



Construcción interfaz front-end y despliegue con React.js

Breve descripción:

Poder realizar una interfaz web con React.js, ofrece grandes beneficios en cuanto a funcionalidad y rapidez en el servicio web, es una herramienta tecnológica que ayuda y simplifica el escribir código y se integra muy bien con los servicios de JavaScript.

Diciembre 2023

Tabla de contenido

Introducción	1
1. “Front-end”	3
2. Herramientas de desarrollo web.....	6
3. Construcción web con React	14
4. Sitio web, “hosting” y dominio	23
Síntesis	26
Material complementario.....	27
Glosario	28
Referencias bibliográficas	29
Créditos	30

Introducción

Para realizar la interfaz de usuario de una aplicación web con React.JS es importante conocer su importancia, en cuanto a la definición y posicionamiento de los elementos de la interfaz del sitio web, así como la implementación de las interacciones entre todos estos elementos. A continuación se presenta un video que introduce a este componente formativo.

Video 1. Construcción interfaz “front-end” y despliegue con React.JS



[Enlace de reproducción del video](#)

Síntesis del video: Construcción interfaz “front-end” y despliegue con React.JS

Construcción de la interfaz “front-end” y despliegue con ReactJS.

React es la librería de JavaScript especializada en la creación de interfaces de usuario interactivas y efectivas. React utiliza una sintaxis muy parecida al HTML, lo que facilita su aprendizaje y aplicación. Además, puede diseñar vistas simples y eficaces para cada una de las aplicaciones. React es excelente en la actualización y renderizado de componentes, cuando los datos de la aplicación cambian React se encarga de todo el trabajo. React es una librería de código abierto que se enfoca en la visualización; esto permite agilizar la entrega de proyectos, ofrece una función de recarga continua y optimiza el rendimiento de la aplicación. ¡Bienvenidos!

1. “Front-end”

Es la parte del desarrollo web donde se muestra la parte frontal de una aplicación. Es el diseño, en cuanto a colores, letras, tamaños, fuentes, cajas de texto, tamaños, navegación, paginación y otros elementos más que reflejan la estructura gráfica o visual del sitio, haciéndolo más llamativo y agradable con el uso de imágenes, animaciones y efectos para su interacción.

A continuación, se muestran ejemplos de interfaces (“front-end”) de sitios web.

- **Diseño.** El “front-end” es la parte principal donde el usuario interactúa con la web. Un desarrollador “front-end”, se encarga de llevar a la práctica la estructura y diseño de la página web.
- **Desarrollo.** Cuando el cliente aprueba la propuesta de diseño, en cuanto a colores institucionales, nombre de la empresa y demás características propias de la organización, el desarrollador del diseño web, las lleva a la realidad a través del lenguaje de programación.

Para el desarrollo de la interfaz es necesario tener en cuenta las tareas básicas del desarrollo, como son la arquitectura, apariencia e interacción.

- **Arquitectura.** Es la columna vertebral del desarrollo web, muestra el resultado de cómo se verá finalmente el sitio web.

Su apariencia, se relaciona con el lenguaje CSS, que permite establecer la presentación del sitio, como colores y fuentes del contenido de textos o estructura en la web.

La interacción está basada en el lenguaje de programación con JS, JavaScript. Se basa en los eventos que se utilizan para transformar una página estática en una interfaz dinámica para la interacción con el usuario, navegador y servidor.

El “front-end” es la parte del “software” que interactúa con el usuario. Es desarrollado principalmente a través de los lenguajes HTML (“Hyper Text Markup Language”), CSS (“Cascading Style Sheets”) y JS (JavaScript), los cuales se explican a continuación:

- **HTML.** La maquetación web utiliza estructuras como HTML para organizar la página y desplegar su contenido. Este proceso permite situar elementos como ventanas, párrafos, listas con viñetas, imágenes y tablas en campos específicos. Además, se pueden definir características visuales como tipos de fuente, tamaños, colores y resaltados para hacer la web más atractiva para los usuarios. Todo esto se complementa con el lenguaje CSS, que enriquece la estructura de HTML añadiendo estilos y colorido. Juntos, HTML y CSS crean sitios web visualmente atractivos y funcionalmente organizados.
- **CSS.** Es el acrónimo de “Cascading Style Sheets” (hojas de estilo en cascada). Es un lenguaje que se basa en reglas donde el usuario define las características de los grupos de estilos de colores, fuentes, tamaños y demás particularidades que se quieren mostrar en la web.

El archivo .css, se utiliza llamando en las etiquetas html dentro del head.

