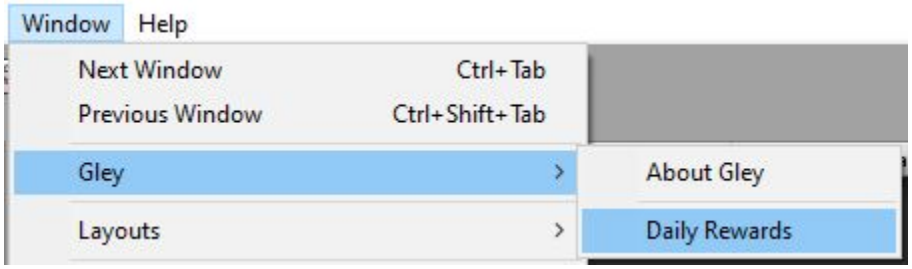
DAILY (TIME BASED) REWARDS

1. WHY DO YOU NEED TO USE THIS PLUGIN

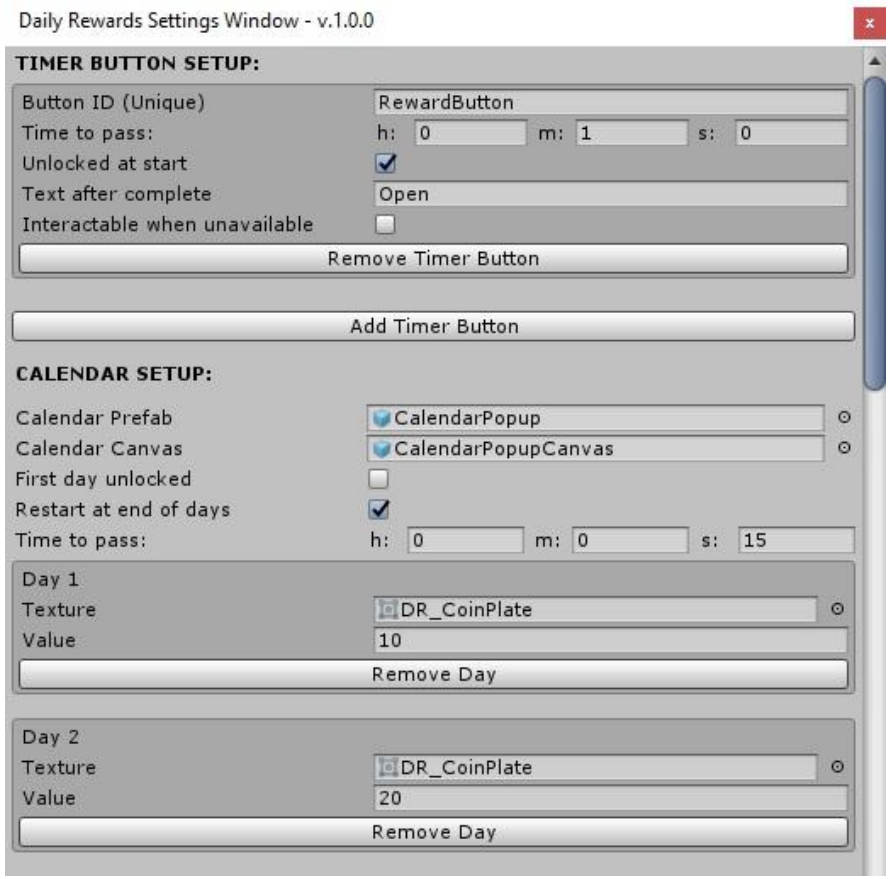
- Increase the retention of your game by using Daily Rewards and Time Based rewards.
- Contains Timer buttons and Calendar popup rewards.
- Easy to use and display with a single line of code.
- Automatically adjusts resolution for both Landscape and Portrait mode.
- All setup is done in a Settings Window.
- Supports any number of days.
- Customizable time between consecutive rewards.
- First day can be open or not by default, based on settings.
- Full commented code, popups and demo scenes included.
- Works with Unity 2017.1 and above with Free, Plus or Pro license.
- Works on all supported Unity platforms.

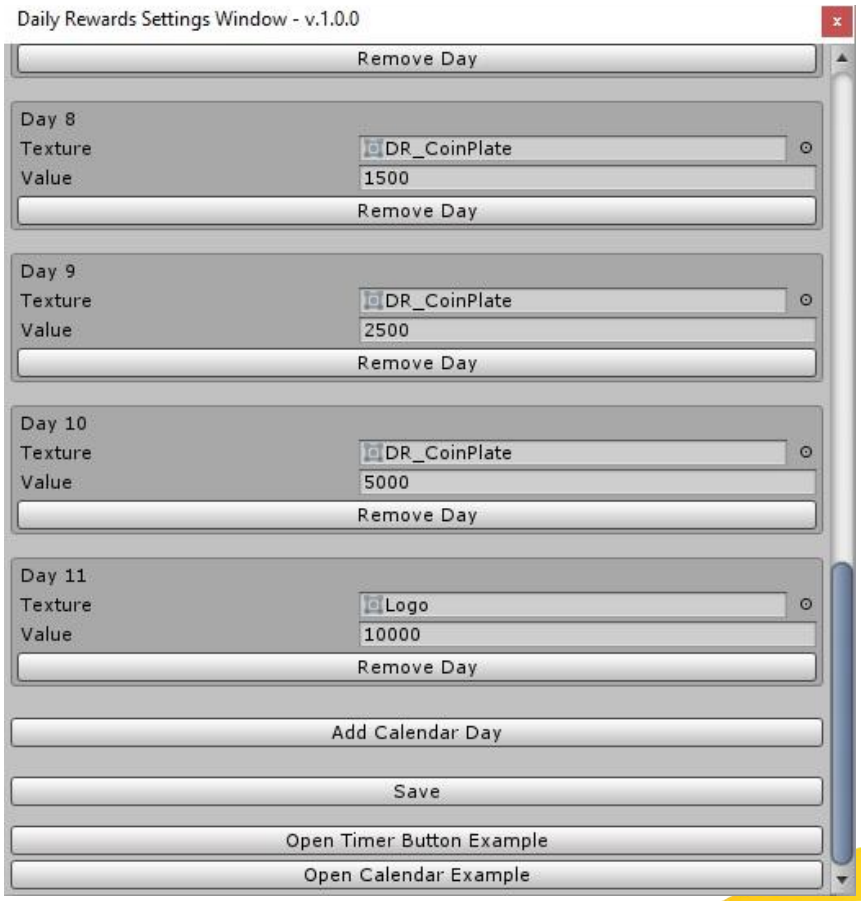
2. SETUP GUIDE

- Import **Gley Daily Rewards Plugin** into Unity.
- Go to **Window->Gley->Daily Rewards** to open the Settings Window.

