

```

/*
#####
#####
# Rocket League (20224.66435.368596/5/2024) SDK
# Generated with the UE3SDKGenerator v2.2.7
#
=====
===== #
# File: ProjectX_classes.hpp
#
=====
===== #
# Credits: TheFeckless, ItsBrank
# Links: www.github.com/itsbrank/UE3SDKGenerator, www.twitter.com/itsbrank
#####
#####
*/
#pragma once

#ifdef _MSC_VER
#pragma pack(push, 0x8)
#endif

/*
#
=====
===== #
# Constants
#
=====
===== #
*/

#define CONST_ReservationType_Reconnect          'Reconnect'
#define CONST_ReservationType_FriendJoin        'FriendJoin'
#define CONST_ReservationType_JoinPrivateMatch  'JoinPrivate'
#define CONST_MaxPrivateMatchPasswordLength    16
#define CONST_MaxPrivateMatchNameLength        16
#define CONST_LocalPlayersBroadcastDelay        0.1f
#define CONST_QueueMax                          100
#define CONST_PrimaryControllerId              0
#define CONST_InMatchmakingID                  -2
#define CONST_HttpContentType                  "plain/text"
#define CONST_NumFriendsPerRequest              100
#define CONST_XpTitleId                        'XP'
#define CONST_MaxPing                          1.0f
#define CONST_ClosedReason_DuplicateLogin      "DuplicateLogin"

/*
#
=====
===== #
# Enums
#

```

```

=====
===== #
*/

// Enum ProjectX._Types_X.ESeverityType
enum class ESeverityType : uint8_t
{
    SeverityType_None           = 0,
    SeverityType_Low            = 1,
    SeverityType_Medium         = 2,
    SeverityType_High           = 3,
    SeverityType_END            = 4
};

// Enum ProjectX._Types_X.EBanType
enum class EBanType : uint8_t
{
    BanType_None                = 0,
    BanType_Auth                = 1,
    BanType_Chat                = 2,
    BanType_Voice               = 3,
    BanType_Club                = 4,
    BanType_Trade               = 5,
    BanType_QuitPenalty         = 6,
    BanType_Social              = 7,
    BanType_END                 = 8
};

// Enum ProjectX._Types_X.EConsoleQualityMode
enum class EConsoleQualityMode : uint8_t
{
    ConsoleQualityMode_Performance = 0,
    ConsoleQualityMode_Quality     = 1,
    ConsoleQualityMode_END         = 2
};

// Enum ProjectX._Types_X.EReservationStatus
enum class EReservationStatus : uint8_t
{
    ReservationStatus_None        = 0,
    ReservationStatus_Reserved    = 1,
    ReservationStatus_Joining     = 2,
    ReservationStatus_InGame      = 3,
    ReservationStatus_END         = 4
};

// Enum ProjectX._Types_X.EJoinMatchType
enum class EJoinMatchType : uint8_t
{
    JMT_Public                    = 0,
    JMT_Private                   = 1,
    JMT_END                      = 2
};

```

```
// Enum ProjectX._Types_X.EDatabaseEnvironment
enum class EDatabaseEnvironment : uint8_t
{
    DBE_DevLocal                = 0,
    DBE_Dev                     = 1,
    DBE_Test                    = 2,
    DBE_Production              = 3,
    DBE_END                     = 4
};
```

```
// Enum ProjectX._Types_X.EAxisSign
enum class EAxisSign : uint8_t
{
    AxisSign_None               = 0,
    AxisSign_Positive           = 1,
    AxisSign_Negative           = 2,
    AxisSign_END                = 3
};
```

```
// Enum ProjectX._Types_X.EButtonPressType
enum class EButtonPressType : uint8_t
{
    BPT_Normal                  = 0,
    BPT_Tap                     = 1,
    BPT_Hold                    = 2,
    BPT_DoubleTap               = 3,
    BPT_Toggle                  = 4,
    BPT_END                     = 5
};
```

```
// Enum ProjectX._Types_X.ERemappable
enum class ERemappable : uint8_t
{
    Remappable_All              = 0,
    Remappable_Key              = 1,
    Remappable_Axis             = 2,
    Remappable_KeyTriggerAxis   = 3,
    Remappable_None             = 4,
    Remappable_END              = 5
};
```

```
// Enum ProjectX._Types_X.ETradePermissionLevel
enum class ETradePermissionLevel : uint8_t
{
    TPL_Everybody               = 0,
    TPL_FriendsAndParty         = 1,
    TPL_Friends                  = 2,
    TPL_Nobody                   = 3,
    TPL_END                     = 4
};
```

```
// Enum ProjectX._Types_X.EChatPermissionLevel
enum class EChatPermissionLevel : uint8_t
{

```

```

CPL_Everybody                = 0,
CPL_FriendsAndTeam           = 1,
CPL_Friends                   = 2,
CPL_Nobody                   = 3,
CPL_END                       = 4
};

// Enum ProjectX._Types_X.EPartyMatchmakingRestriction
enum class EPartyMatchmakingRestriction : uint8_t
{
    PMR_NotOriginalAppOwner    = 0,
    PMR_PendingLicenseAgreement = 1,
    PMR_InMainMenu              = 2,
    PMR_NotLoggedInPsyNet      = 3,
    PMR_END                    = 4
};

// Enum ProjectX._Types_X.EOnlinePlayerPermission
enum class EOnlinePlayerPermission : uint8_t
{
    OPP_PrivateMatchAdmin      = 0,
    OPP_SuperPrivateMatchAdmin = 1,
    OPP_END                    = 2
};

// Enum ProjectX._Types_X.EOnlinePlayerRole
enum class EOnlinePlayerRole : uint8_t
{
    OPR_PrivateMatchAdmin      = 0,
    OPR_SuperPrivateMatchAdmin = 1,
    OPR_END                    = 2
};

// Enum ProjectX._Types_X.EConnectionType_X
enum class EConnectionType_X : uint8_t
{
    ConnectionType_Unknown     = 0,
    ConnectionType_Wired       = 1,
    ConnectionType_WiFi        = 2,
    ConnectionType_Mixed       = 3,
    ConnectionType_Docked      = 4,
    ConnectionType_END         = 5
};

// Enum ProjectX.EpicLogin_X.AgeGateState
enum class EAgeGateState : uint8_t
{
    AGS_Unknown                = 0,
    AGS_Active                  = 1,
    AGS_Complete                = 2,
    AGS_END                    = 3
};

// Enum ProjectX.EpicLogin_X.ELoginMethod

```

```

enum class ELoginMethod : uint8_t
{
    LoginMethod_RefreshToken          = 0,
    LoginMethod_PlatformAuthTicket    = 1,
    LoginMethod_END                    = 2
};

// Enum ProjectX.OnlineGameParty_X.PartyProcessingStatus
enum class EPartyProcessingStatus : uint8_t
{
    PPS_None                          = 0,
    PPS_Create                        = 1,
    PPS_Join                          = 2,
    PPS_END                           = 3
};

// Enum ProjectX.EncodeObject_X.EObjectEncoding
enum class EObjectEncoding : uint8_t
{
    OPSE_UObject                      = 0,
    OPSE_Json                         = 1,
    OPSE_END                           = 2
};

// Enum ProjectX.RPC_X.ERPCPriority
enum class ERPCPriority : uint8_t
{
    RPC_Low                          = 0,
    RPC_Medium                       = 1,
    RPC_High                         = 2,
    RPC_END                           = 3
};

// Enum ProjectX.AnimNodeSeries_X.EAnimNodeSeriesChild
enum class EAnimNodeSeriesChild : uint8_t
{
    AnimNodeSeries_Idle               = 0,
    AnimNodeSeries_Intro               = 1,
    AnimNodeSeries_Loop                = 2,
    AnimNodeSeries_Outro               = 3,
    AnimNodeSeries_END                 = 4
};

// Enum ProjectX.BlogTile_X.ELinkType
enum class ELinkType : uint8_t
{
    LinkType_None                     = 0,
    LinkType_RocketPass                = 1,
    LinkType_Event                     = 2,
    LinkType_ESports                   = 3,
    LinkType_END                       = 4
};

// Enum ProjectX.ClanforgeReservation_X.EReserveState

```

```
enum class EReserveState : uint8_t
```

```
{  
ReserveState_Initialized          = 0,  
ReserveState_Unreserved          = 1,  
ReserveState_Reserved            = 2,  
ReserveState_END                  = 3  
};
```

```
// Enum ProjectX.ClubUtil_X.EClubColorChange
```

```
enum class EClubColorChange : uint8_t
```

```
{  
ClubColorChange_None              = 0,  
ClubColorChange_SwapTeams         = 1,  
ClubColorChange_SwapPrimaryAccent0 = 2,  
ClubColorChange_SwapPrimaryAccent1 = 3,  
ClubColorChange_SwapPrimaryAccent = 4,  
ClubColorChange_WhiteVsBlack      = 5,  
ClubColorChange_Gray1             = 6,  
ClubColorChange_END               = 7  
};
```

```
// Enum ProjectX.DynamicValue_X.EDynamicValueModType
```

```
enum class EDynamicValueModType : uint8_t
```

```
{  
ModType_Add                       = 0,  
ModType_Multiply                  = 1,  
ModType_MultiplyAdd               = 2,  
ModType_Set                       = 3,  
ModType_END                       = 4  
};
```

```
// Enum ProjectX.FakeData_X.FakeDataEnum
```

```
enum class EFakeDataEnum : uint8_t
```

```
{  
FakeData                         = 0,  
FakeData01                       = 1,  
FakeData02                       = 2,  
FakeData_Count                   = 3,  
FakeData_END                     = 4  
};
```

```
// Enum ProjectX.FXActor_X.EFXComponentState
```

```
enum class EFXComponentState : uint8_t
```

```
{  
FXComponentState_Detached        = 0,  
FXComponentState_Attached        = 1,  
FXComponentState_PendingDetach   = 2,  
FXComponentState_END             = 3  
};
```

```
// Enum ProjectX.FXActor_X.EFXComponentTarget
```

```
enum class EFXComponentTarget : uint8_t
```

```
{  
FXComponentTarget_All            = 0,  
};
```

```

FXComponentTarget_Local          = 1,
FXComponentTarget_NonLocal       = 2,
FXComponentTarget_END            = 3
};

// Enum ProjectX.FXActor_X.EAttachActorLocationUnlockFlags
enum class EAttachActorLocationUnlockFlags : uint8_t
{
    EAALUF_None                    = 0,
    EAALUF_X                      = 1,
    EAALUF_Y                      = 2,
    EAALUF_Z                      = 3,
    EAALUF_END                    = 4
};

// Enum ProjectX.FXActor_X.EFXLocality
enum class EFXLocality : uint8_t
{
    FXLocality_Controller          = 0,
    FXLocality_AlwaysLocal        = 1,
    FXLocality_NeverLocal         = 2,
    FXLocality_END                = 3
};

// Enum ProjectX.FXActor_X.EFXOverrideAttachBehavior
enum class EFXOverrideAttachBehavior : uint8_t
{
    FXOverrideAttach_None         = 0,
    FXOverrideAttach_Owner        = 1,
    FXOverrideAttach_END          = 2
};

// Enum ProjectX.GFxDataCallback_X.EDataCallbackType
enum class EDataCallbackType : uint8_t
{
    DataCallbackType_All          = 0,
    DataCallbackType_Table        = 1,
    DataCallbackType_Row          = 2,
    DataCallbackType_Value        = 3,
    DataCallbackType_END          = 4
};

// Enum ProjectX.MaterialEffect_X.EMaterialEffectStage
enum class EMaterialEffectStage : uint8_t
{
    EFFECT_INACTIVE               = 0,
    EFFECT_FADE_IN                = 1,
    EFFECT_ACTIVE                 = 2,
    EFFECT_FADE_OUT               = 3,
    EFFECT_END                    = 4
};

// Enum ProjectX.OnlineGameDLC_X.DLCOwnershipState
enum class EDLCOwnershipState : uint8_t

```

```

{
DLCOwnershipState_Unknown          = 0,
DLCOwnershipState_Owned             = 1,
DLCOwnershipState_NotOwned          = 2,
DLCOwnershipState_Error              = 3,
DLCOwnershipState_END               = 4
};

```

// Enum ProjectX.WordFilterTypes\_X.EWordFilterUsage

```

enum class EWordFilterUsage : uint8_t
{
WordFilterUsage_PlayerName          = 0,
WordFilterUsage_TrainingName        = 1,
WordFilterUsage_LanServerName       = 2,
WordFilterUsage_CustomTeamName      = 3,
WordFilterUsage_TourName             = 4,
WordFilterUsage_TourTeamName        = 5,
WordFilterUsage_ClubName             = 6,
WordFilterUsage_ClubTag              = 7,
WordFilterUsage_ClubMotD             = 8,
WordFilterUsage_ClubTagPlayerName   = 9,
WordFilterUsage_MatchChat            = 10,
WordFilterUsage_PartyChat            = 11,
WordFilterUsage_END                 = 12
};

```

// Enum ProjectX.OnlinePlayerRegionRestrictions\_X.ERegionRestriction

```

enum class ERegionRestriction : uint8_t
{
RegionRestriction_Unknown           = 0,
RegionRestriction_KeyCrate           = 1,
RegionRestriction_Trade              = 2,
RegionRestriction_KeyRocketPassTier = 3,
RegionRestriction_END               = 4
};

```

// Enum ProjectX.PerCon\_X.EPerConStatus

```

enum class EPerConStatus : uint8_t
{
PerConStatus_Disabled               = 0,
PerConStatus_Enabled                = 1,
PerConStatus_END                    = 2
};

```

// Enum ProjectX.RegionPing\_X.RegionPingResult

```

enum class ERegionPingResult : uint8_t
{
PingResult_NotUsable                = 0,
PingResult_UsingCache               = 1,
PingResult_UsingSuccessfulPing      = 2,
PingResult_END                      = 3
};

```

// Enum ProjectX.RemoteAvatarPermissions\_X.EAvatarRequestTryGetResult



```
enum class EAvatarRequestTryGetResult : uint8_t
{
    ARTGR_NotFound          = 0,
    ARTGR_False             = 1,
    ARTGR_True              = 2,
    ARTGR_Hidden            = 3,
    ARTGR_END               = 4
};
```

// Enum ProjectX.RemoteAvatarPermissions\_X.EAvatarPermission

```
enum class EAvatarPermission : uint8_t
{
    AP_True                 = 0,
    AP_False                = 1,
    AP_Hidden               = 2,
    AP_END                  = 3
};
```

// Enum ProjectX.ReservationsPasswordMessage\_X.EPasswordFailureReason

```
enum class EPasswordFailureReason : uint8_t
{
    PFR_Empty              = 0,
    PFR_Mismatch           = 1,
    PFR_END                 = 2
};
```

// Enum ProjectX.RPC\_ReportExploiter\_X.EExploitType

```
enum class EExploitType : uint8_t
{
    ET_FakeProducts        = 0,
    ET_InvisibleCar         = 1,
    ET_END                  = 2
};
```

// Enum ProjectX.ShakeComponent\_X.EShakeReceiver

```
enum class EShakeReceiver : uint8_t
{
    ShakeReceiver_All      = 0,
    ShakeReceiver_Local    = 1,
    ShakeReceiver_NonLocal = 2,
    ShakeReceiver_END      = 3
};
```

// Enum ProjectX.TcpConnection.EWebSocketState

```
enum class EWebSocketState : uint8_t
{
    WebSocketState_NotRequired = 0,
    WebSocketState_NeedsHandshake = 1,
    WebSocketState_Connected    = 2,
    WebSocketState_END          = 3
};
```

// Enum ProjectX.TcpConnection.ETcpConnectionState

```
enum class ETcpConnectionState : uint8_t
```

```
{
TcpConnectionState_Invalid          = 0,
TcpConnectionState_Connecting        = 1,
TcpConnectionState_Connected         = 2,
TcpConnectionState_Disconnected      = 3,
TcpConnectionState_END               = 4
};
```

```
// Enum ProjectX.WebRequest_X.EWebRequestState
enum class EWebRequestState : uint8_t
{
WebRequestState_PendingSend          = 0,
WebRequestState_PendingResponse      = 1,
WebRequestState_Success               = 2,
WebRequestState_Error                 = 3,
WebRequestState_END                   = 4
};
```

```
/*
#
=====
===== #
# Classes
#
=====
===== #
*/
```

```
// Class ProjectX._SharedHelpers
// 0x0000 (0x0060 - 0x0060)
class U_SharedHelpers : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX._SharedHelpers");
}

return uClassPointer;
};
```

```
static int32_t ScoreDeltaFromDefault(class UObject* Object);
static void DrawShadowedText(class UCanvas* Canvas, class FString Text, int32_t
ShadowOffsetX, int32_t ShadowOffsetY, int32_t ShadowAlpha);
static bool IsInTransientPackage(class UObject* Obj);
static class APlayerController* GetAPlayerController(class UClass* PlayerControllerClass);
static class UObject* GetDefaultObject(class UClass* ObjClass);
```

```

static void DumpUnreferencedAnims();
static class FString GetLanguageExtension();
};

// Class ProjectX._Types_X
// 0x0048 (0x0060 - 0x00A8)
class U_Types_X : public UObject
{
public:
    struct FScriptDelegate      __OnPriviledgeChecked__Delegate;          // 0x0060
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
    struct FScriptDelegate      __ServiceExecutionDelegate__Delegate;     // 0x0078
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
    struct FScriptDelegate      __OnTimer__Delegate;                      // 0x0090 (0x0018)
    [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX._Types_X");
        }

        return uClassPointer;
    };

    static void RenderProfileRecordSample(struct FRenderProfile& Profile);
    static void JoinCredentialsFromString(class FString Credentials, class FString& JoinName, class
    FString& JoinPassword);
    static class FString JoinCredentialsToString(class FString JoinName, class FString
    JoinPassword);
    static class FString GenerateRandomPrivateMatchPassword();
    static class FString GenerateRandomPrivateMatchName();
    static class FString GenerateRandomName(int32_t Length);
    void OnTimer();
    static struct FServerReservationData GetServerReservationData(struct
    FCheckReservationResponse& Response);
    void ServiceExecutionDelegate(class UPsyNetClientService_X* Service);
    void OnPriviledgeChecked(class UPrivilegeCheck_X* PrivilegeCheck);
};

// Class ProjectX.ActorComponent_X
// 0x0007 (0x009D - 0x00A4)
class UActorComponent_X : public UActorComponent
{
public:
    unsigned long                bTick : 1;                                // 0x00A0 (0x0004)
    [0x0000000000000000] [0x00000001]
    unsigned long                bIgnoreScriptAttach : 1;                  // 0x00A0 (0x0004)
    [0x0000000000000200] [0x00000002] (CPF_Const | CPF_Transient)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ActorComponent_X");
}

return uClassPointer;
};

void Reattach();
void eventTick(float DeltaTime);
void eventDetached();
void eventAttached();
void eventBeginPlay();
};

// Class ProjectX.ExplosionComponent_X
// 0x003C (0x00A4 - 0x00E0)
class UExplosionComponent_X : public UActorComponent_X
{
public:
float RBVelocityChange; // 0x00A8 (0x0004)
[0x0000000000000001] (CPF_Edit)
float StartRadius; // 0x00AC (0x0004)
[0x0000000000000001] (CPF_Edit)
float EndRadius; // 0x00B0 (0x0004)
[0x0000000000000001] (CPF_Edit)
float Speed; // 0x00B4 (0x0004)
[0x0000000000000001] (CPF_Edit)
float Falloff; // 0x00B8 (0x0004)
[0x0000000000000001] (CPF_Edit)
unsigned long bPassThroughWorldGeometry : 1; // 0x00BC
(0x0004) [0x0000000000000001] [0x00000001] (CPF_Edit)
unsigned long bIgnoreInstigator : 1; // 0x00BC (0x0004)
[0x0000000000000001] [0x00000002] (CPF_Edit)
unsigned long bDebug : 1; // 0x00BC (0x0004)
[0x0000000000000001] [0x00000004] (CPF_Edit)
struct FVector MomentumOffset; // 0x00C0 (0x000C)
[0x0000000000000001] (CPF_Edit)
float LifeTime; // 0x00CC (0x0004)
[0x0000000000000200] (CPF_Transient)
TArray<class AActor*> DamagedActors; // 0x00D0 (0x0010)
[0x0000000000040200] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)

```

```

{
uClassPointer = UObject::FindClass("Class ProjectX.ExplosionComponent_X");
}

return uClassPointer;
};

void eventProcessHit(class AActor* Victim, struct FVector HitLocation, struct FTraceHitInfo
ExtraHitInfo, float DamageScale);
void eventDetached();
};

// Class ProjectX.ExplosionHitHandler_X
// 0x0050 (0x00A4 - 0x00F4)
class UExplosionHitHandler_X : public UActorComponent_X
{
public:
float                               StartTime;                               // 0x00A8 (0x0004)
[0x000000000000000001] (CPF_Edit)
float                               EndTime;                               // 0x00AC (0x0004)
[0x000000000000000001] (CPF_Edit)
class UExplosionCollisionShapes_X* CollisionInformation;                     // 0x00B0
(0x0008) [0x0000000004080009] (CPF_Edit | CPF_ExportObject | CPF_Component |
CPF_EditInline)
unsigned long                       bIgnoreInstigator : 1;                   // 0x00B8 (0x0004)
[0x000000000000000001] [0x00000001] (CPF_Edit)
unsigned long                       bPassThroughWorldGeometry : 1;           // 0x00B8
(0x0004) [0x0000000000000001] [0x00000002] (CPF_Edit)
unsigned long                       bOnlyApplyImpulseOnce : 1;               // 0x00B8 (0x0004)
[0x000000000000000001] [0x00000004] (CPF_Edit)
unsigned long                       bDebug : 1;                             // 0x00B8 (0x0004)
[0x000000000000000001] [0x00000008] (CPF_Edit)
unsigned long                       bPullTowardHitLocation : 1;              // 0x00B8 (0x0004)
[0x000000000000000001] [0x00000010] (CPF_Edit)
float                               Falloff;                               // 0x00BC (0x0004)
[0x000000000000000001] (CPF_Edit)
float                               RBVelocityChange;                       // 0x00C0 (0x0004)
[0x000000000000000001] (CPF_Edit)
float                               Speed;                                   // 0x00C4 (0x0004)
[0x000000000000000001] (CPF_Edit)
struct FVector                      MomentumOffset;                         // 0x00C8 (0x000C)
[0x000000000000000001] (CPF_Edit)
struct FBox                         GoalWorldBox;                           // 0x00D4 (0x001C)
[0x000000000000002000] (CPF_Transient)
float                               LifeTime;                               // 0x00F0 (0x0004)
[0x000000000000002000] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{

```

```

uClassPointer = UObject::FindClass("Class ProjectX.ExplosionHitHandler_X");
}

return uClassPointer;
};

void Destroy();
void Init(class UActorComponent_X* ExplosionGoal);
void eventProcessHit(class AActor* Victim, struct FVector HitLocation, float DamageScale, struct
FContactInformation& ContactInfo);
};

// Class ProjectX.MICParamDispenserComponent_X
// 0x0024 (0x00A4 - 0x00C8)
class UMICParamDispenserComponent_X : public UActorComponent_X
{
public:
struct FPointer                VfTable_IISetParameter;                // 0x00A8 (0x0008)
[0x000000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
int32_t                        MaterialIndex;                        // 0x00B0 (0x0004)
[0x00000000000000001] (CPF_Edit)
struct FName                    AttachedComponentName;                // 0x00B4 (0x0008)
[0x00000000000000001] (CPF_Edit)
class UMeshComponent*          MeshComp;                            // 0x00C0 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.MICParamDispenserComponent_X");
}

return uClassPointer;
};

bool __MICParamDispenserComponent_X__Attached_0x1(struct FFXAttachment A);
void eventAttached();
void SetActorParameter(struct FName Key, class AActor* Value);
void SetLinearColorParameter(struct FName Key, struct FLinearColor Value);
void SetVectorParameter(struct FName Key, struct FVector Value);
void SetFloatParameter(struct FName Key, float Value);
void SetNameParameter(struct FName Key, struct FName Value);
};

// Class ProjectX.ShakeComponent_X
// 0x0114 (0x00A4 - 0x01B8)
class UShakeComponent_X : public UActorComponent_X
{
public:
float                            InnerRadius;                        // 0x00A8 (0x0004)

```

```

[0x0000000000000001] (CPF_Edit)
float                                OuterRadius;                                // 0x00AC (0x0004)
[0x0000000000000001] (CPF_Edit)
float                                Falloff;                                    // 0x00B0 (0x0004)
[0x0000000000000001] (CPF_Edit)
unsigned long                        bAutoPlay : 1;                            // 0x00B4 (0x0004)
[0x0000000000000001] [0x00000001] (CPF_Edit)
unsigned long                        bAutoDetach : 1;                            // 0x00B4 (0x0004)
[0x0000000000000001] [0x00000002] (CPF_Edit)
unsigned long                        bOnlyAffectPlayersWithPawns : 1;          // 0x00B4
(0x0004) [0x0000000000000001] [0x00000004] (CPF_Edit)
unsigned long                        bAttenuate : 1;                            // 0x00B4 (0x0004)
[0x0000000000000001] [0x00000008] (CPF_Edit)
unsigned long                        bPlaying : 1;                            // 0x00B4 (0x0004)
[0x0000000400000200] [0x00000010] (CPF_Transient)
unsigned long                        bStopping : 1;                            // 0x00B4 (0x0004)
[0x0000000400000200] [0x00000020] (CPF_Transient)
uint8_t                             Receiver;                                // 0x00B8 (0x0001)
[0x0000000000000001] (CPF_Edit)
float                                RadiusScale;                                // 0x00BC (0x0004)
[0x0000000000000001] (CPF_Edit)
float                                BlendInTime;                                // 0x00C0 (0x0004)
[0x0000000000000001] (CPF_Edit)
float                                BlendOutTime;                                // 0x00C4 (0x0004)
[0x0000000000000001] (CPF_Edit)
class UCameraShake*                  ShakeParams;                            // 0x00C8 (0x0008)
[0x0000000000000001] (CPF_Edit)
struct FRawDistributionFloat          AmplitudeScale;                            // 0x00D0 (0x0028)
[0x0000000000048001] (CPF_Edit | CPF_Component | CPF_NeedCtorLink)
struct FRawDistributionFloat          FrequencyScale;                            // 0x00F8 (0x0028)
[0x0000000000048001] (CPF_Edit | CPF_Component | CPF_NeedCtorLink)
struct FRawDistributionFloat          DurationScale;                            // 0x0120 (0x0028)
[0x0000000000048001] (CPF_Edit | CPF_Component | CPF_NeedCtorLink)
float                                CurrentAmplitudeScale;                    // 0x0148 (0x0004)
[0x00000000000002002] (CPF_Const | CPF_Transient)
float                                CurrentFrequencyScale;                    // 0x014C (0x0004)
[0x00000000000002002] (CPF_Const | CPF_Transient)
class UForceFeedbackWaveform*         ForceFeedbackWaveform;                  // 0x0150
(0x0008) [0x0000000000000001] (CPF_Edit)
float                                ForceFeedbackScale;                        // 0x0158 (0x0004)
[0x0000000000000001] (CPF_Edit)
float                                ForceFeedbackBalance;                      // 0x015C (0x0004)
[0x0000000000000001] (CPF_Edit)
TArray<struct FParticleSysParam>       InstanceParameters;                    // 0x0160
(0x0010) [0x0000000004400001] (CPF_Edit | CPF_NeedCtorLink | CPF_EditInline)
float                                PlayTime;                                // 0x0170 (0x0004)
[0x0000000400000200] (CPF_Transient)
float                                StopTime;                                // 0x0174 (0x0004)
[0x0000000400000200] (CPF_Transient)
struct FVector                        ShakeLocTimeOffset;                      // 0x0178 (0x000C)
[0x0000000400000200] (CPF_Transient)
struct FVector                        ShakeRotTimeOffset;                      // 0x0184 (0x000C)
[0x0000000400000200] (CPF_Transient)
float                                ShakeFOVTimeOffset;                        // 0x0190 (0x0004)

```

```

[0x0000004000002000] (CPF_Transient)
float CameraShakeDuration; // 0x0194 (0x0004)
[0x0000004000002000] (CPF_Transient)
float ForceFeedbackDuration; // 0x0198 (0x0004)
[0x0000004000002000] (CPF_Transient)
float ForceFeedbackSamplesTime; // 0x019C (0x0004)
[0x0000004000002000] (CPF_Transient)
class AActor* PrevOwner; // 0x01A0 (0x0008)
[0x0000004000002000] (CPF_Transient)
TArray<struct FShakeReceiver> Receivers; // 0x01A8 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ShakeComponent_X");
}

return uClassPointer;
};

```

```

bool GetColorParameter(struct FName InName, struct FColor& OutColor);
bool GetVectorParameter(struct FName InName, struct FVector& OutVector);
bool GetFloatParameter(struct FName InName, float& OutFloat);
void SetColorParameter(struct FName ParameterName, struct FColor Param);
void SetVectorRandParameter(struct FName ParameterName, struct FVector& Param, struct
FVector& ParamLow);
void SetVectorParameter(struct FName ParameterName, struct FVector Param);
void SetFloatRandParameter(struct FName ParameterName, float Param, float ParamLow);
void SetFloatParameter(struct FName ParameterName, float Param);
bool IsPlaying();
void eventStopShake(float InBlendOutTime);
void InitParams();
void eventPlayShake();
void eventDetached();
void eventAttached();
};

```

```

// Class ProjectX.AdHocBeacon_X
// 0x0010 (0x0060 - 0x0070)
class UAdHocBeacon_X : public UObject
{
public:
struct FPointer VfTable_FTickableObject; // 0x0060 (0x0008)
[0x00000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
struct FPointer pAdHoc; // 0x0068 (0x0008)
[0x00000000000003002] (CPF_Const | CPF_Native | CPF_Transient)

```

```

public:
static UClass* StaticClass()

```



```
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.AdHocBeacon_X");
}

return uClassPointer;
};

int32_t GetNodeCountMax();
void AdHocJoinMatch();
void AdHocCreateMatch();
bool AdHocIsActive();
void AdHocDisconnectFromAccessPoint();
void AdHocConnectToAccessPoint(int32_t ScanResultIndex);
void AdHocScanForAccessPoints();
void AdHocBecomeStation();
void AdHocRejectStation(int32_t NetworkNodeIndex);
void AdHocBecomeAccessPoint();
void AdHocSetAdvertiseData(class FString AdvertiseData);
void AdHocDestroy();
void AdHocCreate();
void AdHocCreateWithUsername(class FString Username);
void eventConstruct();
};

// Class ProjectX.FXActor_X
// 0x00C0 (0x0268 - 0x0328)
class AFXActor_X : public AActor
{
public:
class UFXActorEvent_X*                SpawnState;                // 0x0268 (0x0008)
[0x0000000000000000]
class UFXActorEvent_X*                ActivationState;          // 0x0270 (0x0008)
[0x0000000000000000]
TArray<struct FFXAttachment>           Attachments;              // 0x0278 (0x0010)
[0x0002000000480001] (CPF_Edit | CPF_Component | CPF_NeedCtorLink)
uint8_t                               Locality;                  // 0x0288 (0x0001)
[0x0000000000000001] (CPF_Edit)
struct FName                          SocketOrBoneName;          // 0x028C (0x0008)
[0x0000000000000001] (CPF_Edit)
unsigned long                          bDeactivateWhenOwnerDestroyed : 1;    // 0x0294
(0x0004) [0x0000000000000001] [0x00000001] (CPF_Edit)
unsigned long                          bAllowShadowCasting : 1;    // 0x0294 (0x0004)
[0x0000000000000001] [0x00000002] (CPF_Edit)
unsigned long                          bAutoActivate : 1;          // 0x0294 (0x0004)
[0x0000000000000001] [0x00000004] (CPF_Edit)
unsigned long                          bRenderInactive : 1;        // 0x0294 (0x0004)
[0x0000000000000001] [0x00000008] (CPF_Edit)
unsigned long                          bActive : 1;                // 0x0294 (0x0004)
[0x0000004000002000] [0x00000010] (CPF_Transient)
unsigned long                          bHadOwner : 1;              // 0x0294 (0x0004)
}
```

```

[0x0000004000002002] [0x00000020] (CPF_Const | CPF_Transient)
class AFXActor_X*                               Parent;                               // 0x0298 (0x0008)
[0x00000000000000001] (CPF_Edit)
class AActor*                                   AttachmentActor;                       // 0x02A0 (0x0008)
[0x0000004000002000] (CPF_Transient)
TArray<class UFXActorEvent_X*>                  FXStates;                               // 0x02A8 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<struct FFXActorEventCheck>              FXEvents;                               // 0x02B8 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
float                                           DestroyWaitTime;                       // 0x02C8 (0x0004)
[0x00000000000000001] (CPF_Edit)
float                                           DestroyTime;                           // 0x02CC (0x0004)
[0x00000040000000000]
class UParameterDispenser_X*                  Parameters;                             // 0x02D0 (0x0008)
[0x00000000004080009] (CPF_Edit | CPF_ExportObject | CPF_Component | CPF_EditInline)
int32_t                                         EditID;                                // 0x02D8 (0x0004)
[0x00000000800020003] (CPF_Edit | CPF_Const | CPF_EditConst)
TArray<struct FFXEventSubscription>            EventSubscriptions;                     // 0x02E0
(0x0010) [0x0000004000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
class UFXActorEvent_X*                        AnimationEndState;                       // 0x02F0 (0x0008)
[0x00000000000000000]
struct FScriptDelegate                        __EventFXStatePushed__Delegate;        // 0x02F8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                        __EventFXStatePopped__Delegate;        // 0x0310
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.FXActor_X");
}

return uClassPointer;
};

struct FName __FXActor_X__PrintDebugInfo_0x1(class UFXActorEvent_X* State);
void PrintDebugInfo(class UDebugDrawer* Drawer);
void eventDumpDebugInfo();
class UFXAttachmentTraitBase_X* GetTrait(class UClass* TraitClass, int32_t AttachmentIdx);
void eventOnAnimEnd(class UAnimNodeSequence* SeqNode, float PlayedTime, float ExcessTime);
void eventOnAnimPlay(class UAnimNodeSequence* SeqNode);
void eventDestroyed();
void SubscribeState(class UFXActorEvent_X* FXEvent, struct FScriptDelegate OnPushed, struct FScriptDelegate OnPopped);
void SetStateEnabled(class UFXActorEvent_X* State, unsigned long bEnable);
void OnToggle(class USeqAct_Toggle* Action);
void OnTriggerFXActor_X(class USeqAct_TriggerFXActor_X* SeqAct);
void Inherit(class AFXActor_X* Other);
void ResetParticles();

```

```

void StopAllEffects();
void eventDeactivateAndDestroy();
void eventOnDetached(class UActorComponent* AC);
void eventDeactivateFXComponent(class UActorComponent* AC);
void eventActivateFXComponent(class UActorComponent* AC);
void eventOnAttached(class UActorComponent* AC);
int32_t GetComponentAttachmentIndex(class UActorComponent* ComponentToFind);
void UpdateFXStates();
bool IsLocallyControlled();
class AController* GetControllerOwner();
void ToggleState(class UFXActorEvent_X* NewState);
bool RemoveStateForceUpdate(class UFXActorEvent_X* OldState);
bool RemoveEvent(class UFXActorEvent_X* OldEvent);
bool RemoveState(class UFXActorEvent_X* OldState);
void AddEvent(class UFXActorEvent_X* NewEvent);
void AddState(class UFXActorEvent_X* NewState);
bool CanRemoveEvent(struct FFXActorEventCheck& InEvent);
bool IsStateActive(class UFXActorEvent_X* InState);
void Deactivate();
void Activate();
void HandleParentStatePopped(class AFXActor_X* P, class UFXActorEvent_X* Event);
void HandleParentStatePushed(class AFXActor_X* P, class UFXActorEvent_X* Event);
void BindTo(class AFXActor_X* ParentFXActor);
void SetAttachmentActor(class AActor* AttachToActor);
void PostBeginPlay();
void AllAttachments(class UClass* ComponentClass, class UActorComponent*& OutComponent,
int32_t& OutAttachmentIdx);
class UActorComponent* GetComponentByName(class UClass* ComponentClass, struct FName
ComponentName);
void EventFXStatePopped(class AFXActor_X* FXActor, class UFXActorEvent_X* Event);
void EventFXStatePushed(class AFXActor_X* FXActor, class UFXActorEvent_X* Event);
};

```

```

// Class ProjectX.AnimateParametersComponent_X
// 0x003F (0x009D - 0x00DC)
class UAnimateParametersComponent_X : public UActorComponent
{
public:
TArray<struct FAnimatedFloatParam>          FloatParameters;                // 0x00A0
(0x0010) [0x00000000000400001] (CPF_Edit | CPF_NeedCtorLink)
TArray<struct FAnimatedVectorParam>        VectorParameters;              // 0x00B0
(0x0010) [0x00000000000400001] (CPF_Edit | CPF_NeedCtorLink)
TArray<struct FAnimatedLinearColorParam>    ColorParameters;              // 0x00C0
(0x0010) [0x00000000000400001] (CPF_Edit | CPF_NeedCtorLink)
unsigned long                             bResetMaterialsOnComplete : 1;    // 0x00D0 (0x0004)
[0x00000000000000001] [0x000000001] (CPF_Edit)
float                                     AnimTime;                        // 0x00D4 (0x0004)
[0x000000000000002000] (CPF_Transient)
float                                     EndTime;                        // 0x00D8 (0x0004)
[0x000000000000002002] (CPF_Const | CPF_Transient)

public:
static UClass* StaticClass()
{

```

```

static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.AnimateParametersComponent_X");
}

return uClassPointer;
};

};

// Class ProjectX.ArchetypePrefab_X
// 0x0020 (0x0268 - 0x0288)
class AArchetypePrefab_X : public AActor
{
public:
    TArray<class AActor*> PrefabArchetypes; // 0x0268 (0x0010)
    [0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
    TArray<class AActor*> ArchetypeInstances; // 0x0278 (0x0010)
    [0x0000000000040202] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.ArchetypePrefab_X");
        }

        return uClassPointer;
    };

};

// Class ProjectX.ArchetypeSpawnPoint_X
// 0x0070 (0x0268 - 0x02D8)
class AArchetypeSpawnPoint_X : public AActor
{
public:
    struct FPointer VfTable_IInterface_NavMeshPathObject; // 0x0268
    (0x0008) [0x00000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
    class FString ArchetypePath; // 0x0270 (0x0010)
    [0x0000080000040003] (CPF_Edit | CPF_Const | CPF_NeedCtorLink)
    unsigned long bSeedNavMesh : 1; // 0x0280 (0x0004)
    [0x0000000000000001] [0x00000001] (CPF_Edit)
    unsigned long bSpawnOnLevelStart : 1; // 0x0280 (0x0004)
    [0x0000000000000001] [0x00000002] (CPF_Edit)
    unsigned long bOnlyAllowOneInstance : 1; // 0x0280 (0x0004)
    [0x0000000000000001] [0x00000004] (CPF_Edit)
    class AActor* ActorArchetype; // 0x0288 (0x0008)
    [0x0000000000000001] (CPF_Edit)

```

```

class USpriteComponent*          PlacementSprite;          // 0x0290 (0x0008)
[0x00000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class USkeletalMeshComponent*    ArchetypeSkeletalMesh;    // 0x0298
(0x0008) [0x00000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UStaticMeshComponent*      ArchetypeStaticMesh;      // 0x02A0
(0x0008) [0x00000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
TArray<class AActor*>             SpawnedActors;             // 0x02A8 (0x0010)
[0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
class AActor*                    SpawnedActor;              // 0x02B8 (0x0008)
[0x0000000000002000] (CPF_Transient)
struct FScriptDelegate            __EventSpawned__Delegate; // 0x02C0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ArchetypeSpawnPoint_X");
}

return uClassPointer;
};

```

```

void OnSpawned(class AActor* Spawned);
class AActor* DoSpawn();
void OnToggle(class USeqAct_Toggle* Action);
void OnDestroy(class USeqAct_Destroy* Action);
void eventPostBeginPlay();
void EventSpawned(class AArchetypeSpawnPoint_X* SpawnPoint, class AActor* Spawned);
};

```

```

// Class ProjectX.Aws4Signature_X
// 0x0000 (0x0060 - 0x0060)
class UAws4Signature_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Aws4Signature_X");
}

return uClassPointer;
};

```

```

static void SignRequest(class UHttpRequestInterface* HTTPRequest);

```

```

};

// Class ProjectX.ReservationBeacon_X
// 0x0060 (0x0070 - 0x00D0)
class UReservationBeacon_X : public UComponent
{
public:
TArray<struct FBeaconMessageHandler>      MessageHandlers;          // 0x0070
(0x0010) [0x00000000000400002] (CPF_Const | CPF_NeedCtorLink)
class UPsyNetBeacon_X*                    PsyNetBeacon;            // 0x0080 (0x0008)
[0x000000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
struct FScriptDelegate                    __MessageReceivedDelegate__Delegate; // 0x0088
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __EventConnected__Delegate;    // 0x00A0
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __EventDisconnected__Delegate;  // 0x00B8
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ReservationBeacon_X");
}

return uClassPointer;
};

void __ReservationBeacon_X__Construct_0x4();
void __ReservationBeacon_X__Construct_0x3(class UPsyNetBeaconConnection_X* Connection,
class UObject* Message);
void __ReservationBeacon_X__Construct_0x2(class UPsyNetBeacon_X* _, class
UPsyNetBeaconConnection_X* Connection);
void __ReservationBeacon_X__Construct_0x1(class UPsyNetBeacon_X* _, class
UPsyNetBeaconConnection_X* Connection);
class UIReservationConnection_X* ServerReserveConnection(class FString ReservationID, struct
FUniqueNetId PlayerID, float ConnectionTimeoutSeconds);
class UIReservationConnection_X* BeginPsyNetConnect(class FString ReservationID, class
FString DSRTOKEN);
class UIReservationConnection_X* BeginConnect(struct FServerReservationData Reservation);
void Close();
void CloseConnection(class UIReservationConnection_X* Connection);
void eventBroadcastMessage(class UObject* Message);
void SendReservationMessage(class UObject* Message, struct FServerReservationData&
Reservation);
void eventSendMessageW(class UIReservationConnection_X* Connection, class UObject*
Message);
class UBeaconMessage_X* CreateMessage(class UClass* MessageClass);
void ReceiveMessage(class UIReservationConnection_X* Connection, class UObject* Message);
void RemoveMessageHandler(struct FScriptDelegate Callback);
void AddMessageHandler(class UClass* MessageClass, struct FScriptDelegate Callback);

```

```

void eventConstruct();
void EventDisconnected(class UReservationBeacon_X* Beacon, class
UIReservationConnection_X* Connection);
void EventConnected(class UReservationBeacon_X* Beacon, class UIReservationConnection_X*
Connection);
void MessageReceivedDelegate(class UIReservationConnection_X* Connection, class UObject*
Message);
};

// Class ProjectX.BeaconMessage_X
// 0x0000 (0x0060 - 0x0060)
class UBeaconMessage_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.BeaconMessage_X");
}

return uClassPointer;
};

class UBeaconMessage_X* Broadcast();
class UBeaconMessage_X* Send(class UIReservationConnection_X* Connection);
};

// Class ProjectX.BreadcrumbBroadcaster_X
// 0x0028 (0x0060 - 0x0088)
class UBreadcrumbBroadcaster_X : public UObject
{
public:
class UBreadcrumbConfig_X*          BreadcrumbConfig;          // 0x0060
(0x0008) [0x000000000000000001] (CPF_Edit)
struct FPointer          BreadcrumbInstance;          // 0x0068 (0x0008)
[0x0000000000000001000] (CPF_Native)
struct FScriptDelegate          __OnCrumbAddedDelegate__Delegate;          // 0x0070
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.BreadcrumbBroadcaster_X");
}
}

```

```

return uClassPointer;
};

void PollCrumbs();
void Unsubscribe(class UObject* Subscriber);
void Subscribe(struct FScriptDelegate Callback);
void HandleConfigChanged(class UBreadcrumbConfig_X* Config);
void SetPollTimer(int32_t IntervalSeconds);
void eventOnInit();
void InitGlobalInstance_Native();
static void InitGlobalInstance(class UGameEngine* Engine);
void OnCrumbAddedDelegate(class FString Category, class FString Value);
};

// Class ProjectX.CachedWebData_X
// 0x005C (0x0060 - 0x00BC)
class UCachedWebData_X : public UObject
{
public:
    TArray<uint8_t> Data; // 0x0060 (0x0010)
    [0x000000000000400002] (CPF_Const | CPF_NeedCtorLink)
    int32_t DataCRC; // 0x0070 (0x0004)
    [0x000000000000000002] (CPF_Const)
    class FString ETag; // 0x0078 (0x0010)
    [0x000000000000400002] (CPF_Const | CPF_NeedCtorLink)
    class FString ContentType; // 0x0088 (0x0010)
    [0x000000000000400002] (CPF_Const | CPF_NeedCtorLink)
    class FString Signature; // 0x0098 (0x0010)
    [0x000000000000400002] (CPF_Const | CPF_NeedCtorLink)
    class UError* LoadError; // 0x00A8 (0x0008)
    [0x00000000000002000] (CPF_Transient)
    class UError* DownloadError; // 0x00B0 (0x0008)
    [0x00000000000002000] (CPF_Transient)
    unsigned long bNewData : 1; // 0x00B8 (0x0004)
    [0x00000000000002000] [0x00000001] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CachedWebData_X");
}

return uClassPointer;
};

class FString GetText();
void SetData(TArray<uint8_t> NewData, class FString NewETag, class FString NewContentType,
class FString NewSignature);
};

```



```

// Class ProjectX.Camera_X
// 0x0110 (0x05B8 - 0x06C8)
class ACamera_X : public ACamera
{
public:
TArray<class UCameraState_X*> States; // 0x05B8 (0x0010)
[0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)
struct FRotator PCDeltaRotation; // 0x05C8 (0x000C)
[0x0000000000002000] (CPF_Transient)
struct FRotator OldControllerRotation; // 0x05D4 (0x000C)
[0x0000000000002000] (CPF_Transient)
struct FVector PCDeltaLocation; // 0x05E0 (0x000C)
[0x0000000000002000] (CPF_Transient)
struct FVector OldControllerLocation; // 0x05EC (0x000C)
[0x0000000000002000] (CPF_Transient)
struct FCameraOrientation PreProcessPOV; // 0x05F8 (0x002C)
[0x0000000000002000] (CPF_Transient)
struct FCameraOrientation PostProcessPOV; // 0x0624 (0x002C)
[0x0000000000002000] (CPF_Transient)
class UCameraStateBlender_X* Blender; // 0x0650 (0x0008)
[0x00000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
struct FVector ShakeLocationOffset; // 0x0658 (0x000C)
[0x0000000000002002] (CPF_Const | CPF_Transient)
struct FRotator ShakeRotationOffset; // 0x0664 (0x000C)
[0x0000000000002002] (CPF_Const | CPF_Transient)
float ShakeFOVOffset; // 0x0670 (0x0004)
[0x0000000000002002] (CPF_Const | CPF_Transient)
struct FColor StartFadeColor; // 0x0674 (0x0004)
[0x0000000000002000] (CPF_Transient)
struct FColor EndFadeColor; // 0x0678 (0x0004)
[0x0000000000002000] (CPF_Transient)
struct FVector ClipOffset; // 0x067C (0x000C)
[0x0000000000002000] (CPF_Transient)
TArray<struct FLocationCameraKnock> LocationKnocks; // 0x0688
(0x0010) [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<struct FRotationCameraKnock> RotationKnocks; // 0x0698
(0x0010) [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
unsigned long bDisableCameraShake : 1; // 0x06A8 (0x0004)
[0x0000000000002000] [0x00000001] (CPF_Transient)
unsigned long bSnapNextTransition : 1; // 0x06A8 (0x0004)
[0x0000000000002000] [0x00000002] (CPF_Transient)
struct FScriptDelegate __EventCameraStateChanged__Delegate; // 0x06B0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Camera_X");
}
}

```

```

return uClassPointer;
};

void __Camera_X__Destroyed_0x1(class UCameraState_X* S);
void Destroyed();
void eventOnViewTargetChanged();
void SetViewTarget(class AActor* NewViewTarget, struct FViewTargetTransitionParams
TransitionParams);
bool IsTransitioning();
void HandleBlenderStateChanged(class UCameraStateBlender_X* CameraBlender);
void SnapTransition();
void PrintDebugInfo(class UDebugDrawer* Drawer);
void ApplyCameraModifiers(float DeltaTime, struct FTPOV& OutPOV);
bool CameraTrace(struct FVector End, struct FVector Start, struct FVector Extent, struct
FVector& HitLoc, class AActor*& HitActor);
class UCameraState_X* InstanceCameraState(class UCameraState_X* Archetype);
void eventModifyPostProcessSettings(struct FPostProcessSettings& PP);
void ProcessViewRotation(float DeltaTime, struct FRotator& OutViewRotation, struct FRotator&
OutDeltaRot);
void PostProcessCameraState(float DeltaTime, struct FCameraOrientation& OutPOV);
void ProcessCameraState(float DeltaTime, struct FCameraOrientation& OutPOV);
void CheckViewTarget(struct FTVViewTarget& VT);
void CopyFade(class ACamera_X* Other);
void SetCameraFade(unsigned long bNewEnableFading, struct FColor NewFadeColor, struct
FVector2D NewFadeAlpha, float NewFadeTime, unsigned long bNewFadeAudio);
void UpdateFade(float DeltaTime);
void eventUpdateCamera(float DeltaTime);
void ClampPOV(struct FCameraOrientation& OutPOV);
struct FRotator RemoveRoll(struct FRotator InRot);
class UCameraState_X* GetCameraState();
class UCameraState_X* SelectCameraState();
void UpdateCameraState();
void UpdateCameraKnocks(float DeltaTime, struct FCameraOrientation& OutPOV);
void AddRotationKnock(struct FRotationCameraKnock Knock, float Scale);
void AddLocationKnock(struct FLocationCameraKnock Knock, float Scale, struct FRotator
Transform);
void InstanceCameraStates();
void InitializeFor(class APlayerController* PC);
void ModifyCameraShakeScale(class UCameraShake* Shake, float NewScale);
void OnLoadingMovieClosesd();
void eventPostBeginPlay();
static struct FVector InterpVector(struct FVector OldValue, struct FVector NewValue, struct
FVectorInterpRate Rate, struct FRotator Orientation, float DeltaTime);
static void FinalizeOrientation(struct FCameraOrientation& OutPOV);
static struct FCameraOrientation BlendCameraOrientations(float Alpha, struct
FCameraOrientation& A, struct FCameraOrientation& B);
void EventCameraStateChanged(class ACamera_X* Camera, class UCameraState_X*
CameraState);
};

// Class ProjectX.CameraAttachmentComponent_X
// 0x0017 (0x009D - 0x00B4)
class UCameraAttachmentComponent_X : public UActorComponent
{

```

```

public:
TArray<class UActorComponent*> Attachments; // 0x00A0
(0x0010) [0x0000000004480009] (CPF_Edit | CPF_ExportObject | CPF_Component |
CPF_NeedCtorLink | CPF_EditInline)
unsigned long bDirtyTransform : 1; // 0x00B0 (0x0004)
[0x0000000000002002] [0x00000001] (CPF_Const | CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CameraAttachmentComponent_X");
}

return uClassPointer;
};

};

// Class ProjectX.CanvasTexture_X
// 0x0018 (0x0198 - 0x01B0)
class UCanvasTexture_X : public UTextureRenderTarget2D
{
public:
struct FScriptDelegate __Draw__Delegate; // 0x0198 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CanvasTexture_X");
}

return uClassPointer;
};

void Draw(class UCanvas* C);
};

// Class ProjectX.CanvasTextureComponent_X
// 0x0023 (0x009D - 0x00C0)
class UCanvasTextureComponent_X : public UActorComponent
{
public:
class UMeshComponent* Mesh; // 0x00A0 (0x0008)
[0x000000000408000B] (CPF_Edit | CPF_Const | CPF_ExportObject | CPF_Component |
CPF_EditInline)

```

```

int32_t                MaterialIndex;                // 0x00A8 (0x0004)
[0x0000000000000003] (CPF_Edit | CPF_Const)
struct FName           MaterialParamName;           // 0x00AC (0x0008)
[0x0000000000000003] (CPF_Edit | CPF_Const)
class UCanvasTexture_X* Texture;                   // 0x00B8 (0x0008)
[0x000000000440000B] (CPF_Edit | CPF_Const | CPF_ExportObject | CPF_NeedCtorLink |
CPF_EditInline)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CanvasTextureComponent_X");
}

return uClassPointer;
};

void OnDraw(class UCanvas* C);
};

// Class ProjectX.PlayerController_X
// 0x00A8 (0x07D0 - 0x0878)
class APlayerController_X : public APlayerController
{
public:
class ACamera*         CameraArchetype;            // 0x07D0 (0x0008)
[0x0000000000000001] (CPF_Edit)
class AHUD*            HUDArchetype;               // 0x07D8 (0x0008)
[0x0000000000000001] (CPF_Edit)
class APawn*           OldPawnReference;           // 0x07E0 (0x0008)
[0x0000000000000200] (CPF_Transient)
class AActor*          LockedDebugActor;           // 0x07E8 (0x0008)
[0x0000000000000200] (CPF_Transient)
unsigned long          bPausedForExternalUI : 1;    // 0x07F0 (0x0004)
[0x0000000000000200] [0x00000001] (CPF_Transient)
class UNetConnection*  NetConnection;              // 0x07F8 (0x0008)
[0x0000004000000200] (CPF_Transient)
struct FScriptDelegate __EventReceivedPRI__Delegate; // 0x0800
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventPawnChange__Delegate; // 0x0818
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventReceivedPlayer__Delegate; // 0x0830
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventDestroyed__Delegate; // 0x0848
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventPlayerCameraChange__Delegate; // 0x0860
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()

```

```

{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class ProjectX.PlayerController_X");
}

return UClassPointer;
};

void ToggleDebugCamera();
void DebugCrosshair();
void DebugKill();
void DebugKillAllExcept();
bool GetAutoAimTarget(class AActor*& HitActor, struct FVector& HitLoc);
void __PlayerController_X__ReceivedPlayer_0x1(class UObject* _);
void ClientSetSeasonReward(struct FPlayerSeasonRewardProgress Reward);
void ServerUpdateCustomMatchSettings(struct FCustomMatchSettings Settings);
void ServerSetParty(struct FUniqueNetId MemberId, struct FUniqueNetId NewPartyId);
void DisconnectSplitScreenPlayer(class UPlayer* P);
void ServerDestroy();
void KickPlayerForReason(class FString Reason, class FString Title);
void ClientNetLag(float PktLag, float PktVariance, float PktLoss, float PktDup, float PktSpike);
void ServerNetLag(float PktLag, float PktVariance, float PktLoss, float PktDup, float PktSpike);
void Pause();
float GetConnectionTimeSinceLastReceived();
void RemoveOptionFromLastURL(class FString OptName);
void PrintDebugInfo(class UDebugDrawer* Drawer);
void eventClearOnlineDelegates();
void RegisterOnlineDelegates();
void eventClientPlayForceFeedbackWaveform(class UForceFeedbackWaveform* FFWaveform,
class AActor* FFWaveformInstigator);
void ClientCommand(class FString Command);
void ServerCommand(class FString Command);
void eventUnPossess();
void eventPossess(class APawn* aPawn);
void OnPawnChange(class APawn* OldPawn, class APawn* NewPawn);
void PawnReferenceUpdated();
void ClientRestart(class APawn* NewPawn);
void GivePawn(class APawn* NewPawn);
void eventInitInputSystem();
class ACamera* GetCameraArchetype();
void eventSpawnPlayerCamera();
void SetPlayerCamera(class ACamera* NewCamera);
void AddCheats(unsigned long bForce);
class UOnlineGameAccount_X* GetOnlineAccount();
class UOnlinePlayer_X* GetOnlinePlayer();
void OnReceivedPlayerAndPRI();
void InitPlayerReplicationInfo();
void eventDestroyed();
void OnExternalUIChanged(unsigned long blsOpening);
void SetPaused(unsigned long bPaused);
void eventReceivedPlayer();

```

```

static class UOnlineGame_X* GetOnlineGame();
static class UOnlineGameHost_X* GetOnlineGameHost();
static class UOnlineGameDedicatedServer_X* GetOnlineGameDedicatedServer();
void SpawnDefaultHUD();
void SetPRI(class APlayerReplicationInfo* PRI);
void eventReplicatedEvent(struct FName VarName);
void EventPlayerCameraChange(class APlayerController_X* PC);
void EventDestroyed(class APlayerController_X* PC);
void EventReceivedPlayer(class APlayerController_X* PC);
void EventPawnChange(class APlayerController_X* PC, class APawn* OldPawn, class APawn*
NewPawn);
void EventReceivedPRI(class APlayerController_X* PC);
};

// Class ProjectX.CheatManager_X
// 0x0018 (0x0080 - 0x0098)
class UCheatManager_X : public UCheatManager
{
public:
    struct FScriptDelegate          __EventDelegateTest__Delegate;          // 0x0080
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.CheatManager_X");
        }

        return uClassPointer;
    };

    void HandleImageDownloaded(struct FOnlineImageDownload Image);
    void DownloadImage(class FString URL);
    void RandomSleep(float MinDelay, float MaxDelay, float MinSleep, float MaxSleep);
    void Sleep(float Seconds);
    void LoadKismetLevel(struct FName LeveFileName);
    void DrawBulletContacts(float LifeTime);
    void PrintReproPhysics(class FString Command, class AActor* A);
    void ReproPhysics(class AActor* A, float LX, float LY, float LZ, float RX, float RY, float RZ, float RW,
float VX, float VY, float VZ, float AX, float AY, float AZ);
    void DisableCCD();
    void SetNetServerMaxTickRate(float Rate);
    void ScreenShotDir(class FString Directory);
    void ListVisiblePrimitives();
    void HideVisiblePrimitives(class FString NameMatch, class FString NameIgnore);
    void ToggleReplicateVoiceToSelf();
    void ToggleReplicateVoiceToSender();
    void ShutdownCheatManager();
    void InitCheatManager();
    void EventDelegateTest();

```

```

};

// Class ProjectX.CheatObject_X
// 0x0000 (0x0060 - 0x0060)
class UCheatObject_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CheatObject_X");
}

return uClassPointer;
};

};

// Class ProjectX.ColorPalette_X
// 0x0020 (0x0060 - 0x0080)
class UColorPalette_X : public UObject
{
public:
int32_t DefaultId; // 0x0060 (0x0004)
[0x0000000000000001] (CPF_Edit)
int32_t HueCount; // 0x0064 (0x0004)
[0x0000000000002003] (CPF_Edit | CPF_Const | CPF_EditConst)
int32_t ValueCount; // 0x0068 (0x0004)
[0x0000000000002003] (CPF_Edit | CPF_Const | CPF_EditConst)
TArray<struct FLinearColor> Colors; // 0x0070 (0x0010)
[0x0000000000042003] (CPF_Edit | CPF_Const | CPF_EditConst | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ColorPalette_X");
}

return uClassPointer;
};

int32_t IntColorToID(int32_t ColorInt);
int32_t IDToColorInt(int32_t ColorID);
struct FColorPosition GetColorPosition(int32_t ColorID);
struct FLinearColor GetValidColor(struct FLinearColor InColor);

```

```

int32_t GetClosestID(struct FLinearColor InColor);
struct FLinearColor GetColor(int32_t ColorID);
};

// Class ProjectX.Compression_X
// 0x0000 (0x0060 - 0x0060)
class UCompression_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Compression_X");
}

return uClassPointer;
};

bool Uncompress(TArray<uint8_t>& Compressed, TArray<uint8_t>& Uncompressed);
bool Compress(TArray<uint8_t>& Uncompressed, TArray<uint8_t>& Compressed);
};

// Class ProjectX.ControlPreset_X
// 0x0048 (0x0060 - 0x00A8)
class UControlPreset_X : public UObject
{
public:
TArray<struct FPlayerBinding> PCBindings; // 0x0060 (0x0010)
[0x0000000400040400] (CPF_Config | CPF_NeedCtorLink)
TArray<struct FPlayerBinding> GamepadBindings; // 0x0070
(0x0010) [0x0000000400040400] (CPF_Config | CPF_NeedCtorLink)
unsigned long bRemovedPCOverrides : 1; // 0x0080 (0x0004)
[0x0000000000000000] [0x00000001]
unsigned long bRemovedGamepadOverrides : 1; // 0x0080
(0x0004) [0x0000000000000000] [0x00000002]
TArray<struct FPlayerBinding> SteamInputBindings; // 0x0088
(0x0010) [0x0000000400040400] (CPF_Config | CPF_NeedCtorLink)
struct FName CustomPresetName; // 0x0098 (0x0008)
[0x0000000000000000] (CPF_Const)
struct FName DefaultPresetName; // 0x00A0 (0x0008)
[0x0000000000000000] (CPF_Const)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{

```



```

uClassPointer = UObject::FindClass("Class ProjectX.ControlPreset_X");
}

return uClassPointer;
};

TArray<struct FPlayerBinding> GetGamepadBindings();
TArray<struct FPlayerBinding> GetPCBindings();
};

// Class ProjectX.CrashReport_X
// 0x0000 (0x0060 - 0x0060)
class UCrashReport_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CrashReport_X");
}

return uClassPointer;
};

void eventHandleHttpRequestComplete(class UHttpRequestInterface* Request, class
UHttpResponseInterface* Response, unsigned long bSuccess);
};

// Class ProjectX.CRC_X
// 0x0000 (0x0060 - 0x0060)
class UCRC_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CRC_X");
}

return uClassPointer;
};

static int32_t CrcBytes(TArray<uint8_t>& Bytes);
static int32_t CrcString(class FString Text);

```

```

};

// Class ProjectX.DistributionFloatShakeParameter_X
// 0x0007 (0x00A1 - 0x00A8)
class UDistributionFloatShakeParameter_X : public UDistributionFloatParameterBase
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.DistributionFloatShakeParameter_X");
}

return uClassPointer;
};

};

// Class ProjectX.EffectsMap_X
// 0x0010 (0x0060 - 0x0070)
class UEffectsMap_X : public UObject
{
public:
TArray<struct FEffectsMapping> Effects; // 0x0060 (0x0010)
[0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EffectsMap_X");
}

return uClassPointer;
};

struct FEffectsMapping GetEffects(class UPhysicalMaterial* PhysMat);
};

// Class ProjectX.EngineShare_X
// 0x0110 (0x0060 - 0x0170)
class UEngineShare_X : public UEngineShare
{
public:
class UOnlineGame_Base_X* OnlineGame; // 0x0060 (0x0008)
[0x00000004000002000] (CPF_Transient)

```

```

class ULocalCache_X*                LocalCache;                // 0x0068 (0x0008)
[0x0000004000002000] (CPF_Transient)
class FString                      ArchetypeString_OnlineGame;    // 0x0070 (0x0010)
[0x0000080000400003] (CPF_Edit | CPF_Const | CPF_NeedCtorLink)
class FString                      ArchetypeString_OnlineGameDedicatedServer; // 0x0080
(0x0010) [0x0000080000400003] (CPF_Edit | CPF_Const | CPF_NeedCtorLink)
unsigned long                      bDisableSaving : 1;           // 0x0090 (0x0004)
[0x0000000000002000] [0x00000001] (CPF_Transient)
unsigned long                      bIsShowingLoadmapMovie : 1;   // 0x0090 (0x0004)
[0x0000000000002002] [0x00000002] (CPF_Const | CPF_Transient)
unsigned long                      bUndocked : 1;               // 0x0090 (0x0004)
[0x0000000000000002] [0x00000004] (CPF_Const)
struct FScriptDelegate             __EventPreExit__Delegate;    // 0x0098 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate             __EventLoadingMovieClosed__Delegate; // 0x00B0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate             __EventPreScriptsReloaded__Delegate; // 0x00C8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate             __EventPostScriptsReloaded__Delegate; // 0x00E0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate             __EventOnlineGameInitialized__Delegate; // 0x00F8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate             __EventPreLoadMap__Delegate; // 0x0110
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate             __EventPostLoadMap__Delegate; // 0x0128
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate             __EventCrashed__Delegate;    // 0x0140
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate             __EventUndockedChanged__Delegate; // 0x0158
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EngineShare_X");
}

return uClassPointer;
};

void AddDynamicallyLoadedPackages(TArray<struct FDynamicallyLoadedPackage>&
OutPackages);
void PrintDebugInfo(class UDebugDrawer* Drawer);
void eventPreExit();
static bool IsRequestingExit();
static class FString GetUserAgentHeader();
class FString GetRegionString();
class FString GetPlatformString();
static bool ConfigGetString(class FString SectionName, class FString KeyName, class FString&
OutValue);

```

```

static bool ParseCommandlineStringArr(class FString Param, TArray<class FString>&
OutValues);
static bool ParseCommandlineString(class FString Param, unsigned long
bShouldStopOnComma, class FString& OutValue);
static bool ParseCommandlineQWORD(class FString Param, uint64_t& OutValue);
static bool ParseCommandlineFloat(class FString Param, float& OutValue);
static bool ParseCommandlineInt(class FString Param, int32_t& OutValue);
static bool ParseCommandlineParam(class FString Param);
void InitOnlineGame(class UOnlineSubsystem* NewOnlineSubsystem);
static bool IsUsingNullRHI();
static bool IsAuthoritative();
static bool IsNetServer();
static bool IsNetClient();
static bool IsHeadlessGameClient();
static bool IsGameClient();
static bool IsDedicatedServer();
void EventUndockedChanged();
void EventCrashed();
void EventPostLoadMap();
void EventPreLoadMap(class FString MapName);
void EventOnlineGameInitialized();
void EventPostScriptsReloaded();
void EventPreScriptsReloaded();
void EventLoadingMovieClosed();
void EventPreExit();
};

// Class ProjectX.EOS_ErrorResponse
// 0x0038 (0x0060 - 0x0098)
class UEOS_ErrorResponse : public UObject
{
public:
class FString                ErrorMessage;                // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
int32_t                      NumericErrorCode;            // 0x0070 (0x0004)
[0x000000000000000000]
class FString                ErrorCode;                    // 0x0078 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
TArray<class FString>        MessageVars;                // 0x0088 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EOS_ErrorResponse");
}

return uClassPointer;
};

```

```

};

// Class ProjectX.EOS_GetAccountsResponse
// 0x0010 (0x0060 - 0x0070)
class UEOS_GetAccountsResponse : public UObject
{
public:
    TArray<struct FEOSAccountInfo> Accounts; // 0x0060 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.EOS_GetAccountsResponse");
        }

        return uClassPointer;
    };
};

// Class ProjectX.EOS_AccountSummaryResponse
// 0x0050 (0x0070 - 0x00C0)
class UEOS_AccountSummaryResponse : public UEOS_GetAccountsResponse
{
public:
    TArray<struct FEOSAccountInfo> Friends; // 0x0070 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)
    TArray<struct FEOSAccountInfo> Incoming; // 0x0080 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)
    TArray<struct FEOSAccountInfo> Outgoing; // 0x0090 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)
    TArray<struct FEOSAccountInfo> BlockList; // 0x00A0 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)
    struct FAccountSettings Settings; // 0x00B0 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.EOS_AccountSummaryResponse");
        }

        return uClassPointer;
    };
};

```

```

};

// Class ProjectX.EOS_ManageBlockListResponse
// 0x0020 (0x0060 - 0x0080)
class UEOS_ManageBlockListResponse : public UObject
{
public:
    class FString AccountId; // 0x0060 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)
    class FString Created; // 0x0070 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.EOS_ManageBlockListResponse");
        }

        return uClassPointer;
    };
};

// Class ProjectX.EOS_ManageFriendsListResponse
// 0x0010 (0x0060 - 0x0070)
class UEOS_ManageFriendsListResponse : public UObject
{
public:
    class FString Status; // 0x0060 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.EOS_ManageFriendsListResponse");
        }

        return uClassPointer;
    };
};

// Class ProjectX.EOSMetricEvent_X
// 0x0008 (0x0060 - 0x0068)
class UEOSMetricEvent_X : public UObject
{

```

```

public:
struct FName                                     EventName;                               // 0x0060 (0x0008)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EOSMetricEvent_X");
}

return uClassPointer;
};

};

// Class ProjectX.EOSMetrics_X
// 0x0070 (0x0060 - 0x00D0)
class UEOSMetrics_X : public UObject
{
public:
class UPsyNet_X*                               PsyNet;                               // 0x0060 (0x0008)
[0x0001800000000000]
class UEOSMetricsConfig_X*                   EOSMetricsConfig;                               // 0x0068
(0x0008) [0x0001000000000001] (CPF_Edit)
class ULocalPlayer_X*                       LocalPlayer;                               // 0x0070 (0x0008)
[0x0001004000000000]
unsigned long                                bCrashing : 1;                               // 0x0078 (0x0004)
[0x0001004000000000] [0x00000001]
class FString                                EMGuid;                                       // 0x0080 (0x0010)
[0x0001004000400000] (CPF_NeedCtorLink)
class FString                                QueryParamsTemplate;                         // 0x0090 (0x0010)
[0x0001004000400000] (CPF_NeedCtorLink)
struct FMetricEventJsonStruct                CurrentEvents;                               // 0x00A0 (0x0010)
[0x0001004000400000] (CPF_NeedCtorLink)
class FString                                LatestLocation;                              // 0x00B0 (0x0010)
[0x0001004000400000] (CPF_NeedCtorLink)
class FString                                StreamingServiceType;                        // 0x00C0 (0x0010)
[0x0001004000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EOSMetrics_X");
}

return uClassPointer;
};

```

```

};

void __EOSMetrics_X__Init_0x1();
void __EOSMetrics_X__Init_0x2(class UEOSMetricsConfig_X* EpicMetric);
class FString GetUserIdNative(struct FUniqueNetId LocalId, class FString EpicId);
void SetLocation(class FString InLocation);
void AddEvent(class UEOSMetricEvent_X* InEvent);
void HandleCrash();
class FString GetUploadType();
class FString GetUserId();
class FString GetURL();
class UEOSEvent_Presence_X* CreatePresenceEvent();
void SendPing();
void Init(class ULocalPlayer_X* InLocalPlayer);
};

// Class ProjectX.LocalPlayer_X
// 0x0018 (0x04D0 - 0x04E8)
class ULocalPlayer_X : public ULocalPlayer
{
public:
class UOnlinePlayer_X*                OnlinePlayer;                // 0x04D0 (0x0008)
[0x00000080000002000] (CPF_Transient)
class UCabinedOrGuest*                CabinedOrGuest_Object;        // 0x04D8
(0x0008) [0x00000000000002000] (CPF_Transient)
class UCabinedOrGuest*                CabinedOrGuest_Interface;    // 0x04E0
(0x0008) [0x00000000000002000] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.LocalPlayer_X");
}

return uClassPointer;
};

bool SpawnPlayActor(class FString URL, class FString& OutError);
void eventNotifyServerConnectionOpen();
void NotifyOnValidPlayerController(struct FScriptDelegate Callback, class UClass* PCClass);
bool GetEosNameIfAvailable(class FString& EosName);
class FString eventGetNickname();
void OnRemoved();
void HandleOnlineNameChanged(class UOnlinePlayer_X* PrimaryOnlinePlayer);
void OnCreated();
};

// Class ProjectX.StateObject_X
// 0x0004 (0x0060 - 0x0064)
class UStateObject_X : public UObject

```



```

{
public:
    unsigned long                bDebug : 1;                // 0x0060 (0x0004)
    [0x0000000000000001] [0x00000001] (CPF_Edit)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.StateObject_X");
        }

        return uClassPointer;
    };

    void InitExecution();
    void eventDestroyed();
    void Destroy();
};

// Class ProjectX.Online_X
// 0x004C (0x0064 - 0x00B0)
class UOnline_X : public UStateObject_X
{
public:
    class UOnlineSubsystem*      OnlineSub;                // 0x0068 (0x0008)
    [0x0000000400000200] (CPF_Transient)
    class UPsyNet_X*            PsyNet;                    // 0x0070 (0x0008)
    [0x0000000400000200] (CPF_Transient)
    class UOnlineSubsystem*      EOS;                      // 0x0078 (0x0008)
    [0x0000000800000200] (CPF_Transient)
    struct FScriptDelegate      __EventEOSInitialized__Delegate;    // 0x0080
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
    struct FScriptDelegate      __EventEosInitTimeout__Delegate;    // 0x0098
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.Online_X");
        }

        return uClassPointer;
    };

    void __Online_X__InitializeEOS_0x1(class UOSSConfig_X* OSSConfig);
    void __Online_X__InitializeEOS_0x2(class UEpicConfig_X* EpicConfig);

```

```

class UOnlineSubsystem* GetOrCreateEOS(class FString& SandboxId, class FString&
DeploymentId);
void NotifyWhenEOSInitialized(struct FScriptDelegate Callback);
void HandleLinkStatusChangedEOSInit(unsigned long bHasInternetConnection);
void OnEosInitTimeout();
void InitializeEOS();
static class FString BuildAddress(class FString Host, int32_t Port, class FString IP);
bool IsInOnlineGame();
bool IsInMainMenu();
class AGRI_X* GetGRI();
class AWorldInfo* GetWorldInfo();
static struct FUniqueNetId CreateUniqueNetId(uint8_t Platform, class FString PlatformName,
uint64_t PlatformUID);
static bool UniqueLobbyIDsSet(struct FUniqueLobbyId InId);
static struct FUniqueLobbyId StringToUniqueLobbyId(class FString InId);
static class FString UniqueLobbyIdToString(struct FUniqueLobbyId InId);
static struct FUniqueNetId GetUniqueNetIdWithoutSplitscreenID(struct FUniqueNetId Id);
static struct FUniqueNetId GetPrimaryIDForGuest(struct FUniqueNetId InGuestID);
static bool UniqueNetIdIsGuest(struct FUniqueNetId InId);
static bool UniqueNetIdIsPS4Legacy(struct FUniqueNetId InId);
static bool UniqueNetIdHasValue(struct FUniqueNetId InId);
static bool UniqueNetIdIsValid(struct FUniqueNetId InId);
static struct FUniqueNetId StringToUniqueNetId(class FString InId);
static class FString PlatformIdToString(struct FUniqueNetId& InId);
static class FString UniqueNetIdArrayToString(TArray<struct FUniqueNetId>& Ids);
static class FString UniqueNetIdToString(struct FUniqueNetId InId);
static struct FUniqueNetId CleanUniqueNetId(struct FUniqueNetId InId);
static int32_t FindUniqueNetId(TArray<struct FUniqueNetId>& List, struct FUniqueNetId&
Search);
class UError* GetConnectionStatusError(int32_t ErrorCode, class UError* FallbackError);
void OnExit();
void OnNewGame();
void OnMainMenuOpened();
void HandleGRISpawned(class AGRI_X* GRI);
void OnInit();
void Init(class UOnlineSubsystem* NewOnlineSubsystem);
void EventEosInitTimeout();
void EventEOSInitialized();
};

```

```

// Class ProjectX.OnlinePlayer_X
// 0x01E0 (0x00B0 - 0x0290)
class UOnlinePlayer_X : public UOnline_X
{
public:
class UOnlinePlayerFriends_X*           Friends;           // 0x00B0 (0x0008)
[0x0000008000000001] (CPF_Edit)
class UOnlinePlayerAuthentication_X*     Authentication;    // 0x00B8
(0x0008) [0x0000008000000001] (CPF_Edit)
class UPsyNetConnection_X*             PsyNetConnection;    // 0x00C0
(0x0008) [0x0000004000000001] (CPF_Edit)
class UOnlinePlayerStorage_X*           Storage;           // 0x00C8 (0x0008)
[0x0001004000000001] (CPF_Edit)
class UOnlinePlayerRegionRestrictions_X* RegionRestrictions; // 0x00D0

```

```

(0x0008) [0x0001004000000001] (CPF_Edit)
class UUserBugReportComponent_X*          BugReporter;          // 0x00D8
(0x0008) [0x0001004000000001] (CPF_Edit)
class FString                             PlayerName;           // 0x00E0 (0x0010)
[0x0000000000040200] (CPF_Transient | CPF_NeedCtorLink)
struct FUniqueNetId                       PlayerID;             // 0x00F0 (0x0048)
[0x0000000800040200] (CPF_Transient | CPF_NeedCtorLink)
struct FUniqueNetId                       CleanPlayerID;         // 0x0138 (0x0048)
[0x0000000800040200] (CPF_Transient | CPF_NeedCtorLink)
uint8_t                                   LoginStatus;           // 0x0180 (0x0001)
[0x0000000800000200] (CPF_Transient)
int32_t                                   LocalPlayerNum;         // 0x0184 (0x0004)
[0x0000000000000200] (CPF_Transient)
unsigned long                             bLoggingIn : 1;        // 0x0188 (0x0004)
[0x0000000400000000] [0x00000001]
unsigned long                             bLoggingOut : 1;       // 0x0188 (0x0004)
[0x0000000400000000] [0x00000002]
unsigned long                             bCabinedModeStatusKnown : 1; // 0x0188
(0x0004) [0x0000000000000200] [0x00000004] (CPF_Transient)
class UError*                             LoginError;           // 0x0190 (0x0008)
[0x0000000400000200] (CPF_Transient)
class FString                             CachedEpicID;         // 0x0198 (0x0010)
[0x0000000400040200] (CPF_Transient | CPF_NeedCtorLink)
class UError*                             LoginStatusError;      // 0x01A8 (0x0008)
[0x0000000000000000]
class UError*                             BannedError;          // 0x01B0 (0x0008)
[0x0000000000000000]
struct FScriptDelegate                   __EventLoginComplete__Delegate; // 0x01B8
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                   __EventLogoutComplete__Delegate; // 0x01D0
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                   __EventLoginStatusChanged__Delegate; // 0x01E8
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                   __EventCountryUpdated__Delegate; // 0x0200
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                   __EventCanPlayOnlineChanged__Delegate; // 0x0218
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                   __EventShowKeyboardComplete__Delegate; // 0x0230
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                   __EventOnlineNameChanged__Delegate; // 0x0248
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                   __EventRemoved__Delegate; // 0x0260
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                   __EventCabinedModeStatusKnown__Delegate; //
0x0278 (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlinePlayer_X");

```

```

}

return uClassPointer;
};

void __OnlinePlayer_X__OnInit_0x1(class UOnlinePlayerAuthentication_X* _);
void __OnlinePlayer_X__UpdatePsyNetEnabled_0x1(class UBanMessage_X* _);
void __OnlinePlayer_X__InitPlayer_0x1();
void __OnlinePlayer_X__HandleLoginChanged_0x1();
void GetPlayersLikesDislikes(class UGameSettingPlaylist_X* Playlist, TArray<struct FName>& PlayerLikes, TArray<struct FName>& PlayerDislikes);
void PrintDebugInfo(class UDebugDrawer* Drawer);
void ShowCabinedReminder();
void ShowXboxGuestIsPrimaryPlayerWarningScreen();
void ForceCloseAgeGate();
void UnsupportedCorrectiveActionVerify();
void EpicDisplayNameScreenDone();
void ShowParentEmailScreen();
void ShowDOBScreen();
void ShowWelcomeToCabinedModeScreen();
void ShowWelcomeNotCabinedModeScreen();
void ShowEpicDisplayNameScreen();
void SetParentEmail(class FString Email);
bool ValidateEmail(class FString Email);
bool IsInCabinedModeOrGuest();
bool EnforceCabinedMode();
bool UnsupportedCorrectiveActionNeeded();
void ShowEpicAccountLinkScreen();
void SetAskConfirmDisplayName(unsigned long val);
bool AskConfirmDisplayName();
void SetAskParentEmail(unsigned long val);
bool AskParentEmail();
void SetShowedAgeGate(unsigned long val);
bool ShowedAgeGate();
void SetAskAge(unsigned long val);
bool AskAge();
void SetInCabinedMode(unsigned long val);
bool IsInCabinedMode();
bool PromptForPin();
void SetRequirePinForFriends(unsigned long bVal);
bool IsPinRequiredForFriends();
void NotifyWhenCabinedModeStatusKnown(struct FScriptDelegate Callback);
bool IsGuestAccount();
void SetPlayerName(class FString InName);
void UpdateSplitscreenId();
bool IsUpdateRequired();
class FString GetOnlineName();
class FString GetLocalName();
class FString GetRemoteName(int32_t LocalPlayerIndex);
void HandleCanPlayOnlineChanged(uint8_t LocalUserNum);
uint8_t CanPlayOnline();
void HandlePlayerCountryReceived(struct FUniqueNetId InPlayerId, class FString Country);
void GetPlayerCountry(struct FScriptDelegate Handler);
uint8_t CanCommunicateVoice(unsigned long bTryToResolve);

```

```

uint8_t CanCommunicateText(unsigned long bTryToResolve);
void OpenErrorDialog(uint8_t ErrorCode);
void OpenPS4DisplayMode(uint8_t DisplayMode, TArray<class FString> Targets, int32_t
ServiceLabel);
void ResetControllerColor(int32_t InControllerID);
void SetControllerColor(int32_t InControllerID, struct FColor NewColor);
void PostActivityFeedMessage(class FString Id, TArray<class FString> StringReplaceList);
TArray<bool> GetSyncedAchievements(uint8_t LocalUserNum);
void UnlockAchievement(uint8_t LocalUserNum, int32_t AchievementId, float PercentComplete);
void UpdateStat(struct FName StatName, int32_t Points);
void HandleShowKeyboardComplete(unsigned long bWasSuccessful);
void HideKeyboard();
bool ShowKeyboard(class FString TitleText, class FString DescriptionText, unsigned long
blsPassword, class FString DefaultText, int32_t MaxLength, struct FScriptDelegate
OnCompleteDelegate, int32_t LocalPlayerNumOverride);
void OnNewGame();
void HandleControllerChange(int32_t InControllerID, unsigned long blsConnected);
void HandleUserLoginStatusChange(uint8_t NewStatus, struct FUniqueNetId NewId);
bool IsLoggedIn(unsigned long bRequireOnlineLogin);
void HandleLogoutComplete(unsigned long bWasSuccessful);
void HandleLoginChanged(uint8_t PlayerNum);
class FString GetEpicAccountId();
void HandleLoginFailed(uint8_t PlayerNum, uint8_t ErrorCode);
void Logout(struct FScriptDelegate Callback);
void Login(class FString LoginName, class FString LoginPassword);
void OnRemoved();
void InitPlayer();
bool eventIsPrimaryPlayer();
void HandlePsyNetDisconnect(class UPsyNetConnection_X* Connection);
void UpdatePsyNetEnabled();
void HandleBanned(class UOnlinePlayerAuthentication_X* Auth);
void HandleAuthLoginChange(class UOnlinePlayerAuthentication_X* Auth);
void OnInit();
void eventConstruct();
class UOnlineGame_X* GetOnlineGame();
void EventCabinedModeStatusKnown(class UOnlinePlayer_X* Player);
void EventRemoved(class UOnlinePlayer_X* Player);
void EventOnlineNameChanged(class UOnlinePlayer_X* Player);
void EventShowKeyboardComplete(class FString NewText, unsigned long bCanceled);
void EventCanPlayOnlineChanged(class UOnlinePlayer_X* Player);
void EventCountryUpdated(class FString Country);
void EventLoginStatusChanged(class UOnlinePlayer_X* Player);
void EventLogoutComplete(class UOnlinePlayer_X* Player);
void EventLoginComplete(class UOnlinePlayer_X* Player, class UError* Error);
};

// Class ProjectX.EpicFriendsPlugin_X
// 0x00D8 (0x0060 - 0x0138)
class UEpicFriendsPlugin_X : public UObject
{
public:
class FString                                ServerURL;                                // 0x0060 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString                                FriendsServerURL;                            // 0x0070 (0x0010)

```

```

[0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
class FString                                AddFriendURL;                                // 0x0080 (0x0010)
[0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
class FString                                RemoveFriendURL;                                // 0x0090 (0x0010)
[0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
class FString                                OutgoingFriendRequestsURL;                    // 0x00A0 (0x0010)
[0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
class FString                                IncomingFriendRequestsURL;                    // 0x00B0 (0x0010)
[0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
class FString                                BlockPlayerURL;                                // 0x00C0 (0x0010)
[0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
class FString                                BlockListURL;                                // 0x00D0 (0x0010)
[0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
class FString                                AccountSummaryURL;                            // 0x00E0 (0x0010)
[0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
struct FScriptDelegate                      __HTTPRequestCallback__Delegate;              // 0x00F0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __AddFriendCallback__Delegate;                // 0x0108
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __GetOutgoingFriendsCallback__Delegate;        // 0x0120
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EpicFriendsPlugin_X");
}

return uClassPointer;
};

class UError* ConvertError(class UEOS_ErrorResponse* ErrorResponse);
bool GetAccountSummary();
bool GetOutgoingFriendRequestsWithCustomCallback(struct FScriptDelegate Callback);
bool GetOutgoingFriendRequests();
bool GetIncomingFriendRequests();
bool GetBlockList();
bool Unblock(struct FUniqueNetId UserId);
bool bLock(struct FUniqueNetId UserId);
bool RejectFriendRequest(struct FUniqueNetId FriendId);
bool AcceptFriendRequest(struct FUniqueNetId FriendId, class FString InPin);
bool RemoveFriend(struct FUniqueNetId FriendId);
bool AddFriendWithCustomCallback(struct FUniqueNetId FriendId, struct FScriptDelegate
Callback, class FString Pin);
bool AddFriend(struct FUniqueNetId FriendId, class FString Pin);
void eventConstruct();
void GetOutgoingFriendsCallback(class UEOS_GetAccountsResponse* Response, class UError*
Error);
void AddFriendCallback(class UEOS_ManageFriendsListResponse* Response, class UError*
Error, class FString& FriendId);

```

```

void HandleAccountSummaryResponse(class UEOS_AccountSummaryResponse* Response,
class UError* Error);
void HandleGetOutgoingInvitesResponse(class UEOS_GetAccountsResponse* Response, class
UError* Error);
void HandleGetIncomingInvitesResponse(class UEOS_GetAccountsResponse* Response, class
UError* Error);
void HandleGetBlockListResponse(class UEOS_GetAccountsResponse* Response, class UError*
Error);
void HandleUnblockPlayerResponse(class UEOS_ManageBlockListResponse* Response, class
UError* Error);
void HandleBlockPlayerResponse(class UEOS_ManageBlockListResponse* Response, class
UError* Error);
void HandleRejectFriendRequestResponse(class UEOS_ManageFriendsListResponse* Response,
class UError* Error, class FString& FriendId);
void HandleAcceptFriendRequestResponse(class UEOS_ManageFriendsListResponse*
Response, class UError* Error, class FString& FriendId);
void HandleRemoveFriendResponse(class UEOS_ManageFriendsListResponse* Response, class
UError* Error, class FString& FriendId);
void HandleAddFriendResponse(class UEOS_ManageFriendsListResponse* Response, class
UError* Error, class FString& FriendId);
void HTTPRequestCallback(class UObject* Response, class UError* Error);
};

```

```

// Class ProjectX.OnlinePlayerAuthentication_X

```

```

// 0x0158 (0x00B0 - 0x0208)

```

```

class UOnlinePlayerAuthentication_X : public UOnline_X

```

```

{
public:
class URPC_LoginAuthPlayer_X*          LoginRPC;                // 0x00B0 (0x0008)
[0x00000000000002000] (CPF_Transient)
class URPC_LoginAuthPlayer_X*          LoginSuccessRPC;        // 0x00B8
(0x0008) [0x00000004000002000] (CPF_Transient)
struct FUniqueNetId                    LoggedInPlayerId;        // 0x00C0 (0x0048)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                          PlayerUID;                // 0x0108 (0x0010)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                          Platform;                 // 0x0118 (0x0010)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)
unsigned long                          bLoggedIn : 1;            // 0x0128 (0x0004)
[0x00000004000000000] [0x000000001]
unsigned long                          bPlatformTokenAuthenticationFailed : 1; // 0x0128
(0x0004) [0x00000004000002000] [0x000000002] (CPF_Transient)
unsigned long                          bPlatformAuthTicketFailed_Switch : 1; // 0x0128
(0x0004) [0x00000004000002000] [0x000000004] (CPF_Transient)
unsigned long                          bSkipAuth : 1;            // 0x0128 (0x0004)
[0x00010040000000000] [0x000000008]
unsigned long                          bLastChanceAuthBan : 1;   // 0x0128 (0x0004)
[0x00000004000002000] [0x000000010] (CPF_Transient)
class UError*                          AuthLoginError;           // 0x0130 (0x0008)
[0x00000004000000000]
class UBanMessage_X*                  BanMessage;               // 0x0138 (0x0008)
[0x00000004000002000] (CPF_Transient)
class FString                          EncryptedAuthTicket;      // 0x0140 (0x0010)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)

```

```

class FString                                EpicAuthTicket;                                // 0x0150 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
int32_t                                     AuthRequestFailureMax;                                // 0x0160 (0x0004)
[0x00000000000004000] (CPF_Config)
int32_t                                     AuthRequestRetryTime;                                // 0x0164 (0x0004)
[0x00000000000000002] (CPF_Const)
int32_t                                     AuthRequestFailureCount;                                // 0x0168 (0x0004)
[0x00000000000002000] (CPF_Transient)
class FString                                AuthenticatedName;                                // 0x0170 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class UEpicLogin_X*                         EpicLogin;                                // 0x0180 (0x0008)
[0x00000008000000000]
class UError*                               PrimaryAccountNotSetError;                                // 0x0188 (0x0008)
[0x00000000000000000]
struct FScriptDelegate                      __EventLoginChanged__Delegate;                                // 0x0190
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventBanned__Delegate;                                // 0x01A8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventLoginResult__Delegate;                                // 0x01C0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate
__EventPlatformAuthTicketFailedChanged_Switch__Delegate; // 0x01D8 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventConnectionStatusChanged__Delegate; //
0x01F0 (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlinePlayerAuthentication_X");
}

return uClassPointer;
};

void RequestAuthTicket();
void GoToRequestEpicAuthTicket();
void RequestEpicAuthTicket();
void HandleEpicAuthTicket(class FString AuthTicket, class FString EpicAccountId, class UError*
Error);
void __OnlinePlayerAuthentication_X__OnInit_0x1();
class FString GetDebugName();
void GotoAuthState(struct FName AuthStateName);
void HandleAuthTicket(unsigned long bSuccess, class FString AuthTicket);
bool IsLoginAttemptActive();
struct FName GetFeatureSet();
class FString GetBuildRegion();
void SetPlatformAuthTicketFailed_Switch(unsigned long bNewValue);
bool RequiresEpicAuthTicket();
bool RequiresAuthTicket();

```



```

void ReLogin();
void Logout();
void SetAuthLoginError(class UError* E);
void UpdateLoginState();
class UError* GetAuthLoginError();
void UpdateAuthLoginError();
void HandlePsyNetLoginChanged(class UOnlinePlayerAuthentication_X* Auth);
void HandleLinkStatusChanged(unsigned long B);
void HandlePsyNetConnectionChanged(class UPsyNetConnection_X* C);
void HandleEosTimeout();
void HandleLoginStatusChanged(class UOnlinePlayer_X* Player);
void OnLoginFail(class UError* Error);
void OnLoginFailRPC(class URPC_LoginAuthPlayer_X* RPC);
void OnLoginSuccessRPC(class URPC_LoginAuthPlayer_X* RPC);
void SendLoginRPC();
void PsyNetLogin(struct FScriptDelegate Callback);
void HandleInternetConnectionChanged(unsigned long bConnected);
void HandleConnectionStatusChanged(uint8_t Status);
void OnRemoved();
void OnInit();
void EventConnectionStatusChanged(unsigned long bConnected);
void EventPlatformAuthTicketFailedChanged_Switch(unsigned long bFailure);
void EventLoginResult(class UOnlinePlayerAuthentication_X* Auth);
void EventBanned(class UOnlinePlayerAuthentication_X* Auth);
void EventLoginChanged(class UOnlinePlayerAuthentication_X* Auth);
};

```

```

// Class ProjectX.EpicLogin_X
// 0x0108 (0x0060 - 0x0168)
class UEpicLogin_X : public UObject
{
public:
class FString                                PinGrantCode;                                // 0x0060 (0x0010)
[0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
class FString                                PinGrantURL;                                // 0x0070 (0x0010)
[0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
uint64_t                                     PinGrantExpiration;                        // 0x0080 (0x0008)
[0x0000000040000000] (CPF_EditInlineNotify)
unsigned long                                bLoggedIn : 1;                            // 0x0088 (0x0004)
[0x0000000040000000] [0x00000001] (CPF_EditInlineNotify)
unsigned long                                bCorrectiveActionRequired : 1;            // 0x0088 (0x0004)
[0x0000000040000000] [0x00000002] (CPF_EditInlineNotify)
unsigned long                                bSetAsPrimaryAccount : 1;                // 0x0088 (0x0004)
[0x0000000000000000] [0x00000004]
unsigned long                                bPollDuringAccountLinking : 1;            // 0x0088 (0x0004)
[0x0000004040000000] [0x00000008] (CPF_EditInlineNotify)
unsigned long                                bPinGrantRequestInProgress : 1;          // 0x0088 (0x0004)
[0x0000000000000000] [0x00000010]
unsigned long                                bLoginInProgress : 1;                    // 0x0088 (0x0004)
[0x0000000000000000] [0x00000020]
unsigned long                                bAccountCreationInProgress : 1;          // 0x0088 (0x0004)
[0x0000000000000000] [0x00000040]
unsigned long                                bNintendoServiceAccountLinkingNeeded : 1; // 0x0088
(0x0004) [0x0000000000000000] [0x00000080]

