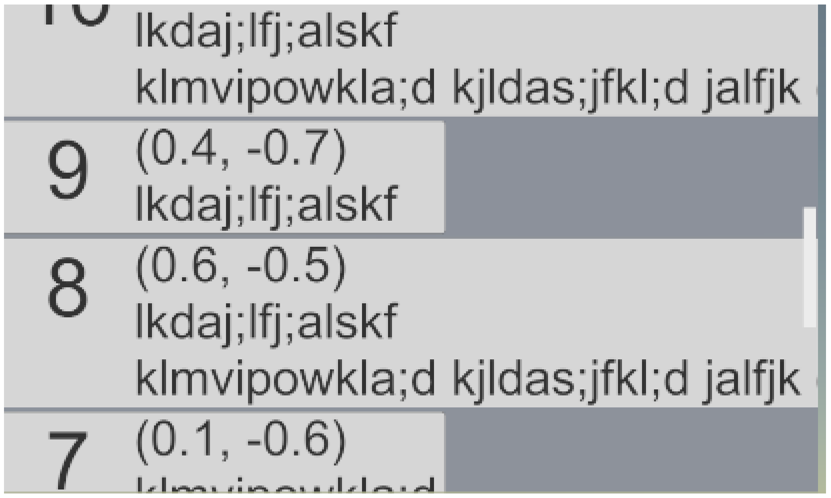
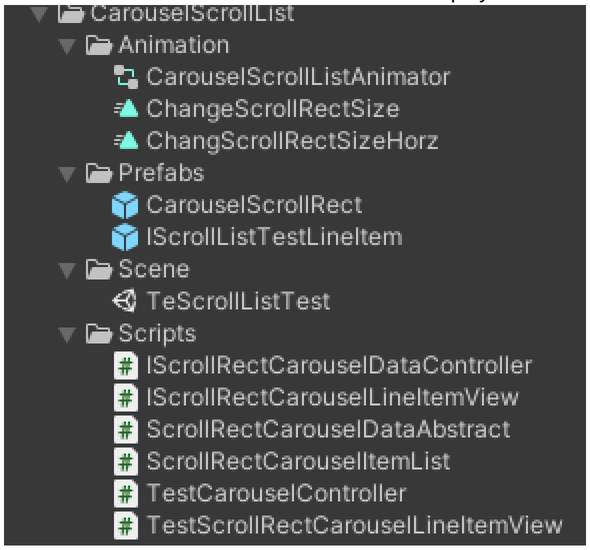
Scroll Rect Carousel



Overview

The ScrollRectCarousel is a UGUI scrollRect companion that displays the data in an efficient manner.
Instead of instantiating a gameObject for every entry in your data, the carousel only instantiates the number needed to visually cover your scrollRect display area. Then it carousels through the data items for display.

The ScrollRectCarousel uses the UGUI ScrollRect along with a ScrollContent gameObject and ItemContainer GameObject to facilitate the drag & scroll then the CarouselItemList container to display the carousel items.

