

## 修改用户信息

1. 上传组件中添加on-success="handleSuccess" , 这样当上传完成之后会调用handleSuccess方法, 方法中得到上传的图片路径
2. 给保存按钮添加点击事件, 调用save方法, 在方法中 准备一个user对象里面装着需要修改的昵称/图片路径和id , 向服务器发出修改请求
3. 在UserController中添加update方法处理请求, 在方法中通过传递过来的信息判断是否需要修改图片, 如果需要修改先查询到之前的图片路径 删除图片, 最后调用mapper的update方法进行修改
4. 实现mapper中的update方法
5. 在application.properties里面设置d:/files文件夹为静态资源文件夹, 这样才能够让客户端访问到上传的图片

## 发布文章

1. 在稿件管理页面中, 点击三种不同的发布按钮跳转页面时传递type参数
2. 在postArticle.html发布页面中的created方法里面 得到传递过来的type值, 并让页面中的单选组件和type进行双向绑定
3. 在created方法中向服务器发请求获取type值对应的二级分类数据, 获取到数据后赋值给categoryArr数组, 让页面中的选择器组件和数组进行绑定
4. 给页面中的类型单选组件添加input事件, 当值发生改变时发请求获取对应分类的数据
5. 让页面中的二级分类选择器组件进行双向绑定
6. 让富文本编辑器和vue对象中的c.content进行关联
7. 实现上传封面的功能
8. 给上传按钮添加点击事件, 调用post方法
9. 在post方法中 判断是否选择了二级分类, 是否选择了封面 然后向服务器发出请求, 并将和页面绑定的c对象提交给服务器
10. 创建ContentController 里面添加addNew方法 处理请求, 在方法中调用mapper的insert方法把数据保存到数据库
11. 实现mapper里面的方法
12. 实现发布视频功能

## 稿件管理页面

1. 将articleManagement.html页面中三个表格对应的数组contentArr 改成三个不同的数组
2. 添加created方法, 在里面发请求获取三种不同类型的数据, 并把得到的数据赋值给三个数组
3. 在ContentController中添加方法 处理上面的请求, 然后调用mapper的selectByType方法, 把接收到的type和当前登录的用户id传递到方法中
4. 实现mapper中的selectByType方法 配置xml里面的SQL语句 (内容表和分类表关联查询)