```

```

unsigned long                bRefusedNintendoAccountAuthorization : 1;    // 0x0088
(0x0004) [0x0000000000000000] [0x00000100]
unsigned long                bUserSpecifiedPrimaryAccount : 1;           // 0x0088
(0x0004) [0x0000000000000000] [0x00000200]
unsigned long                bInitialized : 1;                           // 0x0088 (0x0004)
[0x0000000000000000] [0x00000400]
unsigned long                bTestLinkAccount : 1;                       // 0x0088 (0x0004)
[0x0001000000000000] [0x00000800]
class FString                ParentalConsentURL;                         // 0x0090 (0x0010)
[0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
uint8_t                     CabinedFlowState;                           // 0x00A0 (0x0001)
[0x0000000040000000] (CPF_EditInlineNotify)
uint8_t                     OldCabinedFlowState_Switch;                 // 0x00A1 (0x0001)
[0x0000000000000000]
int32_t                     LoginFailureCount;                          // 0x00A4 (0x0004)
[0x0000000000000000]
class UError*                LoginError;                                // 0x00A8 (0x0008)
[0x0000000000000000]
uint64_t                     LoginErrorTimestamp;                       // 0x00B0 (0x0008)
[0x0000004000000000]
class UEpicConfig_X*         EpicConfig;                                // 0x00B8 (0x0008)
[0x0000804000000000]
struct FScriptDelegate       __EventLoginSucceeded__Delegate;           // 0x00C0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventLoginFailed__Delegate;              // 0x00D8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventReceivedPinGrantCode__Delegate;     // 0x00F0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventLoginError__Delegate;               // 0x0108
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventReceivedEpicAuthTicket__Delegate;    // 0x0120
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventInitialized__Delegate;              // 0x0138 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventCreatedEpicGamesAccount__Delegate;  //
0x0150 (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EpicLogin_X");
}

return uClassPointer;
};

void __EpicLogin_X__HandleLoginChanged_0x2(class FString Ticket, class FString Id, class
UError* Err);
void __EpicLogin_X__HandleLoginFailed_0x1(class FString Ticket, class FString Id, class UError*
Err);

```

```

void __EpicLogin_X__UpdateTwoFactorAuthenticationStatus_0x2(class FString Ticket, class
FString Id, class UError* Err);
void __EpicLogin_X__CreateEpicGamesAccount_0x1(class UWebRequest_X* Response);
void __EpicLogin_X__NintendoAccountLogin_0x1(class FString AuthTicket, class FString __, class
UError* Error);
void __EpicLogin_X__RequestNintendoAccountAuthorization_0x2(class UError* _);
static bool __EpicLogin_X__IsAnyLocalPlayerCabinedOrGuest_0x1(class ULocalPlayer_X*
GamePlayer);
class FString BuildIdentityURL(class FString URLSuffix);
static bool IsAnyLocalPlayerCabinedOrGuest();
bool IsPrimaryPlayerXboxGuest();
class UError* ConvertError(class UEOS_ErrorResponse* ErrorResponse);
void ShowDOBScreen();
void HandleRequestCabinedMode(class UCabinedModeResponse* ResponseAsObject, class
UError* Error);
bool RequestCabinedMode();
void HandleRequestPinGrantTimeout();
void HandleLoginTimeout();
void ClearRequestPinGrantCallbacks();
void ClearLoginCallbacks();
void HandleReceivedPinGrantCode(uint8_t Result, uint8_t InLocalPlayerNum, class FString Code,
class FString URL, int32_t SecondsUntilExpiration);
void HandleCompletedPinGrant();
void RequestPinGrantCode();
void RequestNintendoAccountAuthorization(struct FScriptDelegate Callback);
bool Login(class FString PlatformAuthTicket);
void NintendoAccountLogin();
void HandleEpicAccountCreated();
void CreateEpicGamesAccount();
void UpdateTwoFactorAuthenticationStatus(struct FScriptDelegate Callback);
void RetryAuth();
void HandleConnectionStatusChanged(unsigned long bConnected);
void HandleKickedByEpic();
void HandleAuthTicketExpired();
void HandleUnderageUserDetected(uint8_t InLocalPlayerNum, class FString
InParentalConsentURL);
void HandleLoginResult(class UOnlinePlayerAuthentication_X* Auth);
void HandleLoginFailed(uint8_t InLocalPlayerNum, uint8_t Error);
void UpdateAgeGateModalScreen();
void HandleLoginChanged(uint8_t InLocalPlayerNum);
void ClearLoginError();
void SetLoginError(class UErrorType* Error);
void TriggerAuthTicketDelegate(struct FScriptDelegate Callback);
void ClearReceivedAuthTicketDelegate(struct FScriptDelegate Callback);
bool RequestEpicAuthTicket(struct FScriptDelegate Callback);
bool RequiresEpicAuthTicket();
bool IsLoggedIn();
void UnsubscribeFromInitialized(struct FScriptDelegate Callback);
void NotifyWhenInitialized(struct FScriptDelegate Callback);
void OnRemoved();
void HandleEOSInitialized();
void HandleEpicConfigSet();
void HandlePlatformAuthTicketFailedChanged_Switch(unsigned long bNewValue);
void NotifyWhenLoggedIn(struct FScriptDelegate Callback);

```

```

void eventConstruct();
void EventCreatedEpicGamesAccount();
void EventInitialized();
void EventReceivedEpicAuthTicket(class FString AuthTicket, class FString AccountId, class
UErrors* Error);
void EventLoginError(class UEpicLogin_X* EpicLogin, class UErrorType* Error);
void EventReceivedPinGrantCode();
void EventLoginFailed();
void EventLoginSucceeded();
};

// Class ProjectX.EpochTimers_X
// 0x0000 (0x0060 - 0x0060)
class UEpochTimers_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EpochTimers_X");
}

return uClassPointer;
};

static bool IsActive(struct FScriptDelegate Callback);
static void ClearAll(class UObject* Obj);
static void Clear(struct FScriptDelegate Callback, struct FScriptDelegate Callback2);
static void SetWindow(struct FScriptDelegate StartCallback, uint64_t EpochStartTime, struct
FScriptDelegate EndCallback, uint64_t EpochEndTime);
static void SetTime(struct FScriptDelegate Callback, uint64_t EpochTime);
static void Set(struct FScriptDelegate Callback, uint64_t DelaySeconds);
};

// Class ProjectX.EpochTimerTick_X
// 0x0018 (0x0060 - 0x0078)
class UEpochTimerTick_X : public UObject
{
public:
struct FPointer VfTable_FTickableObject; // 0x0060 (0x0008)
[0x000000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
TArray<struct FEpochTimer> Timers; // 0x0068 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.EpochTimerTick_X");
}

return uClassPointer;
};

};

// Class ProjectX.Errors_X
// 0x0330 (0x0080 - 0x03B0)
class UErrors_X : public UErrorList
{
public:
    class UErrorType*                UnknownError;                // 0x0080 (0x0008)
    [0x0000000000000002] (CPF_Const)
    class UPsyNetErrorType_X*         RequestError;                // 0x0088 (0x0008)
    [0x0000000000000002] (CPF_Const)
    class UErrorType*                HTTPError;                    // 0x0090 (0x0008)
    [0x0000000000000002] (CPF_Const)
    class UErrorType*                SessionNotActive;              // 0x0098 (0x0008)
    [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_NotConnected;              // 0x00A0 (0x0008)
    [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_Connected;                // 0x00A8 (0x0008)
    [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_ConnectionDropped;        // 0x00B0 (0x0008)
    [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_NoNetworkConnection;      // 0x00B8
    (0x0008) [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_ServiceUnavailable;        // 0x00C0 (0x0008)
    [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_UpdateRequired;            // 0x00C8 (0x0008)
    [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_ServersTooBusy;            // 0x00D0 (0x0008)
    [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_DuplicateLoginDetected;    // 0x00D8
    (0x0008) [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_InvalidUser;              // 0x00E0 (0x0008)
    [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_InvalidResponse;          // 0x00E8 (0x0008)
    [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_EpicAccountRequired;        // 0x00F0
    (0x0008) [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_EpicAccountLinkingFailed;  // 0x00F8
    (0x0008) [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_EpicDOBRequired;          // 0x0100 (0x0008)
    [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_EpicParentEmailRequired;   // 0x0108
    (0x0008) [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_EpicConfirmDisplayNameRequired; // 0x0110
    (0x0008) [0x0000000000000002] (CPF_Const)
    class UErrorType*                OSCS_EpicUnsupportedCorrectiveActionRequired; //

```

```

0x0118 (0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType*      ConnectionStatusErrors[0x11];          // 0x0120 (0x0088)
[0x0000000000000002] (CPF_Const)
class UErrorType*      VersionMismatch;                      // 0x01A8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      NoServers;                            // 0x01B0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      FileDoesNotExist;                     // 0x01B8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      FileLoadFail;                         // 0x01C0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      FileLoadCorrupt;                      // 0x01C8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      FileMountCorrupt;                     // 0x01D0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      FileSaveFail;                         // 0x01D8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      FileSaveNoSpace;                      // 0x01E0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      FileSaveCorrupt;                      // 0x01E8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      UserAuthBanned;                       // 0x01F0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      UserSocialBanned;                     // 0x01F8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      NotLoggedInToPsynet;                  // 0x0200 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      FriendsList;                           // 0x0208 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      InviteLimitReached;                   // 0x0210 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      OutOfRetries;                          // 0x0218 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      AuthenticationFailed;                  // 0x0220 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      NoExportCountry;                       // 0x0228 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      UnauthorizedAccess;                    // 0x0230 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      ServiceNotFound;                       // 0x0238 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      FeatureDisabled;                       // 0x0240 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      OtherPlayerFeatureDisabled;           // 0x0248 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      MatchmakingDisabled;                   // 0x0250 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      NoValidPlaylistsSelected;              // 0x0258 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      PlaylistNotAvailable;                  // 0x0260 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      PlayerNameNotSet;                      // 0x0268 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*      NotAuthorized;                         // 0x0270 (0x0008)

```

[0x0000000000000002] (CPF_Const)		
class UErrorType*	GameServerPending;	// 0x0278 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	CacheError;	// 0x0280 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	WordFilterEvil;	// 0x0288 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	WordFilterPlatformError;	// 0x0290 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	WordFilterPlatformChatError;	// 0x0298 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	InvalidSettings;	// 0x02A0 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	Maintenance;	// 0x02A8 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	PlatformAuthError;	// 0x02B0 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	MicrotransactionAuthExpired;	// 0x02B8 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	CannotFriendSelf;	// 0x02C0 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	CannotBlockSelf;	// 0x02C8 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	InvalidPlayer;	// 0x02D0 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	FriendRequestNotFound;	// 0x02D8 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	FriendRequestFailed;	// 0x02E0 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	AlreadyFriends;	// 0x02E8 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	FriendLimitReached;	// 0x02F0 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	LocalFriendLimitReached;	// 0x02F8 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	RemoteFriendLimitReached;	// 0x0300 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	LocalHostConnectionError;	// 0x0308 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	UGCRestricted;	// 0x0310 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	MicrotransactionServiceMaintenance;	// 0x0318
(0x0008) [0x0000000000000002] (CPF_Const)		
class UErrorType*	GenericStorageOutOfSync;	// 0x0320 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	GenericStorageSync_RequestNotFound;	// 0x0328
(0x0008) [0x0000000000000002] (CPF_Const)		
class UErrorType*	SignatureMismatch;	// 0x0330 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	JsonDeserialization;	// 0x0338 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	InvalidPlatform;	// 0x0340 (0x0008)
[0x0000000000000002] (CPF_Const)		
class UErrorType*	InvalidPlatformForCodeRedemption;	// 0x0348

```

(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* InvalidParameters; // 0x0350 (0x0008)
[0x0000000000000002] (CPF_Const)
class UPsyNetErrorType_X* NotMatchmaking; // 0x0358
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* AccessDenied; // 0x0360 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* CallLimitReached; // 0x0368 (0x0008)
[0x0000000000000002] (CPF_Const)
class UPsyNetErrorType_X* PerConDisconnected; // 0x0370
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* AccountNotFound; // 0x0378 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* ServerNotFound; // 0x0380 (0x0008)
[0x0000000000000002] (CPF_Const)
class UPsyNetErrorType_X* ExpiredDsConnectToken; // 0x0388
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* MatchmakingNoInternet; // 0x0390 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* PartyRankDisparity; // 0x0398 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* FriendAddsThrottledGeneric; // 0x03A0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* PlayerLevelNotReached; // 0x03A8 (0x0008)
[0x0000000000000002] (CPF_Const)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Errors_X");
}

return uClassPointer;
};

```

```

static bool DisplayPlatformError(class UErrorType* InErrorType);
};

```

```

// Class ProjectX.EventRecorder_X
// 0x0060 (0x0070 - 0x00D0)
class UEventRecorder_X : public UComponent
{
public:
struct FPointer VfTable_FTickableObject; // 0x0070 (0x0008)
[0x0000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
class UEventRecorderConfig_X* Config; // 0x0078 (0x0008)
[0x0000800000000000]
int32_t MaxQueuedEvents; // 0x0080 (0x0004)
[0x0000000000000001] (CPF_Edit)
int32_t MaxQueuedMinutes; // 0x0084 (0x0004)

```



```

[0x0000000000000001] (CPF_Edit)
unsigned long          bPaused : 1;                // 0x0088 (0x0004)
[0x0000000000000000] [0x00000001]
TArray<struct FMetricsEvent>      QueuedEvents;      // 0x0090 (0x0010)
[0x000000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
struct FGuid              AppSessionID;             // 0x00A0 (0x0010)
[0x00000004000002002] (CPF_Const | CPF_Transient)
struct FGuid              LevelSessionID;           // 0x00B0 (0x0010)
[0x00000004000002002] (CPF_Const | CPF_Transient)
float                    NextSendTime;              // 0x00C0 (0x0004)
[0x00000000000002002] (CPF_Const | CPF_Transient)
float                    FloodPreventionTimeSeconds; // 0x00C4 (0x0004)
[0x0000000000000001] (CPF_Edit)
float                    LastFullSendTime;          // 0x00C8 (0x0004)
[0x00000000000002002] (CPF_Const | CPF_Transient)
int32_t                  AppEventCount;             // 0x00CC (0x0004)
[0x00000000000002002] (CPF_Const | CPF_Transient)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EventRecorder_X");
}

return uClassPointer;
};

```

```

void eventMetricsNotSent(int32_t Count);
void eventMetricsFlood(class FString LastEventName);
void SetLevelSessionID(struct FGuid Id);
bool Send();
void RecordEvent(class FString Category, struct FName Event, struct FUniqueNetId PlayerID,
int32_t Version, class FString& Data);
static void RecordFunction(class FString Category, int32_t Version);
};

```

```

// Class ProjectX.MetricsSystem_X
// 0x0000 (0x00D0 - 0x00D0)
class UMetricsSystem_X : public UEventRecorder_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.MetricsSystem_X");
}
}

```

```

}

return uClassPointer;
};

void eventConstruct();
static void HandleUnpause(class UNetMode_Networked* NetMode, class UMetricsSystem_X*
Metrics, class UPauseMetricsState_X* State);
static void HandlePause(class UNetMode_Networked* NetMode, class UMetricsSystem_X*
Metrics, class UPauseMetricsState_X* State);
};

// Class ProjectX.EventRecorderGroup_X
// 0x0010 (0x0070 - 0x0080)
class UEventRecorderGroup_X : public UComponent
{
public:
class FString                                Category;                                // 0x0070 (0x0010)
[0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EventRecorderGroup_X");
}

return uClassPointer;
};

class FString VectorToString(struct FVector V);
static void RecordFunction(int32_t Version);
class UEventRecorder_X* GetEventRecorder();
};

// Class ProjectX.MetricsGroup_X
// 0x0000 (0x0080 - 0x0080)
class UMetricsGroup_X : public UEventRecorderGroup_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.MetricsGroup_X");
}
}

```

```

return uClassPointer;
};

class UEventRecorder_X* GetEventRecorder();
};

// Class ProjectX.NetMetrics_X
// 0x0018 (0x0080 - 0x0098)
class UNetMetrics_X : public UMetricsGroup_X
{
public:
class UEventRecorderConfig_X*          Config;          // 0x0080 (0x0008)
[0x0000800000000000]
TArray<class FString>                  UnstableConnectionsQueue;          // 0x0088
(0x0010) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.NetMetrics_X");
}

return uClassPointer;
};

static uint8_t GetConnectionType();
void UnstableConnections(TArray<class FString>& Addresses);
void RecordUnstableConnections();
void AddUnstableConnection(class FString& Address);
static void eventStaticUnstableConnection(class FString& Address);
void ReliablePacketsBlocked();
void UseInputBuffer(struct FUniqueNetId PlayerID, struct FName Type);
void LaunchURL(class FString URL);
void ServerReported(struct FUniqueNetId PlayerID);
void RpcSignatureMismatch(int32_t ServiceID);
void ConnectionChangedIP(struct FUniqueNetId PlayerID);
void PlayerNetworkError(struct FUniqueNetId PlayerID, class FString Reason);
void PlayerTimeout(struct FUniqueNetId PlayerID);
void NetStats(int32_t PlaylistId, class FString MatchGuid, int32_t ConnectionType, int32_t
PeriodMS, int32_t PingMin, int32_t PingMax, int32_t PingAvg, int32_t PingMed, int32_t
OutPackets, int32_t InPackets, int32_t OutLost, int32_t InLost, int32_t OutOfOrder, int32_t
OutBytes, int32_t InBytes, int32_t PsyPingMin, int32_t PsyPingMax, int32_t PsyPingAvg, int32_t
PsyPingMed, int32_t PsyReceived, int32_t PsyLost);
};

// Class ProjectX.ExplosionCollisionShapes_X
// 0x0014 (0x0070 - 0x0084)
class UExplosionCollisionShapes_X : public UComponent
{
public:

```

```

struct FVector                Location;                // 0x0070 (0x000C)
[0x0000000000000000]
float                        GrowthAmount;            // 0x007C (0x0004)
[0x0000000000000000]
float                        NormalizedLifetime;      // 0x0080 (0x0004)
[0x0000000000000000]

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ExplosionCollisionShapes_X");
}

return uClassPointer;
};

```

```

void Destroy();
void Init();
};

```

```

// Class ProjectX.BoxCollisionShape_X
// 0x001C (0x0084 - 0x00A0)
class UBoxCollisionShape_X : public UExplosionCollisionShapes_X
{
public:
struct FVector                StartSize;                // 0x0088 (0x000C)
[0x0000000000000001] (CPF_Edit)
struct FVector                EndSize;                // 0x0094 (0x000C)
[0x0000000000000001] (CPF_Edit)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.BoxCollisionShape_X");
}

return uClassPointer;
};

```

```

struct FVector GetCurrentSize();
struct FBox GetCurrentBox();
};

```

```

// Class ProjectX.GoalCollisionShape_X
// 0x0004 (0x0084 - 0x0088)
class UGoalCollisionShape_X : public UExplosionCollisionShapes_X

```

```

{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GoalCollisionShape_X");
}

return uClassPointer;
};

};

// Class ProjectX.SphereCollisionShape_X
// 0x000C (0x0084 - 0x0090)
class USphereCollisionShape_X : public UExplosionCollisionShapes_X
{
public:
float                               StartRadius;                // 0x0088 (0x0004)
[0x00000000000000000000000000000000] (CPF_Edit)
float                               EndRadius;                  // 0x008C (0x0004)
[0x00000000000000000000000000000000] (CPF_Edit)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.SphereCollisionShape_X");
}

return uClassPointer;
};

float GetCurrentRadius();
};

// Class ProjectX.FakeData_X
// 0x0320 (0x0060 - 0x0380)
class UFakeData_X : public UObject
{
public:
int32_t                             RandomSeed;                // 0x0060 (0x0004)
[0x00000000400000000000000000000000]
struct FFakeData1                    Data1;                    // 0x0068 (0x0038)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FFakeData2                    Data2;                    // 0x00A0 (0x0090)

```

```

[0x000000000000400000] (CPF_NeedCtorLink)
struct FFakeData3          Data3;          // 0x0130 (0x0130)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FFakeData4          Data4;          // 0x0260 (0x0120)
[0x000000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.FakeData_X");
}

return uClassPointer;
};

```

```

static int32_t RandomizeValues(class UObject* Target, int32_t InRandomSeed);
static class UFakeData_X* RandomizeFakeData(class UFakeData_X* FakeData, int32_t
InRandomSeed);
static class UFakeData_X* GenerateFakeData(int32_t InRandomSeed);
};

```

```

// Class ProjectX.FuncTestCommandPair_X
// 0x0018 (0x0060 - 0x0078)
class UFuncTestCommandPair_X : public UObject
{
public:
struct FName                Device;          // 0x0060 (0x0008)
[0x000000000000000000]
class FString               Command;         // 0x0068 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.FuncTestCommandPair_X");
}

return uClassPointer;
};

};

```

```

// Class ProjectX.FuncTestDesc_X
// 0x0030 (0x0060 - 0x0090)
class UFuncTestDesc_X : public UObject
{

```

```

public:
class FString                                     TestName;                                // 0x0060 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
TArray<struct FName>                               Tags;                                // 0x0070 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
TArray<class UFuncTestGroup_X*>                   Groups;                                // 0x0080 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.FuncTestDesc_X");
}

return uClassPointer;
};

};

```

```

// Class ProjectX.FuncTestGroup_X
// 0x0010 (0x0060 - 0x0070)
class UFuncTestGroup_X : public UObject
{
public:
TArray<class UFuncTestCommandPair_X*>             Commands;                                // 0x0060
(0x0010) [0x0000000000040000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.FuncTestGroup_X");
}

return uClassPointer;
};

};

```

```

// Class ProjectX.FXActorEvent_X
// 0x0000 (0x0060 - 0x0060)
class UFXActorEvent_X : public UObject
{
public:

```

```

public:
static UClass* StaticClass()

```

```

{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.FXActorEvent_X");
}

return uClassPointer;
};

};

// Class ProjectX.FXAttachmentTraitBase_X
// 0x0000 (0x0060 - 0x0060)
class UFXAttachmentTraitBase_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.FXAttachmentTraitBase_X");
}

return uClassPointer;
};

class UActorComponent* GetComponent();
void eventInit();
};

// Class ProjectX.GameEngine_X
// 0x0000 (0x0B48 - 0x0B48)
class UGameEngine_X : public UGameEngine
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GameEngine_X");
}

return uClassPointer;
};

```



```

void __GameEngine_X__Construct_0x1(class FString InURL);
void PrintDebugInfo(class UDebugDrawer* Drawer);
void eventOnlineSubsystemInitialized();
void eventConstruct();
};

// Class ProjectX.GameInfo_X
// 0x0058 (0x0470 - 0x04C8)
class AGameInfo_X : public AGameInfo
{
public:
class AGameReplicationInfo*          GameReplicationInfoArchetype;          // 0x0470
(0x0008) [0x0000000000000001] (CPF_Edit)
class APlayerController*             PlayerControllerArchetype;             // 0x0478
(0x0008) [0x0000000000000001] (CPF_Edit)
class APawn*                         PawnArchetype;                         // 0x0480 (0x0008)
[0x0000000000000001] (CPF_Edit)
TArray<struct FPauserData>            PCPausers;                            // 0x0488 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FScriptDelegate               __EventSpawned__Delegate;              // 0x0498
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate               __EventInit__Delegate;                // 0x04B0 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GameInfo_X");
}

return uClassPointer;
};

void PrintDebugInfo(class UDebugDrawer* Drawer);
void eventClientMapLoadFail(struct FUniqueNetId PlayerID, class FString MapName);
void RemovePauser(class APlayerController* PC, unsigned long bForceRemove);
void AddPauser(class APlayerController* PC);
void SetPaused(class APlayerController* PC, unsigned long bPause);
void TogglePause(class APlayerController* PC);
bool IsPaused(class APlayerController* PC);
class UOnlineGameHost_X* GetOnlineGameHost();
static class UOnlineGameDedicatedServer_X* GetOnlineGameDedicatedServer();
static class UOnlineGame_Base_X* GetOnlineGame();
void EndOnlineGame();
void Logout(class AController* Exiting);
void GenericPlayerInitialization(class AController* C);
static void DisconnectExistingPlayer(struct FUniqueNetId& PlayerID);
class APlayerController* eventLogin(class FString Portal, class FString Options, struct
FUniqueNetId UniqueId, class FString& ErrorMessage);

```

```

void eventPreLoginSplitscreen(struct FUniqueNetId PrimaryPlayerUniqueId, class FString
Options, class FString Address, struct FUniqueNetId UniqueId, unsigned long bSupportsAuth,
class FString& ErrorMessage);
void eventPreLogin(class FString Options, class FString Address, struct FUniqueNetId UniqueId,
unsigned long bSupportsAuth, class FString& ErrorMessage);
void UpdateGameSettingsCounts();
void PostBeginPlay();
bool ProcessServerLogin();
void RegisterServer();
class APawn* SpawnDefaultPawnFor(class AController* NewPlayer, class ANavigationPoint*
StartSpot);
void eventPreBeginPlay();
class APlayerController* SpawnPlayerController(struct FVector SpawnLocation, struct FRotator
SpawnRotation);
static class UClass* eventSetGameType(class FString MapName, class FString Options, class
FString Portal);
static class FString eventGetDefaultGameClassPath(class FString MapName, class FString
Options, class FString Portal);
void PDI(class UClass* ActorClass, unsigned long bOnlybDebug);
void eventInitGame(class FString Options, class FString& ErrorMessage);
void EventInit(class AGameInfo_X* G, class FString Options);
void EventSpawned(class AGameInfo_X* G);
};

// Class ProjectX.GameSetting_X
// 0x0004 (0x0060 - 0x0064)
class UGameSetting_X : public UObject
{
public:
    unsigned long                bHidden : 1;                // 0x0060 (0x0004)
    [0x0000000000000001] [0x00000001] (CPF_Edit)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.GameSetting_X");
        }

        return uClassPointer;
    };

    bool __GameSetting_X__IsSettingHidden_0x1(struct FGameSettingHidingOverride SO);
    void PrintDebugInfo(class UDebugDrawer* Drawer);
    bool IsSettingHidden();
    void SetHidden(unsigned long bNewHidden);
    bool IsSetting(struct FName SettingName);
    void AddGameSettingToList(class UOnlineGameSettings_X* OnlineGameSettings, TArray<class
UGameSetting_X*>& GameSettingsArray);
};

```

```

// Class ProjectX.GameSettingCategory_X
// 0x0028 (0x0060 - 0x0088)
class UGameSettingCategory_X : public UObject
{
public:
TArray<struct FName> SelectedNames; // 0x0060 (0x0010)
[0x00000004000400001] (CPF_Edit | CPF_NeedCtorLink)
unsigned long bHidden : 1; // 0x0070 (0x0004)
[0x00000000000000003] [0x000000001] (CPF_Edit | CPF_Const)
unsigned long bUseableWhileHidden : 1; // 0x0070 (0x0004)
[0x00000000000000003] [0x000000002] (CPF_Edit | CPF_Const)
TArray<class UGameSetting_X*> GameSettings; // 0x0078
(0x0010) [0x00000004000400001] (CPF_Edit | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GameSettingCategory_X");
}

return uClassPointer;
};

bool __GameSettingCategory_X__GetGameSettingIndex_0x1(class UGameSetting_X* GS);
void PrintDebugInfo(class UDebugDrawer* Drawer);
int32_t GetGameSettingIndex(struct FName SettingName, unsigned long bIgnoreHidden);
bool eventHasSetting(struct FName SettingName);
void GetSelectedGameSettings(TArray<class UGameSetting_X*>& SelectedGameSettings);
class UGameSetting_X* GetFirstSelectedGameSetting();
void ClearSelected();
void AddSelectedSetting(struct FName NewSelectedName);
void SetGameSettings(TArray<class UGameSetting_X*> NewGameSettings);
};

// Class ProjectX.GameViewportClient_X
// 0x0080 (0x0248 - 0x02C8)
class UGameViewportClient_X : public UGameViewportClient
{
public:
unsigned long bHandCursor : 1; // 0x0248 (0x0004)
[0x00000000000002000] [0x000000001] (CPF_Transient)
unsigned long bMuteSoundOnFocusLost : 1; // 0x0248 (0x0004)
[0x00000000000002000] [0x000000002] (CPF_Transient)
unsigned long bIsFinishedLoadingSaveSettings : 1; // 0x0248
(0x0004) [0x00000000000002000] [0x000000004] (CPF_Transient)
int32_t MaxSplitScreenPlayers; // 0x024C (0x0004)
[0x00000000000004002] (CPF_Const | CPF_Config)
struct FScriptDelegate __EventLocalPlayerJoin__Delegate; // 0x0250
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventLocalPlayerLeave__Delegate; // 0x0268

```

```

(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventNotifyConnectionError__Delegate;    // 0x0280
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventPrimaryPlayerChange__Delegate;      // 0x0298
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventFocusChanged__Delegate;             // 0x02B0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GameViewportClient_X");
}

return uClassPointer;
};

bool HasFocus();
void OnPrimaryPlayerSwitch(class ULocalPlayer* OldPrimaryPlayer, class ULocalPlayer*
NewPrimaryPlayer);
bool ForceRemovePlayer(class ULocalPlayer* ExPlayer);
bool eventRemovePlayer(class ULocalPlayer* ExPlayer);
bool RemovePlayerById(int32_t ControllerId);
void eventSetHardwareMouseCursorVisibility(unsigned long bIsVisible);
void NotifyConnectionError(uint8_t MessageType, class FString Message, class FString Title);
void eventSetProgressMessage(uint8_t MessageType, class FString Message, class FString Title,
unsigned long bIgnoreFutureNetworkMessages);
class ULocalPlayer* eventCreatePlayer(int32_t ControllerId, unsigned long bSpawnActor, class
FString& OutError);
void NotifyPlayerRemoved(int32_t PlayerIndex, class ULocalPlayer* RemovedPlayer);
void NotifyPlayerAdded(int32_t PlayerIndex, class ULocalPlayer* AddedPlayer);
bool ShouldMuteAudioWhenUnfocused();
bool PlatformSupportsMuteOnFocusLostOption();
bool IsGameRecordingEnabled();
float GetDefaultSafeZone();
void FlashWindow();
void SaveCurrentWindowState();
void SaveWindowState(int32_t ResX, int32_t ResY, unsigned long bFullScreen, unsigned long
bBorderless, unsigned long bVsync);
bool GetAvailableRefreshRates(TArray<int32_t>& RefreshRates);
bool GetAvailableResolutions(TArray<struct FIntPoint>& Resolutions);
bool GetAvailableResolutionsStr(TArray<class FString>& Resolutions);
bool IsBorderlessViewport();
struct FIntPoint GetViewportSizePoint();
class FString GetViewportSizeStr();
static class UGameViewportClient_X* GetInstance();
void EventFocusChanged(unsigned long bInHasFocus);
void EventPrimaryPlayerChange(class ULocalPlayer* OldPrimaryPlayer, class ULocalPlayer*
NewPrimaryPlayer);
void EventNotifyConnectionError(uint8_t MessageType, class FString Title, class FString

```

```

Message);
void EventLocalPlayerLeave(class ULocalPlayer* OldPlayer);
void EventLocalPlayerJoin(class ULocalPlayer* NewPlayer);
};

// Class ProjectX.GeForceNow_X
// 0x0000 (0x0060 - 0x0060)
class UGeForceNow_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GeForceNow_X");
}

return uClassPointer;
};

static bool IsStreaming();
};

// Class ProjectX.GFxDataRow_X
// 0x0034 (0x0060 - 0x0094)
class UGFxDataRow_X : public UObject
{
public:
struct FName                                     TableName;                                     // 0x0060 (0x0008)
[0x00000000000000003] (CPF_Edit | CPF_Const)
struct FName                                     PrimaryKeyName;                               // 0x0068 (0x0008)
[0x00000000000000003] (CPF_Edit | CPF_Const)
class UClass*                                    ProxyClass;                                   // 0x0070 (0x0008)
[0x00000000000000003] (CPF_Edit | CPF_Const)
unsigned long                                     bLevelTransitionPersistent : 1;               // 0x0078 (0x0004)
[0x00000000000000001] [0x000000001] (CPF_Edit)
class UObject*                                    ProxyObject;                                   // 0x0080 (0x0008)
[0x000000000000002002] (CPF_Const | CPF_Transient)
class UGFxShell_X*                               Shell;                                         // 0x0088 (0x0008)
[0x000000000000002002] (CPF_Const | CPF_Transient)
int32_t                                           RowIndex;                                    // 0x0090 (0x0004)
[0x000000000000002002] (CPF_Const | CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{

```

```

uClassPointer = UObject::FindClass("Class ProjectX.GFxDataRow_X");
}

return uClassPointer;
};

class UGFxObject* FlashEventObject();
class FString FlashEventString();
float FlashEventFloat();
int32_t FlashEventInt();
void FlashEventVoid();
void SetProxy(class UObject* InProxyObject);
void eventOnRemoved();
void eventOnShellSet();
};

// Class ProjectX.GFxDataSingleton_X
// 0x0004 (0x0094 - 0x0098)
class UGFxDataSingleton_X : public UGFxDataRow_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GFxDataSingleton_X");
}

return uClassPointer;
};

};

// Class ProjectX.GFxShell_X
// 0x0098 (0x0098 - 0x0130)
class UGFxShell_X : public UGFxDataSingleton_X
{
public:
class UGFxMoviePlayer_X* MoviePlayerArchetype; // 0x0098
(0x0008) [0x0000000000000001] (CPF_Edit)
class ULocalPlayer_X* Player; // 0x00A0 (0x0008)
[0x00000000000002002] (CPF_Const | CPF_Transient)
unsigned long bGamePaused : 1; // 0x00A8 (0x0004)
[0x0000000040000000] [0x00000001] (CPF_EditInlineNotify)
unsigned long bWasFullscreen : 1; // 0x00A8 (0x0004)
[0x00000000000002002] [0x00000002] (CPF_Const | CPF_Transient)
float LeftX; // 0x00AC (0x0004)
[0x0000000040000000] (CPF_EditInlineNotify)
float LeftY; // 0x00B0 (0x0004)
[0x0000000040000000] (CPF_EditInlineNotify)

```

```

float                RightX;                                // 0x00B4 (0x0004)
[0x0000000040000000] (CPF_EditInlineNotify)
float                RightY;                                // 0x00B8 (0x0004)
[0x0000000040000000] (CPF_EditInlineNotify)
TArray<class UGfxMoviePlayer_X*>    Movies;                // 0x00C0 (0x0010)
[0x0000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
class UGfxDataStore_X*    DataStore;                    // 0x00D0 (0x0008)
[0x0000000000408000A] (CPF_Const | CPF_ExportObject | CPF_Component | CPF_EditInline)
uint8_t              InputType;                        // 0x00D8 (0x0001)
[0x00000000000002002] (CPF_Const | CPF_Transient)
uint8_t              OnlinePlatformType;                // 0x00D9 (0x0001)
[0x000000000040000000] (CPF_EditInlineNotify)
class UOnlineSubsystem*    OnlineSub;                    // 0x00E0 (0x0008)
[0x00008000000000001] (CPF_Edit)
struct FScriptDelegate    __EventInputCaptureChanged__Delegate;    // 0x00E8
(0x0018) [0x000000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate    __EventInputTypeChanged__Delegate;    // 0x0100
(0x0018) [0x000000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate    __EventReceivedInput__Delegate;    // 0x0118
(0x0018) [0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GFxShell_X");
}

return uClassPointer;
};

void PrintDebugInfo(class UDebugDrawer* Drawer);
void SetMouseKeyPressed(struct FName Key, unsigned long bPressed);
void HandleOnlineSubSet();
void SetAllowAnyPlayerInput(unsigned long bAllow);
void HandlePauseChanged();
void OnShowKeyboardCanceled();
void OnShowKeyboardComplete(class FString NewText);
void HandleShowKeyboardComplete(class FString NewText, unsigned long bCanceled);
void class FString PasteFromClipboard();
void CopyToClipboard(class FString ClipboardText);
void HideKeyboard();
bool ShowKeyboard(class FString TitleText, class FString DescriptionText, unsigned long
bIsPassword, class FString DefaultText, int32_t MaxLength);
void SetGamePaused(unsigned long bPaused);
void ExitToMainMenu();
void ExitGame();
void HandleMovieInputCaptureChanged(class UGfxMoviePlayer_X* MoviePlayer);
void TriggerDataCallbacks();
void eventTick(float DeltaTime);
void eventOnMovieClosed(class UGfxMoviePlayer_X* Movie);

```

```

void eventOnMovieStarted(class UGfxMoviePlayer_X* Movie);
void StopMovie(class UGfxMoviePlayer_X* Movie);
void StartMovie(class UGfxMoviePlayer_X* Movie);
void eventStop();
void eventOnStart();
void eventStart(class ULocalPlayer_X* InPlayer);
void EventReceivedInput(class UGfxShell_X* InShell);
void EventInputTypeChanged(class UGfxShell_X* InShell);
void EventInputCaptureChanged(class UGfxShell_X* InShell);
};

// Class ProjectX.GfxDataStore_X
// 0x0074 (0x0070 - 0x00E4)
class UGfxDataStore_X : public UComponent
{
public:
TArray<struct FGfxDataStoreTable>          Tables;                                // 0x0070 (0x0010)
[0x000000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
TArray<struct FGfxDirtyTable>              DirtyTables;                        // 0x0080 (0x0010)
[0x000000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
struct FMap_Mirror                        ObjectNameToTable;                  // 0x0090 (0x0050)
[0x00000000000003002] (CPF_Const | CPF_Native | CPF_Transient)
unsigned long                             bDebugGetValue : 1;                  // 0x00E0 (0x0004)
[0x00000000000000001] [0x00000001] (CPF_Edit)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GfxDataStore_X");
}

return uClassPointer;
};

void ExportFakeData(struct FName InTableName);
void PrintData(struct FName InTableName);
int32_t GetObjectRowW(class UGfxDataRow_X* O);
void AllObjectsOfType(class UClass* BaseClass, class UGfxDataRow_X*& OutObject);
void AllObjects(class UClass* ObjClass, class UGfxDataRow_X*& OutObject, int32_t& Row);
class UGfxDataRow_X* GetObjectByPrimaryKeyW(class UClass* ObjClass, class FString Value);
class UGfxDataRow_X* GetObjectW(class UClass* ObjClass, int32_t Row);
void RemoveObject(class UGfxDataRow_X* O);
void BindObject(class UGfxDataRow_X* Object, int32_t Row);
class UGfxDataRow_X* CreateObject(class UClass* ObjClass, int32_t Row, int32_t& PlacedRow);
void RemoveAllObjects(class UClass* ObjClass);
void SetObjectCount(class UClass* ObjClass, int32_t Count);
void SortTable(class UClass* ObjClass);
int32_t GetObjectCountW(class UClass* ObjClass);
void EmptyTables();
void EmptyTable(struct FName Table);

```



```

void RemoveRow(struct FName Table, int32_t Row);
void SetRowCount(struct FName Table, int32_t Count);
int32_t GetRowCount(struct FName Table);
struct FASValue GetValue(struct FName Table, int32_t Row, struct FName Column);
int32_t GetObjectIndexW(class UGfxDataRow_X* TargetObject);
void SetASValue(struct FName Table, int32_t Row, struct FName Column, struct FASValue&
Value);
void SetTextureValue(struct FName Table, int32_t Row, struct FName Column, class UTexture*
Value);
void SetStringValue(struct FName Table, int32_t Row, struct FName Column, class FString
Value);
void SetQWordValue(struct FName Table, int32_t Row, struct FName Column, uint64_t Value);
void SetFloatValue(struct FName Table, int32_t Row, struct FName Column, float Value);
void SetIntValue(struct FName Table, int32_t Row, struct FName Column, int32_t Value);
void SetBoolValue(struct FName Table, int32_t Row, struct FName Column, unsigned long Value);
void SetDirty(struct FName InTableName, int32_t RowNum, struct FName Value);
};

```

```

// Class ProjectX.GFxEngine_X
// 0x0078 (0x0128 - 0x01A0)
class UGfxEngine_X : public UGfxEngine
{
public:
class UGfxShell_X*                ShellArchetype;                // 0x0128 (0x0008)
[0x000000040000000000]
TArray<class UGfxShell_X*>         Shells;                        // 0x0130 (0x0010)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<struct FDirtyObject>       DirtyObjects;                  // 0x0140 (0x0010)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)
unsigned long                     bAnyShellHasInput : 1;         // 0x0150 (0x0004)
[0x00000004000002000] [0x000000001] (CPF_Transient)
uint8_t                           AvailableGamepadType;         // 0x0154 (0x0001)
[0x00000000000002002] (CPF_Const | CPF_Transient)
struct FScriptDelegate            __EventShellCreated__Delegate; // 0x0158
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate            __EventAvailableGamepadTypeChanged__Delegate; //
0x0170 (0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate            __EventGameSessionEnded__Delegate; // 0x0188
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GFxEngine_X");
}

return uClassPointer;
};

void PrintDebugInfo(class UDebugDrawer* Drawer);

```

```

void eventOnGameSessionEnded();
static void GetFlashKey(struct FName UnrealKey, int32_t& KeyCode, int32_t& MouseButton);
void UpdateHardwareMouseCursorVisibility();
void HandleShellInputTypeChanged(class UGfxShell_X* InShell);
void HandleGfxEnabledChanged(class UGameViewportClient* GVC);
void HandleShellInputCaptureChanged(class UGfxShell_X* InShell);
void FlushDirtyObjects();
void eventTick(float DeltaTime);
class UGfxShell_X* FindShell(class ULocalPlayer_X* ForPlayer);
void OnShellDestroyed(class UGfxShell_X* Shell);
void DestroyShell(class ULocalPlayer* ForPlayer);
class UGfxShell_X* eventCreateShell();
class UGfxShell_X* GetShell(class ULocalPlayer_X* ForPlayer);
void SetShellArchetype(class UGfxShell_X* InShellArchetype);
static class UGfxEngine_X* GetInstance(class UClass* Type);
void EventGameSessionEnded();
void EventAvailableGamepadTypeChanged(class UGfxEngine_X* Engine);
void EventShellCreated(class UGfxEngine_X* Engine, class UGfxShell_X* Shell);
};

// Class ProjectX.GfxMoviePlayer_X
// 0x0120 (0x0218 - 0x0338)
class UGfxMoviePlayer_X : public UGfxMoviePlayer
{
public:
class UGfxShell_X* Shell; // 0x0218 (0x0008)
[0x00000000000002002] (CPF_Const | CPF_Transient)
class UGfxDataStore_X* DataStore; // 0x0220 (0x0008)
[0x000000000408200A] (CPF_Const | CPF_ExportObject | CPF_Transient | CPF_Component |
CPF_EditInline)
TArray<class UGfxDataCallback_X*> DataCallbacks; // 0x0228
(0x0010) [0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
TArray<class UGfxObjectReference_X*> ActionScriptClasses; // 0x0238
(0x0010) [0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
struct FMap_Mirror ActionScriptClassMap; // 0x0248 (0x0050)
[0x00000000000003002] (CPF_Const | CPF_Native | CPF_Transient)
TArray<class FString> CursorScenes; // 0x0298 (0x0010)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<class FString> InputScenes; // 0x02A8 (0x0010)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<class FString> CaptureScenes; // 0x02B8 (0x0010)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FName DebugTableCallback; // 0x02C8 (0x0008)
[0x00000000000000001] (CPF_Edit)
struct FName DebugColumnCallback; // 0x02D0 (0x0008)
[0x00000000000000001] (CPF_Edit)
class UGfxObjectReference_X* ShellHooksClass; // 0x02D8
(0x0008) [0x00000000000002002] (CPF_Const | CPF_Transient)
TArray<class UGfxSoundPack_X*> LoadedSoundPacks; // 0x02E0
(0x0010) [0x00000000000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FScriptDelegate __EventStarted__Delegate; // 0x02F0 (0x0018)
[0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventClosed__Delegate; // 0x0308 (0x0018)
[0x00000000000400000] (CPF_NeedCtorLink)

```

```

struct FScriptDelegate          __EventInputCaptureChanged__Delegate;      // 0x0320
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GFxMoviePlayer_X");
}

return uClassPointer;
};

void PrintDebugInfo(class UDebugDrawer* Drawer);
void SetAllowAnyPlayerInput(unsigned long bAllow);
bool PopString(class FString Value, TArray<class FString>& Items);
void SetGlobalRTCP(struct FName Key, float Value);
void eventStopAllSounds(class UGFxSoundPack_X* SoundPack);
void StopSound(struct FName EventName);
void PlaySound(struct FName EventName);
void eventPlaySoundFromTheme(struct FName EventName, struct FName SoundThemeName);
class AActor* GetSoundSource();
void UpdateCursorVisibility();
void eventSetCaptureAllInput(unsigned long bNewCaptureInput, class FString SceneName);
void eventSetReceiveInput(unsigned long bNewReceiveInput, class FString SceneName);
void eventSetCursorVisible(unsigned long bVisible, class FString SceneName);
void InitShellHooks();
void InitDataHooks();
void eventOnClose();
bool eventStart(unsigned long StartPaused);
void EventInputCaptureChanged(class UGFxMoviePlayer_X* Player);
void EventClosed(class UGFxMoviePlayer_X* Player);
void EventStarted(class UGFxMoviePlayer_X* Player);
};

// Class ProjectX.GFxObjectReference_X
// 0x0030 (0x0060 - 0x0090)
class UGFxObjectReference_X : public UObject
{
public:
int32_t          Value[0xC];          // 0x0060 (0x0030)
[0x00000000000001002] (CPF_Const | CPF_Native)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GFxObjectReference_X");
}
}

```

```

}

return uClassPointer;
};

};

// Class ProjectX.GFxDataCallback_X
// 0x0015 (0x0090 - 0x00A5)
class UGFxDataCallback_X : public UGFxObjectReference_X
{
public:
    struct FName                Table;                                // 0x0090 (0x0008)
    [0x00000000000002000] (CPF_Transient)
    int32_t                    Row;                                    // 0x0098 (0x0004)
    [0x00000000000002000] (CPF_Transient)
    struct FName                Column;                                // 0x009C (0x0008)
    [0x00000000000002000] (CPF_Transient)
    uint8_t                    Type;                                    // 0x00A4 (0x0001)
    [0x00000000000002000] (CPF_Transient)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.GFxDataCallback_X");
        }

        return uClassPointer;
    };

};

// Class ProjectX.GFxSoundPack_X
// 0x0024 (0x0060 - 0x0084)
class UGFxSoundPack_X : public UObject
{
public:
    class FString                ExportClassName;                    // 0x0060 (0x0010)
    [0x00000000000400001] (CPF_Edit | CPF_NeedCtorLink)
    TArray<struct FSoundPackSoundRef> Sounds;                        // 0x0070
    (0x0010) [0x00000004000400003] (CPF_Edit | CPF_Const | CPF_NeedCtorLink)
    unsigned long                bHasSubtitles : 1;                  // 0x0080 (0x0004)
    [0x00010000000000001] [0x000000001] (CPF_Edit)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)

```

```

{
uClassPointer = UObject::FindClass("Class ProjectX.GFxSoundPack_X");
}

return uClassPointer;
};

bool ToggleSound(int32_t Idx, class AActor* Target, unsigned long bPlay);
void StopAll(class AActor* Target);
bool StopSound(struct FName SoundName, class AActor* Target);
bool PlaySound(struct FName SoundName, class AActor* Target);
};

// Class ProjectX.Hashes_X
// 0x0000 (0x0060 - 0x0060)
class UHashes_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Hashes_X");
}

return uClassPointer;
};

static int32_t HashInt(int32_t Value);
};

// Class ProjectX.IFuncTestManager_X
// 0x0000 (0x0060 - 0x0060)
class UFuncTestManager_X : public UInterface
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.IFuncTestManager_X");
}

return uClassPointer;
};

```

```

};

// Class ProjectX.InterpComponent_X
// 0x00DB (0x009D - 0x0178)
class UInterpComponent_X : public UActorComponent
{
public:
    struct FMatrix                InterpStart;                // 0x00A0 (0x0040)
    [0x00000000000002002] (CPF_Const | CPF_Transient)
    struct FMatrix                InterpEnd;                // 0x00E0 (0x0040)
    [0x00000000000002002] (CPF_Const | CPF_Transient)
    struct FMatrix                InterpLocalToWorld;        // 0x0120 (0x0040)
    [0x00000000000002002] (CPF_Const | CPF_Transient)
    float                        InterpStartTime;            // 0x0160 (0x0004)
    [0x00000000000002000] (CPF_Transient)
    float                        InterpEndTime;              // 0x0164 (0x0004)
    [0x00000000000002000] (CPF_Transient)
    TArray<struct FAttachment>    Attachments;                // 0x0168 (0x0010)
    [0x00000000000680002] (CPF_Const | CPF_Component | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.InterpComponent_X");
        }

        return uClassPointer;
    };

    void DetachComponent(class UActorComponent* Component);
    void AttachComponent(class UActorComponent* Component, struct FVector RelativeLocation,
        struct FRotator RelativeRotation, struct FVector RelativeScale);
};

// Class ProjectX.IReservationConnection_X
// 0x0000 (0x0060 - 0x0060)
class UIReservationConnection_X : public UInterface
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.IReservationConnection_X");
        }
    }

```

```

return uClassPointer;
};

};

// Class ProjectX.JsonTests_X
// 0x0000 (0x0060 - 0x0060)
class UJsonTests_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.JsonTests_X");
}

return uClassPointer;
};

static bool ToJsonAndBack(class UObject* Target);
static void RunJsonTests(int32_t RandomSeed);
};

// Class ProjectX.LanBeacon_X
// 0x0034 (0x0070 - 0x00A4)
class ULanBeacon_X : public UComponent
{
public:
struct FPointer VfTable_FTickableObject; // 0x0070 (0x0008)
[0x000000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
struct FPointer LanBeacon; // 0x0078 (0x0008)
[0x00000000000003002] (CPF_Const | CPF_Native | CPF_Transient)
int32_t LanAnnouncePort; // 0x0080 (0x0004)
[0x00000000000004002] (CPF_Const | CPF_Config)
uint64_t QueryNonce; // 0x0088 (0x0008)
[0x00000000000000002] (CPF_Const)
uint8_t BeaconState; // 0x0090 (0x0001)
[0x00000000000000002] (CPF_Const)
class UOnlineMessageComponent_X* MessageComponent; // 0x0098
(0x0008) [0x0000000000408000A] (CPF_Const | CPF_ExportObject | CPF_Component |
CPF_EditInline)
int32_t MaxPacketSize; // 0x00A0 (0x0004)
[0x00000000000000002] (CPF_Const)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.LanBeacon_X");
}

return uClassPointer;
};

bool BroadcastMessage(class UObject* Message);
bool BroadcastData(TArray<uint8_t>& Data);
void Stop();
bool Start(uint8_t InitialState);
};

// Class ProjectX.LensFlareComponent_X
// 0x0080 (0x02BC - 0x033C)
class ULensFlareComponent_X : public ULensFlareComponent
{
public:
    struct FPointer                VfTable_IISetParameter;                // 0x02C0 (0x0008)
    [0x000000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
    struct FRawDistributionFloat    AlphaOverTime;                        // 0x02C8 (0x0028)
    [0x000000000000480001] (CPF_Edit | CPF_Component | CPF_NeedCtorLink)
    struct FRawDistributionVector    ColorOverTime;                        // 0x02F0 (0x0028)
    [0x000000000000480001] (CPF_Edit | CPF_Component | CPF_NeedCtorLink)
    TArray<struct FLensFlareFloatParamCurve>    MaterialFloatParamValuesOverTime;    //
    0x0318 (0x0010) [0x000000000000480001] (CPF_Edit | CPF_Component | CPF_NeedCtorLink)
    struct FName                    SourceColorParamName;                // 0x0328 (0x0008)
    [0x000000000000000003] (CPF_Edit | CPF_Const)
    struct FName                    SourceColorAlphaParamName;            // 0x0330
    (0x0008) [0x000000000000000003] (CPF_Edit | CPF_Const)
    float                            AttachTime;                        // 0x0338 (0x0004)
    [0x000000000000002000] (CPF_Transient)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.LensFlareComponent_X");
        }

        return uClassPointer;
    };

    void SetActorParameter(struct FName Key, class AActor* Value);
    void SetLinearColorParameter(struct FName Key, struct FLinearColor Value);
    void SetVectorParameter(struct FName Key, struct FVector Value);
    void SetFloatParameter(struct FName Key, float Value);
    void SetNameParameter(struct FName Key, struct FName Value);
};

```



```

// Class ProjectX.LinkedAccountMap_X
// 0x0050 (0x0070 - 0x00C0)
class ULinkedAccountMap_X : public UComponent
{
public:
    struct FMap_Mirror                PlayerMap;                // 0x0070 (0x0050)
    [0x00000000000001000] (CPF_Native)

    public:
    static UClass* StaticClass()
    {
    static UClass* uClassPointer = nullptr;

    if (!uClassPointer)
    {
    uClassPointer = UObject::FindClass("Class ProjectX.LinkedAccountMap_X");
    }

    return uClassPointer;
    };

    bool Contains(struct FUniqueNetId& Key);
    void Clear(int32_t ExpectedElements);
    void Remove(struct FUniqueNetId& Key);
    bool TryGet(struct FUniqueNetId& Key, struct FUniqueNetId& OutValue);
    struct FUniqueNetId Get(struct FUniqueNetId& Key);
    void Set(struct FUniqueNetId& Key, struct FUniqueNetId& Value);
    int32_t Count();
    void AllValues(int32_t StartIndex, int32_t MaxValues, struct FUniqueNetId& OutFriend);
};

// Class ProjectX.LocalCache_X
// 0x0080 (0x0060 - 0x00E0)
class ULocalCache_X : public UObject
{
public:
    struct FPointer                VfTable_FTickableObject;                // 0x0060 (0x0008)
    [0x000000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
    TArray<struct FPointer>                ImportTasks;                // 0x0068 (0x0010)
    [0x00000000000103000] (CPF_Native | CPF_Transient)
    TArray<struct FCacheImportCallbackData>                ImportCallbacks;                // 0x0078
    (0x0010) [0x00000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FPointer>                ExportTasks;                // 0x0088 (0x0010)
    [0x00000000000103000] (CPF_Native | CPF_Transient)
    TArray<struct FCacheExportCallbackData>                ExportCallbacks;                // 0x0098
    (0x0010) [0x00000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    unsigned long                bDebug : 1;                // 0x00A8 (0x0004)
    [0x0000000000000001] [0x00000001] (CPF_Edit)
    struct FScriptDelegate                __EventImportFinished__Delegate;                // 0x00B0
    (0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate                __EventExportFinished__Delegate;                // 0x00C8
    (0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

    public:

```

```

static UClass* StaticClass()
{
    static UClass* uClassPointer = nullptr;

    if (!uClassPointer)
    {
        uClassPointer = UObject::FindClass("Class ProjectX.LocalCache_X");
    }

    return uClassPointer;
};

void Delete(class FString Path);
uint8_t Flush(float TimeoutSeconds);
bool HasActiveTasks();
void ExportObjectAsync(class UObject* CacheObject, class FString Path, struct FScriptDelegate
    Callback);
void ImportObjectAsync(class UObject* CacheObject, class FString Path, struct FScriptDelegate
    Callback);
class UError* ImportObject(class UObject* CacheObject, class FString Path);
void EventExportFinished(class ULocalCache_X* Cache, class UObject* CacheObject, class
    UError* Error);
void EventImportFinished(class ULocalCache_X* Cache, class UObject* CacheObject, class
    UError* Error);
};

// Class ProjectX.MapFlythroughProfiler_X
// 0x0058 (0x0268 - 0x02C0)
class AMapFlythroughProfiler_X : public AActor
{
public:
    float Accel; // 0x0268 (0x0004)
    [0x0000000000000001] (CPF_Edit)
    float TurnRate; // 0x026C (0x0004)
    [0x0000000000000001] (CPF_Edit)
    float ProfileTime; // 0x0270 (0x0004)
    [0x0000000000000001] (CPF_Edit)
    float FinishTime; // 0x0274 (0x0004)
    [0x0000000000000200] (CPF_Transient)
    float Speed; // 0x0278 (0x0004)
    [0x0000000000000200] (CPF_Transient)
    struct FRenderProfile Profile; // 0x027C (0x0028)
    [0x0000000000000200] (CPF_Transient)
    struct FScriptDelegate __EventFinished__Delegate; // 0x02A8
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.MapFlythroughProfiler_X");

```

```

}

return uClassPointer;
};

void SetupPlayers();
void eventPostBeginPlay();
void EventFinished(class AMapFlythroughProfiler_X* Profiler);
};

// Class ProjectX.NNXErrors_X
// 0x0008 (0x0080 - 0x0088)
class UNNXErrors_X : public UErrorList
{
public:
class UErrorType*                               SwitchServiceMaintenance;           // 0x0080 (0x0008)
[0x00000000000000002] (CPF_Const)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.NNXErrors_X");
}

return uClassPointer;
};

};

// Class ProjectX.OnlineConfig_X
// 0x0018 (0x0060 - 0x0078)
class UOnlineConfig_X : public UObject
{
public:
unsigned long                                     bAllowOnServer : 1;                               // 0x0060 (0x0004)
[0x00000000000000001] [0x000000001] (CPF_Edit)
TArray<struct FModifierSubscription>              AllSubscriptions;                                  // 0x0068
(0x0010) [0x00000000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineConfig_X");
}

return uClassPointer;
};

```

```

};

void UnsubscribeModifiers();
void ModifyObjects(class UClass* ObjClass, struct FScriptDelegate ApplyCallback, struct
FScriptDelegate ResetCallback);
void Undo();
void Apply();
};

// Class ProjectX.BeaconConfig_X
// 0x0018 (0x0078 - 0x0090)
class UBeaconConfig_X : public UOnlineConfig_X
{
public:
    unsigned long                bUdpPingMetrics : 1;                // 0x0078 (0x0004)
    [0x00000000000004000] [0x00000001] (CPF_Config)
    int32_t                      MaxPingsPerAddress;                // 0x007C (0x0004)
    [0x00000000000004000] (CPF_Config)
    int32_t                      MaxPingsWindowSeconds;            // 0x0080 (0x0004)
    [0x00000000000004000] (CPF_Config)
    int32_t                      StatsLogDelaySeconds;            // 0x0084 (0x0004)
    [0x00000000000004000] (CPF_Config)
    float                      WaitForPongSeconds;                // 0x0088 (0x0004)
    [0x00000000000004000] (CPF_Config)
    int32_t                      PingsPerSecond;                // 0x008C (0x0004)
    [0x00000000000004000] (CPF_Config)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.BeaconConfig_X");
        }

        return uClassPointer;
    };
};

// Class ProjectX.ClassPropertyConfig_X
// 0x0010 (0x0078 - 0x0088)
class UClassPropertyConfig_X : public UOnlineConfig_X
{
public:
    TArray<struct FPropertyOverride> Overrides;                // 0x0078 (0x0010)
    [0x00000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.ClassPropertyConfig_X");
}

return uClassPointer;
};

void Apply();
};

// Class ProjectX.ContentConfig_X
// 0x0010 (0x0078 - 0x0088)
class UContentConfig_X : public UOnlineConfig_X
{
public:
    TArray<struct FContentPair>          ContentMap;                // 0x0078 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.ContentConfig_X");
        }

        return uClassPointer;
    };

    void Undo();
    void Apply();
    static bool IsEncryptionKeySetAtIndex(struct FEncryptedKeyIndex KeyIndex);
    static bool EncryptedKeyIndexIsValid(struct FEncryptedKeyIndex KeyIndex);
    static struct FEncryptedKeyIndex FindKeyIndex(struct FName ContentKeyName);
    void UndoNative();
    void ApplyNative();
};

// Class ProjectX.CrossplayConfig_X
// 0x0020 (0x0078 - 0x0098)
class UCrossplayConfig_X : public UOnlineConfig_X
{
public:
    TArray<struct FCrossplayGroup>      Groups;                    // 0x0078 (0x0010)
    [0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)
    TArray<struct FCrossplayGroup>      DisabledCrossplayGroups;  // 0x0088
    (0x0010) [0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()

```

```

{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CrossplayConfig_X");
}

return uClassPointer;
};

bool CanInteractWithCrossplayDisabled(uint8_t Platform1, uint8_t Platform2);
TArray<uint8_t> GetDisabledCrossplayGroup(uint8_t PlayerPlatform);
bool PlatformsShareGroup(uint8_t Platform1, uint8_t Platform2);
};

// Class ProjectX.EventRecorderConfig_X
// 0x002C (0x0078 - 0x00A4)
class UEventRecorderConfig_X : public UOnlineConfig_X
{
public:
unsigned long bEnabled : 1; // 0x0078 (0x0004)
[0x0000000000000001] [0x00000001] (CPF_Edit)
unsigned long bSeparatePlayerIDs : 1; // 0x0078 (0x0004)
[0x0000000000000001] [0x00000002] (CPF_Edit)
unsigned long bExcludePlayerIDs : 1; // 0x0078 (0x0004)
[0x0000000000000001] [0x00000004] (CPF_Edit)
TArray<struct FName> DisabledEvents; // 0x0080 (0x0010)
[0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
class UClass* EventRecorderClass; // 0x0090 (0x0008)
[0x0000000000000001] (CPF_Edit)
float PlayerNetMetricsPeriod; // 0x0098 (0x0004)
[0x0000000000000001] (CPF_Edit)
int32_t MaxUnstableConnections; // 0x009C (0x0004)
[0x0000000000000001] (CPF_Edit)
int32_t UnstableConnectionsTimePeriodSeconds; // 0x00A0
(0x0004) [0x0000000000000001] (CPF_Edit)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EventRecorderConfig_X");
}

return uClassPointer;
};

};

// Class ProjectX.LocalizationConfig_X

```

```

// 0x0010 (0x0078 - 0x0088)
class ULocalizationConfig_X : public UOnlineConfig_X
{
public:
TArray<struct FLocOverride>                Overrides;                // 0x0078 (0x0010)
[0x00000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.LocalizationConfig_X");
}

return uClassPointer;
};

void Apply();
};

// Class ProjectX.PsyNetConfig_X
// 0x0028 (0x0078 - 0x00A0)
class UPsyNetConfig_X : public UOnlineConfig_X
{
public:
unsigned long                bAllowPerCon : 1;                // 0x0078 (0x0004)
[0x0000000000004001] [0x00000001] (CPF_Edit | CPF_Config)
unsigned long                bRequiresPerCon : 1;                // 0x0078 (0x0004)
[0x0000000000004000] [0x00000002] (CPF_Config)
unsigned long                bAllowPsyNetParty : 1;                // 0x0078 (0x0004)
[0x0000000000000001] [0x00000004] (CPF_Edit)
unsigned long                bSendPingMessage : 1;                // 0x0078 (0x0004)
[0x0000000000004001] [0x00000008] (CPF_Edit | CPF_Config)
unsigned long                bAllowServerBacktraceUploads : 1;                // 0x0078
(0x0004) [0x0000000000000001] [0x00000010] (CPF_Edit)
unsigned long                bAllowServerBacktraceLogUploads : 1;                // 0x0078
(0x0004) [0x0000000000000001] [0x00000020] (CPF_Edit)
unsigned long                bAllowClientBacktraceUploads : 1;                // 0x0078 (0x0004)
[0x0000000000000001] [0x00000040] (CPF_Edit)
unsigned long                bAllowClientBacktraceLogUploads : 1;                // 0x0078
(0x0004) [0x0000000000000001] [0x00000080] (CPF_Edit)
unsigned long                bAllowPlayerCancelMatch : 1;                // 0x0078 (0x0004)
[0x0000000000000001] [0x00000100] (CPF_Edit)
float                PerConTimeout;                // 0x007C (0x0004)
[0x0000000000004001] (CPF_Edit | CPF_Config)
float                PerConPingInterval;                // 0x0080 (0x0004)
[0x0000000000004001] (CPF_Edit | CPF_Config)
float                FreshConnectionWindow;                // 0x0084 (0x0004)
[0x0000000000000001] (CPF_Edit)
float                MatchmakingHeartbeatTimeSeconds;                // 0x0088 (0x0004)
[0x0000000000000001] (CPF_Edit)

```

```

TArray<class FString>                                SuppressHeadersForDebug;                // 0x0090
(0x0010) [0x00000000000404001] (CPF_Edit | CPF_Config | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetConfig_X");
}

return uClassPointer;
};

};

// Class ProjectX.PsyNetRetryConfig_X
// 0x0030 (0x0078 - 0x00A8)
class UPsyNetRetryConfig_X : public UOnlineConfig_X
{
public:
TArray<class URetryDelay_X*>                        RetryDelays;                // 0x0078 (0x0010)
[0x00000000000400000] (CPF_NeedCtorLink)
TArray<class URetryPolicy_X*>                        RetryPolicies;              // 0x0088 (0x0010)
[0x00000000000400000] (CPF_NeedCtorLink)
TArray<class URetryPolicy_X*>                        ReAuthPolicies;            // 0x0098 (0x0010)
[0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetRetryConfig_X");
}

return uClassPointer;
};

class URetryDelay_X* FindReAuthDelay(class FString Service, struct FName Error);
class URetryDelay_X* FindRetryDelay(class FString Service, struct FName Error);
class URetryPolicy_X* FindReAuthPolicy(class FString Service, struct FName Error);
class URetryPolicy_X* FindRetryPolicy(class FString Service, struct FName Error);
};

// Class ProjectX.RPCConfig_X
// 0x0024 (0x0078 - 0x009C)
class URPCConfig_X : public UOnlineConfig_X
{
public:

```



```

TArray<class UClass*>                DisabledClasses;                // 0x0078 (0x0010)
[0x00000000000040001] (CPF_Edit | CPF_NeedCtorLink)
TArray<class FString>                DisabledServices;                // 0x0088 (0x0010)
[0x00000000000040001] (CPF_Edit | CPF_NeedCtorLink)
int32_t                             MaxRPCsPerBatch;                // 0x0098 (0x0004)
[0x00000000000000001] (CPF_Edit)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPCConfig_X");
}

return uClassPointer;
};

bool IsRPCDisabled(class URPC_X* RPC);
};

// Class ProjectX.ServerConfig_X
// 0x002C (0x0078 - 0x00A4)
class UServerConfig_X : public UOnlineConfig_X
{
public:
float                             HeartbeatSeconds;                // 0x0078 (0x0004)
[0x000000000000004001] (CPF_Edit | CPF_Config)
float                             HeartbeatRetrySeconds;            // 0x007C (0x0004)
[0x000000000000004001] (CPF_Edit | CPF_Config)
unsigned long                     bUploadLogFiles : 1;              // 0x0080 (0x0004)
[0x000000000000004001] [0x00000001] (CPF_Edit | CPF_Config)
unsigned long                     bUploadReplays : 1;              // 0x0080 (0x0004)
[0x000000000000004001] [0x00000002] (CPF_Edit | CPF_Config)
unsigned long                     bFlatbufferRecordInput : 1;      // 0x0080 (0x0004)
[0x000000000000004001] [0x00000004] (CPF_Edit | CPF_Config)
int32_t                           MaxUploadLogFileSize;            // 0x0084 (0x0004)
[0x000000000000004001] (CPF_Edit | CPF_Config)
float                             NetServerMaxTickRate;            // 0x0088 (0x0004)
[0x000000000000004001] (CPF_Edit | CPF_Config)
float                             IdleNetServerMaxTickRate;        // 0x008C (0x0004)
[0x000000000000004001] (CPF_Edit | CPF_Config)
float                             FlatbufferChance;                // 0x0090 (0x0004)
[0x000000000000004001] (CPF_Edit | CPF_Config)
int32_t                           FlatbufferMaxRecordSizeMB;        // 0x0094 (0x0004)
[0x000000000000004001] (CPF_Edit | CPF_Config)
int32_t                           FlatbufferPacketSendRate;        // 0x0098 (0x0004)
[0x000000000000004001] (CPF_Edit | CPF_Config)
float                             MinSecondsForUnstableConnection;  // 0x009C (0x0004)
[0x000000000000004001] (CPF_Edit | CPF_Config)
float                             MinSecondsUntilConnectionCheck;  // 0x00A0 (0x0004)
[0x000000000000004001] (CPF_Edit | CPF_Config)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ServerConfig_X");
}

return uClassPointer;
};

};

// Class ProjectX.OnlineFriendMap_X
// 0x0050 (0x0070 - 0x00C0)
class UOnlineFriendMap_X : public UComponent
{
public:
struct FMap_Mirror                PlayerMap;                // 0x0070 (0x0050)
[0x0000000000000001000] (CPF_Native)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineFriendMap_X");
}

return uClassPointer;
};

void ValueArray(TArray<struct FOnlineFriend>& OutArray);
void AppendArray(TArray<struct FOnlineFriend>& InArray);
void CopyArray(TArray<struct FOnlineFriend>& InArray);
bool Contains(struct FUniqueNetId& Key);
void Clear(int32_t ExpectedElements);
void Remove(struct FUniqueNetId& Key);
bool TryGet(struct FUniqueNetId& Key, struct FOnlineFriend& OutValue);
void Set(struct FUniqueNetId& Key, struct FOnlineFriend& Value);
int32_t Count();
void AllValues(int32_t StartIndex, int32_t MaxValues, struct FOnlineFriend& OutFriend);
};

// Class ProjectX.OnlineGameSearch_X
// 0x0044 (0x0154 - 0x0198)
class UOnlineGameSearch_X : public UOnlineGameSearch
{
public:

```

```

TArray<class FString>          InclusiveGameTagsArray;          // 0x0158 (0x0010)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<class FString>          ExclusiveGameTagsArray;          // 0x0168
(0x0010) [0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                  MapName;                          // 0x0178 (0x0010)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                  GameTagsDelimiter;                // 0x0188 (0x0010)
[0x000000000000400003] (CPF_Edit | CPF_Const | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameSearch_X");
}

return uClassPointer;
};

bool GetStringProperty(int32_t PropertyId, class FString& Value);
class FString eventGetExclusiveGameTags();
class FString eventGetInclusiveGameTags();
void SetMap(class FString inMapName);
void SetExclusiveGameTags(TArray<class FString> inGameTagsArray);
void SetInclusiveGameTags(TArray<class FString> inGameTagsArray);
};

// Class ProjectX.OnlineGameSettings_X
// 0x0058 (0x00A0 - 0x00F8)
class UOnlineGameSettings_X : public USettings
{
public:
int32_t                        NumPublicConnections;              // 0x00A0 (0x0004)
[0x000000000000000000]
int32_t                        NumOpenPublicConnections;          // 0x00A4 (0x0004)
[0x000000000000000000]
TArray<class UGameSettingCategory_X*> GameSettingCategories;      //
0x00A8 (0x0010) [0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)
TArray<class UGameSetting_X*> GameModeMutatorSettingPresets;      //
0x00B8 (0x0010) [0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)
unsigned long                  bOffline : 1;                      // 0x00C8 (0x0004)
[0x000000000000002000] [0x00000001] (CPF_Transient)
unsigned long                  bIgnoreHiddenMutatorsOverride : 1; // 0x00C8
(0x0004) [0x000100000000000000] [0x00000002]
int32_t                        MinimumPlayersRequired;           // 0x00CC (0x0004)
[0x000000040000000000]
int32_t                        NumSecondsWaitingForPlayers;       // 0x00D0 (0x0004)
[0x000000000000000000]
int32_t                        SearchScore;                       // 0x00D4 (0x0004)
[0x000000000000000000]
float                          LogTime;                          // 0x00D8 (0x0004)

```

```

[0x0000004040002000] (CPF_Transient | CPF_EditInlineNotify)
struct FScriptDelegate      __EventNumPlayersUpdated__Delegate;      // 0x00E0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameSettings_X");
}

return uClassPointer;
};

void PrintDebugInfo(class UDebugDrawer* Drawer);
bool Equals(class UOnlineGameSettings_X* OtherGameSettings);
void CalculateMinAndMaxNumPlayers();
void CopyFrom(class UOnlineGameSettings_X* OtherGameSettings);
class UGameSettingPlaylist_X* GetPlaylist();
bool IsValid();
void RemoveMismatchedTags(class UOnlineGameSettings_X* OtherGameSettings);
class UGameSettingCategory_X* GetGameSettingCategoryByName(struct FName
CategoryName);
void SetIntProperty(int32_t PropertyId, int32_t Value);
void SetGameTag(struct FName NewGameTag);
void UpdateFromURL(class AGameInfo* Game, class FString& URL);
bool HasSetting(struct FName SettingName);
void GetActiveGameSettings(TArray<class UGameSetting_X*>& GameSettings);
class UGameSetting_X* GetGameSetting(class FString SettingName, class
UGameSettingCategory_X*& Category);
class FString GetGameTags();
bool GetStringProperty(int32_t PropertyId, class FString& Value);
void Init();
void EventNumPlayersUpdated(class UOnlineGameSettings_X* GameSettings, int32_t
NewMinimumPlayersRequired);
};

// Class ProjectX.OnlineLegalText_X
// 0x0040 (0x0070 - 0x00B0)
class UOnlineLegalText_X : public UComponent
{
public:
class FString      Folder;      // 0x0070 (0x0010)
[0x00000000000400001] (CPF_Edit | CPF_NeedCtorLink)
class UCachedWebData_X*      CachedData;      // 0x0080 (0x0008)
[0x00000004000002000] (CPF_Transient)
class FString      AppendedPath;      // 0x0088 (0x0010)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FScriptDelegate      __EventNewText__Delegate;      // 0x0098
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineLegalText_X");
}

return uClassPointer;
};

void __OnlineLegalText_X__Sync_0x1(class UUrlConfig_X* UrlConfig);
void HandleWebText(class UCachedWebData_X* InCachedData);
class FString GetWebUrl(class UUrlConfig_X* UrlConfig);
class FString GetPlatformString();
class FString GetRelativeUrl(unsigned long bUseAppendedPath);
class FString eventGetFileSystemUrl();
class FString GetText();
void ClearCache();
void Sync(class FString InAppendPath);
void EventNewText(class UOnlineLegalText_X* Text);
};

// Class ProjectX.OnlineMessage_X
// 0x0000 (0x0060 - 0x0060)
class UOnlineMessage_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineMessage_X");
}

return uClassPointer;
};

};

// Class ProjectX.OnlineMessageComponent_X
// 0x0038 (0x0070 - 0x00A8)
class UOnlineMessageComponent_X : public UComponent
{
public:
class USerializer_X*                               Serializer;                               // 0x0070 (0x0008)
[0x0000000000000001] (CPF_Edit)
class UCompression_X*                               Compressor;                               // 0x0078 (0x0008)

```

```

[0x0000000000000001] (CPF_Edit)
TArray<struct FOnlineMessageHandler>      MessageHandlers;          // 0x0080
(0x0010) [0x0000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
struct FScriptDelegate      __MessageReceivedDelegate__Delegate;    // 0x0090
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineMessageComponent_X");
}

return uClassPointer;
};

```

```

class UOnlineMessageComponent_X* EnableCompression(unsigned long bEnabled);
class UOnlineMessageComponent_X* SetSerializer(class USerializer_X* InSerializer);
void InvokeHandlers(class UObject* Message);
class UObject* DeserializeMessage(class UClass* ExpectedMessageBaseClass, class UObject*
MessageOuter, TArray<uint8_t>& Data);
bool SerializeMessage(class UObject* Message, TArray<uint8_t>& OutData);
class UOnlineMessage_X* CreateMessage(class UClass* MessageClass, class UObject*
MessageOuter);
void RemoveMessageHandler(struct FScriptDelegate Callback);
void AddMessageHandler(class UClass* MessageClass, struct FScriptDelegate Callback);
void MessageReceivedDelegate(class UOnlineMessageComponent_X* Component, class
UObject* Message);
};

```

```

// Class ProjectX.ParameterDispenser_X
// 0x0068 (0x0070 - 0x00D8)
class UParameterDispenser_X : public UComponent
{
public:
struct FPointer      VfTable_IISetParameter;          // 0x0070 (0x0008)
[0x0000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
TArray<struct FNameParamPair>      NameParams;          // 0x0078
(0x0010) [0x00000004000400001] (CPF_Edit | CPF_NeedCtorLink)
TArray<struct FFloatParamPair>      FloatParams;          // 0x0088 (0x0010)
[0x00000004000400001] (CPF_Edit | CPF_NeedCtorLink)
TArray<struct FVectorParamPair>      VectorParams;          // 0x0098
(0x0010) [0x00000004000400001] (CPF_Edit | CPF_NeedCtorLink)
TArray<struct FColorParamPair>      ColorParams;          // 0x00A8 (0x0010)
[0x00000004000400001] (CPF_Edit | CPF_NeedCtorLink)
TArray<struct FActorParamPair>      ActorParams;          // 0x00B8 (0x0010)
[0x00000004000400001] (CPF_Edit | CPF_NeedCtorLink)
TArray<class UISetParameter*>      AllComponents;          // 0x00C8
(0x0010) [0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)

```

```

public:

```

```

static UClass* StaticClass()
{
    static UClass* uClassPointer = nullptr;

    if (!uClassPointer)
    {
        uClassPointer = UObject::FindClass("Class ProjectX.ParameterDispenser_X");
    }

    return uClassPointer;
};

void Inherit(class UParameterDispenser_X* Other);
void ResetMaterials();
void ApplyAllParameters(class UISetParameter* ActorComp);
class AActor* GetActorParameter(struct FName Key);
void SetActorParameter(struct FName Key, class AActor* Value);
void SetLinearColorParameter(struct FName Key, struct FLinearColor Value);
void SetVectorParameter(struct FName Key, struct FVector Value);
void SetFloatParameter(struct FName Key, float Value);
void SetNameParameter(struct FName Key, struct FName Value);
void RemoveComponent(class UISetParameter* ActorComp);
void AddComponent(class UISetParameter* ActorComp);
};

// Class ProjectX.ParticleModuleBeamTarget_X
// 0x0004 (0x010C - 0x0110)
class UParticleModuleBeamTarget_X : public UParticleModuleBeamTarget
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.ParticleModuleBeamTarget_X");
        }

        return uClassPointer;
    };
};

// Class ProjectX.ParticleModuleLocationBoneSocket_X
// 0x0004 (0x00AC - 0x00B0)
class UParticleModuleLocationBoneSocket_X : public UParticleModuleLocationBoneSocket
{
public:

public:
    static UClass* StaticClass()

```

```

{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class ProjectX.ParticleModuleLocationBoneSocket_X");
}

return UClassPointer;
};

};

// Class ProjectX.ParticleModuleLocationSkelVertSurface_X
// 0x0000 (0x00D0 - 0x00D0)
class UParticleModuleLocationSkelVertSurface_X : public
UParticleModuleLocationSkelVertSurface
{
public:

public:
static UClass* StaticClass()
{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class ProjectX.ParticleModuleLocationSkelVertSurface_X");
}

return UClassPointer;
};

};

// Class ProjectX.ParticleModuleVelocitySurfaceNormal_X
// 0x0034 (0x0074 - 0x00A8)
class UParticleModuleVelocitySurfaceNormal_X : public UParticleModuleVelocityBase
{
public:
struct FName MeshActorParamName; // 0x0078 (0x0008)
[0x0000000000000001] (CPF_Edit)
struct FRawDistributionFloat VelocityMultiplier; // 0x0080 (0x0028)
[0x0000000000480001] (CPF_Edit | CPF_Component | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class ProjectX.ParticleModuleVelocitySurfaceNormal_X");
}
}

```



```

return uClassPointer;
};

};

// Class ProjectX.ParticleModuleVelocitySurfaceNormalStatic_X
// 0x0000 (0x00A8 - 0x00A8)
class UParticleModuleVelocitySurfaceNormalStatic_X : public
UParticleModuleVelocitySurfaceNormal_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.ParticleModuleVelocitySurfaceNormalStatic_X");
}

return uClassPointer;
};

};

// Class ProjectX.ParticleModuleWind_X
// 0x0050 (0x0070 - 0x00C0)
class UParticleModuleWind_X : public UParticleModuleWorldForcesBase
{
public:
struct FRawDistributionFloat          StrengthScaleOverLife;          // 0x0070
(0x0028) [0x00000000000480001] (CPF_Edit | CPF_Component | CPF_NeedCtorLink)
struct FRawDistributionFloat          AccelerationOverLife;          // 0x0098 (0x0028)
[0x00000000000480001] (CPF_Edit | CPF_Component | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ParticleModuleWind_X");
}

return uClassPointer;
};

};

```

```

// Class ProjectX.Pawn_X
// 0x0044 (0x0514 - 0x0558)
class APawn_X : public APawn
{
public:
float GravityScale; // 0x0518 (0x0004)
[0x0000000000000001] (CPF_Edit)
class AController* PreviousController; // 0x0520 (0x0008)
[0x0000000000002000] (CPF_Transient)
struct FScriptDelegate __EventDestroyed__Delegate; // 0x0528
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventAnimEnd__Delegate; // 0x0540
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Pawn_X");
}

return uClassPointer;
};

bool eventIsSameTeam(class APawn* Other);
void eventDestroyed();
float GetGravityZ();
void eventOnAnimEnd(class UAnimNodeSequence* SeqNode, float PlayedTime, float
ExcessTime);
void OnControllerChanged();
void PossessedBy(class AController* C);
void UpdateControllerRef();
void eventReplicatedEvent(struct FName VarName);
void EventAnimEnd(class APawn_X* ForPawn, class UAnimNodeSequence* SeqNode);
void EventDestroyed(class APawn_X* Pawn);
};

// Class ProjectX.PerCon_X
// 0x0038 (0x0060 - 0x0098)
class UPerCon_X : public UObject
{
public:
class UPsyNetConfig_X* Config; // 0x0060 (0x0008)
[0x0000800000000000]
class FString URL; // 0x0068 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
uint8_t Status; // 0x0078 (0x0001)
[0x0000004000000000]
struct FScriptDelegate __EventStatusChanged__Delegate; // 0x0080
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PerCon_X");
}

return uClassPointer;
};

static bool UseWebSocket(unsigned long bRpcWebSocket);
void HandleConfigChanged();
void SetStatus(uint8_t InStatus);
void SetEnabled(unsigned long bEnable);
void EventStatusChanged(class UPerCon_X* PerCon);
};

// Class ProjectX.PhysicalMaterialProperty_X
// 0x0008 (0x0060 - 0x0068)
class UPhysicalMaterialProperty_X : public UPhysicalMaterialPropertyBase
{
public:
struct FName                                     SurfaceType;                                // 0x0060 (0x0008)
[0x0000000000000001] (CPF_Edit)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PhysicalMaterialProperty_X");
}

return uClassPointer;
};

};

// Class ProjectX.PlayerInput_X
// 0x0180 (0x02F8 - 0x0478)
class UPlayerInput_X : public UPlayerInput
{
public:
float                                     GamepadDeadzone;                                // 0x02F8 (0x0004)
[0x0000000000000400] (CPF_Config)
TArray<struct FGamepadDeadzoneSettings>      GamepadDeadzones;                                //
0x0300 (0x0010) [0x0000000000040400] (CPF_Config | CPF_NeedCtorLink)
float                                     KeyboardAxisBlendTime;                        // 0x0310 (0x0004)
[0x0000000000000400] (CPF_Config)

```

```

TArray<struct FKeyboardAxisBlendSettings>    KeyboardAxisBlendTimes;           //
0x0318 (0x0010) [0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
struct FName                                CurrentKey;                       // 0x0328 (0x0008)
[0x00000000000002002] (CPF_Const | CPF_Transient)
struct FName                                ActiveDPadButton;                 // 0x0330 (0x0008)
[0x00000000000002002] (CPF_Const | CPF_Transient)
struct FName                                LastDoubleTapKey;                   // 0x0338 (0x0008)
[0x00000000000002002] (CPF_Const | CPF_Transient)
TArray<struct FName>                        DisabledActions;                 // 0x0340 (0x0010)
[0x0000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
TArray<struct FName>                        DisabledActionsUntilNextUse;       // 0x0350
(0x0010) [0x0000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
TArray<struct FBindingAction>               Actions;                          // 0x0360 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
class UControlPreset_X*                     ControlPreset;                       // 0x0370 (0x0008)
[0x00000000000000000]
TArray<struct FPlayerBinding>               PCBindings;                        // 0x0378 (0x0010)
[0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
TArray<struct FPlayerBinding>               GamepadBindings;                  // 0x0388
(0x0010) [0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
TArray<struct FPlayerBinding>               SteamInputBindings;              // 0x0398
(0x0010) [0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
float                                        TapTime;                          // 0x03A8 (0x0004)
[0x0000000000004000] (CPF_Config)
float                                        DoubleTapTime;                      // 0x03AC (0x0004)
[0x0000000000004000] (CPF_Config)
TArray<struct FPointer>                     HeldBindings;                     // 0x03B0 (0x0010)
[0x0000000000003000] (CPF_Native | CPF_Transient)
unsigned long                               bDebugInput : 1;                  // 0x03C0 (0x0004)
[0x0000000000002000] [0x00000001] (CPF_Transient)
unsigned long                               bAbsorbCurrentKeyPress : 1;       // 0x03C0 (0x0004)
[0x0000000000002002] [0x00000002] (CPF_Const | CPF_Transient)
uint8_t                                    UnknownData00[0x50];              // 0x03C8 (0x0050)
UNKNOWN PROPERTY: MapProperty ProjectX.PlayerInput_X.RawAxisValues
struct FScriptDelegate                      __EventActionToggled__Delegate;  // 0x0418
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventBindingsChanged__Delegate; // 0x0430
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventSetBindingsToUserBindings__Delegate; //
0x0448 (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventInitialized__Delegate;    // 0x0460 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PlayerInput_X");
}

return uClassPointer;

```

```

};

void PrintDebugInfo(class UDebugDrawer* Drawer);
static struct FName GetKeyForActionArray(struct FName Action, TArray<struct FPlayerBinding>& PlayerBindings);
void AbsorbCurrentKeyPress();
void SetBindings(TArray<struct FPlayerBinding>& PC, TArray<struct FPlayerBinding>& Gamepad);
void ClearDisableActionUntilNextUseList();
void DisableActionUntilNextUse(struct FName ActionName);
void ClearDisabledActions();
void RemoveFromDisabledActions(struct FName ActionName);
void AddToDisabledActions(struct FName ActionName);
void ReleaseKey(struct FName Key, unsigned long bTriggerEvents);
static class FString GetUIKey(struct FName KeyName);
void ShutdownInputSystem();
void InitInputSystem();
void ResetInput();
float GetRawSplitAxisValue(struct FName AxisNegative, struct FName AxisPositive);
float GetRawAxisValue(struct FName AxisName);
static class UControlPreset_X* GetControlPreset(struct FName PresetName);
void SetControlPreset(struct FName PresetName);
void ResetActiveBindingsToProfileBindings();
void ResetBindingsToDefault();
void EventInitialized(class UPlayerInput_X* PlayerInput);
void EventSetBindingsToUserBindings(class UPlayerInput_X* PlayerInput);
void EventBindingsChanged(class UPlayerInput_X* PlayerInput);
void EventActionToggled(class UPlayerInput_X* PlayerInput, struct FName ActionName, unsigned long bEnabled);
};

// Class ProjectX.PointLightComponent_X
// 0x0010 (0x0264 - 0x0274)
class UPointLightComponent_X : public UPointLightComponent
{
public:
class UDistributionFloatConstantCurve*      BrightnessOverTime;          // 0x0268
(0x0008) [0x0000000004080009] (CPF_Edit | CPF_ExportObject | CPF_Component | CPF_EditInline)
float          AttachTime;          // 0x0270 (0x0004)
[0x0000000000000200] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PointLightComponent_X");
}

return uClassPointer;
};

```

```

};

// Class ProjectX.PsyNet_X
// 0x0100 (0x0060 - 0x0160)
class UPsyNet_X : public UObject
{
public:
uint8_t Environment; // 0x0060 (0x0001)
[0x00000004000002000] (CPF_Transient)
class FString EnvironmentName; // 0x0068 (0x0010)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FPsyNetKeys Keys; // 0x0078 (0x0040)
[0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
class UStringMap* Headers; // 0x00B8 (0x0008)
[0x00000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UPsyNetServiceSubscriptions_X* Services; // 0x00C0
(0x0008) [0x00000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UPsyNetConnection_X* PrimaryEnabledConnection; // 0x00C8
(0x0008) [0x00000004000000000]
class UPsyNetConnection_X* PrimaryAuthedConnection; // 0x00D0
(0x0008) [0x00000004000000000]
class UPsyNetConnection_X* AnonymousConnection; // 0x00D8
(0x0008) [0x00000004000000000]
float LastReportCheaterTime; // 0x00E0 (0x0004)
[0x00000000000000000]
unsigned long bUsePsynetEnvironment : 1; // 0x00E4 (0x0004)
[0x00000000000004000] [0x000000001] (CPF_Config)
struct FScriptDelegate __EventPrimaryConnectionEnabled__Delegate; //
0x00E8 (0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventPrimaryConnectionDisabled__Delegate; //
0x0100 (0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventPrimaryPlayerLoggedIn__Delegate; // 0x0118
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventPrimaryPlayerLoggedOut__Delegate; // 0x0130
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventCheaterReported__Delegate; // 0x0148
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNet_X");
}

return uClassPointer;
};

static void ReportCheater(struct FUniqueNetId Id, class FString Reason);
static class FString AssignQWordToString(uint64_t Q);
static uint64_t AssignStringToQWord(class FString S);

```

```

static float GetRetryDelay(int32_t Failures, TArray<float> Delays);
static void eventConnectionChangedIP(struct FUniqueNetId PlayerID);
static void eventNetworkError(struct FUniqueNetId PlayerID, class FString Reason);
void InitHeaders();
void NotifyWhenLoggedIn(struct FScriptDelegate LoginCallback, struct FScriptDelegate
LogoutCallback);
void NotifyWhenPrimaryConnectionEnabled(struct FScriptDelegate EnabledCallback, struct
FScriptDelegate DisabledCallback);
void SetPrimaryAuthedConnection(class UPsyNetConnection_X* Connection);
void SetPrimaryEnabledConnection(class UPsyNetConnection_X* Connection);
void HandleConnectionChanged(class UPsyNetConnection_X* Connection);
class UPsyNetConnection_X* GetPrimaryConnection();
void HandleServiceExecuted(class UPsyNetServiceProvider_X* P, class UPsyNetClientService_X*
Service);
void eventInit();
void Flush(float TimeoutSeconds);
static class URPC_X* QueueRPC(class URPC_X* RPC);
static class URPC_X* RPC(class UClass* RPCClass);
static class UPsyNet_X* GetInstance();
void EventCheaterReported();
void EventPrimaryPlayerLoggedOut();
void EventPrimaryPlayerLoggedIn();
void EventPrimaryConnectionDisabled();
void EventPrimaryConnectionEnabled();
};

// Class ProjectX.PsyNetBeaconConnection_X
// 0x002C (0x0060 - 0x008C)
class UPsyNetBeaconConnection_X : public UObject
{
public:
    struct FPointer VfTable_IIReservationConnection_X; // 0x0060
    (0x0008) [0x00000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
    class FString ReservationID; // 0x0068 (0x0010)
    [0x00000000000400000] (CPF_NeedCtorLink)
    class FString ConnectionID; // 0x0078 (0x0010)
    [0x00000000000400000] (CPF_NeedCtorLink)
    float TimeoutTime; // 0x0088 (0x0004)
    [0x00000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PsyNetBeaconConnection_X");
        }

        return uClassPointer;
    };
};

```

```

// Class ProjectX.PsyNetClientService_X
// 0x0030 (0x0060 - 0x0090)
class UPsyNetClientService_X : public UObject
{
public:
class FString                               Service;                               // 0x0060 (0x0010)
[0x000000000000400003] (CPF_Edit | CPF_Const | CPF_NeedCtorLink)
int32_t                                     Version;                               // 0x0070 (0x0004)
[0x000000000000000003] (CPF_Edit | CPF_Const)
class UError*                               Error;                               // 0x0078 (0x0008)
[0x000000000000000002] (CPF_Const)
class UPsyNetConnection_X*                 Connection;                               // 0x0080 (0x0008)
[0x000000000000000000]
uint64_t                                    Expiration;                               // 0x0088 (0x0008)
[0x000000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetClientService_X");
}

return uClassPointer;
};

void Execute();
void SetError(class UError* InError);
};

// Class ProjectX.PsyNetClientServiceCollection_X
// 0x0010 (0x0060 - 0x0070)
class UPsyNetClientServiceCollection_X : public UObject
{
public:
TArray<class UPsyNetClientService_X*>      ServiceArchetypes;                               // 0x0060
(0x0010) [0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetClientServiceCollection_X");
}

return uClassPointer;
};

