|  |
| --- |
| **Informe**  **Bases de Datos** |
|  |

| Integrantes: | Leandro Garcia Tomás Guajardo Fernanda Abarca Sebastian Arroyo  Felipe Painenao |
| --- | --- |
|  |  |
|  |  |
|  |  |
| Asignatura: | Bases de Datos |

|  | Abril 2025 |  |
| --- | --- | --- |

**Índice de Contenidos**

[1. Introducción 3](#_myfuse1azhd8)

[2. Formulación Inicial 3](#_ldb2twyyyvom)

[3. Estado del Arte 3](#_eb4emg3a2uen)

[4. Front End 3](#_9hupozo8ktsu)

[5. Back End 3](#_bc7kqls2yr10)

[6. Bases de Datos 3](#_4efnm1j1jlas)

[7. Conclusión 4](#_809zl6v71mvw)

[8. Formato 4](#_py3fbe2j0m04)

[9. Bibliografía 5](#_r4kf8yi44kit)

[10. Anexos 5](#_apjy0suv4kmc)

# **Introducción**

La influencia significativa de la transformación digital en el comercio del sector de los videojuegos ha facilitado la comercialización mediante espacios virtuales, pero al mismo tiempo ha dejado a muchos comercios pequeños en desventaja, ya que no disponen de los medios necesarios para crear sus propios programas. Para abordar esto, la compañía DEBEDE sugiere la creación de una plataforma integrada que permita a estos negocios ofrecer sus artículos en línea. Esta iniciativa tiene como objetivo satisfacer esa demanda, permitiendo a través de una aplicación web la administración, organización, promoción y comercialización de videojuegos por parte de diversas tiendas, todo en una misma plataforma.

# **Formulación Inicial**

Contexto:

El mercado virtual de videojuegos ha experimentado un notable aumento, sin embargo, los establecimientos pequeños no poseen los recursos necesarios para ser competitivos en el ámbito digital. DEBEDE ha reconocido esta problemática y pretende abordarla a través de una plataforma de colaboración.

Problema:

La disparidad tecnológica entre los grandes minoristas y los pequeños ha frustrado la capacidad de los últimos para integrarse en el mercado digital de videojuegos, restringiendo su expansión y competencia.

Objetivo General:

Desarrollar un portal en línea que ofrezca a pequeñas tiendas de videojuegos la capacidad de exhibir y gestionar sus productos a través de la web.

Objetivos Específicos:

- Establecer diferentes tipos de usuarios: administrador, encargado de tienda y consumidor.

- Facilitar a los encargados de tienda la carga y organización de videojuegos.

- Incorporar características como un carro de compras, lista de deseos y clasificaciones de ventas.

- Crear una base de datos sólida para controlar usuarios, videojuegos, transacciones y categorías.

- Analizar y elegir las tecnologías más adecuadas para el Front End, Back End y la Base de Datos.

# 

# **Estado del Arte**

Utilidad del desarrollo web en la actualidad:

El desarrollo web, que en palabras simples es el proceso de desarrollar sitios web o aplicaciones web ejecutadas a través de internet (Webb, 2024), tiene una gran importancia para el éxito de estas, especialmente si están orientadas a un negocio o empresa.

Elementos como una interfaz funcional y amistosa con el usuario, la implementación de comercio electrónico en el sitio web, funcionalidades (plugins, APIs, etc.), optimización que considera cosas como la velocidad o escalabilidad, etc., (Popov, 2024) son clave para hacer destacar el sitio web del propietario y ponerlo en la mira del público/cliente, especialmente si el propósito del sitio es impulsar un negocio.

