

Diseño e implementación back-end de un software libre para el monitoreo y análisis de información en bases de datos enfocado a contribuir con el crecimiento de pequeñas y medianas empresas

Jesús David Arias Estupiñán

Trabajo de grado para optar por el título de Ingeniero de Telecomunicaciones

Directora

Yuli Andrea Álvarez Pizarro

Magister en Ingeniería Eléctrica

Universidad Santo Tomás, Bucaramanga

División de Ingenierías y Arquitectura

Ingeniería de Telecomunicaciones

2022

Dedicatoria

Antes que todo le doy gracias a Dios por permitirme desarrollar este hermoso proceso como estudiante de Ingeniería de Telecomunicaciones, dedicar este proyecto y mis logros a mi familia quienes me han dado siempre el apoyo desde un principio para poder seguir adelante a pesar de las dificultades que se presentan en la vida.

En mi camino como estudiante me encontré con grandes docentes y compañeros quienes también contribuyeron a mi crecimiento profesional y personal; mi directora de tesis, la ingeniera Yuli Andrea Álvarez Pizarro y el director de tesis de mis compañeros, el ingeniero Ricardo Andrés Medina Puentes, quiénes estuvieron siempre pendientes del desarrollo del proyecto y gracias a sus conocimientos y su guía, pude dar feliz término a este proyecto junto con mis compañeros de estudio Angie Ríos y Rubén García a quienes también les agradezco por su apoyo brindado y esa gran amistad que se fortaleció en cada uno de los semestres en los que tuvimos la dicha de compartir diversas experiencias.

Finalmente quiero agradecer a toda la facultad de Ingeniería de Telecomunicaciones, el ingeniero Sergio Andrés Zabala decano de la facultad, Carmen Ayala quién fue la secretaria de la facultad y a todo el cuerpo docente, porque allí fue donde desarrollé todas mis competencias a nivel personal y profesional en el ámbito de las telecomunicaciones, sé que todos los conocimientos adquiridos en este tiempo me ayudarán a crecer como profesional integro y una gran persona.

¡Gracias por todo!

Contenido

Introducción	10
1. Diseño e implementación back-end de un software libre para el monitoreo y análisis de información en bases de datos enfocado a contribuir con el crecimiento de pequeñas y medianas empresas.....	12
1.1 Formulación del Problema	12
1.2 Justificación.....	13
1.3 Objetivos	14
1.3.1 Objetivo General.....	14
1.3.2 Objetivos Específicos	14
2. Marco de Referencia.....	15
2.1 Marco Teórico	15
2.2 Marco Conceptual	18
3. Desarrollo del Proyecto	23
3.1 Investigación Inicial	23
3.1.1 Recolección de información y conceptos pertinentes.....	23
3.1.2 Revisión de la documentación oficial.....	26
3.2 Planeación del desarrollo	27
3.2.1 Definición de las políticas de seguridad.....	27
3.3 Desarrollo Back-end.....	28
3.3.1 Desarrollo de APIs para la integración con las bases de datos.....	28
3.3.2 Integrar funcionalidades de análisis estadístico.....	32
3.3.3 Configuración de rutas para servir al Front-end.....	36

3.3.4 Migrar la plataforma a una aplicación híbrida.....	38
3.4 Puesta en operación del proyecto	40
3.4.1 Validación de funcionalidades	40
3.4.2 Despliegue del software.....	46
3.4.3 Liberación del código fuente	46
3.4.4 Manual de usuarios	47
4. Conclusiones.....	49
5. Trabajo Futuro	49
Referencias.....	51

Lista de Figuras

Figura 1. <i>Arquitectura de la aplicación desarrollada</i>	16
Figura 2. <i>Técnicas estadísticas y analíticas más utilizadas por científicos de datos</i>	21
Figura 3. <i>Crecimiento histórico de MySQL</i>	22
Figura 4. <i>Almacenes de datos de documentos</i>	25
Figura 5. <i>Panel de control de XAMPP</i>	28
Figura 6. <i>Creación base de datos en phpMyAdmin</i>	29
Figura 7. <i>Archivo .env con los parámetros de la base de datos local.</i>	29
Figura 8. <i>Instalación de paquete de soporte para MongoDB en proyecto Laravel.</i>	30
Figura 9. <i>Configuración en el archivo “config/database.php”</i>	30
Figura 10. <i>Configuración en el archivo “.env”</i>	31
Figura 11. <i>Configuración de la clase en el archivo “User.php”</i>	31
Figura 12. <i>Interfaz</i>	32
Figura 13. <i>Límite inferior y límite superior</i>	33
Figura 14. <i>Medidas de Tendencia Centra (MTC)</i>	34
Figura 15. <i>Varianza.</i>	35
Figura 16. <i>Desviación estándar.</i>	35
Figura 17. <i>Acceso para funcionalidades de Base de Datos</i>	36
Figura 18. <i>Acceso para funcionalidades de métricas.</i>	37
Figura 19. <i>Acceso para componentes de Dashboard</i>	37
Figura 20. <i>Acceso para funcionalidades de Dashboards para análisis estadístico.</i>	37
Figura 21. <i>Acceso para componentes de Dashboard de análisis estadístico.</i>	37
Figura 22. <i>Propiedades de la ventana.</i>	38

Figura 23. <i>Funcionalidades de la ventana</i>	39
Figura 24. <i>Metric-OS desde la aplicación de Electron.</i>	39
Figura 25. <i>Cambio de lenguaje en la aplicación.</i>	40
Figura 26. <i>Databases</i>	41
Figura 27. <i>Visualización de proyectos.</i>	41
Figura 28. <i>Elección de tipo de gráfico.</i>	42
Figura 29. <i>Consulta de datos.</i>	42
Figura 30. <i>Selección de propiedades del gráfico.</i>	43
Figura 31. <i>Creación de dashboard.</i>	43
Figura 32. <i>Dashboard</i>	44
Figura 33. <i>Controladores del proyecto en Laravel.</i>	44
Figura 34. <i>Local register.</i>	45
Figura 35. <i>Middleware</i>	45
Figura 36. <i>Component – saved – properties.</i>	45
Figura 37. <i>Mozilla Public License 2.0</i>	46
Figura 38. <i>"Getting Started" - Manual de Usuarios.</i>	47
Figura 39. <i>Proceso de Instalación Front-end para desarrolladores.</i>	48
Figura 40. <i>Proceso de Instalación Front-end para usuarios finales.</i>	48
Figura 41. <i>Proceso de Instalación Back-end.</i>	48

Resumen

El presente proyecto abarca el desarrollo back-end para una solución software que prestará los servicios de monitoreo y análisis de información en base de datos para las pequeñas y medianas empresas. La principal motivación para este desarrollo es la brecha existente entre las MiPymes y sistemas que proporcionen un continuo seguimiento a su información que les permita gestionar de manera eficiente sus organizaciones y tener una visión más clara de su sistema a la hora de tomar decisiones que aporten al desarrollo constante de las compañías.

El proyecto se realizó en diversas fases compuestas por distintas actividades que dan cumplimiento a los objetivos planteados inicialmente, objetivos enfocados en la identificación de los principales motores de bases de datos que las empresas utilizan para poder realizar las debidas conexiones con el aplicativo y de esta manera las empresas puedan acceder a su información. La implementación de técnicas estadísticas en la solución software permitirá que las empresas puedan procesar los datos para una correcta interpretación de estos. El software se presenta como una aplicación híbrida, la cual se podrá acceder a ella de manera local como una aplicación de escritorio mediante la migración del proyecto base usando Electron.

El software presenta una licencia Copyleft, ofreciendo un código accesible al público para su uso, preservando siempre la naturaleza de no comercialización del código fuente, pero si su uso libre y modificación por parte de los usuarios que accedan a él.

Un proyecto pensado en el performance de las empresas, para lograr un desarrollo de estas dentro de los diferentes sectores económicos en los que se encuentren desarrollados.

Palabras clave: Back-end, MiPymes, aplicación híbrida, Electron, Copyleft

Abstract

This project covers the back-end development of a software solution that will provide information monitoring and analysis services in a database for small and medium enterprises. The main motivation for this development is the existing gap between MSMEs and systems that provide a continuous monitoring of their information that allows them to efficiently manage their organizations and have a clearer vision of their system when making decisions that contribute to the constant development of the companies.

The project was carried out in different phases composed by different activities that give fulfillment to the initially proposed objectives, objectives focused on the identification of the main database engines that the companies use to be able to make the proper connections with the application and in this way the companies can access their information. The implementation of statistical techniques in the software solution will allow the companies to process the data for a correct interpretation of these. The software is presented as a hybrid application, which can be accessed locally as a desktop application by migrating the base project using Electron.

The software presents a Copyleft license, offering a code accessible to the public for its use, always preserving the non-commercial nature of the source code, but its free use and modification by the users who access it.

A project designed for the performance of companies, to achieve a development of these within the different economic sectors in which they are developed.

Keywords: Back-end, MiPymes, hybrid application, Electron, Copyleft

Glosario

Back-end: Esta sección en el desarrollo web hace referencia a la parte que el usuario no puede ver, pero que cumple la función de obtener la información a través de la aplicación para de esta manera ser procesada y devolver al usuario final un resultado.

MiPymes: El término MiPymes refiere a las micro, pequeñas y medianas empresas.

Aplicación híbrida: Una aplicación híbrida es una aplicación software que permite la integración de componentes de una aplicación nativa y una aplicación web, lo cual permite ejecutar el aplicativo desde un entorno local.

Electron: Es un framework que permite la creación de aplicaciones de escritorio usando JavaScript, HTML y CSS.

Copyleft: Es una licencia que permite liberar un software para su copia, modificación y redistribución, pero siempre conservando la naturaleza de código abierto.

Introducción

Este trabajo de grado presenta el desarrollo de una solución software a nivel de back-end donde se pretende abarcar ciertas fisuras presentes en la gestión y manipulación de datos en las pequeñas y medianas empresas. Si bien es cierto, la administración de información y recolección de datos en el ámbito de la prestación de un servicio conlleva en gran medida una principal responsabilidad frente a la toma de decisiones dentro de una compañía.

La implementación de soluciones como lo son “Dashboards” que permiten la visualización y gestión de datos recolectados por una empresa, son acciones que a través de los últimos años las empresas han tomado con mucha fuerza y de esta manera lograr minimizar la carga que se genera en la manipulación de forma manual de diversa información proporcionada por los estudios. Se presentan en la actualidad diversas soluciones software que permiten una mayor gestión de la información, una gestión que permita llevar a la toma de decisiones en la organización de una manera acertada ya que estas soluciones cuentan con un proceso de recolección, análisis y posterior predicción de los datos suministrados por los usuarios; si bien es cierto que ante las organizaciones, estas soluciones se encuentran muy llamativas, se debe tener en cuenta que su costo es altamente elevado y en ocasiones las empresas no logran la adquisición de esos sistemas.

La búsqueda de información relacionada con la visualización, estudio y predicción de datos llevó al desarrollo de una solución de software libre al alcance de las organizaciones, principalmente en las pequeñas y medianas empresas. Brindar al sector empresarial una herramienta capaz de almacenar la información suministrada, visualizarla de diversas formas teniendo en cuenta a los distintos tipos de gráficos de datos utilizados por las organizaciones, y de igual manera, permita un posterior análisis de los datos para proporcionar a los usuarios una

predicción de estos con el fin de contribuir a una mejora organizacional gracias a la toma de decisiones de una manera más acertada según los estudios previos.

