UNIVERSIDAD NACIONAL DE CAJAMARCA

FACULTAD DE INGENIERIA

ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA DE SISTEMAS

"APLICACIÓN DE BASES DE DATOS NO RELACIONALES NOSQL PARA LA MEJORA DEL ACCESO A LA INFORMACIÓN EN EL PROCESO DE SEGMENTACIÓN DE CLIENTES EN EL CENTRO DE ACTUALIZACIÓN PROFESIONAL PARA INGENIERÍAS CAPI"

TESIS

PARA OPTAR EL TITULO PROFESIONAL DE

INGENIERO DE SISTEMAS

PRESENTADO POR EL BACHILLER:

LUIS MIGUEL NEPTALÍ CHÁVEZ QUISPE

ASESOR

ING. AMALIA FERNÁNDEZ VARGAS

CAJAMARCA - PERÚ

2014

DEDICATORIA

Dedico el presente trabajo a mis padres, por el apoyo brindado y por estar siempre a mi lado en éste largo camino. A mis abuelos por sus sabios consejos y por ser un gran ejemplo para mí. A todos aquellos que me acompañaron en la realización de este trabajo

AGRADECIMIENTO

Estoy muy agradecido con mi familia, en especial con mi madre por el eterno apoyo incondicional y por estar siempre alentándome a seguir adelante. También agradezco a mis asesores la Ing. Amalia Fernández e Ing. Carlos Aparicio, por el apoyo brindado para el desarrollo de éste trabajo, sin ellos no hubiera sido posible el logro de ésta meta.

A todos, muchas gracias.

CONTENIDO

INDICE DE FIGURAS

INDICE DE TABLAS

INDICE DE GRÁFICOS

RESUMEN

El presente trabajo tiene como objetivo mejorar el acceso a la información en el proceso de segmentación de clientes en el Centro de Actualización Profesional para Ingenierías CAPI aplicando bases de datos no relacionales. En la actualidad en el área de marketing y ventas del CAPI cuenta con información importante de manera desorganizada la cual es gestionada de forma deficiente en el proceso de segmentación de clientes, en aspectos de automatización y tiempos de acceso a la información. Ésta información desorganizada se encuentra en las diferentes aplicaciones web con las que el CAPI cuenta, pero que aún no es explotada y gestionada; por lo cual, como solución se planteó la aplicación de bases de datos no relacionales la cual permita almacenar éste tipo de información y que permita mejorar el acceso a la misma teniendo como beneficios información procesada y precisa a la hora de ofrecer servicios y filiales con más tendencia de clientes; todo esto a través de una aplicación web para el usuario final. Para el desarrollo de la solución, se propuso una metodología propia la cual contiene etapas que están basadas en etapas de otras metodologías que son tradicionales como la de Ralph Kimball y no tradicionales como la de Minería Web. Las etapas importantes de la metodología son el Diseño Físico, en la cual se incluye: un análisis comparativo de los distintos motores de base de datos NoSQL para la respectiva elección, una explicación general del funcionamiento de la base de datos elegida, el diseño de la base de datos y la manipulación de datos de la base creada. Otra etapa importante es la construcción de la aplicación, en la cual se ven proceso de extracción y carga de los datos a la base; y posteriormente la implementación de la aplicación final para el usuario interno. Como resultados se demostró que la aplicación de bases de datos no relacionales si mejora el acceso a la información en el proceso de segmentación de clientes, obteniendo diferencias notables en los resultados de indicadores de nivel de satisfacción de usuarios, automatización y tiempos de acceso a la información.

Palabras clave: Bases de datos no relacionales, acceso a la información, NoSQL, segmentación de clientes, aplicación web, Minería Web y datos no estructurados.

ABSTRACT

This work aims to improve access to information in the process of customer segmentation in the Centro de Actualización Profesional para Ingenierías CAPI applying non-relational data bases. At present in the area of marketing and sales CAPI's has important information in a disorganized manner which is managed poorly in the process of customer segmentation in aspects of automation and access times to information. This information is disorganized in the different web applications that CAPI account, but that is not operated and maintained; thus, as solution application database non-relational data which allows to store this kind of information and allow better access to the same light as benefits information processing and accurate in providing services and subsidiaries more was raised customer trend; all through a web application for end users. To develop the solution, a methodology which contains steps that are based on other stages are traditional methodologies such as Ralph Kimball and nontraditional as Web mining is proposed. The important stages of the methodology are the Physical Design, which includes: a comparative analysis of different engines NoSQL database to the respective election, a general explanation of how the chosen database, design database and data manipulation of the database set. Another important step is to build the application, in which the extraction process and load data are based; and then implementing the ultimate application for the internal user. As results showed that the application of non-relational data bases if improved access to information in the process of customer segmentation, obtaining significant differences in outcome indicators of user satisfaction, automation and access times to information.

Key Words: Non-relational data bases, access to information, NoSQL, customer segmentation, web application, Web mining and unstructured data.

CAPÍTULO I. INTRODUCCIÓN

1. INTRODUCCIÓN

El 80% de la información de una empresa reside en contenidos no estructurados o no relacionales, vitales para las operaciones diarias de los departamentos más estratégicos de la empresa, desde marketing e I+D hasta recursos humanos y finanzas. Una investigación sobre la gestión de éste tipo de contenidos, de la cual entre las principales conclusiones del estudio, destaca el desconocimiento generalizado por parte de las empresas sobre el alcance real de éstos contenidos como herramienta competitiva para mejorar el servicio al cliente e impulsar la productividad.

Actualmente el acceso a la información para el proceso de segmentación de clientes se viene dando de manera deficiente en el Centro de Actualización Profesional para Ingenierías - CAPI, la empresa ofrece servicios de actualizaciones académicas que van de acuerdo al tipo de profesión de los clientes potenciales pero al momento de obtener información crítica o importante para la captación de éstos no se está tomando en cuenta preferencias y tendencias que tendrá el consumidor del servicio, considerado que el mismo debe ofrecerse de acuerdo a dichas preferencias y tendencias, y no sólo por instinto humano. CAPI almacena de forma desorganizada información digital que no tiene una estructura tradicional como correos electrónicos, imágenes, comentarios de facebook, twitts, etc., la cual no es explotada y utilizada para algún proceso específico; ante éste problema el presente trabajo de investigación plantea aplicar bases de datos no relacionales provenientes de aplicaciones web, que permita mejorar el acceso a la información en el proceso de segmentación de clientes: exprimir toda la información que posea y mejorar en la toma de decisiones a la hora de ofrecer un servicio.

El alcance de ésta investigación se tomarán en cuenta los datos recolectados de las aplicaciones web como redes sociales y portal web para luego almacenarlos en una base de datos NoSQL, que posteriormente pueda ser analizada; con la finalidad de

mejorar el acceso a la información en el proceso de segmentación de clientes en CAPI.

Este trabajo de tesis propone la aplicación de bases de datos no relacionales provenientes de las redes sociales en el Centro de Actualización Profesional para Ingenierías CAPI, que sea capaz de almacenar información importante, con esto se tendría un punto de partida estructurado y definido para posteriormente en un futuro y al contar con dicha información almacenada, se pueda desarrollar un sistema que analice secuencias de éstos datos para ayudar en la mejora del proceso de segmentación de clientes; es decir para entender las preferencias y tendencias de los consumidores acerca de servicios y marcas. La importancia de análisis de datos no relacionales NoSQL es la variedad de tópicos y la información que permite analizar. Esto va a permitir contar con información más actualizada determinando de esta manera comprender mejor al cliente, respecto a la información y tomar las medidas correctivas más exitosamente y generará mayores ingresos, permitiendo el desarrollo de la empresa de proyección local, nacional e internacional.

El objetivo general de la investigación es mejorar el acceso a la información en el proceso de segmentación de clientes en el Centro de Actualización Profesional para Ingenierías CAPI aplicando bases de datos no relacionales provenientes de aplicaciones web. Partiendo de él tres objetivos específicos: el primero es describir el proceso actual de acceso a la información para la segmentación de clientes, con la finalidad de identificar aspectos importantes que ayuden a nuestra propuesta. El segundo objetivo es examinar los datos no relacionales o no estructurados de diferentes aplicaciones web para tomarlos en cuenta en la aplicación de bases de datos no relacionales NoSQL. Y finalmente aplicar bases de datos no relacionales, para mejorar el acceso a la información en el proceso de segmentación de clientes en el Centro de Actualización Profesional para Ingenierías CAPI.

Éste trabajo de investigación se ha dividido en 5 capítulos, cuya estructura de resume a continuación:

El Capítulo II constituye la presentación de antecedentes teóricos que existen sobre nuestra investigación. Además se presentan bases teóricas respecto a el mundo relacional: bases de datos relacionales y bases NoSQL, minería web, metodologías tradicionales y nuevas; las cuales servirá como base con el fin de presentar una posible alternativa de nuestra propia metodología propuesta. Finalmente se

presentan una serie de definiciones de términos importantes para entender mejor nuestro trabajo.

El Capítulo III, está constituido primero por una primera parte que es la propuesta de una metodología para el desarrollo de nuestra investigación, detallando etapas y sus respectivas sub-etapas con el sentido de cumplir cada una de ellas. Posteriormente se aplica el desarrollo de ésta metodología de acuerdo a lo propuesto en éste trabajo de investigación. La segunda parte de éste capítulo lo componen los exámenes de diagnósticos Pre y Post Test, los cuales se basan en los indicadores de las variables, con el fin de mostrar la situación del proceso actual y posteriormente la situación con la aplicación implementada.

El Capítulo IV, está compuesto por dos partes el análisis de resultados y la discusión de los mismos. El primero muestra la comprobación de hipótesis de cada indicador considerados de manera estadística. La segunda parte la conforma una discusión de los resultados por cada objetivo planteado con la finalidad de comprobar su cumplimiento.

El Capítulo V, está conformado por las conclusiones y recomendaciones del presente trabajo de investigación, mostrando el logro de los objetivos planteados.

CAPÍTULO II. MARCO TEÓRICO

2. MARCO TEÓRICO

2.1. Antecedentes Teóricos

2.1.1. Antecedentes Internacionales

Brito, D. [1], en su tesis de pregrado, propone el uso de bases de datos NoSQL como alternativa a las bases relacionales. La autora señala que las bases de datos NoSQL han sido creadas con un enfoque diferente a las bases de datos relacionales tradicionales, mientras que las relacionales buscan cada vez mejorar temas como transaccionalidad, características ACID, soporte para triggers, procedimientos almacenados, etc. Mientras que las NoSQL se han concebido como soluciones que ofrezcan altas velocidades, simplicidad, escalabilidad a costa de algunas características comunes a las bases de datos relacionales. Aunque los tradicionales defensores de las bases de datos relacionales generalmente no le den la importancia que las nuevas herramientas merecen, se debería prestarles la debida atención, pues si empresas como Facebook, Twitter, Google, Amazon, etc. que son gigantes de la informática y tienen millones de usuarios suscritos a sus servicios y cuyo principal activo lo constituyen los datos de sus usuarios; han encontrado útiles las herramientas NoSQL debe ser porque cumplen con sus objetivos de excelente manera. La autora en su trabajo realiza un análisis comparativo de los sistemas de bases de datos NoSQL para determinar la mejor opción a de acuerdo a escenarios específicos.

La autora concluye que no debemos dejar de lado también la posibilidad de implementar soluciones hibridas que usen bases de datos relacionales y NoSQL en conjunto, por ejemplo Facebook usa MySQL para ciertos datos y Cassandra para cubrir otros requerimientos.

Barragán, A. y Forero, A. [2], en su trabajo de investigación desarrollan la implementación de una base de datos NoSQL para la generación de una matriz Origen/Destino. Los autores buscan generar una solución práctica y eficiente para apoyar el sector del transporte, concretamente el sector encargado de la planeación de transporte y en específico con herramientas asociadas a la generación de la matriz origen destino, utilizada para el análisis y planeación de tráfico en una ciudad, todo esto mediante un enfoque NoSQL, que ayude a la organización y gestión de toda la información recolectada en el de transporte urbano, logrando contribuir con la creación de nuevas rutas, mejor calidad de servicio y reducción de costos. Los autores en su trabajo mencionan que éste tipo de almacenamiento no pretende desplazar a las bases de datos relacionales, si no que sirve como una herramienta útil en ciertos entornos donde se busque velocidad y rendimiento. Los autores también señalan que la planificación del transporte desempeña un papel muy importante en la sociedad. En la actualidad es un factor determinante para el futuro de las grandes metrópolis. En esta planificación, la matriz Origen/Destino, es la estructura base para la planeación de rutas y trafico dentro de un sistema de transporte, por este motivo es una de las soluciones que más se utiliza en el campo del transporte. El solo hecho de plantear nuevos enfoques de gestión de información como el NoSQL para recolectar los datos de dicha matriz, permite gestionar de forma más eficiente la planeación de tráfico en la ciudad. En este sentido, su trabajo de investigación utiliza un enfoque diferente a los utilizados, específicamente para el almacenamiento y la organización de información, las bases de datos no relacionales, las cuales proporcionan tiempos de respuesta mucho más bajos que las bases tradicionales, además de sus ventajas en escalabilidad y disponibilidad del sistema. Finalmente la opción de motor de base de datos escogida para éste trabajo de investigación, de acuerdo al problema propuesto fue una base de datos de orientadas a grafos.

Los autores concluyen: las bases de datos no relacionales, pueden ser de gran utilidad en muchos sectores y áreas, actualmente están teniendo un auge en cuanto a las redes sociales, debido a la cantidad de información generada. Pero es importante resaltar que el trabajo con bases de datos de NoSQL requiere, en la mayoría de los casos, conocer bien el negocio que se desea modelar para definir adecuadamente la estructura en la que se van a

almacenar los datos. Un esquema de datos bien ajustado a un negocio muy específico permite optimizar los resultados de las consultas desde la etapa de diseño. Es importante mencionar que el resultado de este trabajo de investigación es la generación de un modelo propio de este tipo de bases transformado desde un modelo relacional a uno no relacional por medio del proceso de desnormalización, siendo esto un logro debido a la dificultad encontrada en documentación y experimentación de estos motores, además de ajustarse al tema principal del caso de estudio, puesto que según los autores tenían poco conocimiento del mismo.

Mancilla, S. [3], en su trabajo de graduación propone el uso de bases de datos documentales NoSQL para crear sitios web de alto rendimiento, éste trabajo consiste en la investigación de los conceptos necesarios, para realizar el análisis de rendimiento de las bases de datos NoSQL documentales y de tipo relacional, a través de pruebas de estrés que permitan la obtención de datos para análisis e interpretación. El autor considera en primer lugar los factores necesarios a considerar para la implementación de una base de datos NoSQL documental, entre los cuales se encuentran: el grado de adaptación, la facilidad de aprendizaje, grado de facilidad de implementación y migración de datos. Luego al autor enumera los costos de implementación de una base de datos documental NoSQL, entre los cuales se encuentran los costos de aplicación como: licencias, capacitación, migración, entre otros, los costos de infraestructura en el caso que se usen varios servidores. Finalmente describe las pruebas de estrés correspondientes a la base de datos NoSQL y a las relacionales, para medir y analizar el rendimiento de ambos tipos, tomando en cuenta factores como: tiempo de respuesta, consumo de recursos, escalabilidad

El autor concluye que es posible el desarrollo de sitios web de alto rendimiento utilizando base de datos NoSQL documentales, se han obtenido beneficios sustanciales como la administración de contenido, entretenimiento, redes sociales o para mostrar información en tiempo real. También concluye que las bases de datos NoSQL documentales no poseen seguridad o integridad en los datos, es por esta razón que no se deben utilizar en transacciones importantes, por ejemplo, transacciones bancarias.

