

DOI: <https://doi.org/10.56712/latam.v4i2.906>

Estudio comparativo para la toma de requerimientos de software para el desarrollo de una aplicación web en investigación psicológica del bullying educativo

Comparative study for software requirements gathering for the development of a web application in psychological research on educational bullying

Marco Israel Bermeo Vintimilla

mibermeo@sudamericano.edu.ec

<https://orcid.org/0000-0002-8116-4204>

Instituto Tecnológico Sudamericano de Cuenca

Cuenca – Ecuador

Francis Fernando Buele Lanchi

ffbuele@sudamericano.edu.ec

<https://orcid.org/0000-0001-6215-2868>

Instituto Tecnológico Sudamericano de Cuenca

Cuenca – Ecuador

Galo Patricio Hurtado Crespo

gphurtado@sudamericano.edu.ec

<https://orcid.org/0000-0002-7190-140X>

Instituto Tecnológico Sudamericano de Cuenca

Cuenca – Ecuador

Artículo recibido: 15 de julio de 2023. Aceptado para publicación: 29 de julio de 2023.

Conflictos de Interés: Ninguno que declarar.

Resumen


El objetivo principal es explorar diferentes herramientas para el desarrollo de aplicaciones web que estén relacionadas con la investigación del acoso escolar, se aplicó la metodología de investigación correlacional que permitió comparar las principales herramientas que se utilizan en el desarrollo de aplicaciones web. En el front-end, JavaScript (JS) destaca como el lenguaje más utilizado con el 63.61%, y NextJS se menciona como una opción para mejorar el rendimiento, seguridad, la experiencia del usuario y la facilidad de implementación con una aceptación del 82.52%. En el back-end, se evalúan lenguajes como Python, JavaScript, Java y Kotlin, Kotlin se destaca por su seguridad, escalabilidad y facilidad de implementación con un 76.40%, siendo compatible con bases de datos y adecuado para el desarrollo en la investigación del acoso escolar. En cuanto a las bases de datos, se mencionan PostgreSQL y MongoDB. PostgreSQL destaca por su seguridad y control de acceso avanzado, y MongoDB maneja grandes volúmenes de datos y consultas complejas, además de ofrecer funciones de seguridad como cifrado de datos. Se mencionan frameworks como Spring Boot y Ktor, que proporcionan soluciones completas y confiables para el desarrollo en el back-end. Así también se plantea como sugerencia la aplicación de SCRUM como metodología de desarrollo en la que se detalla cuáles son los roles y funciones de cada uno de los usuarios en el proceso de desarrollo, se plantean un diagrama que permita a los desarrolladores tener todas las herramientas y argumentos necesarios para la implementación de la aplicación web.

Palabras clave: desarrollo de software, investigación correlacional, aplicación web, psicológica, bullying

Abstract

The main objective is to explore different tools for the development of web applications that are related to bullying research, the correlational research methodology was applied to compare the main tools used in the development of web applications. In the front-end, JavaScript (JS) stands out as the most used language with 63.61%, and NextJS is mentioned as an option to improve performance, security, user experience and ease of the implementation with an acceptance of 82.52%. In the back-end, languages such as Python, JavaScript, Java and Kotlin are evaluated, Kotlin stands out for its security, scalability and ease of implementation with 76.40%, being compatible with databases and suitable for development in bullying research. As for databases, PostgreSQL and MongoDB are mentioned. PostgreSQL stands out for its security and advanced access control, and MongoDB handles large volumes of data and complex queries, as well as offering security features such as data encryption. Frameworks such as Spring Boot and Ktor are mentioned, which provide complete and reliable solutions for back-end development. It is also suggested the application of SCRUM as a development methodology in which the roles and functions of each user in the development process are detailed, and a diagram is proposed that allows the developers to identify the roles and functions of each of the users in the development process.

Keywords: software development, correlational research, web application, psychological, bullying

Todo el contenido de LATAM Revista Latinoamericana de Ciencias Sociales y Humanidades, publicados en este sitio está disponibles bajo Licencia Creative Commons . 

Como citar: Bermeo Vintimilla, M. I., Buele Lanchi, F. F., & Hurtado Crespo, G. P. (2023). Estudio comparativo para la toma de requerimientos de software para el desarrollo de una aplicación web en investigación psicológica del bullying educativo. *LATAM Revista Latinoamericana de Ciencias Sociales y Humanidades* 4(2), 4375–4391. <https://doi.org/10.56712/latam.v4i2.906>

INTRODUCCIÓN

El bullying educativo es un problema social y psicológico de relevancia global que afecta a estudiantes en todas las etapas educativas. Este fenómeno se caracteriza por actos repetidos de agresión física, verbal o psicológica, llevados a cabo por uno o varios individuos con la (Ojeda et al., 2023)

intención de intimidar, humillar o dañar a otros estudiantes. Según estudios recientes, el bullying puede tener un impacto significativo y negativo en la salud mental y el bienestar de las víctimas, así como en el ambiente escolar y el rendimiento académico. (Crespo, 2019)

Ante la magnitud de este problema, la investigación psicológica juega un papel esencial en la comprensión, prevención y tratamiento del bullying educativo. En este contexto, el diseño y análisis de requerimientos de software para la investigación psicológica del bullying educativo ha surgido como un área de creciente interés.