Su estructuración como software libre permitirá que los usuarios con conocimiento de desarrollo en programación puedan establecer mejoras al software y de esta manera sea un software adaptable a las necesidades de cada una de las organizaciones quienes hagan uso de esta solución. Así como también la licencia Copyleft permitirá el ejercicio del derecho de autor brindando un libre uso y distribución del software siempre y cuando los usuarios sigan siempre preservando las mismas condiciones al distribuir posibles copias.

1. Diseño e implementación back-end de un software libre para el monitoreo y análisis de información en bases de datos enfocado a contribuir con el crecimiento de pequeñas y medianas empresas

1.1 Formulación del Problema

El sector empresarial colombiano está constituido en su mayoría por las pequeñas y medianas empresas, conocidas comúnmente como MiPymes [1], la recolección y administración de la información que se genera diariamente representa un factor primordial para la organización y toma de decisiones de la empresa, sin embargo, aunque la mayoría de MiPymes conoce las ventajas de implementar una base de datos, una (1) de cada dos (2) empresas, no cuenta actualmente con una, siendo el desconocimiento del tema y la falta de dinero, dos de las razones por las cuales no se toma la decisión administrativa de implementarlas [2], generando de esta forma, una carga relevante de trabajo tanto para la administración de documentos contables, como para su debido análisis financiero en donde el manejo de la información se ha convertido en el factor principal de la planeación para el crecimiento de una empresa.

El desarrollo de este tipo de software se basa en tres aspectos fundamentales:

- Conexión con una base de datos compatible, en un entorno local o en la nube.
- Consumo de información mediante consultas generadas por el usuario.
- Asignación de la información a gráficos para su visualización en un dashboard.

Dependiendo del desarrollador, se pueden encontrar funcionalidades extras que generan un valor agregado a este tipo de software. Estas son (1) el almacenamiento de múltiples dashboards para la organización de la información. Y (2) la operación en la nube para evitar costos adicionales de instalación física.

1.2 Justificación

Las soluciones de software comerciales cuyas funcionalidades incluyan administrar y analizar la información de forma eficiente para tomar decisiones acertadas en el menor tiempo posible son costosas y no están pensadas en ser adaptables; el software libre es una buena alternativa para las pequeñas empresas ya que está adaptado a necesidades específicas y es más económico. Además, el hecho de que sean libres les permite modificarlas según sus requerimientos, lo que les da una mayor flexibilidad para añadir funcionalidades.

El análisis predictivo usa una variedad de técnicas estadísticas, de minería de datos, modelado y aprendizaje automático para analizar datos históricos y actuales con el fin de predecir acontecimientos futuros, identificando patrones en los datos, el análisis puede estimar riesgos y oportunidades en el flujo de actividades de la organización.

El desarrollo back-end de un software libre para la monitorización y predicción de datos tiene como objetivo generar un sistema de uso y modificación libre para el monitoreo y análisis de información obtenida en bases de datos relacionales y no relacionales, mediante gráficos y consultas generados a través de la interfaz de usuario con el fin de facilitar la interacción y la curva de aprendizaje para su uso en pequeñas y medianas empresas orientado a mejorar su crecimiento y planificación estratégica.

1.3 Objetivos

1.3.1 Objetivo General

Desarrollar un software de uso y modificación libre para el monitoreo y análisis de información obtenida en bases de datos relacionales y no relacionales, mediante gráficos y estadísticas generadas a través de la interfaz de usuario con el fin de facilitar la interacción y la curva de aprendizaje para su uso en pequeñas y medianas empresas orientado a mejorar su crecimiento y planificación estratégica.

1.3.2 Objetivos Específicos

- Identificar los estándares de Laravel correspondientes para la administración de base de datos relacionales y no relacionales que permitan el reconocimiento de los tipos de datos suministrados por las empresas para su posterior análisis.
- Implementar técnicas estadísticas que permitan procesar los datos suministrados por las empresas para su debida interpretación prediciendo así el comportamiento de las MiPymes para que sean empleados por las mismas para sus procesos de mejora.
- Generar una aplicación híbrida usando Electron PHP mediante la migración del software obteniendo una aplicación nativa de escritorio y de esta forma el usuario cuente con su uso sin la necesidad de una conexión a internet.
- Registrar el aplicativo en el repositorio de GitHub y la página web oficial bajo licencia CopyLeft ofreciendo un código que sea accesible al público para su uso.
- Definir el manual de usuarios exponiendo de manera intuitiva las funcionalidades del software que facilite el uso del aplicativo a las empresas.

2. Marco de Referencia

2.1 Marco Teórico

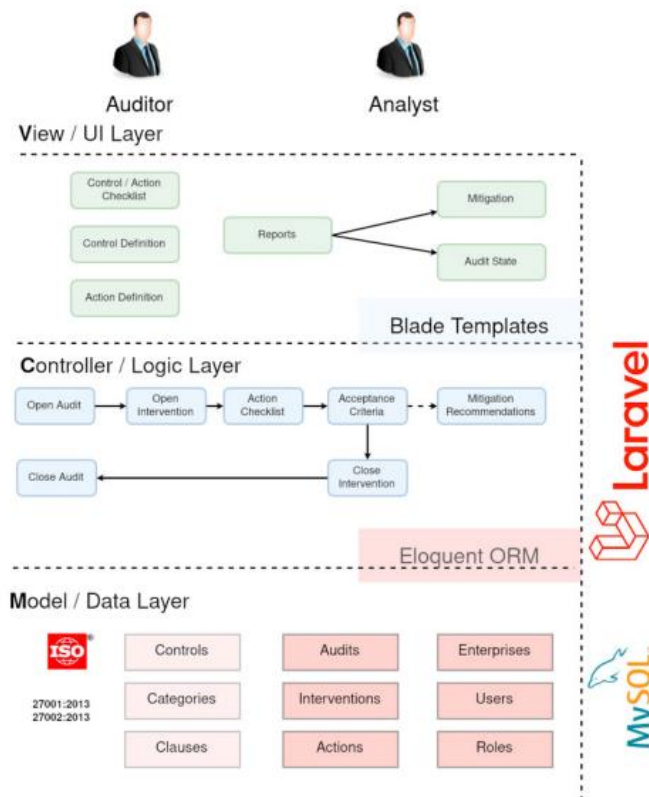
Las soluciones de código abierto para el análisis de datos, y la extracción de métricas en un entorno empresarial, brindan resultados de alto impacto en las empresas; lo cual, es indispensable para evaluar y controlar en el área de trabajo. En el marco del análisis de datos, la estadística reúne la información cuantitativa para organizarla, presentarla, analizar e interpretar los datos numéricos con el fin de realizar una toma de decisiones [3]. En [4] se presenta a los métodos estadísticos como de uno de los componentes del análisis de datos. Se expone el framework EDISON como una solución para el diseño de planes de estudio eficaces en materia de ciencia de los datos [4].

El uso de modelos estadísticos para los procesos de análisis de datos varía dependiendo del uso de datos de cada una de las empresas, sin embargo, [5] nos presenta un método de modelaje basado en 5 fases; la verificación de los datos, selección de variables, análisis preliminar, el análisis de regresión y la verificación del modelo. Presentando como resultado un modelo con una técnica estadística donde supone que todas las variables tienen una distribución normal, para de esta forma poder aplicar que el método de análisis.

Para el desarrollo del aplicativo web se implementa el framework Laravel, el cuál es uno de los frameworks creados para ayudar a los desarrolladores a crear una web haciendo hincapié en la simplicidad y flexibilidad del diseño, tal como lo presenta [6] en donde se desarrolla un sistema de seguimiento de prácticas utilizando Laravel para facilitar el proceso de realización del programa. Por otra parte, una plataforma web personalizable para gestionar el cumplimiento de las normas de seguridad de la información y la auditoría de ciberseguridad que presenta [7] nos

muestra cómo se encuentra desarrollada la aplicación con Laravel y la gestión de base de datos se realiza con el gestor MySQL; para comprender mejor el funcionamiento que presentan en el proyecto se presenta en la figura 1 el proceso de recopilación de pruebas realizada por el equipo consultor, el procesamiento automático de los informes de auditoría y el análisis de los resultados.

Figura 1. *Arquitectura de la aplicación desarrollada*



Tomado de [7]

La aplicación web que se desarrolla en el proyecto mencionado anteriormente, es versátil y personalizable, ya que recibe una lista predefinida de acciones y una lista de correcciones que se deben aplicar en el desarrollo de las actividades.

EL manejo de base de datos implica un cuidado en la manipulación de la información que las empresas suministran en la aplicación web, es por esto, que se implementan gestores de base de datos como MySQL para la gestión de base de datos relacionales, el cual utiliza por defecto

conexiones no cifradas entre el cliente y el servidor [8]. La información que se recolecte de cualquier tipo de fuente puede ser gestionada de forma sencilla en estas bases de datos como lo presenta [9], donde toda la información recolectada se almacena en una base de datos MySQL de un servidor web y se gestiona mediante phpMyAdmin. Esto permite que los usuarios accedan a la información a través de una pantalla de ordenador.

Por su parte, MongoDB para la gestión de base de datos no relacionales, permite representar relaciones, pero de una manera más intuitiva y flexible con documentos estructurados. Otra ventaja que tiene MongoDB es que está construido para ser eficiente, escalable y redundante en su núcleo [10].

Debido a esto, MongoDB ha venido incrementando su uso debido a que los datos se almacenan en estructuras parecidas a un JSON, esto hace que el flujo de datos dentro de la aplicación no tenga mayores cambios en la estructura de datos como lo presenta [11] donde se desarrolla un sistema de asignación de exámenes universitarios usando MongoDB y Python con el fin de asignar salas, deberes y fechas a estudiantes y facultades de acuerdo con las restricciones dadas.

El auge de los sistemas de control de versiones distribuidos, como git, y de las plataformas construidas sobre ellos, como GitHub, ha provocado un cambio en la forma de desarrollar el software [12]. GitHub y GitLab han simplificado enormemente el proceso de desarrollo de software, ya que permiten que los desarrolladores trabajen de forma colaborativa en un entorno unificado. Además, estas plataformas ofrecen una serie de funciones adicionales, como herramientas de revisión de código y un soporte transversal para la integración y el despliegue continuos.

2.2 Marco Conceptual

El desarrollo de un aplicativo software libre implica el hecho de respetar la libertad de los usuarios y la comunidad [13]; es decir, los usuarios tienen la libertad de ejecutar, copiar, modificar y mejorar el software. Bajo el marco de software libre, el proyecto se encuentra con la autorización de una licencia copyleft con el fin de que el software pueda ser utilizado y modificado por un gran número de usuarios con la condición de que mantenga su naturaleza de software libre, esto quiere decir, que no podrá ser comercializado dicho software. El software se considera libre si los usuarios cuentan con ciertos principios de libertad. La libertad de ejecutar el programa como se desee, con cualquier propósito; la libertad de estudiar cómo funciona el programa, y cambiarlo para que el usuario realice lo que desee con él, por otro lado, tendrá la libertad de redistribuir copias para ayudar a otros y, por último, el usuario tendrá la libertad de distribuir copias de sus versiones modificadas a terceros.

En un desarrollo software, backend hace referencia al área lógica de una página web que se conecta con la base de datos y el servidor que utiliza dicho sitio web. Backend es la capa de acceso a datos de un software o cualquier dispositivo, que no es directamente accesible por los usuarios [14]. Por otra parte, esta sección contiene la lógica de la aplicación que maneja los datos mencionados anteriormente. En cuanto a conexión con el servidor que no es más sino la aplicación especializada que comprende la naturaleza con la que el navegador web realiza solicitudes, el backend tiene el acceso a este servidor.

