DTFPS 1.4

User guide

DTFPS v1.4 | DarkTree Development 2020 | Support : darktreedevelopment@gmail.com

Content

Intro

Thank you for purchasing DTFPS. This game template allows you to prototype/create a survival game effortlessly. In this manual, I want to give you comprehensive explanation of tools and methods used in the project and some tutorials to get started with.

The main goal of the asset was to provide simple setting experience and ability to make shooter fast.

But keep in mind that your game requires much more work than just ready template from the assetstore. It will not make a game for you by itself but will give you the great point to start with. And sometimes you still need to work with the code if you want to extend the package and make your game truly unique and original. If you are familiar with C# and can write your own scripts it will be easy to get start with DTFPS setting, however you can still work with provided tools and customize it as you want without scripting.

I hope that you will like it and this bunch of tools and scripts will make your development process much simpler and faster.

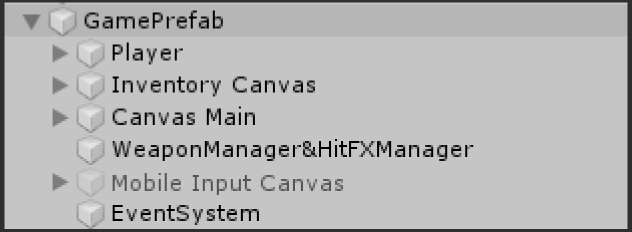
Components overview

It’s time to start from the basics. DTFPS contains a lot of different scripts and gameobjects but we can overview only scripts which we will use in work process.

To start playing with the package you need to drag GamePrefab in your scene. This is a main gameobject that contains player, inventory, save&load system and others.

GamePrefab

Let’s take a look at the GamePrefab expanded hierarchy.



```
▼ GamePrefab
  ▶ Player
  ▶ Inventory Canvas
  ▶ Canvas Main
    WeaponManager&HitFXManager
  ▶ Mobile Input Canvas
    EventSystem
```

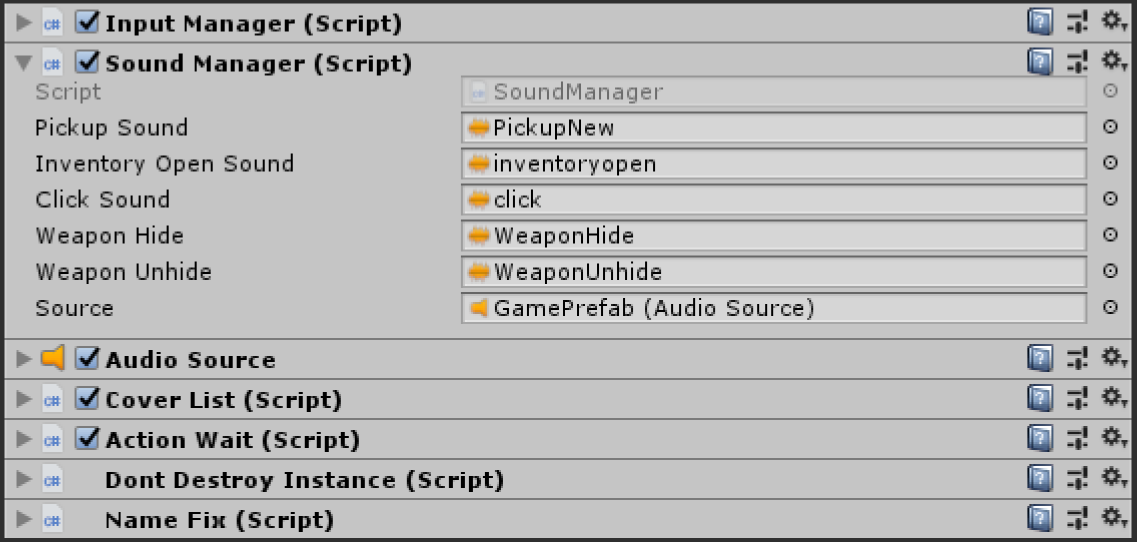
Here we have 6 gameobjects. You can get their functionality with the way they called.
Player – is a player controller that walking, shooting, collecting an items and other.
Inventory Canvas – is the complete inventory system placed on a bunch of canvas objects
Canvas Main – the main UI canvas that contains all visual interfaces like player health, reticle, weapon name and ammo count etc.

WeaponManager&HitFXManager – empty utility gameobject that stores only components required for system work.

Mobile Input Canvas – is a canvas that contains mobile input rig and other mobile stuff.

EventSystem – Unity standard object which required for processing UI events.

Also, we store important components on the GamePrefab. Take a look at the picture bellow.



Input Manager (Script)

Sound Manager (Script)

Script	SoundManager
Pickup Sound	PickupNew
Inventory Open Sound	inventoryopen
Click Sound	click
Weapon Hide	WeaponHide
Weapon Unhide	WeaponUnhide
Source	GamePrefab (Audio Source)

Audio Source

Cover List (Script)

Action Wait (Script)

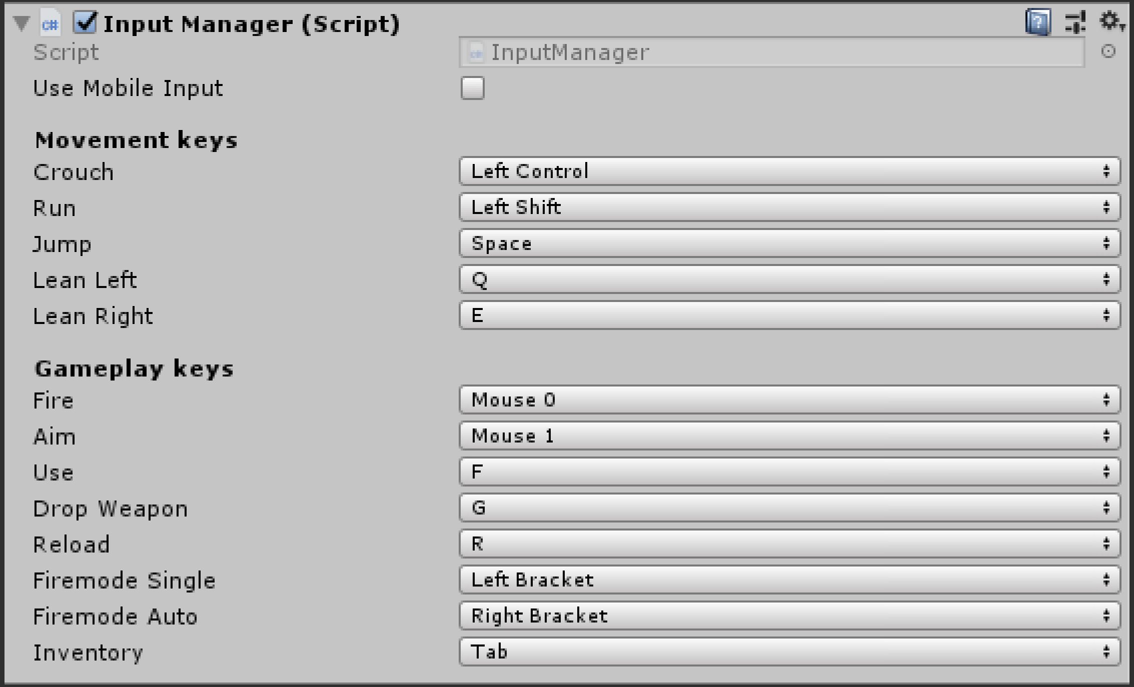
Dont Destroy Instance (Script)

Name Fix (Script)

We will overview the components we need to work with sometimes.

