

SIDE MENU

Vatio.eu

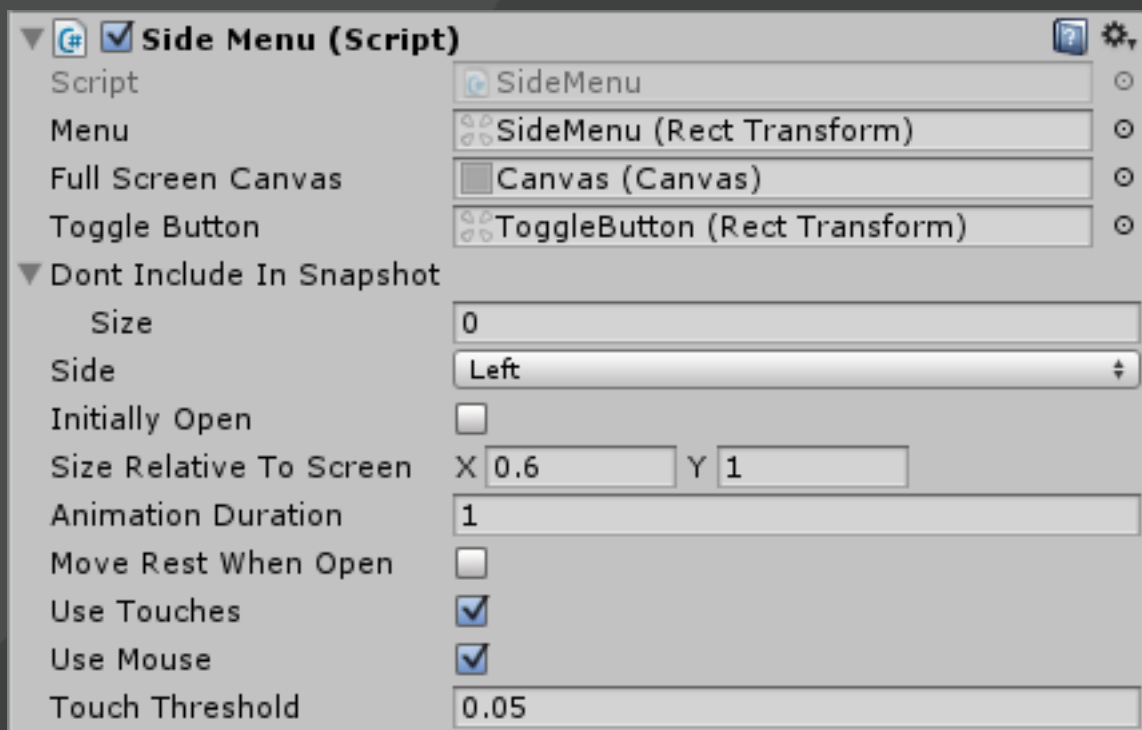
This is the instruction for the **Side Menu** package. It's an easy to use, ready-made Facebook-like sliding side menu mostly designed for mobile apps but working 100% in desktop applications as well. It comes with built-in top-notch gesture control as well as sliding animation and is highly customizable to make it fit in a wide variety of scenarios.

Usage

To use the Side Menu first make a UI GameObject preferably directly under the UI canvas in hierarchy. This will be the topmost object in the side menu view hierarchy. You may add a visual component to this GameObject, however for the sake of the project organization it is recommended to add any visual components in children GameObjects.

When the main Side Menu GameObject is created, attach to it the Side Menu component. The Side Menu could be added to any GameObject in the scene, but it's best to make as little cross-references as possible.

The Side Menu editor view looks like that:

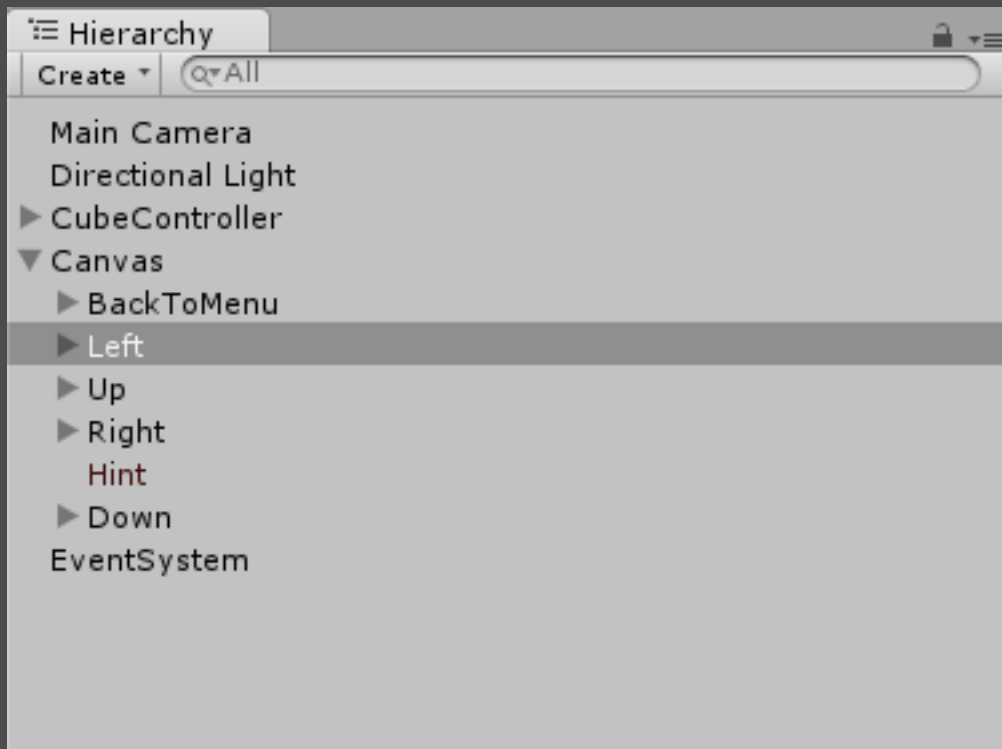


Parameters

These are the parameters available in the editor window:

- **Menu** - [required] the RectTransform component of the main UI GameObject of the side menu. The anchors and other parameters of the RectTransform will be modified by the Side Menu script, but for the sake of the project organization it's best to adjust them appropriately.
- **Full Screen Canvas** - [required only if the "Move Rest When Open" is enabled] the main UI Canvas.
- **Toggle Button** - [optional] the toggle button of the side menu. It will be parented to the side menu RectTransform, so it moves parallel to it and is always visible. It will also be excluded from snapshot to avoid displaying it twice.
- **Dont Include In Snapshot** - [optional] list of GameObjects that should be excluded from the snapshot.
- **Side** - [required] the side of the screen the menu should be displayed on. The menu will be displayed on that side of the screen and centered in the other direction.
- **Initially Open** - [required] if enabled, the menu will be open at the beginning of the scene, otherwise it will be closed.
- **Size Relative To Screen** - [required] the size of the menu relative to the size of its parent object. Assuming the parent object is the main canvas this is the menu size relative to the screen. For example, if set to (0,5:1.0) the menu will be half of the screen width and full screen height.
- **Animation Duration** - [required] the duration in seconds of the full opening or closing animation. If only a part of animation is played (the user drags the menu part of the way and then the animation is played) the time will be reduced proportionally.
- **Move Rest When Open** - [required] if enabled, when menu is being opened, it will take screenshot of the rest of the screen (except GameObjects specified in the "Dont Include In Snapshot" parameter) and display it sliding out of the screen. Requires "Full Screen Canvas" to be set.
- **Use Touches** - [required] should the menu recognize and respond to touch gestures (slide in and out)
- **Use Mouse** - [required] should the menu recognize and respond to mouse gestures (slide in and out)
- **Touch Threshold** - how far the user has to drag the menu for it to start opening or closing. This is to avoid menu animation on every user input.

Example hierarchy



In this hierarchy there are 4 side menus: “Left”, “Up”, “Right” and “Down”. These objects are all just GameObjects with no visual component attached directly, that take the specified part of the screen. Each of them has child GameObject with background image that is the same size and position and children GameObjects with appropriate UI elements. This is an example of a well designed side menu hierarchy, but You are free to adjust hierarchy to Your needs.

Current limitations

- Only one menu can be open at any given time
- Only a single menu can be attached to each side of the screen for gesture control to work (so 4 menus max)

Modifications

The recommended way to make any modifications to the original script is to subclass the SideMenu class. Please look at the methods and information contained in the SideMenu.cs file for details. For changes in the animations modify the methods contained in the “STATES_PROCESSING” region, remember though to include all the needed code in the sub-class method.

Contact and bug reporting

To get additional information regarding the Side Menu package, or to report any problems with it, please send a message to the following address:

unity3d@vatio.eu

Last words

We hope You enjoy using the Side Menu package in many succesful projects.

The team of Vatio.eu