

Tendencias y entornos para el diseño web en 2024

Mtro. Marco Diego Vargas Ugalde

Licenciatura en Comunicación Social, UAM-X



Han pasado casi 35 años desde que Tim Berners-Lee lanzó los primeros desarrollos de la World Wide Web, un *software* que permitió transmitir y recibir información entre computadoras conectadas por medio del protocolo HTTP, y con el lenguaje HTML y una dirección en la red URL¹.

Desde hace unos 20 años, ya podíamos aprender el lenguaje HTML en una simple guía paso a paso que se hallaba en la red. Seguramente, muchos de los que empezaron a crear sitios web, a inicios de este siglo, recordarán Webmaestro como uno de los primeros espacios que tenía bien organizadas las lecciones, una explicación y ejercicio del uso de cada etiqueta HTML. En aquel entonces, el HTML 4.0 era el estándar en el que se escribían las páginas web; todo lo que había ahí era suficiente en términos de código para que cualquier persona tuviera una página web. Las posibilidades gráficas de las computadoras eran bastante limitadas en comparación con la inmensidad de características que tienen los dispositivos

digitales de esta segunda década del siglo XXI, la resolución de las computadoras era baja. La profundidad de color de aquellos monitores CRT era de ocho bits cosa que ha cambiado en computadoras y dispositivos móviles, permitiendo que ciertos equipos modernos tengan profundidades de color de 10 bits. Esto representa un cambio importantísimo en cuanto a la percepción de color para el usuario, pues pasamos de 16 a 1000 millones de colores.

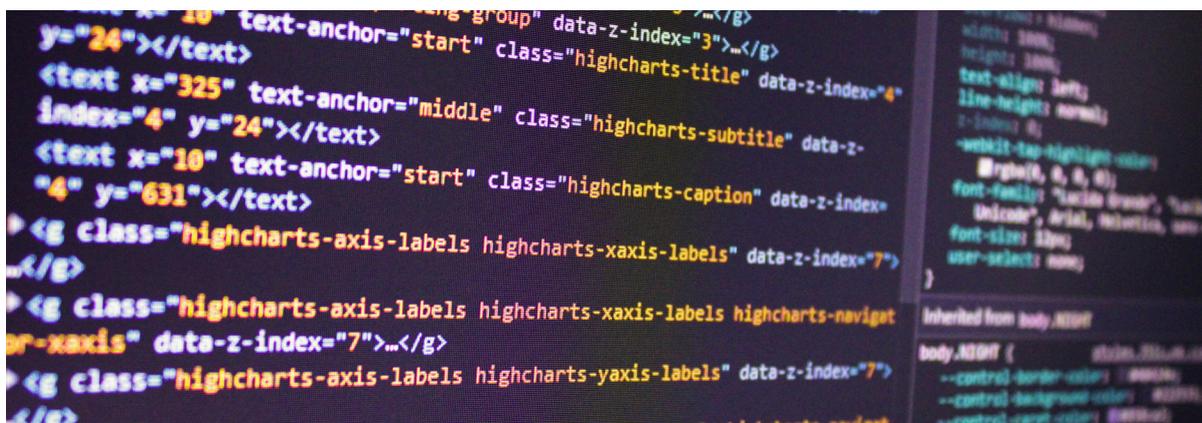
El camino del diseño web ha estado cambiando desde su aparición hasta nuestros días. Podríamos mencionar que en cada una de las áreas de las que depende un sitio web ha habido transformaciones que dieron paso a nuevos tipos de sitios en los que la interactividad, usabilidad y responsividad se han visto nutridas. La web que conocimos a mediados de los 90 y los primeros años del 2000, en

Figura 1: Black flat screen computer monitor.

En: <https://unsplash.com>



¹ Manuel Castells, *La galaxia internet: Reflexiones sobre internet, empresa y sociedad*, Debolsillo, Barcelona, 2001.



la que todo era fijo, no había manera de que los usuarios interactuaran, ni que las páginas identificaran características de los usuarios y les enviaran un contenido específico, es decir, la interactividad estaba limitada a dar clic en enlaces que nos mostraban una nueva página estática. Los colores de fondo cambiantes, los gifs y los bordes redondeados con tablas y decenas de imágenes que podían crear diseños más orgánicos hacían que diseño y desarrollo se vieran como dos líneas paralelas y nunca se encontraran.

