ArioA MineSweeper

AUTHOR: Ario Amin
Version Obfuscated
Mon Mar 29 2021

Hierarchical Index

Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Class Index

Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

File Index

File List

Here is a list of all documented files with brief descriptions:

Class Documentation

FCoordsT Struct Reference

Simple 2d Coordinate struct.

`#include <MineSecret.h>`

Public Member Functions

FCoordsT **operator=** (const FCoordsT &C2)
FCoordsT **operator+** (const FCoordsT &C2) const
FCoordsT **operator-** (const FCoordsT &C2) const
FCoordsT **operator*** (const FCoordsT &C2) const
FCoordsT **operator/** (const FCoordsT &C2) const
FCoordsT **operator%** (const FCoordsT &C2) const
FCoordsT **operator+=** (const FCoordsT &C2)
FCoordsT **operator-=** (const FCoordsT &C2)
FCoordsT **operator*=** (const FCoordsT &C2)
FCoordsT **operator/=** (const FCoordsT &C2)
FCoordsT **operator%=** (const FCoordsT &C2)
bool **operator==** (const FCoordsT &C2) const
bool **operator!=** (const FCoordsT &C2) const
bool **operator<** (const FCoordsT &C2) const
bool **operator>** (const FCoordsT &C2) const
bool **operator<=** (const FCoordsT &C2) const
bool **operator>=** (const FCoordsT &C2) const

Public Attributes

uint16 **X**
uint16 **Y**

Detailed Description

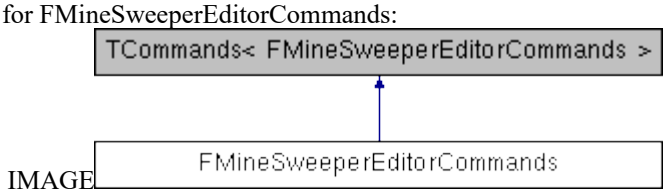
Simple 2d Coordinate struct.

Compiler issues due to constrained hardware, could not create proper default operators, It wasn't doing the logical nor arithmetic operations properly in some cases and in other it was just getting errors. Errors of mismatch types and undefined operators, so I decided I had to define some myself, and fix whatever ails my computer later.

The documentation for this struct was generated from the following file:
MineSweeperEditor/Public/MineSecret.h

FMineSweeperEditorCommands Class Reference

Inheritance diagram for FMineSweeperEditorCommands:



IMAGE

Public Member Functions

virtual void **RegisterCommands** () override

Public Attributes

TSharedPtr< FUICommandInfo > **WindowContext**

The documentation for this class was generated from the following files:

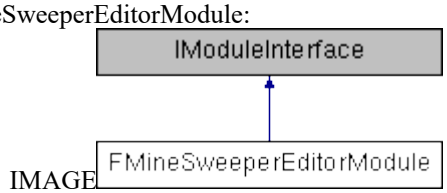
MineSweeperEditor/Public/MineSweeperEditorCommands.h
MineSweeperEditor/Private/MineSweeperEditorCommands.cpp

FMineSweeperEditorModule Class Reference

Minesweeper Editor-module.

`#include <MineSweeperEditor.h>`

Inheritance diagram for FMineSweeperEditorModule:



IMAGE

Public Member Functions

FMineSweeperEditorModule ()
FMineSweeperEditorModule constructor.

virtual void StartupModule () override
StatupModule - IModuleInterface override.

virtual void ShutdownModule () override
Shutdown - IModuleInterface override.

void TabBtnClicked () const
Brings up main plugin window.

void CommittedX (const uint8 NewInt, ETextCommit::Type CommitType) const
Commit value from slider.

void CommittedY (const uint8 NewInt, ETextCommit::Type CommitType) const
Commit text from text-box 1 to value.

void RegenerateGrid (uint8 Xin, uint8 Yin, TSharedRef< class SUniformGridPanel >) const
Regenerates an already existing Slate Grid.

