**岩生网在线答题系统**

摘 要

岩生网在线答题系统是指通过一套完善的集做题、记录、讨论三个方面为一体开发的系统平台，实现用户在平台做题，做统计，以及用户做完题目之后讨论做题的心得的功能，为学习中的用户提供一个基于网络的数字化解决方案。用户在使用系统做题时，能够自动统计用户做题的对错情况，并能够在做完题目之后，和其他用户一起讨论做题的心得体会，提高学习效率，同时能够在做题的时候与其他人一起交流，讨论自己的不同见解，提高做题的效率以及练习的质量

系统基于Spring，MyBatis，SpringMvc框架结合MySql数据建库完成，同时使用了redis进行缓存管理，结合软件工程的开发理论，进行了需求分析、设计、实现和测试。论文首先详细分析实训基地信息管理系统的需求，明确系统所要实现的功能和系统逻辑，确定系统边界并通过系统的用例分析进行系统需求捕获；然后确定实现实训基地信息管理系统需要使用的关键技术，并对其详细阐述。在系统设计时，通过对用例的进一步分析，得出了系统的数据库设计和界面设计。在详细设计阶段，对系统的各个功能模块分别给出了详细的类图和序列图设计。在系统实现部分介绍了系统主要功能模块的实现细节及验证方法。最后运用黑盒测试技术，对系统 的功能和结构进行了测试。系统具有良好的可维护性与可扩展性。

关键词：在线答题系统； SSM；Redis；MySql

Online Answering System Of YanSheng Net

Pick to

Both online system is through a set of complete set, records, to solve the problem. Discuss three aspects of integrated development platform for the system, realize the user in the platform, to solve the problem to do statistics, and user after finish topic to discuss the function of the heart to solve the problem, to learn the user provides a web-based digital solutions.Users in the use of system, to solve the problem that automatically statistical user situation of right and wrong, to solve the problem and can after the finish, and other users to discuss the experience of to solve the problem, improve the learning efficiency, at the same time able to communicate in time with others to solve the problem, discuss their different opinions, to improve the efficiency and the quality of the practice to solve the problem

The system is based on Spring, MyBatis, SpringMvc framework and MySql data base. Meanwhile, redis is used for cache management. Combined with the development theory of software engineering, requirements analysis, design, implementation and testing are carried out.Firstly, the paper analyzes the requirements of the training base information management system in detail, clarifies the functions and logic of the system to be realized, determines the system boundary, and captures the system requirements through the use case analysis of the system.Then the key technologies needed to implement the training base information management system are determined and elaborated.In the system design, through the further analysis of use cases, the system database design and interface design.In the detailed design stage, the detailed class diagram and sequence diagram design are given for each functional module of the system.In the system implementation section, the implementation details and verification methods of the main functional modules of the system are introduced.At last, the function and structure of the system are tested by black box test technology.The system has good maintainability and expansibility.

**Keywords: online answering system:SSM:Redis:MySql**

第1章 前 言

* 1. 项目的背景和意义

随着5G时代的到来，我们已经进入了一个飞速发展的网络时代，传统的纸质考试等手段已经开始落后了，人们更希望能随时随地学知识，而不需要带着繁多的书籍，试卷；同时也不希望考试完成之后还需要等待阅等繁琐的过程。可以说，网络给信息处理、信息传输提供一个更快捷、跟安全的平台，作为练习，巩固知识点并给予网络答题系统设计符合用户的需求。因而建立基于网络的答题系统有很强的现实意义。通过在系统中不断的练习，巩固自己的知识点，为学习知识的学生等人群提供了系统的练习、学习知识的基于网络的途径，同时也能迎接5G时代的到来，让人们在生活中更快、更便捷 的服务。

互联网已经成为21世纪最重要的信息交互平台，能连接到网络的设备已经普通的计算机扩展到只能手机、平板电脑等多种设备上。这一切以为这一坨这些已经存在的设备和资源，我们的传统教育手段已经到了变革的这一刻。

目前出安通的练习手段以纸质试卷为主，这种手段存咋很多缺点：

1. 每道题都需要人工评分，无法自动评分，工作量很大
2. 无法从题库中随机选择试题组成试卷，出卷费时费力
3. 考试完成后，考试无法实现立刻看到成绩，时效性差
4. 考生只能在得到纸质试卷后才能看到自己的错题
5. 相对于纸质试卷，对于很多题目汇总困难

基于计算机的普及和网络的发展，我们需要一套新的考试平台来解决这个问题。奥瑞文在这个方向上开发了岩生网 产品来满足用户的需求。

岩生网在线答题系统是针对于学习的人群，随时随地能巩固，以及学习新知识

* 1. 研究目标及范围

伴随5G时代的到来，将练习获取知识的过程搬上网络；

网上答题系统能够让用户随时随地进行练习学习，学习新知识，同时能够多方面的组合学习，增强自己的能力；

在练习之后能够随即知道自己哪些不足，能够自己查看解析，并与其他联系者一起沟通，能够全方位的了解所学知识点；

并且在练习过程中会记录所有的统计记录，方便自己随时随地的了解自己的不足，以及自己的优势，通过练习错题来补充自己的不足；当然练习时临时需要处理其他事情的时候，平台也将提供“断电续作”的功能，方便随时随地的进行学习

* 1. 定义

|  |  |
| --- | --- |
| **缩写、术语及符号** | **解 释** |
| **SSM** | **即是指Spring、SpringMVC、Mybatis三个基本框架；Spring是一个整合框架，能够将所有的框架整合起来使用，统一管理；并且能够管理所有的对象的生命周期，实现解耦；提供AOP面向切面的变成方法以及思想，实现事物的控制，日志的输入等功能；Mybatis即是整合了jdbc连接数据库的框架，使用起来比jdbc方便很多，同时支持多对多的连接查询，批量的插入，动态sql等多种功能；** |
| **Mysql** | **传统的sql小型数据库，存储所有的数据，练习的统计、练习题目、各种账号等信息；能够使用索引等提高访问查询的速度，提高用户体验；** |
| **Redis** | **能够支持持久化的NoSql非关系型数据库，能够缓存一些用户查询遍历量大的数据，减轻Mysql数据库的压力，同时加快访问速度** |
| **Nosql（非关系型数据库）** | **不仅仅是sql即非关系型数据库；（这类型的数据不需要固定的模式，无需多余的操作就可以横向扩展）Redis，menmecacth** |
| **Sql（关系型数据库）** | **关系模型由关系数据结构、关系操作集合、关系完整性约束三部分组成** |

第2章 技术与原理

B/S结构是浏览器/服务器，直接通过浏览器即可为用户提供服务，将系统的功能实现集中在服务器上，简化了系统的开发、维护以及使用。系统使用了spring，mybatis以及mvc的设计模式，将模块间的耦合度降到最低—spring中使用IOC容器减少对依赖注入的耦合：mvc模式将controller，view，model三层分离，增加model层也就是数据访问dao以及service层的代码逻辑复用，降低耦合；使用了mysql以及索引的技术，结合redis缓存增加了访问的速度；使用了html5，jquery，bootstrap，ajax实现异步访问，局部刷新，使得再不刷新整个页面的情况下为用户提供服务，同时定时的ajax发送请求，是用户能够实时的接收到评论的消息

2.1 系统开发技术

2.1.1 SSM框架技术

SSM框架是Spring，SpringMVC，MyBatis三个框架的简称；通过三个框架整合，是系统的耦合度降到最低，同时能够方便快捷的对数据库进行访问，返回数据到页面。

Spring中的IOC以及AOP介绍：

loc 
Service 
Provider 

图2-1 IOC依赖注入

IOC，又名控制反转；别名#DI 依赖注入：即由spring ioc来控制对象的生命周期以及对象之间的关系；ioc 的代理 的简单实列: 中午肚子饿了，就想吃鸡腿，但是我又不想自己动手做，杀鸡，取鸡腿，再烹饪；我就点外卖，交给其他人来做这些，自己只需要吃就行，吃完餐盒一起丢了，可省的洗碗；也就是ioc的功能：通过ioc控制对象的生命周期，以及对象之间的依赖，我只需要告诉ioc我得需求而已；在类中直接new一个对象以及使用IOC依赖注入的区别：直接new：面向每一个环节，且使用完之后需要自己手动销毁，缺点：耦合度高；通过IOC容器创建无需管理对象的生命周期，降低耦合；AOP：在运行期间，动态的将某段代码切入到指定方法的指定位置运行的方式——能够实现增强代码，前置，后置，环绕，正常执行，异常的时候实现方法切入，不改变代码的情况下，对功能进行增强。AOP通过两种代理方式进行创建代理对象：cjlib以及java原生的动态代理方式进行代理；AOP的用途，能够实现日志功能，以及对mysql事务的管理；事务：一串的mysql命令对数据库数据进行修改时，一条命令报错，spring将会使他们都执行失败，保证数据的正常，减少脏数据的产生。