A finales de la primera década de este siglo, el HTML había tenido varias mejoras, las hojas de estilo en cascada (CSS) y el JavaScript se habían posicionado sólidamente en el diseño web (Front end) y el PHP y ASP se ganaron a los desarrolladores que incorporaron estos lenguajes. Pasó la época del uso casi indiscriminado de páginas con SWF (Flash), animaciones e interactividad bastante vistosa pero que consumía mucho procesador y ancho de banda. Todo estaba lleno de *perloaders*, pantallas de precarga que desesperaba a los usuarios y muchas veces provocaba que se abandonara el sitio. La sólida integración del HTML5, CSS3 y JavaScript se afianzó en la red cuando la proliferación de dispositivos móviles que no soportaron Flash se volvió una realidad. Los smartphones revolucionaron al mercado de la telefonía, pero también provocaron una transformación en el diseño web pues cada vez más usuarios de celulares empezaron a navegar por internet, Para el 2015 más y más compañías de telefonía celular crearon paquetes más accesibles para que más usuarios tuvieran acceso a la red desde la palma de su mano y en cualquier sitio. Evidentemente el uso de pantallas pequeñas implicó que los diseños se ajustaran en tamaño y disposición de textos, imágenes y videos (*layout*), lo que provocó que lo que se veía bien en una pantalla de computadora también tendría que verse bien en una pantalla de celular. Según la Asociación Mexicana de Internet², en 2015 había cerca de 54 millones de personas con acceso a internet en México, con un tiempo aproximado de seis horas diarias en internet a partir del uso de



Figura 2: Monitor de computadora.
En: <https://www.cnnbrasil.com.br/tecnologia/hackers-do-bem-sao-recompensados-ao-testar-os-sistemas-de-grandes-empresas-2/>

distintos dispositivos, principalmente la computadora de escritorio o portátil, sin embargo, el teléfono móvil el 58% de la gente usaba el celular para conectarse. Ahora, casi diez años después, el acceso a internet ha crecido, en el estudio sobre los hábitos de uso que hizo la AMIPCI, se reporta que cerca del 80% de habitantes de México tiene acceso a la red. Plazas, parques, cafeterías, el trabajo, la escuela y el hogar son los lugares donde la gente se conecta a Internet³, es decir, en prácticamente cualquier parte (esto es en ciudades) y las redes WiFi o el 4G son lo que brinda estos accesos. Todos estos usuarios necesitan que las páginas que consultan en la red, se vean y funcionen adecuadamente en su dispositivo, ya sea computadora, tablet o celular, e incluso la TV. Por supuesto que no todo es a través de páginas web, mucho del contenido que está en internet se puede consultar por medio de aplicaciones específicas como las redes sociales, videos, bancos, finanzas, mapas, entre otros. Así que los sitios deben tener disponible su contenido para ser visto y consumido por diversos medios.

² AMIPCI, *Estudio de los Hábitos de los Usuarios de Internet en México*, AMIPCI, 2015.

³ AMIPCI, *19 Estudio sobre los Hábitos de Usuarios de Internet en México*, Ciudad de México, 2023.

A continuación, propongo algunos elementos fundamentales para adentrarse al diseño web y que podrían verse como diagrama de flujo para proyectos de diseño e incluso de desarrollo web.

¿RUTA CRÍTICA PARA CREAR UN SITIO WEB?

• Crear un buen *brief* ⁴.

El documento base para iniciar cualquier proyecto de diseño es también para el trabajo web un documento imprescindible. Sin el resumen del proyecto o *brief*, no hay manera de que este sitio funcione. En el *brief* debemos trabajar con nuestro *cliente*⁵ todos los elementos que desea que tenga el sitio web, por ejemplo, el público al que va dirigido, el objetivo del proyecto, es decir vender productos, difundir, informar o vincular. Deben quedar claras las funcionalidades del sitio, ya que de ahí dependen los requerimientos técnicos para alojar el sitio web, los lenguajes de programación que deberán usarse, niveles de interactividad, adaptabilidad a dispositivos como teléfonos móviles o computadoras portátiles. Igualmente qué tipos de actividades podrán hacer los usuarios cuando entren al sitio, y también qué tipo de contenidos va a presentar cada página, fotos, textos, videos, audios

Figura 3: Monitores.
En: <https://www.pexels.com/es-es/foto/monitores-de-apple-326518/>



o todo junto. El *brief*, como siempre, es el material estructural de cualquier proyecto creativo y solamente puede elaborarse bien cuando trabajamos en conjunto con quien requiere la solución de diseño. En muchas ocasiones, es complicado para quien tiene la necesidad de un sitio, explicar en términos técnicos sus ideas y deseos y por supuesto es imposible que los diseñadores adivinen lo que está en la mente de quien necesita un sitio de internet, por ello, el trabajo en equipo es clave.

