# NetShot样例分析

## 新增内容

每次启动游戏会在随机位置生成一个Mybonus，击中加分且该物体自动销毁

Wasd/上下左右控制人物移动

## Animations

存放动画序列

## Audios

存放音频

## Blueprints

存放主要的蓝图

#### BP\_MuzzleFlash

枪口火焰蓝图

#### BP\_Projectile

疑似击中目标触发

#### CamShake\_01

主场景的镜头抖动

#### MyGameInstance

非业务逻辑的全局管理。玩家加入对局、打开UI页面、退出游戏、在游戏中展示载入页面等等。

#### MyGameMode

游戏模式。

* PostLogin调用MyPlayerController
* Respawn Player Event被MyPlayerController调用，生成枪手

#### MyGameState

记录游戏世界关键状态。目前只有构造函数。（可以在其中加入玩家的得分等信息）

#### MyPlayerController

控制player。

* ClientPostLogin被MyGameMode调用，加载UI控件，抬头显示等，调用MyGameMode中的Respawn Player Event函数生成枪手。
* ShowTheMenu 展示菜单
* Remove the HUD，事件结束后删除HUD

#### MyPlayerState

在此处利用InCrementScore控制玩家分数

* 函数Increment Score负责分数增加
* 变量Player Score记录玩家分数

#### State

枚举类型，存储游戏中的状态

## Blueprints/Widgets

存储控件蓝图

#### AmmoWidget

右下角显示子弹数目的UI

#### ErrorDialog

显示错误的UI

#### InGameMenu

游戏中控制游戏进程的UI，有两个button，分别用于重新开始游戏和退出游戏。

#### LoadingScreen

载入UI

#### MainMenu

主菜单UI，包含四个button，分别为Play，Find games，Quit，

#### MyHUD

抬头显示UI，在底部展示操作方法

#### ScoreboardRow

显示玩家分数的UI

#### ServerList

显示服务器、客户端情况的UI

#### ServerRow

在顶部展示服务器连接情况的UI

## Effects

存储粒子系统

## Environment

场景中的物体

## Gunslinger

枪手（玩家）的动画、骨骼和蓝图

#### Gunslinger\_AinmBP

枪手的动画蓝图，根据IsAiming和IsShooting的状态展示不同的动画

#### Gunslinger\_BP

控制枪手的蓝图

* Reload控制枪手载入
* Firing控制枪手开火。IsAiming=true才能开火，开火时设置IsShooting=true、镜头抖动、生成子弹、生成枪口火光、播放开火音乐
* Aiming，输入分为Aiming和NotAiming，分别将IsAiming设为true和False。在服务器上执行Aiming事件后，在所有机器上执行ClientAiming事件
* Input
* Movement控制枪手移动【暂未实现
* Take Damage 受到伤害【暂未实现
* Tick

## Maps

几个关卡

## Meshes

静态网格体

## Sounds

开火、受到击打等的音效

## Textures

纹理