Spring框架中的其他模块如 图2-2Spring的所有模块；模块的简介以及功能描述：核心容器（Spring Core）核心容器提供Spring框架的基本功能。Spring以bean的方式组织和管理Java应用中的各个组件及其关系。Spring使用BeanFactory来产生和管理Bean，它是工厂模式的实现。BeanFactory使用控制反转(IoC)模式将应用的配置和依赖性规范与实际的应用程序代码分开。应用上下文（Spring Context）Spring上下文是一个配置文件，向Spring框架提供上下文信息。Spring上下文包括企业服务，如JNDI、EJB、电子邮件、国际化、校验和调度功能。Spring面向切面编程（Spring AOP）通过配置管理特性，Spring AOP 模块直接将面向方面的编程功能集成到了 Spring框架中。所以，可以很容易地使 Spring框架管理的任何对象支持 AOP。Spring AOP 模块为基于 Spring 的应用程序中的对象提供了事务管理服务。通过使用 Spring AOP，不用依赖 EJB 组件，就可以将声明性事务管理集成到应用程序中。JDBC和DAO模块（Spring DAO）JDBC、DAO的抽象层提供了有意义的异常层次结构，可用该结构来管理异常处理，和不同数据库供应商所抛出的错误信息。异常层次结构简化了错误处理，并且极大的降低了需要编写的代码数量，比如打开和关闭链接。对象实体映射（Spring ORM）Spring框架插入了若干个ORM框架，从而提供了ORM对象的关系工具，其中包括了Hibernate、JDO和 IBatis SQL Map等，所有这些都遵从Spring的通用事物和DAO异常层次结构。Web模块（Spring Web）Web上下文模块建立在应用程序上下文模块之上，为基于web的应用程序提供了上下文。所以Spring框架支持与Struts集成，web模块还简化了处理多部分请求以及将请求参数绑定到域对象的工作。MVC模块（Spring Web MVC）MVC框架是一个全功能的构建Web应用程序的MVC实现。通过策略接口，MVC框架变成为高度可配置的。MVC容纳了大量视图技术，其中包括JSP、POI等，模型来有JavaBean来构成，存放于m当中，而视图是一个街口，负责实现模型，控制器表示逻辑代码，由c的事情。Spring框架的功能可以用在任何J2EE服务器当中，大多数功能也适用于不受管理的环境。Spring的核心要点就是支持不绑定到特定J2EE服务的可重用业务和数据的访问的对象，毫无疑问这样的对象可以在不同的J2EE环境，独立应用程序和测试环境之间重用。

Spring Framework 
Data Access / Integration Web (MVC / Remoting) 
JDBC 
ORM 
WebSocket 
Web 
Transactions 
AOP 
Beans 
Aspects Instrumentation 
Core Container 
Co re 
Servlet 
Portlet 
Messaging 
SPEL 
Co n text 
Test   
图2-2 Spring所有模块

SpringMVC：实现三层分离，model，controller，view 也就是数据访问层，页面，以及连接三层实现分离，好处就是能够实现model层代码的复用，减少代码量，降低耦合，SpringMVC由前端控制器（DispatcherServlet）接受请求并响应到前端，并负责各组件职责的分配， 处理器 Handler 处理逻辑和业务请求，需要符合适配器的规则， 处理器映射器 HandlerMapping 根据url请求查找对应的处理器Handler，. 处理器适配器 HandlerAdapter 根据类型适配每个处理器并执行处理器 Handler，视图解析器 ViewResolver 根据逻辑视图的名称，将逻辑视图解析为视图对象，视图 view 是一个接口，根据不同的实现类支持不同的类型组成，整个MVC运行的流程图如下图2-3SpingMVC的发送请求以及相应请求：Dispatchersen.'let 
ViewResolver 
void render(Map<String, model, 
HttpServletRequest request, 
HttpSeMetResponse response) 
HandlerMapping 
HandlerExecutionChain 
3. Rn±ÆHandlerfiHandlerAdapter 
Model AndVlew 
HandleAdapter 
RequestMappingHandlerMapping 
Bean NameurlHandlerMapping 
Simple urlHandlerMapping 
WelcomePageHandlerMapping 
RequestMappingHandlerAdapter 
HttpRequestHandlerAdapter 
SimpleControllerHandlerAdapter 
Model AndVlew 
4. #Controller 
Handler 
HandlerMethod 

图2-3用户发送请求到响应的过程

MyBatis框架简介：MyBatis 是一款优秀的持久层框架，它支持定制化 SQL、存储过程以及高级映射。MyBatis 避免了几乎所有的 JDBC 代码和手动设置参数以及获取结果集。MyBatis 可以使用简单的 XML 或注解来配置和映射原生信息，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。

Mybatis的功能以及流程：如 图2-4 Mybatis各层的功能，以及流程；  
Sq1Session 
Executor 
StatementHand1er 
iæ-y-ä#, 
'i List 
MappedStatement 
BoundSq1 
Configuration 
esu1tMap 
State men Q g * 
ParameterHandler 
Result SetHandler 
ÄåJdbcTypeSjavaTvpe ;   
JDB< 
PreparedStatement 
ResultSet 
Statement 
SimpleStatement 
CallableStatement 

图2-4Mybatis各层的功能，以及流程

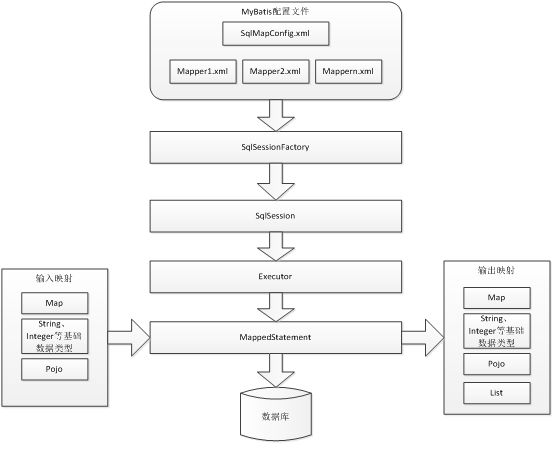
首先加载mybatis的配置文件，加载到内存中；通过Configuration实列维护，通过getConfiguration获取；通过配置文件创建sqlsession 对象，根据statement Id，来获取Configuration对象中对应的MapperStatement 对象，再调用执行器来执行； Executor；再执行查询时：通过sqlsession获取的MapperStatement对象，Executor对象根据传入的参数，完成对sql语句的动态解析，生成一个BoundSql对象，提供给StatementHandler使用；执行返回信息时，为查询创建缓存，提高性能；以及创建Jdbc的Statement链接对象，返回List查询结果；StatementHandler，执行查询时：调用ParameterHandler对象对sql中的占位符？进行赋值，然后通过TypeHandler，将参数转化为jdbcType类型，返回信息时 ： 对查询到的结果，通过TypeHander对jdbcType转化为javaType，再调用ResultHandler封装成相应的list或者java对象  


图2-5 mybatis架构图

Mybatis 缓存机制：一级缓存. 针对每一个sqlsession实现缓存；执行两次相同的查询时，即第二次使用缓存中的数据； 缓存使用哈希表实现，默认开启以及缓存，二级缓存：针对每一个namespace，实现二级缓存； 默认不开启，需手动配置，自带得缓存机制缺陷：针对每次 增删改 都会重新刷新整个缓存，而不是某一个单一得信息；

2.1.2 Redis以及NoSQL

Redis：使用Mybatis的自带缓存机制，有很多缺陷：每次增删改查都需要对所有的进行重新刷新，所以导致服务器压力大；使用Redis缓存机制能够很好的解决当前的问题，Redis是Nosql非关系型数据库，与传统的mysql等数据库关系型数据库相比有所不同。

Nosql简介：不仅仅是sql 即 非关系型数据库；（这类型的数据不需要固定的模式，无需多余的操作就可以横向扩展）Nosql的优势， 易扩展，大数据量高性能，多样灵活的数据模型（增加字段简单）

Nosql的三大特性：Consistence 强一致性 ：所有节点在同一时间具有，相同得数据；数据事物提交之后，所有查询都是一个结果， Availability 可用性可以快速获取数据，在确定的时间内返回操作结果，不管成功与否， Partition tolerance 分区容错性 系统的任意信息丢失或者错误，不会影响整个系统的正常运行。CAP 只能三选二：最多满足两个，分区容错性必须存在。

Redis简介：是以key-value形式存储，和传统的关系型数据库不一样，不一定遵循传统数据库的一些基本要求。优点：对数据高并发读写对海量数据的高效率存储和访问对数据的可扩展性和高可用行缺点：redis(ACID)处理非常简单无法做到太复杂的关系数据库模型

Redis 是以key-value store存储，data structure service 数据结构服务器，键可以包含：（string）字符串、哈希、链表(list)、（set）集合。（Zset）有序集合。这些数据结合都支持pysh/pop、add/remove

及取交集和并集以及更丰富的操作，redis支持各种不同方式排序，为了保证效率，数据丢失缓存在内存中，它也可以周期性的把更新数据写入磁盘后者把修改操作写入追加到文件。除了上述的5种数据结构，Redis还提供了许多额外的功能：提供了键过期功能，可以用来实现缓存提供了发布订阅功能，可以用来实现消息系统支持Lua脚本功能，可以利用Lua创造出新的redis命令提供简单的事务功能，能再一定程度上办证事务特性提供了流水线(pipeline)功能，这样客户端能将一批命令一次性传到redis，减少了网络的开销；Redis速度快的原因：redis把所有数据都是存放在内存中的redis是用C语言实现的，一般来说C语言实现的程序"距离"操作系统最近，执行速度相对更快Redis使用了单线程架构，预防了多线程可能产生的竞争问题Redis 的持久化：redis提供两种持久化方式：RDB和AOF，即可以用两种策略将内存的数据保存在硬盘中，这样就保证了数据的可持久性如 图2-6 Redis的持久化。

高可用： Redis从2.8 版本正式提供了高可用实现Redis Sentinel，它能够保证Redis节点的故障发现和 故障自动转移。Redis从3.0 版本正式提出了分布式Redis Cluster ,它是Redis真正的分布式实现，提供了高可用、读写和容量的扩展性。

