**FORMATO PARA EL DESARROLLO DE COMPONENTE FORMATIVO**

| Programa de formación | Desarrollo Multimedia y Web |
| --- | --- |

| Competencia | 220501102  Integración de los elementos gráficos, audiovisuales e interactivos del proyecto. | Resultado de aprendizaje | 220501102-01  Integrar los componentes del proyecto de acuerdo con la estructura y la experiencia de usuario definida. |
| --- | --- | --- | --- |

| Número del componente formativo | 010 |
| --- | --- |
| Nombre del componente formativo | Integración de componentes del proyecto multimedia |
| Breve descripción | Este contenido está enfocado en la integración de los elementos gráficos y visuales que componen un proyecto que se desarrolle en el entorno *web*, identificando los formatos que se deben utilizar para una óptima funcionalidad de la interfaz del usuario. |
| Palabras clave | CMS, formatos, maquetación, modelo, servidor |

| Área ocupacional | 2 - Ocupaciones en ciencias naturales, aplicadas y relacionadas |
| --- | --- |
| Idioma | Español |

1. **Tabla de contenido**

**Introducción**

**1.Fundamentos de la arquitectura cliente/servidor**

1.1 Tipos de modelos cliente/servidor

1.2 Integración del modelo cliente/servidor

1.3 Funcionamiento del cliente en la *web* – *Frontend*

1.4 Funcionamiento del cliente en la *web* – *Backend*

1.5 Integración del *Frontend* y el *Backend*

**2. Integración multimedia**

**2.1 Formatos para la integración multimedia**

**2.2 Plataformas y gestores de contenido para la integración multimedia**

**3. Estructuras de las aplicaciones**

**Introducción**

Bienvenido a este nuevo componente formativo, que le permitirá continuar con el recorrido por las herramientas tecnológicas que conforman el entorno *web* para el desarrollo de proyectos multimedia. El conocer cómo operan estas tecnologías le permitirá una apropiación sobre lo que ocurre en el funcionamiento por parte de los usuarios y lo que ocurre al interior de la red de Internet donde se encuentran los servidores que prestan los servicios que solicitan los clientes día a día.



Para la elaboración de este componente, se abordaron varios autores conocidos en **integración de componentes del proyecto multimedia**, de quienes se han citado y referenciado conceptos y ejemplos para los fines educativos de esta materia, en el entendido de que el conocimiento es social y, por lo tanto, es para ser usado por quienes necesitan adquirirlo. Se espera que este documento sea útil para todos aquellos, aprendices y lectores en general, que estén interesados en acercarse a asuntos básicos del **desarrollo Multimedia y Web**.

**Fundamentos de la arquitectura cliente/servidor**

Todos los servicios que ofrece la *web* se basan en una relación directa entre los servidores remotos que atienden las peticiones que hacen los usuarios o clientes para acceder a una aplicación, sitio o producto multimedia que se encuentra en Internet. Para entender dicha relación, es necesario apropiar los conceptos de servidor y de cliente en el ecosistema de las redes:



De acuerdo con lo anterior, el modelo cliente/servidor es una configuración arquitectónica de elementos, que se lleva a cabo en el entorno de la *web*. Dichos elementos se componen de equipos o dispositivos que operan los usuarios, equipos servidores con funciones especiales, que prestan servicios a los clientes de manera remota, y redes de Internet que se encargan de establecer la comunicación entre unos y otros respectivamente. Este modelo constituye, entonces, un sistema en el cual se distribuye una infraestructura de elementos, donde los datos parten, hacen un recorrido, se procesan y regresan convertidos en información útil.

* 1. **Tipos de modelos cliente/servidor**

En los modelos, se pueden encontrar diferentes tipos, los cuales se conocerán a continuación:



* **Modelo tradicional o modelo de dos capas:**

* 1. **Interacción del modelo cliente/servidor**

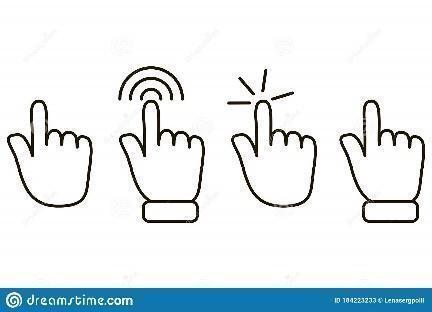
Con lo que se ha explicado hasta ahora del modelo cliente/servidor, se puede determinar que, en el funcionamiento del sistema como tal, se produce una intercomunicación de los dispositivos de *hardware* que intervienen, los cuales requieren de componentes de *software* para completar el ciclo de petición del cliente.













