



# Eadon Survival

A Survival Add-on for Eadon Character Controller

Version 1.2

## Table of Contents

<b>Introduction.....</b>	<b>3</b>
Changelog.....	4
Credits.....	4
<b>Prerequisites.....</b>	<b>5</b>
<b>Configuration.....</b>	<b>6</b>
Eadon Survival Data .....	6
Sickness.....	7
Sickness Cure .....	8
Injury.....	9
<b>Character Creation.....</b>	<b>11</b>
Eadon Survival Manager.....	11
<b>Changes to Consumable Items .....</b>	<b>17</b>
<b>Hunger and Thirst .....</b>	<b>18</b>
<b>Temperature.....</b>	<b>19</b>
<b>Sicknesses .....</b>	<b>20</b>
Infected NPCs .....	20
Infected Areas .....	20
Sickness Cures.....	20
<b>Injuries.....</b>	<b>21</b>
<b>Managing Sicknesses .....</b>	<b>22</b>
<b>Bonfire system.....</b>	<b>23</b>
Normal Bonfires.....	23
Healing Bonfires.....	25
Resting Bonfires.....	26
Resting Area .....	28
Customizing a bonfire .....	29
<b>Bonus Content .....</b>	<b>31</b>
<b>License.....</b>	<b>33</b>

## Introduction

Eadon Survival is an add-on for Eadon Character Controller asset for the Unity game engine. The goal of this asset is to provide a framework for implementing survival games in Unity. The asset does not make any assumptions on the type of survival games, allowing for a wide range of game styles.

The following functionalities are implemented:

- Hunger
- Thirst
- Temperature handling (via Enviro integration)
- Night and day management (via Enviro integration) with different hunger/thirst profiles
- Sickesses
- Specific cures for different sicknesses
- Injuries (from critical attacks or being damaged when below a threshold)
- Custom debilitating effects for sicknesses
- Custom debilitating effects for injuries, including preventing equipping items on broken arms
- Bonfire system

This add-on is currently compatible with Eadon Character Controller version 1.2 and higher and tested on Unity 2019.4 and higher.

## Changelog

V 1.0	Initial release
V 1.1	Enviro 3 support
V 1.2	Updated for Eadon Character Controller v1.5

## Credits

Some models used for the bonfire and the stick in the demo scene are free models licensed under Creative Common Attribution license. Please respect the copyright if you use them in your game and display proper attribution:

Bonfire	"Bonfire-Lowpoly" ( <a href="https://skfb.ly/6yvng">https://skfb.ly/6yvng</a> ) by Christian Gentry is licensed under Creative Commons Attribution ( <a href="http://creativecommons.org/licenses/by/4.0/">http://creativecommons.org/licenses/by/4.0/</a> ).
Stick	"Wooden Stick" ( <a href="https://skfb.ly/6RsL6">https://skfb.ly/6RsL6</a> ) by Thunder is licensed under Creative Commons Attribution ( <a href="http://creativecommons.org/licenses/by/4.0/">http://creativecommons.org/licenses/by/4.0/</a> ).
Bandage	"Bandage" ( <a href="https://skfb.ly/onwMn">https://skfb.ly/onwMn</a> ) by Alex Birdman is licensed under Creative Commons Attribution ( <a href="http://creativecommons.org/licenses/by/4.0/">http://creativecommons.org/licenses/by/4.0/</a> ).
Potion Bottle	"Potion Bottle" ( <a href="https://skfb.ly/6VYZt">https://skfb.ly/6VYZt</a> ) by Jeremy_W is licensed under Creative Commons Attribution ( <a href="http://creativecommons.org/licenses/by/4.0/">http://creativecommons.org/licenses/by/4.0/</a> ).
Bread	"(Game Ready Free) Bread Loaf / Буханка хлеба" ( <a href="https://skfb.ly/6UARz">https://skfb.ly/6UARz</a> ) by Igrodev CG licensed under Creative Commons Attribution ( <a href="http://creativecommons.org/licenses/by/4.0/">http://creativecommons.org/licenses/by/4.0/</a> ).
Chest	"Chest" ( <a href="https://skfb.ly/66CvU">https://skfb.ly/66CvU</a> ) by Aki_Kato is licensed under Creative Commons Attribution ( <a href="http://creativecommons.org/licenses/by/4.0/">http://creativecommons.org/licenses/by/4.0/</a> ).

## Prerequisites

Eadon Survival contains support for the following package:

- Hendrik Haupt's Enviro - Sky and Weather  
(<https://assetstore.unity.com/packages/tools/particles-effects/enviro-sky-and-weather-33963>)

This asset is used for temperature effect, night and day management and managing effects that happen every game hour or game day.

The use of Enviro is completely optional and support is enabled automatically if Enviro is installed in your project.

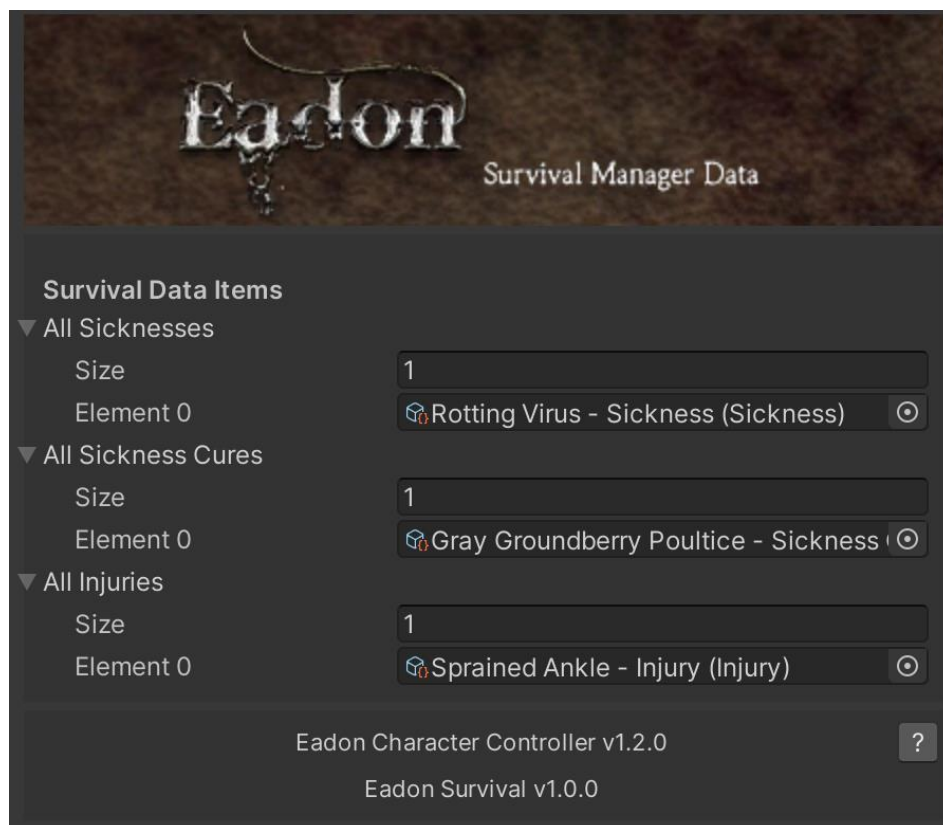
## Configuration

Like every other addon, Eadon Survival needs to be enabled in the Eadon Controller Setting Window:

Before Eadon Survival can be used, it needs to be configured. You need to create all the sicknesses, cures, injuries and remedies you plan to use. This is an optional step, in the sense that if you only want to use hunger and thirst (and temperature if using Enviro) you can skip this step.