Con la intención de administrar el almacenamiento de los datos brindados por las empresas y acceder a los mismos para de estar forma presentar de forma visual en el aplicativo para que los usuarios puedan consumirlos en cualquier dispositivo, se implementan lenguajes de programación tales como Java, C ++, Ruby, PHP o Python, los cuales permiten emplear diversos frameworks de

desarrollo como Django, Laravel, Java EE, ASP.NET; todos ellos presentes para el desarrollo back-end.

En este orden de ideas, para el presente proyecto se emplea el framework de desarrollo back-end, Laravel. Laravel es uno de los frameworks PHP de código abierto más populares en el mundo de la programación, gracias a la sencillez en la sintaxis de su código y su facilidad para desarrollar proyectos de forma rápida y eficaz [15]. La implementación de Laravel trae consigo una variedad de funcionalidades, algunas de ellas ya mencionadas anteriormente, como lo es una sintaxis muy sencilla que permite obtener una curva de aprendizaje rápida. Su arquitectura está en el marco MVC (Modelo-Vista-Controlador), la cuál es un estilo de arquitectura que separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos [16]; el modelo contiene una representación de los datos que maneja el sistema, la vista o la interfaz de usuarios compone la información que se envía al cliente y, por último, el controlador, que actúa como intermediario entre el modelo y la vista.

Continuando con las funcionalidades que nos presenta Laravel, este framework cuenta con un sistema de testing automático, que permite validar la funcionalidad del servicio web que se esté desarrollando en el momento y de esta forma verificar que todo funcione como debe ser.

Para este proyecto, se contará con la implementación de una versión de Laravel estable que permita la mejora en el sistema de rutas, sistema de plantillas y maquetación del sistema.

Se decide implementar este framework de desarrollo back-end gracias a su integración a otros sistemas y bibliotecas, también por su escalabilidad y su facilidad en el mantenimiento del código fuente.

La estructura del proyecto abarca otros apartados que complementan en sí la naturaleza de este, se han mencionado conceptos relacionados con el desarrollo del código fuente que permita

la conexión entre aplicativo y servidor, la inteligencia del sistema en sí y sus demás funcionalidades, ahora bien, el análisis de los datos se realiza bajo el marco de ciencia de datos.

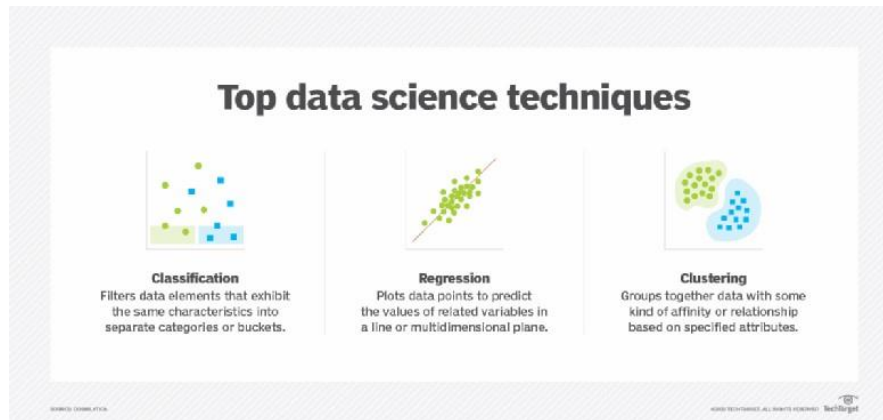
La ciencia de datos es la implementación de técnicas avanzadas y principios estadísticos para extraer información valiosa de los datos para la toma de decisiones comerciales, la planificación estratégica y otros usos [17]. Se trata de una aplicación multidisciplinaria que incorpora la ingeniería de datos, la preparación de los mismo, el análisis predictivo mediante técnicas básicas de estadística, matemática y adicionalmente, la programación de software.

La ciencia de datos obtiene un papel importante en varios aspectos de las operaciones y estrategias que realizan las empresas a nivel comercial. Esta ciencia proporciona información sobre los clientes que ayuda a las empresas a crear acciones de mejora más sólidas frente a temas de marketing y publicidad dirigida a cierto tipo de población, todo esto, con la finalidad de aumentar las ventas de aquellos productos que las empresas puedan comercializar. Todo esto se realiza bajo un análisis informado del comportamiento del cliente, las tendencias del mercado y la competencia [18]. En efecto esta es la finalidad de la implementación de la ciencia de datos en las empresas y así mismo es la intención del desarrollo de este proyecto.

El proceso que se realiza bajo el concepto de ciencia de datos consta de seis pasos, los cuales define Donald Farmer, director de la consultora analítica TreeHive Strategy en un artículo que describe el proceso de ciencia de datos [19], empezando por la identificación de una hipótesis relacionada con el negocio para probar, continuando con la recolección de datos y preparación para su análisis, luego se experimenta con diferentes modelos analíticos para posteriormente elegir el más adecuado y ejecutarlo con los datos; habiendo realizado esto, se presentan los resultados a los encargados de su revisión y a continuación se implementa el modelo para un uso continuo con nuevos datos que suministre la empresa.

A continuación, se presentan algunas de las técnicas estadísticas más utilizadas por los científicos de datos.

Figura 2. *Técnicas estadísticas y analíticas más utilizadas por científicos de datos*



Tomado de [18]

En consecuencia, lo expuesto anteriormente conlleva la implementación de un sistema de monitorización de datos. Estos sistemas ofrecen información del contexto en el que se implementan, permitiendo su posterior análisis para efectuar operaciones diversas, desde el diseño de estrategias hasta la toma de decisiones [20]. Los sistemas de monitorización permiten a las empresas obtener una mejor visión de su negocio a través de la recolección, análisis y monitoreo de datos. Gracias a esta información, las empresas pueden tomar mejores decisiones y mejorar su rendimiento.

Con respecto al sistema de gestión de base de datos relacionales, se implementa MySQL, que cuenta con una doble licencia, por una parte, es de código abierto, pero por otra, cuenta con una versión comercial gestionada por la compañía Oracle [21]. Así mismo es importante mencionar que MySQL es el sistema de gestión de base de datos relacionales más entendido en la actualidad.

Figura 3. *Crecimiento histórico de MySQL*

Tomado de [21]

La implementación de base de datos no relacionales en el proyecto trae la necesidad de implementar un gestor de base de datos adecuado, para el cuál se opta por MongoDB es cuál es una base de datos de documentos que ofrece una escalabilidad y flexibilidad, y un modelo de consultas e indexado avanzado [22]. MongoDB permite almacenar datos en documentos flexibles similares a JSON, por lo que los campos pueden variar entre documentos y la estructura de datos puede cambiarse con el tiempo. Cabe mencionar que MongoDB es de uso gratuito.

Se han mencionados conceptos relacionados con el desarrollo del software, la implementación de técnicas para el análisis y monitorización de datos y la gestión de los datos suministrados por las empresas; ahora con el objetivo de proporcionar el código fuente al público se implementa GitHub.

GitHub es una plataforma de alojamiento, propiedad de Microsoft, que ofrece a los desarrolladores la posibilidad de crear repositorios de código y guardarlos en la nube de forma segura, usando un sistema de control de versiones, llamado Git [23]. GitHub facilita la organización de proyectos y permite la colaboración de varios desarrolladores en tiempo real.

Existen otras plataformas que nos permiten el alojamiento de código como lo son GitLab o Bitbucket, pero hay ciertas razones por las que GitHub es una gran opción para el control y gestión de proyectos de código; el alojamiento de proyectos en repositorios de forma gratuita, la reducción significativamente de los errores que se pueden cometer a nivel humano, son algunas de estas razones por las que se ha elegido GitHub para el control de versiones y gestión del desarrollo del código fuente del proyecto.

De esta forma se presentan los elementos principales para la elaboración de este proyecto y algunos conceptos que serán abarcados en el desarrollo de las actividades propuestas.

3. Desarrollo del Proyecto

3.1 Investigación Inicial

3.1.1 Recolección de información y conceptos pertinentes

Teniendo en cuenta la necesidad de integrar diversas bases de datos a la solución software y de igual manera comprendiendo que las distintas organizaciones recurren a diferentes tipos de bases de datos para el almacenamiento y gestión de su información, se realizó un proceso de investigación partiendo de dar respuesta a lo que se comprende como bases de datos relacionales y bases de datos no relacionales. Teniendo como punto de partida, que una base de datos relacional es un tipo de base de datos que almacena y proporciona acceso a puntos de datos relacionados entre sí [24]. Que una base de datos sea relacional implica que su estructura (compuesta por tablas de datos, vistas, entre otros elementos) se encuentra separada de una estructura de almacenamiento físico; es decir, esta separación permite que los usuarios gestionen el almacenamiento físico de

datos sin que repercuta alguna acción en el acceso de estos datos pero esta vez como una estructura lógica, por ejemplo, el cambio de nombre de cualquier archivo en un base de datos no implica que las tablas que se encuentren en este archivo vayan a cambiar sus nombres.

El uso de un modelo relacional en la gestión de base de datos por lo general se lleva a cabo en el rastreo de inventarios, transacciones de comercio electrónico, así como también la administración de cantidades enormes de información de clientes; las bases de datos relacionales proporcionan una gestión de forma segura de la información y es ampliamente usada por organizaciones de cualquier tipo y de diversos tamaños.

Para la gestión de base de datos relacionales, se encuentra diversos motores que permiten el almacenamiento y recuperación de la información. En las organizaciones, se implementan motores de bases de datos relacionales tales como ORACLE el cuál es el motor relacional más antiguo, el motor más usado en las compañías se conoce como MySQL, a raíz de MySQL se genera MariaDB el cuál es compatible con MySQL para poder cambiar un motor por otro, también se encuentra PostgreSQL inspirado en Oracle. Estos son algunos de los motores de bases de datos relacionales que más se usan en la gestión de información con una estructura relacional.

Ahora bien, del otro lado encontramos las bases de datos no relaciones o NoSQL; estas están diseñadas específicamente para modelos de datos específicos y tienen esquemas flexibles para crear aplicaciones modernas [25]. El uso de base de datos NoSQL encaja muy bien en el desarrollo de aplicaciones modernas, juegos, entre otras aplicaciones que requieren de bases de datos flexibles, escalables, de alto rendimiento y altamente funcionales para proporcionar excelentes experiencias de usuario [25]. El almacenamiento de datos en una base de datos no relacional se lleva a cabo mediante documentos JSON o también se puede llevar como un grafo que se encuentra compuesto de bordes y vértices.

La siguiente imagen presenta un ejemplo de un almacenamiento de datos de documentos en forma de documento JSON.

Figura 4. Almacenes de datos de documentos

Key	Document
1001	<pre>{ "CustomerID": 99, "OrderItems": [{ "ProductID": 2010, "Quantity": 2, "Cost": 520 }, { "ProductID": 4365, "Quantity": 1, "Cost": 18 }], "OrderDate": "04/01/2017" }</pre>
1002	<pre>{ "CustomerID": 220, "OrderItems": [{ "ProductID": 1285, "Quantity": 1, "Cost": 120 }], "OrderDate": "05/08/2017" }</pre>

Tomado de [26]

Si bien es cierto que en los motores de base de datos relacionales encontramos variedad a la hora de elegir un gesto de datos, en los modelos no relacionales o NoSQL encontramos algunos motores como lo son Cassandra y MongoDB quienes son los motores más usados por las compañías a la hora de manipular bases de datos no relacionales.

