《WEB基础与前端开发》项目文档

项目名称： QDU服务网站

学 期： 2022 秋季学期

班 级： 2021级软件 04 班

姓 名： 王林浩

指导教师： 解新峰

一、项目概述

**为了方便传输文件，我仅提交了vue的打包文件，源码已上传到gitee丢人：**

**https://gitee.com/Lin47/just\_try**

了解到大部分同学尤其是大一新生存在对选课流程不了解，对课程内容不了解，对老师考核方式不了解等问题，导致出现的大范围选课困难与大量退改选课，浪费了大量的时间与精力，所以尝试开发了这个网页，由于立项之初在学习vue开发，所以大部分内容都是由vue实现。

站点的整体结构上，将页面分为主页面和其他的子功能页面，子功能以课程的选择辅助为主，辅助以一定的娱乐和论坛内容，同时将课程按照类型分为不同的子类。

具体有:

主页（ class\_main ）：显示按时间排序的课程评价信息流。

必修（ class\_msut ）:必修课程

选修\_核心（ class\_xcore ）:选修课的核心部分，每人需修够3门

选修\_非核心( class\_ncore ): 选修课的核心部分，每人需修够2门

一学分英语（ class\_eng ）: 必须选修的，必修课之外的英语课，每人需修够1门

美育（ class\_art ）:修够1门即可毕业

体育（ class\_pe ）:必修，但是选择多样，大一大二进行，每年1门

论坛( o\_talk ): 论坛面板

登录( u\_login )

注册( u\_reg )

站点的页面布局上，采用了与日常作业时使用的相类似的左右分割模式，左侧存放菜单按钮，右侧作为主要页面显示部分。左侧占比较右侧小很多。

二、页面效果及技术实现

在页面的切换上，由于日常实验已经使用过iframe实现相类似的页面，所以个人网页选择了通过vue-rouetr的插件实现，在router/index.js内实现引入各个vue界面，同时调用了element-ui的库来简化操作，再通过将router-link标签放入左侧el-aside内，将对应的router-view放入了右侧的el-main当中。如下图所示。

Button的实现没有使用传统的h5的input或者button,而是使用了vue中的el-button标签，原因是动画效果比较好看，方便实现。

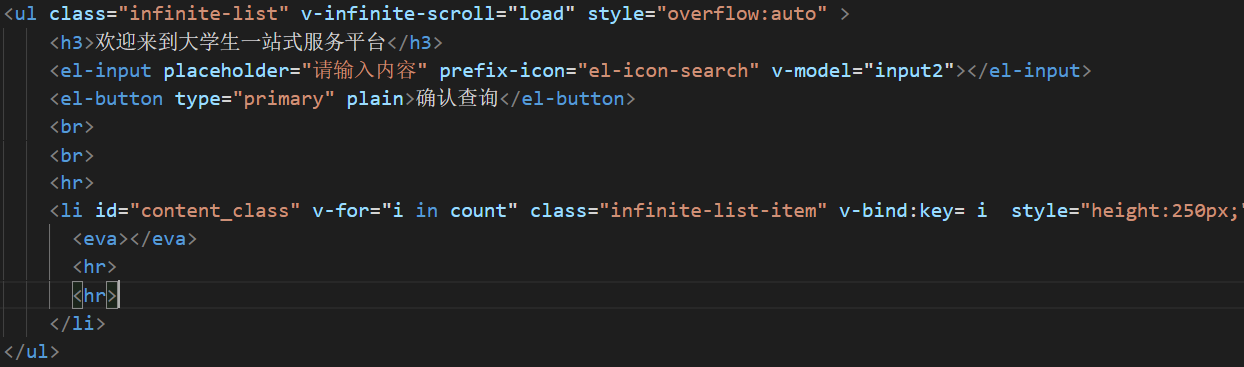
整体的背景使用了background的渐变色。

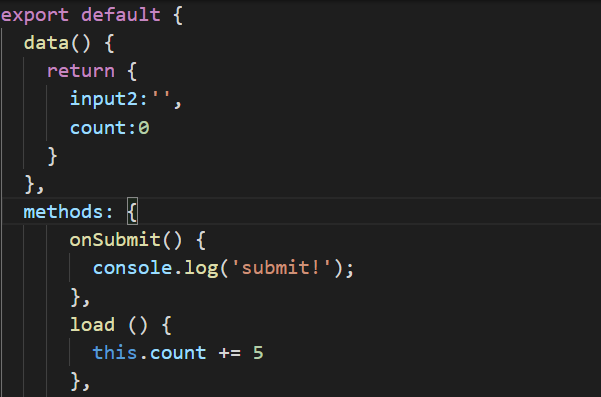


图表 1 总体布局预览

在左侧的el-aside框内，从上到下依次是用img标签实现的图片插入，v-router的跳转按钮，通过elment-ui实现的走马灯效果的广告位，以及最后的登录与注册按钮。