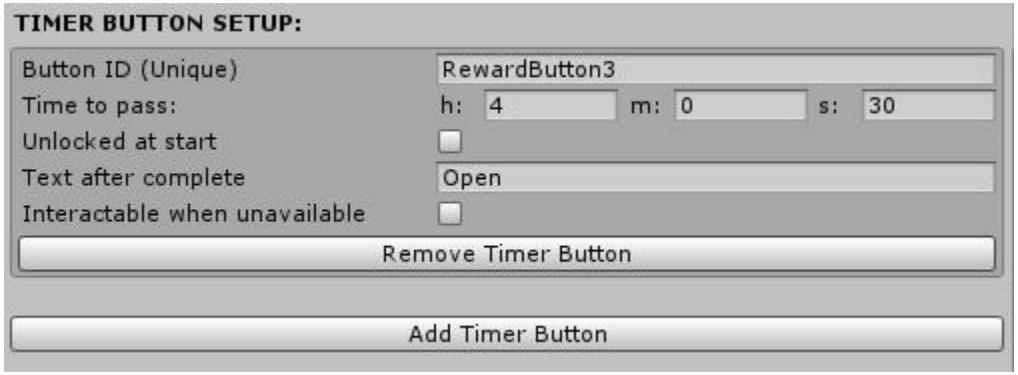


- Settings Window will open



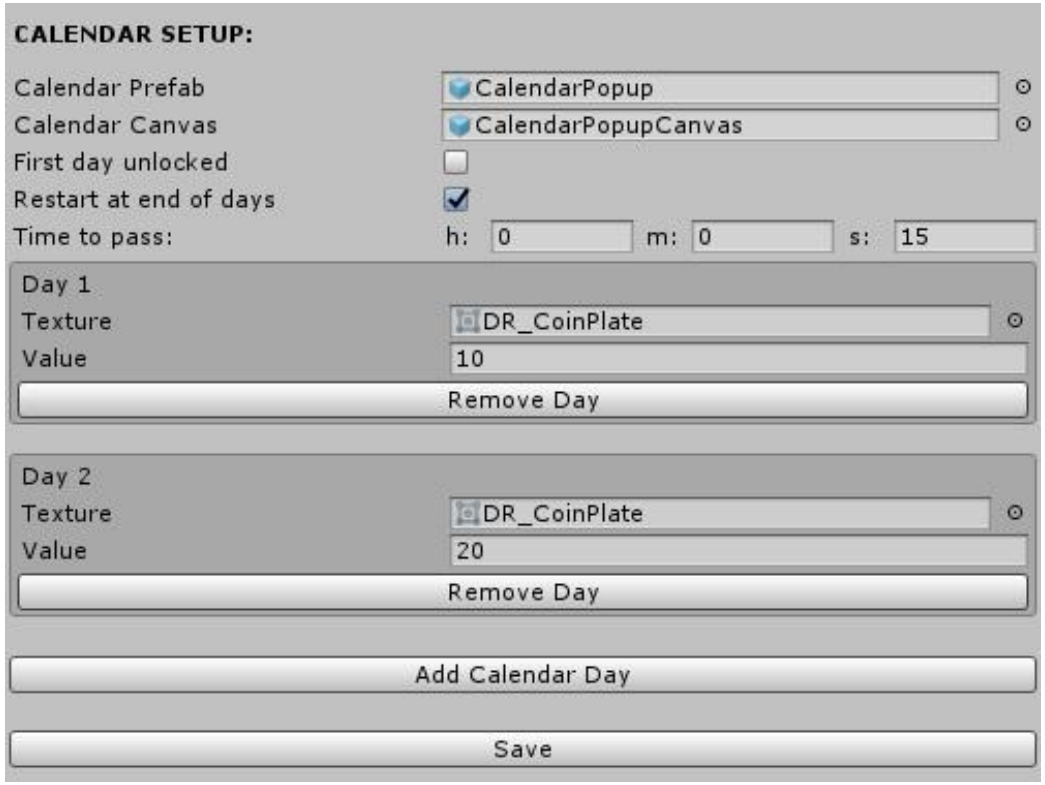


Setup Timer Button



- **Button ID** -> a unique id for each button, it needs to be assigned on timer button script
- **Time to pass** -> time to pass for the reward to be available
- **Unlocked at start** -> if clicked the button starts as available and only after click the timer starts
- **Text after complete** -> text for the button when timer is 00:00:00
- **Interactable when unavailable** -> button can be clicked even when timer is running.
- **Remove Timer Button** -> deletes the current timer button
- **Add Timer Button** -> creates a new timer button

Setup Calendar Popup



Setup Calendar Popup

- **Calendar Prefab** -> the prefab that contains the calendar popup. Is located inside: GleyPlugins/DailyRewards/Prefabs/CalendarPopup
- **Calendar Canvas** -> the prefab that contains the popup canvas. Is located inside: GleyPlugins/DailyRewards/Prefabs/CalendarPopupCanvas
- **First day unlocked** -> if checked the first day is available from start, without the required time to pass
- **Reset at end of days** -> if checked after the last day is unlocked it will start from the beginning
- **Time to pass** -> time for unlocking the next day

Day setup:

- **Texture** -> the texture from the day prefab - should be representative for your rewards
- **Value** -> the reward amount

3. USER GUIDE - TIMER BUTTON

Subscribe to click listener from any script so you can grant and store the reward

- **GleyDailyRewards.TimerButton.AddClickListener(RewardButtonClicked);**

```
// buttonID -> the ID of the clicked button - was set up inside settings window
// timeExpired -> true if time expired, means that the reward can be granted
private void RewardButtonClicked(TimerButtonIDs buttonID, bool timeExpired)
{
    if(timeExpired)
    {
        //if(buttonID == RewardButtonIDs.YourButtonID)
        //{
            //give reward for this button ID
        //}
    }
    else
    {
        //not ready yet, you have to wait
    }
}
```

3. USER GUIDE - TIMER BUTTON

Get the remaining time for a specific button

- **GleyDailyRewards.TimerButton.GetRemainingTime(TimerButtonIDs.ButtonID);**

Reset timer for a specific button

- **GleyDailyRewards.TimerButton.ResetTimer(TimerButtonIDs.ButtonID);**

Add the amount of time to the specified button

- **GleyDailyRewards.TimerButton.AddTime(TimerButtonIDs.ButtonID, new TimeSpan(0, 1, 0));**

Remove the amount of time from the specified button

- **GleyDailyRewards.TimerButton.RemoveTime(TimerButtonIDs.ButtonID, new TimeSpan(0, 1, 0));**

4. USER GUIDE - CALENDAR POPUP

Show the calendar

- **GleyDailyRewards.Calendar.Show();**

Subscribe to click listener from any script so you can grant and store the reward

- **GleyDailyRewards.Calendar.AddClickListener(CalandarButtonClicked);**

```
// dayNumber -> current clicked day
// rewardSprite -> the sprite of the reward
// rewardValue -> the reward value for current day
```


```
private void CalandarButtonClicked(int dayNumber, int rewardValue, Sprite rewardSprite)
{
    yourRewardVariable += rewardValue;
}
```