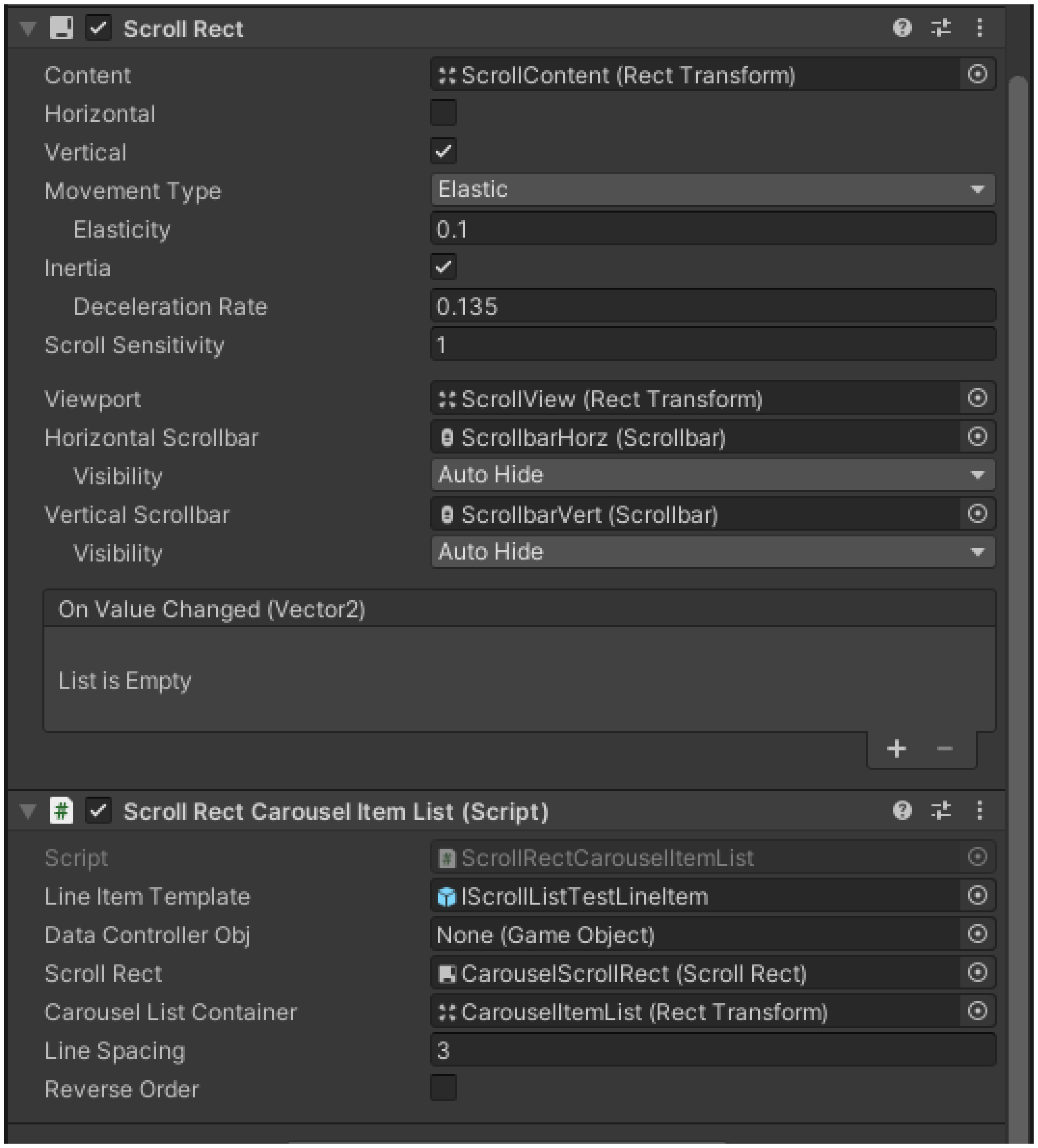


```
CarouselScrollList
  Animation
    CarouselScrollListAnimator
    ChangeScrollRectSize
    ChangScrollRectSizeHorz
  Prefabs
    CarouselScrollRect
    IScrollListTestLineItem
  Scene
    TeScrollListTest
  Scripts
    IScrollRectCarouselDataController
    IScrollRectCarouselLineItemView
    ScrollRectCarouselDataAbstract
    ScrollRectCarouselItemList
    TestCarouselController
    TestScrollRectCarouselLineItemView
```

The ScrollRectCarousel uses an MVC architecture and the package includes the following core files:

1. ScrollRectCarouselDataAbstract - An abstract base class for line item model to inherit from. This provides a way for the Controller & View to work with the model
2. ScrollRectCarouselItemList - This is the core of the carousel scroll rect. It contains logic for instantiating the needed list items and it controls how items are updated and displayed in the list. It has a corresponding example prefab CarouselScrollRect and is used in conjunction with the UGUI scrollRect and must use various ScrollRectCarousel interfaces and the base dataAbstract class to function.
3. IScrollRectCarouselDataController - This interface provides the user a way to request a view refresh by having access to the list of data and reference to the view interface for render.
4. IScrollRectLineItemView - This is the interface for the individual line item display. Implementing this will provide the core scroll list with necessary info like line item width/height, min width/height which is will use to populate the scroll list. This also gives the line item a way to be update given the data.

The ScrollRectCarousel package also includes the CarouselScrollRect prefab, this is a springboard prefab that has the setup for a functioning ScrollRect Carousel that can be customized.