Por ejemplo:

- `<html>`
- `<head>`

- `<link href="styles/style.css" rel="stylesheet" type="text/css">`
- `</head>`
- `<body>`
- `</body>`
- `</html>`

- **JavaScript.**

- Es el lenguaje de programación que se utiliza para añadir características interactivas al sitio web.
- Es un lenguaje de scripts dinámicos que admite construcción de objetos.
- Es el único lenguaje de programación que funciona de una manera ágil y eficiente en todos los navegadores.

Es importante indicar que las empresas dueñas de cada sitio web, con el fin de mantenerlo actualizado con las últimas tendencias, frecuentemente cambian el diseño de la interfaz gráfica (“front-end”) de su sitio, por lo que hoy se puede visualizar una información y si se visita el sitio después, tanto la información como su distribución puede haber cambiado.

2. Herramientas de desarrollo web

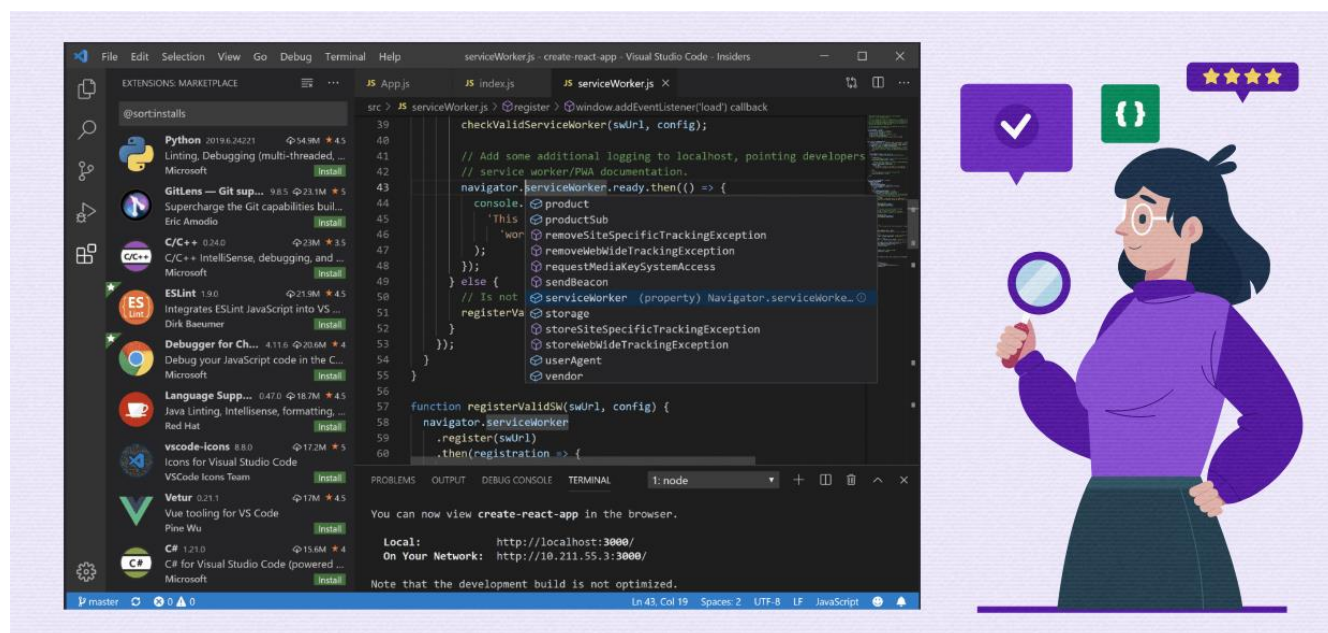
Antes de iniciar el diseño y construcción de la interfaz gráfica (“front-end”) de una aplicación, es importante tener las herramientas necesarias para su construcción.

A continuación, se encuentra el instructivo de instalación de aquellas que se utilizarán en este componente formativo para construir el “front-end” de una aplicación web y, posteriormente, realizar su configuración y despliegue en un servidor de almacenamiento web.

Instalación de Visual Studio Code

Es un editor de texto plano, desarrollado por Microsoft, gratuito y de código abierto para que los usuarios lo usen como herramienta de programación. La siguiente imagen muestra un pantallazo de ejemplo de la interfaz de esta herramienta.

Figura 1. Interfaz Visual Studio Code



Nota. Captura de <https://code.visualstudio.com/>

En los siguientes videos se muestra el paso a paso para los procesos iniciales de Visual Studio Code. Es así, como en este primer video se hace el proceso de descarga.

