



دانشگاه صنعتی اصفهان

پروژه درس برنامه‌نویسی پیشرفته

در این پروژه قصد داریم با استفاده از مفاهیم آموخته‌شده در درس برنامه‌نویسی پیشرفته و با استفاده فریمورک Qt یک پیام‌رسان گرافیکی طراحی نماییم. این پروژه حاوی دو قسمت مهم است: ۱- کد سرور ۲- کد کلاینت

کد سرور پیام‌رسان شما پیش‌تر توسط دستیاران آموزشی درس نوشته شده است و وظیفه شما در این پروژه طراحی و نوشتن کد کلاینت با به کارگیری مفاهیم شیء‌گرایی می‌باشد. برای برقراری ارتباط میان کلاینت و سرور از مجموعه API های آماده که در اختیار شما گذاشته می‌شود استفاده خواهید کرد. به کمک API های آماده، شما درخواست‌های لازم را به سمت سرور ارسال کرده و سپس با توجه به نوع درخواست شما، سرور پاسخ مناسب را به سمت کلاینت بازمی‌گرداند. در نهایت شما می‌توانید از اطلاعات بازگردانده‌شده در پیام‌رسان خود استفاده نمایید.

هر API شامل بخش‌های زیر می‌باشد:

<http://hostname:port/command?parameters>

hostname	در اینجا منظور آدرس سرور مورد نظر است که کد سرور بر روی آن اجرا می‌گردد.
port	آدرس پورتی است که شما از طریق آن با سرور ارتباط برقرار کرده و درخواست‌های خود را ارسال و دریافت می‌کنید.
command	مجموعه دستوراتی است که شما از سرور درخواست می‌کنید.
parameters	گاهی نیاز است برای ارسال یک درخواست اطلاعاتی را نیز به همراه درخواست خود ارسال نمایید. در این صورت اطلاعات لازمه را به صورت جفت‌های (کلید، مقدار) به سرور ارسال می‌نمایید.

به عنوان مثال برای ارسال درخواست ثبت نام یک کاربر با نام کاربری sara و پسورد 1234 می‌بایست درخواست خود را به صورت زیر برای سرور ارسال نمایید (در این درخواست username و password پارامترهای ارسالی به سمت سرور می‌باشند):

<http://api.barafardayebehtar.ml:8080/signup?username=sara&password=1234>

خروجی تمام API ها به فرمت json می‌باشد. برای آشنایی بیشتر با این فرمت [لینک](#) را مطالعه نمایید. پاسخ سرور در قالب json دارای دو بخش code و message خواهد بود که به ترتیب به معنی کد پاسخ و متن پاسخ ناشی از درخواست از جانب سرور می‌باشد. کد پاسخ‌های سرور در این پروژه به شرح زیر است:

200	درخواست شما با موفقیت انجام شده‌است.
404	درخواست شما موفقیت‌آمیز نبوده و اطلاعات درخواستی یافت نشده است. به طور مثال اگر درخواست کنید لیست پیام‌های یک گروه با نام GP1 را از سرور دریافت کنید حال آنکه چنین گروهی اصلاً وجود نداشته باشد، در این صورت درخواست شما یافت نشده و کد ۴۰۴ از جانب سرور بازگردانده می‌شود.
204	درخواست شما موفقیت‌آمیز نبوده و به دلیل تکراری بودن اطلاعات امکان انجام درخواست برای سرور وجود ندارد. به طور مثال اگر بخواهید کاربری با نام کاربری sara ایجاد کنید در حالی که پیش‌تر یک کاربر با این نام ایجاد شده و اطلاعات آن در سرور موجود می‌باشد، در این صورت درخواست ایجاد کاربر جدید موفقیت‌آمیز نبوده و سرور کد ۲۰۴ را به شما بازمی‌گرداند.
401	درخواست شما موفقیت‌آمیز نبوده و اطلاعاتی که در اختیار سرور قرار داده‌اید نادرست است. برای مثال شما کاربر sara را با پسورد ۱۲۳۴ ایجاد کرده‌اید و حال می‌خواهید وارد حساب کاربری خود شوید. چنانچه پسورد نادرستی را برای سرور ارسال کنید، کد ۴۰۱ از جانب سرور به شما بازگردانده می‌شود و ورود به حساب کاربری موفقیت‌آمیز نخواهد بود.

در این پروژه آدرس سروری که کد سرور مورد نظر را اجرا می کند (hostname) و نیز پورت ارتباطی شما با سرور به صورت زیر می باشد:

hostname	api.barafardayebehtar.ml
port	8080

پس تمامی درخواست های شما که به سمت سرور ارسال می گردد باید به فرمت زیر باشد:

<http://api.barafardayebehtar.ml:8080/command?parameters>

سناریو اصلی پیام رسان

در این پروژه شما نخست می بایست به عنوان یک کاربر در برنامه ثبت نام کنید. سپس یک رشته حاوی ۱۶ کاراکتر به نام توکن (token) توسط سرور به شما داده می شود که نمایانگر هویت کاربری است که وارد برنامه شده و login کرده است. در ادامه می بایست تمامی درخواست های هر کاربر همراه با توکن یکتای او به سمت سرور ارسال شود تا سرور کاربر مورد نظر را شناخته و پاسخ مناسب به درخواست او ارسال نماید. لازم به ذکر است زمانی که کاربر از پیام رسان خارج می شود و logout صورت می پذیرد، توکن مورد نظر null شده و پس از login مجدداً توکن یکتای جدید به کاربر تخصیص می یابد.