La implementación de cada uno de estos modelos va a depender principalmente del volumen de datos, es decir, cuando se comprende que el volumen de los datos no tiene a aumentar o lo hace poco a poco, es adecuado usar un modelo de base de datos relacional, en cambio, cuando

el volumen de los datos crece muy rápidamente en momentos puntuales, se recomienda implementar un modelo de base de datos NoSQL o no relacional.

3.1.2 Revisión de la documentación oficial

Para el desarrollo del proyecto se tomaron en cuenta los motores de bases de datos MySQL como gestor de base de datos relacional y MongoDB como gestor de base de datos no relacional.

El proceso de conexión e integración de Laravel con las bases de datos MySQL consta inicialmente de configurar la base de datos en la aplicación Laravel en la ubicación “config/database.php”. En este archivo, se definen todas las conexiones de la base de datos, así como también se especifica qué conexión debe usarse de forma predeterminada [27].

Para el caso de nuestro gestor de base de datos MongoDB se debe primero instalar el paquete Laravel MongoDB en nuestro proyecto de Laravel, esto con el fin de instalar la extensión PHP para MongoDB y de esta manera poder instalar posteriormente las bibliotecas MongoDB para Laravel [28]. La revisión de la documentación oficial de MongoDB proporciona un paso a paso para poder realizar la conexión de una base de datos MongoDB con nuestro proyecto de Laravel, la especificación de cada uno de estos procesos se verá de manera detallada en el desarrollo de la fase 3 la cual presenta el desarrollo de la integración con las bases de datos dentro del desarrollo Back-end.

3.2 Planeación del desarrollo

3.2.1 Definición de las políticas de seguridad

Ya que una brecha de seguridad de datos representa un error al mantener la confidencialidad de los datos en una base de datos [29], esto podría implicar daños en diferentes factores dentro de una compañía, factores como la propiedad intelectual que se vea comprometida en la información suministrada, daños a la reputación de la marca, los costos de reparación de brechas, entre otros factores.

Estas brechas de seguridad pueden estar dadas por varias razones, tales como una mala configuración de software, vulnerabilidades o patrones de descuido [29]. Algunas de las amenazas más comunes que ponen en riesgo una base de datos pueden estar definidas como amenazas internas que no es más sino amenazas de seguridad donde usuarios internos de la compañía tengan la intención de hacer daño, informantes que producen errores para poder vulnerar a las bases de datos. Así como el error humano también puede ser un factor para que se produzca una brecha de seguridad, los ataques de denegación de servicio (DDos) también son muy comunes en donde se engaña al servidor de destino, que para nuestro caso puntual hace referencia al servidor de bases de datos.

Teniendo en cuenta a este contexto sobre la seguridad en base de datos, para el proyecto se definieron algunas políticas de seguridad con la finalidad de prevenir estas brechas de seguridad. Políticas de seguridad basadas en la conexión con la base de datos, en donde se define que la base de datos no se debe declarar manualmente en el código fuente, sino que se solicita que la contraseña que el usuario asigne al registrar la base de datos esté encriptada en todo momento.

3.3 Desarrollo Back-end

3.3.1 Desarrollo de APIs para la integración con las bases de datos

Esta actividad se llevó a cabo en dos etapas, la primera consistió en realizar la conexión de una base de datos local MySQL con el proyecto de Laravel. Para esto, se realizó el siguiente proceso:

1. Para poder crear nuestra base de datos de prueba se inició el servidor Apache y MySQL desde el panel de control de XAMPP.

Figura 5. Panel de control de XAMPP



2. Posterior a ello, en el navegador se ingresó a phpMyAdmin mediante la siguiente dirección en la URL “localhost/phpmyadmin”.
3. Allí en phpMyAdmin creamos nuestra base de datos de prueba.

Figura 6. Creación base de datos en phpMyAdmin

- Una vez creada la base de datos de manera local, en nuestro proyecto de Laravel nos ubicamos en el archivo “**.env**”, allí se ingresan los datos de nuestra base de datos, tales como el nombre, username y password, siempre y cuando se tengan configuradas.

Figura 7. Archivo `.env` con los parámetros de la base de datos local.

```
.env
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8
9 DB_CONNECTION=mysql
10 DB_HOST=127.0.0.1
11 DB_PORT=3306
12 DB_DATABASE=MetricOS
13 DB_USERNAME=root
14 DB_PASSWORD=
```

- De esta manera se realiza la conexión de bases de datos locales MySQL con el software que como ya se ha mencionado, ha sido desarrollado en el framework Laravel.

La segunda etapa de esta actividad consistió en la conexión de una base de datos MongoDB, para ello se realizó el siguiente procedimiento:

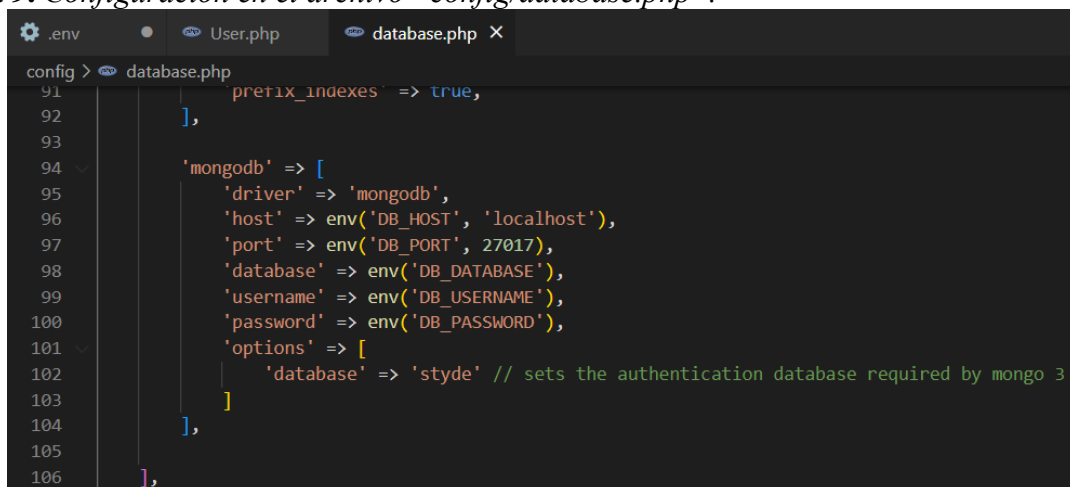
1. En el proyecto de Laravel se tiene que instalar el paquete que genera soporte a MongoDB, este paquete se instaló con el siguiente comando dentro de la terminal de nuestro entorno de desarrollo.

Figura 8. *Instalación de paquete de soporte para MongoDB en proyecto Laravel.*

```
PS C:\xampp\htdocs\MetricOS> composer require jenssegers/mongodb
Using version ^3.9 for jenssegers/mongodb
./composer.json has been updated
Running composer update jenssegers/mongodb
Loading composer repositories with package information
Updating dependencies
```

2. Una vez instalado el soporte de MongoDB, realizamos la configuración en el archivo “config/database.php” en donde se agregó un segmento de código en el arreglo “connections”.

Figura 9. *Configuración en el archivo “config/database.php”.*



```
config > database.php
91         'prefix_indexes' => true,
92     ],
93
94     'mongodb' => [
95         'driver' => 'mongodb',
96         'host' => env('DB_HOST', 'localhost'),
97         'port' => env('DB_PORT', 27017),
98         'database' => env('DB_DATABASE'),
99         'username' => env('DB_USERNAME'),
100        'password' => env('DB_PASSWORD'),
101        'options' => [
102            'database' => 'styre' // sets the authentication database required by mongo 3
103        ]
104    ],
105
106 ],
```

3. Como en la sección de código se hace uso de la función de ayuda “env” la cual nos provee Laravel con la finalidad de buscar coincidencias establecidas en el archivo “.env”.

Figura 10. Configuración en el archivo “.env”

```
9   DB_CONNECTION=mongodb
10  DB_HOST=127.0.0.1
11  DB_PORT=27017
12  DB_DATABASE=MetricOS
13  DB_USERNAME=root
14  DB_PASSWORD=
```

4. Para finalizar se hizo el cambio de la clase que extiende cada modelo dentro de nuestro proyecto de Laravel ya que el paquete que se mencionó en el inciso A de este proceso, extiende clases del ORM Eloquent. Esto se realizó de la siguiente manera, en el archivo “User.php”.

Figura 11. Configuración de la clase en el archivo “User.php”.

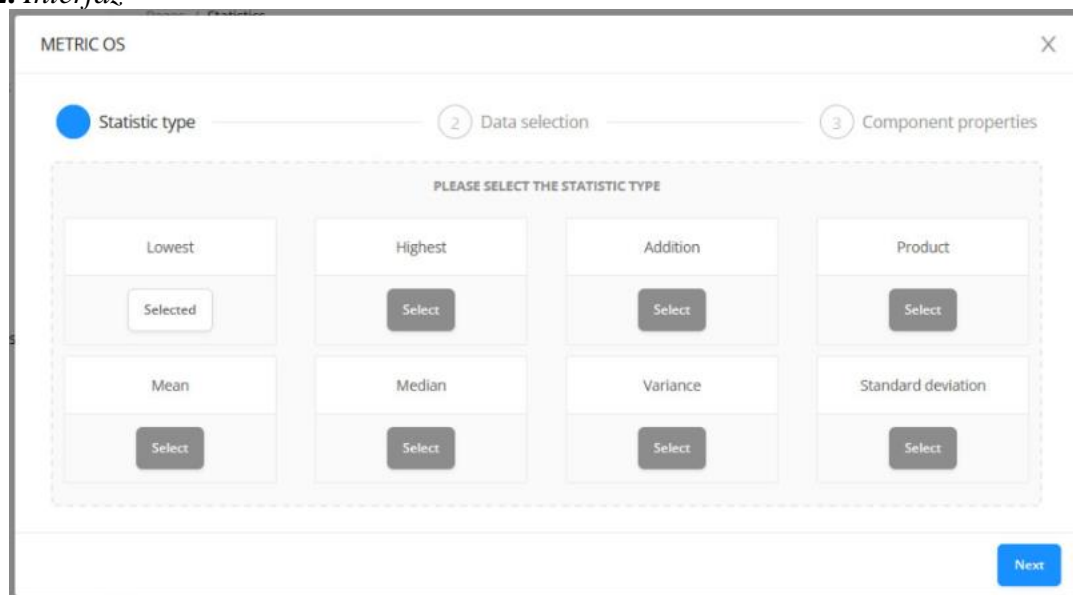
```
.env User.php x database.php
app > User.php > ...
1  <?php
2
3  namespace App;
4
5  use Illuminate\Contracts\Auth\MustVerifyEmail;
6  use Illuminate\Foundation\Auth\User as Authenticatable;
7  use Illuminate\Notifications\Notifiable;
8  use Jenssegers\Mongodb\Eloquent\Model as MongoModel;
9
10 class User extends MongoModel
11 {
12     use Notifiable;
13
14     /**
15      * The attributes that are mass assignable.
16      *
17      * @var array
18      */
19     protected $fillable = [
20         'name', 'email', 'password',
21     ];
```

De esta manera se lograron realizar las conexiones de nuestro proyecto en Laravel con cada uno de los gestores de bases de datos, así, cuando el usuario haga uso de la solución software, podrá realizar la conexión con una base de datos local alojada en MySQL o en MongoDB.