Video 2. Descarga Visual Studio Code



[Enlace de reproducción del video](#)

Síntesis del video: Descarga Visual Studio Code

El video trata sobre la descarga e instalación de Visual Studio Code, un editor de código esencial para escribir código y desarrollar funcionalidades web, interfaces y programación para construir “software” o aplicaciones web. Se explica cómo ingresar al sitio web de Visual Studio Code y cómo el sitio detecta automáticamente el sistema operativo del usuario, en este caso Windows, para sugerir la descarga correspondiente. El proceso de descarga se inicia con un clic en el botón 'Descargar', y se observa la descarga en la parte inferior de la pantalla. El video resalta la importancia de Visual Studio Code como herramienta clave para comenzar a trabajar en la interfaz web.

Y en este segundo video, ya se explica el proceso de instalación.

Video 3. Instalación Visual Studio Code



[Enlace de reproducción del video](#)

Síntesis del video: Instalación Visual Studio Code

El video describe el proceso de instalación de Visual Studio Code. Una vez descargado el ejecutable, se realiza doble clic para iniciar la instalación. Durante este proceso, se aceptan los términos de la licencia y se sigue una serie de pasos, donde se puede elegir la ruta de instalación, aunque esto es opcional. Se opta por crear un acceso directo en el escritorio y se procede a la instalación. Una vez finalizada, el programa se ejecuta automáticamente. Al abrirse Visual Studio Code, se muestra la página principal y, en ocasiones, una breve historia sobre el editor, la licencia y la versión, pero esto es principalmente texto informativo. El video concluye destacando

que la instalación de Visual Studio Code es el primer paso para comenzar a codificar y construir interfaces web.

Instalación de Node.JS

Se utiliza principalmente para desarrollar aplicaciones junto con servidores no bloqueantes, es rápido y efectivo para la visualización de la construcción de código en tiempo real.

Es una herramienta importante que proporciona un buen acompañamiento para el desarrollo ágil de las aplicaciones web (API).

A través del siguiente video se visualiza el paso para su descarga e instalación.

Video 4. Instalación Node.JS



[Enlace de reproducción del video](#)

Síntesis del video: Instalación Node.JS

El video explica cómo instalar Node.js, una herramienta crucial para el desarrollo de interfaces web. El proceso comienza con la búsqueda de Node.js en un navegador, accediendo a su página oficial (nodejs.org) y descargando la versión más reciente, que es recomendable para obtener las últimas funcionalidades. La página detecta automáticamente el sistema operativo del usuario, en este caso, Windows de 64 bits.

Una vez descargado el ejecutable, se procede a la instalación: se hace doble clic en el archivo descargado, se aceptan los términos de la licencia y se sigue el proceso de instalación, dejando las opciones por defecto. Durante la instalación, se puede optar por instalar herramientas adicionales, pero en este caso se elige no hacerlo. Tras aceptar un mensaje de alerta que permite a la aplicación realizar cambios en el dispositivo, la instalación de Node.js se completa. Al finalizar, Node.js queda instalado en el dispositivo, listo para su uso en el desarrollo web.

Instalación de React.JS

Es una herramienta para crear interfaces de usuario (“front-end”) interactivas de manera sencilla. Sirve para realizar diseños de interfaces web, actualizando y renderizando de manera eficiente todos los componentes que la integran como HTML, CSS y JavaScript.

A continuación, se podrá consultar el video que muestra el paso para verificar la versión instalada de Node.JS e instalar los paquetes (componentes) requeridos para crear aplicaciones React.JS.

Video 5. Instalación paquetes (componentes) React JS



[Enlace de reproducción del video](#)

Síntesis del video: Instalación paquetes (componentes) React JS

El video describe la instalación de React JS y la creación de un proyecto React. El proceso comienza con la búsqueda de la página oficial de React JS (reactjs.org) y el acceso a la sección de documentación. En esta sección, se encuentra información útil, tutoriales y ejemplos de proyectos React.

Se enfatiza la importancia de reutilizar proyectos y componentes existentes de React disponibles en la página. Luego, se guía al espectador para crear una nueva aplicación React a través de un enlace específico en el sitio. Antes de proceder con React, se verifica la instalación de Node.js, una herramienta necesaria para el desarrollo en React, utilizando la consola del sistema y comprobando la versión de Node.js instalada.

Finalmente, se regresa a la documentación de React para obtener comandos específicos necesarios para la creación de una nueva aplicación. Se destaca la importancia de seguir las instrucciones al pie de la letra para evitar errores durante la instalación. El proceso de instalación se inicia con el comando `npx create-react-app my-app`, y se espera a que finalice, destacando que esto implica una descarga desde internet.

Una vez instalados los paquetes (componentes) requeridos para crear aplicaciones React.JS se puede crear una aplicación React.JS, como se muestra en el siguiente video.

