计算机学院 软件工程 课程实验报告

|  |  |  |
| --- | --- | --- |
| 实验题目：实验十 设计模块+编写程序（文档） | | 学号：201805130176 |
| 日期：2021年6月3日 | 班级： 计科二班 | 姓名： 李博远 |
| 实验目的：  1. 练习设计模式应用  2. 掌握编程的标准和过程。  3. 编程指导原则：讨论程序编写方式与规则：  4. 文档：内部文档、外部文档  5. 记录项目及小组的工作进度。 | | |
| 实验步骤与内容：   1. 结合自己的项目，指出其中的数据管理、异常处理、UI方面的设计；它是否使用了框架？  数据管理静态数据 静态数据，又称“横截面数据”，是指在运行过程中主要作为控制或参考用的数据,它们在很长的一段时间内不会变化，一般不随运行而变。在此，我们用到的静态数据主要分为以下两方面：  **表4.1.1-1 静态数据表**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | 名称 | 设置值 | 定义 | 格式 | 类型 | | 服务器操作部分 | 模型地址 | /model\_path/.. | 用于保存训练后的模型的地址 | String | String | | sql | ‘select \* from ...’ | 数据库操作的固有命令，用于直接处理信息 | String | String | | finalPath | /image/../.. | 用于保存用户上传的图像信息 | String | String | | …… |  |  |  |  |  动态数据 **表4.1.2-1 动态数据表**   |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | 名称 | 定义 | 格式 | 类型 | | 服务器操作部分 | 用户个人信息 | 用于保存用户个人设置及隐私信息 | struct | struct | | 服务器流量 | 用于保存当日访问人数 | integer | integer | | 用户字体分数 | 用于保存用户拍照字体历史得分数据 | double | double | | 用户天梯分数 | 用于保存用户天梯得分数据 | String | String | | …… |  |  |  |  数据词典 **表4.1.3-1 用户信息表**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 字段 | 描述 | 类型 | 允许空值 | 默认值 | | id | 用户id | int(10) UNSIGNED | No |  | | name | 用户名 | varchar(255) | No |  | | email | 用户邮箱 | varchar(255) | No |  | | email\_verified\_at | 邮箱验证时间 | timestamp | Yes |  | | password | 用户密码 | varchar(255) | No |  | | remember\_token | 找回密码令牌 | varchar(100) | Yes |  | | created\_at | 创建时间 | timestamp | Yes |  | | updated\_at | 更新时间 | timestamp | Yes |  | | avatar\_url | 头像地址 | varchar(255) | No | /avatar/default\_avatar.png' | | phone | 手机号 | varchar(255) | No |  | | birthday | 生日 | date | No | 1901-01-01' | | gender | 性别 | tinyint(1) | No | 0 | | structure | 用户字体布局得分 | int(11) | Yes |  |   表4.1.3-2 用户上传字体信息表   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 字段 | 描述 | 类型 | 允许空值 | 默认值 | | word\_id | 字体id | int(10) UNSIGNED | No |  | | user\_id | 对应用户id | int(10) UNSIGNED | No |  | | user\_img\_path | 用户字体图像存储 | varchar(255) | No |  | | grade | 用户得分 | double | No | 0 | | date | 字体得分日期 | date | No | 1901-01-01' | | YB | 运笔得分 | int(11) | No | 0 | | DX | 大小得分 | int(11) | No | 0 | | JG | 结构得分 | int(11) | No | 0 | | ZX | 中心得分 | int(11) | No | 0 | | JD | 角度得分 | int(11) | No | 0 | | style | 该字体风格 | int(11) | Yes |  | | img\_path | 标准字体图像存储 | varchar(255) | No |  |   表4.1.3-3 天梯信息表   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 字段 | 描述 | 类型 | 允许空值 | 默认值 | | user\_id | 用户id | int(10) UNSIGNED | No |  | | time | 赛季时间 | varchar(255) | No |  | | point | 用户该赛季点数 | int(11) | No | 1200 |   表4.1.3-4找回密码表   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 字段 | 描述 | 类型 | 允许空值 | 默认值 | | email | 验证的邮箱 | varchar(255) | No |  | | token | 找回密码令牌 | varchar(255) | No |  | | created\_at | 创建时间 | timestamp | Yes |  |  框架 落笔云烟后端利用Flask+Celery+Redis+MongoDB组合作为业务处理逻辑，采用Nginx+Gunicorn进行部署，使用pytorch进行模型的构建。   * 采用Flask的原因  1. Flask框架为轻量级 Web 应用框架，其有着极强的扩展性，对比Django等框架，Flask拥有更高的灵活性适合自己设计代码框架。 2. 由于本项目需要使用多种自实现深度学习模型，使用Python应用框架便于加载模型，结合flask扩展性强的特点可以轻松引入。  * 使用Redis的优势  1. 对于纯静态数据，直接使用URL进行缓存，在之后处理中可直接返回数据，无需访问磁盘。 2. 采用Redis的另一个目的便是作为Celery任务队列的中间件。   落笔云烟前端采用RxJava+Retrofit作为网络请求框架，采用GSON作为数据实体处理库，采用Glide作为图片加载框架，UI采用了swiperefreshlayout、stickyDecoration、calendarview、RollingText、MPAndroidChart等社区组件，MVVM调用到了官方架构组件Lifecycle、Viewmodel、Livedata、Paging、Navigation以及Data。   * 采用Retrofit的优点：  1. 可以配置不同HTTP client来实现网络请求，如okhttp、httpclient等 2. 将接口的定义与使用分离开来，实现结构。 3. 支持多种返回数据解析的Converter可以快速进行数据转换。 4. Retrofit和RxJava集成的很好 5. 因为容易和RxJava结合使用，所以对于异步请求，同步请求也不需要做额外的工作。  * 采用官方MVVM组件库的优点：  1. ViewModel：因设备配置改变导致 Activity 重建时，无需从 Model 中再次加载数据，减少了 IO 操作 2. LiveData：更新 UI 时，不用再关注生命周期问题 3. Data Binding： 可以有效减少模板代码的编写，而且目前已经支持双向绑定 4. 上网查询“阿里 编程规范 （如：阿里巴巴JAVA开发手册）；华为 编程军规”等，对照自己的代码看有哪些不符合规范的地方，修改。   规范1：杜绝完全不规范的缩写，避免望文不知义。    规范2：不能使用过时的类或方法。    规范3：Object 的 equals 方法容易抛空指针异常，应使用常量或确定有值的对象来调用 equals。  规范4：所有整型包装类对象之间值的比较，全部使用 equals 方法比较。  规范5：浮点数之间的等值判断，基本数据类型不能用==来比较，包装数据类型不能用 equals 来判断。    规范6：在 JDK7 版本及以上，Comparator 实现类要满足如下三个条件，不然 Arrays.sort， Collections.sort 会抛 IllegalArgumentException 异常。 说明：三个条件如下 1） x，y 的比较结果和 y，x 的比较结果相反。 2） x>y，y>z，则 x>z。 3） x=y，则 x，z 比较结果和 y，z 比较结果相同。  规范7：当 switch 括号内的变量类型为 String 并且此变量为外部参数时，必须先进行 null 判断。    规范8：类、类属性、类方法的注释必须使用 Javadoc 规范，使用/\*\*内容\*/格式，不得使用 // xxx 方式。    规范9：所有的抽象方法（包括接口中的方法）必须要用 Javadoc 注释、除了返回值、参数、 异常说明外，还必须指出该方法做什么事情，实现什么功能。 | | |