

基于Tornado的课程管理系统的设计与实现

刘 涛, 孙 茜, 易善凯

(上海理工大学, 上海 200093)

摘 要: 针对目前智能课堂的研究, 关于课程的管理系统就提出来了。因此, 开发了一种基于Tornado和React技术的课程管理系统。通过Tornado框架作为后端框架, 实现数据库中数据的增、删、改、查操作, Tornado与现在主流的Web服务器框架相比有着明显的优势: 它是非阻塞式服务器, 而且速度相当快, 这对于当下的Web开发来说至关重要。前端部分采用React技术将前台表单中输入的数据传输到后端数据库中, 并且对数据库中的数据进行相应的操作, React的一切都是组件化的, 使得代码更加的模块化, 重用代码将变得更加容易, 可维护性极高。该系统的开发过程还彻底的实现了前后端分离, 前后端的联系仅靠API联系起来, 前端部分的开发引用了Mock-api这一虚拟的服务端接口完成前端部分开发。这一开发技术路线充分展现了Tornado和React对Web开发的优势所在, 也可以提高这一系统的可维护性。

关键词: Tornado; React; Mock-api; 管理系统

中图分类号: TP315 **文献标识码:** A

Design and Implementation of the Curriculum Management System Based on Tornado

LIU Tao, SUN Xi, YI Shankai

(University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: In view of the current researches on intelligent classroom, the curriculum management system has been put forward. Therefore, a curriculum management system based on Tornado and React has been developed where Tornado framework is used as the back-end framework to realize the operation of adding, deleting and checking data in the database. Compared with the current mainstream Web server frameworks, Tornado has obvious advantages: it is a non-blocking server, and its speed is quite fast, which is very important for current Web development. The front-end part uses React technology to transfer the data input from the front-end forms to the back-end database, and carries on the corresponding operation of the data in the database. React is wholly component-based, which makes the codes more modularized, and makes the reuse of the codes easier and more maintainable. The development process of the system also completely realizes the separation of front and back ends. The connection between front and back ends only depends on API. The development of front-end part refers to Mock-api, a virtual server interface, to complete the separation of front and back ends. This development technology fully demonstrates the advantages of Tornado and React for Web development, and also improves the maintainability of this system.

Keywords: Tornado; React; Mock-api; management system

1 引言(Introduction)

如今, 互联网不仅大大方便了我们的日常生活, 还提升了人与人之间的沟通效率, 互联网发展到现在, 不得不与人工智能开始产生交集。人工智能平台大多从互联网企业转型而来, 因此可以说那互联网AI化就是人工智能。互联网的将来也一定是走向人工智能^[1]。随着教学改革的不定期进行, 各个高校对教育管理系统的自动化需求也越来越大。学校应该充分利用计算机所带来的技术来服务于高校, 随着最近几年AI技术的快速进步, 算法、算力和数据之间的进步, 高校对教育系统对系统智能化的需要也越来越高。基于Python的Tornado框架就很好的解决了C10K问题, 大多数社交网络会展示实时更新来提醒新消息、状态变化和用户通知, 这要求

客户端始终有一个打开的连接等待来自服务器的任何响应。这就使得Apache的最大线程池会迅速的饱和。一旦线程池的资源耗尽, 服务器将无法再响应新请求。异步服务器的应用也就提出来了, Tornado正是被设计来减轻基于线程的服务器的限制的, 也就是说, 如果当前请求正在等待来自其他资源的数据, 则异步服务器可以显式控制暂停请求^[2]。随着学校规模的扩大, 学生人数逐渐增加; 学校的课程也越来越多, 课程管理也就变得尤为重要, 因此智能课堂的完善需要这个课程管理系统来进一步实现。

2 Tornado技术介绍(Introduction of tornado technology)

Web开发人员可用的工具已经得到全面发展。随着技

术专家不断突破极限并使Web应用程序无处不在，我们还必须升级我们的工具并不断创建更好的框架以确保更好的应用程序。同时，我们也希望能够使用新工具，以便我们可以编写更清晰、可维护的代码，这些代码在部署到全球用户时具有高度可扩展性。Tornado框架也就提出来了，Tornado是一个编写易创建、扩展和部署的强大Web应用程序的绝佳选择。它是一个用Python编写的功能强大且可扩展的Web服务器。Tornado是FriendFeed网站的基础，于2009年9月10日发布，目前不仅得到了许多社区的支持，而且被应用于各种不同的场合。除FriendFeed外，还有许多公司在生产上转向Tornado，包括Quora、Turntable.fm、Bit.ly、Hiphunk和MyYearbook等。

