





# LARGE CSS FILES

increased complexity

# CSS 2.0

the good 'ol days

```
.selection {  
    font-size: 12px;  
    line-height: 13px;  
    font-family: Arial, Helvetica, sans-serif;  
    text-align: center;  
    margin-top: 10px;  
}  
.selection ul, li {  
    font-size: 12px;  
    line-height: 14px;  
    font-family: Arial, Helvetica, sans-serif;  
    text-align: left;  
    vertical-align: top;  
    margin-left: 10px;  
    width: auto;  
    height: auto;  
}
```

# TABLE-LESS & CSS3

things got much more complicated

```
/* GLOBALHEADER */
#globalheader { position:relative; display:block; width:980px; height:36px; margin:
18px auto; text-align:left; z-index:9998; background:url(/global/nav/images/
globalheader.png) repeat-x;
    -khtml-border-radius:4px; -ms-border-radius:4px; -o-border-radius:4px; -moz-border-
radius:4px; -webkit-border-radius:4px; border-radius:4px;
    -khtml-box-shadow:rgba(0,0,0,0.3) 0 1px 2px; -ms-box-shadow:rgba(0,0,0,0.3) 0 1px
2px; -o-box-shadow:rgba(0,0,0,0.3) 0 1px 2px; -moz-box-shadow:rgba(0,0,0,0.3) 0 1px
2px; -webkit-box-shadow:rgba(0,0,0,0.3) 0 1px 2px; box-shadow:rgba(0,0,0,0.3) 0 1px
2px;
    font:12px/18px "Lucida Grande", "Lucida Sans Unicode", Helvetica, Arial, Verdana,
sans-serif;
}
@media only screen and (max-device-width:768px) {
    #globalheader { z-index:1; }
}
#globalheader #globalnav { position:relative; display:block; margin:0; padding:0;
zoom:1; z-index:3; }
#globalheader #globalnav:after { content:"."; display:block; height:0; clear:both;
visibility:hidden; }
#globalheader #globalnav li { display:inline; }
#globalheader #globalnav li a { float:left; width:102px; height:36px; overflow:hidden;
```



**crap ton  
of CSS**

# TABLE-LESS & CSS3

things got much more complicated

```
/* GLOBALHEADER */
```

```
#globalheader { position:relative; display:
```

```
18px auto; text-align:left; z-index:
```

```
globalheader.png) repeat-x:
```

```
-khtml-border-radius:
```

```
radius:4px; -webkit-
```

```
-khtml-
```

LET'S FIND

a

BETTER WAY

crap ton  
of CSS

```
#globalnav { position:relative; display:block; margin:0; padding:0;
```

```
z-index:100; }
```

```
#globalheader #globalnav:after { content:"."; display:block; height:0; clear:both;
```

```
visibility:hidden; }
```

```
#globalheader #globalnav li { display:inline; }
```

```
#globalheader #globalnav li a { float:left; width:102px; height:36px; overflow:hidden;
```



# CSS PRE-PROCESSORS

Sass, LESS, etc ...



# NEW IDEAS ...

```
/* GLOBAL VARIABLES */
$prefix_defaults: -moz- -webkit- -o- -ms- '' !default;
$webkit_support: -webkit- '' !default;
$moz_support: -moz- '' !default;
$ms_support: -ms- '' !default;
$moz_webkit_support: -moz- -webkit- '' !default;
$moz_ms_support: -moz- -ms- '' !default;
$webkit_ms_support: -webkit- -ms- '' !default;
```

**variables**



```
$grid-width: 980px;
$max-device-width: 768px;
$background-position-full: 100%;
$background-position-half: 50%;
$link-color: #7F7F7F;
$text-color: #333;
```

**mixins**



```
/* Mixins */
@mixin border_radius ($border_radius, $prefixes: $webkit_support) {
  @each $prefix in $prefixes {
    #{$prefix}border-radius: $border_radius;
  }
}

@mixin box_shadow ($shadow_color: $shadow_color, $shadow: $shadow, $prefixes:
$prefix_defaults) {
  @each $prefix in $prefixes {
    #{$prefix}box-shadow: $shadow_color $shadow;
  }
}
```

```
@mixin clearfix () {
```

