

PROYECTO DE SOFTWARE

Conceptos básicos

Cursada 2024

¿QUÉ ABORDAREMOS EN ESTA CLASE?

- Lineamientos básicos a abordar.
- HTML-CSS: aspectos básicos.
- Introducción al DOM.
- **Accesibilidad web.**

LINEAMIENTOS BÁSICOS

Usaremos y desarrollaremos con software libre

El software libre es el software donde los usuarios tienen la libertad para ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software. El acceso al código fuente es una condición necesaria.

La definición oficial pueden encontrarla en el [sitio del proyecto GNU](#)

LINEAMIENTOS BÁSICOS

Utilizaremos git y gitlab para el desarrollo

Git es un *sistemas de control de versiones* distribuido libre diseñado para manejar proyectos con velocidad y eficiencia. **GitLab** es una aplicación opensource que nos permite administrar repositorios en git mediante una interfaz web.

Este tema se explicará en detalle la clase siguiente.

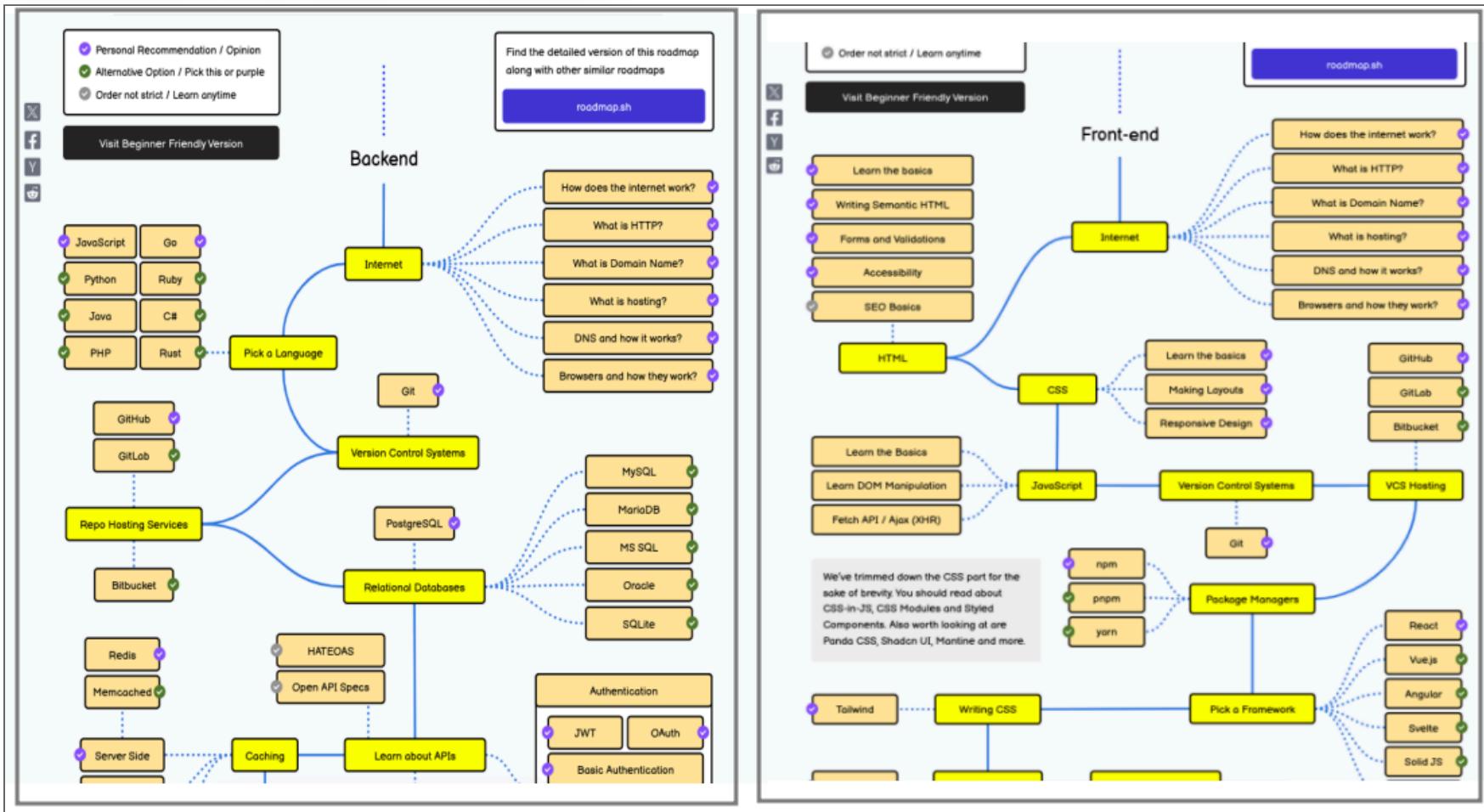
Es requisito en nuestra cursada el **trabajo en grupo**. El uso de git es una ENORME ventaja.