Public Attributes

TSharedPtr< FSysManager > SysManager
System Manager.

Detailed Description

Minesweeper Editor-module.

This is the editor module which is responsible for initializing slate elements. (That regards the related editor window and it's elements)

Constructor & Destructor Documentation

FMineSweeperEditorModule::FMineSweeperEditorModule ()

FMineSweeperEditorModule constructor.

Creates a FSysManager and loads total saved score

Member Function Documentation

void FMineSweeperEditorModule::CommittedX (const uint8 *NewInt*, ETextCommit::Type *CommitType*) const

Commit value from slider.

Commits value to this->SysManager->NextRowSize, is bound to an SSlider which is defined in OnSpawnTab

Parameters

NewInt	New display value
CommitType	

Note

set value when keyboard input

void FMineSweeperEditorModule::CommittedY (const uint8 *NewInt*, ETextCommit::Type *CommitType*) const

Commit text from text-box 1 to value.

Parameters

NewInt	New display value
CommitType	

Note

set value when keyboard input

void FMineSweeperEditorModule::RegenerateGrid (uint8 *Xin*, uint8 *Yin*, TSharedRef< class SUniformGridPanel >) const

Regenerates an already existing Slate Grid.

Parameters

Xin	Maximum size X
Yin	Maximum size Y

Note

FSLocal - Local container to create and bind function

FSLocal::OnTileClick(const Coords, TSharedPtr<FSysManager>)

FSLocal::MakeTile(const Coords, TSharedPtr<FSysManager>)

Returns

Shared reference of Grid panel, type: TSharedRef<SUniformGridPanel>

void FMineSweeperEditorModule::ShutdownModule () [override], [virtual]

Shutdown - IModuleInterface override.

De-initializes the style-set and command, as-well as triggers SaveState in FSysManager, then de-registers callback and the new tab in the editor menu.

void FMineSweeperEditorModule::StartupModule () [override], [virtual]

StatupModule - IModuleInterface override.

Initializes the style-set and command, then registers callback and the new tab in the editor menu.

void FMineSweeperEditorModule::TabBtnClicked () const

Brings up main plugin window.

Tries to invoke tab through FGlobalTabmanager:: and then Triggers the New Game event

Note

This function will be bound to Command

Member Data Documentation

TSharedPtr<FSysManager> FMineSweeperEditorModule::SysManager

System Manager.

A shared pointer to an FSysManager, which manages much of the game and the system functionality.

In contrast to FMineSweeperEditorModule, which mainly is used for initializing the visual elements, as-well as binding them through FTileBinder.

The documentation for this class was generated from the following files:

MineSweeperEditor/Public/MineSweeperEditor.h
MineSweeperEditor/Private/MineSweeperEditor.cpp

FMineSweeperEditorStyle Class Reference

`#include <MineSweeperEditorStyle.h>`

Public Member Functions

template<EBtnStyles BtnStyle> TSharedRef< FSlateStyleSet > **CreateBtn** ()

Static Public Member Functions

static void **Init** ()
static void **Shutdown** ()
static void **ReloadTextures** ()
static const ISlateStyle & Get ()
static FName GetName ()

Detailed Description

MineSweeper button Slate Style

Member Function Documentation

const ISlateStyle & FMineSweeperEditorStyle::Get () `[static]`

reloads textures used by slate renderer

FName FMineSweeperEditorStyle::GetName () `[static]`

Returns

The Slate style set

The documentation for this class was generated from the following files:

MineSweeperEditor/Public/MineSweeperEditorStyle.h
MineSweeperEditor/Private/MineSweeperEditorStyle.cpp

FObfuscator Class Reference

Secret class, kindly ignore this :)

