



Universidad de **Nariño**

UNA APLICACIÓN DE BASES DE DATOS NoSQL PARA LA GESTIÓN DE
INFORMACIÓN DE EGRESADOS DEL PROGRAMA DE INGENIERÍA DE
SISTEMAS DE LA UNIVERSIDAD DE NARIÑO

GRUPO DE INVESTIGACION GRIAS

UNIVERSIDAD DE NARIÑO

SAN JUAN DE PASTO

2021

UNA APLICACIÓN DE BASES DE DATOS NoSQL PARA LA GESTIÓN DE
INFORMACIÓN DE EGRESADOS DEL PROGRAMA DE INGENIERÍA DE
SISTEMAS DE LA UNIVERSIDAD DE NARIÑO

INFORME FINAL

Estudiantes investigadores:

CARLOS ANDRÉS TORO GUERRERO

CARLOS ANDRÉS OCAÑA MORILLO

ESTEFANY YURANI TULCÁN DÍAZ

Co-asesor:

SILVIO RICARDO TIMARAN PEREIRA

UNIVERSIDAD DE NARIÑO

SAN JUAN DE PASTO

2021

Contenido

1. INTRODUCCIÓN	4
2. OBJETIVOS.....	5
2.1. OBJETIVO GENERAL	5
2.2. OBJETIVOS ESPECIFICOS	5
2.2.1. Seleccionar la bibliografía que permita a los investigadores la apropiación del conocimiento necesario sobre bases de datos NoSQL y el gestor MongoDB para desarrollar el proyecto satisfactoriamente	5
3. MARCO TEORICO.....	6
Bases de datos no relacionales NoSQL.....	7
4. MATERIALES Y MÉTODOS.....	7
5. METODOLOGÍA	8
Fase 1: Apropiación del conocimiento.....	8
Fase 2: Exploración y gestión del sistema de bases de datos NoSQL MongoDB.....	8
Fase 3: Diseño e implementación de la base de datos de los egresados del programa de ingeniería de Sistemas de la Universidad de Nariño.....	9
Fase 4: Diseño y desarrollo del aplicativo Ripuy	9
Fase 5: Construcción y realización de pruebas de funcionalidad del aplicativo Ripuy	9
6. RESULTADOS	10
Fase 1: Apropiación del conocimiento.....	10
Fase 2: Exploración y gestión del sistema de bases de datos NoSQL MongoDB.....	10
Fase 3: Diseño e implementación de la base de datos de los egresados del programa de ingeniería de Sistemas de la Universidad de Nariño.....	10
Fase 4: Diseño y desarrollo del aplicativo Ripuy	11
Fase 5: Construcción y realización de pruebas de funcionalidad del aplicativo Ripuy ...	11
7. DISCUSIONES	15
8. CONCLUSIONES	16
9. RECOMENDACIONES	17
REFERENCIAS	19
ADJUNTOS	21

RESUMEN

Se realizó la implementación en la construcción de una base de datos para la gestión de la información de egresados del programa de Ingeniería de Sistemas utilizando este gestor mediante el aplicativo Ripuy, con el fin de adquirir el conocimiento y experiencia necesaria para la implementación de un laboratorio y electiva de bases de datos NoSQL y como un primer paso para el montaje de Bases de Datos NoSQL en la Universidad de Nariño. Ripuy se desarrolló mediante una metodología descriptiva bajo el enfoque cuantitativo, aplicando un diseño no experimental. Se contemplaron tres fases: La apropiación del conocimiento sobre bases de datos NoSQL y el gestor MongoDB, la exploración y gestión del sistema de bases de datos MongoDB; el diseño e implementación de la base de datos de egresados del programa de ingeniería de Sistemas de la Universidad de Nariño y la construcción de pruebas de funcionalidad y de rendimiento del aplicativo; el diseño y desarrollo del aplicativo Ripuy que gestiona la información de egresados de Ingeniería de Sistemas usando el gestor MongoDB, un grupo de pruebas de funcionamiento y de rendimiento del aplicativo Ripuy y el gestor MongoDB. Los resultados principales que se obtuvieron son los siguientes: una documentación de MongoDB, estructurada de forma ordenada y legible para ser usada como guía para la instrucción y el manejo del gestor, la construcción del aplicativo Ripuy que gestiona la información de egresados de Ingeniería de Sistemas usando el gestor MongoDB y las pruebas unitarias de funcionamiento y de rendimiento del aplicativo Ripuy y el gestor MongoDB.

Los resultados de las pruebas permitieron concluir qué el rendimiento del gestor MongoDB demostró ser superior a los gestores SQL relacionales en cuanto al uso de los recursos hardware en circunstancias limitantes y su flexibilidad facilitó implementar nuevos modelos y modificaciones durante el desarrollo sin afectar el diseño del modelo y los datos almacenados.

1. INTRODUCCIÓN

Cada día que pasa, el manejo de información a nivel mundial, se hace más complejo, como, por ejemplo, la gran cantidad de usuarios que deben manejar las aplicaciones web, la proliferación del comercio electrónico y el uso masivo de redes sociales así como la prestación de servicios virtuales a través de internet, lo cual ha ido en aumento en las últimas décadas; esto ha hecho que se vengán desarrollando nuevas tecnologías que apoyen el desarrollo de aplicaciones que resuelvan los nuevos problemas emergentes respecto a el manejo de grandes cantidades de información.

Durante muchos años los servicios y aplicaciones de Internet se han apoyado en las bases de datos relacionales, que funcionan muy bien con fuentes de información que generan datos muy bien estructurados (no variantes). Pero estas bases de datos relacionales tienen ciertas carencias a la hora de enfrentarse a aplicaciones que generan Big Data.

La dificultad más grande ocurre en el momento en el que se pretende mantener el rendimiento a medida que aumenta la cantidad de datos. Es decir, no es posible escalar correctamente a partir de cierto volumen. Dentro del contexto de bases de datos relacionales, normalmente se escala de forma vertical (aumentando recursos de la máquina), pero cuando hay que hacerlo horizontalmente (añadiendo máquinas) pueden aparecer problemas de rendimiento. Cuando una aplicación maneja Big Data, es conveniente escalar horizontalmente, ya que almacenar todos los datos en una máquina es complejo. O bien no existe la máquina capaz de procesar esa gran cantidad de información, o bien es muy costosa.

Como alternativa de solución a este problema, surgen nuevas bases de datos, desarrolladas para trabajar con grandes cantidades de información (Big Data) no estructurada y/o cambiantes, y para escalar horizontalmente en vez de verticalmente. A estas nuevas bases de datos se les denomina bases de datos NoSQL.

