Modular First Person Controller

By: Jess Case

The purpose of this document is to provide information how the “Modular First Person Controller” works along with how to use and modify its components. The “Modular First Person Controller” is an open source first person controller for the Unity Engine. The included Prefab and “FIrstPersonController” script are free to use in both commercial and non-commercial applications. Giving attribution is not required, but is greatly appreciated.

NOTICE:
The demo scene includes artwork from the “Gridbox Prototype Materials” asset, and thus are not to be used in commercial applications.

If you have a problem that you cannot solve or have added a feature that you would like to see added to this package email jesscasedev@gmail.com.

Included In This Package

ModularFirstPersonController - Folder, contains package

- FirstPersonController - Folder, contains free to use work
    - CircleReticle - Image, used for default reticle
    - FirstPersonController - Script, contains all code for package
    - FirstPersonController - Prefab, controller to be placed in scene
- DemoScene - Folder, contains everything used in the demo scene
    - Gridbox Prototype Materials - Folder, package downloaded from Unity Asset Store and used in creation of the demo scene
    - Demo - Scene

Creation

To place the controller in your Unity scene, simply drag the provided prefab into your scene or hierarchy. The prefab must have all the given components to work:

- A collider (capsule collider provided)
- Rigidbody
- First Person Controller Script

The player object's hierarchy must be in this order:
FirstPersonController > Joint > PlayerCamera > CrosshairAndStamina > Reticle, SprintBar > StaminaBG, Stamina

Default movement keys are WASD.

Editor

The "Modular First Person Controller" contains a custom editor that holds every available variable for users to change. These variables are sectioned into the categories "Camera Setup", "Movement Setup", and "Head Bob Setup", and some sections may contain subsections. All features, such as the ability to sprint and even camera movement, can be enabled or disabled in the editor.

List of Properties

Camera Setup

Property Name	Type	Function
Camera	Camera Object	Camera attached to the controller.
Field of View	float	The camera's view angle. Changes the player camera directly.
Enable Camera Rotation	bool	Determines if the camera is allowed to move.
Invert Camera Rotation	bool	Inverts the up and down movement of the camera.
Look Sensitivity	float	Determines how sensitive the mouse movement is.
Max Look Angle	float	Determines the max and min angle the player camera is able to look.
Lock and Hide Cursor	bool	Turns off the cursor

		visibility and locks it to the middle of the screen.
Auto Crosshair	bool	Determines if the basic crosshair will be turned on, and sets is to the center of the screen.
Crosshair Image	Sprite	Sprite to use as the crosshair.
Crosshair Color	Color	Determines the color of the crosshair.
Zoom		
Enable Zoom	bool	Determines if the player is able to zoom in while playing.
Hold to Zoom	bool	Requires the player to hold the zoom key instead if pressing to zoom and unzoom.
Zoom Key	KeyCode	Determines what key is used to zoom.
Zoom FOV	float	Determines the field of view the camera zooms to.
Step Time	float	Determines how fast the FOV transitions while zooming in.

Movement Setup

Property Name	Type	Function
Enable Player Movement	bool	Determines if the player is allowed to move.
Walk Speed	float	Determines how fast the player will move while walking.

Sprint		
Enable Sprint	bool	Determines if the player is allowed to sprint.
Unlimited Sprint	bool	Determines if “Sprint Duration” is enabled. Turning this on will allow for unlimited sprint.
Sprint Key	KeyCode	Determines what key is used to sprint.
Sprint Speed	float	Determines how fast the player will move while sprinting.
Sprint Duration	float	Determines how long the player can sprint while unlimited sprint is disabled.
Sprint Cooldown	float	Determines how long the recovery time is when the player runs out of sprint.
Sprint FOV	float	Determines the field of view the camera changes to while sprinting.
Step Time	float	Determines how fast the FOV transitions while sprinting.
Use Sprint Bar	bool	Determines if the default sprint bar will appear on screen.
Hide Full Bar	bool	Hides the sprint bar when sprint duration is full, and fades the bar in when sprinting. Disabling this will leave the bar on screen at all times when the sprint bar is enabled.

Bar BG	Image Object	Object to be used as sprint bar background.
Bar	Image Object	Object to be used as sprint bar foreground.
Bar Width	float	Determines the width of the sprint bar.
Bar Height	float	Determines the height of the sprint bar.
Jump		
Enable Jump	bool	Determines if the player is allowed to jump.
Jump Key	KeyCode	Determines what key is used to jump.
Jump Power	float	Determines how high the player will jump.
Crouch		
Enable Crouch	bool	Determines if the player is allowed to crouch.
Hold to Crouch	bool	Requires the player to hold the crouch key instead if pressing to crouch and uncrouch.
Crouch Key	KeyCode	Determines what key is used to crouch.
Crouch Height	float	Determines the y scale of the player object when crouched.
Speed Reduction	float	Determines the percent “Walk Speed” is reduced by. 1 being no reduction, and .5 being half.

Head Bob Setup

Property Name	Type	Function
Enable Head Bob	bool	Determines if the camera will bob while the player is walking.
Camera Joint	Transform Object	Joint object position is moved while head bob is active.
Speed	float	Determines how often a bob rotation is completed.
Bob Amount	Vector3	Determines the amount the joint moves in both directions on every axes.

First Person Controller Script

The “First Person Controller” Script contains a number of private variables used for internal representations. These should not be changed unless changing systems in the script directly. It also contains private methods used in conjunction with Update() and FixedUpdated() for organization proposes.