***APUNTES DIW***

Tabla de contenido

[SASS 2](#_Toc160412635)

[-Creación 2](#_Toc160412636)

[-Variables 2](#_Toc160412637)

[-Mixin 2](#_Toc160412638)

[-Operadores 2](#_Toc160412639)

[-Funciones 2](#_Toc160412640)

[ACCESIBILIDAD 2](#_Toc160412641)

[USABILIDAD 2](#_Toc160412642)

[-ISO 9241 2](#_Toc160412643)

[-ISO 25000 2](#_Toc160412644)

[-CONCLUSIÓN 2](#_Toc160412645)

[TABINDEX 2](#_Toc160412646)

[LICENCIAS 2](#_Toc160412647)

[PREGUNTAS 2](#_Toc160412648)

# SASS

Es uno de los preprocesadores más conocidos y usados en la actualidad. Sus siglas significan Syntactically Awesome Style Sheets.

El framework Bootstrap se base en esto.

SASS y SCSS se diferencian principalmente por su sintaxis, ya que SASS es una sintaxis que utiliza la sangría (como python) y SCSS utiliza las llaves y los puntos y comas para.

## -Creación

Para poder usar el archivo deberemos transformarlo en un .css, mediante el comando ***sass –watch style.scss:style.css*** o usando la extensión de vscode LiveSASSCompiler.

## -Variables

Las variables se crean mediante ***${nombre}***

Al transformar el fichero las variables serán sustituidas por su valor.

## -Mixin

Los mixins son bloques reutilizables de código.

Se crean mediante @mixin nombre(parametro1,…){}

Se utilizan mediante @include nombre(parametro1,…);

## -Operadores

En SASS disponemos de los mismos operadores que en cualquier otro lenguaje de programación para realizar operaciones.

Disponemos de aritméticos, comparativos y lógicos.

Además, también podemos usar @if, @else if y/o @else para hacer código condicional.

Podemos usar el @for para iterar sobre algún rango de valores🡪@for $i from 1 through $variables{}

A su vez existe el @each, sirve para iterar por todos los elementos de una lista y extraeremos cada elemento usando nth($elemento,{numero})

Podremos usar el @while para iterar mientras una condición se cumpla.

## -Funciones

* Darken ($color, $cambio)🡪$cambio será el porcentaje a oscurecer.
* Lighten ($color, $cambio)🡪Igual que en el anterior, $cambio será el porcentaje a clarear.
* Mix ($color1, $color2, $proporcion)🡪$proporcion será el porcentaje que se aplicará del color1. Si se deja vacío será 50/50.
* Complement ($color)🡪 Devuelve el color complementario al recibido por parámetro.
* Invert ($color)🡪 Devuelve el color invertido al recibido.
* Percentage ($valor)🡪Convierte el valor del parámetro en un porcentaje.
* Grayscale ($color)🡪Devuelve el color en escala de grises del valor por parámetro.
* Transparentize ($color, $cambio)🡪Transforma el color en la versión transparente mediante el porcentaje pasado por parámetro.
* Nth ($lista, $n)🡪Devuelve el elemento de la lista en la posición indicada.

Existen más funciones como: opacify, change-color, adjust-hue, saturate, is-light, ceil, floor, abs, round, etc.

# ACCESIBILIDAD

La accesibilidad se puede definir como la posibilidad de acceso a un determinado portal web, en lugar de la facilidad de uso. Una web debe ser accesible antes que usable.

El acceso se deberá facilitar a todos los usuarios, más allá de las limitaciones de cada usuario.

## -W3C

El Worl Wide Web es una organización que se encarga del desarrollo de estándares que aseguran el crecimiento y el acceso a la web.

## -WAI

Inciativa para la Accesibilidad a la Web (Web Accessibility Initiative) aparece para desarrollar estrategias, directrices y recursos para garantizar el acceso por igual a la web.

Esta iniciativa desarrolló las directrices de accesibilidad para el contenido web 2.0 más conocido como ***WCAG 2.0*** donde se recogen las pautas y las técnicas que ofrecen soluciones accesibles para el contenido web. En 2018 fue sustituido por la versión ***WCAG 2.1***

## -Principios

Los criterios serán representados con notación x,y,z donde cada uno de los elementos representan x:principio, y:directrices y z:critérios de éxito.