La eficiencia del manejo de datos y la velocidad de respuesta de las aplicaciones que tienen que ver con éstos, es un tema importante y trascendente, por tanto se ha venido prestando mayor atención a los gestores y/o sistemas de bases de datos NoSql, las cuales han sido desarrolladas para brindar esquemas flexibles en el trabajo con datos no estructurados, alta eficiencia ante consultas de datos, y posibilidades de escalar horizontalmente, sacrificando cuestiones como la consistencia de los datos u otras características deseables en las bases de datos SQL.

La Universidad de Nariño ha implementado bases de datos relacionales en las áreas académica, administrativa, financiera e institucional, las cuales se desempeñan bien en la gestión de datos estructurados, pero presentan una gran dificultad con el manejo de datos semiestructurados y no estructurados que se obtienen de diferentes fuentes de datos como la internet.

El contar con una base de datos en MongoDB permite a los estudiantes del programa de Ingeniería de Sistemas conocer de primera mano las ventajas de implementar esta tecnología, sus principales características y los grandes beneficios que pueden brindar las bases de datos no relacionales en la creación de grandes sistemas escalables que permiten el análisis de

grandes volúmenes de datos, para obtener conocimiento e ideas que ayuden a la toma de decisiones.

Éste es un primer paso para el montaje de Bases de Datos NoSQL, de modo que esta información pueda ser utilizada para tomar decisiones más adecuadas respecto al mejoramiento de la calidad de la educación, en nuestra institución.

La Universidad de Nariño tiene un compromiso con la región del sur de Colombia, forjando profesionales con la competitividad e integridad para llevar a cabo su misión y visión institucional. En ese sentido los estudiantes egresados de la Universidad de Nariño son un ente importante en el crecimiento de la región nariñense y es por eso que Ripuy permite tener almacenada toda la información de los egresados del programa de Ingeniería de Sistemas, además de realizar un seguimiento de su estado laboral, crecimiento profesional y personal.

2. OBJETIVOS

2.1.OBJETIVO GENERAL

Construir una base de datos NoSQL para la gestión de la información de egresados del programa de Ingeniería de Sistemas utilizando el gestor MongoDB y acoplarse a un aplicativo, con el fin de adquirir el conocimiento y experiencia necesaria para la implementación de un laboratorio y electiva de bases de datos NoSQL y como un primer paso para el montaje de Bases de Datos NoSQL en la Universidad de Nariño.

2.2.OBJETIVOS ESPECIFICOS

- 2.2.1.** Seleccionar la bibliografía que permita a los investigadores la apropiación del conocimiento necesario sobre bases de datos NoSQL y el gestor MongoDB para desarrollar el proyecto satisfactoriamente
- 2.2.2.** Instalar, configurar y Administrar el gestor de bases de datos NoSQL MongoDB.
- 2.2.3.** Identificar los requerimientos que permita la construcción de la aplicación de egresados del programa de Ingeniería de Sistemas
- 2.2.4.** Diseñar e implementar en MongoDB, la base de datos que gestione la información de los egresados del programa de Ingeniería de Sistemas de la Universidad de Nariño
- 2.2.5.** Diseñar, desarrollar e implementar la aplicación de gestión de información de egresados del programa de Ingeniería de Sistemas de la Universidad de Nariño acoplándola con la base de datos en MongoDB
- 2.2.6.** Realizar pruebas y Analizar y Evaluar el desempeño de la aplicación Ripuy mediante consultas a la base de datos de MongoDB
- 2.2.7.** 0Elaborar la documentación de la investigación, manual de programador, manual de usuario y documentación de MongoDB.

3. MARCO TEORICO

Con el paso de los años el termino de Big Data ha tomado fuerza, gracias a un sinfín de tendencias tecnológicas que venían madurando desde la primera década del siglo XXI y que se han consolidado en los últimos años, como lo son la interacción de los usuarios con las redes sociales, el aumento de banda ancha, la conexión a internet a un bajo costo, internet de las cosas, computación en la nube, entre otros. Debido a estas tendencias, el volumen de los datos ha crecido significativamente y por tal motivo, surge la necesidad de almacenar la información para analizarla, procesarla e interpretarla para dar soluciones a situaciones negativas que se puedan presentar. Por ende, esta nueva tecnología crece con gran rapidez y se cree que utilizándola adecuadamente puede traer grandes ventajas para las empresas y compañías del mundo [9]. Estas tecnologías son: Big Data.

Cuando se habla de Big Data, se habla de grandes volúmenes de datos y de la necesidad de capturarlos, almacenarlos y analizarlos para conseguir grandes ventajas para las empresas que deciden adoptar esta tecnología [9]. Big Data es un término que describe el tratamiento de grandes volúmenes de conjuntos de datos, tanto estructurados como no estructurados, que inundan los negocios cada día. Pero no es la cantidad de datos lo que es importante. Lo que importa con el Big Data es lo que las organizaciones hacen con los datos. Big Data se puede analizar para obtener ideas que conduzcan a mejores decisiones y movimientos de negocios estratégicos [10].

Para IBM, Big Data se resume a toda la información que no puede ser procesada o analizada utilizando herramientas comunes o tradicionales. Sin embargo, el termino como tal no se refiere a una cantidad específica de información, debido a que cuando se habla de Big Data se habla en términos de petabytes y exabytes de datos. IBM explica que además del gran volumen de información, también existe una gran variedad de datos que a su vez requieren una gran velocidad de respuesta, para obtener la información correcta en el momento preciso [11].

Las características más importantes del Big Data son volumen de información, velocidad, variedad y veracidad de los datos, conocidas como las cuatro V del Big Data, además del valor de los datos, aunque en la actualidad, en los últimos artículos se empieza hablar de las siete V del Big Data, agregando a las mencionadas, viabilidad y visualización de los datos [12]. Todas esas montañas de información han generado un costo potencial al no descubrir el gran valor asociado. Desde luego, el ángulo correcto que actualmente tiene el liderazgo en términos de popularidad para almacenar enormes cantidades de información son las bases de datos no relacionales NoSQL.

horizontalmente, ya que almacenar todos los datos en un servidor, se hace difícil. O bien no existe el servidor capaz de manejar todos nuestros datos, o bien es muy cara. Lo que sucedió es que determinadas organizaciones empezaron a trabajar en nuevas bases de datos pensadas desde el inicio para enfrentarse a este tipo de información. Estas nuevas bases de datos están diseñadas para trabajar con grandes fuentes de información (Big Data) no estructuradas y/o cambiantes, y para escalar horizontalmente en vez de verticalmente. Pronto a alguien se le ocurrió acuñar un término para agruparlas: NoSQL [1]. Las bases de datos NoSQL son bases de datos no relacionales optimizadas para modelos de datos sin esquema y de desempeño escalable. También son ampliamente conocidas por su facilidad de desarrollo, baja latencia y resiliencia. Utilizan una variedad de modelos de datos, como los almacenes de valor clave

en memoria, de gráficos, de documentos y en columnas. Las bases de datos NoSQL son ideales para muchas aplicaciones web, de Big Data y para dispositivos móviles que necesitan un mayor nivel de escala y capacidad de respuesta que las bases de datos relacionales tradicionales. Debido al escalado horizontal y a las estructuras de datos más simples, las bases de datos NoSQL normalmente responden más rápido y son más fáciles de escalar que las bases de datos relacionales. En una base de datos NoSQL, los datos se pueden consultar de manera eficiente en un número limitado de maneras, fuera de los cuales es más costoso y lento hacer consultas. [13].

