《软件体系结构》课程架构设计

跑操管理系统”Destination”详细设计文档

整理：71Y16114龚呈

对象：学生

作者：71Y16110 曲正锟

功能1：预定/提醒功能

相关数据库中的参数：Date date

函数1：void Booktime()

功能：预定跑操时间

描述：对预定按钮进行监听，连接数据库，输入年份，月份，日期，时，分，并存入数据库完成预定，允许多次调用进行多次预定。

函数2：void CheckBook()

功能：查看已有的预定

描述：对查看预定按钮进行监听，连接数据库，获取数据库中的所有的月份，日期，时，分并依次打印出来，在每条预定后由删除和修改按钮，对于预定进行删除和修改操作。

函数3：void Monitor()

功能：提醒用户进行跑操

描述：连接数据库，获取系统当前日期，时，分并与数据库中已有预定的日期，时，分进行对比，若全部相符则进行提醒，新建一个子线程设置每分钟运行一次该函数。

功能2：跑步路线定制

功能3：个人账户/个人主页

相关参数：String id int Stu\_id int School\_id String Stu\_name int Stu\_gender

String phone\_number String email

函数1：void InitialInfo()

功能：初始化个人信息

描述：在注册后自动调用该函数，输入个人信息，包括账号，一卡通号，学校编号，性别，电话号，邮箱，姓名并存入数据库。

函数2：void ChangeInfo()

功能：更改个人信息

描述：监听更改信息按钮，并对个人信息做出修改

函数3：void Setvisible()

相关参数：int state;

功能：对不同人设置是否可见

描述：可以用0，1表示该用户的可见状态，访问一个用户的主页时先获取该用户的可见状态如果是0的话显示“该用户屏蔽了个人信息”，如果是1的话就连接数据库，打印出相关的信息。

函数4：void AddFriend(String id)

相关参数：String id；

功能：添加好友

描述：输入账号，监听查找按钮，在数据库中进行搜索，如果有该账号则发出添加好友的请求，添加成功后，将两人的id存入FriendShip，如果没有则提示“未搜索到该用户”

函数5：void DeleteFriend()

功能：删除好友

描述：监听删除按钮，在FriendShip中的column1，2寻找两人id，找到匹配项后删除该项。

函数6：void CheckFriend()

功能：查看好友资料

描述：点击好友时自动调用，若该好友设置可见则连接数据库，找到对应的Id下的个人信息并打印出来，设置不可见则显示“该用户屏蔽了个人信息”。

函数7：void LookLeaderBoard()

功能：查看排行榜

描述：连接数据库，在StudentTable中获取每个用户的跑操分数score，排序后，按照从大到小分别打印前100用户的id和score.

对象：学生

作者：71117432奉捷

目前我分了两个类：一个表示能量的类，一个学生类

类1：

class Energy{

private double energy; //表示新生成的能量数量

private double LossEnergy; //表示被好友收取de 能量

private double ExpirationTime; //表示若未被收取，能量的最长存活期

HashMap WhoSteal = new HashMap(); //一个map，用来存放收取过我能量的用户的id以及收取能量的数量，通过判断map中是否含有某个用户的id来判断某个用户是否收取过我的能量，以此保证一个用户一个能量周期内不能多次收取我的能量

这个类有一个构造函数，以及三个成员各一个set、get函数

void set\*\*\*(double)

void get\*\*\*();

}，

类2：

class Student{

private double ExerciseDistance; //用户的锻炼距离，是计算能量值的关键

private double TotalEnergy; //用户总能量数值

private String UserID; //用户ID

private Energy newenergy = new Energy(); //Energy类对象，用于表示新生成的尚未收取的能量

函数1：private void EnergyTran()

参数：无

功能描述：根据锻炼距离的大小，分一定层次转化成相应数量的能量，锻炼距离超过一定值时，能量达到最大值，不再随距离增加而增加。

函数2：private void EnergyIncrease(double take)

参数：double take，表示收取的能量数值

功能描述：在收取能量与收取他人能量的函数中执行，相应增加我自己的能量

函数3：void TakeEnergy()

参数：无

功能描述：收取自己的能量，并设置Energy类对象的能量、过期时间、记录收取过我能量的用户的map为初始状态，以此表示能量已被收取

函数4：double TakeOthersEnergy(Student OtherUser)

参数：另一个学生类对象，表示将被收取能量的用户

功能描述：首先判断我是否收取过这个用户的能量，若收取过，返回-1；若未收取，判断OtherUser被收取的能量是否超过生成新能量的一定比例（比如40%），若超过，返回-2；若没超过，随机收取一定比例的能量，修改OtherUser剩余的未收取能量，增加我的能量，返回收取的能量数

函数5：static double getDistance(double lng1, double lat1, double lng2, double lat2)

参数：经度1，纬度1，经度2，纬度2

功能描述：调用此函数，返回两个经纬度之间的距离

函数6：static String sendGet(String url, String param)

参数：第一个参数为请求的url，第二个参数为get的参数(e.g.:sendGet(“www.baidu.com”, “key=1&name=test”))