Get the remaining time for next day

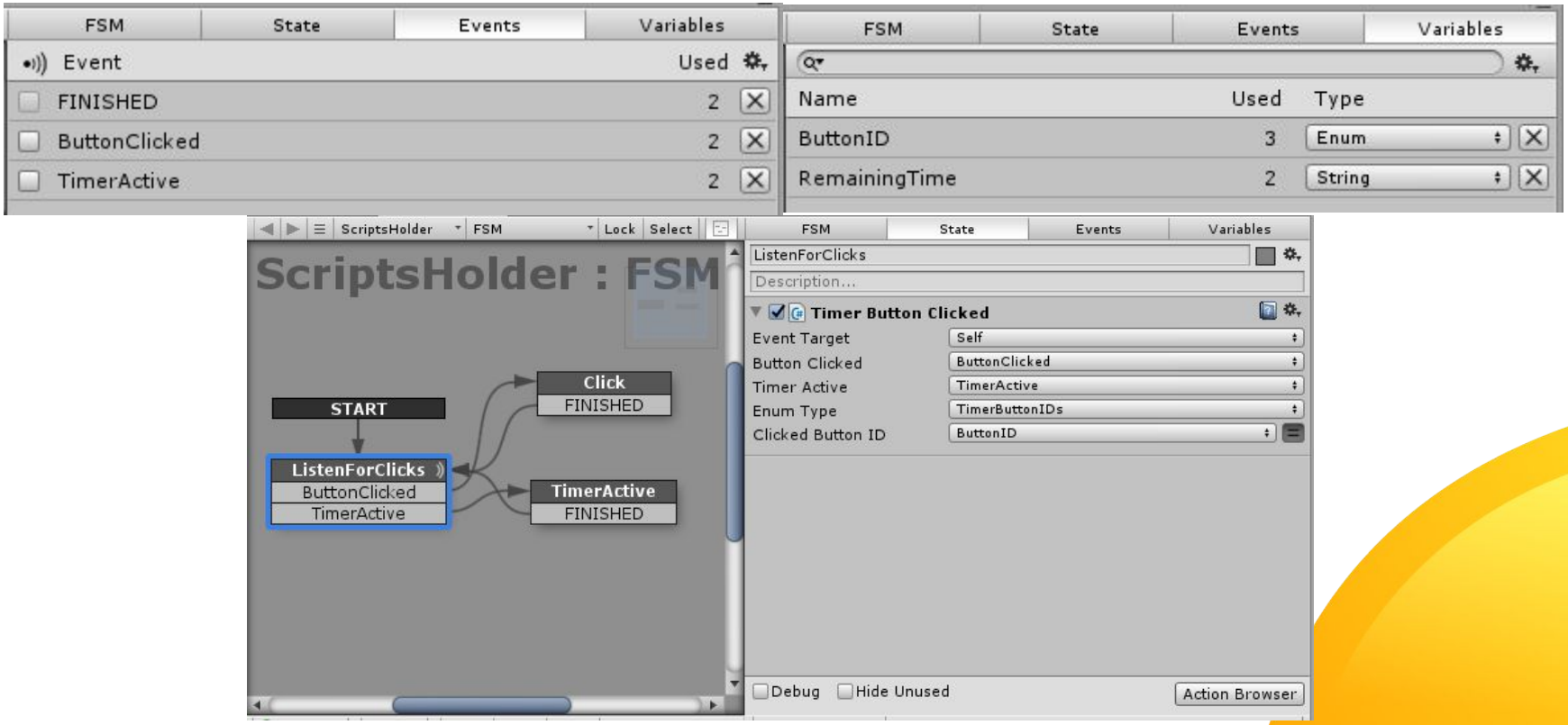
- **GleyDailyRewards.Calendar.GetRemainingTimeSpan();**

Reset calendar to day one

- **GleyDailyRewards.Calendar.Reset();**

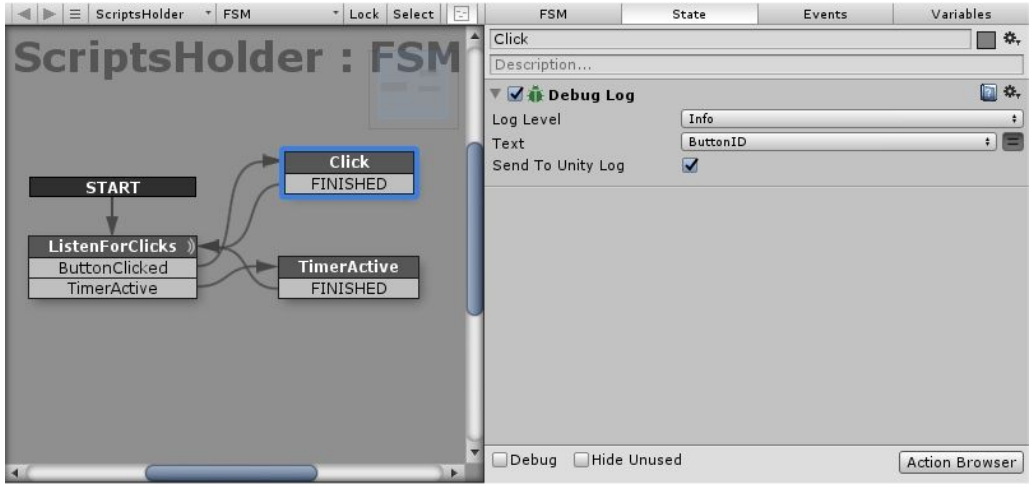
5. PLAYMAKER SUPPORT

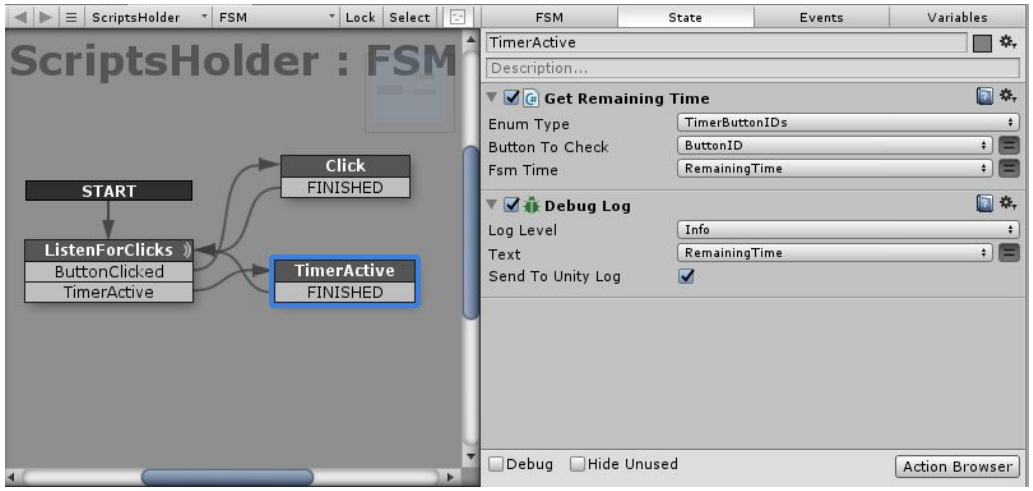
- **Supported Playmaker Actions:**
  - **Open Calendar**
  - **Timer Button Clicked**
  - **Get Remaining Time**
  -
- **Timer Button Clicked details**