# OLD HABITS ...

```
/* GLOBALHEADER */
#globalheader { position:relative; display:block; width:$grid-width; height:36px; margin:
18px auto; text-align:left; z-index:999; background:url(/global/nav/images/
globalheader.png) repeat-x;
    -khtml-border-radius:4px; -ms-border-radius:4px; -o-border-radius:4px; -moz-border-
radius:4px; -webkit-border-radius:4px; border-radius:4px;
    -khtml-box-shadow:rgba(0,0,0,0.3) 0 1px 2px; -ms-box-shadow:rgba(0,0,0,0.3) 0 1px 2px;
-o-box-shadow:rgba(0,0,0,0.3) 0 1px 2px; -moz-box-shadow:rgba(0,0,0,0.3) 0 1px 2px; -
webkit-box-shadow:rgba(0,0,0,0.3) 0 1px 2px; box-shadow:rgba(0,0,0,0.3) 0 1px 2px;
    font:12px/18px "Lucida Grande", "Lucida Sans Unicode", Helvetica, Arial, Verdana,
sans-serif;
}
@media only screen and (max-device-width:$max-device-width) {
    #globalheader { z-index:1; }
}
#globalheader #globalnav { position:relative; display:block; margin:0; padding:0; zoom:1;
z-index:3; }
#globalheader #globalnav:after { content:"."; display:block; height:0; clear:both;
visibility:hidden; }
#globalheader #globalnav li { display:inline; }
#globalheader #globalnav li a { float:left; width:102px; height:36px; overflow:hidden;
text-indent:-9999px; background:url(/global/nav/images/globalnav.png) no-repeat;
cursor:pointer; }

/* VML FOR IE */
v\:roundrect,
v\:fill { behavior:url(#default#VML); }
#globalheader-roundrect,
#globalheader-fill { display:block; position:absolute; width:; height:35px; top:0; left:0;
z-index:2; }
#globalheader-shadow { display:block; position:absolute; width:$grid-width; height:35px;
top:-1px; left:-2px; z-index:1; -ms-
filter:"progid:DXImageTransform.Microsoft.Blur(pixelRadius=2)";
filter:progid:DXImageTransform.Microsoft.Blur(pixelRadius=2); }

/* VARIANTS */
#globalheader #globalnav.tabs6-nosearch li a { width:163px; background-image:url(/global/
```



# OLD HABITS ...

```
/* GLOBALHEADER */
#globalheader { position:relative; display:block; width:$grid-width; height:36px; margin:
18px auto; text-align:left; z-index:1; background:url(/global/nav/images/
globalheader.png) repeat-x;
  -khtml-border-radius:4px; border-radius:4px; -moz-border-
radius:4px; -webkit-border-ra
  -khtml-box-shadow:rgba(0,0,0,0.3) 0 1px 2px;
  -o-box-shadow:rgba(0,0,0,0.3) 0 1px 2px; -
webkit-box-shadow:rgba(0,0,0
  font:12px/18px "Lucida G
sans-serif;
}
@media only screen and (max
  #globalheader { z-index
}
#globalheader #globalnav {
  z-index:3; }
#globalheader #globalnav:a
visibility:hidden; }
#globalheader #globalnav
#globalheader #globalnav
text-indent:-9999px; back
cursor:pointer; }

/* VML FOR IE */
v\:roundrect,
v\:fill { behavior:url(#
#globalheader-roundrect
#globalheader-fill { di
z-index:2; }
#globalheader-shadow { display:block;
top:-1px; left:-2px; z-index:1; -ms-
filter:"progid:DXImageTransform.Microsoft.Blur(pixelRadius=2)";
filter:progid:DXImageTransform.Microsoft.Blur(pixelRadius=2); }

/* VARIANTS */
#globalheader #globalnav.tabs6-nosearch li a { width:163px; background-image:url(/global/
```



# LETS MAKE IT BETTER?

early days of file management



mixins.scss



variables.scss



forms.scss



typography.scss



style.scss



LETS MAKE IT BETTER?

close...



but no cigar.

Worth 1000.com  
ICANHASCHEEZBURGER.COM

style.scss

# CONTROLLER/ACTION BASED STYLES

seemed like a good idea



# THE MVC

file structures of inspiration



app/



models/



views/



controllers/



admin/



blog/



shared/

# DUPLICATE CODE == BAD

where do I put the abstracted stuff?



mixins



variables



views



typography



forms



style.scss



partials



**junk-drawer**



# NESTING ABUSE

felt so right ... was soooooo wrong

## the HTML:

```
<html>
  <body>
    <section>
      <header></header>
      <article>
        <header></header>
      </article>
    </section>
  </body>
</html>
```

## the Sass:

```
html {
  body {
    section {
      header {color: green;}
      article {
        header {}
      }
    }
  }
}
```

## the CSS:

```
html body section header {color: green;}
```

## the HTML:

```
<html>
  <body>
    <section>
      <header>
        <article>
          <header>
            </article>
          </header>
        </article>
      </section>
    </body>
  </html>
```