Scroll Rect

Content	ScrollContent (Rect Transform)
Horizontal	☐
Vertical	✓
Movement Type	Elastic
Elasticity	0.1
Inertia	✓
Deceleration Rate	0.135
Scroll Sensitivity	1
Viewport	ScrollView (Rect Transform)
Horizontal Scrollbar	ScrollbarHorz (Scrollbar)
Visibility	Auto Hide
Vertical Scrollbar	ScrollbarVert (Scrollbar)
Visibility	Auto Hide

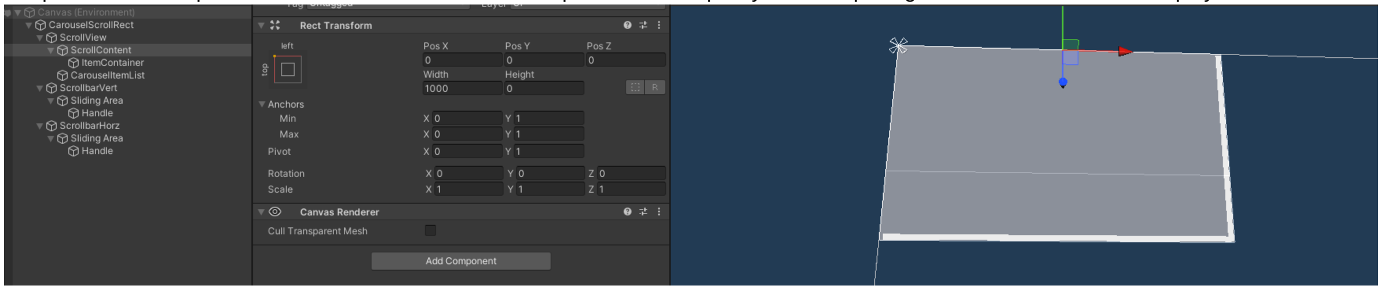
On Value Changed (Vector2)

List is Empty

Scroll Rect Carousel Item List (Script)

Script	ScrollRectCarouselItemList
Line Item Template	IScrollListTestLineItem
Data Controller Obj	None (Game Object)
Scroll Rect	CarouselScrollRect (Scroll Rect)
Carousel List Container	CarouselItemList (Rect Transform)
Line Spacing	3
Reverse Order	☐

As a companion component, user still uses the UGUI ScrollRect to control the scroll direction, movement values and scroll bar behavior. The ScrollRectCarouselItemList allows user to link a Controller object in editor or they can set it in code. User need to link it to the ScrollRect component and to a prefab that'll serve as the line Item template and can specify the line spacing and the if the list will be display in reverse order.



```
Canvas (Environment)
  CarouselScrollRect
    ScrollView
      ScrollContent
        ItemContainer
      CarouselItemList
    ScrollbarVert
      Sliding Area
        Handle
    ScrollbarHorz
      Sliding Area
        Handle
```

Rect Transform

	Pos X	Pos Y	Pos Z
	0	0	0
	Width	Height	
	1000	0	
Anchors Min	X 0	Y 1	
Anchors Max	X 0	Y 1	
Pivot	X 0	Y 1	
Rotation	X 0	Y 0	Z 0
Scale	X 1	Y 1	Z 1

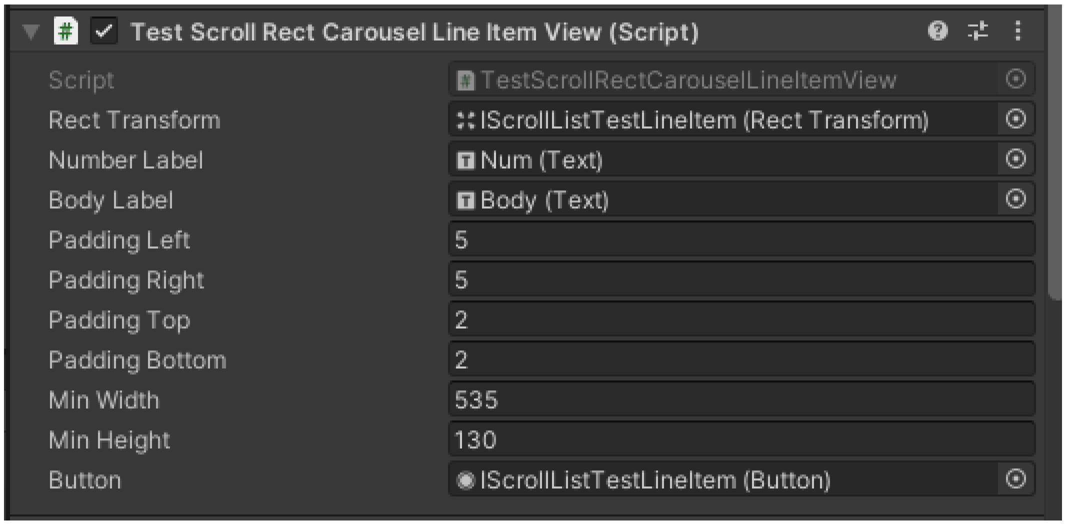
Canvas Renderer

Cull Transparent Mesh ☐

Under the ScrollView object there are ScrollContent and CarouselItemList. Traditionally all of the scroll list items are instantiated and placed under the ScrollContent, in a carousel list however, the ScrollContent is empty but its width & height is set to match the size of the full of list of line items. Then under the CarouselItemList we instantiate just enough items to cover the view port and as the user scrolls we refresh the visuals so we create an illusion of a complete list without the performance penalty.

