



UNIVERSIDAD TÉCNICA DE BABAHOYO

FACULTAD DE ADMINISTRACIÓN, FINANZAS E INFORMÁTICA

**PROCESO DE TITULACIÓN
DICIEMBRE 2021 – ABRIL 2022**

**EXAMEN COMPLEXIVO DE GRADO O DE FIN DE CARRERA
PRUEBA PRÁCTICA**

INGENIERÍA EN SISTEMAS

TEMA:

**ESTUDIO COMPARATIVO DE LAS TECNOLOGÍAS PARA EL
DESARROLLO DEL BACK-END “NODEJS” Y “PHP”.**

EGRESADO(A):

KERLY LUCIA VALLEJO CANO

TUTOR:

ING. ANA DEL ROCÍO FERNANDEZ TORRES

BABAHOYO – ECUADOR 2022

INTRODUCCIÓN

La presente documentación tiene como título Estudio comparativo de las tecnologías para el desarrollo del back-end “NodeJS” y “PHP”, como objetivo principal veremos en estas dos tecnologías de back-end: su historia, su instalación, gestor de paquetes, compatibilidad con otros frameworks, conexiones a base de datos, rendimiento, sus beneficios, ventajas y desventajas, se implementara pequeños fragmentos de código para su comparación se cuestionarán preguntas sobre ¿Qué tecnología deberías elegir en el desarrollo de un proyecto?.

El desarrollo de aplicaciones web del lado del Backend va en constante desarrollo dándose a conocer nuevas tecnológicas para la implementación del mismo siendo una necesidad el dicho conocimiento siendo unas que otras de fácil acceso y aprendizaje.

Pero la actualizaciones o evolución de estas tecnologías crean una problemática al elegir que tecnología de desarrollo web debemos utilizar, el desarrollo cuenta para las tecnologías que tienen más demanda en el mercado por lo tanto le hablare de NodeJS es un entorno de ejecución de JavaScript, su entorno de ejecución es en tiempo real, tiene más demanda en el mercado por utilizar su lenguaje de programación JavaScript, lo transformaron de algo que solo podía ejecutarse en el navegador en algo que se puede ejecutar en los ordenadores como una aplicación independiente. Mientras que PHP solo se ejecuta del lado del servidor, pero no en tiempo real también tiene su demanda, en la actualidad su compañía de desarrolladores no le darán más actualizaciones.

En esta investigación se implementó la metodología comparativa y documental la misma que ayudo a saber las diferencias de cada una de estas tecnológicas para el desarrollo del Backend, recaudando información precisa de las características, ventajas,

desventajas y su lenguaje de programación, sobre el rendimiento de ambas y su tiempo de carga, se ejecutó la técnica de observación con su instrumento guía de observación.

Implementando la línea de investigación presente en este proyecto está encaminada en redes y tecnologías de software y hardware, por lo tanto, se realiza este estudio comparativo con el objetivo de dar a conocer las dos tecnologías más ágiles en el desarrollo de aplicaciones del lado del backend de una manera fácil y segura.

DESARROLLO

Las tecnologías de backend es la parte importante para el desarrollo web el mismo que se encarga de que toda la lógica del aplicativo web funcione. Este se trata de un sin número de funciones que pasan por la web y no la vemos porque se comunican del lado del servidor.

Los desarrolladores encargados de realizar el backend pasan días desarrollando la lógica del sitio web para el mismo funcione, de su utilidad o preferencia utilizan varios códigos de lenguajes de programación como NodeJS, PHP, Java o .Net, con todo esto la información puede ser transmitida de manera segura y el desempeño de la aplicación no complique la experiencia del usuario.

En la actualidad el desarrollo de aplicativos webs del lado del servidor o backend nos muestra muchos sobre su rendimiento de la página, por los múltiples procesos que se ejecutan comunicándose con el servidor y el aplicativo del frontend.

Según Ravoof define *“Los sitios web y las aplicaciones son tan comunes hoy en día que es fácil darlos por sentado. Pero detrás de cada desarrollo de un sitio web o de*

una nueva función genial hay una combinación de código y tecnología que lo hace funcionar". (Ravoof, 2021)

El desarrollo del backend es de mucha importancia es de comprender que el frontend sin el backend no se puede vivir, para el enlace de estas dos tecnologías se necesita crear proyectos a la medida, los desarrolladores del backend son más racionales, lógicos, a priori y menos creativos.

Tenemos algunas funciones que se gestionan en el desarrollo del backend:

- Solucionar problemas ya arreglar errores.
- Pruebas sobre el control de calidad.
- Actualizar la documentación de la arquitectura y procesos.
- La simplificación de procesos.
- Acciones lógicas.
- Las conexiones a la base de datos.
- Uso de librerías

Para convertirse en desarrollador de la tecnología backend, se debe tener las siguientes habilidades como:

- Servidor
- Base de datos
- Lenguajes de programación backend
- Api

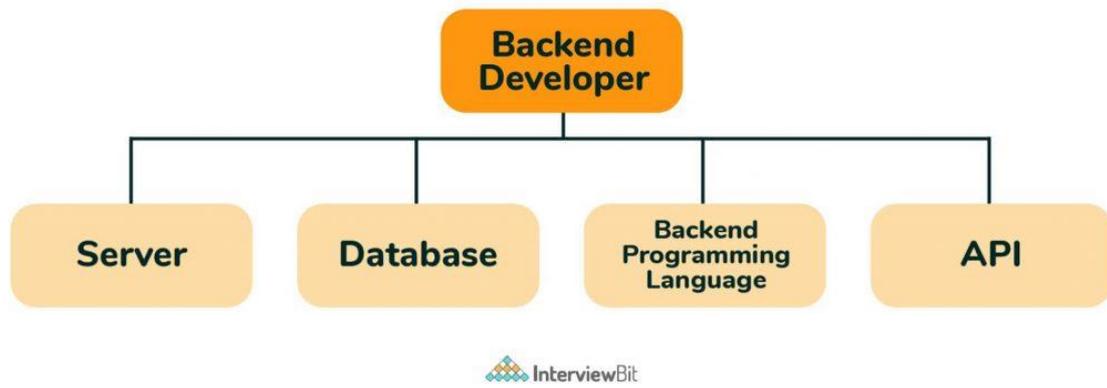


Figura 1: Habilidades para el desarrollo backend

Fuente: (InterviewBit, 2021)

Laura Fitzgibbons define como *“El back end se refiere a las partes de una aplicación informática o del código de un programa que permiten su funcionamiento y a las que no puede acceder un usuario. La mayoría de los datos y la sintaxis de funcionamiento se almacenan y se accede a ellos en el back end de un sistema informático. Normalmente, el código se compone de uno o varios lenguajes de programación. El back end también se denomina capa de acceso a los datos del software o del hardware e incluye cualquier funcionalidad a la que sea necesario acceder y navegar por medios digitales”* (Fitzgibbons, 2019)

El objetivo de este estudio comparativo está enfocado cuales de estas dos tecnologías como NodeJS y PHP es de mayor accesibilidad conociendo sus ventajas, desventajas y sintaxis para el fácil y rápido desarrollo del backend para las aplicaciones webs, dándolas a conocer mediante este estudio a futuros desarrolladores Junior.

