

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

موضوع:

Chat room

استاد:

دکتر زمانی فر

طراح:

جمشید مظفری

در این مقاله پروژه درس سیستم های توزیع شده را توضیح خواهیم داد. در ابتدا در مورد ویژگی های این پروژه توضیحاتی را ارائه می دهیم.

این پروژه شامل دو نرم افزار مختلف به نام های Server و Client می باشد. نرم افزار Server روی سایت خدمتگذار قرار می گیرد و نرم افزار Client روی سایت هایی که قصد چت کردن با یکدیگر دارند قرار می گیرد.

ویژگی ها:

۱. از ارتباطات پیام گرا ناپایدار مانند سوکت برکلی شده است.
۲. هر شخص در ابتدا باید توسط سرور تاییدیه دریافت کند.
۳. امکانات بلاک کردن و رد صلاحیت کاربر وجود دارد.
۴. امکان ارسال پیام به صورت صوتی و متنی وجود دارد.
۵. پیام صوتی به صورت جویبارگرا می باشد.
۶. امکان استفاده از شکلک ها در متن نیز وجود دارد.

توضیحات کلی پروژه:

این پروژه از کلاس های مختلفی تشکیل شده است که بعضی از آنها بین مشتری و خدمتگذار مشترک می باشد و برخی دیگر به صورت مستقل می باشند. در ابتدا کلیات هر کلاس را در زیر توضیح خواهیم داد سپس در بخش بعد در مورد وظیفه هر تابع توضیحاتی مختصر را ارائه می دهیم.

نرم افزار Server:

این نرم افزار باید در سایت خدمتگذار اجرا شود و وظیفه انتقال پیام ها را دارد. ابتدا پیام ها را از مشتری - های مختلف دریافت می کند پس از اعمال قواعد متنوع، به مشتری موردنظر ارسال می کند.

۱. Info Class: اطلاعات مرتبط با حساب کاربری هر مشتری را نگه داری می کند.
۲. MailBox: پیام هایی که بین مشتریان و سرور جابجا می شوند شی ای از این کلاس می باشند.

۳. OnlineUser: اطلاعات مربوط به هر کاربر آنلین را در خود نگه می‌دارد.

۴. Server: پنجره مرتبط با کارگزار را پیاده‌سازی کرده است، همچنین تمام توابع مرتبط با ارسال و دریافت پیام‌ها و ... را در این کلاس پیاده‌سازی شده است.

نرم افزار Client:

۱. MailBox: پیام‌هایی که بین مشتریان و سرور جابجا می‌شوند شیء ای از این کلاس می‌باشند.

۲. OnlineUser: اطلاعات مربوط به هر کاربر آنلین را در خود نگه می‌دارد.

۳. Login: پنجره مرتبط با ورود کاربری می‌باشد. کاربر اطلاعات کاربری خود را وارد می‌کند و سپس در صورت صحیح بودن اطلاعات اجازه استفاده از سرور را به کاربر می‌دهد.

۴. Client: پنجره مرتبط با مشتری را پیاده‌سازی کرده است، که تمامی توابع مربوط به ارسال و ... را پیاده‌سازی کرده است.

توضیحات مربوط به نرم افزار Server:

کلاس Info Class:

این کلاس اطلاعات مرتبط با هر کاربر را نگه می‌دارد. اعضای این کلاس شامل موارد زیر می‌باشد:

۱. Password: رمز برای ورود کاربر را نگه داری می‌کند.

۲. Accept: مقدار مربوط به تاییدیه سرور را نگه‌داری می‌کند.

۳. Block: مقدار مربوط به بلاک شدن کاربر توسط سرور را نگه‌داری می‌کند.

۴. smsBlackList: لیست کاربرانی که توسط کاربر برای ارسال پیام متنی بلاک شده اند.

۵. voiceBlackList: لیست کاربرانی که توسط کاربر برای ارسال پیام صوتی بلاک شده اند.

کلاس MailBox:

این کلاس اطلاعات مربوط به پیام‌ها را نگه‌داری می‌کند. اعضای این کلاس شامل موارد زیر می‌باشد:

۱. Username: نام کاربری مشتری ارسال کننده را نگه می‌دارد.

۲. Title: عنوان پیام را مشخص می‌کند. عنوان پیام نوع عکس العمل سرور و مشتری را در قبال محتوای پیام مشخص می‌کند.

۳. Info: محتویات پیام را نگه می‌دارد.

کلاس OnlineUser:

این کلاس اطلاعات مرتبط با کاربرانی که آنلاین هستند و حق ارسال پیام را دارند نگهداری می‌کند. اعضای این کلاس شامل موارد زیر می‌باشند:

۱. Username: نام کاربری کاربر آنلاین را نگهداری می‌کند.

۲. smsBlock: مقدار مرتبط با بلاک بودن ارسال پیام متنی کاربر توسط سرور را نگهداری می‌کند.

