Bug List UI SRW

- Ao redimensionar a tela, paines não recalculam suas coordenadas de acordo com a resolução atual já janela. (Possível falha no algoritmo de responsividade). [Resolvido]

- Painel scrollbar possuem dimensões indefinidas(não afeta a funcionalidade). [Pendente]

- Ao invocar um painel a partir dos botões do painel “menu”, o botão “close” do cabeçalho do painel invocado está definido com blend = vec3(1.0f). (Possívelmente on, out e over não estão iniciados conforme o pretendido).

- Para paineis que possui scrollbar, ao abrir o painel pela primeira vez o processamento de ventos ainda retorna resultados para partes dos objetos UI pertencentes ao painel que não estão contidos na regão do painel de visualização. (Possivelmente uma questão de que as coordenadas são efetivente atualizadas respeitando a região do painel viewport após lançamento de eventos do mouse para fila de eventos).[Resolvido]

- Para paines scrollbar, após invocá-lo e torná-lo oculto, depois de resetar coordenadas para as posições iniciais, o próximo evento de rolagem aplica deslocamentos maiores do que o pretendido de modo que corresponde a definir a rolagem a partir da ultima posição definida dos objetos. [Resolvido]

- Em modo janela, de alguns objetos a borda apresenta falhas. (Desconhecido)

New Features

- Resolver alinhamentos de objetos dentro de um mesmo objeto container de forma autonoma.

- Arrasto de paineis.

- Enumerações para determinar uso de bordas de paineis.

- Opções de shaders para botões.

- Botão scroll para paines scroll

- Auto padding para paineis

- Labels

Current features

Os objetos UI da namespace srw são baseados em uma árvore hierarquica que tem como topo a classe abstrata UI. UI herda publicamente de Shader e define as seguintes funcionalidades e membros comuns a todo objeto derivado:

Descrição dos atributos da classe UI

Atributos relacionais

UI\* parent: parent pode ser qualquer objeto derivado de UI. Funciona como um objeto container do qual objetos pertencentes a ele herdam seus estados como: se oculto, posição e dimensões relativas ao container.

Se um objeto declarar ter um container, o container o definirá como objeto filho.

vector<UI\*> child: child é todo objeto que declara ter um objeto container (parent). Um container pode possuir diversas childs. Os eventos de objetos child sempre serão processados a partir do objeto container. childs sempre estarão restritas ao objeto container.

UI\* summon: summon define uma relação semantica não restritiva em o objeto invocador tem acesso aos dados do objeto summon.

UI\* summoner: em alguns casos, é preciso acessar os dados do objeto summoner a partir do objeto summon.

Atributos caracteristicos

vec2f position: define a posição do objeto a tela/containeres.

vec2f ratio: a proporção usada para calcular position.

vec2f currScale: a proporção usada para calcular as dimensões do objeto;

int currWidth: a largura já calculada do objeto.

int currHeight: a altura já calculada do objeto.

bool hided: define se o objeto é visível ao usuário ou se está oculto.

bool inside: define a região válida sobre a qual o mouse emite eventos.

Rótulos

Um rótulo define um nome para um objeto UI. É possivel torna-lo visível ou omiti-lo. Um rotulo é composto por uma pequena série de dados sendo:

string uiName: a string que armazena o nome do rótulo.

string fontFamily: a string que armazena o nome da fonte usada no rótulo.

float fontSize: o tamanho da fonte.

Text text: o objeto agregado do tipo Text que é especializado em renderizar textos de uma linha.

Objetos OpenGL

uint VAO: o identificador para um objeto OpenGL que serve como matriz de estados.

uint VAO: o identificador para um objeto OpenGL que serve como buffer de vértices.

Parâmetros de shaders

Há ainda atributos projetados para fornecer informações pertinentes ao sahders de fragmento usados pelos objetos UI. Sendo:

vec3f on: define a intensidade da cor de um objeto quando ativo.

vec3 out: define a intensidade da cor um obejto quando não há eventos sendo emitidos por ele.

vec3 over:define a intensidade da cor de um objeto quando o mouse está sobre a região ativa do objeto.

float borderMin: para paineis scrollbar, define os limites da viewport. Exceder borderMin significa discartar a parte do objeto fora da borda miníma.

float borderMax: para paineis scrollbar, define os limites da viewport. Exceder borderMax significa discartar a parte do objeto fora da borda máxima.

bool resetActiveRegion: um sinalizador usado para tornar a região ativa de um objeto devolta ao padrão (os valores iniciais).

bool enableUI: um sinalizador que rege se os eventos de um objeto serão processados.

bool lockedUI: um sinalizador definido para auxiliar a semantica de enableUI. Defini-lô resulta em compartamentos singulares.

mat4<float> uiMatrix: a matriz compartilhada por todas instâncias de UI que converte coordenadas de janela em coordenadas de clip.

Fila de eventos

Por uma questão de projeto, os objetos UI são colocados em um vetor que atua como uma fila de eventos para os objetos. A ordem em que os objetos estão ordenados deve ser cuidada de acordo com a regra da classificação da profundidade.

A quatro níveis de processamento de eventos:

Nível 1: Os objetos no primeiro nivel(containers) são processados.

Nivel 2: Os objetos na primeira camada desviam o fluxo de processamento para seus objetos summon.

Nivel 3: Os objetos na primeira camada desviam o fluxo de processamento para seus filhos.

Nível 4: Os objetos na terceira camada desviam para o fluxo de processamento para seus filhos.

Os níveis devem ir aumentando conforme as camadas na relação (pai – filho/ summon-summoner) forem aumentando.