También realizamos ciertas preguntas de reflexión para poder determinar la problemática de la investigación de las tecnologías en el desarrollo del Backend como NodeJS y PHP:

¿Cuáles son las ventajas y diferencias entre NodeJS y PHP en el entorno del desarrollo del backend?

¿Cuál es la comparación meticolosa sobre el estudio de estas dos tecnologías en el desarrollo del backend?

Por causa a las limitaciones del tiempo los pequeños ejemplos de prueba como desarrollar un CRUD (Consultar, Crear, Actualizar y Eliminar) en estas dos tecnologías para dar a conocer sus sintaxis, gestor de paquetes, conexión a la base de datos y su rendimiento.

En la actualidad hay dos tipos de lenguajes de programación para backend estos son los orientados a objetos y los funcional:

Orientado a objetos (OOP): su arquitectura es basada en los conceptos de clases y valga la redundancia a objetos, son llamados como clases para estructurar el programa en fragmentos de códigos simples y reutilizables, los mismo que se utilizan para la creación de instancias. Dentro de los lenguajes de programación orientada a objetos para el desarrollo del backend tenemos Java, JavaScript, Python, PHP, Ruby, .NET.

Funcionales: este se basa en el paradigma de la programación en el que se vinculan los estilos de funciones matemáticas, se basa en la creación de funciones puras para el desarrollo del software. Dentro de los lenguajes de programación funcional para el desarrollo del backend tenemos LISP, ML, Haskell, F#, Clojure, SQL.

Los factores fundamentales para el desarrollo del backend tenemos las habilidades y los lenguajes de programación, está de más decir que para desarrollar se necesita un conjunto de habilidades diferentes estas habilidades corresponden a la maquetación del Frontend como HTML y CSS, aunque no se lo use para el Frontend se debe tener estos conocimientos básicos para poder implementar el Backend.

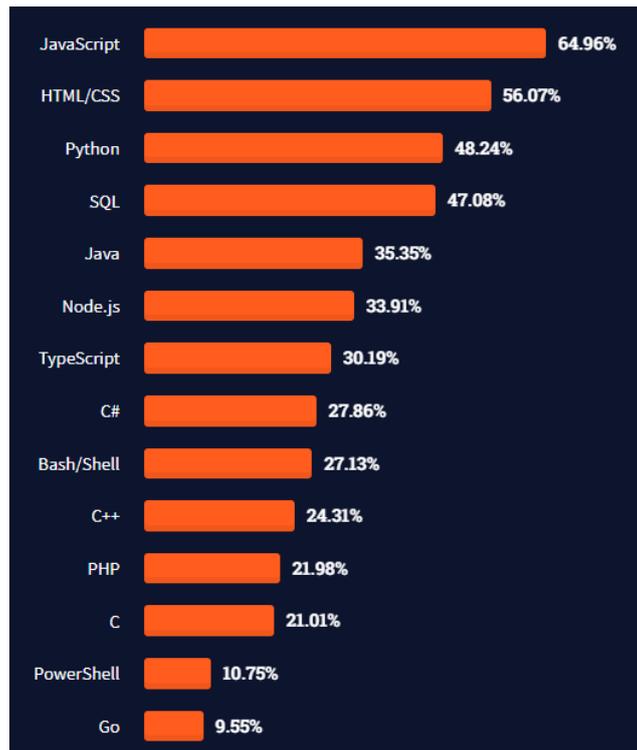


Figura 2: Lenguajes de programación utilizados por desarrolladores

Fuente: (Stack Overflow, 2022)

Podemos observar en la *Figura 2*, mediante la encuesta del Stack Overflow los lenguajes de programación más populares en la actualidad en primer lugar a JavaScript seguido de Python, Java y NodeJS y tenemos a PHP en onceavo lugar claramente su reputación es baja comparado con NodeJS.

NodeJS

Desde el lanzamiento inicial de NodeJS en el año 2009, ha revolucionado el mundo de JavaScript. Empleado sobre las bases del motor V8 de Chrome, su entorno es de código abierto implementado para hacer posibles aplicaciones en el lenguaje JavaScript del lado del servidor. Este permitirá que una vez ya desarrollada se pueda ejecutar en diferentes sistemas, podemos decir que NodeJS nos brinda un conjunto de bibliotecas con módulos en JavaScript para poder simplificar el desarrollo de los aplicativos webs. (Geek, 2020)

Dentro de las últimas décadas se ha visualizado el crecimiento en los sitios webs y como es de esperar el mismo NodeJS está creciendo. Con su popularidad del tiempo en ejecución ha superado los mil millones de descargas desde el 2018 y según (W3techs, W3techs.com, 2022) la tecnología NodeJS ha sido utilizada por el 1,2% de todos los sitios webs del mundo, calculando que son más de 20 millones de sitios webs.

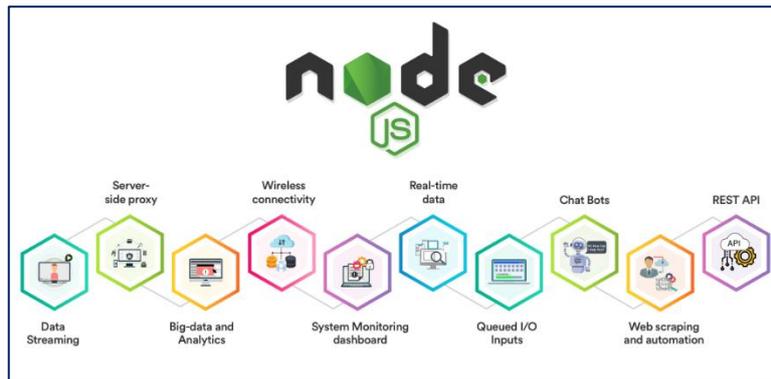


Figura 3: Por que usar NodeJS

Fuente: (Jimenez, 2020)

¿Cuál es la funcionalidad de NodeJS?

Su idea principal es el uso no-bloqueante y el event-driven I/O, permaneciendo ligero y eficiente consumiendo pocos recursos de uso intensivo de datos en tiempo real en el ámbito de la CPU, RAM y Disco.

Dado por entender que NodeJS utiliza menos hilos o subprocesos, utiliza menos recursos del computador, lo que facilita la ejecución rápida de las tareas.

