Atitit.web ui 组件化 vs mvc

组件化 与 mvc并不矛盾。。单双方适用的地方有所不同。。

React推荐以组件的方式去重新思考UI构成，将UI上每一个功能相对独立的模块定义成组件，然后将小的组件通过组合或者嵌套的方式构成大的组件，最终完成整体UI的构建。例如，Facebook的instagram.com整站都采用了React来开发，整个页面就是一个大的组件，其中包含了嵌套的大量其它组件

每个组件都是自我继集成html,css,js的。。

在每一个组件的内部，才使用mvc分离core和ui。。

对于MVC开发模式来说，开发者将三者定义成不同的类，实现了表现，数据，控制的分离。开发者更多的是从技术的角度来对UI进行拆分，实现松耦合。

对于React而言，则完全是一个新的思路，开发者从功能的角度出发，将UI分成不同的组件，每个组件都独立封装。

在React中，你按照界面模块自然划分的方式来组织和编写你的代码，对于评论界面而言，整个UI是一个通过小组件构成的大组件，每个组件只关心自己部分的逻辑，彼此独立。

React认为一个组件应该具有如下特征：

****（1）可组合（Composeable）****：一个组件易于和其它组件一起使用，或者嵌套在另一个组件内部。如果一个组件内部创建了另一个组件，那么说父组件拥有（own）它创建的子组件，通过这个特性，一个复杂的UI可以拆分成多个简单的UI组件；

****（2）可重用（Reusable）****：每个组件都是具有独立功能的，它可以被使用在多个UI场景；

****（3）可维护（Maintainable）****：每个小的组件仅仅包含自身的逻辑，更容易被理解和维护；

React是基于组件化的开发，那么组件化开发最大的优点是什么？毫无疑问，当然是复用，下面我们来看看React中到底是如何实现组件的复用的，这里我们还写一个例子来说吧，代码如下：

许多框架使用Handlebars或Mustache等模板引擎来处理视图层。但React相信视图和控制器应该相互依存在一起而不是使用第三方模板引擎

React不孤立双向绑定，儿angru孤立。。

React的问题与缺点：

资料少

可视化模板不容易得到dw的支持。。作为模版，得到类似dw 这类可视化Html ide的支持的是必须的。。