Cuando hablamos de las tendencias del diseño web, es importante comprender que debemos actualizarnos constantemente, para lograr transmitirle a nuestro cliente qué novedades existen y qué estilos han quedado obsoletos tanto en el aspecto gráfico como en la funcionalidad del sitio, así como en la administración de los contenidos. Una vez más es a partir del *brief* que podríamos dilucidar si lo que requiere nuestro cliente puede resolverse con una simple página en Facebook o si requiere algo más complejo como un Sistema de Gestión de Contenidos (*Content Management System*, cms por sus siglas en inglés) que va a requerir muchas otras cosas para crearse; también si se trata de una plataforma educativa, una tienda virtual, un blog o una revista. Existen cms para cada uno de estos tipos de sitio y además miles de plantillas o *templates* para acondicionar el diseño para cada proyecto. Dentro de éstas, hay diseños gratuitos y con costo. Además de que existen *plugins* o *add-ons* que permiten agregar funcionalidades extra al sitio.

Una vez definido el *brief*, tendremos información suficiente para determinar el tipo de solución que daremos al proyecto. En principio, podríamos elegir entre tres tipos de solución basándonos en el nivel de interactividad, funciones y despliegue de información.

⁴ *Brief* se define como un documento que establece los objetivos de un proyecto o tarea a realizarse.

⁵ Se usará el término *cliente*, como aquella persona u organización que solicita un producto o servicio, en este caso el diseño de un sitio web. Es común que se use esta nomenclatura inclusive si en el proyecto no hay ninguna remuneración económica.

• **Primera opción.** Un sitio realizado con HTML5, CSS3 y algo de JavaScript, hecho a la medida siempre y cuando el nivel de interactividad que se requiera sea mínimo. Es decir, un tipo de página más bien informativa, de consulta de información y con pocas opciones de interacción. Este tipo de soluciones permite que los diseños sean mucho más personalizados para el cliente y pueden enriquecerse con Javascript y lograr un sitio dinámico y con efectos visuales atractivos. Este sitio requiere un plan de administración bien establecido con el cliente, pues cada actualización requiere que el webmaster edite los archivos y los actualice en el servidor por medio de una conexión FTP; esto normalmente supone una tarea casi imposible para los clientes. Sin duda, otra característica de este tipo de proyectos obliga a que el diseñador tenga el control de todos los archivos de trabajo y para cada cambio, semanal, mensual o como sea que se establezca, solamente quien posea los materiales y contraseñas de acceso, podrá modificar los contenidos. La creación de sitios con este modelo es adecuada para sitios pequeños, que no requieren actualizaciones frecuentes, donde se puede hacer una propuesta de diseño

gráfico más amplia y que además tiene un tiempo de entrega corto, pues el desarrollo del sitio con HTML y CSS en manos avanzadas es bastante ágil. Algunos sitios que comúnmente acuden a este tipo de solución tecnológica son páginas personales, despachos de abogados o contadores, productos promocionales, avisos de eventos y boletines mensuales.

• **Segunda opción.** Un sitio basado en un cms como wordpress, joomla, Drupal, Moodle, Wix o SquareSpace. Este tipo de sitios tienen muchas ventajas en cuanto a la administración de los contenidos como textos, fotos, videos y demás materiales pues en principio no hay necesidad de saber programación PHP, HTML, CSS, JavaScript ni ningún otro lenguaje, ya que la administración de los contenidos se hace por medio de un panel de texto e iconos muy intuitivos; sin embargo, la configuración del sitio, diseño, armado y arquitectura del sitio sí son responsabilidad del equipo de diseño y tienen un cierto nivel de complejidad. En caso de requerir funcionalidades extra que no posee el propio cms, se deberá acudir a un equipo de programadores que conozcan básicamente PHP y MySQL que son los lenguajes en los que se basa la mayoría de los cms.