در جدول زیر لیستی از تمامی API های قابل استفاده به همراه نحوه استفاده از آنها و پارامترهای لازم هر یک آورده شده است (پارامترهایی که با رنگ قرمز مشخص شده اند اجباری و سایر پارامترها اختیاری می باشند):

signup	username password firstname lastname	این دستور برای ارسال درخواست ثبت نام یک کاربر می باشد. در صورتی که ثبت نام با موفقیت صورت بپذیرد، سرور کد ۲۰۰ را به همراه پیغام signed up successfully برای شما ارسال می کند.
login	username password	این دستور برای ارسال درخواست ورود یک کاربر می باشد. در صورتی که ورود با موفقیت صورت بپذیرد، سرور کد ۲۰۰ را به همراه پیغام logged in successfully به علاوه یک توکن یکتا به منظور احراز هویت کاربر برای شما ارسال می کند.
logout	username password	این دستور برای ارسال درخواست خروج یک کاربر می باشد. در صورتی که خروج با موفقیت صورت بپذیرد، سرور کد ۲۰۰ را به همراه پیغام logged out successfully برای شما ارسال کرده و توکن کاربر مورد نظر را null می کند.
creategroup	token group_name group_title	این دستور برای ارسال درخواست ساخت گروه می باشد. در صورتی که ساخت گروه با موفقیت صورت بپذیرد، سرور کد ۲۰۰ را به همراه پیغام group created successfully برای شما ارسال می کند. به صورت خودکار کاربر جاری پس از ساخت یک گروه عضو آن گروه نیز می شود.
createchannel	token channel_name channel_title	این دستور برای ارسال درخواست ساخت کانال می باشد. در صورتی که ساخت کانال با موفقیت صورت بپذیرد، سرور کد ۲۰۰ را به همراه پیغام channel created successfully برای شما ارسال می کند. به صورت خودکار کاربر جاری پس از ساخت یک کانال عضو آن کانال نیز شده و مدیر آن می باشد.
getuserlist	token	این دستور برای ارسال درخواست لیست کاربرانی است که کاربر جاری با آنها چت داشته است. در خروجی لیست username کاربرانی که کاربر جاری با آنها چت داشته است به ترتیب زمانی برگردانده می شود.

getgrouplist	token	این دستور برای ارسال درخواست لیست گروه‌هایی است که کاربر جاری در آن‌ها عضویت داشته است. در خروجی لیست group_name گروه‌هایی که کاربر جاری در آن‌ها عضویت داشته است برگردانده می‌شود.
getchannellist	token	این دستور برای ارسال درخواست لیست کانال‌هایی است که کاربر جاری در آن‌ها عضویت داشته است. در خروجی لیست channel_name کانال‌هایی که کاربر جاری در آن‌ها عضویت داشته است برگردانده می‌شود.
getuserchats	token dst date	این دستور برای ارسال درخواست لیست پیام‌هایی است که کاربر جاری با کاربر dst داشته است. در خروجی لیست پیام‌های میان کاربر جاری و dst به ترتیب زمانی برگردانده می‌شود. چنانچه پارامتر date را به صورت "YYYYMMDDHHmmss" مقداردهی کرده و همراه با درخواست ارسال نمایید، تنها لیست پیام‌ها پس از این تاریخ توسط سرور بازگردانده خواهد شد.
getgroupchats	token dst date	این دستور برای ارسال درخواست لیست پیام‌هایی است که در گروه dst میان کاربران ردوبدل شده است. در خروجی لیست پیام‌های گروه dst به ترتیب زمانی برگردانده می‌شود. چنانچه پارامتر date را به صورت "YYYYMMDDHHmmss" مقداردهی کرده و همراه با درخواست ارسال نمایید، تنها لیست پیام‌ها پس از این تاریخ توسط سرور بازگردانده خواهد شد.
getchannelchats	token dst date	این دستور برای ارسال درخواست لیست پیام‌هایی است که در کانال dst ارسال شده است. در خروجی لیست پیام‌های کانال dst به ترتیب زمانی برگردانده می‌شود. چنانچه پارامتر date را به صورت "YYYYMMDDHHmmss" مقداردهی کرده و همراه با درخواست ارسال نمایید، تنها لیست پیام‌ها پس از این تاریخ توسط سرور بازگردانده خواهد شد.
joingroup	token group_name	این دستور برای ارسال درخواست عضو شدن در گروه group_name است. در صورتی که عضو گروه شدن با موفقیت صورت بپذیرد، سرور کد ۲۰۰ را به همراه پیام successfully joined برای شما ارسال می‌کند.
joinchannel	token channel_name	این دستور برای ارسال درخواست عضو شدن در کانال channel_name است. در صورتی که عضو کانال شدن با موفقیت صورت بپذیرد، سرور کد ۲۰۰ را به همراه پیام successfully joined برای شما ارسال می‌کند.
sendmessageuser	token dst body	این دستور برای ارسال پیام به کاربر dst است. در صورتی که پیام با موفقیت ارسال شود، سرور کد ۲۰۰ را به همراه پیام message sent successfully برای شما ارسال می‌کند.
sendmessagegroup	token dst body	این دستور برای ارسال پیام در گروه dst است. در صورتی که کاربر عضو گروه باشد و پیام با موفقیت ارسال شود، سرور کد ۲۰۰ را به همراه پیام message sent successfully برای شما ارسال می‌کند.
sendmessagechannel	token dst body	این دستور برای ارسال پیام در کانال dst است. در صورتی که کاربر مدیر کانال باشد و پیام با موفقیت ارسال شود، سرور کد ۲۰۰ را به همراه پیام message sent successfully برای شما ارسال می‌کند.

