**ChatRoom**

**概要设计说明书**

**编 写 人**：李益军

**编写日期**： 2019年 11月 28日

文档信息：

|  |  |
| --- | --- |
|  |  |
| 文档名称 | ChatRoom概要设计说明书 |
| 描述 | 根据架构设计确定的各子系统、模块，对其中的类和接口进行设计。 |
| 负责人 | 李益军 |
| 状态 | 完成 |

**文档变更记录：**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 版本编号 | 说明：如形成文件、变更内容和变更范围 | 日期 | 变更人 | 批准日期 | 批准人 |
| V0.1 | 形成文件并编辑 | 2019/11/28 | 李益军 | 2019/11/28 | 李益军 |
| V0.2 | 增加程序结构和接口 | 2019/12/1 | 李益军 | 2019/12/1 | 李益军 |
| V0.3 | 增加数据库内容 | 2019/12/2 | 李益军 | 2019/12/2 | 李益军 |
| V0.4 | 增加错误处理条目 | 2019/12/3 | 李益军 | 2019/12/3 | 李益军 |
|  |  |  |  |  |  |

目录

[1.引言 4](#_Toc26714638)

[1.1编写目的 4](#_Toc26714639)

[1.2项目介绍 4](#_Toc26714640)

[1.3技术 4](#_Toc26714641)

[1.4参考资料 4](#_Toc26714642)

[2.总体设计 4](#_Toc26714643)

[2.1功能规定 4](#_Toc26714644)

[2.2运行环境 5](#_Toc26714645)

[2.3基本设计概念和架构 5](#_Toc26714646)

[2.4功能需求与程序的关系 6](#_Toc26714647)

[3.接口设计 7](#_Toc26714648)

[3.1用户接口 7](#_Toc26714649)

[3.2服务器接口 8](#_Toc26714650)

[3.3通讯模块接口 9](#_Toc26714651)

[3.4数据库模块接口 9](#_Toc26714652)

[4.系统数据结构设计 10](#_Toc26714653)

[4.1概念模型 10](#_Toc26714654)

[4.2逻辑模型 10](#_Toc26714655)

[4.3物理模型 11](#_Toc26714656)

[5系统出错处理设计 11](#_Toc26714657)

[5.1出错信息 11](#_Toc26714658)

[5.2补救措施 11](#_Toc26714659)

1.引言

### 1.1编写目的

此概要设计说明书是为了说明整个系统的体系结构，以及需求用例的各个功能点在结构中的体现，为系统的详细设计人员进行详细设计的输入参考文档。

### 1.2项目介绍

   本项目为软件工程的课程项目，主要设计目的是仿照现有即时通讯软件（如QQ，Wechat）设计一个能进行一对一聊天的即时通讯软件。

### 1.3技术

1. Client/Server: 软件结构
2. MySQL：数据库，存储数据
3. JAVA编程语言
4. PowerDesigner绘图工具

### 1.4参考资料

1. 需求分析说明书

## 2.总体设计

### 2.1功能规定

1. 客户端模块：

* 链接服务器：确认客户端与服务器端可以通讯；
* 登录账号：客户端能登录聊天用的账号；
* 加载好友列表：能收到服务端发来的好友列表并显示；
* 聊天：两个客户端能通过服务器进行一对一聊天；

1. 服务端模块：

* 创建服务器：创建一个服务器处理客户请求；
* 为每个活动的客户端创建线程：有客户端连接服务器时为其创建线程并处理其发过来的消息；
* 响应客户端登录账号的请求：对每个登录请求做出响应；
* 响应客户端加载好友列表的请求：对每个客户端显示好友列表的请求进行响应；
* 支持一对一聊天：支持两个客户端之间进行一对一聊天；