۳. voiceBlock: مقدار مرتبط با بلاک بودن ارسال پیام صوتی کاربر توسط سرور را نگهداری می‌کند.

کلاس Server:

این کلاس، کلاس اصلی برنامه می‌باشد که از توابع و بخش‌های مختلفی تشکیل شده است. از اعضای این کلاس می‌توان به موارد زیر اشاره کرد:

۱. Smiles: کد مربوط و متن مشخص کننده شکلک‌ها را نگهداری می‌کند.

۲. Path: آدرس فایل users.inf را که اطلاعات کاربران ثبت نام شده را نگهداری می‌کند، ذخیره می‌کند.

۳. Map: اطلاعاتی که از users.inf خوانده شده در این دیکشنری قرار می‌گیرد.

۴. SelectedUser: برای نگهداری کاربر انتخاب شده در لیست کاربران استفاده می‌شود.

۵. OnlineUsers: اطلاعات کاربران آنلاین را نگهداری می‌کند.

تابع formWindowOpen:

این تابع ابتدا یک نخ برای دریافت درخواست‌ها برای ثبت نام و ورود کاربران اجرا می‌کند. سپس شکلک‌ها را بارگذاری می‌کند و لیست کاربران را بارگذاری می‌کند.

تابع chk_acceptActionPerformed:

این تابع تاییدیه کاربر را مشخص می‌کند. با غیر فعال شدن این مقدار کاربر موردنظر از لیست OnlineUsers حذف می‌شود.

تابع chk_blockActionPerformed:

این تابع کاربر انتخاب شده را بلاک می‌کند و آن را از لیست OnlineUsers حذف می‌کند. سپس تغییرات را در Users.inf ذخیره می‌کند.

تابع btn_SendActionPerformed:

این تابع ابتدا یک پیام با عنوان Message تولید می‌کند و سپس محتوای آن را مقداردهی می‌کند و عنوان فرستنده را نیز با SERVER مشخص می‌کند. پس از ارسال پیام متن آن را در log پنجره نمایش می‌دهد.

تابع RefreshList:

این تابع ابتدا اطلاعات users.inf را بارگذاری می‌کند. سپس با توجه به اطلاعات موجود لیست کاربران و همچنین وضعیت بلاک بودن و ... آنها را مشخص می‌کند و لیست را تولید می‌کند. سپس برای هر یک کاربران آنلاین یک لیست کاربران آنلاین را ارسال می‌کند در حالی که مشخص شده است که این افراد جزء افراد بلاک شده و ... می‌باشند یا خیر.

تابع radioButtonClick:

این تابع با انتخاب هر کاربر از لیست اطلاعات آن را استخراج کرده و وضعیت تاییدیه و بلاک شدن را مشخص می‌کند.

تابع Login:

این تابع اطلاعات کاربری فرستاده شده توسط مشتری را استخراج می‌کند و با توجه به تایید و صحیح بودن اطلاعات پیامی مناسب برای مشتری آماده می‌کند و آن را به عنوان پاسخ به سمت مشتری ارسال می‌کند.

تابع Signup:

این تابع اطلاعات مربوط به ثبت نام کاربر را دریافت می کند و در صورت عدم وجود اشکالی، اطلاعات جدید را در فایل `users.inf` ذخیره می کند و پیامی مناسب را به سمت مشتری ارسال می کند.

تابع SmsBlock:

این تابع کاربر انتخاب شده را برای ارسال پیام متنی بلاک می کند و مجوز ارسال پیام به هیچ مشتری را نمی دهد.

تابع VoiceBlock:

این تابع کاربر انتخاب شده را برای ارسال پیام صوتی بلاک می کند و مجوز ارسال پیام به هیچ مشتری را نمی دهد.

تابع Message:

این تابع پیام را از مشتری دریافت می کند و در صورت داشتن مجوزات و بلاک نبودن توسط کاربر دریافت کننده، پیام را به مشتری دریافت کننده ارسال می کند.

تابع Voice:

این تابع پیام صوتی را به صورت جویباری دریافت می کند و در صورت داشتن مجوزات لازم آن را به مشتری دریافت کننده ارسال می کند.