OBJETIVO GENERAL DE LA MATERIA

Desarrollar una aplicación web integrando distintas herramientas y tecnologías.

Programaremos en **Python** y **JavaScript**.

DESARROLLO WEB: ¿QUÉ INVOLUCRA?



roadmap.sh: developer-roadmap

INICIEMOS POR LO MÁS SIMPLE...

LA ARQUITECTURA WEB BÁSICA



Cliente web



Servidor web



HTML
CSS
scripts

ESPECIFICACIONES PÚBLICAS

¿Dónde?

RFC: Request For Comments

El consorcio de la web

El consorcio de la web se encarga del desarrollo de estándares y guías que aseguren el crecimiento futuro de la web.

¡MUCHAS TECNOLOGÍAS!

Si tenemos que decidir qué tipo de desarrollo proponer o con qué tecnologías trabajar, **¿por dónde empezamos?, ¿qué deberíamos analizar?**
Podemos analizar algunas tendencias y estadísticas.

ALGUNAS ESTADÍSTICAS

- Sobre servidores
- Sobre navegadores
- Estadísticas de Stackoverflow
- Y otras similares ...

PARA ANALIZAR

- Estas estadísticas nos dan un panorama del contexto.
- También nos permiten analizar las tendencias y características de usos.
Decidir qué tecnologías usar...
- Podemos retomar esto al final de la cursada ...

LA WORLD WIDE WEB

- Más de 30 años. A Proposal, Tim Berners-Lee, CERN
- La propuesta era (y sigue siendo) sencilla:
- Un mecanismo para identificar todo recurso accesible en la web: **URL/URI**
- Un protocolo simple: **HTTP**
- Un lenguaje para definir hipertextos: **el lenguaje HTML**

SOBRE LAS URI/URL

- La especificación es la RFC 3986
Ejemplos típicos

```
http://www.servidor.com.ar/especificacion#parte3  
https://www.taller.com.ar/info.php?id=12&qq=11  
mailto:proyecto@info.unlp.edu.ar
```

```
file:///home/claudia/git/proyecto2014/2020/teorias/clase2/index.html
```

- URL encoding
- Las URL se transmiten en ASCII.
- Algunos caracteres deben convertirse.

```
../Clase%201/EjemplosClase1/Ejemplo%20con%20enlaces.html
```

SOBRE EL PROTOCOLO HTTP

- HTTP es un protocolo simple: una transacción HTTP consta de 4 pasos: **inicio conexión - solicitud - respuesta - cierre conexión**
- HTTP es un **protocolo sin estado**: ¿qué significa esto?

ALGUNAS SOLICITUDES HTTP

- **GET**: retorna la información identificada por la URI-solicitada.
- **HEAD**: retorna sólo los encabezados, sin el cuerpo de la respuesta.
- **POST**: se utiliza para el envío de datos al servidor. Estos datos se incluyen en el cuerpo en el mensaje.

La respuesta

- El servidor retorna un código que indica el estado de la solicitud. Por ejemplo, 200 y el recurso, o "el famoso" 404.
- Más info sobre solicitudes HTTP

Retomaremos esto al abordar el desarrollo de API REst.

EL LENGUAJE HTML

```
<!DOCTYPE html>
<html lang="es">

<head>
<title>Título</title>
</head>

<body>
<h1>Proyecto de Software</h1>
<h2>Reglamento de cursada</h2>
</body>

</html>
```

- La última versión en [Web Hypertext Application Technology Working Group](#)

HTML: BÁSICO

¿Qué indica la cláusula **doctype**?

```
<!DOCTYPE html>
```

Los elementos pueden tener atributos.

```
<html lang="es">
```

¿Qué elementos contiene el encabezado? ¿Y el cuerpo?

```
<!DOCTYPE html>
<html lang="es">

<head>
<title>Título</title>
</head>

<body>
<h1>Proyecto de Software</h1>
<h2>Reglamento de cursada</h2>
    Acá va el texto del reglamento
</body>

</html>
```

- Más info: [Tutorial HTML & CSS Is Hard](#)

LOS CAMPOS META

```
<meta name="keywords" content="HTML,CSS,JavaScript">
<meta name="description" content="Guía de la materia
Proyecto de Software-UNLP.">
```

