

关于该系统的后端设计的技术报告

在后端采用的技术是 mysql + django + django rest framework。

其中数据库用 mysql 来储存数据,但是具体的建表之类的操作,不需要用到 mysql,因为 django 自带的模型系统很强大,直接用类来表示,然后进行数据库迁入就 OK 了。

其中详细的版本是使用 mysql 5.6.35。之所以选用 mysql,因为 mysql 体积小、速度快、总体拥有成本低、免费,而且目前网站的构建方面是比较主流的。

其中详细的版本见官网：<https://www.mysql.com/>

mysql 的学习,可以见 mysql 官网的文档：<https://dev.mysql.com/doc/>但是在此项目中,由于 django 的模型系统,可以很方便的进行建设数据库,只需要提供与 mysql 的连接即可。

而选用 python 的 django 框架。因为在产品的多次迭代中,需要一个能进行快速开发的后端 web 框架,那 django 就是一个很好的选择,django 有着很好用的对象关系映射来创建模型,还有强大的后台管理,以及一流的 URL 设计,自带缓存系统以及友好的模板语言。是典型的 MVC 结构。但是在此项目中不需要用到 V,也就是视图,因为我们实行前后端分离,前端对视图进行渲染,往后台发送请求,然后后台取数据库的数据,返回前端去渲染。所以后端的工作主要是建数据库,编写相应的 API 以及设计相应的 URL。

而 django 的版本为当前最新版本 2.0,学习的文档为官网文档：

<https://docs.djangoproject.com/en/2.0/>

而因为使用了前后端分离,那么使用 restful 风格的框架就可以更方便。而这个框架的具体流程为：

1.建立 Models

2.依靠 Serializers 将数据库取出的数据 Parse 为 API 的数据(可用于返回给客户端 , 也可用于浏览器显示)

3.ViewSet 是一个 views 的集合 , 根据客户端的请求 (GET、POST 等) , 返回 Serializers 处理的数据权限 , Permissions 也在这一步做处理

4.ViewSet 可在 Routers 进行注册 , 注册后会显示在 Api Root 页上

5.在 urls 里注册 ViewSet 生成的 view , 指定监听的 url。