### Eadon Survival Data

You need one instance of this ScriptableObject in your project. Its role is to collect all the elements available in your game. To create one, right click in your project window and select Create/Eadon Survival/New Survival Manager Data. A new Eadon Survival Manager Data object will be created in that location. It looks like this:



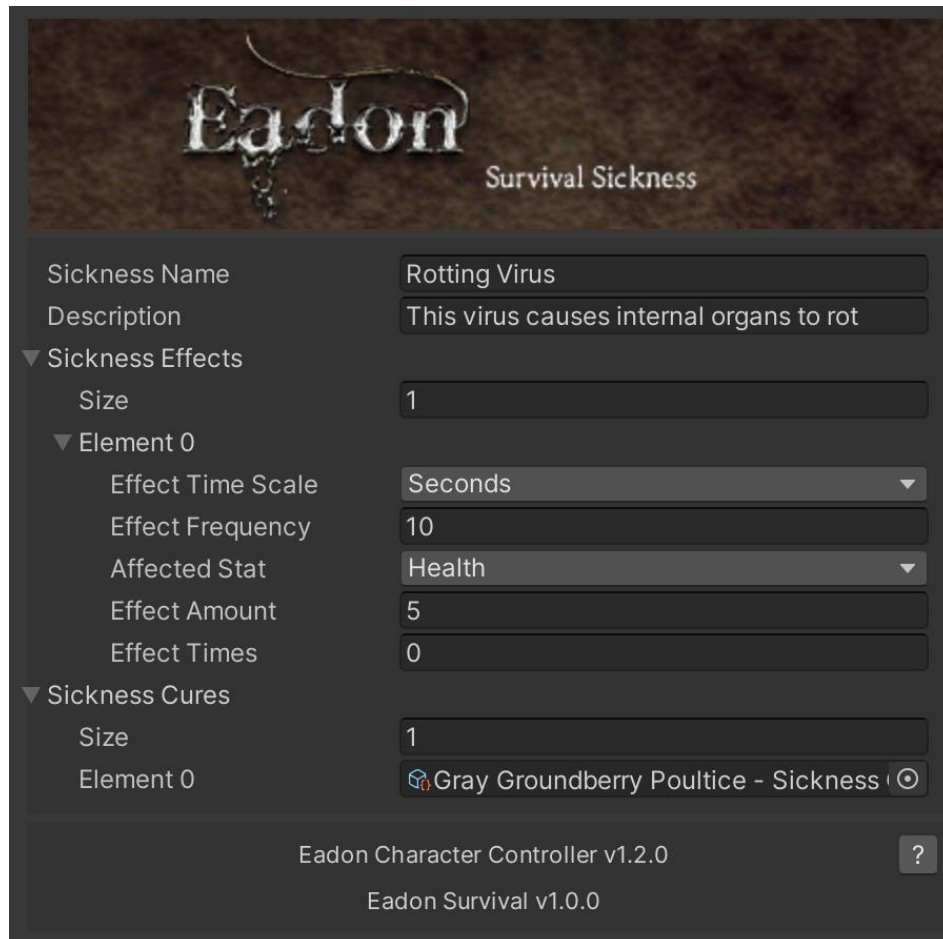
The fields are as follows:

Field	Use
All Sicknesses	A list of all the <a href="#">Sickness</a> ScriptableObjects
All Sickness Cures	A list of all the <a href="#">SicknessCure</a> ScriptableObjects
All Injuries	A list of all the <a href="#">Injury</a> ScriptableObjects

When you create a new element, you need to add it to the relevant section.

## Sickness

A **Sickness** ScriptableObject contains the definition of a sickness or infection available in game. To create one, right click in your project window and select **Create/Eadon Survival/New Sickness**. A new Eadon Survival Sickness object will be created in that location. It looks like this:



The screenshot shows a configuration window for a 'Survival Sickness' in the 'Eadon' game. The window has a dark theme with a title bar at the top. Below the title bar, there are several sections for configuring the sickness. The 'Sickness Name' is 'Rotting Virus' and the 'Description' is 'This virus causes internal organs to rot'. The 'Sickness Effects' section is expanded, showing a 'Size' of 1 and an 'Element 0' with 'Effect Time Scale' set to 'Seconds', 'Effect Frequency' of 10, 'Affected Stat' set to 'Health', 'Effect Amount' of 5, and 'Effect Times' of 0. The 'Sickness Cures' section is also expanded, showing a 'Size' of 1 and an 'Element 0' set to 'Gray Groundberry Poultice - Sickness'. At the bottom, there are version numbers for 'Eadon Character Controller v1.2.0' and 'Eadon Survival v1.0.0'.

Field	Value
Sickness Name	Rotting Virus
Description	This virus causes internal organs to rot
Sickness Effects	
Size	1
Element 0	
Effect Time Scale	Seconds
Effect Frequency	10
Affected Stat	Health
Effect Amount	5
Effect Times	0
Sickness Cures	
Size	1
Element 0	Gray Groundberry Poultice - Sickness

The fields are as follows:

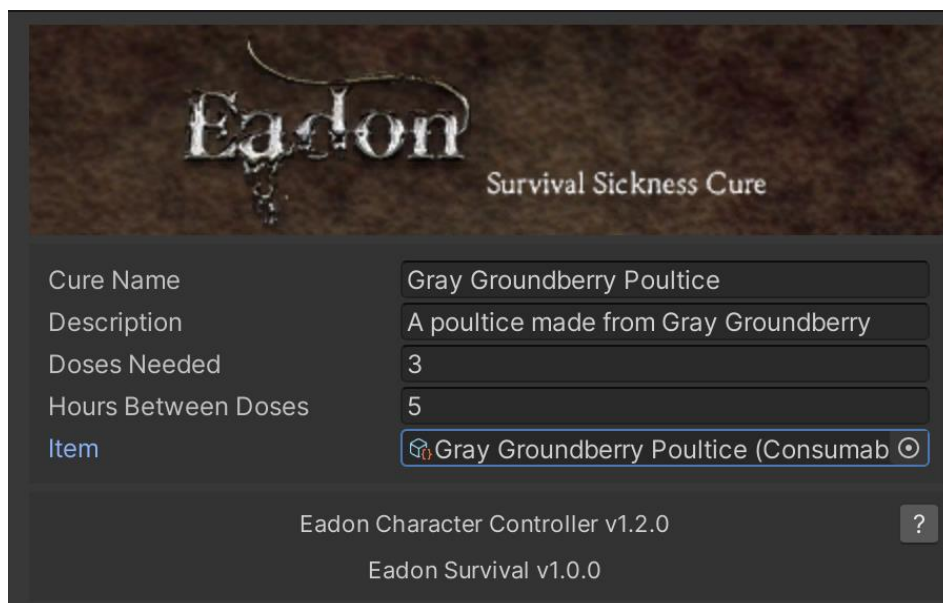
Field	Use
Sickness Name	The name of the sickness
Description	The description of the sickness
Sickness ID	The id of the sickness (needs to be unique across all sicknesses)
Sickness Effects	A list of all the effects caused by this sickness (see below for a description of the fields)
Sickness Cures	A list of all the cures that need to be applied to get rid of the sickness

The sickness effects fields are as follows:

Field	Use
Effect Time Scale	The time scale of the effect, choice of seconds and minutes (real time). If Enviro is installed in the project, game hour and game day are also available
Effect Frequency	How often (based on the time scale) the effect takes place
Affected Stat	Which stat is affected, choice of Health, Stamina, Hunger or Thirst
Effect Amount	The amount of the affected stat lost every time the effect takes place
Effect Times	How many times the effect takes place (if you leave it at 0 or less, it will happen until cured)

### Sickness Cure

A [SicknessCure](#) ScriptableObject contains the definition of a sickness or infection cure available in game. To create one, right click in your project window and select **Create/Eadon Survival/New Sickness**. A new Eadon Survival Sickness object will be created in that location. It looks like this:



The fields are as follows:

Field	Use
Cure Name	The name of the cure
Description	The description of the cure
Cure ID	The id of the cure (needs to be unique across all cures)
Doses Needed	The number of doses needed



<b>Hours Between Doses</b>	The time interval between doses
<b>Item</b>	The consumable item for this cure

## Injury

A **Injury** ScriptableObject contains the definition of an injury available in game. To create one, right click in your project window and select **Create/Eadon Survival/New Sickness**. A new Eadon Survival Sickness object will be created in that location. It looks like this:

The screenshot shows a configuration window for an injury. The fields are as follows:

- Injury Name:** Sprained Ankle
- Description:** A sprained ankle
- Speed Reduction:** 0.5
- Health Reduction:** 0.1
- Stamina Reduction:** 0.2
- Damage Output Reduction:** 0
- Prevents Right Hand Items:** ☐
- Prevents Left Hand Items:** ☐
- Injury Remedies:**
  - Size:** 1
  - Element 0:** Bandage (Consumable)

At the bottom, it shows 'Eadon Character Controller v1.2.0' and 'Eadon Survival v1.0.0'.