1. 通讯模块：

* 打包数据流：将数据打包成数据流；
* 解析数据流：将接收到的数据流还原成原始数据；

1. 数据库模块：

* 操作数据库：能响应服务端请求，对数据库进行相应的增删查改操作并返回操作结果；

### 2.2运行环境

1. 操作系统：

服务端：Windows 10

客户端：Android

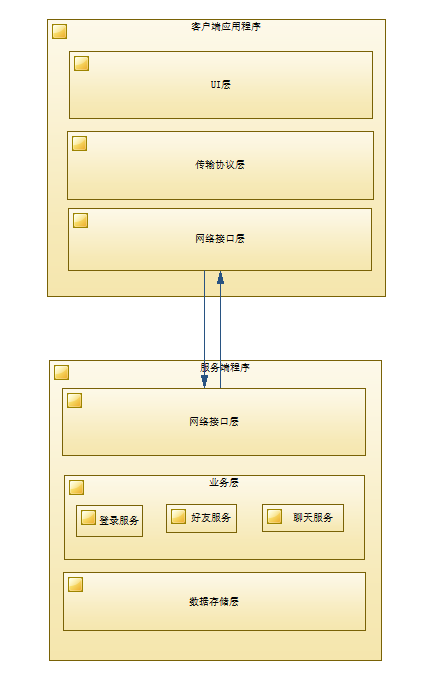
1. 数据库：MySQL
2. Java(TM) SE Runtime Environment (build 1.8.0\_191-b12)

### 2.3基本设计概念和架构

1. 基本设计概念

整个ChatRoom一共分为四个模块，即客户端模块、服务器模块、数据库模块，通讯模块。

1. 架构示意图



### 2.4功能需求与程序的关系

本条用一张如下的矩阵图说明各项功能需求的实现同各块程序的分配关系：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 客户端模块 | 服务端端模块 | 通讯模块 | 数据库模块 |
| 登录 | √ | √ | √ | √ |
| 注册 | √ | √ | √ | √ |
| 聊天 | √ | √ | √ |  |
| 好友 | √ | √ | √ | √ |

## 3.接口设计

### 3.1用户接口

1. public boolean ConnectServer()
   1. 功能描述：请求连接服务器
   2. 参数： 无
   3. 返回值：true(成功连接到服务器),false(不能连接到服务器)
2. public void sendMsg(int to, String Msg)
   1. 功能描述：向uid 为to 的用户发送 Msg 消息
   2. 参数：int to—发送消息接受的用户

String Msg—发送的消息内容

* 1. 返回值： 无

1. public byte[] receiveMsg()
   1. 功能描述：接收服务器发来的消息
   2. 参数：无
   3. 返回值： 字节流byte[]
2. public void processMsg()
   1. 功能描述：处理收到的消息
   2. 参数：无
   3. 返回值： 无
3. public boolean Reg(String NikeName, String PassWord)
   1. 功能描述：注册一个用户
   2. 参数：String NikeName—自定义的昵称

String PassWord—自定义的密码

* 1. 返回值： true(成功发起注册请求)

False(发起注册请求失败)

1. public int Login(int id, String pwd)
   1. 功能描述：用户登录
   2. 参数：int id—用户id

String pwd—用户密码

* 1. 返回值： 0(登录成功)

1(账号或密码错误)

2（已经登录）

3，4（未知错误）

1. public void SendaddFriend(int add\_id, String list\_name)
   1. 功能描述：发送添加好友信息
   2. 参数：int add\_id—好友id

String list\_name—添加到list\_name分组

* 1. 返回值： 无

### 3.2服务器接口

1. public void setupServer(int port)
   1. 功能描述：服务器启动
   2. 参数：int port—服务器监听端口
   3. 返回值： 无
2. public void closeServer()
   1. 功能描述：服务器关闭
   2. 参数：无
   3. 返回值： 无
3. private void processLogin()
   1. 功能描述：处理登录信息
   2. 参数：无
   3. 返回值： 无
4. private void broadcastState()
   1. 功能描述：某个用户上线后向所有其他用户广播其上线状态
   2. 参数：无
   3. 返回值： 无
5. private void sendFriendList()
   1. 功能描述：向用户发送其好友列表
   2. 参数：无
   3. 返回值： 无
6. public void processChat()
   1. 功能描述：处理用户一对一聊天
   2. 参数：无
   3. 返回值： 无
7. public void sendMsg(int from, String msg)
   1. 功能描述：向用户发送来自于from的消息
   2. 参数：int from—消息来源的uid

