

选课交流论坛(bbs)开发计划

一、学习相关基础知识

项目开发之前由于大家对于 python 和 django 具有不同的学习和认知程度，部分同学对于 python 基本语法不太了解，部分同学对于 django 框架不太了解等等

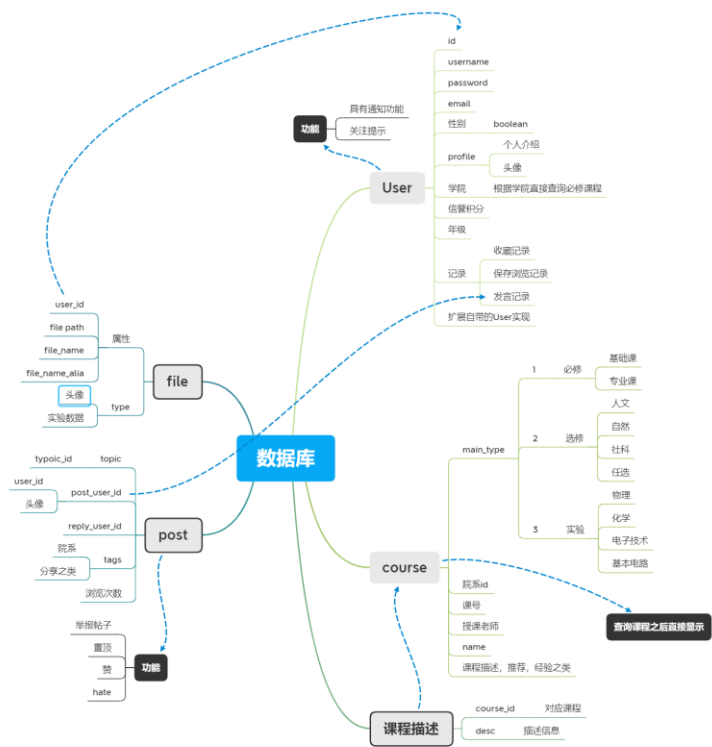
而对于我们的这个项目而言，这两者是必不可少的，但是，知道了这两个也还是远远不够的，对于前端的构建，我们还需要学习 html, css, javascript 等基础知识，这样才能搭建出前端的界面并对其进行美化等。

我们组组员现在大致分为两大类，一类负责前端，一类负责后端，组长邓君海在学习的同时负责对项目成员进行协调工作，并负责大体的框架，但分组并不意味着每个组员的工作是固定的，组员的工作可以灵活安排

方向	人员	学习内容
前端	崔湘晗	HTML, CSS, Javascript bootstrap, layui(学会调用)
	昌洵成	
	于子晗	
	邓君海	
后端	祁山青	django, 数据库
	吴嘉锐	
	汤历欧	

二、进行数据库设计

数据库是网站的一大重要部分，可以说如果没有数据库，那么前端是单调的。经过上一轮的讨论，我们主要需要实现下面这些数据库：



这也只是数据库的一些大体的结果框架，实际设计的时候会有其他的优化删改等。

feedback_feedback	
id	integer
user_id	integer
username	varchar(30)
subject	varchar(100)
content	text
type	bool

(简单的用户反馈数据库设计)

