

- [README](#)
  - 1. 项目介绍
  - 2. 技术栈
  - 3. 项目结构
  - 4. 项目部署
    - 4.1. 编译
    - 4.2. 运行服务器
    - 4.3. 客户端登入
  - 5. 一些测试用例
  - 6. 说明

# README

## 1. 项目介绍

项目名: SimpleWeChat (简易微信)

这是一个基于 C/S (Client/Server) 架构模型及网络套接字 (Socket) 编程的仿微信项目 (更准确的应该叫做 **大型聊天室**)。实现了 **用户的登入登出、私聊、群聊** 等功能。未来可能会加入 **好友管理、微信群建立、UI** 等功能。

## 2. 技术栈

- 技术
  - C 语言
  - Socket 编程
  - 多线程
- 开发工具
  - Visual Studio Code
  - gcc 8.1.0 (MinGW)

## 3. 项目结构

- SimpleWeChat
  - bin
    - **client.exe**: 客户端程序
    - **make.bat**: 用于编译项目的脚本
    - **server.exe**: 服务器端程序

- **doc**: 相关文档
- **src**: 源码根目录

源码结构:

- **entity**: 定义了用户结构体及相关操作
- **main**: 客户端与服务器端主程序及一些通用定义
- **socket**: 套接字相关操作及客户端与服务器端核心代码
- **utils**: 常用工具函数

## 4. 项目部署

### 4.1. 编译

1. 确保 `gcc` (v8.1.0) 已正确安装并配置环境变量。
2. 使用 `cmd` 打开项目根目录。
3. 使用命令 `cd bin` 切换目录。
4. 使用命令 `make` 编译项目。

### 4.2. 运行服务器

1. 使用 `cmd` 打开项目目录。
2. 使用命令 `cd bin` 切换目录。
3. 使用命令 `server <port>` 来启动服务器, 其中 `port` 是给服务器设置的端口号。例:

```
server 8888
```

### 4.3. 客户端登入

1. 使用 `cmd` 打开项目目录。
2. 使用命令 `cd bin` 切换目录。
3. 使用命令 `client <server-ip> <server-port>` 来启动客户端, 其中 `server-ip` 是服务器的 IP 地址, 如果是在同一台机器上运行, 则 `server-ip` 为 `127.0.0.1`, `server-port` 是服务器的端口号。例:

```
client 127.0.0.1 8888
```

## 5. 一些测试用例

用户名	密码
Edward	1234

用户名	密码
Winry	4321
Alphonse	5678
ZhangMay	8765

## 6. 说明

- 所有源码文件 `.h`、`.c` 均为 GBK 编码。
- 所有非源码文件 `.md`、`.bat` 均为 UTF-8 编码。