```



```

class UPsyNetClientService_X* GetServiceArchetype(class FString ServiceName, int32_t
Version);
void CollectServiceArchetypes();
void eventConstruct();
};

// Class ProjectX.PsyNetConnection_X
// 0x0170 (0x0060 - 0x01D0)
class UPsyNetConnection_X : public UObject
{
public:
class UPsyNet_X*                PsyNet;                // 0x0060 (0x0008)
[0x0000800000000000]
class UOnlineSubsystem*        OnlineSub;                // 0x0068 (0x0008)
[0x0000800000000000]
class UPsyNetConfig_X*        Config;                // 0x0070 (0x0008)
[0x0000800000000000]
class UPsyNetRetryConfig_X*    RetryConfig;            // 0x0078 (0x0008)
[0x0000800000000000]
class UPsyNetUrl_X*            URL;                // 0x0080 (0x0008)
[0x0000800000000000]
class UStringMap*              Headers;                // 0x0088 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UPsyNetRequestQueue_X*    RequestQueue;            // 0x0090 (0x0008)
[0x0000004000000000]
class URPCQueue_X*            RPCQueue;                // 0x0098 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UPsyNetServiceProvider_X* ServiceProvider;        // 0x00A0
(0x0008) [0x0000004000000000]
class UPsyNetMessengerHttp_X*    HttpMessenger;        // 0x00A8
(0x0008) [0x0000004000000000]
class UPsyNetMessengerWebSocket_X* PerConMessenger;    // 0x00B0
(0x0008) [0x0000004000000000]
class UPerCon_X*                PerCon;                // 0x00B8 (0x0008)
[0x0000004000000000]
class UPerConMetrics_X*        PerConMetrics;            // 0x00C0 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
unsigned long                    bAuthorized : 1;        // 0x00C8 (0x0004)
[0x0000004000000000] [0x00000001]
unsigned long                    bConnected : 1;        // 0x00C8 (0x0004)
[0x0000004000000000] [0x00000002]
unsigned long                    bFreshConnection : 1;    // 0x00C8 (0x0004)
[0x0000004000000000] [0x00000004]
unsigned long                    bPerConConnected : 1;    // 0x00C8 (0x0004)
[0x0000004000000000] [0x00000008]
float                            ConnectedChangeTime;    // 0x00CC (0x0004)
[0x0000004000000000]
class UError*                    DisabledError;            // 0x00D0 (0x0008)
[0x0000004000000000]
TArray<class UError*>            DisabledErrorStack;        // 0x00D8 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
class UError*                    MaintenanceError;        // 0x00E8 (0x0008)
[0x0000000000000000]

```

```

class UError*                DuplicateLoginError;                // 0x00F0 (0x0008)
[0x0000000000000000]
class UError*                NoInternetError;                    // 0x00F8 (0x0008)
[0x0000000000000000]
class UError*                NoUrlError;                        // 0x0100 (0x0008)
[0x0000000000000000]
class UError*                AuthDisabledError;                // 0x0108 (0x0008)
[0x0000000000000000]
TArray<float>                AuthRetryDelays;                    // 0x0110 (0x0010)
[0x0000000000400001] (CPF_Edit | CPF_NeedCtorLink)
int32_t                      ConsecutiveAuthFailures;          // 0x0120 (0x0004)
[0x0000000000000000]
struct FScriptDelegate       __EventConnected__Delegate;      // 0x0128
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventDisconnected__Delegate;    // 0x0140
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventConnectFailed__Delegate;   // 0x0158
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventPerConConnected__Delegate; // 0x0170
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventPerConDisconnected__Delegate; // 0x0188
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventEnabled__Delegate;        // 0x01A0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventDisabled__Delegate;       // 0x01B8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetConnection_X");
}

return uClassPointer;
};

void eventSendPingMessage();
void WatchPerConConnection(struct FScriptDelegate OnConnected, struct FScriptDelegate
OnDisconnected);
void StopWatchConnection(struct FScriptDelegate OnConnected, struct FScriptDelegate
OnDisconnected);
void WatchConnection(struct FScriptDelegate OnConnected, struct FScriptDelegate
OnDisconnected);
void WatchEnabled(struct FScriptDelegate OnEnabled, struct FScriptDelegate OnDisabled);
void eventDispose();
void UpdateLinkConnection(unsigned long bHasConnection);
void ClearDuplicateLoginError();
void OnDuplicateLogin();
void HandleDuplicateLoginMessage(class UPsyNetClientService_X* Service);
void HandleWebSocketDisconnect(class UPsyNetMessengerWebSocket_X* WS, int32_t Code,

```

```

class FString Reason);
void HandleWebSocketConnect(class UPsyNetMessageWebSocket_X* WS);
void HandleWebSocketStartConnectFail(class UPsyNetMessageWebSocket_X* WS);
void UpdateConnectionState();
void ClearAuthDisabledError();
void SetAuthDisabledError(class UError* Error);
void ConditionalSetAuthRetryDelay(class FString Service, class UError* Error);
void HandleErrorRPC(class URPCQueue_X* InQueue, class URPC_X* InRPC, class UError* Error);
bool IsEnabled();
void UpdateDisabledError(class UErrorType* Type, unsigned long blsError, class UError*& Error);
void eventAddDisabledError(class UError* Error);
void RemoveDisabledError(class UError* Error);
uint8_t Flush(float TimeoutSeconds);
class URPC_X* QueueRPC(class URPC_X* RPC);
class URPC_X* RPC(class UClass* RPCClass);
bool ProcessServiceCall(class UPsyNetMessage_X* Message);
void ProcessMessage(class UPsyNetMessage_X* Message);
void UpdatePsyTime(class UPsyNetMessage_X* Message);
void ReceiveMessage(class UPsyNetMessage_X* Message);
class UAsyncResult_PsyNetMessage_X* SendRequest(class FString Service, class
UPsyNetMessage_X* Request);
class UAsyncTask* SendMessageW(class UPsyNetMessage_X* Message);
void RemoveHeader(class FString Key);
void SetHeader(class FString Key, class FString Value);
void SetAuthorized(unsigned long bAuth);
class UStringMap* BuildHandshakeHeaders();
class UPsyNetMessage_X* eventGetMessenger();
void CreatePerConMessenger();
void CreateHttpMessenger(class FString InURL);
void KillPerConMessenger();
void KillHttpMessenger();
void KillMessengers();
void CreateMessengers();
void InitMessengers();
void InitServiceProvider();
void HandlePerConStatusChanged(class UPerCon_X* P);
void InitPerCon();
void InitRPCQueue();
void InitRequestQue();
void InitHeaders();
void UpdateNoUrlError();
void HandleUrlChanged();
class FString GetPsyNetURL();
void eventConstruct();
void EventDisabled(class UPsyNetConnection_X* Connection);
void EventEnabled(class UPsyNetConnection_X* Connection);
void EventPerConDisconnected(class UPsyNetConnection_X* Connection);
void EventPerConConnected(class UPsyNetConnection_X* Connection);
void EventConnectFailed(class UPsyNetConnection_X* Connection);
void EventDisconnected(class UPsyNetConnection_X* Connection);
void EventConnected(class UPsyNetConnection_X* Connection);
};

```

```
// Class ProjectX.PsyNetErrorType_X
```

```

// 0x0000 (0x0070 - 0x0070)
class UPsyNetErrorType_X : public UErrorType
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetErrorType_X");
}

return uClassPointer;
};

};

// Class ProjectX.PsyNetMessage_X
// 0x0018 (0x0060 - 0x0078)
class UPsyNetMessage_X : public UObject
{
public:
class UStringMap* Headers; // 0x0060 (0x0008)
[0x000000000408000A] (CPF_Const | CPF_ExportObject | CPF_Component | CPF_EditInline)
TArray<uint8_t> Body; // 0x0068 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetMessage_X");
}

return uClassPointer;
};

class FString GetBodyText();
void SetBodyText(class FString Value);
void SetBody(TArray<uint8_t>& Value);
};

// Class ProjectX.PsyNetMessenger_X
// 0x0018 (0x0060 - 0x0078)
class UPsyNetMessenger_X : public UObject
{
public:
struct FScriptDelegate __EventMessageReceived__Delegate; // 0x0060

```

(0x0018) [0x0000000000400000] (CPF\_NeedCtorLink)

```
public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetMessage_X");
}
```

```
return uClassPointer;
};
```

```
void eventDispose();
void EventMessageReceived(class UPsyNetMessage_X* Message);
class UAsyncTask* SendMessageW(class UPsyNetMessage_X* Message);
void Connect();
};
```

```
// Class ProjectX.PsyNetMessageHttp_X
// 0x0010 (0x0078 - 0x0088)
class UPsyNetMessageHttp_X : public UPsyNetMessage_X
{
public:
class FString URL; // 0x0078 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
```

```
public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetMessageHttp_X");
}
```

```
return uClassPointer;
};
```

```
void CopyHeadersToMessage(class UWebRequest_X* From, class UPsyNetMessage_X* To);
void CopyHeadersToRequest(class UPsyNetMessage_X* From, class UWebRequest_X* To);
void HandleWebRequestComplete(class FString PsyRequestID, class UWebRequest_X* Request,
class UAsyncTask* Task);
class UAsyncTask* SendMessageW(class UPsyNetMessage_X* Message);
};
```

```
// Class ProjectX.PsyNetMessageWebSocket_X
// 0x00B8 (0x0078 - 0x0130)
class UPsyNetMessageWebSocket_X : public UPsyNetMessage_X
{
public:
```

```

float                                MessageSendTimeout;                                // 0x0078 (0x0004)
[0x00000000000000001] (CPF_Edit)
class UStringMap*                    HandshakeHeaders;                                // 0x0080 (0x0008)
[0x00000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class FString                        URL;                                              // 0x0088 (0x0010)
[0x00000004000400000] (CPF_NeedCtorLink)
class UWebSocketConnection_X*        Connection;                                    // 0x0098
(0x0008) [0x00000004000000000]
TArray<struct FQueuedPsyNetMessage>   QueuedMessages;                                // 0x00A0
(0x0010) [0x00000000000400000] (CPF_NeedCtorLink)
class UPsyNetConfig_X*               PsyNetConfig;                                    // 0x00B0 (0x0008)
[0x00008000000000000]
struct FScriptDelegate               __EventStartConnect__Delegate;                  // 0x00B8
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate               __EventStartConnectFail__Delegate;              // 0x00D0
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate               __EventConnected__Delegate;                    // 0x00E8
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate               __EventDisconnected__Delegate;                  // 0x0100
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate               __EventInvalidMessageReceived__Delegate;        // 0x0118
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetMessageerWebSocket_X");
}

return uClassPointer;
};

void __PsyNetMessageerWebSocket_X__SendQueuedMessages_0x1(struct
FQueuedPsyNetMessage QueuedMessage);
void eventDispose();
void HandleReceivedBunch(class UTcpConnection* C);
void SerializeMessage(class UPsyNetMessage_X* Message);
void FailQueuedMessages(class UError* Error);
void TimeoutQueuedMessages();
void SendMessageWithTaskW(class UPsyNetMessage_X* Message, class UAsyncTask* Task);
void SendQueuedMessages();
void QueueMessage(class UPsyNetMessage_X* Message, class UAsyncTask* Task);
void StripHandshakeHeaders(class UPsyNetMessage_X* Message);
class UAsyncTask* SendMessageW(class UPsyNetMessage_X* Message);
void HandleDisconnected(class UWebSocketConnection_X* InConnection);
void HandleConnected(class UWebSocketConnection_X* InConnection);
void Disconnect();
void Connect();
bool IsConnected();
void Init(class FString InURL, class UStringMap* InHandshakeHeaders);

```

```

void EventInvalidMessageReceived(class UPsyNetMessageWebSocket_X* Messenger, class
FString MessageSubstring);
void EventDisconnected(class UPsyNetMessageWebSocket_X* Messenger, int32_t Code, class
FString Reason);
void EventConnected(class UPsyNetMessageWebSocket_X* Messenger);
void EventStartConnectFail(class UPsyNetMessageWebSocket_X* Messenger);
void EventStartConnect(class UPsyNetMessageWebSocket_X* Messenger);
};

```

```

// Class ProjectX.PsyNetServiceSerializer_X
// 0x0000 (0x0060 - 0x0060)
class UPsyNetServiceSerializer_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetServiceSerializer_X");
}

return uClassPointer;
};

static class FString SerializeResponse(class UPsyNetClientService_X* Service);
static void DeserializeRequest(class FString JSON, class UPsyNetClientService_X* Service);
};

```

```

// Class ProjectX.PsyNetServiceSubscriptions_X
// 0x0010 (0x0070 - 0x0080)
class UPsyNetServiceSubscriptions_X : public UComponent
{
public:
TArray<struct FServiceSubscription> Subscriptions; // 0x0070
(0x0010) [0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetServiceSubscriptions_X");
}

return uClassPointer;
};

void NotifyServiceExecuted(class UPsyNetClientService_X* Service);

```

```

void UnsubscribeAll(class UObject* Listener);
void Unsubscribe(struct FScriptDelegate Callback);
void Subscribe(class UClass* ServiceClass, struct FScriptDelegate Callback);
};

```

```

// Class ProjectX.PsyNetStaticData_X
// 0x00C8 (0x0070 - 0x0138)
class UPsyNetStaticData_X : public UComponent
{
public:
class FString                                PsyConfigTemplateURL;                // 0x0070 (0x0010)
[0x000000000000404001] (CPF_Edit | CPF_Config | CPF_NeedCtorLink)
float                                LocalCacheTimeoutSeconds;            // 0x0080 (0x0004)
[0x0000000000004001] (CPF_Edit | CPF_Config)
float                                SyncDataTimeoutSeconds;            // 0x0084 (0x0004)
[0x0000000000004001] (CPF_Edit | CPF_Config)
class UPsyNet_X*                        PsyNet;                            // 0x0088 (0x0008)
[0x0000800000000000]
class UOnlineSubsystem*                OnlineSub;                            // 0x0090 (0x0008)
[0x0000800000000000]
class UWebConfig_X*                    WebConfig;                            // 0x0098 (0x0008)
[0x0000800000000000]
unsigned long                        bDebug : 1;                            // 0x00A0 (0x0004)
[0x0000000000000001] [0x00000001] (CPF_Edit)
unsigned long                        bLoaded : 1;                            // 0x00A0 (0x0004)
[0x0000004000002000] [0x00000002] (CPF_Transient)
class UCacheTimer_X*                  CacheTimer;                            // 0x00A8 (0x0008)
[0x00000000004080009] (CPF_Edit | CPF_ExportObject | CPF_Component | CPF_EditInline)
class UOnlineResource_X*              WebData;                            // 0x00B0 (0x0008)
[0x00000000004080009] (CPF_Edit | CPF_ExportObject | CPF_Component | CPF_EditInline)
TArray<class UObject*>                DefaultObjects;                        // 0x00B8 (0x0010)
[0x0000000000400001] (CPF_Edit | CPF_NeedCtorLink)
TArray<class UObject*>                DownloadedObjects;                    // 0x00C8 (0x0010)
[0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<class UObject*>                RegisteredObjects;                    // 0x00D8 (0x0010)
[0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                        PsyConfigQueryParams;                // 0x00E8 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                        PsyConfigSecret;                        // 0x00F8 (0x0010)
[0x0000000000400002] (CPF_Const | CPF_NeedCtorLink)
class FString                        PsyConfigAltEnv;                        // 0x0108 (0x0010)
[0x0000000000400002] (CPF_Const | CPF_NeedCtorLink)
float                                BlockingSyncStartTime;                // 0x0118 (0x0004)
[0x0000000000000000]
struct FScriptDelegate                __EventLoaded__Delegate;            // 0x0120
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{

```



```

uClassPointer = UObject::FindClass("Class ProjectX.PsyNetStaticData_X");
}

return uClassPointer;
};

void __PsyNetStaticData_X__Init_0x2(class UObject* _);
void __PsyNetStaticData_X__Init_0x1(class UObject* _);
void __PsyNetStaticData_X__BlockUntilSyncFinished_0x1(class UOnlineResource_X* _);
static void DisableConnection(class UOnlinePlayer_X* Player, class UStaticDataError_X* InError);
void HandleStaticDataError(class UOnlineResource_X* DataSync, class UError* InError);
class UPsyNetStaticDataMetrics_X* GetMetrics();
void UpdateRegisteredObjects();
bool LoadFromUTF8(TArray<uint8_t>& SignedData);
bool LoadFromSignedUTF8WithSignature(class FString Signature, TArray<uint8_t>& SignedData);
bool LoadFromSignedUTF8(TArray<uint8_t>& SignedData);
void InstantiateObjects();
void HandleDataChanged(class UOnlineResource_X* Resource);
static class FString GetPlatform();
class FString GetPsyConfigUrl(class FString Template);
void UpdateBlockingSyncElapsedTime();
void BlockUntilSyncFinished();
void HandleGetURL(class FString URL);
void Sync();
void ForceFullBlockingSync();
void HandleCacheExpired(class UCacheTimer_X* Timer);
void UpdateLinkConnection(unsigned long bHasConnection);
void UpdateCacheTimerEnabled();
bool ShouldEnableCacheTimer();
void HandleWebConfigChanged();
void AddQueryParam(class FString Param);
static void SetSyncPaused(unsigned long bPause, class UPauseStaticDataSync_X* Instance);
void eventInit();
void EventLoaded(class UPsyNetStaticData_X* Data);
};

// Class ProjectX.PsyNetUrl_X
// 0x0028 (0x0060 - 0x0088)
class UPsyNetUrl_X : public UObject
{
public:
    unsigned long                bUseSubsystemURL : 1;                // 0x0060 (0x0004)
    [0x00000000000004001] [0x000000001] (CPF_Edit | CPF_Config)
    class FString                URL;                                // 0x0068 (0x0010)
    [0x00000000000004001] (CPF_Edit | CPF_Config | CPF_NeedCtorLink)
    class FString                URLv2;                            // 0x0078 (0x0010)
    [0x00000000000004001] (CPF_Edit | CPF_Config | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)

```

```

{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetUrl_X");
}

return uClassPointer;
};

class FString GetURL(uint8_t Environment);
};

// Class ProjectX.RadialBlurComponent_X
// 0x0008 (0x0110 - 0x0118)
class URadialBlurComponent_X : public URadialBlurComponent
{
public:
float                                     FadeTime;                                // 0x0110 (0x0004)
[0x0000000000000001] (CPF_Edit)
float                                     FadeFalloff;                                // 0x0114 (0x0004)
[0x0000000000000001] (CPF_Edit)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RadialBlurComponent_X");
}

return uClassPointer;
};

};

// Class ProjectX.RandomStream_X
// 0x0008 (0x0060 - 0x0068)
class URandomStream_X : public UObject
{
public:
struct FRandomStream_Mirror                Stream;                                // 0x0060 (0x0004)
[0x0000000000000100] (CPF_Native)
int32_t                                    CurrentSeed;                                // 0x0064 (0x0004)
[0x0000004000002000] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RandomStream_X");
}
}

```

```

return uClassPointer;
};

void SetCurrentSeed(int32_t NewSeed);
int32_t GetNextIntRange(int32_t MinValue, int32_t MaxValue);
float GetNextFloat();
};

// Class ProjectX.RemoteAvatarPermissions_X
// 0x0098 (0x0060 - 0x00F8)
class URemoteAvatarPermissions_X : public UObject
{
public:
class UOnlineGame_X*                OnlineGame;                // 0x0060 (0x0008)
[0x0000800000000000]
class UEpicConfig_X*                EpicConfig;                // 0x0068 (0x0008)
[0x0000800000000000]
TArray<class URemoteAvatarPermissionsRequest_X*>  UnsentRequests;                //
0x0070 (0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
TArray<class URemoteAvatarPermissionsRequest_X*>  AllRequests;                //
0x0080 (0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
struct FMap_Mirror                PermissionCache;                // 0x0090 (0x0050)
[0x0000000000001000] (CPF_Native)
struct FScriptDelegate                __SendRPC__Delegate;                // 0x00E0 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RemoteAvatarPermissions_X");
}

return uClassPointer;
};

struct FUniqueNetId __RemoteAvatarPermissions_X__TimerSendRPC_0x1(class
URemoteAvatarPermissionsRequest_X* Request);
void HandleRequestFail(class URPC_CanShowAvatar_X* RPC);
void HandleRequestSuccess(class URPC_CanShowAvatar_X* RPC);
void TimerSendRPC();
void GetPermission(struct FUniqueNetId PlayerID, struct FScriptDelegate AllowedCallback, struct
FScriptDelegate DisallowedCallback);
void GetPermissions(struct FScriptDelegate OnAllowed, struct FScriptDelegate OnDisallowed,
TArray<struct FUniqueNetId>& PlayerIDs);
void SetAvatarPermission(struct FUniqueNetId PlayerID, uint8_t PermissionStatus);
void SendRPC(class URPC_CanShowAvatar_X* RPC);
void OnAvatarPermissionSet(struct FUniqueNetId PlayerID, uint8_t PermissionStatus);
class URemoteAvatarPermissionsRequest_X* FindRequest(struct FUniqueNetId& PlayerID);
bool Contains(struct FUniqueNetId& PlayerID);

```

```

uint8_t TryGet(struct FUniqueNetId& PlayerID);
uint8_t AvatarPermissionToTryGetResult(uint8_t InPermission);
void CacheResult(uint8_t PermissionStatus, struct FUniqueNetId& PlayerID);
};

// Class ProjectX.RemoteAvatarPermissionsRequest_X
// 0x0078 (0x0060 - 0x00D8)
class URemoteAvatarPermissionsRequest_X : public UObject
{
public:
    struct FUniqueNetId          PlayerID;                // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate      __EventAllowed__Delegate; // 0x00A8
    (0x0018) [0x000000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate      __EventDisallowed__Delegate; // 0x00C0
    (0x0018) [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RemoteAvatarPermissionsRequest_X");
        }

        return uClassPointer;
    };

    void EventDisallowed(struct FUniqueNetId DisallowedPlayerId);
    void EventAllowed(struct FUniqueNetId AllowedPlayerId);
};

// Class ProjectX.RenderProfiler_X
// 0x0040 (0x0268 - 0x02A8)
class ARenderProfiler_X : public AActor
{
public:
    float          TimePerPrimitive;                // 0x0268 (0x0004)
    [0x00000000000000001] (CPF_Edit)
    TArray<struct FPrimitiveComponentProfile> PrimitiveProfiles; // 0x0270
    (0x0010) [0x000000000000482000] (CPF_Transient | CPF_Component | CPF_NeedCtorLink)
    int32_t        CurrentIndex;                    // 0x0280 (0x0004)
    [0x000000000000002000] (CPF_Transient)
    unsigned long   bInclusive : 1;                  // 0x0284 (0x0004)
    [0x000000000000002000] [0x00000001] (CPF_Transient)
    float          PrimitiveTime;                    // 0x0288 (0x0004)
    [0x000000000000002000] (CPF_Transient)
    struct FScriptDelegate __EventFinished__Delegate; // 0x0290
    (0x0018) [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()

```

```

{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RenderProfiler_X");
}

return uClassPointer;
};

void SetupPlayers();
void eventPostBeginPlay();
void EventFinished(class ARenderProfiler_X* Profiler);
};

// Class ProjectX.RetryDelay_X
// 0x0018 (0x0060 - 0x0078)
class URetryDelay_X : public UObject
{
public:
struct FName                                     Id;                                     // 0x0060 (0x0008)
[0x0000000000000000]
TArray<float>                                     DelaySeconds;                             // 0x0068 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RetryDelay_X");
}

return uClassPointer;
};

};

// Class ProjectX.RetryPolicy_X
// 0x0028 (0x0060 - 0x0088)
class URetryPolicy_X : public UObject
{
public:
TArray<struct FName>                             Errors;                                     // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FName                                     RetryDelay;                               // 0x0070 (0x0008)
[0x0000000000000000]
TArray<class FString>                             Services;                                  // 0x0078 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:

```

```

static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RetryPolicy_X");
}

return uClassPointer;
};

bool AppliesTo(class FString Service, struct FName Error);
};

// Class ProjectX.RPC_X
// 0x0088 (0x0060 - 0x00E8)
class URPC_X : public UObject
{
public:
class FString Service; // 0x0060 (0x0010)
[0x000000000000400003] (CPF_Edit | CPF_Const | CPF_NeedCtorLink)
int32_t Version; // 0x0070 (0x0004)
[0x000000000000000003] (CPF_Edit | CPF_Const)
float ServiceFailRetryDelay; // 0x0074 (0x0004)
[0x000000000000000003] (CPF_Edit | CPF_Const)
unsigned long bAllowBatching : 1; // 0x0078 (0x0004)
[0x000000000000000003] [0x00000001] (CPF_Edit | CPF_Const)
unsigned long bRequiresAuth : 1; // 0x0078 (0x0004)
[0x000000000000000003] [0x00000002] (CPF_Edit | CPF_Const)
unsigned long bDisposed : 1; // 0x0078 (0x0004)
[0x00000004000002000] [0x00000004] (CPF_Transient)
uint8_t Priority; // 0x007C (0x0001)
[0x000000000000000003] (CPF_Edit | CPF_Const)
TArray<struct FKeyValuePair> Headers; // 0x0080 (0x0010)
[0x000000000000400003] (CPF_Edit | CPF_Const | CPF_NeedCtorLink)
class UError* Error; // 0x0090 (0x0008)
[0x00000000000002002] (CPF_Const | CPF_Transient)
class UAsyncTask* Task; // 0x0098 (0x0008)
[0x00000000000002000] (CPF_Transient)
struct FScriptDelegate __EventSuccess__Delegate; // 0x00A0
(0x0018) [0x000000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventFail__Delegate; // 0x00B8 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventComplete__Delegate; // 0x00D0
(0x0018) [0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{

```

```

uClassPointer = UObject::FindClass("Class ProjectX.RPC_X");
}

return uClassPointer;
};

void NotifySuccess();
void NotifyError(class UError* InError);
class UErrorType* eventOverrideErrorType(class UErrorType* ErrorType);
void eventOnComplete();
void eventOnFail();
void eventOnSuccess();
void eventDispose();
class UAsyncTask* CreateTask(struct FScriptDelegate InCallback);
class URPC_X* NotifyOnComplete(struct FScriptDelegate Callback);
class URPC_X* NotifyOnFail(struct FScriptDelegate Callback);
class URPC_X* NotifyOnSuccess(struct FScriptDelegate Callback);
class UObject* eventGetResponseObject();
void EventComplete(class URPC_X* RPC);
void EventFail(class URPC_X* RPC);
void EventSuccess(class URPC_X* RPC);
};

// Class ProjectX.RPC_CrashReport_X
// 0x0000 (0x00E8 - 0x00E8)
class URPC_CrashReport_X : public URPC_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_CrashReport_X");
}

return uClassPointer;
};

};

// Class ProjectX.RPC_RecordMetrics_X
// 0x0038 (0x00E8 - 0x0120)
class URPC_RecordMetrics_X : public URPC_X
{
public:
struct FGuid                                AppSessionID;                                // 0x00E8 (0x0010)
[0x0000000000000000]
struct FGuid                                LevelSessionID;                                // 0x00F8 (0x0010)
[0x0000000000000000]
float                                        CurrentTimeSeconds;                            // 0x0108 (0x0004)

```

```

[0x0000000000000000]
int32_t          FirstEventIndex;          // 0x010C (0x0004)
[0x0000000000000000]
TArray<struct FMetricsEvent>          Events;          // 0x0110 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_RecordMetrics_X");
}

return uClassPointer;
};

};

// Class ProjectX.RPCBatch_X
// 0x006C (0x0060 - 0x00CC)
class URPCBatch_X : public UObject
{
public:
float          SendTime;          // 0x0060 (0x0004)
[0x0000000000000000]
TArray<struct FPendingRPC>          Requests;          // 0x0068 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
TArray<struct FRPCResponse>          Responses;          // 0x0078 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
struct FRPCError          Error;          // 0x0088 (0x0020)
[0x0000000000040000] (CPF_NeedCtorLink)
class UObject*          Result;          // 0x00A8 (0x0008)
[0x0000000000000000]
class UPsyNetMessage_X*          RequestMessage;          // 0x00B0
(0x0008) [0x0000000000000000]
class UPsyNetMessage_X*          ResponseMessage;          // 0x00B8
(0x0008) [0x0000000000000000]
class UError*          ResponseError;          // 0x00C0 (0x0008)
[0x0000000000000000]
unsigned long          bUseRpcV2 : 1;          // 0x00C8 (0x0004)
[0x0000000000000000] [0x00000001]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPCBatch_X");
}
}

```



```

return uClassPointer;
};

};

// Class ProjectX.RPCQueue_X
// 0x0100 (0x0070 - 0x0170)
class URPCQueue_X : public UComponent
{
public:
    struct FPointer                Vftable_FTickableObject;                // 0x0070 (0x0008)
    [0x000000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
    float                          AuthTimeoutSeconds;                    // 0x0078 (0x0004)
    [0x00000000000000001] (CPF_Edit)
    class URPCConfig_X*            RPCConfig;                            // 0x0080 (0x0008)
    [0x00008000000000001] (CPF_Edit)
    class UPsyNetRetryConfig_X*    RetryConfig;                        // 0x0088 (0x0008)
    [0x00008000000000001] (CPF_Edit)
    class UPsyNetUrl_X*            PsyNetUrl;                            // 0x0090 (0x0008)
    [0x00008000000000001] (CPF_Edit)
    TArray<struct FPendingRPC>      PendingRPCs;                        // 0x0098 (0x0010)
    [0x000000000000400002] (CPF_Const | CPF_NeedCtorLink)
    TArray<class URPCBatch_X*>     PendingBatches;                      // 0x00A8
    (0x0010) [0x000000000000400000] (CPF_NeedCtorLink)
    struct FMap_Mirror             ServiceFailureDelayTimes;            // 0x00B8 (0x0050)
    [0x00000000000001002] (CPF_Const | CPF_Native)
    unsigned long                  bAuthorized : 1;                      // 0x0108 (0x0004)
    [0x00000000000000000] [0x000000001]
    struct FScriptDelegate         __EventRPCSuccess__Delegate;          // 0x0110
    (0x0018) [0x000000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate         __EventRPCError__Delegate;            // 0x0128
    (0x0018) [0x000000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate         __EventBatchError__Delegate;          // 0x0140
    (0x0018) [0x000000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate         __SendRequestDelegate__Delegate;      // 0x0158
    (0x0018) [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPCQueue_X");
        }

        return uClassPointer;
    };

    class FString __RPCQueue_X__CreateBatch_0x1(struct FPendingRPC P);
    void eventOnError(class URPC_X* RPC, class UError* Error);
    class UPsyNetMetrics_X* GetPsyNetMetrics();

```

```

void RecordBatchMetrics(class URPCBatch_X* Batch);
void HandleBatchComplete(class URPCBatch_X* Batch, class UPsyNetMessage_X* Response,
class UError* Error);
void eventCreateBatchSingleRPC(class UPsyNetMessage_X* Message, struct FPendingRPC&
RPC);
void eventCreateBatch(class UPsyNetMessage_X* Message, TArray<struct FPendingRPC>&
BatchRPCs);
static void eventSignatureMismatch(int32_t ServiceID);
void FailAllPending(class UError* Error);
void FailPending(class URPC_X* RPC, class UError* Error);
void TickReceive();
void TickSend();
void QueueRPC(class URPC_X* RPC);
class UAsyncResult__PsyNetMessage_X* SendRequestDelegate(class FString Service, class
UPsyNetMessage_X* Message);
void EventBatchError(class URPCQueue_X* RPCQueue, class UError* Error);
void EventRPCError(class URPCQueue_X* RPCQueue, class URPC_X* RPC, class UError* Error);
void EventRPCSuccess(class URPCQueue_X* RPCQueue, class URPC_X* RPC);
};

```

```

// Class ProjectX.RuntimeParameterBase_X
// 0x0008 (0x0060 - 0x0068)
class URuntimeParameterBase_X : public UObject
{
public:
    struct FName                                     ParameterName;                                // 0x0060 (0x0008)
    [0x000000000000000001] (CPF_Edit)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RuntimeParameterBase_X");
        }

        return uClassPointer;
    };
};

```

```

// Class ProjectX.RuntimeParameter_Speed_X
// 0x0000 (0x0068 - 0x0068)
class URuntimeParameter_Speed_X : public URuntimeParameterBase_X
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;
    };
};

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.RuntimeParameter_Speed_X");
}

return uClassPointer;
};

};

// Class ProjectX.RuntimeParameters_X
// 0x0000 (0x0060 - 0x0060)
class URuntimeParameters_X : public UObject
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RuntimeParameters_X");
        }

        return uClassPointer;
    };

};

// Class ProjectX.SeqAct_NonNativeUpdate_X
// 0x0000 (0x0160 - 0x0160)
class USeqAct_NonNativeUpdate_X : public USequenceAction
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.SeqAct_NonNativeUpdate_X");
        }

        return uClassPointer;
    };

    bool eventUpdate(float dt);
};

// Class ProjectX.SequenceCondition_X

```

```

// 0x0000 (0x0140 - 0x0140)
class USequenceCondition_X : public USequenceCondition
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.SequenceCondition_X");
}

return uClassPointer;
};

void eventActivated();
};

// Class ProjectX.Serializer_X
// 0x0000 (0x0060 - 0x0060)
class USerializer_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Serializer_X");
}

return uClassPointer;
};

void DeserializeObject(class UObject* Target, TArray<uint8_t>& InData);
void SerializeObject(class UObject* Target, TArray<uint8_t>& OutData);
};

// Class ProjectX.JSONSerializer_X
// 0x0004 (0x0060 - 0x0064)
class UJSONSerializer_X : public USerializer_X
{
public:
unsigned long                                     bDebug : 1;                                     // 0x0060 (0x0004)
[0x00000000000000001] [0x00000001] (CPF_Edit)

public:
static UClass* StaticClass()

```

```

{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.JSONSerializer_X");
}

return uClassPointer;
};

static bool Validate(class FString Stream);
static class FString func();
static void DecodeObject(class UObject* Target, class FString& Stream);
static void EncodeObject(class UObject* Target, class FString& Stream);
void DeserializeObject(class UObject* Target, TArray<uint8_t>& InData);
void SerializeObject(class UObject* Target, TArray<uint8_t>& OutData);
};

// Class ProjectX.ObjectSerializer_X
// 0x0004 (0x0060 - 0x0064)
class UObjectSerializer_X : public USerializer_X
{
public:
unsigned long                bPersistent : 1;                // 0x0060 (0x0004)
[0x0000000000000001] [0x00000001] (CPF_Edit)
unsigned long                bDebug : 1;                    // 0x0060 (0x0004)
[0x0000000000000001] [0x00000002] (CPF_Edit)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ObjectSerializer_X");
}

return uClassPointer;
};

void DeserializeObject(class UObject* Target, TArray<uint8_t>& InData);
void SerializeObject(class UObject* Target, TArray<uint8_t>& OutData);
};

// Class ProjectX.OnlineGame_Base_X
// 0x00D0 (0x00B0 - 0x0180)
class UOnlineGame_Base_X : public UOnline_X
{
public:
class FString                OnlineSubGameInterfaceName;    // 0x00B0 (0x0010)
[0x00000000000404001] (CPF_Edit | CPF_Config | CPF_NeedCtorLink)
class UOnlineGameSettings_X* DefaultGameSettingsArchetype; // 0x00C0

```

```

(0x0008) [0x0000000000000001] (CPF_Edit)
class UOnlineGameSettings_X*          DefaultGameSettings;          // 0x00C8
(0x0008) [0x0000000000000000]
class UOnlineGameSettings_X*          GameSettings;                // 0x00D0
(0x0008) [0x0000000000000200] (CPF_Transient)
class UOnlineGamePlaylists_X*         Playlists;                  // 0x00D8 (0x0008)
[0x0000000000000001] (CPF_Edit)
class UOnlineImageDownloaderWeb*      ImageDownloader;            // 0x00E0
(0x0008) [0x0000000000000000]
class UOnlineGameDLC_X*               DLC;                        // 0x00E8 (0x0008)
[0x0000000000000001] (CPF_Edit)
class UOnlineGameSkill_X*             Skill;                      // 0x00F0 (0x0008)
[0x0000000000000001] (CPF_Edit)
class FString                         StartServerCommand;          // 0x00F8 (0x0010)
[0x0000000000040003] (CPF_Edit | CPF_Const | CPF_NeedCtorLink)
class UWebCache_X*                    WebCache;                   // 0x0108 (0x0008)
[0x0000000000000000]
class UWebImageCache_X*               WebImageCache;              // 0x0110
(0x0008) [0x0000000000000000]
class UPsyNetStaticData_X*            PsyNetStaticData;           // 0x0118 (0x0008)
[0x000000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
unsigned long                         bLockGameSettings : 1;        // 0x0120 (0x0004)
[0x0000000000000200] [0x00000001] (CPF_Transient)
unsigned long                         bUseEnvironmentZone : 1;      // 0x0120 (0x0004)
[0x0000000000000400] [0x00000002] (CPF_Config)
uint8_t                              CurrentConnectionStatus;      // 0x0124 (0x0001)
[0x0000000400000200] (CPF_Transient)
class FString                         ServerName;                  // 0x0128 (0x0010)
[0x0000000800040200] (CPF_Transient | CPF_NeedCtorLink)
class FString                         CachedOptions;               // 0x0138 (0x0010)
[0x0000000800040200] (CPF_Transient | CPF_NeedCtorLink)
class UOnlineConfigDispatcher_X*      OnlineConfigDispatcher;      // 0x0148
(0x0008) [0x0000000000000200] (CPF_Transient)
class UOnlineGameWordFilter_X*        WordFilter;                 // 0x0150 (0x0008)
[0x0000000400000200] (CPF_Transient)
TArray<class UOnline_X*>               OnlineComponents;           // 0x0158 (0x0010)
[0x0000000800040000] (CPF_NeedCtorLink)
struct FScriptDelegate                __EventInternetConnectionChanged__Delegate; //
0x0168 (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGame_Base_X");
}

return uClassPointer;
};

void PrintDebugInfo(class UDebugDrawer* Drawer);

```

```

bool HasInternetConnection();
static int32_t GetBeaconPort();
static class FString GetBeaconExternalHostAddress(unsigned long bWithPort);
static class FString GetBeaconLocalHostAddress(unsigned long bWithPort);
static int32_t GetUdpPort();
static class FString GetUdpExternalHostAddress(unsigned long bWithPort);
static class FString GetUdpLocalHostAddress(unsigned long bWithPort);
void HandleInternetConnectionChanged(unsigned long bConnected);
void HandleConnectionStatusChanged(uint8_t ConnectionStatus);
void SetDefaultSearchType(class UClass* OnlineSearchClass);
void KickAllPlayers(class FString Reason);
void DelayedShutdownServer();
void OnExit();
int32_t GetPlaylistPlayerCount(class UGameSettingPlaylist_X* Playlist);
void OnNewSettingsChosen(int32_t PlaylistId);
void UpdateGameSettingsPlayerCount(int32_t NumPlayers);
void UpdateGameSettings(class UOnlineGameSettings_X* InGameSettings, class FString
Options);
class UOnlineGameSettings_X* CreateOnlineGameSettings(class FString Options);
void OnNewGameInfoCreated(class AGameInfo_X* Game);
void InitOnlineSubsystemInterfaces();
void OnInit();
void eventConstruct();
void EventInternetConnectionChanged(unsigned long bConnected);
};

```

```

// Class ProjectX.OnlineGame_X

```

```

// 0x0120 (0x0180 - 0x02A0)

```

```

class UOnlineGame_X : public UOnlineGame_Base_X

```

```

{
public:
class UOnlineGameAccount_X*          Account;                // 0x0180 (0x0008)
[0x00000000000000001] (CPF_Edit)
class UOnlineGameMatchmaking_X*      Matchmaking;           // 0x0188
(0x0008) [0x00000000000000001] (CPF_Edit)
class UOnlineGamePrivateMatch_X*     PrivateMatch;          // 0x0190
(0x0008) [0x00000000000000001] (CPF_Edit)
class UOnlineGameServerBrowser_X*    ServerBrowser;          // 0x0198
(0x0008) [0x00000000000000001] (CPF_Edit)
class UOnlineGameLeaderboards_X*     Leaderboards;           // 0x01A0
(0x0008) [0x00000000000000001] (CPF_Edit)
class UOnlineGamePopulation_X*       Population;              // 0x01A8 (0x0008)
[0x00000004000002000] (CPF_Transient)
class UOnlineGameVoice_X*            Voice;                   // 0x01B0 (0x0008)
[0x00000004000002000] (CPF_Transient)
class UOnlineGameParty_X*            Party;                   // 0x01B8 (0x0008)
[0x00000000000000001] (CPF_Edit)
class UOnlineGameInvite_X*           GamelInvite;             // 0x01C0 (0x0008)
[0x00000000000000001] (CPF_Edit)
class UOnlineGameJoinGame_X*         JoinGame;                // 0x01C8
(0x0008) [0x00000000000000001] (CPF_Edit)
class UOnlineGameRegions_X*          Regions;                 // 0x01D0 (0x0008)
[0x00000000000000001] (CPF_Edit)
class UOnlineGamePrivileges_X*       Privileges;              // 0x01D8 (0x0008)

```

```

[0x0000000000000000]
class UOnlineGameWordFilterProcessor_X*      WordFilterProcessor;          // 0x01E0
(0x0008) [0x0000004000002000] (CPF_Transient)
class UOnlineGameVersion_X*                  GameVersion;                  // 0x01E8 (0x0008)
[0x0000000000000001] (CPF_Edit)
class UOnlinePlayer_X*                       PlayerArchetype;              // 0x01F0 (0x0008)
[0x0000000000000001] (CPF_Edit)
TArray<class UOnlinePlayer_X*>                OnlinePlayers;                // 0x01F8 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class UOnlineLegalText_X*                    Eula;                             // 0x0208 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UOnlineLegalText_X*                    PrivacyPolicy;                // 0x0210 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UOnlineLegalText_X*                    ToS;                             // 0x0218 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UOnlineLegalText_X*                    SCT;                             // 0x0220 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UOnlineLegalText_X*                    PaymentServices;              // 0x0228 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
unsigned long                                bRequestPublicIP : 1;              // 0x0230 (0x0004)
[0x00000000000004000] [0x00000001] (CPF_Config)
class FString                                PublicIP;                          // 0x0238 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class UAvatarRequester_X*                    AvatarRequester;                  // 0x0248 (0x0008)
[0x0000000000000000]
class URemoteAvatarPermissions_X*             AvatarPermissions;                // 0x0250
(0x0008) [0x0000000000000000]
struct FScriptDelegate                       __EventConnectionError__Delegate; // 0x0258
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                       __EventPlayerPsyNetLogin__Delegate; // 0x0270
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                       __EventPlayerPsyNetLogout__Delegate; // 0x0288
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGame_X");
}

return uClassPointer;
};

void PrintDebugInfo(class UDebugDrawer* Drawer);
class UOnlinePlayer_X* GetOnlinePlayerFromEpicId(class FString EpicAccountId);
void HandlePublicIPComplete(class URPC_GetPublicIP_X* RPC);
void HandlePsyNetLogin(class UPsyNetConnection_X* Connection);
void NotifyPlayerPsyNetLogout(struct FScriptDelegate Callback);
void NotifyPlayerPsyNetLogin(struct FScriptDelegate Callback);
bool IsSignedIn(int32_t ControllerId);

```



```

bool CheckUpToDate(class UError*& Error);
bool CheckNotTooYoung(class UError*& Error);
class UError* GetPsyNetLoginError(class UOnlinePlayer_X* OnlinePlayer);
void HandleLoginCompleted(class UOnlinePlayerAuthentication_X* Auth, class UAsyncTask*
Task);
class UAsyncTask* CheckPsyNetConnection();
bool RequiresSystemNetworkErrorHandling();
bool ValidateUserOnlineAccount();
bool CheckInternetConnection(class UError*& Error);
bool IsOnlinePlayerID(struct FUniqueNetId PlayerID);
class UOnlinePlayer_X* eventGetOnlinePlayerByID(struct FUniqueNetId PlayerID);
void OnNewSettingsChosen(int32_t PlaylistId);
struct FUniqueNetId eventGetPrimaryPlayerID();
class UOnlinePlayer_X* eventGetPrimaryPlayer();
void RemoveOnlinePlayer(class ULocalPlayer_X* LocalPlayer);
void AddOnlinePlayer(class UOnlinePlayer_X* OnlinePlayer);
class UOnlinePlayer_X* CreateOnlinePlayer(class ULocalPlayer_X* LocalPlayer);
void SetPlayedWith(struct FUniqueNetId PlayerID);
void GetPlayerAvatar(struct FUniqueNetId PlayerID, struct FScriptDelegate
ReadOnlineAvatarCompleteDelegate, uint8_t Size);
void GetPlayerAvatars(struct FScriptDelegate ReadOnlineAvatarCompleteDelegate, uint8_t Size,
TArray<struct FUniqueNetId>& PlayerIds);
void ClearPendingAvatarDownloads();
void SetVoiceVolume(float NewValue);
void OnNewGameInfoCreated(class AGameInfo_X* Game);
void OnMainMenuOpened();
void HandlePlayerLogin(class UOnlinePlayer_X* OnlinePlayer, class UError* Error);
void HandlePsyNetLoginChanged(class UOnlinePlayerAuthentication_X* Auth);
void SyncLegal(class FString InAppendedPath);
bool IsLoggedInToPsyNet(class UOnlinePlayer_X* OnlinePlayer);
void OnInit();
void EventPlayerPsyNetLogout(class UOnlinePlayer_X* Player);
void EventPlayerPsyNetLogin(class UOnlinePlayer_X* Player);
void EventConnectionError(class UError* Error, class UOnlinePlayer_X* Player);
};

```

```

// Class ProjectX.OnlineGameServer_X
// 0x0008 (0x0180 - 0x0188)
class UOnlineGameServer_X : public UOnlineGame_Base_X
{
public:
class UPsyNetConnection_X*          PsyNetConnection;          // 0x0180
(0x0008) [0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameServer_X");
}
}

```

```
return uClassPointer;
};
```

```
void OnInit();
};
```

```
// Class ProjectX.OnlineGameDedicatedServer_X
// 0x0258 (0x0188 - 0x03E0)
class UOnlineGameDedicatedServer_X : public UOnlineGameServer_X
{
public:
class FString                                GameServerID;                                // 0x0188 (0x0010)
[0x0000008000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                                GameServerHost;                                // 0x0198 (0x0010)
[0x0000008000402000] (CPF_Transient | CPF_NeedCtorLink)
int32_t                                     EnvironmentZone;                                // 0x01A8 (0x0004)
[0x0000008000002000] (CPF_Transient)
class UOnlineGameStats_X*                    GameStats;                                    // 0x01B0 (0x0008)
[0x0000004000002000] (CPF_Transient)
class UOnlineGameDedicatedServerRegistration_X* Registration;                            // 0x01B8
(0x0008) [0x0000004000000001] (CPF_Edit)
class UOnlineGameReservations_X*            Reservations;                            // 0x01C0
(0x0008) [0x0000004000000001] (CPF_Edit)
class UClanforgeReservation_X*              Clanforge;                                // 0x01C8 (0x0008)
[0x0001004000002000] (CPF_Transient)
class FString                                Region;                                    // 0x01D0 (0x0010)
[0x0000008000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                                DataCenter;                                // 0x01E0 (0x0010)
[0x0000008000402000] (CPF_Transient | CPF_NeedCtorLink)
int32_t                                     CurrentPlaylistId;                            // 0x01F0 (0x0004)
[0x0000008000000000]
int32_t                                     MutatorIndex;                                // 0x01F4 (0x0004)
[0x0000008000000000]
float                                        AverageMMR;                                    // 0x01F8 (0x0004)
[0x0000008000002000] (CPF_Transient)
float                                        AverageConservativeMMR;                        // 0x01FC (0x0004)
[0x0000008000002000] (CPF_Transient)
int32_t                                     MachineId;                                // 0x0200 (0x0004)
[0x0000004000002000] (CPF_Transient)
struct FCustomMatchSettings                  CustomMatch;                                // 0x0208
(0x0090) [0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FUniqueNetId                          CustomMatchOwner;                            // 0x0298 (0x0048)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
uint64_t                                    CustomMatchClubId;                            // 0x02E0 (0x0008)
[0x0001004000002000] (CPF_Transient)
unsigned long                               bQueuedShutdown : 1;                        // 0x02E8 (0x0004)
[0x0000004000002000] [0x00000001] (CPF_Transient)
unsigned long                               bFindingReplacementServer : 1;              // 0x02E8 (0x0004)
[0x0000004000002000] [0x00000002] (CPF_Transient)
unsigned long                               bHasValidMigrationServer : 1;                // 0x02E8 (0x0004)
[0x0000008000000000] [0x00000004]
unsigned long                               bDisableMatchmakingBan : 1;                 // 0x02E8 (0x0004)
[0x0000004000002000] [0x00000008] (CPF_Transient)
unsigned long                               bBotTest : 1;                                // 0x02E8 (0x0004)
```

```

[0x0000004000002000] [0x00000010] (CPF_Transient)
class FString ReplacementServerID; // 0x02F0 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class UServerPlayerTracker_X* PlayerTracker; // 0x0300 (0x0008)
[0x0000004004082008] (CPF_ExportObject | CPF_Transient | CPF_Component | CPF_EditInline)
class UServerExploitManager_X* ExploitManager; // 0x0308
(0x0008) [0x0000000000002000] (CPF_Transient)
class UClass* MatchRecorderClass; // 0x0310 (0x0008)
[0x0000000000000001] (CPF_Edit)
class UMatchRecorder_X* MatchRecorder; // 0x0318 (0x0008)
[0x0000008000002000] (CPF_Transient)
class UMatchLog_X* MatchLog; // 0x0320 (0x0008)
[0x0000000000002000] (CPF_Transient)
class UServerConfig_X* Config; // 0x0328 (0x0008)
[0x0000804000002001] (CPF_Edit | CPF_Transient)
class UPsyNetConfig_X* PsyNetConfig; // 0x0330 (0x0008)
[0x0000800000002000] (CPF_Transient)
struct FScriptDelegate
__EventFoundNewDedicatedServerForPlayers__Delegate; // 0x0338 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventAverageMMRChanged__Delegate; // 0x0350
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventPlaylistSet__Delegate; // 0x0368
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventPrivateMatchSettingsChanged__Delegate; //
0x0380 (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventInactive__Delegate; // 0x0398 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventActive__Delegate; // 0x03B0 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventMatchGUIDChanged__Delegate; // 0x03C8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameDedicatedServer_X");
}

return uClassPointer;
};

void HandlePsyNetDataLoaded(class UPsyNetStaticData_X* D);
void PrintDebugInfo(class UDebugDrawer* Drawer);
class URPC_RecordMatch_X* SendRecordMatchRPC();
void ReportMatch();
void HandleTrackerPlayerRemoved(class UServerPlayerTracker_X* Tracker, struct FUniqueNetId
PlayerID);
void CreateMatchRecorder();
void HandleTrackerPlayerAdded(class UServerPlayerTracker_X* Tracker, struct FUniqueNetId

```

```

PlayerID);
class FString MatchGuid();
struct FName GetCurrentMapName();
void ClearMatchmakingBan(struct FUniqueNetId PlayerID);
void IssueMatchmakingBan(struct FUniqueNetId PlayerID, class FString Reason);
uint8_t GetTeamNum(struct FUniqueNetId PlayerID);
bool KickPlayersOnGameEventDestroyed();
bool IsBotMatch();
bool CanIssueMatchmakingBan();
bool IsRankedMatch();
class UGameSettingPlaylist_X* GetPlaylist();
void OnExit();
void OnMigrationStarted();
bool DidRematchVotePass();
void HandleNewServerIPRPC(class URPC_CheckReplacementDedicatedServer_X* RPC);
void CheckNewServerIP();
void CheckNewServerTimeout();
void CheckStartShutdown(int32_t GameTimeRemaining);
bool AllowServerMigration();
void UpdateGameTime(int32_t TimeSeconds);
void FindNewDedicatedServerForPlayers(struct FScriptDelegate InDelegate);
bool IsInactive();
int32_t GetTeamScore(int32_t TeamIdx);
bool IsInPostGameLobby();
void UpdateOnlineGame();
bool AllowSplitscreenJoinRankedMatch();
bool AllowSplitscreenJoin(struct FUniqueNetId PrimaryPlayerId, struct FUniqueNetId PlayerID,
class FString PlayerName, class FString& Error);
void PlayerLoggedOut(class APlayerReplicationInfo* PRI);
void PlayerLoggedIn(class APlayerReplicationInfo* PRI);
void AllowPlayerLogin(class FString Options, struct FUniqueNetId PlayerID, class FString&
ErrorMessage);
void OnNewGameInfoCreated(class AGameInfo_X* Game);
bool AllPlayersHaveNetworkIssues();
void TravelToMap(class FString ServerCommand);
void GotoPrivateMatchMap(struct FName MapName, int32_t GameMode, class FString
GameTags);
void GotoNextPrivateMatchMap();
void OnPrivateEventDestroyed();
void GoInactive();
class FString GetStartServerCommand();
struct FName GetNextRandomMapName(int32_t GameMode, unsigned long bStandardOnly,
unsigned long bRocketLabsMedleyOnly);
struct FName GetNextMapName(int32_t GameMode, unsigned long bStandardOnly, unsigned
long bRocketLabsMedleyOnly);
struct FCustomMatchSettings GetCustomMatchSettings();
struct FUniqueNetId GetCustomMatchOwner();
void UpdateCustomMatchOwner(struct FUniqueNetId NewOwner);
void SetCustomSettings(struct FCustomMatchSettings Settings);
void SetupPrivateMatch(struct FCustomMatchSettings Settings, struct FUniqueNetId Creator);
void GoToNextMap();
struct FName GetNextMutatorMapName();
void GoToMap(class FString MapName);
class FString GetPlaylistTags();

```

```

void UpdateAverageMMR();
void SetCustomMatchClubID(uint64_t InClubID);
void ClearSettings();
void SetPlaylist(int32_t PlaylistId);
void HandleReservationsUpdated();
void InitClanforge();
void OnInit();
void EventMatchGUIDChanged(class UOnlineGameDedicatedServer_X* Server);
void EventActive(class UOnlineGameDedicatedServer_X* Server);
void EventInactive(class UOnlineGameDedicatedServer_X* Server);
void EventPrivateMatchSettingsChanged(class UOnlineGameDedicatedServer_X* Server, struct
FCustomMatchSettings NewSettings);
void EventPlaylistSet(class UOnlineGameDedicatedServer_X* Server);
void EventAverageMMRChanged(class UOnlineGameDedicatedServer_X* Server);
void EventFoundNewDedicatedServerForPlayers(class UConnectionInfoMessage_X* Message);
};

// Class ProjectX.OnlineGameDLC_X
// 0x0040 (0x00B0 - 0x00F0)
class UOnlineGameDLC_X : public UOnline_X
{
public:
int32_t SyncInterval; // 0x00B0 (0x0004)
[0x0000000000000000] (CPF_Edit)
float PlayerOwnershipSyncInterval; // 0x00B4 (0x0004)
[0x0000000000000000]
TArray<struct FPlayerDLCInfo> PlayerDLCInfos; // 0x00B8 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
TArray<struct FSteamWebRequestData> SteamWebRequests; // 0x00C8
(0x0010) [0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FScriptDelegate __EventValidationReady__Delegate; // 0x00D8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameDLC_X");
}

return uClassPointer;
};

bool DoesPlayerOwnDLC(struct FUniqueNetId& PlayerID, struct FName& DLCName);
void HandleGetPublisherAppOwnership(class UWebRequest_X* Request);
static bool PlatformRequiresDLCVerification(uint8_t Platform);
void OnMatchFinished();
void SyncPlayerDLC(struct FUniqueNetId PlayerID, struct FScriptDelegate Callback);
void UpdatePlayerDLCInfo(int32_t PlayerInfoIndex);
int32_t CreatePlayerDLCInfo(struct FUniqueNetId& PlayerID);
bool IsPlayerReadyForValidation(struct FUniqueNetId& PlayerID);

```

```

void EventValidationReady();
};

// Class ProjectX.OnlineGameLeaderboards_X
// 0x0090 (0x00B0 - 0x0140)
class UOnlineGameLeaderboards_X : public UOnline_X
{
public:
    int32_t NumResults; // 0x00B0 (0x0004)
    [0x0000000000000001] (CPF_Edit)
    TArray<struct FCachedLeaderboardData> CachedLeaderboards; // 0x00B8
    (0x0010) [0x0000000000040200] (CPF_Transient | CPF_NeedCtorLink)
    int32_t NumFriendsLeaderboardRequests; // 0x00C8 (0x0004)
    [0x0000000000000200] (CPF_Transient)
    struct FName CurrentLeaderboardID; // 0x00CC (0x0008)
    [0x0000000000000200] (CPF_Transient)
    int32_t CurrentFriendIndex; // 0x00D4 (0x0004)
    [0x0000000000000200] (CPF_Transient)
    unsigned long bGlobalLeaderboards : 1; // 0x00D8 (0x0004)
    [0x0000000000000200] [0x00000001] (CPF_Transient)
    class URPC_X* GetLeaderboardPlayerValueRPC; // 0x00E0
    (0x0008) [0x0000000000000200] (CPF_Transient)
    TArray<class URPC_X*> LeaderboardRequests; // 0x00E8
    (0x0010) [0x0000000000040200] (CPF_Transient | CPF_NeedCtorLink)
    struct FScriptDelegate __OnGetLeaderboardPlayerValue__Delegate; // 0x00F8
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
    struct FScriptDelegate __EventNewLeaderboard__Delegate; // 0x0110
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
    struct FScriptDelegate __EventGetLeaderboardFailed__Delegate; // 0x0128
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameLeaderboards_X");
        }

        return uClassPointer;
    };

    void __OnlineGameLeaderboards_X__ClearPreviousTasks_0x1(class URPC_X* RPC);
    void ClearPreviousTasks();
    bool IsSkillLeaderboard(class FString NewLeaderboardID, int32_t& Playlist);
    TArray<struct FLeaderboardData> SortLeaderboardDataMMR(TArray<struct FLeaderboardData>
    Leaderboard);
    TArray<struct FLeaderboardData> SortLeaderboardData(TArray<struct FLeaderboardData>
    Leaderboard);
    void HandleGetLeaderboardFriendsRPC(class URPC_GetLeaderboardRankForUsersBase_X*
    RPC);
    void DispatchLeaderboardFriendsRequest();

```

```

void GetLeaderboardsForAllFriends(struct FName LeaderboardId);
void GetLeaderboardFriends(class UOnlinePlayer_X* OnlinePlayer, struct FName
NewLeaderboardID, struct FScriptDelegate NewDelegate, struct FScriptDelegate FailedDelegate);
void RequestClearPendingAvatarDownloads();
void HandleGetLeaderboardRPC(class URPC_GetLeaderboardBase_X* RPC);
void GetLeaderboard(struct FName NewLeaderboardID, unsigned long bGlobal, struct
FScriptDelegate NewDelegate, struct FScriptDelegate FailedDelegate);
bool GetCachedLeaderboardData(struct FName LeaderboardId, TArray<struct
FLeaderboardData>& LeaderboardDataList);
int32_t CalcDivision(float TopValue, float Value);
void GetRankForPrimaryPlayer(struct FName LeaderboardId, int32_t Value, int32_t& Rank,
int32_t& Division);
void HandleGetLeaderboardPlayerValueRPC(class URPC_GetLeaderboardValueForUserBase_X*
RPC);
void GetLeaderboardPlayerValue(struct FName NewLeaderboardID, struct FScriptDelegate
NewDelegate);
void EventGetLeaderboardFailed(class UError* Error);
void EventNewLeaderboard(struct FName NewLeaderboardID, TArray<struct
FLeaderboardData>& NewLeaderboardData);
void OnGetLeaderboardPlayerValue(class UOnlineGameLeaderboards_X* LeaderboardsRef,
struct FName LeaderboardId, class UError* Error, struct FLeaderboardData&
PlayerLeaderboardData);
};

```

```

// Class ProjectX.OnlineGameReservations_X
// 0x0118 (0x00B0 - 0x01C8)
class UOnlineGameReservations_X : public UOnline_X
{
public:
float                               ExtraMapLoadTime;                               // 0x00B0 (0x0004)
[0x00000000000000001] (CPF_Edit)
float                               WaitForReservationsTimeout;                     // 0x00B4 (0x0004)
[0x00000000000000001] (CPF_Edit)
float                               InitialReservationTimeoutSeconds;                 // 0x00B8 (0x0004)
[0x00000000000000001] (CPF_Edit)
float                               InitialMigrationTimeoutSeconds;                 // 0x00BC (0x0004)
[0x00000000000000001] (CPF_Edit)
unsigned long                       bAllowPrivateMatchCrossPlayDisable : 1;         // 0x00C0
(0x0004) [0x00000000000000001] [0x00000001] (CPF_Edit)
unsigned long                       bSkipReservationCheck : 1;                     // 0x00C0 (0x0004)
[0x0000004000002000] [0x00000002] (CPF_Transient)
unsigned long                       bSinglePlayerMatchmaking : 1;                 // 0x00C0 (0x0004)
[0x0000004000002000] [0x00000004] (CPF_Transient)
unsigned long                       bServerTraveling : 1;                         // 0x00C0 (0x0004)
[0x0000004000002000] [0x00000008] (CPF_Transient)
unsigned long                       bScrambleTeams : 1;                           // 0x00C0 (0x0004)
[0x00000000000002000] [0x00000010] (CPF_Transient)
unsigned long                       bMatchStarted : 1;                            // 0x00C0 (0x0004)
[0x0008000000002000] [0x00000020] (CPF_Transient)
unsigned long                       bLockTeams : 1;                               // 0x00C0 (0x0004)
[0x00000000000002000] [0x00000040] (CPF_Transient)
unsigned long                       bIsBotMatch : 1;                              // 0x00C0 (0x0004)
[0x0000004000002000] [0x00000080] (CPF_Transient)
unsigned long                       bMigrationInProgress : 1;                     // 0x00C0 (0x0004)

```

```

[0x0000008000002000] [0x00000100] (CPF_Transient)
class UReservationBeacon_X* Beacon; // 0x00C8 (0x0008)
[0x0000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UUDPpingBeaconServer_X* PingBeacon; // 0x00D0
(0x0008) [0x0000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
TArray<struct FReservationData> Players; // 0x00D8 (0x0010)
[0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<struct FTeamPairHistory> TeamPairHistories; // 0x00E8
(0x0010) [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<class FString> BotNames; // 0x00F8 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class UCrossplayConfig_X* Crossplay; // 0x0108 (0x0008)
[0x0000800000000001] (CPF_Edit)
class UMapPrefsConfig_X* MapPrefsConfig; // 0x0110 (0x0008)
[0x0000800000000001] (CPF_Edit)
class UBeaconConfig_X* BeaconConfig; // 0x0118 (0x0008)
[0x0000800000000001] (CPF_Edit)
TArray<struct FPendingReservation> PendingReservations; // 0x0120
(0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
class UTAsyncResult__ClubDetails_X* GetClubTask; // 0x0130
(0x0008) [0x0001000000000000]
class FString RankedMatchReservationID; // 0x0138 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
TArray<struct FUniqueNetId> RankedMatchPushedPlayerIDs; // 0x0148
(0x0010) [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
class UOnlineClubProvider_X* Clubs; // 0x0158 (0x0008)
[0x0001800000000000]
class FString JoinName; // 0x0160 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
class FString JoinPassword; // 0x0170 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventReservationsUpdated__Delegate; // 0x0180
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventMigrationJoinOccurred__Delegate; // 0x0198
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __bMatchStarted__ChangeNotify; // 0x01B0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameReservations_X");
}

return uClassPointer;
};

void __OnlineGameReservations_X__OnInit_0x3(class UIReservationConnection_X* Connection,
class UObject* Message);
void __OnlineGameReservations_X__OnInit_0x2(class UIReservationConnection_X* Connection,

```



```

class UObject* Message);
void __OnlineGameReservations_X__OnInit_0x1(class UIReservationConnection_X* Connection,
class UObject* Message);
bool __OnlineGameReservations_X__SetPlayersWithMigrationData_0x2(struct
FMigrationReservationData P);
struct FUniqueNetId
__OnlineGameReservations_X__HandlePsyNetBeaconReservation_0x1(struct
FPsyNetBeaconPlayerReservation P);
void __OnlineGameReservations_X__HandlePublicReservation_0x2(struct FReservationData P);
void __OnlineGameReservations_X__HandlePublicReservation_0x1(struct FReservationData P);
struct FUniqueNetId __OnlineGameReservations_X__IsFull_0x2(struct FReservationData P);
bool __OnlineGameReservations_X__IsFull_0x1(struct FReservationData P);
struct FUniqueNetId __OnlineGameReservations_X__HasTimeoutPlayers_0x2(struct
FReservationData P);
bool __OnlineGameReservations_X__HasTimeoutPlayers_0x1(struct FReservationData P);
bool __OnlineGameReservations_X__HasMultipleReservedTeams_0x2(int32_t TeamID);
int32_t __OnlineGameReservations_X__HasMultipleReservedTeams_0x1(struct
FReservationData P);
bool __OnlineGameReservations_X__AllPlayersReserved_0x1(struct FReservationData P);
struct FUniqueNetId __OnlineGameReservations_X__AllPlayersInGame_0x2(struct
FReservationData P);
bool __OnlineGameReservations_X__AllPlayersInGame_0x1(struct FReservationData P);
void __OnlineGameReservations_X__SetServerTraveling_0x1(struct FPendingReservation P);
struct FUniqueNetId __OnlineGameReservations_X__GetPlayerIDs_0x1(struct FReservationData
Player);
void __bMatchStarted__ChangeNotifyFunc();
TArray<struct FUniqueNetId> GetPlayerIDs();
float GetMapLoadTimeout(uint8_t Platform);
void GetAccumulatedPrefs(TArray<struct FName>& Likes, TArray<struct FName>& Dislikes);
void PrintDebugInfo(class UDebugDrawer* Drawer);
class UReservationsMetrics_X* GetMetrics();
void OnNewGameInfoCreated(class AGameInfo_X* Game);
bool HasSplitScreenPlayers();
void SetParty(struct FUniqueNetId MemberId, struct FUniqueNetId PartyID);
struct FUniqueNetId GetPartyID(struct FUniqueNetId PlayerID);
uint8_t GetTeamNum(struct FUniqueNetId PlayerID);
class FString GetPlayersPlatformsString();
class FString GetPlatformsString(TArray<uint8_t>& Platforms);
TArray<uint8_t> GetPlayerPlatforms();
bool CanPlayTogether(uint8_t Platform1, uint8_t Platform2);
bool CanAcceptReservationsForPlatform(class UAddReservationMessagePublic_X* Message);
bool ShouldCheckPlatformRestrictions(class UAddReservationMessagePublic_X* Message);
uint8_t GetExclusivePlatform();
bool CanAcceptReservations(class UAddReservationMessagePublic_X* Message);
void GetPlayerCounts(int32_t& Team0Reserved, int32_t& Team0Joined, int32_t&
Team1Reserved, int32_t& Team1Joined);
void NotifyReservationsUpdated();
void OnReservationsUpdated();
void CheckMapLoadTimeout();
void StartMapLoadTimeout();
void SyncSkills();
void HackForceNotServerTravelling();
void SwapTeams();
void SetServerTraveling(unsigned long bTraveling);

```

```

void CloseConnection(int32_t Index);
void RemoveReservationIndex(int32_t Index);
int32_t AddSpecialReservation(struct FUniqueNetId PlayerID, class FString PlayerName, struct
FUniqueNetId PartyID);
void HandleTeamChanged(class APRI_X* PRI);
void RemoveReservation(struct FUniqueNetId PlayerID);
void PlayerLoggedOut(class APlayerReplicationInfo* PRI);
void PlayerLoggedIn(class APlayerReplicationInfo* PRI);
void AllowPlayerLogin(class FString Options, struct FUniqueNetId PlayerID, class FString&
ErrorMessage);
bool AllPlayersInGame(TArray<struct FUniqueNetId>& OutPlayersInGame);
bool AllPlayersReserved();
bool HasGameEnded();
bool HasGameStarted();
bool CanStartMatch();
bool HasMultipleReservedTeams();
bool HasTimeoutPlayers(TArray<struct FUniqueNetId>& OutTimeoutPlayers);
bool HasJoiningPlayers();
bool CanGoInactive();
bool IsEmpty();
bool IsNearlyFull();
bool IsFull(TArray<struct FUniqueNetId>& OutPlayersReserved);
int32_t GetNumReservedPlayers();
bool IsCustomMatch();
void HandleReservationDisconnected(class UReservationBeacon_X* _, class
UIReservationConnection_X* Connection);
void ClearTeamHistory();
void ClearReservations();
void Reset();
static void RecordTeamHistory(TArray<struct FReservationData>& TestPlayers, TArray<struct
FTeamPairHistory>& History);
void ScrambleTeams();
void UpdateTeams();
void AssignTeamsByPartySize();
static bool AssignScrambledTeams(int32_t TeamSize, TArray<struct FTeamPairHistory>&
History, TArray<struct FReservationData>& TestPlayers);
static bool AssignTeams(int32_t TeamSize, unsigned long bKeepTeams, TArray<struct
FReservationData>& TestPlayers);
void HandleSkillSynced(class UOnlineGameSkill_X* SkillSystem, struct FUniqueNetId PlayerID,
class UError* Error);
bool AllowSplitscreenJoin(struct FUniqueNetId PrimaryPlayerId, struct FUniqueNetId PlayerID,
class FString PlayerName, class FString& Error);
bool IsCheatingSplitscreenReservation(class UAddReservationMessage_X* Message);
bool IsCheatingSplitscreen(struct FUniqueNetId PrimaryPlayerId, struct FUniqueNetId
SplitscreenPlayerId);
struct FReservationData CreateReservationData(struct FUniqueNetId PlayerID, class FString
PlayerName, struct FUniqueNetId PartyID, uint8_t Status);
void HandlePlayerCancel(class UIReservationConnection_X* Connection, class UObject*
Message);
struct FUniqueNetId GetConnectionPlayerID(class UIReservationConnection_X* Connection);
void InitialReservationTimeout();
void NotAllPlayersJoined();
void DisconnectExistingPlayers(class UAddReservationMessage_X* ReservationMessage);
void SetPlayers(TArray<struct FReservationData>& TempPlayers);

```

```

void HandleClubSyncComplete(class UOnlineClubProvider_X* _);
TArray<struct FReservationData> AddPlayersFromReservationMessage(class
UAddReservationMessage_X* ReservationMessage, class UIReservationConnection_X*
Connection);
bool IsUnique(TArray<struct FName>& Prefs);
void ForcePrivateMatch(struct FCustomMatchSettings InSettings);
void SetClubServer(class UClubDetails_X* Details);
void StartPrivateMatch(struct FCustomMatchSettings MatchSettings, struct FUniqueNetId
Creator);
bool HandlePrivateReservation(class UIReservationConnection_X* Connection, class
UAddReservationMessagePrivate_X* Message);
void ProcessReservationMessage(class UIReservationConnection_X* Connection, class
UAddReservationMessage_X* Message);
void StartMatch();
class UReservationsReadyMessage_X* CreateReadyMessage();
void SendReadyMessage(struct FUniqueNetId PlayerID, class UReservationsReadyMessage_X*
Message, class UIReservationConnection_X* Connection);
void NotifyReady(struct FUniqueNetId PlayerID, class UIReservationConnection_X* Connection);
void BroadcastReady();
bool ShouldSyncSkills();
bool IsSoloPlaylist(int32_t PlaylistId);
bool ShouldCheckRankedMatchReservationID();
bool CheckRankedMatchReservation(struct FUniqueNetId PlayerID, class FString&
InReservationID);
bool HandlePublicReservation(class UIReservationConnection_X* Connection, class
UAddReservationMessagePublic_X* Message);
void HandleBeaconReservationMessage(class UIReservationConnection_X* Connection, class
UAddReservationMessage_X* Message);
bool IsCheatingReservationMessage(class UIReservationConnection_X* Connection, class
UAddReservationMessage_X* Message);
void RecordReservation(class UAddReservationMessage_X* Message);
void HandleRegionPingMessage(class UIReservationConnection_X* Connection, class
URegionPingMessage_X* Message);
void HandlePingMessage(class UIReservationConnection_X* Connection, class
UPingMessage_X* Message);
void HandleMatchStartedChanged();
class UObject* CreateClientReservationMessage(struct FServerReservationData& Reservation,
struct FUniqueNetId& PlayerID);
void HandlePsyNetBeaconReservation(struct FPsyNetBeaconReservation Reservation, float
ConnectionTimeoutSeconds);
void NotifyMigrationStarted();
void SetPlayersWithMigrationData(TArray<struct FMigrationReservationData> MigratedPlayers);
TArray<struct FMigrationReservationData> GetMigrationReservationData();
class UServerToServerMessage_X* GetMatchInfoMessage();
void NotifyNewReplacementServer();
void HandleNewServerConnectionInfo(class UConnectionInfoMessage_X* Message);
void HandleMigrationStartedMessage(class UMigrationStartedMessage_X* Message);
void SendMigratedServerConnectionInfo(class FString ServerId);
void SetJoinNameAndPassword(class FString InJoinName, class FString InJoinPassword);
void HandleServerMigrationMessage(class UMatchInfoMessage_X* Message);
void HandleReconnectReservation(class UPsyNetService_Reconnect_X* Notification);
void HandleFriendJoinReservation(class UPsyNetService_FriendJoin_X* Notification);
void HandleJoinPrivateReservation(class UPsyNetService_JoinPrivate_X* Notification);
void HandleCreatePrivateReservation(class UPsyNetService_CreatePrivate_X* Notification);

```

```

void HandleBackfillReservation(class UPsyNetService_Backfill_X* Notification);
void HandleNewGameReservation(class UPsyNetService_NewGame_X* Notification);
void StartBeacons();
void OnInit();
void EventMigrationJoinOccurred();
void EventReservationsUpdated();
};

```

```

// Class ProjectX.OnlineGameStats_X
// 0x0014 (0x00B0 - 0x00C4)
class UOnlineGameStats_X : public UOnline_X
{
public:
class FString                                MatchGuid;                                // 0x00B0 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
int32_t                                    PlaylistId;                                // 0x00C0 (0x0004)
[0x0000000000000000]

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameStats_X");
}

```

```

return uClassPointer;
};

```

```

void UploadStats(class FString StatType, TArray<struct FUploadStatDataSet>& DataSet);
void SetPlaylistID(int32_t InID);
void SetMatchGUID(class FString InGuid);
};

```

```

// Class ProjectX.TickableStateObject_X
// 0x0028 (0x0064 - 0x008C)
class UTickableStateObject_X : public UStateObject_X
{
public:
struct FPointer                                VfTable_FTickableObject;                                // 0x0068 (0x0008)
[0x000000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
unsigned long                                bTickWhenGamePaused : 1;                                // 0x0070 (0x0004)
[0x00000000000000001] [0x000000001] (CPF_Edit)
unsigned long                                bAutoTick : 1;                                // 0x0070 (0x0004)
[0x00000000000000003] [0x000000002] (CPF_Edit | CPF_Const)
TArray<struct FTimerData>                    Timers;                                // 0x0078 (0x0010)
[0x000000000000400002] (CPF_Const | CPF_NeedCtorLink)
float                                LatentFloat;                                // 0x0088 (0x0004)
[0x00000000000000002] (CPF_Const)

```

```

public:
static UClass* StaticClass()

```

```

{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.TickableStateObject_X");
}

return uClassPointer;
};

void Sleep(float Seconds);
void eventTick(float DeltaTime);
void ProcessState(float DeltaSeconds);
void UpdateTimers(float DeltaSeconds);
void NativeTick(float DeltaTime);
void Destroy();
void SetTickable(unsigned long bTick);
bool IsTimerActive(struct FName inTimerFunc, class UObject* inObj);
void PauseTimer(unsigned long bPause, struct FName inTimerFunc, class UObject* inObj);
void ClearAllTimers(class UObject* inObj);
void ClearTimer(struct FName inTimerFunc, class UObject* inObj);
void SetStateTimer(float InRate, unsigned long inbLoop, struct FName inTimerFunc);
void SetTimer(float InRate, unsigned long inbLoop, struct FName inTimerFunc, class UObject*
inObj);
};

// Class ProjectX.SystemInfo_X
// 0x0000 (0x0060 - 0x0060)
class USystemInfo_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.SystemInfo_X");
}

return uClassPointer;
};

static class FString GetComputerName();
static void GetNetworkInfo(TArray<class FString>& Types);
static void GetOSInfo(class FString& Type, class FString& Bit);
static void GetVideoCardInfo(class FString& CardName, int32_t& RAM);
static void GetMemoryInfo(float& Physical, float& PageFile, float& Virtual, int32_t& Used, int32_t&
Allocated);
static void GetCPUInfo(class FString& Type, class FString& Description, class FString& Cores,
class FString& Threads);

```

```

static bool IsSupported();
};

// Class ProjectX.TickActorComponent_X
// 0x0033 (0x009D - 0x00D0)
class UTickActorComponent_X : public UActorComponent
{
public:
TArray<struct FTimerData> Timers; // 0x00A0 (0x0010)
[0x00000000000400002] (CPF_Const | CPF_NeedCtorLink)
unsigned long bTick : 1; // 0x00B0 (0x0004)
[0x00000000000000002] [0x000000001] (CPF_Const)
struct FScriptDelegate __EventTick__Delegate; // 0x00B8 (0x0018)
[0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.TickActorComponent_X");
}

return uClassPointer;
};

void SetTickable(unsigned long bWantsTick);
void ClearAllTimers(class UObject* inObj);
void ClearTimer(struct FName inTimerFunc, class UObject* inObj);
void SetTimer(float InRate, unsigned long inbLoop, struct FName inTimerFunc, class UObject*
inObj);
void EventTick(float DeltaTime);
};

// Class ProjectX.Timers_X
// 0x0020 (0x0060 - 0x0080)
class UTimers_X : public UObject
{
public:
class UTickComponent_X* TickComponent; // 0x0060 (0x0008)
[0x000000000004082008] (CPF_ExportObject | CPF_Transient | CPF_Component | CPF_EditInline)
struct FScriptDelegate __TickDelegate__Delegate; // 0x0068 (0x0018)
[0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Timers_X");
}
}

```

```

}

return uClassPointer;
};

static void ClearTick(struct FScriptDelegate Callback);
static void SetTick(struct FScriptDelegate Callback);
static bool IsActive(struct FScriptDelegate Callback);
static float GetElapsedTime(struct FScriptDelegate Callback);
static float GetRemainingTime(struct FScriptDelegate Callback);
static float GetRate(struct FScriptDelegate Callback);
static void ClearAll(class UObject* Obj);
static void Clear(struct FScriptDelegate Callback);
static void SetStateTimer(struct FScriptDelegate Callback, float Rate, unsigned long bLoop);
static void Set(struct FScriptDelegate Callback, float Rate, unsigned long bLoop);
void TickDelegate(float DeltaTime);
};

// Class ProjectX.TimersComponent_X
// 0x0010 (0x0070 - 0x0080)
class UTimersComponent_X : public UComponent
{
public:
	TArray<struct FTimerData> Timers; // 0x0070 (0x0010)
[0x0000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.TimersComponent_X");
}

return uClassPointer;
};

void UpdateTimers(float DeltaSeconds);
bool IsActive(struct FScriptDelegate Callback);
float GetElapsedTime(struct FScriptDelegate Callback);
float GetRemainingTime(struct FScriptDelegate Callback);
float GetRate(struct FScriptDelegate Callback);
void ClearAll(class UObject* Obj);
void Clear(struct FScriptDelegate Callback);
void SetStateTimer(struct FScriptDelegate Callback, float Rate, unsigned long bLoop);
void Set(struct FScriptDelegate Callback, float Rate, unsigned long bLoop);
};

// Class ProjectX.TickComponent_X
// 0x0028 (0x0080 - 0x00A8)
class UTickComponent_X : public UTimersComponent_X
{

```

```

public:
struct FPointer          VfTable_FTickableObject;          // 0x0080 (0x0008)
[0x000000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
unsigned long           bTickWhenGamePaused : 1;          // 0x0088 (0x0004)
[0x0000000000000001] [0x00000001] (CPF_Edit)
unsigned long           bTick : 1;                        // 0x0088 (0x0004)
[0x0000000000000002] [0x00000002] (CPF_Const)
struct FScriptDelegate  __TickDelegate__Delegate;        // 0x0090 (0x0018)
[0x0000000000040000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.TickComponent_X");
}

return uClassPointer;
};

```

```

void SetTickable(unsigned long bWantsTick);
void TickDelegate(float DeltaTime);
};

```

```

// Class ProjectX.UIStrings_X
// 0x0000 (0x0060 - 0x0060)
class UUIStrings_X : public UObject
{
public:

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.UIStrings_X");
}

return uClassPointer;
};

```

```

static class FString FormatAsNumber(int32_t Number, class FString Delimiter);
};

```

```

// Class ProjectX.WebRequest_X
// 0x0078 (0x0060 - 0x00D8)
class UWebRequest_X : public UObject
{
public:

```



```

unsigned long                bZipRequest : 1;                // 0x0060 (0x0004)
[0x0000000000000001] [0x00000001] (CPF_Edit)
unsigned long                bZipResponse : 1;              // 0x0060 (0x0004)
[0x0000000000000001] [0x00000002] (CPF_Edit)
unsigned long                bLogRequestContent : 1;        // 0x0060 (0x0004)
[0x0000000000000001] [0x00000004] (CPF_Edit)
unsigned long                bLogResponseContent : 1;       // 0x0060 (0x0004)
[0x0000000000000001] [0x00000008] (CPF_Edit)
unsigned long                bLogSendRecv : 1;              // 0x0060 (0x0004)
[0x0000000000000001] [0x00000010] (CPF_Edit)
unsigned long                bConnectionReset : 1;          // 0x0060 (0x0004)
[0x00000000000002002] [0x00000020] (CPF_Const | CPF_Transient)
uint8_t                      RequestState;                  // 0x0064 (0x0001)
[0x00000000000002002] (CPF_Const | CPF_Transient)
class FString                URL;                            // 0x0068 (0x0010)
[0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
class UHttpRequestInterface* HTTPRequest;                  // 0x0078 (0x0008)
[0x00000004000002000] (CPF_Transient)
class UHttpResponseInterface* HttpResponse;                // 0x0080 (0x0008)
[0x00000004000002002] (CPF_Const | CPF_Transient)
TArray<uint8_t>              Content;                        // 0x0088 (0x0010)
[0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
class FString                StringContent;                 // 0x0098 (0x0010)
[0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
TArray<uint8_t>              ResponseData;                  // 0x00A8 (0x0010)
[0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
class UError*                Error;                         // 0x00B8 (0x0008)
[0x00000000000002002] (CPF_Const | CPF_Transient)
struct FScriptDelegate        __EventCompleted__Delegate;  // 0x00C0
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.WebRequest_X");
}

return uClassPointer;
};

void PrintDebugInfo(class UDebugDrawer* Drawer);
void SetError(class UError* InError);
bool eventHasInternetConnection();
void HandleHttpRequestComplete(class UHttpRequestInterface* OriginalRequest, class
UHttpResponseInterface* Response, unsigned long bDidSucceed);
void eventConstructHttpRequest();
class FString GetResponseHeader(class FString Key);
class FString GetResponseString();
bool HasNewData();
bool IsSuccess();

```

```

bool IsError();
bool IsComplete();
bool IsSent();
class UHttpRequestInterface* GetHttpRequest();
class UWebRequest_X* Send(class FString InURL, struct FScriptDelegate OnCompleteCallback);
void PrepareRequest(class FString InURL);
void SetContentFromString(class FString ContentString);
void SetContent(TArray<uint8_t>& ContentBytes);
class FString GetETag();
void SetETag(class FString ETag);
class UWebRequest_X* SetVerb(class FString InVerb);
class UWebRequest_X* SetHeader(class FString Key, class FString Value);
class UWebRequest_X* AddPlayerIDParam(struct FUniqueNetId PlayerID);
class UWebRequest_X* AddStringParam(class FString Key, class FString Value);
class UWebRequest_X* AddFloatParam(class FString Key, float Value);
class UWebRequest_X* AddQwordParam(class FString Key, uint64_t Value);
class UWebRequest_X* AddIntParam(class FString Key, int32_t Value);
static uint8_t FlushWebRequests(float TimeoutSeconds);
static class UWebRequest_X* Create();
void EventCompleted(class UWebRequest_X* Request);
};

```

```

// Class ProjectX.WordFilterTypes_X

```

```

// 0x0030 (0x0060 - 0x0090)

```

```

class UWordFilterTypes_X : public UObject

```

```

{
public:
struct FScriptDelegate      __CommentSanitizedDelegate__Delegate;      // 0x0060
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __CommentErrorDelegate__Delegate;          // 0x0078
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

```

```

public:

```

```

static UClass* StaticClass()

```

```

{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)

```

```

{
uClassPointer = UObject::FindClass("Class ProjectX.WordFilterTypes_X");
}

```

```

return uClassPointer;

```

```

};

```

```

static void SortRequestTimeAscending(TArray<struct FWordFilterPair>& Filtered);

```

```

static void SortPendingFront(TArray<struct FWordFilterPair>& Filtered);

```

```

void CommentErrorDelegate(class FString OriginalPhrase, class UError* Error);

```

```

void CommentSanitizedDelegate(class FString Original, class FString Sanitized);

```

```

};

```

```

// Class ProjectX.AnimNodeBlendBase_X

```

```

// 0x0020 (0x016C - 0x018C)

```

```

class UAnimNodeBlendBase_X : public UAnimNodeBlendList

```

```

{
public:
float BlendTime; // 0x0170 (0x0004)
[0x0000000000000001] (CPF_Edit)
TArray<float> ChildBlendTimes; // 0x0178 (0x0010)
[0x0000000000400001] (CPF_Edit | CPF_NeedCtorLink)
unsigned long bRebuildChildren : 1; // 0x0188 (0x0004)
[0x0000000000000001] [0x00000001] (CPF_Edit)
unsigned long bChildrenUpdated : 1; // 0x0188 (0x0004)
[0x0000000000002000] [0x00000002] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.AnimNodeBlendBase_X");
}

return uClassPointer;
};

void SetActiveChildNamed(struct FName ChildName, unsigned long bForce);
float GetAnimDuration(int32_t ChildIndex);
float GetBlendTime(int32_t ChildIndex, unsigned long bGetDefault);
};

// Class ProjectX.AnimNodeBlendList_X
// 0x0004 (0x018C - 0x0190)
class UAnimNodeBlendList_X : public UAnimNodeBlendBase_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.AnimNodeBlendList_X");
}

return uClassPointer;
};

};

// Class ProjectX.AnimNodeCustomSeries_X
// 0x0004 (0x018C - 0x0190)
class UAnimNodeCustomSeries_X : public UAnimNodeBlendBase_X
{

```

```
public:
```

```
public:
```

```
static UClass* StaticClass()
```

```
{
```

```
static UClass* uClassPointer = nullptr;
```

```
if (!uClassPointer)
```

```
{
```

```
uClassPointer = UObject::FindClass("Class ProjectX.AnimNodeCustomSeries_X");
```

```
}
```

```
return uClassPointer;
```

```
};
```

```
};
```

```
// Class ProjectX.AnimNodeSeries_X
```

```
// 0x001C (0x018C - 0x01A8)
```

```
class UAnimNodeSeries_X : public UAnimNodeBlendBase_X
```

```
{
```

```
public:
```

```
unsigned long bActive : 1; // 0x0190 (0x0004)
```

```
[0x00000000000002000] [0x00000001] (CPF_Transient)
```

```
unsigned long bInitialized : 1; // 0x0190 (0x0004)
```

```
[0x00000000000002002] [0x00000002] (CPF_Const | CPF_Transient)
```

```
class UAnimNodeSequence* IntroAnim; // 0x0198 (0x0008)
```

```
[0x00000000000002002] (CPF_Const | CPF_Transient)
```

```
class UAnimNodeSequence* OutroAnim; // 0x01A0 (0x0008)
```

```
[0x00000000000002002] (CPF_Const | CPF_Transient)
```

```
public:
```

```
static UClass* StaticClass()
```

```
{
```

```
static UClass* uClassPointer = nullptr;
```

```
if (!uClassPointer)
```

```
{
```

```
uClassPointer = UObject::FindClass("Class ProjectX.AnimNodeSeries_X");
```

```
}
```

```
return uClassPointer;
```

```
};
```

```
};
```

```
// Class ProjectX.AnimNodeSequence_X
```

```
// 0x0018 (0x01A8 - 0x01C0)
```

```
class UAnimNodeSequence_X : public UAnimNodeSequence
```

```
{
```

```
public:
```

```
float AnimSetBlendTime; // 0x01A8 (0x0004)
```

```
[0x0000000000000001] (CPF_Edit)
```

```
unsigned long bResetOnRelevant : 1; // 0x01AC (0x0004)
```

```

[0x0000000000000001] [0x00000001] (CPF_Edit)
class UAnimSequence*                               PrevAnimSeq;                               // 0x01B0 (0x0008)
[0x0000000000000200] (CPF_Transient)
int32_t                                             PrevAnimLinkupIndex;                               // 0x01B8 (0x0004)
[0x0000000000000200] (CPF_Transient)
float                                              BlendTimeLeft;                                   // 0x01BC (0x0004)
[0x0000000000000200] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.AnimNodeSequence_X");
}

return uClassPointer;
};

};

// Class ProjectX.AnimNodePose_X
// 0x0000 (0x01C0 - 0x01C0)
class UAnimNodePose_X : public UAnimNodeSequence_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.AnimNodePose_X");
}

return uClassPointer;
};

void PlayAnim(unsigned long bLoop, float InRate, float StartTime);
};

// Class ProjectX.AnimNotify_PlayParticleEffect_X
// 0x0008 (0x0098 - 0x00A0)
class UAnimNotify_PlayParticleEffect_X : public UAnimNotify_PlayParticleEffect
{
public:
class UParticleSystemComponent*                  PSC;                               // 0x0098 (0x0008)
[0x0000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)

public:

```

```

static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.AnimNotify_PlayParticleEffect_X");
}

return uClassPointer;
};

void OnParticleSystemFinished(class UParticleSystemComponent* Component);
void EndParticles();
void eventNotifyEnd(class AActor* Owner, class UAnimNodeSequence* AnimSeqInstigator);
void eventNotify(class AActor* Owner, class UAnimNodeSequence* AnimSeqInstigator);
};

// Class ProjectX.PsyNetBeacon_X
// 0x0080 (0x0070 - 0x00F0)
class UPsyNetBeacon_X : public UComponent
{
public:
float ConnectionTimeoutSeconds; // 0x0070 (0x0004)
[0x00000000000004000] (CPF_Config)
class UPsyNetConnection_X* PsyNetConnection; // 0x0078
(0x0008) [0x00000000000000000]
TArray<class UPsyNetBeaconConnection_X*> Connections; // 0x0080
(0x0010) [0x00000000000400000] (CPF_NeedCtorLink)
TArray<class UDSPendingMessage_X*> PendingMessages; // 0x0090
(0x0010) [0x00000000000400000] (CPF_NeedCtorLink)
class UPsyNetConfig_X* PsyNetConfig; // 0x00A0 (0x0008)
[0x000080000000000000]
struct FScriptDelegate __EventConnected__Delegate; // 0x00A8
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventDisconnected__Delegate; // 0x00C0
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventMessageReceived__Delegate; // 0x00D8
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetBeacon_X");
}

return uClassPointer;
};

void SerializeMessage(class UObject* Message, class FString& OutMessageType, class FString&

```

```

OutMessagePayload);
class UObject* DeserializeMessage(class FString MessageType, class FString MessagePayload);
void TimeoutPendingMessages();
void TimeoutConnections();
void Tick();
bool IsGameClient();
void IssuePendingMessages(class UPsyNetBeaconConnection_X* Connection);
void AddPendingMessage(class UDSR_DSMessage_X* Message);
void OnMessageReceived(class UPsyNetBeaconConnection_X* Connection, class UObject*
Message);
void ClientHandleMessage(class UDSR_ClientMessage_X* Message);
void ServerProcessMessage(class UPsyNetBeaconConnection_X* Connection, class
UDSR_DSMessage_X* Message);
void ServerHandleServerMessage(class UDSR_DSToDSMessage_X* Message);
void ServerHandleMessage(class UDSR_DSMessage_X* Message);
void HandleClientRpcFail(class UError* Error, class UPsyNetBeaconConnection_X* Connection);
void ServerSendMessageToServerW(class FString ServerId, class UObject* Message);
void SendMessageToServerW(class UPsyNetBeaconConnection_X* Connection, class UObject*
Message);
void SendMessageToClientsW(TArray<class UPsyNetBeaconConnection_X*> ToConnections,
class UObject* Message);
class UPsyNetBeaconConnection_X* FindConnection(class FString ReservationID, class FString
ConnectionID);
void Close();
void CloseConnection(class UPsyNetBeaconConnection_X* Connection);
void eventBroadcastMessage(class UObject* Message);
void SendReservationMessage(class UObject* Message, struct FServerReservationData&
Reservation);
void eventSendMessageW(class UPsyNetBeaconConnection_X* Connection, class UObject*
Message);
class UPsyNetBeaconConnection_X* ReserveConnection(class FString ReservationID, class
FString ConnectionID, float OverrideConnectionTimeoutSeconds);
void SetPsyNetConnection(class UPsyNetConnection_X* InPsyNetConnection);
void EventMessageReceived(class UPsyNetBeaconConnection_X* Connection, class UObject*
Message);
void EventDisconnected(class UPsyNetBeacon_X* Beacon, class UPsyNetBeaconConnection_X*
Connection);
void EventConnected(class UPsyNetBeacon_X* Beacon, class UPsyNetBeaconConnection_X*
Connection);
};

```

```

// Class ProjectX.TcpConnection

```

```

// 0x00E0 (0x0060 - 0x0140)

```

```

class UTcpConnection : public UObject

```

```

{
public:
struct FPointer                VfTable_FTickableObject;                // 0x0060 (0x0008)
[0x000000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
struct FPointer                VfTable_IIReservationConnection_X;      // 0x0068
(0x0008) [0x000000000000801002] (CPF_Const | CPF_Native | CPF_NoExport)
float                          InitialConnectionTimeout;              // 0x0070 (0x0004)
[0x00000000000004000] (CPF_Config)
float                          ConnectionTimeout;                      // 0x0074 (0x0004)
[0x00000000000004000] (CPF_Config)

