Documentation-Jan: Input System

Input System:

General:

1. Unitys Input System is used for processing genereal Game Input
2. A Scriptable Object calles ‚InputReaderSO.cs‘ was created which consists of callback Methods that are called on specific Hardware Inputs (Keyboard, Mouse, Gamepad etc.) (compare Fig. 1-5)
3. This InputReaderSO.cs (compare Fig. 7) then fires Events, on every received Input, that are subscribed to in specific Classes like ‚PlayerController.cs‘ or ‚PlayerWeaponHandling.cs’ e.g. for handling the Player Movement or WeaponHandling etc.

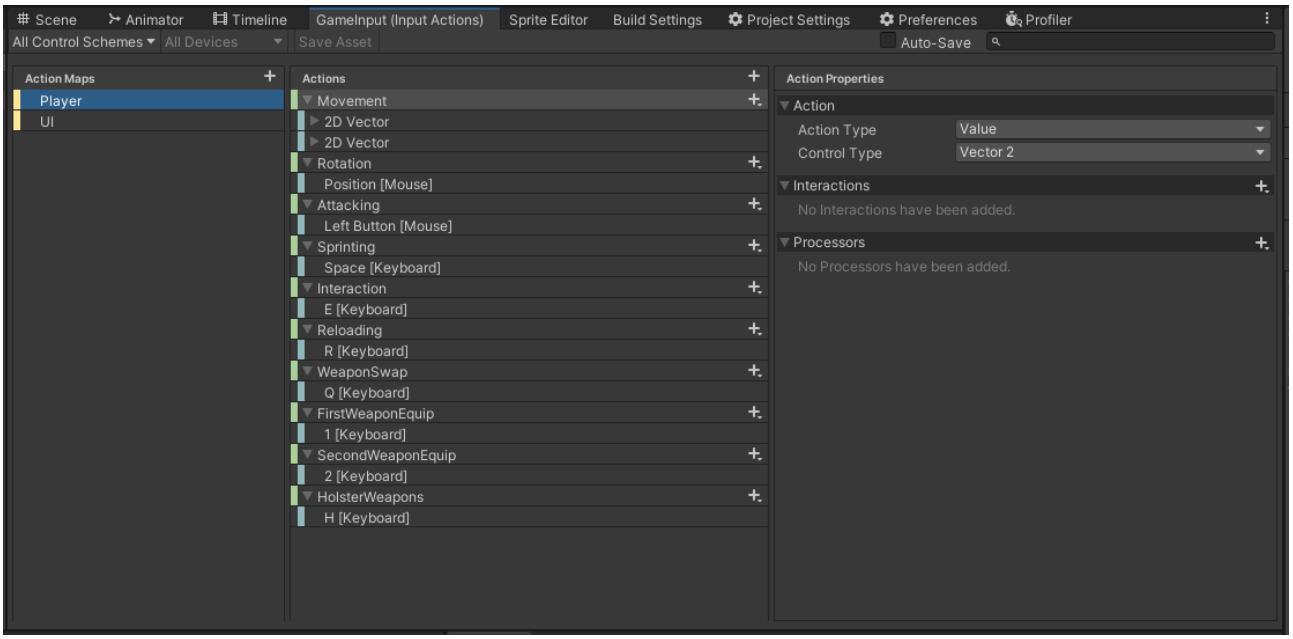

Fig. 1: Setup of the Input Actions for Player related Input

