**Arquitecturas, filosofía modular y componentes**   
componente modular, es un patrón visual repetitivo. Es un fragmento de la interfaz con determinada función, puede tener su respectivo código html5, css y js

**SMACSS** scalable and modular architecture for css  
categorizar las reglas, cinco tipos de reglas:  
1. Archivo con reglas base, donde se apliquen estilos a las etiquetas html  
2. Estilos de layout, los que definen la organización de los bloques grandes contenedores. A cada contenedor como clase se le debe poner l-algo  
3. Módulos, los componentes internos como un menú  
4.Estados, los eventos de javascript por ejemplo que se oculten los menos que aparezcan, etc. Es decir en estados se recomienda poner las clases que vas a cambiar por javascript la clase con el prefijo “is” 🡪 .is-visible  
5.temas: colores y tipografías,

**OOCSS:** CSS Orientado a objetos  
Se basa en generar una mayor velocidad ,tratando de no repetir código con dos principios  
1. Separara la estructura(html) de la presentación (css)  
2. Separar el contenido de sus contenedores  
  
1: Casi todos los elementos de estilo de una página web tiene diferentes características visuales que se repiten en diferentes contextos. Piense en la marca de un sitio web: los colores, usos sutiles de gradientes o bordes visibles. Por otro lado, las estructuras igualmente se repiten.  
es decir si 3 elementos comparten estilos, es mejor hacer una clase aparte con los estilos que comparten y aplicárselo a la etiqueta html  
2: Si por ejemeplo ponemos estilos para un #sideabar h3, solo se aplicarían a los h3 dentro de este id, pero si tenemos otro h3 con esos mismos estilos? Se están duplicando estilos, entonces se recomienda separar estos estilos en modulos u objetos que pueden ser reutilizados en cualquier lugar, por ejemplo .subitutle-h3 {estilos} esta clase se pueda aplicar a cualquier elemento del html  
  
**ITCSS:** Triangulo invertido CSS  
Settings 🡪 variables  
Tools 🡪 mixins,mediaqueries,etc  
Generic🡪 normalize,etc  
Base 🡪 estilos a etiquetas  
Objet 🡪 estilos a objetos   
Componets 🡪 estilos a componentes como un menu  
trumps 🡪 sobreescrituras, solo afectan a una pieza en el dom en el tiempo, como un ¡important

**BEM:** Bloque Elemento Modificador   
Es una nomenclatura para escribir las clases  
**Bloque**: Es el componente/modulo 🡪 pieza de UI  
| pueden llevar guion intermedio como prefijo  
.menu-container  
**Elemento**: Cada elemento que conforma el bloque  
| Se separan del bloque con doble guion bajo \_\_  
**Modificador**: Una variante de un elemento del bloque  
Se separan del bloqueo o elemento con doble guion medio --  
  
**SUIT CSS**Componente: lo que en BEM es bloque  
**.MyComponent {estilos}**Los componentes pueden tener estados y se le agrega el prefijo is  
.MyComponente.is-animating {}  
.MyComponent—modifier  
  
utilitarios , como un enlace .u-link