前端架构

经过开会需求讨论之后，在群里曾经提醒过前端的架构统一，但执行情况不理想，有必要进行文档的规定统一，有据可依。该文档并非定稿，请组员详细阅读，如果有意见要提前提出，没有的话请严格执行，如果后期因为不遵循该文档规定的架构，尤其是UI组件和css样式，造成的返工自己负责。

1. 前端分工。

主页，由冯俊钧负责，优先实现需求文档中优先级为1的功能。

审稿人、主编、个人设置，由王骞负责，这里注意个人设置，每个角色个人中心都有个人设置部分，为了保证风格统一，交由同一个人负责。

作者、编辑、管理员，由李西炜负责。

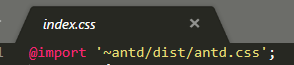
1. 项目技术架构

前端以Reactjs为核心技术，采用以Reactjs为支持对象的一系列技术，具体包括：

Router：单页面应用必须要有路由控制实现页面跳转，Reactjs的路由刚刚经历过版本更新，router4与之前的版本变化较大，强调了动态路由的理念，技术学习理解存在一定的困难，为了提高开发效率，加上router3比较推出已久，且**图书馆项目也采用router3**，学习成本低，网上文档资料也相对丰富，一定程度上更加稳定。所以这里项目组统一要求使用router3，router4可以作为个人学习方向，但为了保证后期代码合并，请确保后期不要影响整体进度即可。

Antd：UI组件库，这是为了保证前端页面风格统一，采用统一的组件库，满足了正常的美化需要。如果各自用css实现各种组件效果，个人风格混乱不堪，组件的开发一般情况下是不应该和项目开发并行的，所以在这种情况下请尽量采用统一的、已有的组件库。使用组件包括两步：