5. PLAYMAKER SUPPORT

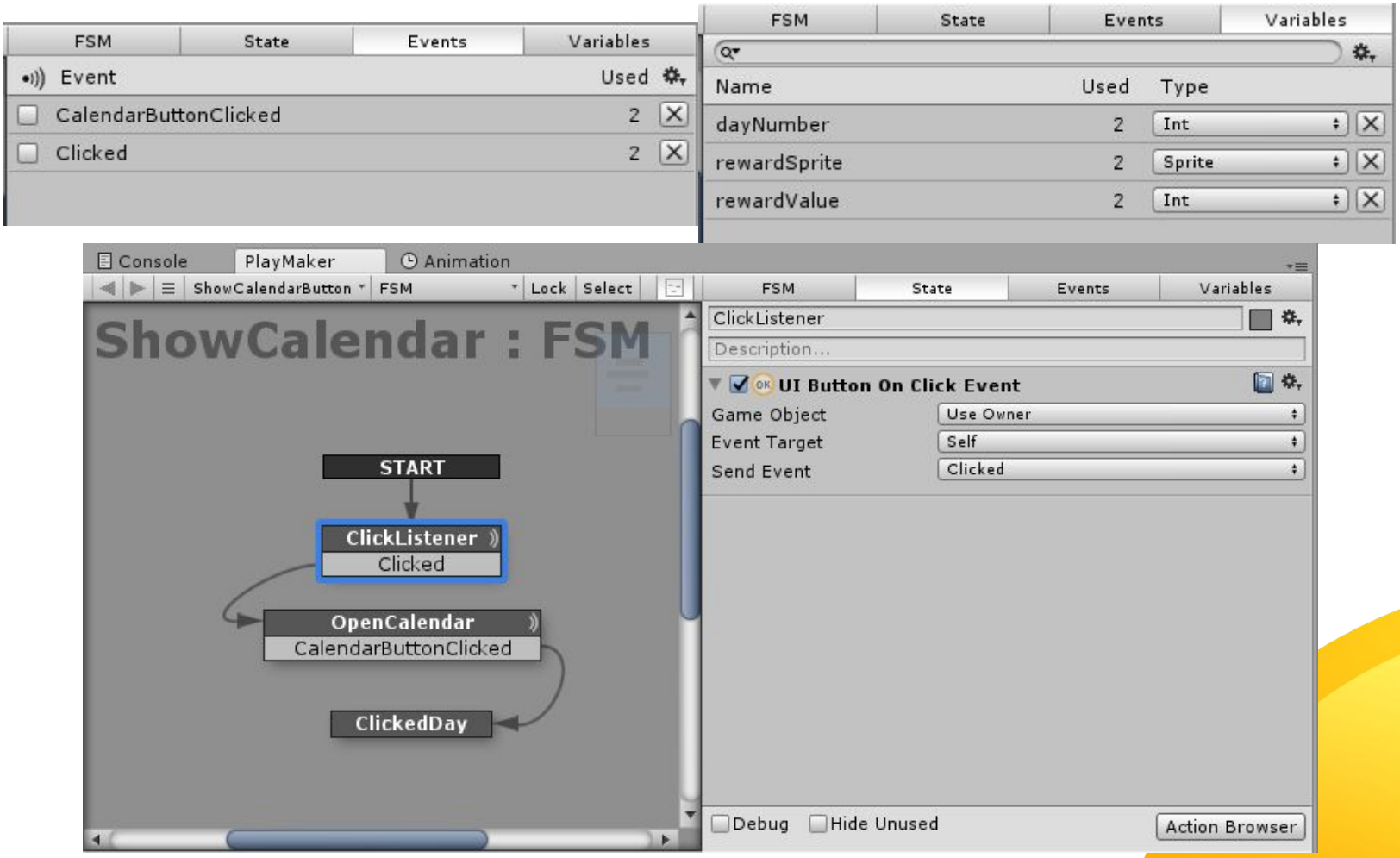
- **Timer Button Clicked details**





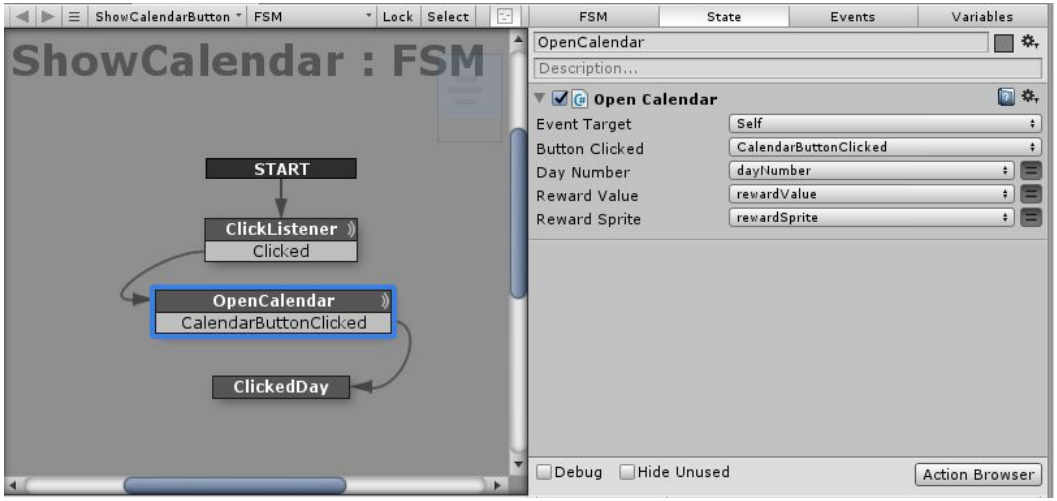
5. PLAYMAKER SUPPORT

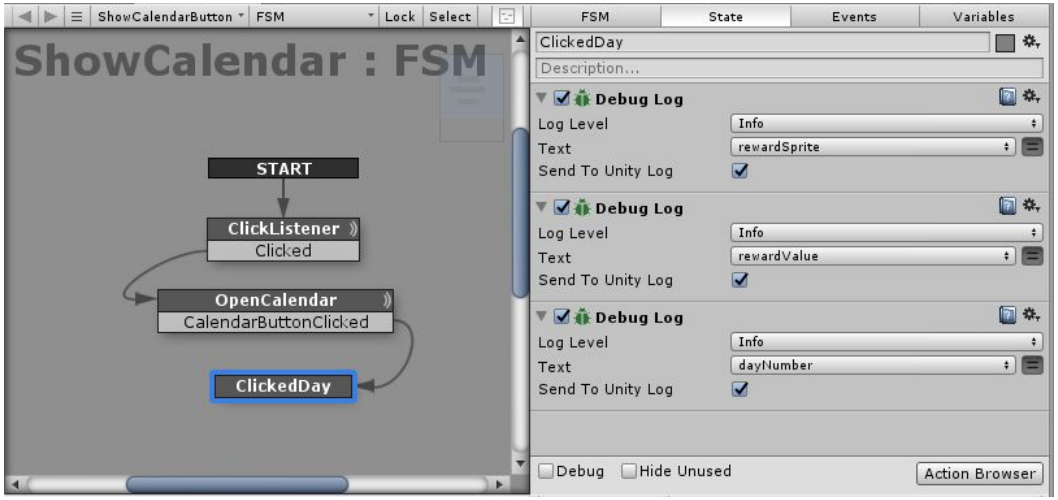
- **Open Calendar details**



5. PLAYMAKER SUPPORT

- **Open Calendar details**



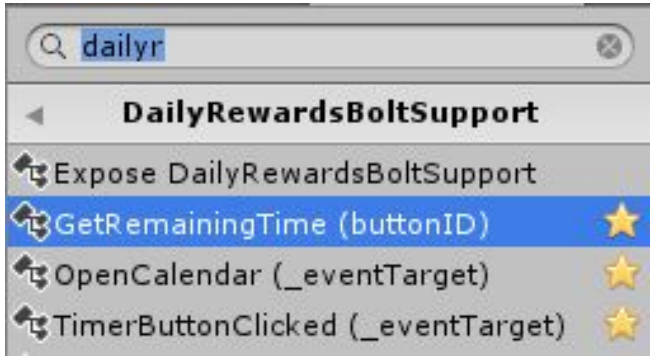


6. BOLT SUPPORT

- **Supported Bolt Actions:**
  - **Open Calendar**
  - **Timer Button Clicked**