López, C. [4], en su trabajo de pregrado Análisis de las bases de datos NoSQL propone una alternativa a las bases de datos SQL a los sistemas de gestión de bases de datos que utilizan las empresas en Medellín, de forma que puedan ser competitivas y hagan uso de las tecnologías emergentes que abordan las necesidades de hoy en día. Para lograr esta propuesta, se realizan varias encuestas a personas que estén relacionadas con la adquisición, operación y mantenimiento de las bases datos en organizaciones con alto grado de madurez y que se encuentren en la ciudad de Medellín, de tal forma que permitan conocer las necesidades y requisitos organizacionales para posteriormente optar por una opción de bases de datos NoSQL. Para éste trabajo de grado se seleccionaron las bases de datos NoSQL más relevantes, correspondientes a: Cassandra, MongoDB y Neo4j. Sobre éstas se investigaron sus características de acuerdo a varios factores como la mantenibilidad, características de la máquina, instalación e implementación, usabilidad, versión, soporte, madurez, documentación y características adicionales. El objetivo del autor es analizar características existentes en una base de datos NoSQL que permitirían a una empresa utilizarlas en vez de una base de datos SQL, el cual se cumplirá después de la aplicación de encuestas a diferentes empresas en Medellín relacionadas con las características que requieren a la hora de adoptar los sistemas de gestión de bases de datos, luego se procederá a proponer una alternativa de base de datos NoSQL. La encuesta consta de 16 preguntas basadas en su mayoría en conocer qué bases de datos utiliza la empresa para la que trabaja el encuestado y las actividades que involucra poseerlas, como mantenimiento, políticas de uso, requisitos, entre otras; y la parte restante, en decisiones sobre el uso de los motores relacionadas también con las políticas de la empresa y conocimiento breve de NoSQL.

El autor concluye: de acuerdo a lista de chequeo de las características que deberían tener las bases de datos NoSQL sobre cuál deberían utilizar las empresas: DynamoDB, Cassandra y MongoDB están más concentradas en ofrecer funcionalidades generales, además se diferencian las modalidades de adquisición de los tres motores considerados para este trabajo de grado, desde gratuita hasta pagada, considerando también las dos licencias de Neo4j; se debe tener en cuenta que la adquisición está asociada al valor agregado y características adicionales que los distribuidores ofrecen de sus bases de datos. Depende entonces de las políticas internas de la empresa,

así como de la orientación a la solución que se necesita, para hacer un buen uso de los sistemas NoSQL. Igualmente se debe tener entendimiento en varios lenguajes de programación, ya que la mayoría son compatibles con estos motores y una buena gestión de recursos para embarcarse en los nuevos modelos de las bases de datos y sacarle provecho a las ventajas que conllevan. También Finalmente resalta que introducir un sistema NoSQL, al igual que cualquier otro componente a la arquitectura de una empresa, exige una ardua gestión del cambio y un proceso constante de adquisición de conocimiento por parte de los empleados y consideración de los riesgos.

Ropero, J. [5] en su tesis doctoral expone una metodología de extracción de Información basado en el Uso de Lógica Difusa. El autor en primer lugar expone el planteamiento de su problema el cual radica que en la gran cantidad de información disponible en la actualidad provocada por el auge de las Tecnologías de la Información constituye una enorme ventaja para las necesidades de búsqueda de esta por parte de los usuarios de las nuevas tecnologías. Sin embargo, al mismo tiempo, surge también un gran problema derivado de la dificultad existente para distinguir la información necesaria de entre toda la enorme cantidad de datos innecesarios. Luego presenta conceptos relacionados con la minería de datos y la minería web; donde muestra una metodología para el uso de la minería web la cual tiene pasos, de acuerdo a éstos conceptos también presenta un Modelo de Espacio Vectorial, en el cual está basado el diseño del Agente Inteligente o Asistente Virtual que realiza la Extracción de la Información. Cómo es que éste Agente necesita interactuar con los usuarios en Lenguaje Natural, también se introducen las técnicas mediante las cuáles éste es procesado. Finalmente el autor presenta un método general basado en Lógica Difusa para la extracción de conocimiento en entornos en los que la información relevante es difícil de distinguir de la que no lo es.

El autor concluye y expone el principal aporte de su tesis es la confección de un método general de búsqueda de conocimiento mediante el uso de un Agente Inteligente basado en la Lógica Borrosa. El problema que surge es que, en la actualidad, la web no provee aún de un gran número de ontologías o esquemas: hay pocas disponibles y en muy pocas materias. Es más,

construir una ontología desde el principio puede resultar una tarea costosa y muy dependiente del ingeniero de conocimiento que la desarrolle.

De acuerdo con Gallardo L., Bermeo F. y Cedeño V. [6], en su trabajo de investigación publicado en un artículo; Sistema de reportes y análisis sobre tendencias en la Web de la ESPOL usando Hadoop para el procesamiento masivo de los datos, exponen: en las cuentas de Facebook se obtienen una gran cantidad de palabras las cuales se repiten muchas veces, para este tipo de casos usan el WordCount, para que los agrupe en una sola palabra y el número de repeticiones de la misma. El WordCount es una sencilla aplicación que se encarga de analizar un texto, señalando en una lista: las palabras utilizadas y el número de repeticiones existentes, el formato de los datos es: como entrada un archivo de texto plano (.txt) y como salida la palabra que es la clave y el número de repetición que es el valor. La implementación de WordCountMapper, vía el método mapper, toma cada línea del archivo de texto, según lo dispuesto por el TextInputFormat. La línea es dividida en palabras y emite pares clave/valor (palabra, 1). La implementación del WordCountReducer, vía el método reducer, recibe la salida del mapper que contiene todas las palabras existentes en el archivo de entrada y cuenta la frecuencia de ellas emitiendo un nuevo par clave/valor (palabra, número de repetición). Por último se almacenan todos estos resultados en una base de datos para luego ser leídos y usados para mostrar los reportes estadísticos. La presente investigación nace como una iniciativa que busca explorar la viabilidad de la aplicación de los sistemas de reportes y análisis sobre tendencias en la web de la universidad, y hallar un escenario para la implementación de este tipo de sistemas dentro de ella.

Los autores concluyen que su trabajo está basado en MapReduce, que es un modelo de programación diseñado para procesar grandes volúmenes de datos y que conceptualmente transforma listas de elementos de datos de entrada en listas de elementos de datos de salida. Un programa de éste tipo hará esto dos veces, usando dos diferentes lenguajes de procesamiento de listas: Map y Reduce, estos términos son tomados de varios lenguajes de procesamiento de listas, tales como LISP, Scheme o ML.

De la Rosa, F. [7] en su Tesis Doctoral expone un sistema de inteligencia web enfocado a análisis de redes sociales. Este trabajo se centra en el estudio de los sistemas de VTIC (Vigilancia Tecnológica e Inteligencia Competitiva) que utilizan fuentes de información Web (Sistemas de Inteligencia Web, SIW) que tienen muchas similitudes con los sistemas VTIC, pero que tienen que afrontar una serie de problemas producidos por el tipo de fuente de información utilizada. Los principales problemas que plantean las fuentes de información web es que son heterogéneas y dinámicas (cambian en periodos cortos de tiempo).

El autor menciona que la principal aportación de su trabajo es el desarrollo de un esquema algorítmico que combina la extracción de redes sociales con la búsqueda de información. De esta forma se define un marco de trabajo que permite especificar fácilmente las necesidades informativas de los usuarios, extrayendo información que no son cubiertas por las bases convencionales, mediante la implementación de sistemas de búsquedas dirigidos por heurísticas.

McKinsey Global Institute [8], en su informe, señala que Big Data ha logrado relevancia en los últimos tiempos ya que las empresas se están dando cuenta de que puede ser y es una mina de oro para encontrar una ventaja competitiva respecto a la competencia. La consultora identifica formas distintas a través de las cuales se puede utilizar Big Data para crear valor, pero sólo una menciona a los clientes y lo hace para hablar de mejoras en la segmentación de los consumidores. También muestran varios casos de éxito de distintas marcas en sus blogs sobre Big Data, pero se centra casi exclusivamente en cuestiones operativas, gestión de procesos y otros aspectos que mejoran la eficiencia. La eficiencia es un objetivo claro que vale la pena perseguir pero desde la óptica de un cliente, el uso de Big Data tiene mucha más relevancia en el terreno de los contenidos o del servicio al cliente. Ahora que los consumidores han visto lo que los medios sociales y la personalización en masa son capaces de hacer, cada vez más esperan estas formas de contacto desde sus marcas preferidas; no sólo son usuarios pasivos que están a la espera de recibir un mensaje, son participativos.

Los autores del informe concluyen que cuando se aplica al entorno de negocios o de marketing, toda la conversación sobre Big Data gira en torno a las tendencias del consumidor, la planificación de nuevos productos y otros insights del mercado.

Linthicum, D. [9], en un reporte especial dedicado a Big Data, trata de dar respuesta a la pregunta: *¿Cómo se tratan los datos no relacionales?* El reporte expone que los datos no relacionales y relacionales se recogen en un sistema de archivos (como ejemplo toma a Hadoop Distributed File System, o HDFS). Luego estos datos se almacenan en bloques en los distintos nodos en el clúster Hadoop. Seguidamente el sistema de archivos crea muchas repeticiones los bloques de datos, la distribución de ellos en el grupo de seguro formas que se pueden recuperar más rápidamente, los tamaños de los bloques puede variar, pero un tamaño típico HDFS bloque es de 128, y se replican a múltiples nodos en el clúster. Nos ocupamos sólo con los archivos, lo que significa que el contenido no se adhiere a una estructura antes de que exista en el archivo sistema. Los mapas de datos se aplican entonces sobre el contenido no relacional para definir los metadatos de núcleo para dicho contenido. Se pueden asignar y reasignar cualquier número de veces para apoyar a los cambiantes requisitos de los metadatos de las herramientas analíticas u otras personas que aprovechan los datos. En algunos casos, se emplea Hadoop Hive. Hive es un sistema de almacenamiento de datos que proporciona el resumen de datos, consultas ad-hoc y análisis de grandes conjuntos de datos almacenados en el clúster Hadoop. Hive proporciona un mecanismo para estructura del proyecto sobre estos datos y para consultar los datos utilizando un lenguaje SQL, llamado HiveQL. Otra opción para tratar datos no relacionales es Pig Apache. Pig es una plataforma de alto nivel para la creación de programas utilizados con Hadoop MapReduce. Se abstrae la programación del motor MapReduce. Como Hive, Pig utiliza su propio lenguaje para interactuar con los datos

Los autores concluyen que, las herramientas puede tener un enfoque diferente. Muchas de las herramientas utilizan mappers o asignadores que hacen que los datos aparecen como si se almacenan en una base de datos relacional tradicional. Otros toman ventaja de las características nativas de la tecnología de datos grandes, incluyendo la capacidad de tratar los datos

relacionales y no relacionales de manera diferente dentro de los modelos analíticos.

Maureen, L., Espinoza, O. y Núñez, K. [10], en su trabajo de investigación titulado Segmentación basada en el valor de cliente expone que existen varias técnicas de segmentación que permite identificar una amplia gama de perfiles de clientes, tanto actuales como potenciales. Asimismo, realizar las estrategias de marketing y comunicación eficientemente. En consecuencia, el mayor conocimiento de los clientes ayuda a definir una cartera diferenciada de clientes entre los de mayor y menor valor para la empresa. El autor menciona que para conocer, analizar y evaluar las necesidades, preferencias, motivaciones y comportamiento de compra de los clientes ha comenzado a ser un tema relevante para las empresas, especialmente, porque han tenido que gestionar grandes volúmenes de información con relación a su cartera de clientes. Para ello, la utilización de nuevas tecnologías de la información y las comunicaciones (NTIC's) ha sido un factor clave para el desarrollo de los procesos de gestión del conocimiento y de la relación que mantienen las empresas con sus clientes. Las diferentes tecnologías de información como las bases de datos, software de análisis, multimedia, entre otros, en conjunto con el desarrollo de la inteligencia de negocios han abierto importantes expectativas que antiguamente no existían. Por otra parte, procesos tecnológicos de captación y almacenamiento de datos como el DataWarehouse y el Data Mining permiten a las organizaciones diseñar productos que cubran las necesidades esperadas de cada cliente. En conjunto, nuevos medios informáticos como la World Wide Web permiten agilizar la comunicación y procesos de venta de cualquier organización con sus clientes de una manera personalizada. Sin embargo para la correcta implantación de un sistema de gestión de relaciones es imprescindible conocer de qué trata este sistema y cuáles son los objetivos estratégicos que se persiguen.

Los autores concluyen, la importancia del mayor conocimiento de los hábitos de los clientes permite a las empresas establecer estrategias comerciales dirigidas y efectivas que contribuyan a cumplir los objetivos empresariales de la organización. Por ello se puede decir que las NTIC's permiten a escala

global gestionar la cartera de cliente de manera personalizada, a tiempo real y optimizar la relación de lealtad y valor entre cliente-empresa.

2.1.2. Antecedentes Nacionales

Herrera, G. [11], en su trabajo de investigación, implementa una solución cloud computing usando una base de datos NoSQL de pacientes con diabetes, el cual tiene como objetivo principal desarrollar una solución a la gestión de datos de pacientes con diabetes en Lima, dicha solución utiliza el paradigma de la computación en la nube y el uso de bases de datos relaciones. El autor pone énfasis que para el año 2030 se duplicarán las cifras de diagnósticos de diabetes en el Perú con relación al año 2000 (754 000 peruanos con diabetes), por lo que si se detecta a tiempo y se lleva un correcto control y seguimiento la enfermedad es tratable. En base a ésta situación problema el autor propone implementar una aplicación que mediante celulares y/o internet permitan a los pacientes enviar recordatorios para tomar medicinas, fechas para sus controles médicos, recibir mensajes educativos de acuerdo al tipo de riesgo. También permite a los usuarios hacer consultas no urgentes y enviar datos de su salud como glucosa, peso y otros que sirvan para monitorear el cumplimiento de sus metas terapéuticas.

El autor concluye que el desarrollo en Cloud Computing ofrece muchas ventajas para el desarrollo de aplicaciones que pueden ser accedidas desde distintas tecnologías y que son importantes en el aporte al sector salud. Además desataca que las bases de datos NoSQL tienen ventajas sobre las bases de datos relaciones para trabajar con grandes volúmenes de información como puede generarse en pacientes con diabetes.

Ballón, J. [12], en su trabajo de tesis implementa un sistema de propuestas de proyectos de software en la empresa Avantica Technologies. Para el desarrollo de éste trabajo al autor propone utilizar la metodología SCRUM y también bases de datos NoSQL. El autor expone que la problemática está en la ineficiente generación de propuestas de preventas en consultoría de proyectos de software y gestión de la información histórica en el área de Preventas. El objetivo de su trabajo es implementar el sistema para automatizar la generación de propuestas y gestionar adecuadamente los

datos históricos obtenidos, contribuyendo a la toma de decisiones del proceso de preventas.

El autor concluye que se reduce en un 60% el tiempo de elaboración de propuestas en base al registro de información histórica en bases de datos NoSQL. Además comenta que el principal hallazgo de su investigación es un prototipo funcional y recomienda la implementación del mismo tomando como referencia la documentación de requerimientos, documentación de diseño, usando ASP. NET MVC 5 y el motor de base de datos no relacional MongoDB como pilares de la arquitectura del sistema

Coronel, N. [13], en su trabajo de tesis propone una aplicación de inteligencia de negocios tradicional en una organización de seguros masivos. El autor menciona que su trabajo esta incentivado por la necesidad de reducir costos, tiempo y optimizar los procesos de los riesgos masivos, estos conllevan a la reducción en el tiempo de entrega de la documentación al cliente, generando una ventaja competitiva con las demás empresas en el rubro, el mismo que permitirá el reconocimiento y la calidad. Actualmente se viene trabajando de forma manual; es decir todo lo concerniente a ambos riesgos se desarrollaba con las herramientas tradicionales de Office, como: Word, Excel y gestores de base de datos Acces. Hay muchas causas naturales o errores humanos que pueden producir deficiencias en la calidad: unos son errores humanos (digitación) y a las quejas reincidentes de los clientes, para evitar la pérdida de la cartera de clientes se optó por desarrollar un datamart, que ayudara a mejorar el control de todos los procesos que se desarrollan en el área.