تابع InsertString:

این تابع پیام ارسالی را در قسمت `log` چاپ می کند. وظیفه این تابع تبدیل کدهای شکلک ها به عکس شکلک ها و همچنین نوشتن برخی بخش های پیام ها به صورت رنگی می باشد.

تابع IndexOfSmile:

این تابع اندیس معادل با کد یک شکلک را برمی گرداند.

تابع LoadUserFile:

این تابع محتویات فایل users.inf را در داخل Map بارگذاری می‌کند. این اطلاعات مربوط به اطلاعات کاربران ثبت نام شده می‌باشد.

تابع SaveUserFile:

این تابع محتویات مربوط به کاربران را که در Map ذخیره شده است در فایل users.inf می‌نویسد و آن را ذخیره می‌کند.

کلاس SmileActionListener:

این کلاس مشخص می‌کند که در هنگام انتخاب هر یک از شکلک‌ها چه عکس العملی نشان داده شود. وظیفه تابع اصلی این کلاس اضافه کردن کد مرتبط با شکلک به متن می‌باشد.

کلاس ServerThread:

این کلاس یک نخ می‌باشد که در ابتدای شروع نرم افزار اجرا می‌شود. وظیفه این نخ گوش دادن از طریق یک پورت برای برقراری اتصال از طرف مشتری می‌باشد. پس از برقراری اتصال یک نخ جداگانه به هر مشتری اختصاص می‌دهد.

کلاس SocketClient:

این کلاس به ازای هر مشتری به صورت جداگانه ساخته می‌شود. وظیفه این کلاس این است که هر پیام که فرستاده می‌شود توسط این نخ دریافت شود و با توجه به عنوان پیام نسبت به آن عکس العمل مناسب انجام دهد. از انواع عنوان‌های مختلف می‌توان به موارد زیر اشاره کرد:

۱. Login: برای صدور مجوز برای اتصال و برقراری ارتباط مشتری با خدمتگزار می‌باشد.

۲. Signup: برای ثبت نام مشتری به کار می‌رود.

۳. smsBlock: برای بلاک کردن پیام متنی برای یک مشتری به کار می‌رود.

۴. voiceBlock: برای بلاک کردن پیام صوتی برای یک مشتری به کار می‌رود.

۵. Message: برای دریافت پیام متنی از طرف مشتری به کار می‌رود.

۶. Voice: برای دریافت پیام صوتی به صورت جویباری از طرف مشتری به کار می‌رود.

توضیحات مربوط به نرم افزار Client:

کلاس MailBox:

این کلاس اطلاعات مربوط به پیام‌ها را نگهداری می‌کند. اعضای این کلاس شامل موارد زیر می‌باشد:

۴. Username: نام کاربری مشتری ارسال کننده را نگه می‌دارد.

۵. Title: عنوان پیام را مشخص می‌کند. عنوان پیام نوع عکس العمل سرور و مشتری را در قبال محتوای پیام مشخص می‌کند.

۶. Info: محتویات پیام را نگه می‌دارد.

کلاس OnlineUser:

این کلاس اطلاعات مرتبط با کاربرانی که آنلاین هستند نگهداری می‌کند. اعضای این کلاس شامل موارد زیر می‌باشند:

۴. Username: نام کاربری کاربر آنلاین را نگهداری می‌کند.

۵. smsBlock: مقدار مرتبط با بلاک بودن ارسال پیام متنی کاربر توسط سرور را نگهداری می‌کند.

۶. voiceBlock: مقدار مرتبط با بلاک بودن ارسال پیام صوتی کاربر توسط سرور را نگهداری می‌کند.

کلاس Login:

این کلاس پنجره مربوط به دریافت اطلاعات از طرف مشتری برای اجازه ورود به محیط چت روم می‌باشد.

تابع Login:

این تابع سازنده کلاس Login می‌باشد که دو پنجره را برای دریافت ip و port سرور نمایش می‌دهد.

تابع btn_LoginActionPerformed:

این تابع اطلاعات مشتری را از ورودی خوانده و آن را به سمت سرور ارسال می‌کند و منتظر می‌ماند تا پاسخ درخواست از طرف سرور ارسال شود. در صورت دریافت مجوز پنجره Client را نمایش می‌دهد.

تابع btn_signupActionPerformed:

این تابع اطلاعات مشتری را برای ثبت نام به سرور ارسال می کند و منتظر پاسخ می ماند. در صورت عدم اشکال کاربر جدید ثبت نام می شود ولی باید منتظر بماند تا از طرف سرور تاییدیه دریافت کند سپس اجازه ورود به محیط چت را دارا خواهد بود.

