منبع تصاویر کارت ها، هیروها؛ پسیوها :

<http://www.hearthcards.net/>

منبع دکمه ها:

<https://www.imagefu.com/create/button>

منبع بک گراند:

<https://www.google.com/>

منبع HintTextField :

<https://stackoverflow.com/>

کتابخانه های external :

Gson 2.8.2

Org.imgsclar 4.2 ( Resizing image)

کتابخانه های internal :

Java.swing

Java.awt

Java.awt.event.MouseAdaptor

Java.awt.event.MouseEvent

Java.util

Java 10.0.2

روش کار :

اول در مورد رابط گرافیک برنامه . یک jframe کلی وطیفه نشون دادن همه پنل هارو بر عهده داره . همچنین این jframe وظیفه تغییر پنل نشون داده شده رو هم برعهده داره . هر صفحه بازی برای ابزار و تصاویر موجود در خودش یک کلاس جدا داره مثلا StorePage برای قسمت store هست که داخل خودش یکسری button , lable داره که به خودش که پنله ادد میشه . حالا همه این کلاس ها ، کلاس UIState رو اکستند میکنن که خود UIState میاد و Jpanel رو اکستند میکنه پس در واقع همه این کلاس ها jpanel هستن. Jframe هم میاد با تابع changeState وضعیت خودش رو از pre به next میبره و بعد برگشتن از next دوباره pre رو بر روی خودش قرار میده.

حالا برای قرار دادن next میاد runState این پنل رو صدا میزنه . برای هر کلاس که قراره ران شه این متد runState وظیفه پردازش اطلاعات اون صفحه رو برعهده داره و از یک while(true) تشکیل شده که تا وقتی دکمه برگشت زده نشده بر return نمیکنه.

در مورد button ها هم اینکه بعد از کلیک کردن روشون ، معمولا یک متغیر تغییر میکنه و همزمان داخل runState چک میشه و کار مربوط رو انجام میده مثلا دوباره با changeState درون jframe بازیکنو به صفحه جدید منتقل میکنه

تمام تلاش هم شده که گرافیک از لاجیک جدا باشه و همه درخواستایی که به لاجیک مربوط باشن میرن و متد مربوط داخل کلاس خودشون صدا میزنن و ارور یا هرچی دیگه رو برمیگردونن و کلا تو پکج UI هیچ کار مربوط به لاجیک انجام نمیشه

برای اپدیت راحتتر هر کلاس تصمیم گرفتم کلا singleton رو داخل UI استفاده نکنم ولی GameState و GameFrame بصورت singleton .