La autora concluye, la implementación de un proceso de inteligencia de negocio en una empresa, permite que la información fluya de una forma ordenada y controlada desde donde se producen las transacciones del día a día de la organización, hasta convertirlas en información y conocimiento que permiten a los usuarios finales tomar mejores y efectivas decisiones.

2.1.3. Antecedentes Locales

Para tener una base sobre las soluciones de Inteligencias de negocios tradicional, según Uceda, P. [14], describe la implementación de tecnologías DataWarehouse mediante la creación de un DataMart, que permita satisfacer las necesidades de información, considerada como recurso fundamental para poder realizar una adecuada toma de decisiones. Para el desarrollo del proyecto se utilizaron algunas herramientas del Planeamiento Estratégico y la metodología propuesta por Ralph Kimball para el desarrollo de DataWarehouse, abarcando cada una de las etapas desde el análisis hasta la operación. Además herramientas de Microsoft, como: SQL Server 2000, como gestor de base de datos; Analysis Manager, como herramienta OLAP; para el análisis y migración de información se hizo uso de DTS (Paquetes de transformación de datos) y para las estaciones usuarias de utilizó Microsoft Excel 2003 con la herramienta de tablas dinámicas.Los beneficios obtenidos con el sistema son: mayor seguridad, eficacia y eficiencia en el proceso de toma de decisiones al contar con información confiable y oportuna.

La autora concluye que la implementación del sistema generó una mayor efectividad en el proceso de la toma de decisiones, optimizando dicho proceso en un 90%, ya que se disminuyó el tiempo que tardaban en preparar los reportes de 15 a 1 día, dependiendo de la capacidad de análisis del tomador de decisiones. Gracias a la utilización de tecnología, dejando de lado los procesos manuales. Ademas menciona que la metodología de Ralph Kimball permite desarrollar de una manera más ordenada e integrada Data Warehouse o Data Marts, gracias al detalle de cada una de las actividades a desarrollar en las etapas consideradas.

2.2. Bases Teóricas

2.2.1. Bases de datos NoSQL

2.2.1.1. Definición

De acuerdo con Martin, A., Chávez, S. y Murazzo, M. [15]: son sistemas de almacenamiento de información que no cumplen con el esquema entidad-relación. Mientras que las tradicionales bases de datos relacionales basan su funcionamiento en tablas, joins y transacciones. Las bases de datos NoSQL no imponen una estructura de datos en forma de tablas y relaciones entre ellas sino que proveen un esquema mucho más flexible.

Según Torre, A. e Illarramendi, A. [16]: el concepto NoSQL se entiende como una solución posible para el almacenamiento y manejo de cantidad de información fuera de todo lo que sea el mundo relacional. En este punto aparecen dos tecnologías a tener en cuenta, las bases de datos NoSQL y los modelos de programación para el desarrollo de sistemas altamente escalables y de alto rendimiento.

2.2.1.2. Diferencias de NoSQL con SQL

López C. [4] afirma: las bases de datos no relacionales son listas de datos almacenados en una sola tabla sin definir relaciones entre los registros. Por otro lado las relacionales reparten los datos en varias tablas más pequeñas eliminando datos duplicados y asegurando consistencia y estableciendo restricciones y relaciones con otras tablas por medio de claves primarias y foráneas; esto genera que se ocupe menos espacio ya que no tiene redundancias y hasta cierto punto es conveniente esta repartición ya que de otro modo, si se realiza un SELECT múltiple en la no relacional se tendría que recorrer la única tabla varias veces para devolver toda la información debido a duplicidad en ciertos datos.

Si vemos la diferencia con enfoque similar: DataWarehouse. Un DataWarehouse requiere el diseño de un proyecto a largo plazo y la construcción de modelos de datos, procesos de ETL, reportes, etc. A menudo se trata de un proyecto que tiene nuevas exigencias de forma y cada interacción es lenta. La propuesta detrás de las bases de datos NoSQL se puede traducir a través de la filosofía de trabajo que tiene el acrónimo MAD, lo que significa: Magnético, Ágil y Profunda (Magnetic, Agile y Deep).

- **Magnético**. Este enfoque es opuesto al que se utiliza en un almacén de datos empresarial (Enterprise DataWarehouse) que tiende a "repeler" a los datos de nuevas fuentes, que sólo se pueden utilizar tras su limpieza e integración. En el mundo analítico, incluso en ausencia de datos de ciertos valores pueden tener una relevancia estadística.

- ❖ **Ágil**. Por el contrario, como ya se ha dicho, un Enterprise DataWarehouse requiere de un diseño a largo plazo y una planificación. ¿Cómo los datos pueden ser útiles si no es posible extraer la información útil de una manera oportuna?
- ❖ **Profundo**. Sofisticados métodos estadísticos se utilizan para ver "los árboles de un bosque, y no sólo el bosque".

También es importante mencionar que una dificultad inherente a entender la diferencia que supone Big data es hacerse a la idea de lo que supone pasar del esquema de base de datos que todos conocemos a distintos niveles, a la idea de bases de datos no relacionales o NoSQL. Un mundo que suele definirse en negativo, por "lo que no es", lo que añade todavía más dificultad conceptual.

2.2.1.3. Tipos de Bases de datos NoSQL

Según Acens [17], dependiendo de la forma en la que almacenen la información, nos podemos encontrar varios tipos distintos de bases de datos NoSQL:

- ✓ **Bases de datos clave-valor**: Son el modelo de base de datos NoSQL más popular, además de ser la más sencilla en cuanto a funcionalidad. En este tipo de sistema, cada elemento está identificado por una llave única, lo que permite la recuperación de la información de forma muy rápida, información que habitualmente está almacenada como un objeto binario (BLOB). Se caracterizan por ser muy eficientes tanto para las lecturas como para las escrituras.
- ✓ **Bases de Datos Documentales**: Este tipo almacena la información como un documento, generalmente utilizando para ello una estructura simple como JSON o XML y donde se utiliza una clave única para cada registro. Este tipo de implementación permite, además de realizar búsquedas por clave–valor, realizar consultas más avanzadas sobre el contenido del documento. Son las bases de datos NoSQL más versátiles. Se pueden utilizar en gran cantidad de proyectos, incluyendo muchos que tradicionalmente funcionarían sobre bases de datos relacionales. Algunos ejemplos de este tipo son MongoDB o CouchDB.

- ✓ **Bases de datos orientadas a grafos**: En este tipo de bases de datos, la información se representa como nodos de un grafo y sus relaciones con las aristas del mismo, de manera que se puede hacer uso de la teoría de grafos para recorrerla. Para sacar el máximo rendimiento a este tipo de bases de datos, su estructura debe estar totalmente normalizada, de forma que cada tabla tenga una sola columna y cada relación dos. Este tipo de bases de datos ofrece una navegación más eficiente entre relaciones que en un modelo relacional. Algunos ejemplos de este tipo son Neo4j, InfoGrid o Virtuoso.
- ✓ **Base de datos orientadas a objetos:** En este tipo, la información se representa mediante objetos, de la misma forma que son representados en los lenguajes de programación orientada a objetos (POO) como ocurre en JAVA, C# o Visual Basic .NET. Algunos ejemplos de este tipo de bases de datos son Zope, Gemstone o Db4o. Algunos ejemplos de este tipo son Cassandra, BigTable o HBase.

2.2.1.4. MongoDB

2.2.1.4.1. Modelado de Datos en MongoDB

De acuerdo con Brito, D. [1], la unidad básica de almacenamiento se define como un "documento" que sería el equivalente a un registro en una base de datos relacional. Los "documentos" con contenidos en una estructura de mayor jerarquía denominada "colección" que almacena los datos de similar tipo; las colecciones serían el equivalente las tablas en un sistema relacional. Finalmente las colecciones se agrupan en una base de datos que las contendrá. MongoDB almacena documentos en un formato similar a JSON (para ser más exactos internamente usa BSON).

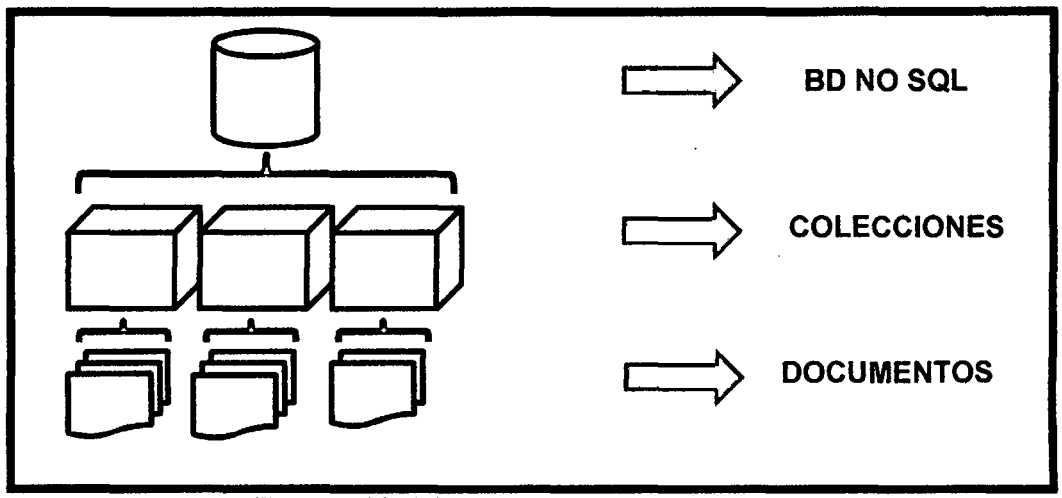


Figura 1. Modelo de Datos de MongoDB

(Por el autor)

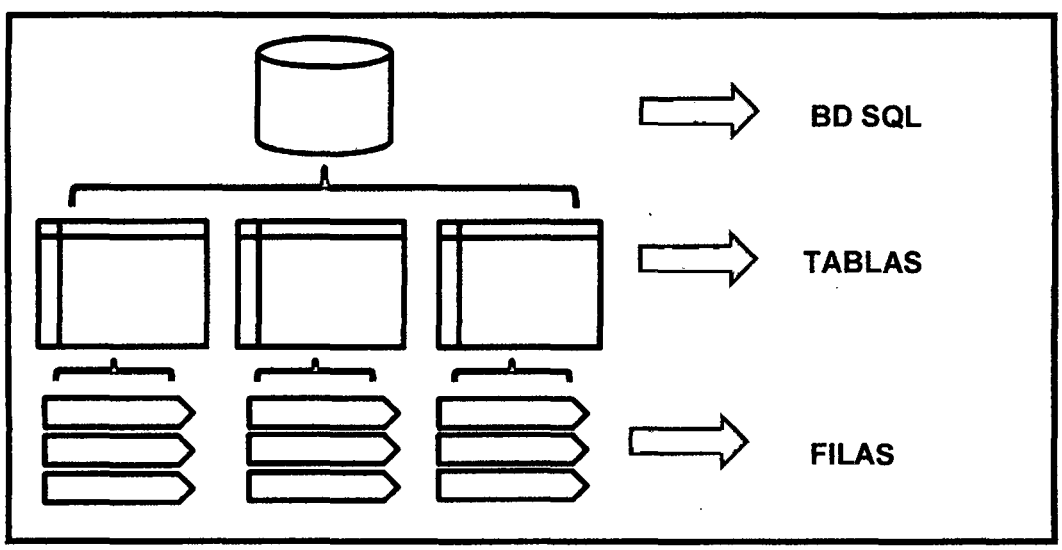


Figura 2. Modelo de Datos Relacional
(Por el autor)

2.2.1.4.2. BSON Y JSON

Para continuar con el análisis de MongoDB es necesario mencionar que dicha ésta herramienta con el fin de maximizar el rendimiento, tanto del almacenamiento como de las búsquedas, MongoDB hace uso de BSON como formato de almacenamiento de datos. BSON es la forma binaria de JSON que, junto con los datos, almacena información adicional sobre los objetos que almacena como pueden ser índices y las longitudes de los campos. Sin embargo, los datos extraídos

mediantes las consultas son convertidos a formato JSON para su fácil manejo y entendimiento. [4]

MongoDB no utiliza SQL para sus procesos de consultas sino que hace uso de librerías específicas para cada lenguaje de programación mediante las que operar sus consultas.

Debemos recordar que MongoDB es una base de datos documental que no usa esquemas de datos con columnas predefinidas. El principal beneficio se refleja en la flexibilidad que ganamos al almacenar los datos, pues de requerir un campo extra no tendremos realizar el proceso de cambiar la estructura de nuestra base de datos como lo haríamos en una relacional. [2]

Las colecciones de datos de MongoDB son extremadamente flexibles, por ejemplo supongamos la colección "*media*" que en JSON podría tener los siguientes documentos.

Tabla 1. Ejemplo Documento JSON

```
{
  Id_: "ObjectId(f5g4fgg8fr245frg4h4ff)",          // El
Id_ es generado por el motor de MongoDB

  Tipo: "CD",
  Artista: "Chayanne",
  Genero: "Baladas",
  ListaCanciones: [
                {
                NumeroPista: "1",
                Titulo: "Me enamore de ti"
                },
                {
                NumeroPista: "2",
                Titulo: "Tu Boca"
                }
              ]
```

```
}
{
  Tipo: "Libro",
  Titulo: "Tesis de Grado",
  Autores: ["Chávez, Luis", "Chávez, Miguel"]
}
```

Como podemos ver en el ejemplo anterior los dos documentos de la colección "*media*"; si bien es cierto tienen dos campos en común: "***Tipo***" y "***Titulo***", los demás campos son diferentes entre ellos a pesar de pertenecer a una misma colección. MongoDB tiene un esquema flexible, donde las colecciones no fuerzan una estructura idéntica para todos los documentos. Esto significa que los documentos de la misma colección no necesitan tener el mismo número de campos o estructura, y los campos comunes pueden contener diferentes tipos de datos. Cada documento solo necesita contener un número relevante de campos de la entidad u objeto que el documento representa. El ejemplo anterior sería difícil de implementar en una base de datos relacional, la única opción sería crear una tabla con todas las posibilidades que podría darse; con pocos campos podría hacerse, pero si hablamos de un número más alto de campos entonces no sería realizable. [1]

No debemos abusar de la flexibilidad de MongoDB, pues si usamos documentos extremadamente diferentes los unos de los otros no explotaremos todo el potencial de la herramienta. Para nombrar las columnas, colecciones y bases de datos debemos usar letras y números sin símbolos. El máximo de caracteres permitido para los nombres es de 128, pero es recomendable mantener los nombres cortos. [1]

2.2.1.4.3. Los Documentos en MongoDB

Los documentos son un conjunto de pares llave/valor. Por ejemplo el par "Tipo": "CD" consiste de una llave denominada "Tipo" y de su respectivo valor "CD". Las llaves generalmente

son un string, pero los valores pueden ser de varios tipos, pueden ser arreglos o hasta datos binarios. [4]

Tabla 2. Clave-Valor

Llave/Clave	Valor
Tipo	**"CD"**

Los tipos de datos que regularmente podemos usar en MongoDB son:

Tabla 3. Tipos de Datos de MongoDB

TIPO	DESCRIPCIÓN
String	Puede contener cualquier colección de caracteres. Por ejemplo: Nombre: "Luis"
Integer (32 y 64)	Permite almacenar valores numéricos sin decimales, por ejemplo: Cantidad: 1987
Boolean	Puede contener valores de falso o verdadero (true y false)
Double	Puede contener valores numéricos con decimales, por ejemplo: Valor: 10,65
Arrays	Puede contener arreglos, por ejemplo: Autores: ["Chávez, Luis", "Chávez, Miguel"]
Timestamp	Puede contener fechas y horas.
Object	Puede contener subdocumentos.
Null	Puede contener valores nulos (NULL).
Symbol	Puede contener caracteres al igual que String, pero se aplica en lenguajes que usan símbolos.

