

ИНСТРУКЦИЯ

ИНВЕНТАРЬ И СИСТЕМА МОДДИНГА ОРУЖИЯ (ICWM) версия 7.9.0b

ВВЕДЕНИЕ

Если что-либо непонятно, вы можете связаться со мной и задать интересующиеся вопросы.

Данная инструкция создана для того, чтобы дать пользователю представление о данном активе. Что это за актив и его особенности были рассмотрены в документации. ICWM состоит на 94% из кода (C#). Пользователям, которые знают азы этого языка программирования или подобных языков, будет легко разобраться с логикой актива. Изначально планировалось создать только систему моддинга, но в итоге стало ясно, что трудно было бы осуществить взаимосвязь между инвентарем и системой моддинга. Было решено совместить их.

Вероятней всего, что пользователь будет находить ошибки. На данный момент у актива есть свои требования по работе с ним, их вы найдете в приложении. Не соблюдение этих требований может привести к ошибкам или программа будет работать не должным образом. В будущем они будут упрощаться до автоматизма. Это сложная задача, т.к. нужно перестраивать всю логику ICWM.

Код ICWM прокомментирован в тех местах, где непонятен смысл, который несет данный раздел скрипта. Я не стал комментировать элементарные операции кода.

Оглавление

ВВЕДЕНИЕ.....	2
ДО ЗАПУСКА.....	4
КАК НАЧАТЬ?	5
КАК ДОБАВИТЬ ЭТО НА НОВУЮ СЦЕНУ?	8
КАК АКТИВИРОВАТЬ STASH?	10
КАК ДОБАВИТЬ НОВЫЙ РАЗМЕР ДЛЯ РЮКЗАКА?	11
КАК ДОБАВИТЬ НОВУЮ КНОПКУ В МИНИ-МЕНЮ?.....	14
КАК ДОБАВИТЬ НОВЫЙ ТЭГ ДЛЯ АТТАЧЕЙ?.....	15
КАК ДОБАВИТЬ НОВЫЙ ПИСТОЛЕТ?.....	16
КАК ДОБАВИТЬ НОВОЕ КОЛЮЩЕЕ ОРУЖИЕ?.....	17
КАК ДОБАВИТЬ НОВУЮ РАЗГРУЗКА, БРОНЮ, РЮКЗАК И ТД?.....	18
КАК ДОБАВИТЬ ВЕЩЬ В СПИСОК ЛУТА	18
ПРИЛОЖЕНИЕ	19

ДО ЗАПУСКА

Выставить соответствующее разрешение экрана. Убрать Free Aspect или Aspect Ration и поставить Fixed Resolution, в противном случае некоторые элементы инвентаря будут не помещаться на экране, как показано на первом изображении. На втором изображении приведен список разрешений экрана, которые были протестированы.



Free Aspect
5:4
4:3
3:2
16:10
16:9
Standalone (1024x768)
10x10
✓ 1920x1080
1024x1024 (1024:1024)
2048x1152 (2048:1152)
3200x1800 (3200:1800)
1024x576 (1024:576)
1600x900 (1600:900)
2880x1800 (2880:1800)
1024x768 (1024:768)
1280x720 (1280:720)
1366x768 (1366:766)
1280x760 (1280:760)
1152x768 (1152:768)
2560x1440 (2560:1440)
+

КАК НАЧАТЬ?

При скачивании актива у вас появится папка с именем *AssetICWM*:



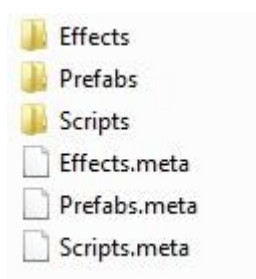
Открыв его:

DocumentationManualAndLinks	16.02.2019 15:24	Папка с файлами	
GamePrefabs	15.02.2019 18:47	Папка с файлами	
ghtmap	30.08.2018 19:33	Папка с файлами	
Inventory	16.02.2019 11:17	Папка с файлами	
Mesh	17.02.2019 14:52	Папка с файлами	
ModdingPrefabs	16.02.2019 17:00	Папка с файлами	
Resources	30.08.2018 19:33	Папка с файлами	
SCENESToExpore	16.02.2019 11:30	Папка с файлами	
DocumentationManualAndLinks.meta	27.09.2017 22:52	Файл "МЕТА"	1 КБ
GamePrefabs.meta	20.05.2017 20:48	Файл "МЕТА"	1 КБ
ghtmap.meta	18.03.2017 1:23	Файл "МЕТА"	1 КБ
ICWM_CarModding.unity	16.02.2019 11:10	Unity scene file	427 КБ
ICWM_CarModding.unity.meta	19.11.2017 17:00	Файл "МЕТА"	1 КБ
ICWM_Inv_EFT.unity	16.02.2019 17:00	Unity scene file	227 КБ
ICWM_Inv_EFT.unity.meta	07.11.2017 23:44	Файл "МЕТА"	1 КБ
ICWM_Inv_RE4_OriginalStyle.unity	16.02.2019 12:08	Unity scene file	105 КБ
ICWM_Inv_RE4_OriginalStyle.unity.meta	07.11.2017 23:44	Файл "МЕТА"	1 КБ
ICWM_Inv_RE4_VooStyle.unity	16.02.2019 12:09	Unity scene file	105 КБ
ICWM_Inv_RE4_VooStyle.unity.meta	07.11.2017 23:44	Файл "МЕТА"	1 КБ
Inventory.meta	29.04.2017 15:27	Файл "МЕТА"	1 КБ
Mesh.meta	28.04.2017 22:38	Файл "МЕТА"	1 КБ
ModdingPrefabs.meta	05.02.2019 19:50	Файл "МЕТА"	1 КБ
Resources.meta	12.11.2017 16:28	Файл "МЕТА"	1 КБ
SampleScene.unity	16.02.2019 12:13	Unity scene file	160 КБ
SampleScene.unity.meta	15.06.2018 8:58	Файл "МЕТА"	1 КБ
SCENESToExpore.meta	18.11.2017 17:01	Файл "МЕТА"	1 КБ

Где:

- 1) DocumentationManualAndLinks – текстовая документация;
- 2) GamePrefabs – все игровые префабы;
- 3) ghtmap – её можете удалить, это карта освещения baking of lightmap;
- 4) Inventory – все, что необходимо для этого актива (скрипты, иконки, материалы и т.д.);
- 5) Mesh – все игровые модели, текстуры, материалы;
- 6) ModdingPrefabs – все игровые префабы, с которыми взаимодействует инвентарь;
- 7) Resources – некоторый материал для меню в верхней панели и скриптов;
- 8) ScensToExpore – сцены для изучения особенностей актива по отдельности.

Inventory:



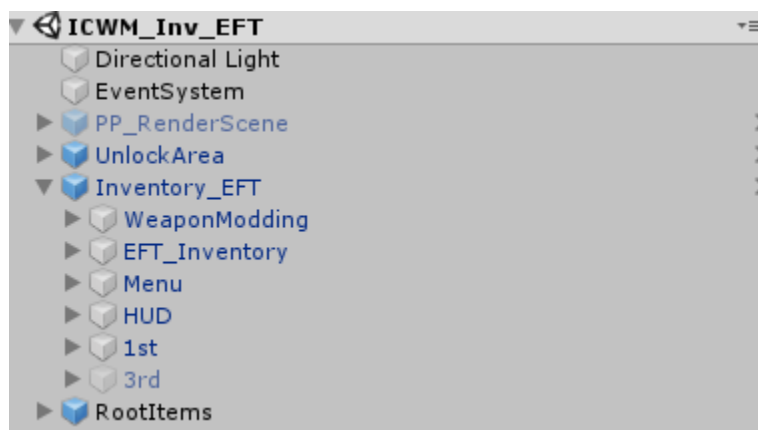
Где:

- 1) Effects – тут все иконки, текстуры, материалы, которые используются в инвентаре и моддинге;
- 2) Prefabs – все префабы, которые необходимы для работы ассета;
- 3) Scripts – скрипты, я постарался расформировать все по папкам, что к чему относится.

После запуска Unity нужно открыть любую сцену. После того, как вы откроете его:

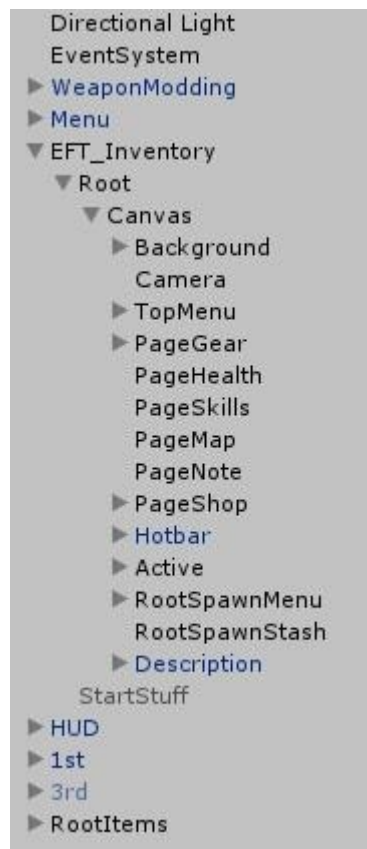


Рассмотрим подробнее окно иерархии:



Где:

- 1) Элементы PP_RenderScene и UnlockArea рассмотрены в документации по Unlock System;
- 2) WeaponModding – тут происходит все, что связано с моддингом оружия (это полотно со своими особенностями);
- 3) EFT_Inventory – наш инвентарь;
- 4) Menu - это меню паузы (по нажатию на Esc);
- 5) HUD – самый простой, там ничего сложного нет;
- 6) Player (1st – 3rd) – наш игрок FPS и TPS;
- 7) RootItems – все вещи которые есть в сцене, а также туда спавнятся все предметы, которые были выброшены из инвентаря.