Bases de datos no relacionales NoSQL.

Durante muchos años los servicios y aplicaciones de Internet se han apoyado en las bases de datos relacionales, que funcionan muy bien con fuentes de información que generan datos muy bien estructurados (no variantes). Pero pronto se observó que estas bases de datos relacionales tenían ciertas carencias a la hora de enfrentarse a aplicaciones que generan Big Data. El principal problema se presentaba a la hora de mantener el rendimiento conforme aumentaba la cantidad de datos. Es decir, no escalan muy bien a partir de cierto volumen. En el mundo relacional, en general se escala verticalmente (aumentando recursos del servidor), pero cuando hay que hacerlo horizontalmente (añadiendo servidores) pueden aparecer problemas de rendimiento. Cuando una aplicación maneja Big Data, es conveniente escalar

4. MATERIALES Y MÉTODOS

El proyecto empezó a ejecutarse el 17 de octubre del año 2019 en la Universidad de Nariño.

Se utilizaron los siguientes insumos como material base para el diseño y construcción del proyecto:

- a. Se hizo uso de la base de datos de los egresados de ingeniería de sistemas de la Universidad de Nariño, como punto de partida para la creación de una base de datos no relacional en MongoDB.
- b. Para la implementación del motor de la base de datos no relacional se hizo uso de la documentación pública disponible de mongoDB.
- c. Documentación de NodeJs como entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor basado en el lenguaje de programación JavaScript.
- d. Documentación de Docker que automatiza el despliegue de aplicaciones dentro de contenedores de software, proporcionando una capa adicional de abstracción y automatización de virtualización de aplicaciones.
- e. Documentación de Express como marco de aplicación web de back-end para Node.js
- f. Documentación de GraphQL como lenguaje de consulta y manipulación de datos para APIs, y un entorno de ejecución para realizar consultas con datos existentes.
- g. Documentación del gestor de dependencias y instalador de paquetes Yarn
- h. Documentación de NuxtJs como una biblioteca JavaScript gratuita y de código abierto basada en Vue.js, Node.js, Webpack y Babel.js.
- i. Documentación de Apollo Client como una biblioteca completa de administración de estado para JavaScript que le permite administrar datos locales y remotos con GraphQL.

Inicialmente se hizo un levantamiento de requisitos para poder establecer los requerimientos funcionales y no funcionales de la aplicación, y un estudio de antecedentes.

Después de haber leído cada una de las documentaciones nombradas anteriormente se procede a la instalación y configuración del entorno de desarrollo, como lo es la instalación de nodejs, la instalación de docker, la instalación de mongoDB y la instalación de dependencias.

Haciendo uso de estas herramientas, se hace la conexión con la base de datos, de la previa documentación de graphql, se hacen todas las peticiones a la base de datos.

Todo esto, se encuentra establecido de manera ampliamente detallada en el manual del programador, haciendo referencia al paquete API del repositorio, al paquete API se puede acceder a través del siguiente enlace: <https://charlitoro.notion.site/Paquete-API-6bad08c9f2c34089b4dc26c428fbc530>.

Después de esto se hace la configuración del nuevo paquete que funciona como interprete entre la API (Backend) y quien lo utiliza (Ripuy o cliente Web) por medio de Apollo Client; la explicación en detalle de esta fase del proceso de construcción del proyecto se encuentra en el paquete requets, al cual se puede acceder desde el siguiente enlace : <https://charlitoro.notion.site/Paquete-Request-c0feb7d3a3f74ed093db9c29ce20bf7f>.

Por último, haciendo uso de la herramienta de NuxtJs, se inicia la construcción del front-end del aplicativo Ripuy, el cual se encarga de usar los datos suministrados por las queries a la base de datos, y crear las diferentes vistas que hacen parte de la app, en el repositorio se lo encuentra como el paquete web, al cual se accede a través del siguiente link : <https://charlitoro.notion.site/Paquete-Web-6ec6392fb6e248aaa1f43966e7a8c4bc>.

5. METODOLOGÍA

La investigación se desarrolló de forma descriptiva bajo el enfoque cuantitativo, aplicando un diseño no experimental. Esta metodología contempla varias fases por las que tuvo que atravesar el proyecto para su finalización, las cuales se describen a continuación:

Fase 1: Apropiación del conocimiento.

En esta fase se realizó una revisión documental exhaustiva sobre bases de datos NoSQL en general y particularmente sobre el gestor de bases de datos MongoDB que permitió a los investigadores apropiarse del conocimiento acerca de estas tecnologías. También se abordó una exploración de las tecnologías que se utilizaron para el diseño, desarrollo, pruebas y puesta en funcionamiento del aplicativo Ripuy.

Fase 2: Exploración y gestión del sistema de bases de datos NoSQL MongoDB.

Fue necesario relacionarse a fondo con el sistema de bases de datos NoSQL MongoDB, analizando la documentación oficial y diferentes medios de información, de modo que se llegó a una comprensión suficiente del gestor. Esto permitió la implementación de dichos conocimientos en el proyecto dentro de la aplicación Ripuy.

Además de haber hecho la revisión de la documentación, se hizo también una traducción al español de los principales aspectos de la información de MongoDB disponible a través de su página oficial, haciendo uso de la herramienta de Gitbook, de modo que dicha información podrá ser accedida de forma ordenada y fácil, para su uso en futuros proyectos.

Fase 3: Diseño e implementación de la base de datos de los egresados del programa de ingeniería de Sistemas de la Universidad de Nariño.

Respecto a la base de datos, se realizó todo el proceso correspondiente al diseño e implementación de ésta, empezando con el diagrama entidad relación que muestra cuales son las colecciones usadas para Ripuy tomando como base los datos reales y ficticias pre-existentes de los estudiantes egresados; seguido a esto, se realizó la construcción de la base de datos creando las colecciones y las relaciones entre ellas, y por último se pobló la base de datos.

Fase 4: Diseño y desarrollo del aplicativo Ripuy

Inicialmente se establecen los requisitos funcionales y no funcionales.

Seguido a esto, se crean los mockups que brindan una vista previa de las vistas que conforman la interfaz de la aplicación, poniendo un punto de partida para la construcción del código que conforma la lógica de cada funcionalidad, de modo que se cumplan los requisitos funcionales y no funcionales.