Video 6. Creación aplicación React.JS



[Enlace de reproducción del video](#)

Síntesis del video: Creación aplicación React.JS

El video se enfoca en la creación de una aplicación React JS después de haber instalado las herramientas necesarias. El proceso comienza con el acceso al escritorio del computador. Una vez en el escritorio, se sigue una instrucción específica de la página oficial de React JS, que implica escribir el comando `npx create-react-app`, seguido del nombre deseado para la aplicación, como "mi primera aplicación".

Tras ingresar el comando, se inicia la creación del proyecto, que incluye la descarga de las herramientas necesarias para el mismo. Este proceso toma algunos segundos. El video termina en este punto, indicando que se ha completado esta segunda parte del proceso de instalación y creación de una aplicación React JS.

3. Construcción web con React

React ayuda a crear interfaces de usuario de manera sencilla, diseñando vistas simples en cada estado de la aplicación, actualiza y renderiza de inmediato una vez se estén cambiando los datos del código fuente.

Es ideal para crear el “front-end”, pues permite múltiples opciones a nivel de diseño gracias a que es una herramienta integral, rápida y eficiente. Para ello, hay que tener en cuenta lo siguiente.

Maquetación HTML, CSS y JavaScript

- Para crear la maquetación con React se necesitan saberes previos, sobre HTML en sus estructuras, CSS sobre los diferentes estilos relacionados con Bootstrap y JavaScript en sus funcionalidades normales, desde las básicas hasta las más avanzadas.
- El archivo padre (principal) donde se escribe el código va directamente con funciones de JavaScript al mostrarlo en el navegador.
- Tiene herramientas importantes e integradas que hacen de React una herramienta muy eficiente; antes se necesitaba trabajar cada una de estas herramientas de manera independiente.

Pruebas y visualización en el servidor

Para lograr iniciar un proyecto React, este debe ir acompañado con la consola, ya sea del sistema operativo del equipo o la consola del editor de texto.

Después de crearse el proyecto y haberlo codificado integralmente con todas las herramientas necesarias, se debe ejecutar el servidor desde la consola, pero se debe tener en cuenta que previamente debe haberse instalado Node JS para que exista el servidor local, así se podrá visualizar la web en paralelo una vez se guarden los cambios en el código.

A continuación, se podrá consultar el video que muestra el paso a paso para cargar y ejecutar un proyecto web con React.

Video 7. Cargue y ejecución de proyecto React



[Enlace de reproducción del video](#)

Síntesis del video: Cargue y ejecución de proyecto React

El video detalla cómo cargar y ejecutar un proyecto de React JS. Después de completar la instalación de las herramientas necesarias, se accede al escritorio del ordenador desde la consola. Aquí, se navega hasta el proyecto React creado anteriormente, denominado "mi primera", utilizando comandos específicos mostrados en la documentación oficial de React.

Una vez localizado el proyecto en el escritorio, se abre con Visual Studio Code (el editor de texto utilizado) para visualizar y editar el proyecto. Se explica cómo arrastrar y soltar el proyecto en el editor para configurarlo en el entorno de trabajo. El proyecto viene con una estructura de carpetas predefinida que incluye componentes esenciales de React y archivos de configuración.

El proceso continúa con la explicación de cómo ejecutar el proyecto React desde la consola utilizando el comando `npm start`. Este comando inicia un servidor de desarrollo local, permitiendo ver la aplicación en un navegador web. Se destaca que React no requiere un servidor externo ya que Node.js facilita este proceso.

El video muestra cómo realizar cambios en la interfaz de la aplicación, utilizando archivos como `App.css` y `App.js` para editar estilos y componentes. Se enfatiza que React actualiza automáticamente el navegador con los cambios realizados en el código sin necesidad de recargar la página manualmente. Esta funcionalidad agiliza el desarrollo y permite ver cambios en tiempo real.

Finalmente, el video concluye destacando la facilidad y eficiencia de React para desarrollar interfaces web, y cómo se puede continuar integrando y desarrollando la interfaz a partir de este punto.

Después de cargar y ejecutar el proyecto, se puede iniciar la edición del código fuente de la aplicación. El siguiente video muestra un ejemplo de edición de código para cambiar el texto inicial de la aplicación y su color de fondo.

Video 8. Edición de código fuente



[Enlace de reproducción del video](#)

Síntesis del video: Edición de código fuente

El video se centra en la edición del código fuente en un proyecto de React JS. Muestra cómo acceder al proyecto llamado "mi primera" a través de la consola y del editor de código. Se enfatiza la importancia de ejecutar el servidor local para visualizar los cambios en la interfaz de la aplicación React en tiempo real.

