

AIO project tooling

This document gives an overview of the tools that we use to generate the content for the angular.io website.

The application that actually renders this content can be found in the `/aio/src` folder. The handwritten content can be found in the `/aio/content` folder.

Each subfolder in this `/aio/tools/` folder contains a self-contained tool and its configuration. There is a `README.md` file in each folder that describes the tool in more detail.

cli-patches

The AIO application is built using the Angular CLI tool. We are often trialling new features for the CLI, which we apply to the library after it is installed. This folder contains git patch files that contain these new features and a utility to apply those patches to the CLI library.

See the `README.md` for more details.

examples

Many of the documentation pages contain snippets of code examples. We extract these snippets from real working example applications, which are stored in subfolders of the `/aio/content/examples` folder. Each example can be built and run independently. Each example also provides e2e specs, which are run as part of our CI build tasks, to verify that the examples continue to work as expected, as changes are made to the core Angular libraries.

In order to build, run and test these examples independently we need to install dependencies into their sub-folder. Also there are a number of common boilerplate files that are needed to configure each example's project. We maintain these common boilerplate files centrally to reduce the amount of effort if one of them needs to change.

This `examples` tool folder contains three utilities:

- `example-boilerplate.js` - install/remove the npm dependencies and boilerplate files into/from each of the examples' subfolders.
- `run-example-e2e.mjs` - run the e2e tests for one or more examples
- `create-example.js` - create a new example from the `example-scaffold/` directory or by importing files from a CLI project.

See the `README.md` for more details.

example-zipper

In the AIO application, we offer the reader the option to download each example as a full self-contained runnable project packaged as a zip file. These zip files

are generated by the utility in this folder.

See the README.md for more details.

stackblitz-builder

In the AIO application, we can embed a running version of the example as a Stackblitz session. We can also provide a link to create a runnable version of the example in the Stackblitz editor.

See the README.md for more details.

transforms

All the content that is rendered by the AIO application, and some of its configuration files, are generated from source files by Dgeni. Dgeni is a general purpose documentation generation tool.

Markdown files in `/aio/content`, code comments in the core Angular source files and example files are processed and transformed into files that are consumed by the AIO application.

Dgeni is configured by “packages”, which contain services and processors. Some of these packages are installed as `node_modules` from the dgeni-packages and some are specific to the AIO project.

The project specific packages are stored in the `aio/tools/transforms` folder. See the README.md for more details.