- ¿Qué pasa con los caracteres especiales?

```
<meta charset="UTF-8">
```

- ¿Qué les parece que indicamos con este campo?

```
<meta name="robots" content="noindex" />
```

FORMULARIOS EN LA ARQUITECTURA WEB

Observemos este código

```
<!DOCTYPE html>
<html lang="es">

<head>
    <title>Formularios web</title>
</head>

<body>
    <h1> Mi color favorito</h1>
    <form method="GET" action="proceso.php">
        Nombre: <input name="nombre" type="text"
placeholder="Ingresá tu nombre acá " />
        Color Favorito: <input name="color_favorito"
list="colores" />
        <datalist id="colores">
            <option value="Azul" />
            <option value="Rojo" />
            <option value="Verde" />
            <option value="Rosa" />
            <option value="Amarillo" />
        </datalist>
        <input type="submit" />
    </form>
</body>

</html>
```

- Más info sobre formularios

METHOD="GET"

El mensaje HTTP sería así:

```
GET /proceso.php?nombre=claudia&color_favorito=Azul HTTP/1.1
Host: www.servidor.com
User-Agent: Mozilla/5.0
Accept: image/gif, image/jpeg, text/html
Accept-language: es, en
Accept-Charset: iso-8859-1
```

METHOD="POST"

El mensaje HTTP sería así:

```
POST /proceso.php HTTP/1.1
Host: www.servidor.com
User-Agent: Mozilla/5.0
Accept: image/gif, image/jpeg, text/html
Accept-language: es, en
Accept-Charset: iso-8859-1

nombre=claudia&color_favorito=Azul
```

GET VS. POST

GET

POST

Los datos son visibles en la URL

No se ven

Hay límite de caracteres

No tiene

Los resultados pueden guardarse como favoritos

No se guardan

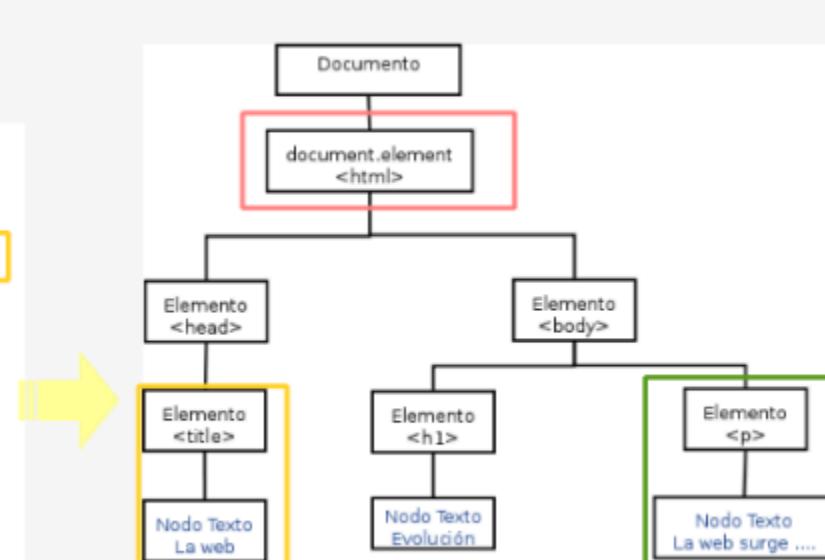
Quedan en el historial del navegador

No se guardan

- Entonces, ¿cuándo uso uno u otro?
- Más info este [artículo](#)

DOM: EL MODELO DE OBJETOS DEL DOCUMENTO

```
<html>  
  
<head>  
  <title> La web </title>  
</head>  
  
<body>  
  <h1>Evoluci&acute;n </h1>  
  <p> La web surge .....</p>  
</body>  
  
</html>
```



DOM: EL MODELO DE OBJETOS DEL DOCUMENTO

- El modelo de objetos del documento es una **API**, que permite acceder a los contenidos de un documento HTML.
- Proporciona una **representación** estructurada, **orientada a objetos**, de los elementos individuales y el contenido del documento, con métodos para recuperar y fijar las propiedades de dichos objetos.
- Proporciona métodos para agregar y eliminar objetos al documento.
- También proporciona una **interfaz estándar** para trabajar con **eventos**.
- La especificación en: <https://dom.spec.whatwg.org/>

DOM: EL MODELO DE OBJETOS DEL DOCUMENTO

Entre otras cosas, define una forma de acceder a un elemento a través del atributo **id**.

```
<h1>Proyecto de Software</h1>
<h1 id="reglamento">Reglamento de cursada</h1>
```

```
document.getElementById('reglamento')
```

Retomaremos esto con mayor profundidad.