Del lado del servidor, la interacción se logra cuando en estos dispositivos se incorpora *software* que procese lenguajes de programación y bases de datos como *PHP, MySQL, MongoDB*, entre otras.

El servidor ejecuta las acciones necesarias mediante el *software* que posee, es decir, el sistema operativo Linux y el *software* de servidor ***web*** Apache, el cual interpreta y ejecuta el lenguaje de programación PHP, los cuales se conectan con la base de datos MySQL, realizando un proceso para generar un archivo resultante HTML, CSS y JavaScript, que es el que envían al navegador para devolver el sitio ***web*** solicitado por el cliente. Las combinaciones de las herramientas tecnológicas -servidor ***web***, sistema operativo, lenguaje de programación, base de datos- del lado del servidor, llamadas ***Stack,*** pueden ser otras diferentes a las que se muestran en el ejemplo, esto depende de los lenguajes de programación y bases de datos que utiliza la entidad para la prestación de servicios ***web***.

El proceso anterior muestra la interacción que se lleva a cabo entre cliente y servidor(es), cuando se hace una solicitud a la ***web***, la cual es un ciclo de desarrollo cliente-servidor-cliente para acceder a las bases de datos, y, una vez resuelta la solicitud, el ciclo se rompe. Lo anterior significa que el ciclo solamente se activa cuando el servidor ejecuta las instrucciones del lenguaje en el que están escritas las páginas ***web*** solicitadas por el navegador, es decir, que el cliente nunca puede acceder directamente a las bases de datos de los servidores.

| Lectura Complementaria | |
| --- | --- |
| Para profundizar en los conceptos del modelo cliente/servidor consultar: | Sistemas Distribuidos. Capítulo 3. Modelos de Arquitecturas. (página 37 a la 52)  <http://dccd.cua.uam.mx/libros/archivos/03IXStream_sistemas_distribuidos.pdf> |

* 1. **Funcionamiento del cliente en la *web* -*Frontend***

Como se explicó antes, el cliente es un dispositivo de *hardware* que dispone de herramientas de *software* para establecer comunicación con el usuario y los servidores de la *web*.

“ Para que el cliente pueda iniciar un ciclo de interacción con la *web,* debe poseer un navegador, dentro del cual se ejecuten lenguajes de código que no solo configuren la apariencia de lo que el usuario ve en la interfaz (pantalla), sino que realice peticiones de servicios a los equipos remotos en la red, y a esta forma de funcionamiento del cliente se le llama ***Frontend***.***”***

A continuación, podrá observar un video explicativo sobre *Backend y Frontend*

| Video Explicativo  DI\_CF010\_1.3A\_Backend-Fronted\_Video |
| --- |

| El *Frontend* es entonces la forma de funcionamiento del cliente dentro del ciclo de interacción cliente-servidor-cliente, y se lleva a cabo en el navegador en el que se ejecutan el lenguaje de etiquetas *HTML,* para estructurar y organizar el contenido de la web, el lenguaje de código de estilo CSS, para darle el formato visual al contenido, y el lenguaje de programación *JavaScript,* que contiene los *scripts* que, al ser activados a través de una petición, conectan al cliente con el servidor. | Top 8 Tendencias Desarrollo Web para 2019 | by Hektor Profe | Medium |
| --- | --- |

.

Para que desde el cliente se pueda realizar la conexión con los servidores remotos, existen varios métodos, los más utilizados son: *Get, Post, AJAX, Sockets*. Estos métodos consisten en incluir líneas de código que se activarán para establecer contacto con los servidores y entregar la información solicitada al cliente. A continuación, se explicarán *GET y POST,* que son los métodos más comunes.

