**版本号V1.0**

多人在线聊天系统

需求分析说明书

殷康龙、张猛

2018

目录

[1引言 2](#_Toc508991312)

[1.1项目背景 2](#_Toc508991313)

[1.2参考资料 2](#_Toc508991314)

[1.3 编写目的 2](#_Toc508991315)

[2 需求概述 3](#_Toc508991316)

[2.1 系统目标 3](#_Toc508991317)

[2.2 用户特点 3](#_Toc508991318)

[2.3 假定和约束 3](#_Toc508991319)

[2.4 系统范围 3](#_Toc508991320)

[3功能需求 3](#_Toc508991321)

[3.1用例01：服务器开启（MFC--01） 3](#_Toc508991322)

[3.2用例02：客户端开启（MFC--02） 4](#_Toc508991323)

[3.3用例03：开启TCP通信（MFC--03） 4](#_Toc508991324)

[3.4用例04：开启UDP通信（MFC--04） 4](#_Toc508991325)

[3.5用例05：服务器端监听在线用户（MFC--05） 5](#_Toc508991326)

[3.6用例06：服务器消息的群发（MFC--06） 5](#_Toc508991327)

[3.7用例07：消息的发送与显示（MFC--07） 5](#_Toc508991328)

[4性能需求 7](#_Toc508991329)

[3.1时间特性要求 7](#_Toc508991330)

[3.1输入输出要求 7](#_Toc508991331)

[5界面需求 7](#_Toc508991332)

[5.1服务器端： 7](#_Toc508991333)

[5.2客户端： 8](#_Toc508991334)

[6其他非功能性需求 8](#_Toc508991336)

[6.1.数据管理能力要求 8](#_Toc508991337)

[6.2 灵活性要求 8](#_Toc508991338)

[7 尚未解决的问题 8](#_Toc508991339)

# 1引言

## 1.1项目背景

在学校实验室和老师的教学安排和实验计划下，我们小组选择了一个学期大作业，为了完成学校的安排，TCP—UDP网络通信简单聊天软件成为了我们组开发的一个应用程序。

## 1.2参考资料

Visual\_C++\_6.0程序设计从入门到精通

VC++深入讲解

Windows程序设计（王艳平，第二版）

GDI+SDK参考（翻译版本）

## 1.3 编写目的

此项目为西北工业大学网络编程的课程设计选题。目标是在VS下，使用Win32网络编程和MFC框架实现多人在线聊天的服务器端和客户端。用户能够选择网络中的聊天对象，通过UDP广播或者通过TCP对指定用户进行通信，方便用户的交流。最终，在windows操作系统下实现多人在线即时聊天功能。

# 2 需求概述

## 2.1 系统目标

系统仅适用于局域网络通信，可以实现多人通信。

## 2.2 用户特点

此软件为局域网聊天软件，用户可以随时开启聊天，用户可选择与在线用户连接并发送消息。简化了繁琐的注册和登录问题，用户可直接输入IP和端口号进行连接，适用于广大用户。

## 2.3 假定和约束

假定：实现用户需求软件系统具有良好的可扩充性。

约束：硬件平台，开发语言，开发工具，灵活和配置要求。

## 2.4 系统范围

# 3功能需求

实现TCP的服务器/客户端程序：服务器端能接收多个客户端的连接请求，并能够区分不同的客户端；连接成功后，服务器能接收客户端发过来的文本信息并显示，客户端在本地能够显示自己的发送信息，信息的发送模式为TCP和UDP，客户端在本地可以获取在线用户列表，显示在线IP。根据与广域网聊天工具如QQ，MSN等的对比，及对用户的需求了解，本程序主要有以下功能：