Durante el video, se realizan modificaciones en el archivo `index.css` para cambiar los estilos de la aplicación. Por ejemplo, se borran ciertas líneas de código y se observa el impacto inmediato en la interfaz. Experimenta con diferentes estilos, incluyendo el cambio de color de fondo usando un degradado obtenido de un sitio web especializado, y demuestra cómo copiar y pegar este código en el proyecto.

También se exploran cambios en el archivo `App.js`, donde se agrega un párrafo como ejemplo de maquetación, y se muestra cómo afectan estos cambios a la estructura de la interfaz. Se destaca la necesidad de considerar el impacto de los cambios en el código en la interfaz final.

El video concluye mostrando cómo se pueden buscar y aplicar diferentes estilos de fondo para mejorar la interfaz del proyecto React. Este proceso incluye la selección de un degradado diferente, copiándolo y aplicándolo en el `index.css` para ver cómo cambia la apariencia de la aplicación.

Un aspecto importante dentro del desarrollo de aplicaciones web corresponde al diseño de la interfaz gráfica, en la cual se estructura, distribuyen y organizan sus elementos gráficos (textos, imágenes, botones, formularios, enlaces, títulos, entre

otros), lo cual corresponde a la maquetación de la aplicación. El siguiente video muestra algunos de los aspectos involucrados en este proceso.

Video 9. Maquetación parte 1



[Enlace de reproducción del video](#)

Síntesis del video: Maquetación parte 1

El video describe el proceso de maquetación de una interfaz utilizando React JS. Comienza con la creación de una nueva aplicación de React a través de la consola, nombrando la aplicación como "*básico app*". Enfatiza la importancia de tener abierta la documentación oficial de React (reactjs.org) para referencia y muestra cómo las librerías de React se integran con HTML, CSS y JavaScript.

Una vez creada la aplicación, se abre en un editor de código y se muestra cómo utilizar la terminal integrada del editor para ejecutar la aplicación usando el comando

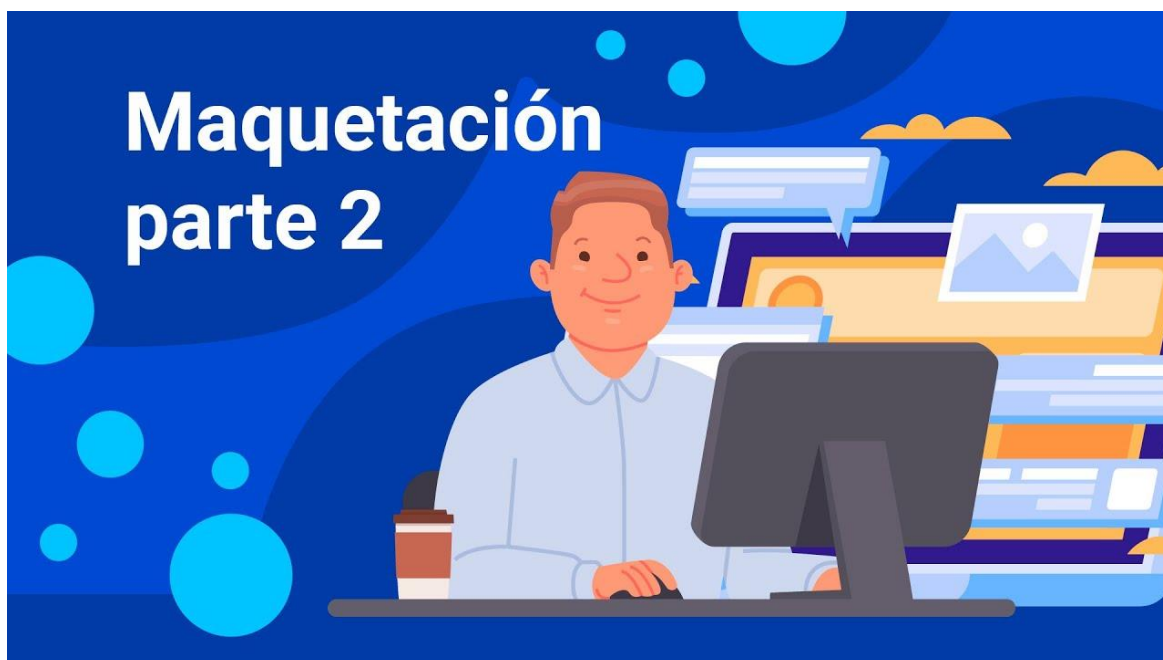
npm start, lo que permite visualizar los cambios en un servidor local. Se menciona la importancia de Node.js para funcionar como servidor local.

El video también cubre la edición de estilos en React. Se demuestra cómo hacer cambios en el archivo App.css, ajustando estilos como el tamaño y la rotación de un logo. Además, se muestra cómo importar estilos de Bootstrap para aplicarlos globalmente al proyecto, copiando y pegando un enlace de Bootstrap en el archivo index.html.

