目录

[第1章 系统架构 2](#_Toc476139703)

[1.1 系统架构图 2](#_Toc476139704)

[1.2 系统架构说明 2](#_Toc476139705)

[1.2.1 负载均衡服务器Keepalived+Nginx 3](#_Toc476139706)

[1.2.2 应用服务器Tomcat 3](#_Toc476139707)

[1.2.3 缓存服务器Memcache 3](#_Toc476139708)

[1.2.4 缓存服务器Redis 4](#_Toc476139709)

[1.2.5 数据库服务器 4](#_Toc476139710)

[第2章 技术实现参数 4](#_Toc476139711)

[2.1 技术参数概述 4](#_Toc476139712)

[2.2 技术参数说明 5](#_Toc476139713)

[2.2.1 开发框架Spring 5](#_Toc476139714)

[2.2.2 MVC框架Spring MVC 5](#_Toc476139715)

[2.2.3 持久层框架Mybatis 5](#_Toc476139716)

[2.2.4 项目构建Maven 6](#_Toc476139717)

[2.2.5 代码管理Git 6](#_Toc476139718)

[2.2.6 权限管理Shiro 6](#_Toc476139719)

[2.2.7 页面布局Sitemesh 7](#_Toc476139720)

[2.2.8 CSS框架Less 7](#_Toc476139721)

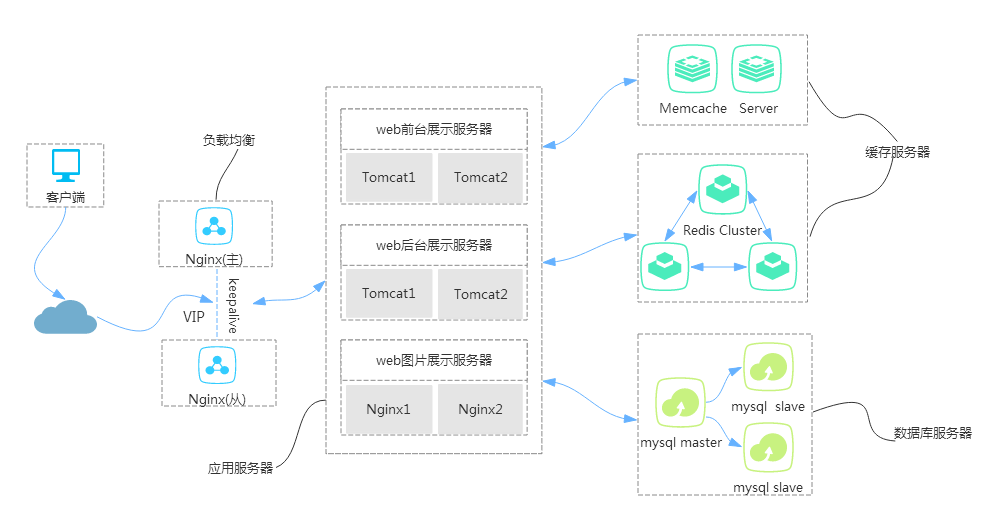
[第3章 系统功能架构 8](#_Toc476139722)

[3.1 设计原则 8](#_Toc476139723)

[3.2 功能架构图 9](#_Toc476139724)

# 系统架构

## 系统架构图

为了保证系统的高可用性，并且能够支持较高的并发，系统架构采用集群式的部署方案，能够根据使用人数的多少快速的对机器进行扩容，系统的部署架构图如下所示：

## 系统架构说明

各个软件的安装版本信息如下：

|  |  |  |
| --- | --- | --- |
| 业务模块 | 软件名称及版本 | 备注 |
| 负载均衡服务器 | keepalived-1.2.24 | 负载高可用 |
| nginx-1.9.8 | 负载均衡 |
| 应用服务器 | tomcat7.0.50 | Web应用服务器 |
| java1.7.0 | JDK |
| openoffice\_4.1.2 | 文件转换 |
| swftools-0.9.2 | 文件转换 |
| rsync-3.0.6 | 文件同步 |
| inotify-3.14 | 文件同步 |
| 缓存服务器 | memcache-1.4.33 |  |
| redis-3.2.4 cluster |  |
| 数据库服务器 | mysql-5.6.16 |  |

### 负载均衡服务器Keepalived+Nginx

负载均衡服务器使用Keepalived+Nginx实现负载的高可用性。

keepalived是一个类似于layer3, 4 & 7交换机制的软件，也就是我们平时说的第3层、第4层和第7层交换。Keepalived的作用是检测服务器的状态，如果有一台web服务器死机，或工作出现故障，Keepalived将检测到，并将有故障的服务器从系统中剔除，同时使用其他服务器代替该服务器的工作，当服务器工作正常后Keepalived自动将服务器加入到服务器群中，这些工作全部自动完成，不需要人工干涉，需要人工做的只是修复故障的服务器。