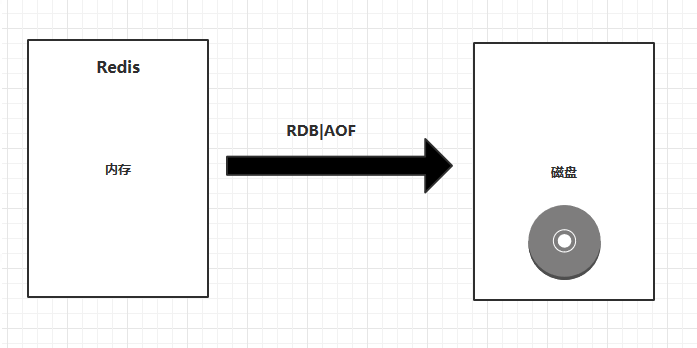


图2-6 Redis的持久化

2.1.3 ajax

为什么要使用ajax？传统的提交请求需要重新刷新整个页面，加重浏览器负担，如果服务器长时间未响应，浏览器也会无响应，用户体验差，提交表单，发送的数据大，可能造成网络问题，使用ajax可以很好的解决这方面的问题：局部刷新数据及页面 无响应时，页面还是显示原有网页 可以在后台与服务器进行少量的数据交互，实现网页的异步刷新 异步刷新 即无需等待服务器响应，发送完请求之后，还能对页面进行操作，浏览

2.1.3 MySQL

目前最流行的关系型数据库，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，分为社区版和商业版，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，一般中小型网站的开发都选择 MySQL 作为网站数据库。

MySQL存储引擎：MyISAM：不支持外键，不支持事务。可以存储null，访问速度快，对于事务没有要求，频繁insert select 的表可以使用这个引擎,支持三种不同的存储格式:压缩表:对每条数据进行压缩，所以只有非常小的访问开支；动态表：存储的字段大小不固定，优点是存储的空间占用小；但是容易产生碎片，需要定期优化；静态表：存储固定长度的内容表，占用的空间比动态表大，不满足长度会自动补足空格，读取时又去掉空格，所以可能存在空格丢失情况；Innodb：支持外键，支持事务，支持行锁，是mysql的默认存储引擎，相对于myisam引擎写能力差，并且会占用更多的磁盘空间保存索引以及数据，没有指定主键，innodb会自动生成一个主键；innodb和 myisam区别：myisam： 表级锁，支持全文索引，不支持事务，外键，保存表中的具体行数；适用于读多写少的场景innodb：行级锁，支持外键，事务，不直接保存表中的具体行数；memory：使用在内存中的数据建表，访问速度块，每个表对应磁盘中的。frm文件，但是当服务器关闭时，表中的数据即丢失，有两种索引 hash和 btre默认使用hash；hash优点：精确查找快，时间复杂度为 o（1），btree则需要遍历；缺点：不支持 不精确查找like等语句，主要用于数据不频繁发生变化的数据，或者存放中间操作数据的表；Merge：是一组myisam表的组合。所有myisam表的结构必须完全一样，本身数据库无数据，都是存在myisam的表中，数据的操作也是对于这些表进行

四大特性ACID 原子性，一致性，持久性，隔离性：原子性 atomicity：一个事物包含的操作要么全部成功，要么全部失败。一致性 consistency：事物必须使数据库从一个一致性状态到下一个一致性状态。事物提交之前，必须保持数据库的一致性状态；一致性读：只能从一个一致性状态中读取数据，不能从T1第一部分，从T2读一部分；一致性写：一个状态只能有一个事物提交数据，不能有两个；隔离性：将事物隔离开了，相互之间不干扰；例如同时有T1和T2两个并发事务，从T1角度来看，T2要不在T1执行之前就已经结束，要么在T1执行完成后才开始。将多个事务隔离开，每个事务都不能访问到其他事务操作过程中的状态。持久性：事物一单提交，对数据的修改是永久性

where 语句后边 少用 ！= 和 >、< ，会放弃使用索引，进行全表扫描；where 语句 or 少用，会放弃索引 可以用 [union 连接多个](onenote:#mysql数据库&section-id={1D601721-C829-42FA-A62B-1837F7E1BA44}&page-id={071B2A78-9D01-4BB3-9A91-AC5991C439AF}&object-id={42DF873D-DF1A-446E-AD24-FDF5A1235B8C}&E7&base-path=https://d.docs.live.net/f411899aa13cc679/科%20的笔记本/新学习.one) 和 union all 替换 or ；in // not in 都会导致全表扫描 如数据连续 in （1，2，3） 使用 between 1 and 3 替换；使用like 会导致全表扫描；所以可以使用 [全文索引：（只能在MyISAM引擎上才能使用）](onenote:#mysql数据库&section-id={1D601721-C829-42FA-A62B-1837F7E1BA44}&page-id={071B2A78-9D01-4BB3-9A91-AC5991C439AF}&object-id={DDE7D9E5-0FF4-4FED-9C88-67E7248E06AC}&53&base-path=https://d.docs.live.net/f411899aa13cc679/科%20的笔记本/新学习.one) 替换；使用表 变量 下面语句将进行全表扫描： select id from t where num = @num 可以改为强制查询使用索引： select id from t with (index(索引名)) where num = @num；不能再where = 号的左边使用 函数，算术运算符，或其他表达式，系统不会正确的使用索引；使用 exists 代替 in select \* form user where num in （select num form b） 代替 select \* from user where num exits （select num from b where [exits 和 in 的区别](onenote:#mysql数据库&section-id={1D601721-C829-42FA-A62B-1837F7E1BA44}&page-id={071B2A78-9D01-4BB3-9A91-AC5991C439AF}&object-id={9D7B3001-8D42-40BF-9FBD-0271D6A9AD31}&13&base-path=https://d.docs.live.net/f411899aa13cc679/科%20的笔记本/新学习.one)；exists 代替 distinct仅仅满足在 一对多 之间的关系；SELECT DISTINCT `uid` FROM `tb\_share`将仅有查询结果为1 的表放exists 之前， 当作外表，连接内表及没有重复结果；SELECT `uid` FROM `tb\_user` u WHERE EXISTS ( SELECT u.`uid` FROM `tb\_share` s WHERE u.`uid` = s.`uid`)；避免使用游标MySQL 数据查询优化：应当尽量避免再where条件中进行null判断 where name is null 而是使用一个默认值，来代替null 例如 0；where name = 0； 因为再where语句中使用null会使mysql放弃索引，而扫描全表（索引中不会包含null数剧）；索引并不是大部分时候都有效，对于拥有很多重复字段的列，可能会放弃使用索引 例如： sex 0 ，1 大量的重复数据；索引并不是越多越好，索引会提高相应的select效率，占用的物理内存空间大，而且 update， delete， insert 可能会重建索引； 一般索引不超过6个；建立的索引数据列尽量少进行update，insert，delete操作；能使用数字型字段，尽量使用数字型字段，若设计成字符型字段，再引擎处理查询和连接时，字符串会逐一一位一位得进行比较，而数字型字段及只要比较一次；尽量使用[varchar/nvarchar](onenote:#mysql数据库&section-id={1D601721-C829-42FA-A62B-1837F7E1BA44}&page-id={071B2A78-9D01-4BB3-9A91-AC5991C439AF}&object-id={18F9B792-34B2-4E3A-90F5-F7E7D17CB6AE}&2F&base-path=https://d.docs.live.net/f411899aa13cc679/科%20的笔记本/新学习.one) 替换 char nchar； 因为定长字符 char nchar（20） 再字符不满足二十时，自动补全英文半角空格，查询出来再去除，所以占用得大小会比varchar 大；尽量使用[临时表 和 表变量](onenote:#mysql数据库&section-id={1D601721-C829-42FA-A62B-1837F7E1BA44}&page-id={071B2A78-9D01-4BB3-9A91-AC5991C439AF}&object-id={42DF873D-DF1A-446E-AD24-FDF5A1235B8C}&15&base-path=https://d.docs.live.net/f411899aa13cc679/科%20的笔记本/新学习.one) 表变量来替换临时表，以减少系统资源的消耗；避免频繁创建，删除临时表，以减少系统表资源消耗；临时表并不是不可用，再需要重复的查询大型表或查用表的某个数据集时，可以使用临时表，select into 语句 替代 create table 可以减少大量的日志文件生成；如果数据量不大，想create table 再 insert

MySQL的逻辑结构如图：图2-7 MySQL的逻辑结构

一 800 +'•d • pod dHdEN • 0E00090 「 VO N 
SWISu3 06g 」 0 Otqe66nld 
! 0100uu00 
SVN •WS •S:IN 图2-7 MySQL的逻辑结构

2.2 系统开发工具介绍

### 2.2.1 Intellij IDEA

是用于java语言开发的集成环境（也可用于其他语言），IntelliJ在业界被公认为最好的java开发工具之一，尤其在智能代码助手、代码自动提示、重构、J2EE支持、Ant、JUnit、CVS整合、代码审查、 创新的GUI设计等方面...