功能描述：包括但不限于Get请求一个地图定位的api，返回包含位置信息的字符串，可用于get数据，具体取决与网址与参数

函数7：static String sendPost(String url, String param)

参数：第一个参数为请求的url，第二个参数为get的参数(e.g.:sendPost(“www.baidu.com”, “key=1&name=test”))

功能描述：包括但不限于Post请求一个地图定位的api，返回包含位置信息的字符串，课用于post数据，具体取决于网址与参数

}

对象：学生

作者：71Y16126金杰

**功能1：获取天气信息及锻炼建议**

类：WeatherInformation //根据申请的api获得实时天气信息(实时天气，温度)

具体实现：

|  |  |  |  |
| --- | --- | --- | --- |
| **函数** | **逻辑实现** | 输入数据 | 输出数据 |
| getTodayWeather() | 根据中央气象台的api,建立Connection,获得返回的json格式数据，调用相应的jar包，对该数据进行处理，处理后的数据用map封装，然后返回 | 城市id | Map<String,Object>类型的天气数据 |
| getExeAdvice() | 调用getTodayWeather()的返回值，通过判断语句对相应的天气数据进行判断（map.get(key))，给出对应天气状况下的跑操建议（String） | Map<String,Object>类型的天气数据 | String类型的跑操建议 |

**Addition:对于天气信息的呈现（present()），输入数据为（map.get(“weather”),component），第一个参数为具体的天气，第二个参数为要呈现的component，无具体返回值,逻辑实现:**

**通过判断语句为天气和图标一一对应，再加上具体的温度信息，利用component.set()呈现。**

**功能2：对于学生各类跑操信息的获取及给予建议**

**核心操作：通过点击相应的标签向服务器发送请求，接受服务器回传的数据并处理，最后呈现在跳转的界面上**

**各类跑操信息包括：1，过去跑操的详细数据;2,早晚操的完成情况；3，当天的早晚操情况；4，已经获得的分数/未获得的分数，已经经过的日期/剩余日期**

**其中1，3两点建立在已有日历的基础上**

类：1，MainFrame //各类标签的集合体，包括早晚操完成情况查询，分数/日程进度，日历，跑操建议获取。

2，Calendar //MainFrame的主体,提供对过去跑操信息的查询，当天跑操状况的查询

注：各函数中需要调用一个共有函数ConnectToServer(Map<type,data> map):建立与服务器的连接，向服务器发送信息（请求类型和请求数据内容），并接受服务器的返回数据

|  |  |  |  |
| --- | --- | --- | --- |
| **函数** | **逻辑实现** | 输入数据 | 输出数据 |
| 监听函数1（主要是MainFrame包含的标签，即早晚操的完成情况和跑操建议的获取） | CompletenceOfAP():早晚操的完成情况  调用ConnectToServer(Map<A,time)  (依照约定，服务器应当返回Map<key,data>类型的数据)  对返回的数据调用map.get(key)获取早晚操的完成次数及得分，最后将相应数据通过component.set()进行设置 | Map<A,time> map:  A代表早晚操完成程度的请求类型，time代表希望获得跑操次数的请求  A:id | 早晚操的完成次数及得分 |
| getProAdvice():  获得关于跑操进度的建议，具体实现还没想好 |  |  |
| 监听函数2（主要是在日历上实现的请求，包括历史和当天的跑操情况的请求） | getDailyInfo/getHistoryInfo()  首先获取点击日期date  调用ConnectToServer(Map<B,date>  对返回的数据调用map.get(key)(其中key包括早晚操开始结束时间，跑步路线，平均速度，路程长度，最快速度等）  将解析后的数据在新跳转的界面上呈现:component.set() | Map<B,date> map:  B代表具体日期跑操情况的请求类型，date表示所要请求信息的发生日期  B:id | 早晚操开始结束时间，跑步路线，平均速度，路程长度，最快速度等 |
| PresentProcession():呈现已经获得的分数/未获得的分数，已经经过的日期/剩余日期 | 由于是呈现函数，所以只要将服务器返回的数据填入相应的MainFrame中的component  调用ConnectToServer(Map<C,score>)  建立时间间隔(每隔一天自动去调用这个函数，实现更新效果  将返回的数据呈现在MainFrame的相应component中  component.set()  (需要一点计算) | Map<C,score>):C代表获取已经获得的分数的请求类型  C:id | 返回已经获得的分数/未获得的分数，已经经过的日期/剩余日期（日期可以通过调用系统时间就可以计算） |