P.S. Utility objects which are not require settings will be ignored

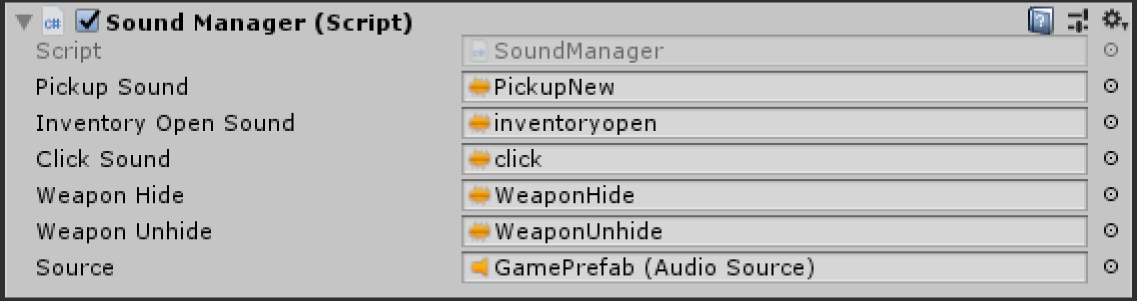
Input Manager – utility script that contains key map for player actions and the toggle for mobile input work



Input Manager (Script)

Script	InputManager
Use Mobile Input	☐
Movement keys	
Crouch	Left Control
Run	Left Shift
Jump	Space
Lean Left	Q
Lean Right	E
Gameplay keys	
Fire	Mouse 0
Aim	Mouse 1
Use	F
Drop Weapon	G
Reload	R
Firemode Single	Left Bracket
Firemode Auto	Right Bracket
Inventory	Tab

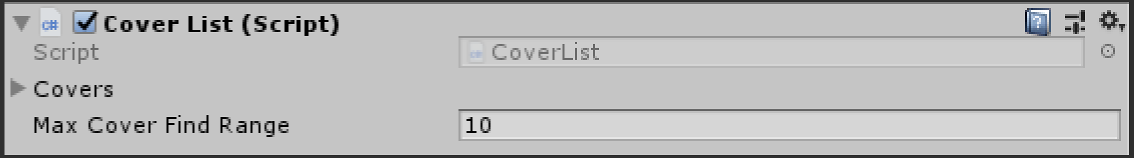
Sound Manager – stores some audio FX for player events



Sound Manager (Script)

Script	SoundManager
Pickup Sound	PickupNew
Inventory Open Sound	inventoryopen
Click Sound	click
Weapon Hide	WeaponHide
Weapon Unhide	WeaponUnhide
Source	GamePrefab (Audio Source)

Cover List is a helper for AI actors. It stores all covers available in the scene and AI takes covers from that. No need to add covers here because it works automatically.

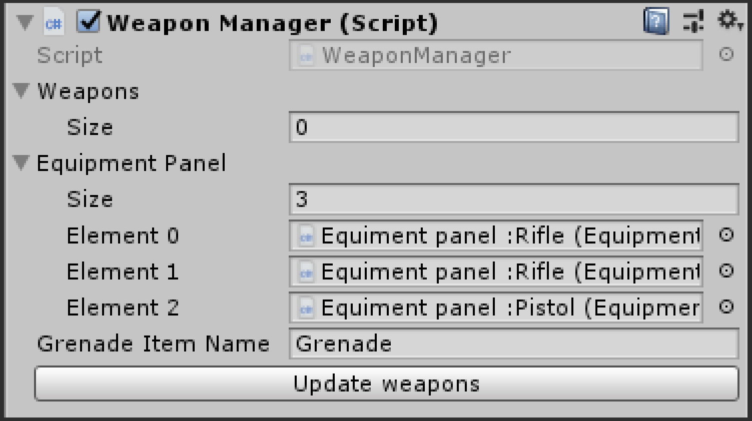


Cover List (Script)

Script	CoverList
Covers	
Max Cover Find Range	10

WeaponManager&HitFXManager

Weapon manager used for weapon states management. This is a component that we need to turn object on/off, hide/unhide weapons, equip/unequip weapons. Shortly, for all thing shared with armed weapon state.

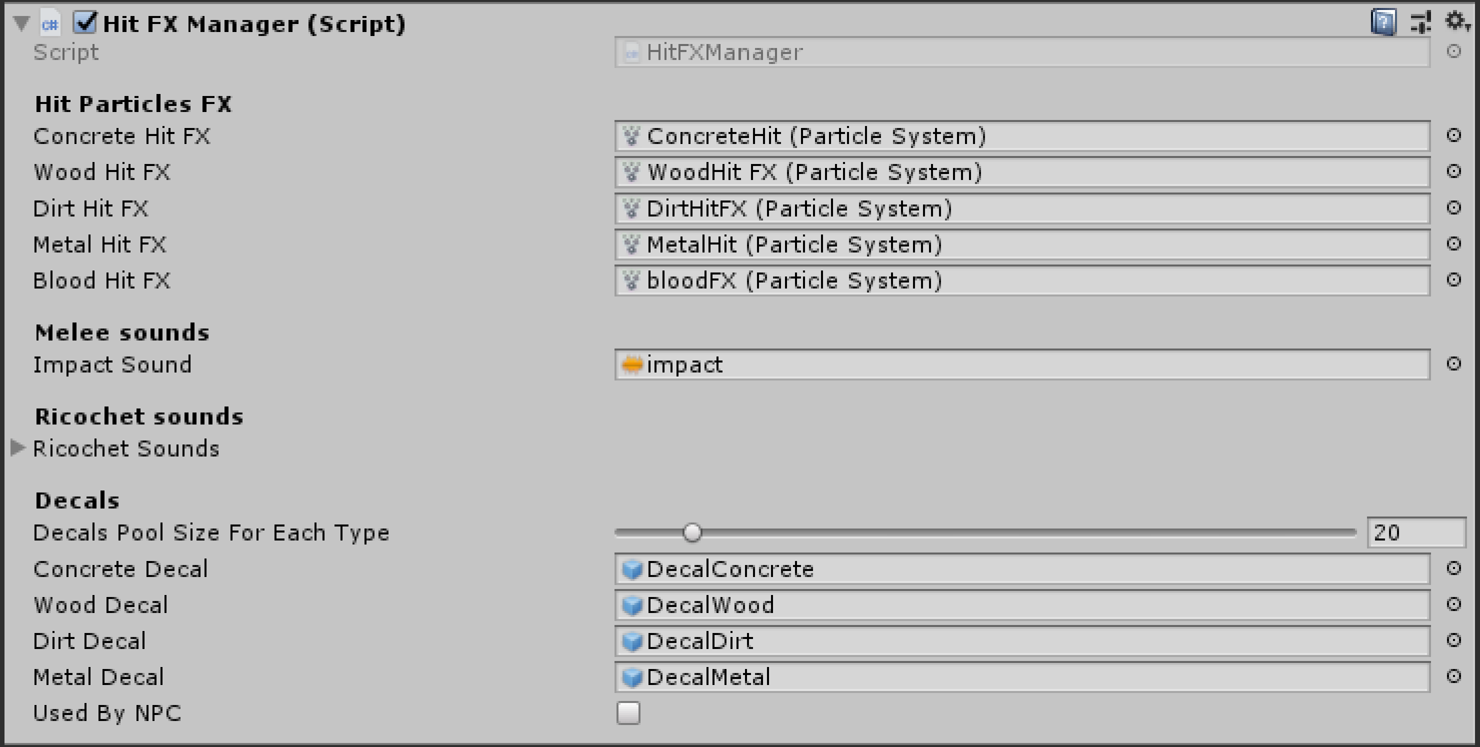


Weapon Manager (Script)

Script	WeaponManager
Weapons	
Size	0
Equipment Panel	
Size	3
Element 0	Equiment panel :Rifle (Equipment
Element 1	Equiment panel :Rifle (Equipment
Element 2	Equiment panel :Pistol (Equipmer
Grenade Item Name	Grenade

Here we have weapons list that we can turn on/off and inventory equipment panels. **Each interactive equipment panel must be put on in this list** in order to give information about equipped weapons.