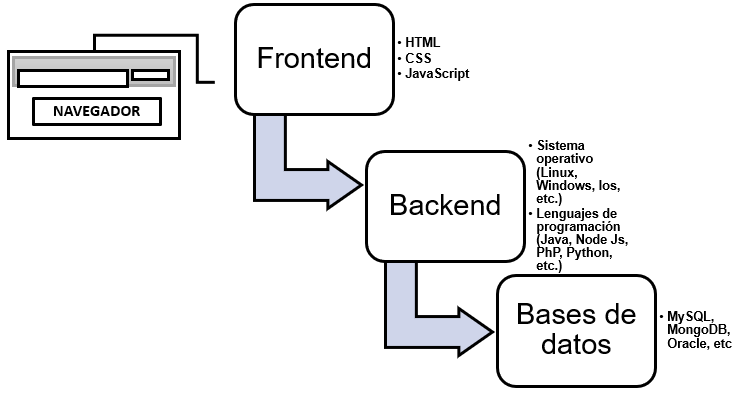


* 1. **Funcionamiento del servidor en la *web - Backend***

El servidor ***web*** es un dispositivo de *hardware* que dispone de herramientas de *software* para establecer comunicación con el cliente y las bases de datos de la ***web***. Para que el servidor pueda seguir con el ciclo de interacción que ha iniciado el cliente en la ***web***, requiere de *software* compuesto por lenguajes de código que procesen los datos enviados y que se conecte con las bases de datos para devolver la respuesta a la petición. A esta forma de funcionamiento del servidor se le llama ***Backend.***

El *Backend* es entonces la forma de funcionamiento del servidor ***web*** dentro del ciclo de interacción cliente-servidor-cliente, en el que se lleva a cabo el procesamiento de los datos que recibe utilizando lenguajes de código (*Java, PHP, Python y Node Js*, entre los más conocidos), para conectar al cliente con las bases datos que solicita.





En la figura, se observa el recorrido que debe hacer el cliente para poder conectarse con las bases de datos de un servidor: primero tiene que pasar, a través del navegador, por el ***Frontend,*** cuyas tecnologías son el *HTML, CSS, JavaScript*; seguir a través del ***Backend,*** el cual recibe los datos y los procesa con tecnologías como el sistema operativo (*Linux, Windows, iOS*, etc.) y los lenguajes de programación (***Java, Node Js, PHP, Python***, etc.); para finalmente enviarlos a las bases de datos (***MySQL, MongoDB, Oracle***, etc.). Es claro entonces que el cliente no puede acceder a las bases de datos directamente, sino a través del *Backend*, además, no puede acceder a toda la base de datos, sino a la información específica que está solicitando. Esto quiere decir que el *Backend* actúa como un filtro que direcciona al cliente hacia los datos que está buscando en el servidor de bases de datos.

| Complemento | |
| --- | --- |
| Para profundizar en los conceptos *Frontend* y *Backend*: | Lectura: Modelo de procesos para el desarrollo de Front-end de Aplicaciones *web*  <https://revistas.ulima.edu.pe/index.php/Interfases/article/view/1245/1205> |

* 1. **Integración del *frontend* y el *backend***

La integración entre el *frontend* y el *backend* se da cuando se establece la comunicación entre las tecnologías del lado del cliente (*HTML, CSS, JavaScript*) y las tecnologías del lado del servidor (lenguajes de programación, bases de datos, servidor *web*). Las primeras se llevan a cabo en el navegador y las segundas, en el servidor, y para que se comuniquen se utilizan aplicaciones que actúan como intermediarias entre unos y otros. A estas aplicaciones se les conoce como API (*Application Programming Interfaces*) o Interfaz de Programación de Aplicaciones.

La API entonces es una interfaz de comunicación entre aplicaciones que requieren compartir datos.

Un ejemplo de API es la plataforma de pago *PayPal*, la cual permite conectar la tienda *online* con las entidades financieras, para recibirle el pago al usuario. En vez de que cada proveedor desarrolle su propia pasarela de pagos, puede utilizar la que ya está creada, pagando los derechos de uso y ahorrando tiempo y dinero de desarrollo.

De acuerdo con el entorno de comunicación que utilicen las aplicaciones, las API pueden ser locales o remotas.

Conozca algunas:



1. **Integración multimedia**

Hacer una integración multimedia consiste en organizar los archivos del proyecto con su respectiva nominación, formato y ubicación al interior de la aplicación o aplicaciones del producto. Igualmente, se realiza la preparación de los programas necesarios para la instalación, accesibilidad y usabilidad de los usuarios. Actualmente, el diseñador cuenta con herramientas que le facilitan la labor de integración de los elementos multimedia (imagen, texto, audio, video), como los CMS - Sistemas de gestión de Contenidos, que se verán más adelante.