当FriendFeed被FaceBook收购后得以开源。与那些只能达到10000个并发连接的传统Web服务器不同, Tornado在设计时充分考虑了性能, 该设计旨在解决C10K问题, 使其成为一个非常高性能的框架。此外, 它还具有处理安全性、用户身份验证、社交网络 and 外部服务(如数据库和用于异步交互的网站API)的工具^[2]。

3 系统设计(System design)

3.1 系统技术支持

该系统的后端基于Tornado技术开发，Tornado本身是一个功能强大且可扩展的Web服务器。不仅在处理高网络流量时表现的足够强大，而且在创建和编写时表现得足够的轻量级，还可以用于各种应用程序和工具。我们这里用Anaconda来运行基于python编写的Tornado程序。Anaconda工具不仅在写程序的时候对我们有很大帮助，在调式程序的时候也能起到很大的帮助，跟踪调试能够帮我们第一时间找出程序问题所在，可以根据报错信息，对代码就像修改。为了方便调试和使用，我们在测试的时候，服务器采用本机服务器，我们需要提前部署好服务器并指定端口号，也就http:localhost:8000,这里的8000指的是端口号，调试的时候端口可能会被占用，这里我们可以根据情况改成8001或者8002。前台采用React技术并通过最基础的CSS技术来对静态页面进行修饰。开发工具采用Sublime，能够对输入的React第一时间进行纠错。同时，为了提高数据库对数据的处理能力和可扩展性，数据库我们选择使用非关系型数据库:MongoDB数据库。

3.2 系统功能设计

课程管理系统整体结构图如图1所示。该系统由数据库模块, 后端模块和前端模块组成。

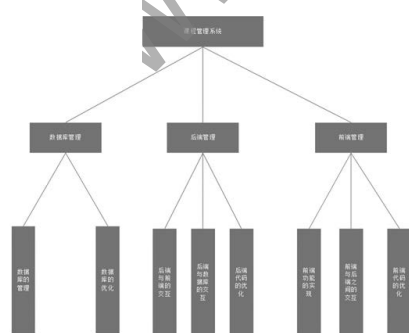


图1 系统框架

Fig.1 System framework

后端模块和前端模块的主要功能是增加、删除、更新和查询课程管理系统中的课程。基于Python的Toanodo框架，使后端代码足够简洁易懂，后端每完成一个方法的定义我们

把这个方法封装成一个API接口，然后把这个API接口提供给前端就可以了；前端部分，我们主要应用React技术，为了不去等待后端提供的API，影响开发效率，我们选择使用Mock-api来模拟后端API接口，首先我们在Mock-api中创建一个数据库，添加一条记录，我们通过React技术把这条技术显示到经过CSS修饰过的页面当中，然后我们在页面上已经创建好的输入行中输入相关课程信息，点击“Create Course”即可将前台输入的数据存入后代数据库中，并且在前台页面也能实时添加这条记录，查看数据库可以发现，我们插入数据的时候，数据库系统会自动为这条记录添加一个ObjectId字段，这是这条记录的主键，这个主键使唯一的，这个字段也是索引和进行修改删除操作的依据。每条数据的后面都有“Edit”和“Delete”按钮，点击“Edit”使数据变成可编辑状态，这个时候可以对数据库里的数据进行修改，修改完成之后单击“Update”以完成此数据的更新，此时后台就是根据前台提供的这个字段来进行索引任何更新数据。点击“Delete”也可以删除这条数据，根据前台传递的ObjectID来索引，找到这条记录然后删除。另外还可以根据任一个字段的信息来查找这条课程记录。课程列表页面如图2所示。