The fields are as follows:

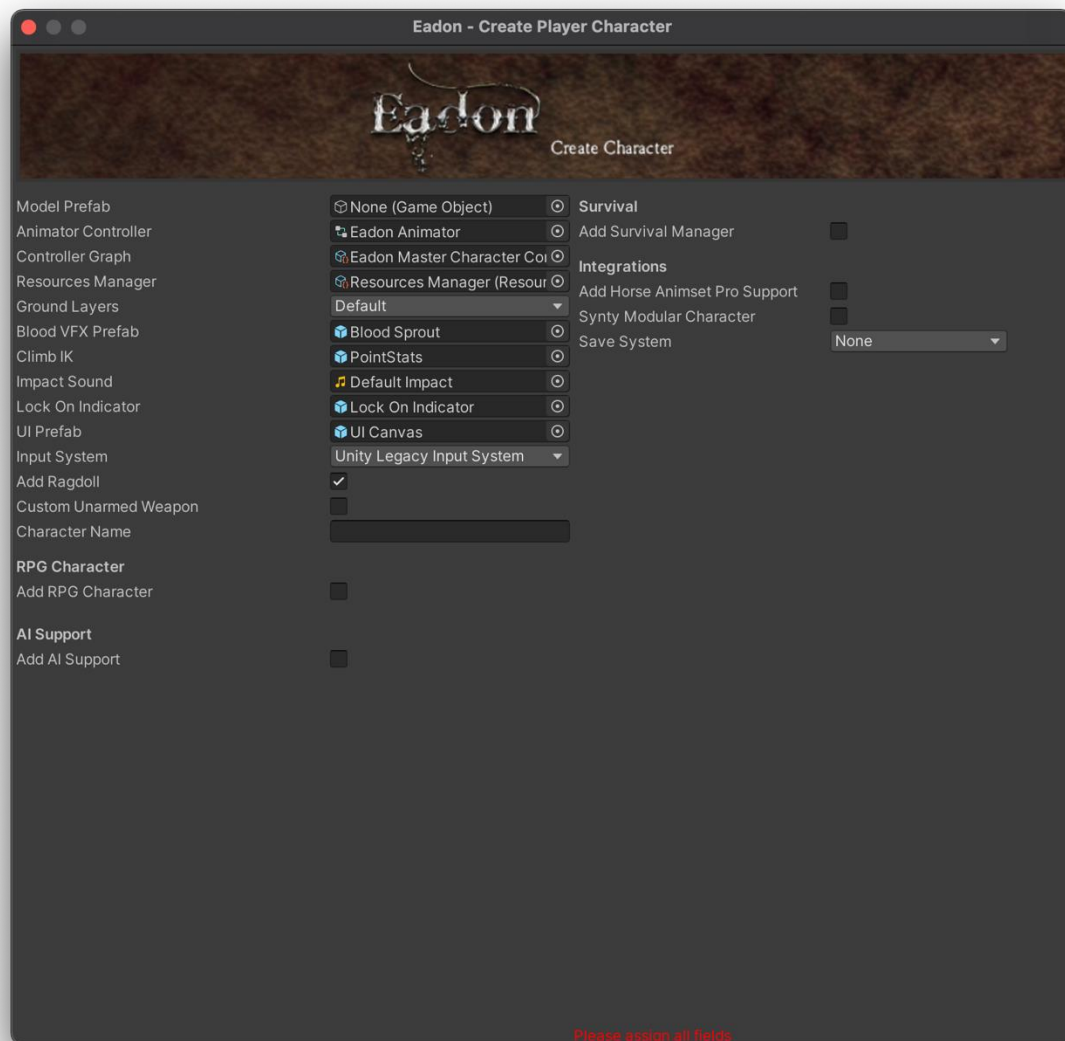
Field	Use
<b>Injury Name</b>	The name of the injury
<b>Description</b>	The description of the injury
<b>Speed Reduction</b>	How much the speed of the character is reduced while injured (range of 0 to 1, 0 means no reduction, 1 means total reduction)
<b>Health Reduction</b>	How much the health of the character is reduced while injured (range of 0 to 1, 0 means no reduction, 1 means total reduction)
<b>Stamina Reduction</b>	How much the stamina of the character is reduced while injured (range of 0 to 1, 0

	means no reduction, 1 means total reduction)
<b>Damage Output Reduction</b>	How much the damage output of the character is reduced while injured (range of 0 to 1, 0 means no reduction, 1 means total reduction)
<b>Prevents Right Hand Items</b>	A flag to prevent equipping items on the right hand
<b>Prevents Left Hand Items</b>	A flag to prevent equipping items on the left hand
<b>Injury Remedies</b>	A list of all the items that can remedy this injury

Speed reduction has no effect on the character if root motion for locomotion is enabled. In that case, it's advised to add a root motion enabled injured walk animation in the Injured tab of the [EadonController](#) inspector.

## Character Creation

Once the addon is imported in your project, you have the option to add the survival elements to your character at creation (top right corner):



This will add an [EadonSurvivalManager](#) component to the character. If you have characters already created to which you want to add support for Eadon Survival, select them in the hierarchy and use the **Eadon/Create/Add Eadon Survival Integration** menu command.

## Eadon Survival Manager

This is the main component for the survival system. It defines the behaviour of the system for the character and tracks what happens to the character.



This is character specific, because you could have different characters in your game with different characteristics, such as an alien from an icy world being adapted to cold temperature. It looks like this:

# Eadon

## Survival Manager

### Data

#### Data

 Demo Eadon Survival Manager 

#### Hunger

Use Hunger	<input checked="" type="checkbox"/>
Max Hunger	<input type="text" value="0"/>
Hunger Level	<input type="text" value="0"/>
Refill Hunger On Start	<input type="checkbox"/>
Hunger Timing	<input type="text" value="Real Seconds"/>
Hunger Interval	<input type="text" value="0"/>
Hunger Loss	<input type="text" value="0"/>
Hunger Loss Night Reduction	<input type="range" value="0.5"/>
Hunger Stamina Loss Threshold	<input type="text" value="0"/>
Hunger Stamina Loss	<input type="text" value="0"/>
Recover Stamina When Hungry	<input type="checkbox"/>
Hunger Health Loss Threshold	<input type="text" value="0"/>
Hunger Health Loss	<input type="text" value="0"/>
Recover Health When Hungry	<input type="checkbox"/>

#### Thirst

Use Thirst	<input checked="" type="checkbox"/>
Max Thirst	<input type="text" value="0"/>
Thirst Level	<input type="text" value="0"/>
Refill Thirst On Start	<input type="checkbox"/>
Thirst Timing	<input type="text" value="Real Seconds"/>
Thirst Interval	<input type="text" value="0"/>
Thirst Loss	<input type="text" value="0"/>
Thirst Loss Night Reduction	<input type="range" value="0.5"/>
Thirst Stamina Loss Threshold	<input type="text" value="0"/>
Thirst Stamina Loss	<input type="text" value="0"/>
Recover Stamina When Thirsty	<input type="checkbox"/>
Thirst Health Loss Threshold	<input type="text" value="0"/>
Thirst Health Loss	<input type="text" value="0"/>
Recover Health When Thirsty	<input type="checkbox"/>

