



Eadon Soul Collection

A Dark Souls-like soul collection system

Version 2.0

Table of Contents

<i>Introduction</i>	3
<i>Changelog</i>	3
<i>Prerequisites</i>	4
<i>Configuration</i>	5
<i>Souls</i>	5
<i>Setting up a game object to release a soul</i>	7
<i>Collecting souls</i>	10
<i>Soul prefabs</i>	12
<i>Soul collection UI</i>	13
<i>License</i>	15

Introduction

Eadon Soul Collection is a system to collect souls from fallen enemies, inspired by Dark Souls and similar games.

The following functionalities are implemented:

- Different soul types
- Souls VFX
- Collection VFX and sound
- Integration with Eadon Character Controller
- Integration with Invector Third Person Controller

This add-on is currently compatible with Eadon Character Controller v1.5.0a and Invector Melee or Shooter v2.6.3.

Changelog

v 1.0	Initial release
v 1.1	Updated for Eadon Character Controller v1.6
v 2.0	Updated for Eadon Character Controller v2.0 Support for networking (Mirror, Fish-Net and Unity Networking for Game Objects, separate assets)

Prerequisites

The Eadon Soul Collection system can be used standalone with any character controller.

If you plan to use this system with Eadon Character Controller, you need to enable support for it by going to the **Eadon/Settings** menu and ticking the relative checkbox to enable conditional compilation of the support for this asset:



No setup is needed to use this asset with Invector TPC Melee or Shooter.

Configuration

Before the system can be used, it must be configured. Eadon Soul Collection is configured through a set of ScriptableObjects that determine what types of souls are available to the system. Souls can be created using the command found under the **Assets -> Create -> Eadon Soul Collection -> New Soul** menu.

Souls

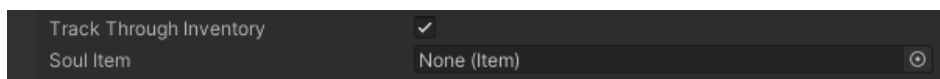
A [Soul](#) defines the structure of a collectible soul. They look like this:



The fields are:

Field name	Purpose
Soul Prefab	The prefab that is spawned when the soul is released
Soul Icon	The soul icon used in the UI
Collection VFX	The VFX spawned when the soul is collected
Collection Sound	The sound palyed when the soul is collected
Collection Action	Choice of add soul or remove soul
Soul Amount	The amount of souls collected (default one)

When used in conjunction with Eadon Character Controller, additional fields are available:



Field name	Purpose
Track Through Inventory	A flag to indicate that souls should be tracked through an inventory item
Soul Item	The item to add to the inventory

Conversely, when used with Invector Third Person Controller, two similar fields are present:

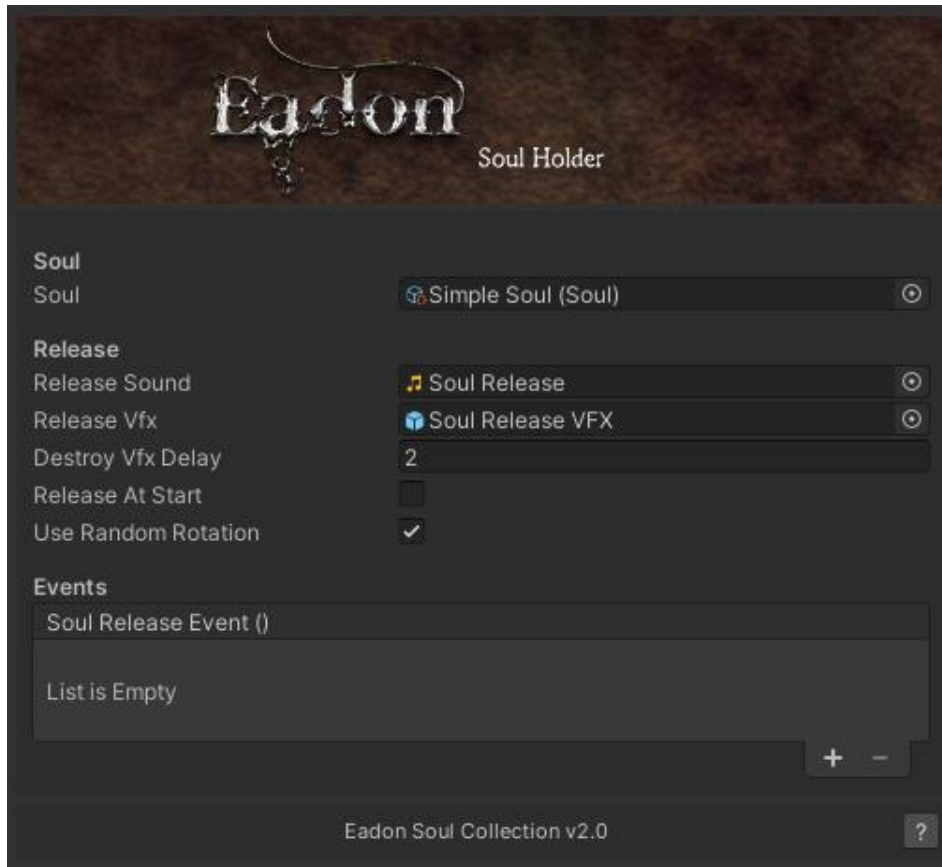
Track Through Inventory	<input checked="" type="checkbox"/>
Soul Item	<input type="text" value="0"/>

