



پروژه چهارم برنامه سازی پیشرفته

دکتر فرید فیضی

ترم 14012

سامانه آموزشیار



دانشگاه گیلان

آموزشیار



سامانه دانشجویی

(فقط امور انتخاب واحد و حذف و اضافه بر اساس جدول زمانبندی اعلام شده)

نام کاربری :

کلمه عبور :

[فراموشی / باز شدن قفل رمز عبور](#)

مقدمه

پس از پیاده‌سازی برنامه مدیریت آموزش در پروژه قبل، دانشجویان و اساتید بسیار از امکانات برنامه شما خشنود شدند و طبق گفته آنها، روند امور آموزشی سریعتر شد؛ در این میان انتقاداتی در مورد رابط کاربری (UI) و تک‌کاربره بودن آن مطرح شد؛ از این رو مدیر دانشگاه، ساخت سامانه‌ای برای امور آموزشی دانشجویان به شما محول کرده.

در این پروژه قصد داریم، با استفاده از برنامه نویسی شبکه (Socket-Programming) و برنامه نویسی چند نخ (Multi-Threading)، برنامه‌ای در محیط گرافیکی (Java-Swing) برای سامانه دانشگاه بنویسید.

شروع برنامه

در پنل ابتدایی انتظار می‌رود یوزرها با وارد کردن Username و Password وارد پنل مربوط به خود شوند.

انواع یوزرها:



- مدیر دانشگاه
- مدیر گروه
- استاد
- دانشجو

قابلیت هر یوزر

مدیر دانشگاه:



- افزودن رشته
- دیدن لیست اساتید
- استخدام استاد
- تعیین مدیر گروه
- ثبت نام دانشجویان

مدیر گروه:



- مدیریت لیست واحدهای ارائه شده مربوط به رشته خود
- باز و بسته کردن انتخاب واحد
- تغییر ظرفیت واحد
- نهایی کردن نمرات

نکته: مدیر گروه خود یکی از اساتید است که مدیریت گروه رشته خود را بر عهده دارد (تمامی قابلیت‌های استاد را هم دارد).

استاد:



- ثبت نمره و در صورت نیاز، تغییر نمره دانشجویان
- مکاتبه با دانشجو و پاسخ به اعتراضات

دانشجویان:



- انتخاب واحد
- دیدن کارنامه و اعتراض به نمره
- مکاتبه با اساتید

نکته: در لیست انتخاب واحد، فقط دروس مربوط به رشته دانشجو موجود است و دانشجو نمیتواند واحدی را بردارد که با واحدی که از قبل برداشته تداخل داشته باشد.

توضیحات مربوط به سرور

سرور وظیفه ارتباط بین کلاینت ها و ارسال دیتا بین آن ها را بر عهده دارد. پس از اجرا شدن، سرور می بایست تمام داده های کاربران از جمله اطلاعات کاربری شان را از فایل بخواند و در خود ذخیره کند. سرور نباید تا زمانی که درخواستی از سوی کلاینت برای آن ارسال نشده است، اطلاعاتی برای هیچ کلاینتی بفرستد.

هر کلاینت پس از اتصال به سرور، بر اساس اطلاعات ورودی خود شامل یوزرنیم و پسورد، داده های مربوط به کاربر مذکور را دریافت میکند و از این طریق برنامه اصلی کار میکند. هرگونه تغییری در کاربر به سرور ارسال شود، در شئی مربوط به آن اعمال می شود. به عنوان مثال اگر استاد A نمرات دانشجوی B را وارد کند، نمرات به سرور ارسال می شود، و در کارنامه دانشجوی B، و همچنین در لیست نمرات وارد شده استاد A ذخیره می شود.

انتظار می رود که با توجه به ساختار کلاینت-سرور، تمامی اطلاعات فقط از طریق سرور قابل دسترسی باشند و کلاینت به هیچ اطلاعاتی از دیگران دسترسی نداشته باشد. توجه کنید که تنها باید اطلاعات لازم به کلاینت فرستاده شود. به عنوان مثال اگر می خواهید اطلاعات پروفایل یک کاربر دیگر را به کلاینت بفرستید، فرستادن اطلاعات محرمانه او، مانند رمز عبور، مجاز نمی باشد! پس دقت کنید که تنها اطلاعات مورد نیاز را به کلاینت ارسال کنید.

نکته: تمامی اطلاعات کاربران باید توسط سرور ذخیره شوند. برای ذخیره سازی این اطلاعات در فایل از دو روش می توانید استفاده کنید. یا اینکه خود شئی کاربران را روی فایل ذخیره کنید یا اینکه مستقیم اطلاعاتشان را در فایل بریزید. (راه های دیگری وجود دارد که در قسمت **نمره مثبت** ذکر شده).

توجه: انجام این پروژه به صورت گرافیکی با استفاده از Swing Java اجباری می باشد.

نمره مثبت +

- ذخیره اطلاعات به کمک دیتابیس (مثل SQL)
- انتقال داده‌های بین سرور و کلاینت با استفاده از JSON
- استفاده از git و آشنایی مختصر با قابلیت‌های آن
- اطلاع‌رسانی عمومی
- ایده خلاقانه با هماهنگی TAها

نکات :

انتخاب ساختار قسمت هایی که درباره آنها داخل مستندات پروژه توضیحی داده نشده، برعهده‌ی دانشجو است.

هیچ توجیهی در استفاده از زبان های برنامه نویسی دیگر وجود نداشته و برابر با تحویل ندادن پروژه ، نمره **صفر** به دانشجو تعلق میگیرد.

پروژه به صورت انفرادی یا گروه‌های دونفره قابل انجام بوده و در صورت وجود هرگونه تشابه بین دو کد یا عدم تسلط به روند، **نمره منفی** به دانشجویان داده می شود (اعضای تیم باید تسلط کامل بر کل پروژه داشته باشند).

برای تسلط بهتر و کسب نمره خوب در این درس، به انجام پروژه های اضافه بر پروژه های ترم اقدام کنید.

کامنت نویسی مناسب و رعایت دندانه گذاری الزامی است.