吉林大学

软件学院

《C++课程设计》

实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| **实验地点** | 计算机大楼实验室 | **座位号** | 38 |
| **班级** | **552203班** | | |
| **学号** | **55220331** | | |
| **姓名** | **陈驰水** | | |

2023-2024学年第1学期

|  |  |
| --- | --- |
| 实验项目 | **模拟即时通信系统实现** |
| 1. 设计任务分析   在整个项目中，我完全用了面向对象的编程范式，尽量遵守了面向对象的六大原则。我设计了合理的类和对象，实现了封装、继承和多态等特性。代码具有良好的安全性和复用性，易于维护和扩展。  我认为我的项目有以下几个显著优点：   1. 完全使用面向对象编程。详情可以见下面的类视图与uml图。 2. 类设计合理，尽可能满足面向对象六原则，有高内聚、低耦合、可维护和可扩展的特点。 3. 安全性高，通过遵循开闭原则和设计合理的类结构，减少错误和漏洞，从而提高系统的安全性。 4. 复用性强，大量的模块化和抽象，通过封装和继承等机制。我的类设计利用了虚函数和纯虚函数，来实现多态性和抽象性。减少重复开发的工作量。 5. 可维护性高，通过单一职责原则、开闭原则等，可以使代码结构清晰、简洁，易于理解和修改，降低维护成本。 6. 扩展性优化。在部分设计中预留了接口（有纯虚函数的抽象类）比如密码找回、文件发送功能等。其中通过接口隔离原则、依赖倒置原则等，实现可扩展的代码结构。当需求变化时，可以通过扩展而不是修改现有代码，提高系统的灵活性和可扩展性。 7. 代码质量较高、有明晰的注释。遵循面向对象的原则和合理的类设计可以提高代码的质量。通过减少耦合度、增加模块的独立性、提高代码的可读性和可测试性，可以降低代码中的bug和错误。 8. 服务端使用了单例设计模式，QSingleApplication类保证了本机服务器只可能产生单一的一个，保证了不会出现多个服务器进程同时运行，造成资源浪费或冲突的情况。可以方便地在不同的客户端之间共享服务器的数据和状态。 9. 程序有良好的鲁棒性。同时有着输入格式验证、异常处理、用户交互等提升鲁棒性的功能。 10. 我的类设计利用了构造函数和析构函数，来完成对象的初始化和销毁。构造函数负责为对象分配内存空间，并赋予对象初始状态。析构函数负责释放对象占用的内存空间，并进行一些清理工作。保证了内存空间不会泄露。 11. 使用本机搭建局域网服务器，实现了网络通讯功能。 12. 使用数据库（SQLite）存储，保证数据的安全性与独立性，并能处理大量数据。 13. 使用图形化套件（5.14.2MinGW套件），对程序进行了ui设计。   我实现的功能如下：   1. 用户可以通过独立的登录和注册界面进行身份验证，并且可以选择记住账号，方便下次登录。用户的账号信息和个人信息都存储在数据库中，保证了数据的安全性和一致性。 2. 用户可以管理自己的好友列表，添加、删除、修改和查询好友。用户可以通过ID或昵称在数据库中搜索其他用户，并发送好友请求。用户也可以设置自己的隐私状态，决定是否允许被其他用户搜索到。 3. 用户可以自定义自己的个人资料，包括头像、生日、性别、昵称、地区、签名等。这些信息都会同步到数据库中，并且可以被好友查看。用户也可以修改自己的个人资料，更新自己的信息。 4. 用户可以在局域网内进行网络通讯，发送和接收文本消息。用户也可以通过服务端监测自己的网络状态，查看是否有未读消息或者离线消息。 5. 用户可以创建和加入群聊，与多个用户同时交流。用户也可以管理群聊中的成员，邀请或踢出其他用户。 6. 我们的项目支持微信和QQ两种不同的设计风格，体现了差异化的特点。例如，微信可以用手机号登录，QQ只能用ID登录；微信只能搜索手机号添加好友，QQ可以搜索昵称添加好友等。 7. 我的项目提供了多个界面之间的跳转功能，使得用户可以方便地切换不同的功能模块。例如，用户可以退出登录，返回到登录界面。 | |
