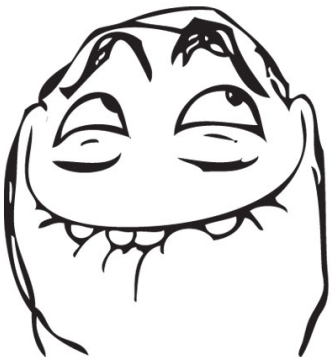


2.1.8

Semantic UI

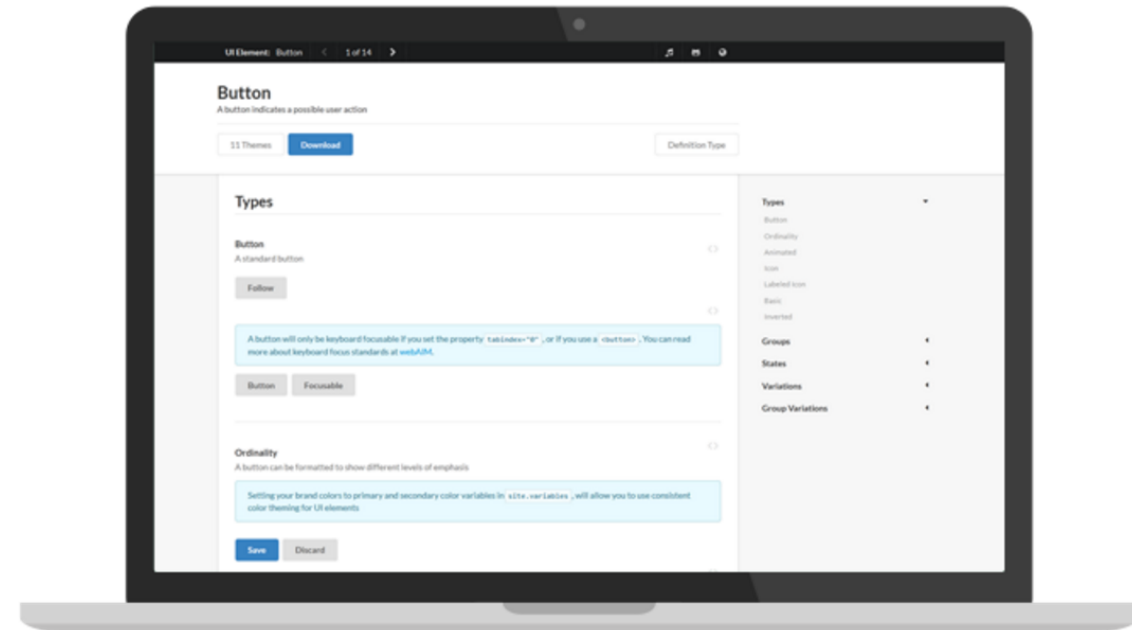
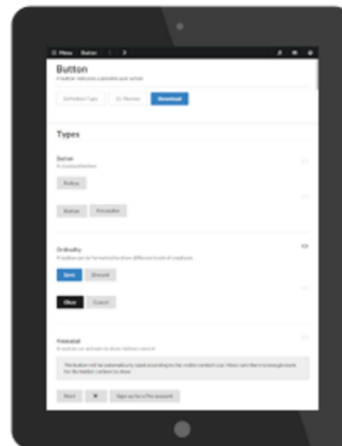
User Interface is the language of the web



~~Not Bootstrap~~

What is Semantic UI?

- Framework providing basic front-end styling and functionality
 - Uses Html, CSS, Javascript
- Uses a Grid System
- Uses JQUERY
- Mobile Friendly!



What are other options?

- Foundation and Bootstrap are also great options
- Quite large in comparison
- Materialize mimics Android UI
- Still in its infancy stage



FOUNDATION 3



So why are we going with Semantic UI?

Semantic UI

- Smaller project to work with
- Better organized (docs, tutorials)
- Friendliest for novices
- Syntactically light
- Easier to fully understand internals
- Style is newer and “edgy”

Bootstrap

- Very large project
- Lots of various files to remember to include in your project
- Docs are less novice friendly
- Syntactically heavier by far
- Looks a bit dated



What can Semantic UI do?

Animations

- Popups
- Dialogues
- Hovers
- etc..

Styling

- Grid System
- Automatic Mobile View Adjusting
- Colour variables/Fixed styles to use
- etc..

Form Checking

- Bad input
- Error messages

Clean API



Concise HTML

Semantic UI treats words and classes as exchangeable concepts.