## the CSS:

```
html body
```



USE

rong

```
h {
  er {color: green;}
  le {
  eader {}
```

```
en;}
```

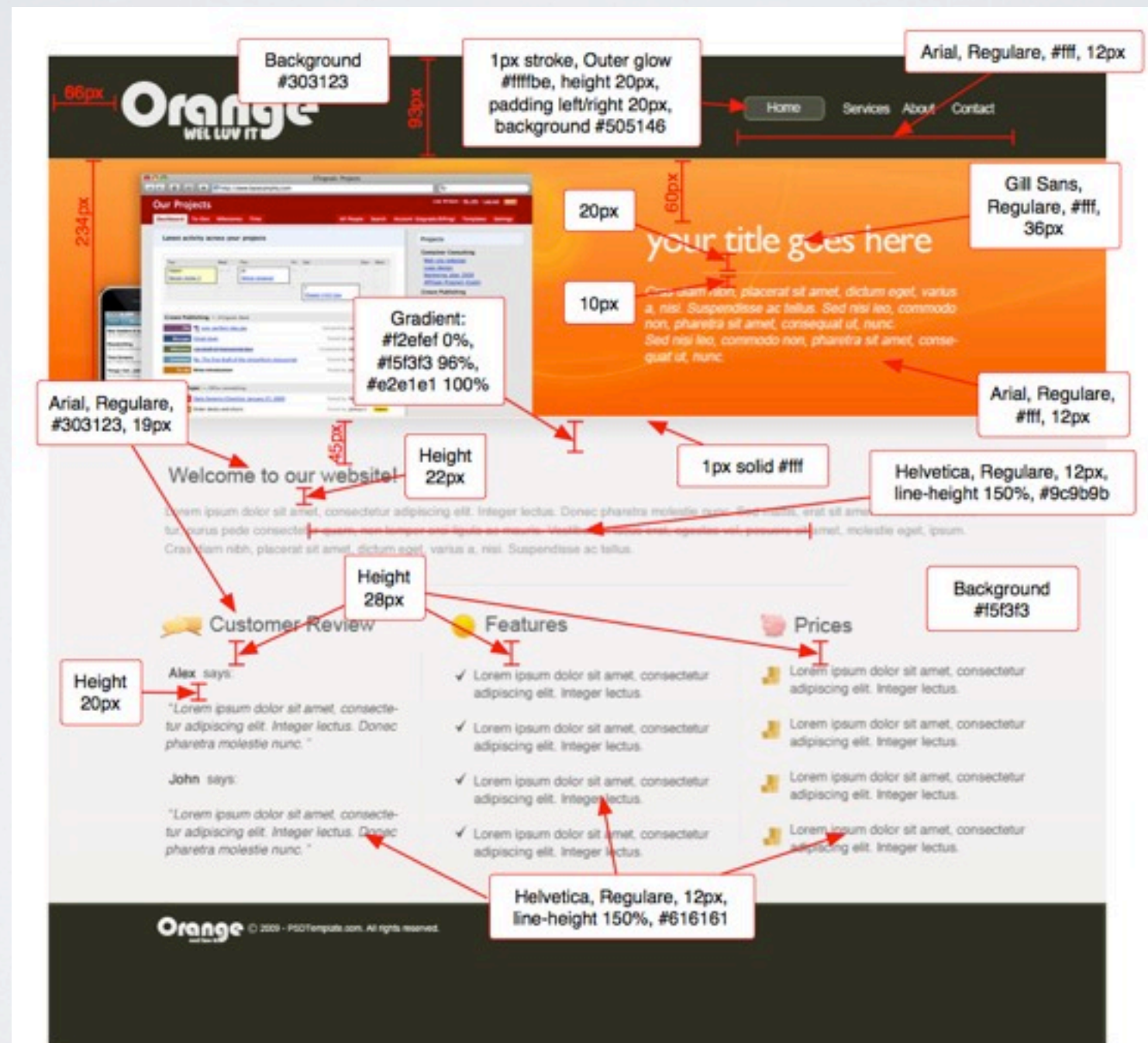


# LIFE IS A JOURNEY

learn from '*doing it wrong*'