三、进行网页设计

数据库的设计自然是核心，但是没有网页上的呈现，最终只是一些单调无比的数据。对于网站，我们主要需要设计如下几个界面：



对于网页的设计方面，我们可以选择参考其他网页的样式，并结合 [bootstrap](#) 和 [layui](#) 所提供的元素排版方式。

在这里输入您的意见

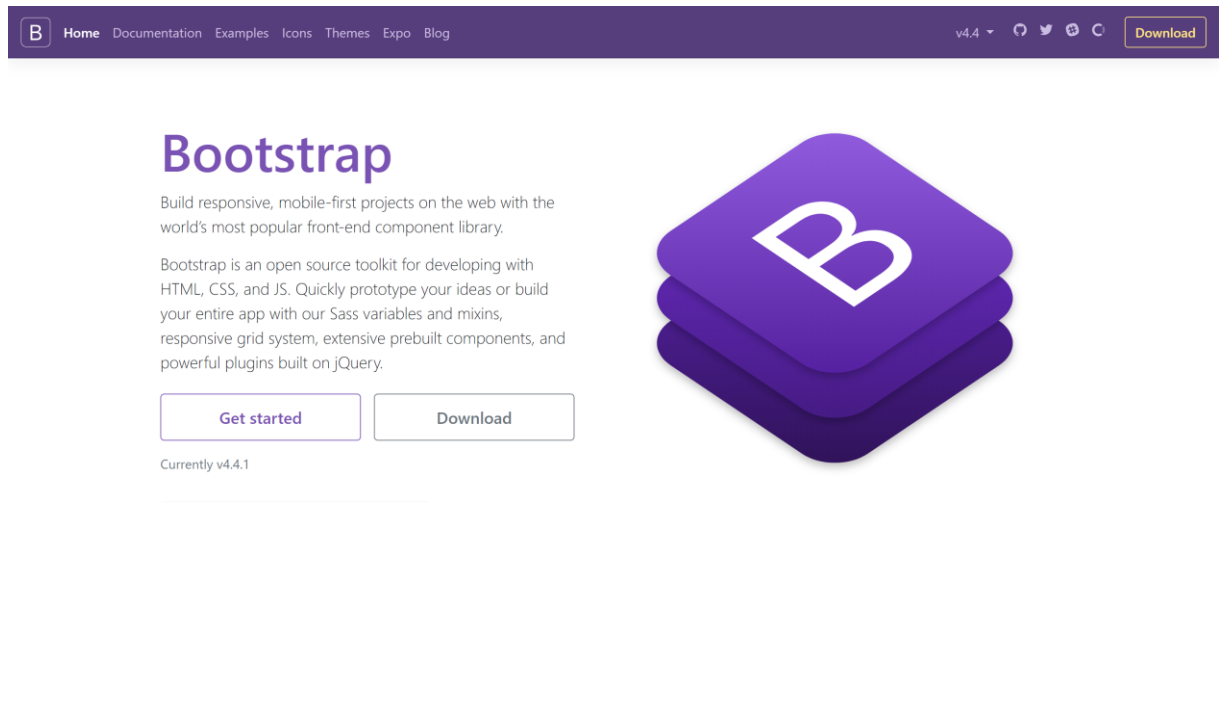
☒ 建议

☐ 举报

提交

(简单的用户反馈界面)

bootstrap 中有许多比较美观的排版方式, 中间也提供了许多 example, 好好利用其中的资源对于我们对网页的美化具有很大的帮助作用



layui 类似于 bootstrap, 但是其中提供了许多 bootstrap 中所没有的模块, 如果我们需要可以直接调用



四、后端设计

后端的话，主要分为用户对于课程的查询，文件的上传下载显示，用户的注册登录等操作，另外还需要对发帖，回复，帖子查询等进行相应的操作。

后端的操作需要前端的需求，后端可以按照前端的意愿来提供相应的操作

```
@login_required
def index(request):
    """反馈界面"""
    if request.method == "POST":
        subject = request.POST.get("subject")
        content = request.POST.get("content")
        type = bool(int(request.POST.get("type")))
        message = None
        if not subject.strip():
            message = {"info": '主题为空', 'tag': "warning"}
        if not content.strip():
            message = {"info": "内容为空", 'tag': "warning"}
        if content.strip() and subject.strip():
            message = {"info": "反馈成功，请等待管理员处理", "tag": "success"}
            feedback = Feedback(user_id=User.objects.get(user_id=request.user.id).id,
                               username=request.user.username,
                               subject=subject,
                               content=content,
                               type=type)
            feedback.save()
        request.session["message"] = message
        print(message)
        return redirect(reverse("feedback:index"))
```

(简单的用户反馈后台处理)

五、总结

在第一个 sprint 周期，我们决定最重要的是学习相关的基础知识，只有拥有了这些基础知识的铺垫，我们才能够进行进一步的迭代，以及对代码进行重构。

另外我们需要大体确定自己的网站需要实现哪一些功能，然后将这些功能从简单到难一步一步实现。

另外一点，由于我们的数据库需要课程的一些信息，所以可能还需要收集一些信息来填充我们的数据集