Se han realizado varios estudios e investigaciones sobre el tema de bullying en instituciones educativas, entre ellos, un estudio realizado en la unidad escolar José Faustino Sánchez Carrión de Trujillo – Perú, en el cual contó con la ayuda de un aplicativo psicológico en adolescentes, la población era 433 estudiantes donde se tomó una muestra de 27 estudiantes del primer año de secundaria. Se realizó una investigación cuantitativa junto a un diseño experimental utilizando el método de cascada como metodología de diseño de software. Se optó por un método de análisis de prueba paramétrica que obtuvo estos resultados basados en 3 factores: a) la incidencia del factor familiar, que disminuyó en un 11 %, b) el factor social y, c) el intrapersonal, donde hubo una disminución del 22.3% de cada uno. Acorde a los resultados se observó un aumento en la autoestima del adolescente y se logró disminuir los casos de bullying dentro de la institución, al igual que se logró intervenir en la conducta violenta del adolescente en el entorno familiar. (García, 2020)

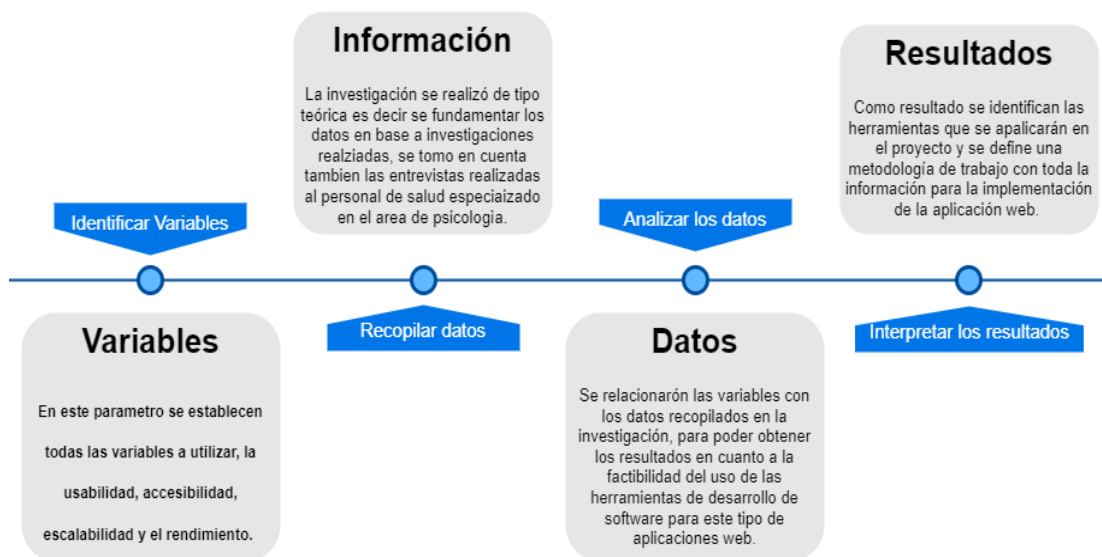
En el presente estudio, el objetivo es profundizar en el diseño y análisis de requerimientos de software para la investigación psicológica del bullying educativo, se busca proporcionar una visión integral de cómo las herramientas de software pueden ser aplicadas para facilitar la investigación y el tratamiento del bullying educativo. Ojeda et al. (2023) mencionan que “en la sociedad se ha encontrado carencia de entornos virtuales de aprendizaje por lo cual se propone implementar herramientas digitales”, por tanto, el análisis para una propuesta de desarrollo de software apoyaría a la investigación del bullying educativo. Para lograr el objetivo, la metodología incluye la recolección de datos a través de tablas comparativas, entrevistas y presentar una propuesta de desarrollo de software con la herramienta que se determine en el análisis comparativo y que permita cumplir con todos los requerimientos de la aplicación web. En la sección de resultados, se analiza cada tabla en relación a cada componente de software y sus tecnologías seleccionadas, así como una propuesta para la aplicación web para su posterior desarrollo. Finalmente, en la sección de discusión, se expone las conclusiones a las cuales se llegaron que justifique el diseño de la aplicación, incluye sugerencias para posibles futuras investigaciones en el campo del diseño de software que tengan un enfoque en la investigación psicológica del bullying educativo.

METODOLOGÍA

Se aplicó la metodología correlacional que permite establecer una comparativa entre dos o más variables, identificando las relaciones que puedan existir en este caso entre las diferentes variables como la usabilidad, accesibilidad, seguridad, y el rendimiento, de las diferentes herramientas de desarrollo de software, a continuación, se presentan las fases de investigación con relación a la propuesta de investigación:

Figura 1

Fases de la metodología de investigación aplicada en el proyecto



Fuente: Elaboración propia.

En la gráfica se pueden observar las fases de la metodología de investigación, y en cada una de las fases se detalla cada uno de los componentes y cómo se obtuvo la información para la ejecución de la metodología de investigación en el proyecto. Cabe recalcar que en el proceso de investigación se trabajó de manera coordinada con expertos en el área de psicología que son los beneficiarios directos del proyecto.

Recolección de datos

La recolección de datos para el estudio se realiza mediante tablas comparativas que permiten identificar los requisitos funcionales y no funcionales. Las tablas comparativas proporcionan un enfoque estructurado para recopilar y presentar datos sobre diferentes sistemas de software. En estas tablas, cada fila representa una variable específica (como el rendimiento, la usabilidad, la escalabilidad, seguridad, accesibilidad) y cada columna representa un sistema de software diferente (como PostgreSQL, MySQL, MongoDB, entre otros). Se realizó una encuesta en la plataforma de Google Forms, donde se detallaron los 4 componentes (back-end, front-end, base de datos y visualizador de datos) de un sitio web con las tecnologías propuestas para que sean calificados de acuerdo a las variables.

Se identifican los requisitos funcionales y no funcionales que desempeñan un papel integral en la recolección de datos. Los requisitos funcionales son las funcionalidades que se espera que realice un sistema de software, mientras que los requisitos no funcionales se refieren a los criterios que se utilizan para juzgar el funcionamiento de un sistema, como la usabilidad, la escalabilidad, seguridad y accesibilidad. Estos resultados proporcionan datos cualitativos que permiten proponer una aplicación web que cumpla con las expectativas de los usuarios finales, como son los profesionales en el campo de la psicología.

Para analizar de manera efectiva los diferentes componentes del software y cómo se relacionan entre sí para cumplir con los requerimientos de la investigación psicológica del bullying educativo, adoptamos una arquitectura MVC. La presente arquitectura tiene como objetivo clasificar el software en front-end, back-end, base de datos y software de análisis de datos.

