

"گزارش تمرین سری هشتم"

شبیه سازی Steam

ساختار کلی:

ابتدا کاربر با ساخت اکانت وارد برنامه می شود یا اگر از قبل دارای اکانت است پس از وارد کردن پسورد و username (در صورتی که درست وارد کند) لاگین کرده و وارد برنامه میشود. در ابتدا نیاز است تا لیست تمامی بازی های موجود را ببیند و عدد متناظر بازی ای که مدنظر گرفته را به خاطر بسپارد، (البته می تواند به ادامه فعالیت خود ادامه ندهد و دوباره به منوی اولیه برگردد) حال پیش روی خود دو انتخاب دارد: ۱) عدد بازی دلخواه خود را جهت مشاهده تمام جزئیات بازی مانند ژانر بازی، نام سازنده، قیمت بازی، سال تولید، نام سازنده و ... را وارد کند. ۲) عدد بازی دلخواه خود را جهت دانلود فایل آن وارد کند.

توجه: میشد بازی به گونه ای طراحی شود که کاربر برای هر فعالیتی نیاز به حساب کاربری نداشته باشد اما طراحی من برای استفاده از هرگونه قابلیت نیاز به حساب کاربری دارد. همچنین میشد بجای عدد بازی نام بازی از کاربر گرفته شود اما چون امکان اشتباه تایپی و یا املایی در نام بازی خیلی بیشتر از احتمال وارد کردن یک عدد بود ترجیح دادم تنها یک عدد از کاربر گرفته شود تا بجای دریافت خطا راحت تر به مطلوب خود دست پیدا کند.

برای تقسیم بندی درخواست های کاربر، تمام درخواست های ممکن مانند login, download, exit, find(view one game),... در قسمت Title

قرار داده شده است تا هر جا که درخواستی فرستاده میشد با استفاده از setTitle نوع درخواست مشخص و تابع مربوط به آن فرخوانده شود. در کل کلاس ها و package های اضافه شده برای خرید کردن پروژه اصلی به بخش های کوچک تر برای تمیزتر شدن کد و به سرانجام رساندن پروژه اصلی است.

توجه: پس از اجرا گرفتن برنامه در نهایت قبل از push , commit کردن پسورد مربوط به postgresSQL خود را برای حفظ امنیت پاک کردم و آن را یک

رشته خالی قرار دادم. پس لازم است برای اجرا و بررسی کد پسورد مربوط به SQL در string pass قرار داده شود تا اتصال دیتابیس برقرار گردد.

Backup: با استفاده از همان روش گفته شده در pgAdmin از دیتابیس backup گرفته شد و در همان دایرکتوری Database Backup در پروژه قرار گرفته است.

"هم چنین برنامه مولتی ترد نوشته شده است."