HitFXManager is the component that stores different effects for melee and bullet hit events.



Hit FX Manager (Script)

Script	HitFXManager
Hit Particles FX	
Concrete Hit FX	ConcreteHit (Particle System)
Wood Hit FX	WoodHit FX (Particle System)
Dirt Hit FX	DirtHitFX (Particle System)
Metal Hit FX	MetalHit (Particle System)
Blood Hit FX	bloodFX (Particle System)
Melee sounds	
Impact Sound	impact
Ricochet sounds	
Ricochet Sounds	
Decals	
Decals Pool Size For Each Type	20
Concrete Decal	DecalConcrete
Wood Decal	DecalWood
Dirt Decal	DecalDirt
Metal Decal	DecalMetal
Used By NPC	☐

Hit Particles are separated by type of surface bullet hit.

Impact sound is a sound being played on melee weapon hit something (will be expanded with different type of surfaces too).

Ricochet sounds are being played on a surface hit.

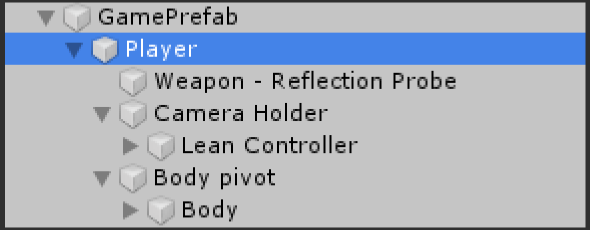
And decals are for different types of surface here as well. You can control Decal pool here and set pool size for each type of decal.

Character

Notice: In previous versions character was made with rigidbody and capsule collider but since version 1.4 physics was switched to CharacterController component.

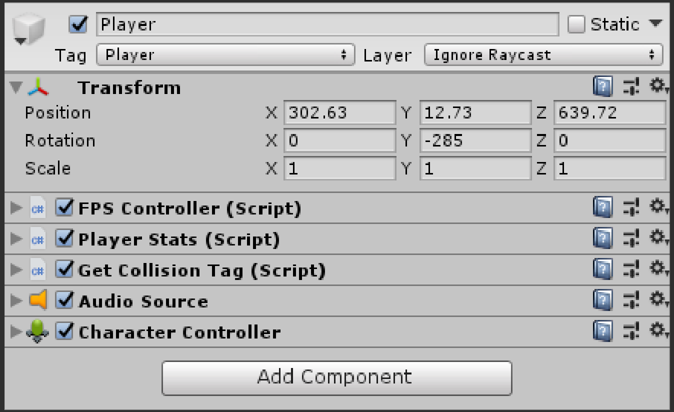
Character is the gameobject that contains all weapons, animations, player stats (health, thirst, hunger), and other controller logic like movement, swimming, and ladder climbing.

Player consists of different gameobjects which you can find in the hierarchy.



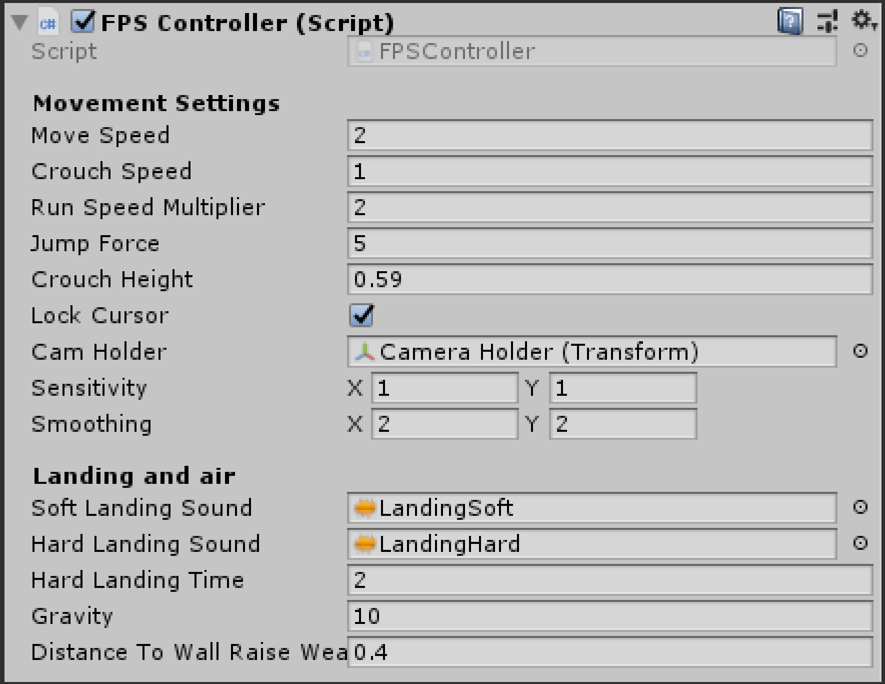
```
▼ GamePrefab
  ▼ Player
      Weapon - Reflection Probe
    ▼ Camera Holder
      ▶ Lean Controller
    ▼ Body pivot
      ▶ Body
```

Player holds 5 components but we interested in FPS Controller and Players Stats only. Let's take a look at them



```
☑ Player                         ☐ Static
Tag Player            Layer Ignore Raycast
▼ Transform
Position    X 302.63   Y 12.73   Z 639.72
Rotation    X 0        Y -285    Z 0
Scale       X 1        Y 1       Z 1
▶ ☑ FPS Controller (Script)
▶ ☑ Player Stats (Script)
▶ ☑ Get Collision Tag (Script)
▶ ☑ Audio Source
▶ ☑ Character Controller
Add Component
```

FPS Controller



FPS Controller (Script)

Script	FPSController
Movement Settings	
Move Speed	2
Crouch Speed	1
Run Speed Multiplier	2
Jump Force	5
Crouch Height	0.59
Lock Cursor	✓
Cam Holder	Camera Holder (Transform)
Sensitivity	X 1 Y 1
Smoothing	X 2 Y 2
Landing and air	
Soft Landing Sound	LandingSoft
Hard Landing Sound	LandingHard
Hard Landing Time	2
Gravity	10
Distance To Wall Raise Wea	0.4

The most of fields are describe themselves with the name they has. I'll describe only important fields here.

Lock Cursor – Is cursor should being disabled on start?
Cam Holder – Is the object that contains camera child
Sensitivity – mouse sensitivity for each axis
Smoothing – smoothing coefficient for mouse axes

Landing and air are settings for player air states.

Soft Landing Sound – sound being played when player hit the ground after jump if air time is not bigger than 'HardLandingTime'

Hard Landing Sound – being played if 'HardLandingTime' is bigger than air time.