Análisis de datos

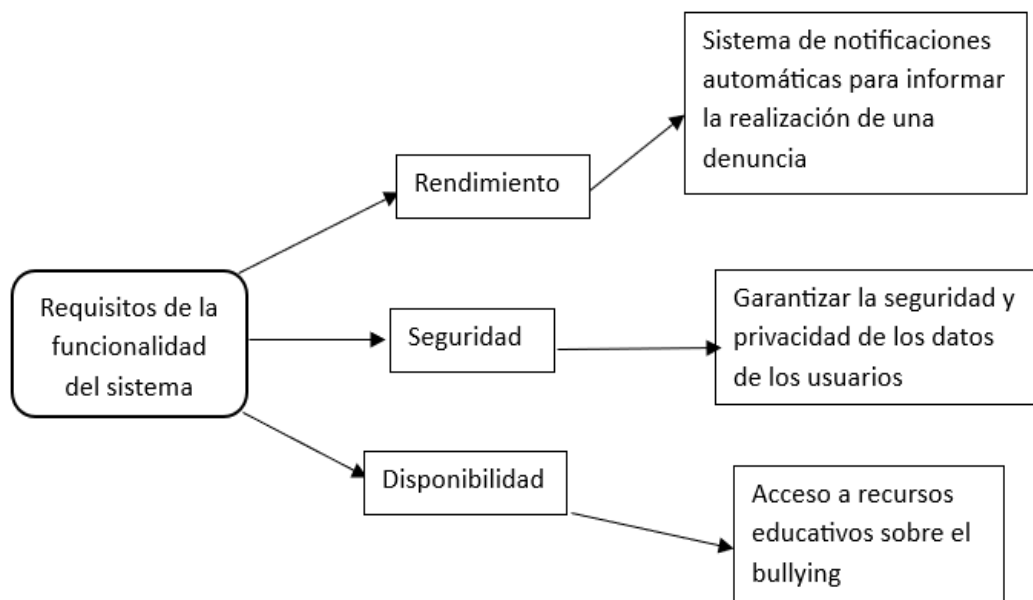
Los datos recogidos en las tablas se analizarán cualitativamente para identificar patrones y tendencias. Cada variable se evaluará en términos de cómo los diferentes sistemas de software se comparan entre sí. Los resultados de este análisis proporcionarán una visión detallada de las fortalezas y debilidades de cada sistema de software en relación con los requerimientos de software para la investigación psicológica del bullying educativo. Con las respuestas obtenidas de las encuestas acorde a las variables se determinarán las tecnologías más apropiadas para el desarrollo del aplicativo web.

RESULTADOS Y DISCUSIÓN

En la sociedad se ha encontrado carencia de entornos virtuales de aprendizaje por lo cual se propone implementar herramientas digitales como elementos web 2.0 y 3.0. Las variables que influyen en la detección y prevención del acoso escolar incluyen la edad de los estudiantes, el género, la personalidad, la apariencia física, las relaciones familiares y el ambiente escolar. Estos factores son fundamentales para comprender cómo se desarrolla y se percibe el acoso en el entorno escolar.

Figura 2

Requisitos



Nota: El gráfico representa los requisitos básicos para la funcionalidad del sistema.

Fuente: Elaboración propia.

Para el desarrollo y diseño de software se debe tener en cuenta cada uno de los componentes del sistema, los cuales son: el back-end, front-end, bases de datos y la visualización de datos. Para la investigación del bullying educativo se propone un sistema que permita el ingreso de la información, para ser analizada mediante las variables identificadas y los resultados sean mostrados mediante tecnologías para la visualización de datos.

Tecnologías de Base de Datos

Las bases de datos se utilizan para almacenar y gestionar información en una amplia gama de aplicaciones y sistemas, desde sitios web y aplicaciones móviles hasta sistemas empresariales y científicos. Proporcionan un método eficiente para almacenar, consultar, actualizar y recuperar datos de manera segura.

Tabla 1

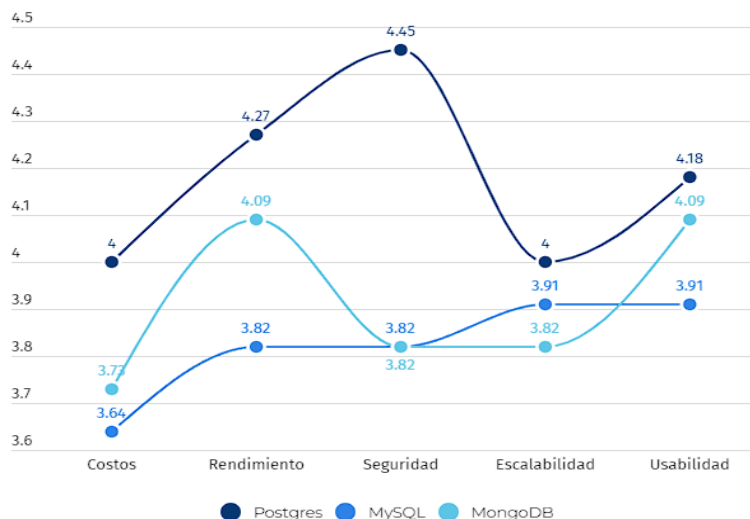
Cuadro comparativo de tecnologías de Base de Datos

Variable	PostgreSQL	MySQL	MongoDB
Rendimiento	Buen rendimiento para aplicaciones transaccionales y consultas complejas.	Buen rendimiento para aplicaciones web y carga de trabajo de lectura intensiva.	Alto rendimiento en aplicaciones que manejan grandes volúmenes de datos y requieren acceso rápido.
Usabilidad	Muy utilizado y tiene una comunidad activa que proporciona documentación y soporte.	Herramienta de código abierto, provee mayor facilidad en la configuración y el manejo de la base de datos.	Su modelo de datos flexible facilita el desarrollo rápido de aplicaciones.
Escalabilidad	Escalabilidad vertical (mejorar el hardware) y la escalabilidad horizontal (replicación y particionamiento).	Maneja cargas de trabajo crecientes añadiendo servidores y la configuración de la replicación.	Proporciona una escalabilidad horizontal nativa y permite la distribución de datos en clústeres.
Seguridad	Seguridad: como cifrado, autenticación sólida y control de acceso avanzado.	Seguridad básica como autenticación y control de acceso, puede configurarse una seguridad avanzada.	Autenticación y autorización, puede requerir configuraciones adicionales para seguridad más exigente.

