

Dishooks 2.0.0 documentation

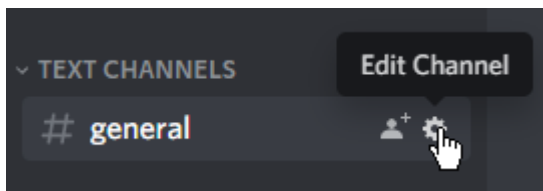
Dishooks makes it easy for everyone to send messages and rich embeds to Discord from your Unity game. Maybe you want to announce when someone completes a hard level? When someone levels up? Or why not a moderation log?

Set-up

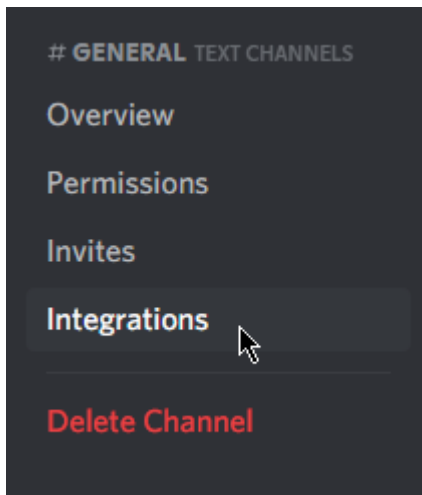
To use Dishooks you will need at least one Discord webhook, located in the Dishook script. This is a wrapper class containing the default username, profile picture and Discord channel to send your webhooks in. These can be overridden later.

To create a Discord webhook and set it as the default one to use, follow these steps:

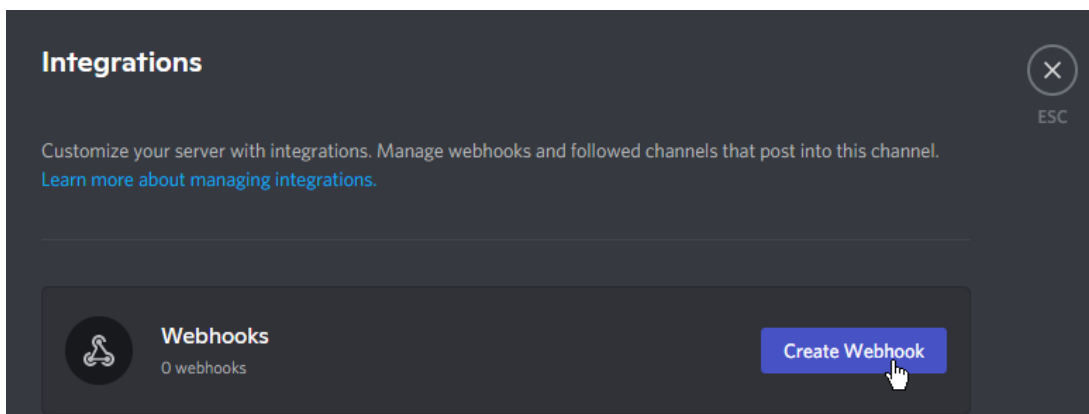
1. Edit a channel



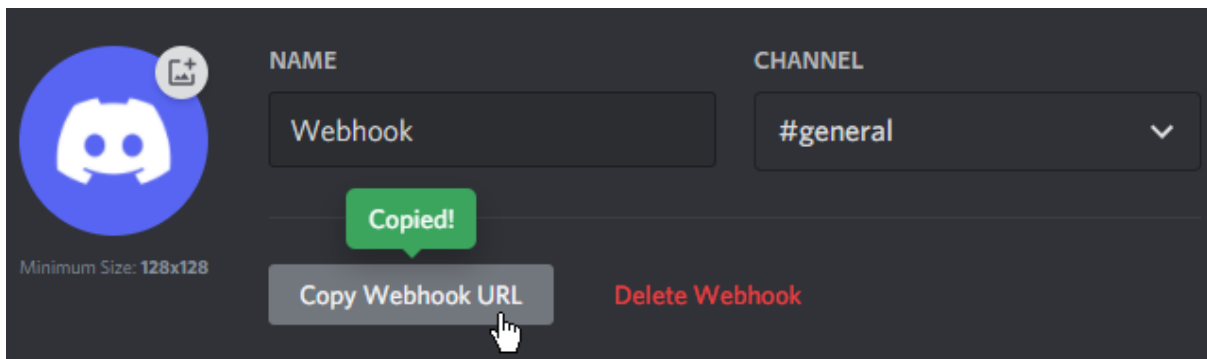
2. Go to Integrations



3. Create a webhook



4. Copy the webhook URL



5. Paste the URL you copied to the *Assets/Dishooks/Scripts/Dishook.cs* script.

```
public class Dishook : MonoBehaviour
{
    public const string DefaultName = "Dishooks";
    public const string DefaultAvatar = "https://i.imgur.com/rbSzmFv.png";
    public const string DefaultUrl = "https://discord.com/api/webhooks/7216752
```

Now you are ready to send messages!

Check the demo scene located at *Assets/Dishooks/Example/Dishook Example*, or keep reading to use the asset!

Sending messages with the wrapper (Simple)

Sending messages can be done in multiple ways. If you just want to send something quickly to a predefined channel you can use the built-in wrapper:

```
Dishook.Send("Hello World!");
```

Easy, right? If needed, you can override the other properties with this method as well. The signature is:

```
Dishook.Send(string msg, string name, string avatar, string url);
```

*For more examples, check out the **Example/DishooksExample_Simple** script.*

Sending rich messages and embeds (Advanced)

Sometimes you want to send a bit more advanced or reusable messages, and that's what the *Webhook* class is for! Here is how you can create a webhook that will send a simple message when executed:

```
Webhook wh = new Webhook
{
    URL = "https://discord.com/api/webhooks/XXX/YYY",
    Username = "Dishooks",
    AvatarUrl = "https://example.com/image.png",
    Content = "Hello World!"
};

wh.Send(); // send the message
```

Want to spice it up a little? Add an embed! Discord supports up to 10 embeds per webhook. You can find the full embed structure on the [Discord Documentation](#).