Cuando se desarrolla un proyecto *web* haciendo uso de lenguajes como el HTML, en el proceso de integración, se requiere utilizar líneas de código, por ejemplo, para:



* 1. **Formatos para la integración multimedia**

Los productos multimedia que posean elementos visuales de imagen o audio y que vayan a ser publicados en la *web* deben integrarse teniendo en cuenta su compatibilidad con los navegadores que se encuentran instalados en el cliente, para que puedan ser vistos por los usuarios sin presentar ningún inconveniente. Actualmente, los formatos de los componentes más utilizados y soportados por la *web* son los que se relacionan a continuación.

**Formatos de Imagen**

Reconozca cuáles son los formatos de imagen:



**Formatos de audio**

| *WAV*  *Waveform Audio Format* | Creado por *Microsoft*, incorpora audios sin perder la calidad, ya que no realiza la función de compresión, aspecto que puede llegar a tener un efecto de distorsión en el sonido. |
| --- | --- |
| *AIFF*  *Audio Interchange File Format* | Desarrollado por *APPLE*, por lo tanto, es el que mejor se adapta a este tipo de equipos de cómputo. No pierden calidad, pero ocupan mucha memoria, por lo que se deben trabajar en dispositivos de gran capacidad de almacenamiento. |
| *MP3*  *MPEG-1 Audiolayer-3* | Creado por *Moving* *Picture Experts*, hace parte del formato de video y posee alta capacidad de compresión. Es el formato de mayor compatibilidad con los dispositivos de *hardware* y *software* de la *web*. |
| *ACC*  *Advanced Audio Coding* | Es la versión mejorada del MP3, ya que posee 48 canales de audio, es decir, diferentes formas de escuchar el sonido, como monofónico, estereofónico, cuadrafónico y en tonos altos hasta de 96 Hz. |
| *OGG*  *Códec Vorbis* | Formato de audio que posee hasta 255 canales de audio, es decir, es multisonido, y los tonos del audio alcanzan hasta los 192 Hz, razón por la cual es el más usado en los entornos *web*. |

**Formatos de video:**

Ahora puede revisar los formatos de video que se utilizan para este fin:



* 1. **Plataformas y gestores de contenido para la integración multimedia**

En el entorno *web,* se pueden utilizar plataformas y gestores de contenido para integrar proyectos o productos multimedia, entre los cuales están los portales *web*, los sitios *web* y los CMS.

Explore algunas plataformas que permiten integrar proyectos Multimedia.



* **Gestores de contenido (CMS)**

| Un CMS, siglas en inglés que traducen *Content Management System*, sistema de gestión de contenidos, es un programa desarrollado para que cualquier usuario pueda administrar y gestionar contenidos de un sitio *web,* con facilidad y sin conocimientos de programación; para ello, existen CMS de código abierto, es decir, para programar de manera gratuita, y otros con licencia de pago.  Un CMS permite administrar y gestionar contenidos en un medio digital o en el entorno de la *web*, creando, clasificando o editando información, ya sea en texto, imagen o video. | Logotipo, nombre de la empresa  Descripción generada automáticamente |
| --- | --- |

En la figura, se observan las funcionalidades que puede administrar un CMS, como escribir líneas de código *HTML, CSS* o *JavaScript*, dado que es de código abierto (*open-source*), hacer uso de los servicios de la *web* para incorporar elementos visuales, o enlazar sitios *web* o un dominio requerido, incorporar archivos PDF, gestionar los datos para el análisis de los usuarios y también elaborar y gestionar contenidos. Dentro de los CMS más utilizados, se encuentra *WordPress* y Drupal, cuya característica principal, además de todas las anteriores, es que son de fácil configuración para diseños personalizados.

Características claves a la hora de definir un CMS:

* Potentes herramientas de publicación.
* Plantillas *responsive* (diseño responsivo), que se adaptan a PC, móvil, o tabletas*.*
* Flexibilidad.
* Escalabilidad.
* Control de publicaciones.
* Herramientas SEO integradas.
* Integración con redes sociales.
* Analítica detallada.
* Fácil adaptación a los sistemas.
* Seguridad.