Fuente: elaboración propia.

Gráfico 1

Resultados de los encuestados sobre Bases de datos



Fuente: elaboración propia.

La elección de PostgreSQL como base de datos se fundamenta en los requisitos del software y su capacidad para el procesamiento y gestión de información. Este sistema ofrece configuraciones tanto simples como avanzadas, lo que garantiza la seguridad e integridad de los datos, especialmente considerando que se trata de información personal y sensible. Es de vital importancia tomar medidas para prevenir cualquier fuga o robo de datos.

Además, destaca por su usabilidad, ya que maneja grandes volúmenes de solicitudes y proporciona muchas funcionalidades para almacenar funciones y tipos de datos. En el contexto de la analítica de datos, las bases de datos relacionales como PostgreSQL ofrecen un rendimiento óptimo y eficiente para satisfacer las necesidades del sistema.

Tecnologías de Front-End

JavaScript es fundamental en el front-end para manipular el DOM, gestionar eventos, comunicarse con servidores y almacenar datos en el navegador.

El uso de frameworks y herramientas de desarrollo agiliza el proceso y mantenerse actualizado es esencial para seguir las tendencias y mejores prácticas en este campo, además es uno de los lenguajes más utilizados según el portal de Stack Overflow (2023) y JetBrains (2022).

Tabla 2

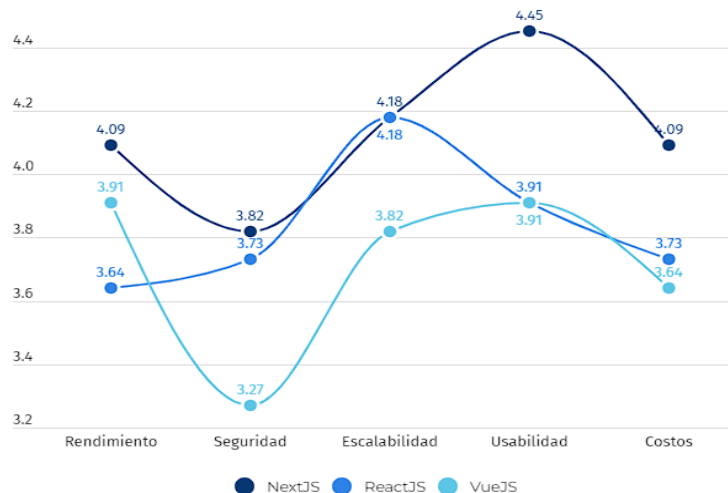
Cuadro comparativo de tecnologías de front-end

Variable	ReactJS	NextJS	VueJS
Rendimiento	Rendimiento eficiente, gracias a su Virtual DOM y su capacidad para realizar actualizaciones rápidas y optimizadas	Renderiza del lado del servidor (SSR) y genera páginas estáticas para lograr un mejor rendimiento en la carga inicial y una mejor indexación en los motores de búsqueda.	Tiene un rendimiento excelente gracias a su sistema de reactividad y la implementación eficiente de su virtual DOM.
Usabilidad	Amplia comunidad de desarrolladores, más fácil de encontrar documentación, recursos y soporte. Tecnología modular.	Un enfoque más estructurado y pensado para el desarrollo de sistemas, lo que facilita la comprensión y la adopción.	Facilidad de aprender y ofrece una sintaxis intuitiva para el desarrollo de interfaces de usuario.

Fuente: Elaboración propia.

Gráfico 2

Resultados de los encuestados sobre tecnologías de front-end



Fuente: elaboración propia.

De acuerdo con el portal de Stack Overflow (2023), JavaScript (JS) es el lenguaje más utilizado por la comunidad en el desarrollo de aplicaciones front-end. Por lo tanto, se analizaron tecnologías que han experimentado un crecimiento significativo en relación con JS.

NextJS tiene mayor rendimiento y tiempo de respuesta, lo cual mejora la experiencia para el usuario, esto es debido a su renderización del lado del servidor, generando páginas estáticas y mediante componentes de cliente permite la interacción del usuario. Cada una de las tecnologías de javascript son altamente escalables y comparten mecanismos similares para garantizar la seguridad de la información, previniendo ataques y la inyección de código malicioso, además, siguiendo las mejores prácticas, se asegura una comunicación segura con el back-end. Por último, NextJS es compatible con una amplia variedad de librerías, lo que facilita la construcción de interfaces de usuario rápidas, intuitivas y amigables.

Tecnologías de Back-End

Al hablar de back-end se trata de la parte de un sistema informático que se encarga del procesamiento y almacenamiento de datos, así como de la lógica y la funcionalidad del software, se compone de servidores, bases de datos, aplicaciones y API's que permiten la comunicación entre el cliente y el servidor. Entre algunos de los lenguajes para el desarrollo del back-end están: Python, JavaScript, Java, Kotlin, entre otros lenguajes. (Presta, 2021).

