SC5 Styleguide

Styleguide

represents your UI

Old school way

mockup ▶ **markup** ▶ **product**
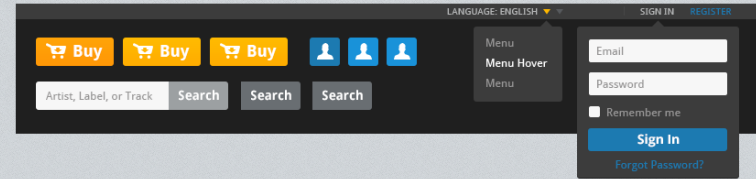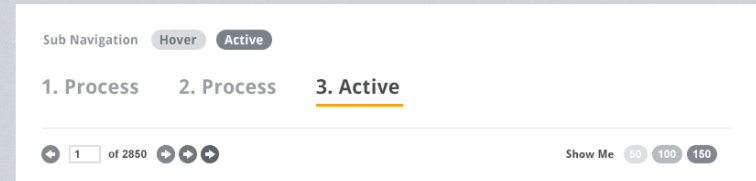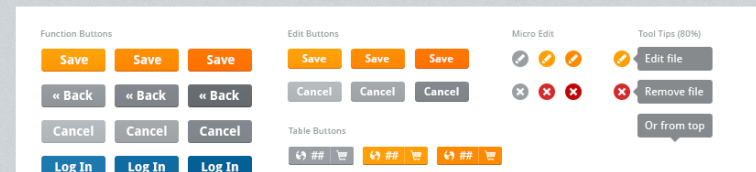
- PSD
- Pack of pages

- HTML/CSS/JS
- Waits for pages
- Or is re-coded

3

Dash process

In your dreams

mockup   markup   integration

In reality

mockup   markup

mockup   markup
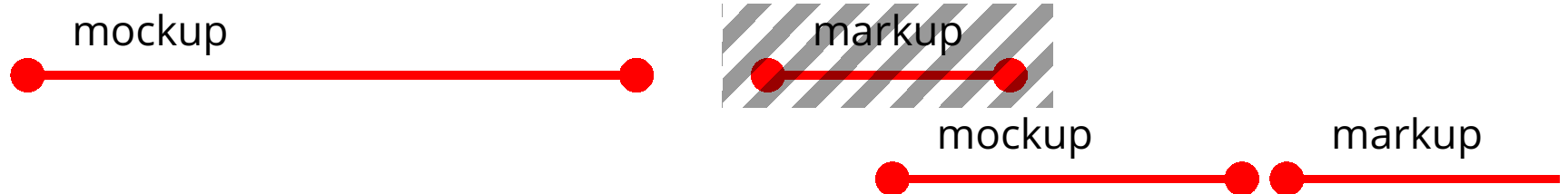
...over and over again

Old school "benefits"

- Inconsistent visual system
- Inconsistent code (duplicant, unreasoned)
- Version confusion
- Time-consuming
- **Unmaintainable**

"

Websites are systems rather than pages and as soon as we stop percerving them as that, the better.

Anna Debenham

Atomic design paradigm
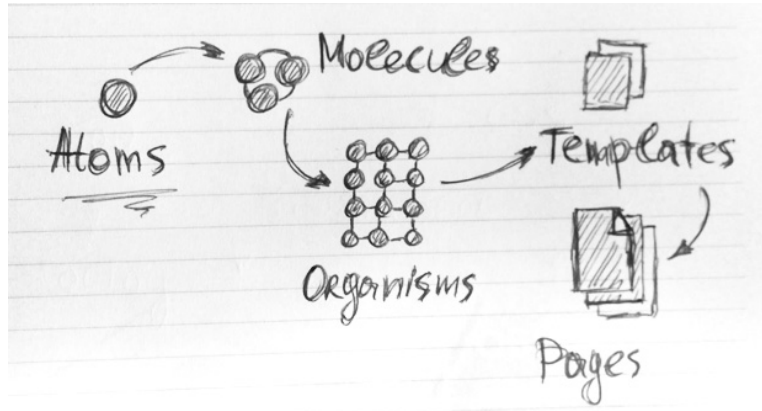
or any other modular solution

Living Styleguides

A **living styleguide** represents UI components of your website with exact the same styles which you use across the project.

Styleguide Driven Development

Styleguide ~~can~~ must be an integral part of your development workflow.