**功能3：体育老师发布跑操信息/班主任（辅导员）发送鼓励信息**

**类：Manage（P.Eteacher/monitor):这个功能只是这个类里的一个方法**

|  |  |  |  |
| --- | --- | --- | --- |
| **函数** | **逻辑实现** | 输入数据 | 输出数据 |
| SendMessageAll():向全体成员发布跑操信息  体育老师 | String message;  首先执行I/O操作，输入message;  调用ConnectToserver(Map<D,message>)  (服务器接受请求后就向全体成员发布消息)  ReceiveMessage()&Present():客户端接收信息并呈现(接受的信息类型应该是Map<type(身份），message>  Reflect():服务器返回给体育老师message是否发布成功（boolean)  条件？ | String message  Map<D,message>  D代表发布信息给全体成员的请求类型） | message的呈现  发布信息是否成功（true/false） |
| SendMessageToGroup():向某个班级或某个院系成员发布消息  班主任/辅导员 | String message;  首先执行I/O操作，输入message;  调用ConnectToserver(Map<班级代码/院系代码,message>)  (服务器接受请求后就向班级或院系内的全体成员发布消息)  ReceiveMessage()&Present():客户端接收信息并呈现(接受的信息类型应该是Map<type(身份），message>  Reflect():服务器返回给班主任或辅导员message是否发布成功（boolean)  条件？ | String message  Map<,message>  D代表班级代码/院系代码） | message的呈现  发布信息是否成功（true/false） |

**注：逻辑实现内的某些没有实现的函数其实只是一些操作，可以不写成一个函数**

**注：有些类型虽然有数据库的支持，但是我还是不好确定**

对象：老师

作者：71Y6122沈梦烨

class Manager{

**函数1**：boolean SetStandard()

参数：

int school\_id , string school\_name , int auther\_id , string period ,

string time , string run\_time , int pass\_score , string score set ,

string run\_limit ,string standard , string run\_standard

功能1：针对不同分组的学生能设定不同标准

描述：对“查询”按钮设置监听，把在界面输入的用来设置标准的各个数据分别存储到上面的对象中，然后(如上)打包到map，其中有个关键字（服务器会根据这个来判断是什么请求）是存储到数据库的UniversityRule表，然后用sendPost（）发送给服务器，服务器用ParseMap（）函数识别关键字，知道是存储请求后，调用相关存储到数据库的函数存储，若此函数返回true，则回给客户端true，本函数return true，否则弹出对话框“失败并重试“。

**函数2**：boolean SetReward()

功能2：设定额外的奖励目标

描述：体育老师可新增 如冬季长跑？阳光长跑？ 等活动 额外增加学生跑操次数。具体实现，还有待思考…总觉得会被鸽（此功能开发中….）。

**函数3**：boolean CheckAllinfo()

**函数4**：boolean CheckClass(string banji)

功能3: 获得全体综合信息报表（全体/班级）

描述：把相关信息（查询全体或者具体某个班级banji等）打包，其中有个关键字（服务器会根据这个来判断是什么请求）是查询数据库的StudentSportsRecord表，然后用sendPost（）发送给服务器，服务器用ParseMap（）函数识别关键字，知道是查询请求后，调用相关存储数据库的函数查询，若此函数返回true，则回给客户端true，并把数据打包到Answer返回给客户端，然后客户端把接收到的信息显示在界面上，并求出数据的平均值显示在界面上，本函数return true，否则弹出对话框“失败并重试“。

**函数5**：boolean CheckStudent(int stu\_id)

boolean CheckStudent(string stu\_name)

功能4：查看所有学生的个人资料

描述：把在界面输入的用来查询某个学生的各个数据打包到map，其中有个关键字（服务器会根据这个来判断是什么请求）是查询数据库的StudentInfo表，然后用sendPost（）发送给服务器，服务器用ParseMap（）函数识别关键字，知道是查询请求后，调用相关查询数据库的函数查询，若此函数返回true，则回给客户端true以及查询到的result，本函数把数据显示在界面上并return true，否则弹出对话框“失败并重试“。

｝

class Teacher｛

**函数1**：boolean CheckClass(string banji)

功能1：辅导员/班主任能够了解本系（分班级）/本班同学某日的跑操整体情况

描述：把相关信息（查询全体或者具体某个年级、班级、学院等）打包，其中有个关键字（服务器会根据这个来判断是什么请求）是查询数据库的StudentSportsRecord表，然后用sendPost（）发送给服务器，服务器用ParseMap（）函数识别关键字，知道是查询请求后，调用相关存储数据库的函数查询，若此函数返回true，则回给客户端true，并把数据打包到Answer返回给客户端，然后客户端把接收到的信息显示在界面上，并求出数据的平均值显示在界面上，本函数return true，否则弹出对话框“失败并重试“。

**函数2**：boolean CheckStudent(int stu\_id)

boolean CheckStudent(string stu\_name)

功能2：辅导员/班主任能够查询某个同学的跑操记录和成绩。

描述：把在界面输入的用来查询某个学生的各个数据打包到map，其中有个关键字（服务器会根据这个来判断是什么请求）是查询数据库的StudentSportsRecord表，然后用sendPost（）发送给服务器，服务器用ParseMap（）函数识别关键字，知道是查询请求后，调用相关查询数据库的函数查询，若此函数返回true，则回给客户端true以及查询到的result，本函数把数据显示在界面上并return true，否则弹出对话框“失败并重试“。

**函数3**：boolean Encourage()

功能3：辅导员/班主任能够对优秀的同学予以表扬与鼓励，不合格的同学予以批评

描述：对“表扬“与”批评“两个按钮设置监听，弹出输入要发消息的编辑框，调用社交功能中的私发消息，发送。此函数主要是调用社交方面的，因不知开发社交功能的同学如何实现，因此此功能暂时不开发。｝