

با نام برنامه نویس هستی



دانشگاه صنعتی شریف

دانشکده مهندسی کامپیوتر

# مستندات و توضیحات مربوط به پروژه پیام رسان

گردآورنده : علیرضا حسین خانی

شماره دانشجویی : ۹۸۱۰۱۴۰۳

استاد: سرکار خانم دکتر بقولی زاده

درس: مبانی برنامه سازی

بهمن ۹۸

## الف – کلاینت

توضیحات کلی : در تابع main پس از کدهای مربوط به راه اندازی که از فایل های کارگاه شیکه کپی شدند! به دو حلقه همواره درست تودرتو می رسیم که با while پیاده سازی شده اند و در حقیقت وظیفه شان بررسی وضعیت برنامه (اکنون در کدام قسمت هستیم؛ محیط ورود و خروج کاربری، محیط ورود به کانال ها یا محیط داخل کانال یعنی ارسال پیام و...) می باشد که به کمک یک متغیر با نام state این امر انجام می شود و تابع مربوطه فراخوانی می شود. (الان که فکر میکنم میشد با یه while هم جمعش کرد!!).

### توابع مهم:

۱ - تابع ساخت سوکت که برای هربار اتصال با سرور فراخوانی میشود. این تابع از یک متغیر گلوبال اسفاده می کند تا بعد از اتصال امکان ارسال و دریافت پیام به کمک تابع send از طریق آن متغیر برای همه توابع فراهم باشد.

۲ - تابع login\_menu که مربوط به محیط ورود به و خروج از حساب کاربری است. از یک حلقه while و متغیر state برای چک کردن وضعیت استفاده میکند. این تابع با دریافت نوع عملیات (login یا register) و سپس نام کاربری و گذرواژه از کاربر و چسباندن آن دو به هم طبق پروتکل سرور، آن را برای سرور ارسال می کند؛ سپس پاسخ را دریافت نموده و پیغام موفقیت یا عدم آن را چاپ میکند. همچنین در صورت موفقیت آمیز بودن ورود، state را برابر ۲ میکند و break و برنامه از این تابع خارج و در main تابع بعدی فراخوانی میشود.

۳ - تابع karbari\_menu که پس از ورود موفق فراخوانی میشود. مشابه تابع قبل از یک حلقه while و متغیر state برای چک کردن وضعیت استفاده میکند. اگر دستور logout باشد و سرور پیام موفقیت آمیز بودن را ارسال کند state را برابر ۱ میکند یعنی برگشت به تابع قبل و... در غیر این صورت نام کانال موردنظر برای ملحق شدن یا ساخته شدن را از کاربر دریافت می کند؛ اطلاعات را برای سرور ارسال و نتیجه را چاپ میکند. در صورت موفقیت آمیز بودن، state را برابر ۳ میکند یعنی ورود به تابع بعد و... .

۴ - تابع chat\_menu که پس از ملحق شدن یا ساختن یک کانال فراخوانی میشود. مشابه تابع قبل از یک حلقه while و متغیر state برای چک کردن وضعیت استفاده میکند. مشابه توابع قبل، دستور را از کاربر میگیرد، به سرور ارسال میکند، پاسخ را دریافت و تفسیر میکند، بنا به نیاز متغیر state را تغییر میدهد.

نکته: آرایه های `global_username` و `global_channel_name` و `user_token` به صورت گلوبال تعریف شده اند تا در صورت نیاز (مثلا برای چاپ نام کاربری هنگام ورود و خروج موفق) همه جا در دسترس باشند و مثلا نیازی به هربار ارسال توکن کاربر به توابع نباشد.

## ب - سرور

توضیحات کلی: بعد از کد های راه اندازی که از کدهای کارگاه شبکه کپی شد! به یک حلقه `while` همواره درست میرسیم که در آن پیام از کاربر در آرایه `buffer` دریافت میشود، سپس دستور مدنظر کاربر از `buffer` در آرایه `command` کپی میشود و بنا به دستور توابع مربوطه فراخوانی میشوند. به غیر از دستور `logout`، `leave` و `join` تقریبا همه اعمال مربوط به هر دستور درون تابع خود انجام میشود (این سه تا حواسم نبود از دستم در رفت!).

