|  |
| --- |
|  |
| MinChat 即时通讯 |
| 设计模型 |

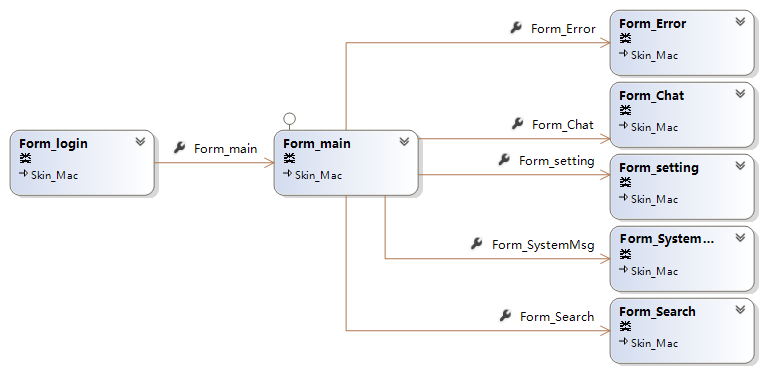
|  |
| --- |
| 胡伟龙 李凌云 郭天翼 汪宇琴 王浩  2016-1-17 |

## 问题域子系统

## 人机交互子系统

### 1.用户相关界面

界面类图如下：



#### 1.1 登录注册

用户可以通过web端注册一个新账号，然后在客户端登陆





#### 1.2 主界面

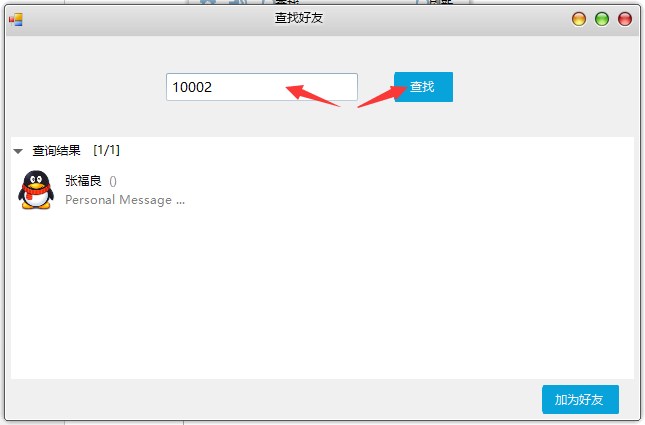


#### 1.3 查找好友

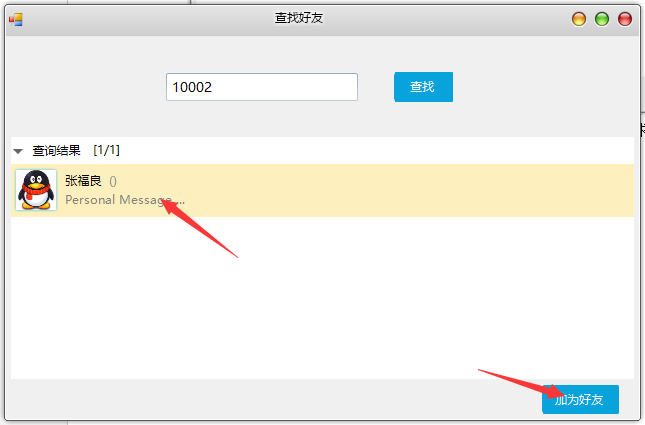
在主界面单机查找按钮可以打开查找好友界面



在输入框中输入用户的ID，单击查找按钮可以查找出该用户

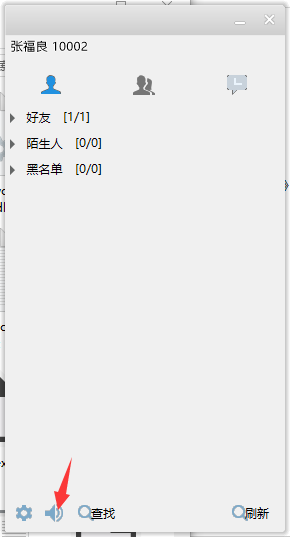


点击选中该用户，单击右下角“加为好友”按钮可以发送加对方为好友请求



#### 1.4 查看系统消息

用户在主界面单击如图按钮可以查看当前未读系统消息



选择同意并确定，通过好友请求

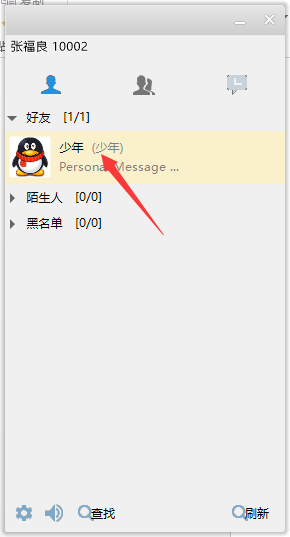


好友便出现在列表中

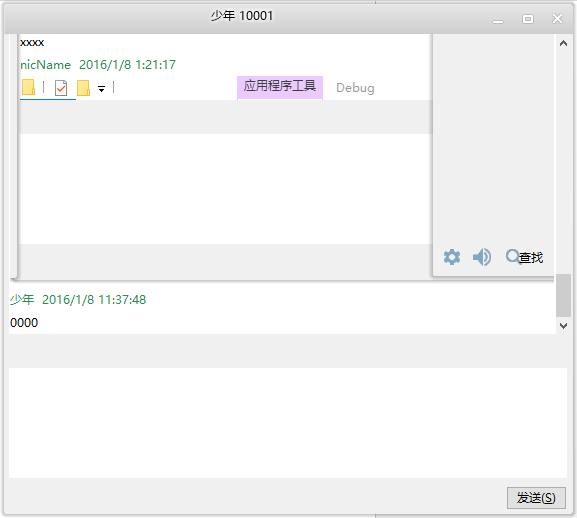


#### 1.5 发送个人消息

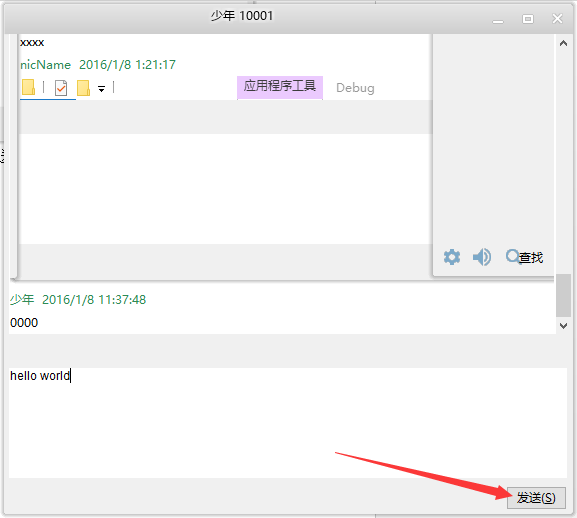
