



Create Angular Desktop App Using Electron



Maragatha Mani

Updated date Jan 24, 2020

16.8k

5

6

[Download Free .NET & JAVA Office Files API](#)
[Try Free File Format APIs for Word/Excel/PDF](#)

Introduction

In this article, we'll learn to build cross-platform Desktop app for Angular 8/7 by integrating it with Electron.

Electron

[Electron](#) is an open-source framework developed and maintained by GitHub. It is a popular platform for building cross-platform desktop apps for Windows, Linux and macOS with JavaScript, HTML, and CSS. It is essentially a web application that is self-contained as a desktop application.

In this guide, we will look at how to create an Electron application with the Angular framework using TypeScript. We will cover,

- Building a project from scratch
- Packaging the desktop application for distribution
- Using live reloading for development

Installing AngularCLI

Let's get started by installing Angular CLI, which is the official tool for creating and working with Angular projects. Open a new terminal and run the below command.

```
npm install -g @angular/cli
```

Steps to follow,

Step 1

Create an Angular project setup using the below command or however you want to create it.

```
ng new projectname
```

Example,

```
ng new angularElectron
```

 Create Angular Desktop App Using Electron

Step 2

Now, we must install Electron in our Angular app. Open a new terminal and run the below command.

```
npm install electron --save-dev
```

 Create Angular Desktop App Using Electron

Even after installing, Electron will be unaware of the Angular app. So, it should be linked with the Angular app in order to work.

Step 3

Create a script file named as main.js and copy the below code in it. This file is used to link the Angular project with Electron.

```
01.  const {  
02.      app,  
03.      BrowserWindow  
04.  } = require('electron')  
05.  // Keep a global reference of the window object, if you don't, the window  
06.  // be closed automatically when the JavaScript object is garbage collected  
07.  let win  
08.  
09.  function createWindow() {  
10.      // Create the browser window.
```

```

11.     win = new BrowserWindow({
12.         width: 800,
13.         height: 600,
14.         webPreferences: {
15.             nodeIntegration: true
16.         }
17.     })
18.     // and load the index.html of the app.
19.     win.loadFile('dist/angularElectron/index.html')
20.     // Open the DevTools.
21.     win.webContents.openDevTools()
22.     // Emitted when the window is closed.
23.     win.on('closed', () => {
24.         // Dereference the window object, usually you would store windows
25.         // in an array if your app supports multi windows, this is the time
26.         // when you should delete the corresponding element.
27.         win = null
28.     })
29. }
30. // This method will be called when Electron has finished
31. // initialization and is ready to create browser windows.
32. // Some APIs can only be used after this event occurs.
33. app.on('ready', createWindow)
34. // Quit when all windows are closed.
35. app.on('window-all-closed', () => {
36.     // On macOS it is common for applications and their menu bar
37.     // to stay active until the user quits explicitly with Cmd + Q
38.     if (process.platform !== 'darwin') {
39.         app.quit()
40.     }
41. })
42. app.on('activate', () => {
43.     // On macOS it's common to re-create a window in the app when the
44.     // dock icon is clicked and there are no other windows open.
45.     if (win === null) {
46.         createWindow()
47.     }
48. })
49. // In this file you can include the rest of your app's specific main process
50. // code. You can also put them in separate files and require them here.

```

Change the path of index.html in the script based on your project path.

Refer line#18

 Create Angular Desktop App Using Electron

Step 4

In Index.html, modify the below line

01. | <base href="/"> to <base href="./">

Refer line#6



Step 5

In package.json, add the below line to link the newly created script file.

```
"main": "main.js",
```

In Scripts, add the below code to run Electron with short code.

```
"electron": "ng build && electron ."
```

Refer line#12



Step 6

Now, run the application

```
npm run electron
```



On successful execution of the above command, the following Angular UI window pops out.



Angular

Create Angular Desktop App

Electron

Node.js

Next Recommended Reading

[Angular V6 Desktop Application With Firebase And Electron](#)

OUR BOOKS



Maragatha Mani *TOP 1000*

598

283.3k

1

6

5



Type your comment here and press Enter Key (Minimum 10 characters)



I tried, but when i put "npm run electron", the app open with a blank page (only html)

[carlos france](#)

2071 3 0

Mar 05, 2020

0

0

Reply



Thanks for sharing this article

[Rathrola Prem Kumar](#)

317 7.4k 1.3m

Jan 27, 2020

1

1

Reply



Thanks Rathrola Prem Kumar

[Maragatha Mani](#)

598 3.4k 283.3k

Jan 28, 2020

0



Nice article on Desktop app Angular using electron.

[Sourav Kumar Das](#)

721 2.5k 126.6k

Jan 25, 2020

1

1

Reply



Thanks Sourav Kumar Das

[Maragatha Mani](#)

598 3.4k 283.3k

Jan 28, 2020

0

FEATURED ARTICLES

How To Upgrade to Windows 11

Exploring Subject <T> In Reactive Extensions For .Net

Micro Frontends With Webpack

What's New In iPhone 13

Understanding Synchronization Context Task.ConfigureAwait In Action

[View All](#) 

TRENDING UP

- 01 The Best VS Code Extensions For Remote Working
- 02 Getting Started With .NET 6.0
- 03 Understanding Numpy With Examples
- 04 C# Evolution
- 05 Understanding Matplotlib With Examples
- 06 How To Post Data In ASP.NET Core Using Ajax
- 07 Top 15 Git Commands With Examples For Every Developers🔖
- 08 Understanding Pandas With Examples
- 09 How To Install Visual Studio 2022
- 10 Using The Roslyn C# Compiler

[View All](#) ○

[About Us](#) [Contact Us](#) [Privacy Policy](#) [Terms](#) [Media Kit](#) [Sitemap](#) [Report a Bug](#) [FAQ](#) [Partners](#)

[C# Tutorials](#) [Common Interview Questions](#) [Stories](#) [Consultants](#) [Ideas](#) [Certifications](#)

©2021 C# Corner. All contents are copyright of their authors.