`#include <FSysManager.h>`

Public Member Functions

void **VC** ()
void **PC** ()
void **BW** ()
void **DW** ()
void **BC** () const
void **CB** () const
void **BF** () const
uint8 & **FK** ()
uint8 & **HG** ()
bool **CC** () const
bool **DC** () const
template<uint16 BitField, uint16 Bit2Field, uint16 Bit4Field, uint16 Bit8Field, uint16 Bit16Field> bool **SCW** ()
uint8 & **SC** ()
void **ObfscDobfsc** (TSharedPtr< FSysManager > ManagerShared)
void **DobfscObfsc** (TSharedPtr< FSysManager > ManagerShared, FSysManager::EGameState)
bool **VH** ()
bool **KP** ()
bool **MW** ()
bool **WV** ()
uint8 & **GS** ()

Static Public Member Functions

template<uint16 BitField> static bool **Obfsc** (const FCoords TileCoords, const uint16 Fieldval)
static void **Binder** (FString &Garble, FString &ReturnParam)
static void Flipper (FString &Flipper, FString &ReturnParam)

Detailed Description

Secret class, kindly ignore this :)

This is a class to be ignored. Will not contain any documentation.

Member Function Documentation

static void FObfuscator::Flipper (FString & *Flipper*, FString & *ReturnParam*) `[inline], [static]`

Global flipper

The documentation for this class was generated from the following files:

MineSweeperEditor/Public/FSysManager.h
MineSweeperEditor/Private/FSysManager.cpp

FSysManager Class Reference

General resource and game manager.

`#include <FSysManager.h>`

Public Types

enum EGameDifficulty : uint8 { **Easy** = 0x0, **Normal** = 0x1, **Hard** = 0x2, **Insane** = 0x3 }
Game-difficulty enum @type EGameDifficulty : uint8.

enum EGameState : uint8 { **L** = 0x0, **W** = 0x1, **P** = 0x2 }
Game-state enum @type EGameState : uint8.

enum EBitField : uint8 { **IsMine** = 0x0, **IsClicked** = 0x1, **HasFlag** = 0x2, **HasQuestion** = 0x3, **NeighbourMines** = 0x4 }
Data-field enum @type EBitField : uint8.

enum EPrivateMember : uint8 { **BoolPlayAgain** = 0x0, **VectorSlateGrid** = 0x1, **VectorDifficultyList** = 0x2, **VectorTileDisplayGrid** = 0x3, **TOptGridWidgetRef** = 0x4, **FObfsctr** = 0x5, **STextEndMsgRef** = 0x6, **STextStatsRef** = 0x7, **STextScoreRef** = 0x8 }
Private-member-name enum @type EPrivateMember : uint8.

Public Member Functions

FSysManager ()
FSysManager constructor.

void InitBtnSBrush ()
Setting the FSlateImageBrushes with actual images.

void UpdateScoreWidget ()
Updating Score Widget.

TOptional< uint16 > DisplayColSize () const
Displays Column Size for widget.

TOptional< uint16 > DisplayRowSize () const
Displays Row Size for widget.

FText DisplayEndMsg () const
Returns End-game message for widget.

void RowSizeCommitted (float NewRowSize)
Assigns 'NewRowSize' to this->NextRowSize.

void ColSizeCommitted (float NewColSize)
Assigns 'NewColSize' to this->NextColSize.

void UpdateGridSize ()
Updates Grid Size variables.

TSharedRef< SButton > GetGridFSlot (FCoords TileCoords)

Get reference to specific Slate SUniformGridPanel::FSlot.

TSharedRef< STextBlock > GetTileTextBlock (FCoords TileCoords)
Get reference to specific Slate Text Block.

EGameState ClickTile (const FCoords TileCoords)
Click Tile.

template<EBitField BitField> uint8 GetAttributes (const FCoords TileCoords) const
Get FSysManager Attributes.

template<EBitField BitField> void SetAttributes (const FCoords TileCoords, const uint8 Fieldval)
Set FSysManager Attributes.

template<EPrivateMember PrivateMember> auto GetPrivateMemberRef () -> auto &
Get FSysManager Private Member References.

void SaveState () const
Save session scores.

void LoadState ()
Loads saved score.

void ResetGame ()
Reset game / end game Pretty self-explanatory.

void RestartGame ()
Restart game / end game. Pretty self-explanatory.

void FSetNextDiff (EGameDifficulty NextDiff)
Updates NextDifficulty Variable.

