

# Hello, SharePoint Framework (SPFx): Getting Started – Quick and Dirty

JUNE 19, 2017JUNE 19, 2017 ~ JASON RIVERA

I know, I know... there's github, pluralsight, PnP, dev.office.com and a plethora of videos and sites where you can find info on getting started with SPFx. We'll just add this one to the list. This is a **quick and dirty** getting started guide. I'll also provide some references for you to dive deeply into related topics.

## Setup

What do I need to install?

For starters, you need to install [Node.js](https://nodejs.org/en/). (https://nodejs.org/en/). Get yourself the LTS version.

Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine. Node.js uses event-driven, non-blocking I/O model that makes it lightweight and efficient. Node.js package ecosystem, npm, is the largest ecosystem of open source libraries in the wor

### Download for Windows (x64)

<b>v6.11.0 LTS</b> Recommended For Most Users	<b>v8.1.2 Current</b> Latest Features
<a href="#">Other Downloads</a>   <a href="#">Changelog</a>   <a href="#">API Docs</a>	<a href="#">Other Downloads</a>   <a href="#">Changelog</a>   <a href="#">API Docs</a>

Or have a look at the [LTS schedule](#).

Once you have that installed, you can open up a command shell of your choice. I'll be using PowerShell. You're going to want to install the **node package manager (npm)** globally so you can use it from the command line for all of your projects. In order to do that, you'll just need to run the following command in PowerShell or whatever command shell you decided to use:

```
1 | npm install npm -g
```

If the above powershell looks a little weird to you, what we're saying in english is that we want to use the node package manager to install the latest node package manager globally.

Next up is the Yeoman Generator. This is a handy little tool that creates your SharePoint project for you. So we need to go back to your trusty npm and install yeoman globally.

```
1 | npm install yo -g
```

Once you have the Yeoman generator, you'll want the SharePoint template generator.

```
1 | npm install @microsoft/generator-sharepoint -g
```

Next up, Gulp. Now, gulp is a task automation tool. We're basically going to use it to run our solution.

```
1 | npm install gulp -g
```

There's one last package that needs to be installed and that's webpack, but I'll skip it for now. We'll set that up when it's time to deploy our solution.

## First Web Part

Now that we have (almost) everything that we need installed, we can start with our project. The installs above were 1 time installs since we specified that they were each to be globally installed.

The first thing that we want to do is create the directory where our solution will live. So in PowerShell, I go to the base directory where I store all of my solutions and then I type the following command to create a directory called "SPFx" (call it whatever you want):

```
1 | md SPFx
```

The above is the mkdir command so you could type the following and it's exactly the same:

```
1 | mkdir SPFx
```

Then we want to go to that newly created directory and run our Yeoman generator to create the project files. (At this point, you should be in your project path in PowerShell)

```
1 | yo @microsoft/sharepoint
```

Next, the generator will start asking you for details like what do you want to call your solution, are you building a web part or an extension, names, descriptions, and what JavaScript library you want to use. (The version of the generator at the time of this post is Version 2.0.0.0). Depending on which version you're on, you may need to provide different inputs.

Here are my inputs. Once I hit enter, it starts installing a ton of stuff so you have time to check your email or maybe get a beverage of your choice.

```
PS C:\Projects\Demo\spfx2> yo @microsoft/sharepoint

  _____
  |-----|
  |  (o)  |
  |_____|
  |  U  |
  |_____|
  |  A  |
  |_____|
  |  ~  |
  |_____|
  |  T  |
  |_____|
  |  V  |
  |_____|

Welcome to the
SharePoint Client-side
Solution Generator

Let's create a new SharePoint solution.
? What is your solution name? SPFx
? Which type of client-side component to create? WebPart
? What is your WebPart name? HelloWorld
? What is your WebPart description? HelloWorld description
? Which framework would you like to use? (Use arrow keys)
> No JavaScript framework
  React
  Knockout
```