¿CÓMO CAMBIAMOS LA VISUALIZACIÓN DE UN DOCUMENTO HTML?

HOJAS DE ESTILO: CSS (CASCADING STYLE SHEETS)

- El documento tiene una estructura y una forma de visualización
- **Estructura:** usando las etiquetas HTML más apropiadas.
- **Visualización:** usando hojas de estilo
- Ver sitio [CSS Zen Garden](#)

CSS: BÁSICO

Reglas: selectores y propiedades

```
p {  
    color: red;  
    font-size: 14px;  
}  
ol, ul {  
    color: blue;  
    font-size: 18px;  
}  
ol li {  
    color: blue;  
}
```

- ¿A qué elementos estamos especificando estilos?
- ¿Cuáles son los selectores y cuáles las propiedades?

CSS: BÁSICO

¿Dónde se ubican estas reglas?

```
<!DOCTYPE html>
<html lang="es">

<head>
    <title>Formularios web</title>
    <style>
        h1 {color:red}
    </style>
</head>

<body>
    <h1 style="color:red"> Mi color favorito</h1>
</body>

</html>
```

Los archivos .css

```
<link rel="stylesheet" href="estilo.css">
```

¿CÓMO SE APLICAN?



- **Estilos definidos en CSS externos.**
- **Estilos definidos en el documento html (tag style).**
- **Estilos en línea: atributo style en las etiquetas.**

Imagen sacada de: <https://internetingishard.com>

TAREA PARA EL HOGAR...

¿De qué color se muestra el texto "Mi color favorito"?

```
<!DOCTYPE html>
<html lang="es">
<head>
    <title>Jugando con estilos</title>
    <style>
        h1 {color:orange}
        h1.verde {color: green}
        .azul {color: blue}
        #amarillo {color: yellow}
    </style>
</head>

<body>
    <h1 class="verde" id="amarillo" style="color:red"> Mi
color favorito</h1>
</body>

</html>
```

- ¿Y si incluimos un archivo css que contiene...?

```
h1 {color:grey}
```

SELECTORES

a

Un elemento

a, b, c

Varios elementos

a b

Elementos descendientes

a > b

Elemento hijo directo

a + b

Elementos adyacentes

a:last-child

Último elemento a

...

...

CSS: MODELO DE CAJAS

- Todos los elementos de un doc HTML se representan mediante cajas rectangulares.
- Cajas creadas por los elementos de línea y los elementos de bloque
- Más info: [cap. 5 del tutorial Interneting Is Hard](#)

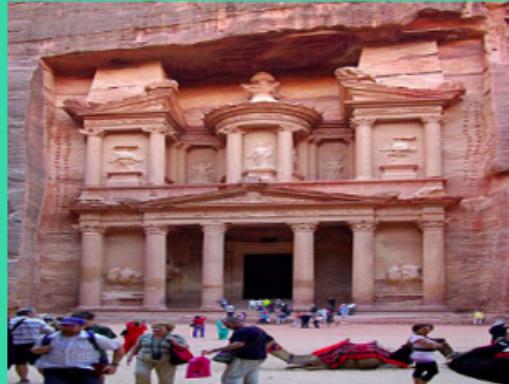
CSS: MODELO DE CAJAS (CONT.)

margin
border
padding

Contenido de la caja

Petra

Está ubicada en Jordania y es una de las nuevas siete maravillas del mundo moderno.



El asentamiento de Petra se localiza en un valle angosto, al este del valle de la Aravá que se extiende desde el mar Muerto hasta el Golfo de Aqaba. Los restos más célebres de Petra son sin duda sus construcciones labradas en la misma roca del valle (hemispeos), en particular, los edificios conocidos como el Khazneh (el Tesoro) y el Deir (el Monasterio). Fundada en la antigüedad hacia el final de siglo VIII a. C. por los edomitas, fue ocupada en el siglo VI a. C. por los nabateos que la hicieron prosperar gracias a su situación en la ruta de las caravanas que llevaban el incienso, las especias y otros productos de lujo entre Egipto, Siria, Arabia y el sur del Mediterráneo.