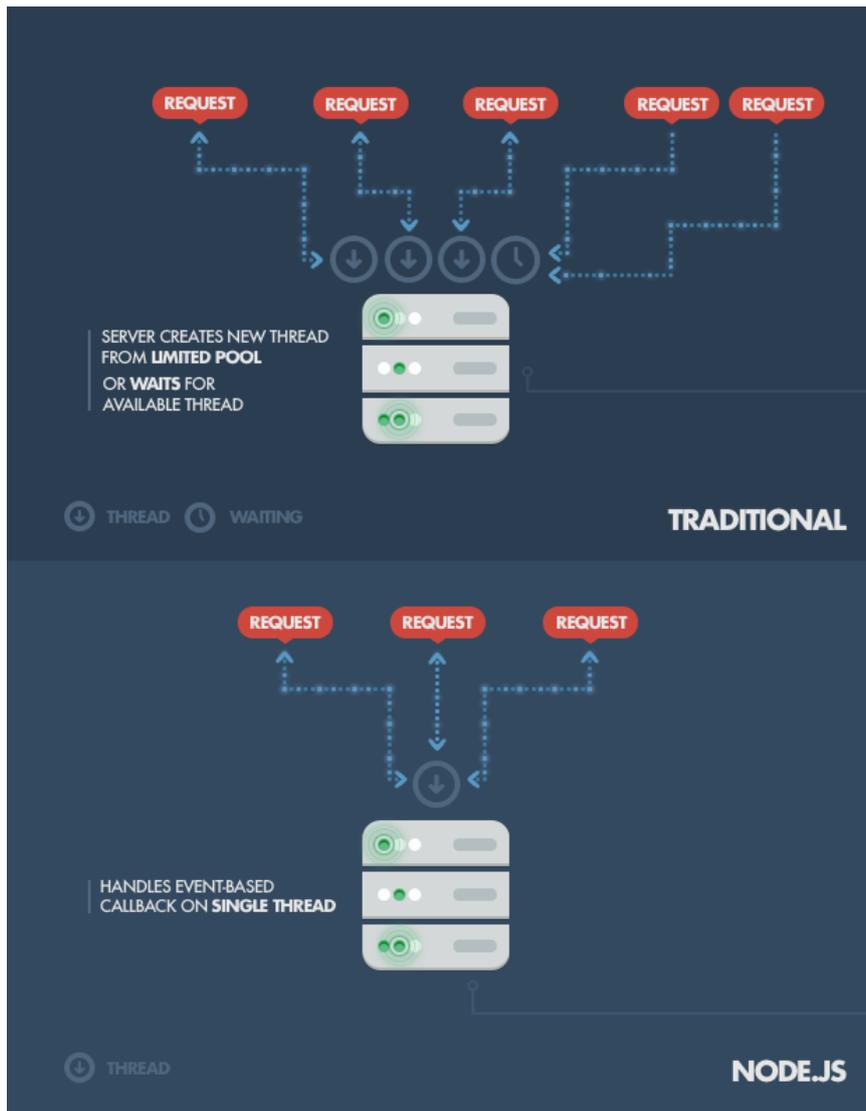


Figura 4: Servidor tradicional y servidor en NodeJS con sus procesos

Fuente: (Capan, 2013)

En la *Figura 4* podemos visualizar dos tipos de servidores primeramente un servidor creando un nuevo subproceso a partir de grupo limitado y mientras ingresen más solicitudes las pone en espera hasta que un subproceso esté disponible, sin embargo, en la segunda ilustración lo cual es un servidor de NodeJS podemos visualizar el manejo en la devolución de llamadas de las solicitudes con un solo subproceso.

Características de NodeJS

Debido al gran crecimiento que tiene NodeJS se ha obtenido la siguiente lista:

- ✓ **Easy.** – es de fácil aprendizaje. Tiene variedad de tutoriales con una gran comunidad y empezar es de mucha facilidad.
- ✓ **Escalable.** - facilita una gran escalabilidad para las aplicaciones y al ser de un solo hilo, es capaz de manejar grandes conexiones simultaneas con alto rendimiento.
- ✓ **Velocidad.** - al ejecutarse los hilos sin bloqueo hace que NodeJS sea más rápido y eficiente.
- ✓ **Paquetes.** - en la actualidad existen un conjunto de paquetes de código abierto que pueden simplificar el trabajo en NodeJS. Teniendo más de un millón de paquetes en el ecosistema NPM.
- ✓ **Respaldo.** - este framework para backend está escrito en C y C++, por esto tenemos rapidez y compatibilidad con redes.
- ✓ **Multiplataforma.** - nos da gran ventaja por permitir crear sitios web SaaS, aplicaciones móviles y de escritorio, todo utilizando solo NodeJS.
- ✓ **Maintainable.** - es de mucha facilidad para los desarrolladores ya que en el frontend y el backend se pueden gestionar con JavaScript como un único lenguaje.

El Node Package Manager “NPM”

Al empezar hablar de Node.js, una de las cosas que tenemos que tener en cuenta que definitivamente no se debe omitir es integrarlo al soporte de la gestión de paquetes que utiliza las herramientas NPM, ya que este gestor nos permite la dependencia de poder gestionar nuestros proyectos, ya que viene integrado por

defecto con cada una de la instalación de Node.js. la intención de los módulos NPM es similar a la de Ruby Gemas. Es un compuesto de componentes que se pueden usar que están disponibles públicamente por medio de una sencilla instalación por medio de repositorio online.

DONDE SE PUEDE UTILIZAR NODE.JS

Aplicaciones web del lado del servidor

En Node.js junto a Express.js estos también pueden ser utilizados para la creación de aplicaciones de web clásica por el medio de ejecución de código en el lado del servidor. Por lo tanto, nuestro lo sea posible, en este paradigma en petición y respuesta de Node.js se lo dirigiera alrededor de HTML, no es lo más típico en los casos de uso. Tenemos algunos contenidos para estar a favor y en contra de este método de enfoque. Aquí tenemos algunos que se consideran:

Ventaja:

Si su aplicación no tiene ningún cálculo intensivo de CPU, puede compilar JavaScript de arriba a abajo, incluso si almacena objetos usando JSON como MongoDB DB, o en el nivel de la base de datos. Esto facilita enormemente el desarrollo (incluido el reclutamiento).

El rastreador recibe la respuesta HTML completa, que es más compatible con SEO que una sola página o una aplicación Websockets que se ejecuta en Node.js.

Desventajas

Las CPU con uso intensivo de cómputo evitan que Node.js responda, por lo que una plataforma con subprocesos es un mejor enfoque. Alternativamente, puede probar el cálculo extendido.

Usar Node.js con bases de datos relacionales sigue siendo una molestia (lea a continuación para obtener más detalles). Si intenta realizar operaciones relacionales, hágase un favor y elija cualquier otro entorno, como Rails, Django o ASP.NET MVC.

