Game

Assumptions:

UI iterates all elements and blits its respective textures.

ELEMENTS states behaviour of element (defined in Screen Module). GUI module detects hover, click and focus through recursive methods and informs the scene module, which has methods to respond to these events.

There are different structs to store Button and Label Data, and Panel, Slider, Scrollbar use Button's struct.

All Ui Elements are defined in an xml, and loaded in the gui module through DeployUi method, called by Scene Module.

All Elements trigger behaviour is defined in Scene Module

GUI Module

- + Start()
- + Awake(xml node)
- + CleanUp()
- + PreUpdate()
- + Update ()
- + bool isInbound(rect &rect);
- + bool isClicked(rect &rect);
- + bool Colorize(texture, color):
- + void DeColorize(texture);
- + UI Element* CreateButton(ButtonInfo &Data)
- + ButtonInfo FillButton(pugi::xml_node &Ulconfig
- + UI Element* CreateLabel(Text &Data):
- + Text FillLabel(pugi::xml_node &Ulconfig);
- + bool RecursiveOnDrag(List item<UI Element*
- + bool RecursiveOnHover(List item<UI_Element'
- + bool RecursiveOnClick(List item<UI Element*)
- + void DebugDraw();
- + void DeployUI(pugi::xml node &Ulconfig)
- + List<UI Element*> Ulelements
- + iPoint MousePos
- + iPoint onClickPos
- + ButtonInfo Type1Data
- + Button* focus = nullptr;
- + int focus index;
- + iPoint dragRef
- +UI Element focus:

UI Element

- + virtual void FixedUpdate ()
- + virtual Getters ()
- + UI Element* Getparent ()
- + UI Element*Getchild()
- + bool * GetActive()
- + int GetChildrenCount()
- + enum class ELEMENTS
- + struct class Booleans
- + struct Buttonrects
- + struct TextBox
- + iPoint position
- + List <UI_Elements*> children
- + UI_Element* parent
- + bool active

Button: public UI Element

- + void FixedUpdate ()
- + ELEMENTS GetType() override;
- + Buttonrects* Getrects() override;
- + Booleans * GetBooleans() override;
- + SDL_Texture* GetTexture();
- + struct ButtonInfo
- + ButtonInfo Data

Label: public UI Element

- + void FixedUpdate ()
- + inline void PlaceAtTop(ButtonInfo &Data);
- + inline void PlaceAtBottom(ButtonInfo &Data);
- + inline void PlaceAtLeft(ButtonInfo &Data);
- + inline void PlaceAtRight(ButtonInfo &Data);
- + inline void PlaceAtMiddle(ButtonInfo &Data);
- + ELEMENTS GetType() override:
- + Booleans * GetBooleans() override:
- + Buttonrects* Getrects() override;
- + textbox * GetTexts() override;
- + SDL_Texture* GetTexture();
- + void ShapeLabel(const char *text);

Text Data;