#### Temperature

Use Temperature	<input type="checkbox"/>
Temperature Check Interval	<input type="text" value="5"/>
Cold Stamina Threshold	<input type="text" value="0"/>
Stamina Loss Per Interval Cold	<input type="text" value="0"/>
Cold Health Threshold	<input type="text" value="0"/>
Health Loss Per Interval Cold	<input type="text" value="0"/>
Hot Stamina Threshold	<input type="text" value="0"/>
Stamina Loss Per Interval Hot	<input type="text" value="0"/>
Hot Health Threshold	<input type="text" value="0"/>
Health Loss Per Interval Hot	<input type="text" value="0"/>

## Sicknesses

► Current Sicknesses

► Pending Cures

## Injuries

Use Injuries

☐

Single Attack Damage Thresh

25

Single Attack Injury Chance

0.2

Low Health Threshold

10

Low Health Injury Chance

0.1

► Current Injuries

## Events

On Infected (String)

List is Empty

+ -

On Cured (String)

List is Empty

+ -

On Injured (String)

List is Empty

+ -

On Injury Remedied (String)

List is Empty

+ -

On Hunger Stamina Loss ()

List is Empty

+ -

On Hunger Health Loss ()

List is Empty

+ -

On Hunger Recovered ()

List is Empty

+ -

On Thirst Stamina Loss ()

List is Empty

+ -

On Thirst Health Loss ()

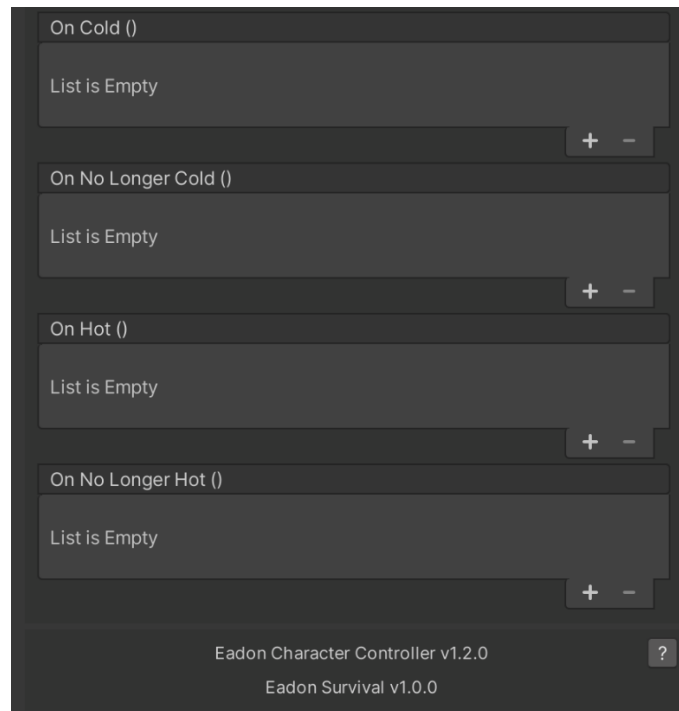
List is Empty

+ -

On Thirst Recovered ()

List is Empty

+ -



The fields are as follows:

Field	Use
<b>Data</b>	The SurvivalManagerData object with the data for the game
<b>Use Hunger</b>	A flag to indicate if the hunger subsystem is in use
<b>Max Hunger</b>	The max hunger resistance level
<b>Hunger Level</b>	The current hunger resistance level
<b>Refill Hunger On Start</b>	A flag to indicate if the current level is set to the max level on start
<b>Hunger Timing</b>	The time scale of hunger loss (choice of Real Seconds or Enviro Hours, if Enviro is installed)
<b>Hunger Interval</b>	How often, based on the previous time scale, hunger loss happens
<b>Hunger Loss</b>	How much hunger is lost
<b>Hunger Loss Night Reduction</b>	How much is loss reduced at night (if Enviro is installed), 0 means no reduction and 1 is total reduction
<b>Hunger Stamina Loss Threshold</b>	The hunger threshold below which stamina loss happens
<b>Hunger Stamina Loss</b>	How much stamina is lost
<b>Recover Stamina When Hungry</b>	A flag to indicate if stamina recovery happens while below the threshold
<b>Hunger Health Loss Threshold</b>	The hunger threshold below which health loss happens
<b>Hunger Health Loss</b>	How much health is lost

<b>Recover Health When Hungry</b>	A flag to indicate if health recovery happens while below the threshold
<b>Use Thirst</b>	A flag to indicate if the thirst subsystem is in use
<b>Max Thirst</b>	The max thirst resistance level
<b>Hunger Level</b>	The current thirst resistance level
<b>Refill Thirst On Start</b>	A flag to indicate if the current level is set to the max level on start
<b>Thirst Timing</b>	The time scale of thirst loss (choice of Real Seconds or Enviro Hours, if Enviro is installed)
<b>Thirst Interval</b>	How often, based on the previous time scale, thirst loss happens
<b>Thirst Loss</b>	How much thirst is lost
<b>Thirst Loss Night Reduction</b>	How much is loss reduced at night (if Enviro is installed), 0 means no reduction and 1 is total reduction
<b>Thirst Stamina Loss Threshold</b>	The thirst threshold below which stamina loss happens
<b>Thirst Stamina Loss</b>	How much stamina is lost
<b>Recover Stamina When Thirsty</b>	A flag to indicate if stamina recovery happens while below the threshold
<b>Hunger Thirst Loss Threshold</b>	The thirst threshold below which health loss happens
<b>Hunger Thirst Loss</b>	How much health is lost
<b>Recover Health When Thirsty</b>	A flag to indicate if health recovery happens while below the threshold
<b>Use Temperature</b>	A flag to indicate if the temperature subsystem is in use (only available if Enviro is installed)
<b>Temperature Check Interval</b>	The interval in seconds between temperature checks
<b>Cold Stamina Threshold</b>	The cold temperature stamina threshold
<b>Stamina Loss Per Interval Cold</b>	How much stamina is lost per interval while below the threshold
<b>Cold Health Threshold</b>	The cold temperature health threshold
<b>Health Loss Per Interval Cold</b>	How much health is lost per interval while below the threshold
<b>Hot Stamina Threshold</b>	The hot temperature stamina threshold
<b>Stamina Loss Per Interval Hot</b>	How much stamina is lost per interval while above the threshold
<b>Hot Health Threshold</b>	The hot temperature health threshold
<b>Health Loss Per Interval Hot</b>	How much health is lost per interval while above the threshold
<b>Current Sicknesses</b>	The list of sicknesses currently affecting the character
<b>Pending Cures</b>	The list of cures the character is taking


<b>Use Injuries</b>	A flag to indicate if the injuries subsystem is in use
<b>Single Attack Damage Threshold</b>	The damage threshold above which a single attack can cause an injury
<b>Single Attack Injury Chance</b>	The chance of a single attack causing an injury with high damage
<b>Low Health Threshold</b>	The health threshold below which an attack can cause an injury
<b>Low Health Injury Chance</b>	The chance of an attack causing an injury when at low health
<b>Current Injuries</b>	The list of injuries the character is currently affected by
<b>On Infected</b>	An event triggered when the character is infected
<b>On Cured</b>	An event triggered when the player has cured an infection or sickness
<b>On Injured</b>	An event triggered when the character has sustained an injury
<b>On Injury Remedied</b>	An event triggered when the player has remedied an injury
<b>On Hunger Stamina Loss</b>	An event triggered when the player loses stamina due to hunger
<b>On Hunger Health Loss</b>	An event triggered when the player loses health due to hunger
<b>On Hunger Recovered</b>	An event triggered when the player recovers hunger
<b>On Thirst Stamina Loss</b>	An event triggered when the player loses stamina due to thirst
<b>On Thirst Health Loss</b>	An event triggered when the player loses health due to thirst
<b>On Thirst Recovered</b>	An event triggered when the player recovers thirst
<b>On Cold</b>	An event triggered when the player is cold
<b>On No Longer Cold</b>	An event triggered when the player is no longer cold
<b>On Hot</b>	An event triggered when the player is hot
<b>On No Longer Hot</b>	An event triggered when the player is no longer hot

For more information, see the following chapters on the various subsystems.