```

```

struct FPointer          Socket;                                // 0x0078 (0x0008)
[0x00000000000003002] (CPF_Const | CPF_Native | CPF_Transient)
uint8_t                 ConnectionState;                        // 0x0080 (0x0001)
[0x00000000000002002] (CPF_Const | CPF_Transient)
uint8_t                 WebSocketState;                        // 0x0081 (0x0001)
[0x00000000000002002] (CPF_Const | CPF_Transient)
float                   LastReceiveTime;                       // 0x0084 (0x0004)
[0x00000000000002002] (CPF_Const | CPF_Transient)
TArray<uint8_t>          InBytes;                              // 0x0088 (0x0010)
[0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
TArray<uint8_t>          OutBytes;                             // 0x0098 (0x0010)
[0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
TArray<int32_t>          OutMessageLengths;                    // 0x00A8 (0x0010)
[0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
TArray<uint8_t>          QueuedWebSocketBytes;                 // 0x00B8 (0x0010)
[0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
TArray<int32_t>          QueuedWebSocketMessageLengths;        // 0x00C8
(0x0010) [0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
TArray<uint8_t>          InBytesDecoded;                       // 0x00D8 (0x0010)
[0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
TArray<uint8_t>          BunchBytes;                           // 0x00E8 (0x0010)
[0x00000000000402002] (CPF_Const | CPF_Transient | CPF_NeedCtorLink)
struct FScriptDelegate   __EventConnected__Delegate;          // 0x00F8
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate   __EventDisconnected__Delegate;        // 0x0110
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate   __EventReceivedBunch__Delegate;       // 0x0128
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.TcpConnection");
}

return uClassPointer;
};

void eventOnDisconnected();
void eventOnConnected();
void Close();
class FString GetRemoteAddress();
class FString GetAddress();
void EventReceivedBunch(class UTcpConnection* Connection);
void EventDisconnected(class UTcpConnection* Connection);
void EventConnected(class UTcpConnection* Connection);
};

// Class ProjectX.WebSocketConnection_X
// 0x001C (0x0140 - 0x015C)

```



```

class UWebSocketConnection_X : public UTcpConnection
{
public:
float          PingInterval;                // 0x0140 (0x0004)
[0x00000000000004000] (CPF_Config)
class FString          ClosedReason;                // 0x0148 (0x0010)
[0x000000000000400002] (CPF_Const | CPF_NeedCtorLink)
int32_t          ClosedCode;                // 0x0158 (0x0004)
[0x00000000000000002] (CPF_Const)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.WebSocketConnection_X");
}

return uClassPointer;
};

void PretendClose(int32_t Code, class FString Reason);
void Close();
bool BeginWebSocketConnect(class FString Address, class UStringMap* HandshakeHeaders);
bool BeginConnect(class FString Address);
};

// Class ProjectX.UdpPingBeaconBase_X
// 0x0028 (0x0070 - 0x0098)
class UUdpPingBeaconBase_X : public UComponent
{
public:
struct FPointer          Beacon;                // 0x0070 (0x0008)
[0x00000000000001002] (CPF_Const | CPF_Native)
struct FPointer          Thread;                // 0x0078 (0x0008)
[0x00000000000001002] (CPF_Const | CPF_Native)
struct FPointer          Runnable;                // 0x0080 (0x0008)
[0x00000000000001002] (CPF_Const | CPF_Native)
int32_t          SocketErrorLogDelaySeconds;                // 0x0088 (0x0004)
[0x00000000000004000] (CPF_Config)
unsigned long          bThreaded : 1;                // 0x008C (0x0004)
[0x00000000000004000] [0x00000001] (CPF_Config)
class UBeaconConfig_X*          Config;                // 0x0090 (0x0008)
[0x00008000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{

```

```

uClassPointer = UObject::FindClass("Class ProjectX.UdpPingBeaconBase_X");
}

return uClassPointer;
};

void TickSocket(float DeltaTime);
void eventSetTickSocket(unsigned long bTick);
void Close();
};

// Class ProjectX.UdpPingBeaconClient_X
// 0x0030 (0x0098 - 0x00C8)
class UUdpPingBeaconClient_X : public UUdpPingBeaconBase_X
{
public:
    struct FScriptDelegate      __EventPong__Delegate;           // 0x0098 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate      __EventLost__Delegate;          // 0x00B0 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.UdpPingBeaconClient_X");
        }

        return uClassPointer;
    };

    void TickPongs(float DeltaTime);
    void eventSetTickPongs(unsigned long bTick);
    void Close();
    void SendPing(struct FName Address);
    bool InitClient();
    static class UUdpPingBeaconClient_X* GetInstance();
    void EventLost(class UUdpPingBeaconClient_X* TheBeacon, struct FName Address);
    void EventPong(class UUdpPingBeaconClient_X* TheBeacon, struct FName Address, float DeltaSeconds);
};

// Class ProjectX.UdpPingBeaconServer_X
// 0x0000 (0x0098 - 0x0098)
class UUdpPingBeaconServer_X : public UUdpPingBeaconBase_X
{
public:

public:
    static UClass* StaticClass()
    {

```

```

static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.UdpPingBeaconServer_X");
}

return uClassPointer;
};

bool InitServer(int32_t Port);
};

// Class ProjectX.PrivilegeCheck_X
// 0x0088 (0x0060 - 0x00E8)
class UPrivilegeCheck_X : public UObject
{
public:
    TArray<int32_t> ControllerIDs; // 0x0060 (0x0010)
    [0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
    unsigned long bHasRestriction : 1; // 0x0070 (0x0004)
    [0x0000004000002000] [0x00000001] (CPF_Transient)
    unsigned long bFinished : 1; // 0x0070 (0x0004)
    [0x0000000000002000] [0x00000002] (CPF_Transient)
    class UError* FailReason; // 0x0078 (0x0008)
    [0x0000004000002000] (CPF_Transient)
    TArray<uint8_t> PendingPrivileges; // 0x0080 (0x0010)
    [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    int32_t FailedControllerID; // 0x0090 (0x0004)
    [0x0000000000002000] (CPF_Transient)
    uint8_t FailedPrivilege; // 0x0094 (0x0001)
    [0x0000000000002000] (CPF_Transient)
    uint8_t FailedPrivilegeLevel; // 0x0095 (0x0001)
    [0x0000004000002000] (CPF_Transient)
    TArray<int32_t> PendingControllerIDs; // 0x0098 (0x0010)
    [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FScriptDelegate> CheckDelegates; // 0x00A8
    (0x0010) [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    struct FScriptDelegate EventFinished; // 0x00B8 (0x0018)
    [0x0000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate __CheckDelegate__Delegate; // 0x00D0
    (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PrivilegeCheck_X");
        }

        return uClassPointer;
    }
};

```

```

};

bool HasFeatureRestriction(int32_t& OutControllerID, uint8_t& OutFailedPrivilege);
void OnFinished();
void Fail(class FString Reason);
void HandlePrivilege(uint8_t LocalUserNum, uint8_t Privilege, uint8_t Level, class UError*
InFailReason);
void CheckPrivilege();
void CheckPsyNetComplete(class UError* ConnectionError);
void CheckPsyNet();
void ValidateUserOnlineAccount();
void CheckInternet();
class UPrivilegeCheck_X* RequirePrivilege(uint8_t Privilege);
class UPrivilegeCheck_X* RequirePsyNet();
class UPrivilegeCheck_X* RequireUserOnlineAccount();
class UPrivilegeCheck_X* RequireInternet();
void StartNextCheck();
void Start();
void Init(int32_t ControllerId);
void CheckDelegate();
};

// Class ProjectX.AddReservationMessage_X
// 0x006C (0x0060 - 0x00CC)
class UAddReservationMessage_X : public UBeaconMessage_X
{
public:
TArray<struct FReservationPlayerData>          Players;                                // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FUniqueNetId                          PartyLeaderID;                        // 0x0070 (0x0048)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString                                ReservationID;                        // 0x00B8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
unsigned long                                bDisableCrossPlay : 1;                      // 0x00C8 (0x0004)
[0x000000000000000000] [0x00000001]
unsigned long                                bMarkSplitScreenAsRemote : 1;          // 0x00C8 (0x0004)
[0x00010000000006000] [0x00000002] (CPF_Transient | CPF_Config)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.AddReservationMessage_X");
}

return uClassPointer;
};

struct FUniqueNetId __AddReservationMessage_X__GetPlayerIDs_0x1(struct
FReservationPlayerData P);
TArray<struct FUniqueNetId> GetPlayerIDs();

```

```

void GetPlayerMapPrefs(struct FUniqueNetId PlayerID, TArray<struct FName>& Likes,
TArray<struct FName>& Dislikes);
class UAddReservationMessage_X* SetDisableCrossPlay(unsigned long bValue);
class UAddReservationMessage_X* AddPlayer(struct FUniqueNetId PlayerID, class FString
PlayerName, unsigned long bRemotePlayer);
class UAddReservationMessage_X* AddOnlinePlayer(class UOnlinePlayer_X* Player, unsigned
long bFriendJoin);
class UAddReservationMessage_X* AddPlayers();
void SendReservation(struct FServerReservationData& Reservation);
class UAddReservationMessage_X* SetReservationID(class FString Id);
};

// Class
ProjectX.____OnlineGameParty_X__UpdatePartyInfo_0x1____OnlineGameParty_X__UpdatePartyIn
fo_0x1_0x1
// 0x0170 (0x0060 - 0x01D0)
class
U____OnlineGameParty_X__UpdatePartyInfo_0x1____OnlineGameParty_X__UpdatePartyInfo_0x1_0
x1 : public UObject
{
public:
struct FPartyMember PM; // 0x0060 (0x0170)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.____OnlineGameParty_X__UpdatePartyInfo_0x1____OnlineGameParty_X__UpdatePartyInf
o_0x1_0x1");
}

return uClassPointer;
};

bool
____OnlineGameParty_X__UpdatePartyInfo_0x1____OnlineGameParty_X__UpdatePartyInfo_0x1_0x
1(struct FLobbyMember LM);
};

// Class ProjectX.____OnlineGameParty_X__UpdatePartyInfo_0x1
// 0x0030 (0x0060 - 0x0090)
class U____OnlineGameParty_X__UpdatePartyInfo_0x1 : public UObject
{
public:
struct FActiveLobbyInfo Lobby; // 0x0060 (0x0030)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()

```

```

{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGameParty_X__UpdatePartyInfo_0x1");
}

return UClassPointer;
};

bool __OnlineGameParty_X__UpdatePartyInfo_0x1(struct FPartyMember PM);
};

// Class ProjectX.SetPlayerStorageResult_X
// 0x0010 (0x0060 - 0x0070)
class USetPlayerStorageResult_X : public UObject
{
public:
TArray<struct FSetPlayerStorageResultItem>      Items;                // 0x0060
(0x0010) [0x00010000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class ProjectX.SetPlayerStorageResult_X");
}

return UClassPointer;
};

};

// Class
ProjectX.__OnlinePlayerStorageQueue_X__HandleStorageSuccess_0x2__OnlinePlayerStora
geQueue_X__HandleStorageSuccess_0x4_0x1
// 0x0010 (0x0060 - 0x0070)
class
U__OnlinePlayerStorageQueue_X__HandleStorageSuccess_0x2__OnlinePlayerStorageQueue_
X__HandleStorageSuccess_0x4_0x1 : public UObject
{
public:
struct FSetPlayerStorageResultItem              R;                // 0x0060 (0x0010)
[0x00010000000000000]

public:
static UClass* StaticClass()
{
static UClass* UClassPointer = nullptr;

```

```

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.____OnlinePlayerStorageQueue_X__HandleStorageSuccess_0x2____OnlinePlayerStorag
eQueue_X__HandleStorageSuccess_0x4_0x1");
}

return uClassPointer;
};

bool
____OnlinePlayerStorageQueue_X__HandleStorageSuccess_0x2____OnlinePlayerStorageQueue_X
__HandleStorageSuccess_0x4_0x1(struct FPendingStorage P);
};

// Class ProjectX.OnlinePlayerStorageQueue_X
// 0x0088 (0x0060 - 0x00E8)
class UOnlinePlayerStorageQueue_X : public UObject
{
public:
class UOnlinePlayerStorageConfig_X*          Config;                      // 0x0060 (0x0008)
[0x0001800000000000] (CPF_Edit)
TArray<struct FPendingStorage>                QueuedObjects;              // 0x0068
(0x0010) [0x0001000000400000] (CPF_NeedCtorLink)
TArray<struct FPendingStorage>                PendingObjects;            // 0x0078
(0x0010) [0x0001000000400000] (CPF_NeedCtorLink)
float                                         DefaultStorageMaxSizeBytes;      // 0x0088 (0x0004)
[0x0001000000000003] (CPF_Edit | CPF_Const)
TArray<struct FStorageMaxSize>               StorageMaxSizes;                // 0x0090
(0x0010) [0x0001000000400003] (CPF_Edit | CPF_Const | CPF_NeedCtorLink)
struct FScriptDelegate                      __EventStorageSuccess__Delegate; // 0x00A0
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventOutOfSync__Delegate;      // 0x00B8
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventStorageError__Delegate;   // 0x00D0
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlinePlayerStorageQueue_X");
}

return uClassPointer;
};

void ____OnlinePlayerStorageQueue_X__Construct_0x1(class FString _);
void ____OnlinePlayerStorageQueue_X__SendBatch_Internal_0x1(class URPC_X* RPC);
bool ____OnlinePlayerStorageQueue_X__HandleStorageSuccess_0x5(struct FPendingStorage P);

```

```

void __OnlinePlayerStorageQueue_X__HandleStorageSuccess_0x3(struct
FSetPlayerStorageRequestItem R);
bool __OnlinePlayerStorageQueue_X__HandleStorageSuccess_0x1(struct
FSetPlayerStorageResultItem Item);
void __OnlinePlayerStorageQueue_X__HandleStorageError_0x2(struct FPendingStorage P);
bool __OnlinePlayerStorageQueue_X__HandleStorageError_0x1(struct FPendingStorage P);
int32_t GetStorageMaxSizeBytes(struct FName Category);
void HandleStorageError(class UError* Error);
struct FPendingStorage MapResultItem(struct FSetPlayerStorageResultItem Item);
void HandleStorageSuccess(class URPC_PlayerStorageSet_X* RPC);
void PartitionStorageRequests(TArray<struct FPendingStorage>& OutQueue, TArray<struct
FSetPlayerStorageRequestItem>& OutOfDate);
void SendBatch_Internal(TArray<struct FPendingStorage>& OutQueue);
void SendBatch();
void QueueBatch();
void SendImmediate(class UObject* DataObj, int32_t ClientTick);
void UpdateQueuedObject(class UObject* DataObj, int32_t ClientTick);
void QueueObject(class UObject* DataObj, int32_t ClientTick);
void eventConstruct();
void EventStorageError(class UError* Error);
void EventOutOfSync(TArray<struct FPendingStorage>& FailedObjects);
void EventStorageSuccess(class UObject* Data, int32_t ServerTick, uint8_t Encoding);
};

// Class ProjectX.__OnlinePlayerStorageQueue_X__HandleStorageSuccess_0x2
// 0x0018 (0x0060 - 0x0078)
class U__OnlinePlayerStorageQueue_X__HandleStorageSuccess_0x2 : public UObject
{
public:
TArray<struct FSetPlayerStorageResultItem>      OutOfSync;                      // 0x0060
(0x0010) [0x0001000000400000] (CPF_NeedCtorLink)
class URPC_PlayerStorageSet_X*                  RPC;                          // 0x0070 (0x0008)
[0x0001000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlinePlayerStorageQueue_X__HandleStorageSuccess_0x2");
}

return uClassPointer;
};

void __OnlinePlayerStorageQueue_X__HandleStorageSuccess_0x4(struct
FSetPlayerStorageResultItem R);
bool __OnlinePlayerStorageQueue_X__HandleStorageSuccess_0x2(struct
FSetPlayerStorageRequestItem R);
};

```



```

// Class ProjectX.__AdHocBrowser_X__CreateErrorDelegate_0x1
// 0x0020 (0x0060 - 0x0080)
class U__AdHocBrowser_X__CreateErrorDelegate_0x1 : public UObject
{
public:
class UAsyncTask* Task; // 0x0060 (0x0008)
[0x0000000000000000]
struct FScriptDelegate ErrorDelegate; // 0x0068 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__AdHocBrowser_X__CreateErrorDelegate_0x1");
}

return uClassPointer;
};

void __AdHocBrowser_X__CreateErrorDelegate_0x2();
void __AdHocBrowser_X__CreateErrorDelegate_0x1(class UError* Error);
};

// Class ProjectX.LanBrowser_X
// 0x0000 (0x0060 - 0x0060)
class ULanBrowser_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.LanBrowser_X");
}

return uClassPointer;
};

class UAsyncTask* JoinServer(class FString ServerId, class FString Options);
class UAsyncResult__array_LanServerRecord_X* GetServerList();
void DestroyServer();
class UAsyncTask* SetServerMetaData(class FString MetaData);
class UAsyncTask* CreateServer(class FString MetaData);
void HandleGameInfoSpawned(class AGameInfo_X* G);
void eventConstruct();

```

```

};

// Class ProjectX.AdHocBrowser_X
// 0x0048 (0x0060 - 0x00A8)
class UAdHocBrowser_X : public ULanBrowser_X
{
public:
    class UAdHocBeacon_X*                AdHocBeacon;                // 0x0060 (0x0008)
    [0x000080000000000000]
    TArray<struct FAdHocAccessPointInfo>    AdHocNodesOnNetwork;      //
    0x0068 (0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
    float                                SearchTimeout;                // 0x0078 (0x0004)
    [0x000000000000000003] (CPF_Edit | CPF_Const)
    class UAsyncResult__array_LanServerRecord_X*    SearchTask;        // 0x0080
    (0x0008) [0x000000000000000000]
    TArray<class ULanServerRecord_X*>        SearchResults;            // 0x0088
    (0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
    TArray<struct FAdHocAccessPointInfo>    AdHocAccessPointsAvailable; //
    0x0098 (0x0010) [0x0000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.AdHocBrowser_X");
        }

        return uClassPointer;
    };

    void __AdHocBrowser_X__GetServerList_0x1();
    void HandleSearchTimeout();
    void AdHocHandleOnNetworkChanged(TArray<struct FAdHocAccessPointInfo> Ahapis);
    int32_t FindIndexFromServerID(class FString InServerID);
    class UAsyncTask* JoinServer(class FString ServerId, class FString Options);
    static void CreateErrorDelegate(class UAsyncTask* Task);
    class UAsyncResult__array_LanServerRecord_X* GetServerList();
    class UAsyncTask* SetServerMetaData(class FString MetaData);
    class UAsyncTask* CreateServer(class FString MetaData);
};

// Class ProjectX.__AdHocBrowser_X__CreateServer_0x1
// 0x0030 (0x0060 - 0x0090)
class U__AdHocBrowser_X__CreateServer_0x1 : public UObject
{
public:
    class FString                                MetaData;                // 0x0060 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)
    class UAsyncTask*                            Task;                    // 0x0070 (0x0008)
    [0x000000000000000000]
    struct FScriptDelegate                        ApcDelegate;            // 0x0078 (0x0018)

```

[0x0000000000040000] (CPF\_NeedCtorLink)

```
public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.__AdHocBrowser_X__CreateServer_0x1");
}

return uClassPointer;
};

void __AdHocBrowser_X__CreateServer_0x2();
void __AdHocBrowser_X__CreateServer_0x1();
};
```

```
// Class ProjectX.__AdHocBrowser_X__JoinServer_0x1
// 0x0028 (0x0060 - 0x0088)
class U__AdHocBrowser_X__JoinServer_0x1 : public UObject
{
public:
class UAsyncTask*                Task;                // 0x0060 (0x0008)
[0x0000000000000000]
class FString                    ServerId;             // 0x0068 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString                    Options;              // 0x0078 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
```

```
public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.__AdHocBrowser_X__JoinServer_0x1");
}

return uClassPointer;
};

void __AdHocBrowser_X__JoinServer_0x1();
};
```

```
// Class ProjectX.__AdHocInterface_X__DisableAdHoc_0x1
// 0x0020 (0x0060 - 0x0080)
class U__AdHocInterface_X__DisableAdHoc_0x1 : public UObject
{
public:
class UAsyncTask*                Task;                // 0x0060 (0x0008)
[0x0000000000000000]
```

```

struct FScriptDelegate DestroyedDelegate; // 0x0068 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.__AdHocInterface_X__DisableAdHoc_0x1");
}

return uClassPointer;
};

void __AdHocInterface_X__DisableAdHoc_0x2();
void __AdHocInterface_X__DisableAdHoc_0x1();
};

// Class ProjectX.AdHocInterface_X
// 0x0020 (0x0060 - 0x0080)
class UAdHocInterface_X : public UObject
{
public:
class ULanBrowser_X* OldLanBrowser; // 0x0060 (0x0008)
[0x000000000000000000]
class UAdHocBeacon_X* AdHocBeacon; // 0x0068 (0x0008)
[0x000000000000000000]
class UOnlineGameParty_X* OnlineGameParty; // 0x0070
(0x0008) [0x000080000000000000]
class UOnlineGameMatchmaking_X* Matchmaking; // 0x0078
(0x0008) [0x000080000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.AdHocInterface_X");
}

return uClassPointer;
};

void CreateErrorDelegate(class UAsyncTask* Task);
class UAsyncTask* DisableAdHoc();
void HandleGameInfoSpawned(class AGameInfo_X* InGameInfo);
class UAsyncTask* EnableAdHoc();
};

// Class ProjectX.__AdHocInterface_X__EnableAdHoc_0x1

```

```

// 0x0020 (0x0060 - 0x0080)
class U__AdHocInterface_X__EnableAdHoc_0x1 : public UObject
{
public:
class UAsyncTask*                               Task;                               // 0x0060 (0x0008)
[0x0000000000000000]
struct FScriptDelegate                          InitDelegate;                      // 0x0068 (0x0018)
[0x00000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.__AdHocInterface_X__EnableAdHoc_0x1");
}

return uClassPointer;
};

void __AdHocInterface_X__EnableAdHoc_0x2();
void __AdHocInterface_X__EnableAdHoc_0x1();
};

// Class ProjectX.__AvatarRequester_X__AddCallbackToRequest_0x1
// 0x0018 (0x0060 - 0x0078)
class U__AvatarRequester_X__AddCallbackToRequest_0x1 : public UObject
{
public:
struct FScriptDelegate                          ReadOnlineAvatarCompleteDelegate;          // 0x0060
(0x0018) [0x00000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__AvatarRequester_X__AddCallbackToRequest_0x1");
}

return uClassPointer;
};

void __AvatarRequester_X__AddCallbackToRequest_0x1(struct FUniqueNetId InPlayerId, class
UTexture* InAvatar, class FString InPlayerName);
};

// Class ProjectX.AvatarRequester_X
// 0x0068 (0x0060 - 0x00C8)

```

```

class UAvatarRequester_X : public UObject
{
public:
class UVanityConfig_X*          VanityConfig;          // 0x0060 (0x0008)
[0x0000800000000000]
TArray<class UPendingAvatarRequest_X*>    AllRequests;    // 0x0068
(0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
TArray<class UPendingAvatarRequest_X*>    PendingPermissionRequests;    //
0x0078 (0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
TArray<class UPendingAvatarRequest_X*>    PendingAvatarRequests;    //
0x0088 (0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate    __RequestAvatars__Delegate;    // 0x0098
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate    __RequestAvatarPermission__Delegate;    // 0x00B0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.AvatarRequester_X");
}

return uClassPointer;
};

struct FUniqueNetId __AvatarRequester_X__TimerReadAvatars_0x2(class
UPendingAvatarRequest_X* Request);
static class UPendingAvatarRequest_X* FindRequestFromArray(struct FUniqueNetId PlayerID,
uint8_t Size, TArray<class UPendingAvatarRequest_X*>& Requests);
static bool RemoveRequestFromArray(struct FUniqueNetId PlayerID, uint8_t Size, TArray<class
UPendingAvatarRequest_X*>& Requests);
void HandlePlayerIDDisallowed(struct FUniqueNetId PlayerID, uint8_t Size);
void HandleOnlineSubsystemReadAvatar(struct FUniqueNetId PlayerID, uint8_t Size, class
UTexture* Avatar, class FString OnlinePlayerName);
void TimerReadAvatars();
void HandlePlayerIDAllowed(struct FUniqueNetId PlayerID, uint8_t Size);
void GetPlayerAvatar(struct FUniqueNetId PlayerID, struct FScriptDelegate
ReadOnlineAvatarCompleteDelegate, uint8_t Size);
void AddCallbackToRequest(class UPendingAvatarRequest_X* Request, struct FScriptDelegate
ReadOnlineAvatarCompleteDelegate);
void GetPlayerAvatars(struct FScriptDelegate ReadOnlineAvatarCompleteDelegate, uint8_t Size,
TArray<struct FUniqueNetId>& PlayerIDs);
void RequestAvatarPermission(struct FUniqueNetId PlayerID, struct FScriptDelegate OnAllowed,
struct FScriptDelegate OnDisallowed);
void RequestAvatars(TArray<struct FUniqueNetId> PlayerIDs, uint8_t Size, struct FScriptDelegate
OnCompleteCallback);
};

// Class ProjectX.__AvatarRequester_X__FindRequestFromArray_0x1
// 0x0049 (0x0060 - 0x00A9)

```

```

class U__AvatarRequester_X__FindRequestFromArray_0x1 : public UObject
{
public:
    struct FUniqueNetId                PlayerID;                // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)
    uint8_t                            Size;                    // 0x00A8 (0x0001)
    [0x000000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__AvatarRequester_X__FindRequestFromArray_0x1");
        }

        return uClassPointer;
    };

    bool __AvatarRequester_X__FindRequestFromArray_0x1(class UPendingAvatarRequest_X*
    Request);
};

// Class ProjectX.PendingAvatarRequest_X
// 0x0068 (0x0060 - 0x00C8)
class UPendingAvatarRequest_X : public UObject
{
public:
    struct FUniqueNetId                PlayerID;                // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)
    uint8_t                            Size;                    // 0x00A8 (0x0001)
    [0x000000000000000000]
    struct FScriptDelegate             OnAvatarRequestComplete; // 0x00B0
    (0x0018) [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PendingAvatarRequest_X");
        }

        return uClassPointer;
    };

};

// Class ProjectX.__AvatarRequester_X__GetPlayerAvatar_0x1

```

```

// 0x0049 (0x0060 - 0x00A9)
class U__AvatarRequester_X__GetPlayerAvatar_0x1 : public UObject
{
public:
    struct FUniqueNetId                PlayerID;                // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)
    uint8_t                            Size;                    // 0x00A8 (0x0001)
    [0x000000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__AvatarRequester_X__GetPlayerAvatar_0x1");
        }

        return uClassPointer;
    };

    void __AvatarRequester_X__GetPlayerAvatar_0x2(struct FUniqueNetId _);
    void __AvatarRequester_X__GetPlayerAvatar_0x1(struct FUniqueNetId _);
};

// Class ProjectX.__AvatarRequester_X__GetPlayerAvatars_0x1
// 0x0019 (0x0060 - 0x0079)
class U__AvatarRequester_X__GetPlayerAvatars_0x1 : public UObject
{
public:
    struct FScriptDelegate              ReadOnlineAvatarCompleteDelegate;    // 0x0060
    (0x0018) [0x000000000000400000] (CPF_NeedCtorLink)
    uint8_t                            Size;                    // 0x0078 (0x0001)
    [0x000000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__AvatarRequester_X__GetPlayerAvatars_0x1");
        }

        return uClassPointer;
    };

    void __AvatarRequester_X__GetPlayerAvatars_0x1(struct FUniqueNetId Id);
};

```



```

// Class ProjectX.__AvatarRequester_X__RemoveRequestFromArray_0x1
// 0x0049 (0x0060 - 0x00A9)
class U__AvatarRequester_X__RemoveRequestFromArray_0x1 : public UObject
{
public:
    struct FUniqueNetId                PlayerID;                // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)
    uint8_t                            Size;                    // 0x00A8 (0x0001)
    [0x000000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__AvatarRequester_X__RemoveRequestFromArray_0x1");
        }

        return uClassPointer;
    };

    bool __AvatarRequester_X__RemoveRequestFromArray_0x1(class UPendingAvatarRequest_X*
    Request);
};

// Class ProjectX.__AvatarRequester_X__TimerReadAvatars_0x1
// 0x0001 (0x0060 - 0x0061)
class U__AvatarRequester_X__TimerReadAvatars_0x1 : public UObject
{
public:
    uint8_t                            BatchAvatarSize;          // 0x0060 (0x0001)
    [0x000000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__AvatarRequester_X__TimerReadAvatars_0x1");
        }

        return uClassPointer;
    };

    void __AvatarRequester_X__TimerReadAvatars_0x3(struct FUniqueNetId PlayerID, class
    UTexture* Texture, class FString OnlinePlayerName);
    bool __AvatarRequester_X__TimerReadAvatars_0x1(class UPendingAvatarRequest_X* Request);
};

```

```

// Class ProjectX.__BlockStatusReporter_X__OnBlockListDownloadComplete_0x1
// 0x0010 (0x0060 - 0x0070)
class U__BlockStatusReporter_X__OnBlockListDownloadComplete_0x1 : public UObject
{
public:
TArray<struct FOnlineFriend>                BlockedPlayersArray;                // 0x0060
(0x0010) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__BlockStatusReporter_X__OnBlockListDownloadComplete_0x1");
}

return uClassPointer;
};

void __BlockStatusReporter_X__OnBlockListDownloadComplete_0x1(struct
FPlayerBlockListenData Sub);
};

// Class ProjectX.BlockStatusReporter_X
// 0x0050 (0x0060 - 0x00B0)
class UBlockStatusReporter_X : public UObject
{
public:
TArray<struct FPlayerBlockListenData>        PlayerSubscriptions;                // 0x0060
(0x0010) [0x0000000000040000] (CPF_NeedCtorLink)
class UPlatformBlockListStatus*              PrimaryStatus;                // 0x0070 (0x0008)
[0x000000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UBlockStatusReporterConfig_X*          Config;                // 0x0078 (0x0008)
[0x000080000000000000]
struct FScriptDelegate                      __EventBlockListDownloaded__Delegate;    // 0x0080
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventPlayerStatusDownloaded__Delegate; // 0x0098
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.BlockStatusReporter_X");
}

return uClassPointer;
};

```

```

};

void __BlockStatusReporter_X__Construct_0x1();
void TriggerPlayerCallback(class FString EpicId, struct FScriptDelegate Callback, TArray<struct FOnlineFriend>& BlockedPlayersArray);
void OnBlockListDownloadComplete(unsigned long bSuccess);
void ClearDownloadTimeout();
void HandleBlockListDownloadTimeout();
void HandleStatusChanged();
void HandlePsyNetDisconnected(class UPsyNetConnection_X* InConnection);
void HandlePsyNetConnected(class UPsyNetConnection_X* InConnection);
void HandleBlockListStatusCreated(class UPlatformBlockListStatus* Status, uint8_t ControllerId);
void ListenForBlockListDownloaded(struct FScriptDelegate Callback);
void ListenForPlayerBlockStatusDownloaded(class FString EpicId, struct FScriptDelegate Callback);
void eventConstruct();
void EventPlayerStatusDownloaded(class FString EpicId, unsigned long bBlocked);
void EventBlockListDownloaded();
};

// Class ProjectX.__BlockStatusReporter_X__TriggerPlayerCallback_0x1
// 0x0010 (0x0060 - 0x0070)
class U__BlockStatusReporter_X__TriggerPlayerCallback_0x1 : public UObject
{
public:
    class FString                EpicId;                // 0x0060 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.__BlockStatusReporter_X__TriggerPlayerCallback_0x1");
        }

        return uClassPointer;
    };
};

bool __BlockStatusReporter_X__TriggerPlayerCallback_0x1(struct FOnlineFriend BlockedPlayer);
};

// Class ProjectX.__CheatManager_X__RandomSleep_0x1
// 0x0010 (0x0060 - 0x0070)
class U__CheatManager_X__RandomSleep_0x1 : public UObject
{
public:
    float                        MinDelay;                // 0x0060 (0x0004)
    [0x000000000000000000]
    float                        MaxDelay;                // 0x0064 (0x0004)
};

```

```

[0x0000000000000000]
float                               MinSleep;                               // 0x0068 (0x0004)
[0x0000000000000000]
float                               MaxSleep;                               // 0x006C (0x0004)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class ProjectX.__CheatManager_X__RandomSleep_0x1");
}

return UClassPointer;
};

void __CheatManager_X__RandomSleep_0x1();
};

// Class ProjectX.__ClubUtil_X__IsClubTeam_0x1
// 0x0010 (0x0060 - 0x0070)
class U__ClubUtil_X__IsClubTeam_0x1 : public UObject
{
public:
TArray<uint64_t>                    TeamClubs;                               // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class ProjectX.__ClubUtil_X__IsClubTeam_0x1");
}

return UClassPointer;
};

bool __ClubUtil_X__IsClubTeam_0x1(uint64_t Id);
};

// Class ProjectX.ClubUtil_X
// 0x0000 (0x0060 - 0x0060)
class UClubUtil_X : public UObject
{
public:

public:
static UClass* StaticClass()

```

```

{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ClubUtil_X");
}

return uClassPointer;
};

static int32_t Wrap(int32_t Column, int32_t Count);
static struct FClubColorSet SwapPrimaryAccentColors(struct FClubColorSet Colors, class
UColorPalette_X* TeamPalette, class UColorPalette_X* AccentPalette);
static bool AreColorsDifferent(class UColorPalette_X* Palette, int32_t Color0, int32_t Color1);
static uint8_t EnsureDifferentColors(class UColorPalette_X* TeamPalette, class
UColorPalette_X* AccentPalette, int32_t DefaultColorID0, int32_t DefaultColorID1, struct
FClubColorSet& Set0, struct FClubColorSet& Set1);
static bool IsClubTeam(int32_t TeamSize, TArray<uint64_t>& TeamClubs);
static bool IsClubMatch(int32_t TeamSize, TArray<uint64_t>& Team0Clubs, TArray<uint64_t>&
Team1Clubs);
};

// Class ProjectX.__CrossplayConfig_X__GetDisabledCrossplayGroup_0x1
// 0x0001 (0x0060 - 0x0061)
class U__CrossplayConfig_X__GetDisabledCrossplayGroup_0x1 : public UObject
{
public:
uint8_t                                     PlayerPlatform;                // 0x0060 (0x0001)
[0x00000000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__CrossplayConfig_X__GetDisabledCrossplayGroup_0x1");
}

return uClassPointer;
};

bool __CrossplayConfig_X__GetDisabledCrossplayGroup_0x1(struct FCrossplayGroup P);
};

// Class ProjectX.__EOSHelpers_X__RequestEASAuth_0x1
// 0x0018 (0x0060 - 0x0078)
class U__EOSHelpers_X__RequestEASAuth_0x1 : public UObject
{
public:
struct FScriptDelegate                     Callback;                // 0x0060 (0x0018)

```

[0x0000000000400000] (CPF\_NeedCtorLink)

```
public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.__EOSHelpers_X__RequestEASAuth_0x1");
}
```

```
return uClassPointer;
};
```

```
void __EOSHelpers_X__RequestEASAuth_0x1(class UWebRequest_X* Response);
};
```

```
// Class ProjectX.EASAuthResponse
// 0x0010 (0x0060 - 0x0070)
class UEASAuthResponse : public UObject
{
public:
class FString                                access_token;                // 0x0060 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
```

```
public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EASAuthResponse");
}
```

```
return uClassPointer;
};
```

```
};
```

```
// Class ProjectX.EOSHelpers_X
// 0x0030 (0x0060 - 0x0090)
class UEOSHelpers_X : public UObject
{
public:
struct FScriptDelegate                      __HTTPRequestCallback__Delegate;    // 0x0060
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __ConvertErrorFunction__Delegate;    // 0x0078
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
```

```
public:
static UClass* StaticClass()
{
```

```

static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.EOSHelpers_X");
}

return uClassPointer;
};

static void SendHTTPRequestSelectAuth(class FString Verb, class FString URL, unsigned long
bAddContentTypeHeader, class UClass* ResponseClass, struct FScriptDelegate Callback, struct
FScriptDelegate ConvertErrorNew, class UOnlineSubsystem* EOS, int32_t LocalPlayerNum, class
FString BodyJson, class FString Continuation, class FString DisplayNameHint, class FString DOB,
class FString BearerAuthTicket, class FString ContentType);
static void SendHTTPRequestEASAuth(class FString Verb, class FString URL, unsigned long
bAddContentTypeHeader, class UClass* ResponseClass, struct FScriptDelegate Callback, struct
FScriptDelegate ConvertErrorNew, class UOnlineSubsystem* EOS, int32_t LocalPlayerNum, class
FString AuthURL, class FString BodyJson, class FString Continuation);
static void RequestEASAuth(class FString URL, class FString ClientCredentials, class FString
ClientID, class FString ClientSecret, struct FScriptDelegate Callback);
static bool SendHTTPRequest(class FString Verb, class FString URL, unsigned long
bAddContentTypeHeader, class UClass* ResponseClass, struct FScriptDelegate Callback, struct
FScriptDelegate ConvertErrorNew, class UOnlineSubsystem* EOS, int32_t LocalPlayerNum, class
FString BodyJson);
class UError* ConvertErrorFunction(class UEOS_ErrorResponse* ErrorResponse);
void HTTPRequestCallback(class UObject* Response, class UError* ErrorSending);
};

// Class ProjectX.__EOSHelpers_X__SendHTTPRequest_0x1
// 0x0080 (0x0060 - 0x00E0)
class U__EOSHelpers_X__SendHTTPRequest_0x1 : public UObject
{
public:
    class FString URL; // 0x0060 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate Callback; // 0x0070 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)
    class FString Verb; // 0x0088 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)
    unsigned long bAddContentTypeHeader : 1; // 0x0098 (0x0004)
    [0x000000000000000000] [0x000000001]
    class UClass* ResponseClass; // 0x00A0 (0x0008)
    [0x000000000000000000]
    struct FScriptDelegate ConvertErrorNew; // 0x00A8 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)
    class UOnlineSubsystem* EOS; // 0x00C0 (0x0008)
    [0x000000000000000000]
    int32_t LocalPlayerNum; // 0x00C8 (0x0004)
    [0x000000000000000000]
    class FString BodyJson; // 0x00D0 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:

```

```

static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.__EOSHelpers_X__SendHTTPRequest_0x1");
}

return uClassPointer;
};

void __EOSHelpers_X__SendHTTPRequest_0x1(unsigned long bSuccess, class FString
EpicAuthTicket);
};

// Class ProjectX.EpicErrors_X
// 0x0100 (0x0080 - 0x0180)
class UEpicErrors_X : public UErrorList
{
public:
class UErrorType* EpicAccountNotFound; // 0x0080 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EpicAccountLinkingFailed; // 0x0088 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EOSFailedToConnect; // 0x0090 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EOSFailedToConnectSplitscreen; // 0x0098
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* NotLoggedInToEOS; // 0x00A0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EOSUpdateRequired; // 0x00A8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* PinGrantCodeExpired; // 0x00B0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* PrimaryAccountNotSet; // 0x00B8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EpicInvalidPlayer; // 0x00C0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EpicThrottlingHit; // 0x00C8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EpicNotAcceptingFriendInvites; // 0x00D0
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* EpicRemoteFriendLimitReached; // 0x00D8
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* EpicMaxOutgoingInvitesReached; // 0x00E0
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* EpicLocalFriendLimitReached; // 0x00E8
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* EpicAccountLookupFailed; // 0x00F0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EpicAccountCreationFailed; // 0x00F8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EpicAccountRestricted; // 0x0100 (0x0008)

```



```

[0x0000000000000002] (CPF_Const)
class UErrorType* EpicPartyInviteFailed; // 0x0108 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* IncorrectSaveLoaded; // 0x0110 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EpicSocialBanned; // 0x0118 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* MaxPendingFriendInvitesReached; // 0x0120
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* PinRequiredForFriends; // 0x0128 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* InvalidPin; // 0x0130 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EpicDOBRequired; // 0x0138 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EpicParentEmailRequired; // 0x0140 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EpicUnsupportedCorrectiveActionRequired; // 0x0148
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* EpicConfirmDisplayNameRequired; // 0x0150
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* EpicInvalidDateOfBirth; // 0x0158 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EpicInvalidEmail; // 0x0160 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType* EpicContinuationTokenNotFound; // 0x0168
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* EpicCabinedExtAuthCreateNotAllowed; // 0x0170
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType* EpicEmailNotAllowed; // 0x0178 (0x0008)
[0x0000000000000002] (CPF_Const)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EpicErrors_X");
}

```

```

return uClassPointer;
};

```

```

};

// Class ProjectX.__EOSHelpers_X__SendHttpRequestEASAuth_0x1
// 0x0090 (0x0060 - 0x00F0)
class U__EOSHelpers_X__SendHttpRequestEASAuth_0x1 : public UObject
{
public:
class FString URL; // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

```

```

struct FScriptDelegate          Callback;                // 0x0070 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString                   Verb;                    // 0x0088 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
unsigned long                   bAddContentTypeHeader : 1; // 0x0098 (0x0004)
[0x000000000000000000] [0x000000001]
class UClass*                   ResponseClass;            // 0x00A0 (0x0008)
[0x000000000000000000]
struct FScriptDelegate          ConvertErrorNew;         // 0x00A8 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)
class UOnlineSubsystem*         EOS;                     // 0x00C0 (0x0008)
[0x000000000000000000]
int32_t                         LocalPlayerNum;          // 0x00C8 (0x0004)
[0x000000000000000000]
class FString                   BodyJson;                // 0x00D0 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString                   Continuation;            // 0x00E0 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__EOSHelpers_X__SendHTTPRequestEASAuth_0x1");
}

```

```

return uClassPointer;
};

```

```

void __EOSHelpers_X__SendHTTPRequestEASAuth_0x1(unsigned long bSuccess, class FString
EASAuthTicket);
};

```

```

// Class ProjectX.__EOSHelpers_X__SendHTTPRequestSelectAuth_0x1
// 0x0038 (0x0060 - 0x0098)
class U__EOSHelpers_X__SendHTTPRequestSelectAuth_0x1 : public UObject
{

```

```

public:
struct FScriptDelegate          Callback;                // 0x0060 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate          ConvertErrorNew;         // 0x0078 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)
class UClass*                   ResponseClass;            // 0x0090 (0x0008)
[0x000000000000000000]

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class
ProjectX.__EOSHelpers_X__SendHTTPRequestSelectAuth_0x1");
}

return uClassPointer;
};

void __EOSHelpers_X__SendHTTPRequestSelectAuth_0x1(class UWebRequest_X* Response);
};

// Class ProjectX.__EpicFriendsPlugin_X__AcceptFriendRequest_0x1
// 0x0048 (0x0060 - 0x00A8)
class U__EpicFriendsPlugin_X__AcceptFriendRequest_0x1 : public UObject
{
public:
    struct FUniqueNetId                FriendId;                // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__EpicFriendsPlugin_X__AcceptFriendRequest_0x1");
        }

        return uClassPointer;
    };

    void __EpicFriendsPlugin_X__AcceptFriendRequest_0x1(class
UEOS_ManageFriendsListResponse* R, class UError* E);
};

// Class ProjectX.__EpicFriendsPlugin_X__AddFriendWithCustomCallback_0x1
// 0x0060 (0x0060 - 0x00C0)
class U__EpicFriendsPlugin_X__AddFriendWithCustomCallback_0x1 : public UObject
{
public:
    struct FScriptDelegate                Callback;                // 0x0060 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)
    struct FUniqueNetId                FriendId;                // 0x0078 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)

```

```

{
    UClassPointer = UObject::FindClass("Class
ProjectX.__EpicFriendsPlugin_X__AddFriendWithCustomCallback_0x1");
}

return UClassPointer;
};

void __EpicFriendsPlugin_X__AddFriendWithCustomCallback_0x1(class
UEOS_ManageFriendsListResponse* R, class UError* E);
};

// Class ProjectX.__EpicFriendsPlugin_X__GetOutgoingFriendRequestsWithCustomCallback_0x1
// 0x0018 (0x0060 - 0x0078)
class U__EpicFriendsPlugin_X__GetOutgoingFriendRequestsWithCustomCallback_0x1 : public
UObject
{
public:
    struct FScriptDelegate                Callback;                // 0x0060 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* UClassPointer = nullptr;

        if (!UClassPointer)
        {
            UClassPointer = UObject::FindClass("Class
ProjectX.__EpicFriendsPlugin_X__GetOutgoingFriendRequestsWithCustomCallback_0x1");
        }

        return UClassPointer;
    };

    void __EpicFriendsPlugin_X__GetOutgoingFriendRequestsWithCustomCallback_0x1(class
UEOS_GetAccountsResponse* R, class UError* E);
};

// Class ProjectX.__EpicFriendsPlugin_X__RejectFriendRequest_0x1
// 0x0048 (0x0060 - 0x00A8)
class U__EpicFriendsPlugin_X__RejectFriendRequest_0x1 : public UObject
{
public:
    struct FUniqueNetId                FriendId;                // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* UClassPointer = nullptr;

        if (!UClassPointer)
        {

```

```

uClassPointer = UObject::FindClass("Class
ProjectX.__EpicFriendsPlugin_X__RejectFriendRequest_0x1");
}

return uClassPointer;
};

void __EpicFriendsPlugin_X__RejectFriendRequest_0x1(class
UEOS_ManageFriendsListResponse* R, class UError* E);
};

// Class ProjectX.__EpicFriendsPlugin_X__RemoveFriend_0x1
// 0x0048 (0x0060 - 0x00A8)
class U__EpicFriendsPlugin_X__RemoveFriend_0x1 : public UObject
{
public:
    struct FUniqueNetId                FriendId;                // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__EpicFriendsPlugin_X__RemoveFriend_0x1");
        }

        return uClassPointer;
    };

    void __EpicFriendsPlugin_X__RemoveFriend_0x1(class UEOS_ManageFriendsListResponse* R,
class UError* E);
};

// Class ProjectX.__EpicLogin_X__HandleLoginChanged_0x1
// 0x0001 (0x0060 - 0x0061)
class U__EpicLogin_X__HandleLoginChanged_0x1 : public UObject
{
public:
    uint8_t                            InLocalPlayerNum;        // 0x0060 (0x0001)
    [0x000000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.__EpicLogin_X__HandleLoginChanged_0x1");
        }
    }

```

```
return uClassPointer;  
};
```

```
void __EpicLogin_X__HandleLoginChanged_0x1();  
};
```

```
// Class ProjectX.EpicConfig_X
```

```
// 0x0178 (0x0078 - 0x01F0)
```

```
class UEpicConfig_X : public UOnlineConfig_X
```

```
{
```

```
public:
```

```
unsigned long                bAllowRemoteAvatars : 1;                // 0x0078 (0x0004)  
[0x0000000000000001] [0x00000001] (CPF_Edit)  
unsigned long                bPollDuringAccountLinking : 1;        // 0x0078 (0x0004)  
[0x0000000000000001] [0x00000002] (CPF_Edit)  
unsigned long                bEnforcePinRequirementForFriends : 1;   // 0x0078  
(0x0004) [0x0000000000000000] [0x00000004]  
unsigned long                bEnforceCabinedMode : 1;              // 0x0078 (0x0004)  
[0x0000000000000000] [0x00000008]  
unsigned long                bPromptForPin : 1;                    // 0x0078 (0x0004)  
[0x0000000000000000] [0x00000010]  
unsigned long                bForceEnableTrade : 1;                // 0x0078 (0x0004)  
[0x0001000000000000] [0x00000020]  
float                        RemoteAvatarPermissionRequestDelay;    // 0x007C (0x0004)  
[0x0000000000000001] (CPF_Edit)  
int32_t                      SecondsBetweenPolling;                // 0x0080 (0x0004)  
[0x0000000000000001] (CPF_Edit)  
int32_t                      SecondsBeforeRequestsTimeout;         // 0x0084 (0x0004)  
[0x0000000000000001] (CPF_Edit)  
int32_t                      SecondsToWaitBeforeRetryingAuth;      // 0x0088 (0x0004)  
[0x0000000000000001] (CPF_Edit)  
int32_t                      SecondsToWaitBeforeUpdatingFriendsList; // 0x008C  
(0x0004) [0x0000000000000001] (CPF_Edit)  
TArray<class ULocalizedAccountLinkURL*> AccountLinkURLOverrides;    //  
0x0090 (0x0010) [0x0000000000400001] (CPF_Edit | CPF_NeedCtorLink)  
class FString                DefaultAccountLinkURL;                // 0x00A0 (0x0010)  
[0x0000000000400001] (CPF_Edit | CPF_NeedCtorLink)  
class FString                PermissionServerURL;                  // 0x00B0 (0x0010)  
[0x0000000000400000] (CPF_NeedCtorLink)  
class FString                IdentityServerURL;                    // 0x00C0 (0x0010)  
[0x0000000000400000] (CPF_NeedCtorLink)  
class FString                CabinedModeURL;                       // 0x00D0 (0x0010)  
[0x0000000000400000] (CPF_NeedCtorLink)  
class FString                SetDOBURL;                             // 0x00E0 (0x0010)  
[0x0000000000400000] (CPF_NeedCtorLink)  
class FString                CreateAccountWithDOBURL;              // 0x00F0 (0x0010)  
[0x0000000000400000] (CPF_NeedCtorLink)  
class FString                SetParentEmailURL;                    // 0x0100 (0x0010)  
[0x0000000000400000] (CPF_NeedCtorLink)  
class FString                AgeGateURL;                            // 0x0110 (0x0010)  
[0x0000000000400000] (CPF_NeedCtorLink)  
class FString                ConfirmDisplayNameURL;                 // 0x0120 (0x0010)  
[0x0000000000400000] (CPF_NeedCtorLink)
```

```

class FString                                EASAuthURL;                                // 0x0130 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString                                CreateAccountURL;                            // 0x0140 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
TArray<struct FChatPermissionPair>            ChatPermissionPairs;                            // 0x0150
(0x0010) [0x0000000000040000] (CPF_NeedCtorLink)
uint8_t                                       DefaultChatPermissionLevel;                        // 0x0160 (0x0001)
[0x0000000000000000]
struct FPlayerPermissions                    DefaultPermissions;                                // 0x0164 (0x0008)
[0x0000000000000000]
class FString                                ForgotPinURL;                                       // 0x0170 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString                                ParentalControlsURL;                              // 0x0180 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString                                PrivacyPolicyURL;                                  // 0x0190 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString                                ActivateURL;                                       // 0x01A0 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString                                AccountURL;                                       // 0x01B0 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString                                TOSURL;                                           // 0x01C0 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString                                SandboxId;                                        // 0x01D0 (0x0010)
[0x00000000000404001] (CPF_Edit | CPF_Config | CPF_NeedCtorLink)
class FString                                DeploymentId;                                     // 0x01E0 (0x0010)
[0x00000000000404001] (CPF_Edit | CPF_Config | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EpicConfig_X");
}

return uClassPointer;
};

uint8_t GetChatPermissionLevel(class FString PermissionLabel);
void Apply();
};

// Class ProjectX.__EpicLogin_X__TriggerAuthTicketDelegate_0x1
// 0x0060 (0x0060 - 0x00C0)
class U__EpicLogin_X__TriggerAuthTicketDelegate_0x1 : public UObject
{
public:
struct FScriptDelegate                      Callback;                                           // 0x0060 (0x0018)
[0x0000000000040000] (CPF_NeedCtorLink)
struct FUniqueNetId                        NetId;                                             // 0x0078 (0x0048)
[0x0000000000040000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__EpicLogin_X__TriggerAuthTicketDelegate_0x1");
}

return uClassPointer;
};

void __EpicLogin_X__TriggerAuthTicketDelegate_0x1(unsigned long bSuccess, class FString
AuthTicket);
};

// Class ProjectX.__EpicLogin_X__UpdateTwoFactorAuthenticationStatus_0x1
// 0x0018 (0x0060 - 0x0078)
class U__EpicLogin_X__UpdateTwoFactorAuthenticationStatus_0x1 : public UObject
{
public:
struct FScriptDelegate          Callback;          // 0x0060 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__EpicLogin_X__UpdateTwoFactorAuthenticationStatus_0x1");
}

return uClassPointer;
};

void
__EpicLogin_X__UpdateTwoFactorAuthenticationStatus_0x1__EpicLogin_X__UpdateTwoFactor
AuthenticationStatus_0x1_0x1(class FString Ticket, class FString Id, class UError* Err);
void __EpicLogin_X__UpdateTwoFactorAuthenticationStatus_0x1(unsigned long bSuccess, class
FString AuthTicket);
};

// Class ProjectX.__EpicLogin_X__RequestNintendoAccountAuthorization_0x1
// 0x0018 (0x0060 - 0x0078)
class U__EpicLogin_X__RequestNintendoAccountAuthorization_0x1 : public UObject
{
public:
struct FScriptDelegate          Callback;          // 0x0060 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)

```



```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__EpicLogin_X__RequestNintendoAccountAuthorization_0x1");
}

return uClassPointer;
};

void __EpicLogin_X__RequestNintendoAccountAuthorization_0x1(class FString
NintendoAccountToken);
};

// Class ProjectX.CabinedModeResponse
// 0x0010 (0x0060 - 0x0070)
class UCabinedModeResponse : public UObject
{
public:
TArray<struct FCabinedModeData>          ArrayOfCabinedModeData;          // 0x0060
(0x0010) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CabinedModeResponse");
}

return uClassPointer;
};
};

// Class ProjectX.__LocalClubData_X__GetClubByID_0x1
// 0x0008 (0x0060 - 0x0068)
class U__LocalClubData_X__GetClubByID_0x1 : public UObject
{
public:
uint64_t                                Id;                                // 0x0060 (0x0008)
[0x0001000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.__LocalClubData_X__GetClubByID_0x1");
}

return uClassPointer;
};

bool __LocalClubData_X__GetClubByID_0x1(class UClubDetails_X* C);
};

// Class ProjectX.ClubSettings_X
// 0x0028 (0x0060 - 0x0088)
class UClubSettings_X : public UObject
{
public:
    class FString                      ClubName;                      // 0x0060 (0x0010)
    [0x0001000000400000] (CPF_NeedCtorLink)
    class FString                      ClubTag;                      // 0x0070 (0x0010)
    [0x0001000000400000] (CPF_NeedCtorLink)
    int32_t                            PrimaryColor;                // 0x0080 (0x0004)
    [0x0001000000000000]
    int32_t                            AccentColor;                 // 0x0084 (0x0004)
    [0x0001000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.ClubSettings_X");
        }

        return uClassPointer;
    };

    bool IsAccentColorSet();
    bool IsPrimaryColorSet();
    static class UClubSettings_X* Create(class FString InName, class FString InTag, int32_t
    InPrimaryColor, int32_t InSecondaryColor);
};

// Class ProjectX.ClubDetails_X
// 0x0080 (0x0088 - 0x0108)
class UClubDetails_X : public UClubSettings_X
{
public:
    uint64_t                            ClubID;                      // 0x0088 (0x0008)
    [0x0001000000000000]
    struct FUniqueNetId                  OwnerPlayerID;              // 0x0090 (0x0048)
    [0x0001000000400000] (CPF_NeedCtorLink)

```

```

class FString                                MotD;                                // 0x00D8 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
unsigned long                                bVerified : 1;                                // 0x00E8 (0x0004)
[0x0001000000000000] [0x000000001]
uint64_t                                    LastUpdatedTime;                                // 0x00F0 (0x0008)
[0x0001000000000000]
TArray<struct FClubMember>                    Members;                                // 0x00F8 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ClubDetails_X");
}

return uClassPointer;
};

class FString GetMemberDebugString(struct FClubMember Member);
class FString GetMembersDebugString();
class FString GetDebugString();
bool IsMember(struct FUniqueNetId PlayerID);
bool IsDestroyed();
void Destroy();
};

// Class ProjectX.LocalClubData_X
// 0x0024 (0x0060 - 0x0084)
class ULocalClubData_X : public UObject
{
public:
TArray<class UClubDetails_X*>                Clubs;                                // 0x0060 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
TArray<struct FPlayerClubPair>                Invites;                                // 0x0070 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
int32_t                                    NextClubID;                                // 0x0080 (0x0004)
[0x0001000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.LocalClubData_X");
}

return uClassPointer;
};

```

```

class UClubDetails_X* GetClubForPlayer(struct FUniqueNetId PlayerID);
class UClubDetails_X* GetClubByID(uint64_t Id);
void Save();
void Load();
};

// Class ProjectX.__LocalClubData_X__GetClubForPlayer_0x1
// 0x0048 (0x0060 - 0x00A8)
class U__LocalClubData_X__GetClubForPlayer_0x1 : public UObject
{
public:
    struct FUniqueNetId                PlayerID;                // 0x0060 (0x0048)
    [0x0001000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__LocalClubData_X__GetClubForPlayer_0x1");
        }

        return uClassPointer;
    };

    bool __LocalClubData_X__GetClubForPlayer_0x1(class UClubDetails_X* C);
};

// Class ProjectX.__MatchRecorder_X__GetPlayerData_0x1
// 0x0048 (0x0060 - 0x00A8)
class U__MatchRecorder_X__GetPlayerData_0x1 : public UObject
{
public:
    struct FUniqueNetId                PlayerID;                // 0x0060 (0x0048)
    [0x0000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.__MatchRecorder_X__GetPlayerData_0x1");
        }

        return uClassPointer;
    };

    bool __MatchRecorder_X__GetPlayerData_0x1(class UMatchPlayerData_X* P);
};

```

```

};

// Class ProjectX.MatchPlayerData_X
// 0x010C24 (0x0060 - 0x016C84)
class UMatchPlayerData_X : public UObject
{
public:
    struct FUniqueNetId PlayerID; // 0x0060 (0x0048)
    [0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
    class FString PlayerName; // 0x00A8 (0x0010)
    [0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
    uint64_t ConnectTimestamp; // 0x00B8 (0x0008)
    [0x0000000000000000]
    uint64_t JoinTimestamp; // 0x00C0 (0x0008)
    [0x0000000000000000]
    uint64_t LeaveTimestamp; // 0x00C8 (0x0008)
    [0x0000000000000000]
    struct FUniqueNetId unsigned long InPartyLeaderID; : 1; //
    0x00D0 (0x00048) [0x0000000040040000] (CPF_NeedCtorLink00000) [0x00000001]
    (CPF_EditInlineNotify)
    unsigned long bAbandoned : 1; // 0x01180D0 (0x0004)
    [0x0000000000000000] [0x000000012]
    unsigned long bMvp : 1; // 0x01180D0 (0x0004)
    [0x0000000040000000] [0x000000024] (CPF_EditInlineNotify)
    struct FUniqueNetId PartyLeaderID; // 0x00D8 (0x0048)
    [0x0000000000400000] (CPF_NeedCtorLink)
    int32_t LastTeam; // 0x011C20 (0x0004)
    [0x0000000040000000] (CPF_EditInlineNotify)
    class FString TeamColor; // 0x0128 (0x0010)
    [0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
    float SecondsPlayed; // 0x012038 (0x0004)
    [0x0000000000000000]
    int32_t Goals; // 0x01243C (0x0004)
    [0x0000000040000000] (CPF_EditInlineNotify)
    int32_t Assists; // 0x012840 (0x0004)
    [0x0000000040000000] (CPF_EditInlineNotify)
    int32_t Saves; // 0x012C44 (0x0004)
    [0x0000000040000000] (CPF_EditInlineNotify)
    int32_t Shots; // 0x013048 (0x0004)
    [0x0000000040000000] (CPF_EditInlineNotify)
    int32_t Demolishes; // 0x0134C (0x0004)
    [0x0000000040000000] (CPF_EditInlineNotify)
    int32_t Score; // 0x013850 (0x0004)
    [0x0000000040000000] (CPF_EditInlineNotify)
    int32_t OwnGoals; // 0x013C54 (0x0004)
    [0x0000000040000000] (CPF_EditInlineNotify)
    int32_t ClubID; // 0x014058 (0x0004)
    [0x0000000000000000]
    struct FMatchSkillUpdate Skills; // 0x01445C (0x0024)
    [0x0000000000000000]
    int32_t ActorID; // 0x01680 (0x0004)
    [0x0000000000000000]

public:

```

```

static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.MatchPlayerData_X");
}

return uClassPointer;
};

void SetSkills(struct FUpdatedPlayerSkillRating Update);
};

// Class ProjectX.MatchRecorder_X
// 0x0018 (0x0060 - 0x0078)
class UMatchRecorder_X : public UObject
{
public:
class UClass* MatchDataClass; // 0x0060 (0x0008)
[0x0000000000000001] (CPF_Edit)
class UClass* PlayerDataClass; // 0x0068 (0x0008)
[0x0000000000000001] (CPF_Edit)
class UMatchData_X* Match; // 0x0070 (0x0008)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.MatchRecorder_X");
}

return uClassPointer;
};

bool __MatchRecorder_X_GetAbandonedPlayers_0x1(class UMatchPlayerData_X* P);
struct FUniqueNetId __MatchRecorder_X_GetAbandonedPlayerIDs_0x1(class
UMatchPlayerData_X* P);
void Finished();
void UpdateServer(class UOnlineGameDedicatedServer_X* Server);
class UMatchPlayerData_X* CreatePlayer(struct FUniqueNetId PlayerID, class FString
PlayerName);
TArray<struct FUniqueNetId> GetAbandonedPlayerIDs();
TArray<class UMatchPlayerData_X*> GetAbandonedPlayers();
class UMatchPlayerData_X* GetPlayerData(struct FUniqueNetId PlayerID);
void RemovePlayer(struct FUniqueNetId PlayerID);
void ReservePlayer(struct FUniqueNetId PlayerID, struct FUniqueNetId PartyID);
void AddPlayer(struct FUniqueNetId PlayerID, class FString PlayerName);
class FString MatchGuid();

```

```

void eventConstruct();
};

// Class ProjectX.__OnlineClubCache_X__GetClubDetails_0x1
// 0x0008 (0x0060 - 0x0068)
class U__OnlineClubCache_X__GetClubDetails_0x1 : public UObject
{
public:
uint64_t                ClubID;                // 0x0060 (0x0008)
[0x0001000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineClubCache_X__GetClubDetails_0x1");
}

return uClassPointer;
};

bool __OnlineClubCache_X__GetClubDetails_0x1(class UClubDetails_X* C);
};

// Class ProjectX.OnlineClubCache_X
// 0x0028 (0x0070 - 0x0098)
class UOnlineClubCache_X : public UComponent
{
public:
TArray<class UClubDetails_X*>                ClubDetailsCache;                // 0x0070
(0x0010) [0x0001000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                __EventClubUpdated__Delegate;                // 0x0080
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineClubCache_X");
}

return uClassPointer;
};

uint64_t __OnlineClubCache_X__ClearAllExcept_0x1(class UClubDetails_X* C);
void ClearAllExcept(TArray<uint64_t>& KeepClubs);
void Clear();

```

```

void Destroy(uint64_t ClubID);
void Add(class UClubDetails_X* ClubDetails);
uint64_t GetPlayerClubID(struct FUniqueNetId PlayerID);
class UClubDetails_X* GetPlayerClubDetails(struct FUniqueNetId PlayerID);
class UClubDetails_X* GetClubDetails(uint64_t ClubID);
void NotifyWhenClubUpdated(struct FScriptDelegate Callback);
void EventClubUpdated(class UOnlineClubCache_X* Cache, class UClubDetails_X* ClubDetails);
};

// Class ProjectX.__OnlineClubCache_X__GetPlayerClubDetails_0x1
// 0x0048 (0x0060 - 0x00A8)
class U__OnlineClubCache_X__GetPlayerClubDetails_0x1 : public UObject
{
public:
    struct FUniqueNetId                PlayerID;                // 0x0060 (0x0048)
    [0x0001000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__OnlineClubCache_X__GetPlayerClubDetails_0x1");
        }

        return uClassPointer;
    };

    bool __OnlineClubCache_X__GetPlayerClubDetails_0x1(class UClubDetails_X* C);
};

// Class ProjectX.__OnlineClubManager_X__LeaveClub_0x1
// 0x0008 (0x0060 - 0x0068)
class U__OnlineClubManager_X__LeaveClub_0x1 : public UObject
{
public:
    uint64_t                ClubID;                // 0x0060 (0x0008)
    [0x0001000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.__OnlineClubManager_X__LeaveClub_0x1");
        }

        return uClassPointer;
    };
};

```



```

void __OnlineClubManager_X__LeaveClub_0x1();
};

// Class ProjectX.OnlineClubManager_X
// 0x0020 (0x0060 - 0x0080)
class UOnlineClubManager_X : public UObject
{
public:
class UPsyNetConnection_X*          Connection;          // 0x0060 (0x0008)
[0x0001000000000000]
struct FScriptDelegate              __EventClubChanged__Delegate;      // 0x0068
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineClubManager_X");
}

return uClassPointer;
};

void OnLeaveClub(uint64_t ClubID);
void OnClubChanged(class UClubDetails_X* Club);
class UAsyncTask* LeaveClub(uint64_t ClubID);
class UAsyncTask* RejectClubInvite(uint64_t ClubID);
class UAsyncResult__ClubDetails_X* AcceptClubInvite(uint64_t ClubID);
class UAsyncResult__array_ClubInvite_X* SyncClubInvites();
class UAsyncResult__ClubDetails_X* SetClubOwner(struct FUniqueNetId NewOwner);
class UAsyncResult__ClubDetails_X* RemoveFromClub(struct FUniqueNetId PlayerID);
class UAsyncTask* InviteToClub(struct FUniqueNetId PlayerID);
class UAsyncResult__ClubDetails_X* UpdateClubMotD(class FString Text);
class UAsyncResult__ClubDetails_X* UpdateClubColors(int32_t Primary, int32_t Accent);
class UAsyncResult__ClubDetails_X* UpdateClubName(class FString ClubName, class FString ClubTag);
void OnClubCreated(class UClubDetails_X* Club, class UError* ActionError);
class UAsyncResult__ClubDetails_X* CreateClub(class UClubSettings_X* Settings);
void EventClubChanged(class UOnlineClubManager_X* Manager, uint64_t ClubID);
};

// Class ProjectX.__OnlineClubProvider_X__HandleClubSynced_0x1
// 0x0008 (0x0060 - 0x0068)
class U__OnlineClubProvider_X__HandleClubSynced_0x1 : public UObject
{
public:
class URPC_GetClubDetails_X*          RPC;          // 0x0060 (0x0008)
[0x0001000000000000]

public:

```

```

static UClass* StaticClass()
{
    static UClass* uClassPointer = nullptr;

    if (!uClassPointer)
    {
        uClassPointer = UObject::FindClass("Class
        ProjectX.__OnlineClubProvider_X__HandleClubSynced_0x1");
    }

    return uClassPointer;
};

void __OnlineClubProvider_X__HandleClubSynced_0x1(struct FClubMember Member);
};

// Class ProjectX.RPC_ClubDetailsBase_X
// 0x0010 (0x00E8 - 0x00F8)
class URPC_ClubDetailsBase_X : public URPC_X
{
public:
    class UClubDetails_X*                ClubDetails;                // 0x00E8 (0x0008)
    [0x00010000000002000] (CPF_Transient)
    class UTAsyncResult__ClubDetails_X*  ClubDetailsTask;          // 0x00F0
    (0x0008) [0x00010000000002000] (CPF_Transient)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_ClubDetailsBase_X");
        }

        return uClassPointer;
    };

    class UClubDetails_X* __RPC_ClubDetailsBase_X__CreateClubDetailsTask_0x1();
    class UTAsyncResult__ClubDetails_X* CreateClubDetailsTask();
};

// Class ProjectX.RPC_GetClubDetails_X
// 0x0004 (0x00F8 - 0x00FC)
class URPC_GetClubDetails_X : public URPC_ClubDetailsBase_X
{
public:
    int32_t                ClubID;                // 0x00F8 (0x0004)
    [0x000100000000000000]

public:
    static UClass* StaticClass()
    {

```

```

static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetClubDetails_X");
}

return uClassPointer;
};

class URPC_GetClubDetails_X* SetClubID(uint64_t InClubID);
};

// Class ProjectX.OnlineClubProvider_X
// 0x0068 (0x0060 - 0x00C8)
class UOnlineClubProvider_X : public UObject
{
public:
    class UOnlineClubCache_X*          Cache; // 0x0060 (0x0008)
    [0x0001800004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
    TArray<class URPC_GetClubDetails_X*> SyncingClubs; // 0x0068
    (0x0010) [0x0001000000400000] (CPF_NeedCtorLink)
    TArray<class URPC_GetPlayerClubDetails_X*> SyncingPlayers; // 0x0078
    (0x0010) [0x0001000000400000] (CPF_NeedCtorLink)
    TArray<struct FPlayerClubSyncResult> PlayerClubSyncResults; // 0x0088
    (0x0010) [0x0001000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate __EventSyncComplete__Delegate; // 0x0098
    (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate __EventPlayerSynced__Delegate; // 0x00B0
    (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.OnlineClubProvider_X");
        }

        return uClassPointer;
    };

    void ClubSynced(class UClubDetails_X* Club);
    bool IsSyncing();
    void NotifyWhenSyncComplete(struct FScriptDelegate Callback);
    struct FPlayerClubSyncResult GetPlayerClubSyncResult(struct FUniqueNetId PlayerID);
    void SetPlayerClubSyncResult(struct FUniqueNetId PlayerID, class UError* Error);
    void HandlePlayerSynced(class URPC_GetPlayerClubDetails_X* RPC);
    class UAsyncResult__ClubDetails_X* SyncPlayerClubDetails(struct FUniqueNetId PlayerID);
    class UAsyncResult__ClubDetails_X* GetPlayerClubDetails(struct FUniqueNetId PlayerID);
    void HandleClubSynced(class URPC_GetClubDetails_X* RPC);
    class UAsyncResult__ClubDetails_X* SyncClubDetails(uint64_t ClubID);

```

```

class UAsyncResult__ClubDetails_X* GetClubDetails(uint64_t ClubID);
void EventPlayerSynced(class UOnlineClubProvider_X* Provider, struct FUniqueNetId PlayerID);
void EventSyncComplete(class UOnlineClubProvider_X* Provider);
};

// Class ProjectX.__OnlineClubProvider_X__SyncClubDetails_0x1
// 0x0010 (0x0060 - 0x0070)
class U__OnlineClubProvider_X__SyncClubDetails_0x1 : public UObject
{
public:
uint64_t ClubID; // 0x0060 (0x0008)
[0x0001000000000000]
class URPC_GetClubDetails_X* RPC; // 0x0068 (0x0008)
[0x0001000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineClubProvider_X__SyncClubDetails_0x1");
}

return uClassPointer;
};

void __OnlineClubProvider_X__SyncClubDetails_0x2(class URPC_X* _);
bool __OnlineClubProvider_X__SyncClubDetails_0x1(class URPC_GetClubDetails_X* R);
};

// Class ProjectX.__OnlineClubProvider_X__SyncPlayerClubDetails_0x1
// 0x0050 (0x0060 - 0x00B0)
class U__OnlineClubProvider_X__SyncPlayerClubDetails_0x1 : public UObject
{
public:
struct FUniqueNetId PlayerID; // 0x0060 (0x0048)
[0x0001000000400000] (CPF_NeedCtorLink)
class URPC_GetPlayerClubDetails_X* RPC; // 0x00A8 (0x0008)
[0x0001000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineClubProvider_X__SyncPlayerClubDetails_0x1");
}
}

```

```

return uClassPointer;
};

void __OnlineClubProvider_X__SyncPlayerClubDetails_0x2(class URPC_X* _);
bool __OnlineClubProvider_X__SyncPlayerClubDetails_0x1(class URPC_GetPlayerClubDetails_X*
R);
};

// Class ProjectX.RPC_GetPlayerClubDetails_X
// 0x0048 (0x00F8 - 0x0140)
class URPC_GetPlayerClubDetails_X : public URPC_ClubDetailsBase_X
{
public:
    struct FUniqueNetId                PlayerID;                // 0x00F8 (0x0048)
    [0x0001000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetPlayerClubDetails_X");
        }

        return uClassPointer;
    };

    class URPC_GetPlayerClubDetails_X* SetPlayerID(struct FUniqueNetId InPlayerId);
};

// Class ProjectX.__OnlineGame_X__CheckPsyNetConnection_0x1
// 0x0008 (0x0060 - 0x0068)
class U__OnlineGame_X__CheckPsyNetConnection_0x1 : public UObject
{
public:
    class UAsyncTask*                Task;                // 0x0060 (0x0008)
    [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGame_X__CheckPsyNetConnection_0x1");
        }

        return uClassPointer;
    };
};

```

```

void __OnlineGame_X__CheckPsyNetConnection_0x1(class UOnlinePlayerAuthentication_X*
Auth);
};

// Class ProjectX.__OnlineGame_X__GetOnlinePlayerFromEpicId_0x1
// 0x0010 (0x0060 - 0x0070)
class U__OnlineGame_X__GetOnlinePlayerFromEpicId_0x1 : public UObject
{
public:
class FString                                EpicAccountId;                // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGame_X__GetOnlinePlayerFromEpicId_0x1");
}

return uClassPointer;
};

bool __OnlineGame_X__GetOnlinePlayerFromEpicId_0x1(class UOnlinePlayer_X* P);
};

// Class ProjectX.__OnlineGameMatchmaking_X__AddRecommendedServers_0x1
// 0x0004 (0x0060 - 0x0064)
class U__OnlineGameMatchmaking_X__AddRecommendedServers_0x1 : public UObject
{
public:
float                                AccPingThreshold;                // 0x0060 (0x0004)
[0x000000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGameMatchmaking_X__AddRecommendedServers_0x1");
}

return uClassPointer;
};

bool __OnlineGameMatchmaking_X__AddRecommendedServers_0x1(class URegionPing_X* R);
};

```

```

// Class ProjectX.RegionPing_X
// 0x0039 (0x0060 - 0x0099)
class URegionPing_X : public UObject
{
public:
class URegion_X*                Region;                // 0x0060 (0x0008)
[0x0000000000000000]
class FString                  RegionID;                // 0x0068 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FName                   Address;                // 0x0078 (0x0008)
[0x0000000000000000]
class UTcpConnection*          Connection;            // 0x0080 (0x0008)
[0x0000000000000000]
float                          PingSendTime;          // 0x0088 (0x0004)
[0x0000000000000000]
float                          Ping;                  // 0x008C (0x0004)
[0x0000000000000000]
float                          AvgPing;               // 0x0090 (0x0004)
[0x0000000000000000]
int32_t                        PingCount;              // 0x0094 (0x0004)
[0x0000000000000000]
uint8_t                        PingResult;             // 0x0098 (0x0001)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RegionPing_X");
}

return uClassPointer;
};

void Reset();
};

// Class ProjectX.OnlineGameMatchmakingBase_X
// 0x0080 (0x00B0 - 0x0130)
class UOnlineGameMatchmakingBase_X : public UOnline_X
{
public:
class UCheckReservation_X*      CheckReservation;      // 0x00B0
(0x0008) [0x0000000000002000] (CPF_Transient)
class UAsyncTask*              StartMatchmakingTask;   // 0x00B8 (0x0008)
[0x0000000000002000] (CPF_Transient)
class FString                  PingingRegionsString;   // 0x00C0 (0x0010)
[0x000000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
struct FScriptDelegate         __EventFindGameComplete__Delegate; // 0x00D0
(0x0018) [0x000000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate         __EventFindGameStatus__Delegate;   // 0x00E8

```

```

(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventFindGameError__Delegate;          // 0x0100
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventFindGameStateChanged__Delegate;    // 0x0118
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameMatchmakingBase_X");
}

return uClassPointer;
};

```

```

void HandleRegionsPinged(class UOnlineGameRegions_X* InRegions);
void HandleGameStarted(class AGRI_X* GRI);
void HandleStatusUpdate(class FString NewStatus);
void HandleJoinGameComplete(unsigned long bSuccess, class FString FailReason);
struct FName GetMatchmakingStateName();
class UMatchmakingMetrics_X* GetMetrics();
class UCheckReservation_X* CreateCheckReservation();
void GotoMatchmakingState(struct FName StateName);
void ClearStartMatchmakingRPC();
void ClearCheckReservation();
void OnSearchComplete(unsigned long bCanceled);
void Cancel();
bool IsSearching();
class UOnlineGameMatchmakingBase_X* AddFindGameStateChangedDelegate(struct
FScriptDelegate NewDelegate);
class UOnlineGameMatchmakingBase_X* AddFindGameCompleteDelegate(struct
FScriptDelegate HandleFindGameComplete);
class UOnlineGameMatchmakingBase_X* AddFindGameErrorDelegate(struct FScriptDelegate
HandleFindGameError);
class UOnlineGameMatchmakingBase_X* AddFindGameStatusChangedDelegate(struct
FScriptDelegate HandleFindGameStatusChanged);
void EventFindGameStateChanged(struct FName NewState);
void EventFindGameError(class FString NewStatus);
void EventFindGameStatus(class FString NewStatus);
void EventFindGameComplete(unsigned long bCancelled);
};

```

```

// Class ProjectX.OnlineGameMatchmaking_X
// 0x00F0 (0x0130 - 0x0220)
class UOnlineGameMatchmaking_X : public UOnlineGameMatchmakingBase_X
{
public:
TArray<int32_t> PreferredPlaylists;          // 0x0130 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<struct FDSRegionInfo> PreferredRegions;          // 0x0140

```



```

(0x0010) [0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                               SearchingString;                // 0x0150 (0x0010)
[0x0000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                               StartSearchFailString;            // 0x0160 (0x0010)
[0x0000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                               FoundServerString;                // 0x0170 (0x0010)
[0x0000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                               PlaylistsHaveChangedString;      // 0x0180 (0x0010)
[0x0000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                               RegionsHaveChangedString;        // 0x0190 (0x0010)
[0x0000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                               MatchmakingAttemptString;        // 0x01A0 (0x0010)
[0x0000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
float                                       MatchmakingStartTime;            // 0x01B0 (0x0004)
[0x0000004000002000] (CPF_Transient)
int32_t                                    MatchmakingBanTime;              // 0x01B4 (0x0004)
[0x0000004000000000]
float                                       EstimatedQueueTime;              // 0x01B8 (0x0004)
[0x0000004000000000]
unsigned long                               bIgnoreSkill : 1;                // 0x01BC (0x0004)
[0x0000004000002000] [0x00000001] (CPF_Transient)
float                                       MatchmakingDisabledDuration;      // 0x01C0 (0x0004)
[0x0000000000000001] (CPF_Edit)
float                                       MatchmakingDisabledUntilTime;    // 0x01C4 (0x0004)
[0x0000004000002000] (CPF_Transient)
class FString                               LastReservationID;                // 0x01C8 (0x0010)
[0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FScriptDelegate                     __EventStartSearch__Delegate;    // 0x01D8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                     __EventMatchmakingError__Delegate; // 0x01F0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate
__EventMatchmakingCanceledOnPartySizeChanged__Delegate; // 0x0208 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameMatchmaking_X");
}

return uClassPointer;
};

bool AddRecommendedServers(float PingThreshold, float PingIncrement, float MaxPing);
void RecordStart(unsigned long bUseRecommendedRegions);
void StartMatchmaking();
void UpdateMatchmaking();
void SendMatchmakingState();
void HandleError(class UError* Error);

```

```

void HandleMatchmakingStartSuccessRPC(class URPC_StartMatchmaking_X* RPC);
void HandleStartSearch(class UAsyncTask* Task);
void OnReceiveGameServer(struct FServerReservationData Reservation);
void SetupForTimeConstraints();
void OnPlaylistTimeEnded();
struct FDSRegionInfo __OnlineGameMatchmaking_X__StartSearch_0x1(class URegionPing_X* R);
class FString __OnlineGameMatchmaking_X__GetLocalizedPlaylistsString_0x1(int32_t Playlist);
class FString __OnlineGameMatchmaking_X__GetRegionsString_0x1(struct FDSRegionInfo R);
class FString __OnlineGameMatchmaking_X__GetLocalizedRegionsString_0x3(class USuperRegion_X* SuperRegion);
class USuperRegion_X* __OnlineGameMatchmaking_X__GetLocalizedRegionsString_0x2(class URegion_X* R);
class URegion_X* __OnlineGameMatchmaking_X__GetLocalizedRegionsString_0x1(struct FDSRegionInfo R);
class FString __OnlineGameMatchmaking_X__GetDebugSuperRegionString_0x1(class URegionPing_X* R);
class FString __OnlineGameMatchmaking_X__BeginState_0x1(struct FDSRegionInfo R);
struct FDSRegionInfo __OnlineGameMatchmaking_X__AddRecommendedServers_0x3(class URegionPing_X* R);
struct FDSRegionInfo __OnlineGameMatchmaking_X__AddRecommendedServers_0x2(class URegionPing_X* R);
class FString __OnlineGameMatchmaking_X__RecordStart_0x1(struct FDSRegionInfo R);
void PrintDebugInfo(class UDebugDrawer* Drawer);
bool IsMatchmakingDisabled();
void SetSkillIgnored(unsigned long bIgnore);
void ToggleSkill();
void ClearMatchmakingBanTime();
void SetMatchmakingBanTime(int32_t Duration);
void HandlePartyLeaderChanged(class UOnlineGameParty_X* PartyObject, struct FUniqueNetId NewLeader);
void HandlePartySizeChanged(class UOnlineGameParty_X* PartyObject, int32_t NewSize, int32_t OldSize);
void HandlePartyDestroyed(class UOnlineGameParty_X* PartyObject);
void OnFindGameWarning(class FString FailReason);
class FString GetGameServerID();
int32_t SortPlaylists(int32_t A, int32_t B);
class FString GetPartyMembersString();
TArray<class URegionPing_X*> GetSubRegionPings(class FString SuperRegionID);
class FString GetDebugSuperRegionString(class USuperRegion_X* SuperRegion);
class FString GetLocalizedRegionsString();
class FString GetRegionsString();
class FString GetLocalizedPlaylistsString();
class FString GetPlaylistsString();
void OnExit();
void OnSearchComplete(unsigned long bCanceled);
void HandleInternetConnectionChanged(unsigned long bConnected);
class UError* StartSearch(TArray<int32_t> InPreferredPlaylists, TArray<class FString> InPreferredSuperRegions);
int32_t ShiftPingToMS(float Ping);
bool VerifyPlaylist(int32_t InPlaylistID);
void OnInit();
void EventMatchmakingCanceledOnPartySizeChanged(class UOnlineGameMatchmaking_X* InMatchMaking);

```

```

void EventMatchmakingError(class UOnlineGameMatchmaking_X* InMatchMaking, class UError*
Error);
void EventStartSearch(class UOnlineGameMatchmakingBase_X* InMatchMaking);
};

// Class ProjectX.__OnlineGameMatchmaking_X__GetSubRegionPings_0x1
// 0x0010 (0x0060 - 0x0070)
class U__OnlineGameMatchmaking_X__GetSubRegionPings_0x1 : public UObject
{
public:
class FString                               SuperRegionID;                // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGameMatchmaking_X__GetSubRegionPings_0x1");
}

return uClassPointer;
};

bool __OnlineGameMatchmaking_X__GetSubRegionPings_0x1(class URegionPing_X* R);
};

// Class ProjectX.Region_X
// 0x0040 (0x0060 - 0x00A0)
class URegion_X : public UObject
{
public:
class FString                               SuperRegionID;                // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString                               Id;                            // 0x0070 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString                               Label;                          // 0x0080 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString                               Secret;                         // 0x0090 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Region_X");
}
}

```

```

return uClassPointer;
};

};

// Class ProjectX.__OnlineGameMatchmaking_X__OnReceiveGameServer_0x1
// 0x0070 (0x0060 - 0x00D0)
class U__OnlineGameMatchmaking_X__OnReceiveGameServer_0x1 : public UObject
{
public:
    struct FServerReservationData                Reservation;                // 0x0060 (0x0070)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__OnlineGameMatchmaking_X__OnReceiveGameServer_0x1");
        }

        return uClassPointer;
    };

    bool __OnlineGameMatchmaking_X__OnReceiveGameServer_0x1(struct FDSRegionInfo R);
};

// Class ProjectX.__OnlineGameMatchmaking_X__RecordStart_0x2
// 0x0010 (0x0060 - 0x0070)
class U__OnlineGameMatchmaking_X__RecordStart_0x2 : public UObject
{
public:
    TArray<class FString>                RegionIds;                // 0x0060 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__OnlineGameMatchmaking_X__RecordStart_0x2");
        }

        return uClassPointer;
    };

    bool __OnlineGameMatchmaking_X__RecordStart_0x2(class URegionPing_X* R);
};

```

```

// Class ProjectX.__OnlineGameParty_X__BroadcastAllLocalPlayers_0x2
// 0x0170 (0x0060 - 0x01D0)
class U__OnlineGameParty_X__BroadcastAllLocalPlayers_0x2 : public UObject
{
public:
    struct FPartyMember                Primary;                                // 0x0060 (0x0170)
    [0x00000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__OnlineGameParty_X__BroadcastAllLocalPlayers_0x2");
        }

        return uClassPointer;
    };

    bool __OnlineGameParty_X__BroadcastAllLocalPlayers_0x2(struct FPartyMember PM);
};

// Class ProjectX.OnlineGameParty_X
// 0x0390 (0x00B0 - 0x0440)
class UOnlineGameParty_X : public UOnline_X
{
public:
    TArray<struct FPartyMember>          PartyMembers;                        // 0x00B0
    (0x0010) [0x0000000800040200] (CPF_Transient | CPF_NeedCtorLink)
    struct FUniqueLobbyId                PartyID;                            // 0x00C0 (0x0010)
    [0x0000000400000200] (CPF_Transient)
    struct FUniqueNetId                  PartyLeader;                        // 0x00D0 (0x0048)
    [0x0000000400040200] (CPF_Transient | CPF_NeedCtorLink)
    int32_t                              MaxPartySize;                       // 0x0118 (0x0004)
    [0x00000000000000002] (CPF_Const)
    class UOnlineMessageComponent_X*      MessageComponent;                 // 0x0120
    (0x0008) [0x0000000000408000A] (CPF_Const | CPF_ExportObject | CPF_Component |
    CPF_EditInline)
    class UPartyMessage_SearchStatus_X*   PendingSearchStatus;              // 0x0128
    (0x0008) [0x0000000000000200] (CPF_Transient)
    class UPartyMessage_SearchStatus_X*   NullSearchStatus;                 // 0x0130
    (0x0008) [0x0000000000000000]
    int32_t                              PartyTimeout;                       // 0x0138 (0x0004)
    [0x00000000000000003] (CPF_Edit | CPF_Const)
    int32_t                              CurrentPartySize;                   // 0x013C (0x0004)
    [0x0000000400000200] (CPF_Transient)
    unsigned long                         LastbSearchingStatus : 1;           // 0x0140 (0x0004)
    [0x0000000000000200] [0x00000001] (CPF_Transient)
    unsigned long                         LastLockStatus : 1;                 // 0x0140 (0x0004)
    [0x0000000000000200] [0x00000002] (CPF_Transient)

```

```

unsigned long                bPendingIncomingTradeInvite : 1;           // 0x0140 (0x0004)
[0x0001004000002000] [0x00000004] (CPF_Transient)
unsigned long                bTradeLocked : 1;                         // 0x0140 (0x0004)
[0x0009004000000000] [0x00000008]
struct FName                 LastSearchState;                          // 0x0144 (0x0008)
[0x0000000000002000] (CPF_Transient)
struct FPartyJoinMatchSettings MatchSettings;                          // 0x0150 (0x0058)
[0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
uint8_t                     ProcessingStatus;                          // 0x01A8 (0x0001)
[0x0000004000002000] (CPF_Transient)
class FString                NotInSameOnlineGameError;                // 0x01B0 (0x0010)
[0x0000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                MissingLicenseAgreementError;            // 0x01C0 (0x0010)
[0x0000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class UPartySequence_InvitedToPlatformParty_X*
SequenceInvitedToPlatformParty; // 0x01D0 (0x0008) [0x0000004000000000]
class UPartySequence_PsyNetPartyUpgrade_X* SequencePsyNetPartyUpgrade; //
0x01D8 (0x0008) [0x0001004000000000]
class UPartySequence_InvitedToPsyNetParty_X* SequenceInvitedToPsyNetParty; //
0x01E0 (0x0008) [0x0001004000000000]
class UPartyConfig_X*        PartyConfig;                             // 0x01E8 (0x0008)
[0x0000800000000001] (CPF_Edit)
class UPsyNetConfig_X*       Config;                                   // 0x01F0 (0x0008)
[0x0000800000000001] (CPF_Edit)
class UOnlineLobbyInterface* PlatformLobbyInterface_Object;          // 0x01F8
(0x0008) [0x0000004000000000]
class UOnlineLobbyInterface* PlatformLobbyInterface_Interface;        // 0x0200
(0x0008) [0x0000004000000000]
class UParties_X*            PsyNetLobbyInterface;                    // 0x0208 (0x0008)
[0x0001004000000000]
int32_t                     CreatePartyLocalPlayerNum;                // 0x0210 (0x0004)
[0x0000004000000000]
class UCrossplayConfig_X*    CrossplayConfig;                         // 0x0218 (0x0008)
[0x0000800000000000]
class UFindServerTask_X*     FindServerTask;                          // 0x0220 (0x0008)
[0x0000000000000000]
class FString                JoinLobbyError;                          // 0x0228 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
TArray<int32_t>              LeaderPreferredPlaylists;                 // 0x0238 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventPartyCreated__Delegate;          // 0x0248
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __OnPartyInviteAccepted__Delegate;      // 0x0260
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventPlayerInvited__Delegate;         // 0x0278
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventPlayerInvitedSilent__Delegate;    // 0x0290
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventPartyChanged__Delegate;          // 0x02A8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventPartyMemberXPLevelChanged__Delegate; //
0x02C0 (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate       __EventPartySizeChanged__Delegate;      // 0x02D8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

```

```

struct FScriptDelegate      __EventPartyLeaderChanged__Delegate;      // 0x02F0
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventSearchStatusChanged__Delegate;    // 0x0308
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventPartyError__Delegate;             // 0x0320
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventPartyJoinGameError__Delegate;     // 0x0338
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventPartyJoinGameSuccess__Delegate;   // 0x0350
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventPartyDestroyed__Delegate;         // 0x0368
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventConfirmJoinGameMessage__Delegate; //
0x0380 (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventLeaderLeftOnlineGame__Delegate;   // 0x0398
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventProcessingStatusChanged__Delegate; //
0x03B0 (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventDisableCrossPlayChanged__Delegate; //
0x03C8 (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventPlayerInMatchChanged__Delegate;   // 0x03E0
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventStartedJoinFriend__Delegate;      // 0x03F8
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventAddPartyMember__Delegate;         // 0x0410
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventRemovePartyMember__Delegate;      // 0x0428
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