Finalizado el diseño, se creó un repositorio en gitlab que contiene todo el código correspondiente al desarrollo de la aplicación Ripuy para facilitar la continuidad y la construcción del código entre los desarrolladores.

Se construyeron los principales paquetes para garantizar el funcionamiento de Ripuy, los cuales son:

- a. api: Es el paquete que contiene toda la configuración de backend
- b. request: Aquí se configura un cliente que le permite al paquete web contar con las interfaces necesarias para poder consumir el API graphql del backend.
- c. web: Contiene todas las configuraciones del front-end, en este caso se hace uso del framework Nuxt JS

Fase 5: Construcción y realización de pruebas de funcionalidad del aplicativo Ripuy

Se diseñaron las pruebas unitarias correspondientes para garantizar el correcto funcionamiento de la interfaz, de modo que cumpla con criterios de usabilidad y funcionalidad suficientes; dichas pruebas tuvieron resultados positivos como se evidencia en este informe en la sección de adjuntos.

Se hicieron también test para probar la funcionalidad y los tiempos de respuesta de las peticiones en las consultas a la base de datos.

El aplicativo se desarrolló haciendo uso de las siguientes tecnologías:

- a. Scrum: como metodología de desarrollo
- b. MongoDB: como motor de base de datos

- c. Nuxt JS: Como framework de apoyo para el desarrollo de Frontend
- d. GitLab: para las versionamiento y el desarrollo colaborativo entre los desarrolladores
- e. Lerna: mejorando el flujo de trabajo entre los paquetes request y web del repositorio Ripuy.
- f. GraphQL: permite la comunicación entre la API y la base de datos, configurada dentro de un entorno de ejecución con los datos existentes.

6. RESULTADOS

Los resultados obtenidos para cada fase de la metodología aplicada son:

Fase 1: Apropiación del conocimiento.

Considerando la exploración, el análisis del marco teórico necesario y el estado del arte que hacen parte de esta investigación, se obtuvo como resultado la apropiación del conocimiento para definir cada uno de los ítems que conforman el presente documento.

Fase 2: Exploración y gestión del sistema de bases de datos NoSQL MongoDB.

- a. Se hizo la investigación de la información correspondiente al manejo e instalación de MongoDB contenida en la documentación disponible en la página oficial, la cual se encontraba disponible en idioma inglés.
- b. Se elaboró un documento con base en los principales aspectos dentro de la documentación oficial de MongoDB, en idioma español, haciendo uso de la herramienta GitBook, la cual facilita el acceso a dicha información de forma ordenada y legible, de modo que pueda ser usado por los estudiantes de Ingeniería de Sistemas como guía para la creación de otras bases de datos con MongoDB y en un futuro para la implementación de grandes bases de datos NoSQL en la Universidad de Nariño.
A GitBook se puede acceder mediante el link: <https://dev-group.gitbook.io/mongodb-docs/>

Fase 3: Diseño e implementación de la base de datos de los egresados del programa de ingeniería de Sistemas de la Universidad de Nariño.

- a. Se adquirieron los datos correspondientes a la información de los estudiantes egresados de la Universidad de Nariño, proporcionados por la misma institución, lo cual sirvió como base para el diseño de la base de datos que se implementó en el aplicativo Ripuy.
- b. Se hizo el diseño de la base de datos, basada en el esquema SQL preexistente que proporcionó la Universidad, siendo este reinterpretado y complementado en los diferentes documentos y colecciones que conforman la base de datos en MongoDB.
- c. Para la administración del modelo de la base de datos se utilizó la librería mongoose, posteriormente se usó el lenguaje de consulta como alternativa a las API REST como lo es GraphQL cuyo funcionamiento se hace a partir de los esquemas y los resolvers.
- d. Se generaron datos auxiliares que sirvieron como complemento para el poblamiento de la base de datos del aplicativo.

Fase 4: Diseño y desarrollo del aplicativo Ripuy

Diseño y desarrollo del aplicativo Ripuy que gestiona la información de egresados de Ingeniería de Sistemas, la base de datos NoSQL de egresados del programa de Ingeniería de Sistemas bajo el gestor MongoDB.

- a. Se establecieron los requisitos funcionales y no funcionales de la aplicación, los cuales están detallados en el siguiente enlace: <https://drive.google.com/file/d/19-nMSiN1M9by3RtMSAmkJV3EBR7eUbP8/view?usp=sharing>
- b. Se crearon los mockups que definen el diseño de interfaces de la aplicación usando la plataforma para la maquetación Figma.
- c. Se codificaron las vistas utilizando como base los mockups, usando para ello las funcionalidades que proporciona el framework Nuxt.
- d. Se hizo la implementación de un sistema de búsqueda intuitivo que complementa la funcionalidad de la aplicación en todas las vistas, aplicando varios filtros, de modo que se cumplan los requisitos de usabilidad para el usuario final.
- e. Se hizo el acoplamiento de la base de datos con las interfaces usando el paquete request dentro del paquete web, lo cual está especificado en el archivo readme del repositorio o también se encuentra en la página de notion en el Manual del programador ripuy.
- f. Se tiene el aplicativo RIPUY , para el cual se creó código almacenado en el siguiente repositorio: <https://gitlab.com/ripuy-mongo/ripuy-monorepo.git>
- g. Se creó el manual del programador, el cual está basado en los paquetes request y web. Al manual se puede acceder desde: <https://charlitoro.notion.site/Manual-Del-Programador-Ripuy-c3fbc385a9e44e30b10179a9bb2d7066>.
- h. Se creó el manual de usuario en el cual se esclarecen las funcionalidades del aplicativo Ripuy. Este documento se encuentra disponible en el siguiente enlace: <https://drive.google.com/file/d/1tYdh6pEc64a2bEfOpCCcSzpuRb5XG-I-/view>

Fase 5: Construcción y realización de pruebas de funcionalidad del aplicativo Ripuy

a. Pruebas de Rendimiento

Las pruebas de rendimiento son realizadas al servidor donde se encuentran los servicios de la base de datos y el api que conecta con el cliente web. Estas pruebas se realizan sobre un entorno de desarrollo que sea lo más similar al entorno de producción final de la aplicación.

Para el desarrollo de esta actividad se ha realizado búsqueda de servicios o software que permita automatizar estos test y permitan obtener reportes que se puedan analizar. En el proceso de búsqueda se encontró el software **Apache JMeter**. JMeter permitió configurar y realizar peticiones al servidor todos los encabezados necesarios y variables que se deben enviar para que las peticiones sean exitosas. Por otra parte, JMeter cuenta con muchas más herramientas para poder monitorear las pruebas al programar y automatizar pruebas. Pero lo más llamativo son los tipos de reporte que proporciona, tanto tabular como gráficos.

