# 1、前后端分离的好处

（1）彻底解放前端

（2）提高工作效率，分工更加明确。

（3）局部性能提升

（4）降低维护成本

# 2、前后端分离的概念

后台只需要提供API接口，前端调用Ajax实现数据呈现。

# 3、场景和要求

对于前后端分离的应用场景，不是所有的场景都适合，但是大多数项目都能够通过前后端分离来实现。大多数后台应用我们可以做成SPA应用（单页应用），而单页应用最主要的特点是局部刷新，这通过前端控制路由调用Ajax，后台提供接口就可以实现。而且这样的方式用户体验更加友好，网页加载更加快速，开发和维护成本也降低了不少，效率明显提升。同样的，在展示类网站和移动APP页面中前后端分离也同样试用。前后端不分离的情况下，服务端要单独针对Web端做处理，返回完整HTML，这样势必增加服务端的复杂度，可维护性差，而web端需要加载完整的HTML，一定程度上影响网页性能，这对于移动端性能为王的地方非常的不友好。随着前端技术的发展和迭代，前端MVC框架应运而生，利用目前主流的前端框架，如React、Vue、Angular等我们可以轻松的构建起一个无需服务器端渲染就可以展示的网站，同时这类框架都提供了前端路由功能，后台可以不再控制路由的跳转，将原本属于后端的业务逻辑全部丢给前端，这样前后端分离可以说是最为彻底。前后端分离的实现对技术人员尤其是前端人员的要求会上升一个层次，前端的工作不只是切页面写模板或是处理一些简单的js逻辑，前端需要处理服务器返回的各种数据格式，还需要掌握一系列的数据处理逻辑、MVC思想和各种主流框架。

# Web前端路由

就是根据url分配到对应的处理程序，原理和路由器原理相同，不同的请求地址会交给路由处理来转发给相应的控制器处理，所以说路由就可以在转发前修改转发地址。举个例子，在 MVC 结构中，route 的会将 uri 映射到 controller，具体业务逻辑在 controller 里面进行。

为什么要使用路由？传统web开发是每一个请求地址都会请求服务器来进行处理，但是用户有些操作则无需请求服务器，直接页面端修改下逻辑就能达到目的，这种最好使用路由，也许题主会有疑问：直接使用js处理下不就行了。使用js直接处理这些是可以的，事实上以前我们也这么做，但是这样做不便于用户收藏当前页，因为使用js时并不更新url，但是使用路由时，url也是随着改变的，用户浏览到一个网页时可以直接复制或收藏当前页的url给别人，这种方式对于搜索引擎和用户来说都是友好的。

# 4、心得和体会

（1）项目一开始制作前端页面的时候，我不再需要后台给我配置服务器环境了。

（2）项目的前端文件可以在需要调用后台接口的时候丢进服务器就好了，完全不需要事先放进去。

（3）增加一个项目页面需要配置路由的时候不再需要让后台同事给我加了，自己前端搞定。

（4）前端文件里不再掺杂后台的代码逻辑了，看起来舒服多了

（5）页面跳转比之前更加流畅了，局部渲染局部加载非常快速

（6）页面模板可以重复使用了，前端组件化开发提高了开发效率。