```

public:

```
static UClass* StaticClass()
```

```
{
```

```
static UClass* uClassPointer = nullptr;
```

```
if (!uClassPointer)
```

```
{
```

```
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameParty_X");
```

```
}
```

```
return uClassPointer;
```

```
};
```

```
void __OnlineGameParty_X__HandleJoinLobby_0x1();
```

```
void __OnlineGameParty_X__UpdatePartyInfo_0x3(struct FPartyMember PM);
```

```
bool __OnlineGameParty_X__UpdatePartyInfo_0x2(struct FLobbyMember LM);
```

```
bool __OnlineGameParty_X__BroadcastAllLocalPlayers_0x1(struct FPartyMember PM);
```

```
struct FUniqueNetId __OnlineGameParty_X__GetPlayersWithPrimaryMemberID_0x2(struct FPartyMember Member);
```

```
void __bTradeLocked__ChangeNotifyFunc();
```

```
void HandlePlatformPartyIdChanged(class UPartyPlatformSession_X* Session);
```

```
void HandleSetPlatformPartyMessage(class UOnlineMessageComponent_X* Component, class UPartyMessage_SetPlatformParty_X* Message);
```

```
void RequestRejoinPartyInfo();
```

```
class UError* CheckForPlayerInviteError(struct FUniqueNetId PlayerID);
```

```
void SendPartyInvite(struct FUniqueNetId PlayerID);
void HandlePartyConfigChanged();
void HandleMessageForMetrics(class UOnlineMessageComponent_X* Component, class
UObject* Message);
struct FGuid ResolveGuids(struct FGuid& A, struct FGuid& B);
class FString GetMemberName(struct FUniqueNetId InMemberId);
TArray<struct FUniqueNetId> GetPlayersWithPrimaryMemberID(struct FUniqueNetId
InPrimaryID);
TArray<struct FUniqueNetId> GetLocalMemberIDs();
TArray<struct FUniqueNetId> GetOrderedPartyMemberIDs();
TArray<struct FUniqueNetId> GetPartyMemberIDs();
void HandleSessionCreated();
bool IsProcessing();
void SetProcessingStatus(uint8_t Status);
void PrintDebugInfo(class UDebugDrawer* Drawer);
class FString GetPlayerRatingString(struct FUniqueNetId PlayerID, int32_t Playlist);
void OnPartyError(class UError* Error);
void HandlePartyError(class FString Error);
void OnPartyChanged(uint8_t NewProcessingStatus);
bool TryGetValueInt(class FString Key, TArray<struct FLobbyMetaData>& MetaData, int32_t&
Value);
bool PartyHasDisableCrossPlay();
void KickForCrossplayDisabled(struct FUniqueNetId MemberId);
bool HasMultiplePlatforms();
void KickCrossplayDisabledMembers();
void SetCrossPlayTextChatForMember(struct FUniqueNetId PartyMemberID, uint8_t
CrossChatState);
void BroadcastCrossTextChatState();
void HandleCrossPlatformTextChatMessage(class UOnlineMessageComponent_X* Component,
class UObject* Message);
void SetDisableCrossPlayForMember(struct FUniqueNetId PartyMemberID, unsigned long
bDisableCrossPlay);
void HandleDisableCrossPlayMessage(class UOnlineMessageComponent_X* Component, class
UObject* Message);
void BroadcastDisableCrossPlay();
class FString GetAnyLoggedInRestrictionPlayerName();
class FString GetAnyAppOwnerRestrictionPlayerName();
class FString GetMatchmakingRestrictionPlayerName(uint8_t Restriction);
class FString GetMatchmakingRestrictionError();
bool IsPlayerInMainMenu(struct FUniqueNetId MemberId);
bool IsPlayerInMatch(struct FUniqueNetId MemberId);
void SetAvailableForMatchmakingForMember(struct FUniqueNetId PartyMemberID, int32_t
MatchmakeRestrictions);
void HandleMatchmakingAvailability(class UOnlineMessageComponent_X* Component, class
UObject* Message);
int32_t BuildMatchmakingRestrictions();
void BroadcastMatchmakingAvailabilityDelayed();
void BroadcastMatchmakingAvailability();
void BroadcastAllLocalPlayers();
void BroadcastLocalPlayersDelayed();
void BroadcastLocalPlayers();
bool ShouldLeavePartyOnDisconnect();
void HandlePsyNetLoginChanged(class UOnlinePlayerAuthentication_X* Auth);
void HandleLocalPlayerLoginStatusChanged(class UOnlinePlayer_X* Player);
```



```
void HandleLocalPlayerLeave(class ULocalPlayer* Player);
void UpdatePartyMember(class ULocalPlayer_X* Player);
void HandleOnlinePlayerNameChanged(class UOnlinePlayer_X* Player);
void HandleLocalPlayerJoin(class ULocalPlayer* Player);
void ProcessLocalPlayersMessage(class UPartyMessage_LocalPlayers_X* Message);
void HandleLocalPlayers(class UOnlineMessageComponent_X* Component, class UObject*
ObjMessage);
bool AllowSplitScreenPlayer(int32_t ControllerId, class UError*& OutError);
bool HasPartyMembers();
int32_t GetPartySize();
void HandleClientReservationMessage_ConnectionValid(struct FServerReservationData
Reservation, struct FJoinMatchSettings Settings);
void HandleClientReservationMessage(class UIReservationConnection_X* Connection, class
UClientReservationMessage_X* Message);
void HandleSearchStatus(class UOnlineMessageComponent_X* Component, class UObject*
Message);
void BroadcastSearchStatus();
class UPartyMessage_SearchStatus_X* GetSearchStatusMessage();
bool TogglePartyLock();
int32_t GetMaxPartySize();
bool AllowPartySize(int32_t InSize);
bool IsPartyOverFull();
bool AllowNewMember();
void UpdatePartyLock();
void HandleFindGameStateChanged(struct FName NewState);
void ClearLastServerAddress();
void OnNewGameInfoCreated(class AGameInfo_X* Game);
void OnNewSettingsChosen(int32_t PlaylistId);
void OnNewGame();
void SetMatchmakingSearching(unsigned long bState);
void SetSearchState(struct FName SearchState);
int32_t GetLocalMemberIndex();
struct FUniqueNetId GetLocalMemberId();
bool IsMemberLocal(struct FUniqueNetId MemberId);
bool HasRemoteMember();
void OnPartyLeaderChanged();
class UError* GetKickedFromPartyError(uint8_t Reason);
void KickedFromParty(uint8_t Reason);
void HandleKickMessage(class UOnlineMessageComponent_X* Component, class UObject*
Message);
void HandleLobbySettingsUpdateMessage(class UOnlineMessageComponent_X* Component,
class UPartyMessage_LobbySettings_X* LobbySettingsMessage);
void HandleLobbyDestroyed(uint8_t Reason, struct FUniqueLobbyId& LobbyId);
void HandleLobbyReceiveBinaryData(int32_t MemberIndex, struct FActiveLobbyInfo& LobbyInfo,
TArray<uint8_t>& Data);
void RemoveAllPartyMembers();
void RemovePartyMemberByLocalPlayer(class ULocalPlayer* Player);
void RemovePartyMemberByIdWithNotify(struct FUniqueNetId MemberId);
void RemovePartyMemberById(struct FUniqueNetId MemberId);
int32_t AddPartyMember(struct FUniqueNetId PrimaryMemberId, struct FUniqueNetId MemberId,
class FString MemberName, int32_t ControllerId);
void OnEditExistingPartyMember(int32_t PartyMemberIdx);
void OnAddNewPartyMember(int32_t PartyMemberIdx);
void UpdatePartyInfo(struct FActiveLobbyInfo& Lobby);
```

```

void UpdatePartyLeader(struct FActiveLobbyInfo& Lobby);
void HandleLobbySettingsUpdated(struct FActiveLobbyInfo& Lobby);
void OnNewLobby();
void BroadcastPartyInfo();
void HandleJoinLobby_ConnectionValid();
void HandleJoinLobby(unsigned long bWasSuccessful, class FString Error, struct
FActiveLobbyInfo& LobbyInfo, struct FUniqueLobbyId& LobbyUID);
class UError* GetJoinPartyError(struct FActiveLobbyInfo& LobbyInfo);
class UError* CheckForJoinPartyError(struct FActiveLobbyInfo& LobbyInfo);
void BroadcastStateDelayed();
void BroadcastState();
void CheckForJoiningPlayerKick(struct FUniqueNetId PlayerID);
void HandleLobbyMemberStatusUpdate(int32_t MemberIndex, int32_t InstigatorIndex, class
FString Status, struct FActiveLobbyInfo& LobbyInfo);
void CancelJoinGameFromPartyDestroyed(class UOnlineGameParty_X* PartyObject);
void HandlePartyJoinGameComplete(unsigned long bSuccess, class FString FailReason);
void HandleFindServerError(class UError* Error);
void HandleFindServer(struct FServerReservationData Reservation, struct FJoinMatchSettings
Settings);
void HandleConfirmJoinGame_ConnectionValid(struct FPartyJoinMatchSettings InSettings);
void HandleConfirmJoinGame(struct FPartyJoinMatchSettings InSettings);
bool JoinFriend(struct FUniqueNetId PartyMemberID);
bool WantsToFollowTheLeaderOutOfGame();
class UGameSettingPlaylist_X* GetPartyMemberPlaylist(struct FUniqueNetId
PartyMemberNetId);
bool IsInSameMatch(struct FUniqueNetId PartyMemberA, struct FUniqueNetId PartyMemberB);
class FString GetServerName(struct FUniqueNetId PartyMember);
void HandlePartyJoinGame(class UOnlineMessageComponent_X* Component, class
UPartyMessage_JoinGame_X* Message);
bool CanBroadcastMatchmakingMessages();
void BroadcastCancelJoinMessage();
void BroadcastPartyServer();
void HandleServerReserved();
void HandleJoinGameComplete(unsigned long bSuccess, class FString FailReason);
struct FPartyMemberServer GetPartyMemberServer();
void HandlePartyMemberJoinGame(class UOnlineMessageComponent_X* Component, class
UPartyMessage_PartyMemberJoinGame_X* MessageObject);
bool ClearServersForPostGameRankedMatch();
bool SetPartyMemberJoinGame(struct FUniqueNetId& PlayerID, struct FPartyMemberServer&
Server);
void BroadcastPartyMemberServer(struct FPartyMemberServer Server);
void BroadcastLobbySettings(struct FActiveLobbyInfo LobbyInfo);
bool SetLeader(struct FUniqueNetId NewLeader);
bool IsPrimaryPlayerIndex(int32_t MemberIdx);
bool IsPrimaryPlayer(struct FUniqueNetId& PlayerID);
bool IsPlayerInParty(struct FUniqueNetId& PlayerID);
bool KickPlayer(struct FUniqueNetId PlayerID, uint8_t Reason);
bool ShowPlatformInviteUI(uint8_t LocalUserNum);
class FString GetShowPlatformInviteUIError();
bool CanShowPlatformInviteUI();
bool ShowInviteUI(uint8_t LocalUserNum);
bool LeaveParty(class FString Reason);
bool IsInCurrentGame(struct FUniqueNetId MemberId);
bool IsPartyLeader();

```

```

bool IsInPartyId(struct FUniqueLobbyId InPartyId);
bool IsInParty();
void JoinParty(int32_t LocalPlayerNum, struct FUniqueLobbyId& InPartyId);
void CheckPartyTimeout();
void StartPartyTimeout();
void OnPartyCreated(unsigned long bWasSuccessful, class FString Error, struct
FUniqueLobbyId& InPartyId);
void CreatePartyInternal_ConnectionChecked(class UError* ConnectionError, class
UOnlineLobbyInterface* LobbyInterface, int32_t LocalPlayerNum, struct FScriptDelegate
Handler);
void CreatePartyInternal(class UOnlineLobbyInterface* LobbyInterface, int32_t LocalPlayerNum,
struct FScriptDelegate Handler);
void CreatePlatformParty(int32_t LocalPlayerNum, struct FScriptDelegate Handler);
void CreateParty(int32_t LocalPlayerNum, struct FScriptDelegate Handler);
class UOnlineLobbyInterface* GetCreatePartyLobbyInterface();
bool ShouldCreatePsyNetParty();
void SetLobbyInterfacePsyNet();
void SetLobbyInterfacePlatform();
void ShortCircuitPartyInvitedPrompt(struct FUniqueLobbyId& InLobbyId, struct FUniqueNetId&
InviterId);
void HandlePartySizeTracker(class UOnlineGameParty_X* PartyObject);
void HandlePrimaryPlayerIdChanged(class UOnlineGameAccount_X* InAccount, struct
FUniqueNetId PlayerId);
class UPartyMetrics_X* GetMetrics();
void SetLobbyInterface(class UOnlineLobbyInterface* Lobby);
bool IsUsingPsyNetParty();
void InitLobbyInterfaces();
void OnExit();
void OnInit();
void EventRemovePartyMember(class UOnlineGameParty_X* PartyObject, struct FUniqueNetId
InMemberId);
void EventAddPartyMember(class UOnlineGameParty_X* PartyObject, struct FUniqueNetId
InMemberId);
void EventStartedJoinFriend();
void EventPlayerInMatchChanged(class UOnlineGameParty_X* PartyObject, struct FUniqueNetId
InMemberId);
void EventDisableCrossPlayChanged(class UOnlineGameParty_X* PartyObject);
void EventProcessingStatusChanged(class UOnlineGameParty_X* PartyObject);
void EventLeaderLeftOnlineGame(class UOnlineGameParty_X* PartyObject);
void EventConfirmJoinGameMessage(class UOnlineGameParty_X* PartyObject);
void EventPartyDestroyed(class UOnlineGameParty_X* PartyObject);
void EventPartyJoinGameSuccess(class UOnlineGameParty_X* PartyObject);
void EventPartyJoinGameError(class UOnlineGameParty_X* PartyObject, class FString
LocalizedErrorString);
void EventPartyError(class UOnlineGameParty_X* PartyObject, class UError* Error);
void EventSearchStatusChanged(class UOnlineGameParty_X* PartyObject, class
UPartyMessage_SearchStatus_X* StatusMessage);
void EventPartyLeaderChanged(class UOnlineGameParty_X* PartyObject, struct FUniqueNetId
NewLeader);
void EventPartySizeChanged(class UOnlineGameParty_X* PartyObject, int32_t NewSize, int32_t
OldSize);
void EventPartyMemberXPLevelChanged(class UOnlineGameParty_X* PartyObject);
void EventPartyChanged(class UOnlineGameParty_X* PartyObject);
void EventPlayerInvitedSilent(class UOnlineGameParty_X* PartyObject, struct FScriptDelegate

```

```

Callback, struct FUniqueLobbyId& InPartyId, struct FUniqueNetId& InviterId);
void EventPlayerInvited(class UOnlineGameParty_X* PartyObject, struct FScriptDelegate
Callback, struct FUniqueLobbyId& InPartyId, struct FUniqueNetId& FriendId);
void OnPartyInviteAccepted(int32_t LocalPlayerNum, struct FUniqueLobbyId& InPartyId);
void EventPartyCreated(class UOnlineGameParty_X* PartyObject, unsigned long
bWasSuccessful);
};

// Class ProjectX.GRI_X
// 0x03D0 (0x02D8 - 0x06A8)
class AGRI_X : public AGameReplicationInfo
{
public:
int32_t ReplicatedGamePlaylist; // 0x02D8 (0x0004)
[0x0000004100002020] (CPF_Net | CPF_Transient)
int32_t ReplicatedGameMutatorIndex; // 0x02DC (0x0004)
[0x0000004100002020] (CPF_Net | CPF_Transient)
class FString ReplicatedServerRegion; // 0x02E0 (0x0010)
[0x0000004000402020] (CPF_Net | CPF_Transient | CPF_NeedCtorLink)
struct FReplicatedReservationData Reservations[0x8]; // 0x02F0
(0x0300) [0x0000004100402020] (CPF_Net | CPF_Transient | CPF_NeedCtorLink)
class FString GameServerID; // 0x05F0 (0x0010)
[0x0000004000402020] (CPF_Net | CPF_Transient | CPF_NeedCtorLink)
unsigned long bGameStarted : 1; // 0x0600 (0x0004)
[0x0000004100002020] [0x00000001] (CPF_Net | CPF_Transient)
unsigned long bGameEnded : 1; // 0x0600 (0x0004)
[0x0000000000000200] [0x00000002] (CPF_Transient)
class FString MatchGuid; // 0x0608 (0x0010)
[0x0000004100400020] (CPF_Net | CPF_NeedCtorLink)
struct FScriptDelegate __EventSpawned__Delegate; // 0x0618
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventGameDataSelected__Delegate; // 0x0630
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventReservationsUpdated__Delegate; // 0x0648
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventServerNameChanged__Delegate; // 0x0660
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventGameStarted__Delegate; // 0x0678
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventMatchGUIDChanged__Delegate; // 0x0690
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GRI_X");
}

return uClassPointer;
};

```

```

void PrintDebugInfo(class UDebugDrawer* Drawer);
void SetMatchGUID(class FString Id);
void HandleMatchGUIDChanged(class UOnlineGameDedicatedServer_X* OnlineGameDS);
bool IsPlaylistRanked();
class UGameSettingPlaylist_X* GetPlaylist();
bool PlayerIsInCurrentGame(struct FUniqueNetId MemberId);
class FString GetReservationDebugString(struct FReplicatedReservationData Data);
class UOnlineGameDedicatedServer_X* GetOnlineGameDedicated();
class UOnlineGame_Base_X* GetOnlineGameBase();
class UOnlineGame_X* GetOnlineGame();
bool HasSelectedGameData();
void SetGameStarted();
void SetOfflineGameData(struct FName PlaylistName);
void NotifyOnGameDataSelected(struct FScriptDelegate Callback);
void SetGameData(int32_t NewPlaylistID, int32_t MutatorIndex);
void HandleGamePlaylistSet(class UOnlineGameDedicatedServer_X* DedicatedServer);
void OnReservationsUpdated();
void SetReservations(TArray<struct FReplicatedReservationData> InReservations);
struct FReplicatedReservationData ConvertReservation(struct FReservationData Data);
void UpdateReservations();
void eventReplicatedEvent(struct FName VarName);
void eventPostBeginPlay();
void EventMatchGUIDChanged(class AGRI_X* GRI);
void EventGameStarted(class AGRI_X* GRI);
void EventServerNameChanged(class AGRI_X* GRI);
void EventReservationsUpdated(class AGRI_X* GRI);
void EventGameDataSelected(int32_t PlaylistId, int32_t MutatorIndex);
void EventSpawned(class AGRI_X* GRI);
};

```

```

// Class ProjectX.OSSConfig_X
// 0x0008 (0x0078 - 0x0080)
class UOSSConfig_X : public UOnlineConfig_X
{
public:
    unsigned long                bEnablePresence : 1;                // 0x0078 (0x0004)
    [0x000000000000004001] [0x00000001] (CPF_Edit | CPF_Config)
    float                        EosInitTimeoutSeconds;              // 0x007C (0x0004)
    [0x00000000000000001] (CPF_Edit)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.OSSConfig_X");
        }

        return uClassPointer;
    };
};

```

```

};

// Class ProjectX.PartyMessage_X
// 0x0048 (0x0060 - 0x00A8)
class UPartyMessage_X : public UOnlineMessage_X
{
public:
    struct FUniqueNetId                Sender;                // 0x0060 (0x0048)
    [0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartyMessage_X");
        }

        return uClassPointer;
    };

    bool BroadcastOn(class UOnlineLobbyInterface* LobbyInterface, struct FUniqueLobbyId
    LobbyId);
    bool Broadcast();
};

// Class ProjectX.PartyMessage_SearchStatus_X
// 0x0020 (0x00A8 - 0x00C8)
class UPartyMessage_SearchStatus_X : public UPartyMessage_X
{
public:
    struct FName                        SearchState;            // 0x00A8 (0x0008)
    [0x000000000000000000]

    unsigned long                      blsSearching : 1;        // 0x00B0 (0x0004)
    [0x000000000000000000] [0x000000001]

    TArray<int32_t>                    PreferredPlaylists;      // 0x00B8 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartyMessage_SearchStatus_X");
        }

        return uClassPointer;
    };

    class UPartyMessage_SearchStatus_X* SetPreferredPlaylists(TArray<int32_t>& InPlaylists);

```

```

class UPartyMessage_SearchStatus_X* SetIsSearching(unsigned long bValue);
class UPartyMessage_SearchStatus_X* SetSearchState(struct FName InSearchState);
};

// Class ProjectX.PartyMessage_LobbySettings_X
// 0x0004 (0x00A8 - 0x00AC)
class UPartyMessage_LobbySettings_X : public UPartyMessage_X
{
public:
    int32_t BuildID; // 0x00A8 (0x0004)
    [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartyMessage_LobbySettings_X");
        }

        return uClassPointer;
    };

    class UPartyMessage_LobbySettings_X* SetBuildID(int32_t InBuildID);
};

// Class ProjectX.PartyMessage_Kick_X
// 0x0049 (0x00A8 - 0x00F1)
class UPartyMessage_Kick_X : public UPartyMessage_X
{
public:
    struct FUniqueNetId KickPlayer; // 0x00A8 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)
    uint8_t KickReason; // 0x00F0 (0x0001)
    [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartyMessage_Kick_X");
        }

        return uClassPointer;
    };

    class UPartyMessage_Kick_X* SetReason(uint8_t InKickReason);
    class UPartyMessage_Kick_X* SetKicked(struct FUniqueNetId InPlayer);
};

```

```

// Class ProjectX.PartyMessage_LocalPlayers_X
// 0x0058 (0x00A8 - 0x0100)
class UPartyMessage_LocalPlayers_X : public UPartyMessage_X
{
public:
    struct FUniqueNetId          PrimaryMemberId;          // 0x00A8 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)
    TArray<struct FSimplePartyMember> Members;             // 0x00F0
    (0x0010) [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartyMessage_LocalPlayers_X");
        }

        return uClassPointer;
    };

    void AddMember(struct FPartyMember Member);
    class UPartyMessage_LocalPlayers_X* AddPlayer(struct FUniqueNetId PlayerID, class FString
    PlayerName);
    class UPartyMessage_LocalPlayers_X* AddOnlinePlayer(class UOnlinePlayer_X* Player);
    class UPartyMessage_LocalPlayers_X* SetPrimaryMemberId(struct FUniqueNetId
    InPrimaryMemberId);
    class UPartyMessage_LocalPlayers_X* AddOnlinePlayers();
};

// Class ProjectX.PartyMessage_MatchmakingAvailability_X
// 0x0004 (0x00A8 - 0x00AC)
class UPartyMessage_MatchmakingAvailability_X : public UPartyMessage_X
{
public:
    int32_t          MatchmakeRestrictions;          // 0x00A8 (0x0004)
    [0x000000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartyMessage_MatchmakingAvailability_X");
        }

        return uClassPointer;
    };
};

```



```

class UPartyMessage_MatchmakingAvailability_X* SetMatchmakeRestrictions(int32_t
InRestrictions);
};

// Class ProjectX.PartyMessage_JoinGame_X
// 0x0058 (0x00A8 - 0x0100)
class UPartyMessage_JoinGame_X : public UPartyMessage_X
{
public:
    struct FPartyJoinMatchSettings          Settings;                // 0x00A8 (0x0058)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartyMessage_JoinGame_X");
        }

        return uClassPointer;
    };

    class UPartyMessage_JoinGame_X* SetSettings(struct FPartyJoinMatchSettings& InSettings);
};

// Class ProjectX.PartyMessage_PartyMemberJoinGame_X
// 0x0048 (0x00A8 - 0x00F0)
class UPartyMessage_PartyMemberJoinGame_X : public UPartyMessage_X
{
public:
    struct FPartyMemberServer              Server;                // 0x00A8 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartyMessage_PartyMemberJoinGame_X");
        }

        return uClassPointer;
    };

    class UPartyMessage_PartyMemberJoinGame_X* SetPartyMemberServer(struct
FPartyMemberServer& InServer);
};

// Class ProjectX.PartyMessage_DisableCrossPlay_X

```

```

// 0x0004 (0x00A8 - 0x00AC)
class UPartyMessage_DisableCrossPlay_X : public UPartyMessage_X
{
public:
    unsigned long                bDisableCrossPlay : 1;                // 0x00A8 (0x0004)
    [0x0000000000000000] [0x00000001]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartyMessage_DisableCrossPlay_X");
        }

        return uClassPointer;
    };

    class UPartyMessage_DisableCrossPlay_X* SetDisableCrossPlay(unsigned long
    InDisableCrossplay);
};

// Class ProjectX.PartyMessage_CrossPlayTextChat_X
// 0x0001 (0x00A8 - 0x00A9)
class UPartyMessage_CrossPlayTextChat_X : public UPartyMessage_X
{
public:
    uint8_t                      CrossChatState;                      // 0x00A8 (0x0001)
    [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartyMessage_CrossPlayTextChat_X");
        }

        return uClassPointer;
    };

    class UPartyMessage_CrossPlayTextChat_X* SetCrossPlayTextChatState(uint8_t
    InCrossChatState);
};

// Class ProjectX.PartyMessage_SetPlatformParty_X
// 0x0010 (0x00A8 - 0x00B8)
class UPartyMessage_SetPlatformParty_X : public UPartyMessage_X
{
public:

```

```

struct FUniqueLobbyId                                PlatformPartyID;                                // 0x00A8 (0x0010)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PartyMessage_SetPlatformParty_X");
}

return uClassPointer;
};

};

// Class ProjectX.OnlineGameJoinGame_X
// 0x03A0 (0x00B0 - 0x0450)
class UOnlineGameJoinGame_X : public UOnline_X
{
public:
int32_t                                JoinCountdownTime;                                // 0x00B0 (0x0004)
[0x0000000000000001] (CPF_Edit)
class FString                                FailCommand;                                // 0x00B8 (0x0010)
[0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
class FString                                LoadingScreenCommand;                                // 0x00C8 (0x0010)
[0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
class UShakeComponent_X*                                JoinGameShake;                                // 0x00D8
(0x0008) [0x0000000004080009] (CPF_Edit | CPF_ExportObject | CPF_Component |
CPF_EditInline)
class UReservationBeacon_X*                                ReservationBeacon;                                // 0x00E0
(0x0008) [0x0000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class FString                                WaitingForPlayersString;                                // 0x00E8 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                                ReservationNotRespondingString;                                // 0x00F8 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                                ReservationFullString;                                // 0x0108 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                                PartyTeamReservationFullString;                                // 0x0118 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                                NoFriendJoinPrivateMatchString;                                // 0x0128 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                                BeaconTimedOutString;                                // 0x0138 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                                NotAllPlayersJoinedString;                                // 0x0148 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                                CanceledString;                                // 0x0158 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                                SecurityKeyAcquisitionFailed;                                // 0x0168 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                                SecurityKeyVerificationFailed;                                // 0x0178 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)

```

```

class FString                SendingReservationMessage;                // 0x0188 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                JoiningPartyLeadersGame;                // 0x0198 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                InvalidPassword;                        // 0x01A8 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                WrongPlaylistString;                    // 0x01B8 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                WrongRankedMatchString;                // 0x01C8 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                MatchEndedString;                      // 0x01D8 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                CrossplayDisabled;                    // 0x01E8 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                AnotherPlayerCanceled;                // 0x01F8 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
struct FActiveServerData     ActiveServer;                        // 0x0208 (0x00A0)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FJoinMatchSettings    Settings;                            // 0x02A8 (0x0020)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                PendingFailMessage;                    // 0x02C8 (0x0010)
[0x00000000000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<class UPlayer*>       JoinedPlayers;                        // 0x02D8 (0x0010)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FCustomMatchSettings  CustomMatch;                        // 0x02E8
(0x0090) [0x00000004000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventJoinGameComplete__Delegate;    // 0x0378
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventStatusUpdate__Delegate;        // 0x0390
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventCountdownStarted__Delegate;    // 0x03A8
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventCountdownEnded__Delegate;      // 0x03C0
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventActiveServerChanged__Delegate; // 0x03D8
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventServerReserved__Delegate;      // 0x03F0
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventPasswordRequired__Delegate;    // 0x0408
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventJoiningGame__Delegate;         // 0x0420
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate      __EventMaxPlayersChanged__Delegate;    // 0x0438
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameJoinGame_X");
}
}

```

```

return uClassPointer;
};

void HandleCanPlayOnline(class UPrivilegeCheck_X* PrivilegeCheck);
void GoToNextState();
void HandleCancelJoin(class UIReservationConnection_X* Connection, class UObject*
Message);
void SendReservation();
void SendReservationMessage();
class UMatchmakingMetrics_X* GetMetrics();
void HandleConnection(class UReservationBeacon_X* Beacon, class
UIReservationConnection_X* Connection);
void HandleDisconnection(class UReservationBeacon_X* Beacon, class
UIReservationConnection_X* Connection);
void JoinServer(class UReservationsReadyMessage_X* Message);
void TravelToServer();
class UAddReservationMessage_X* CreateReservationMessage();
void SendServerReservedEvent();
void StartConnectToServer();
void CheckAllLocalPlayersHaveJoined(class UPlayer* JoinedPlayer);
void HandleInternetConnectionChanged(unsigned long bConnected);
void OnAllPlayersJoined();
void MigrateToNewServer(struct FServerConnectionInfo ConnectionInfo);
void PrintDebugInfo(class UDebugDrawer* Drawer);
bool IsInTransition();
void SendPing();
void GotoJoinGameState(struct FName NewStateName);
void HandleConnectionResponse(class UIReservationConnection_X* Connection, class UObject*
Message);
void HandleReservationResponse(class UIReservationConnection_X* Connection, class UObject*
Message);
void HandleConnectionFailed(uint8_t MessageType, class FString Title, class FString Message);
void OnJoinGameComplete(unsigned long bSuccess, class FString FailReason);
void SetActiveServerData(struct FActiveServerData& NewValue);
void ClearActiveServerData();
void HandleGameInfoSpawned(class AGameInfo_X* InGameInfo);
bool IsJoiningGame();
void CancelJoin();
bool StartJoinCustomMatch(struct FServerReservationData Reservation, struct
FJoinMatchSettings JoinSettings, struct FCustomMatchSettings& InSettings);
bool StartJoinPrivateMatch(struct FServerReservationData Reservation, struct
FJoinMatchSettings JoinSettings, struct FCustomMatchSettings& InSettings);
bool StartJoin(struct FServerReservationData Reservation, struct FJoinMatchSettings
JoinSettings);
void OnInit();
void EventMaxPlayersChanged(class UOnlineGameJoinGame_X* OnlineGameJoinGame, int32_t
MaxPlayers);
void EventJoiningGame(class UOnlineGameJoinGame_X* OnlineGameJoinGame);
void EventPasswordRequired();
void EventServerReserved();
void EventActiveServerChanged();
void EventCountdownEnded();
void EventCountdownStarted();

```

```

void EventStatusUpdate(class FString NewStatus);
void EventJoinGameComplete(unsigned long bSuccess, class FString FailReason);
};

// Class ProjectX.OnlineGameAccount_X
// 0x0040 (0x00B0 - 0x00F0)
class UOnlineGameAccount_X : public UOnline_X
{
public:
class UOnlinePlayer_X*          PrimaryPlayer;          // 0x00B0 (0x0008)
[0x00000004000002000] (CPF_Transient)
float          UpdateCurrentGameTaskTimer;          // 0x00B8 (0x0004)
[0x00000000000000001] (CPF_Edit)
int32_t          CurrentPlaylistId;          // 0x00BC (0x0004)
[0x00000004000002000] (CPF_Transient)
struct FScriptDelegate          __EventPrimaryPlayerIdChanged__Delegate;    // 0x00C0
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate          __EventCurrentPlaylistSet__Delegate;    // 0x00D8
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameAccount_X");
}

return uClassPointer;
};

void PrintDebugInfo(class UDebugDrawer* Drawer);
void OnMainMenuOpened();
int32_t GetNumLocalPlayers();
void SendUpdatePlayerCurrentGameRPC();
void SendUpdatePlayerCurrentGameRequest();
void UpdateCurrentGame();
void ClearCurrentPlaylist();
void SetCurrentPlaylist(int32_t PlaylistId);
void HandleActiveServerChanged();
void HandlePsyNetConnectionChanged(class UPsyNetConnection_X* C);
void HandlePlayerLoginStatusChanged(class UOnlinePlayer_X* Player);
void OnPrimaryPlayerIdChanged();
void SetPrimaryPlayer(class UOnlinePlayer_X* Player);
void OnInit();
void EventCurrentPlaylistSet(class UOnlineGameAccount_X* InAccount, class
UGameSettingPlaylist_X* NewPlaylist);
void EventPrimaryPlayerIdChanged(class UOnlineGameAccount_X* InAccount, struct
FUniqueNetId PlayerID);
};

// Class ProjectX.PartySequence_InvitedToPlatformParty_X

```

```

// 0x0000 (0x0060 - 0x0060)
class UPartySequence_InvitedToPlatformParty_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PartySequence_InvitedToPlatformParty_X");
}

return uClassPointer;
};

void HandleJoinPlatformLobby(unsigned long bWasSuccessful, class FString Error, struct
FActiveLobbyInfo& LobbyInfo, struct FUniqueLobbyId& LobbyUID);
void JoinPlatformPartyCallback(int32_t LocalPlayerNum, struct FUniqueLobbyId& InLobbyId);
void HandlePlayerInvited(unsigned long bAccepted, struct FUniqueLobbyId& InLobbyId, struct
FUniqueNetId& FriendId);
void Init();
};

// Class ProjectX.Parties_X
// 0x02D0 (0x0060 - 0x0330)
class UParties_X : public UObject
{
public:
struct FActiveLobbyInfo                ActiveLobby;                // 0x0060 (0x0030)
[0x0001008000400000] (CPF_NeedCtorLink)
struct FUniqueLobbyId                LastActiveLobby;                // 0x0090 (0x0010)
[0x0001004000000000]
struct FUniqueNetId                PartyLeaderID;                // 0x00A0 (0x0048)
[0x0001004000400000] (CPF_NeedCtorLink)
class UPsyNet_X*                PsyNet;                // 0x00E8 (0x0008)
[0x0001800000000000]
class UOnlineGameParty_X*                OnlineGameParty;                // 0x00F0
(0x0008) [0x0001800000000000]
class UPartyMessageQueue_X*                MessageQueue;                // 0x00F8
(0x0008) [0x0001000004080009] (CPF_Edit | CPF_ExportObject | CPF_Component |
CPF_EditInline)
class UPsyNetConnection_X*                PsyNetConnection;                // 0x0100
(0x0008) [0x0001004000000000]
class UPsyNetChannel_X*                PsyNetChannel;                // 0x0108 (0x0008)
[0x0001004000000000]
class UPsyNetServiceSubscriptions_X*                Subscriptions;                // 0x0110
(0x0008) [0x0001004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UPartyPlatformSession_X*                PlatformSession;                // 0x0118
(0x0008) [0x0001004000000000]
class UPartySequence_CreateParty_X*                SequenceCreateParty;                // 0x0120
(0x0008) [0x0001004000000000]

```

```

class UPartySequence_JoinParty_X*          SequenceJoinParty;          // 0x0128
(0x0008) [0x0001004000000000]
class UPartySequence_LeaveParty_X*          SequenceLeaveParty;          // 0x0130
(0x0008) [0x0001004000000000]
class UPartySequence_InviteToParty_X*      SequenceInviteToParty;      // 0x0138
(0x0008) [0x0001004000000000]
unsigned long                             bRejoiningParty : 1;          // 0x0140 (0x0004)
[0x0001000000000000] [0x00000001]
float                                     SecondToWaitForPerConReconnect;    // 0x0144 (0x0004)
[0x0001000000000001] (CPF_Edit)
class FString                             PartySetting_PsyNetPartyId;        // 0x0148 (0x0010)
[0x0001000000400001] (CPF_Edit | CPF_NeedCtorLink)
class FString                             PartySetting_JoinKey;            // 0x0158 (0x0010)
[0x0001000000400001] (CPF_Edit | CPF_NeedCtorLink)
struct FScriptDelegate                    __EventShowInviteUI__Delegate;    // 0x0168
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __EventLobbyInviteComplete__Delegate; // 0x0180
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnCreateLobbyComplete__Delegate; // 0x0198
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnLobbyError__Delegate;        // 0x01B0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnLobbySessionCreated__Delegate; // 0x01C8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnFindLobbiesComplete__Delegate; // 0x01E0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnJoinLobbyComplete__Delegate;   // 0x01F8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnLobbySettingsUpdate__Delegate; // 0x0210
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnLobbyMemberSettingsUpdate__Delegate; //
0x0228 (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnLobbyMemberStatusUpdate__Delegate; //
0x0240 (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnLobbyReceiveMessage__Delegate; // 0x0258
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnLobbyReceiveBinaryData__Delegate; // 0x0270
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnLobbyJoinGame__Delegate;      // 0x0288
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnLobbyInvite__Delegate;        // 0x02A0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __EventLobbyInvitePending__Delegate; // 0x02B8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnLobbyDestroyed__Delegate;      // 0x02D0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __EventPartyChatRecieved__Delegate; // 0x02E8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __EventLobbyMessageFailed__Delegate; // 0x0300
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __OnHostStartPlayTogether__Delegate; // 0x0318
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

```

public:



```

static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Parties_X");
}

return uClassPointer;
};

void __Parties_X__CreateLobby_0x1(class UError* Error);
void __Parties_X__HandlePsyNetPartyJoined_0x1(struct FPsyNetPartyMember Member);
void __Parties_X__SetLobbyOwner_0x1(class URPC_PartyChangeOwner_X* RPC);
void HandlePsyNetPartyInviteAccepted(class FString PartyID);
class FString GetPsyNetPartyID();
struct FUniqueNetId FindPartyLeaderID(TArray<struct FPsyNetPartyMember>& Members);
void HandleOwnerChanged(class UPsyNetService_PartyOwnerChanged_X* Notification);
void HandleChatNotification(class UPsyNetService_PartyChat_X* Notifications);
void HandleSystemNotification(class UPsyNetService_PartySystem_X* Notifications);
void HandleUserKicked(class UPsyNetService_PartyUserKicked_X* Notification);
void HandleUserDisconnected(class UPsyNetService_PartyUserDisconnected_X* Notification);
void HandleUserLeft(class UPsyNetService_PartyUserLeft_X* Notification);
void HandleUserJoined(class UPsyNetService_PartyUserJoined_X* Notification);
void HandleMemberStatusUpdate(struct FUniqueNetId MemberUID, class FString Status);
void HandleUserInvitedResponse(struct FUniqueLobbyId LobbyId, struct FUniqueNetId ForUserId,
unsigned long bAccepted);
void HandleUserInvited(class UPsyNetService_PartyUserInvited_X* Notifications);
void SetRejoiningParty(unsigned long bRejoining);
void HandleChannelClosed(class UPsyNetChannel_X* InChannel);
class UPsyNetChannel_X* CreatePsyNetChannel(struct FUniqueLobbyId PsyNetPartyId);
void SetPsyNetSubscriptions(class UPsyNetServiceSubscriptions_X* InSubscriptions);
void SetPsyNetChannel(class UPsyNetChannel_X* InChannel);
void TimeoutPerconReconnect();
void SetPsyNetConnection(class UPsyNetConnection_X* InConnection);
void ClearLobbyData();
void DestroyLobby(uint8_t Reason);
void HandlePerConDisconnected(class UPsyNetConnection_X* Connection);
void HandlePerConConnected(class UPsyNetConnection_X* Connection);
void HandlePartyInfoFail(class URPC_X* RPC);
void HandlePartyInfoSuccess(class URPC_PartyInfo_X* RPC_PartyInfo);
void GetPartyInfo();
class UOnlinePlayer_X* GetPrimaryPlayer();
bool IsInLobby();
static struct FUniqueLobbyId CreatePsyNetPartyID(class FString S);
void ClearHostStartPlayTogetherDelegate(struct FScriptDelegate InDelegate);
void AddHostStartPlayTogetherDelegate(struct FScriptDelegate InDelegate);
void OnHostStartPlayTogether(uint8_t LocalUserNum);
int32_t FindMemberId(struct FLobbyMember& MemberData);
int32_t PlayerIdToMemberId(struct FUniqueLobbyId& LobbyId, struct FUniqueNetId& PlayerId,
int32_t& MemberIndex);
struct FUniqueNetId MemberIdToPlayerId(int32_t MemberId, struct FUniqueLobbyId& LobbyId,
int32_t& MemberIndex);

```

```

bool KickPlayer(uint8_t Reason, struct FUniqueLobbyId& LobbyId, struct FUniqueNetId&
PlayerId);
bool ShowInviteUI(uint8_t LocalUserNum, struct FUniqueLobbyId& LobbyId);
bool InviteToPsyNetLobbyExclusively(struct FUniqueLobbyId& LobbyId, struct FUniqueNetId&
PlayerId);
bool InviteToLobby(struct FUniqueLobbyId& LobbyId, struct FUniqueNetId& PlayerId);
bool CanInviteToLobby(struct FUniqueLobbyId& LobbyId, struct FUniqueNetId& PlayerId);
bool InviteToActiveLobby(struct FUniqueNetId& PlayerId);
bool SetLobbyOwner(struct FUniqueLobbyId& LobbyId, struct FUniqueNetId& NewOwnerId);
bool SetLobbyLock(unsigned long bLocked, struct FUniqueLobbyId& LobbyId);
bool SetLobbyType(uint8_t Type, struct FUniqueLobbyId& LobbyId);
bool SetLobbyServer(class FString ServerIP, struct FUniqueLobbyId& LobbyId, struct
FUniqueNetId& ServerUID);
bool RemoveLobbySetting(class FString Key, struct FUniqueLobbyId& LobbyId);
bool SetLobbySetting(class FString Key, class FString Value, struct FUniqueLobbyId& LobbyId);
bool GetLobbyAdmin(struct FUniqueLobbyId& LobbyId, struct FUniqueNetId& AdminId);
bool SendLobbyBinaryData(struct FUniqueLobbyId& LobbyId, TArray<uint8_t>& Data);
void HandleLobbyMessageFailed(class URPC_PartyChat_X* RPC);
bool SendLobbyMessage(class FString Message, struct FUniqueLobbyId& LobbyId);
bool SetLobbyUserSetting(class FString Key, class FString Value, struct FUniqueLobbyId&
LobbyId);
bool LeaveLobby(struct FUniqueLobbyId& LobbyId);
void RemoveLobbyMember(struct FUniqueNetId PlayerUID);
void AddLobbyMember(struct FUniqueNetId PlayerUID, class FString Username);
void HandleLobbyJoinFailed(class UError* InError);
void HandlePsyNetPartyJoined(struct FUniqueLobbyId PsyNetPartyId, TArray<struct
FPsyNetPartyMember>& Members);
void HandlePsyNetPartyCreated(struct FUniqueLobbyId PsyNetPartyId, TArray<struct
FPsyNetPartyMember>& Members);
bool JoinLobbyWithKey(class FString PsyNetPartyId, class FString JoinKey);
bool JoinLobby(int32_t LocalPlayerNum, struct FUniqueLobbyId& LobbyId);
bool UpdateFoundLobbies(struct FUniqueLobbyId LobbyId);
int32_t FindMemberIndex(struct FUniqueNetId MemberId);
bool FindLobbies(int32_t MaxResults, TArray<struct FLobbyFilter> Filters, TArray<struct
FLobbySortFilter> SortFilters, int32_t MinSlots, uint8_t Distance);
bool CreateLobby(int32_t LocalPlayerNum, int32_t MaxPlayers, uint8_t Type, TArray<struct
FLobbyMetaData> InitialSettings);
void EventLobbyMessageFailed(class UError* Error);
void EventPartyChatRecieved(struct FUniqueNetId PlayerId, class FString Text);
void ClearLobbyDestroyedDelegate(struct FScriptDelegate LobbyDestroyedDelegate);
void AddLobbyDestroyedDelegate(struct FScriptDelegate LobbyDestroyedDelegate);
void OnLobbyDestroyed(uint8_t Reason, struct FUniqueLobbyId& LobbyId);
void RemoveLocalPlayerFromSession(struct FUniqueNetId& PartyMember);
void AddLocalPartyMemberToSession(struct FUniqueNetId& NewPartyMember);
bool GetLobbyMembers(struct FUniqueLobbyId& LobbyId, TArray<struct FLobbyMember>&
Members);
void EventLobbyInvitePending(struct FUniqueLobbyId LobbyId, struct FUniqueNetId FromUserId,
class FString FromUserName);
void ClearLobbyInviteDelegate(struct FScriptDelegate LobbyInviteDelegate);
void AddLobbyInviteDelegate(struct FScriptDelegate LobbyInviteDelegate);
void OnLobbyInvite(unsigned long bAccepted, struct FUniqueLobbyId& LobbyId, struct
FUniqueNetId& FriendId);
void ClearLobbyJoinGameDelegate(struct FScriptDelegate LobbyJoinGameDelegate);
void AddLobbyJoinGameDelegate(struct FScriptDelegate LobbyJoinGameDelegate);

```

```

void OnLobbyJoinGame(class FString ServerIP, struct FActiveLobbyInfo& LobbyInfo, struct
FUniqueNetId& ServerId);
void TriggerLobbyReceiveBinaryDataDelegate(int32_t MemberIndex, TArray<uint8_t> Data);
void ClearLobbyReceiveBinaryDataDelegate(struct FScriptDelegate
LobbyReceiveBinaryDataDelegate);
void AddLobbyReceiveBinaryDataDelegate(struct FScriptDelegate
LobbyReceiveBinaryDataDelegate);
void OnLobbyReceiveBinaryData(int32_t MemberIndex, struct FActiveLobbyInfo& LobbyInfo,
TArray<uint8_t>& Data);
void ClearLobbyReceiveMessageDelegate(struct FScriptDelegate
LobbyReceiveMessageDelegate);
void AddLobbyReceiveMessageDelegate(struct FScriptDelegate
LobbyReceiveMessageDelegate);
void OnLobbyReceiveMessage(int32_t MemberIndex, class FString Type, class FString Message,
struct FActiveLobbyInfo& LobbyInfo);
void ClearLobbyMemberStatusUpdateDelegate(struct FScriptDelegate
LobbyMemberStatusUpdateDelegate);
void AddLobbyMemberStatusUpdateDelegate(struct FScriptDelegate
LobbyMemberStatusUpdateDelegate);
void eventTriggerLobbyMemberStatusUpdateDelegates(int32_t MemberIndex, int32_t
InstigatorIndex, class FString Status);
void OnLobbyMemberStatusUpdate(int32_t MemberIndex, int32_t InstigatorIndex, class FString
Status, struct FActiveLobbyInfo& LobbyInfo);
void ClearLobbyMemberSettingsUpdateDelegate(struct FScriptDelegate
LobbyMemberSettingsUpdateDelegate);
void AddLobbyMemberSettingsUpdateDelegate(struct FScriptDelegate
LobbyMemberSettingsUpdateDelegate);
void OnLobbyMemberSettingsUpdate(int32_t MemberIndex, struct FActiveLobbyInfo&
LobbyInfo);
void ClearLobbySettingsUpdateDelegate(struct FScriptDelegate LobbySettingsUpdateDelegate);
void AddLobbySettingsUpdateDelegate(struct FScriptDelegate LobbySettingsUpdateDelegate);
void OnLobbySettingsUpdate(struct FActiveLobbyInfo& LobbyInfo);
void ClearJoinLobbyCompleteDelegate(struct FScriptDelegate JoinLobbyCompleteDelegate);
void AddJoinLobbyCompleteDelegate(struct FScriptDelegate JoinLobbyCompleteDelegate);
void OnJoinLobbyComplete(unsigned long bWasSuccessful, class FString Error, struct
FActiveLobbyInfo& LobbyInfo, struct FUniqueLobbyId& LobbyId);
void ClearFindLobbiesCompleteDelegate(struct FScriptDelegate FindLobbiesCompleteDelegate);
void AddFindLobbiesCompleteDelegate(struct FScriptDelegate FindLobbiesCompleteDelegate);
void OnFindLobbiesComplete(unsigned long bWasSuccessful, TArray<struct FBasicLobbyInfo>&
LobbyList);
void ClearLobbySessionCreatedDelegate(struct FScriptDelegate LobbySessionCreatedDelegate);
void AddLobbySessionCreatedDelegate(struct FScriptDelegate LobbySessionCreatedDelegate);
void OnLobbySessionCreated();
void ClearLobbyErrorDelegate(struct FScriptDelegate LobbyErrorDelegate);
void AddLobbyErrorDelegate(struct FScriptDelegate LobbyErrorDelegate);
void OnLobbyError(class FString Error);
void ClearCreateLobbyCompleteDelegate(struct FScriptDelegate
CreateLobbyCompleteDelegate);
void AddCreateLobbyCompleteDelegate(struct FScriptDelegate CreateLobbyCompleteDelegate);
void OnCreateLobbyComplete(unsigned long bWasSuccessful, class FString Error, struct
FUniqueLobbyId& LobbyId);
void OnExit();
void Init();
void eventConstruct();

```

```
void EventLobbyInviteComplete(unsigned long bSucceeded, struct FUniqueNetId InviteeID, class UError* InError);
void EventShowInviteUI();
};
```

```
// Class ProjectX.PartySequence_PsyNetPartyUpgrade_X
// 0x0000 (0x0060 - 0x0060)
class UPartySequence_PsyNetPartyUpgrade_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PartySequence_PsyNetPartyUpgrade_X");
}

return uClassPointer;
};
```

```
void HandleJoinPsyNetLobby(unsigned long bWasSuccessful, class FString Error, struct FActiveLobbyInfo& LobbyInfo, struct FUniqueLobbyId& LobbyUID);
void TryUpgrade(struct FActiveLobbyInfo& LobbyInfo, struct FUniqueLobbyId& PlatformLobbyUID);
};
```

```
// Class ProjectX.PartySequence_InvitedToPsyNetParty_X
// 0x0000 (0x0060 - 0x0060)
class UPartySequence_InvitedToPsyNetParty_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PartySequence_InvitedToPsyNetParty_X");
}

return uClassPointer;
};
```

```
void JoinPsyNetPartyCallback(int32_t LocalPlayerNum, struct FUniqueLobbyId& InLobbyId);
class UAsyncTask* HandlePlayerInvited(unsigned long bAccepted, struct FUniqueLobbyId& InLobbyId, struct FUniqueNetId& InviterId);
void HandlePlayerInvitedPrompt(unsigned long bAccepted, struct FUniqueLobbyId& InLobbyId, struct FUniqueNetId& InviterId);
```

```

void HandlePlayerInvitedSilent(struct FUniqueLobbyId& InLobbyId, struct FUniqueNetId&
InviterId);
void Init();
};

// Class ProjectX.PartyPlatformSession_X
// 0x0028 (0x0060 - 0x0088)
class UPartyPlatformSession_X : public UObject
{
public:
    struct FUniqueLobbyId          PartyID;                // 0x0060 (0x0010)
    [0x0001004000000000]
    struct FScriptDelegate         __EventPartyIdChanged__Delegate;    // 0x0070
    (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartyPlatformSession_X");
        }

        return uClassPointer;
    };

    void ClearPlatformParty();
    void LeaveParty();
    void BroadcastPlatformParty();
    void HandleJoinPlatformParty(struct FUniqueLobbyId InPartyId);
    bool IsInParty();
    void EventPartyIdChanged(class UPartyPlatformSession_X* Session);
};

// Class ProjectX.PartyMetrics_X
// 0x000C (0x0080 - 0x008C)
class UPartyMetrics_X : public UMetricsGroup_X
{
public:
    struct FPartyMetricsData        PartyData;            // 0x0080 (0x000C)
    [0x0000000000002000] (CPF_Transient)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartyMetrics_X");
        }
    }

```

```

return uClassPointer;
};

void PartyChannelError(struct FUniqueLobbyId PartyID, struct FUniqueNetId LeaderID, int32_t MissingMessageID);
void PartyMessage(struct FUniqueLobbyId PartyID, struct FUniqueNetId LeaderID, struct FName MessageType);
void PartyChanged(struct FUniqueLobbyId PartyID, struct FUniqueNetId LeaderID, unsigned long bLeader, int32_t PartySize, int32_t LocalPlayers, int32_t RemotePlayers);
struct FPartyMetricsData CreatePartyMetricsData(class UOnlineGameParty_X* Party);
void RecordPartyChanged(class UOnlineGameParty_X* Party);
void PartyError(struct FUniqueLobbyId PartyID, struct FUniqueNetId LeaderID, class FString Error);
void PartyKickRemotePlayer(struct FUniqueLobbyId PartyID, struct FUniqueNetId LeaderID, class FString Reason);
void PartyKickLocalPlayer(struct FUniqueLobbyId PartyID, struct FUniqueNetId LeaderID, class FString Reason);
void PartyLeave(struct FUniqueLobbyId PartyID, struct FUniqueNetId LeaderID, class FString Reason);
void PartyCreationError(struct FUniqueLobbyId PartyID, struct FUniqueNetId LeaderID, class FString Error);
void PartyCreated(struct FUniqueLobbyId PartyID, struct FUniqueNetId LeaderID);
};

// Class ProjectX.__OnlineGameParty_X__CreatePartyInternal_0x1
// 0x0030 (0x0060 - 0x0090)
class U__OnlineGameParty_X__CreatePartyInternal_0x1 : public UObject
{
public:
class UOnlineLobbyInterface* LobbyInterface_Object; // 0x0060
(0x0008) [0x0000000000000000]
class UOnlineLobbyInterface* LobbyInterface_Interface; // 0x0068
(0x0008) [0x0000000000000000]
int32_t LocalPlayerNum; // 0x0070 (0x0004)
[0x0000000000000000]
struct FScriptDelegate Handler; // 0x0078 (0x0018)
[0x00000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGameParty_X__CreatePartyInternal_0x1");
}

return uClassPointer;
};

void __OnlineGameParty_X__CreatePartyInternal_0x1(class UError* ConnectionError);
};

```

```

// Class ProjectX.PartyErrors_X
// 0x0130 (0x0080 - 0x01B0)
class UPartyErrors_X : public UErrorList
{
public:
class UErrorType*                CreatePartyFailed;                // 0x0080 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                CreatePartyFailedTeamFull;        // 0x0088 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                CreatePlayerFailedPartyFull;      // 0x0090 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                InvitePlayerToPartyFailedPartyFull; // 0x0098
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType*                JoinPartyFailedPartyFull;        // 0x00A0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                JoinPartyFailedPartyMatchmaking; // 0x00A8
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType*                JoinPartyFailedPartyInGame;      // 0x00B0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                KickedFromParty;                // 0x00B8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                PartyDestroyedConnectionError;    // 0x00C0
(0x0008) [0x0000000000000002] (CPF_Const)
class UErrorType*                PartyDestroyedSignedOut;          // 0x00C8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                JoinPartyFailed;                // 0x00D0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                JoinPartyFailedNotAllowed;        // 0x00D8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                JoinPartyFailedNotJoinable;      // 0x00E0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                PartyBuildID;                   // 0x00E8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                PartyNotFound;                  // 0x00F0 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                UserNotOwner;                   // 0x00F8 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                PartyIsFull;                    // 0x0100 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                CannotAddToParty;               // 0x0108 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                UserListInvalid;                // 0x0110 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                InvalidOwner;                   // 0x0118 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                ChatDisabled;                   // 0x0120 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                MissingOrExpiredInvite;          // 0x0128 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                MemberNotFound;                 // 0x0130 (0x0008)
[0x0000000000000002] (CPF_Const)
class UErrorType*                PartyIdConflict;                // 0x0138 (0x0008)
[0x0000000000000002] (CPF_Const)

```

```

class UErrorType*                CannotKickSelf;                // 0x0140 (0x0008)
[0x00000000000000002] (CPF_Const)
class UErrorType*                RegionRestrictedTrade;          // 0x0148 (0x0008)
[0x00010000000000002] (CPF_Const)
class UErrorType*                CannotCrossPlayInvite;          // 0x0150 (0x0008)
[0x00000000000000002] (CPF_Const)
class UErrorType*                RestrictedPlatformInvite;        // 0x0158 (0x0008)
[0x00000000000000002] (CPF_Const)
class UErrorType*                PlatformNotSupported;           // 0x0160 (0x0008)
[0x00000000000000002] (CPF_Const)
class UErrorType*                KickedCrossplayDisabled;        // 0x0168 (0x0008)
[0x00000000000000002] (CPF_Const)
class UErrorType*                KickedLeaderPartyUp;            // 0x0170 (0x0008)
[0x00000000000000002] (CPF_Const)
class UErrorType*                VoterDisconnected;              // 0x0178 (0x0008)
[0x00000000000000002] (CPF_Const)
class UErrorType*                VoteEnded;                      // 0x0180 (0x0008)
[0x00000000000000002] (CPF_Const)
class UErrorType*                UsePlatformPartySystem;          // 0x0188 (0x0008)
[0x00000000000000002] (CPF_Const)
class UErrorType*                InvitationRejected;              // 0x0190 (0x0008)
[0x00000000000000002] (CPF_Const)
class UErrorType*                SplitScreenNotAllowedInLan;      // 0x0198 (0x0008)
[0x00000000000000002] (CPF_Const)
class UErrorType*                PartyChatBlockedNotFriendsWithSomeone; // 0x01A0
(0x0008) [0x00000000000000002] (CPF_Const)
class UErrorType*                PartyChatBlockedChatDisabled;    // 0x01A8
(0x0008) [0x00000000000000002] (CPF_Const)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PartyErrors_X");
}

return uClassPointer;
};

};

// Class ProjectX.__OnlineGameParty_X__JoinParty_0x1
// 0x0018 (0x0060 - 0x0078)
class U__OnlineGameParty_X__JoinParty_0x1 : public UObject
{
public:
int32_t                LocalPlayerNum;                // 0x0060 (0x0004)
[0x00000000000000000]
struct FUniqueLobbyId    InPartyId;                    // 0x0068 (0x0010)
[0x00000000000000000]

```



```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.__OnlineGameParty_X__JoinParty_0x1");
}

return uClassPointer;
};

void __OnlineGameParty_X__JoinParty_0x1(class UPrivilegeCheck_X* PrivilegeCheck);
};

// Class ProjectX.OnlineGamePrivileges_X
// 0x0048 (0x00B0 - 0x00F8)
class UOnlineGamePrivileges_X : public UOnline_X
{
public:
TArray<class UPrivilegeCheck_X*> PendingChecks; // 0x00B0
(0x0010) [0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
unsigned long blsCheckingPrivileges : 1; // 0x00C0 (0x0004)
[0x0000004000002000] [0x00000001] (CPF_Transient)
struct FScriptDelegate __EventCheckingPrivilegesChanged__Delegate; //
0x00C8 (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventPrivilegeCheckRestriction__Delegate; // 0x00E0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGamePrivileges_X");
}

return uClassPointer;
};

bool IsCheckingPrivileges();
void RemoveCallback(struct FScriptDelegate Callback);
void HandlePrivilegeCheckFinished(class UPrivilegeCheck_X* PrivilegeCheck);
class UPrivilegeCheck_X* CreatePrivilegeCheck(int32_t ControllerId, struct FScriptDelegate
Callback);
void TryToPlayOnlineAndUseUGC(int32_t ControllerId, struct FScriptDelegate Callback);
class UPrivilegeCheck_X* AddUGCChecks(class UPrivilegeCheck_X* Check);
void TryToUseUGC(int32_t ControllerId, struct FScriptDelegate Callback);
class UPrivilegeCheck_X* AddPlayOnlineChecks(class UPrivilegeCheck_X* Check);
void TryToPlayOnline(int32_t ControllerId, struct FScriptDelegate Callback);
void TryToUsePsyNet(int32_t ControllerId, struct FScriptDelegate Callback);

```

```

void TryToViewLeaderboards(int32_t ControllerId, struct FScriptDelegate Callback);
void TryToBrowseInternet(int32_t ControllerId, struct FScriptDelegate Callback);
void EventPrivilegeCheckRestriction(class UPrivilegeCheck_X* Check);
void EventCheckingPrivilegesChanged(class UOnlineGamePrivileges_X* PrivilegesObject);
};

```

```

// Class ProjectX.OnlineGamePlaylists_X
// 0x0068 (0x00B0 - 0x0118)
class UOnlineGamePlaylists_X : public UOnline_X
{
public:
class UClass* PlaylistClass; // 0x00B0 (0x0008)
[0x0000000000000001] (CPF_Edit)
TArray<class UGameSettingPlaylist_X*> DownloadedPlaylists; // 0x00B8
(0x0010) [0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class URankedConfig_X* RankedConfig; // 0x00C8 (0x0008)
[0x0000800000000001] (CPF_Edit)
class FString CompetitiveCategory; // 0x00D0 (0x0010)
[0x0000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString CasualCategory; // 0x00E0 (0x0010)
[0x0000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString ExtraModeCategory; // 0x00F0 (0x0010)
[0x0000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
struct FScriptDelegate __EventPlaylistsChanged__Delegate; // 0x0100
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGamePlaylists_X");
}

```

```

return uClassPointer;
};

```

```

bool __OnlineGamePlaylists_X__GetPrivateMatch_0x1(class UGameSettingPlaylist_X* P);
bool __OnlineGamePlaylists_X__GetLanMatch_0x1(class UGameSettingPlaylist_X* P);
int32_t __OnlineGamePlaylists_X__GetRankedPlaylistIDs_0x2(class UGameSettingPlaylist_X*
Playlist);
bool __OnlineGamePlaylists_X__GetRankedPlaylistIDs_0x1(class UGameSettingPlaylist_X*
Playlist);
TArray<int32_t> GetRankedPlaylistIDs();
class FString GetLocalizedPlaylistIDCategory(int32_t PlaylistId);
class UGameSettingPlaylist_X* GetLanMatch();
class UGameSettingPlaylist_X* GetPrivateMatch();
TArray<struct FName> GetAccessiblePlaylists(TArray<struct FName>& SelectedPlaylists);
bool IsRankedEnabled();
bool IsRankedPlaylistID(int32_t PlaylistId);
bool IsRankedPlaylistName(struct FName PlaylistName);
bool IsUnrankedPlaylistName(struct FName PlaylistName);

```

```

bool IsStandardPlaylistID(int32_t PlaylistId);
bool IsStandardPlaylistName(struct FName PlaylistName);
bool IsNonStandardPlaylistName(struct FName PlaylistName);
void NamesToIDs(TArray<struct FName>& Names, TArray<int32_t>& Ids);
class FString GetPlaylistFriendlyName(int32_t PlaylistId);
void CopyPlaylistGameTags();
void HandlePlaylistsChanged(class UObjectProvider* Provider);
struct FName IdToName(int32_t PlaylistId);
int32_t NameToId(struct FName PlaylistName);
int32_t GetTimeRemaining(int32_t PlaylistId);
bool IsTimeConstrained(int32_t PlaylistId);
class UGameSettingPlaylist_X* GetPlaylistById(int32_t PlaylistId);
class UGameSettingPlaylist_X* GetPlaylistByName(struct FName PlaylistName);
bool IsPlaylistEnabled(class UGameSettingPlaylist_X* Playlist);
void NotifyWhenChanged(struct FScriptDelegate Callback);
void OnInit();
void EventPlaylistsChanged(class UOnlineGamePlaylists_X* PlaylistsObj);
};

```

```

// Class ProjectX.GameSettingPlaylist_X
// 0x00FC (0x0064 - 0x0160)
class UGameSettingPlaylist_X : public UGameSetting_X
{
public:
    class FString                                Title;                                // 0x0068 (0x0010)
    [0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
    class FString                                Description;                                // 0x0078 (0x0010)
    [0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
    class FString                                BadgeTitle;                                // 0x0088 (0x0010)
    [0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
    int32_t                                        PlayerCount;                                // 0x0098 (0x0004)
    [0x0000004000000001] (CPF_Edit)
    unsigned long                                bStandard : 1;                                // 0x009C (0x0004)
    [0x0000004000000001] [0x00000001] (CPF_Edit)
    unsigned long                                bRanked : 1;                                // 0x009C (0x0004)
    [0x0000004000000001] [0x00000002] (CPF_Edit)
    unsigned long                                bSolo : 1;                                // 0x009C (0x0004)
    [0x0000004000000001] [0x00000004] (CPF_Edit)
    unsigned long                                bExtraMode : 1;                                // 0x009C (0x0004)
    [0x0000004000000001] [0x00000008] (CPF_Edit)
    unsigned long                                bPrivate : 1;                                // 0x009C (0x0004)
    [0x0000004000000001] [0x00000010] (CPF_Edit)
    unsigned long                                bTournament : 1;                                // 0x009C (0x0004)
    [0x0000004000000001] [0x00000020] (CPF_Edit)
    unsigned long                                bApplyQuitPenalty : 1;                                // 0x009C (0x0004)
    [0x0000004000000001] [0x00000040] (CPF_Edit)
    unsigned long                                bAllowForfeit : 1;                                // 0x009C (0x0004)
    [0x0000004000000001] [0x00000080] (CPF_Edit)
    unsigned long                                bDisableRankedReconnect : 1;                                // 0x009C (0x0004)
    [0x0000004000000001] [0x00000100] (CPF_Edit)
    unsigned long                                bIgnoreAssignTeams : 1;                                // 0x009C (0x0004)
    [0x0000004000000001] [0x00000200] (CPF_Edit)
    unsigned long                                bAllowBotFills : 1;                                // 0x009C (0x0004)
    [0x0000004000000001] [0x00000400] (CPF_Edit)

```

```

unsigned long          bKickOnMigrate : 1;                // 0x009C (0x0004)
[0x0000004000000001] [0x00000800] (CPF_Edit)
unsigned long          bCheckRankedMatchReservationID : 1;    // 0x009C
(0x0004) [0x0000004000000001] [0x00001000] (CPF_Edit)
unsigned long          bServerBroadcastCancellations : 1;    // 0x009C
(0x0004) [0x0000004000000001] [0x00002000] (CPF_Edit)
unsigned long          bSkipGameModeVerification : 1;        // 0x009C (0x0004)
[0x0000004000000001] [0x00004000] (CPF_Edit)
unsigned long          bNoBackFill : 1;                    // 0x009C (0x0004)
[0x0000004000000001] [0x00008000] (CPF_Edit)
unsigned long          bIsMicroEventPlaylist : 1;           // 0x009C (0x0004)
[0x0000004000000001] [0x00010000] (CPF_Edit)
unsigned long          bHasVariablePlayerCount : 1;         // 0x009C (0x0004)
[0x0000004000000001] [0x00020000] (CPF_Edit)
unsigned long          bNew : 1;                            // 0x009C (0x0004)
[0x0000004000000001] [0x00040000] (CPF_Edit)
unsigned long          bAllowClubs : 1;                     // 0x009C (0x0004)
[0x0001004000000001] [0x00080000] (CPF_Edit)
unsigned long          bDisableSaveReplays : 1;             // 0x009C (0x0004)
[0x0000004000000001] [0x00100000] (CPF_Edit)
unsigned long          bOpenDetailsOnFirstTimeClicked : 1;   // 0x009C
(0x0004) [0x0000004000000001] [0x00200000] (CPF_Edit)
unsigned long          bAllowStayAsParty : 1;               // 0x009C (0x0004)
[0x0000000000000001] [0x00400000] (CPF_Edit)
class FString          PlaylistImageURL;                    // 0x00A0 (0x0010)
[0x0000004000400001] (CPF_Edit | CPF_NeedCtorLink)
class FString          PlaylistImageTexture;                // 0x00B0 (0x0010)
[0x0000004000400001] (CPF_Edit | CPF_NeedCtorLink)
class FString          PlaylistIconActiveURL;                // 0x00C0 (0x0010)
[0x0000004000400001] (CPF_Edit | CPF_NeedCtorLink)
class FString          PlaylistIconInactiveURL;              // 0x00D0 (0x0010)
[0x0000004000400001] (CPF_Edit | CPF_NeedCtorLink)
class FString          PlaylistNodeThumbnailURL;             // 0x00E0 (0x0010)
[0x0000004000400001] (CPF_Edit | CPF_NeedCtorLink)
class FString          PlaylistNodeDefaultThumbnailPackage;  // 0x00F0
(0x0010) [0x0000004000400001] (CPF_Edit | CPF_NeedCtorLink)
class FString          SecondaryTitleOverride;               // 0x0100 (0x0010)
[0x0000004000400001] (CPF_Edit | CPF_NeedCtorLink)
int32_t               PlaylistId;                            // 0x0110 (0x0004)
[0x0000004000000001] (CPF_Edit)
class UTimeWindow*     PlaylistTimeWindow;                  // 0x0118 (0x0008)
[0x0000000000000001] (CPF_Edit)
TArray<class UPresetMutators_X*> PresetMutators;            // 0x0120
(0x0010) [0x0000008000400001] (CPF_Edit | CPF_NeedCtorLink)
struct FName           MapName;                              // 0x0130 (0x0008)
[0x0000000000000001] (CPF_Edit)
class FString          ServerCommand;                        // 0x0138 (0x0010)
[0x0000000000400001] (CPF_Edit | CPF_NeedCtorLink)
struct FName           MapSetName;                           // 0x0148 (0x0008)
[0x0000000000000001] (CPF_Edit)
TArray<int32_t>         PopulationBuckets;                   // 0x0150 (0x0010)
[0x0000000000400001] (CPF_Edit | CPF_NeedCtorLink)

```

public:

```

static UClass* StaticClass()
{
    static UClass* uClassPointer = nullptr;

    if (!uClassPointer)
    {
        uClassPointer = UObject::FindClass("Class ProjectX.GameSettingPlaylist_X");
    }

    return uClassPointer;
};

bool UseRandomizedNameAndPassword();
bool ShouldAllowRankedReconnect();
bool HasBackfillPolicy();
void GetBackfillAmount(class UOnlineGameDedicatedServerRegistration_X* DS, int32_t&
BackfillTeam1, int32_t& BackfillTeam2);
bool IsLanMatch();
bool IsTournamentMatch();
bool IsPrivateMatch();
bool IsRankedMatch();
bool ShouldUpdateSkills();
static bool IsValidID(int32_t InPlaylistID);
bool IsValid();
class FString GetLocalizedBadgeTitle();
class FString GetLocalizedDescription();
class FString GetLocalizedName();
void Setup(class UPlaylistSettings_X* Settings);
};

// Class ProjectX.__OnlineGameParty_X__HandleConfirmJoinGame_0x1
// 0x0058 (0x0060 - 0x00B8)
class U__OnlineGameParty_X__HandleConfirmJoinGame_0x1 : public UObject
{
public:
    struct FPartyJoinMatchSettings          InSettings;                // 0x0060 (0x0058)
    [0x00000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGameParty_X__HandleConfirmJoinGame_0x1");
        }

        return uClassPointer;
    };

    void __OnlineGameParty_X__HandleConfirmJoinGame_0x1();
};

```

```

// Class ProjectX.__OnlineGameParty_X__HandleConfirmJoinGame_ConnectionValid_0x1
// 0x0020 (0x0060 - 0x0080)
class U__OnlineGameParty_X__HandleConfirmJoinGame_ConnectionValid_0x1 : public UObject
{
public:
    struct FJoinMatchSettings                Settings;                // 0x0060 (0x0020)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__OnlineGameParty_X__HandleConfirmJoinGame_ConnectionValid_0x1");
        }

        return uClassPointer;
    };

    void __OnlineGameParty_X__HandleConfirmJoinGame_ConnectionValid_0x1(struct
    FServerReservationData Reservation);
};

// Class ProjectX.FindServerTask_X
// 0x0028 (0x00D0 - 0x00F8)
class UFindServerTask_X : public UAsyncTask
{
public:
    float                SearchTimeout;                // 0x00D0 (0x0004)
    [0x00000000000000001] (CPF_Edit)
    class URPC_X*                RPC;                // 0x00D8 (0x0008)
    [0x000000000000000000]
    struct FScriptDelegate                __EventResult__Delegate;                // 0x00E0 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.FindServerTask_X");
        }

        return uClassPointer;
    };

    void HandleClientReservationMessage(class UIReservationConnection_X* Connection, class
    UClientReservationMessage_X* Message);

```

```

void HandleSearchTimeout();
void HandleJoinMatchError(class URPC_X* InRPC);
void Cleanup();
void Init(class URPC_X* InRPC);
class UFindServerTask_X* NotifyOnResult(struct FScriptDelegate Callback);
static class UFindServerTask_X* FindUsingRPC(class URPC_X* InRPC);
static class UFindServerTask_X* FindUsingNamePassword(class FString JoinName, class
FString JoinPassword, struct FName ReservationType);
void EventResult(struct FServerReservationData OutResult);
};

// Class ProjectX.PartyJoinedEvent_X
// 0x0000 (0x0060 - 0x0060)
class UPartyJoinedEvent_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PartyJoinedEvent_X");
}

return uClassPointer;
};

};

// Class ProjectX.ClientReservationMessage_X
// 0x0070 (0x0060 - 0x00D0)
class UClientReservationMessage_X : public UBeaconMessage_X
{
public:
struct FServerReservationData          Reservation;          // 0x0060 (0x0070)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ClientReservationMessage_X");
}

return uClassPointer;
};

class FString GetDSRToken();

```

```

class FString GetReservationID();
};

// Class ProjectX.__OnlineGameParty_X__HandleClientReservationMessage_0x1
// 0x0028 (0x0060 - 0x0088)
class U__OnlineGameParty_X__HandleClientReservationMessage_0x1 : public UObject
{
public:
class UClientReservationMessage_X*          Message;                // 0x0060
(0x0008) [0x0000000000000000]
struct FJoinMatchSettings                  Settings;                // 0x0068 (0x0020)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGameParty_X__HandleClientReservationMessage_0x1");
}

return uClassPointer;
};

void __OnlineGameParty_X__HandleClientReservationMessage_0x1();
};

// Class ProjectX.__OnlineGameParty_X__HasMultiplePlatforms_0x1
// 0x0010 (0x0060 - 0x0070)
class U__OnlineGameParty_X__HasMultiplePlatforms_0x1 : public UObject
{
public:
TArray<uint8_t>                          PartyLeaderCrossPlayGroup;    // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGameParty_X__HasMultiplePlatforms_0x1");
}

return uClassPointer;
};

bool __OnlineGameParty_X__HasMultiplePlatforms_0x1(struct FPartyMember P);
};

```



```

// Class ProjectX.OnlineGameSkill_X
// 0x0058 (0x00B0 - 0x0108)
class UOnlineGameSkill_X : public UOnline_X
{
public:
    TArray<class UPlaylistSkillCache_X*>          Playlists;                // 0x00B0 (0x0010)
    [0x0000008000402000] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FPlayerSeasonRewardProgress>    SeasonRewards;          // 0x00C0
    (0x0010) [0x0000008000402000] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FSkillSyncRequest>             SyncRequests;            // 0x00D0
    (0x0010) [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FUniqueNetId>                  SkillsSyncedPlayers;      // 0x00E0
    (0x0010) [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    struct FScriptDelegate                      __EventSkillSynced__Delegate; // 0x00F0
    (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameSkill_X");
        }

        return uClassPointer;
    };

    void __OnlineGameSkill_X__Construct_0x1(class UOnlineGameParty_X* PartyObject);
    void __OnlineGameSkill_X__CreateSyncPlayerSkillRPC_0x2(class URPC_GetPartyMemberSkill_X*
    RPC);
    void __OnlineGameSkill_X__CreateSyncPlayerSkillRPC_0x1(class URPC_GetPlayerSkill_X* RPC);
    void __OnlineGameSkill_X__ClearPartyMembersSkill_0x2(struct FPartyMember Member);
    void __OnlineGameSkill_X__CacheSkills_0x1(struct FPlayerSkillRating Rating);
    void __OnlineGameSkill_X__SyncPartyMembersSkills_0x2(struct FPartyMember Member);
    bool __OnlineGameSkill_X__SyncPartyMembersSkills_0x1(struct FPartyMember Member);
    bool SkillsSynced(struct FUniqueNetId PlayerID);
    int32_t GetHighestSkillTier(struct FUniqueNetId PlayerID);
    struct FPlayerSeasonRewardProgress GetPlayerSeasonRewardProgress(struct FUniqueNetId
    PlayerID);
    class UPlaylistSkillCache_X* GetPlaylistSkillCache(int32_t Playlist);
    void HandleSkillsUpdateFailed(class URPC_X* RPC);
    void SetPlayerSeasonReward(struct FPlayerSeasonRewardProgress Reward);
    void ReplicateSeasonReward(struct FPlayerSeasonRewardProgress Reward);
    void HandleSkillsUpdated(class URPC_UpdateSkills_X* RPC);
    void SubmitMatch(class UMatchData_X* Match, class FString MatchGuid);
    void Clear();
    static float GetConservativeMMR(float Mu, float Sigma);
    float GetPlayerConservativeMMR(struct FUniqueNetId PlayerID, int32_t Playlist);
    static float GetMMR(float Mu, float Sigma);
    float GetPlayerMMR(struct FUniqueNetId PlayerID, int32_t Playlist);
    struct FUpdatedPlayerSkillRating GetUpdatedPlayerRating(struct FUniqueNetId PlayerID, int32_t

```

```

Playlist);
struct FPlayerSkillRating GetPlayerRating(struct FUniqueNetId PlayerID, int32_t Playlist);
void OnSkillSynced(struct FUniqueNetId PlayerID, class UError* Error);
void CacheSeasonReward(struct FPlayerSeasonRewardProgress Reward);
void HandleSyncedPlayerSkill(class URPC_X* RPC, struct FUniqueNetId PlayerID, TArray<struct
FPlayerSkillRating> PlayerSkillRatings, class UError* Error);
void SyncPlayerSkill(struct FUniqueNetId PlayerID, struct FScriptDelegate Callback);
void SyncPartyMembersSkills();
void CacheSkill(struct FUpdatedPlayerSkillRating Rating, int32_t Playlist);
void CacheSkills(TArray<struct FPlayerSkillRating> Ratings);
void PreCacheSkill(struct FPlayerSkillRating Rating, int32_t Playlist);
void HandlePartyDestroyed(class UOnlineGameParty_X* PartyObject);
void HandlePartyLeaderChanged(class UOnlineGameParty_X* PartyObject, struct FUniqueNetId
NewLeader);
void HandlePartyMemberRemoved(class UOnlineGameParty_X* PartyObject, struct
FUniqueNetId InMemberId);
void HandlePartyMemberAdded(class UOnlineGameParty_X* PartyObject, struct FUniqueNetId
InMemberId);
static int32_t GetSkillPlaylistID(int32_t PlaylistId);
void ClearSkill(struct FUniqueNetId PlayerID);
void ClearPartyMembersSkill();
bool AllPlayersAreAroundTheSameRank(int32_t PlaylistId);
class URPC_X* CreateSyncPlayerSkillRPC(struct FUniqueNetId PlayerID);
void eventConstruct();
void EventSkillSynced(class UOnlineGameSkill_X* Skill, struct FUniqueNetId PlayerID, class
UError* Error);
};

// Class ProjectX.__OnlineGameParty_X__GetPlayersWithPrimaryMemberID_0x1
// 0x0048 (0x0060 - 0x00A8)
class U__OnlineGameParty_X__GetPlayersWithPrimaryMemberID_0x1 : public UObject
{
public:
struct FUniqueNetId                InPrimaryID;                                // 0x0060 (0x0048)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGameParty_X__GetPlayersWithPrimaryMemberID_0x1");
}

return uClassPointer;
};

bool __OnlineGameParty_X__GetPlayersWithPrimaryMemberID_0x1(struct FPartyMember
Member);
};

```

```

// Class ProjectX.PartyConfig_X
// 0x0004 (0x0078 - 0x007C)
class UPartyConfig_X : public UOnlineConfig_X
{
public:
    unsigned long                bCompressMessages : 1;                // 0x0078 (0x0004)
    [0x00000000000004000] [0x00000001] (CPF_Config)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartyConfig_X");
        }

        return uClassPointer;
    };
};

// Class ProjectX.__OnlineGameReservations_X__GetMigrationReservationData_0x1
// 0x0018 (0x0060 - 0x0078)
class U__OnlineGameReservations_X__GetMigrationReservationData_0x1 : public UObject
{
public:
    TArray<struct FMigrationReservationData>    CurrentPlayers;                // 0x0060
    (0x0010) [0x00000000000400000] (CPF_NeedCtorLink)
    class UEngine*                            GEngine;                // 0x0070 (0x0008)
    [0x000000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGameReservations_X__GetMigrationReservationData_0x1");
        }

        return uClassPointer;
    };

    void __OnlineGameReservations_X__GetMigrationReservationData_0x1(struct FReservationData
P);
};

// Class ProjectX.__OnlineGameReservations_X__RecordReservation_0x1
// 0x0008 (0x0060 - 0x0068)
class U__OnlineGameReservations_X__RecordReservation_0x1 : public UObject

```

```

{
public:
class UAddReservationMessage_X*           Message;           // 0x0060
(0x0008) [0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGameReservations_X__RecordReservation_0x1");
}

return uClassPointer;
};

void __OnlineGameReservations_X__RecordReservation_0x1(struct FReservationPlayerData P);
};

// Class ProjectX.__OnlineGameReservations_X__SetPlayersWithMigrationData_0x1
// 0x0018 (0x0060 - 0x0078)
class U__OnlineGameReservations_X__SetPlayersWithMigrationData_0x1 : public UObject
{
public:
TArray<struct FReservationData>           CurrentPlayers;           // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class UEngine*                           GEngine;                   // 0x0070 (0x0008)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGameReservations_X__SetPlayersWithMigrationData_0x1");
}

return uClassPointer;
};

void __OnlineGameReservations_X__SetPlayersWithMigrationData_0x3(struct
FMigrationReservationData P);
void __OnlineGameReservations_X__SetPlayersWithMigrationData_0x1(struct
FMigrationReservationData P);
};

// Class ProjectX.__OnlineGameSkill_X__ClearPartyMembersSkill_0x1
// 0x0050 (0x0060 - 0x00B0)

```

```

class U__OnlineGameSkill_X__ClearPartyMembersSkill_0x1 : public UObject
{
public:
    struct FUniqueNetId                PrimaryPlayerId;                // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)
    class UOnlineGameParty_X*          PartyObject;                    // 0x00A8 (0x0008)
    [0x000000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__OnlineGameSkill_X__ClearPartyMembersSkill_0x1");
        }

        return uClassPointer;
    };

    bool __OnlineGameSkill_X__ClearPartyMembersSkill_0x1(struct FPartyMember Member);
};

// Class ProjectX.__OnlineGameSkill_X__ClearSkill_0x1
// 0x0048 (0x0060 - 0x00A8)
class U__OnlineGameSkill_X__ClearSkill_0x1 : public UObject
{
public:
    struct FUniqueNetId                PlayerID;                        // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.__OnlineGameSkill_X__ClearSkill_0x1");
        }

        return uClassPointer;
    };

    void __OnlineGameSkill_X__ClearSkill_0x1(class UPlaylistSkillCache_X* P);
};

// Class ProjectX.PlaylistSkillCache_X
// 0x0030 (0x0060 - 0x0090)
class UPlaylistSkillCache_X : public UObject
{
public:

```

```

TArray<struct FUpdatedPlayerSkillRating>      Players;                                // 0x0060 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class URankedConfig_X*                      RankedConfig;                          // 0x0070 (0x0008)
[0x0001800000000001] (CPF_Edit)
struct FScriptDelegate                      __EventPlaylistSkillChanged__Delegate;    // 0x0078
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PlaylistSkillCache_X");
}

return uClassPointer;
};

```

```

struct FPlayerSkillRating ConvertUpdatedSkillRating(struct FUpdatedPlayerSkillRating
InUpdatedRating);
int32_t GetPlayerIndex(struct FUniqueNetId PlayerID);
struct FPlayerSkillRating GetPlayerRating(struct FUniqueNetId PlayerID);
struct FUpdatedPlayerSkillRating GetUpdatedPlayerRating(struct FUniqueNetId PlayerID);
void ClearSkill(struct FUniqueNetId PlayerID);
void CacheSkill(struct FUpdatedPlayerSkillRating Rating);
void PreCacheSkill(struct FPlayerSkillRating Rating);
bool AllPlayersAreAroundTheSameRank();
void EventPlaylistSkillChanged();
};

```

```

// Class ProjectX.__OnlineGameSkill_X__HandleSyncedPlayerSkill_0x1
// 0x0058 (0x0060 - 0x00B8)
class U__OnlineGameSkill_X__HandleSyncedPlayerSkill_0x1 : public UObject
{
public:
TArray<struct FPlayerSkillRating>           PlayerSkillRatings;                    // 0x0060 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
struct FUniqueNetId                        PlayerID;                               // 0x0070 (0x0048)
[0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGameSkill_X__HandleSyncedPlayerSkill_0x1");
}

return uClassPointer;
};

```

```

};

void __OnlineGameSkill_X__HandleSyncedPlayerSkill_0x1(int32_t RankedPlaylistID);
};

// Class ProjectX.__OnlineGameSkill_X__OnSkillSynced_0x1
// 0x0048 (0x0060 - 0x00A8)
class U__OnlineGameSkill_X__OnSkillSynced_0x1 : public UObject
{
public:
    struct FUniqueNetId                PlayerID;                // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.__OnlineGameSkill_X__OnSkillSynced_0x1");
        }

        return uClassPointer;
    };

    bool __OnlineGameSkill_X__OnSkillSynced_0x1(class ULocalPlayer* P);
};

// Class ProjectX.__OnlineGameWordFilter_X__InternalSanitize_0x1
// 0x0010 (0x0060 - 0x0070)
class U__OnlineGameWordFilter_X__InternalSanitize_0x1 : public UObject
{
public:
    class FString                Id;                // 0x0060 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__OnlineGameWordFilter_X__InternalSanitize_0x1");
        }

        return uClassPointer;
    };

    void __OnlineGameWordFilter_X__InternalSanitize_0x1(struct FWordFilterResult Result);
};

```

```

// Class ProjectX.OnlineGameWordFilter_X
// 0x0020 (0x00B0 - 0x00D0)
class UOnlineGameWordFilter_X : public UOnline_X
{
public:
TArray<struct FWordFilterPair>                Filtered;                // 0x00B0 (0x0010)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)
class UWordFilterConfig_X*                  Config;                  // 0x00C0 (0x0008)
[0x000080000000000001] (CPF_Edit)
class UPsyNetWordFilter_X*                  PsyNetWordFilter;        // 0x00C8 (0x0008)
[0x0000000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameWordFilter_X");
}

return uClassPointer;
};

class FString SanitizePhraseAndNotify(class FString Comment, struct FScriptDelegate Callback);
static class FString SanitizePhrase(class FString Comment);
class UError* CreateError(uint8_t Usage, struct FWordFilterResult& Result);
static bool IsChat(uint8_t Usage);
void HandleCommentSanitized(class FString Id, struct FWordFilterResult& Result);
bool StartWordFilterTask(uint8_t Usage, class FString Comment, struct FScriptDelegate SanitizeDelegate, struct FUniqueNetId PlayerID);
class FString InternalSanitize(class FString Comment, struct FScriptDelegate Callback, struct FScriptDelegate ErrorCallback, uint8_t Usage, struct FUniqueNetId PlayerID);
bool IsPending(class FString Comment);
class FString Sanitize(uint8_t Usage, class FString Comment, struct FScriptDelegate Callback, struct FScriptDelegate ErrorCallback, struct FUniqueNetId PlayerID);
class FString SanitizePlayerName(uint8_t PlayerPlatform, class FString PlayerName, struct FScriptDelegate Callback, struct FScriptDelegate ErrorCallback, struct FUniqueNetId PlayerID);
void OnMainMenuOpened();
};

// Class ProjectX.__OnlinePlayerFriends_X__AcceptEpicFriendInvite_0x1
// 0x0058 (0x0060 - 0x00B8)
class U__OnlinePlayerFriends_X__AcceptEpicFriendInvite_0x1 : public UObject
{
public:
struct FUniqueNetId                        FriendId;                // 0x0060 (0x0048)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString                              InPin;                    // 0x00A8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()

```



```

{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlinePlayerFriends_X__AcceptEpicFriendInvite_0x1");
}

return uClassPointer;
};

void __OnlinePlayerFriends_X__AcceptEpicFriendInvite_0x1(class UEpicFriendsPlugin_X* Plugin);
};

// Class ProjectX.OnlinePlayerFriends_X
// 0x02D0 (0x00B0 - 0x0380)
class UOnlinePlayerFriends_X : public UOnline_X
{
public:
unsigned long                bAllowSamePlatformPsyNetFriends : 1;          // 0x00B0
(0x0004) [0x0000000000000400] [0x00000001] (CPF_Config)
unsigned long                bRepeatFriendsListDownloadsUntilSuccess : 1; // 0x00B0
(0x0004) [0x0000000000000402] [0x00000002] (CPF_Const | CPF_Config)
unsigned long                bFetchingLinkedAccounts : 1;                 // 0x00B0 (0x0004)
[0x0000000000000000] [0x00000004]
class UOnlineFriendMap_X*    CachedFriends;                             // 0x00B8 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UPlatformFriends_X*    PlatformFriends;                          // 0x00C0 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UEpicFriends_X*        EpicFriends;                              // 0x00C8 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
TArray<struct FOnlineFriend>  EpicInvites;                              // 0x00D0 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
class UOnlineFriendMap_X*    BlockedPlayers;                            // 0x00E0 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UOnlineFriendMap_X*    EpicBlockList;                            // 0x00E8 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UOnlineFriendMap_X*    PlatformBlockList;                        // 0x00F0 (0x0008)
[0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class ULinkedAccountMap_X*    PlatformToEpicAccountMap;                 // 0x00F8
(0x0008) [0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class ULinkedAccountMap_X*    EpicToPlatformAccountMap;                 // 0x0100
(0x0008) [0x0000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class URetryDelayer_X*        PlatformFriendsDownloadDelayer;           // 0x0108
(0x0008) [0x0000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class URetryDelayer_X*        PsyNetFriendsDownloadDelayer;             // 0x0110
(0x0008) [0x0000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class URetryDelayer_X*        EpicFriendsDownloadDelayer;               // 0x0118
(0x0008) [0x0000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
TArray<struct FUniqueNetId>   PendingFriendAccepts;                     // 0x0120
(0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
class UBlockStatusReporter_X* BlockStatusReporter;                      // 0x0130
(0x0008) [0x0000000000000000]

```

```

class UOSSConfig_X*                                OSSConfig;                                // 0x0138 (0x0008)
[0x000080000000000001] (CPF_Edit)
class UEpicConfig_X*                               EpicConfig;                               // 0x0140 (0x0008)
[0x000080000000000000]
class UEpicFriendsPlugin_X*                       EpicFriendsPlugin;                       // 0x0148 (0x0008)
[0x000080000000000001] (CPF_Edit)
TArray<struct FEpicSocialTaskData>                 BlockPlayerCallbacks;                    // 0x0150
(0x0010) [0x0000000000040000] (CPF_NeedCtorLink)
TArray<struct FEpicSocialTaskData>                 UnblockPlayerCallbacks;                  // 0x0160
(0x0010) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventSocialCallback__Delegate;        // 0x0170
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventFriendsListChanged__Delegate;    // 0x0188
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventBlockedListChanged__Delegate;    // 0x01A0
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventAcceptEpicFriendInvite__Delegate; // 0x01B8
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventDeclineEpicFriendInvite__Delegate; // 0x01D0
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventChatMessage__Delegate;           // 0x01E8
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventPsyNetStatusUpdate__Delegate;    // 0x0200
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventPlatformStatusUpdate__Delegate;  // 0x0218
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventEpicStatusUpdate__Delegate;      // 0x0230
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventEpicFriendInviteReceived__Delegate; // 0x0248
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventEpicFriendInviteRemoved__Delegate; // 0x0260
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventEpicFriendInviteSucceeded__Delegate; // 0x0278
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventEpicFriendInviteFailed__Delegate; // 0x0290
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventEpicPlayerUnfriended__Delegate;  // 0x02A8
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventQueriedUserByEpicDisplayName__Delegate; //
0x02C0 (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventPlatformFriendsListDownloadCompleted__Delegate; // 0x02D8 (0x0018)
[0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventPsyNetFriendsListDownloadCompleted__Delegate; // 0x02F0 (0x0018)
[0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventEpicFriendsListDownloadCompleted__Delegate; //
0x0308 (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventBlockListDownloaded__Delegate;   // 0x0320
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventBlockStatusReceived__Delegate;   // 0x0338
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                            __EventEpicFriendsPluginAdded__Delegate; // 0x0350
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

```

```

struct FScriptDelegate          __EpicFriendsPlugin__ChangeNotify;      // 0x0368
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlinePlayerFriends_X");
}

return uClassPointer;
};

void __OnlinePlayerFriends_X__OnInit_0x2(class UEpicFriendsPlugin_X* InPlugin);
void __OnlinePlayerFriends_X__OnInit_0x1();
void __OnlinePlayerFriends_X__DownloadBlockedList_0x1(class UEpicFriendsPlugin_X* Plugin);
bool __OnlinePlayerFriends_X__UpdateFriendsFromEpicSub_0x1(struct FOnlineFriend Friend);
void __EpicFriendsPlugin__ChangeNotifyFunc();
void RebuildFriendsCache();
void RemoveEpicFriendLocally(struct FUniqueNetId RemovedID);
void SendChatMessage(class FString InMessage, struct FUniqueNetId Recipient);
void ShowPlayerCard(struct FUniqueNetId FriendId, class FString FriendName);
void SetEpicRichPresence(uint8_t LocalUserNum, class FString PresenceString, class FString
GameDataString);
void SetPlatformRichPresence(uint8_t LocalUserNum, class FString PresenceString, class
FString GameDataString);
void HandleEpicFriendRemoved(unsigned long bWasSuccessful, struct FUniqueNetId
RemovedID);
void RemoveEpicFriend(struct FUniqueNetId FriendId);
bool IsFriend(struct FUniqueNetId FriendId);
bool IsPlatformFriend(struct FUniqueNetId FriendId);
bool IsEpicFriend(struct FUniqueNetId FriendId);
void ConditionalUpdateFriendInMap(class UOnlineFriendMap_X* FriendMap, struct
FOnlineFriend NewFriendData);
void HandleEpicPresenceChanged(struct FUniqueNetId FriendId);
void HandlePlatformPresenceChange(struct FUniqueNetId FriendId);
struct FOnlineStatus ExtractStatusData(struct FOnlineFriend InFriend);
void HandleReadEpicFriendsAbandoned();
void HandleDownloadEpicFriendsTryComplete(unsigned long bSuccess);
void HandleLinkedAccountsToBlockReceived(unsigned long bSuccess, TArray<struct
FLinkedAccountData> RequestedAccounts);
void OnPlayerUnblocked(uint8_t LocalUserNum, struct FUniqueNetId UnblockedPlayerId, class
UErrors* Error);
void OnPlayerBlocked(uint8_t LocalUserNum, struct FUniqueNetId BlockedPlayerId, class UErrors*
Error);
void TriggerSocialCallback(struct FUniqueNetId InPlayerId, class UErrors* Error, TArray<struct
FEpicSocialTaskData>& TaskList);
void OnBlockListUpdated(uint8_t LocalUserNum);
void DownloadEpicFriendsList();
void BeginEpicFriendsDownloadAttempts();
void HandleOnReceivedLinkedAccounts(unsigned long bSuccess, TArray<struct