## Changes to Consumable Items

Once Eadon Survival is enabled, five new fields appear in the inspector for [ConsumableItem](#) scriptable objects:

Survival Effects	
Hunger Recovery	<input type="text" value="0"/>
Hunger Grace	<input type="text" value="0"/>
Thirst Recovery	<input type="text" value="0"/>
Thirst Grace	<input type="text" value="0"/>
Empty Container	<input type="text" value="None (Item)"/> 

The fields are:

Attribute	Use
Hunger	Amount of hunger recovered when the item is consumed
HungerGrace	Grace period, in seconds, during which hunger checks are paused
Thirst	Amount of thirst recovered when the item is consumed
ThirstGrace	Grace period, in seconds, during which thirst checks are paused
EmptyContainer	The <a href="#">Item</a> for an empty container which is added to the inventory when the item is consumed, used to “convert”, for example, a full bottle into an empty bottle so that it can be crafted again into a full bottle

## Hunger and Thirst

The hunger and thirst subsystem work in the same way. Both have a stat (hunger or thirst) which depletes over time and needs to be replenished. The two system are autonomous, and the loss interval for both are independent.

In order to replenish the relevant stat, you need to use a consumable item with data filled in the new extra fields (see previous section).

The use of the grace value is optional, but these values are checked only after restoring the stat, i.e. you cannot have a “grace only” item.

Both subsystems can drain stamina and health, with different threshold.

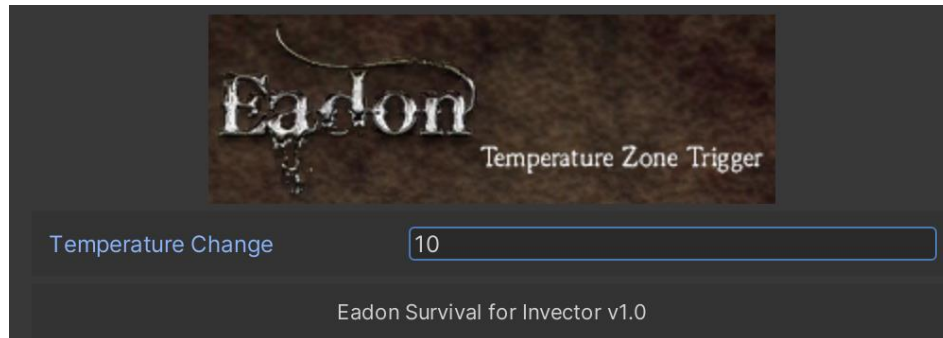
If Enviro is installed in your project, time of day can be used to reduce hunger and thirst loss at night.

Both subsystems can prevent health or stamina recovery while hungry/thirsty.

## Temperature

**The temperature subsystem is only available if Enviro is installed, as it relies on the temperature provided by this asset.**

The subsystem defines cold and hot thresholds for stamina and health and handles loss if the character is below the cold thresholds or above the hot thresholds. The actual temperature compared to the thresholds is adjusted by a localised temperature change which can be manipulated by a trigger collider on a game object with the [TemperatureZoneTrigger](#) component. It looks like this:



And it changes the perceived temperature by that amount while the character is within the volume of the attached trigger collider.

This can be used with a small area for the effect of a campfire or with bigger areas for greater effects.

If Eadon Survival is used in conjunction with Eadon RPG v1.1 or higher, equipped items can provide bonuses to cold and heat protection and mitigate loss of health and stamina due to hunger and thirst. See Eadon RPG documentation for more information.

## Sicknesses

A character can get sick or infected in two cases only. Either he's hit by an attack from a NPC which is an infector carrier or he spends time in an infectious zone. In both cases, players can be infected by a disease only once at the same time. Getting infected by a sickness you're already suffering from has no change in your situation.

### Infected NPCs

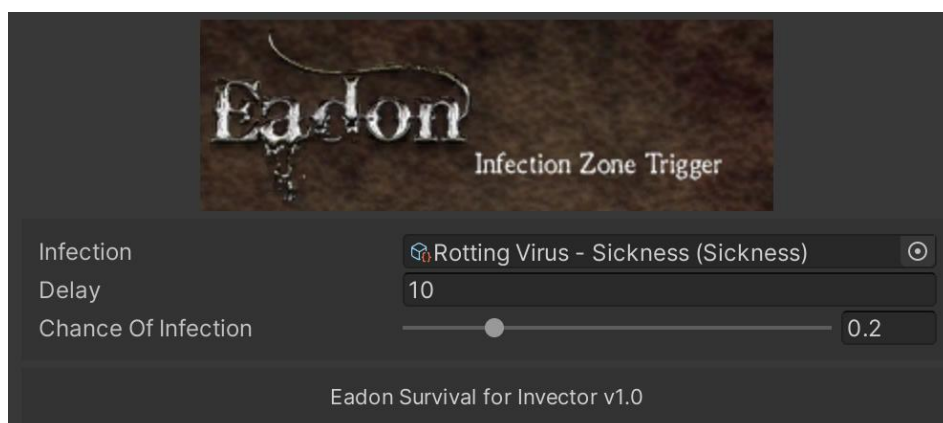
If you attach the [EadonInfectionCarrier](#) component to an NPC, every time the character is damaged by an attack there's a chance of getting infected. The component looks like this:



The fields indicate the actual sickness transmitted and the chance of an attack infecting the player.

### Infected Areas

You can create a game object with a trigger collider and attach to it the [InfectionZoneTrigger](#) component. This component tracks if the player is within the trigger volume. While inside, the component will check every x seconds for the chance to infect the character. It looks like this:



## Sickness Cures

In order to get rid of a sickness or infection, the player needs to take the appropriate cure. Every sickness can be cured in multiple ways, but you need at least one cure. The cure determines how many times it must be taken and how often.

## Injuries

Injuries are “extra effects” which happen to the player when he’s injured by an attack or by any other effect which deals damage (including fall damage).

The injury subsystem has two ways of dealing injuries to the player:

- By defining a damage threshold, every time the player receives damage equal or greater than that amount there’s a chance he’ll receive an injury
- By defining a low health threshold, whenever the player is below that level of health, every time he gets damage there’s a chance he’ll sustain an injury

If you enable the injuries system, you need to have some injuries in the [SurvivalManagerData](#) object. Whenever an injury is sustained (as determined by the two criteria above), a random injury is picked from the list of defined injuries and applied to the player. You can get a specific injury only once at the time.

## Managing Sickneses

Eadon Survival does not have a predefined UI for telling the player he's sick or injured.

There are four events in the [EadonSurvivalManager](#) component which are triggered when the player is infected, cured, injured or remedies an injury.

All these events have an int parameter which is the id of the sickness that has infected the player or has been cured or of the injury sustained or remedied.

To display this information, you need a reference to the [SurvivalManagerData](#) and some methods to listen to the events.