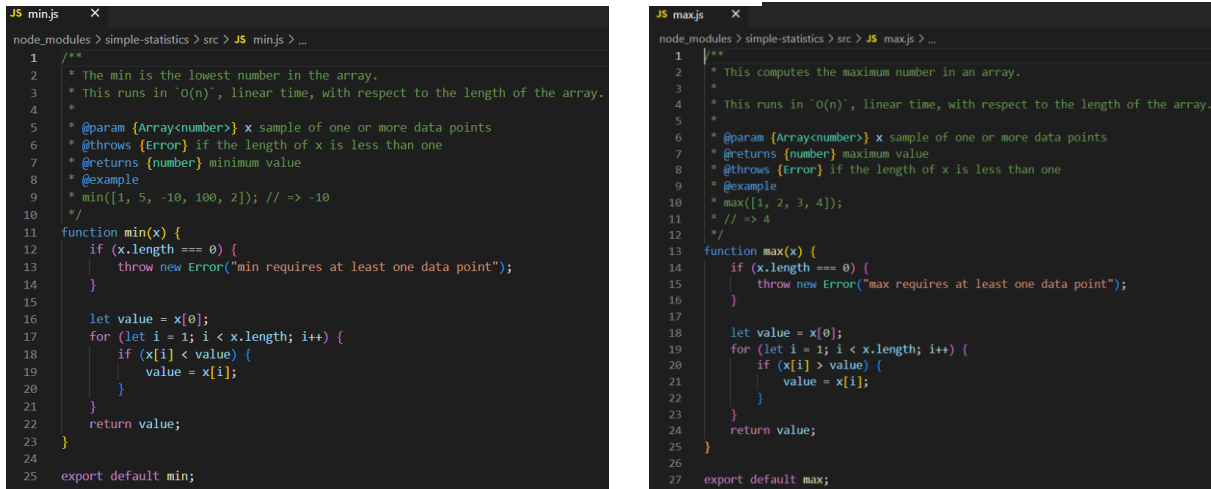
3.3.2 Integrar funcionalidades de análisis estadístico

El proyecto en principio presenta algunas funciones estadísticas para que el usuario pueda seleccionar la indicada para el análisis que desee obtener de la información que suministran. Estas funciones se desarrollaron a nivel de front-end como un proyecto en lenguaje Java Script para que el usuario realizara la selección de las diversas funciones, obteniendo como resultado final la siguiente interfaz:

Figura 12. Interfaz



Se integraron funciones como el valor más bajo y el valor más alto para poder identificar mínimos y máximos en los datos proporcionados por los usuarios, así como también la suma de datos o la serie que sea ingresada.

Figura 13. Límite inferior y límite superior

```
JS min.js X
node_modules > simple-statistics > src > JS min.js > ...
1  /**
2  * The min is the lowest number in the array.
3  * This runs in `O(n)`, linear time, with respect to the length of the array.
4  *
5  * @param {Array<number>} x sample of one or more data points
6  * @throws {Error} if the length of x is less than one
7  * @returns {number} minimum value
8  * @example
9  * min([1, 5, -10, 100, 2]); // => -10
10 */
11 function min(x) {
12   if (x.length === 0) {
13     throw new Error("min requires at least one data point");
14   }
15
16   let value = x[0];
17   for (let i = 1; i < x.length; i++) {
18     if (x[i] < value) {
19       value = x[i];
20     }
21   }
22   return value;
23 }
24
25 export default min;
```

```
JS max.js X
node_modules > simple-statistics > src > JS max.js > ...
1  /**
2  * This computes the maximum number in an array.
3  * This runs in `O(n)`, linear time, with respect to the length of the array.
4  *
5  * @param {Array<number>} x sample of one or more data points
6  * @returns {number} maximum value
7  * @throws {Error} if the length of x is less than one
8  * @example
9  * max([1, 2, 3, 4]);
10 * // => 4
11 */
12
13 function max(x) {
14   if (x.length === 0) {
15     throw new Error("max requires at least one data point");
16   }
17
18   let value = x[0];
19   for (let i = 1; i < x.length; i++) {
20     if (x[i] > value) {
21       value = x[i];
22     }
23   }
24   return value;
25 }
26
27 export default max;
```

Tomado de [30]

Se implementaron Medidas de Tendencia Central (MTC) para que los usuarios puedan identificar cuáles son las tendencias en ciertos conjuntos de datos. La media para poder identificar el valor promedio de un grupo de datos, la mediana para localizar el valor central en una lista de datos y por último la moda para observar el valor que se repite con más frecuencia en los datos suministrados por los usuarios.

Figura 14. Medidas de Tendencia Centra (MTC).

The figure consists of two side-by-side screenshots of a code editor. The left screenshot, titled 'median.js', shows the implementation of a median function. It imports a 'quantile' function from './quantile.js' and includes a detailed JSDoc comment explaining that the median is the middle value of a sorted list, with an example of calculating the median of [10, 2, 5, 100, 2, 1] as 3.5. The function 'median(x)' calls 'quantile(x, 0.5)'. The right screenshot, titled 'mean.js', shows the implementation of a mean function. It imports a 'sum' function from './sum.js' and includes a JSDoc comment explaining that the mean is the sum of all values divided by the number of values, with an example of calculating the mean of [0, 10] as 5. The function 'mean(x)' calculates 'sum(x) / x.length' and includes an error check for empty arrays.

(A) Mediana

(B) Media

The screenshot shows the implementation of a mode function in a file named 'mode.js'. It imports 'modeSorted' from './mode_sorted.js' and 'numericSort' from './numeric_sort.js'. The JSDoc comment defines the mode as the number that appears most frequently in a list, noting that it returns the most recent mode in case of a tie. It states that the algorithm runs in O(n log(n)) due to internal sorting. An example shows 'mode([0, 0, 1])' returning 0. The function 'mode(x)' uses 'modeSorted(numericSort(x))' to find the mode.

(C) Moda

Tomado de [30]

En cuanto a medidas de dispersión, se agregó la funcionalidad de poder calcular la varianza, esto para poder representar la variabilidad del conjunto de datos que el software reciba, esa variabilidad se representa respecto a la media aritmética de los mismos datos.

Figura 15. *Varianza.*

```

JS variance.js X
node_modules > simple-statistics > src > JS variance.js > ...
8   * see the "sampleVariance" method if you want a sample measure.
9   *
10  * @param {Array<number>} x a population of one or more data points
11  * @returns {number} variance: a value greater than or equal to zero.
12  * zero indicates that all values are identical.
13  * @throws {Error} if x's length is 0
14  * @example
15  * variance([1, 2, 3, 4, 5, 6]); // => 2.9166666666666665
16  */
17  function variance(x) {
18    if (x.length === 0) {
19      throw new Error("variance requires at least one data point");
20    }
21
22    // Find the mean of squared deviations between the
23    // mean value and each value.
24    return sumNthPowerDeviations(x, 2) / x.length;
25  }
26
27  export default variance;

```

Tomado de [30]

Por último, se integra la desviación estándar para que los usuarios puedan conocer no solo las medidas de tendencia central, sino también la desviación que tienen los datos en su distribución con el fin de poder tener una visión más clara de los mismos a la hora de interpretar los datos para poder tomar buenas decisiones en la empresa.

Figura 16. *Desviación estándar.*

```

JS standard_deviation.js X
node_modules > simple-statistics > src > JS standard_deviation.js > ...
1  import variance from "./variance.js";
2
3  /**
4   * The [standard deviation](http://en.wikipedia.org/wiki/standard_deviation)
5   * is the square root of the variance. This is also known as the population
6   * standard deviation. It's useful for measuring the amount
7   * of variation or dispersion in a set of values.
8   *
9   * Standard deviation is only appropriate for full-population knowledge: for
10  * samples of a population, {@link sampleStandardDeviation} is
11  * more appropriate.
12  *
13  * @param {Array<number>} x input
14  * @returns {number} standard deviation
15  * @example
16  * variance([2, 4, 4, 4, 5, 5, 7, 9]); // => 4
17  * standardDeviation([2, 4, 4, 4, 5, 5, 7, 9]); // => 2
18  */
19  function standardDeviation(x) {
20    if (x.length === 1) {
21      return 0;
22    }
23    const v = variance(x);
24    return Math.sqrt(v);
25  }
26
27  export default standardDeviation;

```

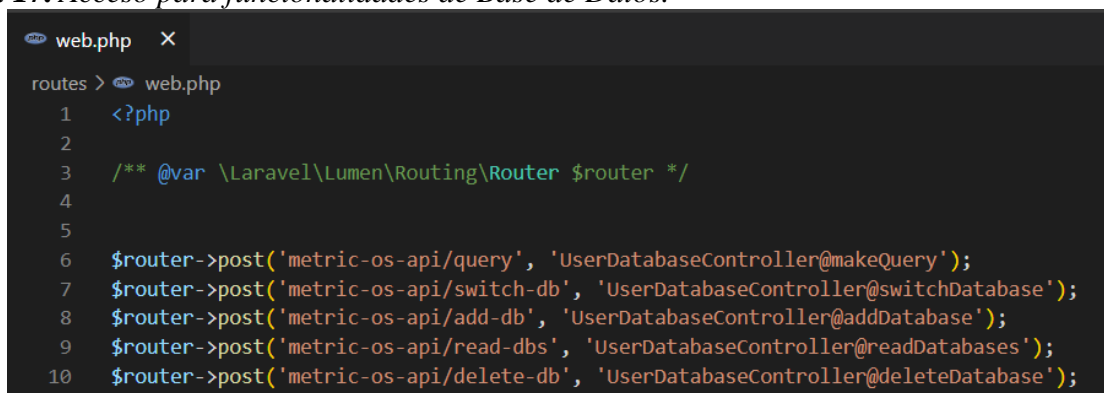
Tomado de [30]

3.3.3 Configuración de rutas para servir al Front-end

Teniendo en cuenta que la información recopilada en el Back-end debe ser suministrada como servicio al Front-end para su debida manipulación y presentación visual al usuario, se configuran unas rutas dentro del proyecto para poder controlar cada uno de los elementos que serán presentados al Front-end; es preciso mencionar que el proyecto que abarca la sección del desarrollo Front-end para la solución software presentada, ha sido realizado por Angie Ríos y Rubén García [31].

Estas rutas se presentan en el archivo “*web.php*” del proyecto Back-end, allí se encuentran como primera sección las rutas encargadas de controlar y servir los datos de las bases de datos agregadas al aplicativo.

Figura 17. Acceso para funcionalidades de Base de Datos.



```
web.php X
routes > web.php
1 <?php
2
3 /** @var \Laravel\Lumen\Routing\Router $router */
4
5
6 $router->post('metric-os-api/query', 'UserController@makeQuery');
7 $router->post('metric-os-api/switch-db', 'UserController@switchDatabase');
8 $router->post('metric-os-api/add-db', 'UserController@addDatabase');
9 $router->post('metric-os-api/read-dbs', 'UserController@readDatabases');
10 $router->post('metric-os-api/delete-db', 'UserController@deleteDatabase');
```

Seguido de estas rutas se encuentran las rutas encargadas de servir los datos proporcionados para la configuración de Dashboards.

Figura 18. Acceso para funcionalidades de métricas.

```
12 $router->post('metric-os-api/add-dashboard', 'DashboardController@addDashboard');  
13 $router->post('metric-os-api/read-dashboards', 'DashboardController@readDashboard');
```

Como tercera sección se configuraron las rutas que brindan los datos de los componentes de las Dashboards que el usuario genere.

Figura 19. Acceso para componentes de Dashboard

```
15 $router->post('metric-os-api/read-components', 'ComponentController@readComponents');  
16 $router->post('metric-os-api/save-component', 'ComponentController@saveComponent');  
17 $router->post('metric-os-api/move-component', 'ComponentController@moveComponents');  
18 $router->post('metric-os-api/resize-component', 'ComponentController@resizeComponents');
```

Posterior a ello, se establecieron las rutas para poder servir los datos de las Dashboards que contienen los análisis estadísticos.