Course code

Course type

Course name

School department

Credit

Total hours

Computery

Semester

Description

Create Courses

Course code	Course type	Course name	School department	Credit	Total hours	Computery	Semester	Remarks	Description	Actions
20180601001	we			1	1	1	1510404376			Edit Delete
20180601002	typical	math	机械工程学院	3	1	1510404808	1	这是第一个主键不为空的课程。		Edit Delete
20180601003	专业必修课程	机械原理	机械工程学院	3	18	1510424263	1	这是机械专业必修的一门课。		Edit Delete
20180601004	专业选修课程	材料力学	机械工程学院	3	18	1510428061	1	材料力学第一、二两个学期的课程。		Edit Delete
20180602001	专业选修课程	理论力学	机械工程学院	4	72	6107	春季	学位课程	这是一个三方课程表	Edit Delete

图2 课程录入列表页面

Fig.2 Course entry list page

3.3 数据库设计

目前关系型数据库还是主流的数据库，比如Mysql。根据快速发展的网站规模的要求，关系型数据库需要满足三个高要求：需要高度并发读取和写入数据库，需要对大量数据进行高效存储访问，以及需要高度可扩展的数据库高可用性。为了满足这些需求，关系型数据库采用的技术有分区、分表和分库，但这需要修改业务逻辑层，这改变起来代价相当大。这个时候非关系型数据库的优势就开始体现出来了，非关系型数据库例如MongoDB就可以很好解决横向扩展问题。因为MongoDB中的Collection没有模式，所以不存在模式变更，由于Collection相互之间没有约束关系，也不能执行复杂的连接操作，所以MongoDB很容易支持分布式部署^[3-6]。该系统后台数据库部分使用MongoDB数据库，数据库名称为Course，Collection名称为courses。该数据库中有一个ObjectId字段是在插入数据的时候系统自动生成的，这个字段使唯一的。除了ObjectId，其余字段都可以为空。我们这里可以把任意一个字段作为索引依据，方便我们在搜索的时候可以依据每一个字段进行搜索，数据库中的数据如图3所示。

[illegible]

图3 数据库数据

Fig.3 Database data

3.4 系统安全设计

Web应用程序中访问控制机制中的漏洞,允许攻击者未经授权能够访问其无权访问的资源。Monshizadeh等人提出的MACR工具^[7]基于多条路径不一致的行为特征,能够自动地检测Web应用程序中的越权漏洞。Pellegrino等人^[8]提出了一种新型的黑盒测试的方法。通过从用户和Web应用程序交互的网络记录中识别Web软件的行为模式,再加上一些常见的攻击场景,生成测试样例。使用这种方法成功地在商用Web应用程序中发现多个位置漏洞。Weissbacher等人^[9]指出:浏览器中部分函数并不会受到同源策略(SOP)的限制,这就导致Web程序可能会因这些渠道而受到攻击。

在安全方面,Tornado Web服务器在设计初期就着重考虑到了,这也使得其能够更容易的防范那些常见的漏洞。安全cookies可以防止用户的本地状态被浏览器中的恶意代码暗中修改。此外,浏览器cookies可以与HTTP请求参数作比较来抵御跨站请求伪造攻击。事实上,有很多种方式可以拦截浏览器中的cookies。JavaScript和Flash对于它们所执行的页面的域有读写cookies的权限。浏览器插件还可以用编程方法访问这些数据,跨站脚本攻击可以有效利用这些访问从而修改访客浏览器中的cookies的值。Tornado的安全cookies使用加密签名的方式来验证cookies的值是否已经被服务器软件以外的任何人修改。由于任何恶意脚本都不知道正确的安全密钥,因此它不能在应用完全不知情的情况下去任意修改cookies的值。然而,Tornado的安全cookies仍然容易被窃听。恶意攻击者可以通过脚本或浏览器插件截获cookies的值,或者干脆直接窃听未加密的网络数据。记住cookies值是签名的而不是加密的。恶意程序可以读取已存储的cookies,并可以将数据传输到任何一个服务器,或者通过发送没有修改的数据给应用程序伪造一个请求。因此,避免将敏感的用户数据存储在互联网cookies中非常重要。我们还需要注意的是,用户也有可能需要去修改他自己的cookies,在这个过程中也会导致遭受攻击。例如,如果我们在cookies中存储了用户已付费的文章剩余的浏览数,我们希望阻止用户自己去更新有关的数值来获取免费的内容。HttpOnly和secure属性就可以有效帮助我们防范这种攻击。