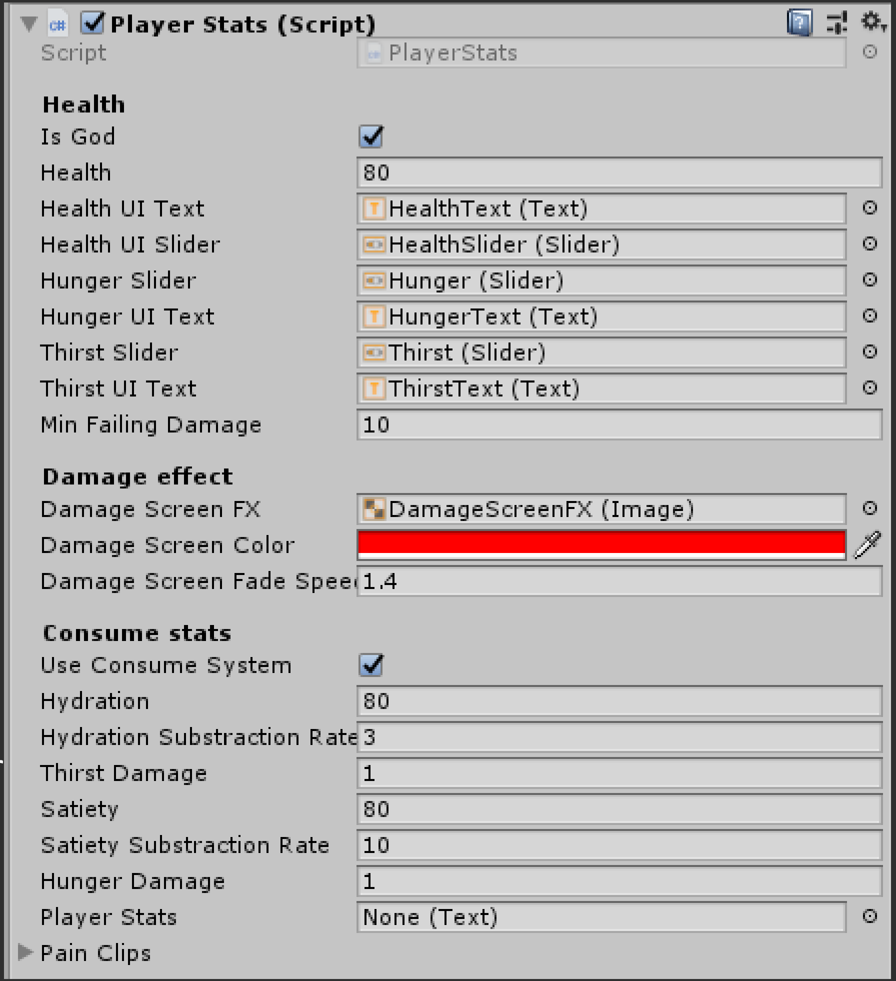
Gravity – gravity value applied to the player down force

Distance To Wall Raise weapon – distance to wall when player should play weapon raise animation.

Player stats

The script has player life stats, linked settings and objects.

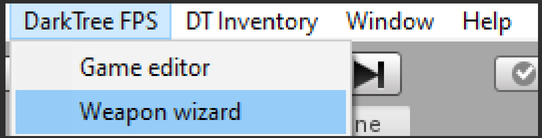
- Is God – player immortal if true
- Health – player health point (int)
- Health UI Text – a text to draw hp on the screen
- Health UI Slider – a slider to draw hp stripe
(All the same for hunger and thirst)
- Min Failing Damage – Minimal damage if player falls from height
- Damage Screen FX – an UI image that will be shown if player get damage
- Damage Screen Color – additional color to control fx image color.
- Damage Screen Fade Speed – how fast damage fx will disappear
- Use Consume system – if true player's life stats will decrease after time
- Hydration, satiety are survival live stats.
- Substraction rate – is a time to decrease one point of stat (in sec).
- Hunger/satiety damage – damage point to decrease player health if stat is zero and subtraction time up to time.
- Pain Clips – Audioclips to play on player hurt (height failing, bullet hit etc.)


Player Stats (Script)

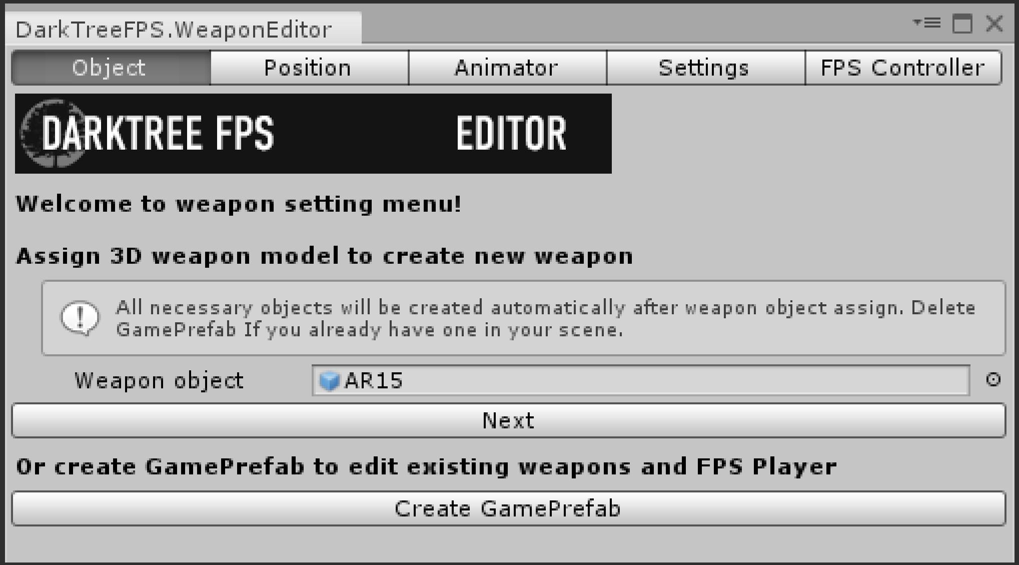
Script	PlayerStats
Health	
Is God	✓
Health	80
Health UI Text	HealthText (Text)
Health UI Slider	HealthSlider (Slider)
Hunger Slider	Hunger (Slider)
Hunger UI Text	HungerText (Text)
Thirst Slider	Thirst (Slider)
Thirst UI Text	ThirstText (Text)
Min Failing Damage	10
Damage effect	
Damage Screen FX	DamageScreenFX (Image)
Damage Screen Color	🟥
Damage Screen Fade Spee	1.4
Consume stats	
Use Consume System	✓
Hydration	80
Hydration Substraction Rate	3
Thirst Damage	1
Satiety	80
Satiety Substraction Rate	10
Hunger Damage	1
Player Stats	None (Text)
Pain Clips	

Weapons

You can create any weapon with DTFPS. You can create rifle, pistol, sniper, rocket launcher, and any other firearm weapon with Weapon Wizard



To start creating new weapon you need to apply your weapon model to weapon object field and press next


DarkTreeFPS.WeaponEditor

Object | Position | Animator | Settings | FPS Controller

DARKTREE FPS EDITOR

Welcome to weapon setting menu!

Assign 3D weapon model to create new weapon

All necessary objects will be created automatically after weapon object assign. Delete GamePrefab If you already have one in your scene.

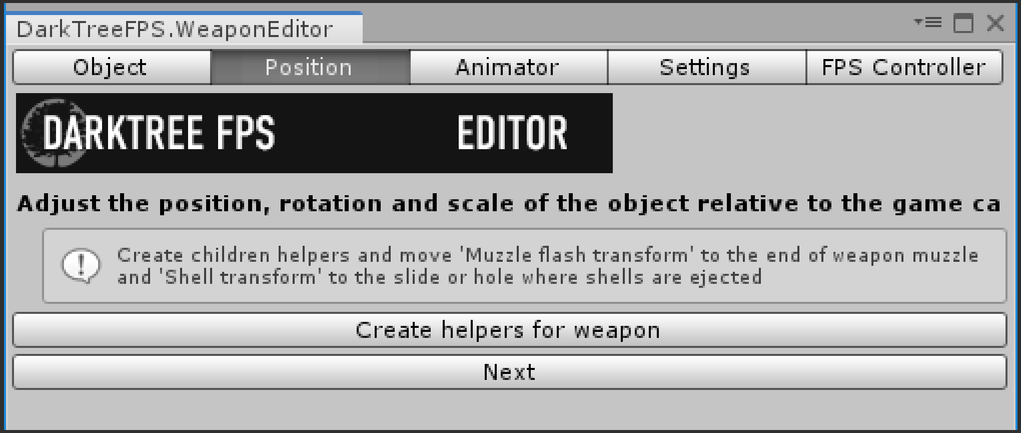
Weapon object: AR15

Next

Or create GamePrefab to edit existing weapons and FPS Player

Create GamePrefab

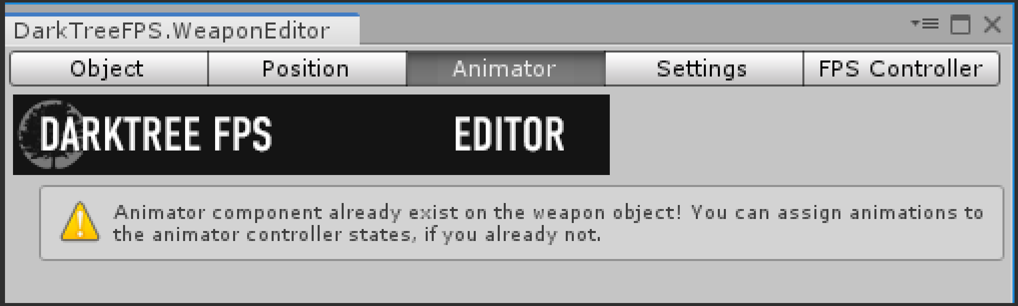
After that you need to create weapon helpers and set them up to correct position. Muzzle flash transform must be at the end of weapon barrel and shell transform must be at the place where shell are being ejected. Also setup your weapon with the camera position and make sure that all weapon child are on ‘Weapon’ layer.