DISEÑO RESPONSIVO

MOBILE



TABLET



DESKTOP



Imagen obtenida de: <https://www.internetingishard.com/html-and-css/responsive-design/>

DISEÑO RESPONSIVO: VIEWPORT

```
<meta name="viewport" content="width=device-width,  
initial-scale=1.0">
```

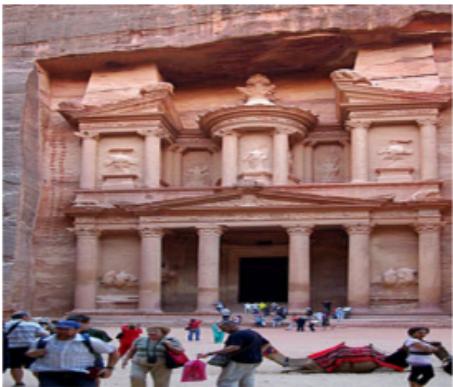


Diseño responsivo

**Veamos cómo se ve CON
viewport**

La ciudad de Petra

Está ubicada en Jordania y es una de las nuevas siete maravillas del mundo moderno.



El asentamiento de Petra se localiza en un



Diseño responsivo

Veamos cómo se ve SIN viewport

La ciudad de Petra

Está ubicada en Jordania y es una de las nuevas siete maravillas del mundo moderno.



El asentamiento de Petra se localiza en un valle angosto, al este del valle de la Aravá que se extiende desde el mar Muerto hasta el Golfo de Aqaba. Los restos más célebres de Petra son sin duda sus construcciones labradas en la misma roca del valle (hemispeos), en particular, los edificios conocidos como el Khazneh (el Tesoro) y el Deir (el Monasterio). Fundada en la antigüedad hacia el final de siglo VIII a. C. por los edomitas, fue ocupada en el siglo VI a. C. por los nabateos que la hicieron prosperar gracias a su situación en la ruta de las caravanas que llevaban el incienso, las especias y otros productos de lujo entre Egipto, Siria, Arabia y el sur del



DISEÑO RESPONSIVO: MEDIA QUERIES

Permiten definir estilos específicos dependiendo del medio donde se mostrará la página.

```
<!DOCTYPE html>
<html lang="es">
    <head>
        <title>Diseños responsivos</title>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width,
initial-scale=1.0">
        <style>
            @media (min-width: 600px) {
                body {
                    background-color: lightgreen;
                }
            }
            @media (max-width: 599px) {
                body {
                    background-color: blue;
                }
            }
        </style>
    </head>
    <body>
        <h1> Diseño responsivo</h1>
    </body>
```

</html>

DISEÑO RESPONSIVO: MÁS OPCIONES

- Establecer estilo en imágenes: max-width: 100%
- Usar flex y grid
- Más info: <https://web.dev/articles/responsive-web-design-basics?hl=es>

HAY MUCHAS HERRAMIENTAS

- Bootstrap
 - Materialize
 - Bulma
 - Y muchos otros ...
- ¿Qué son? ¿Usaron?**

¿QUÉ PASA CUANDO ACCEDEMOS A LA WEB CON LA PANTALLA APAGADA?

¿QUÉ PASA SI NO DISPONEMOS DE UN MOUSE? ¿O UN TECLADO?

Probemos algo...

- Sitio educ.ar
- Sitio Diario El Día

ALGUNAS REFLEXIONES

- OMS: discapacidad y salud - Datos marzo-2023
- Según la OMS, **1300 millones de personas** (es decir, 1 de cada 6 personas en todo el mundo) sufren una discapacidad importante.
- OMS: ceguera y discapacidad visual - Datos agosto-2023
- En el mundo hay al menos **2200 millones de personas** con deterioro de la visión cercana o lejana.

ACCESIBILIDAD WEB

"The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect".

Tim Berners-Lee, W3C

TAREA PARA EL HOGAR ...

Veamos el siguiente video:



¿QUÉ INVOLUCRA PENSAR EN LA ACCESIBILIDAD?

- Hay personas que tienen dificultad para leer o comprender textos.
- Hay personas que no pueden ver, escuchar o moverse.
- No todos utilizan mouse y teclados.
- Conexión a Internet lenta.
- Se trabaja con tecnologías muy diversas.
- Pantallas de sólo texto, pequeñas o de distintas resoluciones.