And this is what you'll see when it's done. If you look inside the box in the image, it tells you what you need to "start" your solution. You can go look at the **252 MB** of SharePointy goodness that get's created in the directory above. (The size on disk is 297 MB). I'll let you inspect the files but for our purposes, we just want to run it. This part I find kind of cool. You DON'T need SharePoint installed on your machine for this.

```
added 1613 packages in 266.088s

#####
###/ (##) (@)
### #####
###/ /###
##### ##
### /## (@)
#####
**=+#####

Congratulations!
Solution sp-fx is created.
Run gulp serve to play with it!

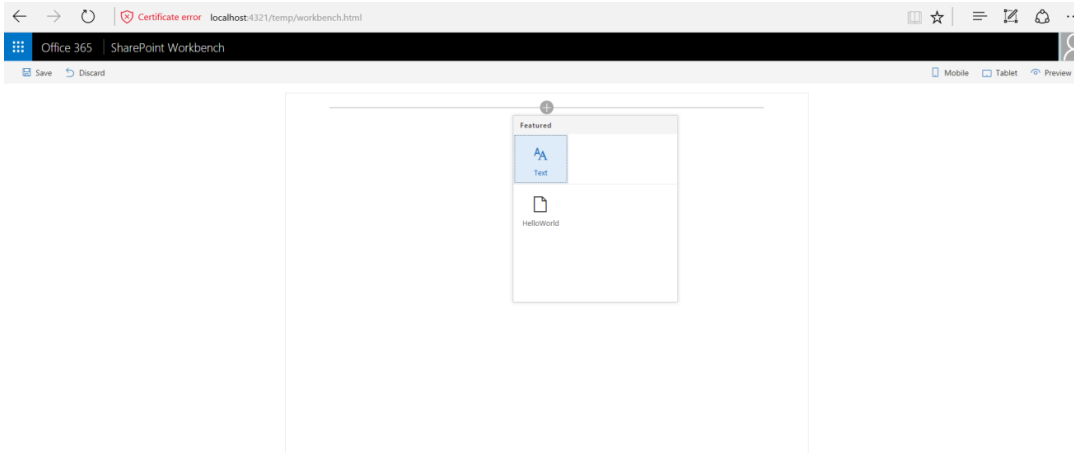
PS C:\Projects\Demo\spfx2>
```

Type the following to launch your solution inside the SharePoint Workbench!

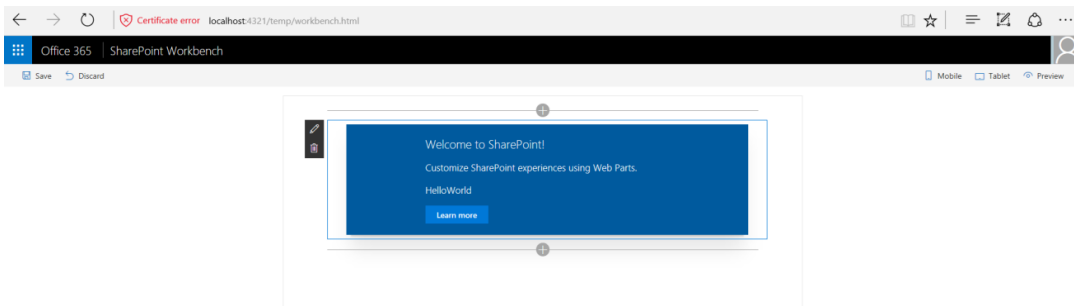
1 | gulp serve

There are a few things to note here.

First, this isn't SharePoint but it functions like it in that you can click on the plus symbol to add a new web part. You'll also see a few buttons at the top that will simulate what the page will look like on a mobile device or tablet.



Also, you may notice a certificate error and if you pay attention to the last bit of text outputted by the gulp serve command, you'll see a warning regarding this. The workbench is using https and won't load your scripts so you'll want to run 1 more gulp command which we'll do next, but first, I'll add my web part to the page.



Run the next command to fix the cert issue. The command will install a certificate. You'll also get a prompt to confirm and the next time you run gulp serve, you won't see the cert issue.

```
1 | gulp trust-dev-cert
```

There's something pretty cool about the workbench. If you have it running, you can go to your SharePoint Online and test out the page from there. Just go to your site and append `/_layouts/15/workbench.aspx`. (Example: [https://<tenant>sharepoint.com/\\_layouts/15/workbench.aspx](https://<tenant>sharepoint.com/_layouts/15/workbench.aspx))

That's what you need to get started. To recap, we ran a few 1 time installs (nodejs, npm, gulp, yeoman, and the SharePoint generator). We created a solution using the yeoman generator, and we took a look at the SharePoint Workbench.

This was meant to be a quick and dirty starter guide (mainly for me). Below are additional resources if you want to start diving deeper into some of the concepts mentioned above. In another post, I'll talk about webpack and deploy a solution.

## Additional Resources

- What is npm? (<https://docs.npmjs.com/getting-started/what-is-npm>). (Includes pretty much everything to you need to know)
- Getting Started with Yeoman (<http://yeoman.io/learning/index.html>)
- Getting Started with Gulp (<https://github.com/gulpjs/gulp/blob/master/docs/getting-started.md>)