WeaponModding:

Где:

- 1) WMRoot – туда спавнятся наше оружие и там же создаются иконки;
- 2) WeaponRenderScreen – тут происходит рендер изображения для иконки оружия;
- 3) PlayerInspectWindow – рендер картинки для окна изучения (Inspect);
- 4) Root... - туда спавнятся аттачи, оружия. Т.к. нам не нужно видеть их.

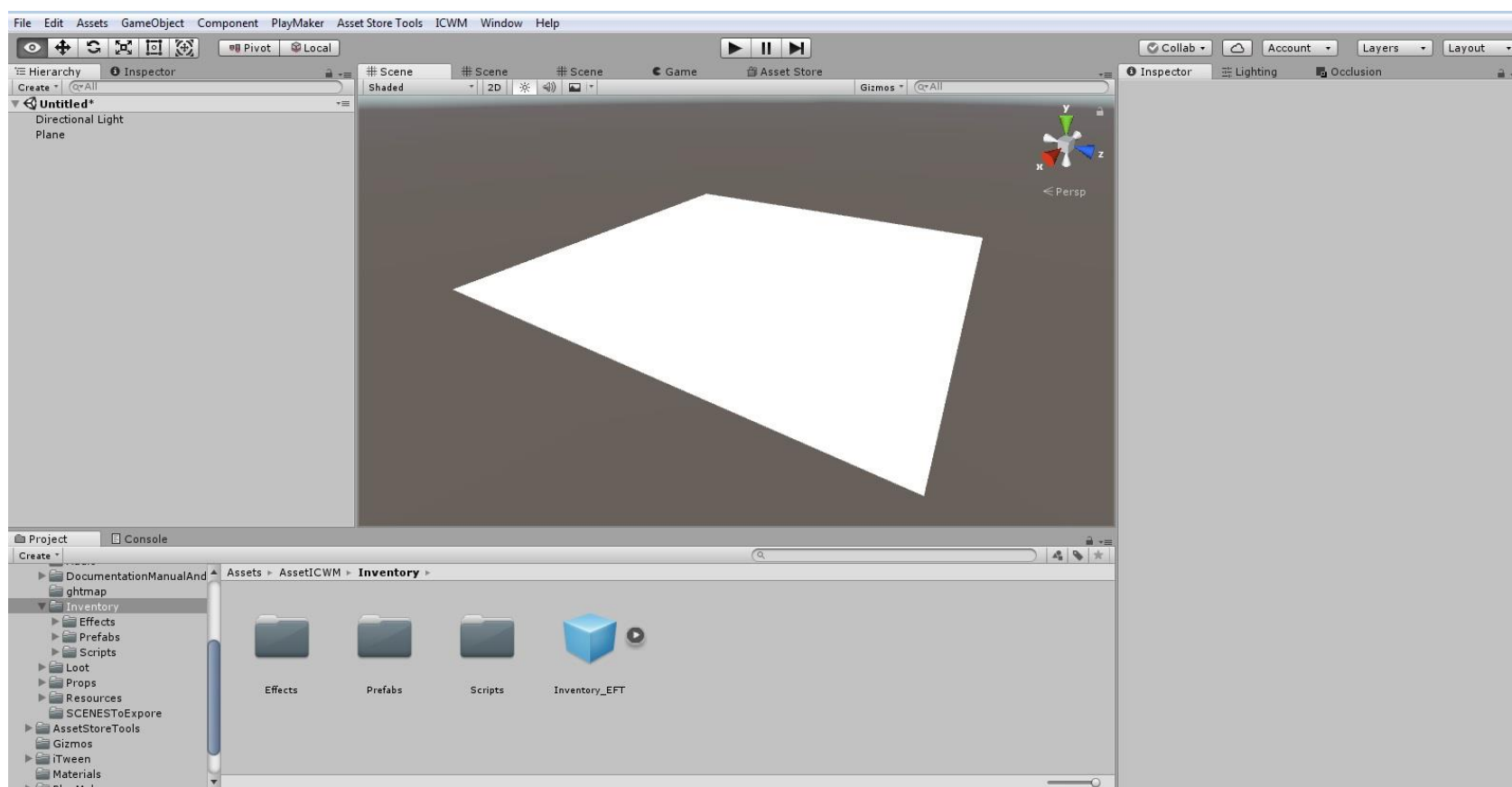
Inventory:

Где:

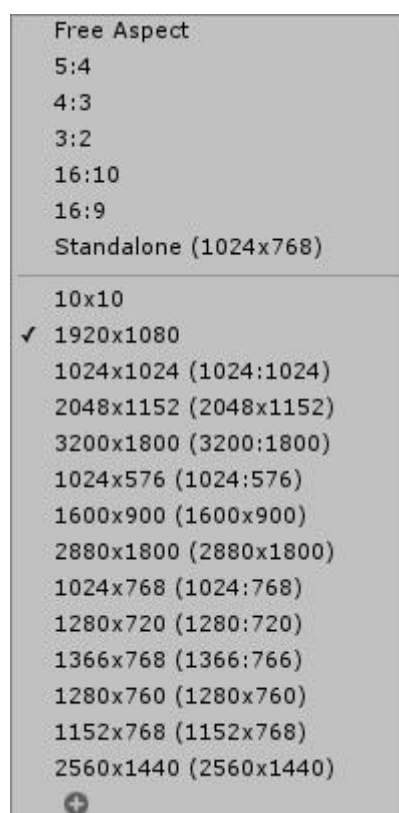
- 1) Background – задний фон (самый задний который);
- 2) TopMenu – панель кнопок в верхней части экрана;
- 3) PageGear – тут и область рюкзака, иконок игрока, меню другого игрока и тд;
- 4) PageShop – меню магазина;
- 5) Hotbar – панель быстрого доступа к предметам;
- 6) Active – все активные кнопки, изображения и тд, которые будут видны в режиме моддинга;
- 7) Root... – также, для спавна в них необходимых компонентов.

КАК ДОБАВИТЬ ЭТО НА НОВУЮ СЦЕНУ?

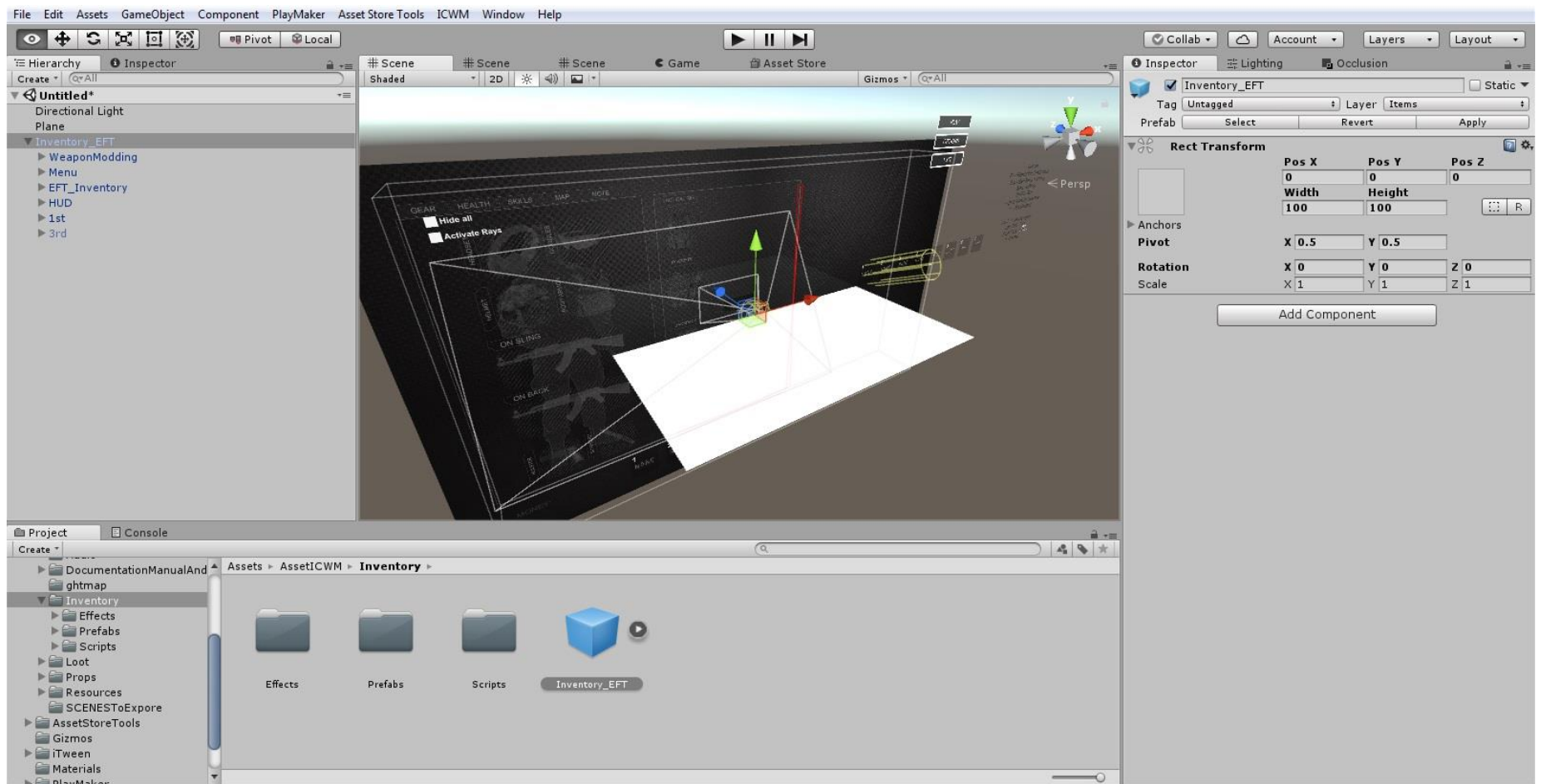
- 1) Создадим новую сцену или откроем готовую сцену.