Tabla 3

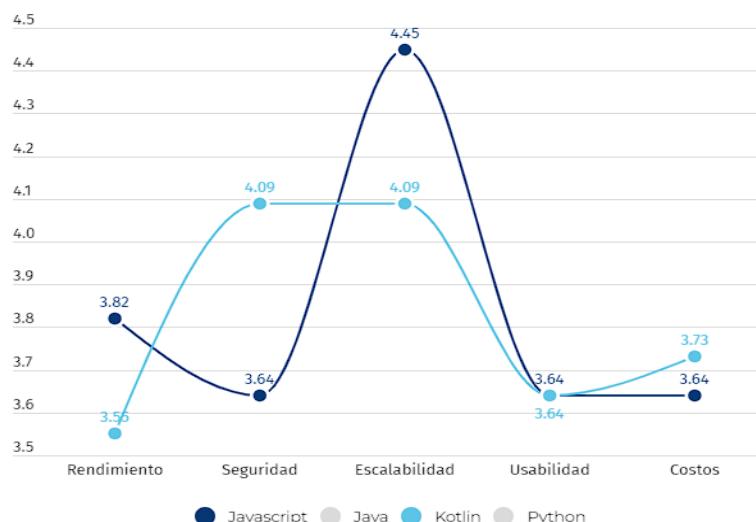
Cuadro comparativo de lenguajes de back-end

Variable	Python	JavaScript	Java	Kotlin
Rendimiento	Lenguaje interpretado, por tanto, menos eficiente en términos de rendimiento en relación a lenguajes compilados.	Lenguaje interpretado, se ejecuta en el navegador del cliente o en el entorno de ejecución de Node.js en el servidor	Capacidad para manejar grandes volúmenes de datos y aplicaciones de alto rendimiento.	Alto rendimiento en aplicaciones, con una amplia gama de optimizaciones y herramientas de rendimiento disponibles para la JVM.
Usabilidad	Sintaxis legible y su enfoque en la legibilidad del código, lo que facilita su aprendizaje y uso.	Mayor cantidad de herramientas y bibliotecas para desarrollar, compatibilidad multiplataforma.	Más estructurado, tiene una curva de aprendizaje más pronunciada en relación a los otros lenguajes.	Aprendizaje más fluido, mantiene la compatibilidad total con las pilas de tecnología basadas en Java.
Escalabilidad	Utilizado en proyectos grandes y complejos, altamente escalable.	Altamente escalable, lo que facilita su ampliación según demande la aplicación.	Escalabilidad alta y se usa en aplicaciones empresariales y de gran escala.	Escalan a una gran cantidad de clientes con requisitos de hardware modestos.
Seguridad	Seguridad relativa, sin diseñar para aspectos avanzados.	Seguridad relativa, vulnerable a ataques si no se siguen buenas prácticas.	Seguridad en el sistema de gestión de memoria y en la estructura de control de acceso.	Hereda características de seguridad de Java. Enfoque en la seguridad del tipo y la prevención de errores.

Fuente: elaboración propia.

Gráfico 3

Resultados de los encuestados sobre tecnologías de back-end



Fuente: Elaboración propia.

En el ámbito del desarrollo del back-end, se considera una variedad de lenguajes, pero Kotlin destaca por su rendimiento y disponibilidad mejorados en la JVM (Java Virtual Machines). Su enfoque en la seguridad y nulidad garantiza la protección de datos sensibles y previene errores, además, la concisión y expresividad facilitan la implementación de algoritmos y análisis de datos en la investigación del acoso escolar. Asimismo, interactúa de manera efectiva con bases de datos, asegurando la privacidad y confiabilidad de la información recopilada, por lo tanto, se posiciona como una opción sólida para abordar el desarrollo del back-end en la investigación del acoso escolar, brindando mayor rendimiento, seguridad y facilidad de implementación.

Frameworks de Kotlin

Kotlin es un lenguaje de programación moderno y versátil con sintaxis concisa. Ofrece interoperabilidad con Java, prioriza la seguridad y la nulidad, combina programación orientada a objetos y funcional, además simplifica la programación concurrente con corrutinas (flujo de ejecución asincrónico). Su creciente popularidad y capacidad para desarrollar aplicaciones multiplataforma lo convierten en una opción atractiva tanto para estudiantes como para profesionales de la programación.

Tabla 4

Cuadro comparativo de frameworks de Kotlin

Variable	Ktor	Spring Boot	Micronaut
Rendimiento	Alto rendimiento y bajo consumo de recursos.	Buen rendimiento y se usa a menudo en aplicaciones empresariales.	Es económico con los recursos y rápido de lanzar.
Usabilidad	Sintaxis clara y concisa, y sus características lo hacen fácil de usar y comprender.	Fácil de usar y comprender debido a la excelente comunidad, recursos y documentación que tiene.	Sistema basado en inyección y su API fácil de usar, reduce la necesidad de procesamiento manual.

Escalabilidad	Es escalable y capaz de manejar tráfico pesado.	Se utiliza para ejecutar aplicaciones de alta velocidad y es conocido por su escalabilidad.	Escalabilidad ideal en los entornos de nube y microservicios.
Seguridad	Admite el cifrado, la autorización y otras funciones de seguridad.	Numerosas funciones y protocolos de seguridad, incluidos OAuth y JWT.	Seguridad integrada como escaneo, autenticación y defensa contra ataques comunes.

Fuente: elaboración propia.

Se propone utilizar Spring Boot como framework principal para implementar el sistema que garantice la disponibilidad e integridad de los datos. Otra característica es la seguridad, como autenticación y autorización, que protegen la información contra amenazas externas. La validación y control de integridad de los datos aseguran su precisión y calidad. En resumen, Spring Boot proporciona una solución completa y confiable para salvaguardar la información sensible en un sistema de análisis de información.

Spring Boot es especialmente compatible con Kotlin, ya que ofrece una amplia gama de servicios que incluyen entrada de datos, módulos de seguridad y servicios web. Esto permite desarrollar aplicaciones completas de manera eficiente, otra característica es que se basa en el principio de inyección, lo que simplifica el diseño y desarrollo del código, además ofrece un sólido soporte de pruebas que se integra sin problemas.