Public Attributes

TSharedPtr< FSlateImageBrush > **FlagBrush**
TSharedPtr< FSlateImageBrush > **QuestionBrush**
TSharedPtr< FSlateImageBrush > **BombBrush**
uint16 **FreeTilesCount** = 0x0
uint16 **ClickedTiles** = 0x0
uint16 **CurrRowSize** = 0x5
uint16 **CurrColSize** = 0x5
uint16 **Ws** = 0x0
uint16 **Ls** = 0x0
FString **SContainer** = FString(TEXT(""))
FString **RContainer** = FString(TEXT(""))
std::array< std::array< uint8, Gmax_Size >, Gmax_Size > GridData = {0}
Grid Data array.

Static Public Attributes

static constexpr uint16 **Gmax_Size** = 0x40

Detailed Description

General resource and game manager.

Is the general Systems and Game manager. It enforces the rules for the game, loads/saves total win/loss, And it interacts with the Slate-widgets on the game-windows

---

Constructor & Destructor Documentation

FSysManager::FSysManager ()

FSysManager constructor.

Loads some FStrings into a TArray<FString>, then constructing some shared pointers to some of FSysManager members

---

Member Function Documentation

FSysManager::EGameState FSysManager::ClickTile (const FCoords *TileCoords*)

Click Tile.

Parameters

TileCoords	Coordinato to tile to be clicked

void FSysManager::ColSizeCommitted (float *NewColSize*)

Assigns 'NewColSize' to this->NextColSize.

Is used to bind to an STextBlock through .Text_Raw()

Parameters

NewColSize	

TOptional< uint16 > FSysManager::DisplayColSize () const

Displays Column Size for widget.

Is used to bind to an SNumericEntryBox through .Value()

Returns

TOptional<uint16> Optional type which holds, if anything, a uint16.

FText FSysManager::DisplayEndMsg () const

Returns End-game message for widget.

Is used to bind to an STextBlock through .Text_Raw()

Returns

FText containing the End-game message

TOptional< uint16 > FSysManager::DisplayRowSize () const

Displays Row Size for widget.

Is used to bind to an SNumericEntryBox through .Value()

Returns

TOptional<uint16> Optional type which holds, if anything, a uint16.

template<FSysManager::EBitField BitField> uint8 FSysManager::GetAttributes (const FCoords *TileCoords*) const

Get FSysManager Attributes.

Templated attributes getter/setter functions

Parameters

TileCoords	Struct with coords for tile to change attribute value in

Template Parameters

BitField	Enum of type FSysManager::EBitField Which is an enum that has fields for each attribute

TSharedRef< SButton > FSysManager::GetGridFSlot (FCoords *TileCoords*)

Get reference to specific Slate SUniformGridPanel::FSlot.

Parameters

TileCoords	Position struct, x & y coordinates

template<FSysManager::EPrivateMember PrivateMember> auto FSysManager::GetPrivateMemberRef

Get FSysManager Private Member References.

Template Parameters

PrivateMember	Enum of type FSysManager::EPrivateMember Which is an enum that has fields for several of FSysManagers private members

Returns

-> auto&, Returns reference to given private member

TSharedRef< STextBlock > FSysManager::GetTileTextBlock (FCoords *TileCoords*)

Get reference to specific Slate Text Block.

Parameters

TileCoords	Position struct, x & y coordinates

void FSysManager::LoadState ()

Loads saved score.