# THE OUTSIDE-IN

our perspective was all wrong







# DECOMPOSE THE UI

start with the smaller parts



**Buttons**

**Typography**

**borders and  
line widths**

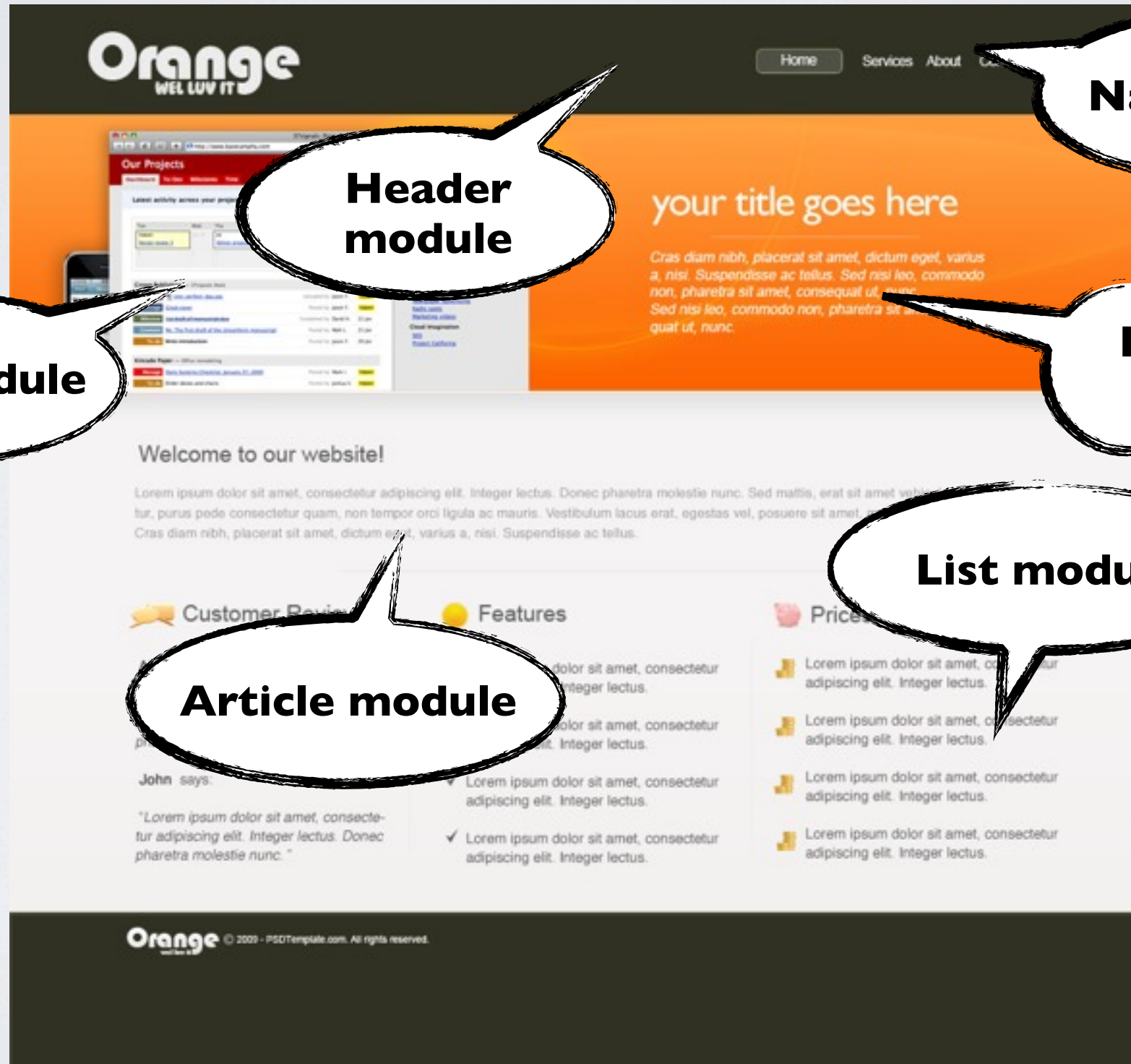
**standard  
icons**

**Color pallet**



# THE INSIDE-OUT

start small and build up



**Nav module**

**Header module**

**Hero module**

**Hero copy module**

**List module**

**Article module**

# ELEMENT, MODULE & LAYOUT

nature has taught us everything



# THE FILE STRUCTURE

code the element, create the module and assemble the layout



buttons/



color/



forms/



layouts/



modules/



typography/



ui\_patterns/



\_buttons.scss



\_config.scss



\_forms.scss



\_global\_design.scss



\_reset.scss



\_typography.scss



style.scss

# PARTIALS & THE MANIFEST

a powerful weapon in the Sass arsenal

```
// Following sequence will load the necessary Stipe libraries  
//
```

---

```
@import "compass/css3"; // Including the Compass CSS3 mixins  
@import "stipe/manifest";  
@import "color/extends";
```

```
// This where you start building your own styles.  
//
```

---

```
@import "typography";  
@import "forms";  
@import "buttons";  
@import "design";  
@import "ui_patterns/*"; // manifest files  
@import "modules/*"; // manifest files  
@import "layouts/*"; // manifest files
```



# PARTIALS & THE MANIFEST

a powerful weapon in the Sass arsenal

```
// Following sequence will load the necessary Stipe libraries  
//
```

```
-----  
@import "compass/css3"; // Including the Compass CSS3 mixins  
@import "stipe/manifest";  
@import "color/extends";
```

```
// This where you start building your own styles.  
//
```

```
-----  
@import "typography";  
@import "forms";  
@import "buttons";  
@import "design";  
@import "ui_patterns/*"; // manifest files  
@import "modules/*"; // manifest files  
@import "layouts/*"; // manifest files
```

**keep sub-  
directory files  
nicely  
organized**



# MANUAL VS GLOB-IMPORTS

there is no right or wrong here

## manual:

```
@import "grid/lib/the_grid";  
@import "grid/lib/grid_placement";  
@import "grid/lib/grid_margin";  
@import "grid/lib/push_logic";  
@import "grid/mixins";  
@import "grid/extends";
```