- 2) Выставить соответствующее разрешение экрана. Убрать Free Aspect или Aspect Ration и поставить Fixed Resolution, в противном случае некоторые элементы инвентаря будут не помещаться на экране. На втором изображении я привел список разрешений, которые я тестировал.

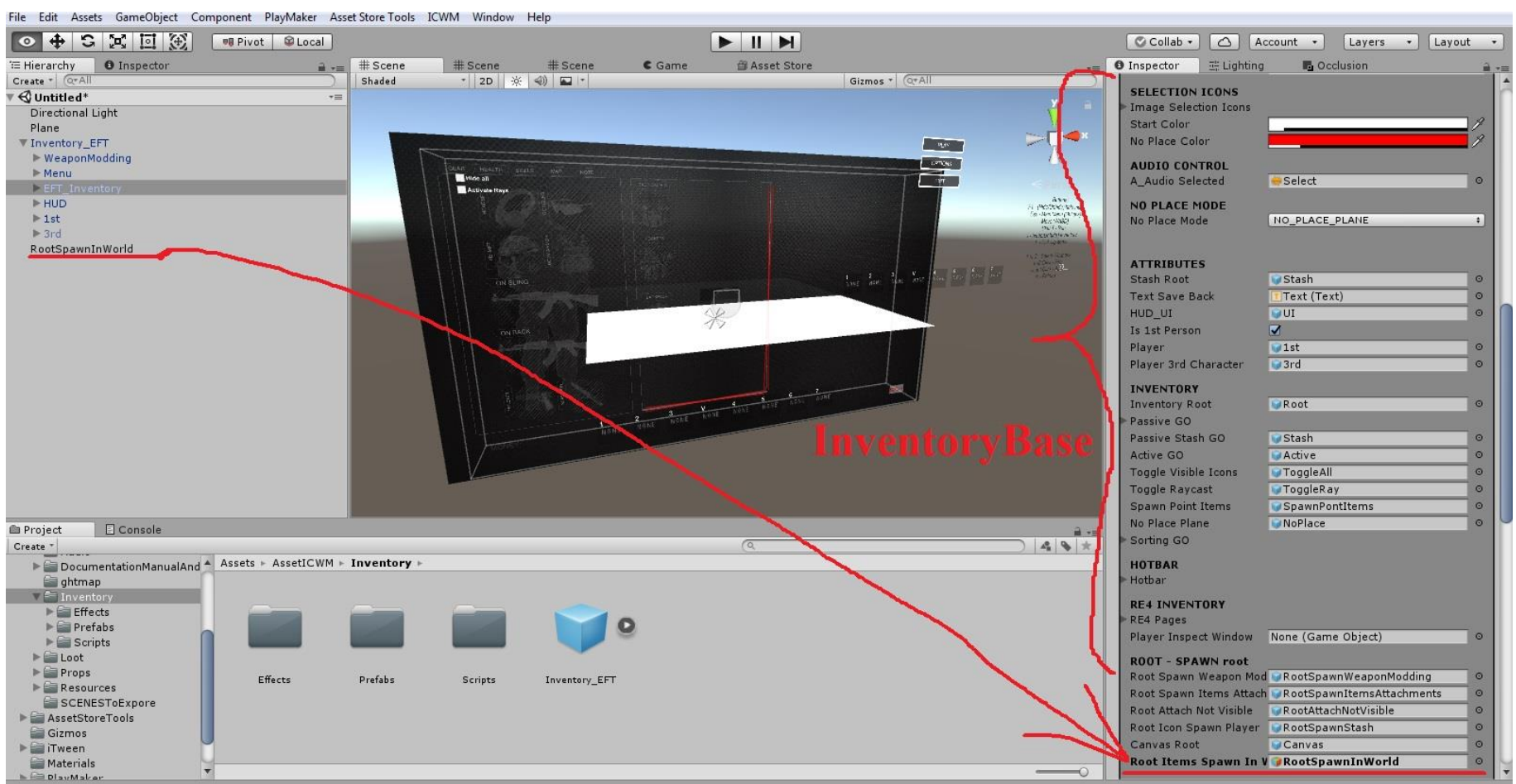


3) Добавить в сцену нужный инвентарь из папки Inventory.



4) Настройте позицию игрока (1st или 3rd), так, чтобы он не провалился сквозь землю, добавьте в сцену плоскость с коллайдером;

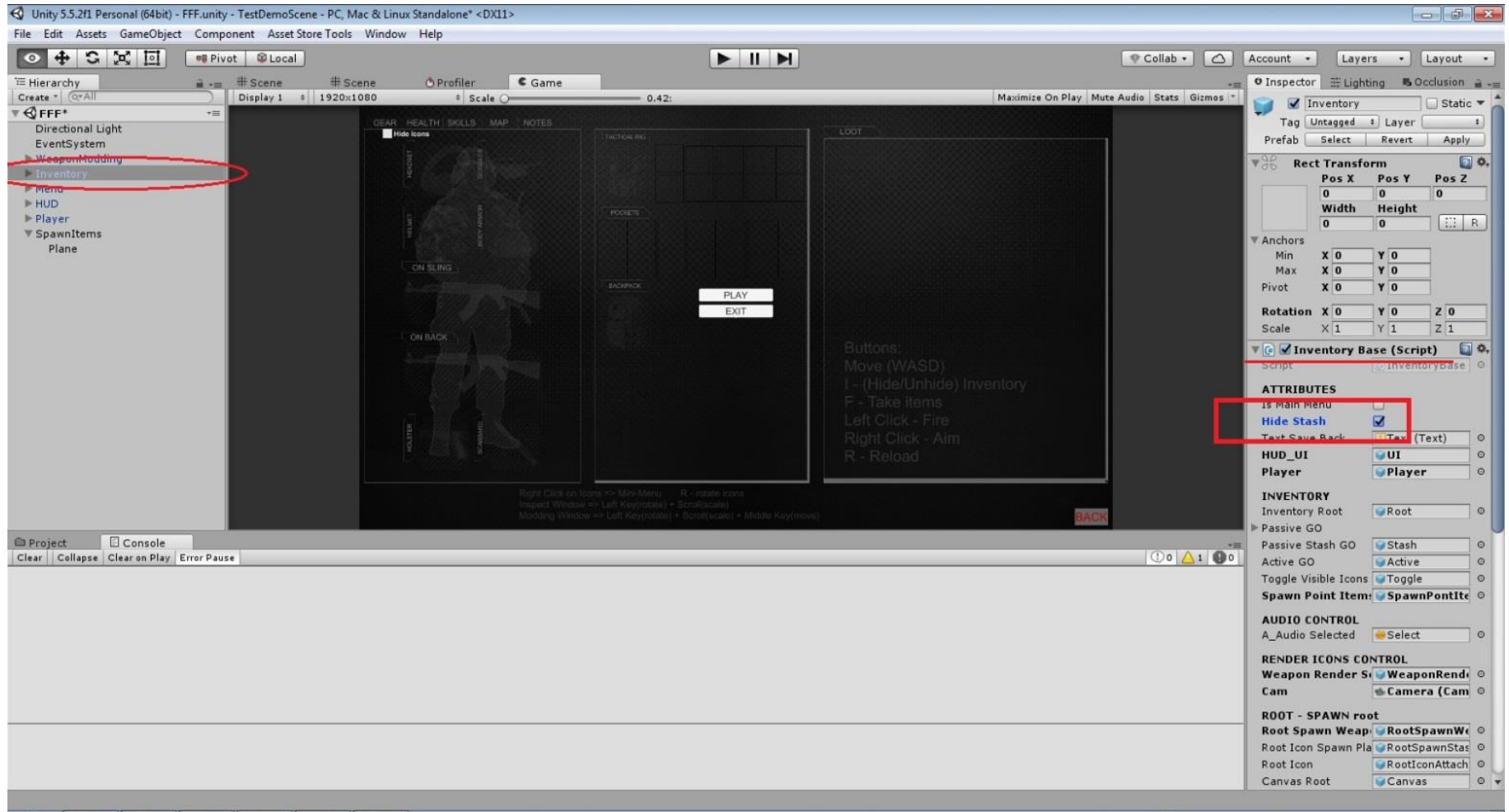
5) Создайте GameObject и поместите его куда в иерархии в любое место, затем откройте скрипт InventoryBase в инвентаре (см. изображение) и переместите ваш созданный GameObject в место указанное на изображении;



6) Play (если вы сделали все правильно то не должно быть ошибок). О да, не забудьте добавить в сцену EventSystem (добавляется таким же образом, как и GameObject).

КАК АКТИВИРОВАТЬ STASH?

Очень полезен для тестирования. Поставить галочку!