Loads score from file, in plugin/MineSweeperEditor/Resources/data/ If directory/file do not exist, then it will not load any data into the total score.

void FSysManager::RowSizeCommitted (float *NewRowSize*)

Assigns 'NewRowSize' to this->NextRowSize.

Is used to bind to an STextBlock through .Text_Raw()

void FSysManager::SaveState () const

Save session scores.

Saves sessions score to file, in plugin/MineSweeperEditor/Resources/data/ If directory does not exist, then it will be created.

template<FSysManager::EBitField BitField> void FSysManager::SetAttributes (const FCoords *TileCoords*, const uint8 *Fieldval*)

Set FSysManager Attributes.

Parameters

TileCoords	Struct with coords for tile to change attribute value in
Fieldval	Actual value to be set in the attribute bit-field

Template Parameters

BitField	Parameter is an enum of type EBitField. Which is an enum that has fields for each attribute

void FSysManager::UpdateGridSize ()

Updates Grid Size variables.

Updates the members CurrRowSize and CurrColSize. Assigns them the values of NextRowSize & NextColSize.

void FSysManager::UpdateScoreWidget ()

Updating Score Widget.

Updates the STextBlock widget, which is used for scores with current Wins/Losses.

Member Data Documentation

std::array<std::array<uint8, Gmax_Size>, Gmax_Size> FSysManager::GridData = {0}

Grid Data array.

Note

64^2 bytes = 4kb, on a mcu it would be unacceptable, on a pc cpu with megabytes of cache it's negligible

Per element:

bit[0] = isMine?

bit[1] = Clicked?

bit[2] = HasFlag?

bit[3] = HasQuestionMark?

bits[7,4] = Neighbour Mines Count

Don't be scared by the notation above; as this does not regard actual bitfields but instead regard values, and uses bit operators to access these "fields", thus it is still a fully portable solution

---

The documentation for this class was generated from the following files:

MineSweeperEditor/Public/FSysManager.h
MineSweeperEditor/Private/FSysManager.cpp
MineSweeperEditor/Private/MineSweeperEditor.cpp

FTileBinder Struct Reference

Function binder.

`#include <MineSweeperEditor.h>`

Static Public Member Functions

static FReply NewGameBind (const FMineSweeperEditorModule *Owner, TSharedPtr< FSysManager > Manager)

New Game event.

static FReply RestartGameBind (const FMineSweeperEditorModule *Owner, TSharedPtr< FSysManager > Manager)

Restart Board event.

static FReply OnDifficultyClick (FSysManager::EGameDifficulty Difficulty, TSharedPtr< FSysManager > ManagerShared)

OnTileClick, function to bind to game-board tiles.

static FReply OnTileClick (FCoords TileCoords, TSharedPtr< FSysManager > ManagerShared)

OnTileClick, function to bind to game-board tiles.

static TSharedRef< SWidget > MakeTile (const FCoords TileCoords, TSharedPtr< FSysManager > ManagerShared)

Make Tile and bind OnClick to it.

Detailed Description

Function binder.

Implements functions which are then used to bind to delegates in FMineSweeperEditorModule

---

Member Function Documentation

TSharedRef< SWidget > FTileBinder::MakeTile (const FCoords *TileCoords*, TSharedPtr< FSysManager > *ManagerShared*) `[static]`

Make Tile and bind OnClick to it.

Parameters

TileCoords	Given coordinates at which I tile will be created
ManagerShared	FSysManager pointer to pass through to internal function call

Returns

TSharedRef<SWidget>, a shared reference to the tile (SAButton widget)

Make Tile and bind OnClick to it

FReply FTileBinder::NewGameBind (const FMineSweeperEditorModule * *Owner*, TSharedPtr< FSysManager > *Manager*) `[static]`

New Game event.

Starts a new game, uses the slider values to generate new board-dimensions.

Parameters

Owner	
Manager	@retun returns an FReply::Handled() when finished.