```

```

FLinkedAccountData> LinkedAccountData);
void GetLinkedFriendData();
void UpdateFriendsFromOnlineSub();
void DelayedUpdateFriendsFromOnlineSub();
void UpdateFriendsFromEpicSub();
void DelayedUpdateFriendsFromEpicSub();
void HandleDeclineEpicFriendInviteComplete(struct FUniqueNetId DeclinedId, class UError*
Error);
void DeclineEpicFriendInvite(struct FUniqueNetId FriendId);
void HandleAcceptEpicFriendInviteComplete(struct FUniqueNetId AcceptedId, class UError*
Error);
void AcceptEpicFriendInvite(struct FUniqueNetId FriendId, class FString InPin);
void HandleEpicFriendInviteCompleted(struct FUniqueNetId InvitedPlayerId, class UError* Error);
bool FilterIncomingFriendInvite(struct FOnlineFriend& Friend);
void UpdateEpicInvites();
void HandleEpicFriendInviteRemoved(uint8_t LocalUserNum, struct FUniqueNetId
PlayerToRemove);
void HandleEpicFriendInviteReceived(uint8_t LocalUserNum, struct FUniqueNetId
RequestingPlayer, class FString RequestingNick, class FString Message);
void InviteEpicFriend(struct FUniqueNetId FriendPlayerId, class FString InPin);
bool RequestLinkedAccounts(TArray<struct FUniqueNetId> AccountIds, struct FScriptDelegate
Callback);
void HandleQueriedUserByEpicDisplayName(unsigned long bWasSuccessful, class FString
QueriedDisplayName, struct FUniqueNetId QueriedPlayerId);
bool QueryUserByEpicDisplayName(class FString DisplayName);
void OnPlatformFriendsDownloadAbandoned();
void HandleDownloadPlatformFriendsTryComplete(unsigned long bSuccess);
void DownloadPlatformFriendsList();
void BeginDownloadPlatformFriendsListAttempts();
class USocialMetrics_X* GetSocialMetrics();
void WaitForBlockListDownload(struct FScriptDelegate Callback);
void WaitForPlayerBlockedStatus(class FString EpicId, struct FScriptDelegate Callback);
void UnblockPlayer(struct FUniqueNetId UnblockID, struct FScriptDelegate Callback);
void BlockPlayer(struct FUniqueNetId BlockedID, class FString BlockedPlayerName, struct
FScriptDelegate Callback);
void DownloadBlockedList();
void HandlePsyNetConnected(class UPsyNetConnection_X* C);
static struct FOnlineFriend ConvertPsyNetResponseToOnlineFriend(struct FPsyNetPersonaData
InData);
void eventConstruct();
void HandleOSSConfigChanged(class UOSSConfig_X* InOSSConfig);
void SubscribeToEpicFriendsPlugin(struct FScriptDelegate Callback);
void OnInit();
void EventEpicFriendsPluginAdded(class UEpicFriendsPlugin_X* Plugin);
void EventBlockStatusReceived(class FString EpicId, unsigned long bBlocked);
void EventBlockListDownloaded();
void EventEpicFriendsListDownloadCompleted(unsigned long bSuccess);
void EventPsyNetFriendsListDownloadCompleted(unsigned long bSuccess);
void EventPlatformFriendsListDownloadCompleted(unsigned long bSuccess);
void EventQueriedUserByEpicDisplayName(unsigned long bWasSuccessful, class FString
QueriedDisplayName, struct FUniqueNetId QueriedPlayerId);
void EventEpicPlayerUnfriended(struct FUniqueNetId RemovedID);
void EventEpicFriendInviteFailed(struct FUniqueNetId InvitedPlayerId, class UError*
InviteFriendError);

```

```

void EventEpicFriendInviteSucceeded(struct FUniqueNetId InvitedPlayerId);
void EventEpicFriendInviteRemoved(struct FUniqueNetId PlayerToRemove);
void EventEpicFriendInviteReceived(struct FUniqueNetId RequestingPlayer, class FString
RequestingNick);
void EventEpicStatusUpdate(struct FOnlineStatus StatusData);
void EventPlatformStatusUpdate(struct FOnlineStatus StatusData);
void EventPsyNetStatusUpdate(struct FOnlineStatus StatusData);
void EventChatMessage(class FString InMessage, struct FUniqueNetId SenderId, unsigned long
bIsLocal);
void EventDeclineEpicFriendInvite(struct FUniqueNetId DeclinedId, class UError* RequestError);
void EventAcceptEpicFriendInvite(struct FUniqueNetId AcceptedId, class UError* RequestError);
void EventBlockedListChanged(class UOnlinePlayerFriends_X* FriendsObject);
void EventFriendsListChanged(class UOnlinePlayerFriends_X* FriendsRef, class UError* Error);
void EventSocialCallback(struct FUniqueNetId TargetPlayerID, class UError* RequestError);
};

```

```

// Class ProjectX.__OnlinePlayerFriends_X__BlockPlayer_0x1
// 0x00A8 (0x0060 - 0x0108)

```

```

class U__OnlinePlayerFriends_X__BlockPlayer_0x1 : public UObject
{
public:
struct FUniqueNetId                BlockedID;                // 0x0060 (0x0048)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FUniqueNetId                EpicAccountId;            // 0x00A8 (0x0048)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate              Callback;                // 0x00F0 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlinePlayerFriends_X__BlockPlayer_0x1");
}

```

```

return uClassPointer;
};

```

```

void __OnlinePlayerFriends_X__BlockPlayer_0x1(class UEpicFriendsPlugin_X* Plugin);
};

```

```

// Class ProjectX.__OnlinePlayerFriends_X__SubscribeToEpicFriendsPlugin_0x1
// 0x0018 (0x0060 - 0x0078)

```

```

class U__OnlinePlayerFriends_X__SubscribeToEpicFriendsPlugin_0x1 : public UObject
{
public:
struct FScriptDelegate              Callback;                // 0x0060 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)

```

```

public:

```

```

static UClass* StaticClass()
{
    static UClass* uClassPointer = nullptr;

    if (!uClassPointer)
    {
        uClassPointer = UObject::FindClass("Class
ProjectX.__OnlinePlayerFriends_X__SubscribeToEpicFriendsPlugin_0x1");
    }

    return uClassPointer;
};

void __OnlinePlayerFriends_X__SubscribeToEpicFriendsPlugin_0x1();
};

// Class ProjectX.RetryDelayer_X
// 0x0050 (0x0070 - 0x00C0)
class URetryDelayer_X : public UComponent
{
public:
    TArray<float>                                RetryDelays;                                // 0x0070 (0x0010)
    [0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)
    int32_t                                     ReattemptsAtMaxBackoff;                                // 0x0080 (0x0004)
    [0x000000000000000001] (CPF_Edit)
    unsigned long                               bRepeatUntilSuccess : 1;                                // 0x0084 (0x0004)
    [0x000000000000000001] [0x00000001] (CPF_Edit)
    int32_t                                     FailedAttempts;                                // 0x0088 (0x0004)
    [0x000000000000000000]
    struct FScriptDelegate                     __RepeatedDelegate__Delegate;                                // 0x0090
    (0x0018) [0x000000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate                     __NotifyAbandonedDelegate__Delegate;            // 0x00A8
    (0x0018) [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RetryDelayer_X");
        }

        return uClassPointer;
    };

    void Abandon();
    void ExecuteNextStep();
    void Cancel();
    void RetryOrAbandon();
    void Start(struct FScriptDelegate ToRepeat, struct FScriptDelegate Abandoned);
    void NotifyAbandonedDelegate();
    void RepeatedDelegate();

```

```

};

// Class ProjectX.__OnlinePlayerFriends_X__UnblockPlayer_0x1
// 0x00A8 (0x0060 - 0x0108)
class U__OnlinePlayerFriends_X__UnblockPlayer_0x1 : public UObject
{
public:
    struct FUniqueNetId                UnblockID;                // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)
    struct FUniqueNetId                EpicAccountId;            // 0x00A8 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate             Callback;                // 0x00F0 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__OnlinePlayerFriends_X__UnblockPlayer_0x1");
        }

        return uClassPointer;
    };

    void __OnlinePlayerFriends_X__UnblockPlayer_0x1(class UEpicFriendsPlugin_X* Plugin);
};

// Class ProjectX.SocialMetrics_X
// 0x0000 (0x0080 - 0x0080)
class USocialMetrics_X : public UMetricsGroup_X
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.SocialMetrics_X");
        }

        return uClassPointer;
    };

    void UnblockedPlayer(struct FUniqueNetId PlatformId, struct FUniqueNetId EpicAccountId);
    void BlockedPlayer(struct FUniqueNetId PlatformId, struct FUniqueNetId EpicAccountId);
};

```

```

// Class ProjectX.__OnlinePlayerFriends_X__InviteEpicFriend_0x1
// 0x0058 (0x0060 - 0x00B8)
class U__OnlinePlayerFriends_X__InviteEpicFriend_0x1 : public UObject
{
public:
    struct FUniqueNetId                FriendPlayerId;                // 0x0060 (0x0048)
    [0x0000000000040000] (CPF_NeedCtorLink)
    class FString                    InPin;                    // 0x00A8 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__OnlinePlayerFriends_X__InviteEpicFriend_0x1");
        }

        return uClassPointer;
    };

    void __OnlinePlayerFriends_X__InviteEpicFriend_0x1(class UEpicFriendsPlugin_X* Plugin);
};

// Class ProjectX.__OnlinePlayerFriends_X__DeclineEpicFriendInvite_0x1
// 0x0048 (0x0060 - 0x00A8)
class U__OnlinePlayerFriends_X__DeclineEpicFriendInvite_0x1 : public UObject
{
public:
    struct FUniqueNetId                FriendId;                // 0x0060 (0x0048)
    [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__OnlinePlayerFriends_X__DeclineEpicFriendInvite_0x1");
        }

        return uClassPointer;
    };

    void __OnlinePlayerFriends_X__DeclineEpicFriendInvite_0x1(class UEpicFriendsPlugin_X* Plugin);
};

// Class ProjectX.__OnlinePlayerFriends_X__TriggerSocialCallback_0x1
// 0x0048 (0x0060 - 0x00A8)

```



```

class U__OnlinePlayerFriends_X__TriggerSocialCallback_0x1 : public UObject
{
public:
    struct FUniqueNetId                InPlayerId;                // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__OnlinePlayerFriends_X__TriggerSocialCallback_0x1");
        }

        return uClassPointer;
    };

    bool __OnlinePlayerFriends_X__TriggerSocialCallback_0x1(struct FEpicSocialTaskData Element);
};

// Class ProjectX.__OnlinePlayerFriends_X__RemoveEpicFriend_0x1
// 0x0048 (0x0060 - 0x00A8)
class U__OnlinePlayerFriends_X__RemoveEpicFriend_0x1 : public UObject
{
public:
    struct FUniqueNetId                FriendId;                // 0x0060 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__OnlinePlayerFriends_X__RemoveEpicFriend_0x1");
        }

        return uClassPointer;
    };

    void __OnlinePlayerFriends_X__RemoveEpicFriend_0x1(class UEpicFriendsPlugin_X* Plugin);
};

// Class ProjectX.RPC_PsyNetSendIndividualChat_X
// 0x0058 (0x00E8 - 0x0140)
class URPC_PsyNetSendIndividualChat_X : public URPC_X
{
public:
    struct FUniqueNetId                PlayerID;                // 0x00E8 (0x0048)

```

```

[0x000000000000400000] (CPF_NeedCtorLink)
class FString                                     Message;                                     // 0x0130 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_PsyNetSendIndividualChat_X");
}

return uClassPointer;
};

class URPC_PsyNetSendIndividualChat_X* SetMessage(class FString InMessage);
class URPC_PsyNetSendIndividualChat_X* SetPlayerID(struct FUniqueNetId InPlayerId);
};

// Class ProjectX.EpicFriends_X
// 0x0000 (0x00C0 - 0x00C0)
class UEpicFriends_X : public UOnlineFriendMap_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EpicFriends_X");
}

return uClassPointer;
};

};

// Class ProjectX.PlatformFriends_X
// 0x0000 (0x00C0 - 0x00C0)
class UPlatformFriends_X : public UOnlineFriendMap_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)

```

```

{
uClassPointer = UObject::FindClass("Class ProjectX.PlatformFriends_X");
}

return uClassPointer;
};

};

// Class ProjectX.__OnlinePlayerStorageQueue_X__GetStorageMaxSizeBytes_0x1
// 0x0008 (0x0060 - 0x0068)
class U__OnlinePlayerStorageQueue_X__GetStorageMaxSizeBytes_0x1 : public UObject
{
public:
    struct FName                                Category;                                // 0x0060 (0x0008)
    [0x0001000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__OnlinePlayerStorageQueue_X__GetStorageMaxSizeBytes_0x1");
        }

        return uClassPointer;
    };

    bool __OnlinePlayerStorageQueue_X__GetStorageMaxSizeBytes_0x1(struct FStorageMaxSize C);
};

// Class ProjectX.EncodeObject_X
// 0x0018 (0x0060 - 0x0078)
class UEncodeObject_X : public UObject
{
public:
    uint8_t                                    Encoding;                                    // 0x0060 (0x0001)
    [0x0000004000000000]
    int32_t                                    Checksum;                                    // 0x0064 (0x0004)
    [0x0000008000000000]
    class FString                                Encoded;                                // 0x0068 (0x0010)
    [0x0000008000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.EncodeObject_X");

```

```

}

return uClassPointer;
};

};

// Class ProjectX.RPC_PlayerStorageSet_X
// 0x0060 (0x00E8 - 0x0148)
class URPC_PlayerStorageSet_X : public URPC_X
{
public:
    struct FUniqueNetId                PlayerID;                // 0x00E8 (0x0048)
    [0x0001000000400000] (CPF_NeedCtorLink)
    TArray<struct FSetPlayerStorageRequestItem>    Items;                // 0x0130
    (0x0010) [0x0001000000400000] (CPF_NeedCtorLink)
    class USetPlayerStorageResult_X*        Result;                // 0x0140 (0x0008)
    [0x0001000000002000] (CPF_Transient)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_PlayerStorageSet_X");
        }

        return uClassPointer;
    };

    class UObject* eventGetResponseObject();
};

// Class ProjectX.__OnlinePlayerStorageQueue_X__MapResultItem_0x1
// 0x0010 (0x0060 - 0x0070)
class U__OnlinePlayerStorageQueue_X__MapResultItem_0x1 : public UObject
{
public:
    struct FSetPlayerStorageResultItem        Item;                // 0x0060 (0x0010)
    [0x0001000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__OnlinePlayerStorageQueue_X__MapResultItem_0x1");
        }
    }

```

```

return uClassPointer;
};

bool __OnlinePlayerStorageQueue_X__MapResultItem_0x1(struct FPendingStorage P);
};

// Class ProjectX.__OnlinePlayerStorageSync_X__SyncObjects_0x1
// 0x0008 (0x0060 - 0x0068)
class U__OnlinePlayerStorageSync_X__SyncObjects_0x1 : public UObject
{
public:
class UAsyncTask*                ResponseTask;                // 0x0060 (0x0008)
[0x0001000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__OnlinePlayerStorageSync_X__SyncObjects_0x1");
}

return uClassPointer;
};

void __OnlinePlayerStorageSync_X__SyncObjects_0x1(class URPC_X* RPC);
};

// Class ProjectX.OnlinePlayerStorageSync_X
// 0x0018 (0x0060 - 0x0078)
class UOnlinePlayerStorageSync_X : public UObject
{
public:
struct FScriptDelegate           __EventSyncSuccess__Delegate;           // 0x0060
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlinePlayerStorageSync_X");
}

return uClassPointer;
};

void __OnlinePlayerStorageSync_X__HandleSyncSuccess_0x1(struct
FOnlinePlayerStorageSyncResult Result);

```

```

class UObject* InstanceSyncData(class UObject* LocalObject, class UDecodeObject_X*
DecodeObj);
void HandleSyncSuccess(class URPC_PlayerStorageGet_X* RPC, class UAsyncTask*
ResponseTask);
int32_t CrcObject(uint8_t Encoding, class UObject* Data);
struct FGetPlayerStorageRequestItem MapRequestItem(struct
FOnlinePlayerStorageSyncRequest Request);
class UAsyncTask* SyncObjects(TArray<struct FOnlinePlayerStorageSyncRequest>& Requests);
void EventSyncSuccess(struct FOnlinePlayerStorageSyncResult& Result);
};

```

```

// Class ProjectX.__Parties_X__HandleUserInvited_0x1
// 0x0008 (0x0060 - 0x0068)
class U__Parties_X__HandleUserInvited_0x1 : public UObject
{
public:
class UPsyNetService_PartyUserInvited_X*      Notifications;                // 0x0060
(0x0008) [0x0001000000000000]

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.__Parties_X__HandleUserInvited_0x1");
}

```

```

return uClassPointer;
};

```

```

void __Parties_X__HandleUserInvited_0x1(class FString EpicId, unsigned long bBlocked);
};

```

```

// Class ProjectX.PsyNetService_Party_X
// 0x00D8 (0x0090 - 0x0168)
class UPsyNetService_Party_X : public UPsyNetClientService_X
{
public:
class FString          PartyID;                // 0x0090 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString          NotificationType;        // 0x00A0 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString          Content;                // 0x00B0 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
struct FUniqueNetId    FromUserId;             // 0x00C0 (0x0048)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString          FromUserName;           // 0x0108 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
int32_t                CreatedAt;              // 0x0118 (0x0004)
[0x0000000000000000]
struct FUniqueNetId    ForUserId;              // 0x0120 (0x0048)
[0x0000000000040000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_Party_X");
}

return uClassPointer;
};

};

// Class ProjectX.PsyNetService_PartyUserInvited_X
// 0x0048 (0x0168 - 0x01B0)
class UPsyNetService_PartyUserInvited_X : public UPsyNetService_Party_X
{
public:
struct FUniqueNetId                               FromEpicUserID;                                // 0x0168 (0x0048)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_PartyUserInvited_X");
}

return uClassPointer;
};

};

// Class ProjectX.__PartyMessageQueue_X__SendMessage_0x1
// 0x0008 (0x0060 - 0x0068)
class U__PartyMessageQueue_X__SendMessage_0x1 : public UObject
{
public:
class UAsyncTask*                                Task;                                // 0x0060 (0x0008)
[0x000000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{

```

```

uClassPointer = UObject::FindClass("Class
ProjectX.__PartyMessageQueue_X__SendMessage_0x1");
}

return uClassPointer;
};

void __PartyMessageQueue_X__SendMessage_0x1W(class UError* _);
};

// Class ProjectX.PartyMessageQueue_X
// 0x0040 (0x0070 - 0x00B0)
class UPartyMessageQueue_X : public UComponent
{
public:
TArray<struct FPendingMessage> Pending; // 0x0070 (0x0010)
[0x0000000000040200] (CPF_Transient | CPF_NeedCtorLink)
TArray<class UAsyncTask*> CurrentBatch; // 0x0080 (0x0010)
[0x0000000000040200] (CPF_Transient | CPF_NeedCtorLink)
unsigned long bPaused : 1; // 0x0090 (0x0004)
[0x0000000400000000] [0x00000001]
float BatchDelayTime; // 0x0094 (0x0004)
[0x0000000000000001] (CPF_Edit)
struct FScriptDelegate __SendMessageServiceDelegate__Delegate; // 0x0098
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PartyMessageQueue_X");
}

return uClassPointer;
};

void __PartyMessageQueue_X__CancelAll_0x1(class UAsyncTask* Task);
void HandleTaskComplete(class UAsyncTask* Task);
class UAsyncTask* SendMessageW(struct FUniqueLobbyId LobbyId, class FString Message);
void SendBatch();
void SendBatchTimer();
void QueueBatch();
void CancelAll();
void QueueMessage(struct FUniqueLobbyId LobbyId, class FString Message);
void SetPaused(unsigned long bPause);
class UAsyncTask* SendMessageServiceDelegateW(struct FUniqueLobbyId LobbyId, class
FString Message);
};

// Class ProjectX.__PartySequence_CreateParty_X__CreateParty_0x1
// 0x0019 (0x0060 - 0x0079)

```



```

class U__PartySequence_CreateParty_X__CreateParty_0x1 : public UObject
{
public:
TArray<struct FLobbyMetaData> InitialSettings; // 0x0060 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
int32_t LocalPlayerNum; // 0x0070 (0x0004)
[0x0001000000000000]
int32_t MaxPlayers; // 0x0074 (0x0004)
[0x0001000000000000]
uint8_t Type; // 0x0078 (0x0001)
[0x0001000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__PartySequence_CreateParty_X__CreateParty_0x1");
}

return uClassPointer;
};

void __PartySequence_CreateParty_X__CreateParty_0x1(class URPC_PartyCreate_X* RPC);
};

// Class ProjectX.RPC_PartyCreate_X
// 0x0088 (0x00E8 - 0x0170)
class URPC_PartyCreate_X : public URPC_X
{
public:
unsigned long bForcePartyonix : 1; // 0x00E8 (0x0004)
[0x0000000000000000] [0x00000001]
struct FPsyNetPartyInfo Info; // 0x00F0 (0x0070)
[0x0001000000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<struct FPsyNetPartyMember> Members; // 0x0160
(0x0010) [0x0001000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_PartyCreate_X");
}

return uClassPointer;
};

```

```

class URPC_PartyCreate_X* SetRequirePsyNetParty(unsigned long bValue);
};

// Class ProjectX.PartySequence_CreateParty_X
// 0x0020 (0x0060 - 0x0080)
class UPartySequence_CreateParty_X : public UObject
{
public:
class URPC_PartyCreate_X*          PendingRPC;          // 0x0060 (0x0008)
[0x0001000000000000]
struct FScriptDelegate             __EventPartyCreated__Delegate;      // 0x0068
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PartySequence_CreateParty_X");
}

return uClassPointer;
};

void __PartySequence_CreateParty_X__CreateParty_0x2(class URPC_X* _);
void HandleCreatePartyComplete();
void HandlePlatformPartyCreated(unsigned long bWasSuccessful, class FString Error, struct
FUniqueLobbyId& PlatformPartyId);
class UAsyncTask* CreateParty(int32_t LocalPlayerNum, int32_t MaxPlayers, uint8_t Type,
TArray<struct FLobbyMetaData> InitialSettings);
void EventPartyCreated(struct FUniqueLobbyId PsyNetPartyId, TArray<struct
FPsyNetPartyMember>& Members);
};

// Class ProjectX.__PartySequence_InvitedToPlatformParty_X__HandlePlayerInvited_0x1
// 0x0058 (0x0060 - 0x00B8)
class U__PartySequence_InvitedToPlatformParty_X__HandlePlayerInvited_0x1 : public UObject
{
public:
struct FUniqueLobbyId              InLobbyId;              // 0x0060 (0x0010)
[0x0000000000000000]
struct FUniqueNetId                FriendId;                // 0x0070 (0x0048)
[0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class

```

```

ProjectX.__PartySequence_InvitedToPlatformParty_X__HandlePlayerInvited_0x1");
}

return uClassPointer;
};

void __PartySequence_InvitedToPlatformParty_X__HandlePlayerInvited_0x1();
};

// Class ProjectX.__PartySequence_InvitedToPsyNetParty_X__HandlePlayerInvitedPrompt_0x1
// 0x0058 (0x0060 - 0x00B8)
class U__PartySequence_InvitedToPsyNetParty_X__HandlePlayerInvitedPrompt_0x1 : public
UObject
{
public:
struct FUniqueLobbyId                InLobbyId;                // 0x0060 (0x0010)
[0x0001000000000000]
struct FUniqueNetId                InviterId;                // 0x0070 (0x0048)
[0x0001000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__PartySequence_InvitedToPsyNetParty_X__HandlePlayerInvitedPrompt_0x1");
}

return uClassPointer;
};

void __PartySequence_InvitedToPsyNetParty_X__HandlePlayerInvitedPrompt_0x1();
};

// Class ProjectX.__PartySequence_InvitedToPsyNetParty_X__HandlePlayerInvitedSilent_0x1
// 0x0058 (0x0060 - 0x00B8)
class U__PartySequence_InvitedToPsyNetParty_X__HandlePlayerInvitedSilent_0x1 : public
UObject
{
public:
struct FUniqueLobbyId                InLobbyId;                // 0x0060 (0x0010)
[0x0001000000000000]
struct FUniqueNetId                InviterId;                // 0x0070 (0x0048)
[0x0001000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)

```

```

{
uClassPointer = UObject::FindClass("Class
ProjectX.__PartySequence_InvitedToPsyNetParty_X__HandlePlayerInvitedSilent_0x1");
}

return uClassPointer;
};

void __PartySequence_InvitedToPsyNetParty_X__HandlePlayerInvitedSilent_0x1();
};

// Class ProjectX.__PartySequence_JoinParty_X__HandleGetPlatformPartyMessage_0x1
// 0x0008 (0x0060 - 0x0068)
class U__PartySequence_JoinParty_X__HandleGetPlatformPartyMessage_0x1 : public UObject
{
public:
class UPartyMessage_GetPlatformParty_X*      Message;                                // 0x0060
(0x0008) [0x0001000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__PartySequence_JoinParty_X__HandleGetPlatformPartyMessage_0x1");
}

return uClassPointer;
};

bool __PartySequence_JoinParty_X__HandleGetPlatformPartyMessage_0x1(struct
FPartyMember M);
};

// Class ProjectX.PartyMessage_GetPlatformParty_X
// 0x0000 (0x00A8 - 0x00A8)
class UPartyMessage_GetPlatformParty_X : public UPartyMessage_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PartyMessage_GetPlatformParty_X");
}

return uClassPointer;
};

```

```

};

};

// Class ProjectX.PartySequence_JoinParty_X
// 0x0018 (0x0060 - 0x0078)
class UPartySequence_JoinParty_X : public UObject
{
public:
    struct FScriptDelegate          __EventJoinedParty__Delegate;          // 0x0060
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PartySequence_JoinParty_X");
        }

        return uClassPointer;
    };

    void HandlePlatformPartyCreated(unsigned long bWasSuccessful, class FString Error, struct FUniqueLobbyId& PlatformPartyId);
    void HandleJoinLobby(unsigned long bWasSuccessful, class FString Error, struct FActiveLobbyInfo& LobbyInfo, struct FUniqueLobbyId& LobbyUID);
    void HandleGetPlatformPartyResponseMessage(class UOnlineMessageComponent_X* Component, class UPartyMessage_GetPlatformPartyResponse_X* Message);
    void HandleGetPlatformPartyMessage(class UOnlineMessageComponent_X* Component, class UPartyMessage_GetPlatformParty_X* Message);
    void HandlePsyNetLobbyJoinSucceeded(class URPC_PartyJoin_X* RpcPartyJoin);
    class UAsyncTask* JoinLobbyWithKey(class FString PsyNetPartyId, class FString JoinKey);
    class UAsyncTask* JoinLobby(struct FUniqueLobbyId& LobbyId);
    void Init();
    void EventJoinedParty(struct FUniqueLobbyId PsyNetPartyId, TArray<struct FPsyNetPartyMember>& Members);
};

// Class ProjectX.__PRI_X__RegisterPlayerWithSession_0x1
// 0x0008 (0x0060 - 0x0068)
class U__PRI_X__RegisterPlayerWithSession_0x1 : public UObject
{
public:
    class UOnlineSubsystem*          OnlineSub;          // 0x0060 (0x0008)
    [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.__PRI_X__RegisterPlayerWithSession_0x1");
}

return uClassPointer;
};

void __PRI_X__RegisterPlayerWithSession_0x1(struct FUniqueNetId _);
};

// Class ProjectX.PRI_X
// 0x0048 (0x0410 - 0x0458)
class APRI_X : public APlayerReplicationInfo
{
public:
    struct FScriptDelegate          __EventPlayerNameChanged__Delegate;          // 0x0410
    (0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate          __EventUniqueldChanged__Delegate;          // 0x0428
    (0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
    struct FScriptDelegate          __EventDestroyed__Delegate;          // 0x0440
    (0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PRI_X");
        }

        return uClassPointer;
    };

    void eventDestroyed();
    void OnUniqueldChanged();
    void SetUniqueld(struct FUniqueNetId PlayerUniqueld);
    void UnregisterPlayerFromSession();
    void RegisterPlayerWithSession();
    void AddPlayerHistoryKey(class UOnlineRecentPlayersList* PlayersList);
    void eventOnOwnerChanged();
    void eventSetPlayerName(class FString S);
    void eventReplicatedEvent(struct FName VarName);
    void EventDestroyed(class APRI_X* PRI);
    void EventUniqueldChanged(class APRI_X* PRI);
    void EventPlayerNameChanged(class APRI_X* PRI);
};

// Class ProjectX.__PsyNetBeacon_X__SendMessageToClients_0x1
// 0x0008 (0x0060 - 0x0068)
class U__PsyNetBeacon_X__SendMessageToClients_0x1 : public UObject
{

```

```

public:
class UPsyNetBeaconConnection_X*          Connection;          // 0x0060
(0x0008) [0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__PsyNetBeacon_X__SendMessageToClients_0x1");
}

return uClassPointer;
};

bool __PsyNetBeacon_X__SendMessageToClients_0x1W(class URPC_RelayToClient_X*
OtherRPC);
};

// Class ProjectX.RPC_RelayToClient_X
// 0x0048 (0x00E8 - 0x0130)
class URPC_RelayToClient_X : public URPC_X
{
public:
TArray<class FString>          PlayerIds;          // 0x00E8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString          ReservationID;          // 0x00F8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString          MessageType;          // 0x0108 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString          MessagePayload;          // 0x0118 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
unsigned long          AllowPartialSuccess : 1;          // 0x0128 (0x0004)
[0x000000000000000000] [0x000000001]
unsigned long          QueueOffline : 1;          // 0x0128 (0x0004)
[0x000000000000000000] [0x000000002]
int32_t          OfflineTTLSeconds;          // 0x012C (0x0004)
[0x000000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_RelayToClient_X");
}

return uClassPointer;
};

```

```

};

// Class ProjectX.__PsyNetBeacon_X__SendMessageToServer_0x1
// 0x0010 (0x0060 - 0x0070)
class U__PsyNetBeacon_X__SendMessageToServer_0x1 : public UObject
{
public:
class URPC_RelayToServer_X*          RPC;          // 0x0060 (0x0008)
[0x0000000000000000]
class UPsyNetBeaconConnection_X*    Connection;    // 0x0068
(0x0008) [0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__PsyNetBeacon_X__SendMessageToServer_0x1");
}

return uClassPointer;
};

void __PsyNetBeacon_X__SendMessageToServer_0x1W(class URPC_X* _);
};

// Class ProjectX.RPC_RelayToServer_X
// 0x0040 (0x00E8 - 0x0128)
class URPC_RelayToServer_X : public URPC_X
{
public:
class FString          DSConnectToken;          // 0x00E8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString          ReservationID;            // 0x00F8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString          MessageType;             // 0x0108 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString          MessagePayload;          // 0x0118 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_RelayToServer_X");
}
}

```



```

return uClassPointer;
};

};

// Class ProjectX.__PsyNetConnection_X__ProcessServiceCall_0x1
// 0x0008 (0x0060 - 0x0068)
class U__PsyNetConnection_X__ProcessServiceCall_0x1 : public UObject
{
public:
class UPsyNetMessage_X*                Response;                // 0x0060 (0x0008)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__PsyNetConnection_X__ProcessServiceCall_0x1");
}

return uClassPointer;
};

void __PsyNetConnection_X__ProcessServiceCall_0x1(class UError* _);
};

// Class ProjectX.__PsyNetMessageHttp_X__SendMessage_0x1
// 0x0018 (0x0060 - 0x0078)
class U__PsyNetMessageHttp_X__SendMessage_0x1 : public UObject
{
public:
class FString                PsyRequestID;                // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class UAsyncTask*            Task;                // 0x0070 (0x0008)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__PsyNetMessageHttp_X__SendMessage_0x1");
}

return uClassPointer;
};

```

```

void __PsyNetMessageHttp_X__SendMessage_0x1W(class UWebRequest_X* Request);
};

// Class ProjectX.__PsyNetRequestQueue_X__SendRequest_0x1
// 0x0008 (0x0060 - 0x0068)
class U__PsyNetRequestQueue_X__SendRequest_0x1 : public UObject
{
public:
    struct FName                                RequestID;                                // 0x0060 (0x0008)
    [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__PsyNetRequestQueue_X__SendRequest_0x1");
        }

        return uClassPointer;
    };

    void __PsyNetRequestQueue_X__SendRequest_0x1();
};

// Class ProjectX.PsyNetRequestQueue_X
// 0x0030 (0x0060 - 0x0090)
class UPsyNetRequestQueue_X : public UObject
{
public:
    float                                RequestTimeout;                                // 0x0060 (0x0004)
    [0x0000000000000001] (CPF_Edit)
    TArray<struct FPsyNetRequest>          Requests;                                // 0x0068 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)
    struct FScriptDelegate                __SendMessageDelegate__Delegate;          // 0x0078
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PsyNetRequestQueue_X");
        }

        return uClassPointer;
    };

    void FailAllPending(class UError* Error);
};

```

```

void TimeoutRequests();
bool ProcessResponseMessage(class UPsyNetMessage_X* Message);
void RemoveRequest(struct FName RequestID);
void SetRequestComplete(struct FName RequestID, class UPsyNetMessage_X* Response, class UError* Error);
class UAsyncResult__PsyNetMessage_X* eventSendRequest(class FString Service, class UPsyNetMessage_X* Message);
class UAsyncTask* SendMessageDelegateW(class UPsyNetMessage_X* Message);
};

// Class ProjectX.__PsyNetServiceProvider_X__CreateChannel_0x1
// 0x0010 (0x0060 - 0x0070)
class U__PsyNetServiceProvider_X__CreateChannel_0x1 : public UObject
{
public:
    class FString                ChannelName;                // 0x0060 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__PsyNetServiceProvider_X__CreateChannel_0x1");
        }

        return uClassPointer;
    };
};

bool __PsyNetServiceProvider_X__CreateChannel_0x1(class UPsyNetChannel_X* C);
};

// Class ProjectX.PsyNetChannel_X
// 0x0050 (0x0060 - 0x00B0)
class UPsyNetChannel_X : public UObject
{
public:
    class FString                ChannelName;                // 0x0060 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)
    unsigned long                bOpen : 1;                // 0x0070 (0x0004)
    [0x000000000000000000] [0x000000001]
    unsigned long                bReceivedFirstMessage : 1;    // 0x0070 (0x0004)
    [0x000000000000000000] [0x000000002]
    unsigned long                bClosed : 1;                // 0x0070 (0x0004)
    [0x000000000000000000] [0x000000004]
    unsigned long                bTimedOut : 1;                // 0x0070 (0x0004)
    [0x000000000000000000] [0x000000008]
    int32_t                    NextMessageID;                // 0x0074 (0x0004)
    [0x000000000000000000]
    float                        WaitForMessageTime;        // 0x0078 (0x0004)
    [0x000000000000000001] (CPF_Edit)

```

```

TArray<struct FPendingChannelService>      ServiceQueue;                // 0x0080
(0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
class UPsyNetServiceSubscriptions_X*      Subscriptions;                // 0x0090
(0x0008) [0x00000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
struct FScriptDelegate                    __EventClosed__Delegate;      // 0x0098 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetChannel_X");
}

return uClassPointer;
};

```

```

class FString GetDebugName();
void Close();
void Timeout();
void ClearTimeout();
void UpdateTimeout();
void ExecuteService(class UPsyNetClientService_X* Service);
void ExecuteNext();
class UAsyncTask* CreatePendingService(class UPsyNetClientService_X* Service, int32_t
MessageId);
class UAsyncTask* QueueServiceCall(class UPsyNetClientService_X* Service, class
UPsyNetMessage_X* Message);
void Open();
void Unsubscribe(struct FScriptDelegate Callback);
void Subscribe(class UClass* ServiceClass, struct FScriptDelegate Callback);
void Init(class FString InChannelName);
void EventClosed(class UPsyNetChannel_X* Channel);
};

```

```

// Class ProjectX.PsyNetServiceProvider_X
// 0x0038 (0x0060 - 0x0098)
class UPsyNetServiceProvider_X : public UObject
{
public:
class UPsyNetClientServiceCollection_X*   ServiceCollection;            // 0x0060
(0x0008) [0x0000000400000000]
class UPsyNetServiceSubscriptions_X*     Subscriptions;                // 0x0068
(0x0008) [0x00000004004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
TArray<class UPsyNetChannel_X*>          Channels;                      // 0x0070 (0x0010)
[0x0000000400040000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __EventServiceExecuted__Delegate; // 0x0080
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()

```

```

{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetServiceProvider_X");
}

return uClassPointer;
};

void FinalizeServiceTask(class UPsyNetClientService_X* Service, class
UTAsyncResult__PsyNetClientService_X* Task, class UError* Error);
void PrintServiceResult(class FString ServiceName, class UError* Error);
class UTAsyncResult__PsyNetClientService_X* ExecuteServiceMessage(class
UPsyNetConnection_X* Connection, class UPsyNetMessage_X* Message);
bool IsServiceRequest(class UPsyNetMessage_X* Message);
class UTAsyncResult__PsyNetClientService_X* ExecuteNotification(class
UPsyNetConnection_X* Connection, class UPsyNetMessage_X* Message);
void SetResponse(class UPsyNetClientService_X* Service, class UError* Error, class
UPsyNetMessage_X* Response);
class UTAsyncResult__PsyNetClientService_X* ExecuteRequest(class UPsyNetConnection_X*
Connection, class UPsyNetMessage_X* Request, class UPsyNetMessage_X* Response);
void HandleChannelClosed(class UPsyNetChannel_X* Channel);
class UPsyNetChannel_X* CreateChannel(class FString ChannelName);
void Unsubscribe(struct FScriptDelegate Callback);
void Subscribe(class UClass* ServiceClass, struct FScriptDelegate Callback);
void eventConstruct();
void EventServiceExecuted(class UPsyNetServiceProvider_X* ServiceProvider, class
UPsyNetClientService_X* Service);
};

// Class ProjectX.__PsyNetServiceProvider_X__ExecuteRequest_0x1
// 0x0008 (0x0060 - 0x0068)
class U__PsyNetServiceProvider_X__ExecuteRequest_0x1 : public UObject
{
public:
class UPsyNetMessage_X*                Response;                // 0x0060 (0x0008)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__PsyNetServiceProvider_X__ExecuteRequest_0x1");
}

return uClassPointer;
};

```

```

void __PsyNetServiceProvider_X__ExecuteRequest_0x1(class UPsyNetClientService_X* Result,
class UError* Error);
};

// Class ProjectX.__PsyNetServiceProvider_X__ExecuteServiceMessage_0x1
// 0x0020 (0x0060 - 0x0080)
class U__PsyNetServiceProvider_X__ExecuteServiceMessage_0x1 : public UObject
{
public:
class FString                               ServiceName;                // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class UPsyNetClientService_X*              Service;                    // 0x0070 (0x0008)
[0x000000000000000000]
class UAsyncResult__PsyNetClientService_X* Task;                        // 0x0078
(0x0008) [0x000000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__PsyNetServiceProvider_X__ExecuteServiceMessage_0x1");
}

return uClassPointer;
};

void __PsyNetServiceProvider_X__ExecuteServiceMessage_0x2(class UError* Err);
void __PsyNetServiceProvider_X__ExecuteServiceMessage_0x1(class UPsyNetClientService_X*
R, class UError* E);
};

// Class ProjectX.TAsyncResult__PsyNetClientService_X
// 0x0050 (0x00D0 - 0x0120)
class UAsyncResult__PsyNetClientService_X : public UAsyncTask
{
public:
class UPsyNetClientService_X*              Result;                    // 0x00D0 (0x0008)
[0x000000004000000000]
struct FScriptDelegate                    __EventResult__Delegate;    // 0x00D8 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __EventResultComplete__Delegate; // 0x00F0
(0x0018) [0x000000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                    __ResultDelegate__Delegate;    // 0x0108
(0x0018) [0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.TAsyncResult__PsyNetClientService_X");
}

return uClassPointer;
};

static class UAsyncResult__PsyNetClientService_X* CreateResultError(class UError* InError);
static class UAsyncResult__PsyNetClientService_X* CreateResult(class
UPsyNetClientService_X* InResult);
class UAsyncResult__PsyNetClientService_X* Copy();
void eventClearCallbacks();
class UAsyncResult__PsyNetClientService_X* eventSetResultWhen(class UAsyncTask* Other,
struct FScriptDelegate GetResultDelegate);
class UAsyncResult__PsyNetClientService_X* eventSetResult(class UPsyNetClientService_X*
InResult, class UError* InError);
class UAsyncResult__PsyNetClientService_X* NotifyOnResultComplete(struct FScriptDelegate
Callback);
class UAsyncResult__PsyNetClientService_X* NotifyOnResult(struct FScriptDelegate Callback);
class UPsyNetClientService_X* ResultDelegate();
void EventResultComplete(class UPsyNetClientService_X* OutResult, class UError* OutError);
void EventResult(class UPsyNetClientService_X* OutResult);
};

// Class ProjectX.__RegionConfig_X__GetSubRegions_0x1
// 0x0010 (0x0060 - 0x0070)
class U__RegionConfig_X__GetSubRegions_0x1 : public UObject
{
public:
    class FString                               SuperRegionID;                // 0x0060 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.__RegionConfig_X__GetSubRegions_0x1");
        }

        return uClassPointer;
    };

    bool __RegionConfig_X__GetSubRegions_0x1(class URegion_X* R);
};

// Class ProjectX.RegionConfig_X
// 0x0020 (0x0078 - 0x0098)
class URegionConfig_X : public UOnlineConfig_X
{
public:

```

```

TArray<class USuperRegion_X*> SuperRegions; // 0x0078
(0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
TArray<class URegion_X*> Regions; // 0x0088 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RegionConfig_X");
}

return uClassPointer;
};

struct FRegionSecret __RegionConfig_X__GetRegionSecrets_0x2(class URegion_X* R);
bool __RegionConfig_X__GetRegionSecrets_0x1(class URegion_X* R);
TArray<struct FRegionSecret> GetRegionSecrets();
TArray<class URegion_X*> GetSubRegions(class FString SuperRegionID);
class URegion_X* FindRegion(class FString RegionID);
class USuperRegion_X* FindSuperRegion(class FString SuperRegionID);
void Apply();
};

// Class ProjectX.__RemoteAvatarPermissions_X__GetPermissions_0x1
// 0x0008 (0x0060 - 0x0068)
class U__RemoteAvatarPermissions_X__GetPermissions_0x1 : public UObject
{
public:
class URemoteAvatarPermissionsRequestBatch_X* Batch; // 0x0060
(0x0008) [0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__RemoteAvatarPermissions_X__GetPermissions_0x1");
}

return uClassPointer;
};

void __RemoteAvatarPermissions_X__GetPermissions_0x1(struct FUniqueNetId PlayerID);
};

// Class ProjectX.RemoteAvatarPermissionsRequestBatch_X
// 0x0060 (0x0060 - 0x00C0)

```



```

class URemoteAvatarPermissionsRequestBatch_X : public UObject
{
public:
TArray<struct FUniqueNetId> Requested; // 0x0060 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
TArray<struct FUniqueNetId> Allowed; // 0x0070 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
TArray<struct FUniqueNetId> Disallowed; // 0x0080 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventAllowed__Delegate; // 0x0090
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventDisallowed__Delegate; // 0x00A8
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.RemoteAvatarPermissionsRequestBatch_X");
}

return uClassPointer;
};

void CheckFinished();
void SetDisallowed(struct FUniqueNetId PlayerID);
void SetAllowed(struct FUniqueNetId PlayerID);
void EventDisallowed(TArray<struct FUniqueNetId> DisallowedIds);
void EventAllowed(TArray<struct FUniqueNetId> AllowedIds);
};

// Class ProjectX.__RemoteAvatarPermissions_X__SetAvatarPermission_0x1
// 0x0048 (0x0060 - 0x00A8)
class U__RemoteAvatarPermissions_X__SetAvatarPermission_0x1 : public UObject
{
public:
struct FUniqueNetId PlayerID; // 0x0060 (0x0048)
[0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__RemoteAvatarPermissions_X__SetAvatarPermission_0x1");
}
}

```

```

return uClassPointer;
};

bool __RemoteAvatarPermissions_X__SetAvatarPermission_0x1(class
URemoteAvatarPermissionsRequest_X* R);
};

// Class ProjectX.__RPC_X__CreateTask_0x1
// 0x0018 (0x0060 - 0x0078)
class U__RPC_X__CreateTask_0x1 : public UObject
{
public:
    struct FScriptDelegate                InCallback;                // 0x0060 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.__RPC_X__CreateTask_0x1");
        }

        return uClassPointer;
    };

    void __RPC_X__CreateTask_0x1(class UError* Err);
};

// Class ProjectX.__RPCQueue_X__CreateBatch_0x2
// 0x0008 (0x0060 - 0x0068)
class U__RPCQueue_X__CreateBatch_0x2 : public UObject
{
public:
    class URPCBatch_X*                    Batch;                    // 0x0060 (0x0008)
    [0x000000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.__RPCQueue_X__CreateBatch_0x2");
        }

        return uClassPointer;
    };

    void __RPCQueue_X__CreateBatch_0x2(class UPsyNetMessage_X* Response, class UError*
Error);
};

```

```

};

// Class ProjectX.__RPCQueue_X__CreateBatchSingleRPC_0x1
// 0x0008 (0x0060 - 0x0068)
class U__RPCQueue_X__CreateBatchSingleRPC_0x1 : public UObject
{
public:
    class URPCBatch_X*          Batch;                // 0x0060 (0x0008)
    [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__RPCQueue_X__CreateBatchSingleRPC_0x1");
        }

        return uClassPointer;
    };

    void __RPCQueue_X__CreateBatchSingleRPC_0x1(class UPsyNetMessage_X* Response, class
    UError* Error);
};

// Class ProjectX.__ServerPlayerTracker_X__AddPlayer_0x1
// 0x0048 (0x0060 - 0x00A8)
class U__ServerPlayerTracker_X__AddPlayer_0x1 : public UObject
{
public:
    struct FUniqueNetId          PlayerID;                // 0x0060 (0x0048)
    [0x00000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.__ServerPlayerTracker_X__AddPlayer_0x1");
        }

        return uClassPointer;
    };

    bool __ServerPlayerTracker_X__AddPlayer_0x1(struct FUniqueNetId P);
};

// Class ProjectX.ServerPlayerTracker_X
// 0x0058 (0x0070 - 0x00C8)

```

```

class UServerPlayerTracker_X : public UComponent
{
public:
TArray<struct FUniqueNetId>          Players;                // 0x0070 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
TArray<class UAddReservationMessage_X*> Messages;            // 0x0080
(0x0010) [0x000000000000400000] (CPF_NeedCtorLink)
class UOnlineGameReservations_X*    Reservations;           // 0x0090
(0x0008) [0x000080000000000000]
struct FScriptDelegate              __EventPlayerAdded__Delegate; // 0x0098
(0x0018) [0x000000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate              __EventPlayerRemoved__Delegate; // 0x00B0
(0x0018) [0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ServerPlayerTracker_X");
}

return uClassPointer;
};

TArray<struct FUniqueNetId> __ServerPlayerTracker_X__Refresh_0x3(class
UAddReservationMessage_X* Message);
struct FUniqueNetId __ServerPlayerTracker_X__Refresh_0x2(struct FReservationData P);
bool __ServerPlayerTracker_X__Refresh_0x1(struct FReservationData P);
class FString GetPlayerName(struct FUniqueNetId PlayerID);
void Reset();
void Refresh();
void RemoveReservationMessage(class UAddReservationMessage_X* Message);
void AddReservationMessage(class UAddReservationMessage_X* Message);
void RemovePlayer(struct FUniqueNetId PlayerID);
void AddPlayer(struct FUniqueNetId PlayerID);
void EventPlayerRemoved(class UServerPlayerTracker_X* Tracker, struct FUniqueNetId PlayerID);
void EventPlayerAdded(class UServerPlayerTracker_X* Tracker, struct FUniqueNetId PlayerID);
};

// Class ProjectX.__StatusObserver_X__FindByAcceptedType_0x1
// 0x0008 (0x0060 - 0x0068)
class U__StatusObserver_X__FindByAcceptedType_0x1 : public UObject
{
public:
class UClass*          InType;                // 0x0060 (0x0008)
[0x000000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class
ProjectX.__StatusObserver_X__FindByAcceptedType_0x1");
}

return uClassPointer;
};

bool __StatusObserver_X__FindByAcceptedType_0x1(class UTriggerInfo* TT);
};

// Class ProjectX.StatusTrigger_X
// 0x0020 (0x0060 - 0x0080)
class UStatusTrigger_X : public UObject
{
public:
    unsigned long                bTriggered : 1;                // 0x0060 (0x0004)
    [0x0000000000000000] [0x00000001]
    struct FScriptDelegate        EventPropertyChange;          // 0x0068 (0x0018)
    [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.StatusTrigger_X");
        }

        return uClassPointer;
    };

    void EventPropertyChangeFunc();
    void EvaluateCondition(unsigned long InConditionalValue);
    bool IsTriggered();
    void ToggleTriggered();
    void DebugPrintInfo(class FString AddedInfo);
    void DebugPrint();
};

// Class ProjectX.TriggerInfo
// 0x0048 (0x0060 - 0x00A8)
class UTriggerInfo : public UObject
{
public:
    class UClass*                AcceptedType;                  // 0x0060 (0x0008)
    [0x0000000000000000]
    TArray<class UStatusTrigger_X*>    Triggers;                // 0x0068 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)
    struct FScriptDelegate        Callback;                     // 0x0078 (0x0018)

```

```

[0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate      __UpdateTrigger__Delegate;           // 0x0090
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.TriggerInfo");
}

return uClassPointer;
};

class UTriggerInfo* SetType(class UClass* InType);
void UpdateTrigger(class UStatusTrigger_X* InTrigger);
};

// Class ProjectX.StatusObserver_X
// 0x0040 (0x0060 - 0x00A0)
class UStatusObserver_X : public UObject
{
public:
TArray<class UTriggerInfo*>      TriggerTypes;                  // 0x0060 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
TArray<class UClass*>      AcceptedTypes;                      // 0x0070 (0x0010)
[0x0000000000040002] (CPF_Const | CPF_NeedCtorLink)
unsigned long      bInitComplete : 1;                          // 0x0080 (0x0004)
[0x0008000000000000] [0x00000001]
struct FScriptDelegate      __bInitComplete__ChangeNotify;    // 0x0088
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.StatusObserver_X");
}

return uClassPointer;
};

class UTriggerInfo* __StatusObserver_X__Init_0x1(class UClass* T);
void __bInitComplete__ChangeNotifyFunc();
void DebugPrintTriggers();
void DebugPrint();
bool ObservesTriggerType(class UStatusTrigger_X* InTrigger);
void UpdateTriggers(class UClass* InType);

```

```

void UnRegisterTriggers(TArray<class UStatusTrigger_X*>& InTriggers);
void RegisterTriggers(TArray<class UStatusTrigger_X*>& InTriggers);
void AddCallback(class UClass* InType, struct FScriptDelegate InCallback);
class UTriggerInfo* FindByAcceptedType(class UClass* InType);
void SetInitComplete();
void Init(class ULocalPlayer_X* LP);
};

// Class ProjectX.__StatusObserver_X__ObservesTriggerType_0x1
// 0x0008 (0x0060 - 0x0068)
class U__StatusObserver_X__ObservesTriggerType_0x1 : public UObject
{
public:
class UStatusTrigger_X*                InTrigger;                // 0x0060 (0x0008)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__StatusObserver_X__ObservesTriggerType_0x1");
}

return uClassPointer;
};

bool __StatusObserver_X__ObservesTriggerType_0x1(class UClass* C);
};

// Class ProjectX.__TAsyncResult__array_ClubInvite_X__Copy_0x1
// 0x0008 (0x0060 - 0x0068)
class U__TAsyncResult__array_ClubInvite_X__Copy_0x1 : public UObject
{
public:
class UTAsyncResult__array_ClubInvite_X*    Instance;                // 0x0060
(0x0008) [0x0001000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__array_ClubInvite_X__Copy_0x1");
}

return uClassPointer;
};

```

```

void __TAsyncResult__array_ClubInvite_X__Copy_0x1(TArray<class UClubInvite_X*> R, class
UErrors* E);
};

// Class ProjectX.TAsyncResult__array_ClubInvite_X
// 0x0058 (0x00D0 - 0x0128)
class UAsyncResult__array_ClubInvite_X : public UAsyncTask
{
public:
TArray<class UClubInvite_X*> Result; // 0x00D0 (0x0010)
[0x0001004000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventResult__Delegate; // 0x00E0 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventResultComplete__Delegate; // 0x00F8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __ResultDelegate__Delegate; // 0x0110
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.TAsyncResult__array_ClubInvite_X");
}

return uClassPointer;
};

static class UAsyncResult__array_ClubInvite_X* CreateResultError(class UErrors* InError);
static class UAsyncResult__array_ClubInvite_X* CreateResult(TArray<class UClubInvite_X*>
InResult);
class UAsyncResult__array_ClubInvite_X* Copy();
void eventClearCallbacks();
class UAsyncResult__array_ClubInvite_X* eventSetResultWhen(class UAsyncTask* Other, struct
FScriptDelegate GetResultDelegate);
class UAsyncResult__array_ClubInvite_X* eventSetResult(TArray<class UClubInvite_X*>
InResult, class UErrors* InError);
class UAsyncResult__array_ClubInvite_X* NotifyOnResultComplete(struct FScriptDelegate
Callback);
class UAsyncResult__array_ClubInvite_X* NotifyOnResult(struct FScriptDelegate Callback);
TArray<class UClubInvite_X*> ResultDelegate();
void EventResultComplete(TArray<class UClubInvite_X*> OutResult, class UErrors* OutError);
void EventResult(TArray<class UClubInvite_X*> OutResult);
};

// Class ProjectX.ClubInvite_X
// 0x00C8 (0x0060 - 0x0128)
class UClubInvite_X : public UObject
{
public:

```



```

uint64_t          ClubID;          // 0x0060 (0x0008)
[0x0001000000000000]
class FString          ClubName;          // 0x0068 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
class FString          ClubTag;          // 0x0078 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
struct FClubMember          InvitedBy;          // 0x0088 (0x00A0)
[0x0001000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ClubInvite_X");
}

return uClassPointer;
};

};

// Class ProjectX.__TAsyncResult__array_ClubInvite_X__NotifyOnResult_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__array_ClubInvite_X__NotifyOnResult_0x1 : public UObject
{
public:
struct FScriptDelegate          Callback;          // 0x0060 (0x0018)
[0x0001000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__array_ClubInvite_X__NotifyOnResult_0x1");
}

return uClassPointer;
};

void __TAsyncResult__array_ClubInvite_X__NotifyOnResult_0x1();
};

// Class ProjectX.__TAsyncResult__array_ClubInvite_X__NotifyOnResultComplete_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__array_ClubInvite_X__NotifyOnResultComplete_0x1 : public UObject
{
public:

```

```

struct FScriptDelegate          Callback;                // 0x0060 (0x0018)
[0x0001000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__array_ClubInvite_X__NotifyOnResultComplete_0x1");
}

return uClassPointer;
};

void __TAsyncResult__array_ClubInvite_X__NotifyOnResultComplete_0x1(class UError* Err);
};

// Class ProjectX.__TAsyncResult__array_ClubInvite_X__SetResultWhen_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__array_ClubInvite_X__SetResultWhen_0x1 : public UObject
{
public:
struct FScriptDelegate          GetResultDelegate;        // 0x0060 (0x0018)
[0x0001000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__array_ClubInvite_X__SetResultWhen_0x1");
}

return uClassPointer;
};

void __TAsyncResult__array_ClubInvite_X__SetResultWhen_0x1();
};

// Class ProjectX.__TAsyncResult__array_LanServerRecord_X__Copy_0x1
// 0x0008 (0x0060 - 0x0068)
class U__TAsyncResult__array_LanServerRecord_X__Copy_0x1 : public UObject
{
public:
class U__TAsyncResult__array_LanServerRecord_X*   Instance; // 0x0060
(0x0008) [0x0000000000000000]

public:

```

```

static UClass* StaticClass()
{
    static UClass* uClassPointer = nullptr;

    if (!uClassPointer)
    {
        uClassPointer = UObject::FindClass("Class
        ProjectX.__TAsyncResult__array_LanServerRecord_X__Copy_0x1");
    }

    return uClassPointer;
};

void __TAsyncResult__array_LanServerRecord_X__Copy_0x1(TArray<class
ULanServerRecord_X*> R, class UError* E);
};

// Class ProjectX.TAsyncResult__array_LanServerRecord_X
// 0x0058 (0x00D0 - 0x0128)
class UTAsyncResult__array_LanServerRecord_X : public UAsyncTask
{
public:
    TArray<class ULanServerRecord_X*>          Result;                // 0x00D0 (0x0010)
    [0x0000000400040000] (CPF_NeedCtorLink)
    struct FScriptDelegate      __EventResult__Delegate;            // 0x00E0 (0x0018)
    [0x0000000000040000] (CPF_NeedCtorLink)
    struct FScriptDelegate      __EventResultComplete__Delegate;    // 0x00F8
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
    struct FScriptDelegate      __ResultDelegate__Delegate;        // 0x0110
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.TAsyncResult__array_LanServerRecord_X");
        }

        return uClassPointer;
    };

    static class UTAsyncResult__array_LanServerRecord_X* CreateResultError(class UError* InError);
    static class UTAsyncResult__array_LanServerRecord_X* CreateResult(TArray<class
    ULanServerRecord_X*> InResult);
    class UTAsyncResult__array_LanServerRecord_X* Copy();
    void eventClearCallbacks();
    class UTAsyncResult__array_LanServerRecord_X* eventSetResultWhen(class UAsyncTask*
    Other, struct FScriptDelegate GetResultDelegate);
    class UTAsyncResult__array_LanServerRecord_X* eventSetResult(TArray<class
    ULanServerRecord_X*> InResult, class UError* InError);
    class UTAsyncResult__array_LanServerRecord_X* NotifyOnResultComplete(struct

```

```

FScriptDelegate Callback);
class UAsyncResult__array_LanServerRecord_X* NotifyOnResult(struct FScriptDelegate
Callback);
TArray<class ULanServerRecord_X*> ResultDelegate();
void EventResultComplete(TArray<class ULanServerRecord_X*> OutResult, class UError*
OutError);
void EventResult(TArray<class ULanServerRecord_X*> OutResult);
};

```

```

// Class ProjectX.LanServerRecord_X
// 0x0020 (0x0060 - 0x0080)
class ULanServerRecord_X : public UObject
{

```

```

public:
class FString                               ServerId;                               // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString                               MetaData;                               // 0x0070 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.LanServerRecord_X");
}

```

```

return uClassPointer;
};

```

```

};

```

```

// Class ProjectX.__TAsyncResult__array_LanServerRecord_X__NotifyOnResult_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__array_LanServerRecord_X__NotifyOnResult_0x1 : public UObject
{

```

```

public:
struct FScriptDelegate                     Callback;                               // 0x0060 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__array_LanServerRecord_X__NotifyOnResult_0x1");
}

```

```

return uClassPointer;

```

```

};

void __TAsyncResult__array_LanServerRecord_X__NotifyOnResult_0x1();
};

// Class ProjectX.__TAsyncResult__array_LanServerRecord_X__NotifyOnResultComplete_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__array_LanServerRecord_X__NotifyOnResultComplete_0x1 : public
UObject
{
public:
    struct FScriptDelegate                Callback;                // 0x0060 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__TAsyncResult__array_LanServerRecord_X__NotifyOnResultComplete_0x1");
        }

        return uClassPointer;
    };

    void __TAsyncResult__array_LanServerRecord_X__NotifyOnResultComplete_0x1(class UError*
    Err);
};

// Class ProjectX.__TAsyncResult__array_LanServerRecord_X__SetResultWhen_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__array_LanServerRecord_X__SetResultWhen_0x1 : public UObject
{
public:
    struct FScriptDelegate                GetResultDelegate;        // 0x0060 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__TAsyncResult__array_LanServerRecord_X__SetResultWhen_0x1");
        }

        return uClassPointer;
    };

```

```

void __TAsyncResult__array_LanServerRecord_X__SetResultWhen_0x1();
};

// Class ProjectX.__TAsyncResult__ClubDetails_X__Copy_0x1
// 0x0008 (0x0060 - 0x0068)
class U__TAsyncResult__ClubDetails_X__Copy_0x1 : public UObject
{
public:
class UTAAsyncResult__ClubDetails_X*          Instance;                      // 0x0060 (0x0008)
[0x0001000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__ClubDetails_X__Copy_0x1");
}

return uClassPointer;
};

void __TAsyncResult__ClubDetails_X__Copy_0x1(class UClubDetails_X* R, class UError* E);
};

// Class ProjectX.TAsyncResult__ClubDetails_X
// 0x0050 (0x00D0 - 0x0120)
class UTAAsyncResult__ClubDetails_X : public UAsyncTask
{
public:
class UClubDetails_X*          Result;                      // 0x00D0 (0x0008)
[0x0001004000000000]
struct FScriptDelegate          __EventResult__Delegate;      // 0x00D8 (0x0018)
[0x00000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate          __EventResultComplete__Delegate; // 0x00F0
(0x0018) [0x00000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate          __ResultDelegate__Delegate;   // 0x0108
(0x0018) [0x00000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.TAsyncResult__ClubDetails_X");
}

return uClassPointer;
};

```

```

static class UAsyncResult__ClubDetails_X* CreateResultError(class UError* InError);
static class UAsyncResult__ClubDetails_X* CreateResult(class UClubDetails_X* InResult);
class UAsyncResult__ClubDetails_X* Copy();
void eventClearCallbacks();
class UAsyncResult__ClubDetails_X* eventSetResultWhen(class UAsyncTask* Other, struct
FScriptDelegate GetResultDelegate);
class UAsyncResult__ClubDetails_X* eventSetResult(class UClubDetails_X* InResult, class
UError* InError);
class UAsyncResult__ClubDetails_X* NotifyOnResultComplete(struct FScriptDelegate Callback);
class UAsyncResult__ClubDetails_X* NotifyOnResult(struct FScriptDelegate Callback);
class UClubDetails_X* ResultDelegate();
void EventResultComplete(class UClubDetails_X* OutResult, class UError* OutError);
void EventResult(class UClubDetails_X* OutResult);
};

// Class ProjectX.__TAsyncResult__ClubDetails_X__NotifyOnResult_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__ClubDetails_X__NotifyOnResult_0x1 : public UObject
{
public:
    struct FScriptDelegate          Callback;                                // 0x0060 (0x0018)
    [0x0001000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__ClubDetails_X__NotifyOnResult_0x1");
        }

        return uClassPointer;
    };

    void __TAsyncResult__ClubDetails_X__NotifyOnResult_0x1();
};

// Class ProjectX.__TAsyncResult__ClubDetails_X__NotifyOnResultComplete_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__ClubDetails_X__NotifyOnResultComplete_0x1 : public UObject
{
public:
    struct FScriptDelegate          Callback;                                // 0x0060 (0x0018)
    [0x0001000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__ClubDetails_X__NotifyOnResultComplete_0x1");
}

return uClassPointer;
};

void __TAsyncResult__ClubDetails_X__NotifyOnResultComplete_0x1(class UError* Err);
};

// Class ProjectX.__TAsyncResult__ClubDetails_X__SetResultWhen_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__ClubDetails_X__SetResultWhen_0x1 : public UObject
{
public:
    struct FScriptDelegate                GetResultDelegate;                // 0x0060 (0x0018)
    [0x0001000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__ClubDetails_X__SetResultWhen_0x1");
        }

        return uClassPointer;
    };

    void __TAsyncResult__ClubDetails_X__SetResultWhen_0x1();
};

// Class ProjectX.__TAsyncResult__PsyNetClientService_X__Copy_0x1
// 0x0008 (0x0060 - 0x0068)
class U__TAsyncResult__PsyNetClientService_X__Copy_0x1 : public UObject
{
public:
    class U__TAsyncResult__PsyNetClientService_X*    Instance;                // 0x0060
    (0x0008) [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__PsyNetClientService_X__Copy_0x1");
        }
    }

```



```

}

return uClassPointer;
};

void __TAsyncResult__PsyNetClientService_X__Copy_0x1(class UPsyNetClientService_X* R,
class UError* E);
};

// Class ProjectX.__TAsyncResult__PsyNetClientService_X__NotifyOnResult_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__PsyNetClientService_X__NotifyOnResult_0x1 : public UObject
{
public:
    struct FScriptDelegate                Callback;                // 0x0060 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__TAsyncResult__PsyNetClientService_X__NotifyOnResult_0x1");
        }

        return uClassPointer;
    };

    void __TAsyncResult__PsyNetClientService_X__NotifyOnResult_0x1();
};

// Class ProjectX.__TAsyncResult__PsyNetClientService_X__NotifyOnResultComplete_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__PsyNetClientService_X__NotifyOnResultComplete_0x1 : public UObject
{
public:
    struct FScriptDelegate                Callback;                // 0x0060 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__TAsyncResult__PsyNetClientService_X__NotifyOnResultComplete_0x1");
        }

        return uClassPointer;
    };

```

```

};

void __TAsyncResult__PsyNetClientService_X__NotifyOnResultComplete_0x1(class UError* Err);
};

// Class ProjectX.__TAsyncResult__PsyNetClientService_X__SetResultWhen_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__PsyNetClientService_X__SetResultWhen_0x1 : public UObject
{
public:
    struct FScriptDelegate          GetResultDelegate;          // 0x0060 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__TAsyncResult__PsyNetClientService_X__SetResultWhen_0x1");
        }

        return uClassPointer;
    };

    void __TAsyncResult__PsyNetClientService_X__SetResultWhen_0x1();
};

// Class ProjectX.__TAsyncResult__PsyNetMessage_X__Copy_0x1
// 0x0008 (0x0060 - 0x0068)
class U__TAsyncResult__PsyNetMessage_X__Copy_0x1 : public UObject
{
public:
    class UAsyncResult__PsyNetMessage_X*      Instance;          // 0x0060
    (0x0008) [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__TAsyncResult__PsyNetMessage_X__Copy_0x1");
        }

        return uClassPointer;
    };

    void __TAsyncResult__PsyNetMessage_X__Copy_0x1(class UAsyncResult__PsyNetMessage_X* R, class
    UError* E);

```

```

};

// Class ProjectX.TAsyncResult__PsyNetMessage_X
// 0x0050 (0x00D0 - 0x0120)
class UAsyncResult__PsyNetMessage_X : public UAsyncTask
{
public:
    class UPsyNetMessage_X*          Result;                // 0x00D0 (0x0008)
    [0x0000000400000000]
    struct FScriptDelegate          __EventResult__Delegate;    // 0x00D8 (0x0018)
    [0x0000000000040000] (CPF_NeedCtorLink)
    struct FScriptDelegate          __EventResultComplete__Delegate;    // 0x00F0
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
    struct FScriptDelegate          __ResultDelegate__Delegate;    // 0x0108
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.TAsyncResult__PsyNetMessage_X");
        }

        return uClassPointer;
    };

    static class UAsyncResult__PsyNetMessage_X* CreateResultError(class UError* InError);
    static class UAsyncResult__PsyNetMessage_X* CreateResult(class UPsyNetMessage_X* InResult);
    class UAsyncResult__PsyNetMessage_X* Copy();
    void eventClearCallbacks();
    class UAsyncResult__PsyNetMessage_X* eventSetResultWhen(class UAsyncTask* Other, struct FScriptDelegate GetResultDelegate);
    class UAsyncResult__PsyNetMessage_X* eventSetResult(class UPsyNetMessage_X* InResult, class UError* InError);
    class UAsyncResult__PsyNetMessage_X* NotifyOnResultComplete(struct FScriptDelegate Callback);
    class UAsyncResult__PsyNetMessage_X* NotifyOnResult(struct FScriptDelegate Callback);
    class UPsyNetMessage_X* ResultDelegate();
    void EventResultComplete(class UPsyNetMessage_X* OutResult, class UError* OutError);
    void EventResult(class UPsyNetMessage_X* OutResult);
};

// Class ProjectX.__TAsyncResult__PsyNetMessage_X__NotifyOnResult_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__PsyNetMessage_X__NotifyOnResult_0x1 : public UObject
{
public:
    struct FScriptDelegate          Callback;                // 0x0060 (0x0018)
    [0x0000000000040000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__PsyNetMessage_X__NotifyOnResult_0x1");
}

return uClassPointer;
};

void __TAsyncResult__PsyNetMessage_X__NotifyOnResult_0x1();
};

// Class ProjectX.__TAsyncResult__PsyNetMessage_X__NotifyOnResultComplete_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__PsyNetMessage_X__NotifyOnResultComplete_0x1 : public UObject
{
public:
struct FScriptDelegate          Callback;                // 0x0060 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__PsyNetMessage_X__NotifyOnResultComplete_0x1");
}

return uClassPointer;
};

void __TAsyncResult__PsyNetMessage_X__NotifyOnResultComplete_0x1(class UError* Err);
};

// Class ProjectX.__TAsyncResult__PsyNetMessage_X__SetResultWhen_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__PsyNetMessage_X__SetResultWhen_0x1 : public UObject
{
public:
struct FScriptDelegate          GetResultDelegate;        // 0x0060 (0x0018)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__PsyNetMessage_X__SetResultWhen_0x1");
}

return uClassPointer;
};

void __TAsyncResult__PsyNetMessage_X__SetResultWhen_0x1();
};

// Class ProjectX.__TAsyncResult__Texture2DDynamic__Copy_0x1
// 0x0008 (0x0060 - 0x0068)
class U__TAsyncResult__Texture2DDynamic__Copy_0x1 : public UObject
{
public:
    class U__TAsyncResult__Texture2DDynamic*      Instance;                // 0x0060
    (0x0008) [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__Texture2DDynamic__Copy_0x1");
        }

        return uClassPointer;
    };

    void __TAsyncResult__Texture2DDynamic__Copy_0x1(class UTexture2DDynamic* R, class
    UError* E);
};

// Class ProjectX.TAsyncResult__Texture2DDynamic
// 0x0050 (0x00D0 - 0x0120)
class U__TAsyncResult__Texture2DDynamic : public UAsyncTask
{
public:
    class UTexture2DDynamic*      Result;                // 0x00D0 (0x0008)
    [0x0000000400000000]
    struct FScriptDelegate      __EventResult__Delegate;    // 0x00D8 (0x0018)
    [0x0000000000040000] (CPF_NeedCtorLink)
    struct FScriptDelegate      __EventResultComplete__Delegate;    // 0x00F0
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)
    struct FScriptDelegate      __ResultDelegate__Delegate;    // 0x0108
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:

```

```

static UClass* StaticClass()
{
    static UClass* uClassPointer = nullptr;

    if (!uClassPointer)
    {
        uClassPointer = UObject::FindClass("Class ProjectX.TAsyncResult__Texture2DDynamic");
    }

    return uClassPointer;
};

static class UTexture2DDynamic* CreateResultError(class UError* InError);
static class UTexture2DDynamic* CreateResult(class UTexture2DDynamic*
InResult);
class UTexture2DDynamic* Copy();
void eventClearCallbacks();
class UTexture2DDynamic* eventSetResultWhen(class UAsyncTask* Other, struct
FScriptDelegate GetResultDelegate);
class UTexture2DDynamic* eventSetResult(class UTexture2DDynamic* InResult,
class UError* InError);
class UTexture2DDynamic* NotifyOnResultComplete(struct FScriptDelegate
Callback);
class UTexture2DDynamic* NotifyOnResult(struct FScriptDelegate Callback);
class UTexture2DDynamic* ResultDelegate();
void EventResultComplete(class UTexture2DDynamic* OutResult, class UError* OutError);
void EventResult(class UTexture2DDynamic* OutResult);
};

// Class ProjectX.__TAsyncResult__Texture2DDynamic__NotifyOnResult_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__Texture2DDynamic__NotifyOnResult_0x1 : public UObject
{
public:
    struct FScriptDelegate          Callback;                // 0x0060 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
ProjectX.__TAsyncResult__Texture2DDynamic__NotifyOnResult_0x1");
        }

        return uClassPointer;
    };

    void __TAsyncResult__Texture2DDynamic__NotifyOnResult_0x1();
};

```

```

// Class ProjectX.__TAsyncResult__Texture2DDynamic__NotifyOnResultComplete_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__Texture2DDynamic__NotifyOnResultComplete_0x1 : public UObject
{
public:
    struct FScriptDelegate          Callback;                // 0x0060 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__TAsyncResult__Texture2DDynamic__NotifyOnResultComplete_0x1");
        }

        return uClassPointer;
    };

    void __TAsyncResult__Texture2DDynamic__NotifyOnResultComplete_0x1(class UError* Err);
};

// Class ProjectX.__TAsyncResult__Texture2DDynamic__SetResultWhen_0x1
// 0x0018 (0x0060 - 0x0078)
class U__TAsyncResult__Texture2DDynamic__SetResultWhen_0x1 : public UObject
{
public:
    struct FScriptDelegate          GetResultDelegate;        // 0x0060 (0x0018)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class
            ProjectX.__TAsyncResult__Texture2DDynamic__SetResultWhen_0x1");
        }

        return uClassPointer;
    };

    void __TAsyncResult__Texture2DDynamic__SetResultWhen_0x1();
};

// Class ProjectX.__WebCache_X__DownloadData_0x1
// 0x0008 (0x0060 - 0x0068)
class U__WebCache_X__DownloadData_0x1 : public UObject
{

```

```

public:
class UWebRequest_X*                               WebRequest;                               // 0x0060 (0x0008)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.__WebCache_X__DownloadData_0x1");
}

return uClassPointer;
};

void __WebCache_X__DownloadData_0x1(class FString Key, class FString Value);
};

// Class ProjectX.WebCache_X
// 0x0028 (0x0060 - 0x0088)
class UWebCache_X : public UObject
{
public:
TArray<struct FCachedDataRequest> Requests; // 0x0060
(0x0010) [0x00000000000482000] (CPF_Transient | CPF_Component | CPF_NeedCtorLink)
struct FScriptDelegate __CachedDataCallback__Delegate; // 0x0070
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.WebCache_X");
}

return uClassPointer;
};

class FString GetRequestDebugString(struct FCachedDataRequest Request);
void HandleWebRequest(class UWebRequest_X* WebRequest);
void DownloadData(class FString URL, class FString ETag, class UStringMap* Headers, unsigned
long bZipResponse);
void HandleLoadFromCache(class ULocalCache_X* Cache, class UCachedWebData_X*
CacheObject, class UError* Error);
void RaiseUrlEmptyError(struct FScriptDelegate Callback);
void SyncDataInternal(class FString URL, struct FScriptDelegate Callback, unsigned long
bZipResponse, class UStringMap* Headers);
void SyncDataZipped(class FString URL, struct FScriptDelegate Callback, class UStringMap*
Headers);

```



```

void SyncData(class FString URL, struct FScriptDelegate Callback, class UStringMap* Headers);
void ClearCache(class FString URL);
static class FString GetCachedPath(class FString URL);
void CachedDataCallback(class UCachedWebData_X* CachedData);
};

// Class ProjectX.__WebImageCache_X__HandleImageData_0x1
// 0x0008 (0x0060 - 0x0068)
class U__WebImageCache_X__HandleImageData_0x1 : public UObject
{
public:
class UWebImageDownload_X*          Download;          // 0x0060 (0x0008)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__WebImageCache_X__HandleImageData_0x1");
}

return uClassPointer;
};

void __WebImageCache_X__HandleImageData_0x1(class FString __, struct FImageLayout Image);
};

// Class ProjectX.WebImageDownload_X
// 0x0028 (0x0060 - 0x0088)
class UWebImageDownload_X : public UObject
{
public:
class FString          URL;          // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
unsigned long          bSRGB : 1;          // 0x0070 (0x0004)
[0x0000000000000000] [0x00000001]
class UTexture2DDynamic* Texture;          // 0x0078 (0x0008)
[0x0000000000000000]
class UAsyncResult__Texture2DDynamic* AsyncResult;          // 0x0080
(0x0008) [0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.WebImageDownload_X");
}
}

```

```

return uClassPointer;
};

};

// Class ProjectX.WebImageCache_X
// 0x0028 (0x0060 - 0x0088)
class UWebImageCache_X : public UObject
{
public:
class UImageDecoder*                               Decoder;                               // 0x0060 (0x0008)
[0x0000000000000000]
class UStringObjectMap*                             Downloads;                             // 0x0068 (0x0008)
[0x00000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
struct FScriptDelegate                             __ImageSyncCallback__Delegate;         // 0x0070
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.WebImageCache_X");
}

return uClassPointer;
};

void SyncImageUrl(class FString URL, struct FScriptDelegate Callback);
void HandleImageDecoded(class UWebImageDownload_X* Download, struct FImageLayout Image);
void SetDownloadError(class UWebImageDownload_X* Download, class UError* Error);
void HandleImageData(class UWebImageDownload_X* Download, class UCachedWebData_X* Data);
class UWebImageDownload_X* StartSync(class FString URL);
void HandleUncachedImageDownload(class UTAsyncResult__Texture2DDynamic* Task, class UTexture2DDynamic* Texture);
class UTAsyncResult__Texture2DDynamic* SyncUncachedImage(class FString URL, unsigned long bSRGB);
class UWebImageDownload_X* GetDownload(class FString URL);
class UTAsyncResult__Texture2DDynamic* SyncImageSRGB(class FString URL);
class UTAsyncResult__Texture2DDynamic* SyncImage(class FString URL);
void eventConstruct();
void ImageSyncCallback(class UTexture2DDynamic* Texture);
};

// Class ProjectX.__WebImageCache_X__StartSync_0x1
// 0x0008 (0x0060 - 0x0068)
class U__WebImageCache_X__StartSync_0x1 : public UObject
{
public:

```

```

class UWebImageDownload_X*           Download;           // 0x0060 (0x0008)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.__WebImageCache_X__StartSync_0x1");
}

return uClassPointer;
};

void __WebImageCache_X__StartSync_0x1(class UCachedWebData_X* Data);
};

// Class ProjectX.__WebImageCache_X__SyncUncachedImage_0x1
// 0x0008 (0x0060 - 0x0068)
class U__WebImageCache_X__SyncUncachedImage_0x1 : public UObject
{
public:
class UTaskAsyncResult__Texture2DDynamic*           Task;           // 0x0060
(0x0008) [0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.__WebImageCache_X__SyncUncachedImage_0x1");
}

return uClassPointer;
};

void __WebImageCache_X__SyncUncachedImage_0x1(struct FOnlineImageDownload
ImageInfo);
};

// Class ProjectX._ReservationTypes_X
// 0x0000 (0x0060 - 0x0060)
class U_ReservationTypes_X : public UObject
{
public:

public:
static UClass* StaticClass()
{

```

```

static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX._ReservationTypes_X");
}

return uClassPointer;
};

};

// Class ProjectX.ActionQueue_X
// 0x0038 (0x0070 - 0x00A8)
class UActionQueue_X : public UComponent
{
public:
    float ActionsPerSecond; // 0x0070 (0x0004)
    [0x0000000000000001] (CPF_Edit)
    TArray<struct FScriptDelegate> Queue; // 0x0078 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)
    int32_t QueueIndex; // 0x0088 (0x0004)
    [0x0000000000000000]
    struct FScriptDelegate __ActionDelegate__Delegate; // 0x0090
    (0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.ActionQueue_X");
        }

        return uClassPointer;
    };

    void Tick();
    void Add(struct FScriptDelegate Callback);
    void ActionDelegate();
};

// Class ProjectX.ActivateAnimSeriesComponent_X
// 0x000C (0x00A4 - 0x00B0)
class UActivateAnimSeriesComponent_X : public UActorComponent_X
{
public:
    struct FName AnimNodeName; // 0x00A8 (0x0008)
    [0x0000000000000001] (CPF_Edit)

public:
    static UClass* StaticClass()

```

```

{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class ProjectX.ActivateAnimSeriesComponent_X");
}

return UClassPointer;
};

void SetAnimSeriesActiveInComponent(class USkeletalMeshComponent* SKC, unsigned long
bActive);
void SetAnimSeriesActive(unsigned long bActive);
void eventDetached();
void eventAttached();
};

// Class ProjectX.AddReservationMessagePrivate_X
// 0x0094 (0x00CC - 0x0160)
class UAddReservationMessagePrivate_X : public UAddReservationMessage_X
{
public:
struct FCustomMatchSettings          Settings;          // 0x00D0 (0x0090)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class ProjectX.AddReservationMessagePrivate_X");
}

return UClassPointer;
};

class FString GetDebugString();
class UAddReservationMessagePrivate_X* SetSettings(struct FCustomMatchSettings&
InSettings);
};

// Class ProjectX.AddReservationMessagePublic_X
// 0x0034 (0x00CC - 0x0100)
class UAddReservationMessagePublic_X : public UAddReservationMessage_X
{
public:
struct FJoinMatchSettings            Settings;            // 0x00D0 (0x0020)
[0x000000000000400000] (CPF_NeedCtorLink)
TArray<struct FOnlinePlayerMapPrefs> MapPrefs;            // 0x00F0
(0x0010) [0x000000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.AddReservationMessagePublic_X");
}

return uClassPointer;
};

class UAddReservationMessage_X* AddPlayers();
void SetPlayerPref(class UGameSettingPlaylist_X* Playlist, class UOnlinePlayer_X* OnlinePlayer);
void GetPlayerMapPrefs(struct FUniqueNetId PlayerID, TArray<struct FName>& Likes,
TArray<struct FName>& Dislikes);
class UAddReservationMessage_X* AddPlayer(struct FUniqueNetId PlayerID, class FString
PlayerName, unsigned long bRemotePlayer);
class FString GetDebugString();
class FString GetPlayerDebugString(struct FReservationPlayerData Player);
class UAddReservationMessagePublic_X* SetSettings(struct FJoinMatchSettings& InSettings);
};

// Class ProjectX.AgeGateRequiredResponse
// 0x0004 (0x0060 - 0x0064)
class UAgeGateRequiredResponse : public UObject
{
public:
unsigned long                                ageGateRequired : 1;                // 0x0060 (0x0004)
[0x0000000000000000] [0x00000001]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.AgeGateRequiredResponse");
}

return uClassPointer;
};

};

// Class ProjectX.AppConfig_X
// 0x0004 (0x0078 - 0x007C)
class UAppConfig_X : public UOnlineConfig_X
{
public:
unsigned long                                bSpinSleep : 1;                        // 0x0078 (0x0004)
[0x00000000000004001] [0x00000001] (CPF_Edit | CPF_Config)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.AppConfig_X");
}

return uClassPointer;
};

void Undo();
void Apply();
};

// Class ProjectX.AprilConfig_X
// 0x0030 (0x0078 - 0x00A8)
class UAprilConfig_X : public UOnlineConfig_X
{
public:
uint64_t StartTime; // 0x0078 (0x0008)
[0x0000000000000001] (CPF_Edit)
uint64_t EndTime; // 0x0080 (0x0008)
[0x0000000000000001] (CPF_Edit)
unsigned long bChangeRankedIcons : 1; // 0x0088 (0x0004)
[0x0000000000000001] [0x00000001] (CPF_Edit)
unsigned long bChangePlaylists : 1; // 0x0088 (0x0004)
[0x0000000000000001] [0x00000002] (CPF_Edit)
unsigned long bChangeCrowd : 1; // 0x0088 (0x0004)
[0x0001000000000001] [0x00000004] (CPF_Edit)
TArray<struct FQuickChatOverridePair> QuickChatDisplayOverrides; // 0x0090
(0x0010) [0x0000000000400001] (CPF_Edit | CPF_NeedCtorLink)
float HatScale; // 0x00A0 (0x0004)
[0x0000000000000001] (CPF_Edit)
float AntennaScale; // 0x00A4 (0x0004)
[0x0000000000000001] (CPF_Edit)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.AprilConfig_X");
}

return uClassPointer;
};

bool IsActive();

```

```

};

// Class ProjectX.SeqEvent_Spawned_X
// 0x000C (0x017C - 0x0188)
class USeqEvent_Spawned_X : public USequenceEvent
{
public:
    class AActor*                Spawned;                // 0x0180 (0x0008)
    [0x00000000000002000] (CPF_Transient)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.SeqEvent_Spawned_X");
        }

        return uClassPointer;
    };

    static void TriggerFor(class AActor* A);
};

// Class ProjectX.VanityConfig_X
// 0x0008 (0x0078 - 0x0080)
class UVanityConfig_X : public UOnlineConfig_X
{
public:
    float                AvatarRequestDelaySeconds;        // 0x0078 (0x0004)
    [0x00000000000000001] (CPF_Edit)
    int32_t                AvatarRequestMaxBatchSize;        // 0x007C (0x0004)
    [0x00000000000000001] (CPF_Edit)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.VanityConfig_X");
        }

        return uClassPointer;
    };

};

// Class ProjectX.BanMessage_X
// 0x00A8 (0x0060 - 0x0108)
class UBanMessage_X : public UObject

```



```

{
public:
struct FUniqueNetId                PlayerID;                // 0x0060 (0x0048)
[0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
uint8_t                            BanType;                // 0x00A8 (0x0001)
[0x0000000040000000] (CPF_EditInlineNotify)
uint8_t                            SeverityType;            // 0x00A9 (0x0001)
[0x0000000040000000] (CPF_EditInlineNotify)
class FString                      Title;                    // 0x00B0 (0x0010)
[0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
class FString                      Body;                     // 0x00C0 (0x0010)
[0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
int32_t                            BannedMinutes;           // 0x00D0 (0x0004)
[0x0000000040000000] (CPF_EditInlineNotify)
unsigned long                      bContributedToBan : 1;     // 0x00D4 (0x0004)
[0x0000000040000000] [0x00000001] (CPF_EditInlineNotify)
TArray<class FString>              Citations;                // 0x00D8 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
float                             BannedUntilTime;           // 0x00E8 (0x0004)
[0x0000000040000000] (CPF_EditInlineNotify)
struct FScriptDelegate             __EventBanEnded__Delegate; // 0x00F0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.BanMessage_X");
}

return uClassPointer;
};

void __BanMessage_X__Init_0x1();
bool IsPermanentlyBanned();
bool IsBanned();
int32_t GetMinutesRemaining();
float GetSecondsRemaining();
void Init();
void EventBanEnded(class UBanMessage_X* BanMessage);
};

// Class ProjectX.BlockStatusMetrics_X
// 0x0000 (0x0080 - 0x0080)
class UBlockStatusMetrics_X : public UMetricsGroup_X
{
public:

public:
static UClass* StaticClass()
{

```

```

static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.BlockStatusMetrics_X");
}

return uClassPointer;
};

void BlockListDownloadTimeout(unsigned long bValidStatus);
};

// Class ProjectX.BlockStatusReporterConfig_X
// 0x0008 (0x0078 - 0x0080)
class UBlockStatusReporterConfig_X : public UOnlineConfig_X
{
public:
    float DownloadTimeoutSeconds; // 0x0078 (0x0004)
    [0x0000000000000001] (CPF_Edit)
    unsigned long bSubmitBugReportOnTimeout : 1; // 0x007C
    (0x0004) [0x0000000000000001] [0x00000001] (CPF_Edit)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.BlockStatusReporterConfig_X");
        }

        return uClassPointer;
    };
};

// Class ProjectX.BlogConfig_X
// 0x0020 (0x0078 - 0x0098)
class UBlogConfig_X : public UOnlineConfig_X
{
public:
    TArray<class UBlogTile_X*> Entries; // 0x0078 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)
    class FString MotD; // 0x0088 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)

```

```

{
uClassPointer = UObject::FindClass("Class ProjectX.BlogConfig_X");
}

return uClassPointer;
};

int32_t __BlogConfig_X__Apply_0x2(class UBlogTile_X* L, class UBlogTile_X* R);
void __BlogConfig_X__Apply_0x1(class UBlogTile_X* X);
void Apply();
};

// Class ProjectX.BlogTile_X
// 0x00B8 (0x0060 - 0x0118)
class UBlogTile_X : public UObject
{
public:
class FString Title; // 0x0060 (0x0010)
[0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
class FString DescriptionHeader; // 0x0070 (0x0010)
[0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
class FString Description; // 0x0080 (0x0010)
[0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
class FString WebURL; // 0x0090 (0x0010)
[0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
class FString CarName; // 0x00A0 (0x0010)
[0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)
class FString ImageURL; // 0x00B0 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
class FString StartTime; // 0x00C0 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
uint64_t StartTimeEpoch; // 0x00D0 (0x0008)
[0x0000000000000000]
class FString EndTime; // 0x00D8 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
uint64_t EndTimeEpoch; // 0x00E8 (0x0008)
[0x0001000000000000]
uint8_t LinkType; // 0x00F0 (0x0001)
[0x0000000040000000] (CPF_EditInlineNotify)
int32_t ShopID; // 0x00F4 (0x0004)
[0x0000000040000000] (CPF_EditInlineNotify)
int32_t ShopItemID; // 0x00F8 (0x0004)
[0x0000000040000000] (CPF_EditInlineNotify)
int32_t ProductID; // 0x00FC (0x0004)
[0x0000000040000000] (CPF_EditInlineNotify)
int32_t CategoryID; // 0x0100 (0x0004)
[0x0000000040000000] (CPF_EditInlineNotify)
int32_t PlaylistID; // 0x0104 (0x0004)
[0x0000000040000000] (CPF_EditInlineNotify)
class FString InfoURL; // 0x0108 (0x0010)
[0x0000000040400000] (CPF_NeedCtorLink | CPF_EditInlineNotify)

public:
static UClass* StaticClass()

```

```

{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.BlogTile_X");
}

return uClassPointer;
};

class FString GetTileID();
};

// Class ProjectX.BreadcrumbConfig_X
// 0x0004 (0x0078 - 0x007C)
class UBreadcrumbConfig_X : public UOnlineConfig_X
{
public:
int32_t PollCrumbsIntervalSeconds; // 0x0078 (0x0004)
[0x0000000000000001] (CPF_Edit)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.BreadcrumbConfig_X");
}

return uClassPointer;
};
};

// Class ProjectX.BugMetrics_X
// 0x0010 (0x0080 - 0x0090)
class UBugMetrics_X : public UMetricsGroup_X
{
public:
TArray<struct FName> ReportedIDs; // 0x0080 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.BugMetrics_X");
}
}

```

```

return uClassPointer;
};

void Report(struct FName Id, class FString Details, class FString Trace);
void BugReport(struct FName Id, class FString Details);
static void StaticReport(struct FName Id, class FString Details);
};

// Class ProjectX.CacheTimer_X
// 0x0028 (0x0070 - 0x0098)
class UCacheTimer_X : public UComponent
{
public:
    float                CacheTimeoutSeconds;                // 0x0070 (0x0004)
    [0x0000000000000001] (CPF_Edit)
    float                CacheExpireTime;                    // 0x0074 (0x0004)
    [0x0000004000002000] (CPF_Transient)
    unsigned long        bEnabled : 1;                      // 0x0078 (0x0004)
    [0x0000004000002000] [0x00000001] (CPF_Transient)
    struct FScriptDelegate __EventExpired__Delegate;        // 0x0080 (0x0018)
    [0x0000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.CacheTimer_X");
        }

        return uClassPointer;
    };

    float GetRandomCacheTimeoutTime();
    void OnExpired();
    void OnDisabled();
    void OnEnabled();
    void SetEnabled(unsigned long bEnableExpiration);
    void Reset(float TimeoutTime);
    bool IsExpired();
    void EventExpired(class UCacheTimer_X* Timer);
};

// Class ProjectX.CameraStateBlender_X
// 0x00A0 (0x0070 - 0x0110)
class UCameraStateBlender_X : public UComponent
{
public:
    struct FCameraTransition Transition;                    // 0x0070 (0x0050)
    [0x0000004000002000] (CPF_Transient)
    struct FCameraOrientation TransitionDelta;                // 0x00C0 (0x002C)

```

```

[0x00000000000002000] (CPF_Transient)
class UCameraState_X* CameraState; // 0x00F0 (0x0008)
[0x00000004000002000] (CPF_Transient)
struct FScriptDelegate __EventBlenderStateChanged__Delegate; // 0x00F8
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CameraStateBlender_X");
}

return uClassPointer;
};

bool IsTransitioning();
void BlendCameraState(float DeltaTime, struct FCameraOrientation& OutPOV);
void PostProcessPOV(float DeltaTime, struct FCameraOrientation& OutPOV);
void UpdatePOV(float DeltaTime, struct FCameraOrientation& OutPOV);
void Tick(float DeltaTime);
void ClearTransitionDelta();
void ClearTransition();
bool TransitionToState(class UCameraState_X* NewState);
void Snap();
void EventBlenderStateChanged(class UCameraStateBlender_X* CameraBender);
};

// Class ProjectX.CameraState_X
// 0x0028 (0x0064 - 0x008C)
class UCameraState_X : public UStateObject_X
{
public:
struct FViewTargetTransitionParams DefaultBlendParams; // 0x0068
(0x0010) [0x0000000000000001] (CPF_Edit)
class AWorldInfo* WorldInfo; // 0x0078 (0x0008)
[0x00000004000002000] (CPF_Transient)
class ACamera_X* Camera; // 0x0080 (0x0008)
[0x00000004000002000] (CPF_Transient)
unsigned long bCanSwivel : 1; // 0x0088 (0x0004)
[0x00000000000000002] [0x00000001] (CPF_Const)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CameraState_X");
}
}

```

```

return uClassPointer;
};

void OnSnap();
void ModifyPostProcessSettings(struct FPostProcessSettings& PP);
void ProcessViewRotation(float DeltaTime, struct FRotator& OutViewRotation, struct FRotator& OutDeltaRot);
void UpdatePOV(float DeltaTime, struct FCameraOrientation& OutPOV);
void Tick(float DeltaTime);
void EndCameraState();
void BeginCameraState();
struct FViewTargetTransitionParams GetEndBlendParams(class UCameraState_X* NewState);
struct FViewTargetTransitionParams GetStartBlendParams(class UCameraState_X* PreviousState);
bool ShouldKeepExecuting();
bool ShouldExecute();
void ShutDown();
void Init(class ACamera_X* InCamera);
};

// Class ProjectX.CameraUtils_X
// 0x0000 (0x0060 - 0x0060)
class UCameraUtils_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CameraUtils_X");
}

return uClassPointer;
};

static void UpdateAspectRatioFOV(float AspectRatio, struct FCameraOrientation& OutPOV);
static void BlendPOVs(struct FCameraOrientation FromPOV, struct FCameraOrientation ToPOV, float Percent, struct FCameraOrientation& OutPOV);
static float GetBlendPercent(struct FViewTargetTransitionParams BlendParams, float Time);
static bool IsPointInView(struct FVector Point, struct FVector ViewStart, struct FRotator ViewRotation, float FOV);
static float GetFrustumDistanceFromWidth(float FOV, float Width, float AspectRatio);
static float GetFrustumDistanceFromHeight(float FOV, float Height);
static void GetFrustumSize(float Dist, float FOV, float AspectRatio, float& OutWidth, float& OutHeight);
static void CalculateDistanceRotation(struct FVector Focus, struct FVector Origin, float& OutDistance, struct FRotator& OutRotation);
};