3.5 系统流程

本系统以为智能课堂服务的目的,来设计满足各种功能要求,使课程信息得以实时的更新,同时还保证了系统内部数据的信息安全。首先我们利用React技术搭建好最基本的前台页面和实现与在Mock-api中建立的数据库的各种通信,对数据库中的数据进行添加、删除、修改和查询等操作。在前后台对接的时候,我们使用axios请求后台API数据,进而对数据库进行操作,tornado通过路由的形式把API提供给前台,前台在实现对数据库中数据进行操作的同时也对页面上显示的数据进行实时的更新。同时了为了保证系统的安全性,我们优化了前后台代码。从前台React基本功能的实现,到使用axios请求后台API数据,到后台Tornado处理前台传入的数据,成功的实现了React和Tornado前后台对接的课程管理系统。

4 系统测试(System test)

通过对该课程管理系统采用分步测试的方法,因为前后端使完全分离式进行开发的,我们在前台进行前台开发测试的时候,只需要满足对页面数据的更新和操作,而不需要

对数据库中的数据去定义各种方法,所以测试起来更加的方便和简介。后端部分定义好接口之后能够实现各个功能就够了,在测试过程中分别对前后台采取断点跟踪调试。对前台进行测试的时候,我们根据每一个功能采用断点跟踪调试的方法对从前台页面中输入的数据实时跟踪。在后台部分,我们从前台接收到数据然后与数据库之间进行交互也是一步步测试,通过多次测试,最终代码完全没有报错,能够成功实现前后台对接的功能。保证了系统的正常使用需求。

5 结论(Conclusion)

通过这个课程管理系统,使用Tornado和React框架满足了系统大部分需求;考虑到数据的安全性需求,使用Tornado作为后台代码并充分利用Tornado框架自带的一些安全机制,不仅保证了信息的安全性,而且代码还足够的轻量化。在高性能Web技术基础之上,摒弃了关系型数据库的种种约束,引进非关系型数据库MongoDB集群和相适应的开发模型以保证系统的高性能、高可用性和高安全性。React创建的用户交互界面相当友好且设计简洁,当数据发生变化时,React可以做到高效地更新渲染界面,使得数据能够实时的更新同步。当然对于MongoDB数据库在这里的应用也有不足之处,对于一些需要连接起来的数据表使用MongoDB就不行了,再就是Tornado常被用作大型站点的接口服务框架,而不能像Django那样着眼于建立完整的大型网站。关于根据任意字段进行搜索的功能也还有待进一步进行开发。页面的完整性和观赏性也有待提高。

参考文献(References)

- [1] 时宗胜.人工智能技术在移动互联网中的应用研究[J].数字通信世界,2017(10):164.
- [2] Michale Dory,Adam parrish,Brendam Berg.Introduction to Tornado[M].US, O'Reilly Media,2012.
- [3] 潘凡.从My SQL到Mongo DB—视觉中国的NoSQL之路[J].程序员,2010(6):78-80.
- [4] 卢冬海,何先波.浅析No SQL数据库[J].中国西部科技,2011(2):14-16.
- [5] 黄贤立.NoSQL非关系型数据库的发展及应用初探[J].福建电脑,2010(7):30-32.
- [6] 王光磊.MongoDB数据库的应用研究和方案优化[J].中国科技信息,2011(20):93-96.
- [7] Monshizadeh M,Naldurg P,Venkatakrishnan VN.MACE:Detecting privilege escalation vulnerabilities in Web applications.In:Proc.of the 2014 ACM SIGSAC Conf.on Computer and Communications Security.ACM Press,2014:690-701.
- [8] Pellegrino G,Balzarotti D.Toward black-box detection of logic flaws in Web applications.In:Bauer L,ed.Proc.of the 21st Annual Network and Distributed System Security Symp(NDSS 2014). San Diego:Internet Society,2014.
- [9] Weissbacher M,Robertson W,Kirda E,et al.Zigzag:Automatically hardening Web applications against client-side validation vulnerabilities.In:Jung J,ed.Proc.of the 24th USENIX Security Symp.San Antonio:USENIX Association,2015:737-752.

作者简介:

刘涛(1995—),男,硕士生.研究领域:智能制造,人工智能。
孙茜(1995—),男,硕士生.研究领域:智能制造,人工智能。
易善凯(1994—),男,硕士生.研究领域:智能制造,人工智能。