  - **Get Remaining Time**

The above Bolt actions behavior is equivalent with corresponding methods from Section 3 - User Guide.

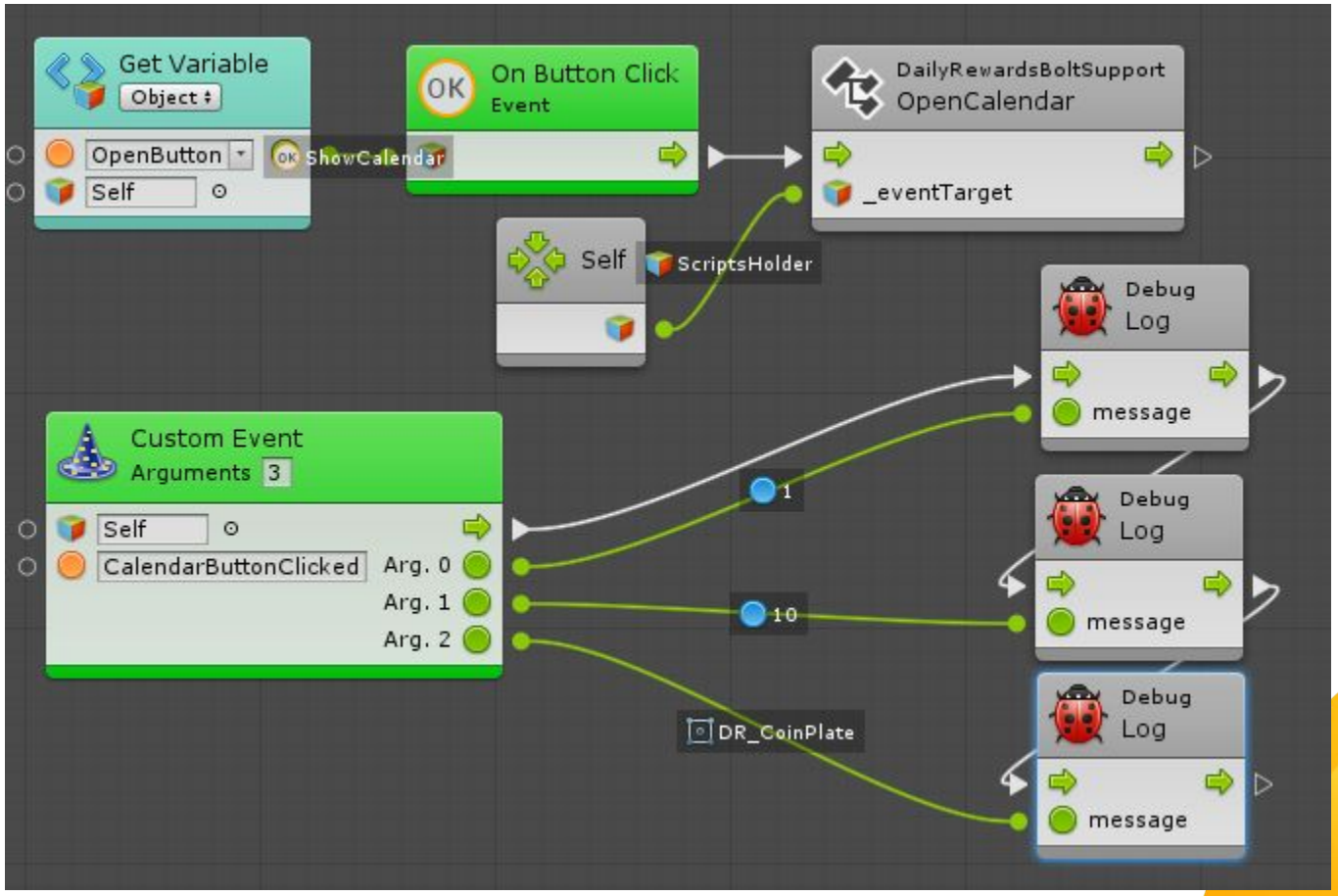
The above Bolt actions are located inside **DailyRewardsBoltSupport** script as shown below:



6. BOLT SUPPORT

- **Open Calendar details**
  Custom event arguments:

Arg. 0 -> clicked day number; Arg. 1 -> clicked day reward; Arg. 2 -> clicked day sprite



