### **پروژه ششم**

سیدمحمدمهدی مصطفوی غمام

### **گزارش کامل پروژه چت‌روم چندمنظوره با قابلیت مدیریت نقش‌ها**

#### **1. معرفی پروژه**

این پروژه یک چت‌روم چندمنظوره مبتنی بر سوکت‌های پایتون است که امکان ارتباط همزمان بین چندین کاربر را فراهم می‌کند. چت‌روم دارای قابلیت‌های پیشرفته‌ای از جمله ارسال پیام خصوصی، مدیریت کاربران، نقش‌دهی به کاربران (مدیر، ناظر، کاربر عادی)، بلاک کردن کاربران و رمزنگاری پیام‌ها با الگوریتم سزار سایفر است.

#### **2. ویژگی‌های کلیدی**

1. **پیام‌های عمومی و خصوصی**:
   * کاربران می‌توانند به صورت عمومی پیام ارسال کنند.
   * با فرمت @username message می‌توان پیام خصوصی ارسال کرد.
2. **مدیریت نقش‌ها**:
   * نقش‌های تعریف‌شده:
     + **مدیر (Admin):** دسترسی کامل به مدیریت کاربران و تغییر نقش‌ها.
     + **ناظر (Moderator):** دسترسی به حذف کاربران و بلاک کردن آنها.
     + **کاربر عادی (User):** فقط می‌تواند پیام‌های عمومی و خصوصی ارسال کند.
   * دستورات مدیریتی:
     + setrole username role: تغییر نقش کاربران (فقط برای مدیر).
     + kick username: حذف کاربر از چت‌روم (مدیر و ناظر).
     + block username: بلاک کردن یک کاربر (عدم دریافت پیام خصوصی از او).
     + unblock username: برداشتن بلاک کاربر.
3. **رمزنگاری پیام‌ها**:
   * پیام‌ها با استفاده از الگوریتم سزار سایفر رمزنگاری و رمزگشایی می‌شوند.
   * سرور پیام‌ها را به صورت رمزنگاری‌شده ارسال می‌کند و رمزگشایی در کلاینت انجام می‌شود.
4. **مدیریت اتصالات**:
   * کاربران هنگام ورود باید نام خود را وارد کنند.
   * نام کاربران و نقش‌های آنها در سرور ذخیره می‌شود.
5. **بلاک کردن کاربران**:
   * کاربران می‌توانند کاربران خاصی را بلاک کنند و پیام خصوصی از آنها دریافت نکنند.
6. **خروج از چت‌روم**:
   * با وارد کردن دستور quit کاربران می‌توانند از چت‌روم خارج شوند.

#### **3. پیش‌نیازها**

برای اجرای این پروژه به موارد زیر نیاز است:

1. **سیستم عامل**:
   * ویندوز، مک یا لینوکس
2. **پایتون**:
   * نسخه 3.6 یا بالاتر
3. **کتابخانه‌های مورد نیاز**:
   * کتابخانه‌های استاندارد پایتون:
     + socket
     + threading

#### **4. راه‌اندازی و نصب**

1. **دانلود پروژه**:
   * فایل‌های server.py و client.py را دانلود کنید.
2. **اجرای سرور**:
3. ابتدا فایل server.py را اجرا کنید:  
    python3 server.py
4. پیام زیر نمایش داده خواهد شد:  
    Server is running and listening for connections...
5. **اجرای کلاینت**:
6. هر کاربر فایل client.py را اجرا کند:  
    python client.py
   * کاربران باید نام خود را وارد کنند و پس از اتصال پیام خوش‌آمدگویی را دریافت خواهند کرد.

#### **5. نحوه استفاده**

1. **ورود به چت‌روم**:
   1. پس از اجرای کلاینت، نام خود را وارد کنید. به عنوان مثال:  
       Enter your name: Alice
   2. سرور پیام خوش‌آمدگویی و نقش کاربر را ارسال می‌کند:  
       Welcome Alice! Your role is Admin. Type 'quit' to exit. Type 'help' for commands.
2. **ارسال پیام عمومی**:
   1. پیام خود را تایپ کرده و اینتر بزنید. پیام به تمام کاربران ارسال خواهد شد.  
       Hello everyone!
3. **ارسال پیام خصوصی**:
   1. پیام خصوصی را با فرمت @username message ارسال کنید:  
      ? Bob Hi Bob, how are you@
4. **دستورات مدیریتی** (فقط برای مدیر و ناظر):
   1. تغییر نقش:  
       setrole Bob Moderator
   2. حذف کاربر:  
       kick Charlie
   3. بلاک کردن کاربر:  
       block Charlie
   4. آنبلاک کردن کاربر:  
       unblock Charlie
5. **خروج از چت‌روم**:
   1. دستور quit را وارد کنید:  
       quit

#### **6. دستورات کلیدی**

|  |  |
| --- | --- |
| **دستور** | **توضیحات** |
| quit | خروج از چت‌روم |
| @username message | ارسال پیام خصوصی به یک کاربر |
| setrole username role | تغییر نقش یک کاربر (فقط برای مدیران) |
| kick username | حذف کاربر از چت‌روم (فقط برای مدیر و ناظر) |
| block username | بلاک کردن یک کاربر (عدم دریافت پیام خصوصی از او) |
| unblock username | برداشتن بلاک یک کاربر |

#### **7. معماری و توضیحات فنی**

1. **ارتباط کلاینت و سرور**:
   * ارتباط بین کلاینت‌ها و سرور با استفاده از سوکت‌های TCP برقرار می‌شود.
   * سرور به صورت چندنخی (Multithreaded) طراحی شده است تا بتواند چندین اتصال همزمان را مدیریت کند.
2. **مدیریت نقش‌ها**:
   * نقش هر کاربر در یک دیکشنری (roles) ذخیره می‌شود.
   * سرور دسترسی کاربران به دستورات خاص را با توجه به نقش آنها بررسی می‌کند.
3. **رمزنگاری و رمزگشایی**:
   * کلاینت‌ها پیام‌ها را قبل از ارسال رمزنگاری می‌کنند.
   * پیام‌های دریافتی از سرور در کلاینت رمزگشایی می‌شوند.
   * سرور پیام‌ها را بدون هیچ تغییری به کاربران ارسال می‌کند.
4. **مدیریت بلاک کاربران**:
   * سرور لیست کاربران بلاک‌شده را برای هر کاربر ذخیره می‌کند.
   * پیام‌های خصوصی از کاربران بلاک‌شده ارسال نمی‌شود.

#### 

#### 

#### **8. مزایا و قابلیت گسترش**

1. **امنیت**:
   * رمزنگاری پیام‌ها امنیت ارتباطات را افزایش می‌دهد.
2. **انعطاف‌پذیری**:
   * نقش‌های قابل تعریف و مدیریت کاربران قابلیت گسترش پروژه را فراهم می‌کند.
3. **قابلیت توسعه**:
   * می‌توان قابلیت‌هایی مانند ذخیره پیام‌ها، احراز هویت کاربران، یا اتصال به پایگاه داده را به پروژه اضافه کرد.