A continuación, se relacionan algunos de los CMS más utilizados actualmente:

| CMS Joomla  Es un gestor de contenidos dinámico, es decir, que permite la creación de elementos interactivos para el usuario. *Joomla* es un *software* de código abierto, desarrollado en PHP, y en el que pueden trabajar de manera colaborativa varios desarrolladores, implementando cada vez más diferentes funcionalidades. Utiliza un panel de administración o escritorio con el que se puede crear, editar o eliminar contenido de una página *web*, añadir nuevas funciones a través de los *plugins* o programas complementarios, con los que se puede hacer cambios a los sitios *web*. | Icono  Descripción generada automáticamente |
| --- | --- |

Con *Joomla*, se pueden desarrollar:

* Páginas *web* utilizando las plantillas prediseñadas.
* Páginas *web* personales y blogs.
* Páginas *web* corporativas.
* Blogs de empresa.
* Foros y chats.
* *Web* de noticias o revistas *online*.
* Tiendas online.

Para su instalación, solo se necesita un plan de alojamiento o *hosting,* y, una vez adquirido, se procede a instalar *Joomla* con un autoinstalador. Los proveedores de *hosting* incluyen instaladores automáticos, para hacer el proceso en pocos minutos. Revise los pasos a continuación:



El proceso para instalar ***Joomla*** puede ser de modo automático, aunque también se puede realizar el proceso de instalación manualmente.

| * *WordPress*   *WordPress* es otro de los CMS más utilizados para la creación de sitios *web*. Su uso es gratuito y, al igual que Joomla, permite la creación y edición de contenidos de un sitio *web*. Posee una interfaz amigable, fácil de manejar y contiene una gran cantidad de *plugins* o complementos para extender su funcionalidad, por ejemplo, para crear una tienda online, enviar formularios de contacto, o galería de fotografías, entre otros. | Logotipo  Descripción generada automáticamente |
| --- | --- |

Al igual que *Joomla* y otros *CMS*, el procedimiento para la instalación de *WordPress* comienza con la adquisición de un servicio de hosting, se ingresa al *cPanel* y se continúa con los pasos tal como lo va solicitando la página de instalación.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Observe el siguiente Tutorial, que le indicará cómo descargar *Wordpress:*

| DI\_CF010\_2-2C\_TutorialDescargaWordpress\_VideoInstructivo |
| --- |

Una vez instalado WordPress, se puede hacer uso de todas sus herramientas, como, por ejemplo, escoger una plantilla de cualquier categoría de la lista para la creación del sitio *web*, incluyendo comercio electrónico, noticias, negocios, finanzas, educación, deportes, y fotografía, entre otras. Es importante definir el objetivo del sitio para que los resultados de utilizar estas herramientas sean realmente efectivos.

Al momento de definir la plantilla, se añade y se instalará el tema en *WordPress*; posteriormente, se va a visualizar la plantilla, su forma, color, tipografías, imágenes, menús, etc. *WordPress* posee herramientas para la edición de la plantilla escogida, donde se incorporan los textos, las imágenes (logo, colores, fotografías), tipografías, contactos, redes sociales y toda aquella información que se crea importante para el proyecto, como se observa en la siguiente figura.

Interfaz de usuario gráfica, Aplicación, Correo electrónico

Descripción generada automáticamente

Una de las herramientas importantes de los CMS son los ***Plugin***, o ***Plug-in***, cuyo significado en la lengua inglesa es “**inserción”,** y esaquella aplicación que añade una funcionalidad adicional o una nueva característica al *software*. En nuestro idioma, por lo tanto, puede nombrarse al *plugin* como un complemento que se añade a una aplicación existente para ampliarla.

Existen más de 50.000 *plugins* gratuitos en *WordPress,* para añadir todo tipo de funcionalidades, pero también se pueden adquirir plugins de pago, todo queda a elección del cliente.

Para instalar un *plugin* desde la interfaz de *WordPress*, se deben seguir los siguientes pasos:

1. En la barra de herramientas del administrador *WordPress*, seleccionar *plugins* – dar añadir.
2. En el buscador, escribir el nombre o palabra clave del *plugin*.
3. Dar clic al botón instalar. Una vez instalado, este mismo botón se activa – dar clic.

Después de activar el *plugin*, se deberá comprobar si se necesita realizar algún ajuste, o si el *plugin* viene ya configurado para ser utilizado sin necesidad de modificar algún parámetro.