The ScrollContent's dimension is automatically adjusted by the ScrollRectCarouselItemList script

The IScrollListTestLineItem is an example of how the line item prefab can be implemented



Test Scroll Rect Carousel Line Item View (Script)

Script	TestScrollRectCarouselLineItemView
Rect Transform	IScrollListTestLineItem (Rect Transform)
Number Label	Num (Text)
Body Label	Body (Text)
Padding Left	5
Padding Right	5
Padding Top	2
Padding Bottom	2
Min Width	535
Min Height	130
Button	IScrollListTestLineItem (Button)

The Min Width & Height are important here because the scroll list uses these values to calculate how many line items it need to generate to cover the entire view port. The RectTransform should point to the line item's top level RectTransform, the scroll list is set to support non uniform sized line items and it needs this reference to change the dimension of the line item to fit the content and correctly calculate the list's total size.

Reference

Use the namespace "ScrollRectCarousel"

ScrollRectCarouselItemList

```
//when interacting in code the key function to call is InitCarousel,
//this will either initialize a new carousel or append to the existing
one depending
//on if the “recalcListDimension” parameter

// Template to the item to display in the scroll list template must
implement IScrollRectCarouselLineItemView
[SerializeField]
private GameObject lineItemTemplate;

// The UGUI scrollRect
[SerializeField]
private ScrollRect scrollRect;

// This contains the instantiated line items for the carousel
[SerializeField]
private RectTransform carouselListContainer;

// Spacing between line items
[SerializeField]
private float lineSpacing;

// Display the list in reverse order
[SerializeField]
private bool reverseOrder;

// Reference to the line item template
public IScrollRectCarouselLineItemView LineItemTemplate { get { return
lineItemInterface; } }

// Link to the data controller can be either linked in editor or use
this set function to do it in code
public void SetDataController(IScrollRectCarouselDataController
controller)

/// <summary>
/// This initialize the carousel scroll list and depending on number of
data items and the
/// UGUI scroll rect's horizontal/vertical setting it'll initialize the
item pool and scroll content sizing
/// </summary>
/// <param name="totalLineItem">Total numer of line times</param>
/// <param name="recalcListDimension">If we want to recalculate the
list dimension or just calcuate dimension from num of existing list
items to new totalLineItem</param>
public void InitCarousel(int totalLineItem, bool recalcListDimension =
true)
```

IScrollRectCarouselDataController

```
/// <summary>
/// Data controller that creates a way to update the item by index in
the data array the implemented function would look like this
///
///     public void UpdateLineItemView(int index,
IScrollRectCarouselLineItemView lineItemView)
///     {
///         lineItemView.UpdateViewBaseOnModel(dataList[index]);
///     }
///
/// </summary

/// <summary>
/// A way to update the line item template base on an index
/// </summary>
/// <param name="index">index of the data you want to render</param>
/// <param name="lineItemView">reference to the template view</param>
void UpdateLineItemView(int index, IScrollRectCarouselLineItemView
lineItemView);

/// <summary>
/// Returns the length of the data list
/// </summary>
/// <returns>length of the data list</returns>
int GetDataListCount();

/// <summary>
/// Returns if the data is selected
/// </summary>
/// <param name="index">index of the data</param>
/// <returns></returns>
bool IsDataSelected(int index);

/// <summary>
/// Selects an item, should set the data's IsSelected to true
/// </summary>
/// <param name="index">index of the data</param>
void SetDataSelected(int index, bool newIsSelected);

/// <summary>
/// Clicks a item by index
/// </summary>
/// <param name="index">index of the data</param>
void OnItemClicked(int index, IScrollRectCarouselLineItemView
lineItemView);
```