Nginx是一款轻量级的Web 服务器/反向代理服务器及电子邮件（IMAP/POP3）代理服务器，其特点是占有内存少，并发能力强，事实上nginx的并发能力确实在同类型的网页服务器中表现较好，国内知名大型互联网公司都在使用。

### 应用服务器Tomcat

应用服务器包括两种，一种部署系统前台和后台系统的服务器；另一种是文件服务器，存储一些资源文件和静态文件，这部分资源可以通过cdn加速提高页面的访问速度。web应用服务器采用Tomcat7.0，Jdk的版本采用1.7。同时服务器上部署了文件转换工具openoffice和swftools。

Tomcat是Apache 软件基金会（Apache Software Foundation）的Jakarta 项目中的一个核心项目，由Apache、Sun 和其他一些公司及个人共同开发而成。由于有了Sun 的参与和支持，最新的Servlet 和JSP 规范总是能在Tomcat 中得到体现。Tomcat 服务器是一个免费的开放源代码的Web 应用服务器，属于轻量级应用服务器。

### 缓存服务器Memcache

memcache是一套分布式的高速缓存系统，目前被许多网站使用以提升网站的访问速度，尤其对于一些大型的、需要频繁访问数据库的网站访问速度提升效果十分显著。

Memcache是一个高性能的分布式的内存对象缓存系统，通过在内存里维护一个统一的巨大的hash表，它能够用来存储各种格式的数据，包括图像、视频、文件以及数据库检索的结果等。简单的说就是将数据调用到内存中，然后从内存中读取，从而大大提高读取速度。

### 缓存服务器Redis

redis是一个key-value存储系统。和Memcached类似，它支持存储的value类型相对更多，包括string(字符串)、list(链表)、set(集合)、zset(sorted set --有序集合)和hash（哈希类型）。这些数据类型都支持push/pop、add/remove及取交集并集和差集及更丰富的操作，而且这些操作都是原子性的。在此基础上，redis支持各种不同方式的排序。与memcached一样，为了保证效率，数据都是缓存在内存中。区别的是redis会周期性的把更新的数据写入磁盘或者把修改操作写入追加的记录文件，并且在此基础上实现了master-slave(主从)同步。

Redis 是一个高性能的key-value数据库。 redis的出现，很大程度补偿了memcached这类key/value存储的不足，在部 分场合可以对关系数据库起到很好的补充作用。

### 数据库服务器

MySQL是一个关系型数据库管理系统，由瑞典MySQL AB 公司开发，目前属于 Oracle 旗下产品。MySQL 是最流行的关系型数据库管理系统之一，在 WEB 应用方面，MySQL是最好的 RDBMS (Relational Database Management System，关系数据库管理系统) 应用软件。

MySQL是一种关系数据库管理系统，关系数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。

MySQL所使用的 SQL 语言是用于访问数据库的最常用标准化语言。MySQL 软件采用了双授权政策，由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，一般大中小型网站的开发都选择MySQL作为网站数据库。

# 技术实现参数

## 技术参数概述

系统采用纯B/S架构并基于成熟稳定的开发平台，使用JAVA语言开发，采用Spring+Spring MVC+Mybatis的技术架构，使用MySQL数据库作为结构化的数据库。

|  |  |  |
| --- | --- | --- |
| 技术场景 | 技术选型 | 备注 |
| 开发语言 | Java |  |
| 开发框架 | Spring |  |
| MVC框架 | Sping MVC |  |
| 技术架构 | Sping+Sping MVC+Mybatis |  |
| 数据库 | Mysql |  |
| 项目构建 | Maven |  |
| 代码管理 | Git |  |
| 权限系统 | Shiro |  |
| 页面布局 | Sitemesh |  |
| CSS框架 | less |  |
| JavaScript框架 | Jquery |  |

## 技术参数说明

### 开发框架Spring

Spring是一个开源框架，Spring是于2003 年兴起的一个轻量级的Java 开发框架，由Rod Johnson创建。简单来说，Spring是一个分层的JavaSE/EEfull-stack(一站式) 轻量级开源框架。

Spring是全面的和模块化的。Spring有分层的体系结构，这意味着你能选择使用它孤立的任何部分，它的架构仍然是内在稳定的。Spring致力于J2EE应用的各层的解决方案，而不是仅仅专注于某一层的方案。可以说Spring是企业应用开发的“一站式”选择，并贯穿表现层、业务层及持久层。然而，Spring并不想取代那些已有的框架，而是与它们无缝地整合。