کلاس Client:

این کلاس، کلاس اصلی برنامه می باشد که از توابع و بخش های مختلفی تشکیل شده است. از اعضای این کلاس می توان به موارد زیر اشاره کرد:

۱. Smiles: کد مربوط و متن مشخص کننده شکلک ها را نگهداری می کند.
۲. OIS: ارتباط جویبارگرا برای دریافت اطلاعات از طرف سرور نگهداری می کند.
۳. OOS: ارتباط جویبارگرا برای ارسال اطلاعات به طرف سرور نگهداری می کند.
۴. SelectedUser: برای نگهداری کاربر انتخاب شده در لیست کاربران استفاده می شود.
۵. Username: نام کاربری کاربر فعلی را نگهداری می کند.
۶. OnlineUsers: اطلاعات کاربران آنلاین را نگهداری می کند.
۷. SLine: برای پخش و ضبط پیام های صوتی مورد استفاده قرار می گیرد.

تابع formWindowOpen:

این تابع ابتدا یک نخ برای دریافت پیام ها از طرف سرور اجرا می کند. سپس شکلک ها را بارگذاری می کند.

تابع chk_blockSmsActionPerformed:

این تابع پیامی را به سمت سرور ارسال می کند که باعث می شود که کاربر انتخابی بلاک شود و پیام های آن از آن پس برای این مشتری ارسال نشوند.

تابع btn_sendActionPerformed:

این تابع پیامی متنی را آماده کرده و آن را به سمت سرور ارسال می کند و سرور با توجه به اطلاعات درون پیام آن را به سمت مشتری مشخص شده ارسال خواهد کرد.

تابع chk_blockVoiceActionPerformed:

این تابع پیامی را به سمت سرور ارسال می کند که باعث می شود که کاربر انتخابی بلاک شود و پیام های صوتی آن از آن پس برای این مشتری پخش نشوند.

تابع btn_voiceActionPerformed:

این تابع یک نخ برای ارسال مداوم صوت ورودی به سمت سرور ایجاد می کند که سرور نیز آن را برای مشتری مورد نظر ارسال می کند. همچنین این نخ خروجی سرور را نیز که از طرف مشتری دیگر ارسال شده است پخش می کند. این نخ صوت را به صورت جویباری به سمت سرور ارسال می کند.

تابع Message:

این تابع پیام دریافت شده از طرف سرور را در log مشتری نمایش می دهد.

تابع Voice:

این تابع جویبار صوت دریافت شده از طرف سرور را دریافت می کند و آن را پخش می کند.

تابع Users:

این تابع اطلاعات کاربران آنلاین را از طرف سرور دریافت می کند و لیست کاربران آنلاین را نمایش می دهد.

تابع radioButtonClick:

این تابع با انتخاب هر کاربر از لیست اطلاعات آن را استخراج کرده و وضعیت بلاک شدن و ... را مشخص می کند.

تابع InsertString:

این تابع پیام ارسالی را در قسمت log چاپ می کند. وظیفه این تابع تبدیل کدهای شکلک ها به عکس شکلک ها و همچنین نوشتن برخی بخش های پیام ها به صورت رنگی می باشد.

تابع IndexOfSmile:

این تابع اندیس معادل با کد یک شکلک را برمی گرداند.

تابع getAudioFormat:

این تابع تنظیمات مرتبط با پیام صوتی را برمی گرداند. این تنظیمات مربوط به فرکانس ضبط صدا و همچنین فرکانس برای پخش صدا می باشد.

کلاس SmileActionListener:

این کلاس مشخص می کند که در هنگام انتخاب هر یک از شکلک ها چه عکس العملی نشان داده شود. وظیفه تابع اصلی این کلاس اضافه کردن کد مرتبط با شکلک به متن می باشد.

کلاس VoiceThread:

این کلاس یک نخ می باشد. در صورت اجرا بودن این نخ امواج صوتی با تنظیماتی مشخص از ورودی دریافت می شود و آن را به پیامی با عنوان مشخص تبدیل می کند. سپس پیام را به سرور ارسال کرده و سرور آن را به مشتری مورد نظر تحویل می دهد.

کلاس ClientThread:

این کلاس پیام های ارسالی از طرف سرور را دریافت می کند و با توجه به عنوان آن نسبت به پیام عکس العمل نشان می دهد. پیام ها عناوین مختلفی دارند که در زیر به آنها اشاره می کنیم:

۱. Message: برای پیام های متنی به کار برده می شود.
۲. Voice: برای پیام های صوتی به کار برده می شود.
۳. Users: برای پیام هایی که از طرف سرور برای ارسال اطلاعات کاربران آنلاین استفاده می شود.