Figura 20. Acceso para funcionalidades de Dashboards para análisis estadístico.

```
20 $router->post('metric-os-api/add-st-dashboard', 'StatisticsController@addStDashboard');  
21 $router->post('metric-os-api/read-st-dashboards', 'StatisticsController@readStDashboard');
```

Por último, se presentan las rutas para servir los datos de los componentes que contienen las Dashboards creadas para los análisis estadísticos.

Figura 21. Acceso para componentes de Dashboard de análisis estadístico.

```
23 $router->post('metric-os-api/read-st-components', 'ComponentStController@readStComponents');  
24 $router->post('metric-os-api/save-st-component', 'ComponentStController@saveStComponent');  
25 $router->post('metric-os-api/move-st-component', 'ComponentStController@moveStComponents');  
26 $router->post('metric-os-api/resize-st-component', 'ComponentStController@resizeStComponents');
```

De esta manera se presenta un servicio que entrega los datos recibidos en el Back-end al proyecto Front-end para su debida interpretación.

3.3.4 Migrar la plataforma a una aplicación híbrida

Para el desarrollo de esta actividad se tuvo en cuenta que la implementación de una aplicación de escritorio implica la migración del proyecto a nivel de Front-end mediante Electron, permitiendo así que el usuario pueda acceder a la interfaz del software y de esta manera realizar la interacción con la aplicación y hacer uso de las funcionalidades que esta provee. Es por esto, por lo que se presenta el desarrollo de esta migración desde el proyecto de Front-end de la siguiente manera:

- Como primer punto, en el proyecto de VueJS (Framework empleado para el desarrollo de Front-end) se ejecuta el comando “*vue add electron-builder*” el cuál agrega los archivos y scripts necesarios para construir la aplicación Vue en una aplicación Electron.
- Seguido de esto, se agrega un archivo en la ruta */src/* llamado “*background.js*” que contiene las propiedades y las funcionalidades de la ventana que se va a crear.

Figura 22. *Propiedades de la ventana.*

```
src > JS background.js > createWindow
1  "use strict";
2
3  import { app, protocol, BrowserWindow } from "electron";
4  import { createProtocol } from "vue-cli-plugin-electron-builder/lib";
5  import installExtension, { VUEJS_DEVTOOLS } from "electron-devtools-installer";
6  const isDevelopment = process.env.NODE_ENV !== "production";
7
8  // Scheme must be registered before the app is ready
9  protocol.registerSchemesAsPrivileged([
10   { scheme: "app", privileges: { secure: true, standard: true } },
11  ]);
12
13  async function createWindow() {
14    // Create the browser window.
15    const win = new BrowserWindow({
16      width: 1280,
17      height: 720,
18      webPreferences: {
19        // Use pluginOptions.nodeIntegration, leave this alone
20        // See nklayman.github.io/vue-cli-plugin-electron-builder/guide/security.html#node-integrati
21        nodeIntegration: process.env.ELECTRON_NODE_INTEGRATION,
22        contextIsolation: !process.env.ELECTRON_NODE_INTEGRATION,
23      },
24    });
```

Figura 23. Funcionalidades de la ventana

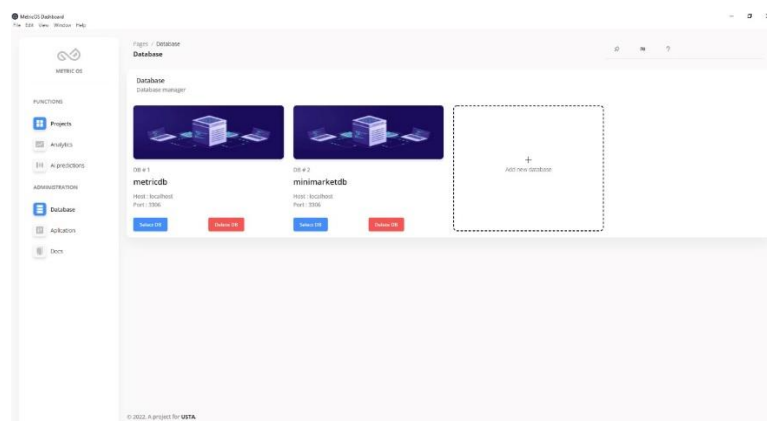
```

37 // Quit when all windows are closed.
38 app.on("window-all-closed", () => {
39   // On macOS it is common for applications and their menu bar
40   // to stay active until the user quits explicitly with Cmd + Q
41   if (process.platform !== "darwin") {
42     app.quit();
43   }
44 });
45
46 app.on("activate", () => {
47   // On macOS it's common to re-create a window in the app when the
48   // dock icon is clicked and there are no other windows open.
49   if (BrowserWindow.getAllWindows().length === 0) createWindow();
50 });
51
52 // This method will be called when Electron has finished
53 // initialization and is ready to create browser windows.
54 // Some APIs can only be used after this event occurs.
55 app.on("ready", async () => {
56   if (isDevelopment && !process.env.IS_TEST) {
57     // Install Vue Devtools
58     try {
59       await installExtension(VUEJS_DEVTOOLS);
60     } catch (e) {
61       console.error("Vue Devtools failed to install:", e.toString());
62     }
63   }
64   createWindow();
65 });

```

- Para iniciar la visualización del proyecto ejecutamos el comando “*npm run electron:serve*” en la terminal de comandos, lo cual permitirá visualizar los cambios en tiempo real durante el proceso de desarrollo.

Con esto ya se puede iniciar el desarrollo del proyecto en una aplicación de Electron y de esta manera es como se visualiza el proyecto en la aplicación Electron:

Figura 24. Metric-OS desde la aplicación de Electron.

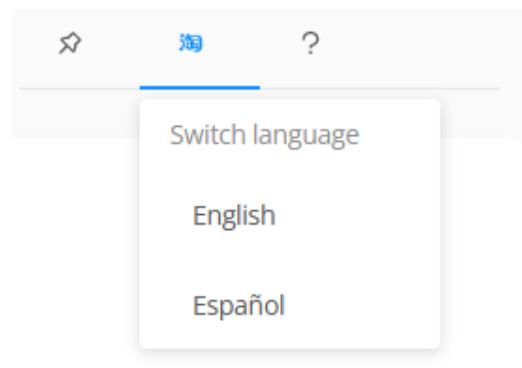
3.4 Puesta en operación del proyecto

3.4.1 Validación de funcionalidades

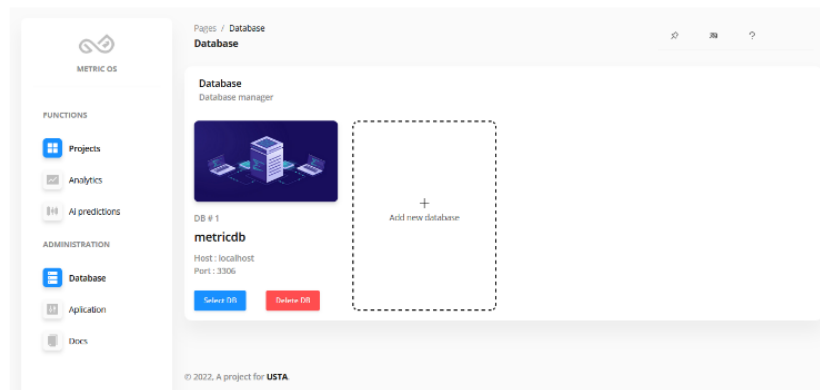
Como resultado final se obtiene una aplicación funcional con diferentes características de las que el usuario puede hacer uso según corresponda el propósito que tienen para la información que sea suministrada en la aplicación.

Inicialmente se visualiza la integración de la funcionalidad de cambio de lenguaje (inglés – español) como se observa en la siguiente figura.

Figura 25. Cambio de lenguaje en la aplicación.



La manipulación de las bases de datos que se integran al software podrá realizarse en el apartado “*Database*” que se encuentra en la sección “*Administration*” tal como se muestra a continuación.

Figura 26. *Databases*

Al igual que la gestión de las bases de datos, los proyectos se pueden visualizar en las funcionalidades que presenta la solución software en el apartado “*Functions*”.

Figura 27. *Visualización de proyectos.*

El software presenta una funcionalidad que permite la creación de gráficos estadísticos para la debida visualización de los datos suministrados por las empresas. El proceso de creación consta de la elección del tipo de gráfico mostrado en la *Figura 28*, seguido de la elección de los datos *Figura 29*; en esta sección el usuario podrá elegir los datos a graficar mediante el proceso de

consultas que permitirán obtener los datos deseados por los usuarios. Por último, se realiza la selección de las propiedades que se requieren para el gráfico, según lo presenta la *Figura 30*.

Figura 28. Elección de tipo de gráfico.

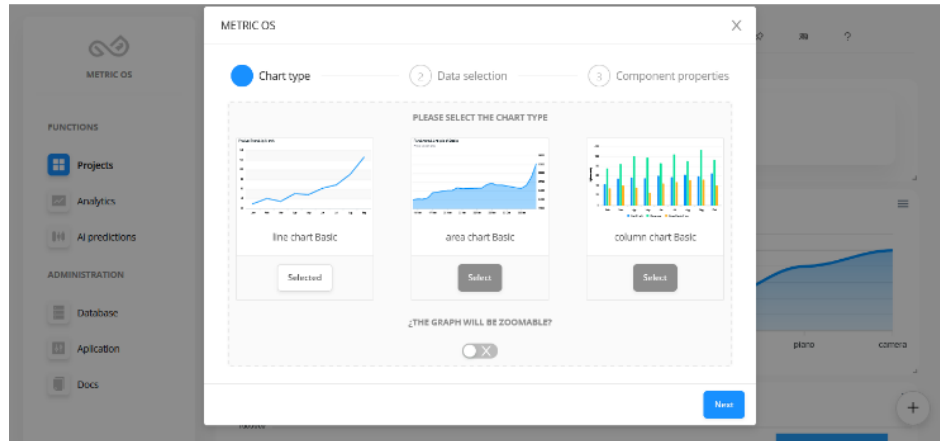


Figura 29. Consulta de datos.