Tecnologías de visualización de datos

La visualización de datos es el proceso de representar información y datos complejos de manera gráfica y comprensible, facilitando la identificación de patrones, tendencias y relaciones, lo que ayuda a tomar decisiones informadas y comunicar eficazmente los hallazgos.

Tabla 5

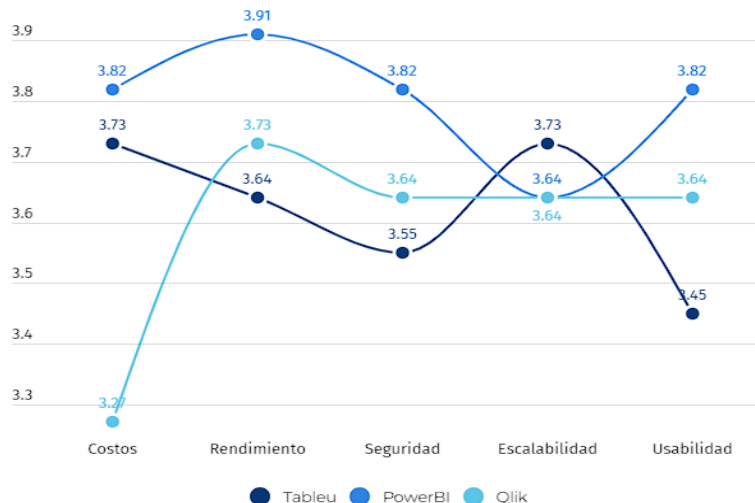
Cuadro comparativo de tecnologías para visualización de datos

Variable	Tableau	Power BI	Qlick
Rendimiento	Brinda un rendimiento sólido y eficiente.	Muy buen rendimiento en general.	Puede manejar grandes volúmenes de datos y soporta consultas complejas.
Usabilidad	Tiene una interfaz muy fácil de usar.	Interfaz intuitiva y amigable para los usuarios.	Interfaz intuitiva y sencilla para la creación de visualizaciones.
Escalabilidad	La potencia de procesamiento se puede aumentar mediante el manejo de grandes conjuntos de datos.	Puede soportar una carga de trabajo creciente.	Es calculable y capaz de manejar datos para grandes cantidades de información.
Seguridad	Sólidas funciones de seguridad como cifrado de datos, control de acceso y autenticación.	Seguridad para proteger los datos, incluidas opciones de autenticación y cifrado.	Control de acceso y la identificación del usuario, entre otras características de seguridad.

Fuente: Elaboración propia.

Figura 6

Resultados de los encuestados sobre visualizadores de datos



Fuente: Elaboración propia.

Para la visualización de datos Power BI con su diversa visualización de relaciones, integración de múltiples fuentes de datos y capacidades de análisis de relaciones, permite descubrir patrones y tendencias importantes en sus datos. Además, brinda la capacidad de crear historias consistentes y compatibles, tiene una facilidad para traer nueva información provocativa. La automatización y la programación permiten generar informes sin asistencia manual.

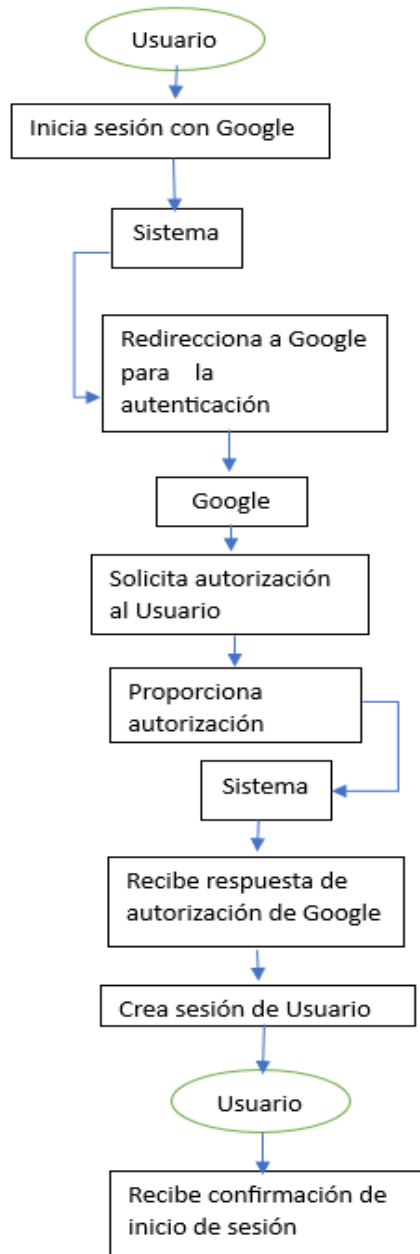
CONCLUSIÓN

Después de analizar las tecnologías recopiladas y propuestas para la comparación, se propone: a) en el back-end hacer uso del lenguaje Kotlin para ser desarrollado en Spring Boot debido a su rendimiento y seguridad para tratar la información que será almacenada en b) la base de datos de PostgreSQL, la cual permite gestionar desde una configuración simple a una avanzada según lo requiera el sistema, también permite el procesamiento de solicitudes masivas, c) en el front-end se plantea NextJS por el manejo y renderización de páginas estáticas desde el lado del servidor, y desde el cliente mediante el uso de componentes, por último, d) Power BI se considera como la mejor opción para la visualización de gráficos e interacción con los resultados.

Para la propuesta de desarrollo del sistema de software se propone un diseño prototipo sobre la interfaz y flujos de trabajo, donde se consideran pantallas como login, registro de usuarios, formularios de denuncia, panel del psicólogo para acceder a la visualización de denuncias y resultados, mientras que, el estudiante para realizar encuestas o denuncias. El prototipo fue desarrollado en el programa de figma para tener un acercamiento más real al producto que se quisiera llegar a obtener.

Figura 7

Diagrama de Secuencia Inicio de Sesión con Autenticación de Google



Nota: Propuesta de una secuencia de autenticación de sesión con Google.

Fuente: Elaboración propia.

