

پروژه برنامه نویسی پیشرفته فاز سوم

ابتدا نقاط ضعف و قوت پروژه را ذکر میکنیم. سپس منابعی که از آن استفاده شده را ذکر میکنیم.

نقاط قوت :

- ۱- اولین نکته مهم این است که برنامه کاملاً شی گرا است و سعی شده پکیج بندی ها و کلاس بندی ها با نهایت دقت انجام شود و سطح دسترسی تابع ها و متد ها حداقل قرار داده شده. پکیج اصلی بازی hearthstone است که خود دارای چند پکیج است.
- ۲- دومین مورد گرافیک برنامه است که سعی شده زیبا طراحی شده و همه دکمه ها و ایکون ها ساخته شده باشند و مستقیم از خود کتابخانه سوییچ گرفته نشوند. (که البته در این بخش از یکی از دوستان کمک گرفتیم که در بخش منابع ذکر خواهیم کرد).
- ۳- سومین مورد استفاده از جلوه های صوتی زیبا برای هم صفحه اصلی و هم صفحه بازی (به صورت جداگانه) است. البته برای کارت ها هنگام بازی شدن روی زمین هم صداهایی قرار داده شده. همچنین برای دکمه end turn و صحنه آخر بازی که برد و باخت را مشخص می کند هم صداهای مناسب قرار داده شده.
- ۴- همچنین تمام اجزای بازی اعم از هیرو ها و کارت ها قابل تغییر به آسانی هستند. یعنی ما قالب کارت یا هیرو یا حتی یک دکمه را میگیریم، سپس متن یا تعداد مانا یا خون یا ... را در آن مینویسیم. البته این کار در فاز قبل انجام شد و در این فاز توسعه یافت. برای مثال برای هیرو پاور ها یا اسلحه ها!
- ۵- در هر مرحله ارور های درستی به کاربر نشان داده شده است. در حین بازی هم تمامی ارور ها به کاربر نشان داده می شوند. برای مثال اگر با یک کارت rush قصد حمله به هیرو را داشت، هشدار لازم داده می شود.
- ۶- اطلاعات کانفیگ برای گرافیک و اطلاعات بازی در دو فایل جدا و در بیرون از بازی ذخیره شده اند.
- ۷- لاجیک بازی با استفاده از اینترفیس های گوناگون زده شده و شبیه پترن observer می باشد، اما اندکی تفاوت وجود دارد! یعنی ما صرفاً برای موجودات اینترفیس ها یا رفتار آن ها را تعریف میکنیم، و در صفحه بازی موقع لزوم این رفتار ها بروز پیدا می کنند!
- ۸- گرافیک و لاجیک بازی از هم کاملاً جدا هستند و تنها راه ارتباطی آن ها Mapper می باشد. همچنین یک کلاس DataTransform هم ساخته شده که کارش فقط دادن اطلاعات است و کاری انجام نمی دهد. با اینکار در فاز ۴ به هنگام شبکه، کار ما بای Mapper و DataTransform بسیار راحت می شود.

۹- انواع مختلفی برای موس در این فاز طراحی شده. برای مثال حالت موس برای هیل، حمله، فریز کردن و دنبال چیزی گشتن فرق می‌کند. و همه موارد ذکر شده با حالت معمولی موس متفاوت هستند. این کار به کاربر کمک می‌کند متوجه شود اکنون منتظر چه کاری هست. همچنین با راست کلیک می‌توان عملیات فعلی را کنسل کرد. یعنی اگر در حال حمله هستیم، با زدن راست کلیک به حالت عادی بر میگردیم.

۱۰- همچنین کلاس‌هایی برای ارتباط با بیرون از برنامه درست شده اند. کار این کلاس این است که اطلاعات را از resource برنامه بخوانند. البته اگر قبلاً این اطلاعات خوانده شده بود، دوباره نمیخوانند و همان اطلاعات قبلی را بر می‌گردانند. این کار باعث شد مشکل سرعتی که در فاز قبل داشتیم حل شود و سرعت برنامه چندین برابر شد.