## glob:

```
@import "grid/lib/*";  
@import "grid/mixins";  
@import "grid/extends";
```



# CONFIGURABLE OPTIONS

this can really save your bacon

# \_CONFIG.SCSS

manage the smart defaults for your UI

```
///////// Typography configuration/////////
```

```
// -----
```

```
$font_size: 12;
```

```
//
```

```
$heading_1: 46;
```

```
$heading_2: 32;
```

```
$heading_3: 28;
```

```
$heading_4: 18;
```

```
$heading_5: 18;
```

```
$heading_6: 18;
```

```
//
```

```
$line: $font_size * 1.5;
```

```
//
```

```
$small_point_size: 10;
```

```
$large_point_size: 14;
```

```
//
```

```
$primary_font_family: #{"Helvetica Neue", Arial, sans-serif};
```

```
$secondary_font_family: #{"Helvetica Neue", Arial, sans-serif};
```

```
$heading_font_family: #{"Helvetica Neue", Arial, sans-serif};
```

```
$icon_font_alpha: #{'FontAwesome'};
```

```
$icon_font_bravo: #{'zocial'};
```

```
///////// Default webfont directory/////////
```

```
// -----
```

```
$webfont_directory: "/fonts/";
```

```
///////// default image directory //////////
```

```
//
```

**all logic  
NO CSS!**





# THE ELEMENT PARTIALS

where we get to work

# YOUR FOUNDATIONAL UI

all CSS, NO LOGIC!

```
button {
  font-size: em(16);

  @include linear_gradient_w3c (lighten($button_color, 15%),
#{lighten($button_color, 15%) 0%, $button_color 100%});
  &:hover {
    @include linear_gradient_w3c (lighten($button_color, 20%),
#{lighten($button_color, 20%) 0%, $button_color 100%});
  }

  .no-cssgradients & {
    @include linear_gradient_legacy_ie (lighten($bravo_color, 15%),
$bravo_color);
    &:hover {
      @include linear_gradient_legacy_ie (lighten($bravo_color, 20%),
$bravo_color);
    }
  }
  border: 1px solid $primary_button_border_color;
  @include border-radius(em(3));
  color: $white;
  padding: em(6) em(10);
  margin: 0;
  @include linear_gradient_legacy_ie (lighten($bravo_color, 15%), $bravo_color);
}
```