1. 用户之间能够进行信息传递，这是需要实现的最基本功能。
2. 在线用户客户端主机名列表的显示和刷新。
3. 客户端能够监听退出和登录的新客户端。

## 3.1用例01：服务器开启（MFC--01）

**用例描述：**开启服务器，建立连接中心。

**参与者：**用户

**执行者：**用户

**前置条件：**在Windows10操作系统上开启服务器。

**事件流：**

1. 用户开启服务器端 。
2. 用户输入端口号开启连接点。

**后置条件：**服务器开启后开始监听和广播消息，接收连接。

## 3.2用例02：客户端开启（MFC--02）

**用例描述：**用户开启客户端，输入服务器端口号和IP地址建立连接。

**参与者：**用户

**执行者：**用户

**前置条件：**在Windows10操作系统上开启客户端。

**事件流：**

1. 用户开启客户端。
2. 用户输入服务器端口号。
3. 用户输入服务器IP地址。
4. 点击连接建立连接。

**后置条件：**服务器开启后开始监听和广播消息，接收连接。

## 3.3用例03：开启TCP通信（MFC--03）

**用例描述：**用户开启客户端后点击界面上TCP选项，开启TCP通信功能。

**参与者：**用户

**执行者：**用户

**前置条件：**在Windows10操作系统上开启服务。

**事件流：**

1用户开启客户端。

2用户输入服务器端口号。

3用户输入服务器IP地址建立连接。

4连接成功后用户点解界面上的TCP按钮开启TCP通信功能。

**后置条件：**客户端开启TCP通信功能，可以向服务器发送TCP消息。

## 3.4用例04：开启UDP通信（MFC--04）

**用例描述：**用户点击界面上的UDP按钮，开启UDP通信功能。

**参与者：**用户

**执行者：**用户

**前置条件：**在Windows10操作系统上开启服务器

**事件流：**

1用户开启客户端。

2用户输入服务器端口到。

3用户输入服务器IP地址建立连接。

4连接成功后用户点解界面上的UDP按钮，开启UDP通信功能。

**后置条件：**客户端开启UDP通信功能，可以向服务器发送UDP消息。

## 3.5用例05：服务器端监听在线用户（MFC--05）

**用例描述：**服务器开启端口后进入监听模式，获取连接的用户并显示在用户栏。

**参与者：**用户

**执行者：**用户

**前置条件：**服务器开启端口，客户端连接成功。

**事件流：**

1.服务器开启端口。