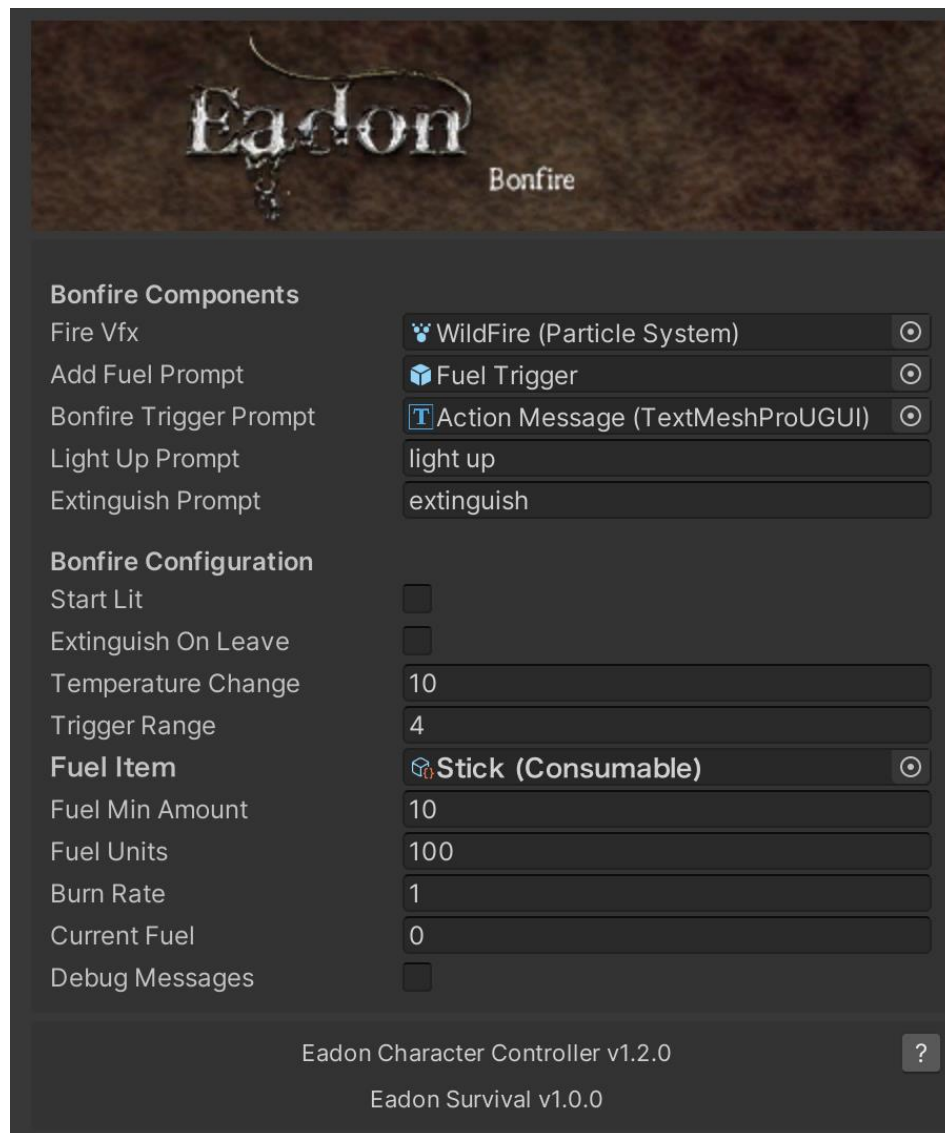
There is a sample component, [DemoEventListener](#), which writes messages to the console and which you can use as a template.

## Bonfire system

Eadon Survival comes with a bonfire system. There are three different types of bonfires: normal bonfires, healing bonfires and resting bonfires. The temperature effects require Enviro (like all other temperature effects).

### Normal Bonfires

A normal bonfire is used to raise the temperature in an area around it. It looks like this:



The fields are:

Field	Use
Fire Vfx	The particle system to use for the fire
Add Fuel Prompt	The game object for the add fuel action prompt

<b>Bonfire Trigger Prompt</b>	The text field for the prompt (to change from “light up” to “extinguish” and vice versa)
<b>Light Up Prompt</b>	The prompt message for the light up action
<b>Extinguish Prompt</b>	The prompt message for the extinguish action
<b>Start Lit</b>	A flag to indicate if the bonfire should start lit on scene load
<b>Extinguish on leave</b>	A flag to indicate if the bonfire should auto extinguish when the player leaves or not
<b>Temperature Change</b>	How many degrees should the temperature raise
<b>Trigger range</b>	The range of the temperature change
<b>Fuel Item</b>	The Item to use as fuel
<b>Fuel Min Amount</b>	How many units of fuel are needed to replenish the bonfire
<b>Fuel Units</b>	How many units of fuel the bonfire starts with
<b>Burn Rate</b>	How many seconds it takes to burn one unit of fuel
<b>Current Fuel</b>	A counter for the current (runtime) amount of fuel in the bonfire. Values in this field will be ignored
<b>Debug Messages</b>	A flag to indicate if messages should be logged to the console when events happen

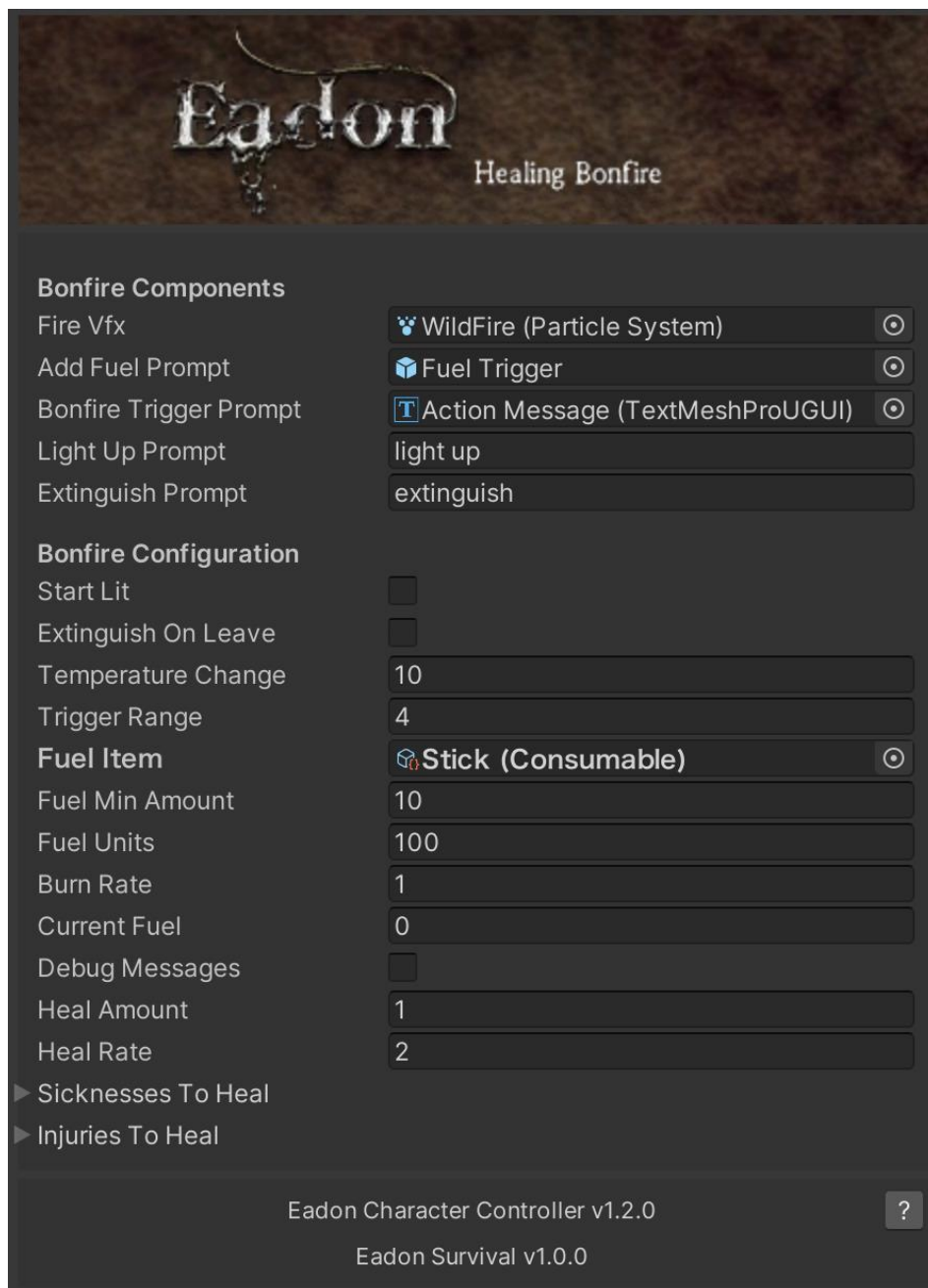
Lighting a bonfire will change the action prompt from the light up prompt to the extinguish prompt, so that you can save fuel for later. When the fuel can be replenished (i.e. when the current fuel is lower than (fuel units – fuel min amount) a second action prompt appears to let you add fuel to the bonfire:





## Healing Bonfires

Healing bonfires are bonfires that in addition to warming you up can heal you. They look like this:



The extra fields are:

Field	Use
Heal Amount	The amount of health to recover in an interval of time (see next field)
Heal Rate	How often does the character recover health, in seconds

<b>Sicknesses To Heal</b>	A list of sicknesses that can be healed by standing in front of the bonfire
<b>Injuries To Heal</b>	A list of injuries that can be healed by standing in front of the bonfire

### Resting Bonfires

A resting bonfire, when lit, allows you to rest for a specified period and recover full health and stamina. If used in conjunction with [Eadon RPG](#), full mana is also recovered. Resting will fade the screen to a rest screen and fade it back to normal after a configurable amount of time. If Enviro is in the project, in game time will also progress for a specified amount of hours while the player is resting. Resting bonfire look like this:





## Resting Bonfire

### Bonfire Components

Fire Vfx	WildFire (Particle System)	⊙
Add Fuel Prompt	Fuel Trigger	⊙
Bonfire Trigger Prompt	Action Message (TextMeshProUGUI)	⊙
Light Up Prompt	light up	
Extinguish Prompt	extinguish	





### Bonfire Configuration

Start Lit	<input type="checkbox"/>
Extinguish On Leave	<input type="checkbox"/>
Temperature Change	10
Trigger Range	4
Fuel Item	 Stick (Consumable) 
Fuel Min Amount	10
Fuel Units	100
Burn Rate	1
Current Fuel	0
Debug Messages	<input type="checkbox"/>

### Rest

Enviro Hours To Rest	6
Recover Full Health	<input checked="" type="checkbox"/>
Recover Full Stamina	<input checked="" type="checkbox"/>

### Rest Components

Fade Screen	 Rest Canvas (Canvas Group)	
Rest Prompt	 Action Message	
Fade In Time	3	
Fade Screen Time Length	5	
Lock Inputs While Resting	<input checked="" type="checkbox"/>	

### Events

On Rest Started ()

List is Empty

+ -

On Rest Ended ()

List is Empty

+ -

The extra fields are:

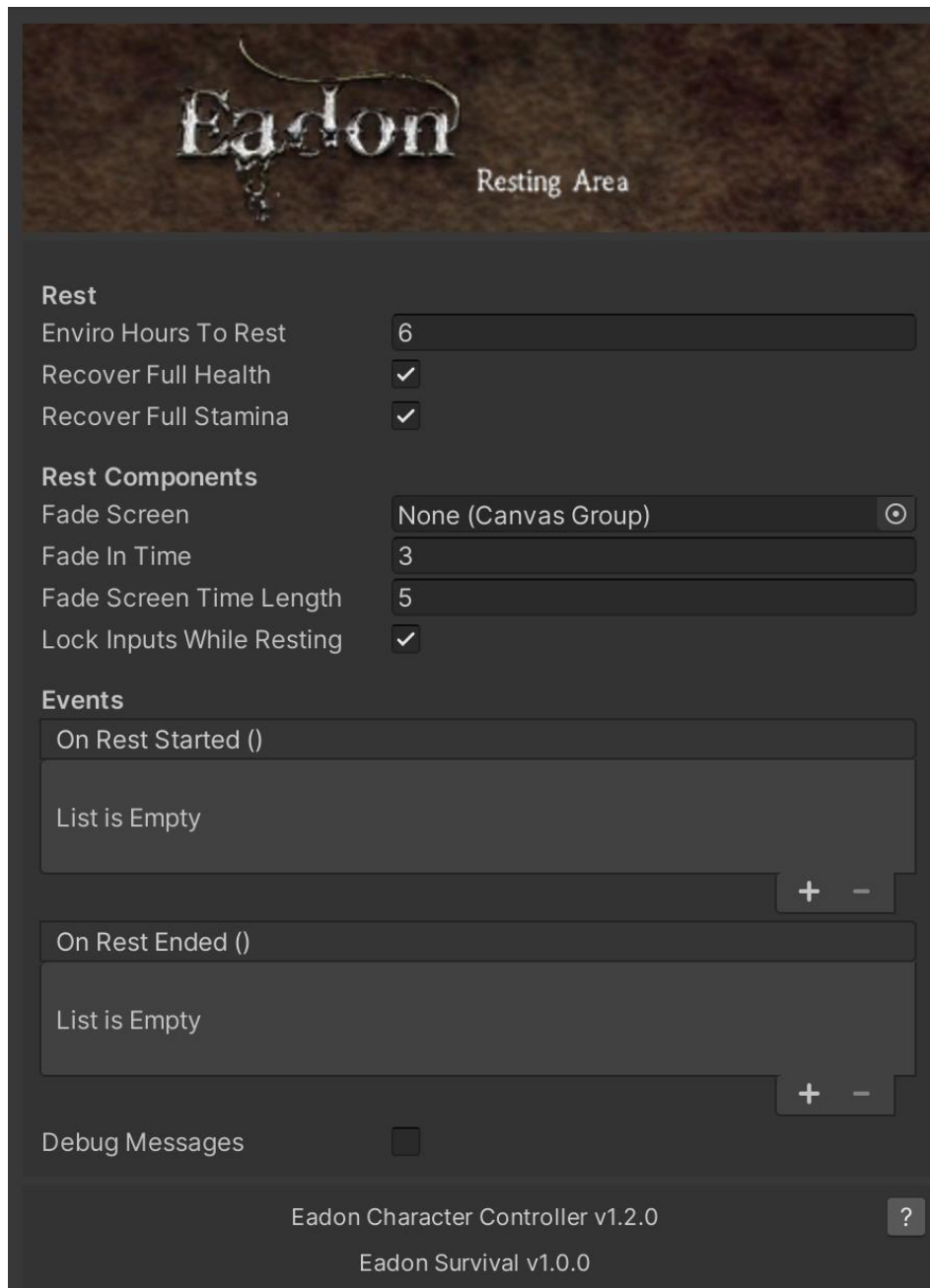
Field	Use
<b>Enviro Hours To Rest</b>	How many hours will Enviro advance while the character is resting (this option is enabled only if Enviro is I the project)
<b>Recover Full Health</b>	A flag to indicate if full health is restored
<b>Recover Full Stamina</b>	A flag to indicate if full stamina is restored
<b>Recover Full Mana</b>	A flag to indicate if full health is restored (this option is enabled only if Eadon RPG is in the project)
<b>Fade Screen</b>	A Canvas Group for the rest screen (used to fade in and out
<b>Rest Prompt</b>	A reference to the rest action trigger
<b>Fade In Time</b>	How long, in seconds, it takes to fade the rest screen
<b>Fade Screen Time Length</b>	How long the rest screen stays active (fade in/out time is in addition)
<b>Lock Inputs While Resting</b>	A flag to determine if player inputs are locked while resting
<b>On Rest Started</b>	An event triggered when rest starts
<b>On Rest Ended</b>	An event triggered when rest ends

These two last events can be used, for example, to hide UI elements and prevent them from being on screen during rest.

The Enviro time progression feature requires an EadonEnviroManager component on an object in your scene (in addition to Enviro Sky Manager). See the chapter on Bonus Content for more information on this component.

### Resting Area

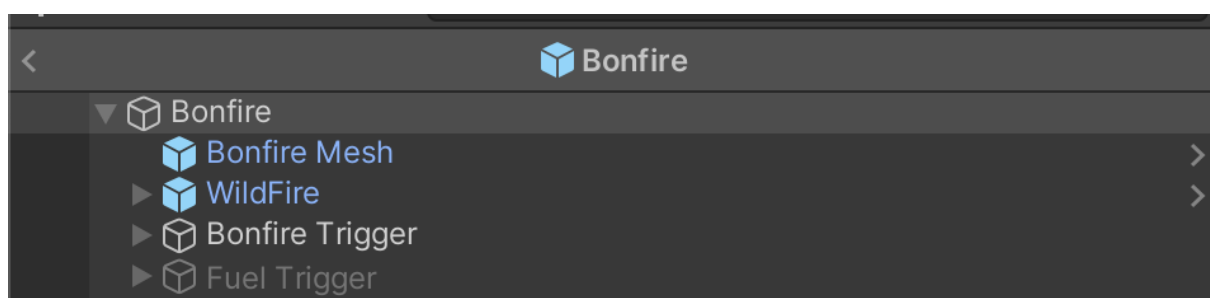
A resting area lets you use the rest functionality without a bonfire. It looks like this:



The fields are the same as for a resting bonfire.