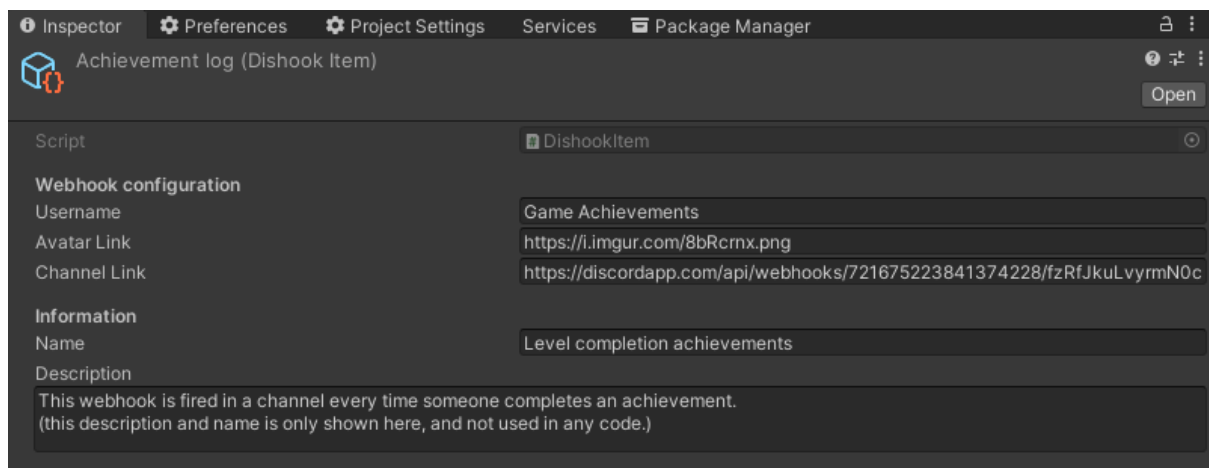
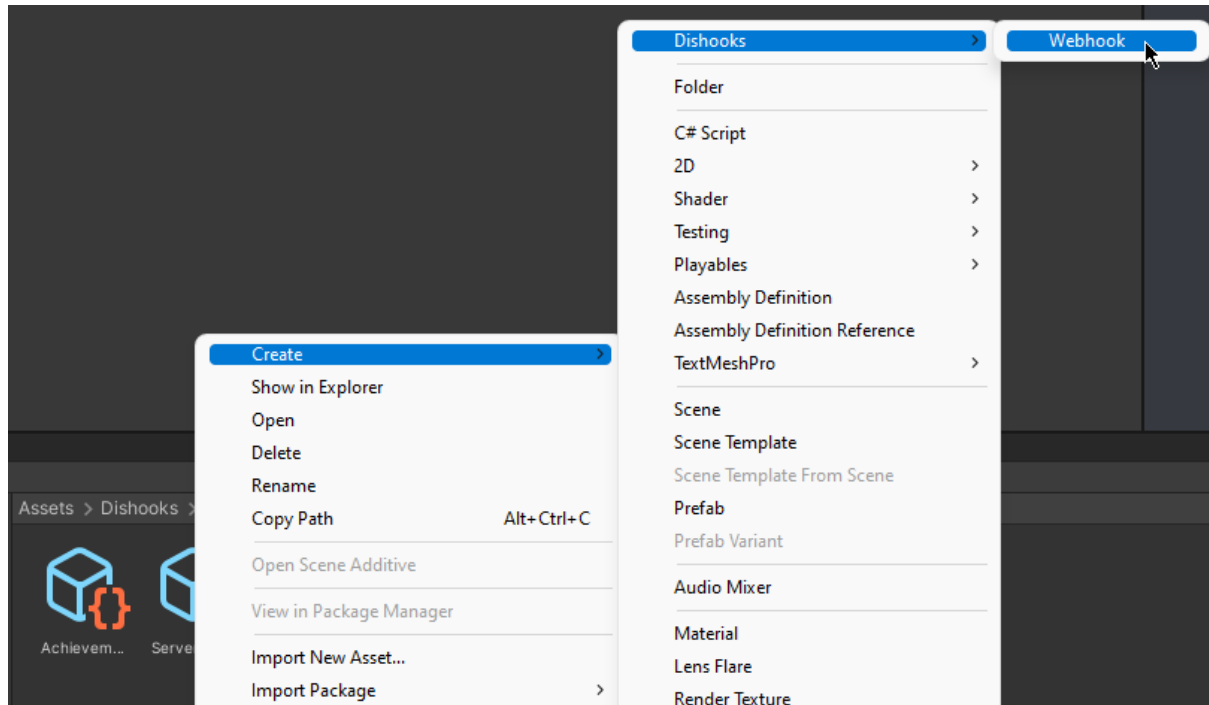
```
Embed embed = new Embed
{
    Color = Color.magenta,
    Author = new Author("Fabian", "example.com/image.png"),
    Title = "Welcome to Dishooks!",
    Description = "Create complex rich embeds with ease.",
    Fields = new []
    {
        new Field("Field title", "Field body text")
    }
};

wh.AddEmbed(embed); // add the embed
wh.Send(); // send the message again
```

For more examples, check out the ***Example/DishooksExample_Advanced*** script.

Dishook items

If you have a lot of different webhooks it can be a pain to keep all the links and configurations organized all the time. To help with this, you can now create an item for each webhook with 1.2 and use them in your script instead!



To use these configurations, simply pass the DishookItem to the constructor when creating a new webhook, like this:

```
Webhook webhook = new Webhook(_achievementLog)
{
    Content = "Hello World!"
};
```

Upgrading from 1.2.0 to 2.0.0

2.0.0 is a complete rewrite of Dishooks, and is *almost* fully compatible with the previous versions, with the exception of the `Dishook.Send()` methods return type. It has been changed from *bool* to *void*.

The reason for this change is mainly because of its low usability. The new `Dishook.Post()` method will now log errors for you, so they can be fixed before you release or continue to expand on your game.

Need help with anything?

You can contact me at lindgrenfabian@gmail.com.

When emailing, please state your invoice number you received when you bought the asset, found [here](#). Thank you for using Dishooks!