Classes use syntax from natural languages like noun/modifier relationships, word order, and plurality to link concepts intuitively.

Get the same benefits as [BEM](#) or [SMACSS](#), but without the tedium.

Three nicely styled buttons with 5 lines of code

```
<div class="ui three buttons">  
  <button class="ui active button">One</button>  
  <button class="ui button">Two</button>  
  <button class="ui button">Three</button>  
</div>
```

One

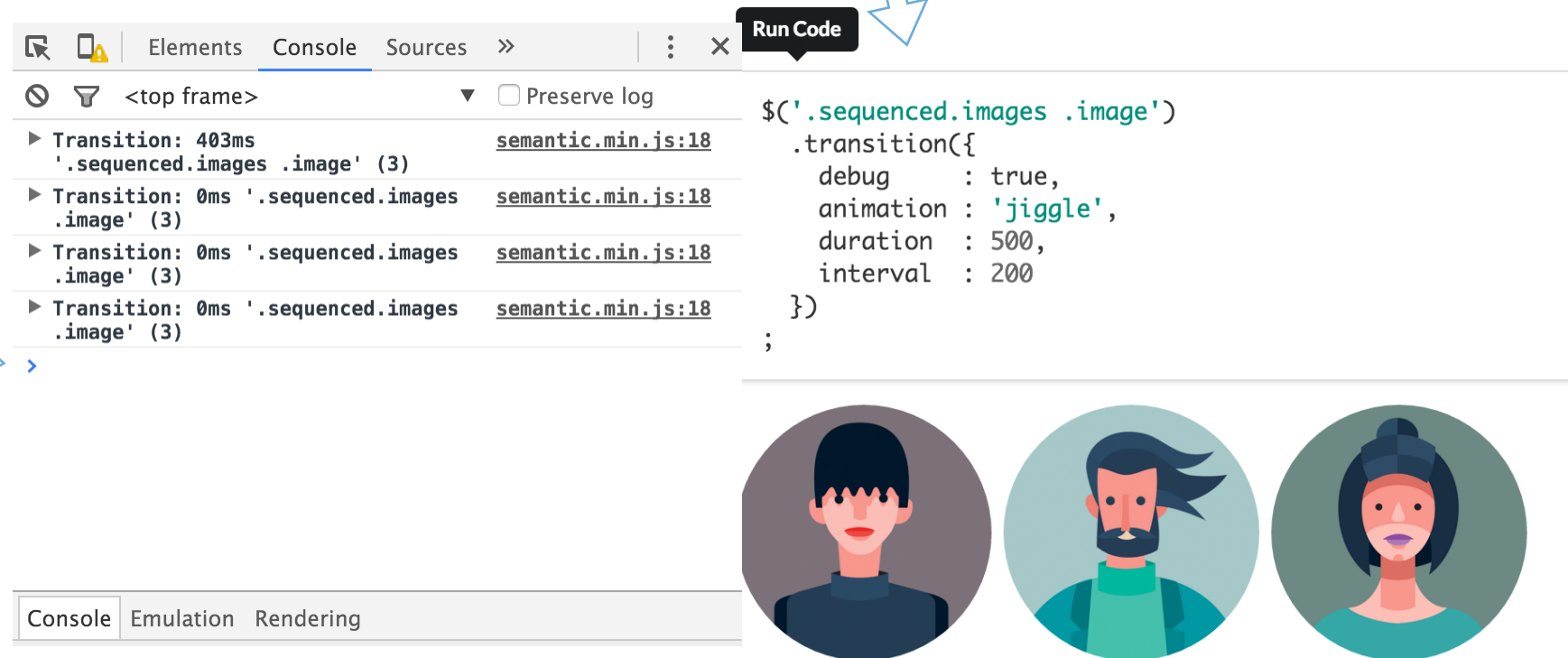
Two

Three

Refined Debugging

Performance logging lets you track down bottlenecks without digging through stack traces.

With debug set to true, relevant output is shown in the browser console



The screenshot shows a browser's developer console with the 'Console' tab selected. The console displays four performance log entries, each showing a transition time of 0ms for the '.sequenced.images .image' (3) selector. The first entry shows a transition time of 403ms. The code snippet on the right shows a jQuery selector `$('.sequenced.images .image')` followed by a `.transition()` method call with options: `debug : true`, `animation : 'jiggle'`, `duration : 500`, and `interval : 200`. A blue arrow points from the text 'Performance logging lets you track down bottlenecks without digging through stack traces.' to the 'Run Code' button. Another blue arrow points from the text 'With debug set to true, relevant output is shown in the browser console' to the console output.