El tipo de pruebas de rendimiento que se van a implementar son las **Pruebas de Carga**. Este tipo de pruebas permitió observar el comportamiento de la aplicación bajo una cantidad esperada de peticiones.

Contexto de las pruebas

Los siguientes datos son los que se configuran en Jmeter simulando un ambiente normal como usuario egresado y administrador:

Configuración / Role	Egresado	Administrador
Número de usuarios Concurrentes	200	4
Periodo de Aceleración (Segundos)	50	10
Número de Bucles	3	5

Tabla No 1 Resultados de simulación de pruebas en Jmeter como egresado y administrador

La configuración para egresados indica que las pruebas se ejecutan 3 veces y que en cada iteración contará con 200 usuarios realizando peticiones al servidor donde estos 200 usuarios se irán conectando durante un tiempo de 50 segundos.

Para el caso de la configuración de administradores, al ser una cantidad de usuarios administrativos reducida solo se establecen 3 usuarios que estarán realizando peticiones y se irán conectando durante un tiempo de 10 segundos.

Resultados pruebas como rol Egresado

El grupo de pruebas para egresados tiene las siguientes peticiones:

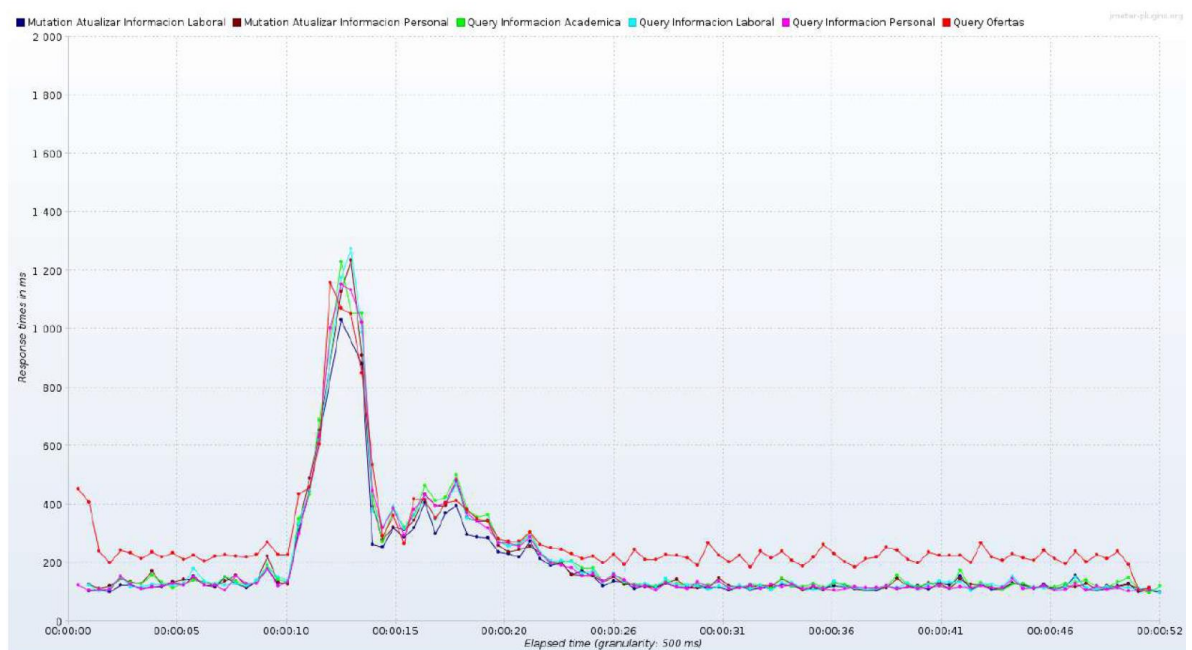
- Query Ofertas: Consulta las ofertas laborales publicadas
- Query Información Personal: Consulta la información personal del usuario en sesión
- Query Información Académica: Consulta la información académica del usuario en sesión
- Query Información Laboral: Consulta la información laboral del usuario en sesión
- Mutation Actualizar Información Personal: Solicitud de actualización de información personal del usuario en sesión
- Mutation Actualizar Información Laboral: Solicitud de actualización de información laboral del usuario en sesión

Del anterior grupo de peticiones se obtuvo la siguiente tabla:

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Egresado Requests:Query Ofertas	600	261	95	1183	182,77	0,00%	11,65501	63,86	11,06	5611
Egresado Requests:Query Informacion Personal	600	187	91	1233	150,76	0,00%	11,73525	6,75	12,07	589
Egresado Requests:Query Informacion Academica	600	189	92	1298	137,72	0,00%	11,73617	9,15	12,76	798
Egresado Requests:Query Informacion Laboral	600	181	92	1274	124,78	0,00%	11,73801	6,94	11,29	605
Egresado Requests:Mutation Atualizar Informacion Personal	600	176	92	1236	115,67	0,00%	11,74743	6,47	13,47	564
Egresado Requests:Mutation Atualizar Informacion Laboral	600	163	93	1032	94,95	0,00%	11,7541	4,43	12,37	386
TOTAL	3600	193	91	1298	140,9	0,00%	69,24675	96,4	71,85	1425,5

Tabla No 2 Resultados de pruebas como egresado

La anterior tabla nos presenta el número de peticiones y los tiempos de las respuestas y los porcentajes de efectividad. Como una mejor visualización JMeter proporciono la siguiente gráfica:



Gráfica No 1 Resultados de pruebas como egresado

De la gráfica se puede concluir que los tiempos de respuesta estuvieron en mayor medida sobre los 0.6 segundos y que las peticiones que tardaron más están aproximadamente por 1.2 segundos. Estos tiempos observados son menores al límite establecido en los resultados esperados que se establecieron entre 1 y 2 segundos.

