P2P聊天软件概要说明书

负责人：

成 员：

分 工：

**1.引言**

**1.1编写目的**

设计一个P2P聊天软件，由多个聊天客户端和一个中心服务器组成。各个聊天客户端可以两两相互发送消息。

**1.2参考资料**

《信息技术提高班培养计划V5.1》

《软件工程——实践与理论》

《软件工程概论》

**2总体设计**

**2.1 功能要求**

1.用户界面（GUI）可自行设计，主要为用户提供方便。

2.用户界面要及时的对用户的操作做出反应。

3.要实现一个用户可以和多个用户同时进行聊天。

4.实现平台Linux或Windows，编程语言C/C++，JAVA。

**2.2运行环境**

服务器：CPU：1G；内存：4G；硬盘：15G

支持系统：Ubuntu16.0.4

**2.3基本设计概念和处理流程**

部分一：P2P聊天中各个成员管理。每个成员称为一个peer，要有一个中心的服务处理peer的登记和对已经登记的peer进行更新。

步骤1：一个peer选择一个名字，通过登记界面进行登记。

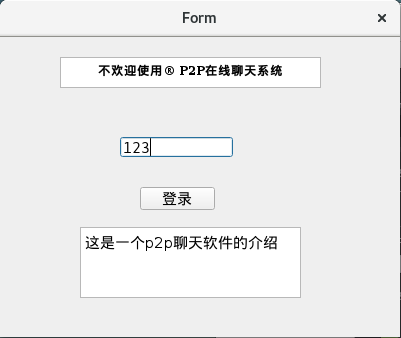


图1 用户1登陆

步骤2：服务器拥有一个公有IP，在一个众所周知的端口进行监听。Peer向服务器发送‘register’请求，同时发送自己的登记信息（如自己取的名字，自己的监听端口号等）。服务器收到‘register’请求后，如果这是第一个登记请求，服务器建立一个用于记录已经登记peer的列表（Peer List），否则更新Peer List。



图2 用户1登陆后分配一个ip

步骤3：服务器向来登记的peer发送Peer List。PeerList 由一些活跃的peer组成，一个活跃的peer是在最近30秒内进行登记的peer。

步骤4：peer收到 Peer List后，以适当的形式显示出来。设计一个显示Peer List的GUI。



图3 当第二个节点登陆后跟新peer list

步骤5：每个已经登记的peer每隔15秒向服务器登记，来使自己保持活跃，同时得到最近新登记的peer信息。如果需要，就在自己的Peer List GUI进行更新。



图4 第一个节点中peer list一起更新

部分二：多个peer间相互发送消息

步骤1：在客户端P1的用户U1（123）在自己Peer List GUI中选择一个peer，假设为U2（321）。

步骤2：P1建立一个到P2的连接。



图5 p1连接到p2

步骤3：U1在自己的发送信息界面（Messaging GUI）编写信息，然后点击发送按钮，P1将信息通过步骤2中建立的连接发送到P2。



图6 p2向p1发送消息



图7 p1接收到p2的消息

步骤4：如果第3个用户U3（222）发送一条消息到U1@P1，如果U1此时正好有一个Messaging GUI与U3进行交互，那么就显示消息，如果没有，就要提示U1有新消息到来。



图8 创建p3后尝试向正在与p2通信的p1发送消息



图9 p1接收到p3的消息

**2.4流程图**

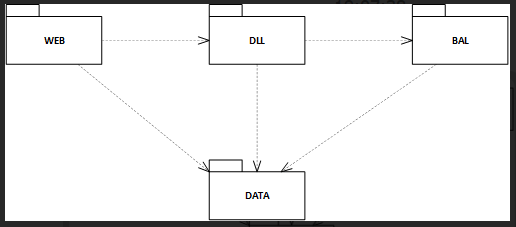


客户端流程图



服务器流程图

**2.5结构**



WEB层：即用户接口层，主要是接收用户的输入，并将处理后的结果显示给用户

BLL层：业务逻辑层，主要是系统业务逻辑的处理，并将处理结果返回给WEB层。

DAL层：数据库访问层，主要负责对数据的访问和操作数据库的操作结果返回给BLL层

DATA层：数据层，作为各层之间数据传递的参数出现

**3.接口设计**

**3.1内部接口**

内部接口方面，各模块之间采用函数调用、参数传递、返回值的方式进行信息传递。具体参数的结构将在下面数据结构设计的内容中说明。接口传递的信息将是以数据结构封装了的数据，以参数传递或返回值的形式在各模块间传输。

**3.2外部接口**

在用户界面部分，根据需求分析的结果，用户需要一个用户友善界面。在界面设计上，应做到简单明了，易于操作，并且要注意到界面的布局，应突出的显示重要以及出错信息。外观上也要做到合理化，考虑到用户多对windows风格较熟悉，应尽量向这一方向靠拢。

**4运行设计**

**4.1运行模块组合**

客户端：每个模块执行单一且相同的功能，模块内部完成高内聚和与其他客户端低耦合，使每个功能互不影响，都能独立与服务器产生连接。

服务器：接收客户端的连接请求，接收并转发消息并维护一张全局的peer list。

**4.2运行控制**

客户端：每个peer登陆之后系统自动分配一个ip，用于与其他的peer连接，其只能够选择由服务器维护的peer list中的peer与其产生对话。

服务器：接收到一个peer节点的登陆消息之后，在peer list中为其创建一项记录，并为其提供所有的peer list中的节点以供其选择。

**4.3运行时间**

本系统个运行模块将本着尽量少占用系统资源、尽快释放占资源的原则进行设计，使得系统运行时间在可以接受的范围内。

1.响应时间：20ms   
2.更新处理时间：10ms   
3.数据的转换和传送时间：5ms   
4.系统数据显示时间：3ms

**5系统数据结构设计**

|  |  |  |
| --- | --- | --- |
| 变量功能 | 变量名称 | 变量类型 |
| 用户昵称 | name\_ | char |
| ip和端口信息 | sockaddr\_ | struct sockaddr |
| 记录上次登陆后经过的时间 | time\_ | int |
| 是否在线 | isonline | bool |