Como podemos instalar NodeJS

En Windows lo podemos hacer desde la página oficial de NodeJS <https://nodejs.org/en/>.

Desde macOS se usa el Bash en Mac y se ingresa la siguiente url

```
curl "https://nodejs.org/dist/latest/node-${VERSION:-$(wget -qO-  
https://nodejs.org/dist/latest/ | sed -nE 's|.*>node-  
(.*)\.pkg.*|\1|p')}.pkg" > "$HOME/Downloads/node-latest.pkg" &&  
sudo installer -store -pkg "$HOME/Downloads/node-latest.pkg" -  
target "/"
```

también con el uso de Homebrew: `brew install node`.

Desde Linux con los comandos *Sudo apt update*, *Sudo apt install nodejs*.

Se empezará desarrollar un programa básico «Hola Mundo NodeJS», donde se implementará un servidor en NodeJS devolviendo el mismo mensaje de salida.

PHP

Diseñado por Rasmus Lerdorf y lanzado en 1995, PHP o Hypertext Preprocessor es un lenguaje de secuencias de comandos del lado del servidor para el desarrollo web y también se utiliza como lenguaje de programación de propósito general. Los lenguajes de implementación incluyen Perl, C, C++, Java.

En el ámbito de PHP puro la utilización de herramientas asíncronas no ha cambiado significativamente desde el año pasado. Pero con la aceptación de los Fibers para PHP 8.1, nos da entender que el 84% no usa ninguna biblioteca, el 7% usa ReactPHP, 7% de Swoole, 2% de Amp y 2% otros, obteniendo un total tenemos un 100% en el uso de PHP puro. Su popularidad de PHP sigue siendo la más alta en Francia: el 43% de los desarrolladores franceses lo utiliza como lenguaje principal.

Es uno de los lenguajes de scripting de más popularidad de la web hasta la actualidad. Según (W3techs, W3techs.com, 2022) nos comenta que PHP es usado por el 78% de todos los sitios webs que utilizan un lenguaje de programación del lado del servidor. Con esto podemos decir que PHP no está muerto y juega un papel importante al respecto del ecosistema de WordPress, ya que su CMS está basado en PHP.

Los desarrolladores junior son más propensos a aprender Java y Python, mientras que los especialistas senior tienden a aprender Go, TypeScript o Rust.

No sorprende que los lenguajes más estudiados sean también los más populares: Python, Java, JavaScript y TypeScript. Los dos últimos parecen ser cada vez más populares como lenguajes en los que formarse.

Características Generales del PHP

- ✓ El lenguaje es fácil de aprender porque se han simplificado varias especificaciones durante su desarrollo, como la definición de variables primitivas, que se pueden ver cuando se usan arreglos de PHP.
- ✓ Este lenguaje está orientado al desarrollo de aplicaciones web dinámicas que pueden acceder a los datos almacenados en una base de datos.
- ✓ Se puede conectar a casi todos los motores de bases de datos que se utilizan hoy en día, y su conectividad es excelente con PostgreSQL y MySQL.
- ✓ El código fuente escrito en este lenguaje no es visible para los clientes y navegadores web porque el servidor es el encargado de ejecutar el código y enviar los resultados del HTML. en el navegador. Esto permite una programación PHP confiable y segura.
- ✓ Contiene muchos documentos en el sitio web oficial, entre los cuales los más importantes son las funciones del sistema descritas y explicadas en un archivo de ayuda.
- ✓ Tiene potencial de extensibilidad usando módulos, también conocidos como extensiones.

Ventajas sobre el lenguaje PHP

- ✓ Su lenguaje de programación es libre y abierto.
- ✓ Su curva de aprendizaje es muy baja.
- ✓ Su entorno de desarrollo es de rápido y fácil configuración.
- ✓ Es de fácil instalación
- ✓ De fácil acceso y se integran muy bien con la base de datos
- ✓ Posee una comunidad extensa
- ✓ Su lenguaje es de mayor demanda

- ✓ Es multiplataforma
- ✓ Orientada al desarrollo de aplicativos webs dinámicas con los accesos a la base de datos
- ✓ Su código se ejecuta del lado del servidor y los resultados en el navegador es HTML.

Desventajas sobre el lenguaje PHP

- ✓ Su gran inconveniente está en que el código fuente no puede ser ocultado de una manera eficiente. La falta de concentración puede dificultar la lectura del código.
- ✓ Si se requiere tener la seguridad de nuestro código se debe crear nuestro propio servidor.
- ✓ Si no se protege el código se tendrá brechas de seguridad a su largo tiempo tendremos problemas.
- ✓ Sus variables no son tipadas, esto provoca que sea difícil que los IDEs ofrezcan asistencias para hacer el tipado del código.

Como instalar PHP

Desde Windows podemos hacer su instalación manual descargando los paquetes todo en uno solo desde el programa Xampp y WampServer son dos de las distribuciones más populares que hay en Windows dentro de la comunidad de PHP.

Comparación rápida de los Frameworks NodeJS y PHP

Muchos desarrolladores se están dando cuenta que conforme se vaya innovando las tecnologías se regirán por implementar la de fácil aprendizaje y su seguridad en este caso NodeJS tiene sus ventajas a favor de PHP. La demanda en estas tecnologías podemos

decir en su tiempo PHP tubo su curva alta, pero ahora en el desarrollo del Backend es favorable NodeJS por motivos de seguridad y actualización hacia los usuarios.

Se ha hecho unas pequeñas comparaciones en el entorno de la Base de datos y su diferencia en el rendimiento de los mismo:

Bases de datos

Habitualmente PHP ha sido asociado con sistemas de bases de datos relacionales (RDBMS) como es MySQL, MS SQL, PostgreSQL y similares. Entre ellos, MySQL la base de datos más reconocida para crear sitios web PHP. Como lo habías hablado anteriormente, es una parte de la popular pila de código abierto LAMP (Linux, Apache, MySQL y PHP). También es posible el uso de base de datos NoSQL como MongoDB con PHP.

En Node.js es más apropiado para el trabajo con base de datos NoSQL o cómo MongoDB y CouchDB. Pero también se puede trabajar con bases de datos SQL, pero muchos de los desarrolladores eligen y prefieren bases de datos NoSQL con Node.js. Con el soporte de JSON incorporado, Node.js trabaja muy bien con las bases de datos NoSQL. MongoDB es una de las opciones más comunes: representa a M en MEAN.

Diferencia entre Node.js vs rendimiento PHP

Node.js es una poderosa plataforma o marco basado en JavaScript construido sobre el motor Google Chrome V8 JavaScript. Se utiliza para desarrollar aplicaciones web intensivas de E/S, como sitios web de transmisión de video, aplicaciones de una sola página y otras aplicaciones web. PHP significa preprocesador de hipertexto. PHP es uno de los lenguajes de secuencias de comandos del lado del servidor que está integrado en una página web o se puede ejecutar como una secuencia de comandos (al igual que Perl) desde la línea de comandos.