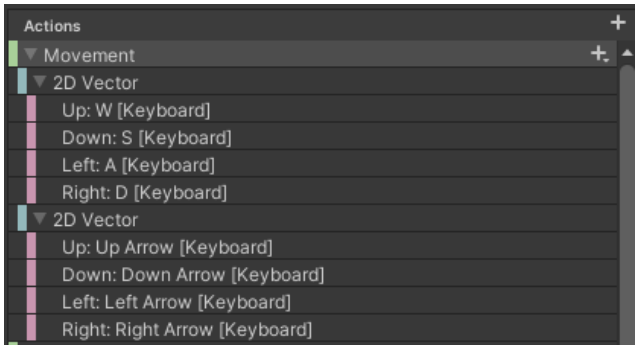

Fig. 2: Player Movement Related Input Action Setup

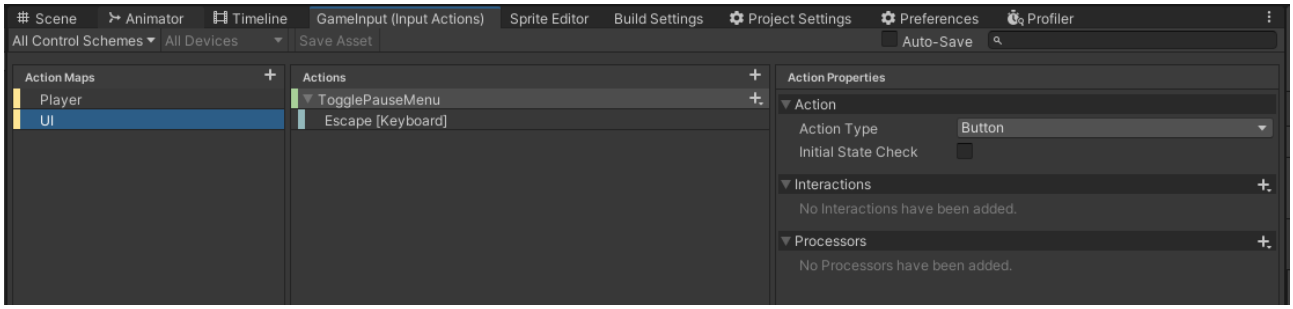

Fig. 3: currently UI Related Setup of the Input Actions

Fig. 4: According to Fig 1-3 the appropriate Input Action Map is located in the Project Folder under 'GameInput'. As you can see there is also a autogenerated C#-Class (called ‚GameInput.cs‘) of the current Input Actions so they can be accessed via code.

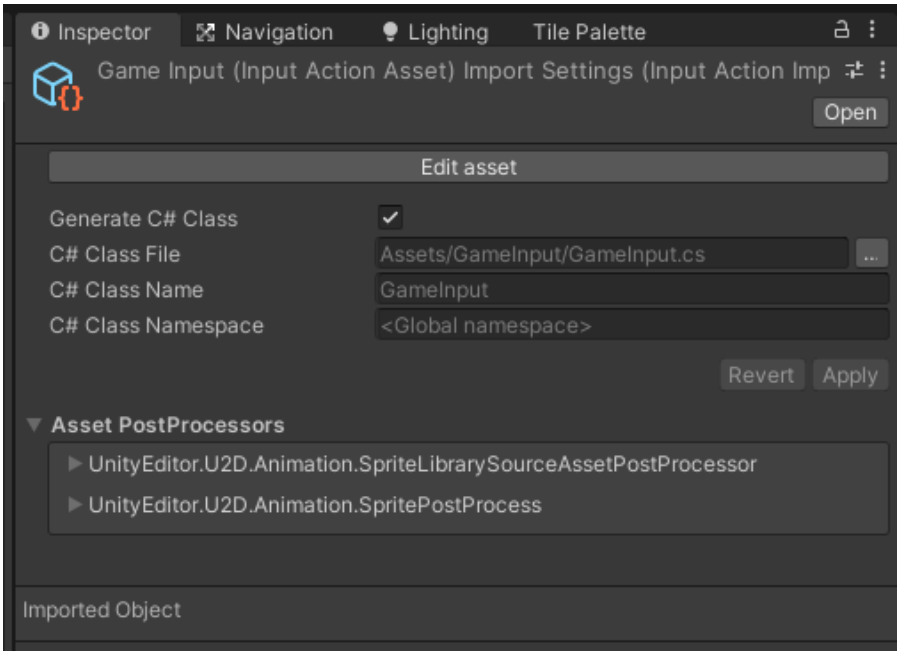

Fig. 5: Inspector Setup of the 'GameInput ActionMap'. Note that the Property to generate an according C#-Class is (and should be) checked.