在主界面双击好友头像，可以打开与该好友聊天的窗口

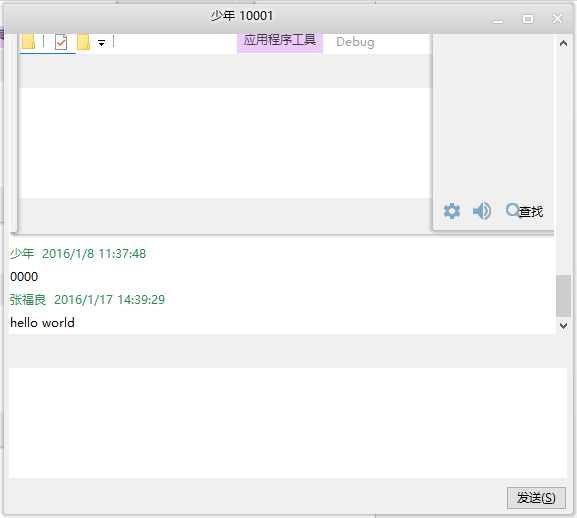


打开后默认载入聊天记录



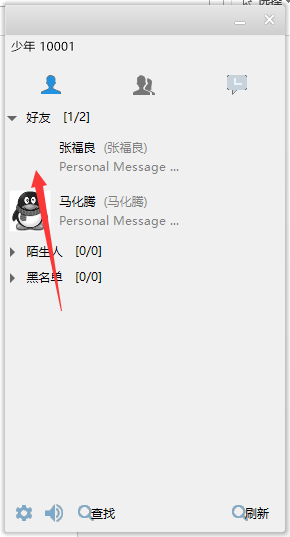
在下方文本框输入要发送的文字，按回车货“发送”按钮发送消息

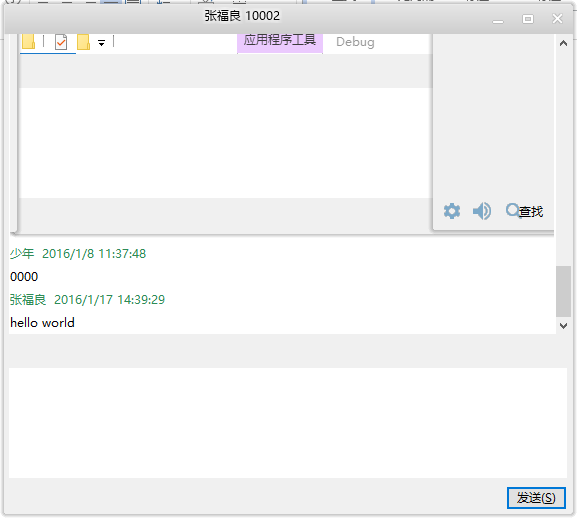




#### 1.6 查看聊天消息

当有好友发来消息，其头像会变成闪烁状态，双击进入聊天，查看消息





#### 1.7 打开设置窗口

单击设置按钮，可以打开设置窗口



### 2.服务器相关界面



## 三、任务管理子系统

### 1.MinChat客户端顺序图

#### 1.1接收消息



#### 1.2发送消息



### 2.MinChat服务端顺序图

#### 2.1登录功能顺序图



【顺序图说明】

1.用户输入账号，密码，系统附加上系统标识，发送给服务端。

2.服务端使用账号密码查询数据库

3.获得查询结果。若存在该用户，则登录成功。若不存在，则报告出错信息。

#### 2.2用户上线顺序图



【顺序图说明】

1.客户端上线触发SomeOneConnected事件。

2.获取上线用户资料。

3.查询该用户是否有离线消息。

4.若有，则发送给用户。

5.发送完离线消息，删除服务器暂存的该用户的离线消息。

#### 2.3查询好友顺序图



【顺序图说明】

1.客户端发起查询好友请求。

2.服务端接收到待查询的用户ID。