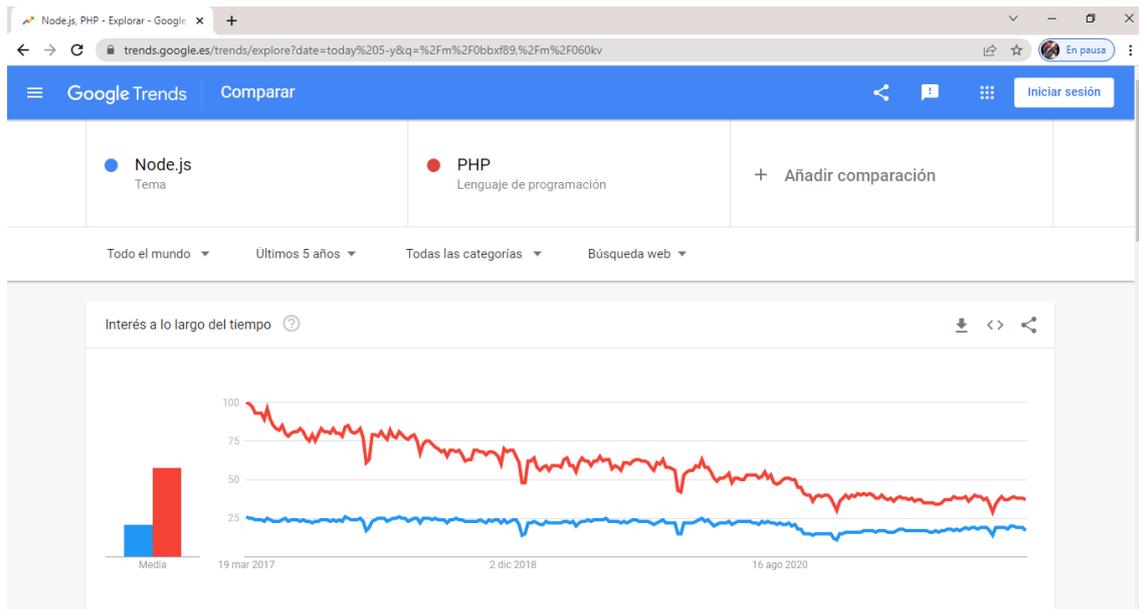


Figura 5: Comparación de NodeJS y PHP (Google Trends)

Fuente: <https://trends.google.es/trends/explore?date=today%205-y&q=%2Fm%2F0bbxf89,%2Fm%2F060kv>

Observando la **Figura 5** se ha realizado una comparación mediante la página de Google Trends sobre estas dos tecnologías la comparación cuenta con un cálculo de búsqueda de los últimos 5 años hasta la actualidad, nos da a visualizar como estas dos herramientas en el desarrollo del backend están en competencia observando que PHP está en una gran curva de descenso y NodeJS manteniéndose muy constante pero no será por mucho tiempo ya que es de mucha facilidad para los desarrolladores facilitando su trabajo en tiempos de ejecución. Dentro de estas búsquedas tenemos su seguridad, rendimiento, su sintaxis, cual es de fácil aprendizaje y que tecnología es de fácil aprendizaje.

Base de comparación entre Node.js vs PHP Performance	Node.js	PHP
Acerca de	Creado por Ryan Dahl y fue lanzado originalmente en 27 de mayo 2009, Node.js es una plataforma o marco que está basado en	Fue creada por el programador Rasmus Lerdorf en el año 1994, PHP es uno de los programas de uso general

	JavaScript para aplicaciones basadas en web. Está escrito en C++ y JavaScript.	que se adapta al desarrollo web, está basado en C, C++ y Java
Conceptos	<p>Los conceptos importantes de Node.js son:</p> <ul style="list-style-type: none"> ✓ Módulos ✓ Consola ✓ Racimo ✓ Complementos ✓ Buffer ✓ Devoluciones de llamada ✓ Cripto ✓ Manejo de errores ✓ Red ✓ Global ✓ Dominio ✓ DNS ✓ Transmisión ✓ Depurador 	<p>Antes de usar PHP, los programadores necesitan saber de lo siguiente:</p> <ul style="list-style-type: none"> ✓ Concepto de OOPS ✓ Gestión de base de datos ✓ Sesiones en PHP ✓ Enrutamiento ✓ Solicitud y respuesta ✓ Controladores ✓ Puntos de vista
Áreas de enfoque de desarrollo	<ol style="list-style-type: none"> 1. Back-End (incluidas las API) 2. Completa pila 3. Interfaz 4. Ops / DevOps 5. Aplicaciones de escritorio 6. Móvil 7. IOT 8. Seguridad 	<ol style="list-style-type: none"> 1. Aplicaciones CRM personalizadas 2. Solicitud de aplicaciones web de entrada 3. Integración inmobiliaria con RETS 4. Aplicaciones de concesionario automotriz 5. Soluciones relacionadas con la salud 6. Desarrollo de aplicaciones móviles. 7. Integración de sitio web ERP 8. Calendarios PHP personalizados

		9. Servidores PDF encriptados
Características	<ol style="list-style-type: none"> 1. Asíncrono y controlado por eventos. 2. Muy rápido. 3. Rosca simple pero altamente escalable. 4. Sin almacenamiento en búfer. 5. Con licencia 	<ol style="list-style-type: none"> 1. Web específica y de código abierto. 2. Integrado en archivos HTML de estatinas. 3. Acceso rápido a una base de datos.
Comparación de rendimiento	<ul style="list-style-type: none"> • Correcciones rápidas del lado del servidor <p>En NodeJS los eventos de JavaScript el bucle y se puede atender de múltiples eventos. Eso hace que el Node.js sea una de buenas opciones para el uso de aplicaciones asíncronas, que están basadas en datos que trabaja en el flujo de vinculadores pesados de E/S. NodeJS nos garantiza un buen rendimiento en el tiempo de ejecución.</p> <ul style="list-style-type: none"> • One ILanguage Across the Stack <p>El uso de Node.js garantiza que se utilice un lenguaje de programación en toda la pila de desarrollo de aplicaciones. El uso del mismo lenguaje para las funciones de front-end y back-end garantiza la capacidad de mantenimiento de la aplicación.</p>	<ul style="list-style-type: none"> • Amplia base de código <p>PHP tiene una base de código muy amplia y rica que nos permite el desarrollo rápido de aplicaciones web.</p> <ul style="list-style-type: none"> • Portátil <p>PHP es muy fácil de transportar, es decir, está diseñado para ejecutarse en cualquier servidor como Nginx, Apache y también en cualquier plataforma como Windows, Linux. El código debe escribirse solo una vez.</p> <ul style="list-style-type: none"> • Diseñado para la web <p>PHP a diferencia a otros lenguajes de programación, este diseñado para la creación de aplicaciones web.</p>