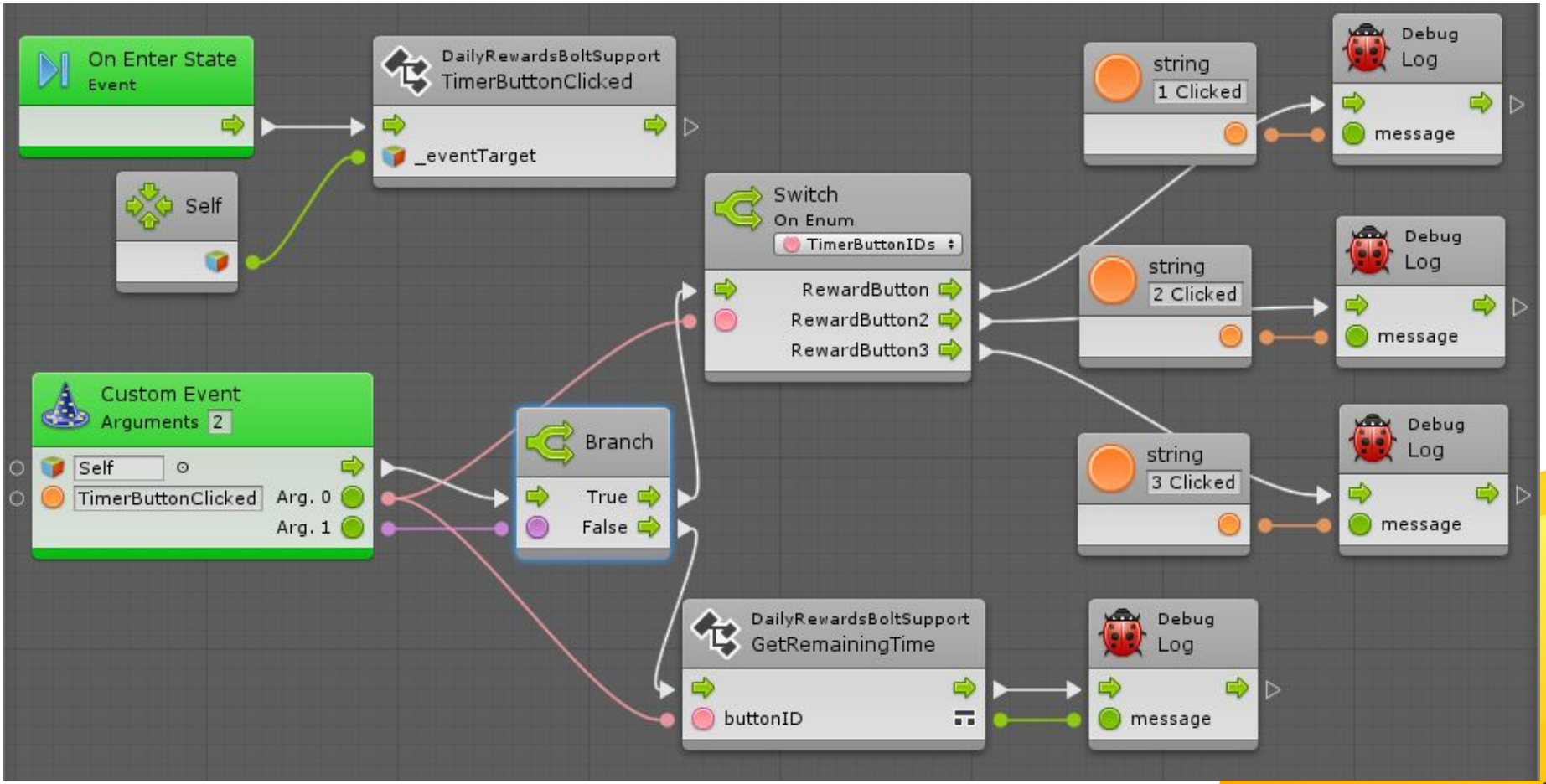
6. BOLT SUPPORT

- **Timer Button details**

  Custom event arguments:

Arg. 0 -> TimerButtonID enum - represents the clicked button;