Create New Game event

FReply FTileBinder::OnDifficultyClick (FSysManager::EGameDifficulty *Difficulty*, TSharedPtr< FSysManager > *ManagerShared*) `[static]`

OnTileClick, function to bind to game-board tiles.

Parameters

Difficulty	Requested Difficulty Setting
ManagerShared	An FSysManager to trigger it's ClickTile() function @retun returns an FReply::Handled() when finished.

On Difficulty Button Click event

FReply FTileBinder::OnTileClick (FCoords *TileCoords*, TSharedPtr< FSysManager > *ManagerShared*) `[static]`

OnTileClick, function to bind to game-board tiles.

Parameters

TileCoords	Coordinates of the tile which has been clicked
ManagerShared	An FSysManager to trigger it's ClickTile() function @retun returns an FReply::Handled() when finished.

On Tile Click event

FReply FTileBinder::RestartGameBind (const FMineSweeperEditorModule * *Owner*, TSharedPtr< FSysManager > *Manager*) `[static]`

Restart Board event.

Restart the current game. can one be used once per board you're playing.

Parameters

Owner	
Manager	@retun returns an FReply::Handled() when finished.

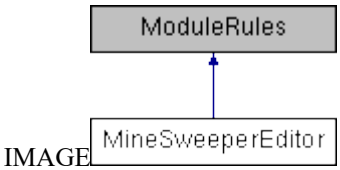
Restart Game event

The documentation for this struct was generated from the following files:

MineSweeperEditor/Public/MineSweeperEditor.h
MineSweeperEditor/Private/MineSweeperEditor.cpp

MineSweeperEditor Class Reference

Inheritance diagram for MineSweeperEditor:



IMAGE

Public Member Functions

MineSweeperEditor (ReadOnlyTargetRules Target)

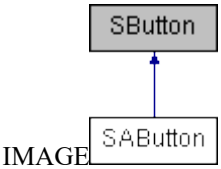
The documentation for this class was generated from the following file:
MineSweeperEditor/MineSweeperEditor.Build.cs

SAButton Class Reference

SButton child class.

`#include <SAButton.h>`

Inheritance diagram for SAButton:



Public Member Functions

void **ExecOnClick** (const FPointerEvent &MouseEvent, FReply &Reply, FKey &LastPressButton)
void **ExecOnRelease** (const FGeometry &MyGeometry, const FPointerEvent &MouseEvent, FReply &Reply, EButtonClickMethod::Type InputClickMethod, FKey &LastPressButton)
bool **MustBePressed** (EButtonClickMethod::Type InputClickMethod)
FReply **RMBExecuteBound** ()
FReply **LMBExecuteBound** ()
virtual FReply **OnMouseButtonDown** (const FGeometry &MyGeometry, const FPointerEvent &MouseEvent) override
virtual FReply **OnMouseButtonUp** (const FGeometry &MyGeometry, const FPointerEvent &MouseEvent) override
virtual FReply **OnMouseButtonDoubleClick** (const FGeometry &InMyGeometry, const FPointerEvent &InMouseEvent) override

Public Attributes

FOnClicked **OnClicked**
FOnClicked **OnRMBClicked**
FOnClicked **OnLMBClicked**

Detailed Description

SButton child class.

Overloads some functions like OnClick to differentiate between LMB and RMB

The documentation for this class was generated from the following files:
MineSweeperEditor/Public/SAButton.h
MineSweeperEditor/Private/SAButton.cpp

File Documentation

MineSweeperEditor/Private/FSysManager.cpp File Reference

```
#include "FSysManager.h"
#include "EditorReimportHandler.h"
#include "Misc/FileHelper.h"
#include "Widgets/Layout/SUniformGridPanel.h"
#include "Interfaces/IPluginManager.h"
#include "Brushes/SlateImageBrush.h"
```

Macros

#define **LOCTEXT_NAMESPACE** "MineSweeperEditorModule"