	<ul style="list-style-type: none"> flexibilidad <p>Node.js tiene pocas reglas y restricciones. Esto nos facilita la creatividad en el desarrollo de aplicaciones web.</p>	
Última versión	v14.16.0	v. 8.1
Los usuarios	<ol style="list-style-type: none"> PayPal LinkedIn Yahoo! Mozilla Netflix Uber Groupon Ve papi Ebay 	<ol style="list-style-type: none"> Facebook Wikipedia Baidu Yahoo! Tumblr Flickr WordPress MailChimp Biblia LAD Birchbox SeatGeek Fotolia Flipkart Digg iStockPhoto

Tabla 1: Comparación rápida de NodeJS y PHP

Fuente: Kerly Lucia Vallejo Cano

CONCLUSIONES

Dado por concluido el análisis del estudio comparativos de las tecnologías NodeJS y PHP en el desarrollo del Backend, diciendo que la implementación de NodeJS del lado del servidor es muy factible, siempre y cuando los proyectos realizados en una de estas dos tecnologías estén en constante mejoramiento y se implementen módulos o funciones con regularidad también se ha demostrado el potencial de cada uno, pero uno del otro tiene su ventaja.

Node.JS es un popular Framework de Backend para el desarrollo de aplicaciones a gran escala que necesitan soportar múltiples solicitudes dentro del servidor. Las E/S (entradas y salidas) no bloqueante desde un solo subproceso o hilo lo convierte en una opción favorable para las aplicaciones en tiempo real y de múltiples flujos de datos.

Luego de estar realizadas las pruebas de comparaciones en las herramientas llamadas Google Trends la misma que nos permite visualizar las búsquedas que los desarrolladores realizan en el día a día, pudiendo observar que NodeJS hasta la actualidad no se da a conocer mucho por el simple hecho que no conocen su metodología, su sintaxis, la función de este Framework de desarrollo del Backend, en pocas palabras su escala es baja pero mientras toda aplicación necesite un Backend robusto no dudaran en usarla.

Desde otra grafica tenemos a PHP que desde sus inicios en los años 1995 creando el primer código en este lenguaje se revoluciono como desarrollo de aplicaciones webs, el mismo tiene su sintaxis y documentación demasiado estructurada pudiendo buscar información desde cualquier página, su grafica como observamos va en descenso y en la actualidad ha llegado solo hasta la versión 8.1.2 y un mensaje de desilusión publicado por sus propios desarrolladores diciendo que PHP no recibirá más actualizaciones dando entender que desarrollar aplicaciones dinámicas no es una opción.

Los desarrollares actualmente estas implementado el Backend en NodeJS por lo robusto que es y este sirve para enlazar diversas tecnologías de Frontend como Angular, Flutter, React.

Dado por concluido este estudio comparativo se puede recomendar el uso de la tecnología NodeJS por sus grandes novedades y amplio ecosistema en el desarrollo multiplataforma, por otro lado, PHP se puede utilizar peor a futuro puede tener varias complicaciones debido a que sus desarrolladores no le darán más soporte en la actualidad se comenta que PHP esta muerto.

Bibliografía

Capan, T. (13 de Agosto de 2013). *Toptal Engineering Blog*. Obtenido de ¿Por qué demonios usaría Node.js? Un tutorial caso por caso: <https://www.toptal.com/nodejs/por-que-demonios-usaria-node-js-un-tutorial-caso-por-caso>

Fitzgibbons, L. (16 de Mayo de 2019). *Front end and back end*. Obtenido de WhatIs.com: <https://whatis.techtarget.com/definition/front-end>

Geek, E. (08 de Abril de 2020). *IfgeekthenNTTdata*. Obtenido de Qué es NodeJS y primeros pasos: <https://ifgeekthen.nttdata.com/es/que-es-node-js-y-primeros-pasos>

InterviewBit. (21 de Junio de 2021). *Backend developer skills you must have*. Obtenido de InterviewBit: <https://www.interviewbit.com/blog/backend-developer-skills/>

Jimenez, A. C. (16 de Noviembre de 2020). *Portal de noticias de tecnología, Realidad Virtual, Aumentada y Mixta, Videojuegos*. Obtenido de ¿Por qué usar node js?: <https://niixer.com/index.php/2020/11/16/node-js-en-el-desarrollo-web-su-importancia-y-uso-actualmente/>

Ravoof, S. (12 de Diciembre de 2021). *Cómo Convertirse en un Desarrollador Backend: Habilidades Importantes que Hay que Aprender*. Obtenido de Kinsta: <https://kinsta.com/es/blog/desarrollador-de-backend/>

Stack Overflow. (12 de Marzo de 2022). Obtenido de Stack overflow developer survey 2021: <https://insights.stackoverflow.com/survey/2021>

W3techs. (19 de marzo de 2022). *W3techs.com*. Obtenido de Usage statistics and market share of Node.js, march 2022: <https://w3techs.com/technologies/details/ws-nodejs>

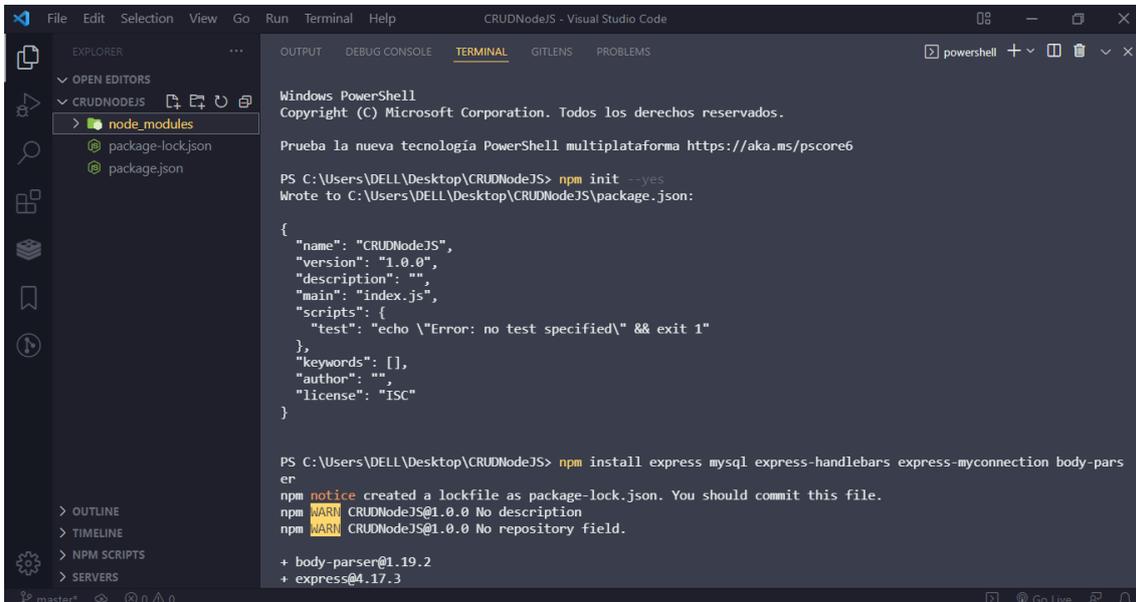
W3techs. (19 de Marzo de 2022). *W3techs.com*. Obtenido de Usage statistics and market share of PHP for websites, march 2022: <https://w3techs.com/technologies/details/pl-php>