Adjust the position, rotation and scale of the object relative to the game ca

Create children helpers and move 'Muzzle flash transform' to the end of weapon muzzle and 'Shell transform' to the slide or hole where shells are ejected

Next step is to set up animations for your new weapon. When you go to Animator tab you'll probably see this message…



Animator component already exist on the weapon object! You can assign animations to the animator controller states, if you already not.

Often, unity generates animator component for each skinned meshes and now you have default animator on your weapon object. Remove that.

Now your Animator tab must look like the picture below.

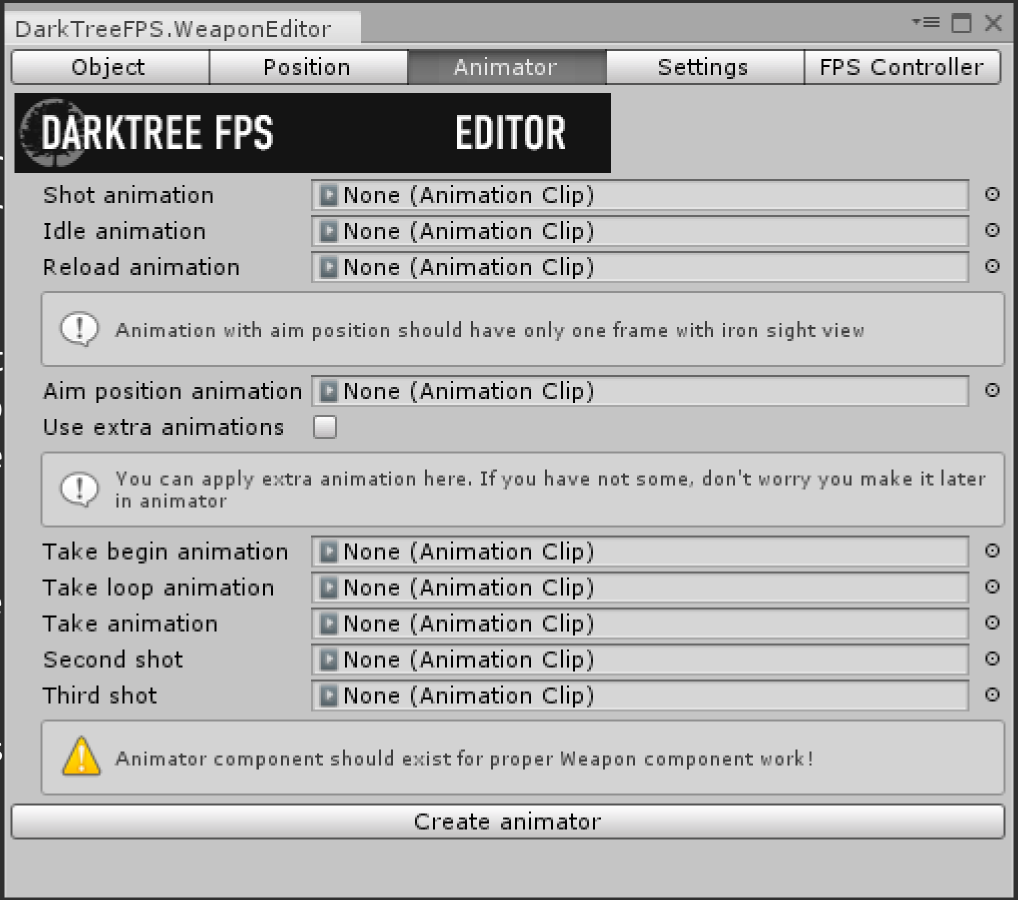
Let's see to fields we have there.

Only 4 animation clips required for weapon proper work. It is shot, idle, reload, aim animations. Other animation are additional and not required by default.

We can animate item take events. When player look at item take begin animation will started then take loop animation. And if you take item take animation will be played. It's advanced item interaction animations.

Second shot and third shot animations are to make shooting animation various.

After you done with animation clips setting press 'Create animator' and go forward.



Shot animation	None (Animation Clip)
Idle animation	None (Animation Clip)
Reload animation	None (Animation Clip)

Animation with aim position should have only one frame with iron sight view

Aim position animation	None (Animation Clip)
Use extra animations	☐

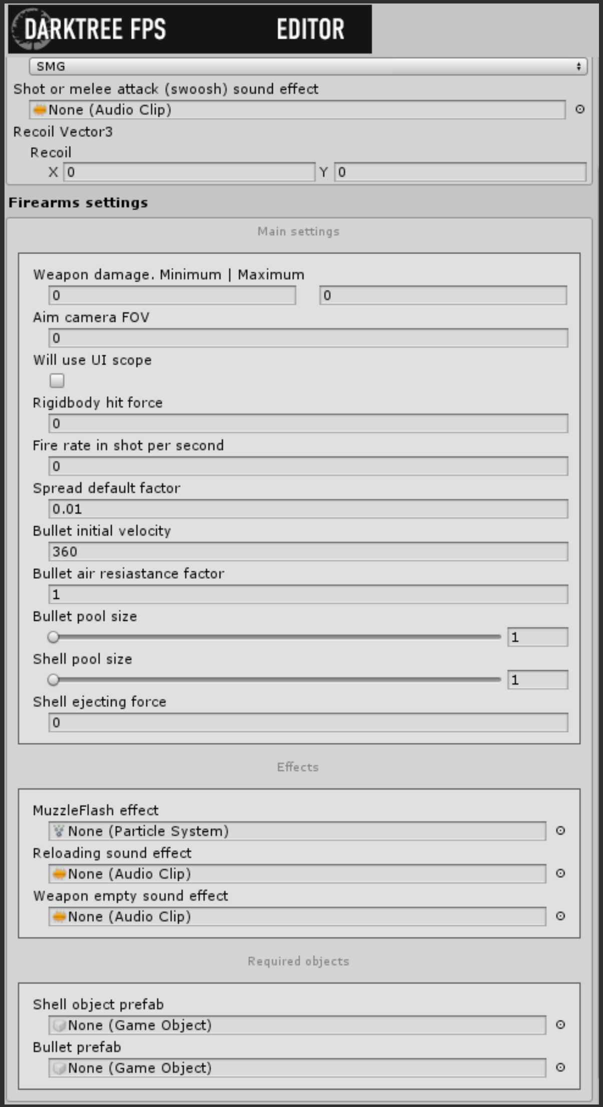
You can apply extra animation here. If you have not some, don't worry you make it later in animator

Take begin animation	None (Animation Clip)
Take loop animation	None (Animation Clip)
Take animation	None (Animation Clip)
Second shot	None (Animation Clip)
Third shot	None (Animation Clip)

Animator component should exist for proper Weapon component work!

Weapon setting tab will ask you to create new weapon scriptable object. You need to click that button and after that you will see new weapon settings list.