主页（ class\_main ）：加入了一个信息流作为主要内容，信息流的实现主要是靠js和v-for实现的懒加载，

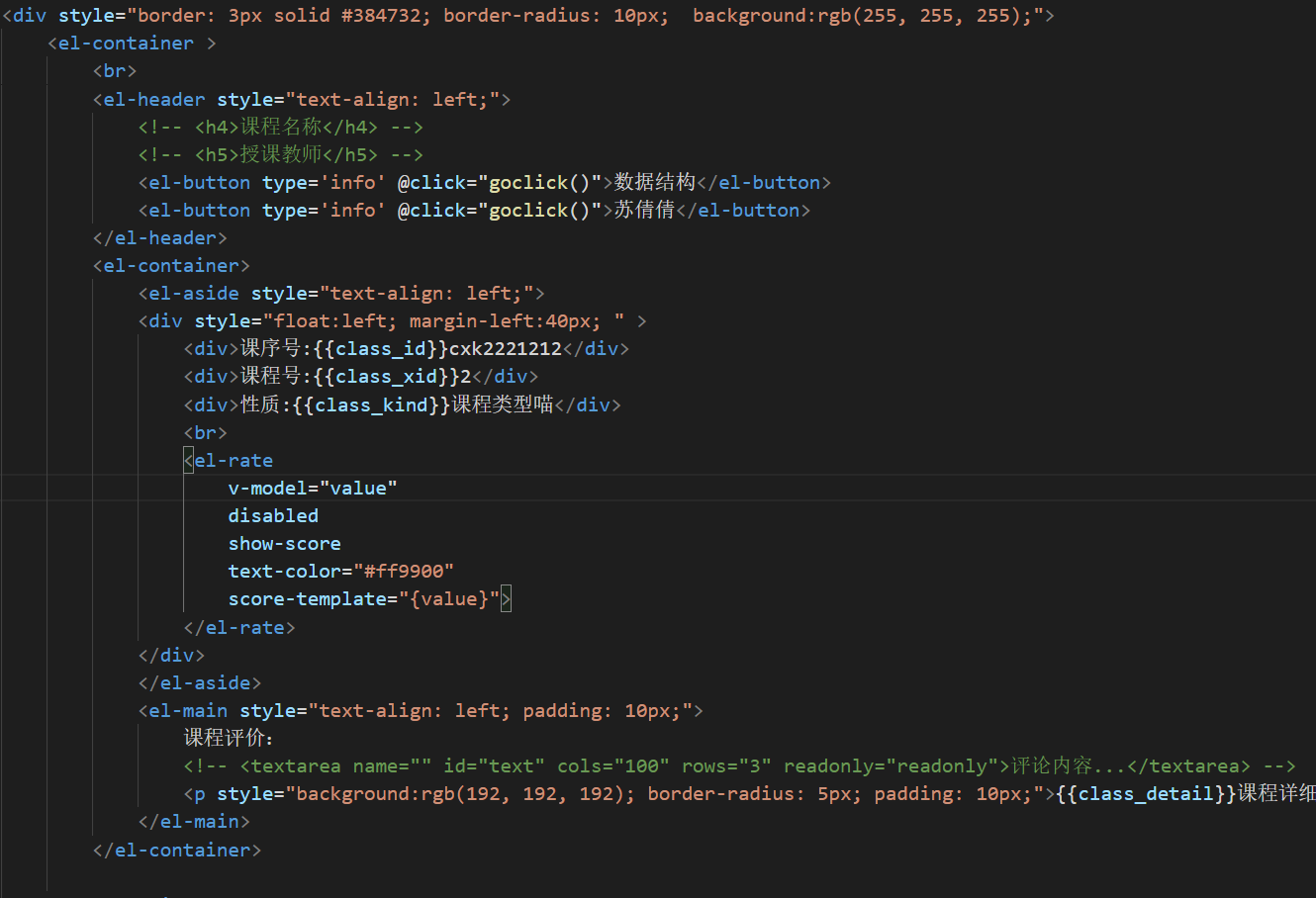




其中加载的，是来自于class\_eva的内容，包括授课教师、课程名称等信息的简单页面，同时将教师和名称都设置为按钮，方便进一步进入详细的内容页面。同时p标签展示详细评论，并使用圆角美化。当其中的eva框加载的大小触及到页面底部，通过load函数实现+5的操作从而读出更多的eva。



图表 2 eva实现的课程评论展示信息流框



图表 3 eva的内部详细代码

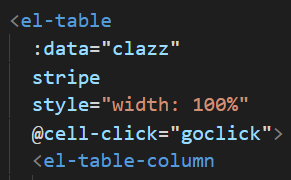
同时，点击事件由 @click 调用js函数实现， js函数部分调用了windows.open去实现新标签页的打开，部分则仍旧使用了router路由实现。

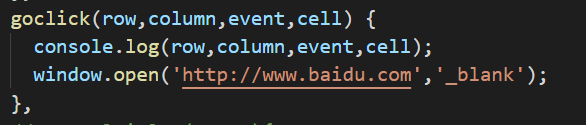
另外，大部分课程的页面其实大同小异，这里以必修页面为例。



图表 4 必修路由页面展示

必修页面主要使用了一个简单的input搜索框和element-ui中的el-table标签来追求更加方便的实现列表的功能。同时为了方便详细信息的查询，将goclick函数通过@cell-click绑在了table标签上，实现对列表内容点击后页面的跳转。