ANEXOS

Guía de Observación

No.	Aspectos evaluados	Si	No	Observaciones
1	Determinar la seguridad de ambas tecnologías en el desarrollo del backend.	X		Se explicó sobre las vulnerabilidades.
2	Las dos tecnologías son de código libre?	X		La licencia es de fácil acceso para cualquier usuario.
3	Se puedo describir lo más importantes de estas dos tecnologías	X		Conociendo los más relevante de estas dos tecnologías.
4	El propósito de este estudio se realizó con la finalidad de dar a conocer a futuros desarrolladores.	X		Se desarrolló para tengan a ala manos sus novedades y que puedan elegir su respectiva metodología para un futuro proyecto.
5	Se calculó el rendimiento de estas dos tecnologías	X		Se descubrió lo que consumen tanto en peticiones hacia el servidor

Anexo 2



```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Users\DELL\Desktop\CRUDNodeJS> npm init --yes
Wrote to C:\Users\DELL\Desktop\CRUDNodeJS\package.json:

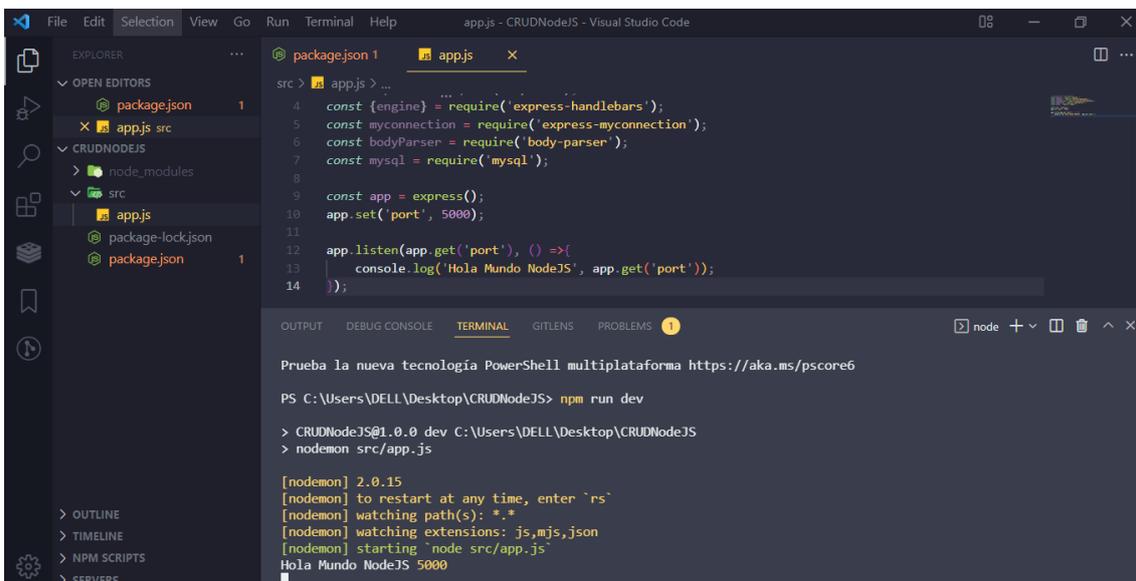
{
  "name": "CRUDNodeJS",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}

PS C:\Users\DELL\Desktop\CRUDNodeJS> npm install express mysql express-handlebars express-myconnection body-parser
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN CRUDNodeJS@1.0.0 No description
npm WARN CRUDNodeJS@1.0.0 No repository field.

+ body-parser@1.19.2
+ express@4.17.3
```

Figura 6: Creación de la aplicación en NodeJS

Fuente: Kerly Lucia Vallejo Cano



```
src > app.js > ...
4 const {engine} = require('express-handlebars');
5 const myconnection = require('express-myconnection');
6 const bodyParser = require('body-parser');
7 const mysql = require('mysql');
8
9 const app = express();
10 app.set('port', 5000);
11
12 app.listen(app.get('port'), () =>{
13   console.log('Hola Mundo NodeJS', app.get('port'));
14 });

PS C:\Users\DELL\Desktop\CRUDNodeJS> npm run dev
> CRUDNodeJS@1.0.0 dev C:\Users\DELL\Desktop\CRUDNodeJS
> nodemon src/app.js

[nodemon] 2.0.15
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node src/app.js`
Hola Mundo NodeJS 5000
```

Figura 7: Ejecución del servidor NodeJS

Fuente: Kerly Lucia Vallejo Cano

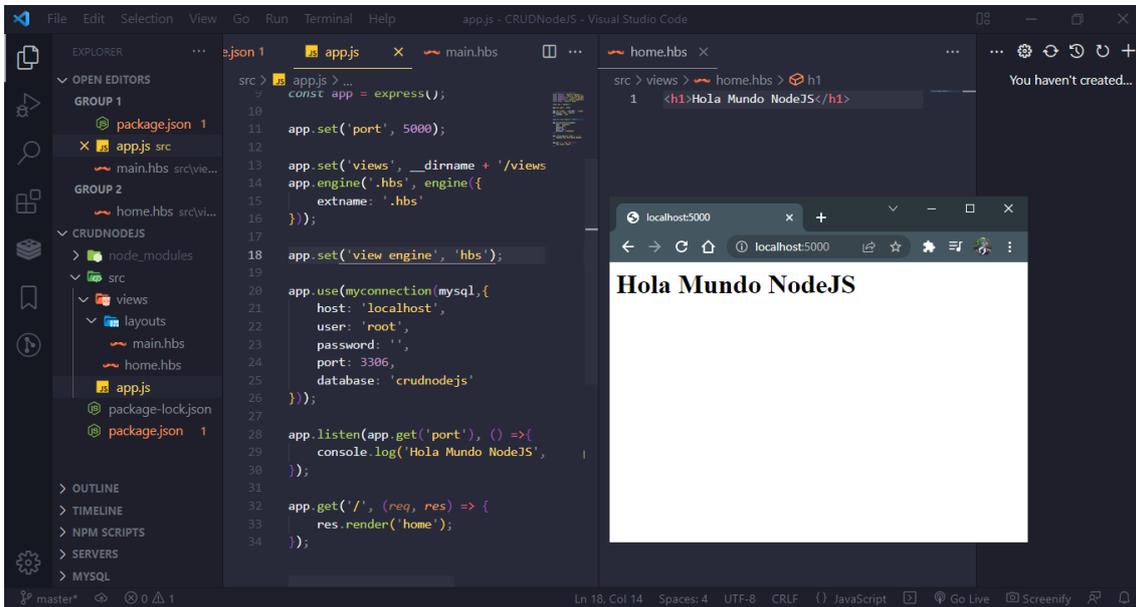


Figura 8: Aplicación NodeJS (muestra un Hola Mundo)

