Web Components

a chance to

Create the future
I’m from Brazil
Create!

Share!

Learn!
But there was no starter-kit =/
GitHub repository for polymer-boilerplate

Replace repository name placeholder

- **zenorocha** authored a day ago

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>dist</td>
<td>Release v0.1.3</td>
<td>2 days ago</td>
</tr>
<tr>
<td>src</td>
<td>Make src more vanilla</td>
<td>2 days ago</td>
</tr>
<tr>
<td>.editorconfig</td>
<td>Insert final newline</td>
<td>a month ago</td>
</tr>
<tr>
<td>.gitignore</td>
<td>Stop to commit bower_components</td>
<td>2 days ago</td>
</tr>
<tr>
<td>Gruntfile.js</td>
<td>Replace repository name placeholder</td>
<td>a day ago</td>
</tr>
<tr>
<td>README.md</td>
<td>Change development instructions from <code>&lt;ol&gt;</code> to <code>&lt;ul&gt;</code></td>
<td>2 days ago</td>
</tr>
<tr>
<td>bower.json</td>
<td>Remove authors and update description</td>
<td>2 days ago</td>
</tr>
<tr>
<td>index.html</td>
<td>Change titles to &quot;my-repo&quot;</td>
<td>2 days ago</td>
</tr>
<tr>
<td>package.json</td>
<td>Add deploy task to send files to gh-pages branch</td>
<td>2 days ago</td>
</tr>
</tbody>
</table>

Link to repository: [github.com/webcomponents/polymer-boilerplate](https://github.com/webcomponents/polymer-boilerplate)
### Fork this repo if you want to start your own Web Component using X-Tag — Edit

- **27 commits**
- **2 branches**
- **1 release**
- **1 contributor**

**branch: master**

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>dist</td>
<td>Release v0.1.0</td>
<td>2 days ago</td>
</tr>
<tr>
<td>src</td>
<td>Add params to attributeChanged()</td>
<td>2 days ago</td>
</tr>
<tr>
<td>.editorconfig</td>
<td>Initial boilerplate based on element-boilerplate</td>
<td>a month ago</td>
</tr>
<tr>
<td>.gitignore</td>
<td>Stop to commit bower_components on master</td>
<td>2 days ago</td>
</tr>
<tr>
<td>Gruntfile.js</td>
<td>Replace repository name placeholder</td>
<td>a day ago</td>
</tr>
<tr>
<td>README.md</td>
<td>Change development instructions from &lt;ol&gt; to &lt;ul&gt;</td>
<td>2 days ago</td>
</tr>
<tr>
<td>bower.json</td>
<td>Update &quot;platform&quot; dependency to v0.2.3</td>
<td>a day ago</td>
</tr>
<tr>
<td>index.html</td>
<td>Change titles to &quot;my-repo&quot;</td>
<td>2 days ago</td>
</tr>
<tr>
<td>package.json</td>
<td>Add deploy task to send files to gh-pages branch</td>
<td>2 days ago</td>
</tr>
</tbody>
</table>

[Read more about GitHub repositories](https://github.com/webcomponents/x-tag-boilerplate)
Fork this repo if you want to start your own Web Component using VanillaJS — Edit

30 commits
2 branches
1 release
2 contributors

Replace repository name placeholder

zenorocha authored a day ago  latest commit d25b6d254b

- src
  - Remove unnecessary comment
    - 2 days ago

- .editorconfig
  - Initial boilerplate based on element-boilerplate
    - A month ago

- .gitignore
  - Stop to commit bower_components
    - 2 days ago

- Gruntfile.js
  - Replace repository name placeholder
    - A day ago

- README.md
  - Change development instructions from <ob> to <ub>
    - 2 days ago

- bower.json
  - Update "platform" dependency to v0.2.3
    - A day ago

- index.html
  - Change titles to "my-repo"
    - 2 days ago

- package.json
  - Add deploy task to send files to gh-pages branch
    - 2 days ago

<my-repo>

github.com/webcomponents/element-boilerplate
What’s included?