```

```

// Class ProjectX.CameraModifier_CameraShake_X
// 0x0004 (0x009C - 0x00A0)
class UCameraModifier_CameraShake_X : public UCameraModifier_CameraShake
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CameraModifier_CameraShake_X");
}

return uClassPointer;
};

void ModifyCameraShakeScale(class UCameraShake* Shake, float NewScale);
void RemoveCameraShake(class UCameraShake* Shake);
};

// Class ProjectX.CameraState_CamActor_X
// 0x0180 (0x008C - 0x020C)
class UCameraState_CamActor_X : public UCameraState_X
{
public:
unsigned long                                     bUseOverridePostProcess : 1;           // 0x0090 (0x0004)
[0x0000000000000001] [0x00000001] (CPF_Edit)
class ACameraActor*                               CamActor;                             // 0x0098 (0x0008)
[0x0000000000000200] (CPF_Transient)
struct FPostProcessSettings                       PrevPost;                             // 0x00A0 (0x0168)
[0x000000000040200] (CPF_Transient | CPF_NeedCtorLink)
float                                              PrevPostAlpha;                        // 0x0208 (0x0004)
[0x0000000000000200] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CameraState_CamActor_X");
}

return uClassPointer;
};

void UpdatePOV(float DeltaTime, struct FCameraOrientation& OutPOV);
void EndCameraState();
void BeginCameraState();
struct FViewTargetTransitionParams GetBlendParams(class UCameraState_X* PrevState);

```



```

bool ShouldExecute();
};

// Class ProjectX.CameraState_CamActorCinematic_X
// 0x0004 (0x020C - 0x0210)
class UCameraState_CamActorCinematic_X : public UCameraState_CamActor_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CameraState_CamActorCinematic_X");
}

return uClassPointer;
};

void UpdatePOV(float DeltaTime, struct FCameraOrientation& OutPOV);
void EndCameraState();
};

// Class ProjectX.CancelJoinMessage_X
// 0x0000 (0x0060 - 0x0060)
class UCancelJoinMessage_X : public UBeaconMessage_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CancelJoinMessage_X");
}

return uClassPointer;
};

};

// Class ProjectX.CDN_X
// 0x0020 (0x0060 - 0x0080)
class UCDN_X : public UObject
{
public:
class FString URL; // 0x0060 (0x0010)
[0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)

```

```

class FString                                BlogURL;                                // 0x0070 (0x0010)
[0x000000000000404000] (CPF_Config | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CDN_X");
}

return uClassPointer;
};

};

// Class ProjectX.HUD_X
// 0x0018 (0x0308 - 0x0320)
class AHUD_X : public AHUD
{
public:
class UObject*                                ShowDebugObject;                                // 0x0308 (0x0008)
[0x00000000000002000] (CPF_Transient)
class UDebugDrawer_X*                        DebugDrawer;                                // 0x0310 (0x0008)
[0x00000000000002000] (CPF_Transient)
class UTexture*                                DebugTexture;                                // 0x0318 (0x0008)
[0x00000000000002000] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.HUD_X");
}

return uClassPointer;
};

void SetShowDebugObject(class UObject* inObj);
void ShowDebugInfo(float& out_YL, float& out_YPos);
void ShowDebug(struct FName DebugType);
void DebugCategory(struct FName DebugType);
void DebugCategorySwitch(unsigned long bForward);
void DrawPauseScreen();
bool ShouldShowConsoleMessage(struct FConsoleMessage InConsoleMessage);
void DrawDebugTexture();
void DrawHUD();
};

```

```

// Class ProjectX.CheckReservation_X
// 0x00E0 (0x0060 - 0x0140)
class UCheckReservation_X : public UObject
{
public:
float          StartDelay;                // 0x0060 (0x0004)
[0x0000000000000001] (CPF_Edit)
float          Rate;                      // 0x0064 (0x0004)
[0x0000000000000001] (CPF_Edit)
float          LastMatchmakingHeartbeatTime; // 0x0068 (0x0004)
[0x0000004000002000] (CPF_Transient)
class UAsyncTask* CheckReservationTask;    // 0x0070 (0x0008)
[0x0000000000002000] (CPF_Transient)
struct FScriptDelegate FoundReservationCallback; // 0x0078
(0x0018) [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FServerReservationData FoundReservation; // 0x0090
(0x0070) [0x0000000000400000] (CPF_NeedCtorLink)
class UPsyNetConfig_X* Config;             // 0x0100 (0x0008)
[0x0000800000000000]
class UReservationBeacon_X* ReservationBeacon; // 0x0108
(0x0008) [0x0000800000408000] (CPF_ExportObject | CPF_Component | CPF_EditInline)
struct FScriptDelegate __OnStartSearch__Delegate; // 0x0110
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __OnFoundReservation__Delegate; // 0x0128
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.CheckReservation_X");
}

return uClassPointer;
};

void HandleClientReservationMessage(class UReservationConnection_X* Connection, class
UClientReservationMessage_X* Message);
void SendRequest();
void HandlePreLoadMap(class FString _);
void Cancel();
bool IsSearching();
void StartSearch(struct FScriptDelegate Callback);
class UCheckReservation_X* NotifyOnStartSearch(struct FScriptDelegate Callback);
void OnFoundReservation(struct FServerReservationData Reservation);
void OnStartSearch(class UAsyncTask* Task);
};

// Class ProjectX.RPC_ReservationHeartbeat_X
// 0x0000 (0x00E8 - 0x00E8)
class URPC_ReservationHeartbeat_X : public URPC_X

```

```

{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_ReservationHeartbeat_X");
}

return uClassPointer;
};

};

// Class ProjectX.MatchmakingMetrics_X
// 0x0020 (0x0080 - 0x00A0)
class UMatchmakingMetrics_X : public UMetricsGroup_X
{
public:
float StartTime; // 0x0080 (0x0004)
[0x0000000000000000]
struct FGuid MMGuid; // 0x0084 (0x0010)
[0x0000000000000000]
unsigned long bIsCrossplayDisabled : 1; // 0x0094 (0x0004)
[0x0000000000000000] [0x00000001]
float PartyLeaderMMR; // 0x0098 (0x0004)
[0x0000000000000000]
int32_t PartyLeaderTier; // 0x009C (0x0004)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.MatchmakingMetrics_X");
}

return uClassPointer;
};

void SendReservationRetrySuccess();
void RankedReconnect(class FString ServerName, struct FGuid MatchMakingGuid);
void FoundServer(class FString ServerName, int32_t Playlist, class FString ReservationID,
unsigned long bFromNotification, float ElapsedTime, struct FGuid MatchMakingGuid, unsigned
long bDisableCrossPlay, float MMR, int32_t Tier);
void RecordFoundServer(unsigned long bFromNotification, struct FGuid MatchMakingGuid,
unsigned long bDisableCrossPlay, float MMR, int32_t Tier, struct FServerReservationData&

```

```

Reservation);
void Cancel(float ElapsedTime, struct FGuid MatchMakingGuid);
void RecordCancel(struct FGuid MatchMakingGuid);
void ErrorID(int32_t Error, struct FGuid MatchMakingGuid);
void ErrorUnknown(class FString Message, struct FGuid MatchMakingGuid);
void RecordError(class FString Error, struct FGuid MatchMakingGuid);
void Start(TArray<class URegionPing_X*> Regions, TArray<int32_t> Playlists, unsigned long
bDisableCrossPlay, unsigned long bUseRecommendedRegions, struct FGuid MatchMakingGuid,
float LeaderMMR, int32_t LeaderTier);
};

```

```

// Class ProjectX.ClanforgeReservation_X
// 0x0068 (0x0060 - 0x00C8)
class UClanforgeReservation_X : public UObject
{
public:
class FString ReserveURLs[0x3]; // 0x0060 (0x0030)
[0x0001004000400000] (CPF_NeedCtorLink)
uint8_t ReserveState; // 0x0090 (0x0001)
[0x0001000000002000] (CPF_Transient)
TArray<float> RetryDelays; // 0x0098 (0x0010)
[0x0001000000400001] (CPF_Edit | CPF_NeedCtorLink)
int32_t SendFailures; // 0x00A8 (0x0004)
[0x0001000000002000] (CPF_Transient)
TArray<class UWebRequest_X*> QueuedRequests; // 0x00B0
(0x0010) [0x0001000000402000] (CPF_Transient | CPF_NeedCtorLink)
class UWebRequest_X* PendingRequest; // 0x00C0 (0x0008)
[0x0001000000002000] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ClanforgeReservation_X");
}

return uClassPointer;
};

void HandleRetry();
void Retry();
void HandleSendComplete(class UWebRequest_X* Request);
void ProcessNextRequest();
void Send(class FString URL);
void SetReserveState(uint8_t NewState);
void HandleActivate(class UOnlineGameDedicatedServer_X* Server);
void HandleInactive(class UOnlineGameDedicatedServer_X* Server);
void Init(class UOnlineGameDedicatedServer_X* Server, class FString ReserveURL, class FString
UnreserveURL);
};

```

```
// Class ProjectX.ClientNetMetrics_X
// 0x0055 (0x0060 - 0x00B5)
class UClientNetMetrics_X : public UObject
{
public:
float RecordPeriod; // 0x0060 (0x0004)
[0x0000000000000000]
float LastRecordTime; // 0x0064 (0x0004)
[0x0000000000000000]
struct FNetPacketStats OldStats; // 0x0068 (0x001C)
[0x0000000000000000]
TArray<float> GamePings; // 0x0088 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
class UGameServerPinger_X* Pinger; // 0x0098 (0x0008)
[0x00000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
TArray<float> PsyPings; // 0x00A0 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
int32_t PsyPacketsLost; // 0x00B0 (0x0004)
[0x0000000000000000]
uint8_t ConnectionType; // 0x00B4 (0x0001)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ClientNetMetrics_X");
}

return uClassPointer;
};

void __ClientNetMetrics_X__StartRecording_0x2(class UGameServerPinger_X* _);
void __ClientNetMetrics_X__StartRecording_0x1(class UGameServerPinger_X* _, float DelaySeconds);
struct FPingStats CalcPingStats(TArray<float>& Pings);
void Record();
void RecordTimer();
void StopRecording();
void StartRecording();
void eventConstruct();
};

// Class ProjectX.GameServerPinger_X
// 0x0048 (0x0070 - 0x00B8)
class UGameServerPinger_X : public UComponent
{
public:
class UBeaconConfig_X* Config; // 0x0070 (0x0008)
[0x0000800000000000]
struct FName Address; // 0x0078 (0x0008)
```

```

[0x0000000000000000]
class UUDPpingBeaconClient_X* PingBeacon; // 0x0080 (0x0008)
[0x00000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
struct FScriptDelegate __EventPong__Delegate; // 0x0088 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventLost__Delegate; // 0x00A0 (0x0018)
[0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GameServerPinger_X");
}

return uClassPointer;
};

void HandleLost(class UUDPpingBeaconClient_X* __, struct FName __);
void HandlePong(class UUDPpingBeaconClient_X* __, struct FName __, float DeltaSeconds);
void SendPing();
void StopPinging();
void StartPinging();
void SetAddress(class FString InAddress);
void EventLost(class UGameServerPinger_X* Pinger);
void EventPong(class UGameServerPinger_X* Pinger, float DeltaSeconds);
};

// Class ProjectX.InitialServerToClientMessage_X
// 0x0000 (0x0060 - 0x0060)
class UInitialServerToClientMessage_X : public UInterface
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.InitialServerToClientMessage_X");
}

return uClassPointer;
};

class FString GetDSRToken();
class FString GetReservationID();
};

```

```

// Class ProjectX.ClubErrors_X
// 0x0098 (0x0080 - 0x0118)
class UClubErrors_X : public UErrorList
{
public:
class UErrorType*                ClubNotFound;                // 0x0080 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                PlayerAlreadyInClub;          // 0x0088 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                PlayerInSameClub;             // 0x0090 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                PlayerInDifferentClub;         // 0x0098 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                NotClubOwner;                 // 0x00A0 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                ClubNameChangeNotAllowed;     // 0x00A8
(0x0008) [0x0001000000000002] (CPF_Const)
class UErrorType*                ClubFull;                     // 0x00B0 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                ClubMemberNotFound;           // 0x00B8 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                ClubInviteNotFound;           // 0x00C0 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                ClubInvalidParameters;        // 0x00C8 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                ClubNameInvalid;              // 0x00D0 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                ClubTagInvalid;               // 0x00D8 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                ClubTagPlusNameInvalid;       // 0x00E0 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                ClubMotdInvalid;              // 0x00E8 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                ClubNameTaken;                // 0x00F0 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                ClubOwnerCannotLeaveUnlessEmpty; // 0x00F8
(0x0008) [0x0001000000000002] (CPF_Const)
class UErrorType*                CrossPlatformClubsDisabled;    // 0x0100 (0x0008)
[0x0001000000000002] (CPF_Const)
class UErrorType*                ClubInviteCrossPlatformClubsDisabled; // 0x0108
(0x0008) [0x0001000000000002] (CPF_Const)
class UErrorType*                ClubBanned;                   // 0x0110 (0x0008)
[0x0001000000000002] (CPF_Const)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ClubErrors_X");
}
}

```



```

return uClassPointer;
};

};

// Class ProjectX.ClubServerResult_X
// 0x0048 (0x0060 - 0x00A8)
class UClubServerResult_X : public UObject
{
public:
class FString                                Host;                                // 0x0060 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
int32_t                                     Port;                                     // 0x0070 (0x0004)
[0x0001000000000000]
class FString                                ServerName;                                // 0x0078 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
class FString                                CustomServerName;                                // 0x0088 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
class FString                                CustomServerPassword;                                // 0x0098 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ClubServerResult_X");
}

return uClassPointer;
};

class FString GetAddress();
};

// Class ProjectX.ServerToServerMessage_X
// 0x0010 (0x0060 - 0x0070)
class UServerToServerMessage_X : public UObject
{
public:
class FString                                ServerId;                                // 0x0060 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ServerToServerMessage_X");
}
}

```

```

return uClassPointer;
};

};

// Class ProjectX.ConnectionInfoMessage_X
// 0x0030 (0x0070 - 0x00A0)
class UConnectionInfoMessage_X : public UServerToServerMessage_X
{
public:
    struct FServerConnectionInfo          ConnectionInfo;          // 0x0070 (0x0030)
    [0x0000000400040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.ConnectionInfoMessage_X");
        }

        return uClassPointer;
    };

    static bool IsConnectionInfoValid(struct FServerConnectionInfo& ServerInfo);
};

// Class ProjectX.DebugDrawer_X
// 0x0040 (0x00C0 - 0x0100)
class UDebugDrawer_X : public UDebugDrawer
{
public:
    float          TextScale;          // 0x00C0 (0x0004)
    [0x0000000000000001] (CPF_Edit)
    class UCanvas* Canvas;          // 0x00C8 (0x0008)
    [0x0000000000000200] (CPF_Transient)
    TArray<struct FName>          DebugCategories;          // 0x00D0 (0x0010)
    [0x000000000040200] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FName>          PossibleDebugCategories;          // 0x00E0
    (0x0010) [0x000000000040200] (CPF_Transient | CPF_NeedCtorLink)
    float          CharWidth;          // 0x00F0 (0x0004)
    [0x0000000000000200] (CPF_Transient)
    float          LastWidestElement;          // 0x00F4 (0x0004)
    [0x0000000000000200] (CPF_Transient)
    unsigned long          bHideDebug : 1;          // 0x00F8 (0x0004)
    [0x0000000000000200] [0x00000001] (CPF_Transient)
    float          OffsetX;          // 0x00FC (0x0004)
    [0x0000000000000200] (CPF_Transient)

public:
    static UClass* StaticClass()

```

```

{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.DebugDrawer_X");
}

return uClassPointer;
};

void DrawHeader();
void DrawSpacer(float Height);
void DrawBackground(float Width, float Height);
void FinishDrawDebug();
void StartDrawDebug(class UCanvas* C);
void PrintText(class FString Text, struct FColor InColor);
void PrintProperty(class FString PropertyName, class FString Value);
void PrintSeperator();
void UpdateCanvasIndentation();
void EndSection();
void StartSection();
void SetPos(float X, float Y);
bool ShouldDisplayDebug(struct FName Category);
};

// Class ProjectX.DecodeObject_X
// 0x0010 (0x0060 - 0x0070)
class UDecodeObject_X : public UObject
{
public:
int32_t Checksum; // 0x0060 (0x0004)
[0x0000000800000000]
class UError* Error; // 0x0068 (0x0008)
[0x0000000800000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.DecodeObject_X");
}

return uClassPointer;
};

void Decode(class UObject* OutObj);
};

// Class ProjectX.DecodeObjectJson_X
// 0x0018 (0x0070 - 0x0088)

```

```

class UDecodeObjectJson_X : public UDecodeObject_X
{
public:
class UJSONSerializer_X*           JsonSerializer;           // 0x0070 (0x0008)
[0x0000000400000000]
class FString                       Stream;                   // 0x0078 (0x0010)
[0x0000000400040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.DecodeObjectJson_X");
}

return uClassPointer;
};

void Decode(class UObject* OutObj);
class UDecodeObjectJson_X* SetStream(class FString& InStream);
};

// Class ProjectX.DecodeObjectTypes_X
// 0x0000 (0x0060 - 0x0060)
class UDecodeObjectTypes_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.DecodeObjectTypes_X");
}

return uClassPointer;
};

};

// Class ProjectX.DecodeObjectUObject_X
// 0x0018 (0x0070 - 0x0088)
class UDecodeObjectUObject_X : public UDecodeObject_X
{
public:
class UObjectSerializer_X*         ObjectSerializer;         // 0x0070 (0x0008)
[0x0000000400000000]
TArray<uint8_t>                    Stream;                    // 0x0078 (0x0010)

```

[0x0000004000400000] (CPF\_NeedCtorLink)

```
public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.DecodeObjectUObject_X");
}

return uClassPointer;
};

void Decode(class UObject* OutObj);
class UDecodeObjectUObject_X* SetStream(int32_t RequestChecksum, class FString&
InStream);
};
```

```
// Class ProjectX.DownloadedPlaylistsData_X
// 0x0010 (0x0060 - 0x0070)
class UDownloadedPlaylistsData_X : public UObject
{
public:
TArray<class UGameSettingPlaylist_X*>          Playlists;                // 0x0060 (0x0010)
[0x0000000000040200] (CPF_Transient | CPF_NeedCtorLink)
```

```
public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.DownloadedPlaylistsData_X");
}

return uClassPointer;
};

};
```

```
// Class ProjectX.DSPendingMessage_X
// 0x000C (0x0060 - 0x006C)
class UDSPendingMessage_X : public UObject
{
public:
class UDSR_DSMessage_X*          Message;                // 0x0060 (0x0008)
[0x0000000000000000]
float          TimeoutTime;                // 0x0068 (0x0004)
[0x0000000000000000]

public:
```

```

static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.DSPendingMessage_X");
}

return uClassPointer;
};

};

// Class ProjectX.DSR_DSMessage_X
// 0x0040 (0x0090 - 0x00D0)
class UDSR_DSMessage_X : public UPsyNetClientService_X
{
public:
class FString                                PlayerID;                                // 0x0090 (0x0010)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                                ReservationID;                                // 0x00A0 (0x0010)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                                MessageType;                                // 0x00B0 (0x0010)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                                MessagePayload;                                // 0x00C0 (0x0010)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.DSR_DSMessage_X");
}

return uClassPointer;
};

};

// Class ProjectX.DSR_ClientMessage_X
// 0x0030 (0x0090 - 0x00C0)
class UDSR_ClientMessage_X : public UPsyNetClientService_X
{
public:
class FString                                ReservationID;                                // 0x0090 (0x0010)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                                MessageType;                                // 0x00A0 (0x0010)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                                MessagePayload;                                // 0x00B0 (0x0010)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.DSR_ClientMessage_X");
}

return uClassPointer;
};

};

// Class ProjectX.DSR_DSToDSMessage_X
// 0x0020 (0x0090 - 0x00B0)
class UDSR_DSToDSMessage_X : public UPsyNetClientService_X
{
public:
class FString                MessageType;                // 0x0090 (0x0010)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                MessagePayload;                // 0x00A0 (0x0010)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.DSR_DSToDSMessage_X");
}

return uClassPointer;
};

};

// Class ProjectX.DynamicValue_X
// 0x0024 (0x0060 - 0x0084)
class UDynamicValue_X : public UObject
{
public:
float                DefaultValue;                // 0x0060 (0x0004)
[0x00000000000000001] (CPF_Edit)
unsigned long                bClampMax : 1;                // 0x0064 (0x0004)
[0x00000000000000001] [0x000000001] (CPF_Edit)
unsigned long                bClampMin : 1;                // 0x0064 (0x0004)
[0x00000000000000001] [0x000000002] (CPF_Edit)
float                MaxValue;                // 0x0068 (0x0004)
[0x00000000000000001] (CPF_Edit)

```

```

float                               MinValue;                               // 0x006C (0x0004)
[0x000000000000000001] (CPF_Edit)
TArray<class UDynamicValueModifier_X*> Modifiers;                          // 0x0070
(0x0010) [0x00000004000400000] (CPF_NeedCtorLink)
float                               CachedValue;                          // 0x0080 (0x0004)
[0x000000080000000000]

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.DynamicValue_X");
}

return uClassPointer;
};

```

```

void PrintDebugInfo(class UDebugDrawer* Drawer);
class FString GetDebugValue();
float GetValue();
void Tick(float DeltaTime);
void RemoveAllModifiers();
void RemoveModifier(class UDynamicValueModifier_X* Mod);
class UDynamicValueModifier_X* AddModifier(class UDynamicValueModifier_X* Mod);
void eventConstruct();
};

```

```

// Class ProjectX.DynamicValueModifier_X
// 0x0018 (0x0060 - 0x0078)
class UDynamicValueModifier_X : public UObject
{
public:
class FString                               DisplayName;                               // 0x0060 (0x0010)
[0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)
uint8_t                                     Type;                                     // 0x0070 (0x0001)
[0x000000000000000001] (CPF_Edit)
int32_t                                     Priority;                                   // 0x0074 (0x0004)
[0x000000000000000001] (CPF_Edit)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.DynamicValueModifier_X");
}

return uClassPointer;
};

```



```

class FString GetDebugValue();
class FString GetDebugString();
bool Expired();
float GetValue();
void Tick(float DeltaTime);
};

// Class ProjectX.DynamicValueModifierCurve_X
// 0x0030 (0x0078 - 0x00A8)
class UDynamicValueModifierCurve_X : public UDynamicValueModifier_X
{
public:
    struct FInterpCurveFloat          Curve;                                // 0x0078 (0x0018)
    [0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)
    struct FScriptDelegate            __GetValueDelegate__Delegate;        // 0x0090
    (0x0018) [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.DynamicValueModifierCurve_X");
        }

        return uClassPointer;
    };

    class FString GetDebugValue();
    float GetValue();
    float GetValueDelegate();
};

// Class ProjectX.DynamicValueModifierDuration_X
// 0x0024 (0x0078 - 0x009C)
class UDynamicValueModifierDuration_X : public UDynamicValueModifier_X
{
public:
    float          Duration;                                // 0x0078 (0x0004)
    [0x000000000000000001] (CPF_Edit)
    float          Value;                                    // 0x007C (0x0004)
    [0x000000000000000001] (CPF_Edit)
    struct FInterpCurveFloat          CurveValue;                                // 0x0080 (0x0018)
    [0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)
    unsigned long          bUseCurve : 1;                                // 0x0098 (0x0004)
    [0x000000000000000001] [0x00000001] (CPF_Edit)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

```

```
if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.DynamicValueModifierDuration_X");
}
```

```
return uClassPointer;
};
```

```
class FString GetDebugString();
float GetValue();
bool Expired();
void Tick(float DeltaTime);
};
```

```
// Class ProjectX.EnableOnlineSave_X
// 0x0000 (0x0060 - 0x0060)
class UEnableOnlineSave_X : public UObject
{
public:
```

```
public:
static UClass* StaticClass()
{
    static UClass* uClassPointer = nullptr;
```

```
if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.EnableOnlineSave_X");
}
```

```
return uClassPointer;
};
```

```
};
```

```
// Class ProjectX.EncodeObjectFactory_X
// 0x0010 (0x0060 - 0x0070)
class UEncodeObjectFactory_X : public UObject
{
public:
class UJSONSerializer_X*                JsonSerializer;                // 0x0060 (0x0008)
[0x00000004000000001] (CPF_Edit)
class UObjectSerializer_X*              ObjectSerializer;              // 0x0068 (0x0008)
[0x00000004000000001] (CPF_Edit)
```

```
public:
static UClass* StaticClass()
{
    static UClass* uClassPointer = nullptr;
```

```
if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.EncodeObjectFactory_X");
```

```

}

return uClassPointer;
};

static class UDecodeObject_X* DecodeObject(uint8_t Encoding, int32_t RequestChecksum, class
FString& Stream);
static class UEncodeObject_X* EncodeObject(uint8_t Encoding, class UObject* Target);
};

// Class ProjectX.EncodeObjectUObject_X
// 0x0008 (0x0078 - 0x0080)
class UEncodeObjectUObject_X : public UEncodeObject_X
{
public:
class UObjectSerializer_X*          ObjectSerializer;          // 0x0078 (0x0008)
[0x0000000400000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EncodeObjectUObject_X");
}

return uClassPointer;
};

class UEncodeObject_X* Encode(class UObject* inObj);
};

// Class ProjectX.EncodeObjectJson_X
// 0x0008 (0x0078 - 0x0080)
class UEncodeObjectJson_X : public UEncodeObject_X
{
public:
class UJSONSerializer_X*          JsonSerializer;          // 0x0078 (0x0008)
[0x0000000400000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EncodeObjectJson_X");
}

return uClassPointer;
};

```

```

class UEncodeObject_X* Encode(class UObject* inObj);
};

// Class ProjectX.EncodeObjectTypes_X
// 0x0000 (0x0060 - 0x0060)
class UEncodeObjectTypes_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EncodeObjectTypes_X");
}

return uClassPointer;
};

};

// Class ProjectX.PreExitEvent_X
// 0x0000 (0x0060 - 0x0060)
class UPreExitEvent_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PreExitEvent_X");
}

return uClassPointer;
};

};

// Class ProjectX.EOSEvent_Presence_X
// 0x0058 (0x0068 - 0x00C0)
class UEOSEvent_Presence_X : public UEOSMetricEvent_X
{
public:
class FString Location; // 0x0068 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString MapName; // 0x0078 (0x0010)

```

```

[0x000000000000400000] (CPF_NeedCtorLink)
class FString                                StreamingService;                // 0x0088 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
int32_t                                     PlaylistId;                        // 0x0098 (0x0004)
[0x000000000000000000]
struct FGuid                                LevelSessionID;                    // 0x009C (0x0010)
[0x000000000000000000]
class FString                                ServerRegion;                      // 0x00B0 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EOSEvent_Presence_X");
}

return uClassPointer;
};

};

// Class ProjectX.EOSMetricsConfig_X
// 0x0014 (0x0078 - 0x008C)
class UEOSMetricsConfig_X : public UOnlineConfig_X
{
public:
class FString                                DataRouterURL;                    // 0x0078 (0x0010)
[0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)
float                                         PingDelayTime;                    // 0x0088 (0x0004)
[0x000000000000000001] (CPF_Edit)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.EOSMetricsConfig_X");
}

return uClassPointer;
};

};

// Class ProjectX.LocalizedAccountLinkURL
// 0x0020 (0x0060 - 0x0080)
class ULocalizedAccountLinkURL : public UObject
{

```

```

public:
class FString                                     Language;                                // 0x0060 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString                                     URL;                                // 0x0070 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.LocalizedAccountLinkURL");
}

return uClassPointer;
};

};

// Class ProjectX.Explosion_X
// 0x00230 (0x0268 - 0x0288)
class AExplosion_X : public AActor
{
public:
class UExplosionComponent_X*                     ExplosionComponent;                                // 0x0268
(0x0008) [0x0000000004080009] (CPF_Edit | CPF_ExportObject | CPF_Component |
CPF_EditInline)
class AFXActor_X*                               FXActorArchetype;                                // 0x0270 (0x0008)
[0x0000000100000021] (CPF_Edit | CPF_Net)
class AFXActor_X*                               FXActor;                                // 0x0278 (0x0008)
[0x0000000000000200] (CPF_Transient)
class APRI_X*                                   Scorer;                                // 0x0280 (0x0008)
[0x0000000000000200] (CPF_Transient)

TArray<class UFXActorEvent_X*>                  PostSpawnEvents;                                // 0x0288
(0x0010) [0x000000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Explosion_X");
}

return uClassPointer;
};

void eventDestroyed();

```

```

void eventForceNetRelevant();
void SetFXActorArchetype(class AFXActor_X* A);
void SpawnFX();
void SetExplosionHandler(TArray<class UExplosionHitHandler_X*> InExplosionHandlers, struct
FBox GoalBox, class UActorComponent_X* ExplosionGoal);
void eventPostBeginPlay();
void eventReplicatedEvent(struct FName VarName);
};

```

```

// Class ProjectX.RPC_JoinMatch_X
// 0x0028 (0x00E8 - 0x0110)
class URPC_JoinMatch_X : public URPC_X
{
public:
    struct FName                                JoinType;                                // 0x00E8 (0x0008)
    [0x0000000000000000]
    class FString                                ServerName;                                // 0x00F0 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)
    class FString                                Password;                                // 0x0100 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_JoinMatch_X");
}

```

```

return uClassPointer;
};

```

```

};

```

```

// Class ProjectX.SeqAct_TriggerFXActor_X
// 0x0010 (0x0160 - 0x0170)
class USeqAct_TriggerFXActor_X : public USequenceAction
{
public:
    class UFXActorEvent_X*                        Event;                                // 0x0160 (0x0008)
    [0x0000000000000001] (CPF_Edit)
    class AActor*                                AttachTo;                                // 0x0168 (0x0008)
    [0x0000000000000001] (CPF_Edit)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.SeqAct_TriggerFXActor_X");

```

```

}

return uClassPointer;
};

};

// Class ProjectX.IOnlineGameHost_X
// 0x0000 (0x0060 - 0x0060)
class UOnlineGameHost_X : public UInterface
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.IOnlineGameHost_X");
}

return uClassPointer;
};

struct FUniqueNetId GetCustomMatchOwner();
struct FCustomMatchSettings GetCustomMatchSettings();
bool AllowSplitscreenJoin(struct FUniqueNetId PrimaryPlayerId, struct FUniqueNetId PlayerID,
class FString PlayerName, class FString& Error);
void PlayerLoggedOut(class APlayerReplicationInfo* PRI);
void PlayerLoggedIn(class APlayerReplicationInfo* PRI);
void AllowPlayerLogin(class FString Options, struct FUniqueNetId PlayerID, class FString&
ErrorMessage);
};

// Class ProjectX.GameInfo_MapProfiler_X
// 0x0000 (0x04C8 - 0x04C8)
class AGameInfo_MapProfiler_X : public AGameInfo_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GameInfo_MapProfiler_X");
}

return uClassPointer;
};

```



```

void GenericPlayerInitialization(class AController* C);
class APlayerController* SpawnPlayerController(struct FVector SpawnLocation, struct FRotator
SpawnRotation);
};

// Class ProjectX.GameSettingConfig_X
// 0x0010 (0x0078 - 0x0088)
class UGameSettingConfig_X : public UOnlineConfig_X
{
public:
TArray<struct FGameSettingHidingOverride>      HidingOverrides;          // 0x0078
(0x0010) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GameSettingConfig_X");
}

return uClassPointer;
};

};

// Class ProjectX.PresetMutators_X
// 0x0038 (0x0060 - 0x0098)
class UPresetMutators_X : public UObject
{
public:
TArray<struct FCategorySettingPair>      PresetTags;          // 0x0060 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
TArray<struct FName>      RequiresMaps;          // 0x0070 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
unsigned long      bLockSettings : 1;          // 0x0080 (0x0004)
[0x0000000000000000] [0x00000001]
unsigned long      bHideSettings : 1;          // 0x0080 (0x0004)
[0x0000000000000000] [0x00000002]
class FString      OverrideName;          // 0x0088 (0x0010)
[0x0000000000040200] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PresetMutators_X");
}
}

```

```
return uClassPointer;  
};
```

```
};
```

```
// Class ProjectX.PlaylistSettings_X  
// 0x00F8 (0x0060 - 0x0158)  
class UPlaylistSettings_X : public UObject  
{  
public:  
    class FString Title; // 0x0060 (0x0010)  
    [0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)  
    class FString Description; // 0x0070 (0x0010)  
    [0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)  
    class FString BadgeTitle; // 0x0080 (0x0010)  
    [0x000000000000400001] (CPF_Edit | CPF_NeedCtorLink)  
    int32_t PlaylistId; // 0x0090 (0x0004)  
    [0x000000000000000001] (CPF_Edit)  
    int32_t PlayerCount; // 0x0094 (0x0004)  
    [0x000000000000000001] (CPF_Edit)  
    unsigned long bStandard : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00000001] (CPF_Edit)  
    unsigned long bRanked : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00000002] (CPF_Edit)  
    unsigned long bCheckRankedMatchReservationID : 1; // 0x0098  
    (0x0004) [0x000000000000000001] [0x00000004] (CPF_Edit)  
    unsigned long bSolo : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00000008] (CPF_Edit)  
    unsigned long bHidden : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00000010] (CPF_Edit)  
    unsigned long bExtraMode : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00000020] (CPF_Edit)  
    unsigned long bPrivate : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00000040] (CPF_Edit)  
    unsigned long bTournament : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00000080] (CPF_Edit)  
    unsigned long bApplyQuitPenalty : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00000100] (CPF_Edit)  
    unsigned long bAllowForfeit : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00000200] (CPF_Edit)  
    unsigned long bDisableRankedReconnect : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00000400] (CPF_Edit)  
    unsigned long bIgnoreAssignTeams : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00000800] (CPF_Edit)  
    unsigned long bKickOnMigrate : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00001000] (CPF_Edit)  
    unsigned long bAllowBotFills : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00002000] (CPF_Edit)  
    unsigned long bServerBroadcastCancellations : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00004000] (CPF_Edit)  
    unsigned long bSkipGameModeVerification : 1; // 0x0098 (0x0004)  
    [0x000000000000000001] [0x00008000] (CPF_Edit)  
    unsigned long bNoBackFill : 1; // 0x0098 (0x0004)
```

```

[0x0000000000000001] [0x00010000] (CPF_Edit)
unsigned long          bIsMicroEventPlaylist : 1;          // 0x0098 (0x0004)
[0x0000000000000001] [0x00020000] (CPF_Edit)
unsigned long          bHasVariablePlayerCount : 1;        // 0x0098 (0x0004)
[0x0000000000000001] [0x00040000] (CPF_Edit)
unsigned long          bNew : 1;                            // 0x0098 (0x0004)
[0x0000000000000001] [0x00080000] (CPF_Edit)
unsigned long          bAllowClubs : 1;                    // 0x0098 (0x0004)
[0x0001000000000001] [0x00100000] (CPF_Edit)
unsigned long          bDisableSaveReplays : 1;            // 0x0098 (0x0004)
[0x0000000000000001] [0x00200000] (CPF_Edit)
unsigned long          bOpenDetailsOnFirstTimeClicked : 1; // 0x0098
(0x0004) [0x0000000000000001] [0x00400000] (CPF_Edit)
unsigned long          bAllowStayAsParty : 1;              // 0x0098 (0x0004)
[0x0000000000000001] [0x00800000] (CPF_Edit)
class FString          PlaylistImageURL;                   // 0x00A0 (0x0010)
[0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
class FString          PlaylistImageTexture;               // 0x00B0 (0x0010)
[0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
class FString          PlaylistIconActiveURL;               // 0x00C0 (0x0010)
[0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
class FString          PlaylistIconInactiveURL;             // 0x00D0 (0x0010)
[0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
class FString          PlaylistNodeThumbnailURL;            // 0x00E0 (0x0010)
[0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
class FString          PlaylistNodeDefaultThumbnailPackage; // 0x00F0
(0x0010) [0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
class FString          SecondaryTitleOverride;              // 0x0100 (0x0010)
[0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
class UTimeWindow*     PlaylistTimeWindow;                  // 0x0110 (0x0008)
[0x0000000000000001] (CPF_Edit)
TArray<class UPresetMutators_X*> PresetMutators;            // 0x0118
(0x0010) [0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
struct FName           MapName;                             // 0x0128 (0x0008)
[0x0000000000000001] (CPF_Edit)
class FString          ServerCommand;                       // 0x0130 (0x0010)
[0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
struct FName           MapSetName;                           // 0x0140 (0x0008)
[0x0000000000000001] (CPF_Edit)
TArray<int32_t>         PopulationBuckets;                   // 0x0148 (0x0010)
[0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class ProjectX.PlaylistSettings_X");
}

return UClassPointer;
};

```

```

};

// Class ProjectX.OnlineGameDedicatedServerRegistration_X
// 0x002C (0x00B0 - 0x00DC)
class UOnlineGameDedicatedServerRegistration_X : public UOnline_X
{
public:
class UServerConfig_X*           MyConfig;           // 0x00B0 (0x0008)
[0x000080000000000001] (CPF_Edit)
class UCacheTimer_X*           HeartbeatTimer;       // 0x00B8 (0x0008)
[0x00000000004080009] (CPF_Edit | CPF_ExportObject | CPF_Component | CPF_EditInline)
int32_t                         GameTimeTimer;       // 0x00C0 (0x0004)
[0x000000000000000001] (CPF_Edit)
int32_t                         GameTimeSeconds;     // 0x00C4 (0x0004)
[0x00000004000002000] (CPF_Transient)
class URPC_CreateGameServer_X* CreateGameServerRPC; // 0x00C8
(0x0008) [0x00000000000002000] (CPF_Transient)
class URPC_UpdateGameServer_X* UpdateGameServerRPC; // 0x00D0
(0x0008) [0x00000000000002000] (CPF_Transient)
float                           PsyNetDisconnectShutdownTime; // 0x00D8 (0x0004)
[0x000000000000000003] (CPF_Edit | CPF_Const)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.OnlineGameDedicatedServerRegistration_X");
}

return uClassPointer;
};

void __OnlineGameDedicatedServerRegistration_X__OnInit_0x1(class FString _);
void __OnlineGameDedicatedServerRegistration_X__HandleCreateServerSucces_0x1(class
UCacheTimer_X* Timer);
void HandleCrashed();
class UServerMetrics_X* GetServerMetrics();
class FString GetExclusivePlatformString(uint8_t Platform);
void UpdateGameTime(int32_t TimeSeconds);
bool ShouldShutdownWhenEmpty();
void TimerShutdownWhenEmpty();
void HandleUpdateServerFailed(class URPC_UpdateGameServer_X* RPC);
void HandleUpdateServerSucces(class URPC_UpdateGameServer_X* RPC);
void HandleCreateServerFailed(class URPC_CreateGameServer_X* RPC);
void HandlePerConDisconnect(class UPsyNetConnection_X* _);
void HandlePerConConnect(class UPsyNetConnection_X* Connection);
void ConnectToPerCon();
void HandleCreateServerSucces(class URPC_CreateGameServer_X* RPC);
void SetServerNotJoinable();

```

```

bool HasBackfillPolicy();
void GetBackfillAmount(int32_t& BackfillTeam1, int32_t& BackfillTeam2);
void SendUpdateServerRPC();
class FString GetServerType();
void SendCreateServerRPC();
void SendUpdateRequest();
void UpdateServer();
void ForceUpdateServer();
void HandleConfigUpdate();
void UnregisterServer();
void RegisterServer();
void OnInit();
};

// Class ProjectX.GetPlayerStorageResult_X
// 0x0010 (0x0060 - 0x0070)
class UGetPlayerStorageResult_X : public UObject
{
public:
    TArray<class UGetPlayerStorageResultItem_X*>    Items; // 0x0060
    (0x0010) [0x00010000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.GetPlayerStorageResult_X");
        }

        return uClassPointer;
    };
};

// Class ProjectX.GetPlayerStorageResultItem_X
// 0x0029 (0x0060 - 0x0089)
class UGetPlayerStorageResultItem_X : public UObject
{
public:
    struct FName                Category; // 0x0060 (0x0008)
    [0x00010000000000000]
    int32_t                    Tick; // 0x0068 (0x0004)
    [0x00010000000000000]
    class FString                Data; // 0x0070 (0x0010)
    [0x00010000000400000] (CPF_NeedCtorLink)
    int32_t                    Checksum; // 0x0080 (0x0004)
    [0x00010000000000000]
    unsigned long                bChecksumMatch : 1; // 0x0084 (0x0004)
    [0x00010000000000000] [0x000000001]
    uint8_t                    Encoding; // 0x0088 (0x0001)
    [0x00010000000000000]

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GetPlayerStorageResultItem_X");
}

return uClassPointer;
};

};

// Class ProjectX.GFxModal_X
// 0x0068 (0x0060 - 0x00C8)
class UGFxModal_X : public UObject
{
public:
class FString                                ActionScriptModalName;                // 0x0060 (0x0010)
[0x0000000000040003] (CPF_Edit | CPF_Const | CPF_NeedCtorLink)
class UGFxObject*                           GFxPopup;                                // 0x0070 (0x0008)
[0x0000000000000000]
TArray<struct FScriptDelegate>               Callbacks;                                // 0x0078 (0x0010)
[0x0000000000040200] (CPF_Transient | CPF_NeedCtorLink)
class FString                                ConfirmString;                            // 0x0088 (0x0010)
[0x0000000000040803] (CPF_Edit | CPF_Const | CPF_Localized | CPF_NeedCtorLink)
struct FScriptDelegate                      __ClickDelegate__Delegate;              // 0x0098 (0x0018)
[0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventClosed__Delegate;                // 0x00B0 (0x0018)
[0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.GFxModal_X");
}

return uClassPointer;
};

class UGFxModal_X* NotifyOnClosed(struct FScriptDelegate Callback);
void Close();
void FlashAddButton(class FString Label);
void FlashAddCancelButton(class FString LocalizedText);
void HandleButtonClicked(int32_t Index);
class UGFxModal_X* AddButtonCallback(int32_t Index, struct FScriptDelegate OnClick);
class UGFxModal_X* AddButton(class FString Label, struct FScriptDelegate OnClick);

```

```

class UGfxModal_X* AddCancelButton(class FString LocalizedText, struct FScriptDelegate
OnClick);
class UGfxModal_X* SetCancellable(unsigned long bCancellable);
class UGfxModal_X* SetDefaultSelectedButtonIndex(int32_t buttonIndex);
class UGfxModal_X* SetIcon(class FString IconName);
class UGfxModal_X* SetBody(class FString LocalizedText);
class UGfxModal_X* SetTitle(class FString LocalizedText);
void SetGfxObject(class UGfxObject* Obj);
void EventClosed(class UGfxModal_X* Modal);
void ClickDelegate(class UGfxModal_X* Modal);
};

```

```

// Class ProjectX.ICabinedOrGuest
// 0x0000 (0x0060 - 0x0060)
class UICabinedOrGuest : public UInterface
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ICabinedOrGuest");
}

return uClassPointer;
};

bool IsInCabinedModeOrGuest();
};

```

```

// Class ProjectX.IOnlineGamePlaylists_X
// 0x0000 (0x0060 - 0x0060)
class UOnlineGamePlaylists_X : public UInterface
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.IOnlineGamePlaylists_X");
}

return uClassPointer;
};

bool IsNonStandardPlaylistName(struct FName PlaylistName);

```

```

bool IsStandardPlaylistName(struct FName PlaylistName);
bool IsUnrankedPlaylistName(struct FName PlaylistName);
bool IsRankedPlaylistName(struct FName PlaylistName);
struct FName IdToName(int32_t PlaylistId);
int32_t NameToId(struct FName PlaylistName);
TArray<struct FName> GetAccessiblePlaylists(TArray<struct FName>& SelectedPlaylists);
};

```

```

// Class ProjectX.LanMessage_X
// 0x0008 (0x0060 - 0x0068)
class ULanMessage_X : public UOnlineMessage_X
{
public:
uint64_t Nonce; // 0x0060 (0x0008)
[0x0000000400000000]

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.LanMessage_X");
}

```

```

return uClassPointer;
};

```

```

bool Broadcast();
class ULanMessage_X* SetNonce(uint64_t InNonce);
};

```

```

// Class ProjectX.LanMessage_HostQuery_X
// 0x0098 (0x0068 - 0x0100)
class ULanMessage_HostQuery_X : public ULanMessage_X
{
public:
struct FCustomMatchSettings Filter; // 0x0068 (0x0090)
[0x0000000000040000] (CPF_NeedCtorLink)
int32_t BuildID; // 0x00F8 (0x0004)
[0x0000000000000000]
unsigned long bHost : 1; // 0x00FC (0x0004)
[0x0000000000000000] [0x00000001]

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.LanMessage_HostQuery_X");
}

```



```

return uClassPointer;
};

class ULanMessage_HostQuery_X* SetHost(unsigned long bValue);
class ULanMessage_HostQuery_X* SetBuildID(int32_t InBuildID);
class ULanMessage_HostQuery_X* SetFilter(struct FCustomMatchSettings InFilter);
};

// Class ProjectX.LanMessage_HostResponse_X
// 0x00D0 (0x0068 - 0x0138)
class ULanMessage_HostResponse_X : public ULanMessage_X
{
public:
    struct FServerResult          Result;                // 0x0068 (0x00B0)
    [0x0000000000040000] (CPF_NeedCtorLink)
    class FString                 ServerId;              // 0x0118 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)
    class FString                 MetaData;              // 0x0128 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.LanMessage_HostResponse_X");
        }

        return uClassPointer;
    };

    class FString GetDebugString();
    class ULanMessage_HostResponse_X* SetMetaData(class FString InMetaData);
    class ULanMessage_HostResponse_X* SetServerID(class FString InServerID);
    class ULanMessage_HostResponse_X* SetResult(struct FServerResult InResult);
};

// Class ProjectX.LanMessage_Ping_X
// 0x0000 (0x0068 - 0x0068)
class ULanMessage_Ping_X : public ULanMessage_X
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.LanMessage_Ping_X");
        }
    }
};

```

```

}

return uClassPointer;
};

};

// Class ProjectX.LocalCacheTests_X
// 0x0010 (0x0060 - 0x0070)
class ULocalCacheTests_X : public UObject
{
public:
class FString                                CacheFilePath;                // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.LocalCacheTests_X");
}

return uClassPointer;
};

static void HandleLocalCacheImported(class ULocalCache_X* Cache, class UObject*
CacheObject, class UError* Error);
static void ImportTest();
static void HandleLocalCacheExported(class ULocalCache_X* Cache, class UObject*
CacheObject, class UError* Error);
static void ToCacheAndBack(int32_t RandomSeed);
};

// Class ProjectX.LoginResponse_X
// 0x0014 (0x0060 - 0x0074)
class ULoginResponse_X : public UObject
{
public:
class FString                                BannedMessage;                // 0x0060 (0x0010)
[0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)
int32_t                                       BannedMinutes;                // 0x0070 (0x0004)
[0x00000000000002000] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.LoginResponse_X");
}

```

```

}

return uClassPointer;
};

};

// Class ProjectX.MapPrefsConfig_X
// 0x000C (0x0078 - 0x0084)
class UMapPrefsConfig_X : public UOnlineConfig_X
{
public:
float                                     PreferenceWeight;                // 0x0078 (0x0004)
[0x00000000000000001] (CPF_Edit)
int32_t                                  MaxLikes;                // 0x007C (0x0004)
[0x00000000000000001] (CPF_Edit)
int32_t                                  MaxDislikes;            // 0x0080 (0x0004)
[0x00000000000000001] (CPF_Edit)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.MapPrefsConfig_X");
}

return uClassPointer;
};

};

// Class ProjectX.MatchData_X
// 0x00E8 (0x0060 - 0x0148)
class UMatchData_X : public UObject
{
public:
class FString                             MatchGuid;                // 0x0060 (0x0010)
[0x00000000000400000] (CPF_NeedCtorLink)
uint64_t                                  RecordStartTimestamp;        // 0x0070 (0x0008)
[0x00000000000000000]
uint64_t                                  RecordEndTimestamp;        // 0x0078 (0x0008)
[0x00000000000000000]
uint64_t                                  MatchStartTimestamp;        // 0x0080 (0x0008)
[0x00000000000000000]
uint64_t                                  MatchEndTimestamp;        // 0x0088 (0x0008)
[0x00000000000000000]
struct FName                             MapName;                // 0x0090 (0x0008)
[0x00000000040000000] (CPF_EditInlineNotify)
int32_t                                  Playlist;                // 0x0098 (0x0004)
[0x00000000040000000] (CPF_EditInlineNotify)
TArray<struct FName>                     Mutators;                // 0x00A0 (0x0010)

```

```

[0x000000000000400000] (CPF_NeedCtorLink)
unsigned long          blsBotMatch : 1;                // 0x00B0 (0x0004)
[0x0000000000000000] [0x00000001]
unsigned long          bClubVsClub : 1;                // 0x00B0 (0x0004)
[0x0001000000000000] [0x00000002]
unsigned long          bOverTime : 1;                  // 0x00B0 (0x0004)
[0x0000000004000000] [0x00000004] (CPF_EditInlineNotify)
unsigned long          bNoContest : 1;                 // 0x00B0 (0x0004)
[0x0000000004000000] [0x00000008] (CPF_EditInlineNotify)
unsigned long          bForfeit : 1;                   // 0x00B0 (0x0004)
[0x0000000004000000] [0x00000010] (CPF_EditInlineNotify)
int32_t                ClubID;                         // 0x00B4 (0x0004)
[0x0001000000000000]
struct FUniqueNetId     CustomMatchCreatorPlayerID;    // 0x00B8
(0x0048) [0x0000000000040000] (CPF_NeedCtorLink)
class FString           CustomServerName;              // 0x0100 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString           CustomServerPassword;          // 0x0110 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
float                   SecondsPlayed;                  // 0x0120 (0x0004)
[0x0000000004000000] (CPF_EditInlineNotify)
float                   OvertimeSecondsPlayed;          // 0x0124 (0x0004)
[0x0000000004000000] (CPF_EditInlineNotify)
int32_t                 WinningTeam;                   // 0x0128 (0x0004)
[0x0000000004000000] (CPF_EditInlineNotify)
int32_t                 Team0Score;                    // 0x012C (0x0004)
[0x0000000004000000] (CPF_EditInlineNotify)
int32_t                 Team1Score;                    // 0x0130 (0x0004)
[0x0000000004000000] (CPF_EditInlineNotify)
TArray<class UMatchPlayerData_X*>    Players;          // 0x0138 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.MatchData_X");
}

```

```

return uClassPointer;
};

```

```

};

```

```

// Class ProjectX.MatchEndedMessage_X
// 0x0000 (0x0060 - 0x0060)
class UMatchEndedMessage_X : public UBeaconMessage_X
{
public:

public:

```

```

static UClass* StaticClass()
{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class ProjectX.MatchEndedMessage_X");
}

return UClassPointer;
};

};

// Class ProjectX.MatchInfoMessage_X
// 0x0048 (0x0070 - 0x00B8)
class UMatchInfoMessage_X : public UServerToServerMessage_X
{
public:
int32_t Playlist; // 0x0070 (0x0004)
[0x0000000000000000]
unsigned long blsBotMatch : 1; // 0x0074 (0x0004)
[0x0000000000000000] [0x00000001]
TArray<class FString> BotNames; // 0x0078 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
class FString ReservationID; // 0x0088 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
class FString JoinName; // 0x0098 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
class FString JoinPassword; // 0x00A8 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class ProjectX.MatchInfoMessage_X");
}

return UClassPointer;
};

struct FPsyNetBeaconReservation GetReservation();
};

// Class ProjectX.MatchLog_X
// 0x0028 (0x0060 - 0x0088)
class UMatchLog_X : public UObject
{
public:
int32_t LogFileStartSize; // 0x0060 (0x0004)

```

```

[0x0000000000000000]
class FString                                URL;                                // 0x0068 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
TArray<uint8_t>                               Content;                                // 0x0078 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.MatchLog_X");
}

return uClassPointer;
};

void ConditionalSendWebRequest();
void End(int32_t MaxSize);
void SetURL(class FString InURL);
class FString GetLogFileName();
int32_t GetLogFileSize();
void Start();
};

// Class ProjectX.MaterialEffect_X
// 0x0038 (0x0090 - 0x00C8)
class UMaterialEffect_X : public UMaterialEffect
{
public:
struct FName                                TimeParamName;                                // 0x0090 (0x0008)
[0x00000000000000001] (CPF_Edit)
float                                FadeInTime;                                // 0x0098 (0x0004)
[0x00000000000000001] (CPF_Edit)
float                                FadeInFalloff;                                // 0x009C (0x0004)
[0x00000000000000001] (CPF_Edit)
float                                FadeOutTime;                                // 0x00A0 (0x0004)
[0x00000000000000001] (CPF_Edit)
float                                FadeOutFalloff;                                // 0x00A4 (0x0004)
[0x00000000000000001] (CPF_Edit)
float                                ActiveTime;                                // 0x00A8 (0x0004)
[0x00000000000000001] (CPF_Edit)
float                                ValueMin;                                // 0x00AC (0x0004)
[0x00000000000000001] (CPF_Edit)
float                                ValueMax;                                // 0x00B0 (0x0004)
[0x00000000000000001] (CPF_Edit)
class UMaterialInstanceConstant*    MatInst;                                // 0x00B8 (0x0008)
[0x00000000000002000] (CPF_Transient)
uint8_t                                Stage;                                // 0x00C0 (0x0001)
[0x00000000000002000] (CPF_Transient)
float                                StageTime;                                // 0x00C4 (0x0004)
[0x00000000000002000] (CPF_Transient)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.MaterialEffect_X");
}

return uClassPointer;
};

class FString GetActiveEffects();
bool HasAnyEffectsActive();
float GetMaterialParameterValue(struct FName MaterialParamName);
void SetMaterialParameterLinearColorValue(struct FName MaterialParamName, struct FLinearColor NewValue);
void SetMaterialParameterValue(struct FName MaterialParamName, float NewValue);
void HandleParametersChanged();
void UpdateFade(float Alpha);
void SetStage(uint8_t NewStage);
bool Tick(float dt);
void End();
void Start();
void Init();
bool IsInitialized();
};

// Class ProjectX.PauseMetricsState_X
// 0x0000 (0x0060 - 0x0060)
class UPauseMetricsState_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PauseMetricsState_X");
}

return uClassPointer;
};

};

// Class ProjectX.NetModeBase
// 0x0000 (0x0060 - 0x0060)
class UNetModeBase : public UObject

```

```

{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.NetModeBase");
}

return uClassPointer;
};

};

// Class ProjectX.NetMode_Networked
// 0x0000 (0x0060 - 0x0060)
class UNetMode_Networked : public UNetModeBase
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.NetMode_Networked");
}

return uClassPointer;
};

};

// Class ProjectX.MigrationStartedMessage_X
// 0x00F0 (0x0070 - 0x0160)
class UMigrationStartedMessage_X : public UServerToServerMessage_X
{
public:
unsigned long bRematch : 1; // 0x0070 (0x0004)
[0x0000000000000000] [0x00000001]
struct FCustomMatchSettings PrivateMatchSettings; // 0x0078
(0x0090) [0x0000000000040000] (CPF_NeedCtorLink)
struct FUniqueNetId PrivateMatchCreator; // 0x0108 (0x0048)
[0x0000000000040000] (CPF_NeedCtorLink)
TArray<struct FMigrationReservationData> MigratingPlayers; // 0x0150
(0x0010) [0x0000000000040000] (CPF_NeedCtorLink)

public:

```



```

static UClass* StaticClass()
{
    static UClass* uClassPointer = nullptr;

    if (!uClassPointer)
    {
        uClassPointer = UObject::FindClass("Class ProjectX.MigrationStartedMessage_X");
    }

    return uClassPointer;
};

};

// Class ProjectX.MirrorUtils_X
// 0x0000 (0x0060 - 0x0060)
class UMirrorUtils_X : public UObject
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.MirrorUtils_X");
        }

        return uClassPointer;
    };

    static struct FRotator MirrorRotatorYaw(struct FRotator InRotator, struct FRotator InMirrorAngle);
    static struct FVector MirrorVectorXY(struct FVector InVectorToMirror, struct FVector InMirrorPoint, struct FVector InMirrorNormal);
    static struct FVector CalculateMirrorLocationNormal(struct FRotator InMirrorAngle);
};

// Class ProjectX.NetMetricsSystem_X
// 0x0000 (0x0060 - 0x0060)
class UNetMetricsSystem_X : public UObject
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.NetMetricsSystem_X");
        }
    }
};

```

```

return uClassPointer;
};

static void Exit(class UPreExitEvent_X* Event, class UClientNetMetrics_X* Metrics);
static void RecordGamePing(class UNetworkPingEvent_X* Ping, class UClientNetMetrics_X* Metrics);
static void PlayerRemoved(class UClientNetMetrics_X* Metrics);
static void PlayerAdded(class UPrimaryPlayer* PP, class UNetMode_DedicatedClient* NetMode, class APlayerReplicationInfo* PRI);
};

// Class ProjectX.NetMode_DedicatedClient
// 0x0000 (0x0060 - 0x0060)
class UNetMode_DedicatedClient : public UNetModeBase
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.NetMode_DedicatedClient");
}

return uClassPointer;
};

};

// Class ProjectX.NetworkPingEvent_X
// 0x0004 (0x0060 - 0x0064)
class UNetworkPingEvent_X : public UObject
{
public:
float DeltaSeconds; // 0x0060 (0x0004)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.NetworkPingEvent_X");
}

return uClassPointer;
};

```

```

};

// Class ProjectX.NetMode_Authoritative
// 0x0000 (0x0060 - 0x0060)
class UNetMode_Authoritative : public UNetModeBase
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.NetMode_Authoritative");
}

return uClassPointer;
};

};

// Class ProjectX.NetMode_Client
// 0x0000 (0x0060 - 0x0060)
class UNetMode_Client : public UNetModeBase
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.NetMode_Client");
}

return uClassPointer;
};

};

// Class ProjectX.NetMode_DedicatedServer
// 0x0000 (0x0060 - 0x0060)
class UNetMode_DedicatedServer : public UNetModeBase
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```
if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.NetMode_DedicatedServer");
}
```

```
return uClassPointer;
};
```

```
};
```

```
// Class ProjectX.NetMode_ListenServer
// 0x0000 (0x0060 - 0x0060)
class UNetMode_ListenServer : public UNetModeBase
{
public:
```

```
public:
static UClass* StaticClass()
{
    static UClass* uClassPointer = nullptr;
```

```
if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.NetMode_ListenServer");
}
```

```
return uClassPointer;
};
```

```
};
```

```
// Class ProjectX.NetMode_Server
// 0x0000 (0x0060 - 0x0060)
class UNetMode_Server : public UNetModeBase
{
public:
```

```
public:
static UClass* StaticClass()
{
    static UClass* uClassPointer = nullptr;
```

```
if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.NetMode_Server");
}
```

```
return uClassPointer;
};
```

```
};
```

```
// Class ProjectX.NetMode_Standalone
```

```

// 0x0000 (0x0060 - 0x0060)
class UNetMode_Standalone : public UNetModeBase
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.NetMode_Standalone");
}

return uClassPointer;
};

};

// Class ProjectX.NetModeReplicator_X
// 0x0000 (0x0268 - 0x0268)
class ANetModeReplicator_X : public AReplicationInfo
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.NetModeReplicator_X");
}

return uClassPointer;
};

};

// Class ProjectX.NetModeSystem_X
// 0x0000 (0x0060 - 0x0060)
class UNetModeSystem_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{

```

```

uClassPointer = UObject::FindClass("Class ProjectX.NetModeSystem_X");
}

return uClassPointer;
};

static void AddModes(TArray<class UClass*>& NetModes);
static void HandleDedicatedClient(class ANetModeReplicator_X* _, class UNetMode_Client*
NetMode);
static void LevelUnloaded(class AWorldInfo* WorldInfo);
static void LevelLoaded(class AWorldInfo* WorldInfo);
};

// Class ProjectX.RPC_CreateClub_X
// 0x0028 (0x00F8 - 0x0120)
class URPC_CreateClub_X : public URPC_ClubDetailsBase_X
{
public:
class FString                ClubName;                // 0x00F8 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
class FString                ClubTag;                // 0x0108 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
int32_t                    PrimaryColor;            // 0x0118 (0x0004)
[0x0001000000000000]
int32_t                    AccentColor;            // 0x011C (0x0004)
[0x0001000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_CreateClub_X");
}

return uClassPointer;
};

class UErrorType* eventOverrideErrorType(class UErrorType* ErrorType);
class URPC_CreateClub_X* SetSettings(class UClubSettings_X* Settings);
};

// Class ProjectX.RPC_UpdateClubName_X
// 0x0020 (0x00F8 - 0x0118)
class URPC_UpdateClubName_X : public URPC_ClubDetailsBase_X
{
public:
class FString                ClubName;                // 0x00F8 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
class FString                ClubTag;                // 0x0108 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_UpdateClubName_X");
}

return uClassPointer;
};

class UErrorType* eventOverrideErrorType(class UErrorType* ErrorType);
class URPC_UpdateClubName_X* SetName(class FString InName, class FString InTag);
};

// Class ProjectX.RPC_UpdateClubColors_X
// 0x0008 (0x00F8 - 0x0100)
class URPC_UpdateClubColors_X : public URPC_ClubDetailsBase_X
{
public:
int32_t PrimaryColor; // 0x00F8 (0x0004)
[0x0001000000000000]
int32_t AccentColor; // 0x00FC (0x0004)
[0x0001000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_UpdateClubColors_X");
}

return uClassPointer;
};

class URPC_UpdateClubColors_X* SetColors(int32_t InPrimary, int32_t InAccent);
};

// Class ProjectX.RPC_UpdateClubMotD_X
// 0x0010 (0x00F8 - 0x0108)
class URPC_UpdateClubMotD_X : public URPC_ClubDetailsBase_X
{
public:
class FString MotD; // 0x00F8 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{

```

```

static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
    UClassPointer = UObject::FindClass("Class ProjectX.RPC_UpdateClubMotD_X");
}

return UClassPointer;
};

class UErrorType* eventOverrideErrorType(class UErrorType* ErrorType);
class URPC_UpdateClubMotD_X* SetMotD(class FString InMotD);
};

// Class ProjectX.RPC_InviteToClub_X
// 0x0048 (0x00E8 - 0x0130)
class URPC_InviteToClub_X : public URPC_X
{
public:
    struct FUniqueNetId                PlayerID;                // 0x00E8 (0x0048)
    [0x0001000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* UClassPointer = nullptr;

        if (!UClassPointer)
        {
            UClassPointer = UObject::FindClass("Class ProjectX.RPC_InviteToClub_X");
        }

        return UClassPointer;
    };

    class UErrorType* eventOverrideErrorType(class UErrorType* ErrorType);
    class URPC_InviteToClub_X* SetPlayerID(struct FUniqueNetId InPlayerId);
};

// Class ProjectX.RPC_RemoveFromClub_X
// 0x0048 (0x00F8 - 0x0140)
class URPC_RemoveFromClub_X : public URPC_ClubDetailsBase_X
{
public:
    struct FUniqueNetId                PlayerID;                // 0x00F8 (0x0048)
    [0x0001000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* UClassPointer = nullptr;

        if (!UClassPointer)
        {

```



```

uClassPointer = UObject::FindClass("Class ProjectX.RPC_RemoveFromClub_X");
}

return uClassPointer;
};

class URPC_RemoveFromClub_X* SetPlayerID(struct FUniqueNetId InPlayerId);
};

// Class ProjectX.RPC_SetClubOwner_X
// 0x0048 (0x00F8 - 0x0140)
class URPC_SetClubOwner_X : public URPC_ClubDetailsBase_X
{
public:
    struct FUniqueNetId          PlayerID;                // 0x00F8 (0x0048)
    [0x0001000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_SetClubOwner_X");
        }

        return uClassPointer;
    };

class URPC_SetClubOwner_X* SetNewOwner(struct FUniqueNetId InPlayerId);
};

// Class ProjectX.RPC_GetClubInvites_X
// 0x0010 (0x00E8 - 0x00F8)
class URPC_GetClubInvites_X : public URPC_X
{
public:
    TArray<class UClubInvite_X*> ClubInvites;                // 0x00E8 (0x0010)
    [0x0001000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetClubInvites_X");
        }

        return uClassPointer;
    };

```

```

TArray<class UClubInvite_X*> __RPC_GetClubInvites_X__CreateClubInvitesTask_0x1();
class UAsyncResult__array_ClubInvite_X* CreateClubInvitesTask();
};

// Class ProjectX.RPC_AcceptClubInvite_X
// 0x0004 (0x00F8 - 0x00FC)
class URPC_AcceptClubInvite_X : public URPC_ClubDetailsBase_X
{
public:
    int32_t                                ClubID;                                // 0x00F8 (0x0004)
    [0x0001000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_AcceptClubInvite_X");
        }

        return uClassPointer;
    };

    class URPC_AcceptClubInvite_X* SetClubID(uint64_t InClubID);
};

// Class ProjectX.RPC_RejectClubInvite_X
// 0x0004 (0x00E8 - 0x00EC)
class URPC_RejectClubInvite_X : public URPC_X
{
public:
    int32_t                                ClubID;                                // 0x00E8 (0x0004)
    [0x0001000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_RejectClubInvite_X");
        }

        return uClassPointer;
    };

    class URPC_RejectClubInvite_X* SetClubID(uint64_t InClubID);
};

// Class ProjectX.RPC_LeaveClub_X
// 0x0000 (0x00E8 - 0x00E8)

```

```

class URPC_LeaveClub_X : public URPC_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_LeaveClub_X");
}

return uClassPointer;
};

};

// Class ProjectX.OnlineClubServerList_X
// 0x0020 (0x0060 - 0x0080)
class UOnlineClubServerList_X : public UObject
{
public:
TArray<class UClubServerResult_X*> Servers; // 0x0060 (0x0010)
[0x0001000000400000] (CPF_NeedCtorLink)
class URPC_GetClubPrivateMatches_X* RPC; // 0x0070 (0x0008)
[0x0001004000000000]
class UError* Error; // 0x0078 (0x0008)
[0x0001004000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineClubServerList_X");
}

return uClassPointer;
};

void __OnlineClubServerList_X__Refresh_0x3(class URPC_X* _);
void __OnlineClubServerList_X__Refresh_0x2(class URPC_X* _);
void __OnlineClubServerList_X__Refresh_0x1(class URPC_X* _);
class UAsyncTask* Refresh();
};

// Class ProjectX.RPC_GetClubPrivateMatches_X
// 0x0010 (0x00E8 - 0x00F8)
class URPC_GetClubPrivateMatches_X : public URPC_X
{

```

```

public:
TArray<class UClubServerResult_X*> Servers; // 0x00E8 (0x0010)
[0x0001000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetClubPrivateMatches_X");
}

return uClassPointer;
};

};

// Class ProjectX.OnlineConfigDispatcher_X
// 0x0004 (0x0060 - 0x0064)
class UOnlineConfigDispatcher_X : public UObject
{
public:
unsigned long bDebug : 1; // 0x0060 (0x0004)
[0x0000000000000001] [0x00000001] (CPF_Edit)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineConfigDispatcher_X");
}

return uClassPointer;
};

class FString GetDebugString(class UOnlineConfig_X* Config);
void UndoConfigObject(class UOnlineConfig_X* Config);
void ApplyConfigObject(class UOnlineConfig_X* Config);
void eventConstruct();
};

// Class ProjectX.UrlConfig_X
// 0x0040 (0x0060 - 0x00A0)
class UUrlConfig_X : public UObject
{
public:
class FString CDN; // 0x0060 (0x0010)
[0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
class FString ESportsURL; // 0x0070 (0x0010)

```

```

[0x0000000000400000] (CPF_NeedCtorLink)
class FString                                CrossPlatformSupportURL;                // 0x0080 (0x0010)
[0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)
class FString                                LegalTextWebFolder;                // 0x0090 (0x0010)
[0x0000000000404000] (CPF_Config | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.UrlConfig_X");
}

return uClassPointer;
};

};

// Class ProjectX.OnlineGameVersion_X
// 0x0004 (0x00B0 - 0x00B4)
class UOnlineGameVersion_X : public UOnline_X
{
public:
unsigned long                                bUpdateRequired : 1;                // 0x00B0 (0x0004)
[0x0000004000002000] [0x00000001] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameVersion_X");
}

return uClassPointer;
};

void __OnlineGameVersion_X__OnInit_0x4(class UOnlineGameServerBrowser_X* __, class UError*
Error);
void __OnlineGameVersion_X__OnInit_0x3(class UOnlineGamePrivateMatch_X* __, class UError*
Error);
void __OnlineGameVersion_X__OnInit_0x2(class UOnlineGameRegions_X* __, class UError* Error);
void __OnlineGameVersion_X__OnInit_0x1(class UOnlineGameMatchmaking_X* __, class UError*
Error);
void HandleError(class UError* Error);
void OnInit();
};

// Class ProjectX.OnlineGameWordFilterProcessor_X

```

```

// 0x0030 (0x00B0 - 0x00E0)
class UOnlineGameWordFilterProcessor_X : public UOnline_X
{
public:
float                                ResponseDelay;                                // 0x00B0 (0x0004)
[0x00000000000000002] (CPF_Const)
TArray<struct FWordFilterRequest>      SanitizeCallbacks;                        // 0x00B8
(0x0010) [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FScriptDelegate               __OnSanitizeStringComplete__Delegate;      // 0x00C8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameWordFilterProcessor_X");
}

return uClassPointer;
};

void TriggerCallbacks(int32_t Index);
void ProcessSanitizedDelayed();
void HandleWordSanitized(class UWebRequest_X* Request);
bool SanitizeString(class FString Comment, struct FScriptDelegate SanitizeDelegate);
void OnSanitizeStringComplete(class FString Original, class FString Sanitized);
};

// Class ProjectX.OnlineGameRegions_X
// 0x0088 (0x00B0 - 0x0138)
class UOnlineGameRegions_X : public UOnline_X
{
public:
float                                PingRegionsCacheTime;                        // 0x00B0 (0x0004)
[0x00000000000000001] (CPF_Edit)
float                                LastPingRegionsTime;                        // 0x00B4 (0x0004)
[0x00000000000002000] (CPF_Transient)
class URegionConfig_X*               Config;                                    // 0x00B8 (0x0008)
[0x00000804000000000]
TArray<class URegionPing_X*>          RegionPings;                            // 0x00C0 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<struct FCachedRegionPing>      CachedRegionPings;                        // 0x00D0
(0x0010) [0x0000008000402000] (CPF_Transient | CPF_NeedCtorLink)
class UBeaconConfig_X*               BeaconConfig;                            // 0x00E0 (0x0008)
[0x00000800000000000]
unsigned long                         bPingingRegions : 1;                        // 0x00E8 (0x0004)
[0x0000004000002000] [0x00000001] (CPF_Transient)
struct FScriptDelegate               __EventRegionsSynced__Delegate;            // 0x00F0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate               __EventRegionsPinged__Delegate;            // 0x0108
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

```

```

struct FScriptDelegate          __EventRegionsError__Delegate;          // 0x0120
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameRegions_X");
}

return uClassPointer;
};

class FString __OnlineGameRegions_X__HandleGetPingRegionPingsRPC_0x4(class
URegionPing_X* R);
bool __OnlineGameRegions_X__HandleGetPingRegionPingsRPC_0x3(class URegionPing_X* R);
void __OnlineGameRegions_X__HandleGetPingRegionPingsRPC_0x2(struct
FGetGameServerPingListData Server);
void __OnlineGameRegions_X__HandleGetPingRegionPingsRPC_0x1(class URegionPing_X*
RegionPing);
class FString GetLocalizedName(class FString RegionID);
bool Exists(class FString RegionID);
void SetCachedRegionPings(TArray<struct FCachedRegionPing>& InCachedRegionPings);
void PrintDebugInfo(class UDebugDrawer* Drawer);
void NotifyWhenSynced(struct FScriptDelegate Callback);
class FString GetRegionDebugString(class URegionPing_X* RegionPing);
class FString GetRegionsDebugString();
int32_t SortRegionDelegate(class URegionPing_X* A, class URegionPing_X* B);
void FinalizeRegionPing(class URegionPing_X* RegionPing);
void OnAllRegionsPinged();
void HandleRegionPinged(class URegionPingData_X* Data);
void HandleRegionsPinged(class URegionPinger_X* Pinger);
void HandleGetPingRegionPingsRPC(class URPC_GetGameServerPingList_X* RPC);
void PingRegions(struct FScriptDelegate Callback);
void UpdateRegionPings();
void OnRegionsSynced();
void AddRegionPing(class URegion_X* Region);
class URegionPing_X* GetRegionPing(class FString RegionID);
void EventRegionsError(class UOnlineGameRegions_X* RegionsObj, class UError* Error);
void EventRegionsPinged(class UOnlineGameRegions_X* RegionsObj);
void EventRegionsSynced(class UOnlineGameRegions_X* RegionsObj);
};

// Class ProjectX.OnlineGameInvite_X
// 0x0108 (0x00B0 - 0x01B8)
class UOnlineGameInvite_X : public UOnline_X
{
public:
struct FUniqueNetId          FriendJoinPlayerID;          // 0x00B0 (0x0048)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                UnableToFindFriendsGameString;          // 0x00F8 (0x0010)

```

```

[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                                AlreadyInSameServerString;                // 0x0108 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                                FriendNotJoinable;                    // 0x0118 (0x0010)
[0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString                                GamelInviteCredentials;              // 0x0128 (0x0010)
[0x00000000000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FName                                ActionRequired;                      // 0x0138 (0x0008)
[0x00000000000002000] (CPF_Transient)
struct FScriptDelegate                      __EventGamelInviteAccepted__Delegate; // 0x0140
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventGamelInviteComplete__Delegate; // 0x0158
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventConfirmationRequired__Delegate; // 0x0170
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventPasswordRequired__Delegate;    // 0x0188
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate                      __EventPsyNetPartyInviteAccepted__Delegate; //
0x01A0 (0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGamelInvite_X");
}

```

```

return uClassPointer;
};

```

```

void HandleJoinGameComplete(unsigned long bSuccess, class FString FailReason);
void HandlePasswordRequired();
void HandleTaskSuccess(struct FServerReservationData Reservation);
void __OnlineGamelInvite_X__BeginState_0x1(class UError* Error);
void OnGamelInviteComplete(unsigned long bSuccess, class FString FailReason);
void JoinGamelInviteGame(struct FJoinMatchSettings Settings);
void OnGamelInviteAccepted(class FString ErrorString, struct FOnlineGameSearchResult&
InviteResult);
void OnInit();
void EventPsyNetPartyInviteAccepted(class FString PartyID);
void EventPasswordRequired();
void EventConfirmationRequired(struct FName ConfirmationReason);
void EventGamelInviteComplete(unsigned long bSuccess, class FString FailReason);
void EventGamelInviteAccepted();
};

```

```

// Class ProjectX.OnlineGameVoice_X
// 0x0018 (0x00B0 - 0x00C8)
class UOnlineGameVoice_X : public UOnline_X
{
public:

```



```

struct FScriptDelegate          __EventPlayerTalking__Delegate;          // 0x00B0
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameVoice_X");
}

return uClassPointer;
};

uint8_t GetControllerID(class APlayerReplicationInfo* PRI);
int32_t GetNumTalkers();
bool IsRemotePlayerTalking(class APlayerReplicationInfo* PRI);
void HandlePlayerTalkingStateChange(struct FUniqueNetId PlayerID, unsigned long bTalking);
void UnregisterTalker(class UOnlinePlayer_X* Player);
void RegisterTalker(class UOnlinePlayer_X* Player);
void OnInit();
void EventPlayerTalking(class UOnlineGameVoice_X* SelfRef, struct FUniqueNetId PlayerID,
unsigned long bTalking);
};

// Class ProjectX.OnlineGamePopulation_X
// 0x0040 (0x00B0 - 0x00F0)
class UOnlineGamePopulation_X : public UOnline_X
{
public:
float          UpdatePopulationDelay;          // 0x00B0 (0x0004)
[0x0000000000000001] (CPF_Edit)
float          LastUpdatePopulationTime;       // 0x00B4 (0x0004)
[0x0000000400000200] (CPF_Transient)
class URPC_X*  GetPopulationRPC;               // 0x00B8 (0x0008)
[0x0000000400000200] (CPF_Transient)
int32_t        TotalPlayers;                   // 0x00C0 (0x0004)
[0x0000000400000200] (CPF_Transient)
TArray<struct FGetPopulationData> Populations; // 0x00C8
(0x0010) [0x0000000400040200] (CPF_Transient | CPF_NeedCtorLink)
struct FScriptDelegate          __EventGetPlaylistPopulations__Delegate; // 0x00D8
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGamePopulation_X");
}
}

```

```

return uClassPointer;
};

int32_t __OnlineGamePopulation_X__HandleGotAllPopulationPlaylistsRPC_0x2(struct
FGetPopulationData P);
bool __OnlineGamePopulation_X__HandleGotAllPopulationPlaylistsRPC_0x1(struct
FGetPopulationData P);
void HandleGotAllPopulationPlaylistsRPC(class URPC_GetPopulation_X* RPC);
void GetPlaylistPopulations(struct FScriptDelegate Callback);
void EventGetPlaylistPopulations(class UOnlineGamePopulation_X* PopulationsObject);
};

// Class ProjectX.OnlineGameServerBrowser_X
// 0x0130 (0x00B0 - 0x01E0)
class UOnlineGameServerBrowser_X : public UOnline_X
{
public:
class FString PreferredRegion; // 0x00B0 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
struct FCustomMatchSettings Filter; // 0x00C0 (0x0090)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString StartSearchFailString; // 0x0150 (0x0010)
[0x0000000000408003] (CPF_Edit | CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString NoResultsFound; // 0x0160 (0x0010)
[0x0000000000408003] (CPF_Edit | CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString NotLoggedInToPsynet; // 0x0170 (0x0010)
[0x0000000000408003] (CPF_Edit | CPF_Const | CPF_Localized | CPF_NeedCtorLink)
struct FScriptDelegate __EventSearchComplete__Delegate; // 0x0180
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventSearchError__Delegate; // 0x0198
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventBrowserError__Delegate; // 0x01B0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventSearchCompleteReservation__Delegate; //
0x01C8 (0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGameServerBrowser_X");
}

return uClassPointer;
};

void HandleGameStarted(class AGRI_X* GRI);
void PerformSearch();
void HandleTaskFail(class UError* Error);
void HandleTaskSuccess(struct FServerReservationData Reservation);

```

```

class FString GetRegionsString();
void OnSearchComplete(TArray<struct FServerResult>& Results);
void OnSearchError(class FString Message);
bool IsSearching();
void Cancel();
bool StartSearch(struct FCustomMatchSettings InFilter, class FString InPreferredRegion);
class UOnlineGameServerBrowser_X* SetCompleteReservationDelegate(struct FScriptDelegate
Callback);
class UOnlineGameServerBrowser_X* SetCompleteDelegate(struct FScriptDelegate Callback);
class UOnlineGameServerBrowser_X* SetErrorDelegate(struct FScriptDelegate Callback);
void EventSearchCompleteReservation(struct FServerReservationData& Reservation);
void EventBrowserError(class UOnlineGameServerBrowser_X* Browser, class UError* Error);
void EventSearchError(class FString NewStatus);
void EventSearchComplete(TArray<struct FServerResult>& Results);
};

// Class ProjectX.OnlineGamePrivateMatch_X
// 0x0120 (0x0130 - 0x0250)
class UOnlineGamePrivateMatch_X : public UOnlineGameMatchmakingBase_X
{
public:
class FString PreferredRegion; // 0x0130 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString SearchingString; // 0x0140 (0x0010)
[0x0000000000408003] (CPF_Edit | CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString StartSearchFailString; // 0x0150 (0x0010)
[0x0000000000408003] (CPF_Edit | CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString FoundServerString; // 0x0160 (0x0010)
[0x0000000000408003] (CPF_Edit | CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString AlreadyJoiningGameString; // 0x0170 (0x0010)
[0x0000000000408003] (CPF_Edit | CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString TimeoutString; // 0x0180 (0x0010)
[0x0000000000408003] (CPF_Edit | CPF_Const | CPF_Localized | CPF_NeedCtorLink)
class FString WrongRegionString; // 0x0190 (0x0010)
[0x0000000000408003] (CPF_Edit | CPF_Const | CPF_Localized | CPF_NeedCtorLink)
struct FCustomMatchSettings Settings; // 0x01A0 (0x0090)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
float SearchTimeout; // 0x0230 (0x0004)
[0x0000000000000003] (CPF_Edit | CPF_Const)
struct FScriptDelegate __EventPrivateMatchError__Delegate; // 0x0238
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineGamePrivateMatch_X");
}

return uClassPointer;
};

```

```

void UpdateStatusMessage();
void HandleError(class UError* Error);
void StartCheckingReservations();
void HandleStartSearch(class UAsyncTask* Task);
void OnReceiveGameServer(struct FServerReservationData Reservation);
void HandleSearchTimeout();
void ResetPrivateMatchSettings();
void UpdatePrivateMatchSettings(struct FCustomMatchSettings InSettings);
void OnPrivateMatchError(class FString Message);
bool StartSearch(class FString InPreferredRegion, struct FCustomMatchSettings InSettings);
void EventPrivateMatchError(class UOnlineGamePrivateMatch_X* InPrivateMatch, class UError*
Error);
};

```

```

// Class ProjectX.UdpLanBrowser_X
// 0x0030 (0x0060 - 0x0090)
class UUdpLanBrowser_X : public ULanBrowser_X
{
public:
float SearchTimeout; // 0x0060 (0x0004)
[0x0000000000000003] (CPF_Edit | CPF_Const)
class ULanBeacon_X* LanBeacon; // 0x0068 (0x0008)
[0x0000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
class UAsyncResult__array_LanServerRecord_X* SearchTask; // 0x0070
(0x0008) [0x0000000000000000]
TArray<class ULanServerRecord_X*> SearchResults; // 0x0078
(0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
class UUdpLanServer_X* LocalServer; // 0x0088 (0x0008)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.UdpLanBrowser_X");
}

return uClassPointer;
};

```

```

class UAsyncTask* JoinServer(class FString ServerId, class FString Options);
void HandleSearchTimeout();
void HandleHostResponse(class UOnlineMessageComponent_X* Component, class
ULanMessage_HostResponse_X* Response);
void SearchTaskDisposed();
class UAsyncResult__array_LanServerRecord_X* GetServerList();
void DestroyServer();
class UAsyncTask* SetServerMetaData(class FString MetaData);
class UAsyncTask* CreateServer(class FString MetaData);
};

```

```

// Class ProjectX.SystemMetrics_X
// 0x0000 (0x0080 - 0x0080)
class USystemMetrics_X : public UMetricsGroup_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.SystemMetrics_X");
}

return uClassPointer;
};

void Specs(struct FOSMetrics OS, struct FCpuMetrics CPU, struct FMemoryMetrics Memory,
struct FVideoCardMetrics Video, struct FNetworkAdapterMetrics Network);
void RecordSpecs();
};

// Class ProjectX.RPC_GetPublicIP_X
// 0x0010 (0x00E8 - 0x00F8)
class URPC_GetPublicIP_X : public URPC_X
{
public:
class FString IP; // 0x00E8 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetPublicIP_X");
}

return uClassPointer;
};

};

// Class ProjectX.RPC_UpdatePlayerPlaylist_X
// 0x0008 (0x00E8 - 0x00F0)
class URPC_UpdatePlayerPlaylist_X : public URPC_X
{
public:
int32_t Playlist; // 0x00E8 (0x0004)

```

```

[0x0000004000000000]
int32_t NumLocalPlayers; // 0x00EC (0x0004)
[0x0000004000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_UpdatePlayerPlaylist_X");
}

return uClassPointer;
};

class URPC_UpdatePlayerPlaylist_X* SetNumLocalPlayers(int32_t InNumLocalPlayers);
class URPC_UpdatePlayerPlaylist_X* SetPlaylist(int32_t InPlaylist);
};

// Class ProjectX.ServerExploitManager_X
// 0x0030 (0x0060 - 0x0090)
class UServerExploitManager_X : public UObject
{
public:
class UOnlineGameDedicatedServer_X* DedicatedServer; // 0x0060
(0x0008) [0x0000000000000000]
TArray<struct FMatchExploitReportData> SubmittedReports; // 0x0068
(0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __SubmitReport__Delegate; // 0x0078
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ServerExploitManager_X");
}

return uClassPointer;
};

void AddReport(struct FUniqueNetId PlayerID, uint8_t Reason);
bool FindPlayerReportIndex(struct FUniqueNetId PlayerID, int32_t& OutIndex);
bool DoesReportExist(struct FUniqueNetId PlayerID, uint8_t Reason);
void Reset();
void Report(struct FUniqueNetId PlayerID, uint8_t Reason, class FString Data);
void SetDedicatedServer(class UOnlineGameDedicatedServer_X* InServer);
void SubmitReport(struct FUniqueNetId PlayerID, uint8_t Reason, class FString Data);
};

```

```

// Class ProjectX.OnlinePlayerPermissions_X
// 0x0040 (0x0060 - 0x00A0)
class UOnlinePlayerPermissions_X : public UObject
{
public:
float                               SyncDelay;                               // 0x0060 (0x0004)
[0x0000000000000001] (CPF_Edit)
TArray<struct FUniqueNetId>          PendingPlayerRequests;                  // 0x0068
(0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
TArray<struct FPlayerPermissionsList> PlayerPermissions;                    // 0x0078
(0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate              __EventUpdated__Delegate;              // 0x0088
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlinePlayerPermissions_X");
}

return uClassPointer;
};

bool HasPermission(struct FUniqueNetId PlayerID, uint8_t Permission);
void HandlePlayerPermissions(class URPC_GetPlayerPermissions_X* RPC);
void SendRequest();
void SyncPermissions(struct FUniqueNetId PlayerID);
void EventUpdated(class UOnlinePlayerPermissions_X* Permissions);
};

// Class ProjectX.RPC_CheckReplacementDedicatedServer_X
// 0x0050 (0x00E8 - 0x0138)
class URPC_CheckReplacementDedicatedServer_X : public URPC_X
{
public:
class FString                       ServerId;                               // 0x00E8 (0x0010)
[0x0000000400040000] (CPF_NeedCtorLink)
unsigned long                        bFoundReplacement : 1;                  // 0x00F8 (0x0004)
[0x0000000400000200] [0x00000001] (CPF_Transient)
struct FCheckReplacementDedicatedServerData Server;                          // 0x0100
(0x0038) [0x0000000400040200] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{

```

```

uClassPointer = UObject::FindClass("Class
ProjectX.RPC_CheckReplacementDedicatedServer_X");
}

return uClassPointer;
};

class FString GetServerAddress();
class URPC_CheckReplacementDedicatedServer_X* SetServerID(class FString InServerID);
};

// Class ProjectX.RPC_AddQuitter_X
// 0x0070 (0x00E8 - 0x0158)
class URPC_AddQuitter_X : public URPC_X
{
public:
    struct FUniqueNetId                PlayerID;                // 0x00E8 (0x0048)
    [0x0000000400040000] (CPF_NeedCtorLink)
    class FString                    Reason;                    // 0x0130 (0x0010)
    [0x0000000400040000] (CPF_NeedCtorLink)
    int32_t                        PlaylistId;                // 0x0140 (0x0004)
    [0x0000000400000000]
    class FString                    MatchGuid;                // 0x0148 (0x0010)
    [0x0000000400040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_AddQuitter_X");
        }

        return uClassPointer;
    };

    class URPC_AddQuitter_X* SetMatchGUID(class FString InMatchGUID);
    class URPC_AddQuitter_X* SetPlaylistID(int32_t InPlaylistID);
    class URPC_AddQuitter_X* SetReason(class FString InReason);
    class URPC_AddQuitter_X* SetPlayerID(struct FUniqueNetId InPlayerId);
};

// Class ProjectX.RPC_RemoveQuitter_X
// 0x0060 (0x00E8 - 0x0148)
class URPC_RemoveQuitter_X : public URPC_X
{
public:
    struct FUniqueNetId                PlayerID;                // 0x00E8 (0x0048)
    [0x0000000400040000] (CPF_NeedCtorLink)
    int32_t                        PlaylistId;                // 0x0130 (0x0004)
    [0x0000000400000000]
    class FString                    MatchGuid;                // 0x0138 (0x0010)

```



[0x0000004000400000] (CPF\_NeedCtorLink)

```
public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_RemoveQuitter_X");
}
```

```
return uClassPointer;
};
```

```
class URPC_RemoveQuitter_X* SetMatchGUID(class FString InMatchGUID);
class URPC_RemoveQuitter_X* SetPlaylistID(int32_t InPlaylistID);
class URPC_RemoveQuitter_X* SetPlayerID(struct FUniqueNetId InPlayerId);
};
```

```
// Class ProjectX.RPC_RecordMatch_X
// 0x0008 (0x00E8 - 0x00F0)
class URPC_RecordMatch_X : public URPC_X
{
```

```
public:
class UObject* Match; // 0x00E8 (0x0008)
[0x0000000000000000]
```

```
public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_RecordMatch_X");
}
```

```
return uClassPointer;
};
```

```
class URPC_RecordMatch_X* SetMatch(class UObject* InMatch);
};
```

```
// Class ProjectX.RPC_UpdateGameServer_X
// 0x00A4 (0x00E8 - 0x018C)
class URPC_UpdateGameServer_X : public URPC_X
{
```

```
public:
class FString ServerId; // 0x00E8 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
int32_t Playlist; // 0x00F8 (0x0004)
[0x0000004000000000]
int32_t MaxPlayers; // 0x00FC (0x0004)
```

```

[0x0000004000000000]
int32_t NumPlayersTeam1; // 0x0100 (0x0004)
[0x0000004000000000]
int32_t NumPlayersTeam2; // 0x0104 (0x0004)
[0x0000004000000000]
int32_t ReservationsTeam1; // 0x0108 (0x0004)
[0x0000004000000000]
int32_t ReservationsTeam2; // 0x010C (0x0004)
[0x0000004000000000]
unsigned long blsPostGame : 1; // 0x0110 (0x0004)
[0x0000004000000000] [0x00000001]
unsigned long blsBotMatch : 1; // 0x0110 (0x0004)
[0x0000004000000000] [0x00000002]
int32_t TimeRemaining; // 0x0114 (0x0004)
[0x0000004000000000]
class FString ExclusivePlatform; // 0x0118 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
TArray<class FString> PlayersPlatforms; // 0x0128 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
TArray<struct FUniqueNetId> PlayerIds; // 0x0138 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
TArray<struct FUniqueNetId> AbandonedPlayerIDs; // 0x0148
(0x0010) [0x0000004000400000] (CPF_NeedCtorLink)
class FString CustomServerName; // 0x0158 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
class FString CustomServerPassword; // 0x0168 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
int32_t ScoreTeam1; // 0x0178 (0x0004)
[0x0000004000000000]
int32_t ScoreTeam2; // 0x017C (0x0004)
[0x0000004000000000]
int32_t NumTeam1BackfillPlayers; // 0x0180 (0x0004)
[0x0000004000000000]
int32_t NumTeam2BackfillPlayers; // 0x0184 (0x0004)
[0x0000004000000000]
int32_t ClubID; // 0x0188 (0x0004)
[0x0001004000000000]

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_UpdateGameServer_X");
}

return uClassPointer;
};

```

```

class URPC_UpdateGameServer_X* SetBackfillTeam2(int32_t InBackfillAmount);
class URPC_UpdateGameServer_X* SetBackfillTeam1(int32_t InBackfillAmount);
class URPC_UpdateGameServer_X* SetTeam2Score(int32_t TeamScore);

```

```

class URPC_UpdateGameServer_X* SetTeam1Score(int32_t TeamScore);
class URPC_UpdateGameServer_X* SetIsBotMatch(unsigned long bBotMatch);
class URPC_UpdateGameServer_X* SetClubID(uint64_t InClubID);
class URPC_UpdateGameServer_X* SetCustomServerPassword(class FString
InCustomServerPassword);
class URPC_UpdateGameServer_X* SetCustomServerName(class FString
InCustomServerName);
class URPC_UpdateGameServer_X* SetPlayersPlatforms(TArray<class FString>
InPlayersPlatforms);
class URPC_UpdateGameServer_X* SetExclusivePlatform(class FString InExclusivePlatform);
class URPC_UpdateGameServer_X* SetTimeRemaining(int32_t InTimeRemaining);
class URPC_UpdateGameServer_X* SetIsPostGame(unsigned long bInIsPostGame);
class URPC_UpdateGameServer_X* SetAbandonedPlayerIDs(TArray<struct FUniqueNetId>
InPlayerIDs);
class URPC_UpdateGameServer_X* SetPlayerIDs(TArray<struct FUniqueNetId> InPlayerIDs);
class URPC_UpdateGameServer_X* SetReservationsTeam2(int32_t InReservationsTeam2);
class URPC_UpdateGameServer_X* SetReservationsTeam1(int32_t InReservationsTeam1);
class URPC_UpdateGameServer_X* SetNumPlayersTeam2(int32_t InNumPlayersTeam2);
class URPC_UpdateGameServer_X* SetNumPlayersTeam1(int32_t InNumPlayersTeam1);
class URPC_UpdateGameServer_X* SetMaxPlayers(int32_t InMaxPlayers);
class URPC_UpdateGameServer_X* SetPlaylist(int32_t InPlaylist);
class URPC_UpdateGameServer_X* SetServerID(class FString InServerID);
};