* Diseño perceptible🡪Mostrar los elementos de manera que se entiendan y puedan ser percibidos.
  + Directrices:
    - Texto alternativo (caracteres grandes)
    - Contenido multimedia dependiente del tiempo
    - Adaptable (representar contenido sin perder información)
    - Distinguible (facilitar la visualización por texto y voz de todo el contenido)
  + Casos prácticos:
    - Contraste de color (4.5:1)
    - Adaptabilidad del texto (@media)
    - Multimedia alternativo (alt=””)
    - Contrastar y presentar (pausar contenido +5s)
    - Reconocimiento de componentes (label=””)
    - Sentido de la información (h1, h2, etc)
* Diseño operable🡪Interactuar y operar con la interfaz y el contenido de manera efectiva.
  + Directrices:
    - Teclado accesible (funciones desde teclado)
    - Tiempo suficiente (tiempo para leer y utilizar)
    - Ataques epilépticos (evitar contenido que pueda provocar ataques al usuario)
    - Navegación (navegar, búsqueda, etc)
  + Casos prácticos:
    - Parpadeo y destello
    - Navegación intuitiva
    - Evitar distracciones
    - Navegación por búsqueda
* Diseño comprensible🡪La información y las operaciones de usuarios deben ser compresibles.
  + Directrices:
    - Legible
    - Previsible (utilización previsible)
    - Asistencia a la entrada de datos (ayudar en errores)
  + Casos prácticos:
    - Lenguaje claro
    - Predecibilidad
    - Entrada de datos (sugerencias)
    - Identificación de errores
    - Consistencia
    - Ayuda contextual
* Diseño robusto🡪El contenido ha de ser robusto para que pueda ser interpretado por una gran variedad de agentes de usuarios, incluyendo a agentes de asistencia.
  + Directriz:
    - Compatible (maximizar la compatibilidad con agentes actuales y futuros + t. asistencia)
  + Casos prácticos:
    - Compatibilidad con T. de Asistencia (aria-label)
    - Compatibilidad con Navegadores
    - Detección de Cambios (aria-live=’polite’)

## -Requisitos de Conformidad

Se usan para verificar la adecuación a la accesibilidad de un sitio web.

* Nivel A🡪Es el mínimo y se deben cumplir 25 criterios.
* Nivel AA🡪Se deben cumplir los anteriores + 13 criterios.
* Nivel AAA🡪Se deben satisfacer todos los anteriores + 23 criterios.

Deberán cumplirse en webs completas.

# USABILIDAD

## -ISO 9241

Es el estándar internacional que trata sobre la ergonomía de la interacción humano-computadora (IHC), proporcionando pautas y principios para el diseño y la evaluación de sistemas interactivos, teniendo en cuenta la usabilidad y la experiencia del usuario.

* UI (Diseño de Interfaces de Usuario): Crear interfaces, iconos, menús y otros elementos visuales sean intuitivos y fáciles de entender. Minimizar la carga cognitiva y facilitar la navegación.
* UX (Diseño de Experiencia del Usuario): Busca hacer que la experiencia sea eficiente, teniendo en cuenta aspectos emocionales y psicológicos.
* Disposición de Hardware y Mobiliario: Disposición de periféricos para minimizar la fatiga y prevenir problemas de salud.
  + ISO 9241-11🡪Principios de usabilidad y diseño.
  + ISO 9241-110🡪Principio de diálogo Usuario-Sistema, comunicación fluida y efectiva entre el usuario y el sistema.
  + ISO 9241-12🡪Presentación visual y organización de información en interfaces de usuario.
  + ISO 9241-171🡪Directrices de accesibilidad para la ergonomía de la interacción. Mejora la accesibilidad en la interacción humano-computadora.

## -ISO 25000

La ISO 25000, también conocida como SQuaRE (System and Software Quality Requirements and Evaluation), se encargan de evaluar la calidad del producto de software.

1. ISO 2501n🡪Modelo de calidad
2. ISO 2502n🡪Medición de calidad
3. ISO 2503n🡪Requisitos de calidad
4. ISO 2504n🡪Evaluación de calidad

La ISO 25010 se centra en el modelo de calidad del software. En el modelo se determinan las características de calidad que se van a tener en cuenta a la hora de evaluar el producto. Se definen 8 características:

1. Adecuación Funcional
2. Eficiencia de Desempeño
3. Compatibilidad
4. Usabilidad
5. Fiabilidad
6. Seguridad
7. Mantenibilidad
8. Portabilidad

## -CONCLUSIÓN

La usabilidad se define de la capacidad del producto de software para ser entendido, aprendido, usado y resultar atractivo al usuario, al ser usado bajo determinadas condiciones.

# TABINDEX

El atributo ***TABINDEX*** se utiliza para permitir (o no) que ciertos elementos sean accesibles y puedan obtener el foco.

* tabindex=”-1”🡪no será enfocable por el teclado, pero si mediante scripts.
* tabindex=”0”🡪seguirá el orden del flujo natural.
* Tabindex=”{numeroPositivo}”🡪 Seguirá el orden por los números que se indica, donde el 1 será el primero de todos.

# LICENCIAS

# PREGUNTAS