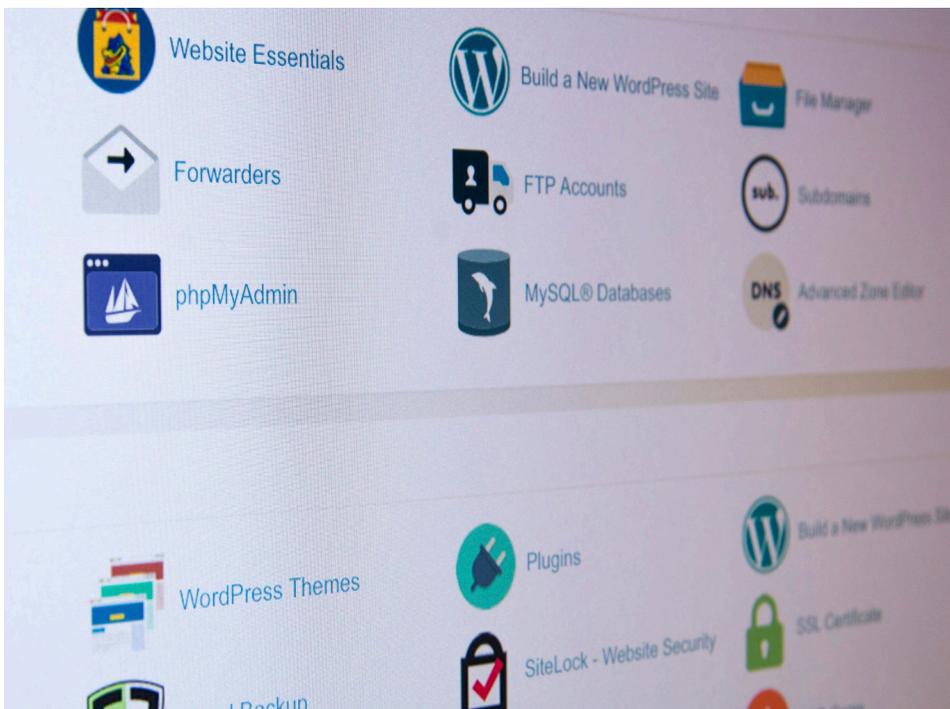


Figura 4: Lote de iconos.

En: https://unsplash.com/es/fotos/lote-de-iconos-surtidos-2kH-6T6x_OI

Este tipo de sitios requieren que todo el contenido sea diseñado por el equipo de diseño y después colocarlo en las plantillas del cms. Textos, imágenes, videos, fondos, banners, etc. La gestión de contenidos es directamente online, tanto el diseño como la incorporación de los contenidos se hace por medio de un sistema de administración de bases de datos que permiten habilitar y deshabilitar módulos del programa a conveniencia del proyecto. Planear un sitio basado en cms responde principalmente a la necesidad de una actualización constante de los contenidos, un manejo de grandes cantidades de información, posibilidades de interacción avanzada con el público usuario, administración de los contenidos por múltiples administradores, colaboradores, editores e incluso visitantes del sitio. Algunos ejemplos de los tipos de sitios que usan cms son las aulas virtuales o plataformas educativas, tiendas virtuales, revistas electrónicas, blogs, galerías y museos virtuales, sitios de universidades y colegios, periódicos online, agencias de información, organizaciones civiles, campañas publicitarias o partidos políticos. Vale la pena destacar que también estas plataformas por su accesibilidad son usadas para páginas personales y todas las que se mencionaron como parte de la primera solución, aunque esto modifica el costo del proyecto. Actualmente estas plataformas de administración web se nutren de Inteligencia Artificial y eso puede ser un mecanismo que ahorra tiempo y recursos, pues reduce el tiempo de producción de imágenes.

- **Tercera opción.** Un sitio creado con HTML5, CSS3, JavaScript, PHP, MySQL y otros lenguajes más para construir un sistema de administración de contenidos diseñado a la medida, específicamente para dar solución a problemáticas concretas y donde los cms tradicionales no brindan el rendimiento, seguridad o funciones requeridas. Este tipo de solución es, sin duda, la más costosa y la que más tiempo requiere para su construcción. Para este tipo de abordajes, se requiere a un equipo conformado por muchas más personas, inicialmente un *web architect*⁶,

seguido por *web designers*⁷ y *web developers*⁸. Una solución integral de este tipo es adecuada para grandes portales de información, donde cada pieza es elaborada cuidadosamente y ensamblada por medio de un equipo de expertos en cada rama. El nivel de interactividad, seguridad, funcionalidad y dinamismo de este tipo de sitios es crítico y por esa razón deben diseñarse las piezas a la medida, teniendo todo el control de los módulos. Algunos tipos de sitios que recurren a este tipo de soluciones son juegos *online*, centros de investigación, oficinas gubernamentales, bibliotecas, instituciones de salud o bancarias, sistemas de ventas por internet, soporte técnico y oficinas virtuales, buscadores web, plataformas de *streaming* de video y música, comunidades virtuales y redes sociales. Normalmente este tipo de portales son respaldados por grandes empresas o bien oficinas de gobierno con presupuestos muy grandes en donde, debido a las características del sitio web, es importante tener contratado al equipo de diseño y desarrollo web que dé mantenimiento y soporte a todas y cada una de las áreas del portal.

