

## فایل های پروژه

برای اجرای این پروژه نیاز به ساخت دو فایل پایتون داریم یکی برای مدیریت `peer` ها و یکی برای `STUN` Server.

در `peer` از کلاس به همان نام استفاده میکنیم که متدهای `connect` (متصل کردن دو سوکت) و `Listen` (اجرا در نقش سرور برای وقتی میخواهد فایلی را دریافت کند) و `send_image` (تابع برای ارسال عکس) و `receive_image` (تابع برای دریافت عکس) و `send_data` (تابع برای ارسال متن یا فایل تکست) و `receive_data` () و `start` () و تابع `main` (تعریف کاربرها و آغاز به کار آن ها) استفاده شده است.

در `Server` از کلاس `BaseHTTPRequestHandler` استفاده شده که از متدهای `do_GET` (پاسخ دهی به درخواست های `get` که برای دریافت اطلاعات یک یا کل کاربران استفاده میشود) و `do_POST` (دادن نام کاربری و `IP` به کاربر جدید) و تابع `main` (ساخت دیتا بیس ردیس و آغاز به کار سرور) استفاده میکنند.

## کتابخانه های مورد استفاده

Socket : ساخت سوکت TCP و UDP و اتصال سوکت ها به هم

Threading : هندل کردن درخواست های مکرر برای دریافت فایل به یک peer یا ارسال فایل از یک peer با استفاده از multi-threading

Redis : دیتابیس مورد استفاده در سرور استان برای ذخیره کردن جفت های key-value که در اینجا نام کاربری و آدرس IP هر peer است.

http.server : یک سرور http است که در رواقع همان سرور استان ماست و وظیفه هندل کردن درخواست های GET و POST از سمت peer ها را دارد.

PIL : نمایش عکس و کار کردن با فایل عکس

Numpy : کار با آرایه و به خصوص تبدیل عکس به آرایه و برعکس برای انتقال فایل عکس

Json : مورد استفاده برای داده انتقالی بین سرور و کاربر

Requests : ارسال درخواست از کاربر به سرور

Pysimplegui : برای رابط گرافیکی ساده برای نمایش peer ها