

# Web publishing definition

Language used



+

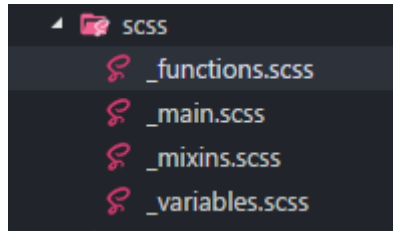
*Sass*

+



# Web publishing definition

global sass



```
1  /* Content variables
2
3  /* Fonts variables
4  = :root {
5    --fn24: rem(24);
6  }
7
8  $mainPdd: rem(18);
9  $mainSecondPdd: rem(15);
10 $mainThirdPdd: rem(13);
11
12 /* Main Colors variables
13 $clGreenLight: #2bc54a;
14 $clGreenBorder: #22ad3d;
15 $clGreenMain: #00c73c;
16 $clGreenLabel: #22b73f;
17 $clBlue: #157efb;
18
19 $clRed: #ff1616;
20
21 $clSilverLight: #f6f6f9;
22 $clSilverBorderLight: #f0f0f0;
23 $clSilverBorder: #d2d2d2;
24 $clSilverButtonBorder: #c9c9c9;
25 $clSilverBorderElements: #dadada;
26
27 $clTextNormal: #6a6a6a;
28
29 $clTextLink: #616161;
30
31 // #777;
32
33 = :root {
34   --silver-bg: $clSilverLight;
35   --green-1: $clGreenLight;
36 }
37
```

```
_variables.scss  _functions.scss x
1  // Function for convert to em & rem
2  $baseFontSize: 16;
3
4  //to rem
5  @function rem($pixels, $context: $baseFontSize) {
6    @if (unitless($pixels)) {
7      $pixels: $pixels * 1px;
8    }
9
10   @if (unitless($context)) {
11     $context: $context * 1px;
12   }
13
14   @return $pixels / $context * 1rem;
15 }
16
17 //to em
18 @function em($pixels, $context: $baseFontSize) {
19   @if (unitless($pixels)) {
20     $pixels: $pixels * 1px;
21   }
22
23   @if (unitless($context)) {
24     $context: $context * 1px;
25   }
26
27   @return $pixels / $context * 1em;
28 }
29
```

## [Explanation]

It is a sass file that defines basic font size, font size calculation, color etc. Just define the file and generate a sass file for each page.

# Web publishing definition

## Bootstrap



<https://getbootstrap.com/>

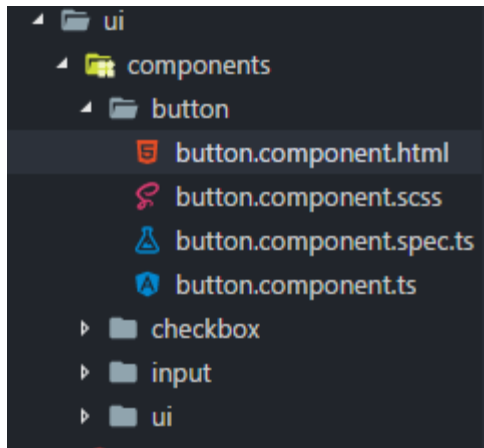
### [Explanation]

I do not want to use bootstrap. Please also make sass separately for each page.

Therefore, please refer to the bootstrap used in our angular to define the basic color or basic value.( v – 4.0.0 using)

# Web publishing definition

## Split UI



### button ui html

```
1 <button class="--green --fillet-3 --lg --full"
2   type="button">
3   <ng-content></ng-content>
4 </button>
5
```

### button ui sass

```
1 /* vars */
2 @import "~assets/scss/functions"
3 @import "~assets/scss/variables"
4
5 $button-radius: 3px;
6 $button-color: white;
7
8 $button-border: $clSilverBorder;
9
10 /* end */
11
12 :host {
13   //display: inline-block;
14
15   font-size: rem(14);
16
17   border-radius: $button-radius;
18   //border: 1px solid $button-b
19
20   background-color: $button-color;
21 }
22
23 .--green {
24   font:{
25     size: rem(18);
26   }
27   color: white;
28   background-color: $clGreenLight;
29   border-color: $clGreenBorder;
30 }
31
32 .--fillet-3 {
33   border-radius: $button-radius;
34 }
35
36 .--lg {
37   height: 60px;
38 }
39
40 .--md {
41   height: 32px;
42 }
43
44 .--full {
45   width: 100%;
46 }
47
```

### [Explanation]

In our work, basically the same UI (mostly components) separates sass files as well as html code for reuse and management.