⁶ El *web architect* es normalmente quien se encarga del trabajo del brief con el cliente y de lograr traducir todas sus necesidades a las posibilidades de creación del sitio. Es también quien se encargará de comunicarse con los diseñadores y programadores sirviendo de puente entre estas dos áreas para lograr que el proyecto avance en la misma dirección siempre.

⁷ Los *web designers* son quienes se encargan del diseño gráfico del sitio, elementos de color, composición, legibilidad, ilustración, experiencia del usuario (UX) y distribución de la información. Es ideal que sepa escribir código como HTML, CSS y algo de JavaScript, pero no es indispensable.

⁸ Los *web developers* son quienes se dedican a la programación del sitio y los hay de dos tipos, Front-end y back-end. Los primeros son los que se encargan de convertir los elementos gráficos del diseño en elementos, etiquetas y objetos web a partir del uso de HTML, CSS y algo de JavaScript, y los segundos son quienes se encargan de la programación más compleja como las bases de datos, interactividad avanzada.

LOS FRAMEWORKS DE CSS LA HERRAMIENTA Y LA TENDENCIA

Hemos visto cómo al paso del tiempo se ha complejizado la tarea de crear un sitio en internet; hace 25 años, prácticamente cualquier persona podía aprender HTML y conectarse a un servidor gratuito como *geocities* y colocar ahí sus archivos para que cualquier otro navegante de internet pudiera encontrarlos. Desde luego que las posibilidades gráficas de las computadoras de aquel tiempo no eran muchas y tampoco las que el código permitía, así que estábamos acostumbrados a ver sitios bastante simples, planos y estáticos salvo por los *GIFs* que se usaban indiscriminadamente pensando que eso le daba “movimiento” a la página. A finales de los noventa y principios del año 2000, las páginas creadas con *Macromedia Flash* inundaron la red. La posibilidad gráfica que tenía el programa permitía crear sitios con mucho mejor diseño, pero también el abuso de esa tecnología complicó las cosas, pues era necesario un flash player para que las computadoras pudieran reproducir el material del sitio, eso sí, muchísimas páginas web lograron avances extraordinarios en animación e interactividad gracias a Flash. Entonces, llegó el HTML5 y el CSS3 impulsados por la tecnología móvil y para el 2010, Flash (ahora ya de la empresa Adobe) se volvió obsoleto, pues los teléfonos celulares y tablets no aceptaron páginas creadas con Flash, sino con los nuevos estándares de HTML. Los teléfonos celulares con conexión a internet revolucionaron el quehacer humano a nivel mundial y dieron un fuerte impulso a la transformación de la red. Conforme fueron avanzando y el HTML5 y CSS3 se propagaron, el diseño responsivo se consolidó. Así nació la web semántica, un gran paso en términos de la comunicación entre las computadoras.

Evidentemente, el crecimiento de usuarios de internet a nivel mundial ha traído consigo la necesidad de más y mejores sitios web que se adapten a las características de cada uno de esos usuarios, que bien pueden estar en una computadora o en un celular; en una televisión o en un avión. Cada usuario deberá tener la posi-

bilidad de ver adecuadamente los contenidos del sitio que visite y el sitio deberá ser capaz de reconocer las características del dispositivo para poder entregar la mejor versión del sitio.

El diseño responsivo comenzó desde que apareció la posibilidad de realizar ajustes automáticos en las páginas web a partir de detectar la resolución del monitor, sin embargo, en aquel tiempo, pensar en dos versiones de sitio para 800 x 600 y 1024 x 768 hacía las cosas simples en comparación con nuestros tiempos, donde una página tiene que ser más flexibles por la inmensa cantidad de tamaños de pantalla en las que se podría ver: celulares en modo vertical y horizontal, tablets grandes y chicas, horizontales y verticales, computadoras, televisiones, relojes, automóviles y hasta refrigeradores.