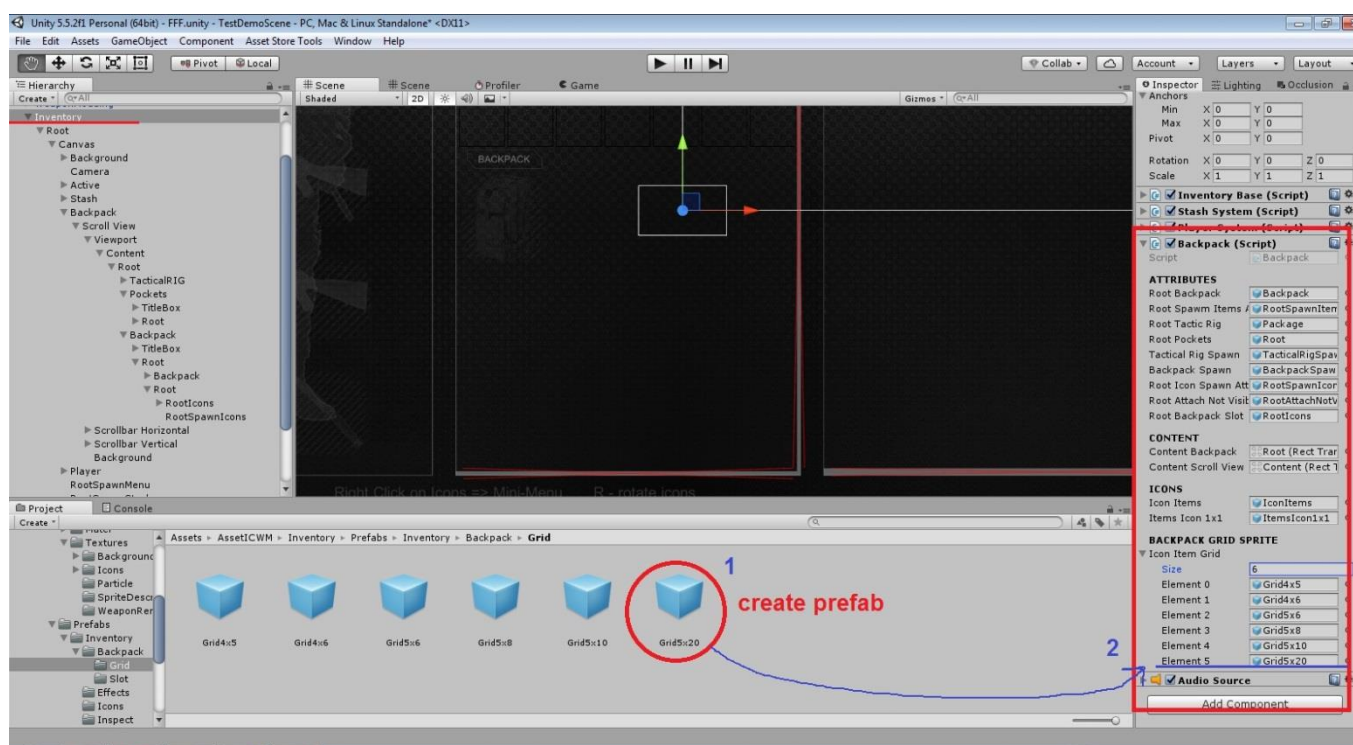
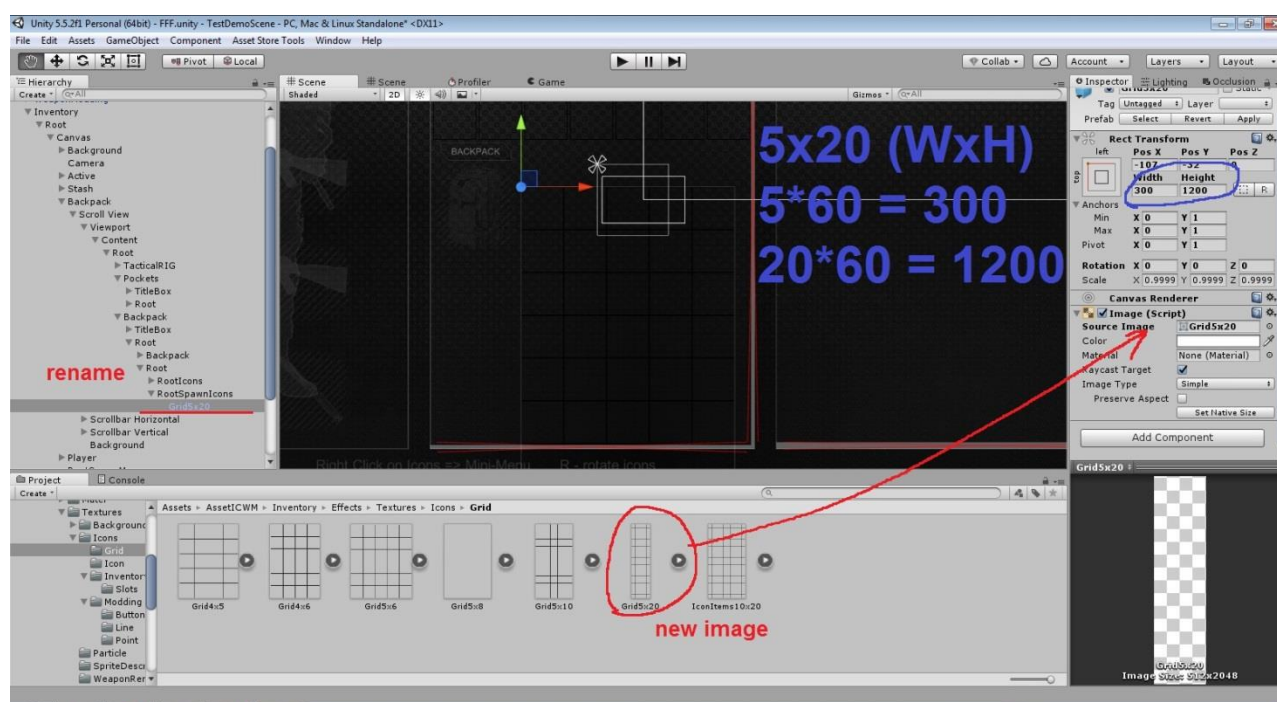
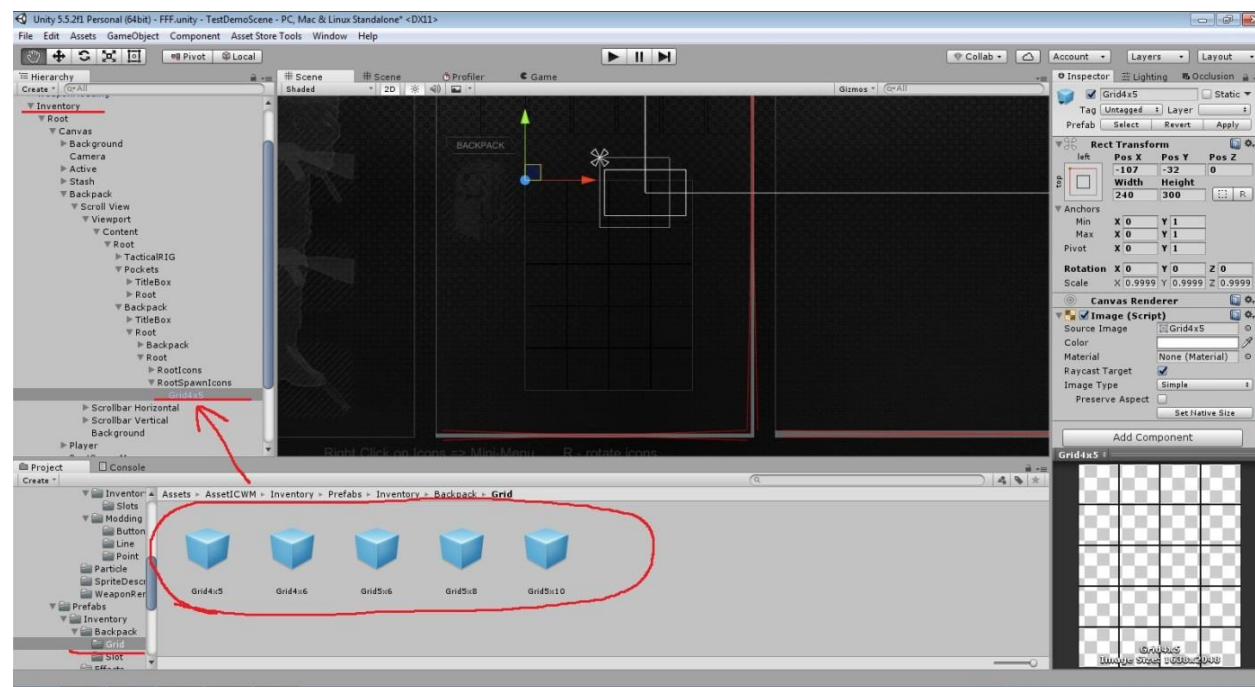


КАК ДОБАВИТЬ НОВЫЙ РАЗМЕР ДЛЯ РЮКЗАКА?