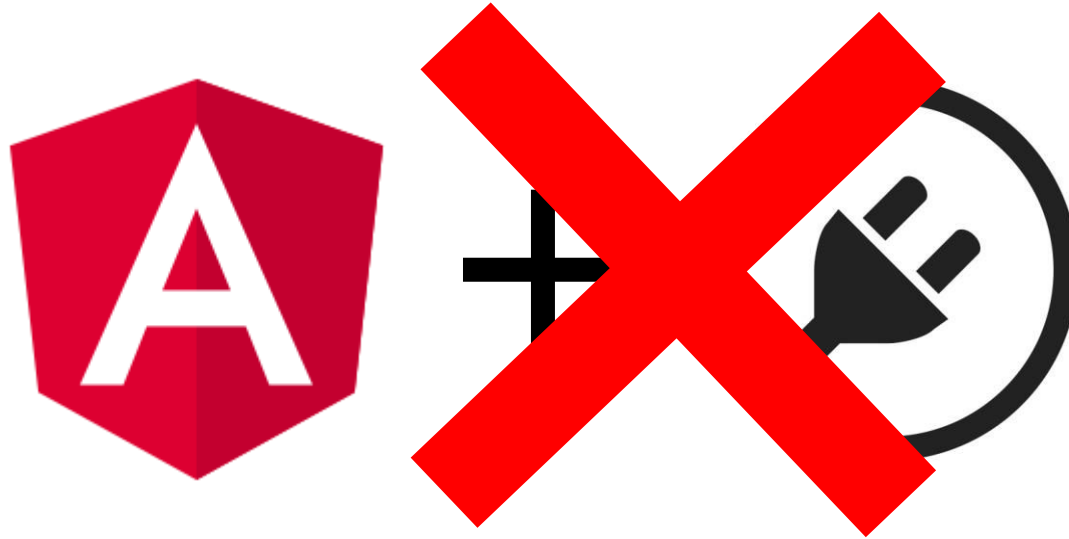
I think it will be difficult to subtract the componet code separately between html operations. At the very least, add a little annotation on top of the reused componet code. ex) For button componet <! - Component UI [button] ->

The sass code is expected to be able to split between tasks. Please do a separate work.

(If you can not do the splitting operation unexpectedly, please separate the sass codes used for componet with comments at the top.)

# Web publishing definition

Using JQuery & js plugin



## [Explanation]

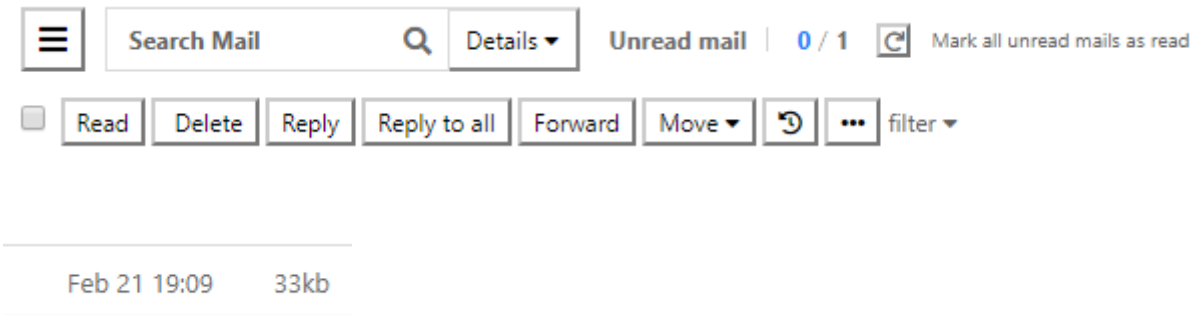
In the client framework, angular, jquery conflicts can prevent some plugins from being used.

It is expected that the page of the planning page is possible even if it is not using the plug-in. Mostly css or simple js code please.

Please let us know first when using the js plugin for page effects. Make sure that the plug-in is enabled.