Los *frameworks* de CSS son pequeños programas que nos presentan un entorno con ciertas funcionalidades de las Hojas de Estilo en Cascada (CSS), en las que podemos apoyarnos para diseñar nuestro sitio; su uso requiere los mismos conocimientos de CSS intermedio. Normalmente los *frameworks* contienen una serie de reglas, estilos y ajustes genéricos, así como una estructura reticular⁹ que permite diseñar en módulos basados en el ancho de cada columna. Este paquete de estilos posibilita crear sitios mucho más rápidos y con más precisión y mejor composición que al hacerlo de manera intuitiva, además los sitios basados en *frameworks* van a ser compatibles con una gran diversidad de dispositivos puesto que, al diseñar la interface, el diseñador podrá prever que el usuario se conecte por medio de un dispositivo distinto cada vez y el sitio siempre le entregará una versión adecuada a las características de su pantalla^{10 11}. Otra de las ventajas de

⁹ La retícula o *grid system* es otro paquete CSS que divide el ancho de la página en varias columnas del mismo ancho y así permite la creación de módulos.

¹⁰ Nikhil. CSS Grid Layout vs CSS Frameworks: When To Use What?, 18 de noviembre de 2019.

¹¹ Themeselection, Best CSS Frameworks in 2021, 20 de diciembre de 2020. Obtenido de dev to. En: https://dev.to/theme_selection/best-css-frameworks-in-2020-1jjh

usar estos marcos de trabajo, es el cumplimiento de estándares web que cada vez son más precisos y al seguirlos, permitimos que nuestro sitio tenga mejores resultados al ser usado, indexado y registrado por los buscadores web. Tanto los sitios creados con *frameworks* como con sistemas de retículas, van a ser visibles por prácticamente todos los navegadores web contemporáneos.

Algunos de los *frameworks* más utilizados por los desarrolladores son Bootstrap, Tailwind, Foundation, Materialize y Unsemantic. Todos estos entornos CSS han aportado en el diseño web, facilitándole a diseñadores front-end la tarea de crear sitios, simplificando labores, estandarizando código y fomentando buenas prácticas en el diseño web.

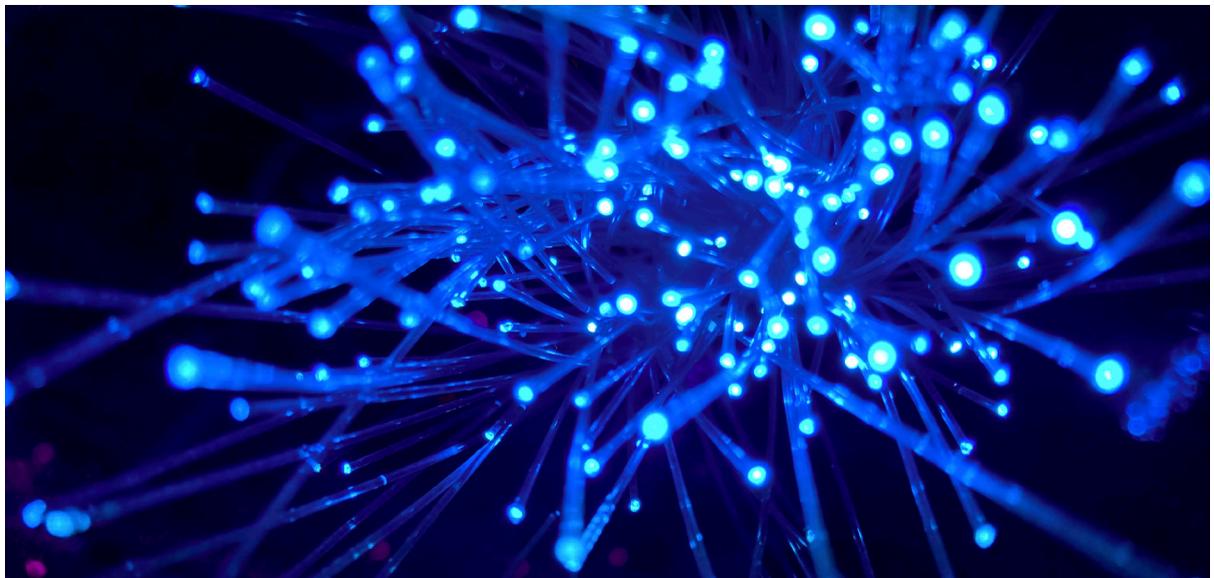
Seguramente, veremos en los sitios de esta década un incremento en el uso de estos entornos y el surgimiento de nuevas tecnologías que fortalecerán la red desde el fondo de las líneas de código.