Field name	Purpose
Track Through Inventory	A flag to indicate that souls should be tracked through an inventory item
Soul Item	The vItem id of the item to add to the inventory

By default, souls are tracked in the [SoulCollector](#) component as abstract numbers. If you're using Eadon Character Controller or Invector TPC, you can choose to track souls as inventory items instead by setting the flag and providing the details of the item to use.

Setting up a game object to release a soul

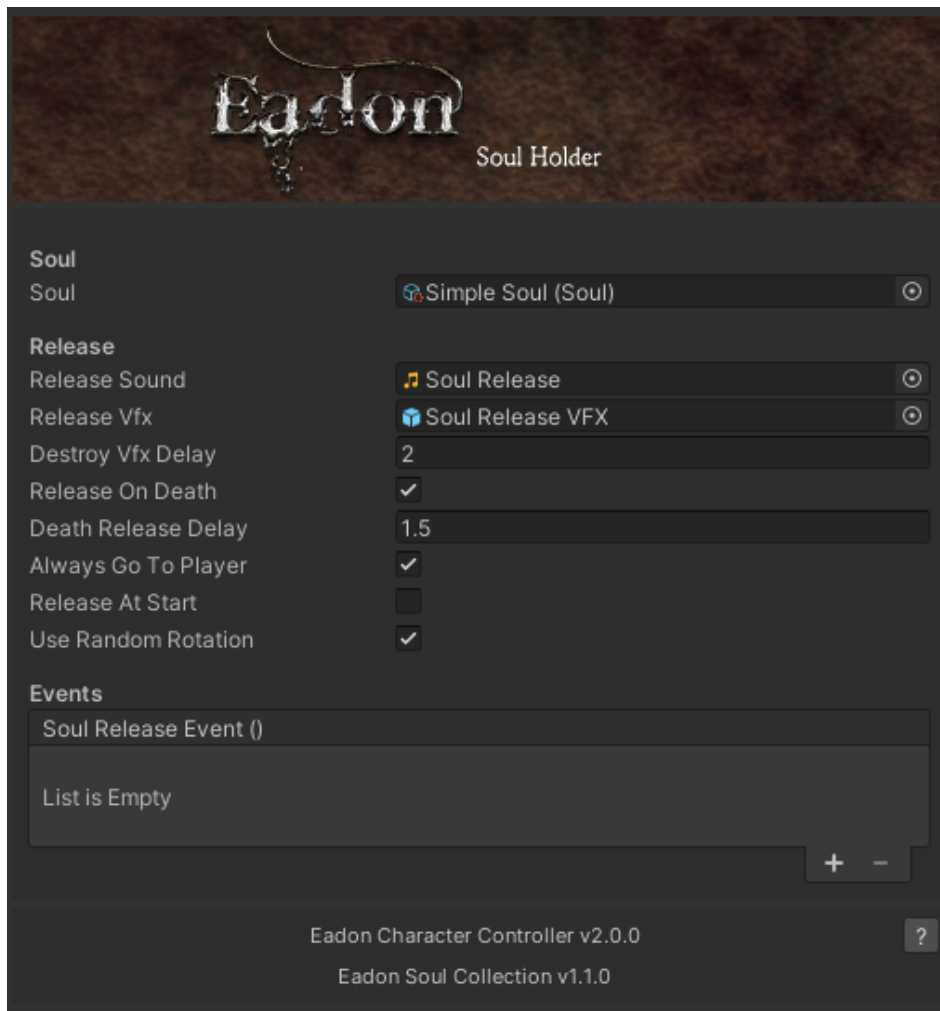
Every game object which can release a soul needs to have the Soul Holder component. In its simplest form, it looks like this:



The fields are as follows:

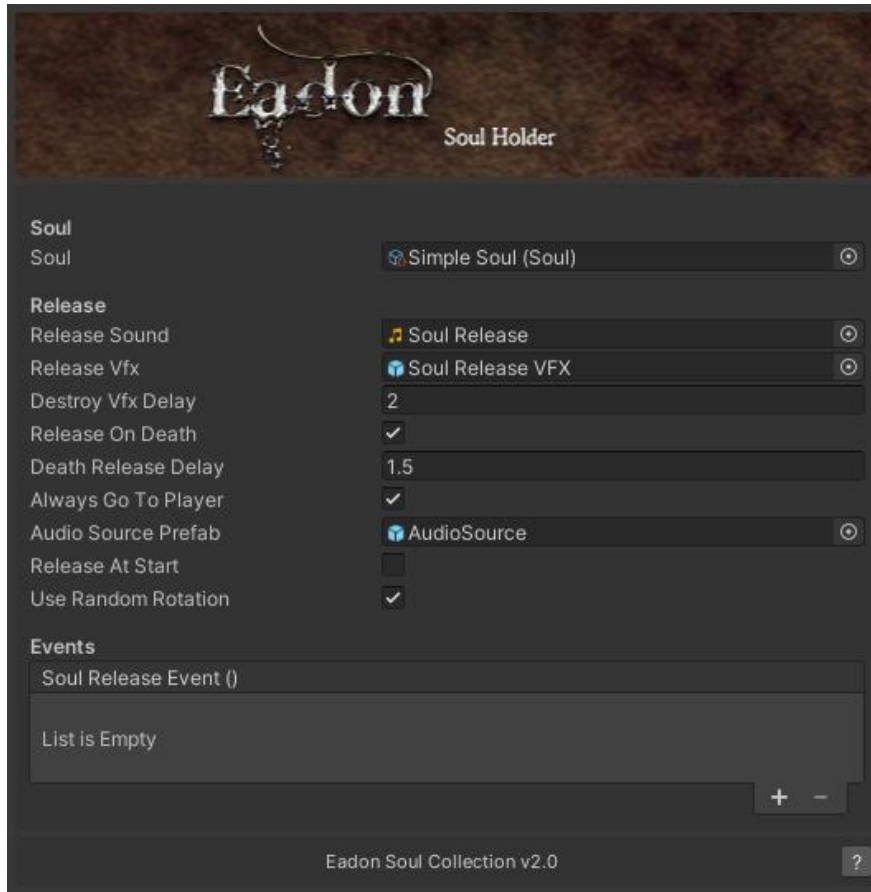
Field name	Purpose
Soul	The type of soul to release
Release Sound	The sound to play on release
Release VFX	The VFX spawned on release
Destroy VFX Delay	The delay after which the release VFX is destroyed
Release On Start	A flag to indicate the soul should be released at start
Use Random Rotation	A flag to indicate the soul prefab is spawned with a random rotation
Soul Release Event	A UnityEvent invoked when the soul is released

If you are using Eadon Soul Collection with Eadon Character Controller or Invector Third Person Controller, the Component will have three extra fields:



Field name	Purpose
Release On Death	A flag to indicate if the soul should be released on death of the holder
Death Release Delay	The delay between the holder death and the release of the soul
Always Go To Player	A flag indicating if the soul should always travel towards the player instead of towards who dealt the killing blow

For Invector there is an extra field for the audio source prefab to use.