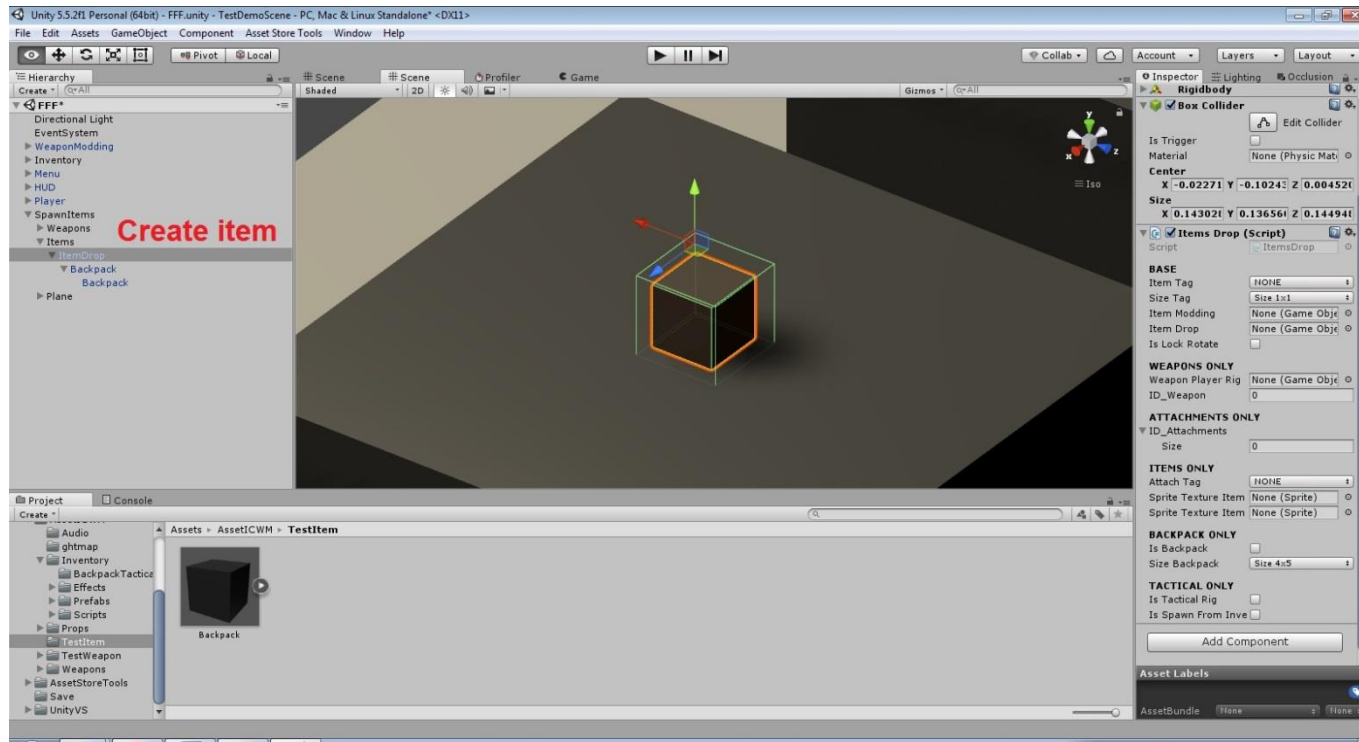
Первое, что нужно сделать, это создать сетку необходимого размера, создадим размер 5x20.

Вы можете воспользоваться примеров common\Photoshop\IconItemsV35x6 и изменить его. Я сохраняю в формате .png. В Unity необходимо преобразовать его в спрайт.

После нам нужно создать префам сетки, для этого воспользуемся имеющимся префабом и немного изменим его. Не забудьте поменять размер, формула на скрине.



Как создать вещь я уже показывал в видео.



Теперь будем немного редактировать скрипт.

Запустите скрипт ClassList.

В нем мы добавим нужный нам размер. Я пометил места, где нужно вставить нужный размер.

```

public enum SizeBackpack
{
    Size4x5,
    Size4x6,
    Size5x6,
    Size5x8,
    Size5x10,
    Size5x20,
    NONE,
}

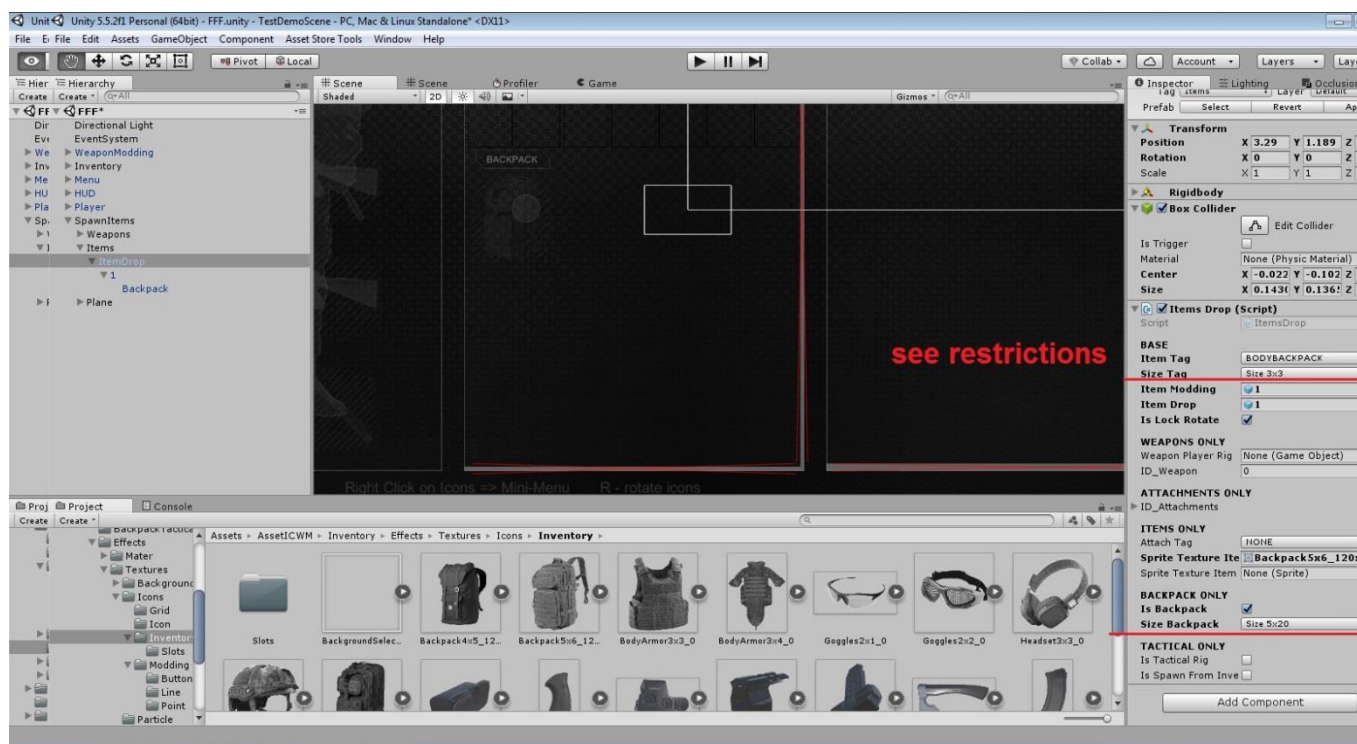
public Vector2 GetSizeBackpack()
{
    if (TagSizeBackpack == ClassList.SizeBackpack.Size4x5)
    {
        return new Vector2(4, 5);
    }
    else if (TagSizeBackpack == ClassList.SizeBackpack.Size4x6)
    {
        return new Vector2(4, 6);
    }
    else if (TagSizeBackpack == ClassList.SizeBackpack.Size5x6)
    {
        return new Vector2(5, 6);
    }
    else if (TagSizeBackpack == ClassList.SizeBackpack.Size5x8)
    {
        return new Vector2(5, 8);
    }
    else if (TagSizeBackpack == ClassList.SizeBackpack.Size5x10)
    {
        return new Vector2(5, 10);
    }
    else if (TagSizeBackpack == ClassList.SizeBackpack.Size5x20)
    {
        return new Vector2(5, 20);
    }
    else
    {
        return new Vector2(0, 0);
    }
}

public int GetIndexSizeBackpack()
{
    if (TagSizeBackpack == SizeBackpack.Size4x5)
        return 0;
    else if (TagSizeBackpack == SizeBackpack.Size4x6)
        return 1;
    else if (TagSizeBackpack == SizeBackpack.Size5x6)
        return 2;
    else if (TagSizeBackpack == SizeBackpack.Size5x8)
        return 3;
    else if (TagSizeBackpack == SizeBackpack.Size5x10)
        return 4;
    else if (TagSizeBackpack == SizeBackpack.Size5x20)
        return 5;
    else
        return 0;
}

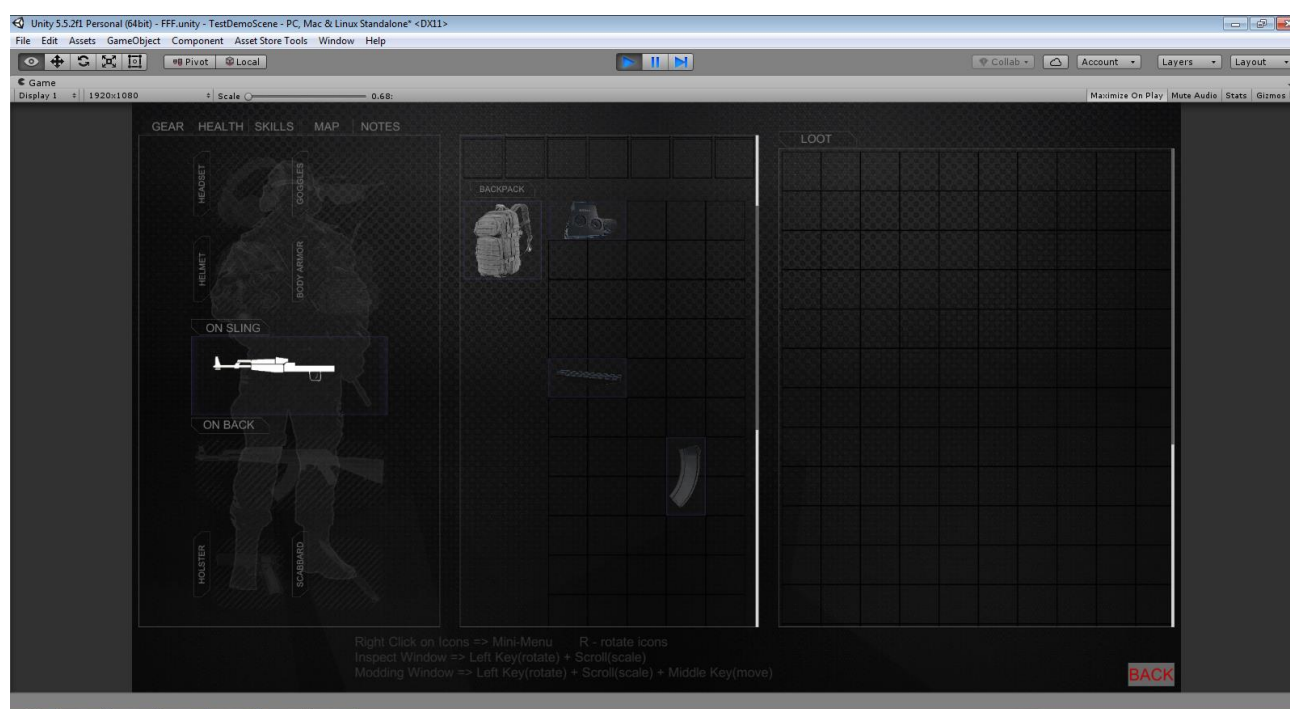
public int GetMasSizeBackpack()
{
    if (TagSizeBackpack == SizeBackpack.Size4x5)
        return 20; // 4 * 5
    else if (TagSizeBackpack == SizeBackpack.Size4x6)
        return 24;
    else if (TagSizeBackpack == SizeBackpack.Size5x6)
        return 30;
    else if (TagSizeBackpack == SizeBackpack.Size5x8)
        return 40;
    else if (TagSizeBackpack == SizeBackpack.Size5x10)
        return 50;
    else if (TagSizeBackpack == SizeBackpack.Size5x20)
        return 100;
    else
        return 0;
}

```

После этого нужно настроить нашу вещь.