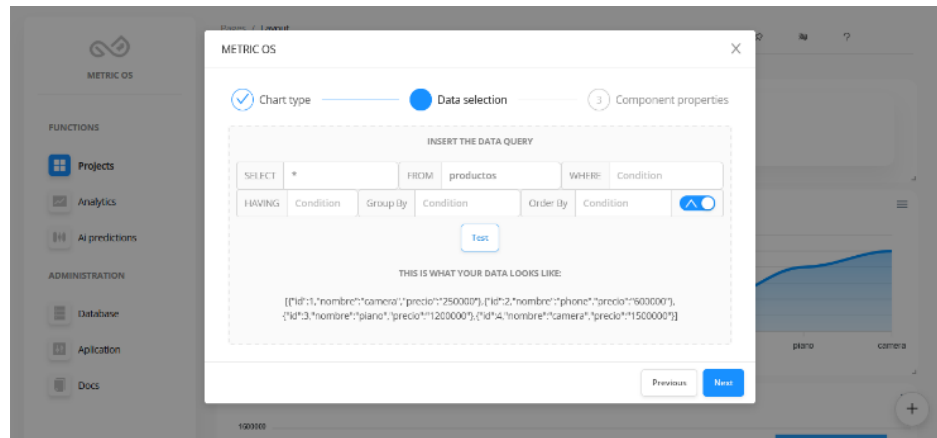
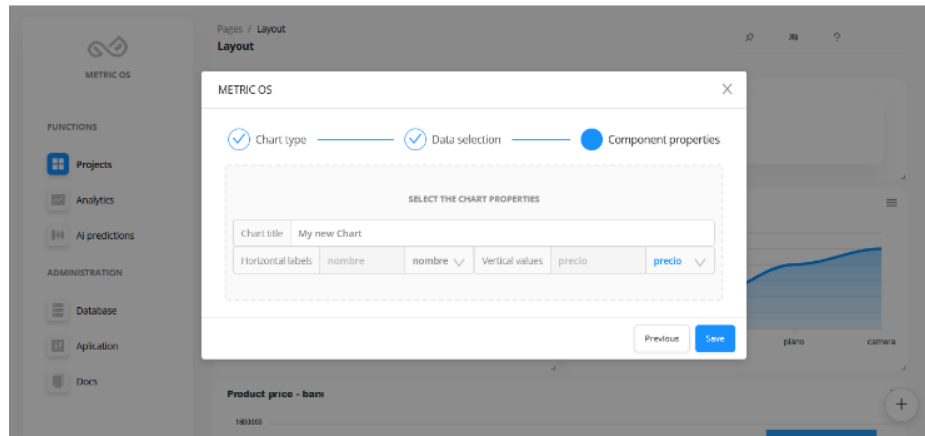


Figura 30. Selección de propiedades del gráfico.

Todos estos datos mencionados anteriormente se guardan para su posterior visualización en una dashboard, Esta dashboard puede ser creada por el usuario; se pueden crear tantas dashboards como bases de datos se deseen analizar. La creación de una dashboard se presenta en la *Figura 31* y la visualización de una dashboard ya creada se puede observar en la *Figura 32*.

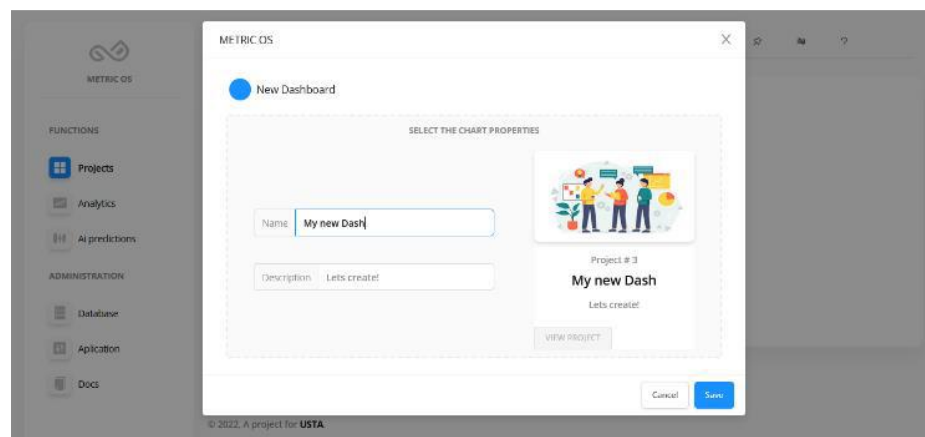
Figura 31. Creación de dashboard.

Figura 32. *Dashboard*

El software, en su estructura de back-end presenta la configuración realizada en el framework Laravel, en la *Figura 33* se puede observar cada uno de los controladores del sistema desarrollados en el proyecto.

Figura 33. *Controladores del proyecto en Laravel.*

- ComponentController.php
- Controller.php
- DashboardController.php
- ExampleController.php
- PlatformDataController.php
- UserDatabaseController.php

Se cuenta también con una serie de archivos generados para llevar un registro de las diferentes variables del entorno del usuario (*Figura 34*), así como también la configuración dedicada para autorizar las peticiones desde el aplicativo, tal como lo muestra la *Figura 35*.

Figura 34. *Local register.***Figura 35.** *Middleware.*

```
public function handle($request, Closure $next)
{
    $headers = [
        'Access-Control-Allow-Origin'      => '*',
        'Access-Control-Allow-Methods'    => 'POST, GET, OPTIONS, PUT, DELETE',
        'Access-Control-Allow-Credentials' => 'true',
        'Access-Control-Max-Age'          => '86400',
        'Access-Control-Allow-Headers'    => 'Content-Type, Authorization, X-Requested-With'
    ];

    if ($request->isMethod('OPTIONS')) {
        return response()->json(['method':"OPTIONS"], 200, $headers);
    }

    $response = $next($request);
    foreach ($headers as $key => $value) {
        $response->header($key, $value);
    }
}
```

Por último, se muestran las propiedades que se guardan para cada componente en un dashboard, esta incluye tanto su posicionamiento en la cuadrícula como sus dimensiones y, finalmente, la consulta SQL con la cual se recopilarán los datos para el componente (*Figura 36*).

Figura 36. *Component – saved – properties.*


```
{
  "i": 0,
  "parentId": "1",
  "component": "text",
  "dataQuery": "Select * FROM productos ORDER BY id",
  "type": "line",
  "zoom": "false",
  "title": "Base Component",
  "xaxis": "nombre",
  "yaxis": "precio",
  "x": 0,
  "y": 0,
  "w": 12,
  "h": 4
},
```

3.4.2 Despliegue del software

Como se mencionó en uno de los objetivos específicos, el software cuenta con una licencia de uso que conserve la misión y visión del proyecto la cual se basa en ser un proyecto de uso libre y gratuito que funcione como base a aquellos usuarios desarrolladores.

El proyecto cuenta con una licencia “Mozilla Public License 2.0” [32], una licencia de tipo CopyLeft cuya finalidad es proporcionar el proyecto de uso gratuito y modificable, así como también permite que los usuarios puedan realizar cambios al proyecto, cambios que se pueden basar en la adición de módulos de análisis estadísticos, personalizar secciones del software, colores y demás. Es necesario aclarar que la licencia a pesar de que permite que se realicen cambios al código fuente, esta así mismo protege que el proyecto siga siendo público y de esta manera los usuarios pueden adaptar el proyecto como un producto interno en sus organizaciones.

Figura 37. *Mozilla Public License 2.0*

 Danilo-GB/MetricOS is licensed under the Mozilla Public License 2.0 Permissions of this weak copyleft license are conditioned on making available source code of licensed files and modifications of those files under the same license (or in certain cases, one of the GNU licenses). Copyright and license notices must be preserved. Contributors provide an express grant of patent rights. However, a larger work using the licensed work may be distributed under different terms and without source code for files added in the larger work.	Permissions <ul style="list-style-type: none"> ✓ Commercial use ✓ Modification ✓ Distribution ✓ Patent use ✓ Private use 	Limitations <ul style="list-style-type: none"> ✗ Liability ✗ Trademark use ✗ Warranty 	Conditions <ul style="list-style-type: none"> ⓘ Disclose source ⓘ License and copyright notice ⓘ Same license (file)
This is not legal advice. Learn more about repository licenses.			

3.4.3 Liberación del código fuente

Para este punto del proyecto se realiza el despliegue del código fuente del aplicativo en el repositorio de la plataforma GitHub, allí el usuario podrá encontrar los archivos pertinentes para el debido funcionamiento del aplicativo software en caso tal de querer realizar algún cambio, o si bien, el usuario solo desea hacer uso del proyecto base, también se encuentra disponible la carpeta con el archivo .exe el cuál es el archivo ejecutable de escritorio.

Estos documentos se encuentran disponibles para el público en la siguiente dirección web: metric-os.netlify.app.

3.4.4 Manual de usuarios

El proyecto en principio está desarrollado para que usuarios con conocimientos en desarrollo web hagan uso de este, sin embargo, se presenta un manual de usuario donde se expone las diferentes funcionalidades del software desarrollado; funcionalidades tanto a nivel Back-end como a nivel Front-end. Este manual de usuario se encuentra disponible en la página web del proyecto, alojada en la siguiente dirección web: metric-os-docs.netlify.app

En el manual de usuario podrán encontrar tres secciones, “Getting Started”, “Folder Structure” y por último “Installation”, allí se presenta toda la información relacionada con la instalación del proyecto, el uso adecuado de sus funcionalidades y algunas características principales a tener en cuenta del proyecto.

Figura 38. "Getting Started" - Manual de Usuarios.

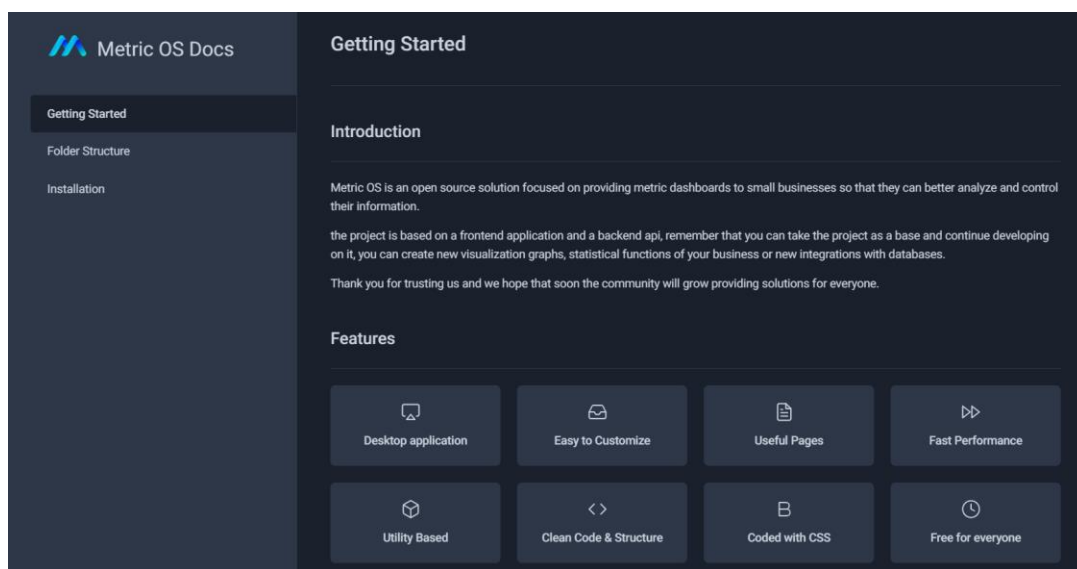
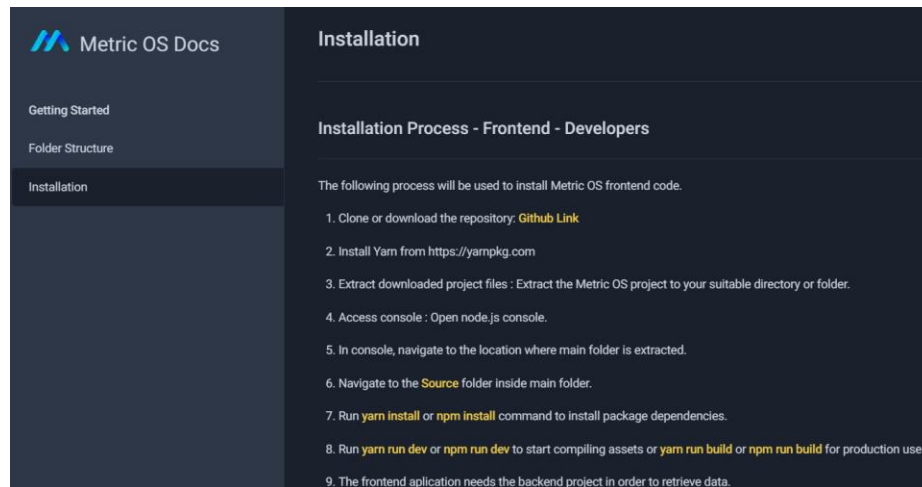
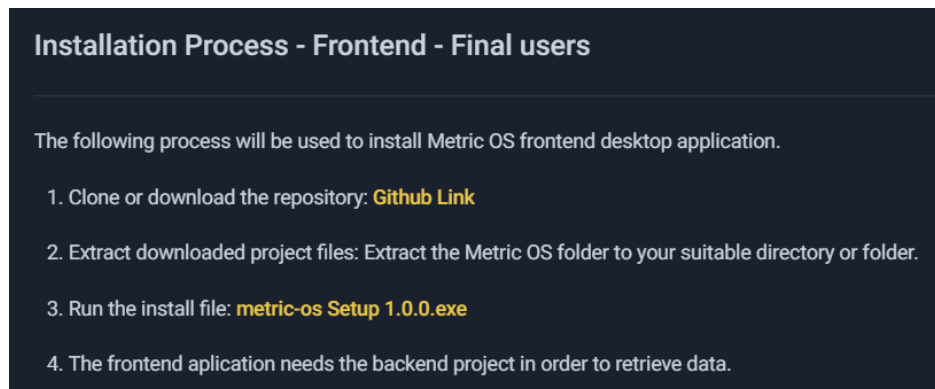
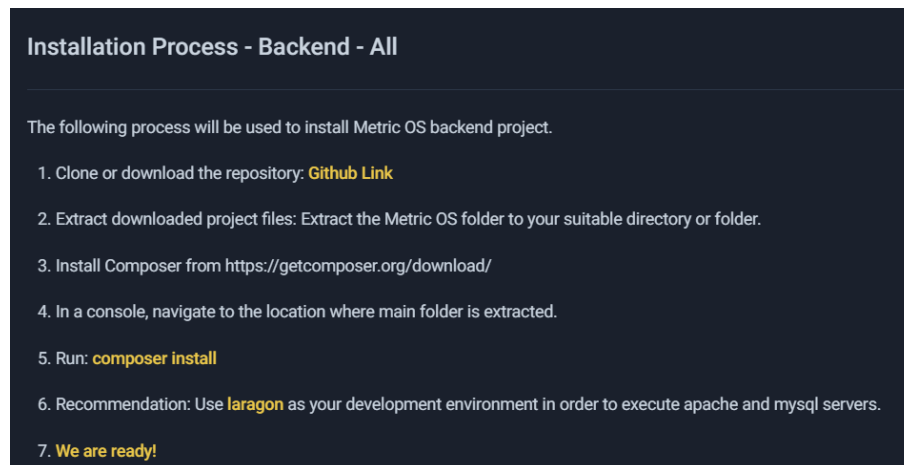