Figura 8

Prototipo y flujos de trabajo del sistema de software

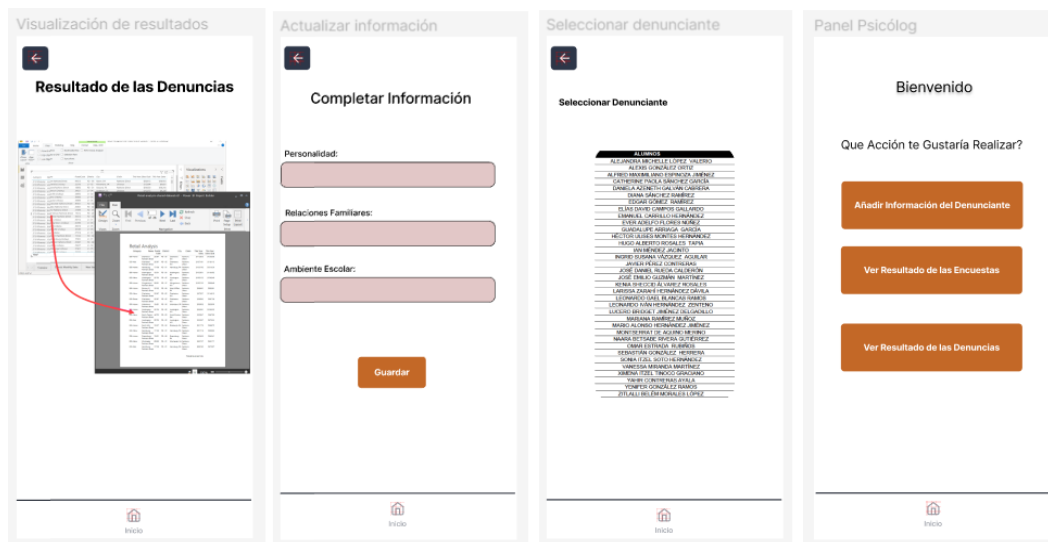


Nota: Propuesta de los flujos e interacción entre las pantallas de usuario para la aplicación

Fuente: Elaboración propia.

Figura 9

Pantallas para la interacción del psicólogo

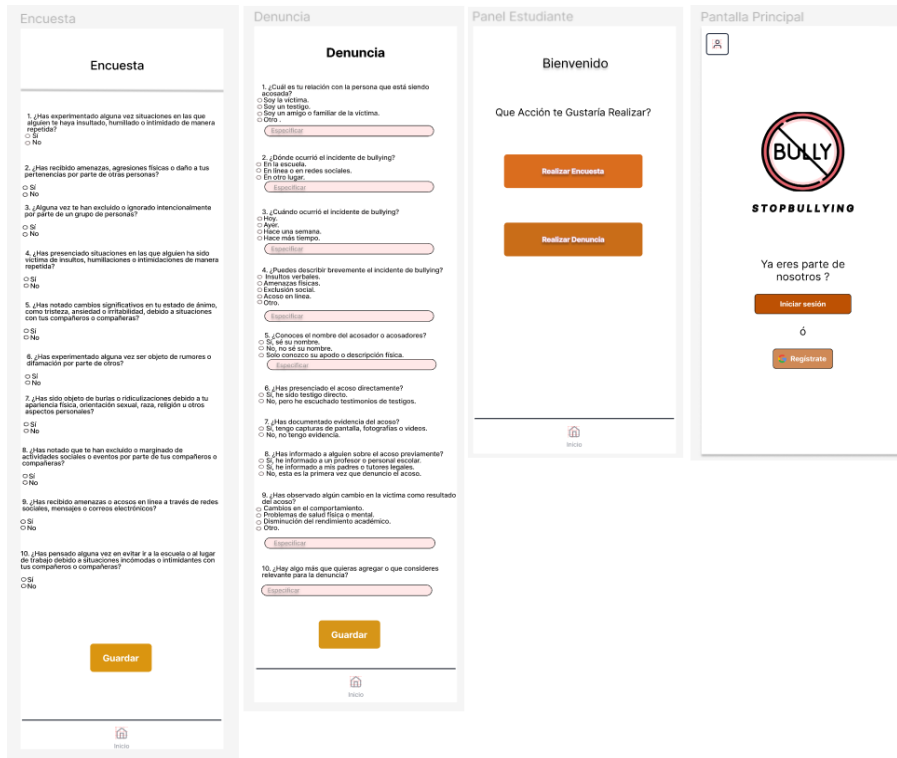


Nota: Pantallas de usuario donde se gestionan los datos, panel y visualización de resultados.

Fuente: Elaboración propia.

Figura 10

Pantallas para interacción del estudiante y login

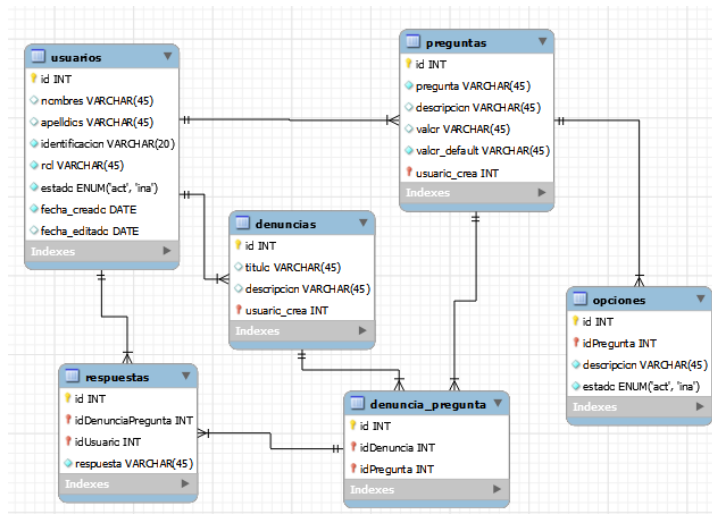


Nota: Modelos de formularios para los estudiantes y panel de navegación. Login de pantalla.

