TPS Character System Guide

Guide for Product Version: 1.4.1

Table of Content

Introduction

Thank you very much for purchasing TPS Character System.

The product was developed as complex system, which includes main gameplay mechanics of game character.

The product will be constantly improved.

Please, send all suggestions for improvement or new functionality, as well as bugs to the mail or to the discord channel.

Contact Email: gameparts.main@gmail.com

Discord Channel: https://discordapp.com/invite/z63hbJr

Guide Overview

This document describes how to customize each of the main systems. The principle of operation system is not described in the guide. All blueprints complimenting comments and the principle of operation will be clear when analyzing the blueprints themselves.

Controls Mapping

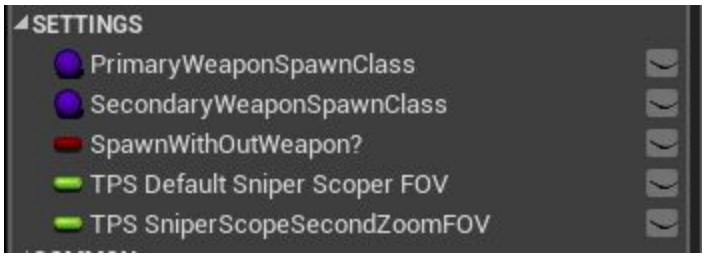
- Shooting: R2 (PS4 / Xbox), Left Mouse Button
- Mini Zoom: L2 (PS4 / Xbox), Right Mouse Button
- Reload Weapon: Square (PS4), X (Xbox), R keyboard button
- Crouch: Circle (PS4), B (Xbox), C keyboard button
- Climb / Jump Over: X (PS4), A (Xbox), Space keyboard button
- Character Movement / Camera Movement: Left / Right Stick (PS4 / Xbox), Mouse
- Swap Weapon: TAP Triangle (PS4), Y (Xbox), Q keyboard button
- Hide Weapon: HOLD Triangle (PS4), Y (Xbox), Q keyboard button
- Pick Up Weapon: HOLD Square (PS4), HOLD X (Xbox), E keyboard button
- Sniper Rifle Zoom: R3 (PS4), R3 (Xbox), Up / Down Wheel Mouse
- Sprint: L3 (PS4), L3 (Xbox), Left Shift keyboard button
- Camera Mirror: R1 (PS4), RB (Xbox), X keyboard button
- Melee Attack: L1 (PS4), LB (Xbox), F keyboard button

Customize Characters Setting

Customize TPS Settings

In BP_TPSCharacter you can customize following settings:

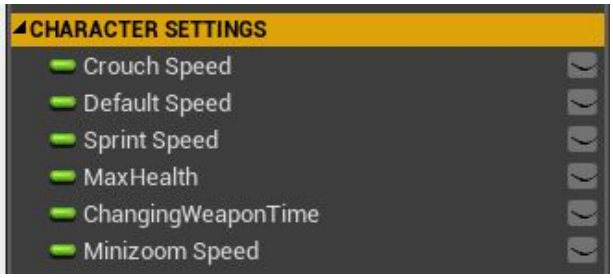
- **TPS Default Sniper Scoper FOV.** The FOV setting, which used when the character has sniper scope state without zoom.
- **TPS SniperScopeSecondZoomFOV.** The FOV setting, which used when the character has sniper scope state with zoom.
- **PrimaryWeaponClass.** The weapon class setting. This class of a weapon will spawn in the primary weapon slot.
- **SecondaryWeaponClass.** The weapon class setting. This class of a weapon will spawn in the secondary weapon slot.
- **SpawnWithOutWeapon?** The setting sets start state of the character: with a weapon or without a weapon.



Customize Common Setting

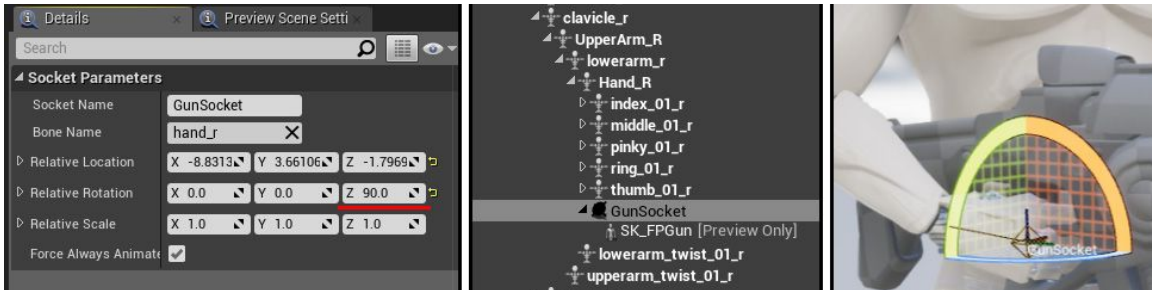
The system support common character setting such as speed, health, etc. This settings can be customy in BP_CommonCharacter.

