TESTING ADDONS IN DEVELOPER MODE

View online: https://www.construct.net/en/make-games/manuals/addon-sdk/guide/using-developer-mode

---

While developing an addon, it is inconvenient to have to package and install a new .c3addon file every time you make a change. To make testing easier, you can load an addon from a local web server. This means every time you reload Construct 3, it re-loads the latest version of your addon from your local web server, saving you from having to make any .c3addon packages. Developer mode also means Construct reloads the addon files on every preview, making it much easier to check changes to runtime code, but note that most of the time changing code that affects the editor will still require a reload of the editor.

Step 1: enable Developer Mode

For these steps to work, you must first enable Developer Mode.

Step 2: start a local web server

There are many ways to run a local web server, e.g. with a Chrome extension, or a standalone server like nginx. Refer to your chosen server's documentation for installation and configuration.

Your local web server must host on *localhost*. Construct 3 will refuse to load addons from any other origin. You can host on any port, but it is recommended to use a port in the ephemeral port range 49152-65535.

The local web server must serve all the addon files with CORS (Cross-Origin Resource Sharing) enabled for Construct 3 to be able to load them, since it will be making a cross-domain request. In practice this means adding this HTTP header to the server response:

```
Access-Control-Allow-Origin: *
```

To do this you need to enable CORS if your web server provides a setting for that, or manually specify the header. For example in nginx, add the following directive in your server location section:

```
add_header Access-Control-Allow-Origin *;
```

You may also wish to review the caching headers to ensure your local server does not return old cached files. Disabling caching entirely will ensure Construct 3 always receives the latest files.

Once the server is fully configured, simply host the contents of the c3addon file in a folder on your local web server. For example the URL *http://localhost:65432/myaddon/addon.json* should serve the addon.json file for your addon.

In modern browsers, `http://localhost` *counts as a secure context even though it does not use HTTPS. Therefore you should not need to set up SSL/TLS for the local server.*

Step 3: update addon.json to include file list

Normally when files are extracted from the zip-format .c3addon file, Construct 3 can obtain a file list from the zip. However when loading from a local web server, Construct 3 needs another way to identify all available files. So you must update addon.json to include a complete file list. For more information see the section *Developer mode addons* in Addon metadata.

The file list can be left in when distributing your .c3addon file — there's no need to later remove it.

Step 4: install the developer mode addon

In Construct 3, open the Addon Manager. After enabling Developer Mode, there should be a new button at the bottom labelled *Add dev addon…*. Click this button. A dialog will appear asking for the URL to the addon's addon.json file on your local web server. Enter its path, e.g. *http://localhost:65432/myaddon/addon.json*, and press OK. If the addon.json file is reachable and parsed successfully, you'll see a message indicating to restart Construct 3 to load the addon. If an error occurs, check the browser console for more details.

Step 5: develop the addon

Now every time you reload Construct 3, or preview a project using the addon, the latest version of your addon is loaded from the local web server. This makes it much quicker to test changes to your addon. For example if you make a change that crashes Construct 3 on startup, there is no need to clear the browser cache to remove the addon; you can simply fix the problem and reload C3. Similarly you can adjust some runtime code and press preview, and your code changes will be immediately reflected.

Your addon will appear in the Addon Manager as having a "Developer" source. Note "Developer" addons cannot be bundled with projects when using the *Bundle addons* feature.

You can also uninstall your developer mode addon like any other addon, by right-clicking it in the Addon Manager and selecting *Uninstall*.