

项目构建思路

项目采用spring MVC的模式来编写。

- 通过编写Dao层实现与数据库的交互
- 通过编写service层实现通过Dao层间接的访问数据库
- 通过编写controller层实现响应前端页面请求，并间接使用service层完成数据交互

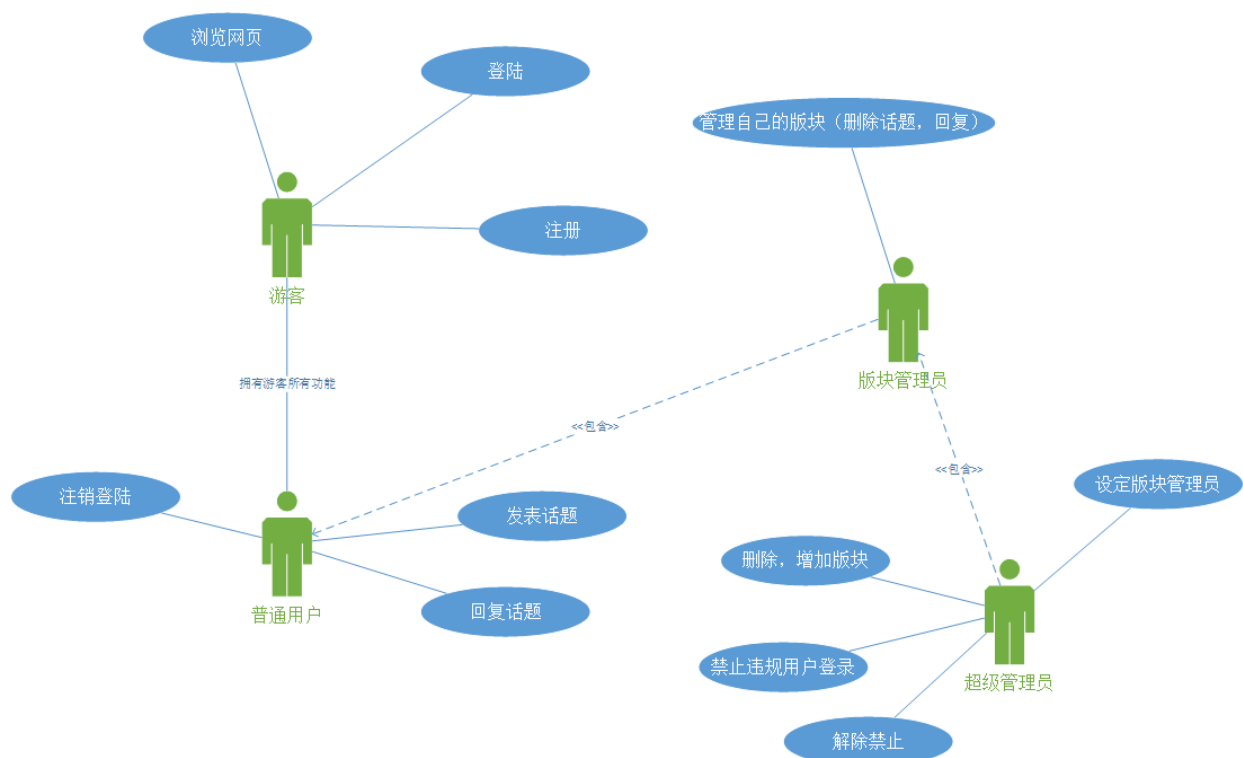
也就是说Dao层是直接与数据库交互的，而service层是直接使用Dao层的函数，而controller层则使用service层的函数，这样编写的目的是降低耦合性，在检测bug的时候只需一层一层的检测。一开始你会觉得非常繁琐，但随着项目的不断变大，你会发现这样分3层来编写的好处。

关于Dao层

在springboot中，其实Dao层并不是直接操作的数据库的，这里引入了面向对象和映射的思想。我们通常新建一张表的同时也会为这张表编写一个实体类，意思就是数据库表的每一项都对应着实体类的一个元素，这样形成一一映射的关系。我们通过Dao层直接操作实体类，spring又通过映射关系来修改数据库里的表。

关于系统角色

- 项目最初是打算编写一个类似论坛的功能，一个人发出话题，其他人在下面发帖。但又考虑到这是一个考试信息共享的系统，大家来看的都是知识点，考点，而不会有太多的讨论，所以决定改成由想要分享考点的大佬来写文章，其他人看，并且可以在下面评论，点赞。
- 为了让其他学院的人也使用这个系统，决定增加版块的功能，也就是说一个学院是一个版块，你可以选择去特定的版块发帖，浏览帖子，也可以全部都看。
- 系统的角色分为，游客，普通用户，版块管理员，超级管理员这4种角色，相应的功能参考下图。



- 当用户没有登录时，当前用户处于游客身份，游客只能在index页面浏览相关文章，但是不能给文章点赞和评论
- 当用户注册并登陆以后，即可到达home页面，可以享有发帖，修改个人信息等功能。

关于home页面

- home页面采用的是一个app.css的主题，left,header,sidebar,footer都是主题框架。我们只需编写content（正文页面即可）。



如图所示，我们只需编写中间空白页面的代码即可。

项目构建思路就先写这些吧。