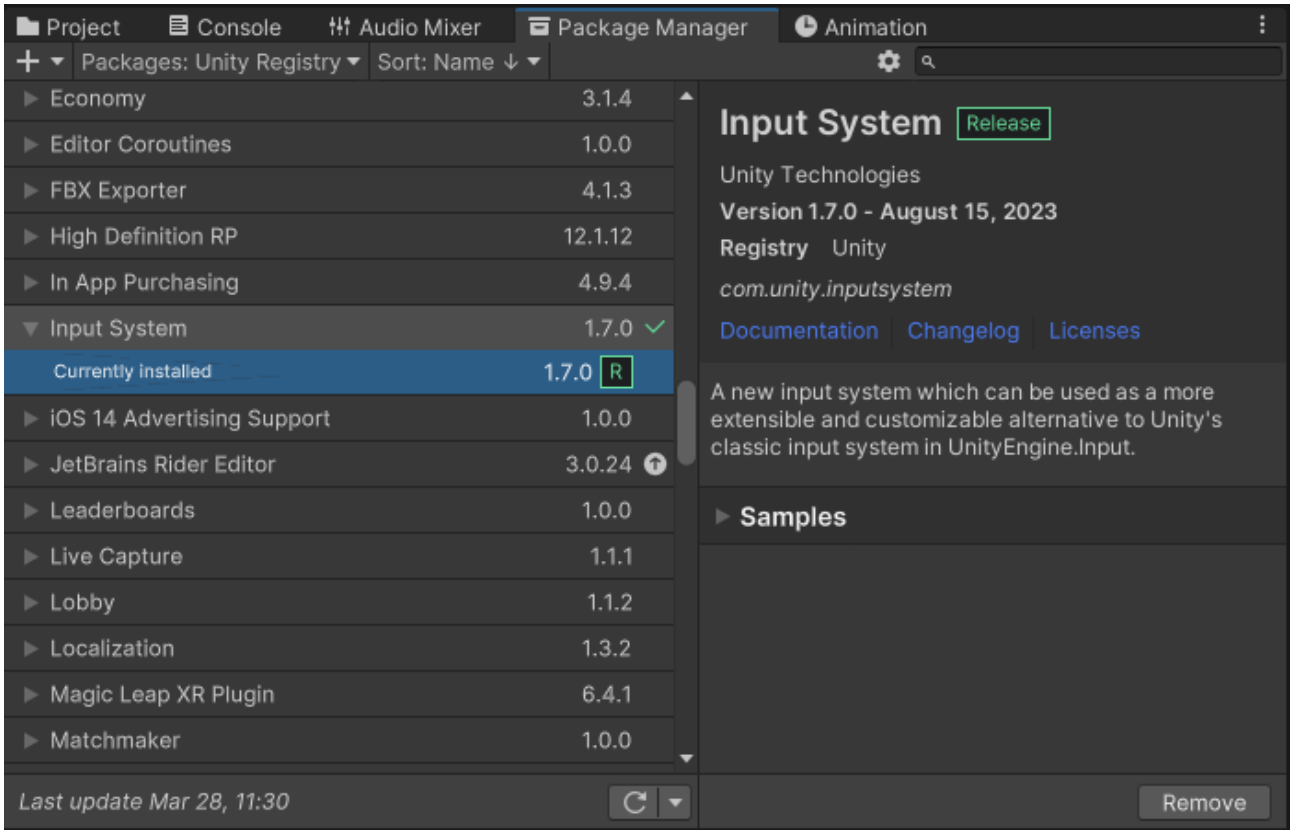

Fig. 6: Of Course the actual Input System Package needs to be installed for beeing able to use it.

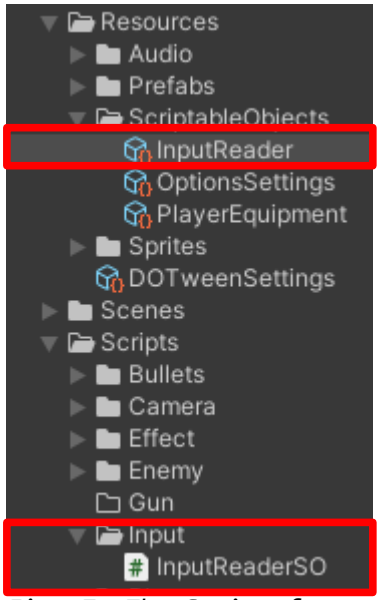

Fig. 7: The Scripz for the created ScriptableObject 'InputReaderSO.cs' can be found in the Project Folder under Scripts/Input. However, the according created ScriptableObject-Asset can be found in the Project Folder under Resources/ScriptableObjects/InputReader.

The InputReaderSO.cs

- Currently implements two Interfaces of the GameInput.cs:
  - ‚IPlayerActions‘
  - ‚IUIActions‘
  - compare Fig. 8 and 9

How does it acutally work?

- When a Key input is registered specified in the Action Map of the Input System The specific Callback-Method of ‚InputReaterSO.cs’ will be executed and the specific Event defined there will be fired.
- The specific Eventlistener will then execute a specific Logic.
- Compare Fig. 10

- Example:
  - The ‚Esc‘ – Key is pressed
    - → the ‚InputReaderSO.cs’ will execute the specific Method, fireses the ‚OnEscPress‘-Event
    - → The ‚PauseMenu.cs‘ listens to this specific Event and will execute the ‚TogglePauseMenu()‘ as soon as the Event ‚OnEscPress‘ was fired by the ‚InputReaderSO.cs’