组件按需加载——

Css样式全部加载——

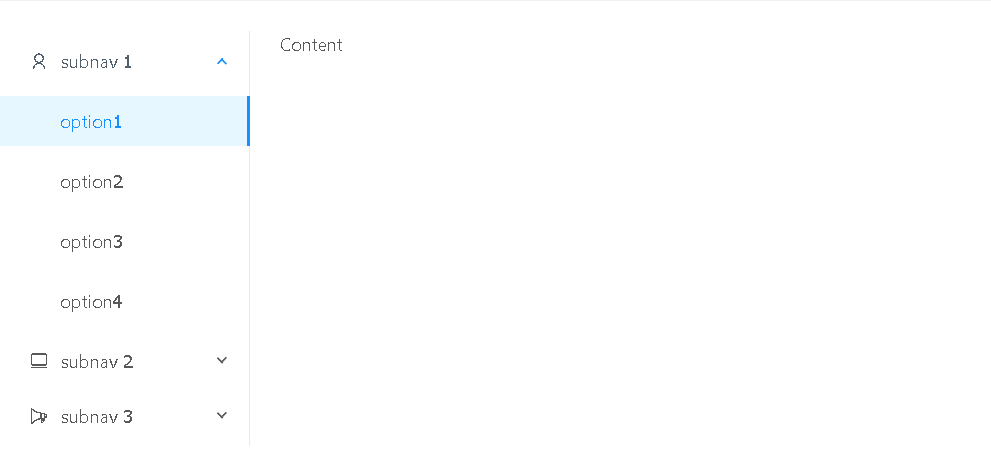
以下是需要考虑的组件：

1. 所有的图标优先使用，不要随便使用图片

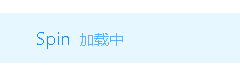


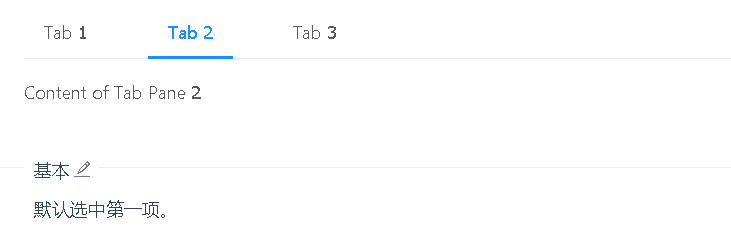
1. 个人中心布局优先使用，左侧功能菜单，右侧内容区域

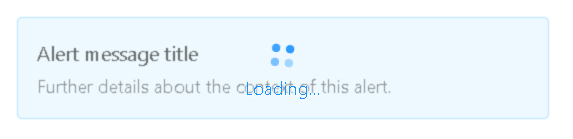




1. 个人中心的布局中，内容区域除了管理员之外，分为“未完成”和“已完成”两个标签页。且标签页内容打开时要覆盖加载提示“加载中……”。

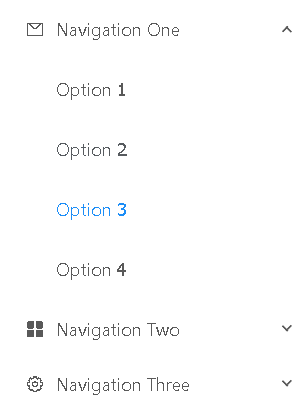






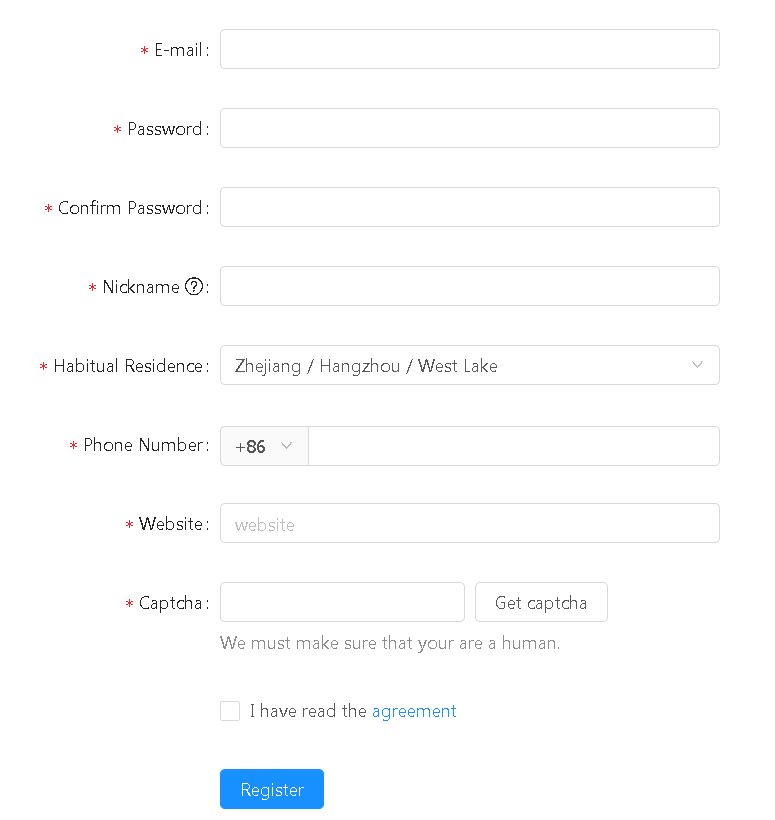
1. 个人中心的布局中，左侧功能菜单使用导航菜单，我们功能较少，不需要折叠，一般分为**任务功能、个人设置**两部分。





