SIMPLE SIDE-MENU

This manual, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. The content of this manual is furnished for informational use only, is subject to change without notice and should not be construed as a commitment by its authors. The author assumes no responsibility or liability for any errors or inaccuracies that may appear in this manual.

Contents Page

Getting Started:

"Simple Side-Menu" Component:

Support:

Getting Started

Overview:

"Simple Side-Menu" is a simple solution to adding sliding side menus into your Unity project which work extremely well for both mobile and desktop applications.

Features:

- ✓ Easy to use and highly customizable.
- ✓ Mouse (desktop) and touch (mobile) support.
- ✓ Ability to place the menu on the left, right, top or bottom of the screen.
- ✓ Control over when the menu is able to change state (based on the threshold drag speed and/or fraction), and how it changes state (by adjusting the transition speed as well as determining whether a handle, a button and/or the menu (itself) can be used to open/close the menu).
- ✓ Customizable overlay that fades in based on the side menu's state progress that can be used to close the menu when pressed.
- ✓ ... and much more!

Included:

- ✓ Two example projects:
  - ➢ Clash of Clans
  - ➢ Google Maps
- ✓ Ready-to-use prefabs:
  - ➢ Left Side-Menu
  - ➢ Right Side-Menu
  - ➢ Top Side-Menu
  - ➢ Bottom Side-Menu
- ✓ In-depth offline documentation

Contact:

In the event you are unable to find the information you are looking for or have found a bug, please feel free to send me an email (daniel@daniellochner.com) and I will get back to you as soon as possible.

Installation:

Once you have downloaded "Simple Side-Menu" from the Unity Asset Store, and the "Import Asset", window appears in Unity, verify that all items to import are selected and then click the "Import" button in the bottom right of the window.

Keep in mind that the "Simple Side-Menu" folder does not have to be in the root directory of your project, so feel free to move it anywhere!

Quick Start:

To add a “Simple Side-Menu” component to your Game Object, go to “Component > UI > Simple Side-Menu” or simply search for the “Simple Side-Menu” script in the “Add Component” menu in the inspector.

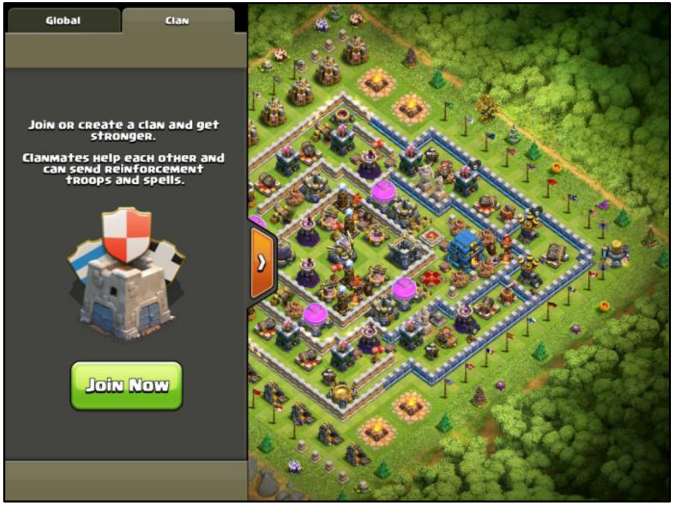
If you are unsure of how to setup your Game Object to use the “Simple Side-Menu” component, there are four basic prefabs that can help get you started (Left, Right, Top and Bottom Side-Menus)

To add any of these prefabs to your scene, simply drag it from the “/Simple Side-Menu/Prefabs” folder and place it under your Canvas, or navigate to “GameObject > UI > Simple Side-Menu“, and select it from the drop-down list.

Examples:

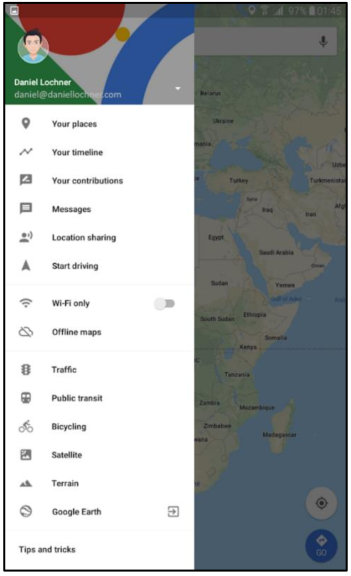
There are two example projects included that illustrate how the “Simple Side-Menu” component can be used in real scenarios:

Example 1: Clash of Clans


Global
Clan
Join or create a clan and get stronger.
Clanmates help each other and can send reinforcement troops and spells.
Join Now

This example shows how the Clash of Clans chat side menu can be recreated with the Simple Side-Menu component. It is achieved using the basic “Left Side-Menu” prefab and then disabling the overlay.

Example 2: Google Maps



This example shows how the Google Maps settings side menu can be recreated with the Simple Side-Menu component. It is achieved creating a transparent drag handle (aligned along the right edge of the Side-Menu), as well as a hamburger-menu button which invokes the Open() method of the Simple Side-Menu when pressed. An overlay is also used, and the option to enable the menu to be dragged is enabled.