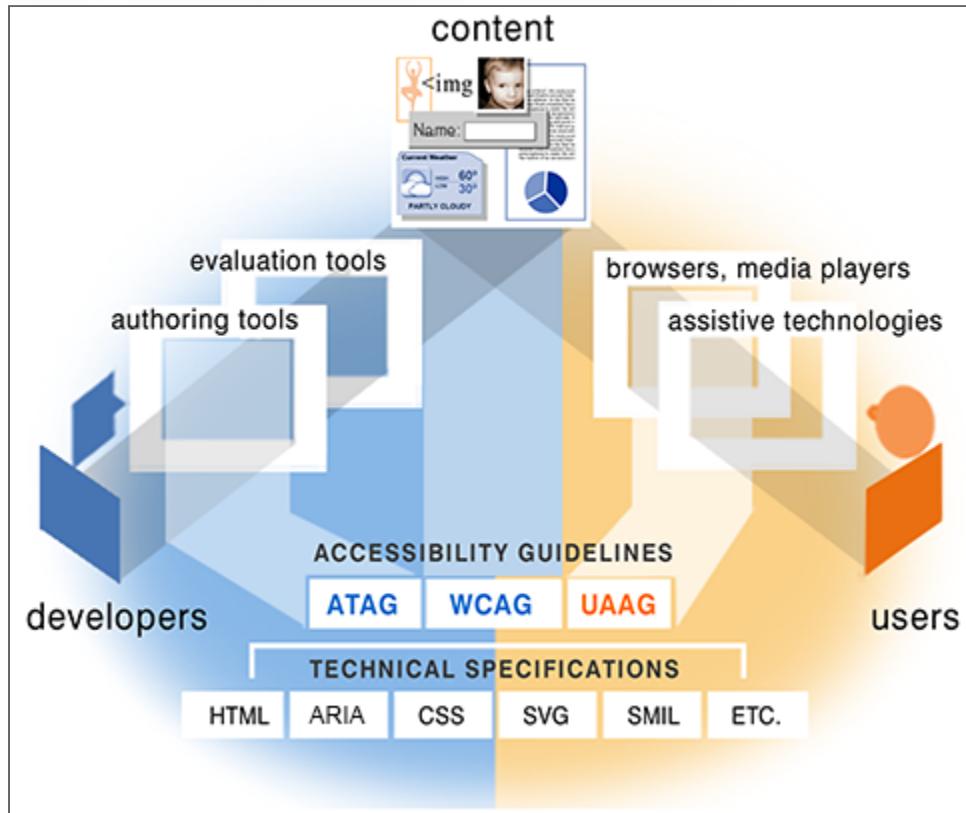


Objetivos

Desarrollar estrategias, pautas, recursos para hacer la Web accesible a personas con discapacidad. Pero también será accesible en otros entornos y aplicaciones, como navegador de voz, teléfono móvil, PC de automóvil; y ante limitaciones bajo las que opere, como entornos ruidosos, habitaciones infra o supra iluminadas, entorno de manos libres.

