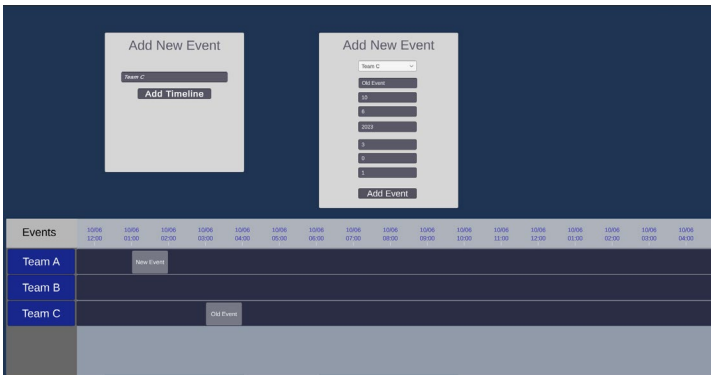
Timeline Viewer

GAAWCITY

The Timeline Viewer can be used for 2D & 3D UI displays, to a scrollable timeline. The start time and end time can be set via the editor or programmatically. Programmatically timeline swim lanes for different sets of events. As well as adding events to different swimlanes.



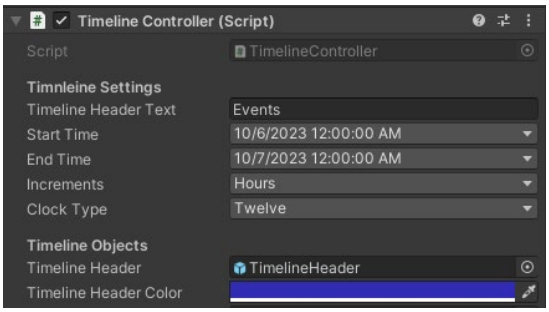
How to use Timeline Viewer

In Unity select the Timeline Viewer from the package manager. Click the install button.

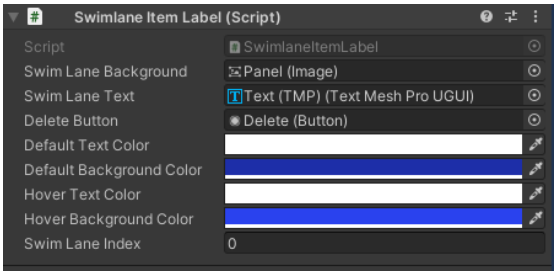
Once Timeline Viewer I installed. You can simply navigate to the Prefabs folder under GAAWCITY → TimelineUI and drag the Timeline prefab under the canvas in the Hierarchy of your choosing.

In the editor under the Timeline Control component, setup the event header label text. Select the timelines start and end date/time. Specify the timeline increments you would like to use. Then specify the timeline clock format. (12 hour or 24 hour)

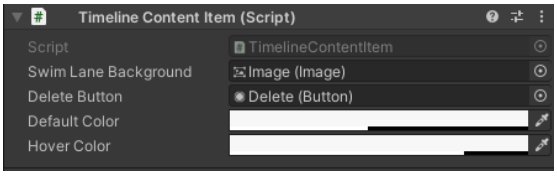
You can also change the color of the timeline tickers as well as the event block background and text colors. To change the header label color, use the color picker on the Timeline Controller component.



To change the color of the swimlane label, navigate to the Components folder and select the SwimlaneItemLabel prefab. Use the color picker settings on the Swimlane Item Label component.



To change the color of the events, select the SwimlaneContentItem prefab. Use the color picker settings on the Timeline Content Item component.



Programmatically add swimlanes & events.

Adding swimlanes and events to swimlanes can be accomplished using the Timeline Controller script. Get a reference to the desired Timeline gameobject (You can set up as many as you want in a scene). Get the Timeline Controller component using "gameobject.GetComponent<TimelineController>()". Now with that reference, Use the AddNewSwimlane method providing the swimalanes name as a string parameter.

```
AddNewSwimlane("Swimlane")
```

That's it, the new swimlane will appear when the code is executed.

You can add events to swim lanes by calling the AddEventToTimeline method which has two signatures taking three or four parameters. The three parameter method takes a string for the