Weapon type	First field is weapon type.
Shot or melee attack sound effect	Second is weapon attack sound (shot, melee attack swoosh etc.)
Recoil Vector	Recoil will move camera to some value on x or y axis on shot.
Weapon damage	Weapon damage value has min and max damage. Final damage calculates with random range between these two values.
Aim camera FOV	Aim camera FOV changes main camera FOV on aim
Will use UI scope	Set will use UI scope if your weapon should use aiming with scope
Rigidbody hit force	Rigidbody hit force value applied to rigidbody object on hit
Fire rate in shot per second	Fire rate in shot per seconds tells how fast your weapon will shot bullets
Spread default factor	Spread factor is value that applied to spread vector for bullet direction
Bullet initial velocity	Bullet initial velocity in meter per seconds. You can set realistic bullet speed.
Bullet air resistance factor	Air resistance factor influence on bullet speed and works like air friction.
Bullet pool size/ Shell pool size	And here is pools for bullets and shells. They are should have the same value as weapon ammo

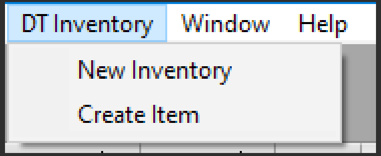


SMG
Shot or melee attack (swoosh) sound effect
None (Audio Clip)
Recoil Vector3
Recoil
X 0 Y 0
Firearms settings
Main settings
Weapon damage. Minimum | Maximum
0 0
Aim camera FOV
0
Will use UI scope
☐
Rigidbody hit force
0
Fire rate in shot per second
0
Spread default factor
0.01
Bullet initial velocity
360
Bullet air resiastance factor
1
Bullet pool size
1
Shell pool size
1
Shell ejecting force
0
Effects
MuzzleFlash effect
None (Particle System)
Reloading sound effect
None (Audio Clip)
Weapon empty sound effect
None (Audio Clip)
Required objects
Shell object prefab
None (Game Object)
Bullet prefab
None (Game Object)

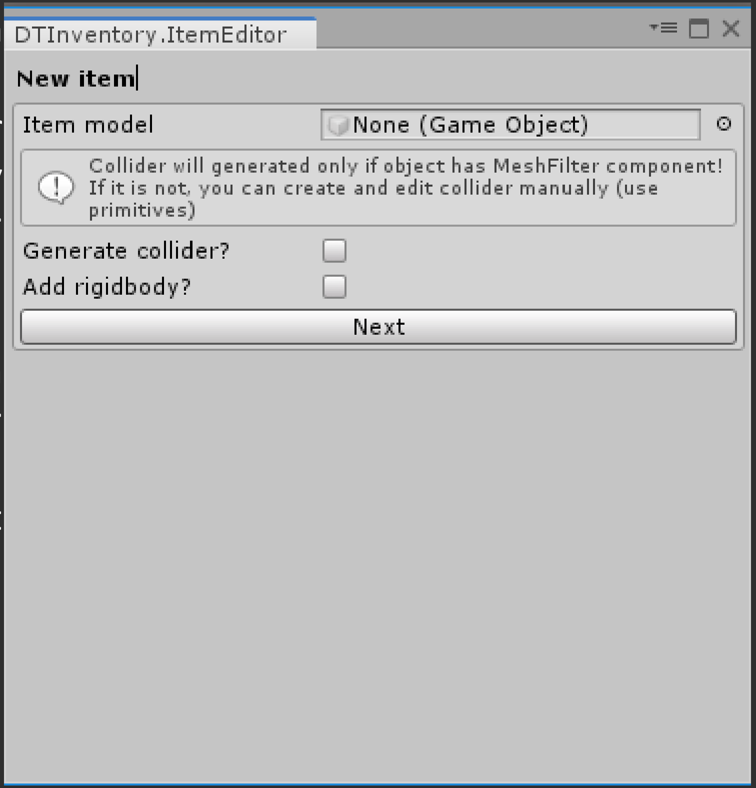
	capacity
Shell ejecting force	Force will be applied to shell rigidbody on shot
Muzzle flash effect	Particle system to play muzzle flash effect on shot. Will be spawned on muzzle flash transform
Reloading sounds effect	
Weapon empty sound effect	Played if weapon has no ammo but player shooting
Shell object prefab	Find in prefabs folder
Bullet prefab	Find in prefabs folder

After setting finish you need to turn off your weapon in inspector and create weapon item. You cant leave weapons enabled due to inventory system limitations. All weapons must be associated with item objects and now we'll create this one.

Go to DTInventory tab and click Create Item button



Drag your weapon model (without hands and animations) to item model field. You can add rigidbody and collider to your object here but in collider will be generated only if your gameobject has meshfilter component (gameobject is mesh). Usually, meshes grouped to empty gameobject and has no meshes on it, In this case you can create collider by yourself. Now press next.


DTInventory.ItemEditor

New item

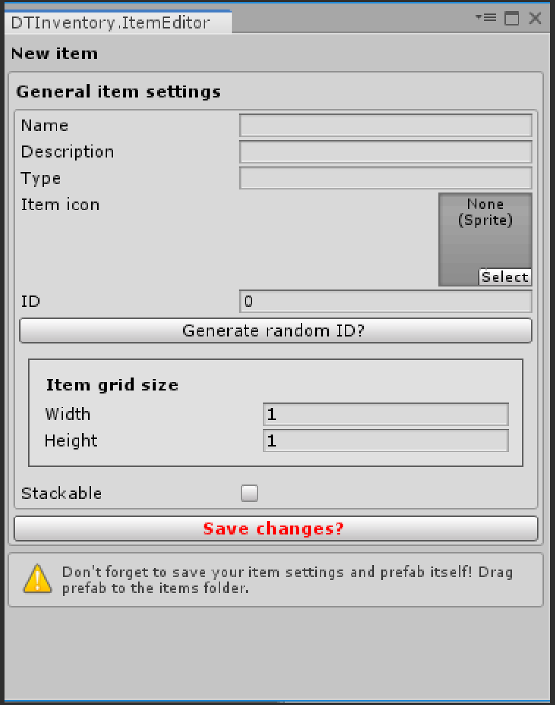
Item model: None (Game Object)

Collider will generated only if object has MeshFilter component! If it is not, you can create and edit collider manually (use primitives)

Generate collider? ☐

Add rigidbody? ☐

Your new item will be instantiated at your scene. Complete item fields and continue. Make sure that item Type is the same as your equipment panel allowed type. In other case, item will not be equipped and used in inventory! To learn more about item fields look at DTInventory manual


DTInventory.ItemEditor

New item

General item settings

Name:

Description:

Type:

Item icon: None (Sprite)

ID: 0

Item grid size

Width: 1

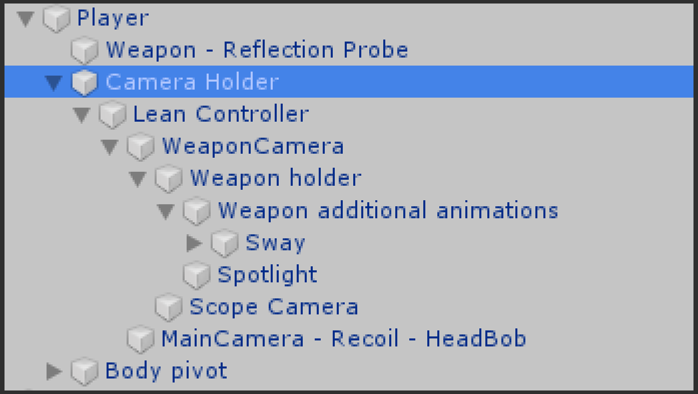
Height: 1

Stackable ☐

Don't forget to save your item settings and prefab itself! Drag prefab to the items folder.

