Prototype Build Instructions

# Build Pre-Requisites

Below I a list of things that need to be on on your local machine in order to build and run the HTML prototype.

1. Install Node.JS
2. Install GIT (not to be used as source control but adds Command line interface tools that are helpful with this process). When installing, configure this to use the standard windows command line for the GIT management tool.
3. Download Code from Source Control

# Building the Project

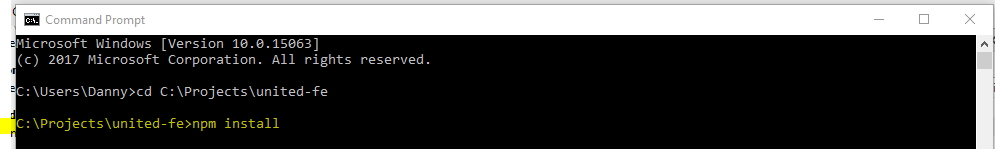
Below are the steps to follow to enable you to build the HTML source

1.) Open a Windows command line tool as Administrator.

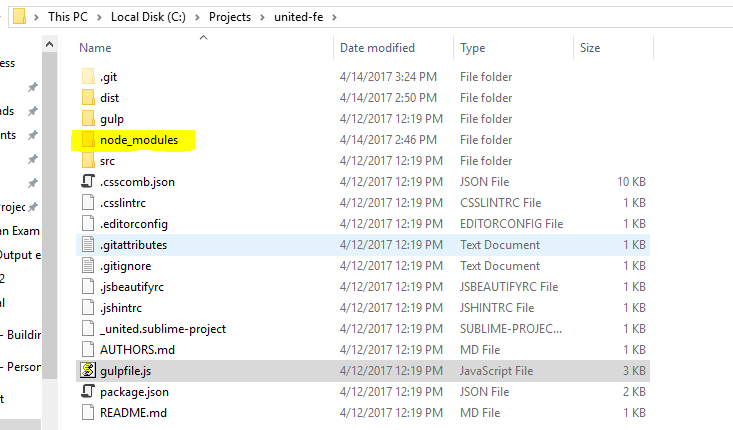
2.) Cd to the root location of your HTML checkout directory



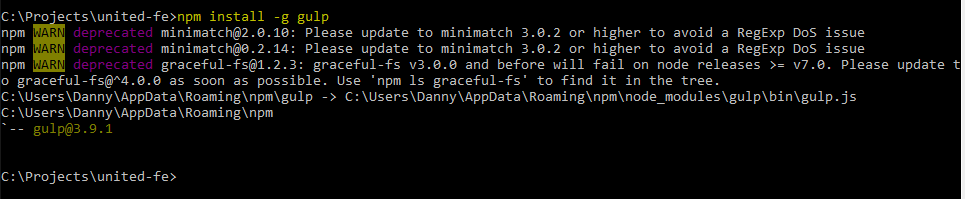
3.) Next run the command ***npm install*** as shown below. This will trigger downloads of Node modules to the root directory for use in the prototype.



You will see the new **node\_modules** folder in the root because of this action.