3.服务端查询用户数据库，查询到用户资料，为字符串。

4.将字符串格式的用户信息序列化为字节数组。

5.将字节数组格式的用户信息返回给发起查询的客户端。

#### 2.4接收离线消息顺序图



【顺序图说明】

1.客户端向服务端发送离线文本消息或者图片。

2.服务端将离线消息插入到待接收用户的离线消息数据库中。

#### 2.5添加好友顺序图



【顺序图说明】

1.客户端发起申请好友请求。

2.服务端根据申请者ID，查询申请者资料信息。

3.判断被申请者是否在线。

4.若在线，则生成申请好友通知给被申请者，若不在线，则生成离线的添加好友消息放到被申请者的离线消息数据库中。

#### 2.6同意添加好友顺序图



【顺序图说明】

1.客户端发出同意添加好友请求。

2.服务端分别在申请者与同意者数据库中加入好友关系。

3.查询同意者的用户信息。

4.查询申请者是否在线。

5.若申请者在线，则直接向其发送同意添加好友通知，否则在其离线消息数据库中插入离线的同意添加好友通知。

### 3.MinChat Web端顺序图

#### 3.1用户注册顺序图



【顺序图说明】

1.用户在浏览器中输入基本注册信息，点击注册。

2.浏览器进行注册信息简单校验。

3.浏览器通过post请求将注册信息发送给we端。

4.web端查询用户是否已存在，若不存在则插入用户信息到用户信息表。

5.web端建立用户数据库。

6.若建立数据库成功，web端返回给客户端注册成功的账号。

## 四、数据管理子系统

### 1.客户端表结构



### 2.服务端表结构