### توابع مهم:

۱ - `make_json_error` که برای ساخت رشته `error` مربوطه استفاده میشود و متن ارور را دریافت میکند و یک پوینتر به رشته ای که ارور در آن جیسون شده برمیگرداند. (در فاز ۳ یعنی بدون `json` بجای استفاده از این تابع از رشته های آماده که گلوبال تعریف شده اند استفاده شده است)

۲- تابع `listen_accept` که برای هربار دریافت دستور از کاربر در ابتدای حلقه `while` در تابع `main` فراخوانی میشود. کدها از کارگاه شبکه کپی شده اند.

۳- توابع `make_json_error` و `make_json_successful` که برای ساخت رشته های جیسون مربوط به ارورها و پیام موفقیت استفاده میشود. تابع مربوط به ارور ورودی اش رشته حاوی نوع ارور است و تابع مربوط به موفقیت ورودی نمیگیرد (چون همیشه یک شکل واحد دارد)

۴ - توابع `check_tekrari_channelname` و `check_tekrari_username` که رشته نام کانال یا یوزرنیم را بعنوان ورودی میگیرند و بنا بر اینکه تکراری هست یا نیست ۱- و ۰ (به ترتیب) برمیگردانند. سازوکار این توابع به کمک تابع `fopen` در حالت "r" میباشد. اگر خروجی اش `NULL` باشد یعنی فایل کاربر یا کانال در دایرکتوری موجود نیست و این یعنی قبلا ساخته نشده و تکراری نیست.

۵ - توابع `register_user` و `login_user` همانطور که از نامشان پیداست وظیفه ثبت و ورود کاربر را بعهده دارند. در صورت ثبت نام موفق، فایل مربوط به کاربر با نام نام کاربری و پسوند `u`. در دایرکتوری فایل سرور ساخته میشود و پسورد کاربر در آن ذخیره میشود. در صورت لاگین موفق در ساختاری به نام `online_list` که

گلوبال میباشد، نام کاربری و توکن تولید شده کاربر ذخیره میشود. تعداد پیامهای خوانده شده و نام کانال نیز در این ساختار برای هر کاربر ذخیره میشود. تعداد کاربران آنلاین و جایگاه ذخیره اطلاعات کاربر جدید در `online_list` به کمک متغیر `list_counter` تعیین میشود.

۶ - تابع `token_make` یک توکن ۳۲ کاراکتری تولید میکند که در آن از تابع `rand` استفاده شده است. البته با دادن `seed` به آن میتوان خاصیت شبه تکراری بودن توکن ها را از بین برد. (اینکه کاربر اول بعد از ورود همیشه توکن یکسانی دریافت میکند و به همین ترتیب سایر کاربران)

۷ - تابع `create_channel` که فرایند آن شبیه `register_user` است با این تفاوت که فایل کانال ها را با پسوند `.c` و داخل فایل پیام های ارسال شده را به صورت جیسون ذخیره میکند. همچنین در ساختار `online_list` نام کانال کاربر سازنده را تغییر میدهد.

۸ - تابع فوق العاده کاربردی `token_check` که وظیفه اش چک کردن صحت توکن کاربر فرستنده پیام از کلاینت است و روی ساختار `online_list` سیر میکند و در صورت وجود توکن، شماره کاربر مربوطه و درغیر آن ۱- برمیگرداند. به کمک این تابع میتوان به همه اطلاعات کاربر آنلاین که در `online_list` ذخیره شده است، دسترسی داشت.

۹ - تابع `channel_members` ساز و کار تابع تقریباً مشخص است. با سیرکردن روی `online_list` و بررسی تطابق نام کانال ثبت شده برای کاربران با نام کانال فرستنده درخواست، لیست اعضا را تهیه میکند و برای کلاینت ارسال میکند.

۱۰ - آخرین تابع مهم که ممکن است کمی گنگ باشد تابع `refresh` میباشد. این تابع به کمک متغیر `read_messages` که در `online_list` برای هر کاربر ذخیره شده، ابتدا یک اشاره گر را تا تعداد پیام خوانده شده در کانال جلو می آورد، سپس لیست پیام ها را تهیه کرده و برای کلاینت ارسال میکند. همچنین تعداد پیام های ارسال شده را به `read_messages` هر کاربر اضافه میکند.