- **Default Speed.** The setting of characters speed, which used when the character has not minizoom state or sprint state or crouch state.
- **Crouch Speed.** The speed setting in crouch state.
- **Sprint Speed.** The speed setting in sprint state.
- **Minizoom Speed.** The speed setting in minizoom state.
- **MaxHealth.** The heath setting.
- **ChangingWeaponTime.** Time for which the character will change weapons or will hide weapons.



Character Skeletal Meshes

The Skeletal Mesh of the character must have the Gun Socket.

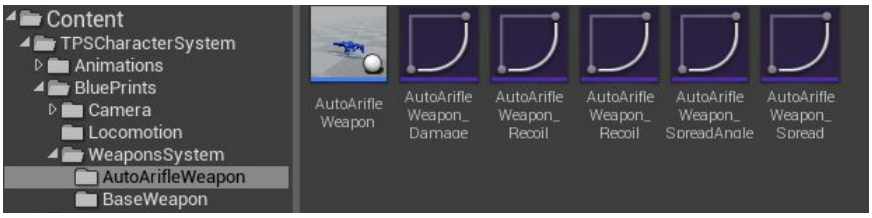


Weapons System

Creating New Weapon

Creating New Weapon Blueprint Class

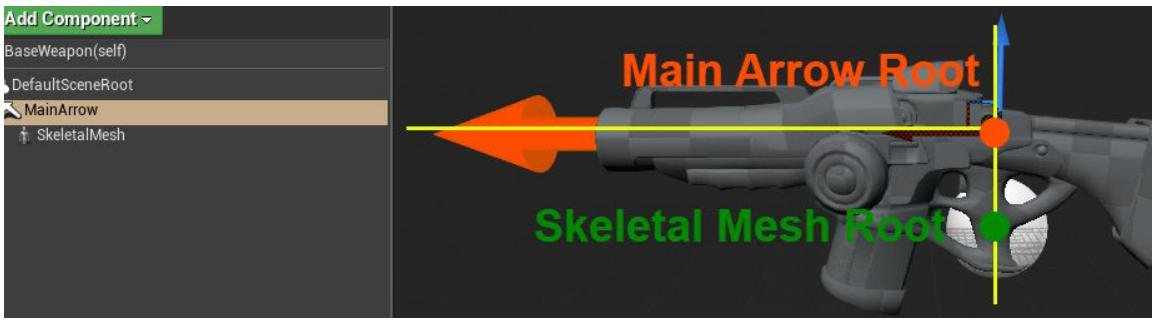
To creating a new types of weapon it is necessary to inherit a new class from the base class of weapons - BP_BaseWeapon. Put the blueprint in a new folder and create new settings curves (for example, see how it was done for auto arifle).



Weapon Blueprint Class & Skeletal Mesh Customize

SkeletalMesh in a weapons blueprint class must be properly configured.

- SkeletalMesh Root should be located in DefaultSceneRoot и and coincide with the place for which this weapon will be held by the character.
- Root of Main Arrow component should be located on one vertical with DefaultSceneRoot and SkeletalMesh Root.
- Main Arrow should be directed along the line of gun barrels, while indicating the direction of shots.



There are should be two sockets in weapons SkeletalMesh:

- Muzzle Socket. In place of the socket, where muzzle sfx is created.
- L_HandSocket. Required for animated blueprints, so that a character's left hand always holds a gun for this place.

Weapons Setting Customize

Weapons must be customize in from an inherited class. The settings there are in a structure of Weapon Settings.

Basic Weapons Settings

- **Fire Rate**. Rate of fire. Number of bullets per minute.
- **Clip Size**. Number of cartridges in clip.
- **Ammo Size**. Total number of cartridges for weapon.
- **Reload Time**. Reload animation corresponds this parameter. For correctly determine a duration of a reload animation, necessary set default time of an animation montage, which using for reload weapon. This parameter can be set in anim blueprint, in reload section.
- **Shooting Max distance**. How far the weapon can shoot. This parameter affects spread.