Se explica cómo maquetar la interfaz utilizando div y clases de Bootstrap para la disposición de los elementos, y cómo crear enlaces y estilos en la cabecera. Se destaca la importancia de una buena indentación para mantener el código organizado.

Finalmente, se muestra cómo utilizar gradientes para los estilos, copiando un código de un sitio web de gradientes y aplicándolo en el proyecto. El video enfatiza la importancia de experimentar con diferentes estilos y cómo los pequeños cambios pueden afectar la apariencia general de la aplicación.

Video 10. Maquetación parte 2



[Enlace de reproducción del video](#)

Síntesis del video: Maquetación parte 2

El video se centra en la maquetación avanzada de una interfaz utilizando React JS y Bootstrap. El proceso comienza con la creación de varios div (divisores) para estructurar la interfaz. Se enfatiza la importancia de una buena indentación para mantener el código organizado y legible. Cada div se asigna una clase específica, como container o “card-header”, para definir su estilo y función en la maquetación.

Se muestra cómo crear tarjetas con encabezados y cuerpos, utilizando etiquetas como h3 y p para los títulos y el contenido, respectivamente. Durante el proceso, se corrigen errores en tiempo real, lo que destaca la importancia de revisar y ajustar constantemente el código.

También demuestra cómo controlar el ancho de los divisores usando el sistema de columnas de Bootstrap, explicando que el ancho total disponible es de 12 columnas y cómo se pueden asignar diferentes tamaños a los divisores.

En la segunda parte del video, se enfoca en la personalización de la interfaz. Se muestra cómo cambiar los estilos de fondo utilizando gradientes de color, copiando códigos de un sitio web especializado y aplicándolos en el proyecto. Se resalta la flexibilidad en la elección de colores y estilos, dependiendo de las necesidades del proyecto o las preferencias institucionales.

El video concluye destacando la versatilidad y la cantidad de opciones disponibles en la maquetación y el diseño de interfaces utilizando React JS y Bootstrap, mostrando cómo se pueden combinar diferentes estilos y estructuras para crear una interfaz atractiva y funcional.

4. Sitio web, “hosting” y dominio

Para cargar un proyecto React.js en un servidor web, se deben tener en cuenta varios aspectos, a saber:

- Su tamaño en código.
- Lenguaje programado.
- Pago anual por el permiso del uso del servidor a nivel mundial.
- Lugar donde se pagarán los derechos para ejecutar la aplicación web.
- Nombre o dominio de la web.
- Si utiliza base de datos.

A continuación, se explica el uso de los anteriores conceptos.

- **Sitio web.** Es un conjunto de páginas web que son accesibles a través de un dispositivo con conexión a internet, convenientemente enlazadas que cumplen una finalidad específica. Está identificado por una dirección URL única que, usualmente, es la página de inicio u “homepage” del sitio web.
- **“Hosting”.** Es provisto por un gran número de empresas a nivel mundial, se debe contratar de acuerdo con las necesidades específicas de la aplicación a implementar (espacio, base de datos, seguridad, entre otros). Algunas compañías de gran tamaño, en ocasiones optan por implementar el servicio de “hosting” directamente en sus servidores, por lo que no requieren contratar un proveedor externo.

- **Dominio.** Corresponde a un nombre único en internet, identifica el nombre del sitio web como tal, por ejemplo: sena.edu.co es el dominio del SENA en Colombia.

Se adquiere a través de múltiples compañías a nivel mundial, tiene un valor que generalmente se paga y renueva anualmente, no obstante, puede hacerse cada 2 o 3 años.

El dominio termina en algunas letras que identifican su tipo, país, entre otros:

- .com: compañías.
- .edu: instituciones de educación.
- .co: para sitios de Colombia.
- .gov o .gob: entidades de gobierno, entre otros.

Para finalizar, en el siguiente video se explica con mayor detalle esos aspectos.

Video 11. “Hosting” web



[Enlace de reproducción del video](#)

Síntesis del video: “Hosting” web

El video aborda el tema de “hosting” web y dominios, aspectos cruciales para la publicación de una aplicación o sitio web en internet. Se explica que tanto el “hosting” como los dominios son servicios pagados. El “hosting” se refiere a los servidores que alojan una aplicación web, permitiendo que sea accesible en internet. Se mencionan varios proveedores de “hosting” como GoDaddy, HostGator, HostPapa, DreamHost, BlueHost, e Ionos, destacando la diversidad de opciones disponibles.

