> *Note: this feature is available with `react-scripts@3.3.0` and higher.*

Custom Templates enable you to select a template to create your project from, while still retaining all of the features of Create React App.

You'll notice that Custom Templates are always named in the format `cra-template-[template-name]`, however you only need to provide the `[template-name]` to the creation command.

Scoped templates are also supported, under the name `@[scope-name]/cra-template` or `@[scope-name]/cra-template-[template-name]`, which can be installed via `@[scope]` and `@[scope]/[template-name]` respectively.

```
npx create-react-app my-app --template [template-name]
```

Finding custom templates

We ship two templates by default:

- `cra-template`
- `cra-template-typescript`

However, you can find many great community templates by searching for "cra-template-*" on npm.

Building a template

If you're interested in building a custom template, first take a look at how we've built `cra-template`.

A template must have the following structure:

```
cra-template-[template-name]/
  README.md (for npm)
  template.json
  package.json
  template/
    README.md (for projects created from this template)
    gitignore
    public/
      index.html
    src/
      index.js (or index.tsx)
```

Testing a template

To test a template locally, pass the file path to the directory of your template source using the `file:` prefix.

```
npx create-react-app my-app --template file:../path/to/your/template/cra-template-[template-name]
```

The `template` folder

This folder is copied to the user's app directory as Create React App installs. During this process, the file `gitignore` is renamed to `.gitignore`.

You can add whatever files you want in here, but you must have at least the files specified above.

The `template.json` file

This is the configuration file for your template. As this is a new feature, more options will be added over time. For now, only a `package` key is supported.

The `package` key lets you provide any keys/values that you want added to the new project's `package.json`, such as dependencies and any custom scripts that your template relies on.

Below is an example `template.json` file:

```json
{
  "package": {
    "dependencies": {
      "eslint-plugin-jsx-a11y": "^6.2.3",
      "serve": "^11.2.0"
    },
    "scripts": {
      "serve": "serve -s build",
      "build-and-serve": "npm run build && npm run serve"
    },
    "eslintConfig": {
      "extends": ["react-app", "plugin:jsx-a11y/recommended"],
      "plugins": ["jsx-a11y"]
    }
  }
}
```

Any values you add for `"dependencies"` and `"scripts"` will be merged with the Create React App defaults. Values for any other keys will be used as-is, replacing any matching Create React App defaults.

For convenience, we always replace `npm run` with `yarn` in your custom `"scripts"`, as well as in your `README` when projects are initialized with yarn.