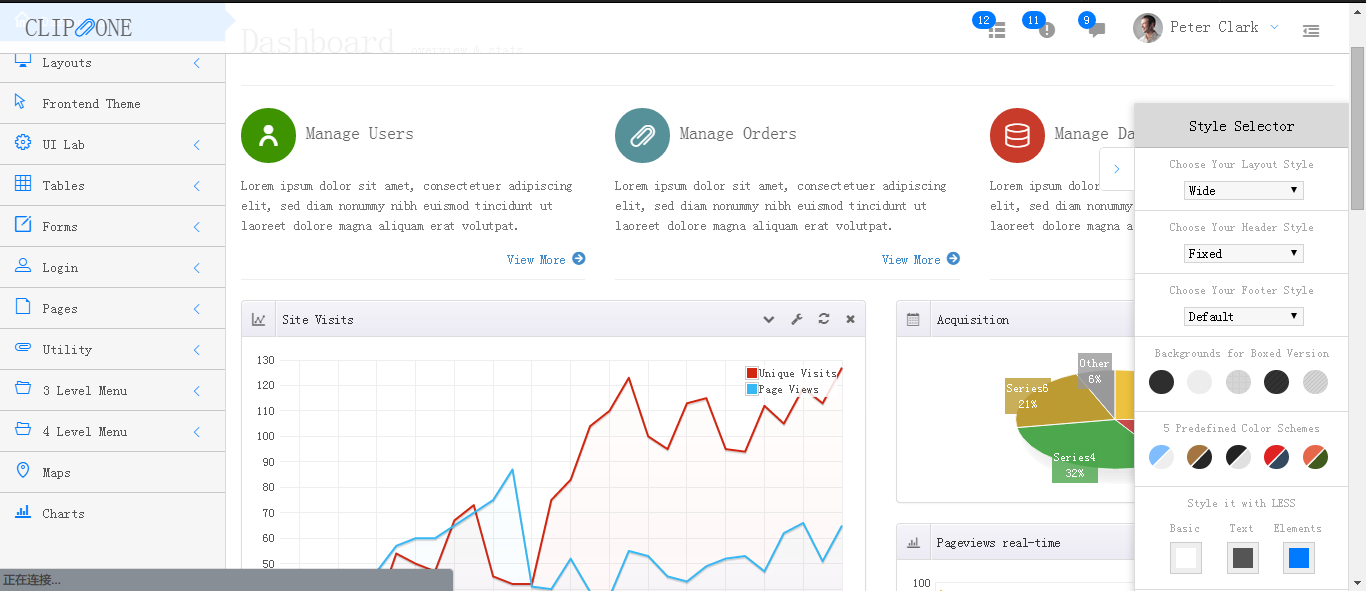
## 抽离公用样式

使用CSS的预编译技术LESS，在一个theme-default.less文件中定义通用的样式设定，在lib.less中编写例如圆角、阴影等特性。

在生产环境中，需要修改风格时，只需要修改这两个文件，重新编译即可。

## 抽离CMS类网站模板

根据实际业务需求，制作模板页面。



# 项目质量

## 使用构建工具Grunt

使用Grunt作为项目的构建工具，自动化以下任务：LESS编译成CSS；JS CSS 压缩，混淆；单元测试；打包最终产品包；

## 单元测试

使用QUnit框架对项目进行单元测试。

## 文档

出具规范的API文档。

# 项目周期