Run Code

<top frame> ☐ Preserve log

- ▶ Transition: 403ms `' .sequenced.images .image' (3)` [semantic.min.js:18](#)
- ▶ Transition: 0ms `' .sequenced.images .image' (3)` [semantic.min.js:18](#)
- ▶ Transition: 0ms `' .sequenced.images .image' (3)` [semantic.min.js:18](#)
- ▶ Transition: 0ms `' .sequenced.images .image' (3)` [semantic.min.js:18](#)

```
$('.sequenced.images .image')  
  .transition({  
    debug      : true,  
    animation  : 'jiggle',  
    duration   : 500,  
    interval   : 200  
  })  
;
```

Console Emulation Rendering



Buttons

<http://semantic-ui.com/elements/button.html>

HTML Code

Example



Save

Discard

```
<button class="ui primary button">  
  Save  
</button>  
<button class="ui button">  
  Discard  
</button>
```

LoadingButtons

<http://semantic-ui.com/elements/button.html>

HTML Code

Very useful when waiting for GET/POST request to return!

Example



```
<button class="ui loading button">Loading</button>  
<button class="ui basic loading button">Loading</button>  
<button class="ui primary loading button">Loading</button>  
<button class="ui secondary loading button">Loading</button>
```

Input

<http://semantic-ui.com/elements/input.html>

Upon Entry

```
<div class="ui input">  
  <input type="text" placeholder="Search...">  
</div>
```

After Entry

```
<div class="ui input error">  
  <input type="text" placeholder="Search...">  
</div>
```

Lists

<http://semantic-ui.com/elements/list.html>



Semantic UI



New York, NY



jack@semantic-ui.com



semantic-ui.com

Horizontally Aligned

Notice the trend with “ui <insert element name here>”

Vertically Aligned



Top Aligned



Middle



Bottom

```
<div class="ui horizontal list">
```

```
<div class="ui list">
  <div class="item">
    <i class="users icon"></i>
    <div class="content">
      Semantic UI
    </div>
  </div>
  <div class="item">
    <i class="marker icon"></i>
    <div class="content">
      New York, NY
    </div>
  </div>
</div>
```

Dropdown

<http://semantic-ui.com/elements/dropdown.html>

This won't work immediately though ...

```
▼ <div class="ui selection dropdown gpa"
  tabindex="0">
  <input type="hidden" name="gpa" id="gpa"
    value="1">
  ▶ <i class="dropdown icon">...</i>
  <div class="text">70-79</div>
  ▶ <div class="menu" tabindex="-1">...</div>
```

Grade Average

70-79 ▼

 90-100

 80-89

 **70-79**

 60-69

 <60

JQUERY initialization

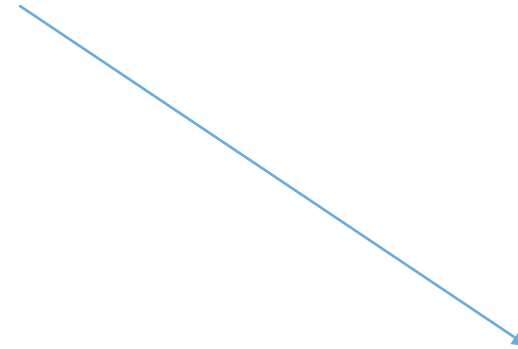
- **Modules**

- For things like dropdowns, accordions, dimmers you need to initialize them

For a dropdown we do:

```
$('.ui.dropdown')  
  .dropdown()  
;
```

The Settings tab provides an API in which we can pass parameters/settings through to the dropdown upon initialization



Definition	Examples	Usage	Settings
------------	----------	-------	----------

Plethora of Icons

<http://semantic-ui.com/elements/icon.html>

Icons can represent types of people

Example



Child



Doctor



Handicap



Spy



Student



User



Users

```
<i class="child icon"></i>
<i class="doctor icon"></i>
<i class="handicap icon"></i>
<i class="spy icon"></i>
<i class="student icon"></i>
<i class="user icon"></i>
<i class="users icon"></i>
```

Theming

<http://semantic-ui.com/elements/theming.html>

- You can create your own themes by instructions below
- Or you can download themes and use/manipulate them to your liking

Packaged Theme

src > themes >  theme-name

Packaged themes are themes bundled together in a folder for distribution. These can be downloaded from the internet, or carried between projects.

Packaged themes can be used by modifying values in your `theme.config` file.