### 2.2.2 Git

Git最大的开源代码托管工具，能够上传任何文件等数据，Git是一个免费的开源 分布式版本控制系统，旨在快速高效地处理从小型到大型项目的所有事务。

Git 易于学习， 占地面积小，具有闪电般快速的性能。它超越了Subversion，CVS，Perforce和ClearCase等SCM工具，具有廉价本地分支，便捷的临时区域和 多个工作流程等功能。

### 2.2.3 Maven

Maven是基于项目对象模型(POM project object model)，可以通过一小段描述信息（配置）来管理项目的构建，报告和文档的软件项目管理工具，Maven的核心功能便是合理叙述项目间的依赖关系，通俗点讲，就是通过pom.xml文件的配置获取jar包，而不用手动去添加jar包，而这里pom.xml文件对于学了一点maven的人来说，就有些熟悉了，怎么通过pom.xml的配置就可以获取到jar包呢？pom.xml配置文件从何而来？等等类似问题我们需要搞清楚，如果需要使用pom.xml来获取jar包，那么首先该项目就必须为maven项目，maven项目可以这样去想，就是在java项目和web项目的上面包裹了一层maven，本质上java项目还是java项目，web项目还是web项目，但是包裹了maven之后，就可以使用maven提供的一些功能了(通过pom.xml添加jar包)。

所以，根据上一段的描述，我们最终的目的就是学会如何在pom.xml中配置获取到我们想要的jar包，在此之前我们就必须了解如何创建maven项目，maven项目的结构是怎样，与普通java,web项目的区别在哪里，还有如何配置pom.xml获取到对应的jar包等等，这里提前了解一下我们如何通过pom.xml文件获取到想要的jar的，具体后面会详细讲解该配置文件。

第3章 需求建模

3.1 系统可行性分析

系统的开发往往有时间和资源的限制。因此，在项目启动之前，需要进行可行性分析，可以降低项目开发的风险，以免造成人力、物力、财力的浪费。下面从技术可行性、社会可行性、操作可行性三个方面来进行分析。

### 3.1.1 技术可行性

项目采用Spring+SpringMVC+MyBatis框架完成后台系统，加上MySQL数据库结合索引加快读取速度，同时结合Redis存入用户读取数据量频繁的数据，减轻数据库的压力，并且增加访问的速度；前端页面ajax使用多，异步刷新数据，减轻浏览器以及服务器的压力

### 3.1.2 社会可行性

1. 法律可行性：系统未触犯任何法律。
2. 人员可行性：系统的使用简单便捷，方便实时练习，查询自己的统计；为学习的学生提供了很好的平台

### 3.1.3 市场需求可行性

随着网络越来越发达，流量越来越便宜智能设备的普及人民获取信息的方式手段比以前更多，使用网络来获取知识也同时大受欢迎，能够随时随地的访问网站获取知识。并同时总结自己的不足，提供错题本

### 3.1.4 操作需求可行性

简单易用，智能设备普及以及网络的发达，系统能够达到简单快速上手的要求，方便广大用户进行学习

3.2 系统需求分析

《岩生网在线答题系统》的目的是能够方便用户随时随地的学习，并且省去自己统计，以及记录错题，并能够随时随地的和其他用户讨论题目的机会，让用户无时无刻的都能在学习中找到自己的不足，并且和别人一起共同的成长，让互联网成为用户获取知识的更有效的有段。无需纸质试卷，无需人工阅卷，无需自己统计记录错题，了解自己的不足，无需等到老师的讲解，练习完即可和一起答题的用户讨论题目，随时随地了解自己的不足。

随着5G时代的到来，网络以及智能设备越来越普及，基于网络的《岩生网在线答题系统》相比传统的纸质试卷能够更加方便，更加的帮助用户提升自己，以及学习新知识

3.2.1 系统流程图



图 3-1 系统总体流程图

如图 3-1 系统总体流程图 。用户进入网站，根据自己是否已有账号，来决定自己下一步：注册，或者登录；没有账号则新注册一个，有账号忘记密码则进入忘记密码流程，重置自己的密码，登录之后用户即可正常的使用平台。

进入平台用户即查看自己的资料，以及修改，查看自己的收藏题目，查看自己的错题，查看自己的做题统计，查看与自己相关的消息，做练习，以及断点续作。断电续作会自动记录上次做题的记录，下次继续学习的时候即可有选择的直接进入练习阶段；练习阶段：可以选择三种方式，错题练习，专项练习，随机练习，系统会自动出题，用户做完题目之后，能够评论题目，以及与各个用户讨论题目，并且查看自己的错题，解析。同时觉得题目的知识点是自己需要的时候，可以收藏起来，留着以后在看。



图 3-2 登录注册忘记密码流程图

用户第一次使用系统时，需要进行邮箱以及手机号的验证保证信息唯一且方便以后修改密码，同时能够只能够在登录之后才能使用系统，系统会根据用户来记录做题统计，以及错题统计等信息，所以用户使用系统时必须登录系统。登录流程图如图 3-2 登录忘记密码流程图



图 3-3 用户修改个人资料

如图 3-3 用户修改个人资料；用户在登录之后，系统会提供一个默认头像，需要用户自己修改用户头像，同时能够修改一些自己的信息，住址、电话号码、邮箱，同时会经过验证才能修改。



图3-4 用户练习

用户登录系统之后能够选择练习的方式，专项练习：顺序练习某一类题目；随机练习，能够选择多个分类，将根据分类的数量随机的输出题目的数量；错题练习：能够随机的练习错题，但是不会更新了正确率；选择好练习的方式，就能选择好题目的类型，java等会自动生成题目，用户在做题时，会记录到缓存中，在用户未做完题目时，能够再次进入系统时能够查询到缓存并提示用户续作，做完练习之后，系统会自动阅卷，返回答案以及解析给用户提醒大致解题逻辑，并且用户做完题目之后能够收藏题目，在之后能够点击收藏的题目继续查看题目知识点；做完题之后还能够对题目进行评论，回复，能够与其他人一起讨论题目的知识点，巩固自己的知识，加深记忆的印象。



图3-5 其他功能简略流程图

系统还提供了很多小功能，如图 3-5 其他功能简略流程图 能够查看自己的错题，能够查看自己收藏的题目，同时也能够查看自己的做题统计，了解自己的不足。

3.2.2 系统总体用例  


图3-6 系统总用例图

用例列表：

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | **名称** | **简要说明** | **参与者** |
| **001** | **注册** | **用户注册** | **用户** |
| **002** | **验证账号信息** | **用户注册、修改密码时认证是否是本人** | **用户** |
| **003** | **浏览个人信息** | **用户登录之后浏览自己的信息** | **用户** |
| **004** | **登录** | **用户登录** | **用户** |
| **005** | **浏览做题总统计** | **用户查看自己的做题总统计** | **用户** |
| **006** | **浏览分类统计** | **用户查看自己每个题目的分类统计** | **用户** |
| **007** | **浏览收藏的题目** | **用户查看自己收藏过的题目，回顾知识点** | **用户** |
| **008** | **浏览错题** | **用户查看自己做错过的题目，回顾知识点** | **用户** |
| **009** | **浏览消息** | **用户查看其他人回复自己的评论，或者回复自己的回复的消息** | **用户** |
| **010** | **查询收藏的题目** | **分页显示收藏的题目基本信息，展示信息** | **用户** |
| **011** | **选择练习方式** | **用户选择练习的方式，专项练习，随即练习，错题练习** | **用户** |
| **012** | **选择练习分类** | **用户选中练习的方式后，选择练习的题目类别** | **用户** |
| **013** | **开始练习** | **用户根据系统的出题，完成练习** | **用户** |
| **014** | **查看练习详情** | **用户完成练习之后，系统反馈做题详情对错** | **用户** |
| **015** | **查看评论** | **用户查看练习过的用户的评论，参与讨论** | **用户** |
| **016** | **查看回复** | **用户评论后，其他用户回复** | **用户** |
| **017** | **显示答题详情** | **用户完成练习后，显示对错** | **用户** |
| **018** | **收藏题目** | **用户做完一个题目之后，能够收藏题目** | **用户** |
| **019** | **取消收藏** | **用户收藏题目之后，能够在收藏接界面取消收藏** | **用户** |
| **020** | **删除评论** | **用户删除自己的评论** | **用户** |
| **021** | **删除回复** | **用户删除自己的回复** | **用户** |
| **022** | **修改头像** | **用户浏览自己的信息时，修改自己的头像** | **用户** |
| **023** | **删除头像** | **修改头像时，上传头像** | **用户** |
| **024** | **删除消息** | **用户浏览消息时，可以选择删除消息** | **用户** |
| **025** | **断点续作** | **用户未做完题目时，会将做题信息存入缓存中，下次进入系统则可以选择继续完成题目** | **用户** |
| **026** | **删除记录** | **用户存在断点续作的记录时，可选择删除记录，或者开始新的做题** | **用户** |

3.3 系统角色定义

### 3.3.1用户

|  |  |
| --- | --- |
| 角色（参与者） | 角色描述 |
| 用户 | 所有注册使用平台的用户 |

所有注册，并进入系统的用户即为系统的使用用户。进入系统必须要登录，并且用户注册时，需要进行邮箱验证，并且在之后修改密码等需要使用邮箱进行验证。用户进入系统时，默认显示首页，能够选择的查看自己收藏的题目、查看和自己相关的消息评论、查看自己的做题统计记录、选择做题的类型、个数、题目的类型、并且修改自己的个人信息，浏览自己的信息。如图3-6 系统总用例图

用户登录注册以及修改密码用例图



图3-7用户登录注册

注册

|  |  |
| --- | --- |
| 用例名称 | 注册 |
| 用例编号 | 001 |
| 行为角色 | 用户 |
| 简要说明 | 用户输入信息，进行注册 |
| 前置条件 | 首次进入系统，没有账号的用户 |
| 后置条件 | 验证信息符合规则，并且验证成功 |

输入账号信息

|  |  |
| --- | --- |
| 用例名称 | 输入账号信息 |
| 用例编号 | 002 |
| 行为角色 | 用户 |
| 简要说明 | 用户输入信息 |
| 前置条件 | 用户注册，登录，修改资料时 |
| 后置条件 | 无 |

登录

|  |  |
| --- | --- |
| 用例名称 | 登录 |
| 用例编号 | 003 |
| 行为角色 | 用户 |
| 简要说明 | 用户输入信息 |
| 前置条件 | 用户以及有账号 |
| 后置条件 | 无 |

修改密码

|  |  |
| --- | --- |
| 用例名称 | 修改密码 |
| 用例编号 | 004 |
| 行为角色 | 用户 |
| 简要说明 | 用户修改自己的密码 |
| 前置条件 | 用户忘记密码，或者主动修改密码的时候 |
| 后置条件 | 验证邮箱 |





图3-8 用户修改信息、练习

选择练习方式

|  |  |
| --- | --- |
| 用例名称 | 选择练习方式 |
| 用例编号 | 005 |
| 行为角色 | 用户 |
| 简要说明 | 用户开始练习时选择练习的方式：专项练习、随机练习、错题I练习 |
| 前置条件 | 用户登录之后，开始练习 |
| 后置条件 | 选择练习的方式以及题目的数量 |