Resultados pruebas como rol Administrador

El grupo de pruebas como usuario administrador tiene las siguientes peticiones:

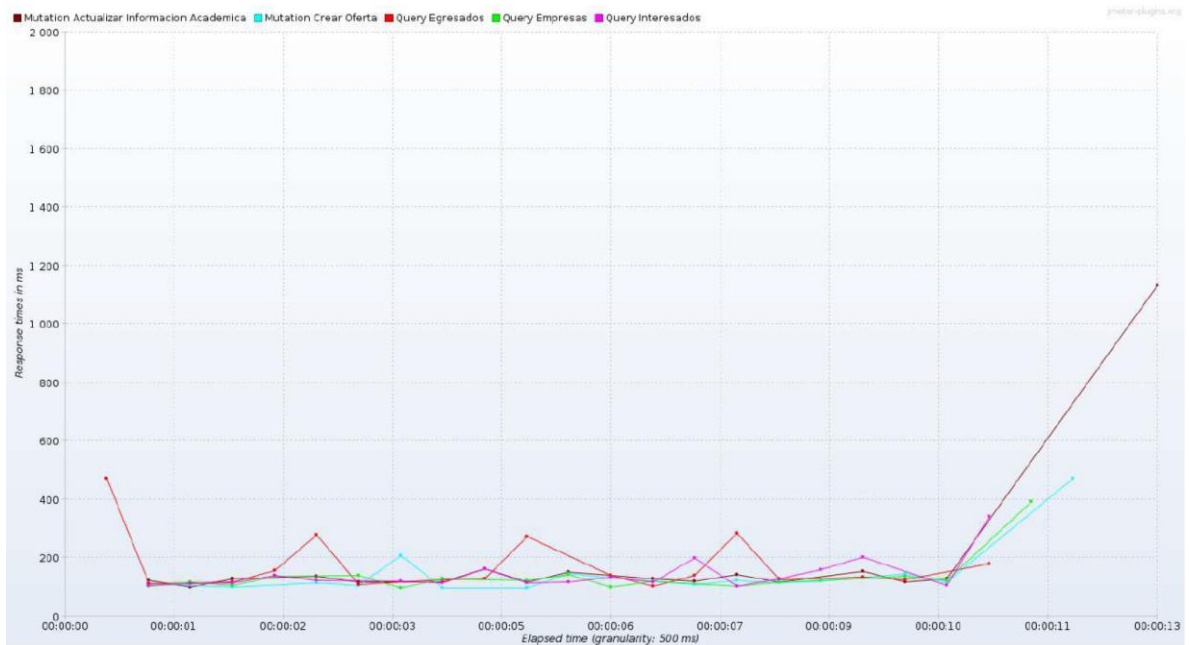
- Query Egresados: Consulta de la lista de egresados del sistema, como usuario administrador en sesión.
- Query Interesados: Consulta de la lista de egresados interesado en una oferta laboral, como usuario administrador en sesión.
- Query Empresas: Consulta de la lista de empresas registradas en el sistema, como administrador en sesión.
- Mutation Crear Oferta: Solicitud de creación de oferta laboral, como administrador en sesión.
- Mutation Actualizar Información Académica; Solicitud de actualizar la información académica de egresados, como administrador en sesión.

Del anterior grupo de peticiones se obtuvo la siguiente tabla

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Query Egresados	20	192	103	473	127,51	0,00%	1,8843	19,88	1,78	10805
Query Interesados	20	142	102	343	54,37	0,00%	1,90767	2,89	1,99	1552
Query Empresas	20	136	99	393	60,85	0,00%	1,85684	13,91	1,69	7671
Mutation Crear Oferta	20	139	98	473	80,65	0,00%	1,79533	1,24	2,22	705
Mutation Actualizar Información Académica	20	179	102	1132	219,09	0,00%	1,64528	0,95	2,08	594
TOTAL	100	158	98	1132	126,6	0,00%	7,71903	32,15	8,33	4265,4

Tabla No 3 Resultados de pruebas como administrador

La grafica correspondiente a los resultados obtenidos es la siguiente:



Gráfica No 2 Resultados de pruebas como administrador

De estos resultados se puede decir que los tiempos de respuesta son muy buenos, todos están por debajo de los 0,6 segundos, y como tendencia los tiempos están sobre los 0.3 segundos. Por otra parte, la petición de consulta de listado de egresados, marcada en rojo, es la que presenta picos que suben hasta cerca de los 0.4 segundos, esto es debido a la

cantidad de datos de prueba insertados para estas pruebas, el número de usuario de prueba que se consultan es de 59000 aproximadamente, la petición con más transferencia de información.

b. Pruebas funcionales.

Estas pruebas se desarrollaron utilizando el framework de testing .para javascript llamado **Jest JS**, las pruebas consisten en simular escenarios de usuarios que en condiciones normales cuáles serían las peticiones que se realizan al servidor. Dentro de estas pruebas se han realizado pruebas tipo Query para solo consultar información y consultar tipo Mutation para hacer modificaciones.

Algunos de resultados están configurados a través de los fixtures establecen la configuración de las respuestas, de lo que se espera obtener, respecto a la información académica, información general que sirve para llenar los componentes de la app, la información personal, profesional y de las ofertas laborales.

En el repositorio están los backups para restaurar los test, estos se ejecutan previniendo que se borren los datos después de cada ejecución, dentro de estos test hay varios archivos que tienen que ver con la conexión de mongoDB.

Como en lo anterior existen test para cada área dentro de la información ya sea académica, personal, profesional, etc, en donde se comprueba que las consultas devuelvan la información esperada.

También se generan errores de forma automatizada para comprobar la efectividad de las pruebas y la capacidad de los test para identificar los errores.

En total se crearon 78 test funcionales agrupados en archivos test diferentes para cada área de la información los cuales después de haber sido ejecutados, se comprobó que fueron aprobados la totalidad de los 78 test.

7. DISCUSIONES

- Es indispensable hacer un análisis de usabilidad de Ripuy que contemple todos aspectos referentes a la experiencia del usuario final en el uso del aplicativo con el fin de lograr obtener la correspondiente retroalimentación, y de esta forma poder realizar mejoras, mejorando así la eficiencia respecto al uso de RIPUY.

- RIPUY innova en cuanto al uso de mongoDB, ya que integra varias tecnologías en un entorno cloud de modo que se desarrollen los flujos de información de una manera eficiente, lo que se demuestra en los resultados de las pruebas de rendimiento de las consultas en cuanto a tiempos de respuesta obtenidos.

8. CONCLUSIONES

- Al término de este proyecto, se obtiene Ripuy , el cual es una aplicación web que gestiona la información de los egresados de la universidad de Nariño del programa de ingeniería de sistemas, haciendo uso de la tecnología de base de datos no relacional MongoDB; logrando culminar cada una de las fases del proyecto se da un primer paso para cada una de las necesidad y los objetivos planteados en un principio, obteniendo resultados con éxito. De este modo, se tiene el aplicativo y los manuales MongoDB para incentivar el uso del motor MongoDB y para futuras investigaciones en la Universidad de Nariño.
- Durante el desarrollo del proyecto se logró adquirir bastante conocimiento acerca de los temas planteados en los objetivos, los cuales abarcan tanto el manejo como la implementación de MongoDB, lo cual es un gran avance en cuanto a la formación personal y profesional de los responsables de éste proyecto.
- También se considera que los logros de este proyecto son un gran aporte al grupo de investigación GRIAS, para la universidad de Nariño y para la región en general, apoyando a los egresados de la institución y constituyendo un seguimiento y retroalimentación de sus datos como profesionales.
- Se consiguieron buenos resultados aplicando la metodología de SCRUM , lo cual permitió llevar un orden en cuanto a las tareas y a la asignación de las mismas, así como un uso eficiente de los tiempos de desarrollo a través de los sprints y de los diferentes procesos que involucró la construcción del software.
- El rendimiento del gestor MongoDB demostró ser superior a los gestores SQL relacionales en cuanto al uso de los recursos hardware en circunstancias limitantes y su flexibilidad facilita implementar nuevos modelos y modificaciones durante el desarrollo sin afectar los datos almacenados.
- El uso de NuxtJs como framework para el frontend fue de gran importancia ya que la construcción de componentes reutilizables hizo que se pueda hacer de manera más fácil y rápida cada de las vistas de RIPUY.
- Se corroboró que gracias a los resultados obtenidos en las pruebas funcionales y de rendimiento realizadas al aplicativo y al conjunto de herramientas que soportan su funcionamiento, que mongoDB tiene un desempeño muy positivo y confiable en cuanto a velocidad de consultas y al tratamiento y acceso a la información en general, lo cual lo hace fácilmente recomendable en futuras implementaciones en proyectos que requieran el manejo de grandes cantidades de información o que requieran un soporte a una alta cantidad de concurrencia de datos o de consultas.