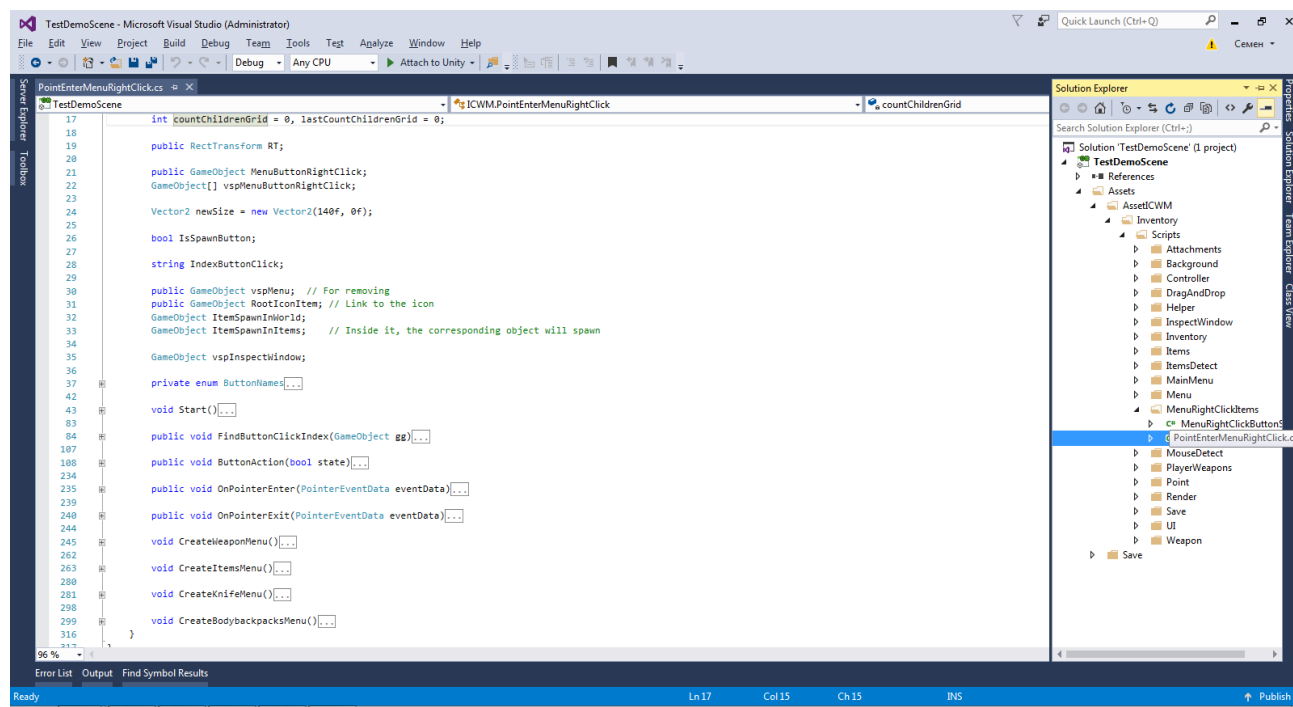
Готова!



КАК ДОБАВИТЬ НОВУЮ КНОПКУ В МИНИ-МЕНЮ?

Теперь нужно определиться, а к какому классу эта кнопка будет относиться. У всех классов разное кол-во кнопок в меню. На данный момент у оружия 3 у обычных вещей 2 и тд. Например, добавим новую кнопку к рюкзаку, броне, шлему и тп, т.к. они все относятся к игроку я сделал их как один класс.

Откроем нужный нам метод `PointEnterMenuRightClick`.



```
private enum ButtonNames
{
    DROP,
    MODDING,
    PUTON
}

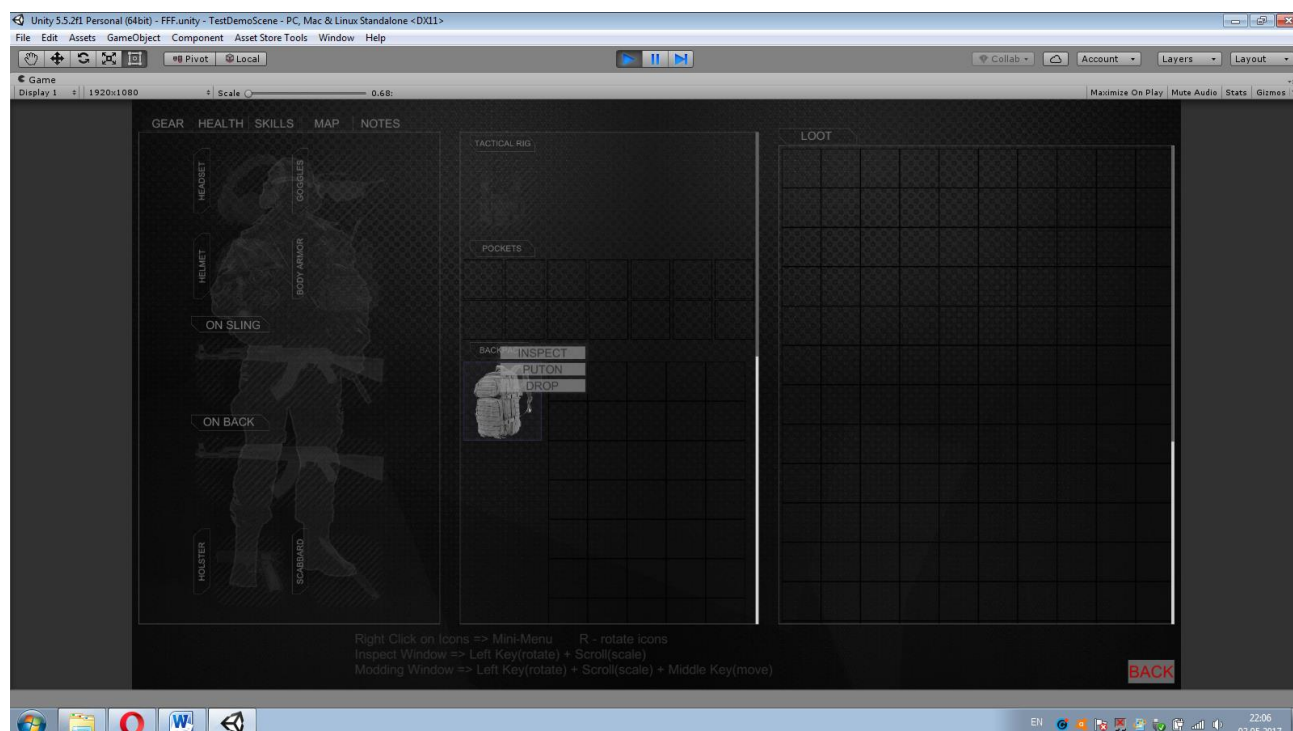
void CreateBodybackpacksMenu()
{
    ButtonNames buttonName = ButtonNames.PUTON;

    vspMenuButtonRightClick = new GameObject[2];
    for (int i = 0; i < vspMenuButtonRightClick.Length; i++)
    {
        vspMenuButtonRightClick[i] = (GameObject)Instantiate(MenuButtonRightClick);
        vspMenuButtonRightClick[i].GetComponent<Transform>().SetParent(transform);
        vspMenuButtonRightClick[i].GetComponent<Transform>().localPosition = Vector3.zero;
        vspMenuButtonRightClick[i].GetComponent<Transform>().localScale = Vector3.one;

        vspMenuButtonRightClick[i].name = buttonName.ToString();
        vspMenuButtonRightClick[i].GetComponentInChildren<Text>().text = buttonName.ToString();
        if (i == 0)
            buttonName -= 2;
        else
            buttonName--;
    }
}

public void ButtonAction(bool state)
{
    if (IndexButtonClick == "PUTON")
    {
        // Action
    }

    IsClick = true;
    IndexButtonClick = null;
}
```



КАК ДОБАВИТЬ НОВЫЙ ТЭГ ДЛЯ АТТАЧЕЙ?

