

## Electron with Typescript application example

This example show how you can use Next.js inside an Electron application to avoid a lot of configuration, use Next.js router as view and use server-render to speed up the initial render of the application. Both Next.js and Electron layers are written in TypeScript and compiled to JavaScript during the build process.

Part	Source code (Typescript)	Builds (JavaScript)
Next.js	/renderer	/renderer
Electron	/electron-src	/main
Production		/dist

For development it's going to run a HTTP server and let Next.js handle routing. In production it use `next export` to pre-generate HTML static files and use them in your app instead of running an HTTP server.

### How to use

Execute `create-next-app` with npm or Yarn to bootstrap the example:

```
npx create-next-app --example with-electron-typescript with-electron-typescript-app
# or
yarn create next-app --example with-electron-typescript with-electron-typescript-app
# or
pnpm create next-app -- --example with-electron-typescript with-electron-typescript-app
```

Available commands:

```
"build-renderer": build and transpile Next.js layer
"build-electron": transpile electron layer
"build": build both layers
"dev": start dev version
"dist": create production electron build
"type-check": check TypeScript in project
```

### Notes

You can create the production app using `npm run dist`.

*note regarding types:*

- Electron provides its own type definitions, so you don't need `@types/electron` installed! source: <https://www.npmjs.com/package/@types/electron>
- There were no types available for `electron-next` at the time of creating this example, so until they are available there is a file `electron-next.d.ts` in `electron-src` directory.