INTELIGENCIA ARTIFICIAL

Hace apenas dos años, se desató a la bestia. Fueron liberadas varias herramientas de inteligencia artificial en la red y su uso provocó un estruendoso impacto en la comunidad de creativos, tanto creadores de imágenes, como de textos vieron en la IA una aparente amenaza. Mucha gente creyó que perdería su trabajo como

diseñador o copy en las agencias, despachos, escuelas y demás. La velocidad con la que la IA era capaz de producir imágenes o textos desconcertó y preocupó a muchos. Docentes y creativos nos volteamos a ver y nos preguntamos si era el fin de una era y el principio de otra, y sí. Así es. Las plataformas para crear imágenes, textos, videos y sonidos a partir de los algoritmos de la conocida IA nos coloca en la era que vimos en el cine hace décadas. Solo que por ahora no son máquinas queriendo tomar el control, sino herramientas para agilizar el trabajo humano. Sin abundar más en ese tema, sí vale la pena mencionar que en muy poco tiempo, la IA ha aprendido y ahora somos capaces de emplear sus recursos para realizar tareas de diseño a velocidades nunca antes vistas. ¿Cómo usarla? es una pregunta que no tiene una respuesta simple, sin embargo, en términos generales podemos decir que debemos aprender a pensar como programadores, haciendo *prompts* que la IA pueda entender con claridad para que nos dé resultados más o menos satisfactorios y que podamos usar. Así, los tres tipos de diseño/desarrollo de páginas web que mencioné pueden verse enormemente nutridos y agilizados si empleamos, por ejemplo ChatGPT para la escritura de código HTML5, CSS y Javascript que podemos incrustar en nuestro sitio prácticamente sin saber lo que dice ese código. Hay plataformas espe-

Figura 5: Fondo de pantalla digital.
En: <https://unsplash.com/es/fotos/fondo-de-pantalla-digital-de-luz-purpura-y-azul-8bghKxNUij0>



cíficas para programadores que brindan revisiones de código o escritura desde cero con un nivel de funcionalidad casi sin margen de error. Existen otras plataformas para la creación de imágenes que permiten una versatilidad casi infinita para vestir un sitio web con estilos, paletas de color, layouts o íconos que reducen el tiempo de ilustración, vectorizado, render o dibujo a unos minutos. Existen también plataformas específicas para crear sitios web completos con IA. Pero claro, no todo es gratis en la vida. Para tener el acceso completo a las funcionalidades de varias de estas plataformas de IA generativa, hay que pagar alguna tarifa. Por suscripción, por tiempo determinado o por ciertas funcionalidades.

CONCLUSIONES

Han pasado 30 años desde que se hizo la primera conexión a internet en México, y en este periodo hemos visto pasar y quedarse tecnologías para navegar en la red. Hemos aprendido distintos lenguajes, modalidades, terminología, dispositivos y usos. Se crean nuevos puestos de trabajo reemplazando el antiguo webmaster (el que lo hacía todo) por *Web developer*, *Front End Designer*, *UX (User Experience Design - Diseño de la experiencia de usuario)*, *Web architect* o *User Interface Design*. Y seguiremos viendo y aprendiendo de los cambios que las nuevas prácticas con IA nos pone sobre la mesa. Sin duda, el diseño asistido por computadora tiene un nuevo componente y deberemos aprender a usarlo para que en verdad sea un recurso y no un impedimento, o como creyeron muchos, el fin de los diseñadores (programadores y escritores también), sino que se convierta en una herramienta específica para hacer avances en tiempo y así podamos dedicar más tiempo a diseñar, crear, imaginar y concebir los proyectos con la mente y además, nos quede tiempo para leer, pasear y estar con nuestros seres queridos, es decir, que la IA nos sirva para tener más tiempo libre y disfrutar la vida en vez de esclavizarnos detrás de una pantalla sin que nos dé el sol esperando que el error de un código o un render

termine. Por ahora, las plataformas generativas están aprendiendo de nuestro uso, las estamos entrenando y dentro de unos años, serán más potentes y se habrán integrado en más actividades del diseño web, así que nuestra tarea está en entrenarlas bien, usarlas con responsabilidad, ética y cuidado. Finalmente, apuntaré algunos sitios que ofrecen distintas herramientas de IA para apoyarnos en el diseño web.



Figura 6: Mano metálica e inteligencia artificial.

En: <https://intelektosoluciones.com/humansite>



Figura 7: Persona con un teléfono celular.

