

目录

项目	1
Web 端	1
1. 注册	1
2 测试功能	2
3.书籍模块	3
4 推荐模块	4
5 帖子模块	4
6 学习模块	5
Android 端	6
1. 登陆注册	6

项目

1. 使用 spring+mybatis+springmvc 框架
2. 使用 mysql5.17 版本数据库
3. 在 web 端对数据进行一定的处理，显示前，客户端也进行一定的处理

Web 端

1. 注册

1.0 使用 userct 页面，页面添加 username, userpasswd,usermodel,以及三个接口

1.1 服务端 service 层提供 useradd 接口，userquery 接口（通过用户名查询密码），userchange 接口

1.2 服务端 dao 层实现上 service 层的功能

1.3 服务端控制层提供 regist 接口，login 接口，change 接口。

1.4 regist 接口获取用户信息模型，调用 useradd 接口，实现注册功能，返回值“0000”，失败则返回“0001”；

1.5 login 接口获取用户输入的用户名和密码，调用 userquery 接口通过用户名查询该用户密码，与从页面获取值进行比对，成功则返回用户个人信息的 json 数据，失败则返回“0001”；

1.6 change 接口从页面获取用户 id 和更改后的用户信息，调用 userchange 进行更改，成功返回“0000”，失败则返回“0001”

1

2 测试功能

2.0 创建 testct 控制层，拥有 result, test, category, test_id , category_id,user_id 属性

2.1 在服务器磁盘目录中创一 Test 目录，存放测试题目，同时构造 tomcat 虚拟目录，用来映射此文件夹

2.2 在 controller 层构建 category 接口，test 接口，download 接口，save 接口，history 接口，addcategory 接口和 addtest 接口，在 service 层构建相应接口，_s。

2.3 在 dao 层实现对 psychotest 表的增加，查询（通过 id 查询其路径，通过分类 id 查询所有）；对 psychotest_category 表的增加和查询（查询所有分类）；对 test_result 实现增加和查询功能，根据 user_id 查询。

2.4，对于 category 接口，调用 category_s,查询所有目录，对于 test 接口，根据分类 id 查询所有试题

2.5 对于 addcategory 接口，从界面获取 category 模型信息，进行插入，对于 addtest,从界面获取 test 信息进行插入

2.6，对于 download 接口，从页面获取 test_id,根据 id 调用 test 的查询方法，返回相应的路径，然后执行跳转，进行下载操作

2.7 对于 save 接口，从页面获取 psychotest_result model 对象，调用接口进行插入，对于 history 接口，调用 result 表的查询方法，根据用户 id 查询结果信息

3.书籍模块

3.0 在 bookct 控制层，需要 category, book_id,category_id,book 属性

3.1 在服务器磁盘构造 book 目录，存放书籍

3.2 在 controller 层创建 category_book 接口， addcategory 接口， download_book 接口（id 查路径）， query_book 接口

3.3 对于 category 接口，查询所有目录信息，对于 addcategory 接口，从页面获取 category 模型，进行插入

3.4 对于 download 接口，从页面获取书籍 id，从表中查询相应路径，

3.4 对于 query 接口，从页面获取分类 id 查询所有书籍，封装成 list 对象，返回若出错，则返回空，对于 addbook，从页面获取 book 模型，及进行插入

4 推荐模块

4.0 在 article 层，拥有 article，article_id 属性

4.1 在服务器磁盘构造 recommend 目录，存放推荐文章

4.2 在 controll 层创建 queryrmd 接口，addrmd 接口，dowondrmd 接口

4.3 对于 queryrmd,查询所有推荐文章（保存不超过 20 条），封装成 list 返回，若失败，则返回空

4.4 对于 addrmd 接口，从页面获取 rmd 模型，进行插入

4.4 对于 download 接口，根据 rmd_id,查询相应路径

5 帖子模块

5.0,在 card 页面, 拥有 card,cemment,diary,user_id 属性

5.1 构建 addcard 接口, query_card 接口, adddiary 接口, query_diary 接口, addcemment 接口, query_myself 接口

5.2 对于 add card 接口, 从页面获取 card 模型, 进行插入, 返回值如上, 对于 querycard 接口, 以时间排序进行分页查找, 每次查询十条数据, 封装 list 返回, , 增加 carddetails 接口, 根据 cardid 查询相应的评论, 并将评论封装成 list 返回