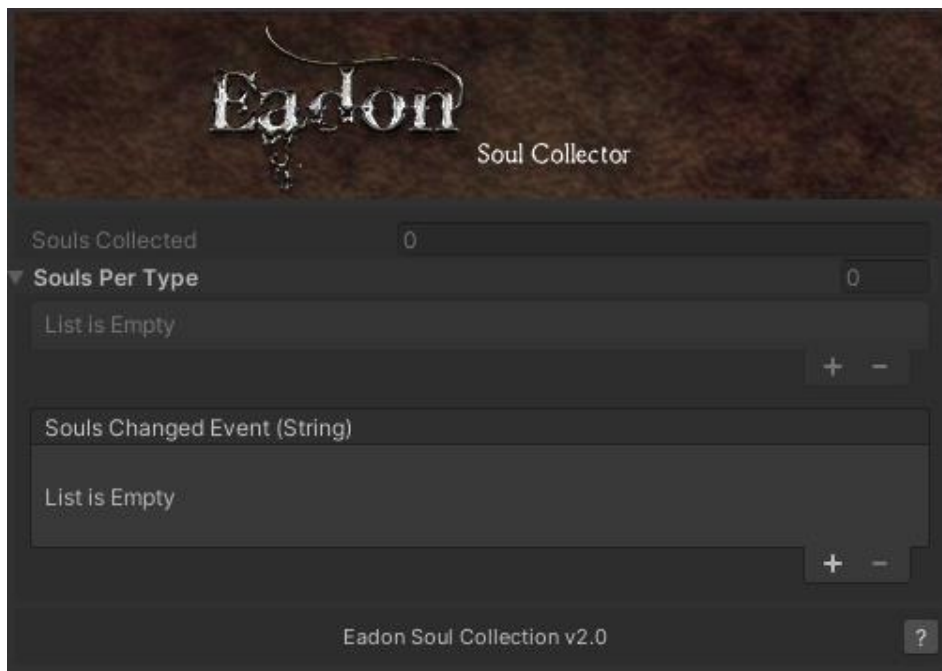


If you are using another controller, the [SoulHolder](#) component has a public method you can call to release the soul:

```
public void ReleaseSoul()
```

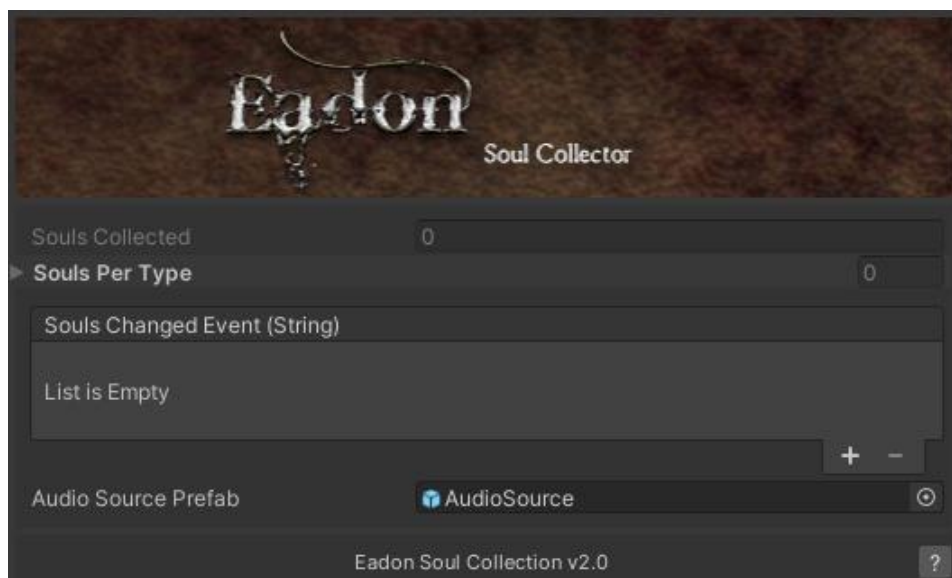
Collecting souls

To collect a soul, the player character needs to have the [SoulCollector](#) component attached in the root. In its simplest form, it looks like this:



Field name	Purpose
Souls Collected	The amount of souls collected (read only)
Souls Per Type	A list of souls collected by type (read only)
Souls Changed Event	An event that is raised when souls change (added or removed)

If used with Invector, there's an extra field for the Audio Source prefab from Invector:



Souls collected are stored inside the [SoulCollector](#) component and the component communicates with the soul collection UI to display a tally of collected souls.

Collecting Souls with Invector and Eadon Character Controller

If you want to be able to use souls in game, for example as currency, you have the option (defined in the [Soul](#) scriptable object) to track souls in the inventory.

If that option is selected for a soul, every time you acquire a soul an item is added to your inventory.

The [SoulCollector](#) component will listen for inventory events indicating that an item has been used or removed and if it's the soul item then the UI will be updated accordingly.

Please note that in this scenario the item used for the soul should not be an item that can be acquired in other ways (like found in a chest) as the UI will not detect souls added outside the soul collection system.

The item added is a normal item (or `VItem` for Invector) and the creation and configuration of the item is not part of this asset.