### MVC框架Spring MVC

Spring MVC属于SpringFrameWork的后续产品，已经融合在Spring Web Flow里面。Spring 框架提供了构建 Web 应用程序的全功能 MVC 模块。使用 Spring 可插入的 MVC 架构，从而在使用Spring进行WEB开发时，可以选择使用Spring的SpringMVC框架或集成其他MVC开发框架。它是一个典型的教科书式的mvc构架，而不像struts等都是变种或者不是完全基于mvc系统的框架。

### 持久层框架Mybatis

MyBatis 是支持普通 SQL查询，存储过程和高级映射的优秀持久层框架。MyBatis 消除了几乎所有的JDBC代码和参数的手工设置以及结果集的检索。MyBatis 使用简单的 XML或注解用于配置和原始映射，将接口和 Java 的POJOs（Plain Old Java Objects，普通的 Java对象）映射成数据库中的记录。

我们把Mybatis的功能架构分为三层：

(1)API接口层：提供给外部使用的接口API，开发人员通过这些本地API来操纵数据库。接口层一接收到调用请求就会调用数据处理层来完成具体的数据处理。

(2)数据处理层：负责具体的SQL查找、SQL解析、SQL执行和执行结果映射处理等。它主要的目的是根据调用的请求完成一次数据库操作。

(3)基础支撑层：负责最基础的功能支撑，包括连接管理、事务管理、配置加载和缓存处理，这些都是共用的东西，将他们抽取出来作为最基础的组件。为上层的数据处理层提供最基础的支撑。

### 项目构建Maven

Maven项目对象模型(POM)，可以通过一小段描述信息来管理项目的构建，报告和文档的软件项目管理工具。

Maven 除了以程序构建能力为特色之外，还提供高级项目管理工具。由于 Maven 的缺省构建规则有较高的可重用性，所以常常用两三行 Maven 构建脚本就可以构建简单的项目。由于 Maven 的面向项目的方法，许多 Apache Jakarta 项目发文时使用 Maven，而且公司项目采用 Maven 的比例在持续增长。

Maven是一个项目管理工具，它包含了一个项目对象模型 (Project Object Model)，一组标准集合，一个项目生命周期(Project Lifecycle)，一个依赖管理系统(Dependency Management System)，和用来运行定义在生命周期阶段(phase)中插件(plugin)目标(goal)的逻辑。

### 代码管理Git

Git是一款免费、开源的分布式版本控制系统，用于敏捷高效地处理任何或小或大的项目。分布式相比于集中式的最大区别在于开发者可以提交到本地，每个开发者通过克隆（git clone），在本地机器上拷贝一个完整的Git仓库。Git有很多优点，比如：适合分布式开发，强调个体；公共服务器压力和数据量都不会太大；速度快、灵活；任意两个开发者之间可以很容易的解决冲突；离线工作。

### 权限管理Shiro

Shiro 是一个用 Java 语言实现的框架，通过一个简单易用的 API 提供身份验证和授权。使用Shiro，您就能够为您的应用程序提供安全性而又无需从头编写所有代码。

这个解决方案提供了一个高度安全的集中式的身份验证基础设施，通过它的 API 或诸如 RADIUS 或 LDAP 这样的标准身份验证协议可很容易地与各种应用程序或 IT 基础设施组件集成。它支持广泛的软硬件或基于手机的身份验证令牌，在选择具体的身份验证方法方面提供了最大限度的灵活性。

由于 Shiro 提供具有诸多不同数据源的身份验证，以及 Enterprise Session Management，所以是实现单点登录（SSO）的理想之选 — 大型企业内的一个理想特性，因为在大型企业内，用户需要在一天内经常登录到并使用不同系统。这些数据源包括 JDBC、LDAP、 Kerberos 和 Microsoft® Active Directory® Directory Services (AD DS)。

Shiro的Session对象允许无需 HttpSession即可使用一个用户会话。通过使用一个通用的 Session 对象，即便该代码没有在一个Web 应用程序中运行，仍可以使用相同的代码。没有对应用服务器或 Web 应用服务器会话管理的依赖，您甚至可以在命令行环境中使用 Shiro。换言之，使用 Shiro 的 API 编写的代码让您可以构建连接到 LDAP 服务器的命令行应用程序并且与 web 应用程序内用来访问 LDAP 服务器的代码相同。

### 页面布局Sitemesh

SiteMesh是一个网页布局和修饰的框架，利用它可以将网页的内容和页面结构分离，以达到页面结构共享的目的。

Sitemesh是由一个基于Web页面布局、装饰及与现存Web应用整合的框架。它能帮助我们在由大量页面工程的项目中创建一致的页面布局和外观，如一致的导航条、一致的banner、一致的版权等。它不仅能处理动态的内容，如JSP、PHP、ASP、CGI等产生的内容，还能处理静态的内容，比如HTML的内容，使得它的内容也符合你的页面结构的要求。甚至它能像include那样将HTML文件作为一个面板的形式嵌入到别的文件中去。所有的这些，都是GOF的Decorator模式的最生动的实现。装饰模式是在不必改变原类文件和使用集成的情况下，动态地扩展一个对象的功能。它能通过创建一个包装对象，也就是装饰来包裹的对象。尽管它是由Java语言来实现的，但是它能与其他Web应用很好的集成。