در این پروژه قصد داریم اطلاعات پیام‌رسان در حالت آفلاین نیز در دسترس باشد لذا می‌بایست تمامی پیام‌هایی که از سرور درخواست می‌شود را در قالب فایل‌های دسته‌بندی شده ذخیره نمایید. برای این کار نخست سه فولدر با نام‌های Users و Groups و Channels ایجاد نمایید. سپس به ازای هر

کاربری که با وی پیام رد و بدل کرده‌اید یک فایل با نام username کاربرد مورد نظر در داخل فولدر Users ایجاد کرده و پیام‌های کاربر جاری با وی را داخل این فایل ذخیره نمایید. به طور مشابه پیام‌های هر گروه و یا کانالی را که کاربر در آن‌ها عضویت دارد در فایل‌های جداگانه ذخیره نمایید تا در صورت عدم اتصال به اینترنت اطلاعاتی که قبلاً از سرور گرفته شده است را نمایش دهید و هنگامی که پیام‌رسان شما به اینترنت متصل شد تنها پیام‌های ردوبدل شده بعد از تاریخ آخرین پیام ذخیره شده را از سرور درخواست نمایید. لازم به ذکر است که در صورت خروج یک کاربر تمامی فایل‌های مربوطه می‌بایست حذف شود و سپس ضمن لاگین، مجدداً برای کاربر جدید ایجاد گردند.

نکات مهمی که می‌بایست حین پیاده‌سازی پروژه در نظر داشته باشید:

- شما می‌بایست از مفاهیم File و Multithreading به درستی استفاده کنید. پیام‌رسان شما می‌بایست در حالت اتصال به اینترنت، بدون نیاز به عمل خاصی از جانب کاربر، پیام‌های جدید را از سرور دریافت نماید.
- استفاده صحیح از مفاهیم شیء‌گرایی در حین پیاده‌سازی پروژه الزامی می‌باشد و شما می‌بایست هدر فایل‌ها و سورس فایل‌های مناسب را برای پروژه خود ایجاد نمایید.
- پروژه را می‌توانید در قالب گروه‌های ۲ یا ۳ نفره انجام دهید (دقت نمایید که الزاماً می‌بایست با دانشجویان هم‌کلاس خودتان گروه تشکیل دهید).
- برنامه شما باید به میزان منطقی از CPU و RAM استفاده کند.
- در انجام این پروژه ملزم به استفاده از ابزار Git هستید و می‌بایست موارد زیر را رعایت نمایید:
 - در حین انجام پروژه از GitHub استفاده نمایید و یک readme مناسب برای پروژه خود بنویسید.
 - الزاماً commit های شما می‌بایست بامعنی باشند و روند اجرا و تکمیل پیوسته پروژه را نشان دهند.
 - تمامی افراد گروه باید از یک repository استفاده نمایند، لذا مشارکت همه اعضای گروه در پتل پروژه شما الزامی است.
- برای سادگی کار می‌توانید پروژه خود را در سه فاز انجام دهید:
 - طراحی کلاس‌ها، attribute ها و method های مورد نیاز
 - راه‌اندازی گیت و پیاده‌سازی توابع و کلاس‌های طراحی شده
 - پیاده‌سازی رابط گرافیکی

نمره اضافه (در صورتی برای انجام موارد زیر در پروژه نمره اضافه دریافت خواهید کرد که حداقل ۹۰ درصد از اصل پروژه را انجام داده باشید):

- اگر پروژه شما دارای installer باشد نمره اضافه دریافت خواهید کرد.
- تمیزی و reusable بودن کد شما از نمره اضافی ویژه‌ای برخوردار است، بدین منظور می‌توانید به این [لینک](#) مراجعه نمایید.
- اگر پروژه خود را داکرایز کرده و بتوانید بر روی هر سیستمی که داکر دارد اجرا نمایید نمره اضافه دریافت خواهید کرد.

موفق باشید