En: <https://unsplash.com/es/fotos/un-primer-plano-de-una-persona-tocando-un-telefono-celular-GZyELVkOmi0?>

PARA PROGRAMACIÓN

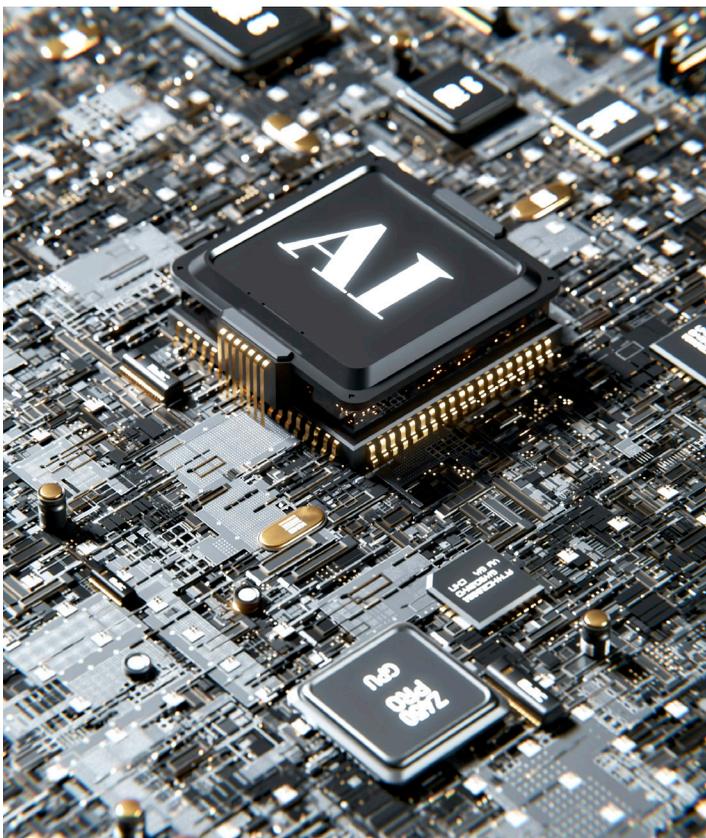
- OpenAI GPT. Permite obtener código funcional de distintos lenguajes de programación.
- GitHub Copilot. Asistente para la escritura de código. Puedes subir tu código y la plataforma te ayuda a corregir y afinar.

PARA CREACIÓN DE IMÁGENES

- Adobe Firefly. Al contar con una licencia de la suite de Adobe, tendrás acceso a la plataforma generativa de imágenes del gigante del software de diseño, Adobe.
- Canva. Sí, esta popular plataforma en la que se pueden crear presentaciones como el antiquísimo powerpoint, también ha integrado una herramienta de IA y por una suscripción puedes acceder a todas sus funciones.
- Dall-E. Una buena herramienta, pero limitada a dos solicitudes al día para la creación de imágenes.
- LimeWire. Tiene plan gratuito y con costo. Es versátil y también puedes generar audios.

Figura 8: Chip de computadora.

En: <https://unsplash.com/es/fotos/un-chip-de-computadora-con-la-letra-a-en-la-parte-superior-eGGFZ5X2LnA>



PARA CREACIÓN DE SITIOS WEB COMPLETOS

- Dora AI. Crea sitios de principio a fin. El resultado puedes modificarlo según tus necesidades.
- Kittl AI. Variedad de herramientas que te permiten crear y editar imágenes y textos.
- Figma. Es un entorno de diseño colaborativo que integra nuevas opciones de trabajo con IA.



Referencias

AMIPCI, *Estudio de los Hábitos de los Usuarios de Internet en México*, AMIPCI, 2015.

AMIPCI, *19 Estudio sobre los Hábitos de Usuarios de Internet en México*, Ciudad de México, 2023.

Castells, Manuel, *La galaxia Internet: Reflexiones sobre internet, empresa y sociedad*, Debolsillo, Barcelona, 2001.

Koishigawa, K, *A Brief History of Responsive Web Design*, 4 de febrero de 2001. Obtenido de Free code camp. En: <https://www.freecodecamp.org/news/a-brief-history-of-responsive-web-design/>

Manovich, L., *El lenguaje de los nuevos medios de comunicación*, Paidós, Barcelona, 2011.

Nikhil., *CSS Grid Layout vs CSS Frameworks: When To Use What?*, 18 de noviembre de 2019.

Obtenido de Lamda test. En: <https://www.lambdatest.com/blog/css-grid-layout-vs-css-frameworks/>

Themeselection, *Best CSS Frameworks in 2021*, 20 de diciembre de 2020. Obtenido de dev to. En: https://dev.to/theme_selection/best-css-frameworks-in-2020-1jjh