# Web publishing definition

## Multilingual display



### [Explanation]

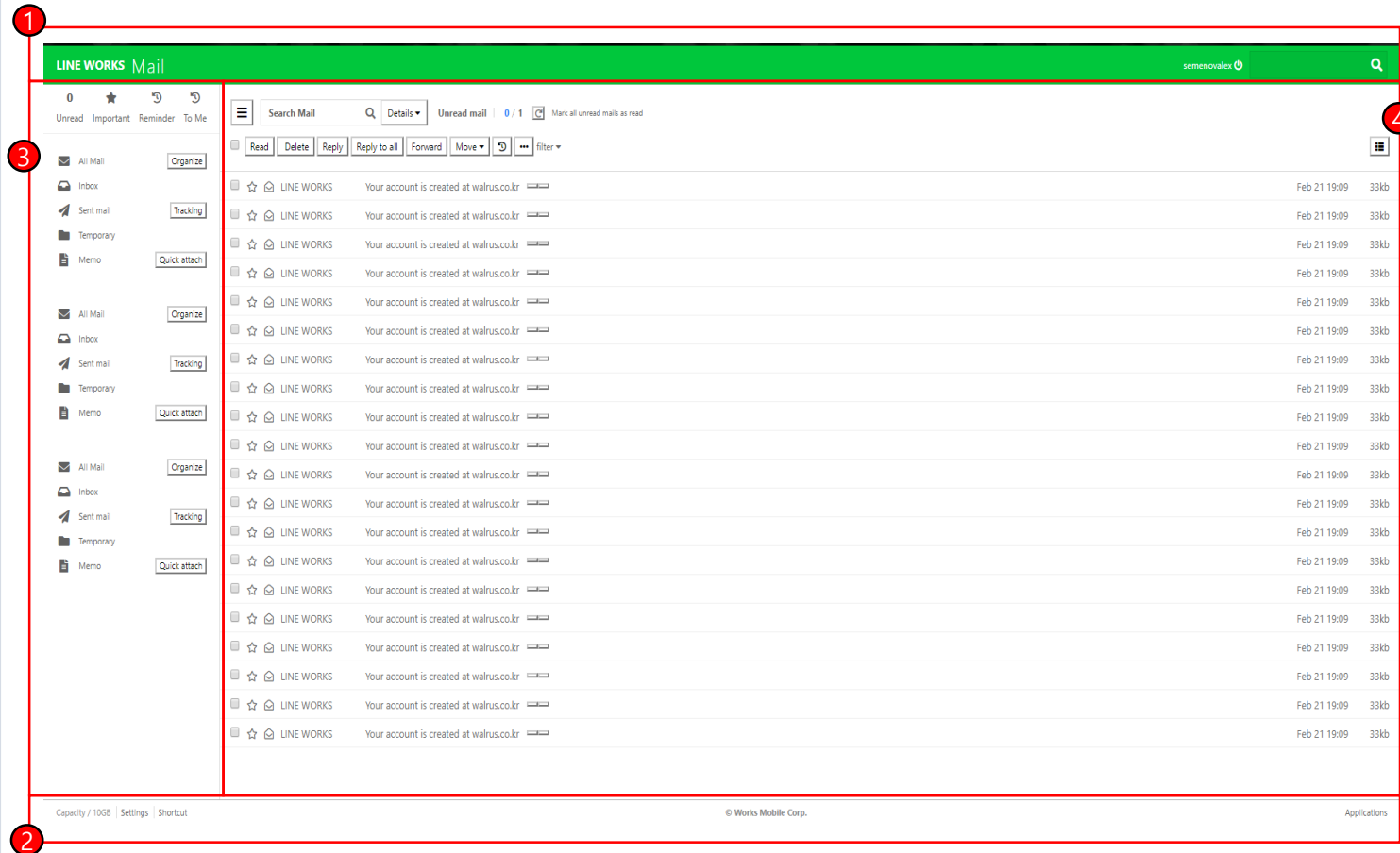
I know that there are some cases where the design is changed due to the difference of the letters in the language on the web page when the multilingual application is supported.

Currently, multilingual services are planned in English, Japanese and Korean. Please work as much as possible so that there is no difference in the length of the language characters.

If you need feedback on multilingual displays, please tell us.

# Web publishing definition

## Page Basic Frame Description



## [Explanation]

1. header : Except for the right page-specific text, it is the same design.
2. footer : The same design is the same except that only the left text part has different contents per page. (Refer to the plan)
3. menu : Except for the basic layout, the design differs depending on the page. (Refer to the plan)
4. content : Except for the basic layout, the design differs depending on the page. (Refer to the plan)

The full frame code is included in the attachment.