**Setting up a Gulp project**

**First time only:**

1. Install node: <https://nodejs.org/en/>
   * check that node has been installed by opening a terminal and typing node -v. If it is installed then it will show you a version.
2. Install NVM (node version manager)
   * In Terminal, paste in this command: curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.33.8/install.sh | bash
   * NVM allows you to switch between different versions of Node (when you download an existing project from Git it may use a different version of node than what you have on your machine).
3. Install Gulp
   * In Terminal paste in this command: sudo npm install gulp -g

**Setting up a basic project:**

1. Download the gulp template from Github:  
   <https://github.com/News-at-Northeastern/gulp-template>  
   **Note**:  Do not upload your project to the same place in Github. Make sure to create a completely new folder for your project if you are planning to upload it to Github.
2. Rename the “gulp-template” folder to whatever you want.
3. Open the renamed project folder in Terminal by dragging it onto the Terminal shortcut
4. Install the project by typing npm install --save-dev in terminal
5. When it is done installing, type gulp and it should open index.html in a browser tab  
   (You may also need to run gulp build once to ensure that the js and css files are compiled to the dist folder)

*Note: If you get an error in the terminal saying “no command ‘gulp’ found”, it means that you don’t have gulp on your machine and you need to do that first (see step three in the first time setup steps).*

**File and folder structure:**

All of the files you need to edit are inside the /app folder. After you are done making your edits you will run a build by typing gulp build in Terminal (see below) and this will create a /dist folder in which all of the compiled/compressed CSS and JS will be located. These are the files you will eventually move into Wordpress.

/app

   /css

   /images

   /js

/library

/plugins

   /scss

   index.html

/dist

   /css

   /js

   index.html

/node\_modules

Gulpfile.js

package.json

CSS:

* If you want to add your own CSS file instead of using the SCSS file main.scss then you need to add it in the css folder and reference it in index.html inside the commented section:  
      <!--build:css css/styles.min.css-->  
          <link rel="stylesheet" href="css/main.css">  
      <!--endbuild-->  
  This will ensure that it is compiled along with main.css.  
  **IMPORTANT**: Do not add your CSS inside main.css. This file gets overwritten each time you run a build (see below)

SCSS

* If you want to use SCSS, then just add it in main.scss or create a new scss file in the /scss folder. When you make changes you will notice that the styles automatically get updated in the browser as you go.

JS

* Files that you create should go in the plugins folder and libraries such as scrollmagic or jquery should go in the library folder
* When you add additional JS files you will need to reference them in the bottom of index.html between the commented out lines:  
  <!--build:js js/main.min.js -->  
      <script src="js/plugins/scrollmagicControls.js"></script>  
  <!--endbuild-->
* The exception to this is if you are adding JS libraries (such as jQuery UI) which are added in the library folder.
* Note that the order in which you add the references sometimes matters if one JS file depends on another to run.

IMAGES

* When you add images and reference them in the HTML, JS and CSS, you will have to use the same structure as Wordpress in order for them to show up correctly on both your local machine and in Wordpress. Since we add our custom projects in the “/interactive” folder, your folder structure would have to be something like this:  
  /dist/interactive/2018/03/YourProject/images
* Browsersync, which is the function that automatically updates your browser, uses files from the “dist” folder, so when you add images in app/images, you need to make sure that they get built to the “dist” folder using the correct folder structure. Otherwise they won’t show up in your browser.
* You can tell gulp to automatically place the images in a folder with the correct folder structure by editing the images task in “gulpfile.js”. I have added a comment in the file about how to do this.

When you are done writing your JS/SCSS/CSS then you need to compile it.

* Go into Terminal and hit CTRL + C. This ends the gulp task.
* Type gulp build. This will create the /dist folder and inside it will be the compiled CSS, images and JS which you’ll use in Wordpress.
* When you upload files to Wordpress, you should use the ones that are compiled to the “dist” folder (e.g. dist/css/styles.min.css)