### CSS框架Less

CSS（层叠样式表）是一门历史悠久的标记性语言，同 HTML 一道，被广泛应用于万维网（World Wide Web）中。HTML 主要负责文档结构的定义，CSS 负责文档表现形式或样式的定义。

作为一门标记性语言，CSS 的语法相对简单，对使用者的要求较低，但同时也带来一些问题：CSS 需要书写大量看似没有逻辑的代码，不方便维护及扩展，不利于复用，尤其对于非前端开发工程师来讲，往往会因为缺少 CSS 编写经验而很难写出组织良好且易于维护的 CSS 代码，造成这些困难的很大原因源于 CSS 是一门非程序式语言，没有变量、函数、SCOPE（作用域）等概念。LESS 为 Web 开发者带来了福音，它在 CSS 的语法基础之上，引入了变量，Mixin（混入），运算以及函数等功能，大大简化了 CSS 的编写，并且降低了 CSS 的维护成本，就像它的名称所说的那样，LESS 可以让我们用更少的代码做更多的事情。

### JavaScript框架Jquery

jQuery是一个快速、简洁的JavaScript框架，是继Prototype之后又一个优秀的JavaScript代码库（或JavaScript框架）。jQuery设计的宗旨是“write Less，Do More”，即倡导写更少的代码，做更多的事情。它封装JavaScript常用的功能代码，提供一种简便的JavaScript设计模式，优化HTML文档操作、事件处理、动画设计和Ajax交互。

jQuery的核心特性可以总结为：具有独特的链式语法和短小清晰的多功能接口；具有高效灵活的css选择器，并且可对CSS选择器进行扩展；拥有便捷的插件扩展机制和丰富的插件。jQuery兼容各种主流浏览器

# 系统功能架构

## 设计原则

在系统的设计和开发过程中，我们遵循如下原则：

1、统一设计原则

统筹规划、统一设计，保证整个系统结构的统一和数据的一致，是非常重要的，因此在系统设计过程中坚持统一设计原则，从全局出发、从长远考虑。

2、先进性原则

在保证实用、可靠的前提下，采用国际先进的技术架构和软件体系结构，构建本项目的系统，以适应将来业务发展需求，确保系统具有良好的拓展性。

3、标准化原则

严格遵循国家有关法律法规和技术规范的要求，从业务、技术、运维、管理等方面对项目的整体建设和实施进行设计，充分体现标准化和规范化。

4、可靠性原则

系统对可靠性有着很高的要求，一方面要强调整个系统高可靠性，不仅能够在正常情况下的稳定运行，而且保证在意外情况故障或重负载情况下系统的稳定性，另一方面要保证在业务高峰期间的系统响应能力。

5. 易用性原则

系统使用的设计要充分考虑使用者的个性化要求，具体体现易操作性。

6．安全性原则

在强化应用层面的安全性设计的同时，充分利用数据库等提供的各种安全机制，并合理利用各类专用安全产品的功能，为用户实现依照其业务规则制定的读、写、删、改的安全规则。

7、可扩展性原则

系统必须具有较好的可扩展性，能适应未来业务发展的需要，通过良好的架构设计和系统配置管理，保证系统的适应性和扩展性。

8、可维护性原则

系统必须具备良好的可维护性，能够方便快捷的对应用系统进行管理和维护，通过良好的技术架构设计，充分考虑应用软件的升级维护，做到客户端零维护。

## 功能架构图

该系统支持PC端和微信端的访问，不同的途径所访问的页面有所不同，都是和各自设备相对应的访问页面。整体系统包括数据库源、基础服务、分析检索平台、网校管理平台、Web网校平台和移动网校平台几部分组成，每一部分都发挥自己的功能。基础服务和各平台之间通过功能模块组合、数据转换抽取的方式进行衔接。

（1）在线学习系统的数据源包括结构化数据库、缓存数据Redis和Memcached、文件数据等几种形式。

（2）在线学习系统的基础服务包括录播服务、网校模板、课程服务、题库服务、视频服务、营销服务、支付服务、短信服务、邮件服务、学员服务、会员服务、问答服务等多种。

（3）在线学习系统包括分析检索平台、网校管理平台、Web网校平台、移动端网校平台等几种形式的方式平台，通过多种形式为用户提供服务。

下图为系统的总体架构图：