IScrollRectCarouselLineItemView

```
/// <summary>
/// The display item must implement this interface and return either
the Height or Width preferably both for the carousel to
/// calculate the number of display items it needs to instantiate
///
/// The prefab IScrollListTestLineItem is the example prefab
/// </summary>

// Returns the height and width of the line item this is used to
calculate how tall our scroll area is and how many line item is needed
for display
float Height { get; }
float Width { get; }

// Returns the min width & height for calculating how many items we
need to fill the scroll rect
float MinHeight { get; }
float MinWidth { get; }

// Returns a reference to the gameObject
GameObject GameObject { get; }

UnityEngine.UI.Button Button { get; }

// Returns a reference to the rect transform
RectTransform RectTran { get; }

// Must implement this to refresh the view of the line item
void UpdateViewBaseOnModel(ScrollRectCarouselDataAbstract data);

// For handling selecting & deselecting, need a silence way to do this
to avoid unnessary visuals as we recyle items
void SilentSelect();
void SilentDeselect();

// Get reference to the current data
ScrollRectCarouselDataAbstract Data { get; }
```

ScrollRectCarouselDataAbstract

```
/// <summary>
/// The line item data must inherit fromt this abstract to give a clean
way to update the view of the item
/// </summary>
public abstract class ScrollRectCarouselDataAbstract
{
    // For holding the index in the data list
    public int index;

    // If the item is selected
    public bool isSelected;
}
```

Quick Start Guide

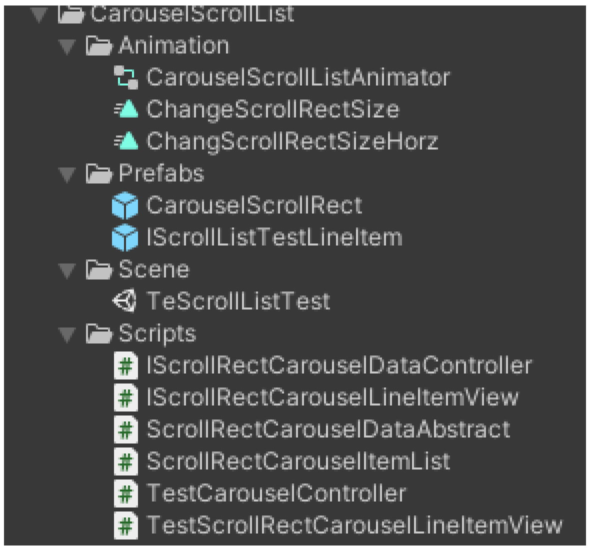
In Editor

1. Import the package
2. Create your line item template, this is how you'll display your data. Make sure it has a component that implements IScrollRectCarouselLineItemView
3. Place the CarouselScrollRect prefab into your scene or prefab and position and size it as you wish.
4. The ItemContainer is just used as a background and can be removed if desired
5. In the Scroll Rect component, specify the scroll direction and the movement values. Also you can adjust your scroll bar visuals
6. In the Scroll Rect Carousel Item List component, point to the line item template you created and the data controller obj if you are linking it in editor.
7. You can also change the spacing and the display order

In Code

1. Create your data object that inherits from ScrollRectCarouselDataAbstract
2. Implement a data controller that implements the IScrollRectCarouselDataController interface
    a. Your data controller should have access to the list of your data. It must be able to access the data by index in order to update the correct item.
3. In your line item template, implement the IScrollRectCarouselLineItemView
    a. You must have a min height/width so Carousel knows how many line item it needs to generate to cover the viewport
    b. You must have an accurate way to calculate the line item's width/height, this is used for scrolling and for non uniform sized content.
    c. If your line items are fixed dimension you can just return the dimension value
    d. If your scroll only in one direction you can just return 0 for the dimension that's not used.
    e. If your line item does not support interaction then return null for the Button property and make the SilentSelect and SilentDeselect do nothing
    f. SilentSelect and SilentDeselect is a way for the line item to be set to the select/deselect state without animation or effect since as the list scrolls we need to turn on/off the highlight.

For example on how to implement the interfaces and setup the prefabs look at the scene TeScrollListTest



```
▼ CarouselScrollList
    ▼ Animation
        CarouselScrollListAnimator
        ChangeScrollRectSize
        ChangScrollRectSizeHorz
    ▼ Prefabs
        CarouselScrollRect
        IScrollListTestLineItem
    ▼ Scene
        TeScrollListTest
    ▼ Scripts
        IScrollRectCarouselDataController
        IScrollRectCarouselLineItemView
        ScrollRectCarouselDataAbstract
        ScrollRectCarouselItemList
        TestCarouselController
        TestScrollRectCarouselLineItemView
```