Plataformas similares al enunciado en el mercado del desarrollo web actual:

Plataformas populares hoy en día para la compra y venta de videojuegos digitales vendrían siendo algunas como Steam, Epic Games Store, Google Play Store, etcétera. Sin embargo, la propuesta del enunciado es bien distinta al mismo tiempo, ya que tiene más en común con plataformas como Amazon, Mercado Libre, o Etsy, aunque DEBEDE tendría un enfoque exclusivo en los videojuegos. Pero la similitud vendría siendo que ambos son mercados en línea donde múltiples tiendas poseen su propio espacio para vender sus cosas. Sin embargo, DEBEDE tiene un matiz extra: posee un sistema compartido que permite la clasificación de los productos (en este caso, los videojuegos) para facilitar su compra por internet, todo en un mismo lugar.

# **Front End**

El front-end es la parte del desarrollo web que interactúa directamente con el usuario, o sea, todo lo que se ve en una página web o aplicación. Por ejemplo, textos, botones, menús, imágenes, etc. Básicamente es la cara visible de cualquier sitio o app. Todo lo que el usuario toca o ve está en el front-end, por lo que es muy importante que se cree una experiencia de usuario (UX) que sea clara y funcional. (*¿Cuál es la diferencia entre el front end y back end en el desarrollo de aplicaciones?*, s.f.)

Para crear estas interfaces, los desarrolladores usan principalmente tres tecnologías esenciales:

HTML (Hypertext Markup Language): Este es el lenguaje que da estructura a la página web. En HTML, se definen los elementos básicos como los encabezados, párrafos, listas o imágenes.

CSS (Cascading Style Sheets): Con CSS se le puede dar estilo a la página, cambiando colores, fuentes, márgenes, y hasta creando efectos visuales. Ayuda a darle a la página un diseño atractivo y organizado, además de solo contenido.

JavaScript: Hace que la página web sea interactiva. Sin él, las páginas solo mostrarían contenido estático, pero gracias a JavaScript, se pueden hacer cosas como que las páginas se actualicen sin necesidad de recargarlas, validar formularios o animar elementos en la pantalla.

### Frameworks y Librerías

HTML, CSS y JavaScript son las bases de todo, pero también hay herramientas que ayudan a hacer el trabajo más rápido y eficiente. Los frameworks y librerías son colecciones de código ya hecho que los desarrolladores pueden usar para no tener que reinventar la rueda.

Algunos de los más populares son:

React: Es una librería de JavaScript creada por Facebook. Se utiliza principalmente para construir interfaces de usuario interactivas. Una de sus principales ventajas es que divide las interfaces en componentes reutilizables, lo que facilita mucho el mantenimiento y la escalabilidad de las aplicaciones.

Vue.js: Es un framework progresivo que también facilita el desarrollo de interfaces interactivas. Es fácil de aprender y se integra rápidamente en proyectos existentes, lo que lo hace ideal para quienes quieren empezar rápido.

Angular: Es un framework más robusto desarrollado por Google. Está diseñado para crear aplicaciones grandes y complejas. Ofrece muchas soluciones "listas para usar" y está pensado para equipos grandes que necesitan trabajar de manera organizada y eficiente.

Comparativa: ¿Cuál es la mejor opción?

No hay una respuesta única, ya que todo depende del proyecto y las necesidades del usuario. Pero en general:

React es usado bastante porque es muy flexible y tiene varias herramientas que lo apoyan. Va perfecto si se quiere hacer una app que necesita mucha interacción con el usuario.

Vue.js es más simple y fácil de aprender. Es liviano, así que es mejor para proyectos medianos o pequeños.

Angular es más completo y está pensado para proyectos más grandes. Tiene todo integrado desde el principio, algo que ayuda a mantener todo ordenado cuando el proyecto crece mucho. Es más pesado, pero sirve para proyectos más complejos.

### ¿Por qué estas tecnologías son tan populares?

Esas tres tecnologías (HTML, CSS y JavaScript) ayudan a que las páginas web se vean bien y sean fáciles de usar. HTML es como la estructura, CSS le da el estilo (colores, tamaños, etc.) y JavaScript hace que todo sea más interactivo, como botones que responden o menús que se abren sin recargar la página.

Además, hay herramientas extra como React, Vue o Angular que hacen más fácil el navegar el sitio, ahorran tiempo y ayudan a que todo esté más ordenado, sobre todo si el proyecto es grande.