Weapons Type

- **ARifle.** This rifles type
- **Pistol.** This pistols type
- **ShotGun.** This weapon type uses unique multitrace system. In one shot creates 6 trace with random spread for create shotgun mechanics.
- **Sniper Rifle.** This weapon type uses addition game logic of the scope. Sniper scope can be customized in Base Character BP. There are two properties: **SniperScopeDefaultZoomFOV** and **SniperScopeSecondZoomFOV**.

Shooting Types

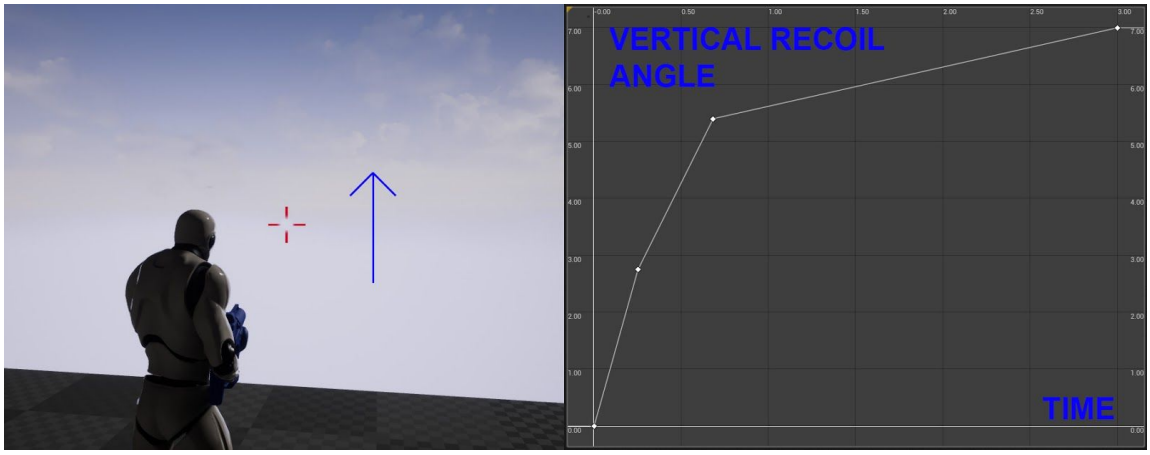
The Weapons System supported various weapons types. Weapons types is determined by the parameter Shooting Type.

- **Auto Fire.** Continuous shooting with a clamped trigger.
- **Single Fire.** A weapon that fires single shots. Play animation twitch shutter after each shot.
- **Auto Single Fire.** A weapon that fires single shots. A cartridge in a barrel is brought automatically after each shot
- **Burst Fire.** Several shots for one click of the trigger. In this case, with a rapid successive second push - a weapon remembers that it is necessary to produce a second shooting cycle. This weapons type is determined by the parameters: **BurstFirePause** (delay between turns) and **BurstFireShotsBeforePause** (shots in one turn)

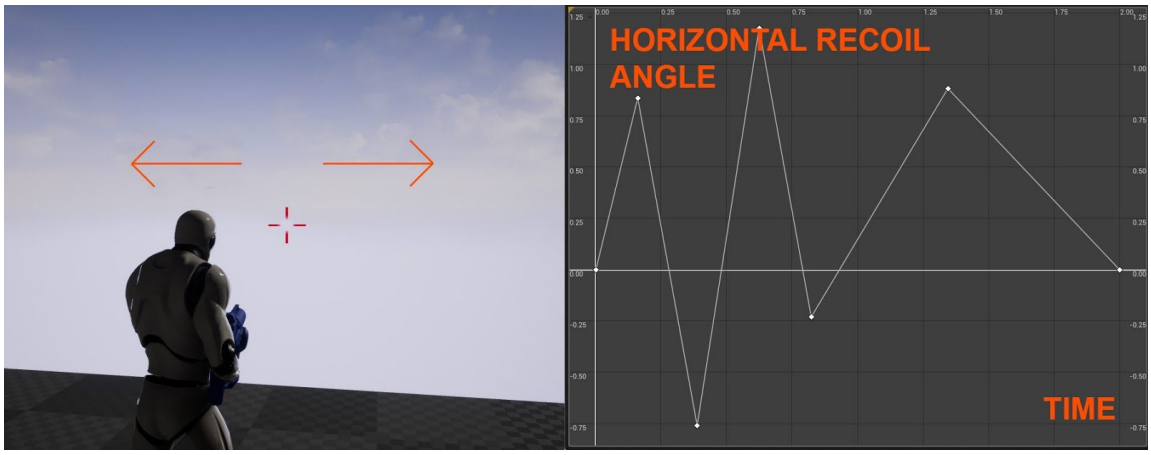
Recoil Settings

The setting of a weapon's recoil is set by 2 curves.

