FigmaConverter

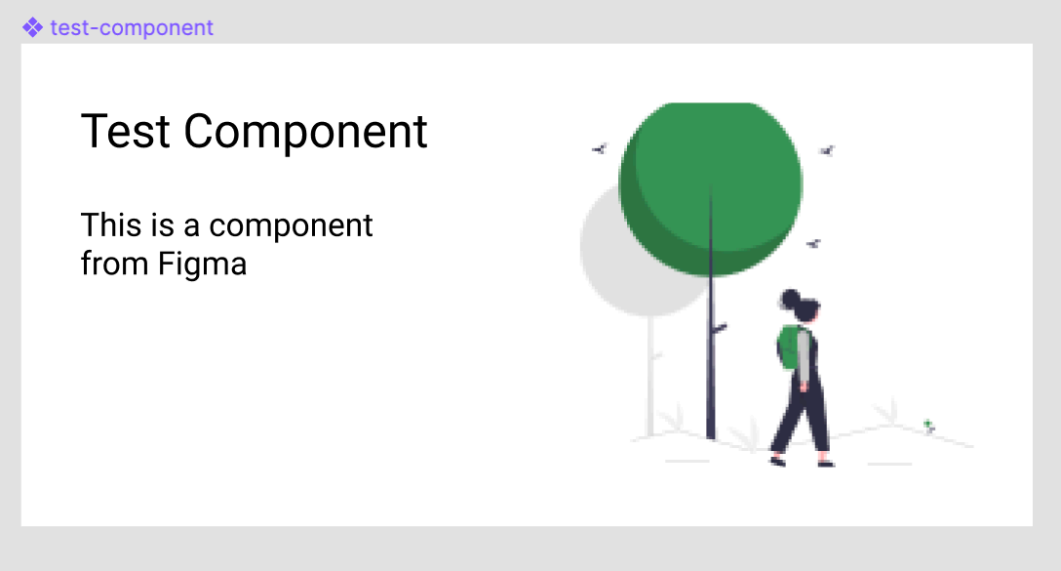
FigmaConverter is a small program that let's you convert Figma components into web-components and Figma text/color styles into SCSS variables and mixins.

Building Figma Components (Designing)

The FigmaConverter uses the components feature from Figma. To get layouts and whitespace correct FigmaConveter also uses the auto-layout feature (which is very similar to *display: flex*). The components that you want to convert must be placed on the canvas itself. (not on in another frame).

Naming elements

FigmaConverter uses the names of the elements in Figma as variable names. Therefore it's important to check that each name is correct. Because of a dependency in web-components, the components must be named with two lowercased words with a dash between them.
Example: "test-component."



Keep in mind that texts should be renamed as below. If this is not done the variable name becomes very long and may crash the styling of the component making it unusable.

Installation (Developing)

Clone the repository from GitHub to install figmaConverter.

```
git clone https://github.com/AlbinFrick/FigmaConverter.git
```

Change directory and run npm install to install all necessary packages to run the program.

```
cd FigmaConverter/
npm install
```

Dependencies

- LitElement
- Any sort of bundler that handles open imports, eg. import 'litElement'
    - Some examples are Webpack, Parcel, Rollup.

The system now relies on CSS *gap* (previously grid-gap) -- Which does not currently work in Safari.

