

# 简易 QQ 聊天系统 - 项目结构简介

## DAO 包简介

### 概述

DAO (Data Access Object) 包用于处理客户端与服务器之间的数据交互。该包通过定义接口和具体实现类, 提供了基于 HTTP 和 MySQL 的两种数据访问方式。DAO 包的设计符合面向接口编程的原则, 提高了代码的模块化和可扩展性。

## 文件结构

DAO 包的设计符合面向接口编程的原则，提高了代码的模块化和可扩展性。其结构如下所示：

```
DAO/
├── HttpConnector/ # http 请求连接控制类
├── DataBaseConnector/ # 基于 JDBC 的 MySQL 数据库请求连接控制类
├── user/ # 用户相关 DAO 类
│   ├── Imp/ # 接口类
│   ├── http/ # 基于 http 实现的数据访问类
│   └── MySQL/ # 基于 MySQL 实现的数据访问类
├── groups/ # 群组相关 DAO 类
│   ├── Imp/ # 接口类
│   ├── http/ # 基于 http 实现的数据访问类
│   └── MySQL/ # 基于 MySQL 实现的数据访问类
└── README.md
```

## users 和 groups 包

`users` 和 `groups` 实现了与用户及群组相关实体类对应的 DAO 类，提供了一个接口和基于 http 协议的和 JDBC - mySQL 数据库的资源请求方案。这两个包都由以下几个文件构成，基于接口回调可实现多态，适配不同的资源请求方式。

- **Imp 文件夹**：包含所有数据访问对象（DAO）的接口。这些接口定义了用户、私信、好友请求和好友关系等操作方法，为具体的实现提供了统一的规范。
- **http 文件夹**：借助 `HttpConnector` 实现了基于 HTTP 协议的 DAO。通过网络请求与服务器交互，实现对用户、私信等数据的远程访问和操作。
- **mySQL 文件夹**：借助 `DataBaseConnector` 实现了基于 MySQL 数据库的 DAO。直接与数据库交互，执行 SQL 操作，实现对数据的增删改查。

## users 包结构

```
user/
├── Imp/ # 接口类
│   ├── UserDAOImp.java
│   ├── PrivateMessageDAOImp.java
│   ├── FriendRequestDAOImp.java
│   └── FriendDAOImp.java
├── http/ # 基于 http 实现的数据访问类
│   ├── UserDAO.java
│   ├── PrivateMessageDAO.java
│   ├── FriendRequestDAO.java
│   └── FriendDAO.java
├── mySQL/ # 基于 mySQL 实现的数据访问类
│   ├── UserDAO.java
│   ├── PrivateMessageDAO.java
│   ├── FriendRequestDAO.java
│   └── FriendDAO.java
└── README.md
```

## groups 包结构

```
user/
├── Imp/ # 接口类
│   ├── GroupDAOImp.java
│   ├── GroupMessageDAOImp.java
│   ├── GroupJoinRequestDAOImp.java
│   └── GroupMemberDAOImp.java
├── http/ # 基于 http 实现的数据访问类
│   ├── GroupDAO.java
│   ├── GroupMessageDAO.java
│   ├── GroupJoinRequestDAO.java
│   └── GroupMemberDAO.java
├── mySQL/ # 基于 mySQL 实现的数据访问类
│   ├── GroupDAO.java
│   ├── GroupMessageDAO.java
│   ├── GroupJoinRequestDAO.java
│   └── GroupMemberDAO.java
└── README.md
```

## HttpConnector 和 DataBaseConnector 类

HttpConnector 和 DataBaseConnector 是两个用于集中控制资源请求方式及路径的连接类。其功能如下所示：

- **HttpConnector:**
  - HttpConnector 类用于发送 HTTP 请求到指定的服务器。它使用 Java 的 HttpClient 库来构建和发送请求，并处理响应。
  - **功能**
    - 同一管理 HTTP 请求的收发，管理连接的请求服务器
    - 构造 HTTP 请求，支持 JSON 格式的请求体。
    - 发送 POST 请求到指定的 URL。
    - 处理服务器返回的响应。

- **DataBaseConnector:**

- MySQLConnector 类用于连接 MySQL 数据库。它提供了一个静态方法 getConnection，该方法返回一个数据库连接对象。通过调用此方法，您可以轻松地连接到指定的 MySQL 数据库。
- **功能**
  - 同一管理控制资源请求数据库
  - 基于设置的 mySQL 数据库 URL 及账户信息创建数据库连接