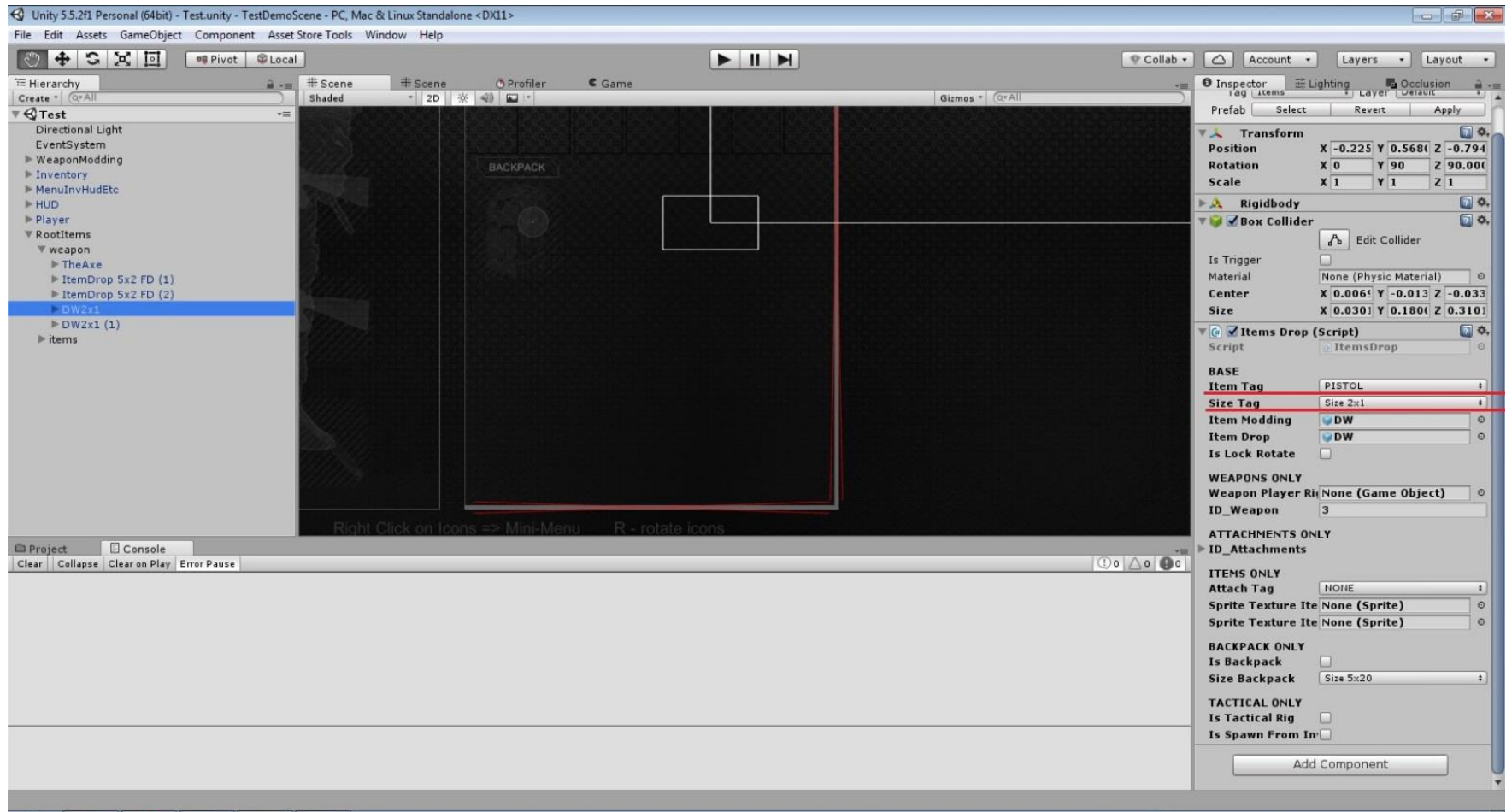
Тут ничего сложного нет. Делается все в одно действие.

Открываем скрипт *ClassList* и добавляет новый тег, вот и все.

```
public enum TagAttachments
{
    NONE, // The weapon does not have a mount
    ATTACH_BARREL_TOP, // Silencer, flash hider etc
    ATTACH_BOTTOM, //
    ATTACH_TOP, //
    ATTACH_TOP_PICATINNY, //
    ATTACH_MAGAZINE, //
    ATTACH_BOTTOM_PICATINNY, // Tactic grip etc
    ATTACH_BOTTOM_PICATINNY_FLASHLIGHT, //
    ATTACH_TOP_RECIVER,
    ATTACH_BOTTOM_PISTOLGRIP,
    ATTACH_TOP_HANDGUARD,
    ATTACH_BOTTOM_STOCK,
    ATTACH_BARREL,
    ATTACH_SLIDER,
    ATTACH_NEWTAG,
}
```

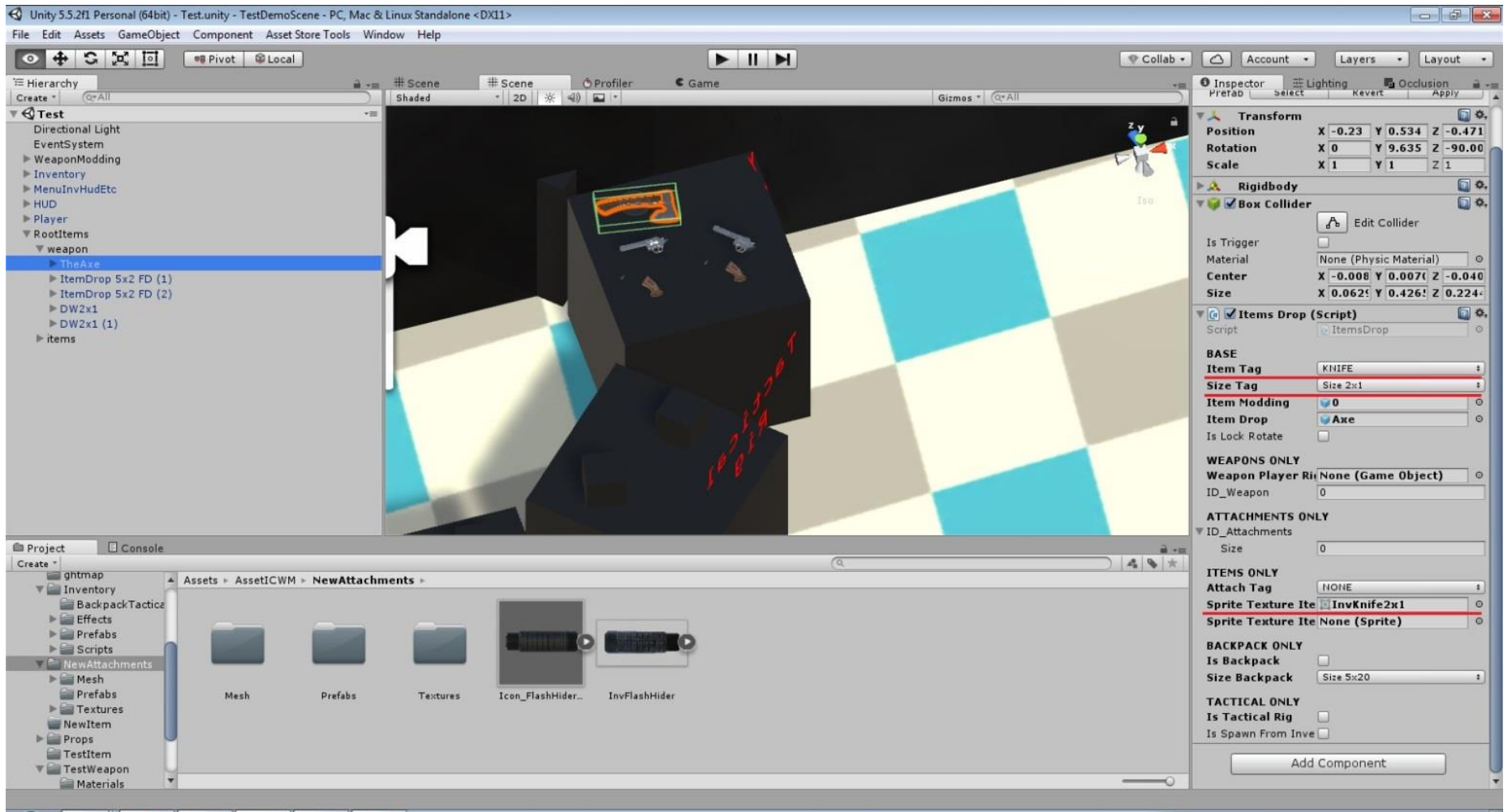
КАК ДОБАВИТЬ НОВЫЙ ПИСТОЛЕТ?

Добавление нового пистолета практически ничем не отличается от добавления нового оружия. Посмотрите видео «Как добавить новое оружие». Отличие заключается только в том, что нужно будет настраивать скрипт ItemDrop иначе. Пример вы можете посмотреть в демо-сцене. Не забываем про ограничения для иконки 1x2 и 2x1. Я добавлю больше размеров в следующих обновлениях.



КАК ДОБАВИТЬ НОВОЕ КОЛЮЩЕЕ ОРУЖИЕ?

Добавление колющего оружия ничем не отличается от добавления аттачей, вам всего лишь нужно добавить нужные параметры в скрипте *ItemDrop*. Пример вы можете посмотреть в демо-сцене. Не забываем про ограничения для иконки 1x2 и 2x1. Я добавлю больше размеров в следующих обновлениях.