1. 个人设置中的表单

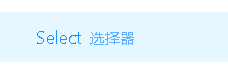




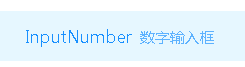
1. 其余的表单组件











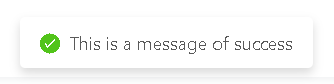


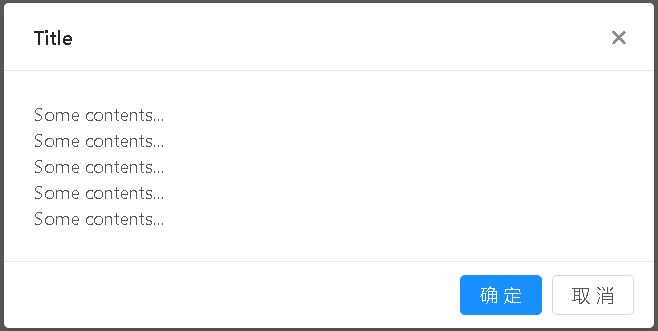


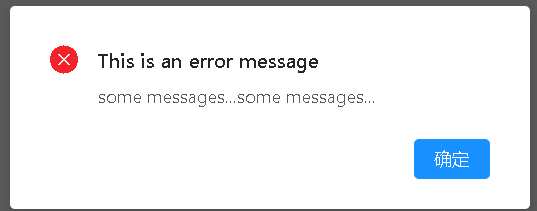
1. 提示。所有的成功提示使用Message全局提示。所有操作弹窗、错误提示、警告提示使用Modal。操作弹窗要求左确定右取消，其余的提示单独一个确定按钮。





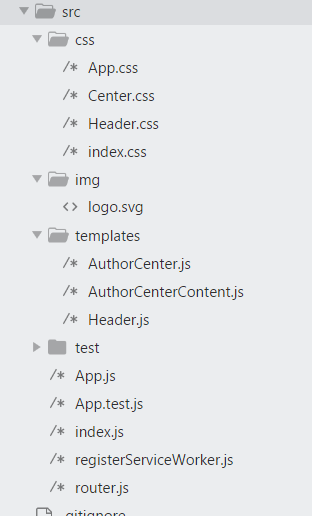






Axios：前后端交互通过接口进行，这里我们使用axios，一种良好封装的ajax，用于http请求。

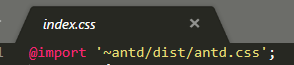
1. 项目文件架构。



通过官方脚手架进行项目创建后，一般情况下我们的代码只涉及src文件的变化。这里为了方便后期合并，参考图书馆的结构进行规定文件架构。文件夹包括css、img、templates，这里的test文件夹和App.test.js是用于测试的，我们暂时不需要，可以不创建，也可以保留。

App.js作为主页；index.js作为项目入口，指向路由；router.js作为统一路由，用于页面跳转；registerServiceWorker.js用于开发缓存资源，无需改动；以上这些和其它文件夹独立出来。

css文件夹：统一放置所有css文件，这里注意index.css作为整个项目的全局css，不允许乱填写其它组件的css，目前暂时只用于引入Antd的css。另外所有的组件css不允许直接对标签进行统一样式修改，比如<html>、<div>等等，这是因为react的css是全局生效的，如果直接修改标签样式，会导致所有的组件受到样式污染和冲突。请使用id、class对html节点元素进行命名，然后规定id、class的样式，而且命名注意不能和其它页面重复，建议命名要体现页面、功能和标签，比如作者中心的上传按钮：authorUploadBtn



img文件夹：统一放置所有图片文件，比如省医院图标图片。

templates文件夹：统一放置所有js组件，js命名建议与组件名字一致，保持首字母大写，其它文件可以采用正常驼峰式命名。

1. 版本控制。

前端版本控制使用阿里云git，https://code.aliyun.com/13402883297/onlineSubmissionSystem。请各位创建好帐号，把帐号发给我，邀请进入项目。然后建立自己的分支，每天把自己的改动上传，到后期我会进行分支合并。

最后再次提醒一下，如果对该文件架构有意见，要及时提出并商讨修改，请不要特立独行。如果你的项目文件结构和规定不一样，那么在后期代码合并的时候，你需要做大量修改来满足合并要求，这是因为组件和资源的import是使用路径引入的，你的结构不一致，就需要大量修改路径，这不是简单复制粘贴能解决。而且到了后期合并上线之后，每次功能修改，如果你的结构不一致，当你修改完代码后提交合并，也需要每次修改结构和路径。所以，为了减少个人的返工，请严格执行规定，到了项目后期，文档统一规定是不会随意为了迎合不符合规定的人而做出修改。