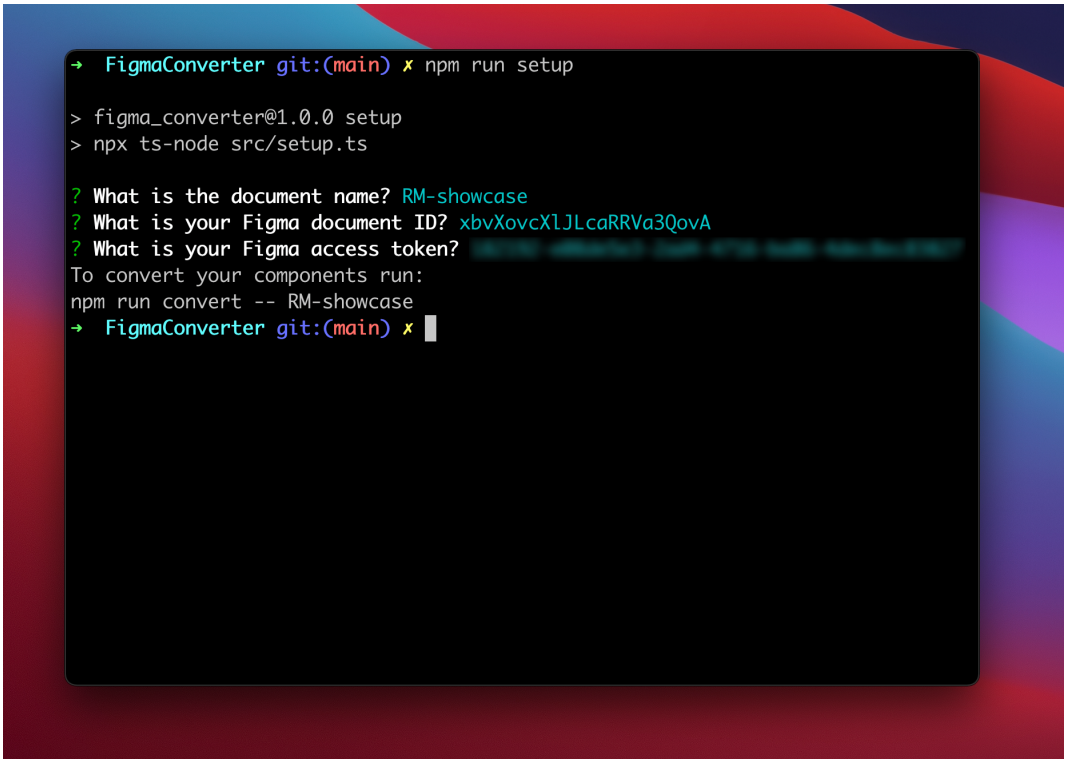
Setup

FigmaConverter focuses on single documents. To be able to get the information from the document an access token is needed. Run the setup script to fill in the information for the Figma document.

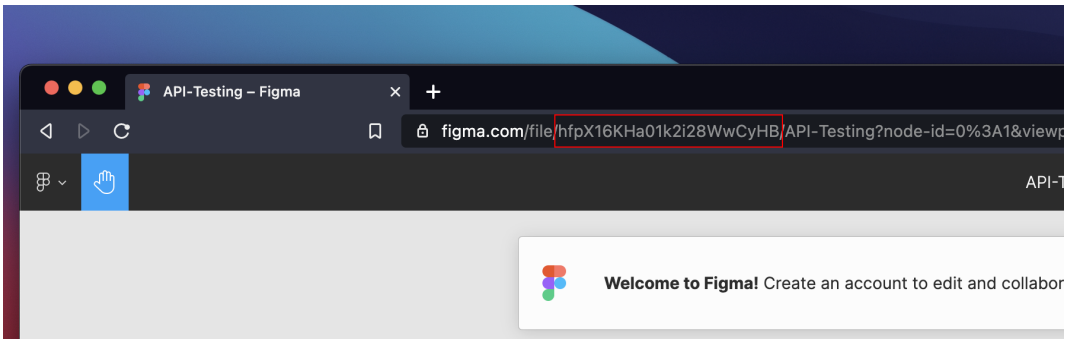
```
npm run setup
```

Three questions are asked:

