

MFPC - Mobile First Person Controller

MFPC is a first person controller template for mobile and other touchscreen devices. Setting up MFPC is as simple as possible, all you need to do is drag and drop MFPCController prefab to the scene and set its position to X:0, Y:0, Z:0.

MFPC got almost all functionality, you may need to create a mobile first person game such as walking, running, crouching, jumping and fast turn with desired angle.

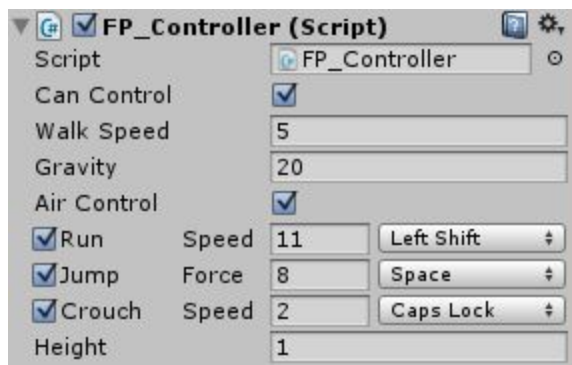
MFPC can work on standalone!

MFPCController prefab contains Player object, Canvas with all needed UI elements and EventSystem object.

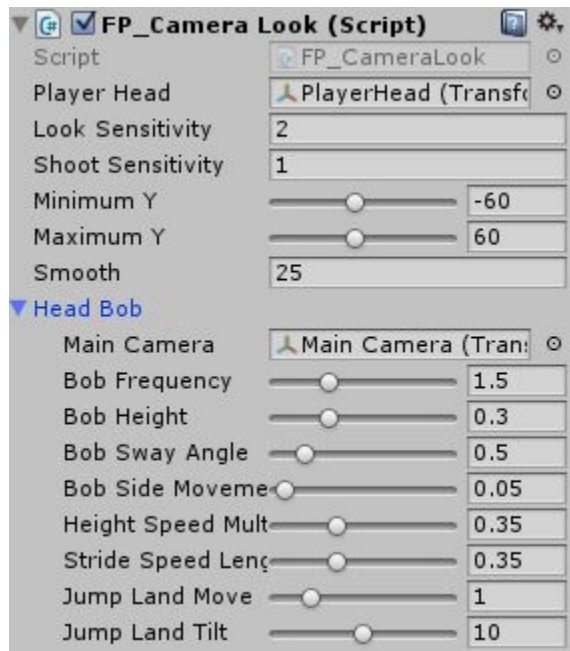


NOTE: If you need reposition player, don't move MFPCController, move Player instead.

PLAYER:



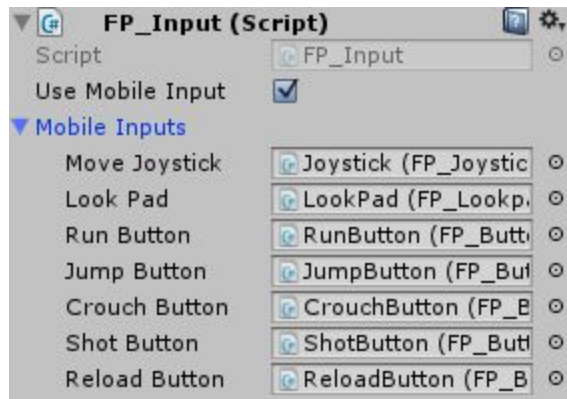
- ❖ **Can Control** - is player controllable or not;
- ❖ **Walk Speed** - player walk speed;
- ❖ **Gravity** - gravity scale;
- ❖ **Air Control** - can move in air, or not;
- ❖ **Run** - can run or not, **Speed** - Run speed, Keyboard key for run;
- ❖ **Jump** - can jump or not, **Force** - jump force, Keyboard button for jump;
- ❖ **Crouch** - can crouch or not, **Speed** - crouch speed, Keyboard button for crouch;
- ❖ **Height** - crouch height.



- ❖ **Player Head** - player's head, empty gameObject which is holding player cameras;
- ❖ **Look Sensitivity** - camera look sensitivity;
- ❖ **Shoot Sensitivity** - camera look sensitivity if shoot button is pressed;
- ❖ **Minimum Y / Maximum Y** - vertical look limits;
- ❖ **Smooth** - camera look smoothing. Less value - more smoothing;
- ❖ **Head Bob** - head bobbing settings.



- ❖ **Jump sound** - jump sound effect;
- ❖ **Land sound** - landing sound effect;
- ❖ **Footsteps** - Array of step options;
 - **Surface Tag** - tag of surface;
 - **Step Sounds** - array of step sounds for this surface.



Use Mobile Input - disable this option to use mouse and keyboard controls;
 Mobile Inputs - touch UI input elements. All the elements can be found under Canvas object.
 If you don't need any of those, just disable or delete it through hierarchy window.



Fast turn script.

Turn speed - rotation speed;

Turn angle - rotation angle;

Left Turn / Right Turn - UI buttons.

If you don't need this mechanic you can simply remove this script and delete buttons.

Hierarchy



Player head is an empty gameObject which is holding cameras;

ItemsCamera is a camera for player objects such as weapons etc. MFPC doesn't contains weapon system, but all the things are already prepared. All the player objects should have layer '**Items**' to ignore wall clipping. If Items layer is missing, you should add it manually.

WeaponExample is an empty object with example script, showing you how to integrate any weapon script into the system.

There is a few steps:

- Declare and assign FP_Input variable;
- Replace your inputs with FP_Input methods. (Lines 9, 26, 30).

INPUT SYSTEM:

FP_Joystick for moving, FP_LookPad for looking and FP_Button for almost all you ever will need. Input system can be used separately with any other project.

Let's take a closer look:



My Canvas is a canvas object;
Default Alpha is idle alpha color;
Active Alpha is pressed alpha color;
Interactable - is button interactable or not;
Dynamic - is button dynamic or not (moving with touch).

FP_Button have next useful functions:

- MoveInput() - returns drag Vector.
- IsPressed() - returns true if button is pressed.
- OnPress() - Fires once when button was pressed.
- OnRelease() - Fires once when button was released.
- Toggle() - Toggle behaviour for a button.
-

FP_Button.Toggle() example on pause script:

```
//Declare FP_Button;
public FP_Button pauseButton;

private bool isPaused;

void Update()
{
    //Assign pause button toggle to pause flag;
    isPaused = pauseButton.Toggle();

    //Pause functionality;
    if (isPaused)
        Time.timeScale = 0;
    else
        Time.timeScale = 1;
}
```

That's all! Thanks for purchase.

If you have any questions or issues please contact me via nbg_yalta@hotmail.com