图片包含 箭头

描述已自动生成

软件综合设计课程项目

VTalk聊天室

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 组长 | 组员1 | 组员2 | 组员3 | 组员4 | 组员5 |
| 学号 | 2018117211 | 2018117223 | 2018117286 | 2018117200 | 2018117191 | 2018117195 |
| 姓名 | 贺泓滔 | 乔凡星 | 蒙宁 | 谢元丰 | 冉弩 | 符泽林 |

**目 录**

[1引言 3](#_Toc66472011)

[1.1编写目的 3](#_Toc66472012)

[1.2背景 3](#_Toc66472013)

[1.3定义 4](#_Toc66472014)

[1.4参考资料 4](#_Toc66472015)

[2总体设计 4](#_Toc66472016)

[2.1需求规定 4](#_Toc66472017)

[2.2运行环境 5](#_Toc66472018)

[2.3基本设计概念和处理流程 5](#_Toc66472019)

[2.3.1 设计构思 5](#_Toc66472020)

[2.3.2 模块设计 6](#_Toc66472021)

[2.4结构 6](#_Toc66472022)

[2.5功能器求与程序的关系 7](#_Toc66472023)

[2.6人工处理过程 8](#_Toc66472024)

[2.7尚未解决的问题 8](#_Toc66472025)

[3接口设计 8](#_Toc66472026)

[3.1用户接口 8](#_Toc66472027)

[3.2外部接口 8](#_Toc66472028)

[3.3内部接口 8](#_Toc66472029)

[4运行设计 10](#_Toc66472030)

[4.1运行模块组合 10](#_Toc66472031)

[4.2运行控制 10](#_Toc66472032)

[4.3运行时间 11](#_Toc66472033)

[5系统数据结构设计 11](#_Toc66472034)

[5.1逻辑结构设计要点 11](#_Toc66472035)

[5.2物理结构设计要点 11](#_Toc66472036)

[5.3数据结构与程序的关系 12](#_Toc66472037)

[6系统出错处理设计 12](#_Toc66472038)

[6.1出错信息 12](#_Toc66472039)

[6.2补救措施 12](#_Toc66472040)

[6.3系统维护设计 13](#_Toc66472041)

# 1引言

## 1.1编写目的

基于互联网时代用户的通信需求，我们团队决定开发一款轻量级的多人聊天室软件，除基本的文字通信功能外，兼具自动回复、定时发送、文件传输等丰富的功能。本文档编制的目的是说明对软件系统的设计考虑，包括对软件系统的基本处理流程、总体设计、接口设计、运行设计、系统数据结构设计、系统出错处理设计等。

通过概要设计构建程序的体系结构，即确定程序的各大模块及模块之间的关系。通过概要设计全面分析用户需求，明确所要开发软件应具有的功能、性能与界面，首先寻找实现目标系统的各种不同的方案，从供选择的方案中选取若干个合理的方案，为每个合理的方案都准备一份系统流程图，列出组成系统的所有物理元素，进行成本和效益分析，从中选出一个最佳方案，最终对确定的最佳方案设计软件结构，通常需要进行必要的数据库设计，网络环境搭建，确定测试要求并制定测试计划。在此基础上进一步完成后续的开发工作。

## 1.2背景

待开发系统名称：VTalk多线程聊天室

任务提出者：开发团队

开发者：开发团队

用户：任何有通信需求的个人用户或团队

实现该软件的计算中心或计算机网络：该聊天平台是使用C++和QT开发，基于Windows平台下TCP/IP网络编程，并在Windows系统上运行的客户端服务端聊天系统。

## 1.3定义

SRS:Software Requirement Specification（软件需求说明）

DFD:Data Flow Diagram（数据流图）

DD:Data Dictionary（数据字典）

## 1.4参考资料

【1】需求概要设计文档格式标准（GB8567--88）

【2】王维波.栗宝鹃.候春望.Qt 5.9 c++开发指南.人民邮电出版社

# 2总体设计

## 2.1需求规定

本系统要达到以下目标：

1.用户登录

2.数据通信（一对多，服务端对客户端）

3.用户信息的数据库存储（账号密码等信息）

4.传文件操作

5.自动回复

6.定时发送

## 2.2运行环境

1.软件环境

|  |  |  |
| --- | --- | --- |
| 分类 | 名称 | 版本 |
| 操作系统 | Windows | 10.0 |
| 数据库平台 | Sql Server | 2017 |

2.硬件环境

|  |  |  |
| --- | --- | --- |
| 分类 | 最低配置 | 推荐配置 |
| 应用和数据库服务器 |  |  |
| 客户端 |  |  |

## 2.3基本设计概念和处理流程

### 2.3.1 设计构思

该部分主要表示系统功能层次。系统主要有用户登录、数据通信、传送文件、消息自动回复、定时发送五大模块。

### 2.3.2 模块设计

图示

描述已自动生成

## 2.4结构

本节将对系统特性作较为详细的描述，并给出系统特性结构图。

系统特性结构图给出系统特性在逻辑层面上相互之间的关系，该系统被划分为多个子系统，系统特性结构图既展现了整个系统的功能特性，同时也展示了子系统的功能特性。

图示

描述已自动生成

## 2.5功能器求与程序的关系

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：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 注册登录模块 | 聊天模块 | 用户信息模块 | 文件传输模块 | 自动操作模块 |
| 注册 | √ |  |  |  |  |
| 登录 | √ |  |  |  |  |
| 注销 | √ |  |  |  |  |
| 公聊 |  | √ |  |  |  |
| 私聊 |  | √ |  |  |  |
| 设置/修改资料 |  |  | √ |  |  |
| 聊天记录 |  |  | √ |  |  |
| 文件传送 |  |  |  | √ |  |
| 自动回复 |  |  |  |  | √ |
| 定时发送 |  |  |  |  | √ |