URL: <https://www.w3.org/WAI/>

WAI: WEB ACCESSIBILITY INITIATIVE



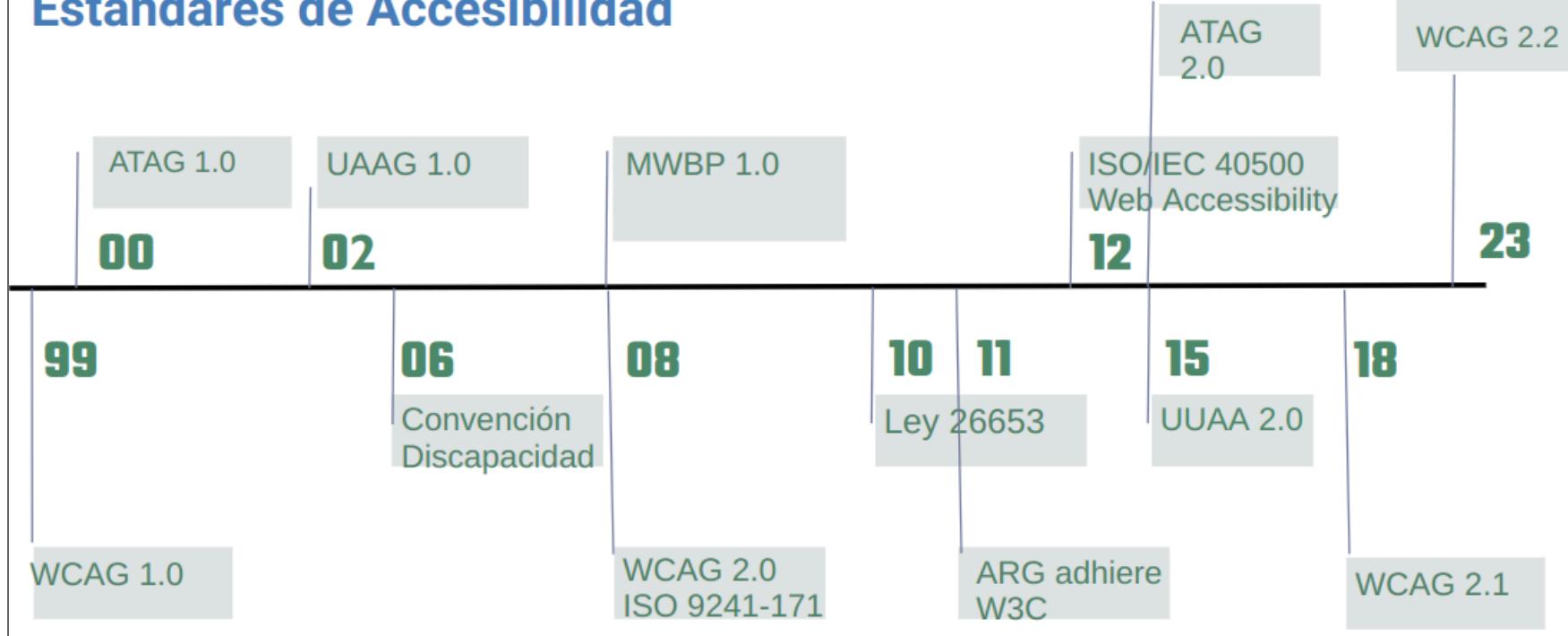
Sacado del sitio de la WAI de la W3C

WAI: WEB ACCESSIBILITY INITIATIVE

- Para aplicaciones de usuario: **WAI-User Agent Accessibility Guidelines (UAAG)**
- Para herramientas de autor: **WAI-Authoring Tool Accessibility Guidelines (ATAG)**
- Para contenido web: **Web Content Accessibility Guidelines (WCAG)**
 - WCAG 1.0
 - WCAG 2.0
 - WCAG 2.1
 - **WCAG 2.2 (2023)**

DESDE HACE AÑOS...

Estándares de Accesibilidad



Material elaborado por la Prof. Ivana Harari, en su presentación "Introducción a la accesibilidad digital".

GUÍAS PARA CONTENIDO WEB ACCESIBLE: WCAG

- Definen **principios de diseño** web.
- Cada principio tiene **pautas**.
- Cada pauta tiene **criterios testeables**.
- Cada criterio tiene **nivel de conformidad: A, AA y AAA**.

PRINCIPIOS DE DISEÑO PARA EL CONTENIDO WEB (WCAG) 2.2

- **1- Perceptible:** la información y los componentes de la interfaz de usuario deben ser mostrados a los usuarios en formas que ellos puedan entender.
- **2- Operable:** los componentes de la interfaz de usuario y la navegación debe ser manejable.
- **3- Comprensible:** la información y las operaciones de usuarios deben ser comprensibles.
- **4- Robustez:** el contenido deber ser suficientemente robusto para que pueda ser bien interpretado por una gran variedad de agentes de usuario, incluyendo tecnologías de asistencia.

CUMPLIR PAUTAS PRINCIPIO 1 - PERCEPTIBLE

- Proporcionar alternativas textuales para contenido no textual.
- Proporcionar subtítulos y otras alternativas para multimedia.
- Crear contenido que se pueda presentar de diferentes formas, incluyendo a las tecnologías de apoyo, sin perder información.
- Facilitar que los usuarios puedan ver y oír el contenido.

CUMPLIR PAUTAS PRINCIPIO 2 - OPERABLE

- Proporcionar acceso a todas las funcionalidades mediante el teclado.
- Conceder a los usuarios tiempo suficiente para leer y usar el contenido.
- No usar contenido que pudiera causar convulsiones o reacciones físicas.
- Ayudar a los usuarios a navegar y encontrar el contenido.
- Facilitar métodos de entrada diferentes al teclado.

CUMPLIR PAUTAS PRINCIPIO 3 - COMPRENSIBLE

- Proporcionar texto legible y comprendible.
- Proporcionar contenido que sea predecible en apariencia y operación.
- Ayudar a los usuarios a evitar y corregir errores.

CUMPLIR PAUTAS PRINCIPIO 4 - ROBUSTEZ

- Maximizar la compatibilidad con herramientas de usuario actuales y futuras.

MARCO LEGAL

En distintos países existen normativas y leyes que obligan a que sitios web y apps nativas sean accesibles.

EN ARGENTINA

Ley Nacional de Accesibilidad N° 26.653

La ley N° 26.653 de "Accesibilidad Web" busca promover la accesibilidad de la información, facilitando especialmente el acceso a todas las personas con discapacidad con el objeto de garantizarles la igualdad real de oportunidades y trato, evitando así todo tipo de discriminación.

Sacado de: <https://www.argentina.gob.ar/jefatura/innovacion-publica/onti/accesibilidad-web>

PARA REFLEXIONAR...

- ¿Por qué les parece que es importante cumplir con los estándares?
- Veamos uno de los videos de perspectivas

¿CÓMO VALIDAMOS?: CHEQUEO INICIAL

La W3C brinda un check list para revisar rápidamente la accesibilidad.