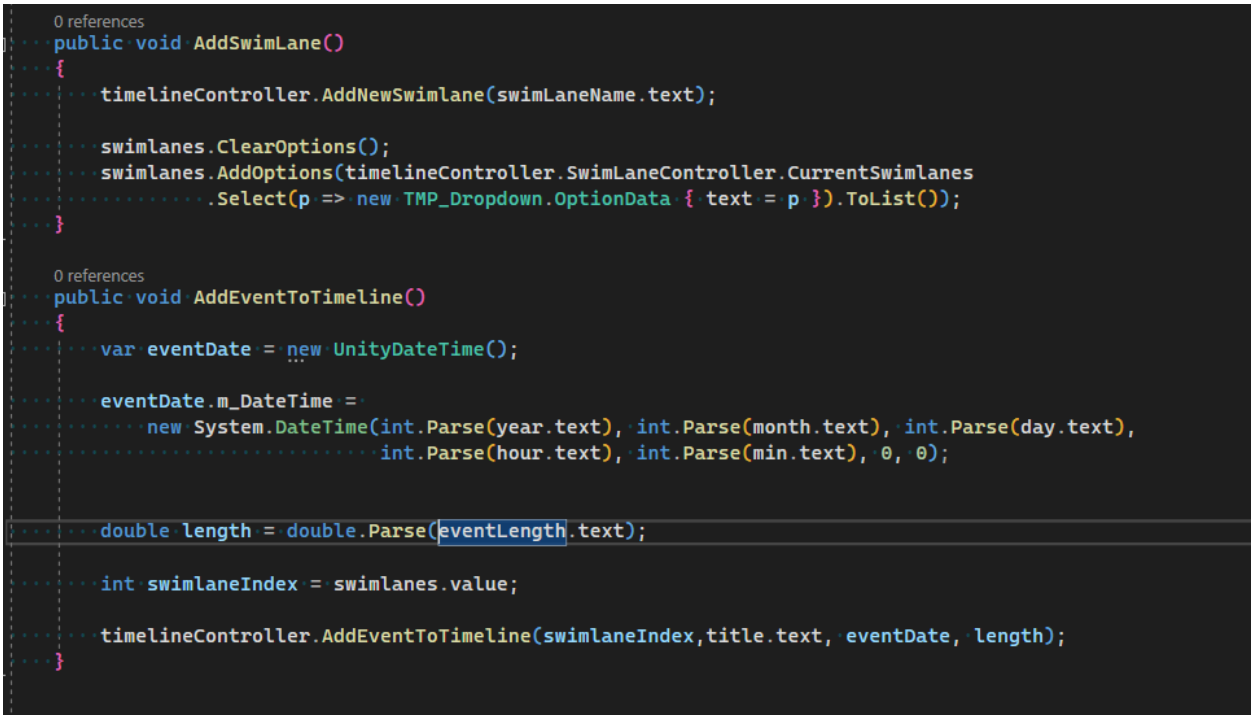
title of the event, a UnityDateTime for the event date and time and a double for the length of the event (ex: 1.5 represents 1 hour and 30 minutes).

AddEventToTimeline("Title", eventDateTime, 1.5)

The four parameters method takes an additional in for the first parameter int to determine the desired swimlane for the event to appear on.

AddEventToTimeline(0, "Title", eventDateTime, 1.5)

SampleScene example code



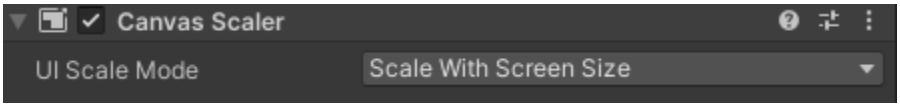
```
0 references
public void AddSwimLane()
{
    timelineController.AddNewSwimlane(swimLaneName.text);

    swimlanes.ClearOptions();
    swimlanes.AddOptions(timelineController.SwimLaneController.CurrentSwimlanes
        .Select(p => new TMP_Dropdown.OptionData { text = p }).ToList());
}

0 references
public void AddEventToTimeline()
{
    var eventDate = new UnityDateTime();

    eventDate.m_DateTime =
        new System.DateTime(int.Parse(year.text), int.Parse(month.text), int.Parse(day.text),
                            int.Parse(hour.text), int.Parse(min.text), 0, 0);


    double length = double.Parse(eventLength.text);

    int swimlaneIndex = swimlanes.value;

    timelineController.AddEventToTimeline(swimlaneIndex,title.text, eventDate, length);
}
```

Different screen resolutions

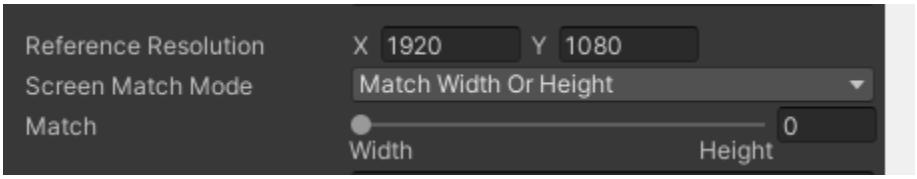
The Timeline Viewer uses Unity's canvas scaler to resize for different screen resolutions.

Make sure the canvas that the Timeline Viewer is under has a canvas scaler on it. (*Best practice is to put the Timeline Viewer in it's on canvas if possible)*

Select the screen resolution that you will be developing in. Set the canvas scalers UI Scale Mode to "Scale With Screen Size".



Set the Reference Resolution to the developing screen resolution. Slide the Match slider all the way towards width, value of 0. Adjust the Timeline Viewer width, to the desired width.



In the inspector set Timeline Viewers anchor position to bottom stretch.