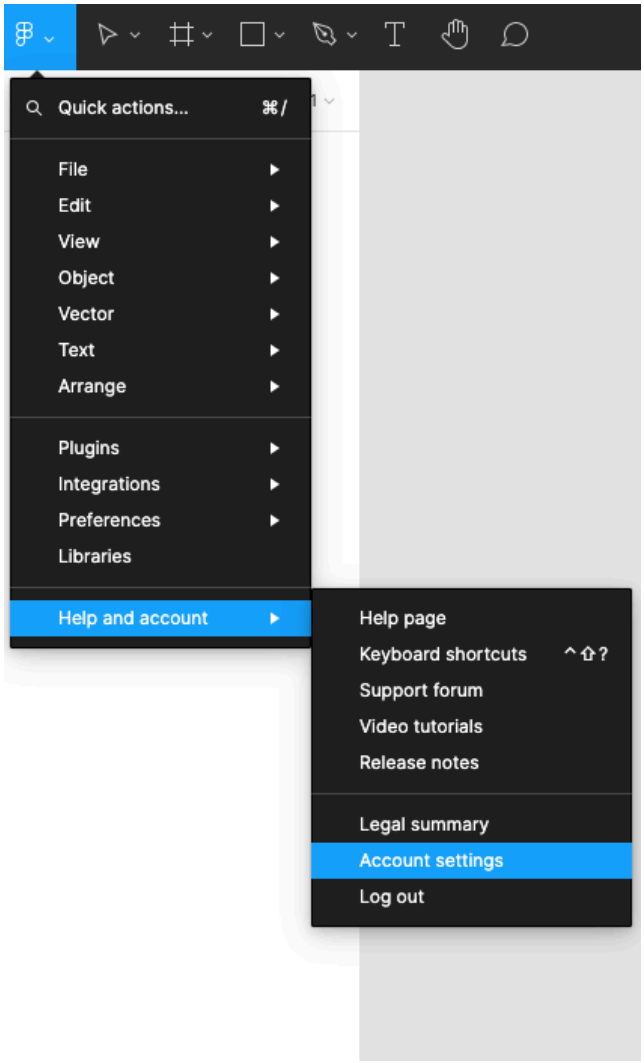
The document name is what you will refer to when running the program. This could be anything that you'll associate with your document.

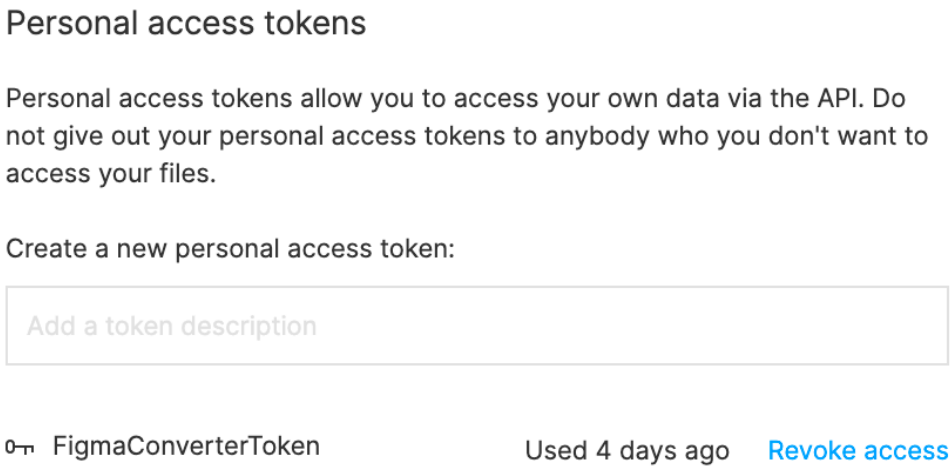


The document ID can be found in the URL to your Figma document (see image below).



The access token can be generated under the account settings tab in Figma.





The configuration is set to a .env-file and read by the figmaConverter.ts-file.

The setup can be run multiple times and to add more than one document. After the first run, you'll be asked if you want to change your access token.

Usage

To run FigmaConverter run the figmaConvert.ts file:

```
npm run convert -- DOCUMENT_NAME
```

When the program has run it will result in some new files in the ***output/*** folder. These are TypeScript files that contain litElements for each Figma component.

These components files can be used directly as is but will most likely be converted to .js-files. To do this with every .ts-file in the output folder:

```
cd output/
tsc
```

Distribute

With these litElement files, an additional package.json file is created. This can be used to distribute the components to NPM.