- ***WeaponClassName*_RecoilVertical.** The curve represents a dependence of a vertical deviation angle on a shooting time.



- ***WeaponClassName*_RecoilHorizontal.** The curve represents a dependence of a horizontal deflection angle on a shooting time.



Recoil Recovery Settings

A weapon return to starting position after shooting is configured by the following parameters.

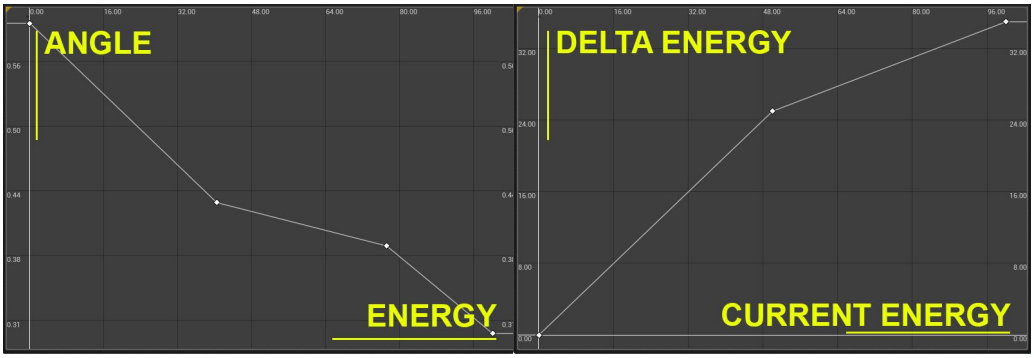
- **Recoil Recovery Vertical Speed.** Recovery speed vertically.
- **Recoil Recovery Horizontal Speed.** Recovery speed horizontal.
- **Recoil Recovery Delay.** After delay return of weapons will start.

Spread Settings

The mechanics of spread is based on the logic of spread energy.

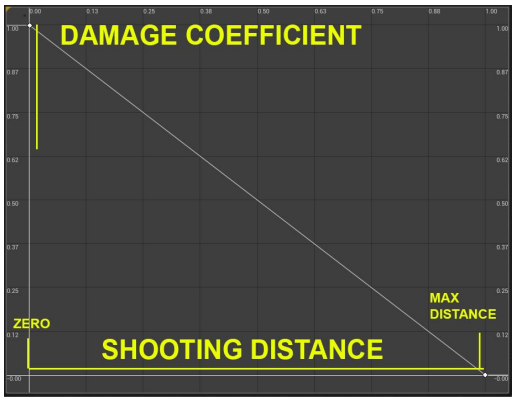
At the beginning of shooting the weapon have maximum value of spread energy, which is equal to 100. During shooting, spread energy drops. It depends on a energy curve **SpreadEnergyCurve.** The curve describes a energy reduction relative to a current energy.

The spread of bullets can be setup on spread angle. The angle is determined from a curve **SpreadAngleCurve** depending on a current energy level.



Weapon Damage Settings

- **Damage Amount.** Base damage per one bullet.
- **Damage Coefficient.** Dependence on curve of distance damage.



Take Damage Logic

Damage value is transferred to the event Take Damage from weapons system component, Create Trace & Damage System & SFX & Sounds Graph, section TAKE DAMAGE. After it processed in BaseCharacter blueprint. If you wish, a transferred damage variable can be connected to a different damage & heath system.

Recoil & Spread Coefficient Settings

During shooting, certain logic calculates recoil and spread. After this, obtained recoil value and spread value are multiplied by the coefficients. The coefficients depend on a character's states.

