

Overview

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.

Example zipper

The `exampleZipper.mjs` tool is very similar to the Stackblitz's `builder.mjs`. The latter creates an HTML file with all the examples' files and the former creates a zip file instead. They both use the `stackblitz.json` file to flag an example as something to stackblitz or zip. For example:

```
{
  "description": "Tour of Heroes: Part 6",
  "files": [
    "**/*.d.ts",
    "**/*.js",
    "**/*. [1,2].*"
  ],
  "tags": ["tutorial", "tour", "heroes", "http"]
}
```

The zipper will use this information for creating new zips.

Two kinds of examples

There are mainly two kinds of AIO docs examples: The ones based on the Angular CLI and the ones based on SystemJS. The majority of the examples are CLI-based with only some of the `ngUpgrade` examples using SystemJS.

Some of the CLI-based examples require small tweaks to the default layout/configuration (for example, to add support for Angular elements, i18n, universal, etc.). These example types have separate boilerplate directories with the files that are different from the default `cli` boilerplate.

There are appropriate `package.json` files for each type of example in the boilerplate directories. If there is no special `package.json` file for an example type, the one from the `cli` boilerplate directory will be used instead.

The `exampleZipper.mjs` won't include any `System.js` related files for CLI-based projects.

The zipper.json

As mentioned, the tool uses the `stackblitz.json` as a flag and also a configuration file for the zipper. The problem is that not all examples have a `stackblitz` but they could offer a zip instead.

In those cases, you can create a `zipper.json` file with the same syntax. It will be ignored by the `stackblitz` tool.

Executing the zip generation

`generateZips.mjs` will create a zip for each `stackblitz.json` or `zipper.json` it finds.

Where? At `src/generated/zips/`.

Then the `<live-example>` embedded component will look at this folder to get the zip it needs for the example.