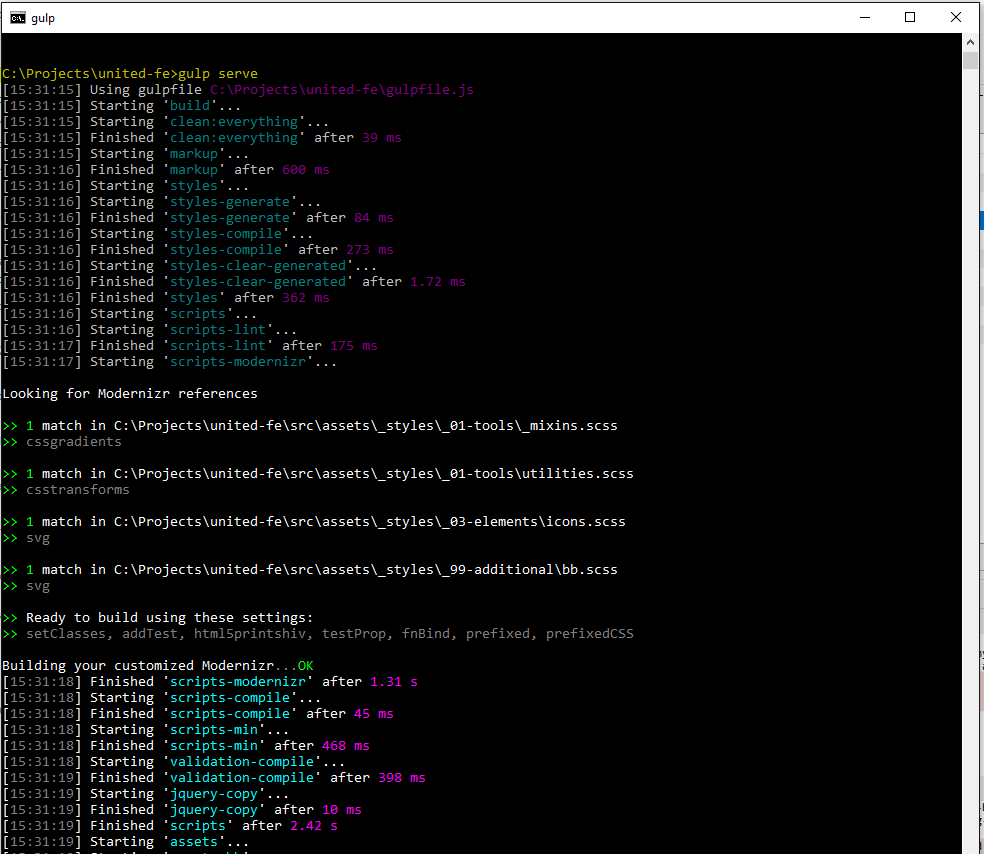
4.) Run the command: ***npm install -g gulp***



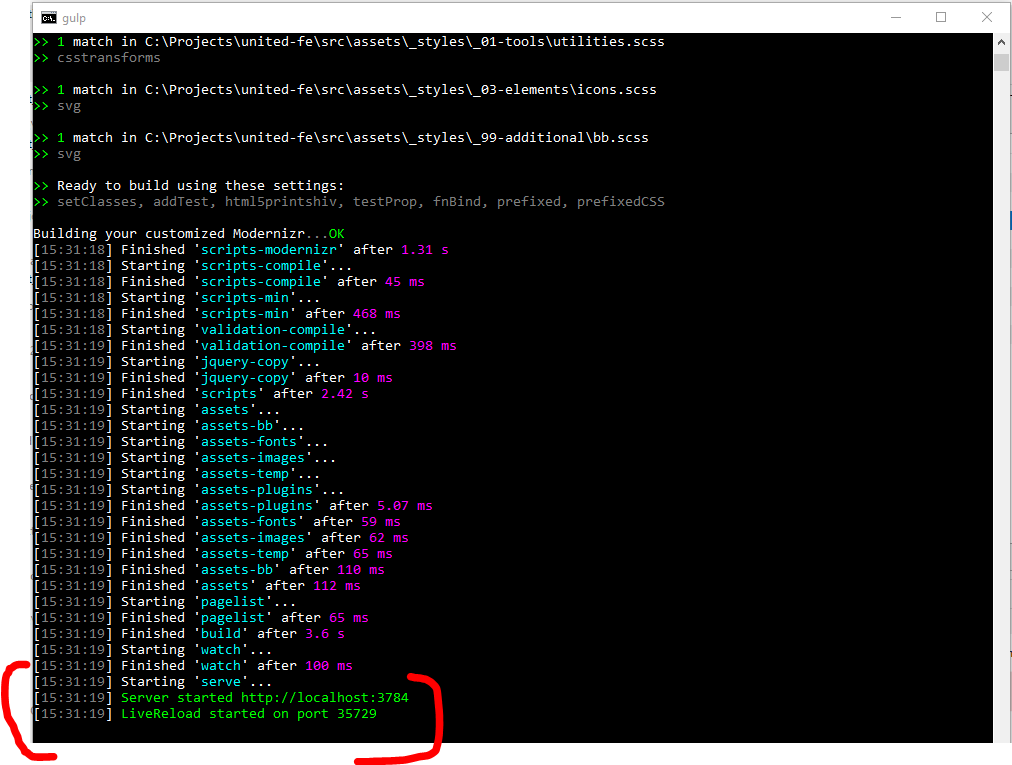
This will install the gulp CLI and other tools into your local AppData folder along with NodeJs etc.