Detailed Description

Author

Ario Amin @project MineSweeper Geodesic Test

MineSweeperEditor/Private/MineSweeperEditor.cpp File Reference

```
#include "MineSweeperEditor.h"
#include "EditorReimportHandler.h"
#include "MineSweeperEditorStyle.h"
#include "MineSweeperEditorCommands.h"
#include "LevelEditor.h"
#include "Widgets/Docking/SDockTab.h"
#include "Widgets/Layout/SWrapBox.h"
#include "Widgets/Views/SListView.h"
#include "Widgets/Layout/SUniformGridPanel.h"
#include "Widgets/Input/SNumericEntryBox.h"
#include "Widgets/Text/STextBlock.h"
#include "Templates/SharedPointer.h"
#include "Misc/FileHelper.h"
#include "Interfaces/IPluginManager.h"
#include "Slate/SlateTextures.h"
#include "ToolMenus.h"
#include "Widgets/Input/SSlider.h"
#include "Widgets/Layout/SGridPanel.h"
#include "Widgets/Layout/SScrollBox.h"
```

Macros

#define **LOCTEXT_NAMESPACE** "MineSweeperEditorModule"

Detailed Description

Author

Ario Amin @project MineSweeper Geodesic Test

MineSweeperEditor/Private/ MineSweeperEditorCommands.cpp File Reference

```
#include "MineSweeperEditorCommands.h"
```

Macros

#define **LOCTEXT_NAMESPACE** "FMineSweeperEditorModule"

Detailed Description

Author

Ario Amin @project MineSweeper Geodesic Test

MineSweeperEditor/Private/MineSweeperEditorStyle.cpp File Reference

```
#include "MineSweeperEditorStyle.h"
#include "Styling/SlateStyleRegistry.h"
#include "Framework/Application/SlateApplication.h"
#include "Slate/SlateGameResources.h"
#include "Interfaces/IPluginManager.h"
```

Macros

#define **IMAGE_BRUSH**(RelativePath, ...) FSlateImageBrush(Style->RootToContentDir(RelativePath, TEXT(".png")), __VA_ARGS__)
#define **BOX_BRUSH**(RelativePath, ...) FSlateBoxBrush(Style->RootToContentDir(RelativePath, TEXT(".png")), __VA_ARGS__)
#define **BORDER_BRUSH**(RelativePath, ...) FSlateBorderBrush(Style->RootToContentDir(RelativePath, TEXT(".png")), __VA_ARGS__)
#define **TTF_FONT**(RelativePath, ...) FSlateFontInfo(Style->RootToContentDir(RelativePath, TEXT(".ttf")), __VA_ARGS__)
#define **OTF_FONT**(RelativePath, ...) FSlateFontInfo(Style->RootToContentDir(RelativePath, TEXT(".otf")), __VA_ARGS__)

Functions

const FVector2D **Icon16x16** (16.0f, 16.0f)
const FVector2D **Icon20x20** (20.0f, 20.0f)
const FVector2D **Icon40x40** (40.0f, 40.0f)

Detailed Description

Author

Ario Amin @project MineSweeper Geodesic Test

MineSweeperEditor/Public/FSysManager.h File Reference

```
#include "CoreMinimal.h"
#include "Modules/ModuleManager.h"
#include "MineSecret.h"
#include <array>
#include <vector>
```

Classes

class FSysManager

General resource and game manager.

class FObfuscator

Secret class, kindly ignore this :)

Detailed Description

Author

Ario Amin @project MineSweeper Geodesic Test

MineSweeperEditor/Public/MineSecret.h File Reference

#defines and FCoordsT definition

```
#include "Misc/Base64.h"
```

Classes

struct FCoordsT

Simple 2d Coordinate struct.