| 二、设计方案  我将主要以UML图来介绍我的设计方案。整体的框架如下：    （为保证架构相对整洁，此UML图删去了部分不太重要的成员函数和成员变量）  类视图如下：    下面开始具体解释类的构成   1. MainWindow作为主体     MainWindow类是程序的主体，在main函数中可以实例化此类。此类和大量类直接进行交互，和部分类通过数据库进行间接交互。  特别的，此类使用了多重继承。MainWindow从QMainWindow公有继承而来。之后公有继承出QFrame类，其中QFrame类包含一些有关图形化的函数，同样也方便类进行管理。之后继承出了QLabel、QPushButton、QWidget三个类分别代表文字、按钮、窗口。此外MainWindow与ui类为依赖关系，进行大量交互（使用现代化工具信号槽）。   1. 账号相关操作     对于账号管理我设计了这几个类：RegistCCS、PwdChangeCCS、loginCCS、loginFindCCS这四个类。其中前三个都是完善的功能，第四个预留了相关的接口（有纯虚函数的抽象类）以体现程序的可扩展性。账号管理的信息存放在数据库中，类分别进行访问以及修改。而数据库也是由类dataBaseCCS创建的。而数据库中的信息进一步和MainWindow交互，相当于有着弱依赖关系。  RegistCCS和loginCCS的关系更为紧密，属于面向对象中的关联关系。RegistCCS和loginCCS之间的关联关系是双向的，比如登录是可以选择进行注册，注册时需要通过登录类检测的其合法性。   1. 聊天功能以及群组功能     聊天功能的核心是messageChatCCS类，其与数据库中的信息交互，并与loginCCS类形成依赖关系，既聊天过程中会使用到登录类的成员函数。而groupCCS类的功能是设计群聊，其相对独立既不直接与数据库进行交互，而是通过messageChatCCS类影响数据。此外，groupCCS的函数也会影响messageChatCCS的成员变量，因此groupCCS类与messageChatCCS同样也为依赖关系。  此外我还设计了settingCCS、tellsCCS、newFriendCCS类分别用来实现更改个人资料、更改显示状态（即QQ和微信的差异功能）。这三个类都可以直接与MainWindows类进行交互，并影响数据。   1. 局域网服务端     局域网服务端是数据库以及聊天功能是实现内核，dataBaseCCS类和messageChatCCS类都与此进行交互。其中QSingleApplication类使用了单例设计模式保证了本机服务器只可能产生单一的一个，保证了不会出现多个服务器进程同时运行，造成资源浪费或冲突的情况。可以方便地在不同的客户端之间共享服务器的数据和状态。MySocketCCS、TcpServerCCS、serverThreadCCS三个类分别与QSingleApplication进行交互，相互为依赖关系。 | |
| 三、详细设计  1. 登录界面及其登录注册功能介绍  在登录过程中，去和数据库进行比较，成功失败都有对应的交互。同时有输入检测，比如不能输入过长或全部空格等。密码进行了隐藏，只显示位长。  特别的，我制作了可以通过账户检测头像的功能，从数据库中读取头像，并投放在窗口中。以及用一个单独的布尔变量存储是否记住密码的选项。同样存放在数据库中。  而注册功能与登录功能类似，同样进行了鲁棒性检测，注册成功后给与一个随机并独立的QQ号，可以通过此账号进行登录。  找回密码也进行了接口预留，制作了相关界面。  密码更改内容和注册类似，不在此赘述了。  2. 好友管理系统  用户可以管理自己的好友列表，添加、删除、修改和查询好友。用户可以通过ID或昵称在数据库中搜索其他用户，并发送好友请求。也可以设置自己的隐私状态，决定是否允许被其他用户搜索到。这意味着用户可以对自己的好友列表进行增删改查的操作。用户可以通过输入ID或昵称，使用模糊匹配的方式来查询数据库中的其他用户，并向他们发送好友请求的数据包。用户也可以通过设置一个布尔型的变量来表示自己的隐私状态，决定是否允许被其他用户搜索到。这样可以保护用户的个人信息和安全。  用户可以自定义自己的个人资料，包括头像、生日、性别、昵称、地区、签名等。