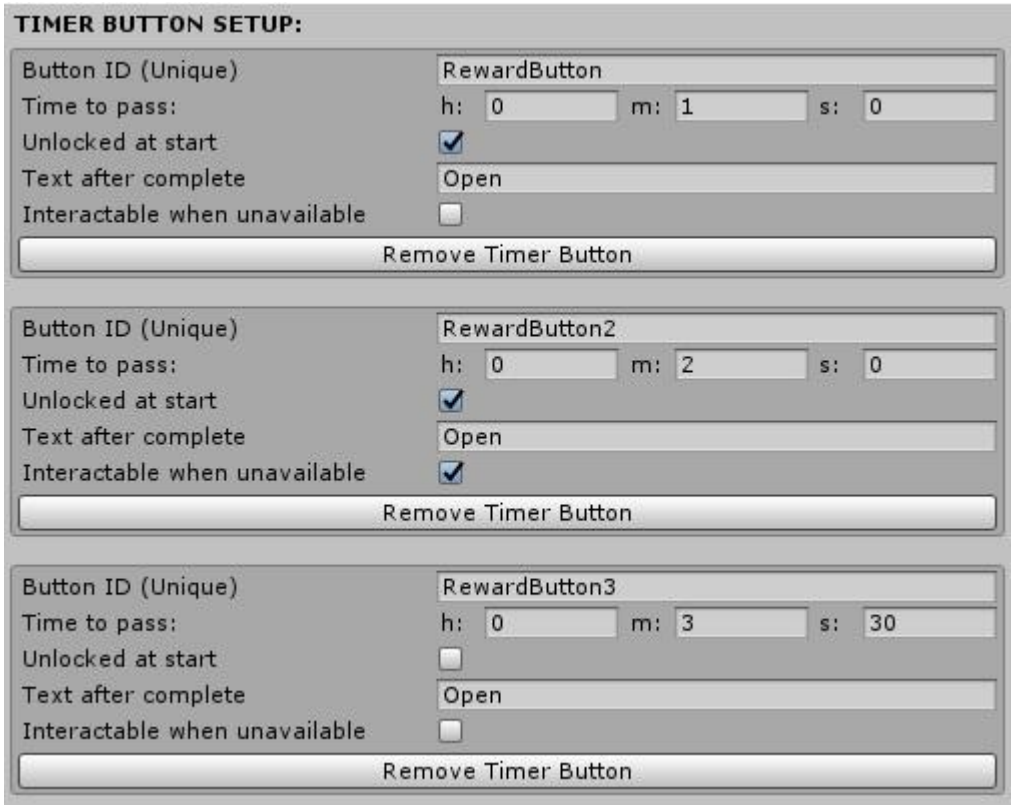
Arg. 1 -> bool - time expired

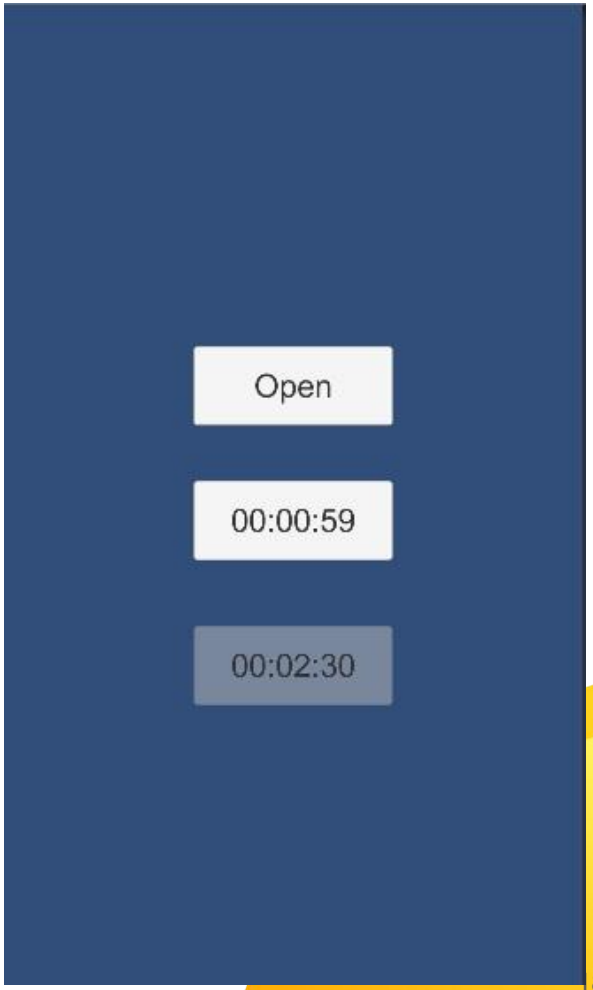


7. EXAMPLE 1 - Timer Buttons

You can find the example test scene here:
Assets/GleyPlugins/DailyRewards/Example/TimerButtonExample.unity

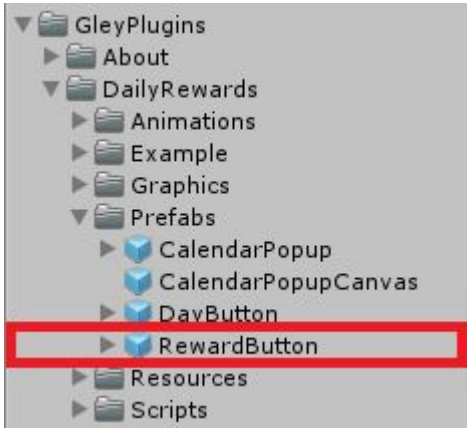
1. Setup all timer buttons from Settings Window:

