

خلاصه ی پروژه ی برنامه نویسی فاز ۱

برنامه نویس: نیما کلیدری

عنوان: نویتر



دانشکده ی علوم ریاضی

مدرس: استاد مجتبی استواری

قسمت اول: منابع مورد استفاده

در این برنامه از کتابخانه ی Gson برای ارتباط با فایل های json و کتابخانه ی Log4j برای ایجاد لاگ های برنامه استفاده شده است. در این پروژه تنها از ۱ نفر در اخطار ها و باگ های کلی برنامه کمک گرفته شده است. اما از ۳ سایت و منبع برای یادگیری و پیاده سازی استفاده شده است.

۱- محمد جعفری دانشجوی مهندسی کامپیوتر

۱- <http://tutorials.jenkov.com/java-json/gson.html>

۲- <https://blog.faradars.org/arraylist-%D8%AF%D8%B1-%D8%AC%D8%A7%D9%88%D8%A7-java>

۳- <https://www.scalyr.com/blog/get-started-quickly-java-logging>

قسمت دوم: توضیح ساختار کد و بررسی نقاط قوت و ضعف

در طراحی این برنامه سعی شده از ساختار شی گرا استفاده شود. بنابر این در ۵۰ کلاس و ۵۳۴۰ خط (۴۷۶۹ خط موثر) نوشته شده است. در ورود در صورت ورود حرف R به صفحه ی ثبت نام وارد میشوید که فیلتر های لازم و چک کردن یکسان نبودن فیلد های لازم اعمال شده است. کاربر در ورود وارد فیلد Feed میشود که به منزله ی صفحه ی اصلی برنامه است.

در این صفحه تعدادی توییت با شماره هایی به ترتیب و مشخص نوشته آمده است. با استفاده از دستور العمل توییت ها ی نوشته شده در صفحه ی شروع برنامه میتوان اعمال مختلف را انجام داد که در برنامه نوشته شده است. تمامی قوانین بلاک هر طرف و میوت های کاربر رعایت شده است و چنانچه توییت نوشته شده ریتوییت یا کامنت یک توییت دیگر باشد به کاربر اطلاع داده میشود.

چون تمام بخش های برنامه برای کاربر مشخص است، بجای توضیح اضافی برای کار با برنامه به توضیح ساختار کلاس ها میپردازیم.

- Category : به مدیریت و اضافه و حذف کردن دسته بندی ها و افراد در دسته بنده ها میپردازد.
- chatPage: این صفحه با گرفتن یوزر نیم فرد اول و دوم کاربر اول را وارد صفحه میکند.
- commentsPage: با دریافت یوزرنیم فرد و آی دی یک توییت، کامنت های آن توییت را نشان میدهد.

- ConsoleColors: برای رنگ بندی صفحه ی کنسول استفاده میشود.
- Editprofile: در این کلاس سنگین به تغییر مشخصات کاربر میپردازیم.
- Expelorer: مانند Feed میماند اما توییت افرادی که فالوشان نکرده اید هم نمایش میدهد.
- getLoginInfo/getReginfo : بعد از ورود به صفحه ی ورود یا ثبت نام برای دریافت سایر اطلاعات از کاربر وارد این صفحات میشود.
- Home: برای اعمال شخصی تر کاربر استفاده میشود. که در داک پروژه نیز گفته شده و مانند آن عمل شده است.
- Info: برای نمایش مشخصات کاربر
- JsonMessage/Notifs/Report/Users/Twittes برای دریافت و ثبت لیست مربوطه در فایل Json آنها استفاده میشود.
- Lists برای نمایش لیست های یک کاربر (۲ پارامتر کاربر بیننده و کاربر صاحب لیست دریافت میکند)
- Login هنگام ورود به برنامه وارد این صفحه میشویم.
- Main برای خوشامد گویی و توضیح کوتاه بکار میرود.
- massMessenger برای ارسال پیام جمعی مطابق داک به کار میرود.
- Messenger برای ارسال پیام های شخصی استفاده میشود.
- newChat برای ساخت صفحه ی چت و شروع به چت
- newTwitte با این کلاس میتوان توییت جدید ایجاد کرد.
- Next Page : برای پایین آمدن خط به تعداد آرگومان ورودی استفاده میشود.
- Notification: برای اعلام اعلان ها و درخواست ها استفاده میشود.
- objMessage/Notifs/Report/Users/Twittes یک شی برای ساخت و دریافت و ساماندهی لیست ها ی مربوطه استفاده میشود.
- Regsiter: برای ثبت نام استفاده میشود.
- Reports: برای ثبت هرزنامه ها در فایل Json استفاده میشود.
- SavedMessages: از بخش پیام ها وارد شده و میتوان توییت ها و مسیج های ذخیره شده را مشاهده کرد. همچنین میتوان پیام هایی نوشت که در همان صفحه ذخیره شود.
- Setting : صفحه ی تنظیمات را نشان میدهد و اعمال گفته شده در داک را به اضافه ی درست کردن دسته بندی ها انجام میدهد.
- sleepTheard متوقف کردن روند اجرای برنامه به اندازه ی عدد داده شده در آرگومان
- Submit Action برای ثبت اعمال یک کاربر بر روی یک کاربر دیگر (فالو و بلاک و میوت و دیلیت و ...)