En resumen, depende de lo que el usuario necesite. Si se quiere algo flexible y que se pueda ampliar fácilmente, React es buena opción, pero si se prefiere algo más simple y rápido de aprender, mejor usar Vue. Y si se está con un proyecto grande y se prefiere que todo venga más armado desde el principio, Angular es buena idea también.

# **Back End**

Antes de definir y analizar las tecnologías utilizadas para el back end, es necesario explicar qué es y para qué sirve esta parte de la página web o aplicación.

El back end es la parte que el usuario de la aplicación no ve. En él se pueden encontrar los aspectos lógicos de la App o página web, se encuentra el almacenamiento de datos, la seguridad y la lógica detrás de la aplicación o página web. (*Backend: ¿Qué es y para qué sirve?*, s.f.)

Como se menciona anteriormente en el Front end, se utilizan esencialmente tres tecnologías: HTML, Javascript y CSS, cada una contando con un framework o librería que facilita la creación del programa.

Framework de Javascript:

Express.js es un framework bastante simple, que utiliza Node.js. es uno de los frameworks más populares a la hora de programar por su simpleza, está mayormente destinado a páginas web de una sola pagina y multipaginas, ofrece enrutamientos, manejo de peticiones HTTP y middleware, haciéndolo flexible y fácil de utilizar. (*¿Qué es Express.js? Todo lo que Debes Saber, 2025)*

Framework de CSS:

Bootstrap es un framework hecho con CSS ya que en sí este lenguaje de programación no usa frameworks, pero al igual que el anterior es bastante simple ya que viene con componentes predefinidos, como botones, formularios, alertas, tarjetas, etc. Es compatible con navegadores principales haciéndolo bastante accesible al público. (Moraguez, 2023)

Para cerrar el tema, comparando las dos herramientas para crear el back end, y pensando en la simpleza, accesibilidad y herramientas que ofrece cada framework se debe destacar que bootstrap es una framework bastante completa la cual facilita demasiado la programación del backend. (Moraguez, 2023)

# **Bases de Datos**

Una base de datos básicamente es un lugar donde se guarda información de forma ordenada para poder encontrarla y usarla fácilmente después. Todo esto se hace desde un computador y se usa un programa especial (que se llama sistema de gestión de bases de datos o DBMS) para manejar todo: guardar, cambiar, organizar y buscar los datos cuando se necesiten. Es como una especie de archivo digital donde todo está bien clasificado para no perder nada. Por ejemplo, las bibliotecas digitales, los registros de clientes o el historial de lo que se ve en internet funcionan gracias a una base de datos.

La selección de la base de datos más adecuada se basa en las demandas específicas del proyecto. PostgreSQL es muy completa si se va a hacer algo grande y complicado, MySQL es muy confiable y funciona bien para la mayoría de las páginas web que no son tan pesadas, MongoDB se usa cuando se tiene mucha información que no está tan ordenada o estructurada, y Firebase es útil si se necesita que la información se actualice en tiempo real (como en chats o apps que cambian frecuentemente). Al final, todo depende del tipo de datos que se va a manejar, cuánta gente va a usar la app, qué tan rápido se necesita que funcione y también qué tan fácil es encontrar ayuda o documentación sobre esa base de datos si algo falla. (*¿Qué es una base de datos?*, 2020)

# **Conclusión**

Creemos que se cumplieron los objetivos, ya que pudimos entender mejor cómo la tecnología puede ayudar a las pequeñas empresas en un mercado tan específico y competitivo como el de los videojuegos, esto especialmente porque la idea del DEBEDE puede ser muy útil y económica para este tipo de negocios que no tienen los recursos ni capital necesario para hacer algo de ese calibre.

Algunas complicaciones y limitaciones fueron simplemente el encontrar fuentes confiables y serias con respecto a esto, ya que la mayoría eran blogs (y terminamos usándolos igual). Asimismo, fue difícil encontrar algunos ejemplos que coincidieran con lo que se estaba buscando, específicamente el DEBEDE.