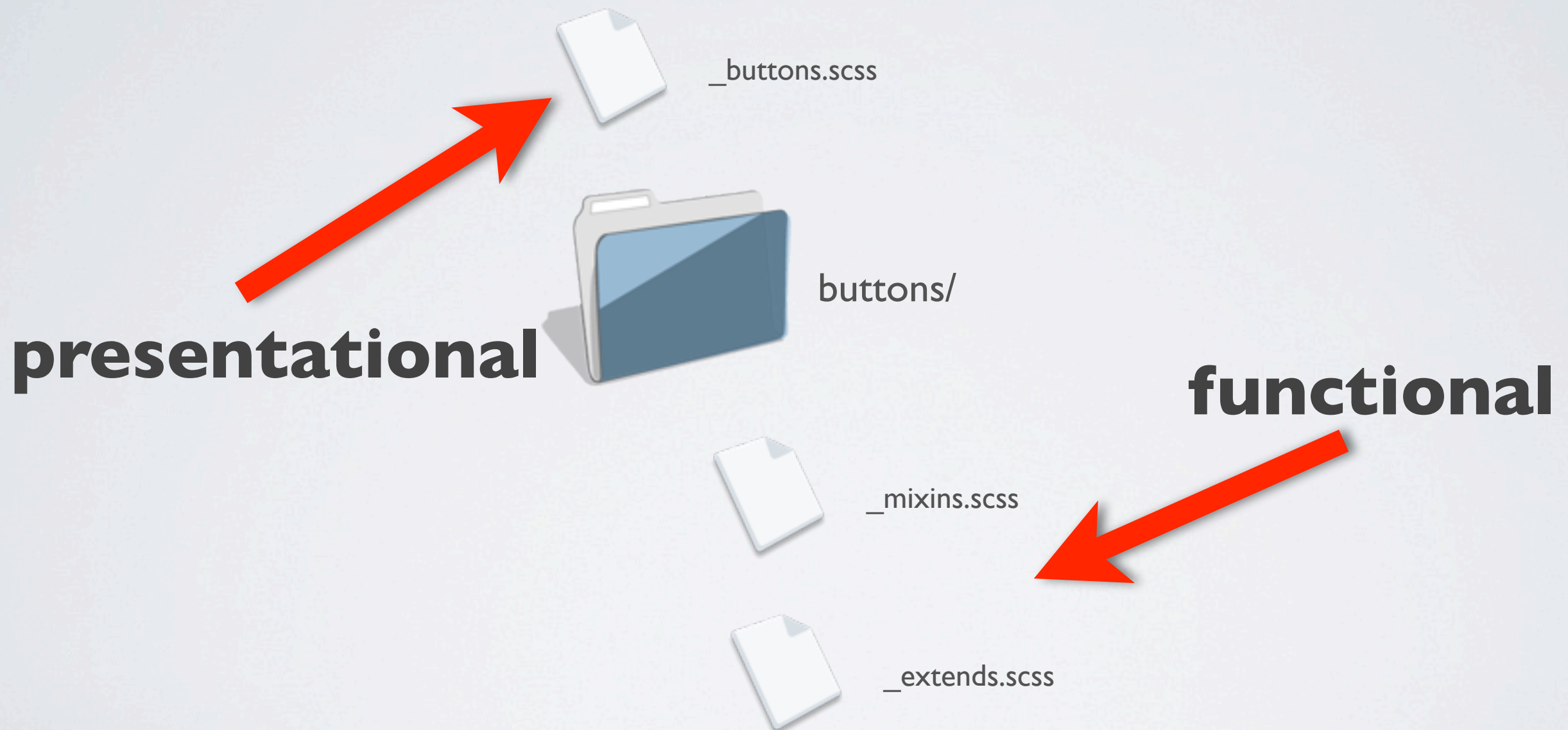


# CUSTOM CODE

where's u go?

# THE FILE STRUCTURE

functional vs. presentational





# FUNCTIONAL vs PRESENTATIONAL

organize your thoughts

```
// Custom button extends and mixins  
//
```

---

```
@import "buttons/mixins";  
@import "buttons/extends";
```

```
// buttons  
//
```

---

```
/* toadstool buttons */  
button, a.button {  
    @include button($button-color);  
}
```

# FUNCTIONAL vs PRESENTATIONAL

organize your thoughts

```
// Custom button extends and mixins  
//
```

**functional**

```
-----  
@import "buttons/mixins";  
@import "buttons/extends";
```

```
// buttons  
//
```

```
-----  
/* toadstool buttons */  
button, a.button {  
    @include button($button-color);  
}
```

**presentational**



# MODULES & UI PATTERNS

seeing is believing

# MODULES & UI PATTERNS

simplicity in organization

**Module Sass** is **exclusive** to a **particular interaction** of the application.

Modules will come in ***all shapes and sizes.***

Larger modules may also **consist** of **smaller modules** or smaller **UI patterns.**



# MODULES & UI PATTERNS

simplicity in organization

UI patterns are a **gray area** and are subject to **personal interpretation.**

In practice when engineering modules, from one to the next, small UI patterns may begin to emerge.

It is practical to try and encapsulate these smaller patterns for reuse,  
but I **don't lose sleep** over them.