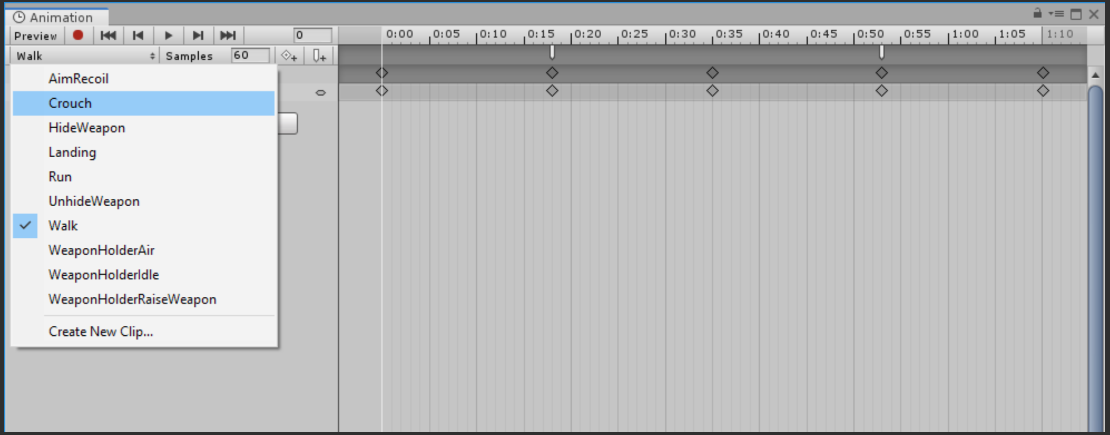
Controller animations

As you may already know, we use animations to make weapon walk bobbing, camera bobbing, animate weapon changing and some other. We need get to know the main components we use and animate in the system.



Lean controller is procedural transform that allows player to lean out of corners. It has special Lean script with obstacle check so you won't dive your head through the wall.

Weapon holder is an animator of walking, running, crouching,
jumping etc. Also there is an animations to show/hide weapon. To edit these animation you need to select weapon holder and open animation window.



Sway gameobject allows your weapons to sway when camera is moving.

And last one is the MainCamera – Recoil – HeadBob. Here we have animator that animate camera bobbing. You can examine animations with opening animator. It is the same process as for weapon holder.

Inventory and Items

You can learn about inventory and items in DTInventory manual. You can find it in DTFPS folder.

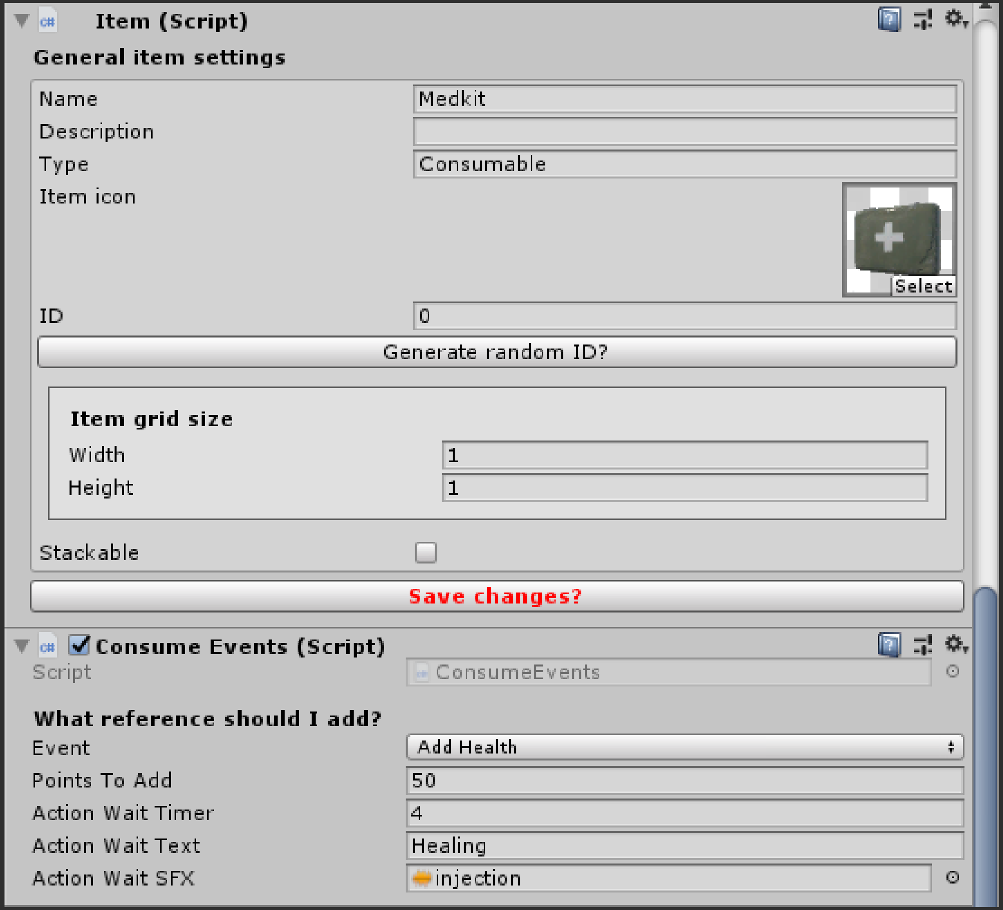
But you need to know about specific components of DTFPS item system. And it is consume events script.

We call this script events when use some item. There is already events to add health, satiety, hydration.

Action Wait Timer will freeze player for some time. You can set timer to zero if you do not want item use waiting.

Action wait text will show what action player performing at this time. For example, we can use medkit and freeze player for healing time and show 'healing' text on screen.

Also we can attach sfx for event action.



Item (Script)

General item settings

Name	Medkit
Description	
Type	Consumable
Item icon	Select
ID	0

Generate random ID?

Item grid size

Width	1
Height	1

Stackable ☐

Save changes?

Consume Events (Script)

Script	ConsumeEvents

What reference should I add?

Event	Add Health
Points To Add	50
Action Wait Timer	4
Action Wait Text	Healing
Action Wait SFX	injection

Building system

Building system allows you to create building using inventory items. For some people it looks complicated and hard but it's very easy in fact.

Let's see how to create new building.

We need to create new building scriptable object first. Right click in project view → create → building data.

Let's what we've got here…

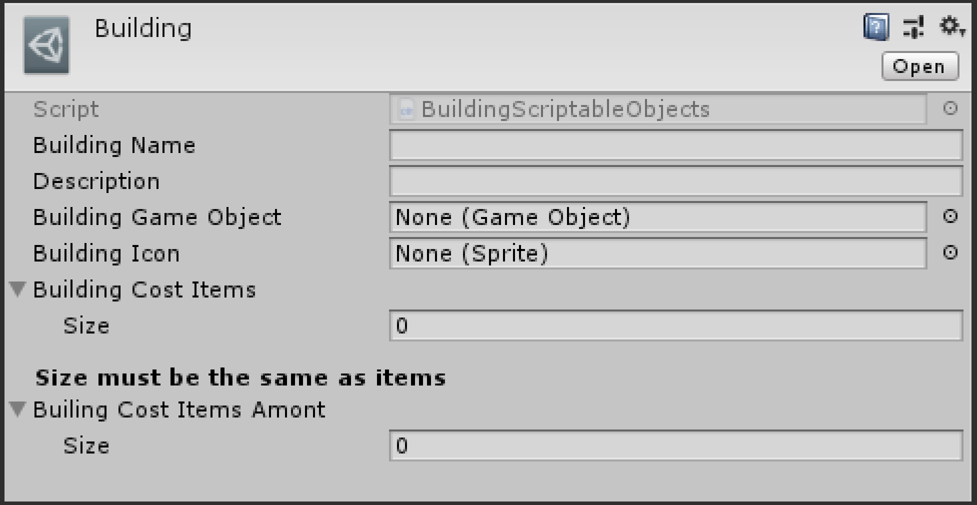
Building name will drawn in building select window

Description is not used for now but will show building description in future.

Building Game Object – it's the next step. I will tell about it little bit later.

