EVENTS:

On showing up on screen for first time (on rented)

When color palette is altered, text styles will be updated. Maybe put in functionality that allows text styles to be bound to observables. Hell maybe make them observables?

Pointer events:

On mouse entered

On mouse down

On mouse exited

On data changed (for tooltips?) pass thru data

Hierarchical events?

**Typed Pool: ElementPool<Element>**

External Actors:

Highlights (the big flashy bars that show what is selected)

External Factors:

Many elements movement has delay delta defined

**Don’t disable anything, just move them offscreen (Don’t disable until they are offscreen)**

The rect transform of the currently highlighted element (if there is one)

The screen position of the mouse

Tab selection changes where the elements come from offscreen

Avoid using any canvas layouts and such

Internal Factors:

Boolean for flagging when the ui element is still moving

Design choices:

Drop shadow for Stats like Attack, defence, health(?), magic

Maybe we could use the principle of the scriptable object wrapping for things and wrap up live data using scriptable objects.

Character data structure first. Then display this in a menu

Character:

IName, IDescription, INextActionDescription

Player:

Health, ActionPoints, Stamina

Area:

IName (e.g. dark hallway, lake)

IDescription (implementation should do encounter.Name + resourceArea.Name)

Encounter (nullable)

ResourceArea (nullable)

Encounter:

List characters

IName (e.g. should reflect a sentence starting with “You can see” -> discarded corpses twitching on ground, a sleeping dragon, thousands of freshly dug tunnels lining the walls. You can hear what sound like scratching noises echoing through the area. , etc.)

IDescription (e.g. should reflect the before battle beginning -> one by one the corpses start to stand, pulled by some invisible force. The dragon awakes to the sound of your footsteps echoing closer. The scratching sounds grow louder eventually turning into what sounds like an avalanche as small creatures come pouring out of the tunnels.)

Bool PlayerGetsFirstAction (lets player disengage from encounter)

ResourceArea:

IName (e.g. forest, lake)

IDescription (e.g. You are in a dense forest)

//link to a canvas window minigame

Interfaces

IName

IDescription

INextActionDescription