۱۱- در این فاز از انیمیشن‌های زیبا استفاده شده که این کلاس‌ها داینامیک بودند و قابلیت استفاده مجدد را دارند. البته انیمیشن‌ها یک مرحله‌ای هستند. ولی به گونه‌ای طراحی شدند که می‌توان مسیر یک انیمیشن را در حین حرکتش تغییر داد!

۱۲- همچنین بعضی قسمت‌هایی که از فاز قبل تکمیل نشده بودند یا به علت فراموشی اضافه نشده بودند، در این فاز اضافه شدند. برای مثال قابلیت حذف کاربر هم اضافه شد.

۱۳- یک قسمت تنظیمات هم برای این فاز ساختیم که علاوه بر تغییر صدا، قابلیت تغییر پشت کارت‌ها، اسم کاربر و ... را هم دارد.

۱۴- تعداد بیشتری کارت علاوه بر چیزهایی که واجب بود زده شده!

۱۵- در آخر نقاط قوتی که از فازهای قبل داشتیم هم حفظ شده اند. برای مثال هش شدن پسورد ها و ذخیره نکردن خود پسورد اکانت ها هم در این فاز رعایت شده.

نقاط ضعف :

۱- قسمت اطلاعات کامپوننت‌های گرافیکی بسیار زیاد است، این به خاطر این است که هر کامپوننت را در جای مشخصی قرار داده‌ایم و چون امکانات بهتری در اختیار نبود، برای همه کامپوننت‌ها مختصاتشان را در اول هر کلاس به صورت دو متغیر استاتیک فاینال نوشته ایم. البته دقت کنید این مقادیر کانفیگ نیستند. چون دارای محاسبات ریاضی هستند و مکان هر کامپوننت بر حسب مکان بقیه کامپوننت‌ها تعیین می‌شود.

توضیحات :

تغییراتی که با بازی اصلی و شاید داک توضیحات شاهد هستیم :

- اول اینکه تعریف کارت قفل کارتی است که کاربر نمیتواند آن را استفاده کند (ممکن است آن را داشته باشد ولی قابل استفاده نیست). برای قابل استفاده شدن این نوع کارت ها، کاربر باید بازی کند (و شاید در آینده مراحل را طی کند که احتمالاً اضافه خواهد شد). پس شما نمیتوانید یک کارت قفل را به دک خود اضافه کنید!

- مورد دوم اینکه هر قهرمان تعدادی دک برای خود دارد (که محدودیت هفت دک برای این فاز قرار داده شده است). پس عوض کردن قهرمان برای یک دک یک چیز بی معنا است و آن را پیاده نکردیم.

برای شروع بازی ابتدا باید اکانت خود را بسازید. سپس به بخش کالکشن رفته و قهرمان خود را انتخاب کنید. برای قهرمان خود یک دک بسازید و دک خود را بچینید (البته باید دک خود را انتخاب هم بکنید!). حال شما آماده بازی با خودتان هستید!

البته می توانید دک مورد نظر خود را به همراه کارت های آن بنویسید، با استفاده از دک ریدر آن را فراخوانی کنید و بازی را به راحتی شروع کنید!

Java Version : 11

Project Build Tool : Maven

External Libraries :

com.google.gson : 2.8.5

منابع اصلی استفاده شده :

<https://www.geeksforgeeks.org/>

<https://stackoverflow.com/>

<https://www7.lunapic.com/>

<https://online-audio-converter.com/>

<https://crop-circle.imageonline.co/>

کلاس CustomScrollBarUI از گیت هاب کپی شده که متأسفانه لینک آن را ذخیره نکردیم.

در آخر تشکر میکنیم از سید مجتبی مدرسی (@Xamarin_Developer) برای کمک هایش در دیزاین و گرافیک برنامه. به طوری که گرافیک زیبای برنامه رو مدیون ایشان و مشاوره هاشون هستیم.

Designed by AliTavassoly & XamarinDev