一．Description

该系统适用于游戏的流程控制，关卡切换，存档，读档，结构体序列化。
The system is applicable for game flow control, level switching, saving and loading, and struct serialization.

二．Setup

把'LQuickFrame' 添加到工程插件文件夹下，并开启插件。
Add 'LQuickFrame' to the project plugin folder and enable the plugin.

在'Content'文件夹下右键，选择'QuickFrame/LSavePreset'创建存档预设。
Right-click in the 'Content' folder, select 'QuickFrame/LSavePreset' to create a save preset.

配置'LSavePreset'并添加到 'ProjectSettings/Game/LQuickFrameSettings'。
Configure 'LSavePreset' and add it to 'ProjectSettings/Game/LQuickFrameSettings'.

设置'Game Instance Class'为 LGameInst 在 'ProjectSettings/Maps & Modes'。
Set 'Game Instance Class' to LGameInst in 'ProjectSettings/Maps & Modes'.

三．Functional interface description

```
/*
* reopen new map
* @param LevelName      the map name
* @param Opens          load some streaming levels
* @param Options        params transfer
*/
UFUNCTION(BlueprintCallable, Category = "LGameInst", meta=(AutoCreateRefTerm = "Opens",AdvancedDisplay="Opens,Options"))
void NewWorld(FName LevelName, const TArray<FName>& Opens, FString Options = TEXT(""));
/*
* load saved slots for world, the function will auto load the saved steaming levels
* @param SlotName       world Slot Name
*/
UFUNCTION(BlueprintCallable, Category = "LGameInst")
void LoadWorld(const FString& SlotName);
/*
* Open and Close streaming Level
* @param OpenLevel      want to load streaming Level
* @param CloseLevel     want to unload streaming Level
*/
UFUNCTION(BlueprintCallable, Category = "LGameInst", meta=(Latent, LatentInfo = "LatentInfo", ExpandEnumAsExecs = "State", UnsafeDuringActorConstruct
void OpenCloseLevel(FName OpenLevel, FName CloseLevel, ELatentRespOne& State, FLatentActionInfo LatentInfo);
```

```cpp
/*
* get slot info
* @param SlotName       the saved slot name
*/
UFUNCTION(Category = "Save", BlueprintPure)
ULOneSlotInfo* GetSlotInfo(const FString& SlotName);
/*
* save the world
* @param SlotName       the saved slot name
*/
UFUNCTION(Category = "Save", BlueprintCallable)
void SaveWorld(const FString& SlotName);
/*
* load saved slots for world, this function needs to be called after manually loading the sub-levels
* @param SlotName       world Slot Name
*/
UFUNCTION(Category = "Save", BlueprintCallable)
void LoadWorld(const FString& SlotName);
/*
* save single object
* @param SlotName       world Slot Name
* @param Object         object to be saved
* @param Info           customizable saved object information
*/
UFUNCTION(Category = "Save", BlueprintCallable)
void SaveObject(const FString& SlotName, UObject* Object, ULOneSlotInfo* Info = NULL);
/*
* load single object
* @param SlotName       world Slot Name
* @param Object         object to be loaded
* @return               customizable saved object information
*/
UFUNCTION(Category = "Save", BlueprintCallable)
ULOneSlotInfo* LoadObject(const FString& SlotName, UObject* Object);
```

四. Others

Documentation Link:
https://github.com/liwei-cd/LQuickFrameDemo/document.pdf

Example Project: https://github.com/liwei-cd/LQuickFrameDemo

email:289423619@qq.com