**Порядок действий для добавления нового элемента в сохранение игры.**

Пусть новый класс, который добавляется в сохранения, называется ANewClass.

1. В файле

MagicTrigger\Source\MagicTrigger\SaveGame\SaveGameManager.h

1. добавить объявление нового класса в шапке class ANewClass;
2. добавить объявление указателя на объект нового класса ANewClass\* NewClass
3. добавить объявление нового таймера FTimerHandle NewClassTimer;
4. увеличить на 1 MaxCountOfLoadedElements
5. добавить методы

void LoadNewClass(USaveGameMT\* InLoadingGame);

void SaveNewClass(USaveGameMT\* InSavingGame);

1. В файле

MagicTrigger\Source\MagicTrigger\SaveGame\SaveGameManager.cpp

1. добавить пустые определения методов

void USaveGameManager::LoadNewClass(USaveGameMT\* InLoadingGame)

{

}

void USaveGameManager::SaveNewClass(USaveGameMT\* InSavingGame)

{

}

заполнять определения методов нужно после всех нижеследующих действий.

1. Добавить инклуд для нового класса

#include "MagicTrigger\SomeFolder\NewClass.h"

1. в метод

void USaveGameManager::LoadAll(USaveGameMT\* InLoadingGame)

добавить вызов

LoadNewClass(InLoadingGame);

1. в метод

void USaveGameManager::SaveAll(USaveGameMT\* InSavingGame)

добавить вызов

SaveNewClass(InSavingGame);

1. В папке

MagicTrigger\Source\MagicTrigger\Data\SaveGame\

1. Создать новый файл

NewClassSaveGameStruct.h

для структуры

FNewClassSaveGameStruct

которая будет содержать сохраняемые данные класса ANewClass

1. Заполнить NewClassSaveGameStruct.h по шаблону подобных структур в папке.
2. В файле

MagicTrigger\Source\MagicTrigger\SaveGame\SaveGameMT.h

1. добавить инклуд

#include "MagicTrigger\Data\SaveGame\NewClassSaveGameStruct.h"

1. добавить объявление

UPROPERTY()

FNewClassGameStruct NewClassSaveGameStruct;

1. В файле

"MagicTrigger\SomeFolder\NewClass.h"

добавить объявление метода

void SendThisToSaveGameManager();

1. В файле

"MagicTrigger\SomeFolder\NewClass.cpp"

1. добавить инклуды

#include "MagicTrigger\SaveGame\SaveGameManager.h"

#include "MagicTrigger\CoreClasses\GameInstanceMagicTrigger.h"

#include "MagicTrigger\Data\DebugMessage.h"

#include "Kismet\GameplayStatics.h"

1. определить новый метод

void ANewClass:: SendThisToSaveGameManager ()

{

UGameInstanceMagicTrigger\* GameInstance = Cast<UGameInstanceMagicTrigger>(UGameplayStatics::GetGameInstance(GetWorld()));

if (GameInstance)

{

GameInstance->SaveGameManager->NewClass = this;

}

else

{

DEBUGMESSAGE("!GameInstance");

}

}

1. Добавить вызов нового метода в

void ANewClass::BeginPlay() { }

1. В файле

MagicTrigger\Source\MagicTrigger\SaveGame\SaveGameManager.cpp

заполнить определения методов

void USaveGameManager::LoadNewClass(USaveGameMT\* InLoadingGame)

{

}

void USaveGameManager::SaveNewClass(USaveGameMT\* InSavingGame)

{

}

по шаблону похожих существующих методов.

1. Проверить сохранение в игре.