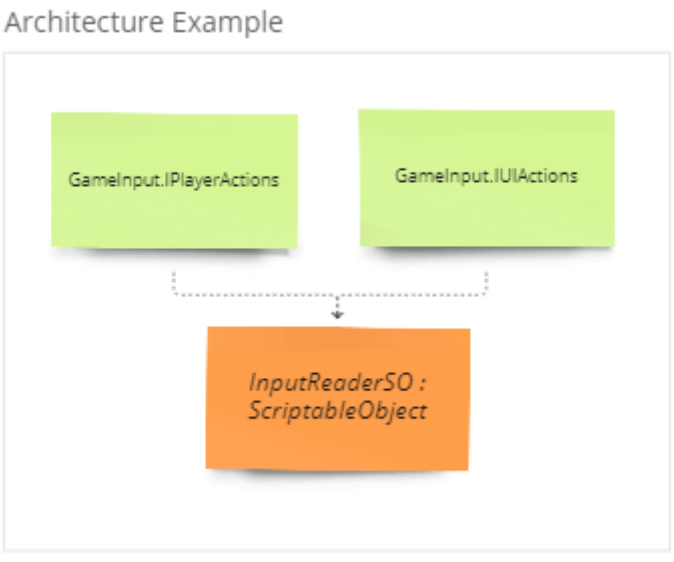

Fig. 8: The 'InputReaderSo'.cs implements the InterFaces 'IPlayerActions' and 'IUIActions' of the 'GameInput.cs'

InputReaderSO.cs

Fig. 9: Basic Setup of the 'InputReaderSO.cs'

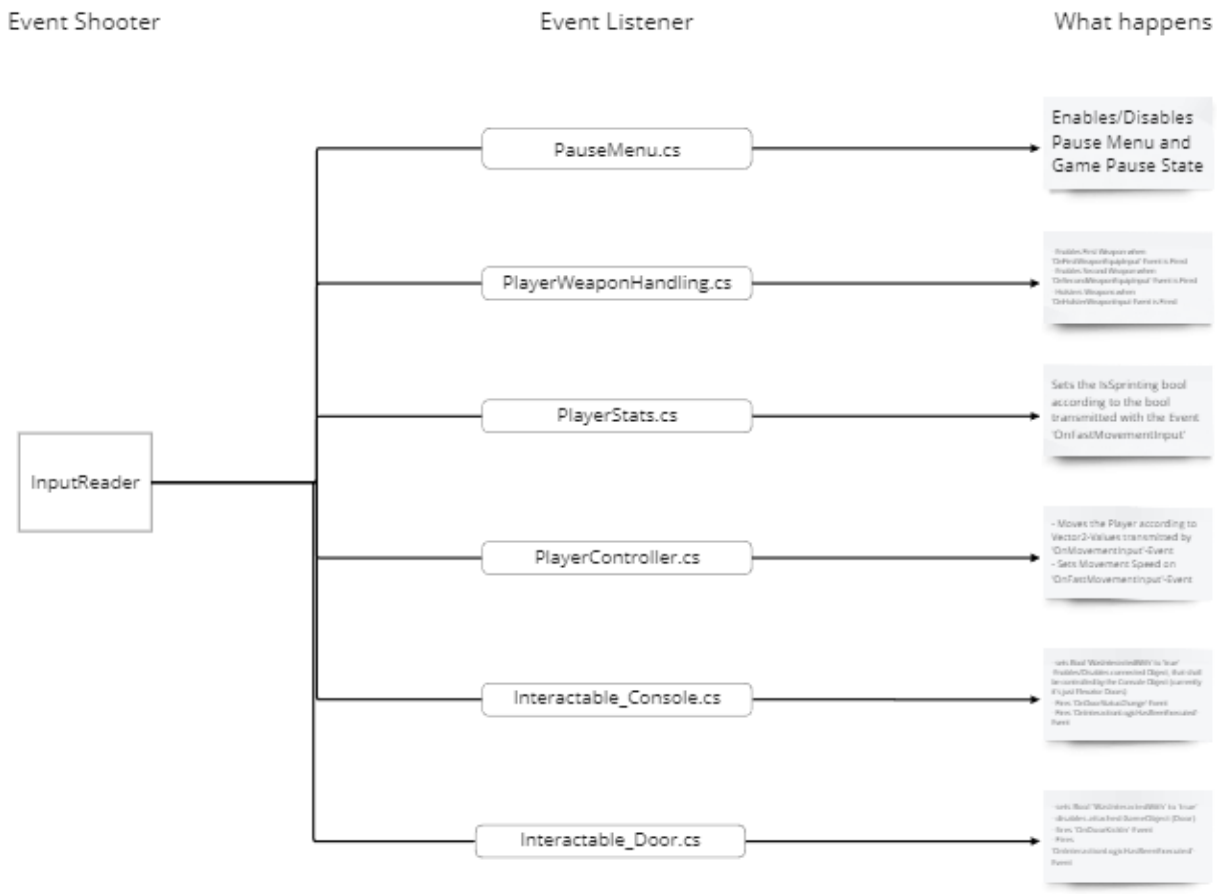

Fig. 10: Schematic Event Flow of the 'InputReaderSo.cs' it listeners and a description what basicaly will happen when a specific event was fired.