Título de la página

Textos alternativos

Imágenes y videos

Textos

Encabezados, contrastes de color y
redimensiones

Interacción

Acceso por teclado, foco visual,
formularios, errores, etc.

General

Contenido en movimiento o parpadeante,
alternativas multimedia (audio y video),
Verificación de estructura básica.

¿CÓMO VALIDAMOS?: EVALUACIÓN MANUAL

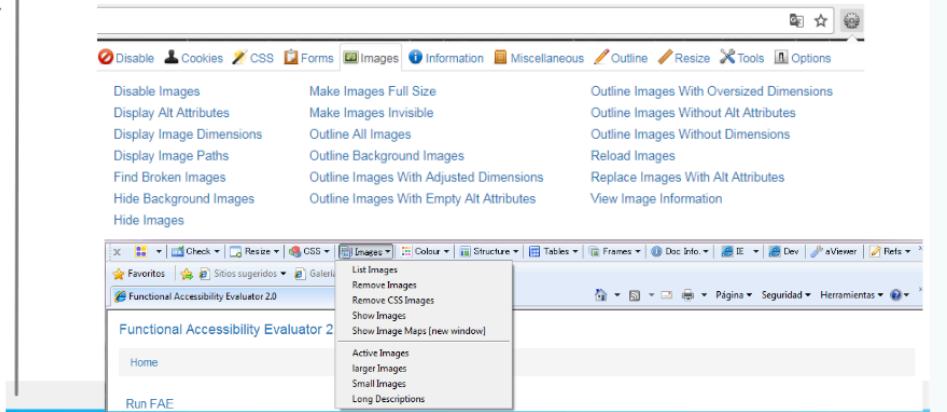
Evaluar el sitio en forma manual

- El sitio debe ser evaluado teniendo en cuenta distintos escenarios de interacción, comprobando su estabilidad y acceso.
- En diferentes circunstancias, el contenido y sus servicios deben poder accederse :

Con teclado solamente
Con lectores de pantalla
Con o sin imágenes
Con o sin color
Con y sin tablas
Con y sin hojas de estilo
Con y sin javascript
Otras configuraciones

Evaluar el sitio en forma manual

- Hay complementos para los navegadores que permiten configurar la página mediante distintos seteos para analizar cómo se comporta la misma.



Material elaborado por la Prof. Ivana Harari, en su presentación "Introducción a la accesibilidad digital".

¿CÓMO VALIDAMOS?: CON USUARIOS FINALES

Evaluar el sitio con usuarios finales

- Es necesario testear el sitio con usuarios con discapacidad para analizar si es posible el acceso a toda la información.
- Trabajar con distintos perfiles del usuario, incluyendo adultos mayores, para observar su interacción y costos para llevar a cabo las tareas.
- Se debe definir la muestra, definir las actividades a realizar, diseñar la planilla de registro y luego analizar los resultados.

Material elaborado por la Prof. Ivana Harari, en su presentación "Introducción a la accesibilidad digital".

VALIDADORES

HTML y CSS

- Validador HTML: <http://validator.w3.org/>
- Validador de Hojas de Estilos: <http://jigsaw.w3.org/css-validator/>
- NU: <https://validator.w3.org/nu/>

VALIDADORES

Accesibilidad

- WAVE: <http://wave.webaim.org>
- TAW: <http://www.tawdis.net>
- Tingtun Checker: <http://checkers.eiii.eu/>
- [Lighthouse](#)
- Herramientas de la W3C: <https://www.w3.org/WAI/ER/tools/>

CURSO DE ACCESIBILIDAD WEB DE LA FACULTAD

Curso a distancia Accesibilidad Web *XIII Edición*



Gratis / Cupos limitados

Inicio: Lunes 13 / 9 . **Finaliza:** Sábado 14 / 11

Inscripción en: <https://actividades.linti.unlp.edu.ar>

Contacto: accesibilidad@info.unlp.edu.ar

Abierto a la comunidad / Se entregarán certificados

+Info

ALGUNAS REFERENCIAS

- Principios WAI
- Componentes WAI
- Descripción general de las WCAG 2
- Accessible Rich Internet Applications (ARIA) (Accessible Rich Internet Applications (ARIA) son un conjunto de roles y atributos que definen formas de hacer que el contenido y las aplicaciones web (especialmente aquellas desarrolladas con JavaScript) sean más accesibles para las personas con discapacidades.)
- All the tags
- Curso HTML and CSS - Lecture 0 - CS50's Web Programming with Python and JavaScript

SEGUIMOS LA PRÓXIMA ...

Speaker notes