2.客户连接服务武器。

3.服务器监听已连接客户端。

4.服务器获取客户端IP地址，已连接用户显示在用户栏。

**后置条件：**服务器获取连接用户，用户栏显示连接用户。

## 3.6用例06：服务器消息的群发（MFC--06）

**用例描述：**服务器在用户栏选择多个已连接用户并同时发送消息。

**参与者：**用户

**执行者：**用户

**前置条件：**有多个客户端连接，服务器选择多个连接用户。

**事件流：**

1.服务器在用户列表中选择多个人在线用户。

2.服务器端编辑消息并点击发送。

**后置条件**：多个客户端接收到服务器端发送的消息并显示在消息栏中。

## 3.7用例07：消息的发送与显示（MFC--07）

**用例描述：**各端口发送和接收消息并将其显示在消息栏上。

**参与者：**用户

**执行者：**用户

**前置条件：**客户端与服务器连接成功。

**事件流：**

1. 用户编辑消息点击发送并将消息显示在消息栏上。
2. 接收端接收消息并将其显示在消息栏上。

**后置条件：**各端口消息栏显示发送或接收的消息。

**用例图：**

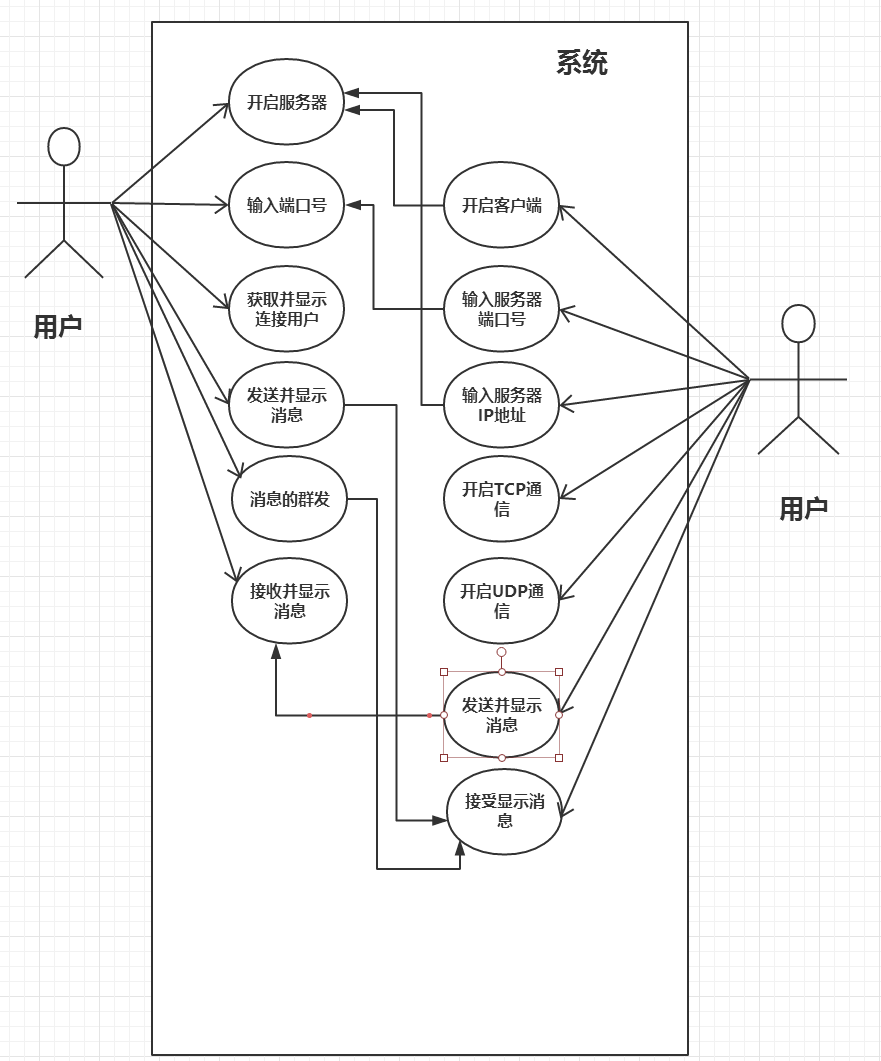


Figure 用例图

用户开启客户端，输入服务器端口号，连接服务器，即登录，在用户列表中找到要发送消息的用户点击消息编辑窗口进行消息的编辑，点击发送按钮进行消息的发送。

# 4性能需求

系统能够稳定的接收和发送消息，不会出现闪退和卡顿及系统崩溃的现象。

## 3.1时间特性要求

系统在处理时间上不能大于5秒，即发送客户端在输入消息点击发送后，接收端能在5秒前能接收到消息给出反馈。

## 3.1输入输出要求

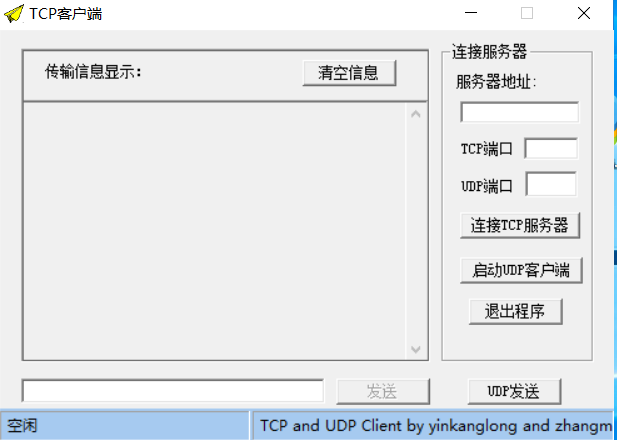
系统输入输出流畅且能在消息编辑框中更改输入的信息，消息接收显示器上显示的文本清楚且带有主机名和发送的时间。

# 5界面需求

## 5.1服务器端：



## 5.2客户端：



# 6其他非功能性需求

## 6.1.数据管理能力要求

系统数据能够保存，且只要用户不点击清楚信息按钮，消息便会持续保留在文本显示编辑框的缓冲内存中。

## 6.2 灵活性要求

系统能够在任何时间和地点使用，且随开随用，不受软件系统的限制。

系统运行速度快，能够即开即用，方便简洁。

# 7 尚未解决的问题

服务器不能先向客户端发送消息，未实现文件的传输。