```

```

// Class ProjectX.RPC_CreateGameServer_X
// 0x00B8 (0x00E8 - 0x01A0)
class URPC_CreateGameServer_X : public URPC_X
{
public:
class FString                ServerName;                // 0x00E8 (0x0010)
[0x0000000400040000] (CPF_NeedCtorLink)
int32_t                      MachineId;                  // 0x00F8 (0x0004)
[0x0000000400000000]
class FString                IP;                        // 0x0100 (0x0010)
[0x0000000400040000] (CPF_NeedCtorLink)
class FString                Region;                    // 0x0110 (0x0010)
[0x0000000400040000] (CPF_NeedCtorLink)
class FString                DataCenter;                // 0x0120 (0x0010)
[0x0000000400040000] (CPF_NeedCtorLink)
int32_t                      BuildID;                    // 0x0130 (0x0004)
[0x0000000400000000]
class FString                ServerType;                // 0x0138 (0x0010)
[0x0000000400040000] (CPF_NeedCtorLink)
class FString                ServerId;                  // 0x0148 (0x0010)
[0x0000000400040200] (CPF_Transient | CPF_NeedCtorLink)
class FString                Host;                      // 0x0158 (0x0010)
[0x0000000400040200] (CPF_Transient | CPF_NeedCtorLink)
int32_t                      Zone;                      // 0x0168 (0x0004)
[0x0000000400000200] (CPF_Transient)
unsigned long                UseWebSocket : 1;          // 0x016C (0x0004)
[0x0000000000000200] [0x00000001] (CPF_Transient)
class FString                PerConURL;                 // 0x0170 (0x0010)
[0x0000000000040200] (CPF_Transient | CPF_NeedCtorLink)
class FString                PerConURLv2;               // 0x0180 (0x0010)

```

```

[0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                                PsyToken;                                // 0x0190 (0x0010)
[0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_CreateGameServer_X");
}

return uClassPointer;
};

class URPC_CreateGameServer_X* SetBuildID(int32_t InBuildID);
class URPC_CreateGameServer_X* SetDataCenter(class FString InDataCenter);
class URPC_CreateGameServer_X* SetRegion(class FString InRegion);
class URPC_CreateGameServer_X* SetIP(class FString InIP);
class URPC_CreateGameServer_X* SetMachineID(int32_t InMachineID);
class URPC_CreateGameServer_X* SetServerType(class FString InServerType);
class URPC_CreateGameServer_X* SetServerName(class FString InServerName);
};

// Class ProjectX.ServerMetrics_X
// 0x0000 (0x0080 - 0x0080)
class UServerMetrics_X : public UMetricsGroup_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ServerMetrics_X");
}

return uClassPointer;
};

void CrashedError();
void ShutDown();
void Created(class FString ServerId, class FString Host);
};

// Class ProjectX.RPC_DeactivateGameServer_X
// 0x0010 (0x00E8 - 0x00F8)
class URPC_DeactivateGameServer_X : public URPC_X
{

```

```

public:
class FString                                ServerId;                                // 0x00E8 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_DeactivateGameServer_X");
}

return uClassPointer;
};

class URPC_DeactivateGameServer_X* SetServerID(class FString InServerID);
};

// Class ProjectX.RPC_SetServerNotJoinable_X
// 0x0010 (0x00E8 - 0x00F8)
class URPC_SetServerNotJoinable_X : public URPC_X
{
public:
class FString                                ServerId;                                // 0x00E8 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_SetServerNotJoinable_X");
}

return uClassPointer;
};

class URPC_SetServerNotJoinable_X* SetServerID(class FString InServerID);
};

// Class ProjectX.ReservationsMetrics_X
// 0x0000 (0x0080 - 0x0080)
class UReservationsMetrics_X : public UMetricsGroup_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.ReservationsMetrics_X");
}

return uClassPointer;
};

void PlayerReservationWithNoPsyNetBeaconError(struct FUniqueNetId PlayerID);
void ServerMigrationPlayersAlreadySetError(class FString ServerId);
void ServerMigrationPlaylistNotSetError(class FString ServerId);
void TournamentMigrationMessageInvalidTournamentPlaylistError();
void TournamentMigrationMessageWhileActiveError();
void JoinExternalMatchReservationWhileInactiveError();
void CreateExternalMatchReservationWhileActiveError();
void DsrServerToServerFailedDeserializeError(class FString MessageType);
void JoinTournamentReservationWhileInactiveError();
void CreateTournamentReservationInvalidPlaylistError();
void CreateTournamentReservationWhileActiveError();
void JoinPrivateReservationWhileInactiveError();
void CreatePrivateReservationWhileActiveError();
void BackfillReservationIncorrectPlaylist();
void BackfillReservationWhileInactiveError();
void NewGameReservationWhileActiveError();
void SplitscreenIdMismatchError(struct FUniqueNetId PrimaryPlayerId, struct FUniqueNetId SplitscreenPlayerId);
void PartyLeaderInDuelError(struct FUniqueNetId PlayerID);
void PlayerReservationWrongIdError(struct FUniqueNetId PlayerID, class FString AttemptedId);
void JoinWhileInactiveError();
void DsrConnectionTimeoutError();
void DsrClientFailedDeserializeError(class FString MessageType);
void DsrUnreservedServerError(class FString MessageType);
void DsrServerFailedDeserializeError(struct FUniqueNetId PlayerID, class FString MessageType);
void DsrMissingReservationError(struct FUniqueNetId PlayerID);
void DsrMissingConnectionError();
void PlayerCanceled();
void NotAllPlayersJoinedError();
void GetKeysInvalidOriginError();
void GetKeysFailedError();
void ReservationsFullError();
void PlatformExclusiveReservationError();
void WrongReservationRankedMatchError();
void WrongReservationPlaylistError();
void InvalidReservationPlaylistError();
void RankedReconnect();
void RankedReconnectFinished();
void PrivateMatchFinished();
void FriendJoin();
void FriendJoinEmptyError();
void FriendJoinRankedError();
void AddReservation(struct FUniqueNetId PlayerID, int32_t Playlist, class FString ReservationID);
void FirstReservation(int32_t Playlist, class FString ReservationID);
};

```

```

// Class ProjectX.QueuedOfflineMessage_X
// 0x0000 (0x0060 - 0x0060)
class UQueuedOfflineMessage_X : public UInterface
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.QueuedOfflineMessage_X");
}

return uClassPointer;
};

int32_t GetOfflineTimeToLiveSeconds();
};

// Class ProjectX.RPC_ServerToServer_X
// 0x0030 (0x00E8 - 0x0118)
class URPC_ServerToServer_X : public URPC_X
{
public:
class FString                                ServerId;                                // 0x00E8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString                                MessageType;                                // 0x00F8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString                                MessagePayload;                                // 0x0108 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_ServerToServer_X");
}

return uClassPointer;
};

};

// Class ProjectX.ReservationsFullMessage_X
// 0x0000 (0x0060 - 0x0060)
class UReservationsFullMessage_X : public UBeaconMessage_X
{

```

public:

public:

static UClass\* StaticClass()

```
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ReservationsFullMessage_X");
}
```

return uClassPointer;

};

};

// Class ProjectX.ReservationsTeamFullMessage\_X

// 0x0000 (0x0060 - 0x0060)

class UReservationsTeamFullMessage\_X : public UBeaconMessage\_X

```
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ReservationsTeamFullMessage_X");
}
```

return uClassPointer;

};

};

// Class ProjectX.ReservationsWaitingMessage\_X

// 0x0000 (0x0060 - 0x0060)

class UReservationsWaitingMessage\_X : public UBeaconMessage\_X

```
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ReservationsWaitingMessage_X");
}
```



```

return uClassPointer;
};

};

// Class ProjectX.ReservationsReadyMessage_X
// 0x0038 (0x0060 - 0x0098)
class UReservationsReadyMessage_X : public UBeaconMessage_X
{
public:
class FString                ServerAddress;                // 0x0060 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString                PingAddress;                // 0x0070 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
TArray<int32_t>                ProductIDs;                // 0x0080 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class UNetworkEncryptionKey* Keys;                // 0x0090 (0x0008)
[0x000000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ReservationsReadyMessage_X");
}

return uClassPointer;
};

class UReservationsReadyMessage_X* SetKeys(class UNetworkEncryptionKey* InKeys);
class UReservationsReadyMessage_X* SetProductIDs(TArray<int32_t> InProductIDs);
class UReservationsReadyMessage_X* SetPingAddress(class FString InAddress);
class UReservationsReadyMessage_X* SetServerAddress(class FString InAddress);
};

// Class ProjectX.ReservationsTimedOutMessage_X
// 0x0000 (0x0060 - 0x0060)
class UReservationsTimedOutMessage_X : public UBeaconMessage_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ReservationsTimedOutMessage_X");
}
}

```

```

return uClassPointer;
};

};

// Class ProjectX.ReservationsPrivateMessage_X
// 0x0000 (0x0060 - 0x0060)
class UReservationsPrivateMessage_X : public UBeaconMessage_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ReservationsPrivateMessage_X");
}

return uClassPointer;
};

};

// Class ProjectX.ReservationsPasswordMessage_X
// 0x0001 (0x0060 - 0x0061)
class UReservationsPasswordMessage_X : public UBeaconMessage_X
{
public:
uint8_t Reason; // 0x0060 (0x0001)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ReservationsPasswordMessage_X");
}

return uClassPointer;
};

class UReservationsPasswordMessage_X* SetReason(uint8_t InReason);
};

// Class ProjectX.ReservationsWrongPlaylistMessage_X
// 0x0000 (0x0060 - 0x0060)
class UReservationsWrongPlaylistMessage_X : public UBeaconMessage_X
{

```

public:

public:

static UClass\* StaticClass()

{

static UClass\* uClassPointer = nullptr;

if (!uClassPointer)

{

uClassPointer = UObject::FindClass("Class ProjectX.ReservationsWrongPlaylistMessage\_X");

}

return uClassPointer;

};

};

// Class ProjectX.ReservationsWrongRankedMatchMessage\_X

// 0x0000 (0x0060 - 0x0060)

class UReservationsWrongRankedMatchMessage\_X : public UBeaconMessage\_X

{

public:

public:

static UClass\* StaticClass()

{

static UClass\* uClassPointer = nullptr;

if (!uClassPointer)

{

uClassPointer = UObject::FindClass("Class  
ProjectX.ReservationsWrongRankedMatchMessage\_X");

}

return uClassPointer;

};

};

// Class ProjectX.ReservationsKeysFailedMessage\_X

// 0x0000 (0x0060 - 0x0060)

class UReservationsKeysFailedMessage\_X : public UBeaconMessage\_X

{

public:

public:

static UClass\* StaticClass()

{

static UClass\* uClassPointer = nullptr;

if (!uClassPointer)

{

uClassPointer = UObject::FindClass("Class ProjectX.ReservationsKeysFailedMessage\_X");

}

```

return uClassPointer;
};

};

// Class ProjectX.ReservationCrossplayDisabledMessage_X
// 0x0000 (0x0060 - 0x0060)
class UReservationCrossplayDisabledMessage_X : public UBeaconMessage_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ReservationCrossplayDisabledMessage_X");
}

return uClassPointer;
};

};

// Class ProjectX.ReservationsMaxPlayersMessage_X
// 0x0004 (0x0060 - 0x0064)
class UReservationsMaxPlayersMessage_X : public UBeaconMessage_X
{
public:
int32_t MaxPlayerCount; // 0x0060 (0x0004)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ReservationsMaxPlayersMessage_X");
}

return uClassPointer;
};

class UReservationsMaxPlayersMessage_X* SetMaxPlayerCount(int32_t inMaxPlayerCount);
};

// Class ProjectX.PingMessage_X
// 0x0004 (0x0060 - 0x0064)
class UPingMessage_X : public UBeaconMessage_X

```

```

{
public:
    unsigned long                bIsResponse : 1;                // 0x0060 (0x0004)
    [0x0000000000000000] [0x00000001]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PingMessage_X");
        }

        return uClassPointer;
    };

    class UPingMessage_X* SetIsResponse();
};

// Class ProjectX.RPC_GetLeaderboardValueForUserBase_X
// 0x0064 (0x00E8 - 0x014C)
class URPC_GetLeaderboardValueForUserBase_X : public URPC_X
{
public:
    struct FUniqueNetId                PlayerID;                // 0x00E8 (0x0048)
    [0x0000000400040000] (CPF_NeedCtorLink)
    class FString                    LeaderboardId;                // 0x0130 (0x0010)
    [0x0000000400040200] (CPF_Transient | CPF_NeedCtorLink)
    unsigned long                bHasSkill : 1;                // 0x0140 (0x0004)
    [0x0000000400000200] [0x00000001] (CPF_Transient)
    unsigned long                bHasValue : 1;                // 0x0140 (0x0004)
    [0x0000000400000200] [0x00000002] (CPF_Transient)
    int32_t                        Value;                // 0x0144 (0x0004)
    [0x0000000400000200] (CPF_Transient)
    float                        MMR;                // 0x0148 (0x0004)
    [0x0000000400000200] (CPF_Transient)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetLeaderboardValueForUserBase_X");
        }

        return uClassPointer;
    };

    class URPC_GetLeaderboardValueForUserBase_X* SetPlayerID(struct FUniqueNetId InPlayerId);
};

```

```

// Class ProjectX.RPC_GetSkillLeaderboardValueForUser_X
// 0x0008 (0x014C - 0x0154)
class URPC_GetSkillLeaderboardValueForUser_X : public
URPC_GetLeaderboardValueForUserBase_X
{
public:
int32_t Playlist; // 0x0150 (0x0004)
[0x0000004000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetSkillLeaderboardValueForUser_X");
}

return uClassPointer;
};

class URPC_GetSkillLeaderboardValueForUser_X* SetPlaylist(int32_t InPlaylist);
};

// Class ProjectX.RPC_GetLeaderboardValueForUser_X
// 0x0014 (0x014C - 0x0160)
class URPC_GetLeaderboardValueForUser_X : public URPC_GetLeaderboardValueForUserBase_X
{
public:
class FString Stat; // 0x0150 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetLeaderboardValueForUser_X");
}

return uClassPointer;
};

class URPC_GetLeaderboardValueForUser_X* SetStat(class FString InStat);
};

// Class ProjectX.RPC_GetLeaderboardBase_X
// 0x0028 (0x00E8 - 0x0110)
class URPC_GetLeaderboardBase_X : public URPC_X
{

```

```

public:
    unsigned long                bDisableCrossPlay : 1;                // 0x00E8 (0x0004)
    [0x0000004000000000] [0x00000001]
    class FString                LeaderboardId;                        // 0x00F0 (0x0010)
    [0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FGetLeaderboardPlatformBaseData> Platforms;          // 0x0100
    (0x0010) [0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
    static UClass* uClassPointer = nullptr;

    if (!uClassPointer)
    {
    uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetLeaderboardBase_X");
    }

    return uClassPointer;
    };

class URPC_GetLeaderboardBase_X* SetDisableCrossPlay(unsigned long InDisableCrossplay);
};

// Class ProjectX.RPC_GetSkillLeaderboard_X
// 0x0004 (0x0110 - 0x0114)
class URPC_GetSkillLeaderboard_X : public URPC_GetLeaderboardBase_X
{
public:
    int32_t                    Playlist;                                // 0x0110 (0x0004)
    [0x0000004000000000]

public:
    static UClass* StaticClass()
    {
    static UClass* uClassPointer = nullptr;

    if (!uClassPointer)
    {
    uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetSkillLeaderboard_X");
    }

    return uClassPointer;
    };

class URPC_GetSkillLeaderboard_X* SetPlaylist(int32_t InPlaylist);
};

// Class ProjectX.RPC_GetLeaderboard_X
// 0x0010 (0x0110 - 0x0120)
class URPC_GetLeaderboard_X : public URPC_GetLeaderboardBase_X
{
public:
    class FString                Stat;                                // 0x0110 (0x0010)

```

[0x0000004000400000] (CPF\_NeedCtorLink)

```
public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetLeaderboard_X");
}

return uClassPointer;
};

class URPC_GetLeaderboard_X* SetStat(class FString InStat);
};

// Class ProjectX.RPC_GetLeaderboardRankForUsersBase_X
// 0x0030 (0x00E8 - 0x0118)
class URPC_GetLeaderboardRankForUsersBase_X : public URPC_X
{
public:
TArray<struct FUniqueNetId> PlayerIds; // 0x00E8 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
class FString LeaderboardId; // 0x00F8 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<struct FGetLeaderboardRankForUserData> Players; // 0x0108
(0x0010) [0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.RPC_GetLeaderboardRankForUsersBase_X");
}

return uClassPointer;
};

class URPC_GetLeaderboardRankForUsersBase_X* SetPlayerIDs(TArray<struct FUniqueNetId>
InPlayerIDs);
};

// Class ProjectX.RPC_GetSkillLeaderboardRankForUsers_X
// 0x0004 (0x0118 - 0x011C)
class URPC_GetSkillLeaderboardRankForUsers_X : public
URPC_GetLeaderboardRankForUsersBase_X
{
public:
```



```

int32_t                                Playlist;                                // 0x0118 (0x0004)
[0x0000004000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetSkillLeaderboardRankForUsers_X");
}

return uClassPointer;
};

class URPC_GetSkillLeaderboardRankForUsers_X* SetPlaylist(int32_t InPlaylist);
};

// Class ProjectX.RPC_GetLeaderboardRankForUsers_X
// 0x0010 (0x0118 - 0x0128)
class URPC_GetLeaderboardRankForUsers_X : public
URPC_GetLeaderboardRankForUsersBase_X
{
public:
class FString                                Stat;                                // 0x0118 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetLeaderboardRankForUsers_X");
}

return uClassPointer;
};

class URPC_GetLeaderboardRankForUsers_X* SetStat(class FString InStat);
};

// Class ProjectX.SuperRegion_X
// 0x0020 (0x0060 - 0x0080)
class USuperRegion_X : public UObject
{
public:
class FString                                Id;                                // 0x0060 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
class FString                                Label;                                // 0x0070 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.SuperRegion_X");
}

return uClassPointer;
};

};

// Class ProjectX.RPC_PlayerCancelMatchmaking_X
// 0x0000 (0x00E8 - 0x00E8)
class URPC_PlayerCancelMatchmaking_X : public URPC_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_PlayerCancelMatchmaking_X");
}

return uClassPointer;
};

};

// Class ProjectX.RPC_StartMatchmaking_X
// 0x0068 (0x00E8 - 0x0150)
class URPC_StartMatchmaking_X : public URPC_X
{
public:
TArray<struct FDSRegionInfo> Regions; // 0x00E8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
TArray<int32_t> Playlists; // 0x00F8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
int32_t SecondsSearching; // 0x0108 (0x0004)
[0x000000000000000000]
class FString CurrentServerId; // 0x0110 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
unsigned long bDisableCrossPlay : 1; // 0x0120 (0x0004)
[0x000000000000000000] [0x000000001]
class FString PartyID; // 0x0128 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
TArray<struct FUniqueNetId> PartyMembers; // 0x0138 (0x0010)

```

```

[0x000000000000400000] (CPF_NeedCtorLink)
float                BannedSecondsRemaining;                // 0x0148 (0x0004)
[0x00000000000002000] (CPF_Transient)
float                EstimatedQueueTime;                    // 0x014C (0x0004)
[0x00000000000002000] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_StartMatchmaking_X");
}

return uClassPointer;
};

class URPC_StartMatchmaking_X* SetIgnoreSkill(unsigned long bInIgnoreSkill);
class URPC_StartMatchmaking_X* SetPartyId(class FString InPartyId);
class URPC_StartMatchmaking_X* SetPartyMembers(TArray<struct FUniqueNetId>
InPartyMembers);
class URPC_StartMatchmaking_X* SetDisableCrossPlay(unsigned long bInDisableCrossplay);
class URPC_StartMatchmaking_X* SetCurrentServerID(class FString InCurrentServerID);
class URPC_StartMatchmaking_X* SetSecondsSearching(int32_t InSecondsSearching);
class URPC_StartMatchmaking_X* SetPlaylists(TArray<int32_t> InPlaylists);
class URPC_StartMatchmaking_X* SetRegions(TArray<struct FDSRegionInfo> InRegions);
};

// Class ProjectX.RankedConfig_X
// 0x0048 (0x0078 - 0x00C0)
class URankedConfig_X : public UOnlineConfig_X
{
public:
int32_t                ReconnectTimeoutSeconds;                // 0x0078 (0x0004)
[0x00000000000000001] (CPF_Edit)
TArray<int32_t>                SkillTierToSeasonRewardLevel;                // 0x0080 (0x0010)
[0x00000000000040001] (CPF_Edit | CPF_NeedCtorLink)
TArray<int32_t>                SeasonRewardRequiredWinsPerLevel;                // 0x0090
(0x0010) [0x00000000000040001] (CPF_Edit | CPF_NeedCtorLink)
unsigned long                bCheckReservationID : 1;                // 0x00A0 (0x0004)
[0x00000000000000001] [0x000000001] (CPF_Edit)
uint64_t                SeasonEndTimeSeconds;                // 0x00A8 (0x0008)
[0x00000000000000001] (CPF_Edit)
int32_t                MaximumRankDisparity;                // 0x00B0 (0x0004)
[0x00000000000000001] (CPF_Edit)
int32_t                PlacementMatchesNeededToReceiveRank;                // 0x00B4
(0x0004) [0x00000000000000001] (CPF_Edit)
int32_t                HighestTierNewPlayersCanPlayWith;                // 0x00B8 (0x0004)
[0x00000000000000001] (CPF_Edit)
int32_t                HighestMuNewPlayersCanHave;                // 0x00BC (0x0004)
[0x00000000000000001] (CPF_Edit)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RankedConfig_X");
}

return uClassPointer;
};

bool HasSeasonEnded();
int32_t GetSeasonTimeRemaining();
};

// Class ProjectX.RPC_GetPopulation_X
// 0x0010 (0x00E8 - 0x00F8)
class URPC_GetPopulation_X : public URPC_X
{
public:
TArray<struct FGetPopulationData>          Playlists;                                // 0x00E8 (0x0010)
[0x00000004000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetPopulation_X");
}

return uClassPointer;
};

};

// Class ProjectX.RPC_PlayerSearchPrivateMatch_X
// 0x0010 (0x00E8 - 0x00F8)
class URPC_PlayerSearchPrivateMatch_X : public URPC_X
{
public:
class FString                             Region;                                // 0x00E8 (0x0010)
[0x00000004000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)

```

```

{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_PlayerSearchPrivateMatch_X");
}

return uClassPointer;
};

class URPC_PlayerSearchPrivateMatch_X* SetRegion(class FString InRegion);
};

// Class ProjectX.RPC_GetGameServerPingList_X
// 0x0020 (0x00E8 - 0x0108)
class URPC_GetGameServerPingList_X : public URPC_X
{
public:
TArray<struct FRegionSecret> Regions; // 0x00E8 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
TArray<struct FGetGameServerPingListData> Servers; // 0x00F8
(0x0010) [0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetGameServerPingList_X");
}

return uClassPointer;
};

class URPC_GetGameServerPingList_X* SetRegionSecrets(TArray<struct FRegionSecret>&
InRegions);
void eventOnSuccess();
};

// Class ProjectX.RegionPinger_X
// 0x0058 (0x0070 - 0x00C8)
class URegionPinger_X : public UComponent
{
public:
float MaxPing; // 0x0070 (0x0004)
[0x00000000000000001] (CPF_Edit)
int32_t PingsPerRegion; // 0x0074 (0x0004)
[0x00000000000000001] (CPF_Edit)
float DelayBetweenPings; // 0x0078 (0x0004)
[0x00000000000000001] (CPF_Edit)
TArray<class URegionPingData_X*> ActivePings; // 0x0080
(0x0010) [0x0000000000400000] (CPF_NeedCtorLink)
class UUdpPingBeaconClient_X* PingBeacon; // 0x0090 (0x0008)
[0x00000000004080008] (CPF_ExportObject | CPF_Component | CPF_EditInline)
int32_t TickIndex; // 0x0098 (0x0004)

```

```

[0x0000000000000000]
TArray<class URegionPingData_X*> PingResults; // 0x00A0
(0x0010) [0x0000000000040000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventRegionsPinged__Delegate; // 0x00B0
(0x0018) [0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RegionPinger_X");
}

return uClassPointer;
};

class URegionPingData_X* __RegionPinger_X__PingRegions_0x1(class FString Address);
void HandlePong(class UUdpPingBeaconClient_X* _, struct FName Address, float DeltaSeconds);
void SendPing(int32_t Idx);
void Tick(float _);
void StopPingBeacon();
void StartPingBeacon();
void PingRegions(TArray<class FString>& Addresses);
void EventRegionsPinged(class URegionPinger_X* Pinger);
};

// Class ProjectX.RegionPingData_X
// 0x0018 (0x0060 - 0x0078)
class URegionPingData_X : public UObject
{
public:
struct FName Address; // 0x0060 (0x0008)
[0x0000000000000000]
int32_t PingsSent; // 0x0068 (0x0004)
[0x0000000000000000]
int32_t PingsReceived; // 0x006C (0x0004)
[0x0000000000000000]
float LowestPing; // 0x0070 (0x0004)
[0x0000000000000000]
float LastSendTime; // 0x0074 (0x0004)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RegionPingData_X");
}
}

```

```

return uClassPointer;
};

};

// Class ProjectX.OnlineGameReservations_AssignTeamsByParty_X
// 0x0000 (0x0060 - 0x0060)
class UOnlineGameReservations_AssignTeamsByParty_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class
ProjectX.OnlineGameReservations_AssignTeamsByParty_X");
}

return uClassPointer;
};

static int32_t PartySort(struct FPartyByTeam Left, struct FPartyByTeam Right);
static void AssignTeams(int32_t TeamSize, TArray<struct FReservationData>& TestPlayers);
};

// Class ProjectX.RegionPingMessage_X
// 0x0004 (0x0060 - 0x0064)
class URegionPingMessage_X : public UObject
{
public:
unsigned long                                bIsResponse : 1;                                // 0x0060 (0x0004)
[0x0000000000000000] [0x00000001]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RegionPingMessage_X");
}

return uClassPointer;
};

class URegionPingMessage_X* SetIsResponse();
};

```

```

// Class ProjectX.PsyNetService_ReservationBase_X
// 0x0020 (0x0090 - 0x00B0)
class UPsyNetService_ReservationBase_X : public UPsyNetClientService_X
{
public:
class FString                                ReservationID;                                // 0x0090 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
TArray<struct FPsyNetBeaconPlayerReservation>  Players;                                // 0x00A0
(0x0010) [0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_ReservationBase_X");
}

return uClassPointer;
};

struct FPsyNetBeaconReservation GetReservation();
};

// Class ProjectX.PsyNetService_NewGame_X
// 0x0028 (0x00B0 - 0x00D8)
class UPsyNetService_NewGame_X : public UPsyNetService_ReservationBase_X
{
public:
int32_t                                     Playlist;                                // 0x00B0 (0x0004)
[0x000000000000000000]
unsigned long                               IsBotMatch : 1;                                // 0x00B4 (0x0004)
[0x000000000000000000] [0x00000001]
TArray<class FString>                      BotNames;                                // 0x00B8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
TArray<struct FPsyNetBeaconPartyReservation> Parties;                                // 0x00C8
(0x0010) [0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_NewGame_X");
}

return uClassPointer;
};

struct FPsyNetBeaconReservation GetReservation();

```



```

};

// Class ProjectX.PsyNetService_Backfill_X
// 0x0018 (0x00B0 - 0x00C8)
class UPsyNetService_Backfill_X : public UPsyNetService_ReservationBase_X
{
public:
    int32_t Playlist; // 0x00B0 (0x0004)
    [0x0000000000000000]
    TArray<struct FPsyNetBeaconPartyReservation> Parties; // 0x00B8
    (0x0010) [0x0000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_Backfill_X");
        }

        return uClassPointer;
    };

    struct FPsyNetBeaconReservation GetReservation();
};

// Class ProjectX.PsyNetService_CreatePrivate_X
// 0x0000 (0x00B0 - 0x00B0)
class UPsyNetService_CreatePrivate_X : public UPsyNetService_ReservationBase_X
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_CreatePrivate_X");
        }

        return uClassPointer;
    };

    struct FPsyNetBeaconReservation GetReservation();
};

// Class ProjectX.PsyNetService_JoinPrivate_X
// 0x0000 (0x00B0 - 0x00B0)
class UPsyNetService_JoinPrivate_X : public UPsyNetService_ReservationBase_X
{

```

public:

public:

static UClass\* StaticClass()

{

static UClass\* uClassPointer = nullptr;

if (!uClassPointer)

{

uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService\_JoinPrivate\_X");

}

return uClassPointer;

};

};

// Class ProjectX.PsyNetService\_FriendJoin\_X

// 0x0000 (0x00B0 - 0x00B0)

class UPsyNetService\_FriendJoin\_X : public UPsyNetService\_ReservationBase\_X

{

public:

public:

static UClass\* StaticClass()

{

static UClass\* uClassPointer = nullptr;

if (!uClassPointer)

{

uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService\_FriendJoin\_X");

}

return uClassPointer;

};

};

// Class ProjectX.PsyNetService\_Reconnect\_X

// 0x0000 (0x00B0 - 0x00B0)

class UPsyNetService\_Reconnect\_X : public UPsyNetService\_ReservationBase\_X

{

public:

public:

static UClass\* StaticClass()

{

static UClass\* uClassPointer = nullptr;

if (!uClassPointer)

{

uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService\_Reconnect\_X");

}

```

return uClassPointer;
};

};

// Class ProjectX.RPC_GetPlayerSkill_X
// 0x00A8 (0x00E8 - 0x0190)
class URPC_GetPlayerSkill_X : public URPC_X
{
public:
    struct FUniqueNetId                PlayerID;                // 0x00E8 (0x0048)
    [0x0000000400040000] (CPF_NeedCtorLink)
    TArray<struct FPlayerSkillRating>    Skills;                // 0x0130 (0x0010)
    [0x0000000400040200] (CPF_Transient | CPF_NeedCtorLink)
    struct FPlayerSeasonRewardProgress    RewardLevels;        // 0x0140
    (0x0050) [0x0000000400040200] (CPF_Transient | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetPlayerSkill_X");
        }

        return uClassPointer;
    };

    class URPC_GetPlayerSkill_X* SetPlayerID(struct FUniqueNetId InPlayerId);
    void eventOnSuccess();
    void AddPlayerIDToResponse();
};

// Class ProjectX.RPC_GetPartyMemberSkill_X
// 0x0020 (0x00E8 - 0x0108)
class URPC_GetPartyMemberSkill_X : public URPC_X
{
public:
    TArray<struct FUniqueNetId>          PlayerIds;            // 0x00E8 (0x0010)
    [0x0000000400040000] (CPF_NeedCtorLink)
    TArray<struct FPartyMemberSkill>     Players;            // 0x00F8 (0x0010)
    [0x0000000400040200] (CPF_Transient | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetPartyMemberSkill_X");
        }
    }

```

```

return uClassPointer;
};

void eventOnSuccess();
void AddPlayerIDsToResponse();
class URPC_GetPartyMemberSkill_X* AddPlayer(struct FUniqueNetId PlayerID);
};

// Class ProjectX.SyncedSkillData_X
// 0x0008 (0x0060 - 0x0068)
class USyncedSkillData_X : public UObject
{
public:
class UOnlineGameSkill_X*      OnlineGameSkill;          // 0x0060 (0x0008)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.SyncedSkillData_X");
}

return uClassPointer;
};

};

// Class ProjectX.RPC_RecordMatchResults_X
// 0x008C (0x00E8 - 0x0174)
class URPC_RecordMatchResults_X : public URPC_X
{
public:
class FString      MatchGuid;          // 0x00E8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FSkillMatchData      Match;          // 0x00F8 (0x0028)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString      MatchName;          // 0x0120 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
TArray<struct FSkillMatchPlayer>      Winners;          // 0x0130 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
TArray<struct FSkillMatchPlayer>      Losers;          // 0x0140 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FGuid      AppSessionID;          // 0x0150 (0x0010)
[0x000000000000000000]
struct FGuid      LevelSessionID;          // 0x0160 (0x0010)
[0x000000000000000000]
unsigned long      blsBotMatch : 1;          // 0x0170 (0x0004)
[0x000000000000000000] [0x00000001]

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_RecordMatchResults_X");
}

return uClassPointer;
};

int32_t GetPartyIndex(struct FUniqueNetId LeaderID, TArray<struct FUniqueNetId>& PartyIDs);
void InitSkillMatchPlayers(class UMatchData_X* MatchData);
struct FSkillMatchPlayer InitSkillMatchPlayer(class UMatchData_X* MatchData, class
UMatchPlayerData_X* Player, TArray<struct FUniqueNetId>& PartyIDs);
class URPC_RecordMatchResults_X* SetMetrics(class UMetricsSystem_X* MetricsSystem);
class URPC_RecordMatchResults_X* SetMatchName(class FString InName);
class URPC_RecordMatchResults_X* SetMatch(class UMatchData_X* InMatch);
class URPC_RecordMatchResults_X* SetIsBotMatch(unsigned long bBotMatch);
class URPC_RecordMatchResults_X* SetMatchGUID(class FString InGuid);
};

// Class ProjectX.RPC_UpdateSkills_X
// 0x0034 (0x0174 - 0x01A8)
class URPC_UpdateSkills_X : public URPC_RecordMatchResults_X
{
public:
int32_t Playlist; // 0x0178 (0x0004)
[0x0000000000000000]
TArray<struct FUpdatedPlayerSkillRating> NewSkills; // 0x0180
(0x0010) [0x0000000000040200] (CPF_Transient | CPF_NeedCtorLink)
TArray<struct FPlayerSeasonRewardProgress> NewRewardLevels; //
0x0190 (0x0010) [0x0000000000040200] (CPF_Transient | CPF_NeedCtorLink)
class UMatchData_X* MatchData; // 0x01A0 (0x0008)
[0x0000000400000200] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_UpdateSkills_X");
}

return uClassPointer;
};

class URPC_RecordMatchResults_X* SetMatch(class UMatchData_X* InMatch);
};

```

```

// Class ProjectX.RPC_UpdateLeaderboard_X
// 0x0024 (0x00E8 - 0x010C)
class URPC_UpdateLeaderboard_X : public URPC_X
{
public:
TArray<struct FUploadStatDataSet>          Updates;                      // 0x00E8 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
class FString                            MatchGuid;                    // 0x00F8 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
int32_t                                PlaylistId;                      // 0x0108 (0x0004)
[0x0000004000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_UpdateLeaderboard_X");
}

return uClassPointer;
};

class URPC_UpdateLeaderboard_X* SetPlaylistID(int32_t InID);
class URPC_UpdateLeaderboard_X* SetMatchGUID(class FString InGuid);
class URPC_UpdateLeaderboard_X* SetUpdates(TArray<struct FUploadStatDataSet>&
InUpdates);
};

// Class ProjectX.PsyNetWordFilter_X
// 0x0010 (0x0070 - 0x0080)
class UPsyNetWordFilter_X : public UComponent
{
public:
class UWordFilterConfig_X*              Config;                        // 0x0070 (0x0008)
[0x0000800000000001] (CPF_Edit)
class URPC_FilterContent_X*              PendingFilterRPC;              // 0x0078 (0x0008)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetWordFilter_X");
}

return uClassPointer;
};

```

```

void __PsyNetWordFilter_X__WordFilterSanitizeString_0x1(class URPC_FilterContent_X* RPC);
void HandlePsyNetWordFilterFail(class URPC_FilterContent_X* RPC);
void HandlePsyNetWordFilterSuccess(class URPC_FilterContent_X* RPC);
void HandlePsyNetWordFilter(class URPC_FilterContent_X* RPC);
void SendPendingFilterRPC();
bool WordFilterSanitizeString(class FString Comment, struct FScriptDelegate SanitizeDelegate,
struct FUniqueNetId PlayerID);
};

```

```

// Class ProjectX.WordFilterConfig_X
// 0x0030 (0x0078 - 0x00A8)
class UWordFilterConfig_X : public UOnlineConfig_X
{
public:
    unsigned long                bSanitizeEntirePhrase : 1;                // 0x0078 (0x0004)
    [0x0000000000000001] [0x00000001] (CPF_Edit)
    unsigned long                PsyNetFilterEnabled : 1;                // 0x0078 (0x0004)
    [0x0000000000000001] [0x00000002] (CPF_Edit)
    int32_t                      NameHistoryCacheLength;                // 0x007C (0x0004)
    [0x0000000000000001] (CPF_Edit)
    TArray<uint8_t>              IgnoreFilterList;                // 0x0080 (0x0010)
    [0x00000000000404001] (CPF_Edit | CPF_Config | CPF_NeedCtorLink)
    TArray<uint8_t>              PlatformFilterList;                // 0x0090 (0x0010)
    [0x00000000000404001] (CPF_Edit | CPF_Config | CPF_NeedCtorLink)
    float                        PsyNetWordFilterBatchDelay;            // 0x00A0 (0x0004)
    [0x0000000000000001] (CPF_Edit)
    int32_t                      PsyNetWordFilterBatchMaxSize;            // 0x00A4 (0x0004)
    [0x0000000000000001] (CPF_Edit)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.WordFilterConfig_X");
        }

        return uClassPointer;
    };
};

```

```

// Class ProjectX.UserBugReportComponent_X
// 0x0000 (0x0060 - 0x0060)
class UUserBugReportComponent_X : public UObject
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.UserBugReportComponent_X");
}

return uClassPointer;
};

void Start(class FString Message);
};

// Class ProjectX.OnlinePlayerRegionRestrictions_X
// 0x0010 (0x00B0 - 0x00C0)
class UOnlinePlayerRegionRestrictions_X : public UOnline_X
{
public:
    TArray<uint8_t> Restrictions; // 0x00B0 (0x0010)
    [0x00010000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.OnlinePlayerRegionRestrictions_X");
        }

        return uClassPointer;
    };

    uint8_t __OnlinePlayerRegionRestrictions_X__HandleLoginChanged_0x1(struct FName S);
    bool IsRestricted(uint8_t Restriction);
    void HandleLoginChanged(class UOnlinePlayerAuthentication_X* Auth);
    void OnExit();
    void OnInit();
};

// Class ProjectX.OnlinePlayerStorage_X
// 0x0018 (0x00B0 - 0x00C8)
class UOnlinePlayerStorage_X : public UOnline_X
{
public:
    class UOnlinePlayerStorageQueue_X* Queue; // 0x00B0 (0x0008)
    [0x0001004000000000]
    class UOnlinePlayerStorageSync_X* Sync; // 0x00B8 (0x0008)
    [0x0001004000000000]
    class UOnlinePlayerStorageManifest_X* Manifest; // 0x00C0
    (0x0008) [0x0001004000000000]

public:
    static UClass* StaticClass()

```



```

{
static UClass* UClassPointer = nullptr;

if (!UClassPointer)
{
UClassPointer = UObject::FindClass("Class ProjectX.OnlinePlayerStorage_X");
}

return UClassPointer;
};

void OnInit();
};

// Class ProjectX.RPC_LoginAuthPlayer_X
// 0x011C (0x00E8 - 0x0204)
class URPC_LoginAuthPlayer_X : public URPC_X
{
public:
class FString Platform; // 0x00E8 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
class FString PlayerName; // 0x00F8 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
class FString PlayerID; // 0x0108 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
class FString Language; // 0x0118 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
class FString AuthTicket; // 0x0128 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
class FString BuildRegion; // 0x0138 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
struct FName FeatureSet; // 0x0148 (0x0008)
[0x0000004000000000]
class FString Device; // 0x0150 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
class FString LocalFirstPlayerID; // 0x0160 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
unsigned long bSkipAuth : 1; // 0x0170 (0x0004)
[0x0001004000000000] [0x00000001]
unsigned long bSetAsPrimaryAccount : 1; // 0x0170 (0x0004)
[0x0000004000000000] [0x00000002]
unsigned long UseWebSocket : 1; // 0x0170 (0x0004)
[0x0000004000002000] [0x00000004] (CPF_Transient)
unsigned long IsLastChanceAuthBan : 1; // 0x0170 (0x0004)
[0x0000004000002000] [0x00000008] (CPF_Transient)
class FString EpicAuthTicket; // 0x0178 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
class FString EpicAccountId; // 0x0188 (0x0010)
[0x0000004000400000] (CPF_NeedCtorLink)
class FString SessionId; // 0x0198 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString PsyToken; // 0x01A8 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class UBanMessage_X* BanMessage; // 0x01B8 (0x0008)

```

```

[0x0000004000002000] (CPF_Transient)
class FString                               VerifiedPlayerName;                // 0x01C0 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                               PerConURL;                        // 0x01D0 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
class FString                               PerConURLv2;                          // 0x01E0 (0x0010)
[0x0000004000402000] (CPF_Transient | CPF_NeedCtorLink)
TArray<struct FName>                        CountryRestrictions;                // 0x01F0 (0x0010)
[0x0001004000402000] (CPF_Transient | CPF_NeedCtorLink)
int32_t                                     EulaResetCounter;                  // 0x0200 (0x0004)
[0x0000004000002000] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_LoginAuthPlayer_X");
}

return uClassPointer;
};

class URPC_LoginAuthPlayer_X* SetAsPrimaryAccount(unsigned long bInPrimary);
class URPC_LoginAuthPlayer_X* SetEpicAccountId(class FString InEpicAccountId);
class URPC_LoginAuthPlayer_X* SetEpicAuthTicket(class FString InEpicAuthTicket);
class URPC_LoginAuthPlayer_X* SetLocalFirstPlayerID(class FString InLocalFirstPlayerID);
class URPC_LoginAuthPlayer_X* SetConsoleType(class FString InType);
class URPC_LoginAuthPlayer_X* SetFeatureSet(struct FName InFeatureSet);
class URPC_LoginAuthPlayer_X* SetBuildRegion(class FString InBuildRegion);
class URPC_LoginAuthPlayer_X* SetAuthTicket(class FString InAuthTicket);
class URPC_LoginAuthPlayer_X* SetLanguage(class FString InLanguage);
class URPC_LoginAuthPlayer_X* SetPlayerID(class FString InPlayerId);
class URPC_LoginAuthPlayer_X* SetPlayerName(class FString InPlayerName);
class URPC_LoginAuthPlayer_X* SetPlatform(class FString InPlatform);
};

// Class ProjectX.RPC_GetPlayerPermissions_X
// 0x0020 (0x00E8 - 0x0108)
class URPC_GetPlayerPermissions_X : public URPC_X
{
public:
TArray<struct FUniqueNetId>                 PlayerIds;                          // 0x00E8 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
TArray<struct FPlayerPermissionsReponse>    PlayerPermissions;                  // 0x00F8
(0x0010) [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetPlayerPermissions_X");
}

return uClassPointer;
};

bool GetPermissionFromString(struct FName PermissionName, uint8_t& Permission);
TArray<uint8_t> ConvertPermissions(TArray<struct FName> PermissionNames);
struct FPlayerPermissionsList ConvertPlayerPermissions(struct FUniqueNetId PlayerID);
TArray<struct FPlayerPermissionsList> GetPlayerPermissions();
class URPC_GetPlayerPermissions_X* SetPlayers(TArray<struct FUniqueNetId>& InPlayerIDs);
};

// Class ProjectX.OnlinePlayerStorageManifest_X
// 0x0010 (0x0060 - 0x0070)
class UOnlinePlayerStorageManifest_X : public UObject
{
public:
    TArray<struct FStorageMetadata> Metadata; // 0x0060 (0x0010)
    [0x0001000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.OnlinePlayerStorageManifest_X");
        }

        return uClassPointer;
    };

    bool ChecksumMatches(struct FName Category, int32_t Checksum, uint8_t Encoding);
    void SetChecksum(struct FName Category, int32_t Checksum, uint8_t Encoding);
    int32_t GetIndex(struct FName Category);
};

// Class ProjectX.OnlinePlayerStorageConfig_X
// 0x0004 (0x0078 - 0x007C)
class UOnlinePlayerStorageConfig_X : public UOnlineConfig_X
{
public:
    float QueueBatchDelay; // 0x0078 (0x0004)
    [0x0001000000000001] (CPF_Edit)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;
    }
};

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.OnlinePlayerStorageConfig_X");
}

return uClassPointer;
};

};

// Class ProjectX.SaveLock_X
// 0x0008 (0x0060 - 0x0068)
class USaveLock_X : public UObject
{
public:
    class UGfxEngine_X*                GfxEngine;                // 0x0060 (0x0008)
    [0x0000800000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.SaveLock_X");
        }

        return uClassPointer;
    };

    static bool StaticAllowSave();
    bool AllowSave();
};

// Class ProjectX.RPC_PlayerStorageGet_X
// 0x0060 (0x00E8 - 0x0148)
class URPC_PlayerStorageGet_X : public URPC_X
{
public:
    struct FUniqueNetId                PlayerID;                // 0x00E8 (0x0048)
    [0x0001000000400000] (CPF_NeedCtorLink)
    TArray<struct FGetPlayerStorageRequestItem>    Items;                // 0x0130
    (0x0010) [0x0001000000400000] (CPF_NeedCtorLink)
    class UGetPlayerStorageResult_X*    Result;                // 0x0140 (0x0008)
    [0x0001000000002000] (CPF_Transient)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {

```

```

uClassPointer = UObject::FindClass("Class ProjectX.RPC_PlayerStorageGet_X");
}

return uClassPointer;
};

void __RPC_PlayerStorageGet_X__OnSuccess_0x2(class UGetPlayerStorageResultItem_X* Item);
bool __RPC_PlayerStorageGet_X__OnSuccess_0x1(class UGetPlayerStorageResultItem_X* Item);
void eventOnSuccess();
class UObject* eventGetResponseObject();
};

// Class ProjectX.OnlineResource_X
// 0x00A0 (0x0070 - 0x0110)
class UOnlineResource_X : public UComponent
{
public:
class FString URL; // 0x0070 (0x0010)
[0x0000000000400001] (CPF_Edit | CPF_NeedCtorLink)
TArray<float> RetryDelays; // 0x0080 (0x0010)
[0x0000000000400001] (CPF_Edit | CPF_NeedCtorLink)
float CacheSeconds; // 0x0090 (0x0004)
[0x0000000000000001] (CPF_Edit)
unsigned long bZipResponse : 1; // 0x0094 (0x0004)
[0x0000000000000001] [0x00000001] (CPF_Edit)
unsigned long bSyncing : 1; // 0x0094 (0x0004)
[0x0000004000002000] [0x00000002] (CPF_Transient)
class UStringMap* Headers; // 0x0098 (0x0008)
[0x0000000004080009] (CPF_Edit | CPF_ExportObject | CPF_Component | CPF_EditInline)
class UCachedWebData_X* CachedData; // 0x00A0 (0x0008)
[0x0000004000002000] (CPF_Transient)
int32_t RetryCount; // 0x00A8 (0x0004)
[0x0000000000002000] (CPF_Transient)
float LastSyncTime; // 0x00AC (0x0004)
[0x0000000000002000] (CPF_Transient)
class UPsyNetRetryConfig_X* RetryConfig; // 0x00B0 (0x0008)
[0x0000800000000000]
class FString ServiceName; // 0x00B8 (0x0010)
[0x0000000000400002] (CPF_Const | CPF_NeedCtorLink)
struct FScriptDelegate __EventSyncComplete__Delegate; // 0x00C8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventSyncError__Delegate; // 0x00E0
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate __EventDataChanged__Delegate; // 0x00F8
(0x0018) [0x0000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.OnlineResource_X");

```

```

}

return uClassPointer;
};

void SetCachedData(class UCachedWebData_X* NewData);
bool IsValidNewData(class UCachedWebData_X* Data);
void ClearRetryTimer();
void SetRetryTimer(float Delay);
void UpdateRetryDelays(class UError* SyncError);
void HandleRetryConfigChanged();
void HandleSync(class UCachedWebData_X* Data);
void RetrySync();
void SyncData(struct FScriptDelegate Callback);
void EventDataChanged(class UOnlineResource_X* DataSync);
void EventSyncError(class UOnlineResource_X* DataSync, class UError* Error);
void EventSyncComplete(class UOnlineResource_X* DataSync);
};

// Class ProjectX.OnlineSaveLock_X
// 0x0004 (0x0060 - 0x0064)
class UOnlineSaveLock_X : public UObject
{
public:
    int32_t EnableCount; // 0x0060 (0x0004)
    [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.OnlineSaveLock_X");
        }

        return uClassPointer;
    };

    static void HandleEnableOnlineSaveRemoved(class UEnableOnlineSave_X* Enable);
    static void HandleEnableOnlineSaveAdded(class UEnableOnlineSave_X* Enable);
    static bool StaticAllowSave();
    bool AllowSave();
};

// Class ProjectX.RPC_PartyBase_X
// 0x0010 (0x00E8 - 0x00F8)
class URPC_PartyBase_X : public URPC_X
{
public:
    class FString PartyID; // 0x00E8 (0x0010)
    [0x00000000000040000] (CPF_NeedCtorLink)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_PartyBase_X");
}

return uClassPointer;
};

class URPC_PartyBase_X* SetPartyIdStr(class FString InPartyId);
class URPC_PartyBase_X* SetPartyId(struct FUniqueLobbyId InLobbyId);
};

// Class ProjectX.RPC_PartyMessage_X
// 0x0010 (0x00F8 - 0x0108)
class URPC_PartyMessage_X : public URPC_PartyBase_X
{
public:
class FString                                     Message;                                     // 0x00F8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_PartyMessage_X");
}

return uClassPointer;
};

class URPC_PartyMessage_X* SetMessage(class FString InMessage);
};

// Class ProjectX.PartySequence_InviteToParty_X
// 0x0000 (0x0060 - 0x0060)
class UPartySequence_InviteToParty_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{

```

```

uClassPointer = UObject::FindClass("Class ProjectX.PartySequence_InviteToParty_X");
}

return uClassPointer;
};

void __PartySequence_InviteToParty_X__InviteToPsyNetParty_0x2(class
URPC_PartySendInvite_X* RPC);
void __PartySequence_InviteToParty_X__InviteToPsyNetParty_0x1(class
URPC_PartySendInvite_X* RPC);
bool InviteToPsyNetParty(struct FUniqueLobbyId& LobbyId, struct FUniqueNetId& PlayerID);
bool InviteToParty(struct FUniqueLobbyId& LobbyId, struct FUniqueNetId& PlayerID);
};

// Class ProjectX.PartySequence_LeaveParty_X
// 0x0000 (0x0060 - 0x0060)
class UPartySequence_LeaveParty_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PartySequence_LeaveParty_X");
}

return uClassPointer;
};

void LeaveParty(struct FUniqueLobbyId& LobbyId);
};

// Class ProjectX.RPC_PartyChat_X
// 0x0010 (0x00F8 - 0x0108)
class URPC_PartyChat_X : public URPC_PartyBase_X
{
public:
class FString                                Message;                                // 0x00F8 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_PartyChat_X");
}
}

```



```

return uClassPointer;
};

class URPC_PartyChat_X* SetText(class FString InText);
};

// Class ProjectX.RPC_PartyChangeOwner_X
// 0x00C8 (0x00F8 - 0x01C0)
class URPC_PartyChangeOwner_X : public URPC_PartyBase_X
{
public:
    struct FUniqueNetId                NewOwnerId;                // 0x00F8 (0x0048)
    [0x0000000000040000] (CPF_NeedCtorLink)
    struct FPsyNetPartyInfo            Info;                      // 0x0140 (0x0070)
    [0x0001000000402000] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FPsyNetPartyMember>    Members;                  // 0x01B0
    (0x0010) [0x0001000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_PartyChangeOwner_X");
        }

        return uClassPointer;
    };

    class URPC_PartyChangeOwner_X* SetNewOwnerId(struct FUniqueNetId InNewOwnerId);
};

// Class ProjectX.RPC_PartyKickMember_X
// 0x0011 (0x00F8 - 0x0109)
class URPC_PartyKickMember_X : public URPC_PartyBase_X
{
public:
    TArray<class FString>                Members;                // 0x00F8 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)
    uint8_t                               KickReason;            // 0x0108 (0x0001)
    [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_PartyKickMember_X");
        }
    }

```

```

return uClassPointer;
};

class URPC_PartyKickMember_X* SetReason(uint8_t InReason);
class URPC_PartyKickMember_X* AddMember(class FString InMemberId);
};

// Class ProjectX.RPC_PartyInfo_X
// 0x0090 (0x00E8 - 0x0178)
class URPC_PartyInfo_X : public URPC_X
{
public:
    TArray<struct FPartyInvite> Invites; // 0x00E8 (0x0010)
    [0x0001000000402000] (CPF_Transient | CPF_NeedCtorLink)
    struct FPsyNetPartyMember Member; // 0x00F8 (0x0080)
    [0x0001000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_PartyInfo_X");
        }

        return uClassPointer;
    };
};

// Class ProjectX.PsyNetService_PartySystem_X
// 0x0000 (0x0168 - 0x0168)
class UPsyNetService_PartySystem_X : public UPsyNetService_Party_X
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_PartySystem_X");
        }

        return uClassPointer;
    };
};

// Class ProjectX.PsyNetService_PartyUserJoined_X

```

```

// 0x0000 (0x0168 - 0x0168)
class UPsyNetService_PartyUserJoined_X : public UPsyNetService_Party_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_PartyUserJoined_X");
}

return uClassPointer;
};

};

// Class ProjectX.PsyNetService_PartyUserKicked_X
// 0x0101 (0x0090 - 0x0191)
class UPsyNetService_PartyUserKicked_X : public UPsyNetClientService_X
{
public:
class FString PartyID; // 0x0090 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString NotificationType; // 0x00A0 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FUniqueNetId Content; // 0x00B0 (0x0048)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FUniqueNetId FromUserId; // 0x00F8 (0x0048)
[0x000000000000400000] (CPF_NeedCtorLink)
int32_t CreatedAt; // 0x0140 (0x0004)
[0x000000000000000000]
struct FUniqueNetId ForUserId; // 0x0148 (0x0048)
[0x000000000000400000] (CPF_NeedCtorLink)
uint8_t KickReason; // 0x0190 (0x0001)
[0x000000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_PartyUserKicked_X");
}

return uClassPointer;
};

};

```

```

// Class ProjectX.PsyNetService_PartyUserLeft_X
// 0x0000 (0x0168 - 0x0168)
class UPsyNetService_PartyUserLeft_X : public UPsyNetService_Party_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_PartyUserLeft_X");
}

return uClassPointer;
};

};

// Class ProjectX.PsyNetService_PartyUserDisconnected_X
// 0x0000 (0x0168 - 0x0168)
class UPsyNetService_PartyUserDisconnected_X : public UPsyNetService_Party_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_PartyUserDisconnected_X");
}

return uClassPointer;
};

};

// Class ProjectX.PsyNetService_PartyOwnerChanged_X
// 0x0100 (0x0090 - 0x0190)
class UPsyNetService_PartyOwnerChanged_X : public UPsyNetClientService_X
{
public:
class FString PartyID; // 0x0090 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
class FString NotificationType; // 0x00A0 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
struct FUniqueNetId Content; // 0x00B0 (0x0048)
[0x000000000000400000] (CPF_NeedCtorLink)

```

```

struct FUniqueNetId                FromUserId;                // 0x00F8 (0x0048)
[0x000000000000400000] (CPF_NeedCtorLink)
int32_t                            CreatedAt;                // 0x0140 (0x0004)
[0x000000000000000000]
struct FUniqueNetId                ForUserId;                // 0x0148 (0x0048)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_PartyOwnerChanged_X");
}

return uClassPointer;
};

};

// Class ProjectX.PsyNetService_PartyChat_X
// 0x0000 (0x0168 - 0x0168)
class UPsyNetService_PartyChat_X : public UPsyNetService_Party_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_PartyChat_X");
}

return uClassPointer;
};

};

// Class ProjectX.PartyMessage_GetPlatformPartyResponse_X
// 0x0058 (0x00A8 - 0x0100)
class UPartyMessage_GetPlatformPartyResponse_X : public UPartyMessage_X
{
public:
struct FUniqueLobbyId                PlatformPartyID;                // 0x00A8 (0x0010)
[0x000000000000000000]
struct FUniqueNetId                OriginalSender;                // 0x00B8 (0x0048)
[0x000000000000400000] (CPF_NeedCtorLink)

public:

```

```

static UClass* StaticClass()
{
    static UClass* uClassPointer = nullptr;

    if (!uClassPointer)
    {
        uClassPointer = UObject::FindClass("Class
        ProjectX.PartyMessage_GetPlatformPartyResponse_X");
    }

    return uClassPointer;
};

};

// Class ProjectX.RPC_PartySendInvite_X
// 0x0048 (0x00F8 - 0x0140)
class URPC_PartySendInvite_X : public URPC_PartyBase_X
{
public:
    struct FUniqueNetId                InviteeID;                // 0x00F8 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_PartySendInvite_X");
        }

        return uClassPointer;
    };

    class URPC_PartySendInvite_X* Invite(struct FUniqueNetId InInviteeUserId);
};

// Class ProjectX.RPC_PartyJoin_X
// 0x0090 (0x00F8 - 0x0188)
class URPC_PartyJoin_X : public URPC_PartyBase_X
{
public:
    class FString                    JoinID;                    // 0x00F8 (0x0010)
    [0x000000000000400000] (CPF_NeedCtorLink)
    struct FPsyNetPartyInfo          Info;                    // 0x0108 (0x0070)
    [0x000100000000402000] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FPsyNetPartyMember> Members;                // 0x0178
    (0x0010) [0x000100000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {

```

```

static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.RPC_PartyJoin_X");
}

return uClassPointer;
};

class URPC_PartyBase_X* SetJoinKey(class FString InJoinKey);
};

// Class ProjectX.RPC_PartyLeave_X
// 0x0000 (0x00F8 - 0x00F8)
class URPC_PartyLeave_X : public URPC_PartyBase_X
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_PartyLeave_X");
        }

        return uClassPointer;
    };
};

// Class ProjectX.PauseStaticDataSync_X
// 0x0000 (0x0060 - 0x0060)
class UPauseStaticDataSync_X : public UObject
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PauseStaticDataSync_X");
        }

        return uClassPointer;
    };
};

```

```

// Class ProjectX.PerConMetrics_X
// 0x001C (0x0080 - 0x009C)
class UPerConMetrics_X : public UMetricsGroup_X
{
public:
    int32_t                StartConnectFailCount;                // 0x0080 (0x0004)
    [0x0000000000000000]
    int32_t                ConnectFailCount;                    // 0x0084 (0x0004)
    [0x0000000000000000]
    int32_t                DisconnectCount;                    // 0x0088 (0x0004)
    [0x0000000000000000]
    int32_t                RPCErrorCount;                    // 0x008C (0x0004)
    [0x0000000000000000]
    int32_t                InvalidMessageCount;                // 0x0090 (0x0004)
    [0x0000000000000000]
    unsigned long          bConnected : 1;                    // 0x0094 (0x0004)
    [0x0000000000000000] [0x00000001]
    float                  StartTime;                        // 0x0098 (0x0004)
    [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PerConMetrics_X");
        }

        return uClassPointer;
    };

    void ServiceError(struct FUniqueNetId PlayerID, int32_t Count, class FString Service, struct FName Error);
    void InvalidMessageError(struct FUniqueNetId PlayerID, int32_t Count, class FString Substring);
    void DisconnectedError(struct FUniqueNetId PlayerID, int32_t Count, int32_t Code, class FString Reason, float ElapsedSeconds);
    void Connected(struct FUniqueNetId PlayerID, float ElapsedSeconds);
    void ConnectError(struct FUniqueNetId PlayerID, int32_t Count, int32_t Code, class FString Reason, float ElapsedSeconds);
    void StartConnectError(struct FUniqueNetId PlayerID, int32_t Count);
    void StartConnect(struct FUniqueNetId PlayerID);
    void Disabled(struct FUniqueNetId PlayerID);
    void Enabled(struct FUniqueNetId PlayerID);
    void HandleRPCError(class URPCQueue_X* InQueue, class URPC_X* RPC, class UError* Error);
    void HandleInvalidMessage(class UPsyNetMessageerWebSocket_X* InMessenger, class FString Substring);
    void HandleDisconnected(class UPsyNetMessageerWebSocket_X* Socket, int32_t Code, class FString Reason);
    void HandleConnected(class UPsyNetMessageerWebSocket_X* Socket);
    void HandleStartConnectFail(class UPsyNetMessageerWebSocket_X* Socket);
    void HandleStartConnect(class UPsyNetMessageerWebSocket_X* Socket);

```



```
void HandleStatusChanged(class UPerCon_X* PerCon);
struct FUniqueNetId GetPlayerID();
void SetMessenger(class UPsyNetMessageWebSocket_X* PerConMessenger);
void eventConstruct();
};
```

```
// Class ProjectX.Platform_Console
// 0x0000 (0x0060 - 0x0060)
class UPlatform_Console : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Platform_Console");
}

return uClassPointer;
};

};
```

```
// Class ProjectX.Platform_DedicatedServer
// 0x0000 (0x0060 - 0x0060)
class UPlatform_DedicatedServer : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Platform_DedicatedServer");
}

return uClassPointer;
};

};
```

```
// Class ProjectX.Platform_Dingo
// 0x0000 (0x0060 - 0x0060)
class UPlatform_Dingo : public UObject
{
public:
```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Platform_Dingo");
}

return uClassPointer;
};

};

// Class ProjectX.Platform_GameClient
// 0x0000 (0x0060 - 0x0060)
class UPlatform_GameClient : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Platform_GameClient");
}

return uClassPointer;
};

};

// Class ProjectX.Platform_NNX
// 0x0000 (0x0060 - 0x0060)
class UPlatform_NNX : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Platform_NNX");
}

return uClassPointer;
};

```

```

};

// Class ProjectX.Platform_Orbis
// 0x0000 (0x0060 - 0x0060)
class UPlatform_Orbis : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Platform_Orbis");
}

return uClassPointer;
};

};

// Class ProjectX.Platform_PC
// 0x0000 (0x0060 - 0x0060)
class UPlatform_PC : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.Platform_PC");
}

return uClassPointer;
};

};

// Class ProjectX.Platform_Server
// 0x0000 (0x0060 - 0x0060)
class UPlatform_Server : public UObject
{
public:

public:
static UClass* StaticClass()
{

```

```

static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.Platform_Server");
}

return uClassPointer;
};

};

// Class ProjectX.PlatformSystem_X
// 0x0000 (0x0060 - 0x0060)
class UPlatformSystem_X : public UObject
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PlatformSystem_X");
        }

        return uClassPointer;
    };

    static void AddPlatforms(TArray<class UClass*>& Platforms);
    static void AddServer(class UClass* PlatformClass);
    static void AddConsole(class UClass* PlatformClass);
    static void Init(class UGameEngine* Engine);
};

// Class ProjectX.PlayerTitleConfig_X
// 0x0030 (0x0078 - 0x00A8)
class UPlayerTitleConfig_X : public UOnlineConfig_X
{
public:
    TArray<struct FPlayerTitleCategory> Categories; // 0x0078 (0x0010)
    [0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
    TArray<struct FPlayerTitleData> Titles; // 0x0088 (0x0010)
    [0x0000000000040001] (CPF_Edit | CPF_NeedCtorLink)
    class FString DefaultColorHexCode; // 0x0098 (0x0010)
    [0x0000000000040002] (CPF_Const | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

```

```

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.PlayerTitleConfig_X");
}

return uClassPointer;
};

struct FPlayerTitleData InitTitleColors(struct FPlayerTitleData Data);
struct FPlayerTitleData GetTitleData(struct FName TitleId);
struct FPlayerTitleCategory GetCategory(struct FName CategoryID);
void Apply();
};

// Class ProjectX.PostProcessManager_X
// 0x0070 (0x0060 - 0x00D0)
class UPostProcessManager_X : public UObject
{
public:
    class ULocalPlayer*          PlayerOwner;                // 0x0060 (0x0008)
    [0x00000008000002000] (CPF_Transient)
    TArray<class UMaterialEffect_X*> ActiveEffects;          // 0x0068 (0x0010)
    [0x00000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    unsigned long                bNeedsReset : 1;            // 0x0078 (0x0004)
    [0x00000000000002000] [0x00000001] (CPF_Transient)
    TArray<struct FPPEffectDefaults> EffectDefaults;        // 0x0080 (0x0010)
    [0x00000000000400001] (CPF_Edit | CPF_NeedCtorLink)
    TArray<struct FPPPersistentEffectDefaults> PersistentEffectDefaults; // 0x0090
    (0x0010) [0x00000000000400001] (CPF_Edit | CPF_NeedCtorLink)
    TArray<struct FPPChainInfo> ActiveChains;                // 0x00A0 (0x0010)
    [0x00000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FPPChainInfo> ChainDefaults;               // 0x00B0 (0x0010)
    [0x00000000000400001] (CPF_Edit | CPF_NeedCtorLink)
    TArray<struct FPostProcessOverride> PostProcessOverrides; // 0x00C0
    (0x0010) [0x00000008000400001] (CPF_Edit | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PostProcessManager_X");
        }

        return uClassPointer;
    };

    int32_t GetNextPostProcessOverride();
    void TogglePostProcessSettings(struct FName Id, unsigned long bEnabled);
    void SetMaterialEffectValue(struct FName EffectName, float NewValue);
    void ResetEffectsToDefaults(unsigned long bRebuildPostProcessChains);
    void ToggleChainNamed(class UPostProcessChain* Chain, struct FName ChainName, unsigned

```

```

long bEnabled);
void StopChainNamed(struct FName ChainName);
void StartChainNamed(class UPostProcessChain* Chain, struct FName ChainName);
bool RemovePostProcessChain(int32_t OldChain);
int32_t AddPostProcessChain(class UPostProcessChain* NewChain, struct FName ChainName);
class UMaterialEffect_X* GetEffect(struct FName EffectName);
void TickPostProcess(float dt);
void ToggleEffect(class UMaterialEffect_X* Effect, unsigned long bEnabled);
void StopEffectNamed(struct FName EffectName);
void StartEffectNamed(struct FName EffectName);
void ToggleEffectNamed(struct FName EffectName, unsigned long bEnabled);
void PrintDebugInfo(class UDebugDrawer* Drawer);
class UUberPostProcessEffect* GetUberPostProcessEffect();
class UPostProcessChain* GetPlayerChain();
void Exit();
void Init(class APlayerController_X* NewOwner);
};

```

```

// Class ProjectX.PrimaryAuthLoggedIn_TA
// 0x0008 (0x0060 - 0x0068)
class UPrimaryAuthLoggedIn_TA : public UObject
{
public:
class UPsyNetConnection_X*          PrimaryAuthedConnection;          // 0x0060
(0x0008) [0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PrimaryAuthLoggedIn_TA");
}

return uClassPointer;
};

};

```

```

// Class ProjectX.RPC_ReportCheater_X
// 0x0058 (0x00E8 - 0x0140)
class URPC_ReportCheater_X : public URPC_X
{
public:
struct FUniqueNetId          PlayerID;          // 0x00E8 (0x0048)
[0x0000000400040000] (CPF_NeedCtorLink)
class FString                Reason;            // 0x0130 (0x0010)
[0x0000000400040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{

```

```

static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.RPC_ReportCheater_X");
}

return uClassPointer;
};

class URPC_ReportCheater_X* SetReason(class FString InReason);
class URPC_ReportCheater_X* SetPlayerID(struct FUniqueNetId InPlayerId);
};

// Class ProjectX.PsyNetBreadcrumbObserver_X
// 0x0000 (0x0080 - 0x0080)
class UPsyNetBreadcrumbObserver_X : public UMetricsGroup_X
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PsyNetBreadcrumbObserver_X");
        }

        return uClassPointer;
    };

    void Breadcrumb(class FString CrumbName, class FString Value);
    void OnBroadcasterAdded(class UBreadcrumbBroadcaster_X* Broadcaster);
    static void InitGlobalInstance();
    static void SubscribeToPrimaryConnection(class UPsyNet_X* PsyNet);
};

// Class ProjectX.PsyNetService_DuplicateLogin_X
// 0x0000 (0x0090 - 0x0090)
class UPsyNetService_DuplicateLogin_X : public UPsyNetClientService_X
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_DuplicateLogin_X");
        }
    }

```

```

return uClassPointer;
};

};

// Class ProjectX.PsyNetMetrics_X
// 0x0020 (0x0080 - 0x00A0)
class UPsyNetMetrics_X : public UMetricsGroup_X
{
public:
TArray<struct FServiceMetricsData>          ServiceData;                // 0x0080 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)
TArray<struct FServiceErrorData>          ErrorData;                // 0x0090 (0x0010)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetMetrics_X");
}

return uClassPointer;
};

void ServiceErrors(TArray<struct FServiceErrorData> Errors);
void ServiceCalls(TArray<struct FServiceMetricsData> Services);
void HandleMapChange(class FString M);
void RecordServiceError(class FString Service, class UError* Error);
void RecordServiceCall(class FString Service, float Latency);
};

// Class ProjectX.PsyNetService_CreateHonorDuel_X
// 0x0098 (0x00B0 - 0x0148)
class UPsyNetService_CreateHonorDuel_X : public UPsyNetService_ReservationBase_X
{
public:
int32_t                                     Playlist;                // 0x00B0 (0x0004)
[0x000000000000000000]
struct FHonorDuelChallenge                 PlayerRoles;                // 0x00B8 (0x0090)
[0x000000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_CreateHonorDuel_X");
}

```



```

}

return uClassPointer;
};

struct FPsyNetBeaconReservation GetReservation();
};

// Class ProjectX.PsyNetService_Echo_X
// 0x0020 (0x0090 - 0x00B0)
class UPsyNetService_Echo_X : public UPsyNetClientService_X
{
public:
class FString RequestString; // 0x0090 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString ResponseString; // 0x00A0 (0x0010)
[0x0004000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_Echo_X");
}

return uClassPointer;
};

void Execute();
};

// Class ProjectX.PsyNetService_PersonalInfo_X
// 0x0078 (0x0090 - 0x0108)
class UPsyNetService_PersonalInfo_X : public UPsyNetClientService_X
{
public:
struct FOnlinePersonaData FromPlayer; // 0x0090 (0x0078)
[0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_PersonalInfo_X");
}

return uClassPointer;
};

```

```

struct FOnlineFriend GetData();
};

// Class ProjectX.PsyNetService_FriendAcceptedRequest_X
// 0x0000 (0x0108 - 0x0108)
class UPsyNetService_FriendAcceptedRequest_X : public UPsyNetService_PersonalInfo_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_FriendAcceptedRequest_X");
}

return uClassPointer;
};
};

// Class ProjectX.PsyNetService_FriendStatusUpdate_X
// 0x0068 (0x0090 - 0x00F8)
class UPsyNetService_FriendStatusUpdate_X : public UPsyNetClientService_X
{
public:
struct FPsyNetOnlineStatus          FromPlayer;          // 0x0090 (0x0068)
[0x00000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_FriendStatusUpdate_X");
}

return uClassPointer;
};
};

struct FOnlineStatus GetData();
};

// Class ProjectX.PsyNetService_IncomingFriendRequest_X
// 0x0000 (0x0108 - 0x0108)
class UPsyNetService_IncomingFriendRequest_X : public UPsyNetService_PersonalInfo_X
{
public:

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_IncomingFriendRequest_X");
}

return uClassPointer;
};

};

// Class ProjectX.PsyNetService_PlayerUnfriended_X
// 0x0048 (0x0090 - 0x00D8)
class UPsyNetService_PlayerUnfriended_X : public UPsyNetClientService_X
{
public:
struct FUniqueNetId                PlayerID;                // 0x0090 (0x0048)
[0x00000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.PsyNetService_PlayerUnfriended_X");
}

return uClassPointer;
};

};

// Class ProjectX.WebConfig_X
// 0x0008 (0x0078 - 0x0080)
class UWebConfig_X : public UOnlineConfig_X
{
public:
unsigned long                bCacheWebImages : 1;                // 0x0078 (0x0004)
[0x0000000000004001] [0x00000001] (CPF_Edit | CPF_Config)
unsigned long                bZipPsyNetStaticData : 1;                // 0x0078 (0x0004)
[0x0000000000004001] [0x00000002] (CPF_Edit | CPF_Config)
float                PsyNetStaticDataCacheMinutes;                // 0x007C (0x0004)
[0x0000000000004001] (CPF_Edit | CPF_Config)

public:
static UClass* StaticClass()
{

```

```

static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.WebConfig_X");
}

return uClassPointer;
};

};

// Class ProjectX.PsyNetStaticDataMetrics_X
// 0x0000 (0x0080 - 0x0080)
class UPsyNetStaticDataMetrics_X : public UMetricsGroup_X
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.PsyNetStaticDataMetrics_X");
        }

        return uClassPointer;
    };

    void SyncDataTime(float Seconds);
};

// Class ProjectX.StaticDataError_X
// 0x0008 (0x0060 - 0x0068)
class UStaticDataError_X : public UObject
{
public:
    class UError*                               Error;                               // 0x0060 (0x0008)
    [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.StaticDataError_X");
        }

        return uClassPointer;
    };
};

```

```

};

// Class ProjectX.RPC_FilterContent_X
// 0x0040 (0x00E8 - 0x0128)
class URPC_FilterContent_X : public URPC_X
{
public:
    TArray<class FString>          Content;                                // 0x00E8 (0x0010)
    [0x0000000000400000] (CPF_NeedCtorLink)
    TArray<class FString>          FilteredContent;                        // 0x00F8 (0x0010)
    [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FScriptDelegate> Callbacks;                            // 0x0108 (0x0010)
    [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FUniqueNetId>     PlayerIds;                            // 0x0118 (0x0010)
    [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_FilterContent_X");
        }

        return uClassPointer;
    };

    class URPC_FilterContent_X* AddComment(class FString Comment, struct FScriptDelegate
    Callback, struct FUniqueNetId PlayerID);
};

// Class ProjectX.RPC_CanShowAvatar_X
// 0x0030 (0x00E8 - 0x0118)
class URPC_CanShowAvatar_X : public URPC_X
{
public:
    TArray<struct FUniqueNetId>     PlayerIds;                            // 0x00E8 (0x0010)
    [0x0000000000400000] (CPF_NeedCtorLink)
    TArray<struct FUniqueNetId>     AllowedPlayerIDs;                      // 0x00F8 (0x0010)
    [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FUniqueNetId>     HiddenPlayerIDs;                      // 0x0108 (0x0010)
    [0x0000000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_CanShowAvatar_X");
        }
    }
};

```

```

}

return uClassPointer;
};

class URPC_CanShowAvatar_X* SetPlayerIDs(TArray<struct FUniqueNetId>& InPlayerIDs);
};

// Class ProjectX.ReservationsSettingUpMatchMessage_X
// 0x0000 (0x0060 - 0x0060)
class UReservationsSettingUpMatchMessage_X : public UBeaconMessage_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.ReservationsSettingUpMatchMessage_X");
}

return uClassPointer;
};

};

// Class ProjectX.RPC_AddPlayerToRole_X
// 0x0004 (0x00E8 - 0x00EC)
class URPC_AddPlayerToRole_X : public URPC_X
{
public:
int32_t                                RoleID;                                // 0x00E8 (0x0004)
[0x0000000000000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_AddPlayerToRole_X");
}

return uClassPointer;
};

class URPC_AddPlayerToRole_X* SetRole(uint8_t Role);
};

// Class ProjectX.RPC_ClearClubInvites_X

```

```

// 0x0000 (0x00E8 - 0x00E8)
class URPC_ClearClubInvites_X : public URPC_X
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_ClearClubInvites_X");
}

return uClassPointer;
};

};

// Class ProjectX.RPC_FileStorage_GetFileDownloadUrl_X
// 0x0020 (0x00E8 - 0x0108)
class URPC_FileStorage_GetFileDownloadUrl_X : public URPC_X
{
public:
class FString Path; // 0x00E8 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString URL; // 0x00F8 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_FileStorage_GetFileDownloadUrl_X");
}

return uClassPointer;
};

};

// Class ProjectX.RPC_FileStorage_GetFileUploadUrl_X
// 0x0030 (0x00E8 - 0x0118)
class URPC_FileStorage_GetFileUploadUrl_X : public URPC_X
{
public:
class FString Path; // 0x00E8 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString ContentType; // 0x00F8 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)

```

```

class FString                                URL;                                // 0x0108 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_FileStorage_GetFileUploadUrl_X");
}

return uClassPointer;
};

};

// Class ProjectX.RPC_GetAntiAddictionData_X
// 0x0050 (0x00E8 - 0x0138)
class URPC_GetAntiAddictionData_X : public URPC_X
{
public:
struct FUniqueNetId                        PlayerID;                                // 0x00E8 (0x0048)
[0x0001004000040000] (CPF_NeedCtorLink)
unsigned long                             bUnderAge : 1;                                // 0x0130 (0x0004)
[0x0001004000002000] [0x00000001] (CPF_Transient)
unsigned long                             bShowMessage : 1;                                // 0x0130 (0x0004)
[0x0001004000002000] [0x00000002] (CPF_Transient)
int32_t                                    HoursPlayed;                                // 0x0134 (0x0004)
[0x0001004000002000] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetAntiAddictionData_X");
}

return uClassPointer;
};

class URPC_GetAntiAddictionData_X* SetPlayerID(struct FUniqueNetId InPlayerId);
};

// Class ProjectX.RPC_GetGenericDataAll_X
// 0x0010 (0x00E8 - 0x00F8)
class URPC_GetGenericDataAll_X : public URPC_X
{
public:
TArray<struct FGetGenericDataAllData>      GenericData;                                // 0x00E8

```



(0x0010) [0x0000004000402000] (CPF\_Transient | CPF\_NeedCtorLink)

```
public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetGenericDataAll_X");
}

return uClassPointer;
};

};

// Class ProjectX.RPC_GetPlayerTitles_X
// 0x0058 (0x00E8 - 0x0140)
class URPC_GetPlayerTitles_X : public URPC_X
{
public:
struct FUniqueNetId                PlayerID;                // 0x00E8 (0x0048)
[0x0000000400040000] (CPF_NeedCtorLink)
TArray<class FString>              Titles;                  // 0x0130 (0x0010)
[0x0000000400040200] (CPF_Transient | CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_GetPlayerTitles_X");
}

return uClassPointer;
};

class URPC_GetPlayerTitles_X* SetPlayerID(struct FUniqueNetId InPlayerId);
};

// Class ProjectX.RPC_PartyChatMessage_X
// 0x0010 (0x00F8 - 0x0108)
class URPC_PartyChatMessage_X : public URPC_PartyBase_X
{
public:
class FString                      Message;                  // 0x00F8 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
```

```

static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.RPC_PartyChatMessage_X");
}

return uClassPointer;
};

class URPC_PartyChatMessage_X* SetMessage(class FString InMessage);
};

// Class ProjectX.RPC_RemovePlayerFromRole_X
// 0x0004 (0x00E8 - 0x00EC)
class URPC_RemovePlayerFromRole_X : public URPC_X
{
public:
    int32_t                                RoleID;                                // 0x00E8 (0x0004)
    [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_RemovePlayerFromRole_X");
        }

        return uClassPointer;
    };

    class URPC_RemovePlayerFromRole_X* SetRole(uint8_t Role);
};

// Class ProjectX.RPC_ReportExploiter_X
// 0x0070 (0x00E8 - 0x0158)
class URPC_ReportExploiter_X : public URPC_X
{
public:
    struct FUniqueNetId                    PlayerID;                                // 0x00E8 (0x0048)
    [0x0000000000040000] (CPF_NeedCtorLink)
    uint8_t                                Reason;                                // 0x0130 (0x0001)
    [0x0000000000000000]
    class FString                          MatchGuid;                            // 0x0138 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)
    class FString                          Data;                                // 0x0148 (0x0010)
    [0x0000000000040000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {

```

```

static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
    uClassPointer = UObject::FindClass("Class ProjectX.RPC_ReportExploiter_X");
}

return uClassPointer;
};

class URPC_ReportExploiter_X* SetData(class FString InData);
class URPC_ReportExploiter_X* SetMatchGUID(class FString InMatchGUID);
class URPC_ReportExploiter_X* SetReason(uint8_t InReason);
class URPC_ReportExploiter_X* SetPlayerID(struct FUniqueNetId InPlayerId);
static void Send(struct FUniqueNetId InPlayerId, uint8_t InReason, class FString InMatchGUID,
class FString InData);
};

// Class ProjectX.RPC_ReportLowFPS_X
// 0x0040 (0x00E8 - 0x0128)
class URPC_ReportLowFPS_X : public URPC_X
{
public:
    class FString                IP;                                // 0x00E8 (0x0010)
    [0x0000000400040000] (CPF_NeedCtorLink)
    int32_t                      MachineId;                          // 0x00F8 (0x0004)
    [0x0000000400000000]
    class FString                ServerId;                          // 0x0100 (0x0010)
    [0x0000000400040000] (CPF_NeedCtorLink)
    class FString                ServerName;                        // 0x0110 (0x0010)
    [0x0000000400040000] (CPF_NeedCtorLink)
    int32_t                      NumHumans;                          // 0x0120 (0x0004)
    [0x0000000400000000]
    int32_t                      NumBots;                            // 0x0124 (0x0004)
    [0x0000000400000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_ReportLowFPS_X");
        }

        return uClassPointer;
    };

    class URPC_ReportLowFPS_X* SetNumBots(int32_t InNumBots);
    class URPC_ReportLowFPS_X* SetNumHumans(int32_t InNumHumans);
    class URPC_ReportLowFPS_X* SetServerName(class FString InServerName);
    class URPC_ReportLowFPS_X* SetServerID(class FString InServerID);
    class URPC_ReportLowFPS_X* SetMachineID(int32_t InMachineID);

```

```

class URPC_ReportLowFPS_X* SetIP(class FString InIP);
};

// Class ProjectX.RPC_SetClubMotD_X
// 0x0010 (0x0120 - 0x0130)
class URPC_SetClubMotD_X : public URPC_CreateClub_X
{
public:
    class FString                Text;                // 0x0120 (0x0010)
    [0x0001000000400000] (CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_SetClubMotD_X");
        }

        return uClassPointer;
    };

    class URPC_SetClubMotD_X* SetText(class FString InText);
};

// Class ProjectX.RPC_SetPlayerSkill_X
// 0x0010 (0x00E8 - 0x00F8)
class URPC_SetPlayerSkill_X : public URPC_X
{
public:
    int32_t                      Playlist;              // 0x00E8 (0x0004)
    [0x0000000400000000]

    float                        Mu;                    // 0x00EC (0x0004)
    [0x0000000400000000]

    float                        Sigma;                  // 0x00F0 (0x0004)
    [0x0000000400000000]

    int32_t                      MatchesPlayed;          // 0x00F4 (0x0004)
    [0x0000000400000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_SetPlayerSkill_X");
        }

        return uClassPointer;
    };
};

```

```

class URPC_SetPlayerSkill_X* SetSigma(float InSigma);
class URPC_SetPlayerSkill_X* SetMu(float InMu);
class URPC_SetPlayerSkill_X* SetPlaylist(int32_t InPlaylist);
};

// Class ProjectX.RPC_SetPlayerSkillTier_X
// 0x000C (0x00E8 - 0x00F4)
class URPC_SetPlayerSkillTier_X : public URPC_X
{
public:
int32_t Playlist; // 0x00E8 (0x0004)
[0x0000000400000000]
int32_t Tier; // 0x00EC (0x0004)
[0x0000000400000000]
int32_t MatchesPlayed; // 0x00F0 (0x0004)
[0x0000000400000000]

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_SetPlayerSkillTier_X");
}

return uClassPointer;
};

class URPC_SetPlayerSkillTier_X* SetMatchesPlayed(int32_t InMatchesPlayed);
class URPC_SetPlayerSkillTier_X* SetTier(int32_t InTier);
class URPC_SetPlayerSkillTier_X* SetPlaylist(int32_t InPlaylist);
};

// Class ProjectX.RPC_SetRichPresence_X
// 0x0020 (0x00E8 - 0x0108)
class URPC_SetRichPresence_X : public URPC_X
{
public:
class FString PresenceInfo; // 0x00E8 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)
class FString PresenceState; // 0x00F8 (0x0010)
[0x0000000000040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_SetRichPresence_X");
}
}

```

```

return uClassPointer;
};

class URPC_SetRichPresence_X* SetPresenceState(class FString InState);
class URPC_SetRichPresence_X* SetPresenceInfo(class FString InInfo);
};

// Class ProjectX.RPC_SetSeasonReward_X
// 0x0008 (0x00E8 - 0x00F0)
class URPC_SetSeasonReward_X : public URPC_X
{
public:
    int32_t SeasonLevel; // 0x00E8 (0x0004)
    [0x0000000000000000]
    int32_t SeasonLevelWins; // 0x00EC (0x0004)
    [0x0000000000000000]

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_SetSeasonReward_X");
        }

        return uClassPointer;
    };

class URPC_SetSeasonReward_X* SetReward(int32_t Level, int32_t Wins);
};

// Class ProjectX.RPC_Test_X
// 0x0170 (0x00E8 - 0x0258)
class URPC_Test_X : public URPC_X
{
public:
    class FString TestParamLocalizedString; // 0x00E8 (0x0010)
    [0x00000000000408002] (CPF_Const | CPF_Localized | CPF_NeedCtorLink)
    class FString TestParamString; // 0x00F8 (0x0010)
    [0x00000000000400000] (CPF_NeedCtorLink)
    struct FRPCTestParam TestParamStruct; // 0x0108 (0x0088)
    [0x00000000000400000] (CPF_NeedCtorLink)
    TArray<struct FRPCTestParam> TestParamStructs; // 0x0190
    (0x0010) [0x00000000000400000] (CPF_NeedCtorLink)
    class FString TestResultLocalizedString; // 0x01A0 (0x0010)
    [0x00000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    class FString TestResultString; // 0x01B0 (0x0010)
    [0x00000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    struct FRPCTestParam TestResultStruct; // 0x01C0 (0x0088)
    [0x00000000000402000] (CPF_Transient | CPF_NeedCtorLink)
    TArray<struct FRPCTestParam> TestResultStructs; // 0x0248

```

(0x0010) [0x0000000000402000] (CPF\_Transient | CPF\_NeedCtorLink)

```
public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.RPC_Test_X");
}

return uClassPointer;
};

bool Check(unsigned long Assertion, class FString Message);
bool CheckParamArraysMatch(TArray<struct FRPCTestParam> A, TArray<struct
FRPCTestParam> B, class FString Message);
bool CheckItemArraysMatch(TArray<struct FRPCTestItem> A, TArray<struct FRPCTestItem> B,
class FString Message);
bool CheckStringArraysMatch(TArray<class FString> A, TArray<class FString> B, class FString
Message);
bool CheckItemsMatch(struct FRPCTestItem A, struct FRPCTestItem B, class FString Message);
bool CheckParamsMatch(struct FRPCTestParam A, struct FRPCTestParam B, class FString
Message);
bool ValidateResults();
void eventOnComplete();
TArray<struct FRPCTestParam> GetRandomTestParams();
struct FRPCTestParam GetRandomTestParam();
TArray<struct FRPCTestItem> GetRandomTestItems();
struct FRPCTestItem GetRandomTestItem();
void Init();
};

// Class ProjectX.TestsHelper_X
// 0x0000 (0x0060 - 0x0060)
class UTestsHelper_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.TestsHelper_X");
}

return uClassPointer;
};

static class FString GetRandomUppercaseString();
```

```

static TArray<class FString> GetRandomStringArray();
static class FString GetRandomString();
};

// Class ProjectX.RPC_TestPlayerID_X
// 0x0090 (0x00E8 - 0x0178)
class URPC_TestPlayerID_X : public URPC_X
{
public:
    struct FUniqueNetId                PlayerID;                // 0x00E8 (0x0048)
    [0x000000000000400000] (CPF_NeedCtorLink)
    struct FUniqueNetId                ResponsePlayerID;        // 0x0130 (0x0048)
    [0x000000000000402000] (CPF_Transient | CPF_NeedCtorLink)

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_TestPlayerID_X");
        }

        return uClassPointer;
    };

    void OnSuccess();
    void Init();
};

// Class ProjectX.RPC_UpdateClub_X
// 0x0000 (0x0120 - 0x0120)
class URPC_UpdateClub_X : public URPC_CreateClub_X
{
public:

public:
    static UClass* StaticClass()
    {
        static UClass* uClassPointer = nullptr;

        if (!uClassPointer)
        {
            uClassPointer = UObject::FindClass("Class ProjectX.RPC_UpdateClub_X");
        }

        return uClassPointer;
    };

};

// Class ProjectX.SeqAct_SpawnArchetype_X
// 0x0040 (0x0160 - 0x01A0)

```



```

class USeqAct_SpawnArchetype_X : public USequenceAction
{
public:
class AActor*                ActorArchetype;                // 0x0160 (0x0008)
[0x0000000000000001] (CPF_Edit)
struct FVector                Velocity;                // 0x0168 (0x000C)
[0x0000000000000001] (CPF_Edit)
float                Speed;                // 0x0174 (0x0004)
[0x0000000000000001] (CPF_Edit)
TArray<class AActor*>                SpawnOwners;                // 0x0178 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
TArray<class AActor*>                SpawnPoints;                // 0x0188 (0x0010)
[0x0000000000400000] (CPF_NeedCtorLink)
class AActor*                Spawned;                // 0x0198 (0x0008)
[0x0000000000002000] (CPF_Transient)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.SeqAct_SpawnArchetype_X");
}

return uClassPointer;
};

void Init(class AActor* SpawnedActor);
void eventActivated();
};

// Class ProjectX.SkelControlBlendTargetComponent_X
// 0x000C (0x00A4 - 0x00B0)
class USkelControlBlendTargetComponent_X : public UActorComponent_X
{
public:
struct FName                SkelControlName;                // 0x00A8 (0x0008)
[0x0000000000000001] (CPF_Edit)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.SkelControlBlendTargetComponent_X");
}

return uClassPointer;
};

```

```

void SetControlActiveInComponent(class USkeletalMeshComponent* SKC, unsigned long
bActive);
void SetControlActive(unsigned long bActive);
void eventDetached();
void eventAttached();
};

```

```

// Class ProjectX.SmoothDynamicValue_X
// 0x0010 (0x0084 - 0x0094)
class USmoothDynamicValue_X : public UDynamicValue_X
{
public:
float                LerpUpSpeed;                // 0x0088 (0x0004)
[0x0000000000000001] (CPF_Edit)
float                LerpDownSpeed;                // 0x008C (0x0004)
[0x0000000000000001] (CPF_Edit)
float                SmoothedCachedValue;          // 0x0090 (0x0004)
[0x0000000000000200] (CPF_Transient)

```

```

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.SmoothDynamicValue_X");
}

return uClassPointer;
};

```

```

class FString GetDebugValue();
float GetValue();
void Tick(float DeltaTime);
void eventConstruct();
};

```

```

// Class ProjectX.StringUtil_X
// 0x0000 (0x0060 - 0x0060)
class UStringUtil_X : public UObject
{
public:

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.StringUtil_X");
}
}

```

```

return uClassPointer;
};

static bool IsStringEmptyOrWhiteSpace(class FString BaseString);
static void SplitStringInHalf(class FString OriginalString, class FString& FirstHalf, class FString&
SecondHalf);
};

// Class ProjectX.TimeWindowOptional_X
// 0x0050 (0x0060 - 0x00B0)
class UTimeWindowOptional_X : public UObject
{
public:
uint64_t                StartTime;                // 0x0060 (0x0008)
[0x000000040000000000]
uint64_t                EndTime;                // 0x0068 (0x0008)
[0x000000040000000000]
class UIEpochNow*      Time_Object;                // 0x0070 (0x0008)
[0x000000040000000000]
class UIEpochNow*      Time_Interface;            // 0x0078 (0x0008)
[0x000000040000000000]
struct FScriptDelegate  __EventWindowStarted__Delegate;    // 0x0080
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)
struct FScriptDelegate  __EventWindowEnded__Delegate;      // 0x0098
(0x0018) [0x00000000000400000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.TimeWindowOptional_X");
}

return uClassPointer;
};

void HandleWindowEnded();
void HandleWindowStarted();
bool WillEnd();
bool HasEnded();
bool HasStarted();
bool IsActive();
class UTimeWindowOptional_X* NotifyOnWindowEnded(struct FScriptDelegate
OnEndedCallback);
class UTimeWindowOptional_X* NotifyOnWindowStarted(struct FScriptDelegate
OnStartedCallback);
class UTimeWindowOptional_X* Init(uint64_t InStartTime, uint64_t InEndTime, class
UIEpochNow* InTimeProvider);
void EventWindowEnded();
void EventWindowStarted();
};

```

```

// Class ProjectX.UdpLanServer_X
// 0x0018 (0x0060 - 0x0078)
class UUdpLanServer_X : public UObject
{
public:
class ULanBeacon_X* Beacon; // 0x0060 (0x0008)
[0x0000000004080009] (CPF_Edit | CPF_ExportObject | CPF_Component | CPF_EditInline)
class FString Metadata; // 0x0068 (0x0010)
[0x0000000400040000] (CPF_NeedCtorLink)

public:
static UClass* StaticClass()
{
static UClass* uClassPointer = nullptr;

if (!uClassPointer)
{
uClassPointer = UObject::FindClass("Class ProjectX.UdpLanServer_X");
}

return uClassPointer;
};

void HandleLanQueryMessage(class UOnlineMessageComponent_X* Component, class
ULanMessage_HostQuery_X* Query);
void Destroy();
class UAsyncTask* SetServerMetaData(class FString InMetaData);
void eventConstruct();
};

/*
#
=====
===== #
#
#
=====
===== #
*/

#ifdef _MSC_VER
#pragma pack(pop)
#endif

```

Removed: 27

Added: 34

Generated at <https://www.textcompare.org/> on 05/06/2024, 18:07:33