



## Downloadable package: Input System integration

Thanks for downloading the Input System integration package for Adventure Creator. [Input System](#) is the name of Unity's new input handler, and can be used to easily define and modify input bindings in a dedicated asset.

This package provides an Input Action asset named **Controls**, which includes some default actions for AC's most common inputs – including those for movement, interactions and UI. This asset can be modified and updated to suit your needs.

### Instructions

1. Add the provided **ControlsReader** prefab into your scene.
2. Configure the **Controls** asset to suit your input needs. It comes provided with a standard set of inputs and device schemes, but all can be modified as necessary.
3. For any new input you wish to detect, ensure the name entered into the Controls asset matches that used by AC.
4. The Controls asset has three Control Schemes defined: Gamepad, Mouse and keyboard, and Touchscreen. The Controls Reader Inspector's **Auto Sync Input Method** option allows it to switch the Settings Manager's Input method dynamically based on the last-pressed input. For further control over what happens when the control scheme is changed, hook a custom script into the Controls Reader's **On Set Control Scheme** event, which passes the currently-used control scheme's name as a parameter.

## Remapping inputs

With Input System, it is possible to allow the user to remap bindings at runtime to suit their preference. This integration provides a new Inputs menu that can be used to display and remap inputs.

To use it, set your Menu Manager to the provided **InputSystem\_MenuManager** asset file, and copy/paste its **Inputs** menu to your own Menu Manager. To view it at runtime, use the **Menu: Change state** Action to turn it on.

By default the Menu will allow for the remapping of four inputs: InteractionA, InteractionB, Run and FlashHotspots. To change these or add more inputs, open the InputsUI prefab and modify the **Input Remapper** component's **Remappable Actions** array – assigning each element an input reference as well as a UI Button inside its Content child object in the Hierarchy.

In order to save changes bindings, create a new Global String variable in the Variables Manager, give it a unique name (e.g. "InputRebindings") and set its **Link to** property to **Options Data**. Then, locate the ControlsReader prefab and set its **Global String Variable** field to match the name of the new variable.