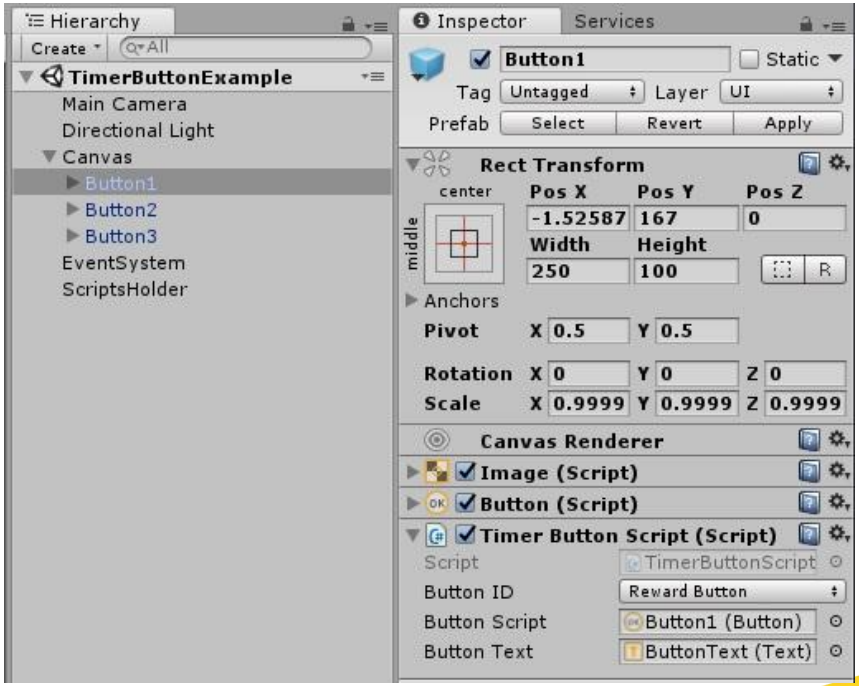




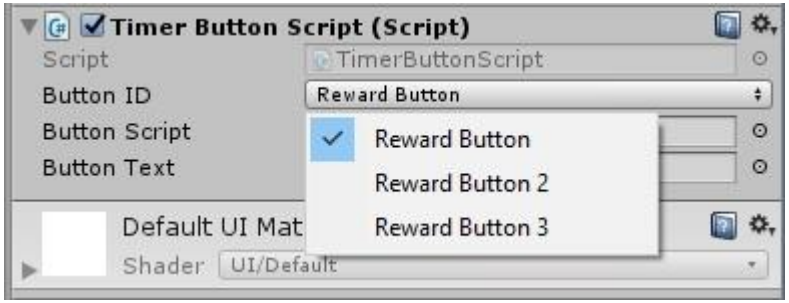
7. EXAMPLE 1 - Timer Buttons

2. Drag reward button prefab to the scene:





3. Associate the button with your settings by selecting the corresponding button ID from button script:



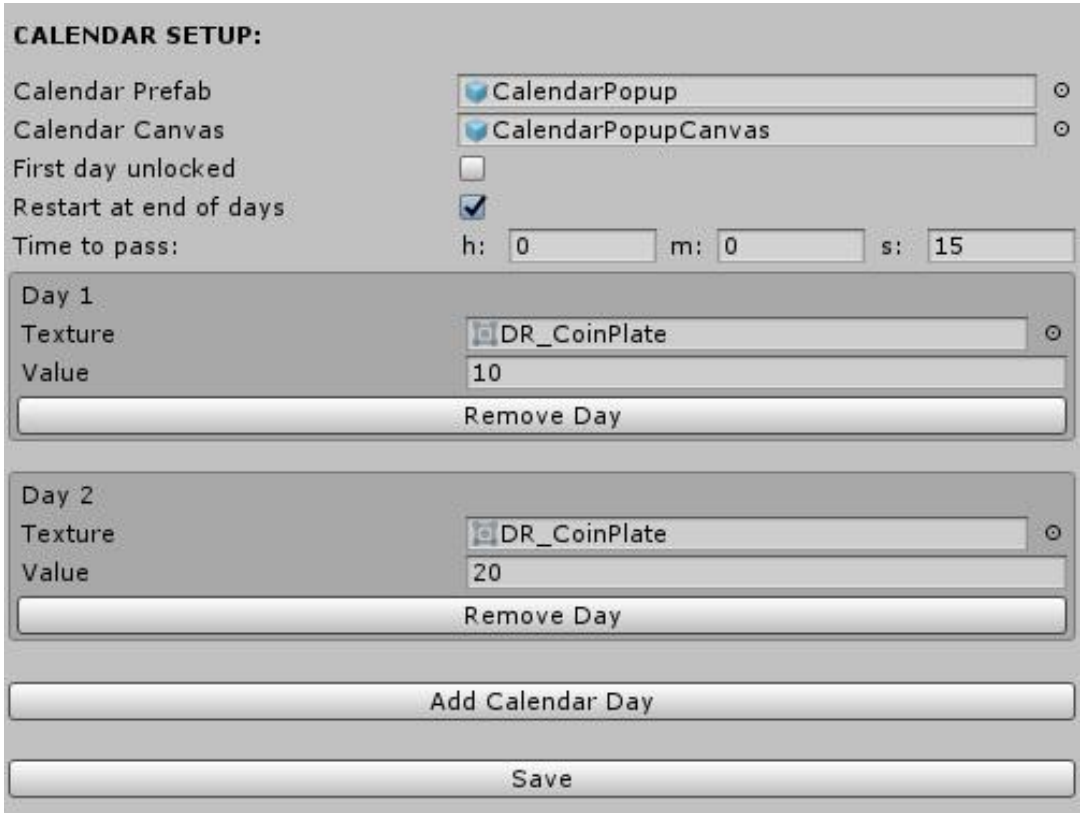
8. EXAMPLE 2 - Calendar

You can find the example test scene here:
Assets/GleyPlugins/DailyRewards/Example/CalendarExample.unity

8. EXAMPLE 2 - Calendar

Calendar customization:

1. **Create your rewards in settings window**

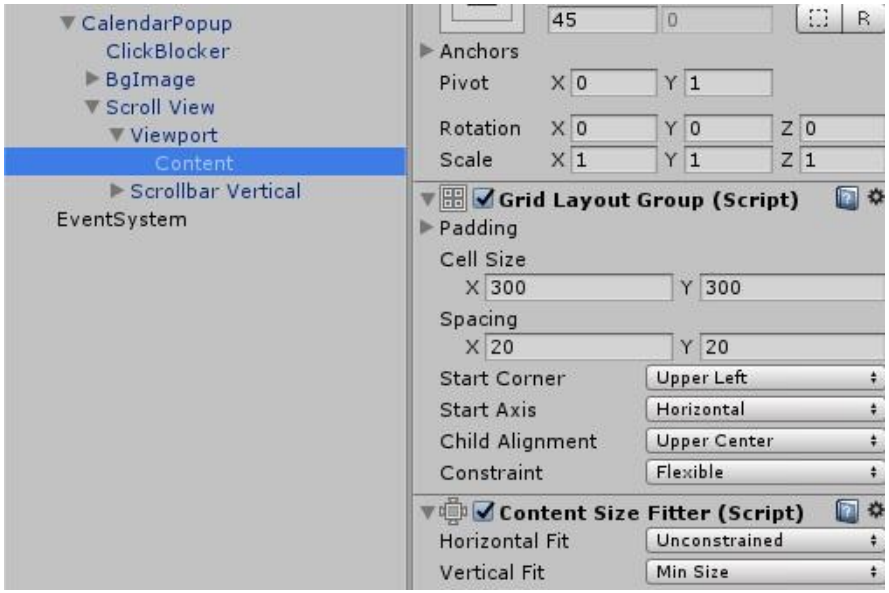


8. EXAMPLE 2 - Calendar

Calendar customization:

2. Modify day size:

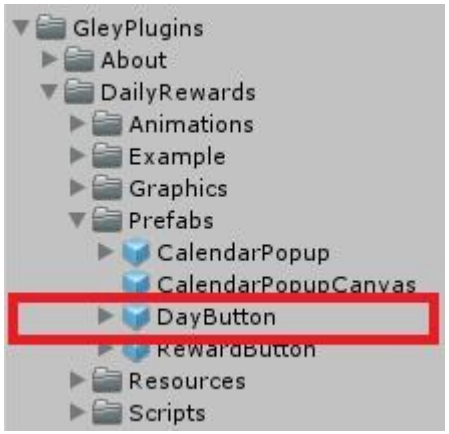
- Go to CalenderPopup prefab -> Scroll View -> Content and change the cell size and spacing



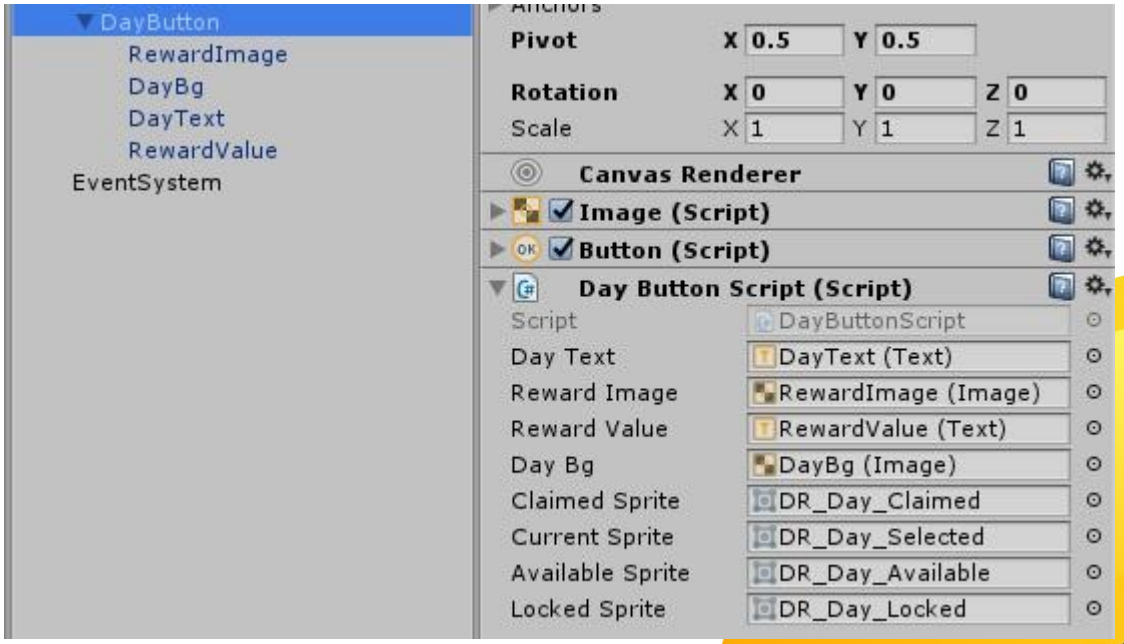
8. EXAMPLE 2 - Calendar

Calendar customization:

2. **Modify day button appearance:**
    - Open prefab from:



- Change the appearance
- Sprites are change with the state of the button as follows:
- **Claimed sprite** - active for past days
- **Current sprite** - active for current day but timer is active(cannot be claimed)
- **Available sprite** - active for current day (can be claimed)
- **Locked Sprite** - active for future days




gLeY