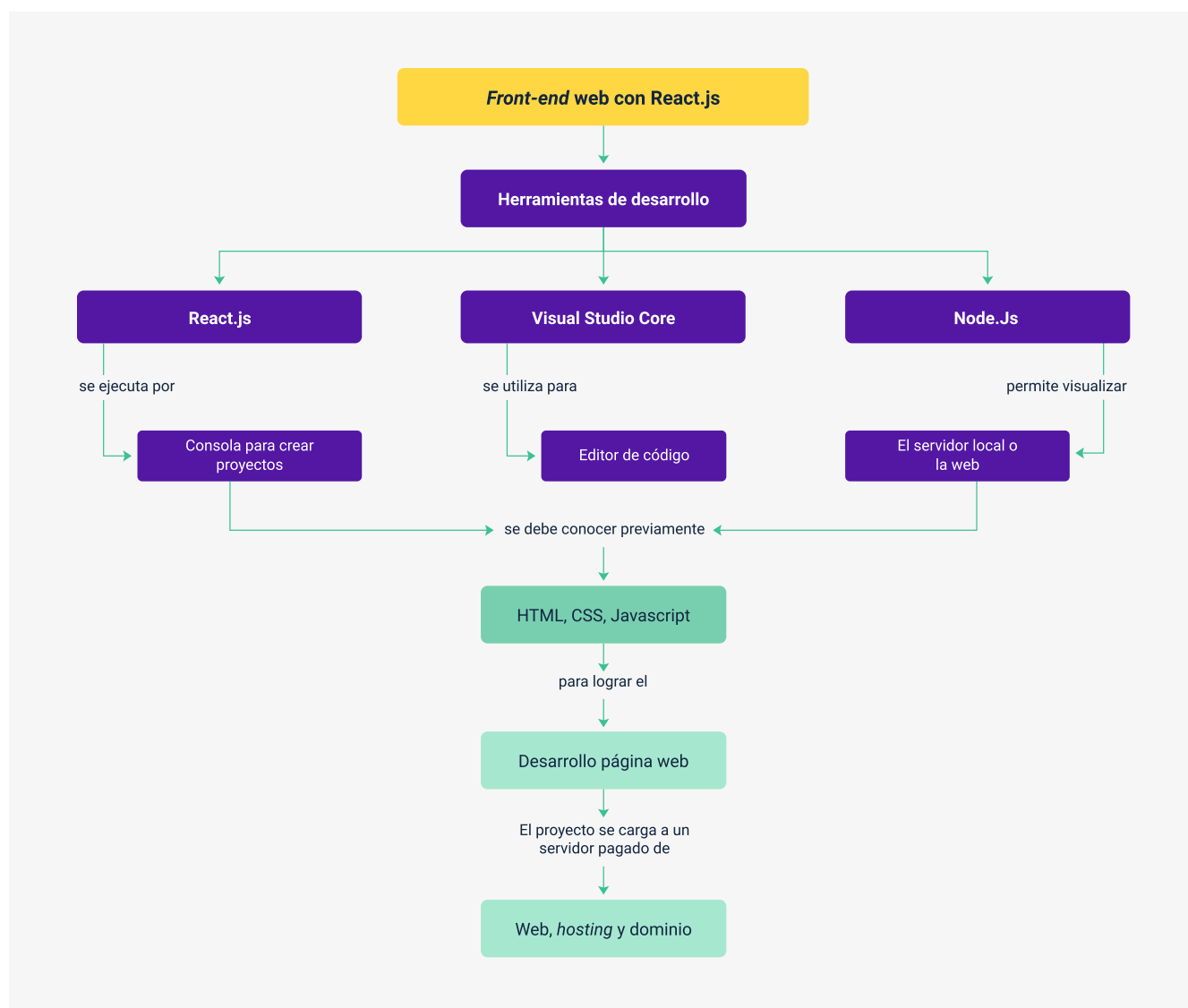
El proceso implica elegir un proveedor de “hosting”, realizar el pago anual (cuyo costo varía según el tamaño y el peso de la aplicación), y subir la carpeta de la aplicación web al “hosting”. Se enfatiza la importancia de conocer el tamaño de la aplicación para elegir un plan de “hosting” adecuado, ya que los precios pueden variar según la capacidad de almacenamiento requerida.

Además del “hosting”, se habla sobre la compra de un dominio, que es el nombre único que identifica un sitio web en internet. Al comprar un “hosting”, normalmente se ofrece un dominio y, a menudo, un correo electrónico corporativo. Se destaca que el dominio asegura que nadie más pueda usar ese nombre específico en la web.

Finalmente, se resalta que todo esto (“hosting”, dominio y correo corporativo) se integra para formar una página web completa y funcional. Se concluye mencionando que tanto los sitios web como los dominios siempre implican un costo adicional.