Fuente: Kerly Lucia Vallejo Cano

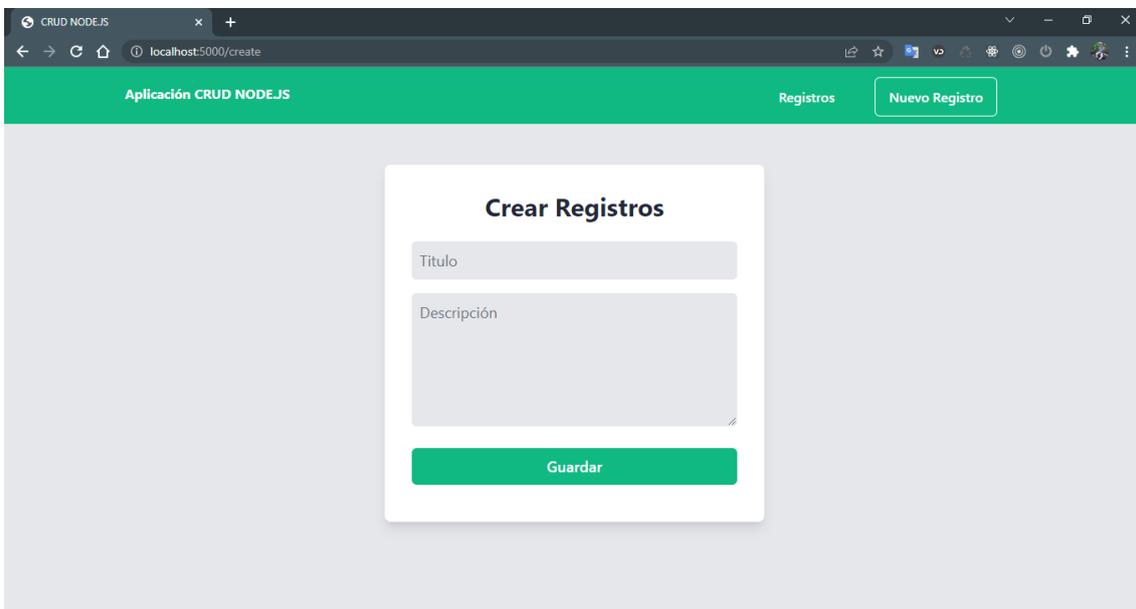


Figura 9: Aplicación CRUD NodeJS

Fuente: Kerly Lucia Vallejo Cano

```
db.php > ...
1 <?php
2
3 $db = new Mysqli;
4
5 $db->connect('localhost','root','','crudPHP');
6
7 if(!$db){
8
9     echo "success";
10 }
11
12
13
14
```

Figura 10: Creación de la aplicación en PHP

Fuente: Kerly Lucia Vallejo Cano

```
index.php > ...
1 <!DOCTYPE html>
2 <?php include 'db.php';
3
4 $page = (isset($_GET['page']) ? $_GET['page'] : 1);
5 $perPage = (isset($_GET['per-page']) && ($_GET['per-page'] <= 50 ? $_GET['per-page'] : 5);
6 $start = ($page > 1) ? ($page * $perPage) - $perPage : 0;
7
8
9 $sql = "select * from registros limit ".$start.", ".$perPage." ";
10 $total = $db->query("select * from registros")->num_rows;
11 $pages = ceil($total / $perPage);
12
13 $rows = $db->query($sql);
14
15
16 ?>
17 <html>
18 <head>
19 <script src="https://ajax.googleapis.com/ajax/libs/jquery/2.2.4/jquery.min.js"></script>
20 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity=
21 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.m
22 <title>Crud App</title>
23 </head>
24 <body>
25 <div class="container">
26 <center><h1>Todo list</h1></center>
27
28 <div class="row" style="margin-top: 70px;">
```

Figura 11: Código Interfaz principal en PHP

Fuente: Kerly Lucia Vallejo Cano

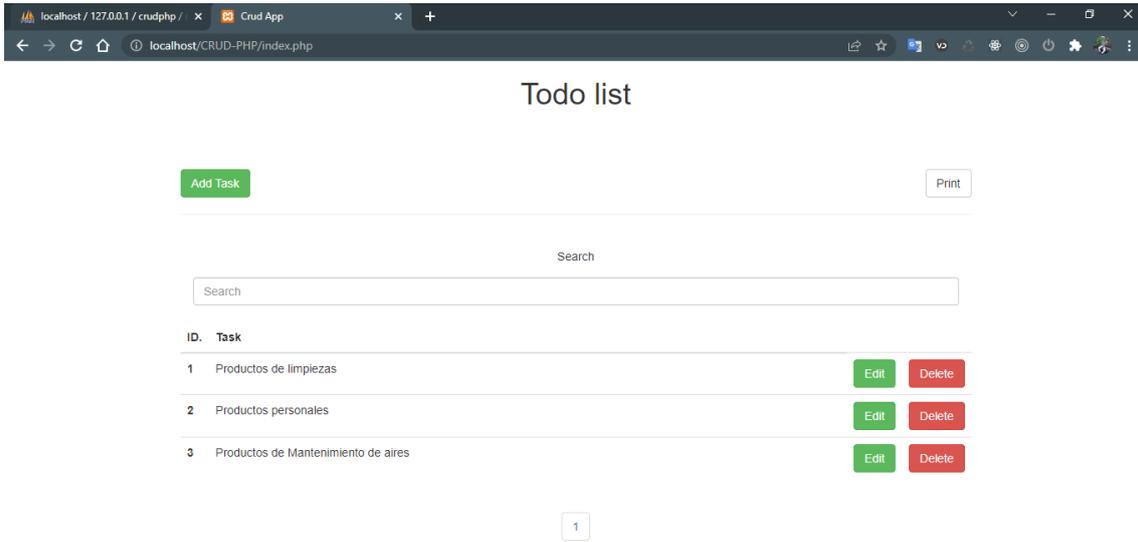


Figura 10: Aplicación CRUD PHP

Fuente: Kerly Lucia Vallejo Cano