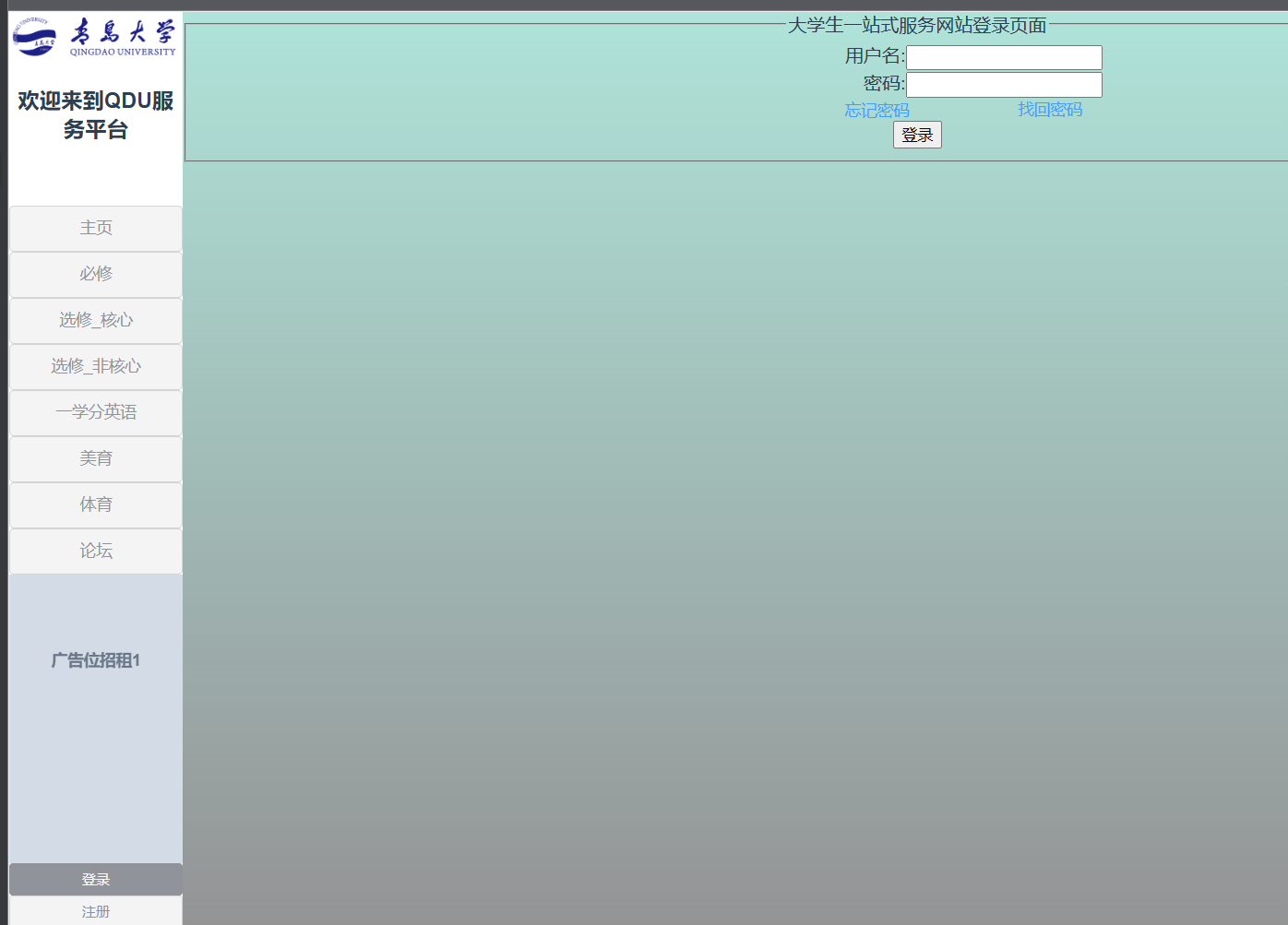
由于未与数据库连接，方便调试，暂时将内容绑定为百度首页。

论坛( o\_talk ): 论坛面板是我最头疼的版块之一，主要是他的实现需要在一个页面内使用多个路由，需要去路由管理文件中设定默认的路由路径。主要的成分是一个简单的走马灯加上一个帖子vue的信息流和一个由elment-ui中 tabs实现的页面跳转。



图表 5 论坛页面展示

此后较为相似的两个页面是左下角的登录和注册，主要目的是锻炼一下十四周学习的表单的相关使用。



图表 6 登录



图表 7 注册

由于刚刚做完相关的实验，所以也没有去详细再重复做。通过div包含lable和input，之后给lable一个固定大小，再加上align:right右对齐实现输入框的对齐，好看很多。

最后则是用npm run build指令完成了项目的打包。

三、项目总结

总体上开发了一个基本上无法正常运行的网页，仍具有巨大的进步空间，比如说基本上没有考虑后端的连接，大量地方实现的方法存在小bug影响美观，没有很好的掌握比例布局导致缩放页面比例后会出现无法正常使用的情况。

我最大的感受就是，虽然说大家都说前端简单，但是真要精通前端开发也完全不是一件轻松的事情，会出现各种各样的浮动错误，很多时候高度封装的各种组件实际上是不完全透明的，会出现很多无法解释的bug，而且因为实现原理的不知晓，也很难靠逻辑推理去解决问题，大概率只能去一点点尝试或者百度前人的经验。通过课程的学习，我对前端页面代码的撰写也有了一定的框架体系，知道出了什么样子的bug后应该去哪里百度什么来解决问题。

老师讲课一直很照顾我们这些听课较慢的同学，一学期下来，我也大体能记住大部分html常用的标签，css部分重要的内容还有小部分JS的相关代码。比如说display、position、float、align、还有对数字的检验等等。关于vue为数不多的收获可能就是有对关命令提示符的相关的操作获得了一定的熟悉，了解了一些vue的特性和用node.js通过npm和Vue.Cli脚手架去构建项目，还有就是在github大量查询代码也提升了我一定的资源检索能力。也顺路学会了vscode配置c++和相关代码的环境。

心得上，由衷的感叹程序化学习的重要性，自己东看一段西看一段代码只会有越来越多的困惑和bug，最好的学习方法还是在了解整体语言的特点之后，从官方文档看起，一点点建立逻辑理性的知识结构。然后就是要加强对JavaScript和css的学习，只有学好这两个东西才能写出动态很强的html页面。只依赖vue的组件是写不出自己想要的网页的。

问题的话，如果前后端交互的话，有没有可能直接让数据库去直接做后端。我常能看到Vue+SpringBoot+myBatis的组合，如果页面简单的话，是不是可以直接省略掉springboot部分的代码。还是感觉只讲前端的没有后端内容实在是非常的难受。

最后，感谢解老师一学期以来的教导！