String msg—消息内容

* 1. 返回值： 无

### 3.3通讯模块接口

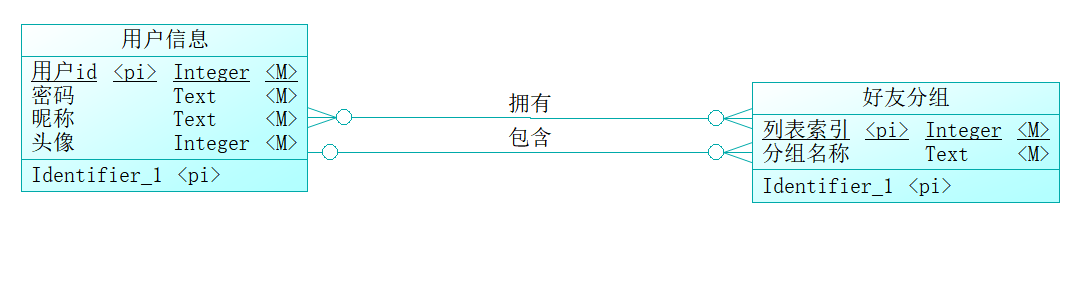
1. public static MsgHead parseMsg(byte[] data)
   1. 功能描述：解析收到的数据流
   2. 参数：byte[] data—收到的字节流
   3. 返回值： 解析出来的消息头
2. public static byte[] packMsg(MsgHead msg)
   1. 功能描述：根据消息类型打包消息
   2. 参数：MsgHead msg—包装好的消息头
   3. 返回值： byte[]—打包好的字节流

### 3.4数据库模块接口

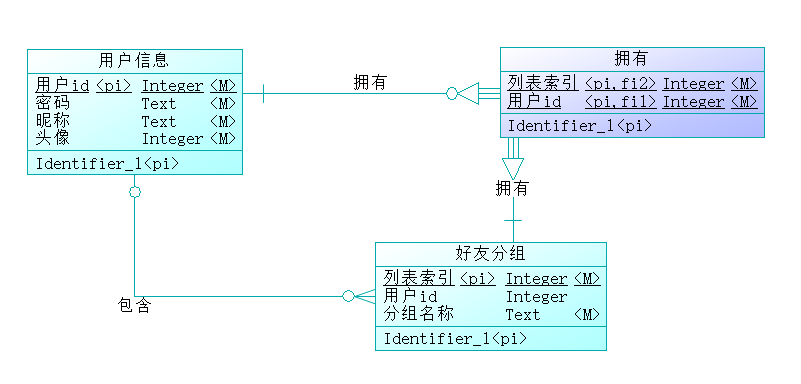
1. public Connection getConnection()
   1. 功能描述：连接数据库
   2. 参数：无
   3. 返回值： 数据库连接对象（Connection）
2. public void release()
   1. 功能描述：释放数据库连接资源
   2. 参数：无
   3. 返回值：无
3. public ResultSet query (String sql)
   1. 功能描述：执行查询sql语句
   2. 参数：String sql—查询语句
   3. 返回值：返回查询结果--ResultSet对象
4. public int update(String sql)
   1. 功能描述：执行更新sql语句
   2. 参数：String sql—更新语句
   3. 返回值：返回更新影响到的条目
5. public int insertAndGet(String sql)
   1. 功能描述：插入一个新条目并获得其索引
   2. 参数：String sql—插入语句
   3. 返回值：返回插入之后的索引

## 4.系统数据结构设计

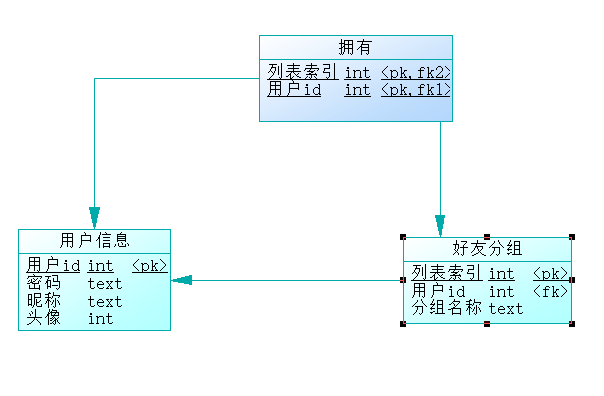
### 4.1概念模型



### 4.2逻辑模型



### 4.3物理模型



## 5系统出错处理设计

### 5.1出错信息

1. 数据库访问出错
2. 数据流解析，包装出错

### 5.2补救措施

1. 手动捕获，返回操作失败。
2. 手动捕获，返回操作失败。