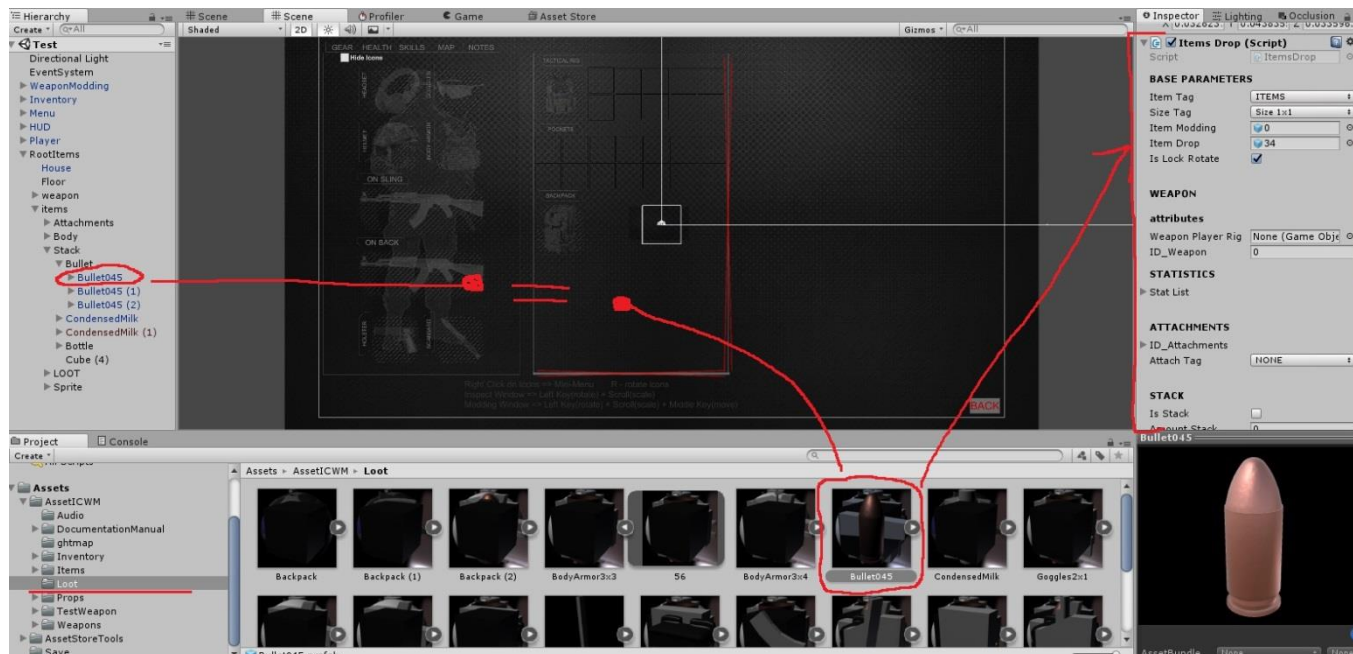


КАК ДОБАВИТЬ НОВУЮ РАЗГРУЗКА, БРОНЮ, РЮКЗАК И ТД?

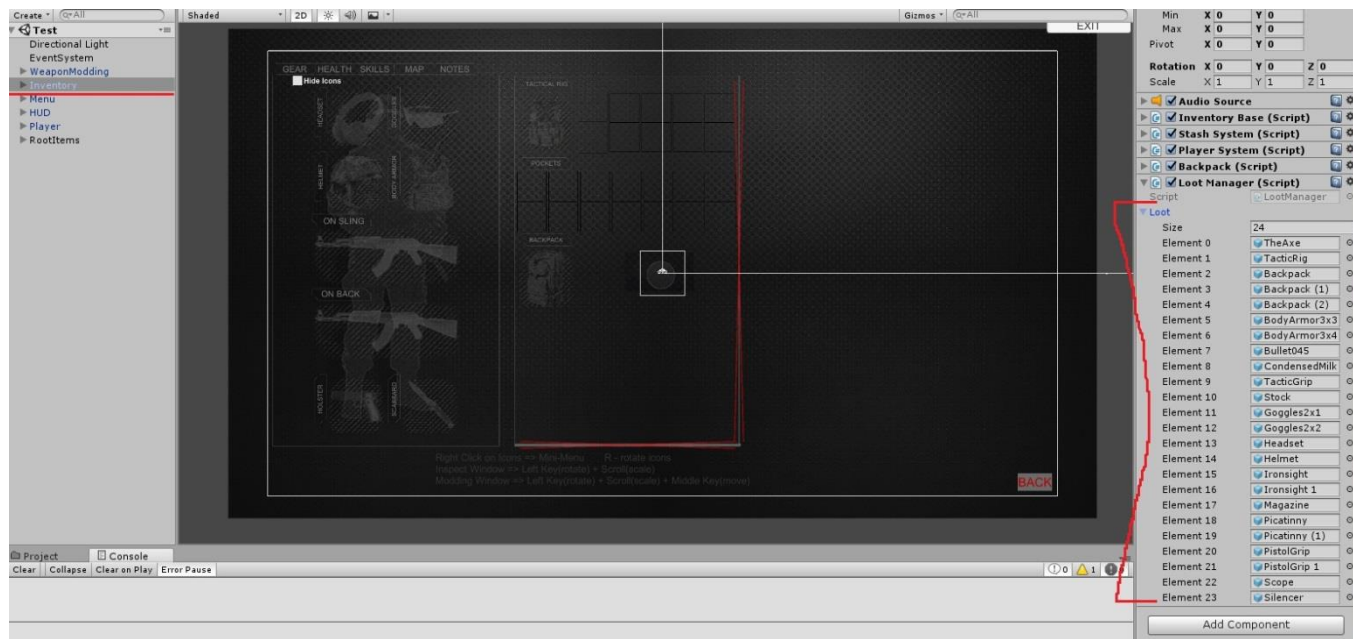
Добавление этих вещей ничем не отличается от добавления вещи в целом. Посмотрите видео урок «Как добавить новую вещь». Отличия будут заключаться только в том, что в скрипте ItemDrop у всех будут свои параметры. Для большей информации изучите демо-сцену. Не забываем про требования, которые предъявляются к иконкам.

КАК ДОБАВИТЬ ВЕЩЬ В СПИСОК ЛУТА

Предмет должен быть настроен, прежде чем помещать его в систему добычи.

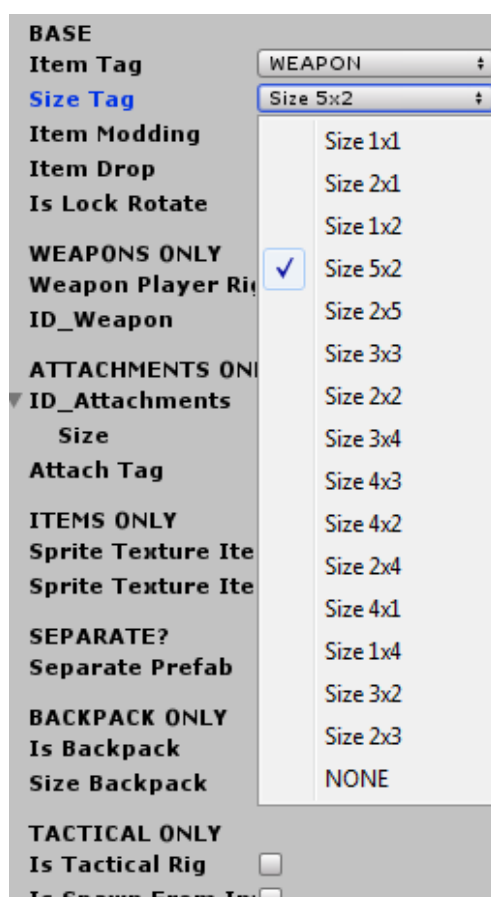


Все вещи, которые случайно появляются в ящиках, хранятся в массиве Loot. Вам нужно всего лишь добавить предмет в список.



ПРИЛОЖЕНИЕ

Размеры иконок:



Иконки, которые можно вращать:

- 1) 2x1
- 2) 1x2
- 3) 5x2
- 4) 2x5
- 5) 4x1 (New Update 2.0)
- 6) 1x4 (New Update 2.0)
- 7) 4x2 (New Update 2.0)
- 8) 2x4 (New Update 2.0)
- 9) 3x2 (New Update 2.0)
- 10) 2x3 (New Update 2.0)

Размер иконки под пистолет:

- 1) 2x1
- 2) 1x2

Размер иконки под оружие:

- 1) 5x2
- 2) 2x5
- 3) 4x1 (New Update 2.0)
- 4) 1x4 (New Update 2.0)
- 5) 4x2 (New Update 2.0)
- 6) 2x4 (New Update 2.0)
- 7) 3x2 (New Update 2.0)
- 8) 2x3 (New Update 2.0)

Размер иконки под броню:

- 1) 3x3
- 2) 3x4

Размер иконки под разгрузку:

- 1) 3x3

Размер иконки под очки или гарнитуру:

- 1) 1x2
- 2) 2x1
- 3) 2x2

Размер иконки под шлем:

- 1) 2x2

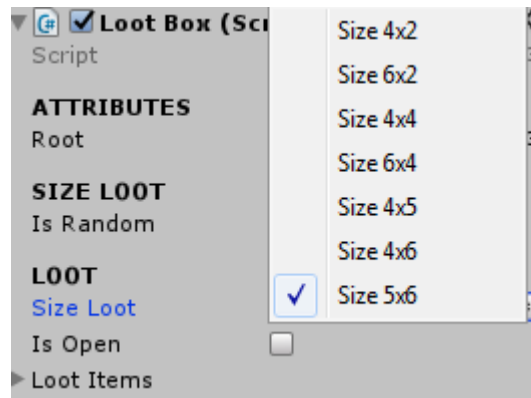
Размер иконки под рюкзак:

- 1) 3x3
- 2) 3x4
- 3) 2x2 (New Update 2.0)
- 4) 4x3 (New Update 2.0)

Размер иконки под нож:

- 1) 1x2
- 2) 2x1

Размеры ящика с добычей (New Update 2.0)



Размер рюкзака, который **нельзя ставить в инвентаре EFT: (New Update 6.0)**

- 1) 15x8
- 2) 11x7
- 3) 10x6