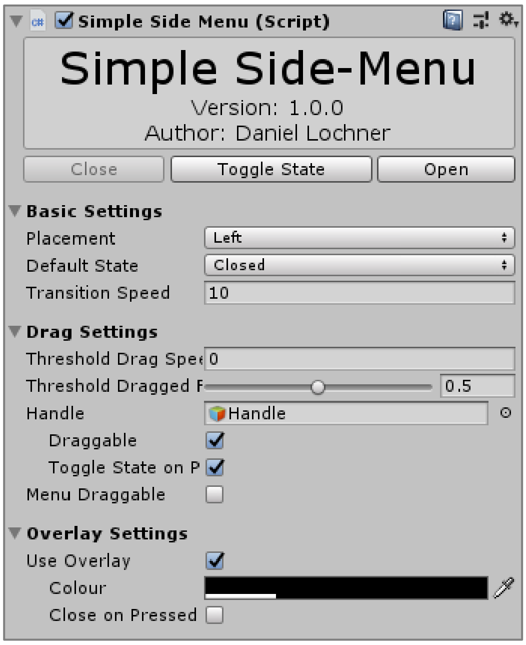
“Simple Side-Menu” component

Inspector Breakdown:

The “Simple Side-Menu” component’s inspector is broken down into three sections that can each be toggled to fold-in or out, to keep your inspector more manageable. Each section, and the configuration of each of their sub-items, are discussed in detail later.

“Simple Side-Menu”

1. Basic Settings:
    a. Placement
    b. Default State
    c. Transition Speed
2. Drag Settings:
    a. Threshold Drag Speed
    b. Threshold Dragged Fraction
    c. Handle:
        i. Draggable
        ii. Toggle State on Pressed
    d. Menu Draggable
3. Overlay Settings:
    a. Use Overlay:
        i. Colour
        ii. Close on Pressed



☑ Simple Side Menu (Script)

Simple Side-Menu
Version: 1.0.0
Author: Daniel Lochner

Basic Settings
Placement: Left
Default State: Closed
Transition Speed: 10

Drag Settings
Threshold Drag Spe: 0
Threshold Dragged F: 0.5
Handle: Handle
Draggable: ☑
Toggle State on P: ☑
Menu Draggable: □

Overlay Settings
Use Overlay: ☑
Colour
Close on Pressed: □

Configuration:

By hovering over a property’s label in the editor, a tooltip will be displayed, providing a basic description of what that property is. For a more in depth guide as to how to configure each property, the following written guide on “Configuration” has been provided:

Basic Settings:

1. Placement:
   The position at which the menu will be placed (Left, Right, Top or Bottom), which determines how the menu will be opened and closed. This sets the pivot, closed position, opened position and min and max anchors of the side menu.
2. Default State:
   Determines whether the menu will be open or closed by default.
3. Transition Speed:
   The speed at which the menu will snap into position when transitioning to the next state. It represents the value that will be multiplied by Time.deltaTime() when the side menu is linearly interpolated to the next state.

Drag Settings:

1. Threshold Drag Speed:
   The minimum speed required when dragging that will allow a transition to the next state.

2. Threshold Dragged Fraction:
   The fraction of the fully opened menu that must be dragged before a transition is allowed to occur to the next state (when the current drag speed is less than the threshold drag speed).
3. Handle:
   An optional GameObject used to open and close the side menu by dragging or pressing (when a Button component has been added) when each attribute has been enabled.
   a. Draggable:
      Determines whether the handle should be able to be used to drag the Side-Menu.
   b. Toggle State on Pressed:
      Determines whether the handle should be able to be used to toggle the Side-Menu's state when it is pressed.
4. Menu Draggable:
   Determines whether the Side-Menu itself should be draggable.

Overlay Settings:

1. Use Overlay:
   Determines whether an overlay should be used when the Side-Menu is opened. If enabled, the overlay is instantiated immediately "below" the Side-Menu on setup.
   a. Colour:
      The colour of the overlay. A state progress property is used to determine the alpha value of this colour which is linearly interpolated between 0 and 1 (0 when the menu is closed, and 1 when the menu is open).
   b. Close on Pressed:
      Determines whether the Side-Menu should be closed when the overlay is pressed.

Scripting References:

Controlling a "Simple Side-Menu" from outside of the class requires that you invoke public methods and/or get/set the values of properties defined. Each class that allows this level of control is discussed below, with each publically accessible property/method listed and described.

"SimpleSideMenu.cs"

1. Namespace: DanielLochner.Assets.SimpleSideMenu
2. Extends: MonoBehaviour
3. Implements: IBeginDragHandler, IEndDragHandler, IDragHandler, IInitializePotentialDragHandler
4. Properties:

Property	Description
CurrentState	The current state of the Side-Menu.
TargetState	The current targeted state of the Side-Menu.
StateProgress	The fraction representing the progress of the Side-Menu concerning its state (0 being closed and 1 being open).

5. Methods:

Method	Description
ToggleState()	Toggles the state of the Side-Menu (if it is currently open, it will set the targeted stated to be closed, whereas if it is currently closed, it will set the targeted state to be open)
Open()	Sets the targeted state to be open.
Close()	Sets the targeted state to be closed.

Support

Frequently Asked Questions (FAQ):

1. Where are all the Frequently Asked Questions?
   Once enough questions have been frequently asked, I will update the documentation and add them here ;)

Contact Information:

In the event you are unable to find the information you are looking for or have found a bug, please feel free to send me an email (daniel@daniellochner.com) and I will get back to you as soon as possible.

Please do not leave a review, regarding an issue with the asset, on the Unity Asset Store unless you have first consulted me asking for help.