To publish to npm you can log in to your npm account.

```
npm login
```

Follow the instructions and then use:

```
npm publish
```

Test

If you wish to test your components locally before you publish them navigate to your ***output/*** folder and type.

```
npm link
```

Then navigate to your local project and type

```
npm link YOUR-PACKAGE-NAME
```

Using the Components

The generated LitElement components are as a base a web-component therefore the components can be used in native HTML5 and in any framework. To use the components import them in a JavaScript-file:

```
import 'YOUR-PACKAGE-NAME'
```

Then use the components in HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Document</title>
</head>
<body>
  <my-component></my-component>
  <script src="./js/main.js"></script>
</body>
</html>
```

Styling

If you wish to style your components after they've been made you can do that by targeting the name of the element you wish to change and then writing CSS to a string.

```
  <test-component
    textBox="
      background-color: red;
      border-radius: 15px;
    "
    title="
      font-family: Roboto;
      font-size: 2em;
    "
  >
  </test-component>
```

Texts

Often components are built as a template where the content is different for each instance of the components. To change the text for a component the name of the element is targeted with the addition of the keyword “Text.” Example:

```
  <test-component
    titleText="This is a new title"
  >
  </test-component>
```

Images

As of now the images that are used are implemented as URLs from Figmas API.
IMPORTANT
These URLs are only available for 14 days. To use these images permanently as of now is to set them as the background image of the element.

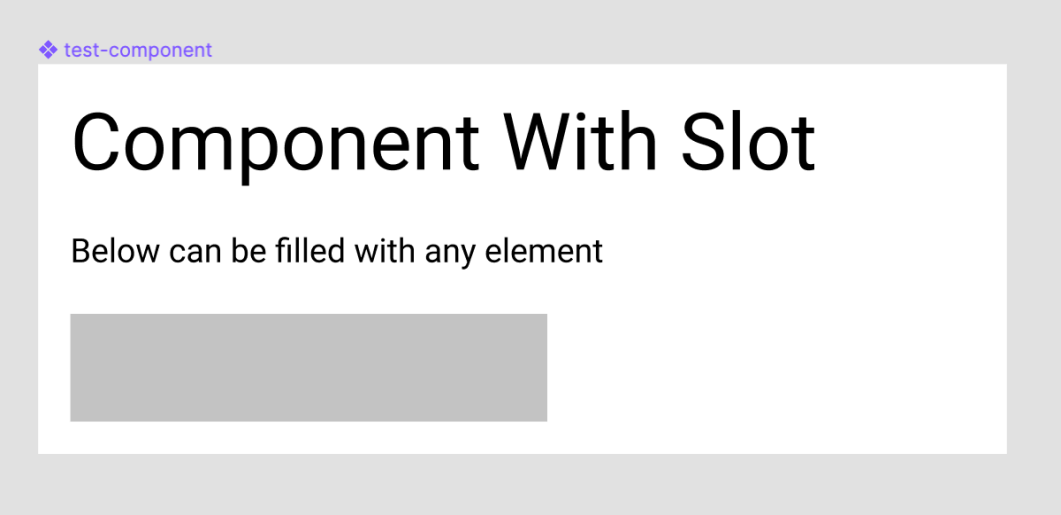
```
<test-component
  img="background-image: url(path/to/your/image)"
>
</test-component>
```

Slots

Slots can be used to insert elements into your web-component. To use a slot with figmaConverter you first need to mark where you want the slot to be. This can be done with any element in Figma but needs to have the keyword "SLOT" before the name.

Example

Here a rectangle is used to fill the slots space.



The slot name is slotName.

To use the slot in your project you can add any element in the component with the slot attribute matching the name from Figma. Here slotName.

```
<test-component>
  <button slot="slotName">Button</button>
</test-component>
```

Contributing

Pull requests are welcome. For major changes, please open an issue first to discuss what you would like to change.

License

MIT

#