Assign building icon

And final building cost items. This array should contain single items that we use for building. Cost Items array should contain cost of each array item for building. Take a look at the picture bellow.



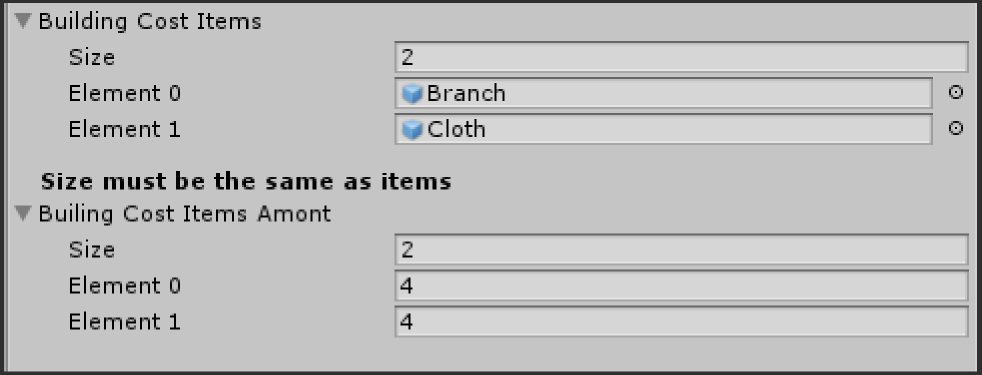
Script	BuildingScriptableObjects
Building Name	
Description	
Building Game Object	None (Game Object)
Building Icon	None (Sprite)
Building Cost Items	
Size	0

Size must be the same as items

Builing Cost Items Amont	
Size	0

Here we have Branch and Cloth items which are required for shelter building. Cost for both items are 4. It means we need to have 4 Branches and 4 Cloth items in inventory in order to build a shelter.

We missed Building Game Object field. Do you remember? Now I will show you how to make it.



Building Cost Items

Size	2
Element 0	Branch
Element 1	Cloth

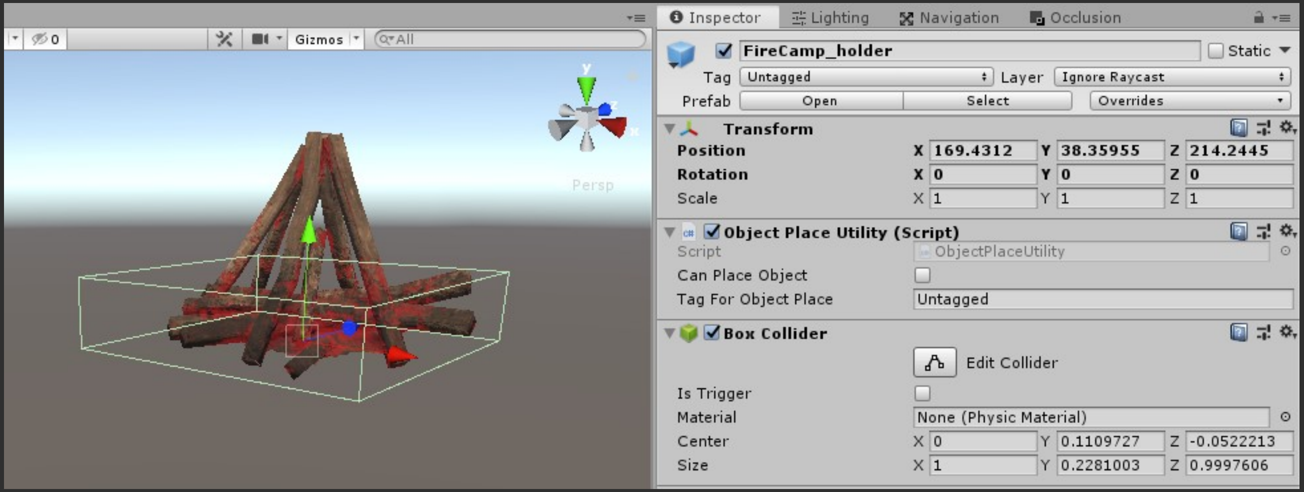
Size must be the same as items

Builing Cost Items Amont

Size	2
Element 0	4
Element 1	4

This is firecamp building (you can find it in your assets folder). Pivot point of the building must be on ground because if your pivot somewhere at the middle, you object will fall through the ground in a half size. Object layer is ‘Ignore Raycast’, set it for your own buildings too. Attach Object Place Utility script and set surface tag for placement.

And we must have a collider for the object in order to touch the ground with it. Collider must be little bit lower of object pivot (bellow the ground little bit). Now drag you new object to prefabs folder and set it to scriptable object field.



FireCamp_holder — Static
Tag: Untagged — Layer: Ignore Raycast
Prefab: Open | Select | Overrides

Transform

	X	Y	Z
Position	169.4312	38.35955	214.2445
Rotation	0	0	0
Scale	1	1	1

Object Place Utility (Script)

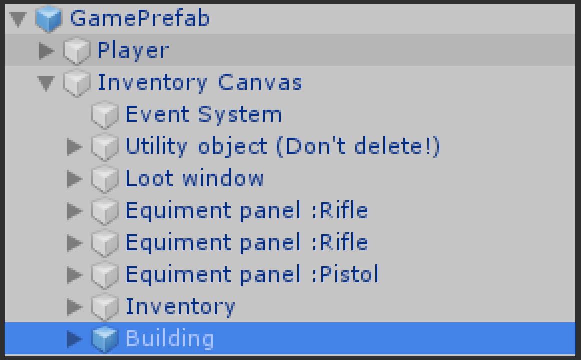
Script	ObjectPlaceUtility
Can Place Object	☐
Tag For Object Place	Untagged

Box Collider

Edit Collider

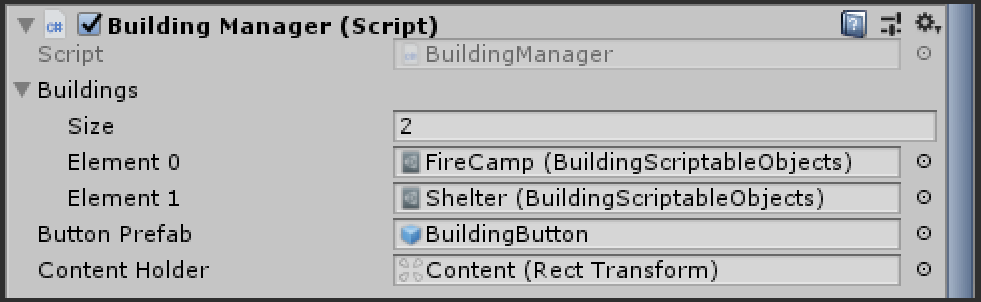
	X	Y	Z
Is Trigger	☐		
Material	None (Physic Material)		
Center	0	0.1109727	-0.0522213
Size	1	0.2281003	0.9997606

Last thing we have to do. Is to attach our new scriptable object to building manager list. Find it in the hierarchy.



GamePrefab
- Player
- Inventory Canvas
  - Event System
  - Utility object (Don't delete!)
  - Loot window
  - Equiment panel :Rifle
  - Equiment panel :Rifle
  - Equiment panel :Pistol
  - Inventory
  - Building

You will see Building Manager Script in the inspector. Drag and drop your new building scriptable object to buildings array. It's all. Now you can build your own objects. Graphics data and UI will be automatically generated.



Building Manager (Script)

Script	BuildingManager
Buildings	
Size	2
Element 0	FireCamp (BuildingScriptableObjects)
Element 1	Shelter (BuildingScriptableObjects)
Button Prefab	BuildingButton
Content Holder	Content (Rect Transform)

Tutorials and help

Check out tutorials on YouTube chanel :

https://www.youtube.com/channel/UCvn3YCMFoIj_nJ5hG4dGjfA

Discord link :

https://discord.gg/r6wRYA2