Object ID	Almacena el ID del documento,
Binary	Puede contener datos *binarios, por ejemplo una* imagen
Regular Expression	Puede contener expresiones regulares.
JavaScript	Puede contener como su nombre lo indica código JavaScript

- **Documentos incrustado o embebidos:**

  Antes de hablar de este tema es importante comprender por qué necesitamos usarlos. Los documentos incrustados o documentos embebidos son las alternativas que MongoDB nos brinda para resolver las dependencias de los datos. En el JSON del ejemplo anterior encontramos el caso en el que cada CD debe tener su lista de canciones, de tal manera que en una base de datos relacional tendríamos una tabla denominada ***CDs*** relacionada de uno a varios con una tabla llamada ***Canciones.*** EN el siguiente ejemplo vemos mejor un documento embebido: [1]

Tabla 4. Documento JSON embebido

```
  Id_: "ObjectId(f5g4fgg8fr245frg4h4ff)",          // El
Id_ es generado por el motor de MongoDB

  Tipo: "CD",
  Artista: "Chayanne",
  Genero: "Baladas",
  ListaCanciones: [
                {
                NumeroPista: "1",
                Titulo: "Me enamore de ti"
                },
                {
                NumeroPista: "2",
```

```
                Titulo: "Tu Boca"
                }
            ]
}
```

El documento de ejemplo usa documentos embebidos o incrustados para almacenar la lista de canciones en el par "***ListaCanciones***".

- **Objetc ID en MongoDB**

  Al igual que una tabla en una base de datos relacional identifica cada una de sus filas con un valor único denominado Llave Primaria; MongoDB también identifica de manera única sus documentos, y lo hace en un par llamado "***_id***". [1]

  En los ejemplos anteriores hemos referenciamos al par "***_id***" para efectos de entendimiento, sin embargo MongoDB se encarga de crearlo de manera automática.

2.2.1.5. Relación con Big Data

Méndez M. [18] menciona: el termino Big Data se puede definir como datawarehouses de varios petabytes a datos provenientes de redes sociales; de aplicaciones basadas en Cloud Computing a sensores y dispositivos móviles; de datos de aplicaciones de comercio electrónico a información geoespacial que no pueden ser capturados, almacenados ni analizados con el software y la infraestructura tradicionales que se han empleado hasta ahora, lo que supone pasar del esquema de base de datos que todos conocemos a distintos niveles, a la idea de bases de datos no relacionales NoSQL. A continuación veremos un framework que usa Big Data:

2.2.1.5.1. Hadoop

De acuerdo con Augsburger M. [19], es un framework de software que soporta aplicaciones distribuidas bajo una licencia libre. Permite a las aplicaciones trabajar con miles de nodos y grandes volúmenes de datos (Big Data). Consta de dos

servicios principales: almacenamiento fiable utilizando el Sistema de Archivos Distribuidos de Hadoop y un proceso de datos en paralelo de alto rendimiento que es una técnica llamada MapReduce.

- **HDFS.** El sistema de archivos distribuidos de Hadoop, HDFS, es un sistema de archivos diseñado para guardar grandes cantidades de datos y proveer acceso de alto flujo de datos a esta información. Los archivos son almacenados de manera redundante a través de múltiples máquinas, para asegurar su resistencia a fallas y su alta disponibilidad a aplicaciones altamente paralelas.
- **MapReduce.** Hadoop implementa un paradigma computacional llamado map-reduce, el cual divide la aplicación en fragmentos de trabajo. Cada uno de los fragmentos puede ser ejecutado o re ejecutado en cualquier nodo del clúster. Un trabajo MapReduce divide el set de datos de entrada en trozos independientes, los cuales son procesados por las tareas map en forma paralela. El framework clasifica el output de los maps, los cuales son entonces entregados como input de las tareas "reduce", que también son ejecutadas de forma paralela. Conceptualmente, un programa escrito en MapReduce transforma listas de elementos entregados como input, en listas de datos de salida. Esto se lleva a cabo dos veces, usando dos procesos de modificación de datos: map y reduce.

Según Díaz, C. [20], MapReduce es el nombre dado a la combinación de dos procesos separados necesarios para la extracción de valores de un gran número de orígenes de datos distintos. La parte map funciona como extractor y asigna valores a determinadas claves para un único documento. La parte reduce realiza la función de acumulación y combina las claves de múltiples documentos para crear un valor reducido (combinado) único para cada clave a partir de los múltiples valores generados.

- **Map**. El método Map recibe como entrada un par (clave, valor) y su salida es uno o varios pares (clave-i, valor-i). Para cada (clave1, valor1) devuelve una lista de (clave2, valor2):

$$(clave1, valor1) \rightarrow [(clave2, valor2)]$$

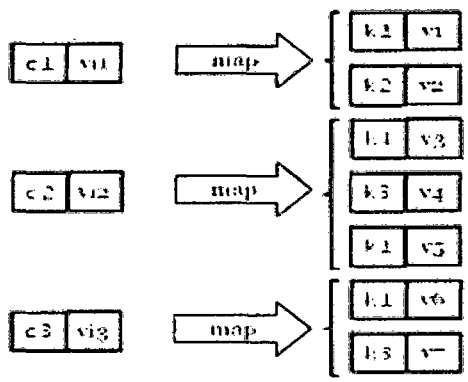


Figura 3. Función Map
(Díaz, 2011, p. 8)

El sistema se encarga de mezclar y ordenar resultados intermedios en función de las claves.

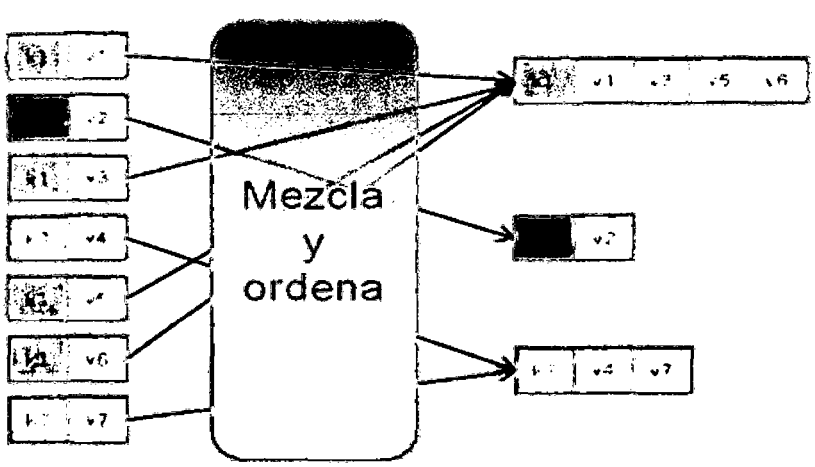


Figura 4. Proceso de Mezcla y Ordenación
(Díaz, 2011, p. 8)

- **Reduce**. El método Reduce recibe como entrada un par (clave, lista de valores) y la salida es un par (clave, valor). Para cada clave2, toma la lista de valores asociada y los combina en uno solo:

(clave2, [valor2]) → (clave2, valor2)

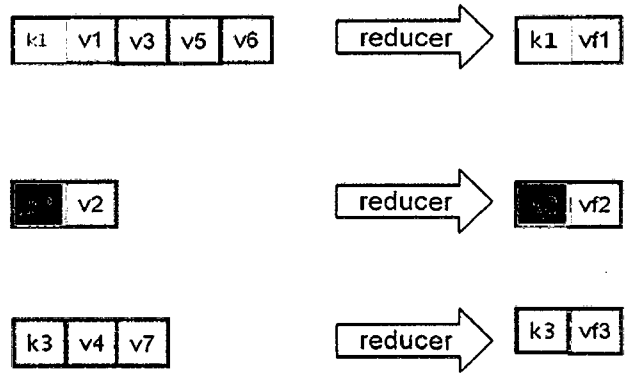


Figura 5. Función Reduce

(Díaz, 2011, p. 9)

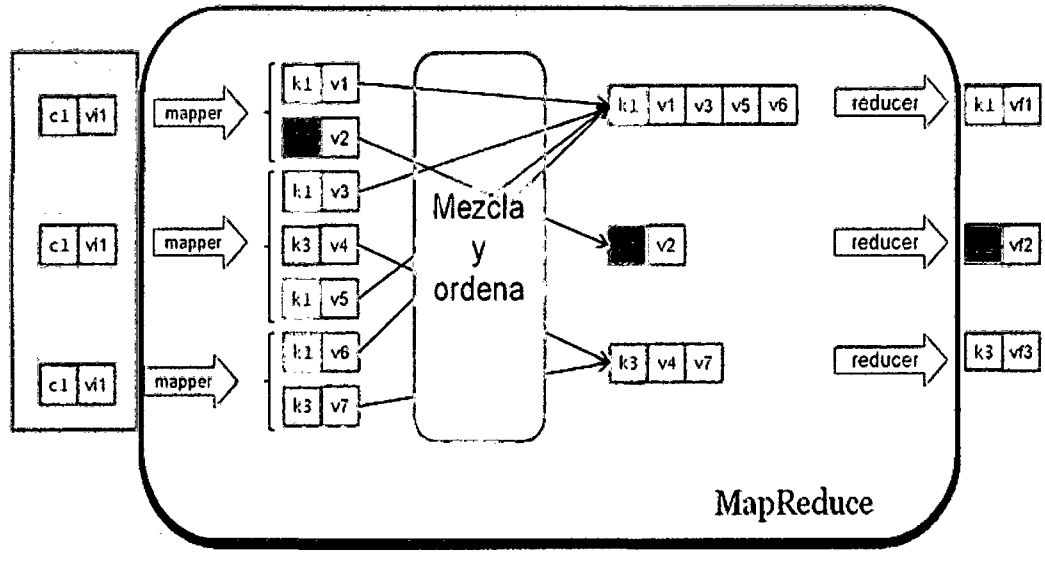


Figura 6. Modelo de Proceso de MapReduce
(Díaz, 2011, p. 9)

2.2.1.6. Minería Web

Según Etzioni, O. [21] se refiere al proceso global de descubrir información o conocimiento potencialmente útil y previamente desconocido a partir de datos de la Web. El autor Scooto, M.[22], la define: es el descubrimiento y análisis de información relevante que involucra el uso de técnicas y acercamientos basados en la minería de datos (Data Mining) orientados al descubrimiento y extracción automática de información de documentos y servicios de la Web, tomando en consideración el comportamiento y preferencias del usuario. De acuerdo con Pal, K., Talwar, V.y Mitra, P. [23], en minería web, los datos pueden ser recogidos por parte del servidor, el cliente u obtenidos de bases de datos de una organización. Dependiendo de la localización de la fuente, el tipo de datos puede diferir, existiendo una gran variación en los contenidos (por ejemplo: textos, imágenes, audio, símbolos). Esto hace que las técnicas utilizadas en minería web cambien según la tarea particular que haya que llevar a cabo. Sin embargo, algunas características comunes de los datos web son las siguientes:

- ✓ No etiquetados.
- ✓ Distribuidos.
- ✓ Heterogéneos
- ✓ Semi-estructurados
- ✓ Variables en el tiempo.

Por lo tanto, la minería web trata básicamente con información de gran tamaño, con hiperenlaces y con las características antes mencionadas. Además, al ser un medio interactivo, la interfaz humana es un componente clave en la mayoría de los usos de la web. Algunas cuestiones que han salido a la luz, por consiguiente, se centran en:

- ❖ La necesidad de manejar preguntas sensibles al contexto e imprecisas.
- ❖ La necesidad de resumir y deducir.
- ❖ La necesidad de que exista una personalización y un aprendizaje

2.2.1.6.1. Categorías de la minería web

Según Tao, U., Hong, Y. y Su, M. [24], la minería web puede ser de tres tipos:

- Minería Web de Contenidos (WCM, *Web Content Mining*).
- Minería Web de Estructuras (WSM, *Web Structure Mining*).
- Minería Web de Uso (WUM, *Web Usage Mining*).

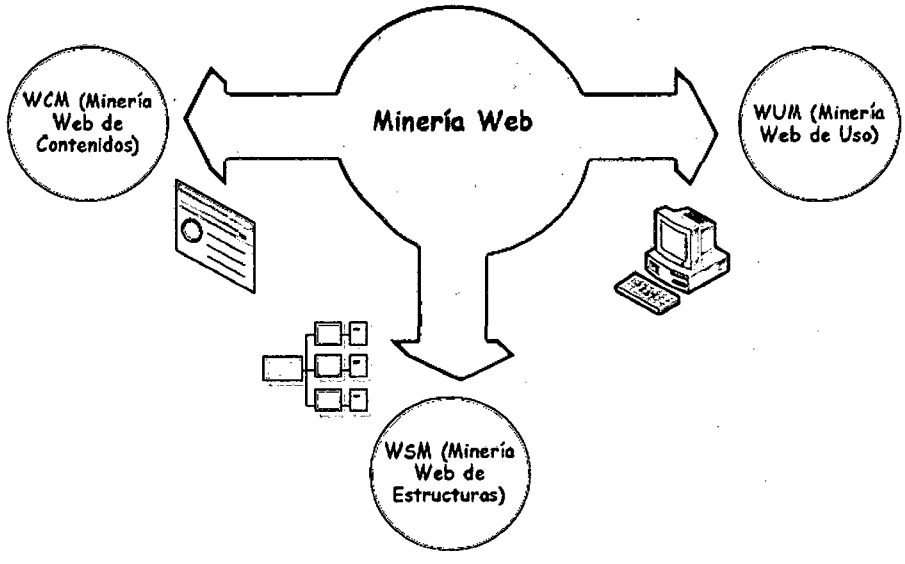


Figura 7. Categorías de Web Mining
(Ropero, 2009, p. 36)

WCM clasifica los documentos automáticamente o construye una base de información web multicapa. WSM extrae la estructura de una página web; WUM descubre patrones de acceso a las páginas en los usuarios.

- **WCM- Web Content Mining**

  WCM trata con el descubrimiento de información útil de los contenidos/datos/documentos/servicios de la web. Sin embargo, los contenidos web no se componen únicamente de texto, sino también de audio, vídeo, datos simbólicos e hiperenlazados y metadatos.

  Por otra parte, según la forma de afrontar el problema, podemos contemplar dos aproximaciones para WCM: la

aproximación basada en agentes y la basada en bases de datos.

- ✓ La aproximación basada en agentes implica el desarrollo de sofisticados sistemas que pueden actuar de forma autónoma o semi-autónoma en nombre de un usuario particular para descubrir y organizar información contenida en la web. Generalmente, la aproximación basada en agentes se puede a su vez subdividir en tres categorías: agentes de búsqueda inteligente, categorización/filtrado de información y agentes web personalizados.
- ✓ La aproximación centrada en bases de datos se centra en las técnicas para organizar datos semi-estructurados de la web en conjuntos de recursos más estructurados, utilizando mecanismos de búsqueda y técnicas de minería de datos para analizarlos.

➢ **WSM-Web Structure Mining.**

WSM extrae la estructura de los hiperenlaces, es decir, como están los documentos estructurados respecto a los otros (estructura interdocumental, a diferencia de la estructura intradocumental de WCM). La estructura se representa como un grafo de los enlaces en un sitio web o entre sitios web. WSM revela más información que la información contenida en los documentos: por ejemplo, los enlaces que apuntan a un documento pueden indicar la popularidad o importancia de un documento, mientras que los enlaces salientes indican la riqueza o variedad de los temas que contiene. Esto nos lleva a una organización jerárquica por temas que puede ser inferida directamente de los patrones de enlazado. Es posible incluso no especificar los documentos mediante palabras clave, sino mediante documentos ejemplares.

➢ **WUM - Web Usage Mining.**

Mientras que la *minería de contenidos* (WCM) y la *minería* de estructura (WSM) usan los datos reales o primarios de

la web, la minería de uso (WUM), utiliza datos secundarios generados por la interacción de los usuarios con la web. WUM incluye datos de las conexiones a los servidores web, servidores Proxy o buscadores, perfiles de usuarios, archivos de registro, sesiones de usuario, búsquedas, clicks de ratón o scrolls, carpetas de favoritos, etc.