En cuanto a lo que aprendimos, fue más que nada el darnos cuenta lo importante que es planificar bien y pensar en el usuario, tanto el que vende como el que compra. Nos dimos cuenta que un buen diseño, tanto del sistema como del proyecto en general, es esencial para que un producto funcione.

# **Bibliografía**

Webb, M. (14 de mayo de 2024). *What is web development? Definition, Processes & types*. Techopedia. <https://www.techopedia.com/definition/23889/web-development>

Popov, I. (13 de agosto de 2024). *How web development can improve any business’s performance (regardless of the niche)*. Forbes. <https://www.forbes.com/councils/forbesbusinesscouncil/2023/04/12/how-web-development-can-improve-any-businesss-performance-regardless-of-the-niche/>

*¿Qué es una base de datos?*. Oracle México. (s.f.). <https://www.oracle.com/mx/database/what-is-database/>

Concepto.de. (2023). Base de datos - Qué es, tipos y ejemplos. Recuperado de https://concepto.de/base-de-datos/

Dataxpertos. (2024). PostgreSQL lidera el ranking de bases de datos más utilizadas en 2024. Recuperado de <https://dataxpertos.com/postgresql-lider-en-bases-de-datos-2024/>

Marketing Web. (2023). ¿Cuáles son las bases de datos más utilizadas en 2023?. Recuperado de <https://marketingweb.blog/bases-de-datos-mas-utilizadas/>

Wikipedia. (2024). MySQL. En Wikipedia, la enciclopedia libre. Recuperado de <https://es.wikipedia.org/wiki/MySQL>

MDN Web Docs. (s.f.). *Front-end web developer*. Mozilla. Recuperado el 25 de abril de 2025, de<https://developer.mozilla.org/en-US/docs/Learn/Front-end_web_developer>

W3Schools. (s.f.). *Learn web development*. W3Schools. Recuperado el 25 de abril de 2025, de<https://www.w3schools.com/> Contenido: Este sitio ofrece tutoriales prácticos y fáciles de seguir sobre HTML, CSS, JavaScript, y otras herramientas esenciales del desarrollo frontend. Es ideal para principiantes.

freeCodeCamp. (s.f.). *Responsive Web Design Certification*. freeCodeCamp. Recuperado el 25 de abril de 2025, de https://www.freecodecamp.org/learn/ Contenido: Proporciona certificaciones gratuitas y ejercicios prácticos para aprender desarrollo web frontend desde cero, con enfoque en diseño responsivo, accesibilidad, y JavaScript.

Amazon Web Services. (s.f.). *La diferencia entre frontend y backend*. AWS. Recuperado el 25 de abril de 2025, de<https://aws.amazon.com/es/compare/the-difference-between-frontend-and-backend/> Contenido: Explica la diferencia entre frontend y backend en el desarrollo de aplicaciones web, con ejemplos claros y orientados a su aplicación en entornos en la nube.

*Backend: ¿Qué es y para qué sirve?*. Gluo. (s.f.). https://www.gluo.mx/blog/backend-que-es-y-para-que-sirve