Interfaz de usuario gráfica, Aplicación, Sitio web

Descripción generada automáticamente

| * Drupal   Drupal es un CMS o sistema de gestión de contenidos, libre, modular, multipropósito y muy configurable, que permite publicar artículos, imágenes y archivos, y que también ofrece la posibilidad de otros servicios añadidos. Incluye la creación y administración de foros, encuestas, *blogs*, administración de usuarios, permisos y otras características. Drupal es un CMS muy flexible para ser utilizado por diseñadores *web* y usuarios con conocimientos técnicos avanzados. | Logotipo  Descripción generada automáticamente |
| --- | --- |

Aunque es un CMS más segmentado y con una comunidad más pequeña que la de WordPress u otras tecnologías, Drupal permite crear proyectos ***web*** complejos, con funcionalidades muy específicas, que reciben grandes cantidades de tráfico y procesan muchos datos. Drupal está muy enfocado a usuarios avanzados y programadores. Lo suelen utilizar grandes empresas que quieren una tecnología muy potente.

Las principales características de Drupal



**Facilidad de uso:** para diseñadores o técnicos más avanzados, que pueden utilizar funcionalidades más genéricas.

**Potencial y funcionalidades:** no se dedica simplemente a la publicación de artículos, pues también permite el diseño y desarrollo de páginas ***web*** de *e-commerce*.

**Seguridad:** permite mantener la seguridad en la ***web*** frente a visitantes y posibles amenazas cibernéticas, realizando copias de seguridad y actualizaciones permanentes del *software*,

**Precio:** se tienen la versión gratuita y la versión de pago con más beneficios, funcionalidades y facilidades de uso.

A continuación, puede revisar los pasos indicados para lograr la instalación de Drupal desde el Cpanel:



1. **Estructuras de las Aplicaciones**

Para definir los diseños de los proyectos multimedia, se deben contemplar estructuras y arquitecturas que permitan una experiencia de uso funcional y accesible para los usuarios, de acuerdo con el dispositivo cliente que utilicen. Para lo anterior, se utilizan lenguajes de programación y maquetación como *HTML, CSS y JavaScript*. Revíselos a continuación:



Antes de comenzar a maquetar o a distribuir una página ***web,*** se debe tener claro cómo se desea diseñar (secciones, elementos, formas); se recomienda hacer un boceto previamente, un mapa de navegación pensado estratégicamente, para dar relevancia a la información que se pretende comunicar, teniendo en cuenta los logos, el menú de navegación y, en sí, el contenido en general.

A continuación, observe un ejemplo de una maquetación simple:

| Utilizar algunas de las mejores prácticas de diseño *web* para garantizar que los sitios *web* sean persuasivos. Esto, en el campo digital, significa atraer usuarios para finalmente convertirlos en clientes potenciales. | Diagrama  Descripción generada automáticamente con confianza media |
| --- | --- |

Entendiendo este concepto, se describirán algunas de las recomendaciones a la hora de iniciar un proyecto de diseño y desarrollo ***web***:



1. **Actividades didácticas (opcionales si son sugeridas)**

| Descripción de actividad didáctica | |
| --- | --- |
| Nombre de la Actividad |  |
| Objetivo de la actividad |  |
| Tipo de actividad sugerida |  |
| Archivo de la actividad  (Anexo donde se describe la actividad propuesta) |  |

1. **Material complementario**

| Tema | Referencia APA del Material | Tipo de material  (Video, capítulo de libro, artículo, otro) | Enlace del Recurso o  Archivo del documento o material |
| --- | --- | --- | --- |
| 1.5 Integración del Frontend y el Backend | Edteam*.* (2019). *¿Qué son las APIs y para qué sirven?* [Video]. YouTube. <https://www.youtube.com/watch?v=u2Ms34GE14U> | [Video] | <https://www.youtube.com/watch?v=u2Ms34GE14U> |
| 1.5 Integración del Frontend y el Backend | Reyes, I. (2020). *Arquitectura Aplicación Web - Frontend/Backend* [Video]. YouTube. <https://www.youtube.com/watch?v=CWJ84RA7LyI> | [Video] | <https://www.youtube.com/watch?v=CWJ84RA7LyI> |
| 2. Integración Multimedia | Pakiu Reviews. (2020). *Tipos de Resoluciones de Pantalla | SD - QHD - HD - FHD - UHD - 4K - 8K* [Video]. YouTube. <https://www.youtube.com/watch?v=mZ1oFNA_-XI> | [Video] | <https://www.youtube.com/watch?v=mZ1oFNA_-XI> |
| 2.2 Plataformas y Gestores de Contenido para la Integración Multimedia | Avanzapyme. (2008). *Gestores de contenidos* [Video]*.* YouTube. <https://www.youtube.com/watch?v=xUMQ4D96Hq8> | [Video] | <https://www.youtube.com/watch?v=xUMQ4D96Hq8> |
| * 1. Plataformas y Gestores de Contenido para la Integración Multimedia | UpEmpresa Tutoriales. (2014). *Tutorial Wordpress desde cero -- Video 2: ¿Qué es Wordpress?* [Video]. YouTube. <https://www.youtube.com/watch?v=u95vocRy1aY&list=PLmIEoKACpcDy5XGw_v7RFxRQ04cvXCF40&index=2> | [Video] | <https://www.youtube.com/watch?v=u95vocRy1aY&list=PLmIEoKACpcDy5XGw_v7RFxRQ04cvXCF40&index=2> |
| * 1. Plataformas y Gestores de Contenido para la Integración Multimedia | Gallardo, Y. (2016). *Qué es Drupal y cómo funciona* [Video]. YouTube. <https://www.youtube.com/watch?v=NNSsCahPsm4> | [Video] | <https://www.youtube.com/watch?v=NNSsCahPsm4> |