- **Spread Floor Blind Coefficient / Recoil Floor Blind Coefficient.** Use when a character has standing position.
- **Spread Floor Minizoom Coefficient / Recoil Floor Minizoom Coefficient.** Use when a character has standing position and has state minizoom.
- **Spread Crouch Blind Coefficient / Recoil Crouch Blind Coefficient.** Use when a character has a crouch state.
- **Spread Crouch Minizoom Coefficient / Recoil Crouch Minizoom Coefficient.** Use when a character has crouch and has state minizoom.
- **Spread Move Coefficient / Recoil Move Coefficient.** The last multiplying coefficient. Use when a character is moving in any state.

SFX & Sound Settings & HUD Icon

- **Hit Point SFX.** The effect that will create in a place collision of line trace, where a weapon gets bullet.
- **Muzzle SFX.** The effect muzzle that will create in near of weapon's end barrel.
- **Trail SFX.** The effect trail that will create in near of weapon's end barrel.
- **Hit Point Decal.** The effect of the scorch after hit.
- **WeaponShotSound.** The sound played in a weapon
- **HitPointShotSound.** The sound is play in a place where line trace hits.
- **HudIcon.** This is icon for hud.

Weapons PickUp

The product supports pickup of weapons (BP_BaseWeaponPickUp). This class can be placed on a level. It has one setting - Weapon Item Class (class of a weapon which will be spawned on begin game).

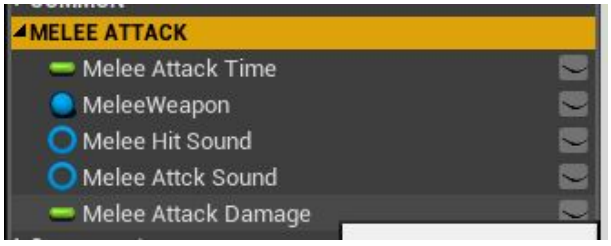
Melee Attack

Melee Attack Overview

To implement the melee attack, use the animation from Mixamo Anim Pack. Also for the example of the weapon use the weapon from Infinity Blade Weapons Pack.

Customize

All melee attack setting there are in BP_CommonCharacter, in section MELEE ATTACK.



- **Melee Attack Time.** Time for which passes the melee attack. Under this time, the animation and the mechanics of damage are adapted.
- **Melee Hit Sound.** A sound that is played at the point of hit by the melee attack.
- **Melee Attack Sound.** A sound that is played when melee attack are used.
- **Melee Attack Damage.** Melee damage.

Player’s Camera

Player’s Camera Overview

This system allows to change a states of a player's camera depending on any actions: transition to sprinting, running, crouching, etc. The system operates with four camera variables: Target Arm Length, Socket Offset, Target Offset, FOV.

System Structure

The camera system there is in the folder BluePrints - Camera. The logic of the work is in AdvancedTPSCamera actor component, which connects to BP_TPSPlayerController. Settings for camera mods are in CameraModeLib.

If you need to add a new mode or remake an old one, the transitions between modes are controlled by the logic in the event CameraController in CAMERA CONTROLLER Graphs, BP_TPSPlayerController blueprint.

Customize Tranzition Camera Modes

In modes settings, you can set a curve for transition. There are 2 types of transitive curve: default and unique for transition from a certain mode. The logic of a camera mode selection depends on a correctness of a name modes in CameraModeLibrary and a name of a curves. Consider example of CrouchIdle mode. This mode has 3 transitions curve:

- **CrouchIdle_Default.**Used to change camera settings in cases where the transition to CrouchIdle from any other mode, except CrouchMiniZoom or CrouchWalk.
- **CrouchIdle_CrouchMiniZoom.** Used when CrouchIdle mode camera goes from CrouchMiniZoom.
- **CrouchIdle_CrouchWalk.** Used when CrouchIdle mode camera goes from CrouchWalk.

Correct spelling of a name of the curve:
ModeToWhereWeWantToGo**From**ModeFromWhereWeAreGoing.

When adjusting curves, it is necessary to take into account that the transition time from one camera modes to another depends on the horizontal time scale.

Cover System

Cover System Overview

The logic of the covers system is in CoverAnalyzer component, which is connected to the player's controller. The system uses traces for analyze level geometry and gives an opportunity to use 2 possibilities: climbing and jump over.