5.3 对于 adddiary,从页面获取 diary 模型, 进行插入; 对于 query_,从页 myself 面获取 user_id, 进行查询

5.4 对于 addcemment,从页面获取 comment 模型进行插入

6 学习模块

6.0 在 know 层, 拥有 category,user_id,category_id,learn_result,knowledge,knowledge_id 属性

6.1 创建 komwledge 目录存放知识点, 创建映射,

6.2 在 controller 层创建 querycategory 接口 addccategory 接口, queryknow 接口, addknow 接口, downknow 接口, saveres 接口, queryres 接口。

6.3 对于 querycategory 接口，查询所有目录，用 list 封装返回，对于 addcategory 接口，从页面获取 category 模型，执行插入

6.4 对于 queryknow 接口，从页面获取 category_id，根据分类 id 进行查询；对于 addknow,从页面获取 know 模型，进行插入；对于 down 接口，从页面获取 know_id, 查询相应路径，进行下载

6.5 对于 saveres，从页面获取 learn_result 模型，进行插入，对于 queryres,从页面获取 user_id,根据 id 查询结果

Android 端

1. 添加 model 类

1. 登陆注册

1.0 验证码模块，网络模块

1.1 创建手机号，用户名，密码，验证码输入框，获取验证码按钮和注册按钮，对于验证码按钮，添加事件，发送验证码，同时在 activity 中保存该值,对于注册按钮，添加事件验证输入验证码值是否正确，然后封装 user 对象转换为 Json，发送到服务端调用 useradd 接口

1.2 登陆 从页面获取用户名和密码，将用户名和密码发送至服务端 login 接口，获取返回值判断是否登陆成功，错误则重新输入信息，成功则跳转入主页

1.3 忘记密码 添加验证码验证，将用户手机号和新密码发送至客户端，根据手机号修改密码，修改完成后跳转至登陆页面

1.4 验证码模块 导入相应的 module 和 jar 包，调用 messageutil 发送验证码，验证码随机生成四位 $(\text{random} * (7999) + 2000)$ ，并转换为 string 类型，保存该值与输入的验证码作比较；两个方法，生成并发送验证码（无参并返回随机数，），验证验证码是否正确（输入页面值，返回 boolean 值）

1.5 网络模块 构建 post 方法，从 getmyparam 中获取 hashmap,迭代 map 获取参数并构建 requestbody 对象，传入 call 对象，执行 execute；创建 asynchrotask,在 doinback 中调用 post 方法，根据执行时根据需要重写 result 方法。

1.6 页面值封装 同步静态方法 setmyparam (string s,view v)创建静态 hashmap 对象，调用方法前线 cleanmap，从页面获取 edittext 中的数据，将键值对传入 map 中,getmyparam(), 返回 hashmap 对象

主页

1. 添加测试按钮，进入测试类型列表，点击具体的测试类型，展示该类型下的所有测试题目，题目展示列表项存在按钮，若该项存在，则显示进行测试，若不存在，则为下载按钮
2. 添加书籍阅读，
3. 显示一个列表视图（推荐）

测试

1. 向服务器发出 category 请求，获取到返回的 json 数据，将其封装为 list 对象，构造 list<map>作为数据源（其中封装名称，）将对象中的目录名称显示到页面，添加点击事件，根据获取到的目录 id,向服务器发出 test 请求，并将该对象传递到测试页面，跳转进去测试项页面（按钮事件和列表项事件会分开）
2. 进入测试页面，获取测试项的列表数据，构建 listmap 封装 list,在其中保存测试名称，下载地址，点击按钮进入测试
3. 测试，测试前进行加载，如果存在测试文件则直接显示，如果不存在，测下载后显示
4. 传递 jsonlist 对象 使用 gson new
TypeToken<List<Person>>().getType() 进行转换

5. 测试文件是否存在 直接根据目录构建 file 对象，返回其是否存在
6. 测试文件存储 存放与公开私有目录下 file 下 document 目录下，根据其目录 id+文件 id 进行拼接存储
7. 测试文件解析

书籍阅读

1. 发出请求，返回分类链表，展示该分类结果，点击某分类时，传输分类 id,请求分类下所有内容，返回书籍模块，链表展示，存放下载按钮，重写 getview,更改按钮显示内容

推荐

1. 发出请求，返回推荐数据，显示，生成文本阅读界面，显示文本内容
2. 文本阅读模块 实现 txt 文档，pdf 文档的阅读