2.2.1.6.2.Metodología Web Mining

Según Kosala, R. y Blockeel, H. [25], la metodología que usa Web Mining se componen en 4 fases:

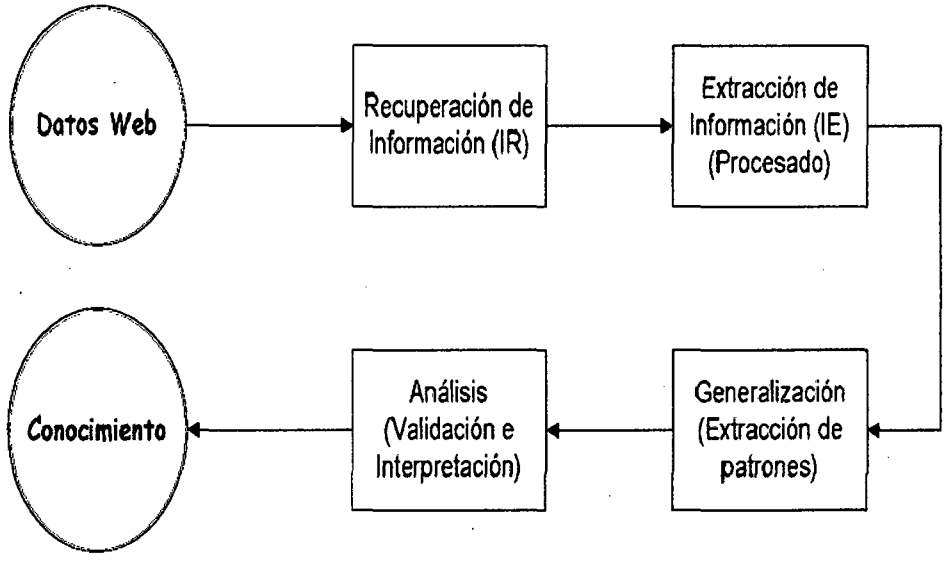


Figura 8. Etapas de la Metodología Web Mining
(Ropero, 2009, p. 38)

- ✓ **Recuperación de Información (IR).**

  La Recuperación de Información (Information Retrieval, IR) trata acerca de la recuperación automática de todos los documentos relevantes de un conjunto de conocimiento, asegurando al mismo tiempo que los documentos no relevantes recuperados sean los menos posibles. El proceso de IR incluye principalmente la representación de documentos, el indexado, y la búsqueda de documentos.

  Existen diferentes técnicas de minería de contenidos web utilizadas por distintos autores para la Recuperación de

Información en documentos semi-estructurados. Estas técnicas se basan generalmente en la utilización de índices. Un índice es, básicamente, una colección de términos con indicadores a los lugares en los que puede encontrarse la información sobre los documentos. Sin embargo, indexar páginas web para facilitar la recuperación es un proceso bastante complejo, y un reto si se compara con el problema correspondiente asociado a bases de datos clásicas, donde las técnicas directas son suficientes. El enorme número de páginas web, su dinamismo, y su frecuente puesta al día hace que las técnicas de indexado parezcan aparentemente imposibles de aplicar. Actualmente, existen cuatro aproximaciones para el indexado de documentos en la web: indexado humano o manual; indexado automático; indexado inteligente o basado en agentes; e indexado basado en metadatos.

- ✓ **Extracción de Información (IE)**

  La Extracción de Información (Information Extraction, IE) consiste en la transformación de una colección de documentos, habitualmente con la ayuda de sistemas de IR, en información más fácil de asimilar y analizar. IE intenta extraer hechos relevantes de los documentos, mientras que IR selecciona los documentos relevantes. Por tanto, podríamos decir que IE trabaja con una granularidad más fina que IR. En todo caso, los conceptos IE e IR pueden llegar a confundirse en la práctica.

  Entonces, una vez que los documentos han sido recuperados, el desafío debe ser extraer el conocimiento y otras informaciones requeridas automáticamente, sin la interacción humana. La Extracción de Información (IE) es la tarea de identificar los fragmentos específicos de un documento o, en general, de cualquier conjunto de conocimiento, que constituyen su principal contenido semántico. Según Kushmerick, N. [26], los sistemas de extracción de información funcionan interpretando los

distintos conjuntos de conocimiento y extrayendo la información de ellos. Por ejemplo, se pueden considerar los diversos sitios web como fuentes de conocimiento. Para hacer eso, el sistema procesa los documentos del sitio web para extraer fragmentos de texto relevantes y se usa una librería de envoltorios en la que cada envoltorio es un sistema personalizado de IE para cada sitio particular de Internet.

- ✓ **Generalización.**

  En esta fase, se usan el reconocimiento de general patrones sobre la información extraída. La mayoría de los sistemas de aprendizaje en máquinas utilizados en la web aprenden más sobre los intereses de los usuarios que sobre la propia web.

  Las técnicas de minería de datos utilizan análisis estadísticos para descubrir reglas de asociación y patrones de interés entre distribuciones de los denominados términos índice o palabras clave (keywords). El problema está en que la asignación de estas palabras clave a cada conjunto de conocimiento (por ejemplo: un texto o una página web) debe ser realizada previamente a la aplicación de dichas técnicas.

  Existen diversas formas de realizar esta asignación, entre las que cabe destacar las siguientes:

  - ❖ Extracción automática de los términos más comunes de un texto: los términos más frecuentes se asignan como palabras clave. Sin embargo, los textos no son pre-procesados por lo que no se analiza el contexto en el que se encuentran dichos términos y se genera un modelo difícil de entender sin una lectura plena del texto.
  - ❖ Extracción automática de términos usando información sintáctica: la idea es no usar todos los términos sino sólo los realmente significativos. El problema viene dado por los errores semánticos

que pueden causar la sinonimia (palabras diferentes para el mismo significado), la polisemia (una palabra con diferentes significados) y las palabras con la misma raíz.

Las técnicas LMI (Linguistically Motivated Indexing,) permiten el tratamiento de términos índice multipalabra de una forma diferente y quizás más elaborada, aunque en ambos casos, precoordinación y postcoordinación son posibles. Un término índice puede ser un término solo (una palabra sola o una palabra raíz) o uno compuesto: este último puede ser un término complejo (encontrado por cualquier técnica LMI) o un término relacionado o similar. En la siguiente figura se clasifican los distintos términos índice. Estos pueden ser términos simples (palabras sencillas o raíces) o bien términos compuestos (complejos o relacionados).

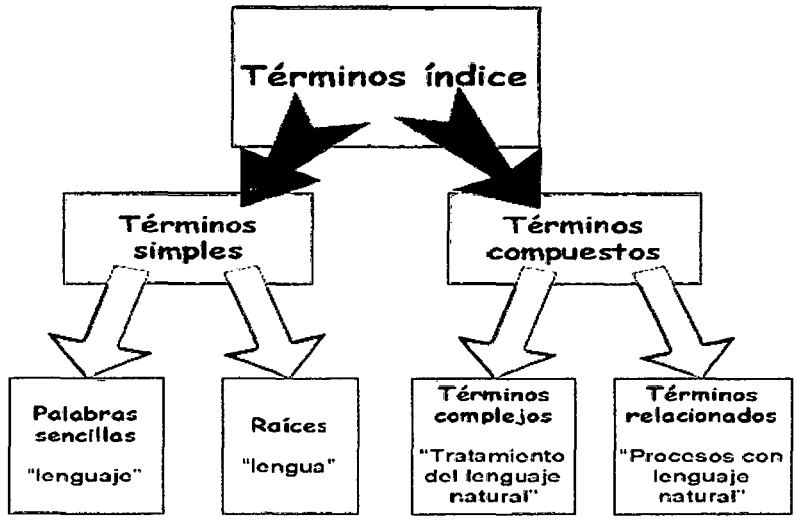


Figura 9. Tecnología LMI- Términos Índice

(Ropero, 2009, p. 52)

- ✓ **Análisis e Interpretación**

  El análisis de los datos es un aspecto que asume que existen suficientes datos para que la información potencialmente útil pueda ser extraída y analizada. Los seres humanos juegan un papel importante en el proceso de descubrimiento del conocimiento y la información en la web, dado que la propia web es un medio interactivo. Esto es especialmente importante para la interpretación de los patrones de minería. Una vez que los patrones han sido descubiertos, los analistas necesitan herramientas apropiadas para comprender, visualizar e interpretar estos patrones.

2.2.2. Segmentación de clientes

2.2.2.1. Definición

De acuerdo con Núñez S. [27], es el proceso de dividir un mercado en grupos uniformes más pequeños que tengan características y necesidades semejantes. Los segmentos son grupos homogéneos y que tienen como particularidad que dentro de cada grupo, es probable que respondan de modo similar a determinadas estrategias de marketing. De esta forma, los integrantes del segmento tendrán probablemente las mismas reacciones acerca del marketing mix de un determinado producto, vendido a un determinado precio, distribuido en un modo determinado y promocionado de una forma dada.

Cuando ya se cuenta con una base de información como los tipos de segmentación y sus variables, es necesario tener un adecuado acceso a la información. Según Martínez C. [28], se refiere al conjunto de técnicas para buscar, categorizar, modificar y acceder a la información que se encuentra en un sistema: bases de datos, bibliotecas, archivos, Internet. Así mismo, el acceso a la Información involucra a muchos otros temas, como los derechos de autor, el Código abierto, la privacidad y la seguridad. En nuestro caso el acceso a la información se aplica a información que ya ha sido procesada por un sistema de base de datos, por lo que el objetivo es tanto encontrar la manera más eficiente de clasificarla y archivarla; como encontrar la mejor manera

de obtener de manera inequívoca la información deseada utilizando para ello el menor número de recursos.

2.2.2.2. Tipos de Segmentación y sus variables

- ✓ **Demográfica:** Sexo, edad, Raza, lugar de residencia.
- ✓ **Socio-económica**: Nivel de ingreso, nivel de educación, clase social, profesión.
- ✓ **Por Uso:** Por cantidad de uso, tipo de uso, *oportunidad de eso, lealtad* de marca.
- ✓ **Por Estilo de Vida**: Se basan en datos estadísticos reales que combinan las variables anteriores obteniendo grupos con individuos que piensan y consumen de manera similar y que comparten además ciertas variables socio-económicas y demográficas.

2.2.2.3. Soluciones para la Segmentación basada en el valor del cliente

2.2.2.3.1. Business Intelligence

De acuerdo con Velásquez E. [29], se refiere al uso de los datos de una empresa para facilitar la toma de decisiones mediante la comprensión del funcionamiento actual y la anticipación de acciones para dar una dirección operativa optima a la misma. Existen muchas maneras de analizar la información, y por este motivo existen un conjunto de soluciones que resuelven las diferentes necesidades analíticas.

2.2.2.3.1.1. DataWarehouse

Según Sevilla, E. [30], es un conjunto de datos orientados hacia una materia, integrados, no transitorios y que varían con el tiempo, los cuales apoyan el proceso de toma de decisiones de una administración.

- ✓ **Características.** Un DataWarehouse se caracteriza por ser:
  - o **Integrado.** Los datos almacenados en el DataWarehouse deben integrarse en una estructura que elimine las inconsistencias existentes entre los diversos sistemas operacionales, que sirven de fuentes de datos. La información suele estructurarse también en

distintos niveles de detalle para adecuarse a las distintas necesidades de los usuarios.

- **Temático:** sólo los datos necesarios para generar información del negocio se integran desde el entorno operacional. Los datos se organizan por temas, no por aplicación, así se facilita el acceso y entendimiento por parte de los usuarios finales a los datos contenidos en el DataWarehouse. Esta característica permite realizar análisis y minería de datos (data mining).
- **Histórico.** El tiempo es parte implícita de la información contenida en un DataWarehouse. En los sistemas operacionales, los datos siempre reflejan el estado de la actividad del negocio en el momento presente. Por el contrario, el DataWarehouse se carga con los distintos valores que toma una variable en el tiempo para permitir comparaciones y análisis de tendencias, entre otras cosas.
- **No volátil.** El DataWarehouse se construye para ser leído, y no modificado. La información que existe en un DataWarehouse es permanente, su actualización consiste en la incorporación de los últimos valores que tomaron las distintas variables contenidas en él sin alteración a los datos que ya existían.

2.2.2.3.1.2. Metodologías DataWarehouse.

A continuación se abordan dos de las metodologías más usadas:

2.2.2.3.1.2.1. Metodología de W. H. Inmon

William H. Inmon es considerado el padre del DataWarehouse, pues fue el primero en acuñar el término, además su libro con el cual dio a conocer su teoría sobre el DataWarehouse, es hoy en día, una década después de su primera publicación,

una de las referencias más completas y utilizadas por los profesionales que ingresan al mundo de la tecnología del DataWarehouse.

La metodología que él propone difiere del plan de migración en varias maneras. El plan de migración describe actividades generales dinámicamente. La metodología describe actividades específicas, los resultados de esas actividades y el orden de las actividades, pero las dinámicas iterativas de crear un DataWarehouse no son descritas. En otras palabras el plan de migración describe un plan impreciso en tres dimensiones, mientras la metodología describe un plan detallado en una dimensión. Juntos ellos forman un retrato completo de lo que se requiere para construir el DataWarehouse.

A continuación la esencia de la metodología propuesta por Inmon:

- **Desarrollo de Sistemas Operacionales**
  - **Actividades Iniciales del Proyecto:** Obtención de los requerimientos preliminares del sistema a través de Entrevistas, recopilación de datos, Diseño de Aplicaciones Conjuntas (JAD), análisis del plan estratégico de negocios, requerimientos de parte de los sistemas existentes para los nuevos sistemas.
  - **Uso de código y datos existentes:** Usar cuanto código y datos sean posibles y preparar para futuros proyectos que usaran código y datos desarrollados en el proyecto actual.
  - **Determinación de tamaño y fases:** Luego de la obtención de los requerimientos generales se deben de determinar su tamaño y dividir el desarrollo del proyecto en fases que funcionen como unidades pequeñas y manejables.

- **Formalización de los requerimientos:** asegurar que los requerimientos sean completos, organizados, leíbles, comprensibles y a un nivel de detallen que permita ser efectivos, que no sean discordantes o se traslapen. Se debe separar los requerimientos operacionales de los de soporte a la toma de decisiones.

- **Modelado de Datos**
  - **Diagrama Entidad Relación:** De la especificación formal de los requerimientos resulta la necesidad de identificar las áreas de mayor interés que constituirán el sistema y las relaciones y cardinalidad entre ellas.
  - **Conjunto de Elementos de Datos:** Cada tema es dividido (en términos de nivel de detalle) en conjunto de elementos de datos (CED). Los CED contiene los atributos de los datos, agrupamiento de los atributos, llaves, tipos de datos, conectores y agrupamiento secundario de los datos. Solo los datos primitivos se manejan aquí.
  - **Análisis de Desempeño:** Se resuelve el asunto de la denormalización física de los datos que permita un ingreso y actualización eficientes, para el caso de grandes cantidades de datos o de procesamiento.
  - **Diseño físico de la Base de Datos:** Obtención de las tablas y bases de datos diseñadas físicamente, luego de transformar todas las consideraciones lógicas de diseño de datos, desempeño, actualización, ingreso, disponibilidad, etc.

- **Especificaciones de Proceso**
  - **Descomposición Funcional:** Resulta de tomar todas las funciones amplias a ser alcanzadas por el sistema y dividirlas en una serie de funciones pequeñas sucesivas. Es la descripción de todas las actividades a ser realizadas durante el desarrollo desde un nivel alto hasta un nivel bajo.
  - **Nivel de Contexto 0.** Diagrama Entidad-Relación, en la especificación de proceso.
  - **Nivel de Contexto 1-n:** Los niveles restantes de la descomposición funcional describen más detalladamente las actividades que ocurren, de manera ordenada, organizada, completa y en concordancia con el flujo de actividades.
  - **Diagramas de Flujo de Datos (DFD):** Existe un DFD para cada nivel de contexto n, indica la entrada de un proceso, la salida del proceso, el almacenamiento de datos necesario para establecer el proceso y una breve descripción del proceso.
  - **Especificaciones Algorítmicas, Análisis de Desempeño:** Es el bosquejo del procesamiento actual paso por paso. Los procesos de cada DFD se dividen en especificaciones algorítmicas detalladas, tomando en cuenta los aspectos de desempeño que deben ser resueltos en el diseño de los programas.
  - **Pseudocódigo:** Los algoritmos y especificaciones de programas se refinan en pseudocódigo, el cual debe incluir completitud, orden de ejecución, todos los casos requeridos, todas las contingencias (manejo de errores, condiciones de excepción), estructura de la codificación.
  - **Codificación:** *Construcción del código fuente.* La traducción completa y eficiente de

pseudocódigo en código, incluyendo la documentación en código.

