

Gui

p2List<j1UIItems\*> items;

SDL\_Texture\* atlas;

p2SSString atlas\_file\_name;

bool debug = false;

bool Awake(pugi::xml\_node&);

bool Start();

bool PreUpdate();

bool Update(float dt);

bool PostUpdate();

bool CleanUp();

SDL\_Texture\* GetAtlas() const;

ImageUI\* CreateImage(Position ali, char\* path, SDL\_Rect texture = { 0,0,0,0 }, iPoint movement = { 0,0 }, j1Module\* auxmodule = nullptr);

ButtonUI\* CreateButton(Position ali, p2SSString text, iPoint movement = { 0,0 }, j1Module\* auxmodule = nullptr);

LabelUI\* CreateText(Position ali, p2SSString text, iPoint movement = { 0,0 }, Fonts = ALLER\_LT, SDL\_Color color = { 255,255,255,255 }, j1Module\* auxmodule = nullptr);

WindowUI\* WindowCreate(Position ali, uint num\_buttons, p2SSString title, iPoint movement = { 0,0 }, j1Module\* auxmodule = nullptr);

SliderUI\* CreateSlider(Position ali, p2SSString text, iPoint movement = { 0,0 }, j1Module\* auxmodule = nullptr);

UI Items

Type guitype;

SDL\_Texture\* TexLoad(p2SSString path);

SDL\_Texture\* texture = nullptr;

SDL\_Rect rect;

iPoint position;

iPoint movement;

j1Module\* auxmodule;

bool show;

bool deleting;

bool kinetic;

bool debug;

iPoint OnClick;

void UpdatePosition();

virtual void ShowDebug();

virtual void Move(iPoint movement);

virtual void MouseInput(Input state);

bool Blit() const

Image UI

void Move(iPoint Movement);

Slider UI

SDL\_Texture \* Rail

SDL\_Texture \* None

SDL\_Texture\* Over

SDL\_Texture\* Pressed

bool active

bool movable

iPoint pos

iPoint limit

void StartDrag()

void Drag()

void Enddrag();

void MouseInput(Input state);

void ChangeEvent(j1UIItems\* item, Event event)

void Move(iPoint movement)

void SettleTextureToSlider(p2SSString path\_idle, p2SSString path\_hover = nullptr, p2SSString path\_pressed = nullptr)

Button UI

SDL\_Texture \* None

SDL\_Texture\* Over

SDL\_Texture\* Pressed

bool active

void MouseInput(Input state);

void ChangeEvent(j1UIItems\* item, Event event);

void Move(iPoint movement);

void SettleTextureToButton(p2SSString path\_idle, p2SSString path\_hover, p2SSString path\_pressed = nullptr);

Label UI

p2SSString text

SDL\_Color text\_color

\_TTF\_Font\* text\_font

void CreateText(p2SSString txt, SDL\_Color color, Fonts font)

void UpdateText(p2SSString newtext)

void Move(iPoint displace)

Window UI

p2SSString title;

uint ButtonCount;

bool movable

p2List<j1UIItems\*> ItemWindow;

p2List<ButtonUI\*> ButtWindow;

p2List<p2SSString> TextBttWindow;

LabelUI\* TextWindow

void Move()

void MoveStart()

void MoveEnd()

void WindowItemsMove(iPoint movement)