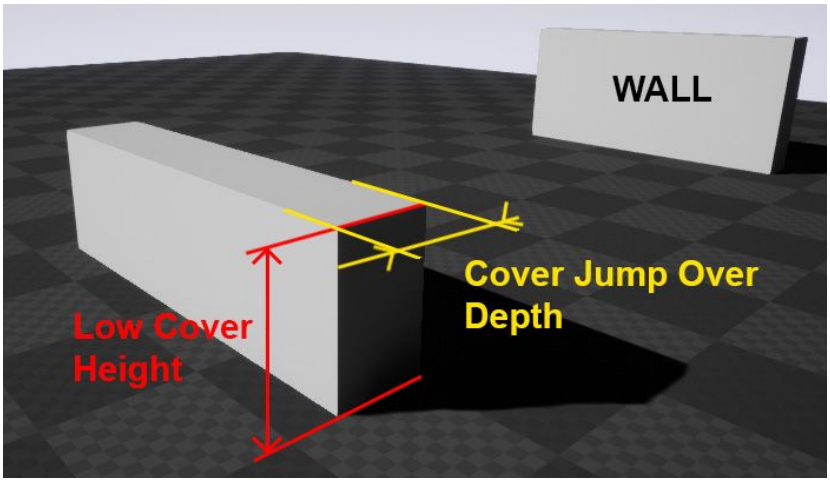
In the first version of the product, mechanics are implemented only for low covers. In future, you will add new features for this system.

Customize Covers Analyzer. Covers Settings

Cover analyzer uses sockets form character mesh for check wall and low covers. This sockets determine height of the low cover.

In Cover Analyzer BP component there are additional setting.

- **Cover Jump Over Depth** (Default Value = 90). Maximal thickness of a cover, for possibility of a jump over.
- **Cover Check Angle** (Default Value = 0.2). The maximum angle of the character's deviation from a cover



Customize Climb & Jump Over Movement

Cover Analyzer transfers variables for climb or jump over to BP_CommonCharacter blueprint.

The logic of moving a character when climbing and jump over is in CLIMB and JUMP OVER sections.

Moves of each type is divided into 2 parts, and uses set location node with time line nodes. Each time line node uses time curve. A time in timeline nodes determines a speed of movement in climb or jump over.

Artificial Intelligence

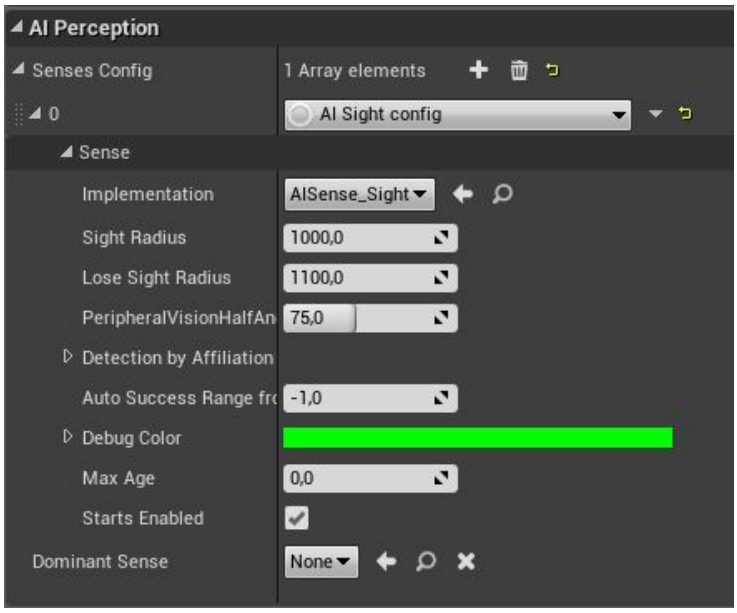
Overview

The system support basic AI character. This AI can detect a player. Check distance. If distance more than attack distance , ai will be move to target.If distance less or equal of attack distance, that ai will be shot to target. AI has opportunity to aim. Ai will be get nearest player as target.

Customize

AI Perception

For vision using AIPerception which connected to BP_AIController. Vision setting can be customize in AIPerception setting.