Styleguide Driven Development is the practise of using the styleguide as the focal point for all front-end UI development tasks.

9

We miss the tool

```
npm install sc5-styleguide
```

http://styleguide.sc5.io

SC5 Styleguide

Informative and easily navigable live styleguide which can be edited directly in the browser

My Styleguide

buttons

Overview

1 Colors and typography

2 Buttons and inputs

3 Structure

4 Designer tool

5 Helper elements

6 Angular Directives

Buttons and inputs

Links, buttons, input boxes etc.

2.3 Buttons

Button styles used in the styleguide

default - Default button

.primary - Primary button

:disabled - Disabled button

Show all variables   ✕

2 Buttons and inputs

This section does not contain any related variables.

Sub sections use the following variables:

secondary-color 👁

#E4E4E4

tertiary-color 👁

#F7FCF1

primary-action-color 👁

#EB7F00

default-action-color 👁

#1695A3

action-color-change 👁

Lift off!

Killer features

- CSS / SCSS / SASS / LESS

- Related variables

- Live editting

- Angular directives
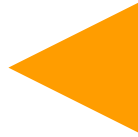
- Easy to integrate

13

```
my-styles/
  _variables.sass
  atoms/
    menu.sass
    footer.sass
  common.sass
  index.sass
  products.sass
```

+

- CLI
- Gulp
- Grunt

Gulp Stream

My Styleguide

Show only related variables

All variables

buttons	
Overview	
1 Colors and typography	
2 Buttons and inputs	
3 Structure	
4 Designer tool	
5 Helper elements	
6 Notifications	

Show all sections

Show all markup

2 Buttons and inputs

Links, buttons, input boxes and other similar components

2.2 Buttons

Button styles used in the styleguide

default - Default button
.primary - Primary button
:disabled - Disabled button

Button text   default
Button text   .primary
Button text   :disabled

Show CSS

`<button class="sg {$modifiers}">Button text</button>`

primary-color
#1c3849

secondary-color
#E4E4E4

tertiary-color
#F7FCF1

primary-action-color
#EB7F00

default-action-color
#1695A3

action-color-change *
10%

primary-font
"Helvetica Neue", Helvetica, Arial, sans-serif

secondary-font
"Helvetica Neue", Helvetica, Arial, sans-serif

designer-tool-width
30%

header-height
60px

Save changes   Reset local changes

KSS Knyle Style Sheets

```
// A button suitable for giving a star to someone.
//
// :hover             - Subtle hover highlight.
// .star-given        - A highlight indicating you've already given a star.
// .star-given:hover  - Subtle hover highlight on top of star-given styling.
// .disabled          - Dims the button to indicate it cannot be used.
//
// Styleguide 2.1.3.
a.button.star{
  ...
  &.star-given{
    ...
  }
  &.disabled{
    ...
  }
}
```

14

Intergration

```
gulp.task("styleguide", function() {
  return gulp.src(["**/*.css", "**/*.scss", "**/*.less"])
    .pipe(styleguide({

      ...

    }))

    .pipe(gulp.dest("<destination path>"));
});
```

Development with SC5 Styleguide

- Living overview of UI

- Quick manual testing

- Quick built-out of new pages

- Unit tests for UI

- Device-wall friendly

- **Easier designer/developer/client collaboration**

16

Developers mindset

Before

What markup, CSS and JS I need?

After

What modules is this page made of? Do I need to create new modules?

Future modules and pages start in styleguide.

Behind the curtain

Gulp streams

- Everything in memory

- Process files in parallel

CSS Parsing

What?

- Get key-value list of variables

- Parse code into KSS blocks

- Detect variables usage

How?

- ~~RegEx~~

- ~~OmetaJS~~

- Gonzales-PE

20

CSS encapsulation

¿Question?

```css
ul li::before {
    content: "·";
}
```

¡Answer!

```html
<div sg-scope-user-styles="mod.wrappedMarkup |
setModifierClass: mod.className | setVariables: variables |
addWrapper | unsafe">
  #shadow-root
    <style>
    @import url('/styleguide.css');
    @import url('/styleguide_pseudo_styles.css');
    </style>
    <button class="sg default">Button text</button>
  </div>
</div>
```

Open source

- Shame driven development

- Reported issues

- Pull requests

- Open source back

Future plans

- Article in Smashing Magazine

- Workflows

- Preview mode

- Variables in any file

- Editting full CSS