- submitComment برای دریافت و ثبت یک کامنت استفاده میشود.
- submitDelete انجام حذف حساب کاربری و جزئیات آن
- submitEdit انجام ویرایش حساب کاربری و جزئیات آن
- submitLike لایک کردن یک توییت را ثبت میکند
- submitMessage با گرفتن متن پیام و فرستنده و گیرنده آن پیام را ثبت و در صفحه ی مربوطه مشاهده میکند.
- submitPrivacy برای ثبت تنظیمات قابل دیدن بودن ایمیل و شماره تلفن و آخرین بازدید و تاریخ تولد برای افراد مختلف
- submitRegistry ثبت مشخصات کاربر هنگام ثبت نام او
- submitRequest ثبت درخواست فالو کردن یک کاربر در فایل جیسون مربوطه
- submitRetwitte ثبت ریتوییت برای یک پست
- submitTwitte ثبت یک توییت در فایل جیسون آن
- Text برای نمایش و دریافت متن از کاربر
- TwittePrinter با گرفتن شاخص یک توییت بطور سازماندهی شده آنرا نمایش میدهد.
- userProfile صفحه ی شخصی هر کاربر که شامل دو یوزر نیم صاحب صفحه و فرد اول است.
- userTwittes توییت های انجام شده توسط هر کاربر را نشان میدهد.

نقاط قوت:

- سازماندهی کامل جزئیات و توجه به لیست فالوینگ و بلاک و ... برای انجام هر عملی.
- دقت و امکانات کاملتری از آنچه در داک عنوان شده بود.
- شی گرای و مستقل بودن هر بخش که باعث کار راحت تر با کد برای افراد دیگر میشود.
- باگ بسیار کم با توجه به حجم پروژه ی انجام شده.
- توجه کامل به چرخه ها و ارث بری توییت های کامنت و ریتوییت های کامنت از هم
- لاگ ساختارمند
- رنگ بندی ها و فاصله گذاری های مناسب برای کنسول

نقاط ضعف:

- باگ های احتمالی جزئی در هنگام اجرا
- تمام تلاش شده است در صورت عدم وجود فایل json ساخته شود و مشکلی بوجود نیاید. اما در بعضی از نقاط برنامه احتمال ارور هایی بر این مبنی در از ذهن نیست.
- تنها در صورتی که کاربر یا از حساب یا برنامه خارج شود آخرین زمان آنلاین بودن او ثبت میشود و از هر راه دیگر خارج شود Online باقی میماند. که البته راه حلی با توجه به جستجو های انجام شده وجود نداشت.

قسمت سوم: ارائه ی دلیل برای ساختار کد

بنده با توجه به اینکه ابتدا با کتابخانه های دیگر جیسون کار میکردم با نصب جیسون تصمیم گرفتم برای هر عملی با تابعی یک لیست از جیسون لود کنم و پس از اعمال تغییراتی آن را با تابعی دیگر در فایل ذخیره کنم. در این روش سر در گمی کمتر شده و حس راحتی بیشتری برای برنامه نویسی ایجاد میشود.

برای پیمایش کاربر تنها با وارد کردن یک حرف در منو ها جابجا میشود و در هیچ جای برنامه بجز وارد کردن متن نیاز به نوشتن کلمه نیست. این به دلیل راحتی کار با برنامه و تبدیل راحت تر آن به حالت گرافیکی انجام شده است.

برای هر قسمت از برنامه تصمیم گرفته شده است با استفاده از متد (تابع+return) کاربر وارد صفحه جدید شود و پس از اتمام کار یک به یک خارج شود. این کار از لحاظ حافظه ی موقت بار زیادی روی رم قرار میدهد اما جای مانور زیادی فعلا وجود نداشت، در فاز بعدی انشالله راه حل کاملی پیدا میگردد.

به این دلیل که شاید مشکلاتی در meven و gradle بوجود بیاید، (برای من بار ها پیش آمده است) فایل jar کتاب خانه های Gson و Log4j در پروژه قرار داده شده است.

قسمت سوم: کتابخانه ها و ورژن ها

جاوا ورژن ۱۵

محیط برنامه نویسی IntelliJ IDEA version 2021.3.2

کتابخانه GSON ورژن ۲.۸.۶

کتابخانه Log4j ورژن ۲.۱۴.۱