开始练习

|  |  |
| --- | --- |
| 用例名称 | 开始练习 |
| 用例编号 | 006 |
| 行为角色 | 用户 |
| 简要说明 | 用户选择好题目数量，类型，做题方式时即可开始做题 |
| 前置条件 | 用户登录 |
| 后置条件 | 无 |

断点续作

|  |  |
| --- | --- |
| 用例名称 | 断点续作 |
| 用例编号 | 007 |
| 行为角色 | 用户 |
| 简要说明 | 用户在练习时，未交卷的情况下，会将做题数据记录，可以重新进入做题 |
| 前置条件 | 练习时未做完 |
| 后置条件 | 无 |

删除记录

|  |  |
| --- | --- |
| 用例名称 | 删除记录 |
| 用例编号 | 008 |
| 行为角色 | 用户 |
| 简要说明 | 存在断点续作的记录下， 用户可以选择删除记录，则可以开始一次练习 |
| 前置条件 | 存在做题记录 |
| 后置条件 | 无 |

修改头像

|  |  |
| --- | --- |
| 用例名称 | 修改头像 |
| 用例编号 | 009 |
| 行为角色 | 用户 |
| 简要说明 | 用户浏览个人信息时，点击修改头像即可 |
| 前置条件 | 浏览个人信息 |
| 后置条件 | 无 |

上传图片

|  |  |
| --- | --- |
| 用例名称 | 上传图片 |
| 用例编号 | 010 |
| 行为角色 | 用户 |
| 简要说明 | 点击修改头像的时候，上传图片修改头像 |
| 前置条件 | 点击修改头像 |
| 后置条件 | 无 |

查看练习详情



查看做题详情

|  |  |
| --- | --- |
| 用例名称 | 查看做题详情 |
| 用例编号 | 011 |
| 行为角色 | 用户 |
| 简要说明 | 用户做完练习点击交卷之后，系统返回用户的做题详情，对错，是否收藏等信息 |
| 前置条件 | 做完练习 |
| 后置条件 | 无 |

收藏题目

|  |  |
| --- | --- |
| 用例名称 | 收藏题目 |
| 用例编号 | 012 |
| 行为角色 | 用户 |
| 简要说明 | 用户做完练习点击交卷之后，可以选择是否收藏题目 |
| 前置条件 | 做完练习 |
| 后置条件 | 无 |

取消收藏

|  |  |
| --- | --- |
| 用例名称 | 取消收藏 |
| 用例编号 | 013 |
| 行为角色 | 用户 |
| 简要说明 | 用户做完练习点击交卷之后，收藏题目之后可以点击取消收藏 |
| 前置条件 | 做完练习，并且收藏题目之后 |
| 后置条件 | 无 |

评论题目

|  |  |
| --- | --- |
| 用例名称 | 评论题目 |
| 用例编号 | 014 |
| 行为角色 | 用户 |
| 简要说明 | 用户做完练习点击交卷之后，可以评论该题目，写出自己的解题看法 |
| 前置条件 | 做完练习 |
| 后置条件 | 无 |

删除评论

|  |  |
| --- | --- |
| 用例名称 | 删除评论 |
| 用例编号 | 015 |
| 行为角色 | 用户 |
| 简要说明 | 用户做完练习点击交卷之后，评论完之后可以选择删除评论 |
| 前置条件 | 评论之后 |
| 后置条件 | 无 |

回复用户评论

|  |  |
| --- | --- |
| 用例名称 | 回复用户评论 |
| 用例编号 | 016 |
| 行为角色 | 用户 |
| 简要说明 | 用户做完练习自后 |
| 前置条件 | 其他用户评论过题目 |
| 后置条件 | 无 |

删除回复

|  |  |
| --- | --- |
| 用例名称 | 删除回复 |
| 用例编号 | 016 |
| 行为角色 | 用户 |
| 简要说明 | 用户做完练习点击交卷之后，评论完之后可以选择删除回复 |
| 前置条件 | 回复了用户的评论之后 |
| 后置条件 | 无 |



浏览个人做题统计

|  |  |
| --- | --- |
| 用例名称 | 浏览个人做题统计 |
| 用例编号 | 017 |
| 行为角色 | 用户 |
| 简要说明 | 用户在登录之后可以查看自己的做题总统计，以及分类下的统计 |
| 前置条件 | 登录之后 |
| 后置条件 | 无 |

查看收藏的题目

|  |  |
| --- | --- |
| 用例名称 | 查看收藏的题目 |
| 用例编号 | 018 |
| 行为角色 | 用户 |
| 简要说明 | 用户在登录之后可以查看自己的收藏的题目，回顾自己不足的知识点 |
| 前置条件 | 登录之后 |
| 后置条件 | 无 |

删除收藏的题目

|  |  |
| --- | --- |
| 用例名称 | 删除收藏的题目 |
| 用例编号 | 019 |
| 行为角色 | 用户 |
| 简要说明 | 用户在登录之后可以删除自己的收藏的题目 |
| 前置条件 | 登录之后 |
| 后置条件 | 无 |

查看消息

|  |  |
| --- | --- |
| 用例名称 | 查看消息 |
| 用例编号 | 020 |
| 行为角色 | 用户 |
| 简要说明 | 用户在登录之后可以查看自己的消息 |
| 前置条件 | 登录之后 |
| 后置条件 | 无 |

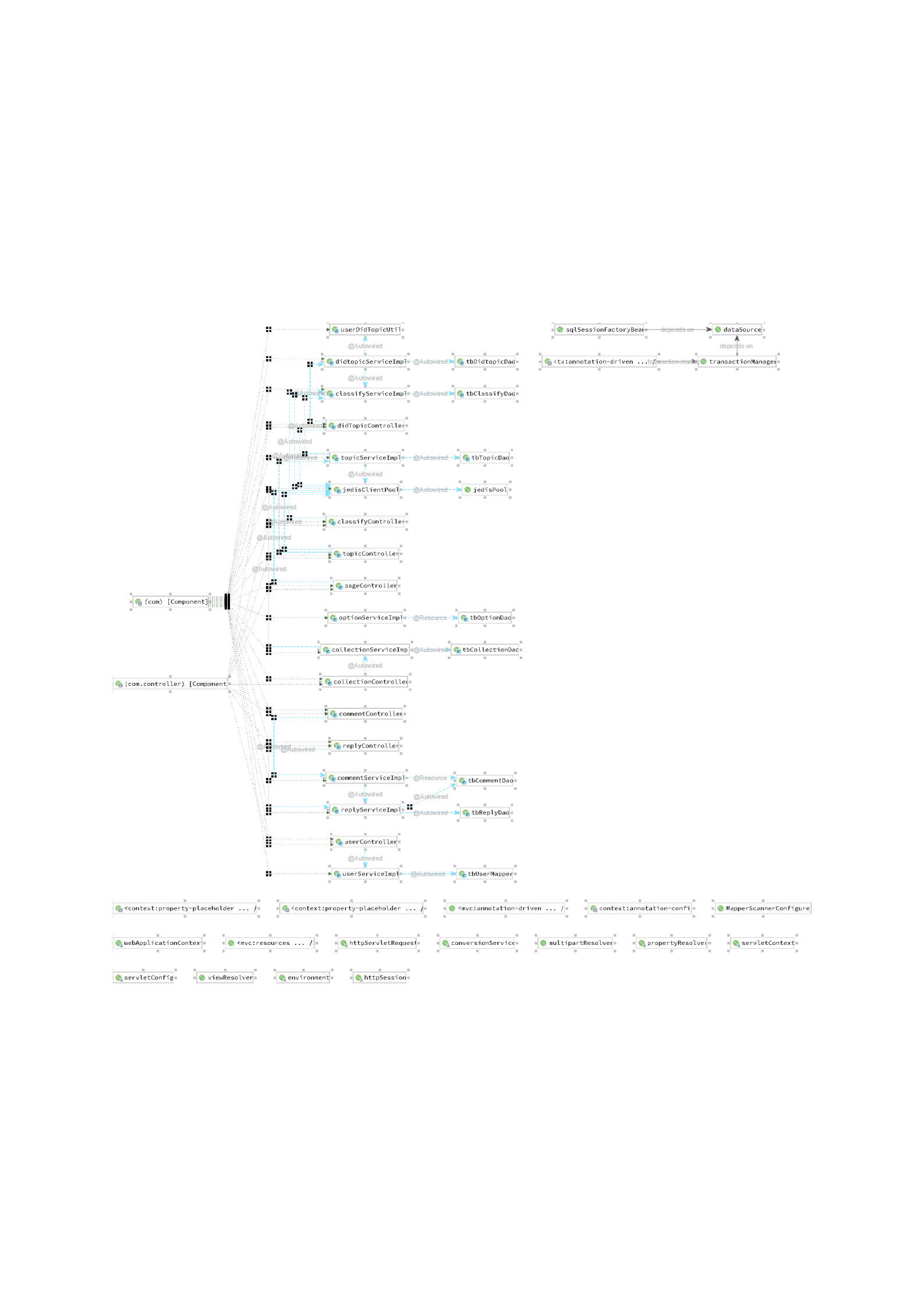
删除消息

|  |  |
| --- | --- |
| 用例名称 | 删除消息 |
| 用例编号 | 020 |
| 行为角色 | 用户 |
| 简要说明 | 用户在登录之后可以删除不想看到的消息 |
| 前置条件 | 登录之后 |
| 后置条件 | 无 |

查看详情

|  |  |
| --- | --- |
| 用例名称 | 查看详情 |
| 用例编号 | 021 |
| 行为角色 | 用户 |
| 简要说明 | 用户登录之后能够通过收藏的题目，以及消息进入题目，查看题目的详情 |
| 前置条件 | 登录之后 |
| 后置条件 | 无 |