- Caminata: Explicación verbal del código a los colegas, para encontrar y corregir la mayor cantidad posible de errores antes de las pruebas.

- **Desarrollo del DW.** Este es el componente de la metodología que se ocupa del desarrollo de sistemas y procesamiento de soporte a la toma de decisiones.
  - **Análisis del Modelo de Datos:** Confirmación que el modelo de datos de la organización es sólido y que contiene la identificación de los temas de mayor interés, cada tema tiene separada su propia definición de datos: subtipos de datos, atributos, relaciones claramente definidas, identificación de llaves entre otros.
  - **Análisis Breadbox:** Permite la determinación del tamaño –estimación bruta- del entorno de los sistemas de soporte a la toma de decisiones. Simplemente proyecta, en términos crudos, que cantidad de datos mantendrá el Data Warehouse. Con esto es posible determinar si será necesario considerar múltiples niveles de granularidad.
  - **Valoración Técnica:** Contiene definiciones técnicas que tienen la habilidad de manejar grandes cantidades de datos, permitir que los datos sean ingresados flexiblemente, organizar los datos de acuerdo al modelo de datos, recibir y enviar datos a una amplia variedad de tecnologías, contemplar descargas de datos masivas, acceder conjunto de datos simultáneamente o registro por registro.
  - **Preparación del entorno técnico:** *Instalación*, ubicación y desarrollo de los componentes

técnicos que recibirán los datos: la red, el almacenamiento secundario, su sistema operativo, la interfaz hacia y desde el DataWarehouse el software administrador del DataWarehouse y el DataWarehouse en sí.

- **Análisis de los temas del DataWarehouse:** Determinación del tema que será el primero en implementarse (poblarse). Debe ser grande lo suficiente para tener sentido y pequeño lo suficiente para permitir una rápida implementación.
- **Diseño del DataWarehouse:** Algunas de las características del diseño incluyen:
  - Acomodación de los diferentes niveles de granularidad, si existen.
  - Orientación de los datos a los principales temas de la organización.
  - La presencia de solo datos primitivos y datos derivados públicamente.
  - La ausencia de datos que no apoyan los sistemas de soporte a las decisiones.
  - Variabilidad de tiempo en cada registro de datos.
  - Denormalización física de los datos donde sea aplicable
  - Adaptación de los datos del entorno operacional al analítico- DataWarehouse.
- **Análisis de los Sistemas Fuente:** Identificación del sistema de registro, es decir el mapeo de los datos del ambiente operacional al ambiente analítico. Se deben resolver los siguientes aspectos relacionados a la integración de los datos:
  - La estructura y resolución de llaves al pasar al ambiente analítico.
  - Atribuciones: Elección de una fuente de datos entre muchas, qué hacer cuando no

hay fuente de datos, las transformaciones a realizar.

- Creación de la variabilidad de tiempo en los registros a partir de datos
- actuales.
- Creación de la estructura analítica a partir de la estructura operacional.
- Reflejo de las *relaciones del entorno* operacional en el ambiente analítico.

- **Especificaciones:** Formalización de la interfaz entre los ambientes operacionales y analíticos en términos de especificaciones de programas, que permitan la extracción e integración de los datos lo más eficiente y simple posible. Verifica que datos operacionales deben ser obtenidos y como guardarlos.
- **Programación:** Programas de transformación que permiten la extracción, integración y ubicación en perspectiva de tiempo de los datos. Incluye todas las actividades estándar de programación como: elaboración de pseudocódigo, codificación, compilación, pruebas. Esto asegura que el código generado sea eficiente, documentado, con capacidad de cambio rápido, eficaz y completo.
- **Población:** Ejecución de los programas desarrollados en las etapas anteriores. Se deben resolver los aspectos de frecuencia de población, reglas de purga, envejecimiento de los datos poblados, administración de múltiples niveles de granularidad y refrescamiento. Con este paso final se obtiene un DataWarehouse poblado y funcional, accesible y comprensible que sirve las necesidades de la comunidad de los sistemas de soporte a la toma de decisiones.

- **Procesamiento Heurístico.** Esta componente de la metodología de lo que Inmon llama entorno arquitectónico describe el uso del DataWarehouse para propósitos de análisis.
  - **Repetición del Desarrollo Estándar**: Para la obtención de reportes estándares, el procesamiento analítico repetitivo debe seguir el procesamiento *normal descrito en el Desarrollo* de Sistemas Operacionales, exceptuando el modelado de datos, porque la fuente de datos es el DataWarehouse.
  - **Determinación de los datos necesarios:** Selección de datos para análisis posteriores y uso potencial en la satisfacción de los requerimientos de reportes.
  - **Programas para extraer datos:** Escritura de un programa para acceder y recuperar los datos seleccionados.
  - **Combinar, Fusionar, Analizar**: Edición, combinación con otros datos y refinamiento de los datos obtenidos para que los datos sean utilizables en análisis.
  - **Análisis de datos:** Revisión de los resultados obtenidos para asegurar que satisfagan las necesidades del analista de datos, si no se inicia el proceso iterativo.
  - **Respuesta a la cuestión**: Producción del reporte final luego del proceso iterativo.
  - **Institucionalización**: Si existe necesidad de elaborar un reporte de forma repetitiva, entonces se debe convertir éste en un conjunto de requerimientos a ser satisfechos mediante una operación de ocurrencia regular.

2.2.2.3.1.2.2. Metodología Ralph Kimball

Según Ilbay, E. [31], el ciclo de vida de Kimball es una metodología paso a paso para diseñar, desarrollar y desplegar DataMarts y DataWarehouses.

- **Planificación del Proyecto**. Este es el primer paso que se debe efectuar al iniciar la construcción de un DataWarehouse: definir el proyecto. En esta etapa se debe determinar la preparación de la organización para afrontar dicho proyecto. También se debe elaborar el plan para el proyecto, así como gestionar la puesta en marcha del mismo, definiendo y manteniendo su alcance.

  A nivel de planificación del proyecto, establece la identidad del mismo, el personal, los usuarios, lideres, gerentes del proyecto, equipos y roles.

- **Definición de Requerimientos del Negocio.** Cada organización es única en sí misma, cada vez que se inicia un DataWarehouse, es imposible conocer en avance los requerimientos de tal instrumento de apoyo a la toma de decisiones, por tanto se debe de hacer uso de entrevistas o sesiones con facilitador para lograr obtener datos de la información necesaria en la empresa para poder definir de manera correcta el contenido y utilización del DataWarehouse.

- **Diseño de Datos o Modelado Dimensional.** Según su creador Ralph Kimball, es el diseño físico y lógico que transformará las antiguas fuentes de datos en las estructuras finales del DataWarehouse, a través de una técnica que busca la presentación de los datos en un marco de trabajo estándar que es intuitivo y permite un acceso de alto desempeño.

Cada modelo dimensional está compuesto de una tabla que tiene una llave compuesta llamada tabla de hechos y un conjunto de tablas más pequeñas llamadas dimensiones. Cada tabla dimensión tiene una llave primaria simple, que corresponde exactamente a una de las partes de la llave compuesta en la tabla de hechos. Esta estructura característica es *usualmente llamada esquema* estrella.

- **Diseño y Desarrollo de Presentación de Datos.** Las principales sub-etapas de esta zona de la metodología son: la extracción, transformación y carga (ETL). Wayne E. y White C. [32] mencionan: ETL, es el proceso que permite a las organizaciones mover datos desde múltiples fuentes, reformatearlos y limpiarlos, y cargarlos en otra base de datos, datamart, o data warehouse para analizar, o en otro sistema operacional para apoyar un proceso de negocio. Los procesos ETL también se pueden utilizar para la integración con sistemas heredados (aplicaciones antiguas existentes en las organizaciones que se han de integrar con los nuevos aplicativos, por ejemplo, ERP´s. La tecnología utilizada en dichas aplicaciones puede hacer difícil la integración con los nuevos programas).

  Todas estas tareas son altamente críticas pues tienen que ver con la materia prima del datawarehouse: los datos.

- **Construcción de Aplicaciones a Usuarios Finales.** El desarrollo de las aplicaciones de los usuarios finales involucra configuraciones de la construcción de reportes específicos.

- **Despliegue y Crecimiento.** El despliegue exitoso de un Data Warehouse requiere planeación consistente y coordinación previa a la culminación de los esfuerzos de desarrollo. Un apropiado mantenimiento y crecimiento evidencian el éxito de llevar a cabo un proyecto importante como es un Data Warehouse, una correcta gestión que ponga en primer lugar satisfacer a los usuarios de negocio, sin sacrificar atención al back room y entorno técnico permite asegurar una adecuada evolución del DataWarehouse si es acompañado de mediciones y rastreo en el Data Warehouse y retroalimentación de parte de los usuarios.

2.3. Definición de Términos Básicos

I + D: Símbolo de Investigación y Desarrollo, que se aplica a los departamentos de investigación públicos o privados encaminados al desarrollo de nuevos productos o la mejora de los existentes por medio de la investigación científica.

NoSQL: Definido como Not only SQL, es un tipo de sistema de gestión de bases de datos que pretende ser la siguiente generación sobre estas tecnologías. Son no relacionales, distribuida, de códigos abiertos, horizontalmente escalables y más rápidos, ya que no implementa las propiedades ACID para asegurar la confiabilidad de las transacciones sobre las bases de datos. Entre las características que posee NoSQL se encuentra que no presentan esquemas, tienen fácil soporte de replicación, API simple, eventualmente consistente (conocido como BASE, y contrario al concepto de ACID) y contienen enormes cantidades de datos.

XML: Es un lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible. Permite definir la gramática de lenguajes específicos para estructurar documentos grandes.

ERP: Sistema de planificación de los recursos y de gestión de la información que, de una forma estructurada, satisface la demanda de necesidades de la

gestión empresarial. Se trata de un programa de software integrado que permite a las empresas evaluar, controlar y gestionar más fácil su negocio en todos los ámbitos.

CASSANDRA: Base de datos no relacional escrita en Java, de tipo Columna-Familia, de código abierto por Facebook en 2008, diseñada por Avinash Lakshman. Altamente escalable, eventualmente consistente, distribuida y almacenamiento estructurado key-value.

MONGODB: Base de datos no relacional creada por la compañía 10gen, su nombre proviene de la palabra "humongous" que se traduce como enorme. Escrita en lenguaje C++, de código abierto, orientada a documentos, y pensada para ser escalable y de desarrollo ágil.

COUCHDB: es un sistema de base de datos documental orientado a documentos, que se distribuye bajo licencia Open Source. CouchDB no utiliza esquemas predefinidos como las bases de datos relacionales tradicionales, es decir no tendremos tablas, columnas, llaves primarias, llaves foráneas, joins, relaciones, etc. En lugar de ello CouchDB almacena los datos como documentos.

TRIGGER: Es un procedimiento que se ejecuta cuando se cumple una condición establecida al realizar una operación. Dependiendo de la base de datos, los triggers pueden ser de inserción, actualización o borrado. Algunas bases de datos pueden ejecutar triggers al crear, borrar o editar usuarios, tablas, bases de datos u otros objetos.

ACID: Acrónimo de Atomicity, Consistency, Isolation, Durability. Modelo que garantiza la atomicidad, consistencia, aislamiento y durabilidad de la base de datos. Esto es que todas las operaciones se deben realizar completamente o de lo contrario se regresa el sistema al estado antes del cambio; los datos a modificar sean los correctos; todas las operaciones sean independientes entre sí; y los cambios realizados exitosamente perduren.

BASE: Acrónimo de Basically Available, Soft state, Eventual consistency. Modelo alternativo a ACID que es básicamente disponible, de estado ligero y eventualmente consistente, es decir que no asegura la disponibilidad, el estado del sistema puede cambiar eventualmente incluso sin modificaciones y

finalmente, el sistema llegará a un estado de consistencia con el tiempo mientras no reciba más ingresos.

JSON: JavaScript Object Notation. Es un formato para almacenar e intercambiar datos, independiente del lenguaje de programación que se esté utilizando

API: Traducido como Interfaz de Programación de Aplicaciones, son librerías para ser utilizadas por otras aplicaciones o servicios que permiten el uso de métodos, funciones, procedimientos, etc.

BI: Business Intelligence, traducido como Inteligencia de Negocios, es un término que incluye aplicaciones, infraestructura, herramientas y las mejores prácticas que permiten el acceso y análisis de la información para mejorar y optimizar decisiones y rendimientos.

ETL: Son las siglas en inglés de Extraer, Transformar y Cargar. Es un proceso que permite manipular para facilitar el uso en diferentes sistemas.

LMI: Son sistemas que utilizan técnicas lingüísticas, usando la semántica y la sintaxis para identificar términos, encontrar unidades compuestas de varias palabras o caracterizar la estructura interna de una frase o documento

Marketing Mix: Es la combinación de los elementos de marketing que se emplean para satisfacer los objetivos de la organización y el individuo. Los elementos de la mezcla original son producto, precio, promoción y plaza o distribución

Startup: Utilizado actualmente en el mundo empresarial que traduce arrancar, emprender o simplemente montar un nuevo negocio y hace referencia como su nombre lo indica a ideas de negocio que apenas empiezan o están en construcción, es decir, son empresas emergentes apoyadas en la tecnología y la calidad con un alto nivel de proyección, a pesar de su corta trayectoria y a la falta de recursos o financiación que puede enfrentar un negocio cuando apenas empieza.

Aplicaciones Web: aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador

CAPÍTULO III. MATERIALES Y MÉTODOS

3. MATERIALES Y MÉTODOS

En el mundo, las empresas deben hacer frente a los desafíos del área de marketing y ventas, por esta razón, en la actualidad existen nuevos enfoques tecnológicos que utilizan técnicas avanzadas para apoyar a varias soluciones en este sector, tales como la información que se genera día a día, debido a la cantidad de información que se maneja. Uno de estos enfoques son las bases de datos NOSQL, por eso, en este capítulo se tiene como objetivo principal abordar en detalle cada uno de los elementos necesarios que para la implementación de una base de datos NOSQL que permita almacenar y accesar a información importantes que luego serán importantes para la toma de decisiones en el proceso de segmentación de clientes

a) Procedimientos

3.1. Metodología Propuesta

En éste apartado veremos la metodología propuesta a seguir en el presente trabajo de investigación. Según Sevilla, E. [30], es un hecho real que casi ningún equipo en un proyecto de DataWarehouse tradicional construye un DataWarehouse utilizando una metodología pura. Básicamente cada equipo está construyendo su propio enfoque y utilizando algunas de sus propias técnicas, una vez que han asimilado lo mejor de las metodologías que elijan.

La metodología propuesta es una metodología hibrida que tiene como base fases de la metodología de Ralph Kimball para DataWarhouse tradicionales y la metodología de Web Mining, cada una de estas fases tendrá subprocesos que serán importantes para el desarrollo de éste trabajo. Cada fase con sus subprocesos serán detalladas y estructuradas de manera que sirva como guía para el desarrollo; todo esto dentro del marco de las bases teóricas citadas en el capítulo anterior y de acuerdo a técnicas y/o

herramientas propuestas por el autor de ésta investigación. Por ello, ésta guía significa una ayuda para entender cómo será el desarrollo de ésta investigación de acuerdo a los que se rige.