Fuente: Elaboración propia.

Figura 11

Modelo de entidad - relación de las tablas para la base de datos



Fuente: Elaboración propia.

REFERENCIAS

Aglaia. (2022). Comparación de Power BI con QlikView y Tableau | Aglaia Consulting. <https://aglaia.es/blog/power-bi/comparacion-de-power-bi-con-qlikview-y-tableau/>

Crespo, Y. (2019). EL ACOSO ESCOLAR: BULLYING. <https://www.redalyc.org/journal/5350/535059263011/html/>

García García, N. F. (2020). Aplicativo psicológico para mejorar la intervención de bullying social en adolescentes gran unidad escolar José Faustino Sánchez Carrión, Trujillo. Repositorio Institucional - UCV. <https://repositorio.ucv.edu.pe/handle/20.500.12692/49407>

Gonzales Troncos, & Alexander Moises. (2019). Implementación de un sistema web para el registro de casos de bullying en el complejo educativo San Cristo. Repositorio Institucional - UCV. <https://repositorio.ucv.edu.pe/handle/20.500.12692/36688>

JetBrains, & colaboradores. (s. f.). Kotlin for server side. Kotlin Help. Recuperado 12 de julio de 2023, de <https://kotlinlang.org/docs/server-overview.html>

JetBrains. (2022). The State of Developer Ecosystem in 2022 infographic. JetBrains: Developer Tools for Professionals and Teams. <https://www.jetbrains.com/lp/devecosystem-2022/>

Ojeda, J., Macancela, P., Hurtado, G., & Perez, J. (2023, February 24). Desarrollo de entornos virtuales de enseñanza aprendizaje con herramientas web 3.0 y 4.0 para la asignatura de inglés. https://scholar.google.es/citations?view_op=view_citation&hl=es&user=wKcd8q4AAAAJ&citation_for_view=wKcd8q4AAAAJ:kNdYIx-mwKoC

Stack Overflow. (2023). Stack overflow developer survey 2023. Stack Overflow. <https://survey.stackoverflow.co/2023/>

Arribas, I. (2019, January 17). Power BI vs Qlik vs Tableau: ¿Cuál es el mejor software de análisis de datos? - Vandalytic - tu blog de Power BI y más en español. <https://vandalytic.com/power-bi-vs-qlik-vs-tableau-cual-es-el-mejor-software-de-analisis-de-datos/>

Mangabo, O. (2019, noviembre 5). 6 riesgos de seguridad del backend web a considerar en el desarrollo. Geekflare. <https://geekflare.com/es/web-backend-security-risk/>


Borges, S. (2019, noviembre 19). ¿Qué es PostgreSQL? - Para qué sirve, Características e Instalación. Infranetworking. <https://blog.infranetworking.com/servidor-postgresql/>

Aegis Softech. (2020, diciembre 9). Java vs. Python vs. Kotlin: Battle of the best. Aegis Softech. <https://www.aegissofttech.com/articles/java-vs-python-vs-kotlin-battle-of-the-best.html>

Latitude Technolabs. (2020, diciembre 15). Vue.js V/s Next.js: Choosing the right one for web app development. Latitude. <https://latitudetechnolabs.com/vue-js-vs-next-js/>

Muradas, Y. (2021, January 29). Kotlin vs Java, diferencias y ventajas | OpenWebinars. <https://openwebinars.net/blog/kotlin-vs-java/>

Presta, M. (2021, junio 29). El mejor lenguaje para el desarrollo de backend. Back4App Blog; Low-code backend to build modern apps. <https://blog.back4app.com/es/el-mejor-lenguaje-para-el-desarrollo-de-backend/>

Flores, A. (2022, May 17).  ¿Qué es Power BI y para qué sirve? | Guía completa. <https://www.crehana.com/blog/transformacion-digital/que-es-power-bi/>

Ravoof, S. (2022, June 28). MongoDB vs PostgreSQL: 15 diferencias críticas. <https://kinsta.com/es/blog/mongodb-vs-postgresql/>

Wickramasinghe, S. (2022, July 29). Vue vs React: ¿Cuál deberías usar? <https://kinsta.com/es/blog/vue-vs-react/>

De Gregorio, A. (2022, agosto 20). Micronaut vs Spring Boot. Baeldung.com. <https://www.baeldung.com/micronaut-vs-spring-boot>

Tokio School. (2022, octubre 25). Qué es Spring Boot: conoce este framework. Tokio School. <https://www.tokioschool.com/noticias/spring-boot/>


Redacción KeepCoding. (2022, noviembre 2). ¿Qué es MySQL? KeepCoding Bootcamps. <https://keepcoding.io/blog/que-es-mysql/>

Karczewski, D. (2022, diciembre 22). Next.js vs react: Which framework is better for your front-end? Ideamotive.Co. <https://www.ideamotive.co/blog/nextjs-vs-react-which-javascript-framework-is-better-for-your-front-end>

Jesús. (2023, enero 21). Javascript como backend: ¿es buena idea? Tutoriales Dongee. <https://www.dongee.com/tutoriales/javascript-como-backend-es-buena-idea/>

McBride, J. (2023, abril). 🛠️ Kotlin for JavaScript and Python devs: A friendly introduction to a powerful language 🚀. LinkedIn.com. <https://www.linkedin.com/pulse/kotlin-javascript-python-devs-friendly-introduction-powerful-mcbride>

Chaewonkong. (2023, abril 28). Ktor: The next generation framework that might replace spring boot. Medium. <https://medium.com/@chaewonkong/ktor-the-next-generation-framework-that-might-replace-spring-boot-868e8d21fc0f>

Todo el contenido de **LATAM Revista Latinoamericana de Ciencias Sociales y Humanidades**, publicados en este sitio está disponibles bajo Licencia [Creative Commons](https://creativecommons.org/licenses/by/4.0/) .