Síntesis

A continuación se presenta un mapa conceptual que sintetiza el componente formativo:



Material complementario

Tema	Referencia	Tipo de material	Enlace del recurso
1. “Front-end”	Platzi. (2021). Qué es Frontend y Backend. [Video]. YouTube.	Video	https://www.youtube.com/watch?v=s95S1_vP-dQ
2. Herramientas de desarrollo web	Visual Studio Code. (2022). Code Editing. Redefined.	Sitio web	https://code.visualstudio.com/
2. Herramientas de desarrollo web	Node.js. (2022). Descarga Node.js.	Sitio web	https://nodejs.org/en
2. Herramientas de desarrollo web	React. (2022). A JavaScript library for building user interfaces.	Sitio web	https://reactjs.org/
4. Sitio web, “Hosting” y Dominio	MarketHax Ecommerce México. (2020). Qué es hosting y dominio de una página web para principiantes. [Video]. YouTube.	Video	https://www.youtube.com/watch?v=VzXbegQtXVs

Glosario

CSS: hoja de estilos en cascada o lenguaje que ayuda a definir la apariencia del documento escrito.

Dominio: nombre único de la página web, con el que nadie más a nivel mundial podrá utilizar el mismo nombre.

“Hosting”: servidor donde se permite el cargue de la aplicación web desarrollada para un fin u objetivo de desarrollo de “software”.

Maquetación: distribución de la página web relacionada con los estilos, como colores, fuentes, tamaños, letras y demás elementos.

Node.JS: entorno de ejecución para JavaScript que ayuda a permanecer el código ligero y eficiente en la superficie del uso intensivo de los datos en tiempo real.

React.JS: librería de JavaScript de código abierto que se enfoca a la visualización.

Referencias bibliográficas

Node.js. (2022). Node.js. <https://nodejs.org/es>

Pickupimage.com. (2022). Free High Resolution Pictures and High Quality Free images. <https://pickupimage.com>

React. (2022). A JavaScript library for building user interfaces. <https://reactjs.org>

Skitterphoto. (2022). Un lugar para encontrar, mostrar y compartir imágenes de dominio público. <https://skitterphoto.com>

StockSnap. (2022). Beautiful Free Stock Photos. <https://stocksnap.io>

Unsplash. (2022). 500+ Scrum Pictures | Download Free Images on Unsplash. <https://unsplash.com/s/photos/scrum>

Visual Studio Code. (2022). Code Editing. Redefined. <https://code.visualstudio.com>

Créditos

Nombre	Cargo	Centro de Formación y Regional
Claudia Patricia Aristizábal	Líder del Ecosistema	Dirección General
Rafael Neftalí Lizcano Reyes	Responsable de Línea de Producción	Centro Industrial del Diseño y la Manufactura - Regional Santander
Luis Eyder Ortiz	Experto Temático	Centro de Teleinformática y Producción Industrial - Regional Cauca
Gustavo Santis Mancipe	Diseñador Instruccional	Centro de Gestión Industrial - Regional Distrito Capital
Andrés Felipe Velandia Espitia	Asesor Metodológico	Centro de Diseño y Metrología - Regional Distrito Capital
José Gabriel Ortiz Abella	Corrector de estilo	Centro de Diseño y Metrología - Regional Distrito Capital
Paola Alexandra Moya	Diseñador Instruccional	Centro Industrial del Diseño y la Manufactura - Regional Santander
Yerson Fabian Zarate Saavedra	Diseñador de Contenidos Digitales	Centro Industrial del Diseño y la Manufactura - Regional Santander
Edward Leonardo Pico Cabra	Desarrollador Fullstack	Centro Industrial del Diseño y la Manufactura - Regional Santander
Daniela Muñoz Bedoya	Animador y Productor Multimedia	Centro Industrial del Diseño y la Manufactura - Regional Santander
Daniela Muñoz Bedoya	Locución	Centro Industrial del Diseño y la Manufactura - Regional Santander
Emilsen Alfonso Bautista	Actividad Didáctica	Centro Industrial del Diseño y la Manufactura - Regional Santander
Zuleidy María Ruiz Torres	Validador de Recursos Educativos Digitales	Centro Industrial del Diseño y la Manufactura - Regional Santander

Nombre	Cargo	Centro de Formación y Regional
Luis Gabriel Urueta Alvarez	Validador de Recursos Educativos Digitales	Centro Industrial del Diseño y la Manufactura - Regional Santander
Daniel Ricardo Mutis Gómez	Evaluador para contenidos inclusivos y accesibles	Centro Industrial del Diseño y la Manufactura - Regional Santander