Plan de instalación Carcesorios (web)

Presenta:

Jhon Jairo Araujo satiaca

Luis Carlos Diaz Carreño

Carlos fabian Cubillos zarate

Miguel Angel Gomez Escobar

Jefferson Estid Moreno Venegas

Instructor(a):

Isaura María Suárez Novoa

Servicio Nacional de Aprendizaje (SENA)

Análisis y desarrollo de software

Ficha: 2558108

Bogotá D.C

2024

Tabla de contenido

[Plan de instalación Carcesorios (web) 3](#_Toc162882994)

[Introducción 3](#_Toc162882995)

[Objetivos 3](#_Toc162882996)

[Requisitos del Sistema 3](#_Toc162882997)

[Tecnologías usadas 4](#_Toc162882998)

[Front-end 4](#_Toc162882999)

[Entorno de desarrollo Front-end 5](#_Toc162883000)

[- *Editor de código*: 5](#_Toc162883001)

[- *Gestión de dependencias*: 5](#_Toc162883002)

[- *Control de versiones*: 5](#_Toc162883003)

[Back-end 5](#_Toc162883004)

[Entorno de desarrollo Back-end 7](#_Toc162883005)

[- Editor de código: 7](#_Toc162883006)

[- Gestion de dependencias: 7](#_Toc162883007)

[- Herramientas de desarrollo: 7](#_Toc162883008)

[- Control de versiones: 7](#_Toc162883009)

# Plan de instalación Carcesorios (web)

Este documento es el plan de instalación del aplicativo web Carcesorios donde se encuentra la introducción, objetivos, requisitos del sistema

# Introducción

El proyecto Carcesorios consiste en un aplicativo web para la venta de autopartes con una interfaz amigable desarrollada en React y una API REST desarrollada en Express. La aplicación permitirá a sus usuarios buscar, visualizar, comprar de manera segura, agregar, editar y eliminar productos, datos etc.

# Objetivos

* Proporcionar una plataforma intuitiva y fácil de usar para la compra de autopartes.
* Ofrecer una experiencia de usuario fluida que incluya funciones de búsqueda y visualización de productos.
* Ofrecer una experiencia dinámica de usuario administrador para agregar, editar y eliminar productos y datos de usuarios clientes.
* Integrar un sistema de carrito de compras que permita a los usuarios agregar y eliminar elementos antes de confirmar su compra.
* Implementar un proceso de pago seguro y confiable para completar las transacciones.

# Requisitos del Sistema

* Sistema operativo: Windows 10 o superior
* Memoria RAM: 6GB en adelante
* Procesador: Intel Core I3 de sexta generación en adelante o Ryzen 3 3200 en adelante
* Espacio en disco duro: 700mb
* Navegador web: Google Chrome o Microsoft Edge

# Tecnologías usadas

## Front-end

* **Node**: es un entorno de ejecución multiplataforma, de código abierto, para la capa del servidor basado en el lenguaje de programación JavaScript, asíncrono, con E/S de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google.

Como descargar:

En el siguiente enlace está el instalador: <https://nodejs.org/en/download>

* **React**: Es una biblioteca JavaScript de código abierto diseñada para crear interfaces de usuario con el objetivo de facilitar el desarrollo de aplicaciones en una sola página.

Como descargar:

1. Crear proyecto en react: npx create-react-app react-carcesorios
2. Arrancar: npm start

**Descargar dependencias:**

1. ***react-dom***: es una biblioteca necesaria para trabajar con React en aplicaciones web y se utiliza para interactuar con el DOM (Document Object Model) del navegador. Para descargar usar este comando: npm install react-dom
2. ***react-hook-form***: es una biblioteca de React que facilita la creación y validación de formularios en aplicaciones web. Esta biblioteca se basa en los "Hooks" de React, lo que significa que aprovecha las características de React Hooks para ofrecer una forma intuitiva y eficiente de manejar formularios. Para descargar usar este comando: npm install react-hook-form
3. ***react-router-dom***: es una biblioteca de enrutamiento para aplicaciones web desarrolladas con React. Permite la navegación entre diferentes componentes de React mientras el usuario interactúa con la aplicación, manteniendo la URL sincronizada con el estado de la aplicación. Para descargar usar este comando: npm install react-router-dom
4. ***Axios***: es un cliente HTTP simple basado en promesas para el navegador y node. js. Axios provee una librería fácil de usar en un paquete pequeño con una interfaz muy extensible. Para descargar usar: npm install axios

## Entorno de desarrollo Front-end

* *Editor de código*: Visual Studio Code
* *Gestión de dependencias*: npm es el sistema de gestión de paquetes por defecto para Node.js, un entorno de ejecución para JavaScript, bajo Artistic License 2.0.
* *Control de versiones*: Git.

## Back-end

1. Crear una carpeta para el proyecto en este caso API CARCESORIOS.
2. Por medio del editor de código abrir esa carpeta y abrir la consola.
3. Ejecutar el siguiente comando para inicializar un proyecto de Node.js y crear un archivo package.json: npm init -y
4. Ahora instalar Express y dependencias necesarias con: npm install express body-parser cors

Dependencias:

* Cors: Es una política de seguridad implementada por los navegadores web que restringe las solicitudes HTTP realizadas entre diferentes dominios. En resumen, CORS es un mecanismo que permite a los servidores indicar a los navegadores si permiten que una solicitud de origen cruzado tenga acceso a un recurso específico. Para descargar usar: npm install cors. Como descargar: npm install cors
* Dotenv: es una biblioteca de Node.js que facilita la carga de variables de entorno desde archivos .env en tu aplicación. Las variables de entorno son valores que se utilizan para configurar y personalizar el comportamiento de una aplicación en función del entorno en el que se ejecuta, como el entorno de desarrollo, el entorno de producción, etc. Como descargar: npm install dotenv
* Express: es un marco de aplicación web de backend para Node.js, diseñado para facilitar la creación de aplicaciones web y APIs (Interfaz de Programación de Aplicaciones) robustas y escalables. Es uno de los marcos más populares y ampliamente utilizados en el ecosistema de Node.js debido a su simplicidad, flexibilidad y potencia. Como descargar: npm install express
* promise-mysql: es un paquete de Node.js que proporciona una envoltura de promesas para el cliente de MySQL, permitiéndote interactuar con bases de datos MySQL utilizando promesas en lugar de callbacks. Esto simplifica la gestión de conexiones y consultas a la base de datos, haciéndolo más compatible con el estilo de programación asincrónica y moderna de JavaScript. Como descargar: npm install promise-mysql
* @babel/preset-env: es un conjunto de configuraciones predefinidas para Babel, que te permite compilar código JavaScript moderno a una versión compatible con las versiones específicas de los navegadores que deseas admitir. Como descargar: npm install @babel/preset-env --save-dev

## Entorno de desarrollo Back-end

* Editor de código: Visual Studio Code
* Gestion de dependencias: npm es el sistema de gestión de paquetes por defecto para Node.js, un entorno de ejecución para JavaScript, bajo Artistic License 2.0.
* Herramientas de desarrollo: Postman (para probar la API), MySQL (para almacenar datos estructurados y relacionales, como usuarios, pedidos, productos, etc.).
* Control de versiones: Git.