3.4系统类图



3.5数据库er图

计算机生成了可选文字:
tbcomment
cc•mmentld
3℃03012
「eateOate
c肱032》
tb_city
、℃「《50
tb_reply
3℃03012
「epO'C「eateOate
「epO'Fath•e「黾
3「《30
C「《32》
tb
didtopic
tbuser
d巧TO卩℃
c「《32》
SS0
C「《32》
3「《30
e「「。「Op卜'Id
3011》
3℃03012
3℃03「崆到
tbcollection
3「《30
deleteFlag
tb_province
“℃「《50
option
0000nd
“℃「《50
tbtopic
to卩℃
“℃「《50
、℃「《50
tb_classiW
0“引f
03if'/N艹

### 3.5.1 数据表结构

tb\_city城市表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| cid | int(11) | NO | PRI | (NULL) | auto\_increment |  |
| name | varchar(50) | YES |  | (NULL) |  |  |
| pid | int(11) | YES | MUL | (NULL) |  |  |

Tb\_classify题目分类表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra | Comment |
| classifyId | int(4) | NO | PRI | (NULL) | auto\_increment | 分类id |
| classifyName | varchar(20) | NO |  | (NULL) |  |  |

tb\_collection 题目收藏表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| collectionId | int(4) | NO | PRI | (NULL) | auto\_increment |  |
| topicId | int(4) | NO | MUL | (NULL) |  |  |
| userId | char(32) | NO | MUL | (NULL) |  |  |
| deleteFlag | tinyint(1) | NO |  | 0 |  |  |

tb\_comment题目评论表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| commentId | int(4) | NO | PRI | (NULL) | auto\_increment |  |
| commentContent | varchar(100) | NO |  | (NULL) |  | 评论内容 |
| commentCreateDate | timestamp | NO |  | CURRENT\_TIMESTAMP | on update CURRENT\_TIMESTAMP |  |
| topicId | int(4) | NO | MUL | (NULL) |  |  |
| uid | char(32) | NO | MUL | (NULL) |  |  |

tb\_didTopic用户做题表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| didTopicId | int(4) | NO | PRI | (NULL) | auto\_increment |  |
| topicId | int(4) | NO | MUL | (NULL) |  |  |
| userId | char(32) | NO | MUL | (NULL) |  |  |
| error | tinyint(1) | YES |  | 1 |  |  |
| errorOptionId | int(4) | YES |  | -1 |  |  |

Tb\_option选项表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| optionId | int(4) | NO | PRI | (NULL) | auto\_increment |  |
| optionComment | varchar(50) | NO |  | (NULL) |  |  |
| topicId | int(4) | NO | MUL | (NULL) |  |  |
| correct | tinyint(1) | NO |  | 0 |  |  |

Tb\_province城市表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| pid | int(11) | NO | PRI | (NULL) | auto\_increment |  |
| name | varchar(50) | YES |  | (NULL) |  |  |

Tb\_reply评论回复表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| replyId | int(4) | NO | PRI | (NULL) | auto\_increment |  |
| replyContent | varchar(100) | NO |  | (NULL) |  |  |
| replyCreateDate | timestamp | NO |  | CURRENT\_TIMESTAMP | on update CURRENT\_TIMESTAMP |  |
| commentId | int(4) | NO | MUL | (NULL) |  |  |
| replyFatherId | int(4) | YES |  | -1 |  |  |
| toUid | char(32) | NO | MUL | (NULL) |  |  |
| topicId | int(4) | YES | MUL | (NULL) |  |  |
| uid | char(32) | YES | MUL | (NULL) |  |  |
| isRead | tinyint(1) | YES |  | 0 |  | 是否已读 0 未读， 1已读 |

Tb\_topic题目表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| topicId | int(4) | NO | PRI | (NULL) | auto\_increment |  |
| topicComment | varchar(50) | NO |  | (NULL) |  |  |
| classifyId | int(4) | NO | MUL | (NULL) |  |  |
| analysis | varchar(50) | YES |  | test |  |  |

Tb\_user用户表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| uid | char(32) | NO | PRI | (NULL) |  |  |
| uname | char(255) | NO | UNI | (NULL) |  |  |
| password | char(32) | NO |  | (NULL) |  |  |
| email | char(32) | NO | UNI | (NULL) |  |  |
| phone | char(11) | YES |  | (NULL) |  |  |
| sex | tinyint(4) | NO |  | 1 |  |  |
| VIP | tinyint(4) | NO |  | 0 |  |  |
| address | varchar(100) | YES |  | -1 |  |  |
| image | varchar(255) | NO |  | /static/img/userHeadImg/user.png |  |  |
| status | tinyint(1) | NO |  | 1 |  |  |

第4章 架构设计

4.1 系统技术架构

本系统采用的是B/S架构模式，B/S架构即为我们常说的[浏览器](http://baike.baidu.com/view/7718.htm)/[服务器](http://baike.baidu.com/view/899.htm)模式，是Browser/Server的简称，是目前最为流行的一种架构模式，它是依托浏览器，是客户端最常用的软件。使用B/S架构模式，统一了访问的客户端，让客户使用更加便捷，系统功能的核心部分都部署在服务器中，简化了开发者的维护与开发。客户只需要在本地计算机上安装浏览器，当系统需要升级时，客户只需要刷新浏览器页面即可获得最新的升级后的效果，服务器安装SQL Service、MySql、oracle等数据库即可以对数据进行存储。方便快捷的操作，使得B/S架构也越来越流行。



图4-1 B/S架构模型

4.2 系统功能结构图



4.3 具体功能模块

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **功能名称** | **功能需求标识** | **优先级** | **简要描述** |
| 1 | 用户注册 | S001 | 高 | 系统中申请人，校审批者的注册功能。 |
| 2 | 用户登录 | S002 | 高 | 用于已注册的用户进行登陆 |
| 3 | 做题总统计信息 | S003 | 高 | 用于统计用户的所有做题详情，并显示 |
| 4 | 做题分类统计信息 | S004 | 高 | 用于统计用户每个分类下的做题详情，并显示 |
| 5 | 显示收藏题目 | S005 | 高 | 用于显示用户收藏的题目 |
| 6 | 删除收藏 | S006 | 高 | 用于删除用户自己收藏的题目 |
| 7 | 浏览消息 | S007 | 高 | 用于浏览和用户相关的评论，回复消息 |
| 8 | 删除消息 | S008 | 高 | 用于用户删除自己不感兴趣的与自己相关的消息 |
| 9 | 练习 | S009 | 高 | 用户选择了做题的类型，分类，以及数量时进行练习 |
| 10 | 查看练习详情 | S010 | 高 | 用于做完题目之后，显示用户的做题详情 |
| 11 | 修改密码 | S011 | 高 | 用户忘记密码时，能够修改密码 |
| 12 | 修改个人信息 | S012 | 高 | 用户可以修改自己的头像、绑定的邮箱、绑定的手机号、居住地等 |
| 13 | 查看评论 | S013 | 高 | 用户查看自己的评论 |
| 14 | 收藏题目 | S014 | 高 | 用户收藏自己喜欢的题目 |
| 15 | 添加评论 | S015 | 高 | 用户对题目进行讨论 |
| 16 | 删除评论 | S016 | 高 | 用户删除自己的评论 |
| 17 | 回复评论 | S017 | 高 | 用户回复别人的评论 |
| 18 | 取消收藏 | S018 | 高 | 用户取消对题目的收藏 |
| 19 | 回复评论 | S019 | 高 | 用户回复别人的评论 |

4.3.1 S001 用户注册

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S001** | **功能名称** | **用户注册** | **所属角色** | **学生** |
| 功能描述 | 需要使用系统的用户可以进行注册。 | | | | |
| 输入项 | 1. 用户名 2. 密码 3. 确认密码 4. 性别 5. 联系电话 6. 邮箱 7. 验证码 | | | | |
| 处理描述 | 1. 用户不能为空，且唯一 2. 密码不能为空，密码需要通过效验 3. 重复密码必须与密码相同 4. 联系电话不能为空，并进行联系电话格式校验 5. 邮箱不能为空，邮箱要进行校验 6. 验证码必须正确 7. 密码存入数据库进行MD5加密 | | | | |
| 输出项 | 1. 若未按要求将需要填写的位置填写完毕，则输出哪些需要修改。 2. 若已经完成表单验证，则提示注册成功 | | | | |
| 界面要求 | 1. 要有输入项中显示的所有需要输入内容的地方 2. 所有错误提示用红色字体显示 | | | | |

4.3.2 S002 用户登录

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S002** | **功能名称** | **用户登录** | **所属角色** | **用户** |
| 功能描述 | 用户需要使用系统的时候，即需要进行登录 | | | | |
| 输入项 | 1. 账号 2. 密码 | | | | |
| 处理描述 | 1. 账号不能为空 2. 密码不能为空 3. 账号和密码必须匹配 4. 未在黑名单 | | | | |
| 输出项 | 1. 用户名为空时：提示用户名不能为空 2. 密码为空时：提示密码不能为空 3. 用户名在数据库中找不到时：提示用户名不存在 4. 用户名密码不匹配时：提示密码错误，请重新输入 | | | | |
| 界面要求 | 1. 要有输入项中显示的所有需要输入内容的地方 2. 所有错误提示用红色字体显示 | | | | |

4.3.3 S003 做题总统计信息

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S003** | **功能名称** | **用户登录** | **所属角色** | **用户** |
| 功能描述 | 用户查看自己在系统中做题的所有统计信息 | | | | |
| 输入项 | 无 | | | | |
| 处理描述 | 1. 第一次做题各项统计都为0 2. 用户做错题练习时，不会更新统计信息 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 能够显示所有的总统计信息 | | | | |