1. **Glosario**

| Término | Significado |
| --- | --- |
| API | Interfaz que tiene como función la comunicación de una aplicación con otra. |
| Back-end | Tareas realizadas del lado del servidor (Valdivia, 2016). |
| CMS | Un CMS (*Content Management System*), o Sistema de Gestión de Contenidos, es un sistema *online* que permite poner en marcha una página *web* de forma sencilla y rápida. Se trata de un *software* que ayuda a administrar contenidos dinámicos, por ejemplo, un blog, un *e-commerce* o cualquier tipo de página *web*. (Bello, 2021) |
| Cpanel | Cpanel es básicamente un panel de control para administrar un servicio de alojamiento. Permite administrar todos los aspectos de un servicio de alojamiento *web,* sin la necesidad de habilidades técnicas avanzadas. (Neubox, 2020). |
| Diseño modular | Como su propio nombre lo dice, una página *web* modular consiste en diagramar cada uno de sus elementos en módulos. Es posible apreciarlo cuando se entra a una página *web* con varios bloques que interactúan con el usuario, ya sea que ese bloque se haga más grande, se ilumine mientras se pone el cursor sobre ellos o, inclusive, se oscurezca un poco y dé la opción de guardar, tal como lo hace Pinterest.  Este diseño modular está pensado mucho en el comportamiento actual de las personas, ya que, como consumidores, se busca la manera más rápida de encontrar los servicios de una web, y normalmente se hace a través de los celulares. (Guerrero, 2018). |
| *Front-end* | Conjunto de tecnologías con las que interactúa el usuario (Valdivia, 2016). |
| *Leads* | Contacto o registro en función del momento de la campaña de *marketing*. El proceso de compra del propio usuario o el nivel de implicación que este ha demostrado con la marca. Es por ello que, incluso dentro del propio ámbito de *marketing* *online,* se suele crear cierta claridad al hablar de *leads*. Un *lead* es un usuario que ha entregado sus datos a una empresa y que, como consecuencia, pasa a ser un registro de su base de datos con el que la organización puede interactuar. Este registro puede realizarse de forma física, con papel, o de manera *online*, a través de un formulario. (Bels, 2018). |
| Pixel | El término surge por la combinación de dos palabras inglesas comunes, «*picture*» (imagen) y «*element*» (elemento). Un píxel es una unidad física, un punto cuadrado o rectangular que utiliza el lenguaje informático para definir la parte más pequeña de una pantalla de computador o un dispositivo móvil. |
| Plugin | Que también puede mencionarse como *plug-in*. Se trata de un concepto de la lengua inglesa que puede entenderse como “inserción” y que se emplea en el campo de la informática. Un *plugin* es aquella aplicación que, en un programa informático, añade una funcionalidad adicional o una nueva característica al *software*. En nuestro idioma, por lo tanto, puede nombrarse al *plugin* como un complemento. |
| SEO | SEO, o *Search Engine Optimization* (Optimización en Motores de Búsqueda), es un conjunto de técnicas y de herramientas que sirven para optimizar tanto las páginas *web* como las *landing pages*, para que los buscadores, como Google, puedan leerlas de forma más fácil y rápida. SEO no solo se utiliza para los buscadores, sino también para los usuarios. Es fundamental que cada página sea relevante con el contenido y que estas hablen de forma clara y concreta. (SN, 2012). |