Macros

#define **MX** "Flym"
#define **M0** "=UWa"
#define **M1** "gMXa"
#define **M2** "sBSY"
#define **M3** "hNGI"
#define **M4** "gU2a"
#define **M5** "yjgD"
#define **M6** "Xtew"
#define **M7** "lhGV"
#define **M_REEE** MX M7
#define **M_REEL** M2 M0
#define **M_RAEL** M2 M1
#define **M_REAL** M4 M3
#define **M_RAAL** M7 M4
#define **M_RAFL** M5 M2
#define **M_RADL** M5 M7
#define **M_REDL** M3 M6
#define **M_ROEL** M3 M1
#define FREETILES FreeTilesCount - ClickedTiles
#define MAKEROBOTO(FontSize) .Font(FSlateFontInfo(FPaths::EngineContentDir() / TEXT("Slate/Fonts/Roboto-Bold.ttf"), FontSize))
#define **ROBOTOARG**(FontSize) (FSlateFontInfo(FPaths::EngineContentDir() / TEXT("Slate/Fonts/Roboto-Bold.ttf"), FontSize))
#define MAKETEXT(InString) .Text(FText::FromString(InString))
#define **TEXTARG**(InString) (FText::FromString(InString))
#define MAKENUMTEXT(InInt) .Text(FText::FromString(FString::FromInt(InInt)))
#define **NUMTEXTARG**(InInt) (FText::FromString(FString::FromInt(InInt)))

Typedefs

using **FDcdeT** = FBase64
using **FDcde** = FDcdeT
using **FCoords** = FCoordsT

Variables

bool **bCh** = false

Detailed Description

#defines and FCoordsT definition

Author

Ario Amin @project MineSweeper Geodesic Test

Macro Definition Documentation

#define FREETILES FreeTilesCount - ClickedTiles

Note

\FREETILES is ONLY to be used within member functions of class FSysManager

#define MAKENUMTEXT(InInt) .Text(FText::FromString(FString::FromInt(InInt)))

Note

\INT_TEXT is ONLY to be used Slate widgets that, when created through SNew(), have the member function .Text(...) as-well as an integer to read as text

#define MAKEROBOTO(FontSize) .Font(FSlateFontInfo(FPaths::EngineContentDir() / TEXT("Slate/Fonts/Roboto-Bold.ttf"), FontSize))

Note

\MAKEROBOTO is ONLY to be used Slate widgets that, when created through SNew(), have the member function .Font(...)

#define MAKETEXT(InString) .Text(FText::FromString(InString))

Note

\MAKETEXT is ONLY to be used Slate widgets that, when created through SNew(), have the member function .Text(...)

MineSweeperEditor/Public/MineSweeperEditor.h File Reference

```
#include "CoreMinimal.h"
#include "FSysManager.h"
#include "Modules/ModuleManager.h"
```

Classes

class FMineSweeperEditorModule

Minesweeper Editor-module.

struct FTileBinder

Function binder.

Detailed Description

Author

Ario Amin @project MineSweeper Geodesic Test

MineSweeperEditor/Public/MineSweeperEditorCommands.h File Reference

```
#include "CoreMinimal.h"
#include "Framework/Commands/Commands.h"
#include "MineSweeperEditorStyle.h"
```

Classes

class FMineSweeperEditorCommands

Detailed Description

Author

Ario Amin @project MineSweeper Geodesic Test

MineSweeperEditor/Public/MineSweeperEditorStyle.h File Reference

```
#include "CoreMinimal.h"
#include "Styling/SlateStyle.h"
```

Classes

class FMineSweeperEditorStyle

Enumerations

enum class **EBtnStyles** : uint8 { **BtnFlag**, **BtnQ**, **BtnBomb**, **BtnInt** }

Detailed Description

Author

Ario Amin @project MineSweeper Geodesic Test

MineSweeperEditor/Public/SAButton.h File Reference

```
#include "Widgets/Input/SButton.h"
#include "FSysManager.h"
```

Classes

class SAButton

SButton child class.

Detailed Description

Author

Ario Amin @project MineSweeper Geodesic Test

Index

INDEX