# MODULES & UI PATTERNS

simplicity in organization



modules/



**registration/**



\_extends.scss



\_functions.scss



\_mixins.scss



\_module\_**registration**.scss



\_module\_personallInfo.scss

# MODULES & UI PATTERNS

simplicity in organization



modules/



**registration/**



\_extends.scss



\_functions.scss



\_mixins.scss



\_module\_**registration**.scss



\_module\_personallInfo.scss

**directory  
name**

**primary  
module**





# MODULES & UI PATTERNS

simplicity in organization



modules/



**registration/**



\_extends.scss



\_functions.scss



\_mixins.scss



\_module\_**registration**.scss



\_module\_personallInfo.scss



**purchase/**



\_extends.scss



\_functions.scss



\_mixins.scss



\_module\_summary.scss



\_module\_**purchase**.scss



# MODULES & UI PATTERNS

name-space by the module itself

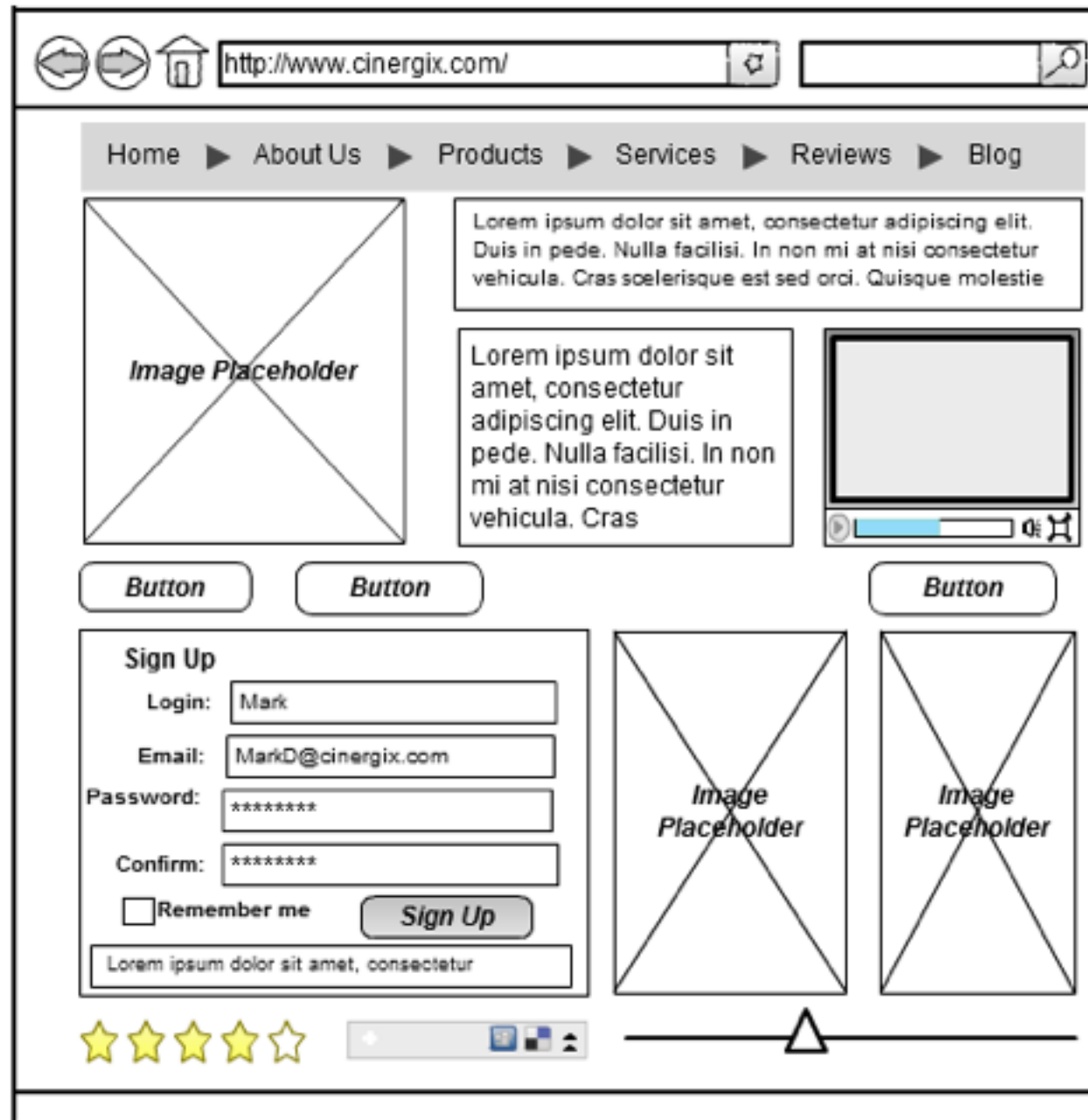
```
%orderSummary {
  .form_header {
    @include border_radius(em(5) em(5) 0 0);
    margin-bottom: 0;
  }
  .orderSummaryContainer {
    @extend %Grid4ThreeSideBox;
  }
  .initialAssetListing {
    @media #{ $mobile } {
      display: none;
    }
  }
  .price {
    text-align: right;
  }
  .total {
    font-size: em($large_point_size);
  }
  footer {
```