Figura 39. *Proceso de Instalación Front-end para desarrolladores.***Figura 40.** *Proceso de Instalación Front-end para usuarios finales.***Figura 41.** *Proceso de Instalación Back-end.*

Es importante mencionar que la sección “Folder Structure” del manual de usuario, no se presenta en este proyecto debido a que su contenido presenta información relevante para el uso de funcionalidades a nivel Front-end, presentes en el proyecto Front-end realizado por mis compañeros Angie Ríos y Rubén García [31].

4. Conclusiones

Una vez implementadas las configuraciones para la conexión con las bases de datos MySQL y MongoDB, se puede observar que el proceso de conexión es similar en cuanto a las variables de conexión con la solución software y en un entorno local permite tener mayor control sobre las bases de datos que se deseen analizar.

La implementación de una aplicación híbrida mediante Electron permitió tener una mayor accesibilidad a la solución software sin la necesidad de iniciar el proyecto desde un editor de código fuente como lo pueden ser Visual Studio Code o Sublime Text.

Las funciones estadísticas permitieron tener una visión más clara de la información suministrada, esto gracias a que se realizaron pruebas con distintas bases de datos (durante el desarrollo del proyecto) para poder verificar la calidad de los resultados arrojados por la solución software.

5. Trabajo Futuro

Gracias a que la solución software desarrollada es de naturaleza código libre, se plantea implementar como trabajo futuro la integración de modelos de predicción basados en CNN para poder brindar a los usuarios una mayor visión frente al análisis de la información suministrada en el software.

La implementación de estos modelos conlleva una mayor dedicación en su investigación y comprensión para poder ser integrados de manera correcta en el software, debido al lenguaje de programación que se emplea para su correcta ejecución.

Referencias

- [1] Franco Ángel, M., & Urbano, D. (2019). Caracterización de las pymes colombianas y de sus fundadores: Un Análisis Desde dos Regiones del País. *Estudios Gerenciales*, 81–91. <https://doi.org/10.18046/j.estger.2019.150.2968>
- [2] M. E. L. P. Rosario de Fátima Suárez Améndola, «Bases de Datos en las Pequeñas y Medianas Empresas,» 2012.
- [3] W. N. E. E. F. M. A. C. E. F. V. Nubia Casquete Baidal, «Aplicación de las estadísticas en las empresas,» *Revista Caribeña de Ciencias Sociales*, 2016.
- [4] Y. Demchenko et al., "EDISON Data Science Framework: A Foundation for Building Data Science Profession for Research and Industry," 2016 IEEE International Conference on Cloud Computing Technology and Science (CloudCom), 2016, pp. 620-626, doi: 10.1109/CloudCom.2016.0107.
- [5] Y. Hao and Y. -f. Zhang, "Statistical prediction modeling for software development process performance," 2011 IEEE 3rd International Conference on Communication Software and Networks, 2011, pp. 703-706, doi: 10.1109/ICCSN.2011.6014187.
- [6] M. Anif, A. Dentha and H. W. S. Sindung, "Designing internship monitoring system web based with Laravel framework," 2017 IEEE International Conference on Communication, Networks and Satellite (Comnetsat), 2017, pp. 112-117, doi: 10.1109/COMNETSAT.2017.8263583.
- [7] M. M. R. G. Mário Antunes, «A Customizable Web Platform to Manage Standards Compliance of Information Security and Cybersecurity Auditing, » vol. 196, pp. 36-43, 2022.

- [8] I. Zoratti, "MYSQL Security Best Practices," 2006 IET Conference on Crime and Security, 2006, pp. 183-198.
- [9] W. Sriratana, V. Khagwian and S. Sathamsakul, "Analysis of Electric Current by Using MySQL Database on Web Server for Machine Performance Evaluation: A Case Study of Air Conditioning System," 2020 20th International Conference on Control, Automation and Systems (ICCAS), 2020, pp. 437-442, doi: 10.23919/ICCAS50221.2020.9268222.
- [10] MongoDB, «MongoDB, » [En línea]. Available: <https://www.mongodb.com/es/what-is-mongodb>.
- [11] N. S. Yerramilli, N. J. Johnson, and Y. Omsri Sainadh Reddy, "College Exam Allocation Using MongoDB and Python3," 2021 2nd Global Conference for Advancement in Technology (GCAT), 2021, pp. 1-3, doi: 10.1109/GCAT52182.2021.9587589.
- [12] A. Fiechter, R. Minelli, C. Nagy, and M. Lanza, "Visualizing GitHub Issues," 2021 Working Conference on Software Visualization (VISSOFT), 2021, pp. 155-159, doi: 10.1109/VISSOFT52517.2021.00030.
- [13] F. S. Foundation, «El sistema operativo GNU,» [En línea]. Available: <https://www.gnu.org/philosophy/free-sw.es.html>. Elissa, "Title of paper if known," unpublished.
- [14] maldeadora, «Platzi, » 2018. [En línea]. Available: <https://platzi.com/blog/que-es-frontend-y-backend/>.
- [15] F. Machuca, «Crehana,» 2021. [En línea]. Available: <https://www.crehana.com/pe/blog/desarrollo-web/que-es-laravel/>.
- [16] U. d. Alicante. [En línea]. Available: <https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista->

- controladormvc.html#:~:text=Modelo%20Vista%20Controlador%20(MVC)%20es,contro
l%20en%20tres%20componentes%20distintos.
- [17] C. Stedman, «ComputerWeekly,» septiembre 2021. [En línea]. Available:
<https://www.computerweekly.com/es/definicion/Ciencia-de-datos>. J. Clerk Maxwell, A
Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–
73.
- [18] G. Stedman, «Tech Accelerator, » Agosto 2021. [En línea]. Available:
<https://www.techtarget.com/searchenterpriseai/definition/data-science>.
- [19] D. Farmer, «The data science process: 6 key steps on analytics applications, » 2020.
- [20] LOGICALIS, «LOGICALIS -Architects of Change, » 06 octubre 2014. [En línea].
Available: [https://blog.es.logicalis.com/analytics/la-monitorizacion-de-datos-en-](https://blog.es.logicalis.com/analytics/la-monitorizacion-de-datos-en-sistemas-de-business)
sistemas-de-business Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron
spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE
Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf.
Magnetics Japan, p. 301, 1982].
- [21] A. Robledano, «OpenWebinars,» septiembre 2019. [En línea]. Available:
<https://openwebinars.net/blog/que-es-mysql/>.
- [22] N. Cottrell, MongoDB Topology Design, Berkeley, CA: Apress, 2020.
- [23] D. Camacho, «Platzi,» 2021. [En línea]. Available: [https://platzi.com/blog/que-es-github-](https://platzi.com/blog/que-es-github-como-funciona/?utm_source=google&utm_medium=cpc&utm_campaign=12915366154&utm_adgroup=&utm_content=&gclid=Cj0KCQjw8_qRBhCXARIsAE2AtRbUGrFoe1Ov8DYf7c6kL9iw5BGLrNKzB_UsI4mFcyPQhw7j7oWw87oaAml6EALw_wcB&gclsrc=aw.ds)
como-
funciona/?utm_source=google&utm_medium=cpc&utm_campaign=12915366154&utm_
adgroup=&utm_content=&gclid=Cj0KCQjw8_qRBhCXARIsAE2AtRbUGrFoe1Ov8DY
f7c6kL9iw5BGLrNKzB_UsI4mFcyPQhw7j7oWw87oaAml6EALw_wcB&gclsrc=aw.ds.

- [24] Oracle, «Oracle,» 2022. [En línea]. Available: <https://www.oracle.com/co/database/what-is-a-relational-database/>.
- [25] A. W. Services, «AWS,» 2022. [En línea]. Available: <https://aws.amazon.com/es/nosql/>.
- [26] Z. Tejada, «Microsoft | Learn,» [En línea]. Available: <https://learn.microsoft.com/es-es/azure/architecture/data-guide/big-data/non-relational-data>.
- [27] Laravel, "Laravel," [Online]. Available: <https://laravel.com/docs/7.x/database>. [Accessed 2022].
- [28] MongoDB, "MongoDB," [Online]. Available: <https://www.mongodb.com/compatibility/mongodb-laravel-intergration>. [Accessed 2022].
- [29] IBM, «IBM Database Security,» [En línea]. Available: <https://www.ibm.com/co-es/cloud/learn/database-security>. [Último acceso: 2022].
- [30] T. MacWright, «Simple Statistics,» 2012.
- [31] A.T. Ríos Sandoval, R.D. García Bautista, “Diseño e implementación de un software libre para el monitoreo y análisis de información en bases de datos enfocado a contribuir con el crecimiento de pequeñas y mediana empresas”, Tesis Pregrado, Universidad Santo Tomás, Bucaramanga, Colombia, 2022.
- [32] Mozilla, «Mozilla Public License,» 2022. [En línea]. Available: <https://www.mozilla.org/en-US/MPL/2.0/>.