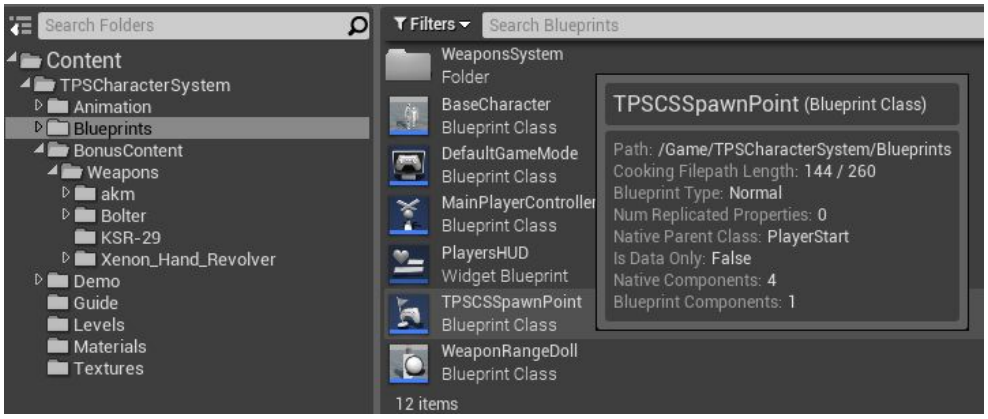
Vision & Detect Valid Targets Logic

For vision logic using BTS_Vision. In this BP can be customized ValidAttackDistance.

Spawn System

Overview

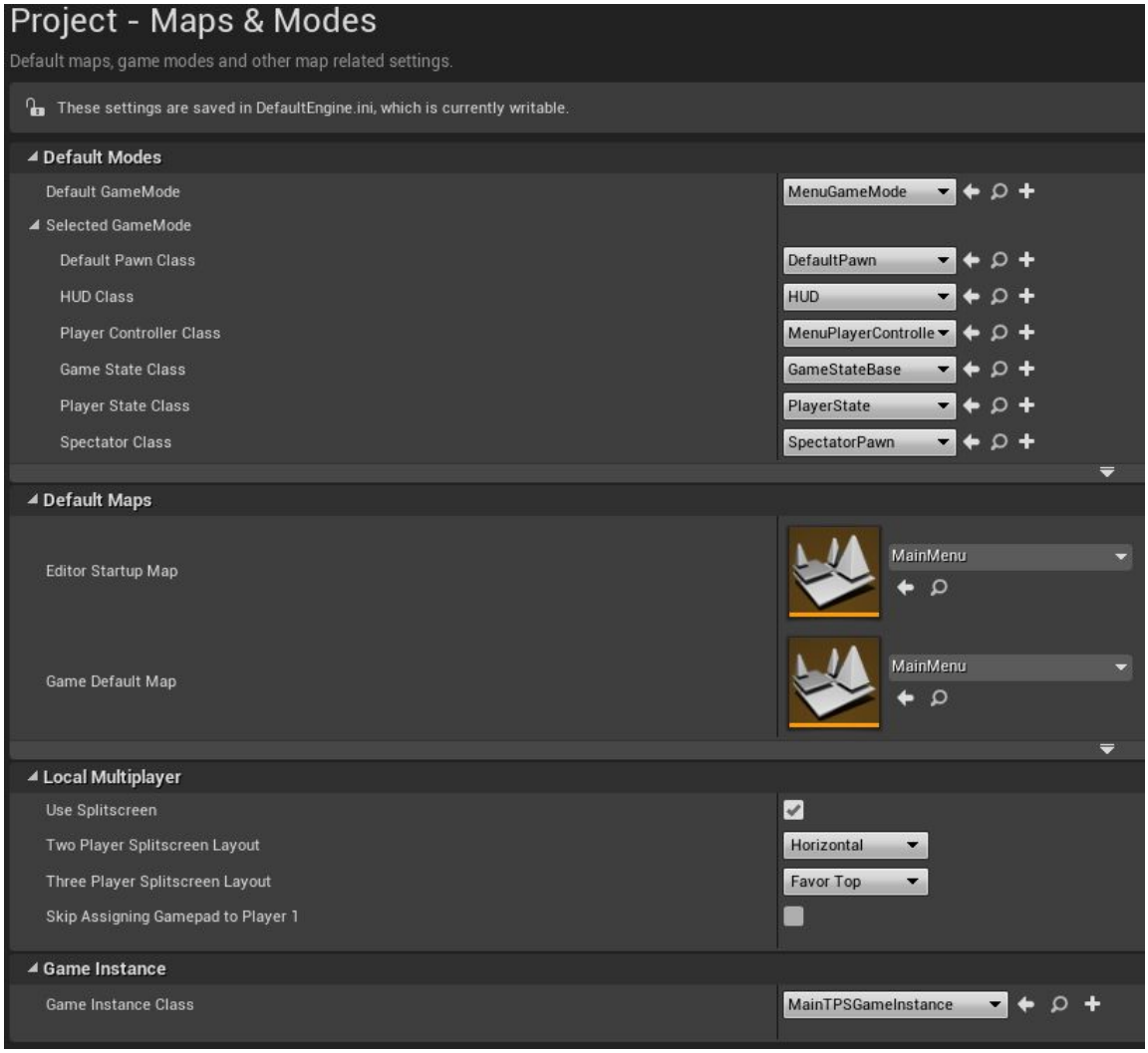
For spawn the characters need to use special spawn points actors: TPSCSSpawnPoint.