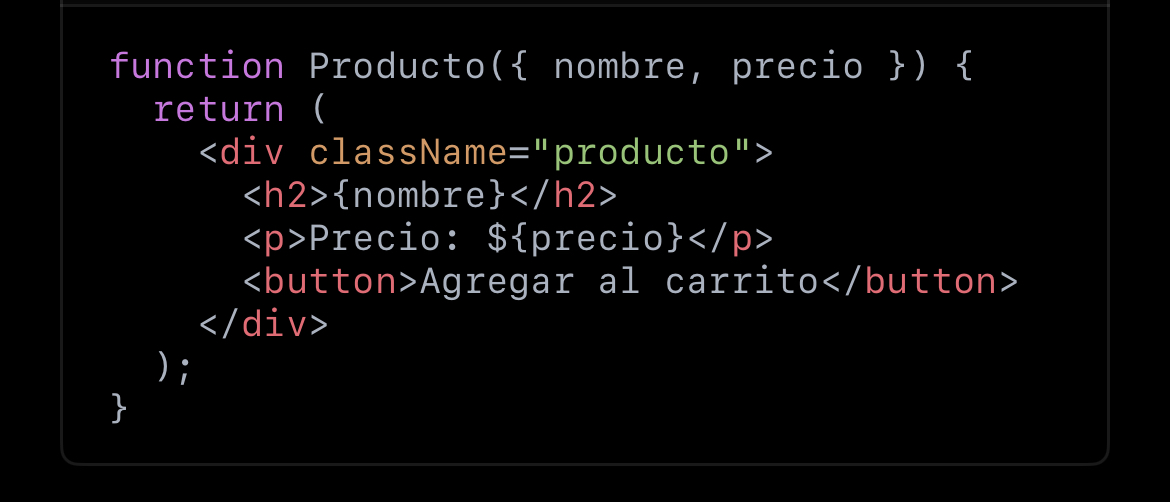
*¿Qué es express.js? Todo Lo Que Debes saber*. Kinsta®. (5 de marzo de 2025). <https://kinsta.com/es/base-de-conocimiento/que-es-express/>

Moraguez, por E. R. (23 de febrero de 2023). *¿Qué es bootstrap: Cómo funciona y para qué sirve?*. LovTechnology. https://lovtechnology.com/que-es-bootstrap-como-funciona-y-para-que-sirve/

# **Anexos**

1. Estado del arte
2. Front end

| Tecnología | Tipo | Ventajas principales | Casos de uso ideales |
| --- | --- | --- | --- |
| React | Librería | Componentes reutilizables, gran comunidad, compatible con múltiples entornos | Aplicaciones web interactivas y escalables |
| Vue.js | Framework | Fácil de aprender, liviano, buena documentación | Proyectos medianos o pequeñas aplicaciones rápidas |
| Angular | Framework | Completo y estructurado, ideal para trabajo en equipo | Aplicaciones grandes y corporativas |
| Bootstrap | Librería CSS | Estilos prediseñados, responsive por defecto | Prototipado rápido, apps con diseño estándar |
| Tailwing CSS | Framework CSS | Altamente personalizable, utilidad por clases, enfoque moderno | Aplicaciones con diseños personalizados |

Fragmentos de código de ejemplo:

Este componente en React muestra un producto con su nombre, precio y un botón de acción. Representa una estructura reutilizable dentro del sistema de venta de videojuegos.

1. Back end

| Framework | Lenguaje base | Tipo | Funcionalidades principales | Ventajas clave | Uso común |
| --- | --- | --- | --- | --- | --- |
| Express.js | JavaScript | Back End | Enrutamiento, peticiones HTTP, middleware, integración con bases de datos | Ligero, rápido, muy usado con Node.js, curva de aprendizaje baja | Aplicaciones web dinámicas |
| Bootstrap | CSS | Front End\* | Componentes predefinidos (botones, menús, formularios, tarjetas) | Compatible con todos los navegadores, diseño responsive rápido | Prototipos y aplicaciones visuales |

\*Nota: Aunque Bootstrap es una herramienta de Front End, se mencionó en el contexto del Back End por su contribución a simplificar la construcción general del sistema.

1. Base de datos

| Base de Datos | Tipo | Ventajas | Casos de uso |
| --- | --- | --- | --- |
| PostgreSQL | Relacional | Soporte para tipos de datos avanzados, escalabilidad, código abierto | Aplicaciones empresariales complejas, análisis de datos |
| MySQL | Relacional | Facilidad de uso, amplia adopción | Aplicaciones web, sistemas de gestión de contenido |
| MongoDB | NoSQL | Manejo de datos no estructurados, escalabilidad horizontal | Aplicaciones con grandes volúmenes de datos no estructurados |
| Firebase | NoSQL | Sincronización en tiempo real, backend como servicio | Aplicaciones móviles y web en tiempo real |