4.3.4 S004 分类题目统计信息

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S004** | **功能名称** | **用户登录** | **所属角色** | **用户** |
| 功能描述 | 用户查看自己在系统中各个分类的统计信息 | | | | |
| 输入项 | 无 | | | | |
| 处理描述 | 1. 第一次做题各项统计都为0 2. 用户做错题练习时，不会更新统计信息 3. 会按照题目的类别进行统计展示 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 能够显示所有分类的总统计信息 | | | | |

4.3.5 S005 显示收藏的题目

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S005** | **功能名称** | **显示收藏的题目** | **所属角色** | **用户** |
| 功能描述 | 用户查看自己的收藏的题目 | | | | |
| 输入项 | 无 | | | | |
| 处理描述 | 1. 展示用户收藏的题目信息 2. 能够点击进入题目详情 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 展示题目的名称，题目的类别 2. 能够点击题目名称进入题目的详情界面 3. 若无收藏数据，即显示暂无数据 4. 分页展示数据量   点击加载更多即加载其他分页 | | | | |

4.3.6 S006 删除收藏

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S006** | **功能名称** | **删除收藏** | **所属角色** | **用户** |
| 功能描述 | 用户查看自己的收藏的题目时，能够删除自己的收藏信息 | | | | |
| 输入项 | 无 | | | | |
| 处理描述 | 1. 在展示用户收藏的信息界面上 2. 点击取消收藏即可删除收藏信息 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 点击取消收藏，即删除收藏信息 2. 若无收藏数据，即显示暂无数据 3. 分页展示数据量 4. 点击加载更多即加载其他分页 | | | | |

4.3.7 S007 **浏览消息**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S007** | **功能名称** | **浏览消息** | **所属角色** | **用户** |
| 功能描述 | 用户能够查看与自己相关的评论、回复的信息，并且能在主页上显示数字提示 | | | | |
| 输入项 | 无 | | | | |
| 处理描述 | 1. 每间隔三十秒获取一次数据，刷新数量提示 2. 点击消息即进入消息的详情页面，显示题目详情 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 能够及时的显示未读消息，提醒用户 2. 点击消息详情能够进入题目的详情，显示评论 3. 显示删除按钮 4. 若无收藏数据，即显示暂无数据 5. 分页展示数据量 6. 点击加载更多即加载其他分页 | | | | |

4.3.8 S008 **删除消息**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S008** | **功能名称** | **删除消息** | **所属角色** | **用户** |
| 功能描述 | 用户浏览自己的消息时，能够点击删除，删除掉自己不想看到的消息 | | | | |
| 输入项 | 无 | | | | |
| 处理描述 | 1. 点击删除之后，系统会更新数据表中的字段，并不是删除数据 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 能够及时的显示未读消息，提醒用户 2. 点击消息详情能够进入题目的详情，显示评论 3. 显示删除按钮 | | | | |

4.3.9 S009 **练习**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S009** | **功能名称** | **练习** | **所属角色** | **用户** |
| 功能描述 | 用户选择了练习的方式，练习题目的类别，练习题目的数量，系统自动出题，给用户做题 | | | | |
| 输入项 | * + - 1. 做题的方式       2. 做题的数量       3. 做题的类别       4. 题目的答案 | | | | |
| 处理描述 | 1. 专项练习，只能选中一个类别；随机练习能够选择多个类别的题目，并且随机选择类别中的题目；错题练习能够选择多个类别的题目进行错题练习；错题练习不会更新正确率等统计信息；用户开始做练习时，每次只能展示一个题目，分页展示其他题目 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 每次只展示一道题的信息； 2. 分页展示所有题目信息 3. 用户选择完答案之后保存到缓存中 | | | | |

4.3.10 S010 查看**练习详情**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S010** | **功能名称** | 查看**练习详情** | **所属角色** | **用户** |
| 功能描述 | 用户做完练习之后点击交卷，显示题目的内容，选项，答案，对错，解析评论，回复，收藏等信息 | | | | |
| 输入项 | 无 | | | | |
| 处理描述 | 1. 用户做完题目，并交卷 2. 显示题目的对错信息，对则显示绿色，错则显示红色 3. 显示收藏信息，收藏即显示取消收藏，没有收藏即显示收藏按钮 4. 显示评论，回复信息 5. 有评论即显示评论信息，并显示评论下的回复信息数量 6. 有数量即显示数量并且显示点击查看详情，没有数量即显示暂无评论 7. 自己的评论显示删除按钮，不显示回复按钮 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 每次只展示一道题的信息； 2. 分页展示所有题目信息 3. 用户选择完答案之后保存到缓存中 | | | | |

4.3.11 S011 修改密码

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S011** | **功能名称** | 修改密码 | **所属角色** | **用户** |
| 功能描述 | 用户忘记密码之后，能够修改密码，但是需要先验证邮箱 | | | | |
| 输入项 | 邮箱验证码，密码，确认密码， | | | | |
| 处理描述 | 1. 用户做完题目之后，提示 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 每次只展示一道题的信息； 2. 分页展示所有题目信息 3. 用户选择完答案之后保存到缓存中 | | | | |

4.3.12 S012 修改个人信息

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S012** | **功能名称** | 修改个人信息 | **所属角色** | **用户** |
| 功能描述 | 用户查看自己的个人信息时，可以选择修改个人信息，绑定邮箱，绑定电话号码，居住地等信息，头像 | | | | |
| 输入项 | 上传头像图片  输入新邮箱  验证码  新手机号  新居住地 | | | | |
| 处理描述 | 1. 用户上传图片之后，出现上传头像的框，能够提示用户选择、缩放图片 2. 用户修改邮箱时需要验证邮箱，确认邮箱的验证码 3. 修改手机号，居住地址 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 头像图片上传能够缩放并选择部分图片 2. 邮箱，手机号需要符合验证规则 3. 居住地能够自己选择 | | | | |

4.3.13 S013查看评论

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S013** | **功能名称** | **查看评论** | **所属角色** | **用户** |
| 功能描述 | 用户完成题目之后，能够查询到题目的评论 | | | | |
| 输入项 | 五 | | | | |
| 处理描述 | 1. 用户交卷之后，显示题目的对错，并且显示题目的评论信息 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 显示评论信息，并且能够显示评论的回复数量 2. 没有数量即显示暂无回复评论，有评论显示数量 | | | | |

4.3.14 S014收藏题目

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S014** | **功能名称** | 收藏题目 | **所属角色** | **用户** |
| 功能描述 | 用户在查看题目详情界面能够点击题目进行收藏 | | | | |
| 输入项 | 五 | | | | |
| 处理描述 | 1. 题目详情页面用户点击收藏，即收藏题目 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 题目默认为不收藏 2. 没有收藏即形式为未收藏按钮 3. 收藏了即显示取消收藏按钮 | | | | |

4.3.15 S015添加评论

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S015** | **功能名称** | 添加评论 | **所属角色** | **用户** |
| 功能描述 | 用户查看题目详情的时候，能够对题目新增评论，输入信息后点击评论按钮即可对题目评论 | | | | |
| 输入项 | 评论消息 | | | | |
| 处理描述 | 1. 评论框默认显示，用户在输入框中输入自己的评论信息时，点击评论时，即发送请求，添加评论信息 2. 评论完毕之后显示自己的平路信息，并且自己仅仅能够删除，不能回复 | | | | |
| 输出项 | 1. 显示评论的信息 | | | | |
| 界面要求 | 1. 评论框默认显示，输入消息后，点击评论能够插入到评论框中 2. 点击评论之后，底部菜单栏默认显示自己新添加的评论 3. 自己的评论仅仅能够删除，不能够自己回复自己 | | | | |

4.3.16 S016删除评论

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S016** | **功能名称** | 删除评论 | **所属角色** | **用户** |
| 功能描述 | 用户能够删除自己的评论消息 | | | | |
| 输入项 | 无 | | | | |
| 处理描述 | 1. 自己的评论能够删除，并且连带删除评论之下的所有回复 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 用户点击删除能删除自己的评论 2. 删除自己的评论即需要连带删除评论之下的所有的回复 | | | | |

4.3.17 S017回复评论

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S017** | **功能名称** | 回复评论 | **所属角色** | **用户** |
| 功能描述 | 用户能够对其他用户的评论进行回复 | | | | |
| 输入项 | 回复的信息 | | | | |
| 处理描述 | 1. 用户仅仅能够回复别人的评论 2. 自己的评论不能回复 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 点击回复弹出输入框，输入回复之后，点击评论能够回复用户 2. 回复之后自动收起输入框，输出评论信息 | | | | |

4.3.18 S018取消收藏

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S018** | **功能名称** | 取消收藏 | **所属角色** | **用户** |
| 功能描述 | 用户能够取消收藏的题目 | | | | |
| 输入项 | 无 | | | | |
| 处理描述 | 1. 点击取消，发送请求取消收藏 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 在查看详情页面点击取消即可以取消收藏 2. 收藏时显示取消收藏按钮 3. 取消收藏时显示收藏按钮 | | | | |

4.3.19 S019回复回复

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **功能编号** | **S019** | **功能名称** | 回复回复 | **所属角色** | **用户** |
| 功能描述 | 用户回复评论之后叫做回复，回复回复就是回复评论下的回复 | | | | |
| 输入项 | 无 | | | | |
| 处理描述 | 1. 点击回复能够回别人的回复 | | | | |
| 输出项 | 1. 无 | | | | |
| 界面要求 | 1. 点击回复能够输入回复内容 2. 点击回复收起输入框 3. 显示回复信息，回复的用户以及回复的内容 | | | | |

第五章 模块设计

5.1 系统设计

系统主旨在于提供用户随时随地的学习机会和其他学霸随时随地的讨论的机会、方便用户查看自己的不足，同时能够方便用户查看错题，而不需要自己手动记录，同时能够省去很多人工的阅卷成本。