5.) Now you can run and build the HTML prototype in a local Node Server using Gulp on your local machine. Simply type the following command: ***gulp serve***

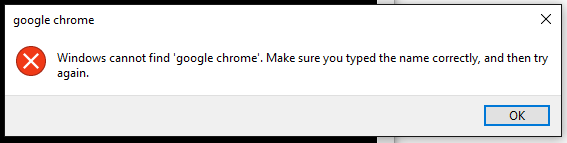
You should see something like following output example start to display as the gulp task executes.



Once it completes it will fire up a Node powered web server and the prototype will be visible at the URL displayed at the end of the gulp output.



If you are on windows you may see the following error. This is not an issue and is caused by the fact that these prototypes are often built on Macs and the Gulp command doesn’t know how to run the Chrome application with the base command that has been used.

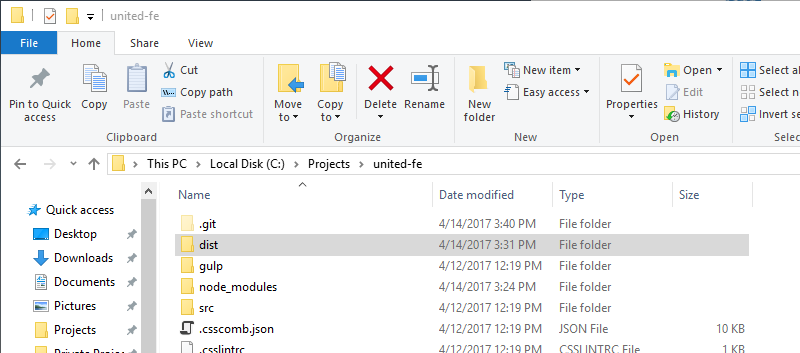


If this happens simply open your chosen web browser and browse to the URL displayed by the output of the gulp server command.

## What did this do?

This gulp task used the gulp configurations to build out compiled CSS and JavaScript as well as used the partials and handlebars templates to create full HTML pages for the HTML Demo Prototype. The command then fired up a temporary webserver using Node.

Along with the rest of the source the compiled output can now be found in a new ***dist*** folder created in the root of your checkout directory.



If you didn’t want to use the standard serve command to run the prototype you could simply point your local IIS webserver to the ***dist*** folder and the site would run just the same.

# Gulp Commands & Configuration

The gulp command above is the default to build and run the prototype. There are other additional commands available such as:

1. ***Gulp build*** – builds a local version on your local machine
2. ***Gulp production*** - does a full build, minification etc. for a production deployment. This task would be called by the

You can also have a watch command applied in the command line that will watch the checkout directory for changes you make to the root SSCS, JavaScript or HTML. When they happen, the watch will re-compile the source and update the **dist** folder versions automatically.

For example, running the command: ***gulp serve watch*** will trigger the watch to look for saves to the source code and recompile.

# References

<http://gulpjs.com/>

<https://nodejs.org/en/>

<https://git-scm.com/download/win>