Designless

To quickly test the project navigate to dist/ and spin up a Python web server:

```
python -m SimpleHTTPServer
```

To install dependencies and build from the source, follow the following instructions modified from the Google Polymer setup guide.

Prerequisites

The following major dependencies are required:

- Node.js, used to run JavaScript tools from the command line.
- npm, the node package manager, installed with Node.js and used to install Node.js packages.
- gulp, a Node.js-based build tool.
- bower, a Node.js-based package manager used to install front-end packages (like Polymer).

To install dependencies:

1. Check your Node.js version.

```
node --version
```

The version should be at or above 0.12.x.

2. If you don't have Node.js installed, or you have a lower version, go to nodejs.org and click on the big green Install button.

3. Install `gulp` and `bower` globally.

```
npm install -g gulp bower
```

This lets you run `gulp` and `bower` from the command line.

4. Install the local `npm` and `bower` dependencies.

```
npm install && bower install
```

This installs the element sets (Paper, Iron, Platinum) and tools the starter kit requires to build and serve apps.

Building

Serve / watch

```
gulp serve
```

This outputs an IP address you can use to locally test and another that can be used on devices connected to your network.

Build & Vulcanize

```
gulp
```

Build and optimize the current project, ready for deployment. This includes linting as well as vulcanization, image, script, stylesheet and HTML optimization and minification.

Features

The original proposal for Designless laid out the following capabilities and their timeline:

- Markdown & JSON to layout engine (Week 6)
- Persistent drag & drop interface (Week 6-Week 10)
- Typography & image controls (Week 10-Week 14)
- Publishing & Documentation (Week 14-Week 16)

I intend to implement all the original features planned. This is possible due to my experimentation with a few technologies before embarking on the project to ensure that the

features would be realistically achieved.

MVP Features

In my original proposal I stated:

" For the Minimum Viable Product I hope to complete a prototype which can take Markdown and design structure as a JSON file and render the document layout in the browser, complete with nested textbox-like element support. "

I have created implemented this core layout and Markdown data storage and specification, the important core of the project that everything will be build on. While I haven't implemented the nested textbox support on the front end I have build the data structure code with this in mind.

Some detail into the components I have implemented to support these features:

- jQuery UI Draggable text boxes
- Persistent layout storage using Lockr.js and HTML5 localstorage
- Pixel and millimetre perfect layout storage through computing the pixel-per-millimetre ratio (layout is based in millimetres as it is for print)
- Export of documents through Google Chrome print dialogue, with perfect layout reproduction
- Live document updating through clean, callback based Markdown structure parsing pipeline (Markdown.structured() in Markdown.js)
- Material design framework derived from the Google Polymer Starter Kit, including customised Grub build chain
- Automatic bower dependency addition to index.html with grub-wiredep build chain in gulpfile.js

- Model/View/Controller separation of code:
    - Markdown.js and Layout.js manage the data structures (Model)
    - Document.js manages the view from the data
    - The DOM stores the view objects
- What-You-See-Is-What-You-Mean Markdown editor which has basic visible formatting controls, derived from SimpleMDE Markdown Editor

Dependencies and Acknowledgements

Designless at this stage makes use of the following libraries and resources for the front end (managed with bower):

- Google Polymer (Starter kit) Including the various webcomponents as part of this package (iron-*, paper-*)
- Page.js client side router
- jQuery
- jQuery UI
- SimpleMDE markdown editor
- “Marked” markdown parser & compiler
- Lockr.js local storage library

Numerous utilities are installed with npm for the gulp build chain.