系统的功能环环相扣，练习模块，做题详情展示模块，收藏模块，评论回复模块，做题统计模块缺一不可，能够为用户提供一系列的功能，方便用户的学习以及减少时间的成本

5.2 登陆注册模块

5.2.1 模块描述

用户使用系统的入口，所有使用平台的用户必须要注册，才能方便用户的做题统计信息，收藏，评论，错题等功能的使用

5.2.2 界面设计

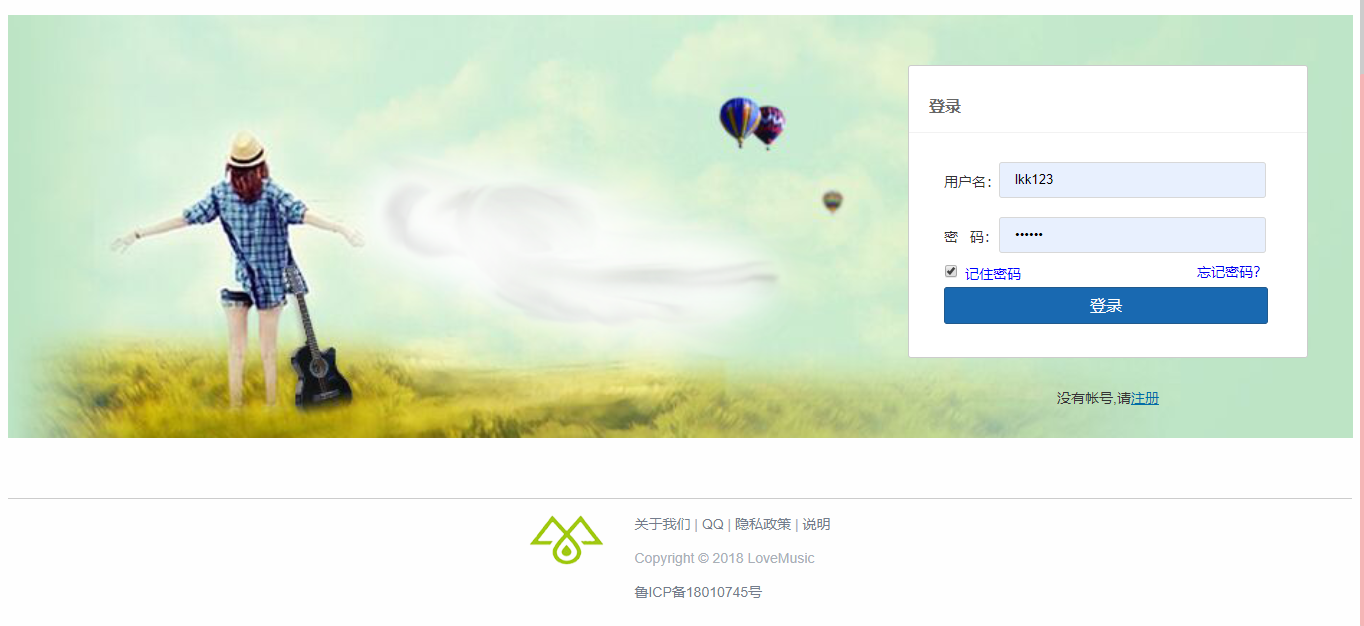


图5-1 登录界面



图5-2 注册界面第一步

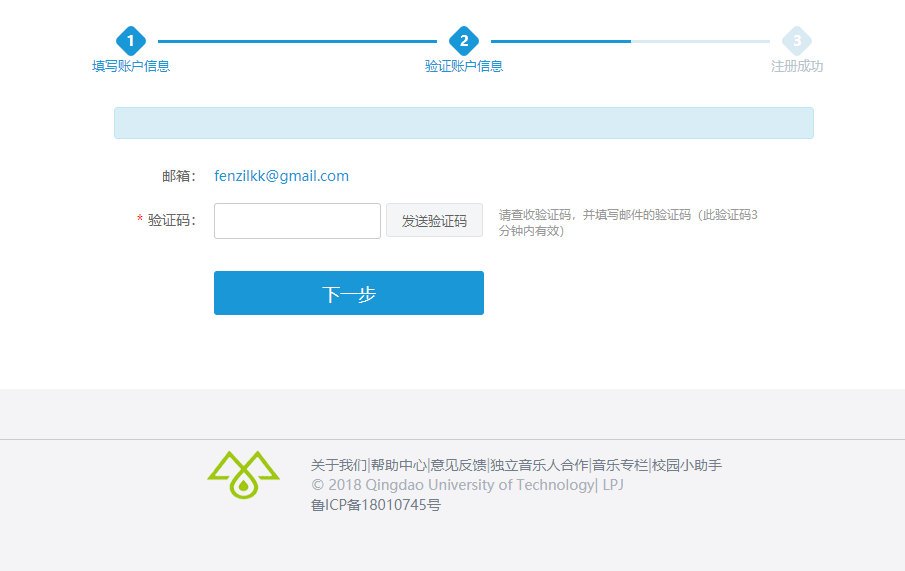


图5-3 注册界面第二步



图5-4 注册完成界面

5.2.3 程序逻辑

登录界面，用户输入账号密码，后台会进行md5加密并进行验证，账号不存在，密码错误时提示用户“账号或者密码错误”，同时会进行黑名单判断，在黑名单的用户将无法登录，并且提示“账号异常，请联系管理员”。用户首次使用系统，没有账号的时候，通过注册进行账号的申请。

注册分为三步：第一步用户输入账号名，账号密码，账号绑定的邮箱，以及图片验证码，并且阅读条款；输入的账号名必须使用英文字母开头，并且3到20位大小，并且和已经注册过的用户进行校验，保证用户名不能重复，重复了提示用户“用户名已经使用”；输入的邮箱也将保证和现有注册用户不重复，并且邮箱的格式正确；密码必须是数字字符的组合，不然提示用户密码安装等级太低，正确的输入了图片验证码的验证码之后，勾选条款，点击下一步进入，第二步邮箱验证的界面；第二步邮箱验证，使用用户第一步输入的邮箱，发送验证码到邮箱上，用户输入验证码后进行效验，验证码错误即提示用户验证码错误，此步骤是为了检验邮箱是本人邮箱；邮箱验证完毕之后进入第三部，输入手机号码，验证手机号规则，同时完成注册，界面在5秒之后自动跳转到登陆界面，或者用户手动点击进入登录界面。

5.3.4 后台处理设计

涉及到的类：contoller层：UserController类, PageController页面跳转类；service：UserService接口以及UserSeviceImpl接口的实现类，dao：TbUserMapper接口，以及配置文件TbUserDao.mapper，POJO类：TbUser、TbUserExample类，发送邮件的工具类：SendEmail，密码加密工具MD5Utils，JedisClient Redis工具类，以及静态常量类UserFinalTool。

5.3 个人信息管理模块

5.3.1 模块描述

用户在登录时，忘记密码能够点击忘记密码来修改自己的密码，修改密码时需要验证邮箱，保证是账号的主人操作修改密码；登录之后，用户能够查看自己的个人资料，绑定的手机号，性别，绑定的邮箱，基本住址信息，头像，用户名等信息，并且能对部分信息进行修改。

5.3.2 界面设计







5.3.3 程序逻辑

修改密码分三步：验证邮箱，保证是本人操作；输入修改密码并确认；修改成功自动跳转登录界面；用户在登录时，点击忘记密码，输入用户名和用户名下绑定的邮箱，输入完用户名时，先效验后台数据中是否存在该用户，不存在即提示用户名不存在；输入用户名之后在输入邮箱地址，验证邮箱地址是否正确，不正确提示用户邮箱格式错误，同时根据用户名和邮箱地址一同确认邮箱是否是该用户绑定的，不是即提示用户邮箱错误，邮箱地址和用户名都符合要求时，才可以点击发送验证码到邮箱，确认了邮箱以及操作是本人之后才能进入下一步；下一步输入新密码，密码必须是字符和数字的组合，并且需要输入两次确认密码；修改成功后自动进入登录页面或者用户手动点击进入。

修改个人信息界面，能够修改绑定的邮箱，电话号码，头像等信息；修改头像时需要用户上传图片，并且手动的截切；修改邮箱以及电话号码时需要验证是否是本人操作，修改性别，居住地时没有约束。

5.3.4 后台处理设计

涉及到的类：contoller层：UserController类, PageController页面跳转类；service：UserService接口以及UserSeviceImpl接口的实现类，dao：TbUserMapper接口，以及配置文件TbUserDao.mapper，POJO类：TbUser、TbUserExample类，发送邮件的工具类：SendEmail，密码加密工具MD5Utils，JedisClient Redis工具类，Base64加密，TimeUtils自定义时间类，JsonUtils 对象转Json字符串工具类，以及静态常量类UserFinalTool。

5.4 数据查询模块

5.4.1 模块描述

用户在系统中的所有做题统计都会记录到系统中，记录系统中所有做题统计信息，以及记录每个分类的统计信息以及正确率。

5.4.2 界面设计



5.4.3 程序逻辑

5.4.4 后台处理设计

5.5 个人信息管理模块

5.5.1 模块描述

5.5.2 界面设计

5.5.3 程序逻辑

5.5.4 后台处理设计

5.3 个人信息管理模块

5.3.1 模块描述

5.3.2 界面设计

5.3.3 程序逻辑

5.3.4 后台处理设计

5.3 个人信息管理模块

5.3.1 模块描述

5.3.2 界面设计

5.3.3 程序逻辑

5.3.4 后台处理设计

5.3 个人信息管理模块

5.3.1 模块描述

5.3.2 界面设计

5.3.3 程序逻辑

5.3.4 后台处理设计

5.3 个人信息管理模块

5.3.1 模块描述

5.3.2 界面设计

5.3.3 程序逻辑

5.3.4 后台处理设计