## 2.6人工处理过程

如果网络服务出现问题，需要开发人员手动配置IP地址和端口号。

## 2.7尚未解决的问题

目前项目处于稳步开发阶段，尚无未解决的问题。

# 3接口设计

## 3.1用户接口

本系统采用图形用户接口，以鼠标和键盘为用户接口，方便用户对系统数据的操作。

## 3.2外部接口

外部接口为鼠标、键盘等I/O设备，另需配置一台中心服务器，本系统与其他系统通过调用相同的数据库中的数据而保持一致性。

## 3.3内部接口

1.服务端

主体窗口类：

* 主体窗口初始化

- 窗口界面初始化

- 获取本地所有可用的IP地址

- IP地址监听

- 确定服务端IP地址和可用端口

- 服务端与首个客户端连接

- 保存客户端的IP地址和端口号并建立连接

- 服务端与新客户端连接（其他客户端）

- 保留原客户端IP地址和端口号

- 保存新客户端IP地址和端口号

- 断开与客户端的连接

- 逐个监听已连接的服务端并找出已关闭的客户端，断开连接

- 发送消息

- 确认客户端状态为可接收状态

- 发送消息并将焦点重定向到文本编辑框

- 接收消息

- 拆解数据包，识别对方身份并读取消息

- 关闭服务端

2.客户端

主体窗口类：

* 主体窗口初始化

- 窗口界面初始化

- 客户端与服务端建立连接

- 输入服务端IP地址和端口号

- 断开与服务端的连接

- 发送信息

- 发送消息并将焦点重定向到文本编辑框

- 接收消息

- 拆解数据包，读取消息

- 关闭客户端

# 4运行设计

## 4.1运行模块组合

共包括IP地址获取模块，服务器对客户端连接、断开模块，数据传输以及数据包装、解析模块。

登录模块：输入正确账号密码进入系统。

IP地址获取模块：获取本机地址和可用端口。

服务器模块：监听客户端，并在用户操作下接入接出。可将监听到的地址和端口号赋给新端口，并且监控客户端的状态。

数据模块：可对客户端的数据信息进行加工，然后解析。包括保存文件和消息记录等。

## 4.2运行控制

用户通过输入正确账号密码后，进入界面，使用鼠标点击获取其他功能并且可以结束使用。

## 4.3运行时间

各模块运行时间小于1秒。

# 5系统数据结构设计

## 5.1逻辑结构设计要点

用户表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 大小 | 约束 | 描述 |
| Userid | Int | 20 | 主键 | 用户id |
| Account number | Int | 20 | 非空 | 账号 |
| password | Int | 20 | 非空 | 密码 |
| nickname | Varchar | 20 | 非空 | 昵称 |
| remark | Int | 100 | 可空 | 备注 |

聊天记录表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 大小 | 约束 | 描述 |
| Logid | Int |  | 主键 | 记录id |
| Senderid | Varchar |  | 外键 | 发送人id |
| recieve | Inr |  | 外键 | 接受 |
| Send time | Varchar | 20 | 非空 | 时间 |
| Send content | Varchar | 200 | 非空 | 内容 |
| Remark | Varchar | 200 |  | 备注 |

## 5.2物理结构设计要点

在用户使用系统时应该有良好的登录界面，用户输入正确的账号密码方可登录。进入个人聊天面板，输入错误的账号密码则弹出对应框，提示退出或重新输入。

## 5.3数据结构与程序的关系

说明各个数据结构与访问这些数据结构的形式:

# 6系统出错处理设计

## 6.1出错信息

1.可能客户端无法与服务器连接，若不能正常连接，应该做出正确处理。

2.对不同的出错状况给出不同的出错信息，一般用对话框的形式提示。

## 6.2补救措施

说明故障出现后可能采取的变通措施，包括：

1. 后备技术说明准备采用的后备技术，当原始系统数据万一丢失时启用的副本的建立和启动的技术，例如周期性地把磁盘信息记录到磁带上去就是对于磁盘媒体的一种后备技术；
2. 降效技术说明准备采用的后备技术，使用另一个效率稍低的系统或方法来求得所需结果的某些部分，例如一个自动系统的降效技术可以是手工操作和数据的人工记录；
3. 恢复及再启动技术说明将使用的恢复再启动技术，使软件从故障点恢复执行或使软件从头开始重新运行的方法。

对出现的错误给出解决方式，引导客户重新输入或者退出重试。

## 6.3系统维护设计

此项目较小，没有特殊的维护设计。不过要求开发期间严格按照科学的方法开发软件，认真建立各种文档，同时在编码时给出详细的注释，方便对各个模块进行维护管理。