Main Menu

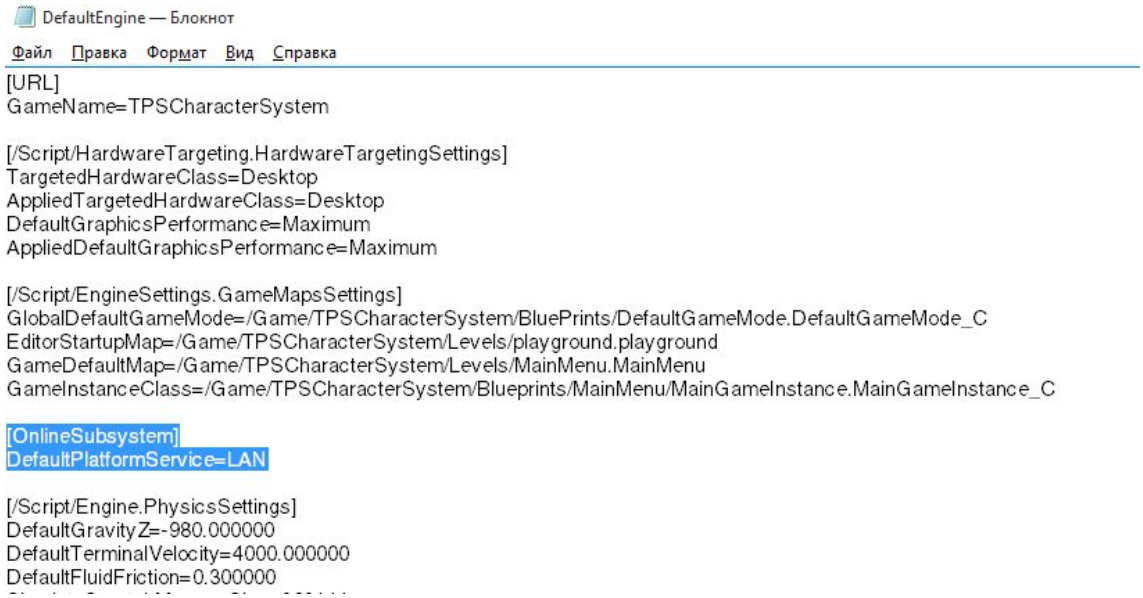
Overview

Main menu can launch if open main menu map.
For the correct operation of the menu, need to set in project setting, in Maps & Modes section:
Default GameMode - MenuGameMode
Editor Startup Map - MainMenu
Game Default Map - MainMenu
Game Instance Class - MainTPSGameInstance



Need to add new strings in DefaultEngine.ini , which is located in TPSCharacterSystem\Config.

[OnlineSubsystem]
DefaultPlatformService=LAN


DefaultEngine — Блокнот

Файл Правка Формат Вид Справка

```
[URL]
GameName=TPSCharacterSystem

[/Script/HardwareTargeting.HardwareTargetingSettings]
TargetedHardwareClass=Desktop
AppliedTargetedHardwareClass=Desktop
DefaultGraphicsPerformance=Maximum
AppliedDefaultGraphicsPerformance=Maximum

[/Script/EngineSettings.GameMapsSettings]
GlobalDefaultGameMode=/Game/TPSCharacterSystem/BluePrints/DefaultGameMode.DefaultGameMode_C
EditorStartupMap=/Game/TPSCharacterSystem/Levels/playground.playground
GameDefaultMap=/Game/TPSCharacterSystem/Levels/MainMenu.MainMenu
GameInstanceClass=/Game/TPSCharacterSystem/Blueprints/MainMenu/MainGameInstance.MainGameInstance_C

[OnlineSubsystem]
DefaultPlatformService=LAN

[/Script/Engine.PhysicsSettings]
DefaultGravityZ=-980.000000
DefaultTerminalVelocity=4000.000000
DefaultFluidFriction=0.300000
```