1. **Referencias bibliográficas**

Aranda, J. (2014). *Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión*. IC Editorial. <https://www-ebooks7-24-com.bdigital.sena.edu.co/?il=8654>

Bel, O. (2020)**.** *¿Qué es un lead, qué tipos hay y para qué sirven?* Inboundcycle.<https://www.inboundcycle.com/blog-de-inbound-marketing/que-es-un-lead>

Bello, E. (2021). *Qué es un CMS, características y cuáles son los más populares*. IEBS. <https://www.iebschool.com/blog/que-es-un-cms-e-commerce/>

Casas, V. (2019). *Qué es Joomla y para qué sirve.* LucusHost. <https://www.lucushost.com/blog/que-es-joomla-y-para-que-sirve/>

DSI-EIE-FCEIA. (2015). *Interfaces para los programas de aplicación (APIs)*. FCEIA. <https://www.dsi.fceia.unr.edu.ar/images/downloads/InfoElectronica/APIs2015.pdf>

Guebs. (2014). *Como instalar y activar un plugin en WordPress* <https://ayuda.guebs.com/como-instalar-plugin-en-wordpress/>

Guerrero, E. (2018). *Diseño Web Modular.* Wataweb. <https://www.wataweb.com/blog/diseno-web-modular/>

Heilborn, C. (2012). *¿Qué es SEO y para qué funciona?* MD Marketing Digital. <https://www.mdmarketingdigital.com/blog/que-es-seo-y-para-que-funciona/>

Human Level. (s. f.). *¿Qué es un gestor de contenidos (CMS)?* <https://www.humanlevel.com/diccionario-marketing-online/gestor-de-contenidos-cms>

López, F. (2015). *Sistemas Distribuidos.* Universidad Autónoma Metropolitana.<http://dccd.cua.uam.mx/libros/archivos/03IXStream_sistemas_distribuidos.pdf>

López, M. (2016). *Programación web en el entorno del cliente*. Ediciones de la U. <https://www-ebooks7-24-com.bdigital.sena.edu.co/?il=7955>

Neubox. (2020).*¿Qué es cPanel y para qué sirve?* <https://neubox.com/tutoriales/base-de-conocimientos/que-es-cpanel-y-para-que-sirve/>

Recio, J. (2018). *HTML5, CSS3 y Jquery. Curso práctico*. Ediciones de la U. <https://www-ebooks7-24-com.bdigital.sena.edu.co/?il=8000>

Valdivia, J. (2016). *Modelo de Procesos para el Desarrollo del Front-End de Aplicaciones Web*. Ulima. <https://revistas.ulima.edu.pe/index.php/Interfases/article/view/1245/1205>

1. **Control del documento**

|  | Nombre | Cargo | Dependencia  *(Para el SENA indicar Regional y Centro de Formación)* | Fecha |
| --- | --- | --- | --- | --- |
| Autor(es) | Eduardo José Velasco Acevedo | Experto Temático | Regional Distrito Capital - Centro para la Industria de la Comunicación Gráfica | Julio de 2021 |
| Juan Manuel Reyes Ramírez | Experto Temático | Regional Distrito Capital - Centro para la Industria de la Comunicación Gráfica | Julio de 2021 |
| Carlos Andrés Suescun Lesmes | Experto Temático | Regional Distrito Capital - Centro para la Industria de la Comunicación Gráfica | Julio de 2021 |
| Olga Lucía Mogollón Carvajal | Experto Temático | Regional Distrito Capital - Centro para la Industria de la Comunicación Gráfica | Julio de 2021 |
| Caterine Bedoya Mejía | Diseñadora Instruccional | Regional Distrito Capital – Centro de Gestión Industrial | Julio de 2021 |
| Alix Cecilia Chinchilla Rueda | Evaluadora Instruccional | Regional Distrito Capital – Centro de Gestión Industrial | Agosto de 2021 |
| Rafael Neftalí Lizcano Reyes. | Asesor Pedagógico | Regional Santander - Centro Industrial del Diseño y la Manufactura | Agosto de 2021 |
| Darío González | Corrección de estilo | Regional Tolima – Centro Agropecuario La Granja | Octubre 2021 |

1. **Control de cambios**

**(Diligenciar únicamente si realiza ajustes a la Unidad Temática)**

|  | Nombre | Cargo | Dependencia | Fecha | Razón del cambio |
| --- | --- | --- | --- | --- | --- |
| Autor(es) |  |  |  |  |  |