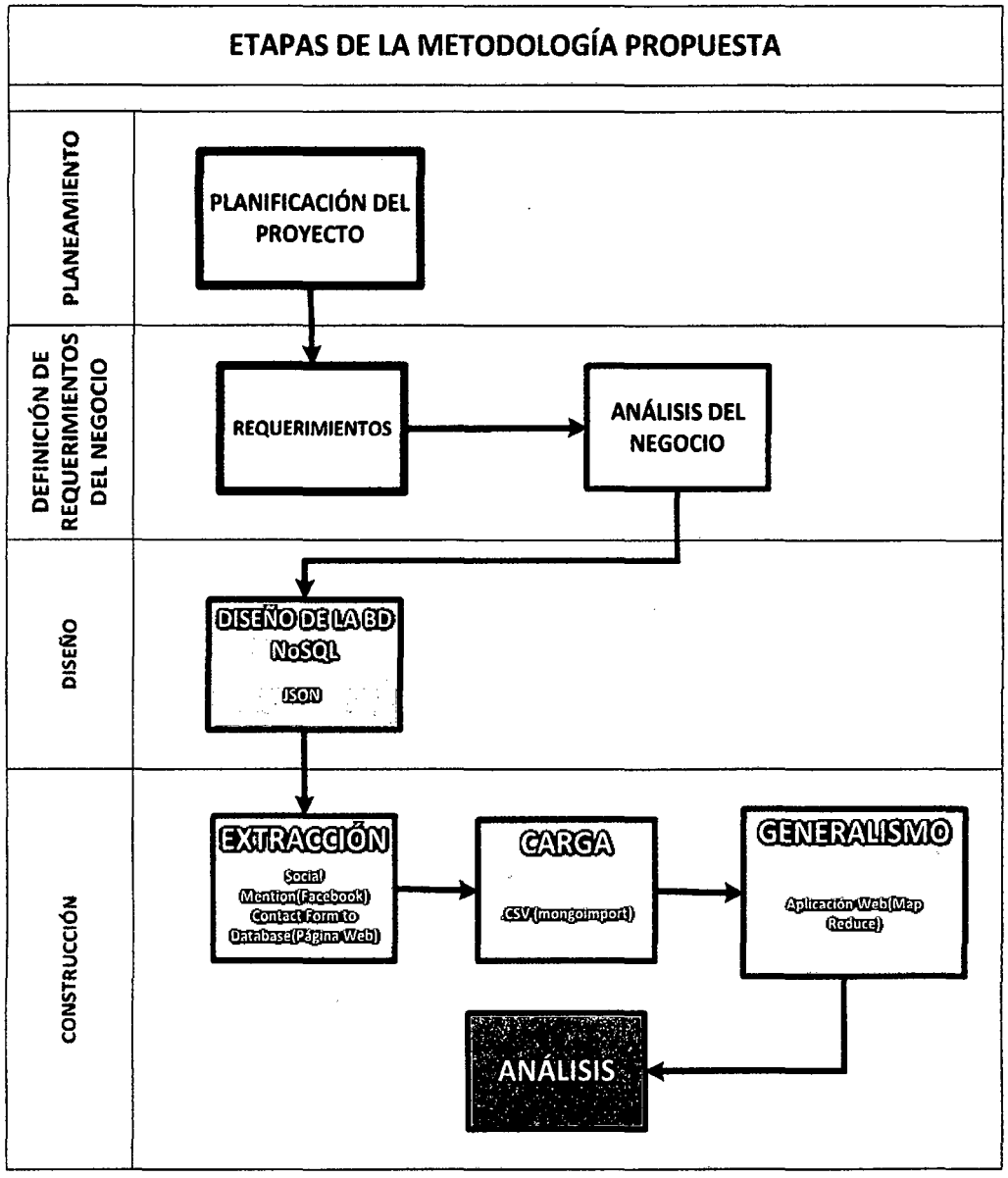


Figura 10. Etapas de Metodología Propuesta
(Por el autor)

3.2. Análisis de las metodologías existentes para la definición de la metodología propuesta

- De acuerdo con las bases teóricas citadas en el Capítulo II, la mayoría de metodologías tradicionales coinciden en que es necesario al iniciar el

proyecto analizar el modelo de datos empresarial con el que cuenta la empresa, es decir son muy importantes las etapas de Planificación del Proyecto y la etapa de Definición de Requerimientos del Negocio. En éste sentido, se está tomando en cuenta ambas etapas para la metodología propuesta.

- A partir de lo mencionado anteriormente, las siguientes etapas como el Diseño y Construcción serán desarrolladas en base a la metodología Web Mining, la cual nos brinda una guía básica para el uso y desarrollo de conceptos, técnicas y/o herramientas que tiene que ver con el proceso global de descubrir información o conocimiento potencialmente útil y previamente desconocido a partir de datos de la Web que es a lo que se rige éste trabajo.

- Cabe resaltar que no se han tomado en cuentas las etapas de la metodología de Kimball a partir de la etapa de Diseño hacia delante ya que dentro de éstas, de acuerdo con Sevilla, E. [30], el punto central de la metodología de Kimball es el modelado dimensional. Kimball propone un esquema de denormalización del Diagrama Entidad-Relación (ERD) para identificar procesos discretos de negocios (áreas de interés) con sus posibles tablas de hechos y dimensiones. Luego, selecciona un subconjunto de datos para modelarlo utilizando el esquema estrella y continuar el desarrollo del DataWarehouse de forma iterativa, modelando un nuevo subconjunto cada vez. Aquí Kimball se enfoca en bases de datos relacionales, ERD, tablas, lenguaje SQL, etc., un enfoque en el cual no está enmarcado éste trabajo.

3.3. Etapas de la metodología propuesta

La metodología propuesta no pretende ahondar o profundizar los conceptos abordados con mayor o menor amplitud en el marco teórico sino más bien componer con los conceptos destilados un nuevo enfoque reconciliado que sirvan para determinar la estrategia de desarrollo de éste trabajo de investigación. Es importante mencionar que para determinar la metodología o los enunciados de la metodología; fue conveniente incluirse en otras, teniendo como base principal el de ser acorde al objetivo del desarrollo.

3.3.1. Planificación del Proyecto. Busca identificar la definición y el alcance del proyecto, las justificaciones del negocio y evaluaciones de factibilidad. La planificación del proyecto se focaliza sobre recursos, perfiles, tareas, duraciones y secuencialidad.

La planificación del proyecto es dependiente al negocio y sus requerimientos, ya que los requerimientos del negocio determinan el alcance del proyecto, definen los recursos necesarios, etc. Esta etapa identifica el escenario del proyecto para saber dónde surge la necesidad del mismo.

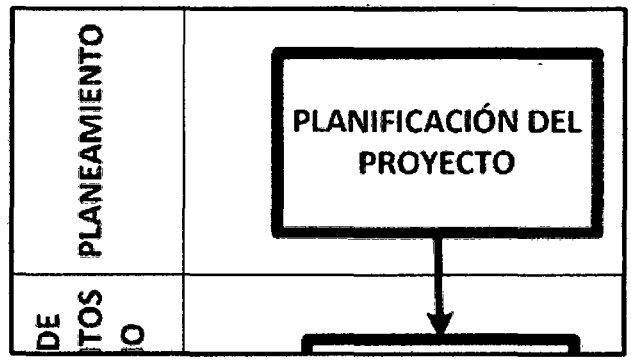


Figura 11. Etapa Planificación del Proyecto

(Por el autor)

Algunos factores asociados con esta etapa son:

- ✓ Identificación de Roles y equipos.
- ✓ Cooperación entre áreas y negocios de sistemas.
- ✓ Cultura analítica de la organización.

3.3.2. Definición de Requerimientos del Negocio. Un factor determinante en el éxito del proyecto es la interpretación de correcta de los diferentes requerimientos expresados por los diferentes niveles de usuarios.

El objetivo de ésta fase es definir las necesidades de negocio y funcionales que puedan requerir los usuarios finales del trabajo. Para ello se necesita tomas de requisitos que mantendrá el equipo de trabajo con los distintos representantes del departamento usuario final del sistema.

Se trata de dar sentido al sistema, porque los requerimientos identificados en ésta fase, serán los que trate de optimizar el diseño del proyecto. El resto de etapas que forman parte del proceso se nutren de la información definida en éste apartado, cuánto más claros sean los requerimientos menos dudas surgirán en el proceso de diseño y construcción a la hora de tomar decisiones sobre cómo dar solución a las necesidades requeridas por los usuarios.

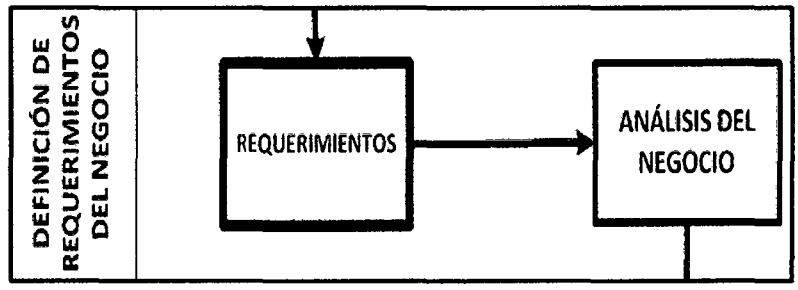


Figura 12. Etapa de Definición de Requerimientos del Negocio
(Por el autor)

Se realizará el estudio de los sistemas de información existentes, que ayudarán a comprender las carencias actuales en los sistemas encargados de generar los informes de negocio y explotación de los datos para que nuestra propuesta optimice su solución. Se conocerán cuáles son las consultas que más se realizan, la periodicidad con que se hacen, el perfil de los usuarios que la realiza etc.

3.3.3. Diseño Físico. En ésta etapa se focaliza en el diseño físico de la base de datos NoSQL para el almacenamiento y gestión de los datos que posteriormente en las siguientes etapas serán extraídos y procesados de la web. El diseño físico de la base de datos NoSQL es importante para identificar que colecciones (tablas en SQL), documentos (filas en SQL), anidaciones o indexaciones (claves foráneas en SQL), serán tomados en cuenta para el desarrollo de la propuesta en el presente trabajo de investigación.

Además es importante, pues en ésta etapa se definirá el tipo de Base de datos NoSQL que será necesario implementar de acuerdo a las necesidades y análisis de requerimientos del negocio.

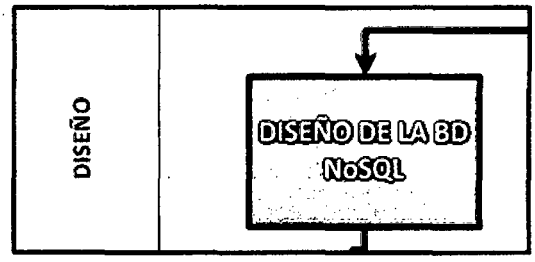


Figura 13. Etapa de Diseño
(Por el autor)

3.3.4. Construcción. Consta de 4 sub-etapas:

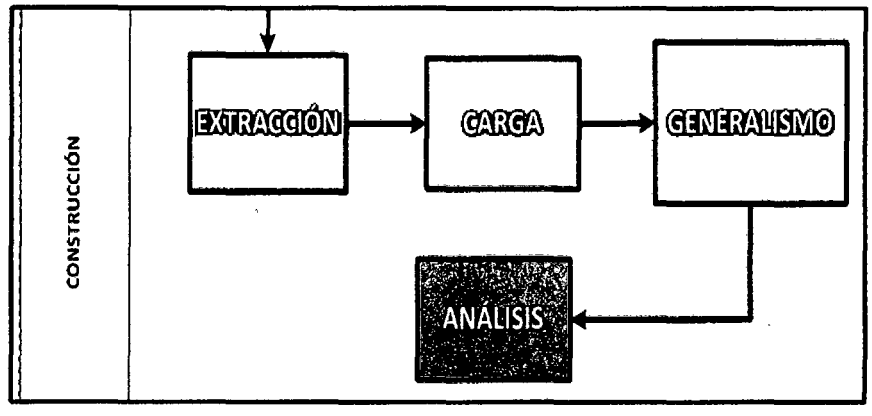


Figura 14. Etapa de Construcción
(Por el autor)

- **Extracción**:

  Es una sub etapa de la construcción, la cual servirá para la identificación de los datos a analizar provenientes de la web y luego la extracción de los mismos a través de técnicas y/o herramientas adecuadas para éste proceso.

  Debido a la naturaleza de la Web, en ésta etapa se centra en sitios Web específicos para extraer información. También en el aprendizaje automático o técnicas de minería de datos para realizar el aprendizaje de patrones y reglas de los documentos de forma automática o semi-automática.

  Los resultados de esta etapa podrían estar en la "forma" de una base de datos estructurada o un resumen de los textos o documentos originales, por poner un ejemplo.

❖ **Carga**:

Una vez extraídos los datos de la web ya de forma estructurada, vamos a almacenarlos en una base de datos no relacional debido a que estos tipos de bases de datos de acuerdo con las bases teóricas citadas tienen ventajas como escalabilidad, almacenamiento de grandes volúmenes de datos, mejores tiempo de respuesta, etc.

Esta etapa es importante porque se verán procesos de gestión de bases de datos NoSQL, de acuerdo al tipo de las mismas. Teniendo una base de conocimiento de la gestión de bases de datos NoSQL, el proceso de carga de datos a la BD se hará de forma manual o masiva; siendo recomendable y convenientemente de forma masiva.

❖ **Generalismo**:

Esta etapa sirve para el proceso de reconocimiento de patrones, este proceso se basa en gestionar la información almacenada en la base de datos y propone técnicas relacionadas con conceptos del mundo no relacional (NoSQL) para obtener un resultado específico.

Según lo mencionado en las bases teóricas, existen diversas formas de realizar esta asignación, entre las que cabe destacar las siguientes:

- ✓ Extracción automática de los términos más comunes de un texto
- ✓ Extracción automática de términos usando información sintáctica.
- ✓ MapReduce.

Se optará por aplicar el modelo de programación Map-Reduce, el cual es menos complejo de aplicar a un conjunto de datos almacenados en una base no relacional.

Además en esta sub-etapa se construirá una pequeña aplicación para el que los usuarios finales interactúen a través de ella.

- **Análisis**:

  Ésta última sube-etapa, tiene como objetivo analizar, comprender, visualizar e interpretar los patrones obtenidos en la sub-etapa anterior, para finalmente ayude en la toma de decisiones.

En definitiva, la metodología propuesta está basada en el uso de las técnicas de minería de datos tradicionales y no tradicionales (Web minnig) aplicadas a la búsqueda, extracción y evaluación automática de información para el descubrimiento del conocimiento de los documentos y servicios de la web.

3.4. Desarrollo de la metodología propuesta

En este apartado de éste subcapítulo pondremos en marcha la metodología propuesta para éste trabajo de investigación.

3.4.1. Planificación del Proyecto

A. El Negocio

- **Descripción de la Organización:**

  El Centro de Actualización Profesional para Ingenierías, CAPI EIRL, creada el 11 de Noviembre del 2008, en la ciudad de Piura, es una empresa de derecho privado, cuya finalidad es promover y desarrollar diplomados y cursos de actualización profesional para ingenierías, así como la asesoría y consultoría a empresas públicas y privadas, capacitando y difundiendo las tecnologías de la construcción e investigación académica, desarrollando proyectos de desarrollo en las diversas especialidades de la ingenierías en el Perú.

  - **Organización**: Centro de Actualización Profesional para Ingenierías CAPI
  - **Área**: Marketing y Ventas
  - **Responsable**: Lic. Tania Rojas

3.4.2. Definición de Requerimientos del Negocio

A. El negocio

- **Misión:** Proporcionar capacitaciones profesionales a nivel superior, Asesoría y consultaría en ingeniería y tecnología, Promover Alianzas Estratégicas, contribuyendo a elevar el nivel de educativo con calidad y competitividad en el mundo globalizado a los profesionales Ingenieros del Perú

- **Visión:** Ser una empresa reconocida, distinguida, renombrada, enfocada a la enseñanza a nivel superior de profesionales de la ingeniería en el Perú, desarrollando diplomados y cursos de actualización técnico-profesional, con excelencia académica, promoviendo la mejora continua con innovación de alta calidad educativa.

- **Descripción del proceso: Segmentación de clientes.** El proceso de segmentación de clientes empieza cuando se apertura algún diplomado o curso de software en alguna filial específica; para hacer las campañas de marketing por internet, los encargados solicitan al área de administración los correos electrónicos de todos los ingenieros, ya sean correos obtenidos por visitas o correos de todos los ingenieros inscritos en el colegio de Ingenieros del Perú. Una vez con la lista de correos se envía información acerca del diplomado o curso a todos de forma masiva sin excepción.