Soul prefabs

The Eadon Soul Collection system comes with five soul prefabs. Soul prefabs are basically a VFX object moving in space. All the included soul prefabs have these components:



The [SoulBehaviour](#) component is responsible for the movement of the soul. The fields are:

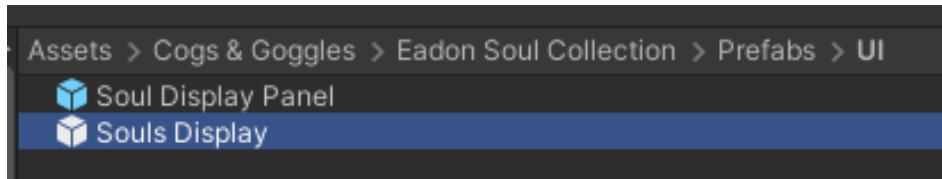
Field name	Purpose
Rising Height	The height the soul rises to from the spawn point (the position of the soul holder)
Rising Time	How long it takes for the soul to rise to the specified height
Use Curve	A flag to indicate if the soul should rise based on a curve or with linear speed
Rising Curve	The curve for the rising speed
Soul Speed	The speed of the soul
Rotation Speed	The speed the soul rotates with towards the target
Target Height	The height on the collector that the soul will reach (for example the head or chest of the player)

The soul also requires a collider set to trigger and a kinematic rigidbody.

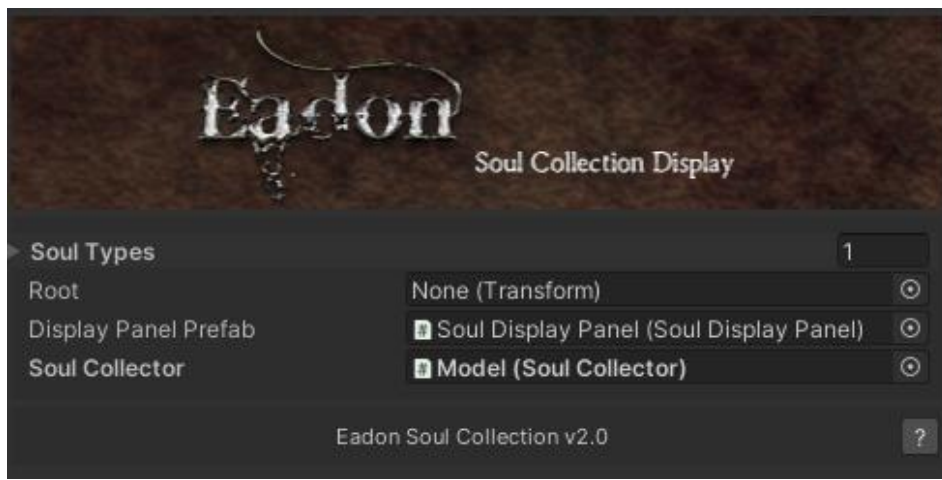
If you want to make your own soul prefabs, you just need to add these three components to the prefab you want to use.

Soul collection UI

Eadon Soul Collection comes with a drag and drop UI prefab to display the souls acquired by the player. It's located here:



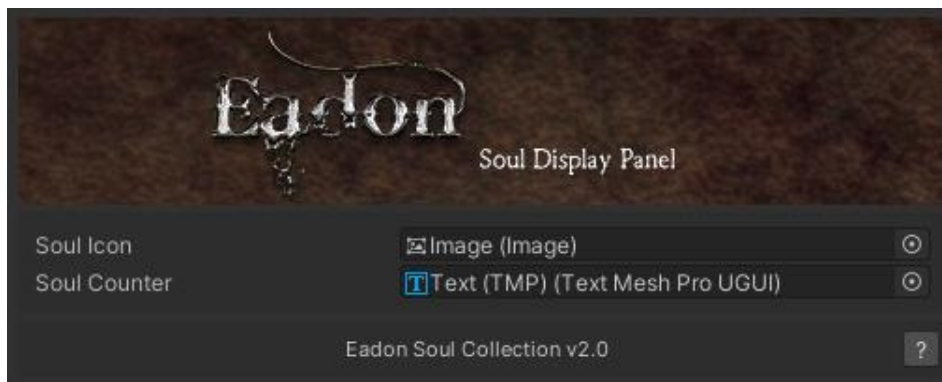
The display is driven by this component:



Field name	Purpose
Soul Types	The soul types to track in the UI
Root	An optional root transform for the list of soul prefabs
Display Panel Prefab	The soul panel to use to display collected souls
Soul Collector	A field that displays the current soul collector. Manual changes to this field will be ignored

The [SoulCollector](#) component will register with the display UI at start automatically.

The [SoulDisplayPanel](#) is a component you can customize to your taste. It needs to have these two references:



Field name	Purpose
Soul Icon	The Image into which to display the soul icon from the Soul scriptable object
Soul Counter	The TMP Text into which to write the current amount of collected souls

License

You agree that Cogs & Goggles own all rights, 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.