

软件设计文档

所使用的技术

一、AJAX:

AJAX 是一种创建交互式网页应用的网页开发技术，主要应用于课程页面的评论发表。之所以选用它是因为相比于传统的网页开发，它有一些明显的优点：

1、无刷新更新数据。

AJAX 最大优点就是能在不刷新整个页面的前提下与服务器通信维护数据。这使得 Web 应用程序更为迅捷地响应用户交互，并避免了在网络上发送那些没有改变的信息，减少用户等待时间，带来非常好的用户体验。

2、异步与服务器通信。

AJAX 使用异步方式与服务器通信，不需要打断用户的操作，具有更加迅速的响应能力。优化了 Browser 和 Server 之间的沟通，减少不必要的数据传输、时间及降低网络上数据流量。

3、前端和后端负载平衡。

AJAX 可以把以前一些服务器负担的工作转嫁到客户端，利用客户端闲置的能力来处理，减轻服务器和带宽的负担，节约空间和宽带租用成本。并且减轻服务器的负担，AJAX 的原则是“按需取数据”，可以最大程度的减少冗余请求和响应对服务器造成的负担，提升站点性能。

4、界面与应用分离。

AJAX 使 WEB 中的界面与应用分离（也可以说是数据与呈现分离），有利于分工合作、减少非技术人员对页面的修改造成的 WEB 应用程序错误、提高效率、也更加适用于现在的发布系统。

二、响应式布局框架

项目中使用的 flatUI 前端框架是基于 Bootstrap 开发的扁平化风格模板。项目中的前端页面模板均使用到了响应式布局，譬如导航栏，消息列表、课程信息等处都可以根据浏览器自适应地改变布局。Bootstrap 是最受欢迎的 HTML、CSS 和 JS 框架，用于开发响应式布局、移动设备优先的 WEB 项目，为所有开发者、所有应用场景而设计。Bootstrap 让前端开发更快速、简单。所有开发者都能快速上手、所有设备都可以适配、所有项目都适用。它的优点有：

1、预处理脚本

虽然可以直接使用 Bootstrap 提供的 CSS 样式表，但是 Bootstrap 的源码是基于最流行的 CSS 预处理脚本——Less 和 Sass 开发的。我们可以采用预编译的 CSS 文件快速开发，也可以从源码定制自己需要的样式。

2、一个框架、多种设备

我们的网站和应用能在 Bootstrap 的帮助下通过同一份代码快速、有效适配手机、平板、PC 设备，这一切都是 CSS Media Query 的功劳。

3、特性齐全

Bootstrap 提供了全面、美观的文档。你能在这里找到关于 HTML 元素、HTML 和 CSS 组件、jQuery 插件方面的所有详细文档。

三、Django MVC 架构框架

Django 大包大揽，用它来快速开发一些 Web 应用是不错的。在源码中，src/目录下的各个子模块都运用了 MVC 架构，其中 M 为 model.py，V 为 template，C 为 view.py 与 url.py。之所以选择 Django 是因为：

1、自助管理后台

admin interface 是 Django 里比较吸引眼球的一项 contrib，让你几乎不用写一行代码就拥有一个完整的后台管理界面。

2、自带 ORM

一般来说可以不怎么使用 SQL 语句，每条记录都是一个对象，而取对象的关联，易如反掌。

3、URL 设计

Django 的 URL 模块设计得看似很复杂，都是使用正则表达式，但真正用到的都是很简单的正则，当我用到一定程度的时候，发现这块东西，做的很细致，地址的表达上，我们可以随心所欲，那些优美的，简洁的，专业的地址，不管哪种，你都能表现出来。

4、App 设计理念

Django 的 App 理念很好。App 可插拔，是不可多得的思想。不需要了，可以直接删除，对系统影响不大。

5、Django 的错误提示

Django 的错误提示做的足够详细了，我有时喜欢上 Django 的出错页面了，那页面先不说有多详细，但光页面的美观上就是一种享受。有时，Django 的错误提示信息，让你马上就能知道在那个页面哪行代码出错了。

四、爬虫技术

在本应用中用到了爬虫来爬取学校的教务系统，src/api/目录下的 sysu.py 文件中使用到了爬虫技术。编程语言上，我们选择使用 Python 来进行爬虫的编写，原因有：

1、各种爬虫框架，方便高效的下载网页；

2、多线程、进程模型成熟稳定，爬虫是一个典型的多任务处理场景，请求页面时会有较长的延迟，总体来说更多的是等待。多线程或进程会更优化程序效率，提升整个系统下载和分析能力。

架构设计：

我们的 WEB App 采用 MVC 模式进行开发，MVC 全名是 Model View Controller，是模型(model)－视图(view)－控制器(controller)的缩写，一种软件设计典范，用一种业务逻辑、数据、界面显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。

在框架的选择上我们选择 Django 来实现 MVC 模式，在 Django 中，用户主要负责完成这几种文件：Model，Url，View，Template，Form 等，其中：

Model：主要用一个 Python 类来描述数据表。运用这个类，你可以通过简单的 Python 的代码来创建、检索、更新、删除 数据库中的记录而无需写一条又一条的 SQL 语句。

Url：指出了什么样的 URL 调用什么的视图。

View：包含并定义了页面的业务层逻辑。

Template：是 html 模板，它描述了这页面的设计是如何的。

Form：用 Python 类来描述表单。运用这个类可以让后台开发人员使用 Python 语言来定义表单。

在 Django 中，Model 文件和 Python 内置的 ORM 构成了 MVC 中的 Model，Url 文件和 Python 中的管理文件 manage.py 构成了 MVC 中的 Controller，View、Template、Form 等文件共同构成了 MVC 的 View。又因为在 Django 中，开发人员主要负责 Model 文件、View 文件和 Template 文件的编写，所以 Django 也称为 MTV 架构。

这种设计模式关键的优势在于各种组件都是松散结合的。这样，每个由 Django 驱动的 Web 应用都有着明确的目的，并且可独立更改而不影响到其它的部分。比如，开发者 更改一个应用程序中的 URL 而不用影响到这个程序底层的实现。设计师可以改变 HTML 页面的样式而不用接触 Python 代码。数据库管理员可以重新命名数据表并且只需更改一个地方，无需从一大堆文件中进行查找和替换。

模块划分

- 1、StaticPage: 静态页面，用于开发过程的参考及预期效果展示。
- 2、api: 教务系统爬虫模块，实现登陆教务系统的功能，并获取相关的信息。
- 3、base: 网页模板基础模块，包括其他模块页面的基础模板及登陆模块。
- 4、comment: 评论模块，实现了发布、查看评论的功能。
- 5、exchange: 换课信息模块，实现了换课信息发布、浏览的功能。
- 6、grade: 成绩模块，实现了成绩查询的功能。
- 7、lesson: 课程信息模块，实现了浏览课程信息的功能。
- 8、material: 资源分享模块，实现了发布、下载课程资源的功能。
- 9、school: 学校模块，拓展用，尚未进行深入开发。

引用的开源项目

1、Django: Django 是一个开放源代码的 Web 应用框架，由 Python 写成。采用了 MVC 的软件设计模式，即模型 M，视图 V 和控制器 C。它最初是被开发来用于管理劳伦斯出版集团旗下的一些以新闻内容为主的网站的。并于 2005 年 7 月在 BSD 许可证下发布。

2、FlatUI: Flat UI 是一个基于 Bootstrap 开发的一个响应式网页设计框架，包括了 HTML 及 PSD 版本。

3、sysu-jwxt-api: 由 dengyh 开发的 python 爬虫模块，能够登陆访问中山大学教务系统。