Por último, se concluye que se logró la divulgación de los resultados logrados con este proyecto gracias a la inclusión del mismo haciendo parte del “Cuarto Congreso Andino en

Computación, Informática y Educación - CACIED 2019” con el póster titulado UNA APLICACIÓN DE BASES DE DATOS NoSQL PARA LA GESTIÓN DE INFORMACIÓN DE EGRESADOS DEL PROGRAMA DE INGENIERIA DE SISTEMAS DE LA UNIVERSIDAD DE NARIÑO en donde se obtuvo el reconocimiento del segundo lugar en la línea Minería de Datos y Big Data, que se evidencia revisando el siguiente enlace :

https://drive.google.com/file/d/1Uu44QLf3iIsqpV8p9ke_qzXxfDSjTlc5/view?usp=sharing y así mismo en el ”Quinto Congreso Andino en Computación, Informática y Educación - CACIED 2021” con la ponencia titulada ANÁLISIS DE DESEMPEÑO DEL GESTOR DE BASES DE DATOS NoSQL MONGODB PARA LA GESTIÓN DE INFORMACIÓN DE EGRESADOS DEL PROGRAMA DE INGENIERIA DE SISTEMAS DE LA UNIVERSIDAD DE NARIÑO, como se puede verificar en el siguiente link :
https://drive.google.com/file/d/1MFNALGojHgYXtXo60WubUH_EcHr7WL0l/view?usp=sharing

La Universidad de Nariño a través del grupo de investigación GRIAS del departamento de Sistemas dará a conocer un estudio sobre el uso de bases de datos no relacionales NoSQL utilizando herramientas de software libre como MongoDB, con el fin de elaborar una base de datos no relacional para almacenar los datos de los egresados del programa de ingeniería de sistemas, con el fin de incentivar su uso en instituciones educativas, empresas y organizaciones de la región y el país para la recolección y el tratamiento de datos masivos, para extraer conocimiento que ayude a la toma de decisiones relacionadas con el mejoramiento de la educación, como por ejemplo métodos de enseñanza más eficaces.

9. RECOMENDACIONES

- Es aconsejable crear una funcionalidad con una vista dentro de la interfaz para recibir sugerencias de los usuarios de modo que se complemente el feedback recibido, y se implementen mejoras en futuras actualizaciones.

- Se sugiere la implementación de pruebas con datos reales en producción, para reafirmar la solidez del comportamiento de la aplicación, debido a que las pruebas se realizaron con datos generados
- Se recomienda hacer la instalación de RIPUY en otras instituciones y en diversos entornos académicos y productivos, determinando así un avance y un aporte en cuando a Big Data se refiere.
- Se recomienda la divulgación activa y continua de la documentación que se obtuvo como resultado acerca de MongoDB y de la aplicación, los cuales fueron los resultados importantes del proyecto de investigación de modo que se pueda compartir el conocimiento en otras instituciones
- Es aconsejable incorporar tecnologías para el uso y la ejecución de pruebas automatizadas de modo que se puedan mejorar los tiempos de testeo y la calidad de los resultados y del feedback en general, en cuanto a los diferentes flujos que puedan existir dentro de la lógica de la aplicación; tecnologías tales como Selenium o puppeteer test, etc.
- Es importante considerar el uso de una herramienta para analizar la audiencia, la adquisición y el comportamiento de RIPUY, como lo es Google analytics la cual nos permitirá tener información agrupada del tráfico y de concurrencia de datos, además de obtener informes sobre las sesiones abiertas, duración de estas sesiones entre otras

REFERENCIAS

- [1] AULA301. MongoDB lo que deberías de saber antes de utilizarlo. [En línea]. Disponible en: <https://aula301.com/cuando-usar-mongodb/>. [Consultado:09/05/2018].
- [2] Paramio, C. (2011).El concepto NoSQL, o cómo almacenar tus datos en una base de datos no relacional.[En línea]. Disponible en: <https://www.genbetadev.com/bases-de-datos/el-concepto-nosql-o-como-almacenar-tus-datos-en-una-base-de-datos-no-relacional>. [Consultado:09/05/2018].
- [3] RUBENFA. (2014). MongoDB: qué es, cómo funciona y cuándo podemos usarlo. [En línea]. Disponible en: <https://www.genbetadev.com/bases-de-datos/mongodb-que-es-como-funciona-y-cuando-podemos-usarlo-o-no>.
- [4] Barranco,R.(2012). ¿Qué es Big Data? [En línea]. Disponible en: <https://www.ibm.com/developerworks/ssa/local/im/que-es-big-data/index.html>. [Consultado: 10/05/2018].
- [5] Lopez, D. (2012). Análisis de las posibilidades de uso de Big Data en las organizaciones. Universidad de Cantabria, Santander, España. [En línea]. Disponible en: <https://repositorio.unican.es/xmlui/bitstream/handle/10902/4528/TFM%20-%20David%20L%C3%B3pez%20Garc%C3%ADa.pdf?sequence=1>. [Consultado: 10/05/2018].
- [6] Abramova, V.Bernardino,J.,&Furtado,P.(2014).Experimental evaluation of NoSQL databases. International. Journal of Database Management Systems, 6(3),1.
- [7] Nayak,A.,Poriya,A.,&Poojary,D(2013).Typeof NOSQL databases and its comparison with relational databases. International Journal of Applied Information Systems, 5(4), 16-19.
- [8] Strauch,C.,Sites,U.L.S.,&Kriha,W(2011).NoSQL databases. LectureNotes,Stuttgart Media University.
- [9] Aguilar, L.J. (2013). Big Data - Análisis de Grandes Volúmenes de Datos en Organizaciones. México: Alfaomega Grupo Editor.
- [10] Zikopolous,P.,Deroos, D.,Deutsch,T. and Lapis, G. (2012). Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data. Editorial McGraw- Hill.
- [11] Fragoso,R.(2012). Que es Big Data.[En línea]. Disponible en: <https://www.ibm.com/developerworks/ssa/local/im/que-esbig-data/index.html>. [Consultado: 09-05-2018].
- [12] IIC (2016). Las 7 V del Big data: Características más importantes. Instituto de ingeniería del conocimiento. [En línea]. Disponible en: <http://www.iic.uam.es/innovacion/big-data-caracteristicas-mas-importantes-7-v/>. [Consultado: 10-05-2018].
- [13] AMAZON. (2017). ¿Que es NoSQL?[En línea].Disponible en: <https://aws.amazon.com/es/nosql/>.
- [14] Morales, A. (2017). Bases de datos NoSQL, MongoDB y GIS. [En línea]. Disponible en: <https://mappinggis.com/2014/07/mongodb-y-gis/#JSON>.