这些信息都会同步到数据库中，并且可以被好友查看。用户也可以修改自己的个人资料，更新自己的信息。  3. 聊天功能  用户可以在局域网内进行网络通讯，发送和接收文本消息。用户也可以通过服务端监测自己的网络状态，查看是否有未读消息或者离线消息。这意味着用户可以使用TCP协议来实现客户端和服务端之间的双向通讯，使用MySocketCCS和TcpServerCCS类来建立和管理TCP连接，实现文本消息的发送和接收。  项目支持微信和QQ两种不同的设计风格，体现了差异化的特点。例如，微信可以用手机号登录，QQ只能用ID登录；微信只能搜索手机号添加好友，QQ可以搜索昵称添加好友等。这意味着我们的项目可以根据用户的选择来切换不同的风格和功能，使用StyleFactory类来设置应用程序的风格，并根据用户的选择来切换微信或QQ风格。我们还在搜索好友功能中添加了一个条件判断，根据用户选择的风格来决定是否允许搜索昵称或手机号。  用户可以管理群聊中的成员，邀请或踢出其他用户。用户可以通过输入群ID，从数据库中查找符合条件的群聊，就可以成为群成员。用户也可以自己创建一个新的群聊，并邀请自己的好友或其他用户加入。用户还可以管理员，对群聊中的成员进行管理，例如邀请或踢出某个成员。 | |
| 1. 总结与体会   我通过这次课程设计，深刻地理解了面向对象编程范式的思想和方法，掌握了面向对象六大原则的应用和意义，提高了自己的编程能力和水平。我认识到面向对象编程范式可以使代码更加清晰、简洁、易于理解和修改，也可以使代码更加模块化、抽象化、复用化和扩展化。我也认识到面向对象编程范式并不是万能的，也有一些局限性和缺点，例如可能导致过度设计、过度封装、过度继承等问题，也可能增加代码量和运行时间等开销。  我的课程设计任务可以让用户在同一网络内进行注册、登录、添加好友、修改个人资料、发送消息、创建群聊等功能。我遵循了面向对象的六大原则，设计了登录类、注册类、个人资料类、消息类、群聊类等多个类和对象，实现了封装、继承和多态等特性。我的代码具有高安全性、高复用性、高可维护性和高可扩展性等优点。我的课程设计还使用了SQLite数据库来存储用户数据，使用Qt图形化套件来设计用户界面，提高了系统的实用性和美观性。项目支持微信和QQ两种不同的设计风格，体现了差异化的特点。  我的课程设计过程中，我遇到了一些问题和挑战，例如如何实现用户之间的网络通讯，以及如何处理离线消息和未读消息的问题。我通过使用MySocketCCS和TcpServerCCS类来建立客户端和服务端之间的TCP连接，实现了数据的传输和接收。我还使用了数据库来存储用户的离线消息和未读消息，并在用户上线时发送给用户。有一些例如找回密码等设计没能完全完成，做成了接口体现可扩展性。  我通过这次课程设计，学习了数据库、图形化套件等相关技术，拓展了自己的知识面和技能范围，增加了自己的实践经验和创新能力。我认识到数据库可以使数据更加安全、独立、持久化，并能处理大量数据。我认识到图形化套件可以使界面更加美观、友好、交互式，并能支持多种风格。我也认识到这些技术都有自己的特点、优势和局限性，需要根据不同的需求和场景来选择合适的技术。  我通过这次课程设计，发现了自己在编程过程中遇到的问题和不足，反思了自己的解决方案和优化方法，总结了自己的收获和改进。我发现了自己在代码规范、注释文档、调试测试等方面还有一些不足，需要进一步改进和提高。我也发现了自己在面向对象编程范式的理解和应用上还有一些不够深入和全面的地方，需要进一步学习和探索。我也总结了自己在这次课程设计中的收获和成就，感到了自己的进步和成长。  在课程设计中，我对面向对象的理解有了更深的层次。我认识到面向对象不仅仅是一种编程范式，而是一种思维方式，一种对现实世界的抽象和模拟。面向对象让我从对象的角度来看待问题，而不是从过程的角度。面向对象让我关注对象的属性和行为，以及对象之间的关系和协作，而不是关注具体的实现细节。面向对象让我将复杂的问题分解为多个简单的模块，通过类和对象来封装数据和逻辑，通过继承和多态来实现代码的复用和扩展，通过接口和抽象类来实现代码的解耦和规范。面向对象让我能够更好地理解需求，更快地设计方案，更高效地编写代码，更容易地进行测试和维护。 | |