We encourage a series of best practices like setting up a live demo, documenting your API, maintaining a changelog for release purposes, and sharing it in a package manager.
A Yeoman generator for Custom Elements https://npmjs.org/generator-element — Edit

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
<th>Last Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>README.md</td>
<td>Add new generator for individual elements</td>
<td>a day ago</td>
</tr>
<tr>
<td>app</td>
<td>Add new generator for individual elements</td>
<td>a day ago</td>
</tr>
<tr>
<td>elements</td>
<td>Add new generator for individual elements</td>
<td>a day ago</td>
</tr>
<tr>
<td>test</td>
<td>Update template to use polymer-boilerplate v0.1.4</td>
<td>2 days ago</td>
</tr>
<tr>
<td>.editorconfig</td>
<td>Standardize indentation and other stuff according to the pattern we’re…</td>
<td>a month ago</td>
</tr>
<tr>
<td>.gitignore</td>
<td>Adding initial yo generator output</td>
<td>2 months ago</td>
</tr>
<tr>
<td>.jshintrc</td>
<td>Adding initial yo generator output</td>
<td>2 months ago</td>
</tr>
<tr>
<td>.travis.yml</td>
<td>Adding initial yo generator output</td>
<td>2 months ago</td>
</tr>
<tr>
<td>README.md</td>
<td>Add new generator for individual elements</td>
<td>a day ago</td>
</tr>
<tr>
<td>banner.js</td>
<td>Add new generator for individual elements</td>
<td>a day ago</td>
</tr>
<tr>
<td>package.json</td>
<td>Add new generator for individual elements</td>
<td>a day ago</td>
</tr>
</tbody>
</table>

github.com/webcomponents/generator-element
Create!

Share!

Learn!
So people started to **create elements**
Example

Basic Usage

This example is a simple way to initialize the user browser camera and start tracking for objects with color magenta.

There are two available callbacks, onFound is fired when the object is detected and onNotFound does the opposite.

```javascript
var videoCamera = new tracking.VideoCapture().render().renderVideoCanvas();
videoCamera.track({
  type: 'color',
  color: 'magenta',
  onFound: function(track) {
    console.log(track.x, track.y, track.z);
  },
});
```
<SF-HTML5>
We needed a place to show these elements
Got some friends together...
Custom Elements
a web components gallery for modern web apps

What is Web Components?
Web Components are a collection of standards which are working their way through the W3C. They enable truly encapsulated and reusable components for the web. And if you think HTML5 changed the web, wait to see what Web Components will do.

For lots more information about it, including articles and presentations, check out webcomponents.github.io.

— Zeno Rocha, project lead.

Submit your own
Got a great idea for a custom element? Awesome! There are boilerplates for Polymer, X-Tag, and VanillaJS that you can fork and get up and running with a simple component.

When you're ready to go, submit it to the form below and it'll appear on this site for others to play and use!

Submit

Latest elements
gravatar-tag
Custom Element for Gravatar without Polymer.

Most popular elements
x-gif
A custom element for flexible GIF playback
CustomElements.io

+ 170 elements
+ 100,000 pageviews

North America: 24%
Southern/Western Europe: 14%
Eastern/Northern Europe: 13%
South America: 13%
How to add an element there?
{  
  "name": "my-repo",
  "version": "0.0.0",
  "description": "My awesome Custom Element",
  "license": "MIT",
  "main": "src/my-element.html",
  "keywords": [  
    "web-components"
  ],
  "ignore": [  
    "**/.*",
    "node_modules",
    "bower_components"
  ],
  "dependencies": {  
    "platform": "Polymer/platform#~0.2.3"
  }
}
What’s the main reference to learn Web Components?
And a bunch of awesome people...

Alex Komoroske
Briza Bueno
Daniel Buchner
Pascal Precht
Eric Bidelman
Bernard De Luna
Soledad Penadés
Bruce Lawson
Rob Dodson
Eduardo Lundgren
Sindre Sorhus
Mathias Bynens
WebComponents.org
the web platform specs that will change the way you build apps

WHAT?
WebComponents.org is where pioneers and community-members of the Web Components ecosystem (like Polymer, X-tags, and other interested parties) document web components best practices so that others can follow the same path instead of needlessly striking out on their own.

SPECs

WEB COMPONENTS
This document is a non-normative reference, which provides an overview of Web Components. It summarizes the normative information in the respective specifications in easy-to-digest prose with illustrations.

CUSTOM ELEMENTS
This specification describes the method for enabling the author to define and use new types of DOM elements in a document.

HTML IMPORTS
HTML Imports are a way to include and reuse HTML documents in other HTML documents.

ARTICLES
WEB COMPONENTS BEST PRACTICES
Web Components (WC) are a new set of web platform features that enable developers to build applications in a declarative, composable way. The following is an initial list of best practices we advocate component authors consider to ensure their elements are good citizens in the Web Component ecosystem.
Gather **best practices**

Introduce **useful tools and resources**

Highlight great **articles & talks**

Discuss **API implementations**
Create!
Share!
Learn!
The best way to predict the future is to create it.
Thank you!

Zeno Rocha
zenorocha.com

bit.ly/wc-mailing
#webcomponents
@webcomponents
+WebComponents
Stickers \o/