- [15] Ortego, D. (2017). Empresas que usan MongoDB. [En línea]. Disponible en: <https://openwebinars.net/blog/empresas-que-usan-mongodb/>
- [16] Rumayor, A., (2014). Modelando datos jerárquicos en bases de datos NoSQL. [En línea]. Tesis para obtener el grado de Maestro en Ingeniería de Software. Centro de Investigación en Matemáticas, A.C.-CIMAT. [En línea]. Disponible en: <http://cimat.repositorioinstitucional.mx/jspui/handle/1008/429>.
- [17] López, M. (2015). Herramientas de soporte analítico dentro de un entorno integrado MongoDB. [En línea]. Tesis de fin de Master en Ingeniería de Redes y Servicios Telemáticos. Escuela Técnica Superior de Ingenieros de Telecomunicación. Universidad Politécnica de Madrid. Madrid,
- [18] Róttoli, G., Zaffaroni, J., López, M. y Pollo-Cattaneo, M. (2016). Metodología para la Evaluación de Impacto de Migración entre Versiones de bases de Datos NoSQL. [En línea]. Universidad Tecnológica Nacional. Buenos Aires, Argentina. Disponible en: http://sedici.unlp.edu.ar/bitstream/handle/10915/52890/Documento_completo.pdf?sequence=1. [Consultado: 11-05-2018]. en: http://sedici.unlp.edu.ar/bitstream/handle/10915/52890/Documento_completo.pdf?sequence=1. [Consultado: 11-05-2018].
- [19] Moreno, F., Quintero, J. y Rueda, R. (2016). Una Comparación de Rendimiento entre Oracle y MongoDB. Ciencia e Ingeniería Neogranadina, 26 (1), pp. 109-129, DOI: <http://dx.doi.org/10.18359/rcin.1669>.
- [20] "Document MongoDB". [mongo. https://mongo.charlitoro.com/](https://mongo.charlitoro.com/) (accedido el 24 de octubre de 2021).
- [21] MongoDB, Inc. "MongoDB Versioning". [mongodb. https://docs.mongodb.com/manual/release-notes/5.0/](https://docs.mongodb.com/manual/release-notes/5.0/) (accedido el 24 de octubre de 2021).
- [22] MongoDB, Inc. "Release Notes for MongoDB 5.0". [mongodb. https://docs.mongodb.com/manual/release-notes/5.0/](https://docs.mongodb.com/manual/release-notes/5.0/) (accedido el 24 de octubre de 2021).
- [23] MongoDB, Inc. "RDBMS to MongoDB Migration Guide". [mongodb. https://www.mongodb.com/collateral/rdbms-mongodb-migration-guide](https://www.mongodb.com/collateral/rdbms-mongodb-migration-guide) (accedido el 24 de octubre de 2021).
- [24] F. Toledo, Introducción a las Pruebas de Sistemas de Información. Montevideo, 2014.
- [25] Performance Testing United. "PtU Certified Performance Tester con JMeter (CPTJM)". Brightest. https://assets.brightest.org/media/resources/PtU_Certified_Performance_Tester_with_JMeter_Syllabus_ES_Ver_1.0.pdf (accedido el 24 de octubre de 2021).
- [26] Departamento Nacional de Planeación. "Guia para Crear y Ejecutar Pruebas de Carga". dnp.

<https://colaboracion.dnp.gov.co/CDTI/Oficina%20Informatica/Sistemas%20de%20informacion/Guías%20Formatos%20Plantillas/Guia%20%20para%20Crear%20y%20Ejecutar%20Pruebas%20de%20Carga.pdf> (accedido el 24 de octubre de 2021).

[27] López, M. (2015). Herramientas de soporte analítico dentro de un entorno integrado MongoDB. [En línea]. Tesis de fin de Master en Ingeniería de Redes y Servicios Telemáticos. Escuela Técnica Superior de Ingenieros de Telecomunicación. Universidad Politécnica de Madrid. Madrid,

ADJUNTOS

1. Link de GitBook (Documentación MongoDB) <https://dev-group.gitbook.io/mongodb-docs/>
2. Acceso al repositorio del código fuente del aplicativo RIPUY <https://gitlab.com/ripuy-mongo/ripuy-monorepo.git>
3. Levantamiento de requisitos <https://drive.google.com/file/d/19-nMSiN1M9by3RtMSAmkJV3EBR7eUbP8/view?usp=sharing>
4. Manual del programador <https://charlitoro.notion.site/Manual-Del-Programador-Ripuy-c3fbc385a9e44e30b10179a9bb2d7066>
5. Manual de Usuario o de referencia <https://drive.google.com/file/d/1tYdh6pEc64a2bEfOpCCcSzpuRb5XG-I/-view>
6. UNA APLICACIÓN DE BASES DE DATOS NoSQL PARA LA GESTIÓN DE INFORMACIÓN DE EGRESADOS DEL PROGRAMA DE INGENIERIA DE SISTEMAS DE LA UNIVERSIDAD DE NARIÑO, Cuarto Congreso Andino en Computación, Informática y Educación – CACIED 2019 https://drive.google.com/file/d/1Uu44QLf3iIsqpV8p9ke_qzXxfDSjTlc5/view?usp=sharing
7. ANÁLISIS DE DESEMPEÑO DEL GESTOR DE BASES DE DATOS NoSQL MONGODB PARA LA GESTIÓN DE INFORMACIÓN DE EGRESADOS DEL PROGRAMA DE INGENIERIA DE SISTEMAS DE LA UNIVERSIDAD DE NARIÑO, Quinto Congreso Andino en Computación, Informática y Educación – CACIED 2021 https://drive.google.com/file/d/1MFNALGojHgYXtXo60WubUH_EcHr7WL0l/view?usp=sharing