### [Customizing a bonfire](#)

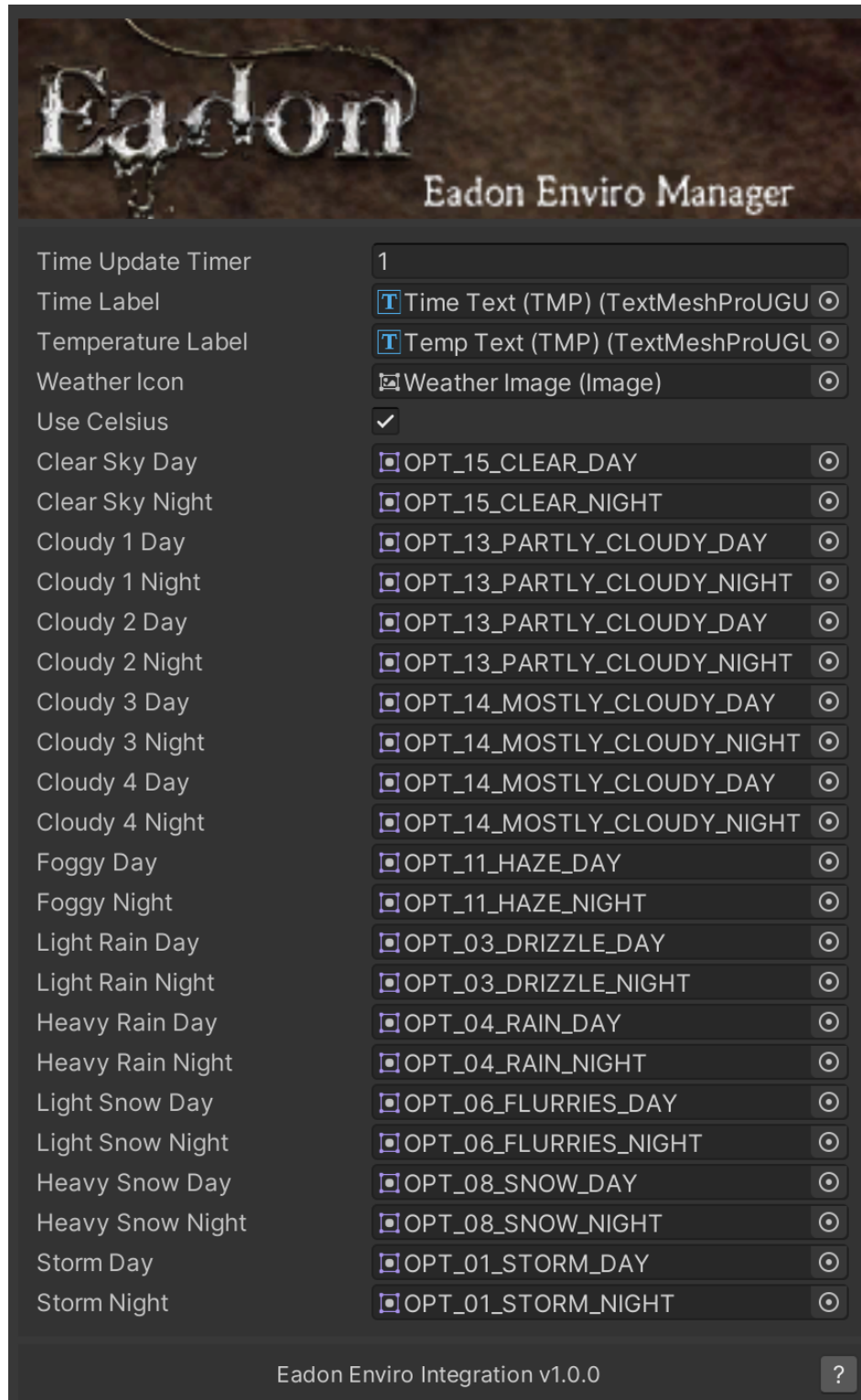
Creating your own bonfires is very simple. The prefab for a bonfire looks like this:



You can replace the bonfire mesh with any other mesh. If you replace the fire VFX prefab, just assign the reference to the new object in the Fire Vfx field in the inspector.

## Bonus Content

In the Integrations/Enviro folder you can find a prefab that you can put in your scene which will show the current time, temperature and weather from Enviro. It uses a component called [EadonEnviroManager](#), which looks like this:



The fields are:

Field	Use
<b>Time Update Timer</b>	Frequency, in seconds, for checking time and weather in Enviro
<b>Time Label</b>	A TMP_Text component in which to display the time
<b>Temperature Label</b>	A TMP_Text component in which to display the temperature
<b>Weather Icon</b>	An Image in which to display the current weather
<b>Use Celsius</b>	Untick if you want temperature displayed in Fahrenheit

The remaining fields are icons to be used with every weather preset.



## License

You agree that Cogs & Goggles own all right, title and interest in this Asset, including without limitation all applicable Intellectual Property Rights. "Intellectual Property Rights" means any and all intellectual property rights wherever in the world and whenever arising (and including any application), including patent laws, copyright, trade secrets, know-how, confidential information, business names and domain names, computer programs, trademark laws, service marks, trade names, utility models, design rights, semi-conductor topography rights, database rights, goodwill or rights to sue for passing off, and any and all other proprietary rights worldwide. You agree that you will not, and will not allow any third party to,

(i) copy, sell, license, distribute, transfer, modify, adapt, translate, prepare derivative works from, decompile, reverse engineer, disassemble or otherwise attempt to derive source code from the Asset, unless otherwise permitted,

(ii) take any action to circumvent or defeat the security or content usage rules provided, deployed or enforced by any functionality (including without limitation digital rights management or forward-lock functionality) in the Asset,

(iv) remove, obscure, or alter Cogs & Goggles' or any third party's copyright notices, trademarks, or other proprietary rights notices affixed to or contained within the Unity Asset Store or Assets.

YOU EXPRESSLY UNDERSTAND AND AGREE THAT YOUR USE OF THE ASSET IS AT YOUR SOLE RISK AND THAT THE ASSET IS PROVIDED "AS IS" AND "AS AVAILABLE" WITHOUT WARRANTY OF ANY KIND, TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW. IN PARTICULAR, COGS & GOGGLES, ITS SUBSIDIARIES, HOLDING COMPANIES AND AFFILIATES, AND ITS LICENSORS DO NOT REPRESENT OR WARRANT TO YOU THAT:

(A) YOUR USE OF THE ASSETS WILL MEET YOUR REQUIREMENTS,

(B) YOUR USE OF THE ASSETS WILL BE UNINTERRUPTED, TIMELY, SECURE OR FREE FROM ERROR,

(C) ANY INFORMATION OBTAINED BY YOU AS A RESULT OF YOUR USE OF THE ASSETS WILL BE ACCURATE OR RELIABLE, AND

(D) THAT DEFECTS IN THE OPERATION OR FUNCTIONALITY OF ANY SOFTWARE PROVIDED TO YOU AS PART OF THE ASSETS WILL BE CORRECTED.

YOUR USE OF THE ASSET IS AT YOUR OWN DISCRETION AND RISK AND YOU ARE SOLELY RESPONSIBLE FOR ANY DAMAGE TO YOUR COMPUTER SYSTEM, OR OTHER DEVICE, OR LOSS OF DATA THAT RESULTS FROM SUCH USE.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, COGS & GOGGLES FURTHER EXPRESSLY DISCLAIMS ALL WARRANTIES TERMS OR CONDITIONS OF ANY KIND, WHETHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO ANY IMPLIED WARRANTIES TERMS AND CONDITIONS OF MERCHANTABILITY, SATISFACTORY QUALITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.