# ASSEMBLE THE LAYOUT

bringing the picture into focus

# ALL THOSE MODULES

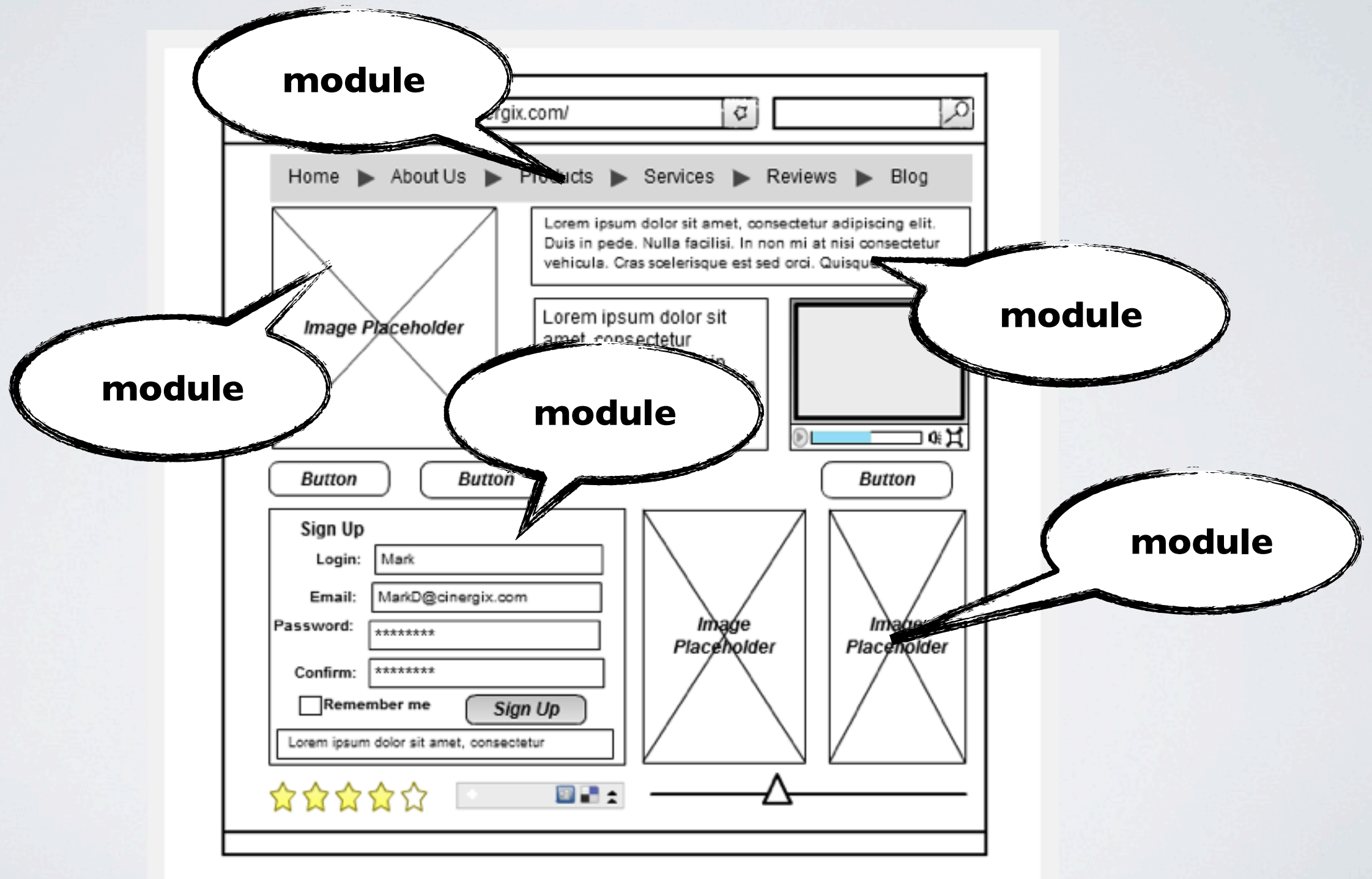
make up the view





# ALL THOSE MODULES

make up the view



# EVERYTHING IS CODED

manage your grid

```
.checkout_container {
  .checkout {
    @extend %grid_7;

    @media #{ $tablet_portrait } {
      @include grid(10, $grid_context: 10);
    }
  }
}

.orderSummary {
  @extend %grid_5;

  @media #{ $tablet_portrait } {
    @include grid(10, $grid_context: 10);
  }

  @media #{ $mobile } {
    @include grid(4, $grid_context: 4);
  }
}

.scrollBox {
```

# EVERYTHING IS CODED

manage your grid

```
.checkout_container {  
  .checkout {  
    @extend %grid_7;  
  
    @media #{ $tablet_portrait } {  
      @include grid(10, $grid_context: 10);  
    }  
  }  
}  
.orderSummary {  
  @extend %grid_5;  
  
  @media #{ $tablet_portrait } {  
    @include grid(10, $grid_context: 10);  
  }  
  
  @media #{ $mobile } {  
    @include grid(4, $grid_context: 4);  
  }  
}  
.scrollBox {
```

**name-spaced selector**

**placement of elements**



# SUMMARY

what have we learned?

**CSS management  
has come a long way**

# SUMMARY

what have we learned?

**CSS management  
has come a long way**

**Vanilla CSS has NO  
TOOLS for code  
management**

# SUMMARY

what have we learned?



**CSS management  
has come a long way**

**Vanilla CSS has NO  
TOOLS for code  
management**

# SUMMARY

learned?

**outside-in  
BAD!**

**inside-out  
BETTER!**

**CSS management  
has come a long way**

**Vanilla CSS has NO  
TOOLS for code  
management**

# SUMMARY

learned?

**outside-in  
BAD!**

**inside-out  
BETTER!**

**modular files with  
@import system  
=== WINNING!**

**CSS management  
has come a long way**

**Vanilla CSS has NO  
TOOLS for code  
management**

SUMMARY

**outside-in  
BAD!**

**inside-out  
BETTER!**

learned?

**elements, modules  
and then the view**

**modular files with  
@import system  
=== WINNING!**



**CSS management  
has come a long way**

**Vanilla CSS has NO  
TOOLS for code  
management**

**Take everything  
you know AND  
SET IT ON FIRE!**

**elements, modules  
and then the view**

**outside-in  
BAD!**

**inside-out  
BETTER!**

**modular files with  
@import system  
=== WINNING!**

**CSS management  
has come a long**

**NO  
de  
t**

**Take even  
you know  
SET IT O**

**Use processors**

**it will  
hurt less**

**modules  
he view**

**inside-out  
BETTER!**

**modular files with  
@import system  
=== WINNING!**

# QUESTIONS?

you, in the front ...



