SWIPY MENU

Swipy Menu – is a Unity asset which allows you quickly create swipable menu with sliders and headers for your project!

INFO

Swipy Menu consists of two main scripts/elements:

· **SwipyMenu** – core part of the asset. Highly optimized Unity's MonoBehavior component witch will be included in your build.

· **SwipyMenuGenerator** – wrapper for SwipyMenu designed to help setup the SwipyMenu in Editor. It won't be included in build and won't made any overhead in your game.

There is also a few scripts that you may like to know about, but you won't have to use them directly, so you can skip this section:

· **SwipyMenuHeader** – component that represents a heder of the menu. It has two properties assigned automatically by SwipyMenuGenerator, but you can also place them manually if you need to.
Text – it is a UnityEngine.UI.Text component. If assign, then its text property will be displayed as a name of a respective item in SwipyMenuGeneraator's reorderable list, instead of a default name.
Button – SwipyMenu use UnityEngine.UI.Button to receive clicks on headers. If assigned, then SwipyMenuGenerator. will automatically update onClick event of a button with the change of a menus order. If you want to use your own button, you will have to call HeaderClickHandler method from the SwipyMenu component.

· **SwipyMenuScrollRect** – it inherits from UnityEngine.UI.ScrollRect and used only in SwipyMenuScrollRectEditor to restrain ScrollRect properties that won't work with SwipyMenu.

· **ScrollRectNested** – unlike standard ScrollRect it allows you to swipe also a parent SwipyMenu when swiped in opposite direction of a MenusOrientations property.

· **Utilities** – just a two helper functions called Normalize and Denormalize.

GETTING STARTED

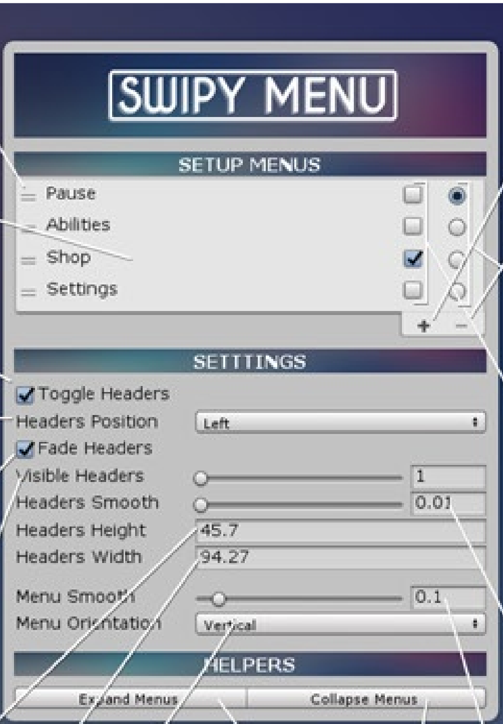
1. Create and empty UI **GameObject** (it must be located somewhere within **Canvas** component).
2. Setup its **RectTransform** as you need your menu to be positioned.
3. To the newly created GameObject attach **SwipyMenuGenerator** component.
4. Click **+ button** on the **menus list** to **add** menus.
5. **Double click** on any menu in **menus list**, and start modifing it!

That's all for getting started with **Swipy Menu**!

TIPS

1. SwipyMenuGenerator will automatically create two RectMask2D components. One for the menus. One for the headers. If you don't need them, you can just delete these components. But it's recommended to use RectMask2D for reducing draw calls and for better performance.

USER MANUAL



SWIPY MENU

SETUP MENUS

Pause	□	●
Abilities	□	○
Shop	☑	○
Settings	□	○

+ −

SETTINGS

☑ Toggle Headers	
Headers Position	Left
☑ Fade Headers	
Visible Headers	1
Headers Smooth	0.01
Headers Height	45.7
Headers Width	94.27
Menu Smooth	0.1
Menu Orientation	Vertical

HELPERS

Drag list items to **reorder menus**

Double click list item to **highlight content** that you will be editing

Enable/Disable headers

Choose **position of the headers**: Left/Top/Right/Bottom

Toggle if headers should **fade** when they out of focus

Determines **how many headers** will be **visible** at the sides of the current header

Set **headers height**

Set **headers width**. This is the width of a one header

Choose **menus orientation**: Horizontal / Vertical

Click for **adding a new menu**

Click for **deleting selected menu**

Select to set the **default menu**

Toggle to add **NestedScrollRect** (standard ScrollRect wont scroll overall menu, only itself)

Determines how **quickly** menu will reach its target position when user don't input anything.

1 – means instantly, 0 – means never. You restricted to only set this value between 0.01 and 1

Determines how **quickly** headers will reach its target position when user don't input anything.

1 – means instantly, 0 – means never. You restricted to only set this value between 0.01 and 1

Collapse menus so they take less space

Expand menus. So it would be much more convenient to edit them

SCRIPTING API

To control Swipy Menu though script you need access SwipyMenu component that is attached to GameObject named Menus (which is a child of main Swipy Menu GameObject, the one with SwipyMenuGenerator component).

SwipyMenu	
Public fields and properties	
public **float** headerSmoothness;	Determines how quickly menu will reach its target position when user don't input anything. 1 – means instantly, 0 – means never. Should be value from 0.01 to 1.
public **float** scrollRectSmoothness;	Determines how quickly headers will reach its target position when user don't input anything. 1 – means instantly, 0 – means never. Should be value from 0.01 to 1.
public **int** visibleHeaders;	Determines how many headers will be visible at the sides of the current header.

public **int** defaultMenuIndex	Set a menu index which you like to show first on load.
public **bool** HeadersEnabled	Enable/Disable headers.
public **float** HeaderWidth	Set headers width. This is width of a one header.
public **float** HeadersHeight	Set headers height.
public **HeaderPositions** HeaderPosition	Set header position: Left / Top / Right / Bottom.
public **MenusOrientations** MenusOrientation	Set Swipy Menu to scroll horizontally or vertically
public **bool** FadeHeaders	Toggle if headers should fade when they out of focus
Public methods	
public void SetCurrentMenu(int number)	Sets current menu as **number.** **number** - menu number (starts from 1)
public void HeaderClickHandler(int number)	If you are using custom button subscribe this to its click event. **number** - menu number (starts from 1)
Enums	

public **enum** HeaderPositions { Top, Bottom, Left, Right }
public **enum** MenusOrientations { Horizontal, Vertical }
	Represents position of a headers.
	Represents menus orientation.

Usage example:

```
using UnityEngine;

public class Example : MonoBehaviour
{
	public SwipyMenu swipyMenu; // set it in the Inspector

	swipyMenu.menuIndexToShowFirstOnLoad = 3;
	swipyMenu.HeaderPosition = SwipyMenu.HeaderPositions.Left;
	swipyMenu.HeadersHeight = 100f;
	swipyMenu.HeaderWidth = 40f;
	swipyMenu.MenusOrientation = SwipyMenu.MenusOrientations.Vertical;
}
```