  De esta manera entonces, consideramos que esta forma de hacer campañas de marketing por internet es deficiente, pues no se toma en cuenta para éste proceso factores críticos como el tipo de diplomado o curso que se está ofreciendo. Estos factores deben considerarse con el objetivo que soporte de manera consistente, acertada y coherente el proceso de segmentación de clientes.

- **Los problemas del negocio**

  El Centro de Actualización Profesional para Ingenierías CAPI, dentro de área de ventas y marketing tiene problemas

en el proceso de la segmentación de clientes, esto debido a la falta de análisis de datos que se deberían tener.

- **Operativa**

  El proceso para la segmentación de clientes de manera virtual dentro del CAPI es ineficiente; se ha hecho durante varios años sin tomar en cuenta factores relevantes que muestren con precisión los servicios a ofrecer. El proceso se hace de manera manual y empírica de manera que se pierde tiempo y la automatización es inadecuada.

- **Económica**

  Otro aspecto importante es que no se toma en cuenta preferencias de los consumidores, las cuales se encuentran en bruto dentro de las redes sociales en donde se promocionan los diplomados y/o cursos. Esto conlleva a que se aperturen estos diplomados de manera espontánea sin un análisis previo, es allí donde se podría estar perdiendo ingresos para la organización.

❖ **Selección de entrevistados**

Encargados del Área de Ventas y Marketing: Para extraer información acerca del proceso de segmentación de clientes dentro de las campañas de marketing por internet en el CAPI.

3.4.3. Diseño

En esta etapa se verá todo lo relacionado a las bases de datos NoSQL y a la vez diseñará la misma.

3.4.3.1. Análisis comparativo de bases de datos NOSQL

El desarrollo web es uno de los principales beneficiados de las bases de datos NoSQL, pero ¿qué pasa si la escalabilidad no es una de nuestras preocupaciones? Pues definitivamente existen otros motivos por los que el uso de NoSQL es

perfectamente viable. Veamos algunas de sus principales características que prevalecen a la las Bases de datos SQL: [1]

3.4.3.1.1. Características

- La base de datos es de crucial importancia en el desarrollo en general, y el desarrollo web no es la excepción. Los desarrolladores web se ven forzados a pensar en entidades, joins, agregados, relaciones, llaves primarias, llaves foráneas, restricciones, etc. Si somos observadores nos daremos cuenta los datos nunca se almacenarán en la base tal como son presentados al usuario final, hasta en la más mínima operación intervendrán dos o más tablas; como desarrolladores tendremos que interpretar los errores dados por ejemplo al tratar de eliminar un registro del que dependen otras tablas, y mostrar un mensaje entendible para el usuario.
- Las bases de datos NoSQL al no tener un esquema estático (pues basta con definir el nombre de la colección) brinda al desarrollador alta flexibilidad al momento de agregar campos por ejemplo de determinado objeto; es decir el desarrollador diseña su esquema al tiempo que programa.
- Las bases de datos NoSQL, principalmente las basadas en documentos (CouchDB y MongoDB) brindan un alto grado de comodidad al desarrollador pues la forma de grabar los datos es mucho más parecida a la realidad, además utilizan JSON como formato para almacenar los datos, la sencillez de JSON lo hace muy fácil de comprender, interpretar y manipular desde nuestro lenguaje de programación.

3.4.3.1.2. Evaluación de las Bases de Datos NoSQL

Para la presente investigación no se incluirá en la evaluación a las bases de datos NoSQL orientadas a grafos. Se ha seleccionado las siguientes herramientas para la evaluación:

- ✓ Cassandra
- ✓ CouchDB
- ✓ MongoDB

A continuación realizaremos un cuadro comparativo de las tres herramientas con el fin de evaluar los escenarios de aplicación de cada una de ellas, ventajas y desventajas, etc.

Tabla 5. Cuadro Comparativo - Almacenamiento y Modelo de Datos

CARACTERÍSTICAS TÉCNICAS	
A. ALMACENAMIENTO Y MODELO DE DATOS	
Cassandra	✓ Utiliza el modelo llave/valor. ✓ Para grabar colecciones dentro de un campos utiliza las denominadas "súper Column Family"
CouchDB	✓ Graba los datos en forma documental, apoyada en el modelo llave/valor. ✓ Utiliza JSON puro. ✓ Podemos almacenar subdocumentos como valores de los campos.
MongoDB	✓ Graba los datos en forma documental, apoyada en el modelo llave/valor. ✓ Utiliza la versión binaria de JSON denominada BSON.

Tabla 6. Cuadro Comparativo - Interfaces

CARACTERÍSTICAS TÉCNICAS	
B. INTERFACES	
Cassandra	✓ Para las conexiones a la base utiliza su propio protocolo de comunicación denominado Thrift.
CouchDB	✓ Para conexiones a la base utiliza una API RESTful HTTP que funciona mediante servicios web. CouchBase (la versión comercial) ofrece drivers para usarlos con varios lenguajes de programación.

MongoDB	✓ Dispone de un gran número de drivers nativos oficiales para usarse con varios lenguajes de programación.

Tabla 7. Cuadro Comparativo - Escalabilidad

CARACTERÍSTICAS TÉCNICAS	
C. ESCALABILIDAD HORIZONTAL	
Cassandra	✓ Basada en replicación. ✓ Alta tolerancia a fallos.
CouchDB	✓ Para la replicación utiliza su API RESTful HTTP. ✓ Permite resumir la sincronización de los datos entre los nodos si existiera algún error de hardware o conectividad.
MongoDB	✓ Permite escalar usando su función de Auto sharding o de auto segmentar los datos en sus diferentes nodos de trabajo.

Tabla 8. Cuadro Comparativo - Consultas Dinámicas

CARACTERÍSTICAS TÉCNICAS	
D. CONSULTAS DINÁMICAS	
Cassandra	✓ Soporta consultas dinámicas.
CouchDB	✓ No soporta consultas dinámicas, las consultas deben ser programadas para luego ser consumidas.
MongoDB	✓ Soporta consultas dinámicas.

Tabla 9. Cuadro Comparativo - Plataformas Soportadas

CARACTERÍSTICAS TÉCNICAS	
E. PLATAFORMAS SOPORTADAS	
Cassandra	✓ Windows ✓ Linux ✓ Mac OS X
CouchDB	✓ Mac OS X ✓ Linux ✓ Solaris ✓ BSD ✓ Android (plataforma móvil)
MongoDB	✓ Windows ✓ Linux ✓ Mac OS X ✓ Solaris

Tabla 10. Cuadro Comparativo - Drivers para Lenguajes de Programación

CARACTERÍSTICAS TÉCNICAS	
E. DRIVER NATIVOS OFICIALES PARA LENGUAJES DE PROGRAMACIÓN	
Cassandra	✓ Java ✓ Python
CouchDB	✓ Soporta todos los lenguajes de programación que puedan trabajar con servicios web vía su API RESTful usando JSON.
MongoDB	✓ C ✓ C++ ✓ Erlang ✓ Haskell ✓ Java ✓ JavaScript ✓ .NET (C# F#, PowerShell, etc) ✓ Perl ✓ PHP ✓ Python ✓ Ruby ✓ Scala

Luego de comparar las características de las herramientas seleccionadas es hora de que evaluemos los mejores escenarios en los que cada una de las bases puede ser aplicada:

- Comencemos por Cassandra, debemos mencionar que independientemente al hecho que esté desarrollada en Java, se integra muy bien en aplicaciones desarrolladas con dicha tecnología. Sería ideal para aplicarla en sistema de logins, y para realizar minería de datos e inteligencia de negocios basada en el análisis de datos. Es un poco difícil de usarla hasta acostumbrarse a la herramienta.
- CouchDB veríamos usarla en aplicaciones cuyos datos no sean cambiantes debido al limitante de no soportar consultas dinámicas, que es uno de los principales inconvenientes que ésta herramienta presta. Aun así esta base de datos es muy recomendada para quienes necesiten implementar réplicas maestro/maestro que estén distribuidas geográficamente y necesiten las capacidades de sincronización que CouchDB utiliza. También es recomendad como una base de datos embebida para

móviles por ejemplo, ya que aprovecharía sus capacidades de sincronización fuera de línea.

- Finalmente MongoDB es una herramienta recomendada para quienes necesitan alto desempeño, para quienes desean altas velocidades de escritura de datos; pero principalmente para quienes necesiten una base de datos flexible, fácil de usar, rápida, con buena integración en los lenguajes de programación, *así como buena comunidad* y documentación.

3.4.3.1.3. Conclusiones del análisis

- ✓ Aunque generalmente los desarrolladores web tengan conocimientos acerca de bases de datos y SQL, el simple hecho de que un analista haya diseñado una base para determinado proyecto web involucra que el desarrollador tenga que aprender de dicho diseño para poder plasmarlo en el aplicativo; vale la pena recalcar que no hablamos de sistemas ERP, o te transacciones en línea de un banco, en lo absoluto, hablamos de aplicaciones web orientadas a servicios del tipo "startup" en los que alguien tuvo una idea de un servicio novedoso que beneficiaría mucho a sus usuarios.
- ✓ No debemos dejar de lado también la posibilidad de implementar soluciones hibridas que usen bases de datos relacionales y NoSQL en conjunto, por ejemplo Facebook usa MySQL para ciertos datos y Cassandra para cubrir otros requerimientos.
- ✓ El simple hecho de conocer más herramientas, sus principales características (ventajas, desventajas, carencias, etc.) y los posibles escenarios de aplicación permite que siempre dispongamos de la herramienta correcta para cubrir determinado trabajo.
- ✓ En conclusión las bases de datos NoSQL deberían considerarse como una herramienta más para el desarrollo de proyectos de software, pues como se mencionó en textos anteriores no buscan reemplazar a las tradicionales relacionales; la razón de ser de las bases de

datos NoSQL se resume en ofrecer características como sencillez, escalabilidad, y demás que la convierten en una alternativa muy funcional para ciertos proyectos.

3.4.3.2. Elección de Motor de Base de Datos NoSQL

De acuerdo al análisis realizado anteriormente se eligió a **MongoDB** como el motor de base de datos a trabajar; es una herramienta recomendada para quienes necesitan alto desempeño, para quienes desean altas velocidades de escritura de datos; pero principalmente para quienes necesiten una base de datos flexible, fácil de usar, rápida, con buena integración en los lenguajes de programación, así como buena comunidad y documentación. En general para todo proyecto en el que usaríamos MySQL podríamos también usar MongoDB sabiendo sus virtudes y sus defectos en función al tipo de proyecto que estemos implementando. En el siguiente punto incluiremos como funciona este tipo de motor de base de datos no relacional NoSQL.

3.4.3.3. Diseño del Modelo NoSQL

Es importante tener en cuenta, que el proceso para transformar un modelo relacional a no relacional se llama desnormalización, este proceso busca optimizar el desempeño de una base de datos. Cabe resaltar que para este trabajo de investigación, es de gran importancia la creación del modelo para la base de datos, pues es el requisito fundamental para la implementación del caso de estudio, el cual debe cumplir con especificaciones para el almacenamiento de la información recolectada. [2]

Para efectos de comprender mejor el modelado de la base de datos NoSQL se presenta un ejemplo sencillo de un esquema o modelo lógico en SQL.

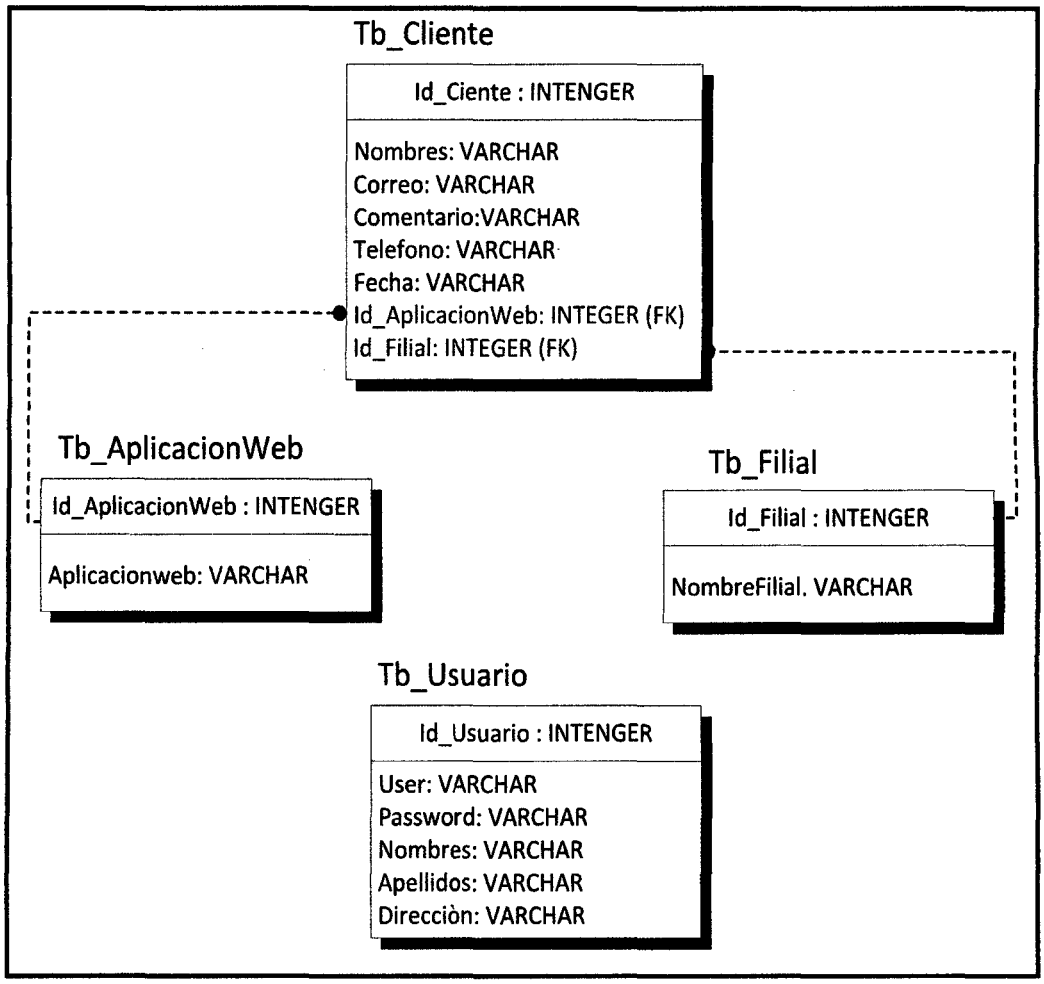


Figura 15. Ejemplo de Esquema Relacional de la Base de Datos

(Por el Autor)

Ahora es posible transformar el esquema anterior a un modelo NoSQL de forma anidada, seleccionando la tabla principal, es decir la que presenta mayor relevancia para en proceso de negocio, puesto que es convertida en la contenedora de todos los atributos de las tablas con las que tiene relación.

Para elegir la tabla madre, es decir la tabla principal del modelo, es necesario saber dónde están las llaves foráneas en el modelo relacional, puesto que en esa tabla se encontrarán relaciones de 1 a 1.

- ✓ Cliente: Esta tabla será considerada como la tabla madre que contendrá a la tabla Profesión y Filial, debido a que estas dos últimas tablas le proporcionan claves foráneas.
- ✓ Filial y Profesión: en el modelo NoSQL anidado, estas tablas serán embebidas dentro de la tabla Cliente.
- ✓ Usuario: Tomando en cuenta que las bases de datos NoSQL no tienen un esquema específico, tomaremos de forma separada la tabla Usuario en el modelo NoSQL.

De aquí en adelante en el modelo NoSQL anidado y de acuerdo a lo citado en el tipo de base de datos elegida MongoDB la tablas recibirán el nombre de Colecciones, las filas recibirán el nombre de documento, etc.

A continuación mostramos de manera gráfica el modelo NoSQL anidado:

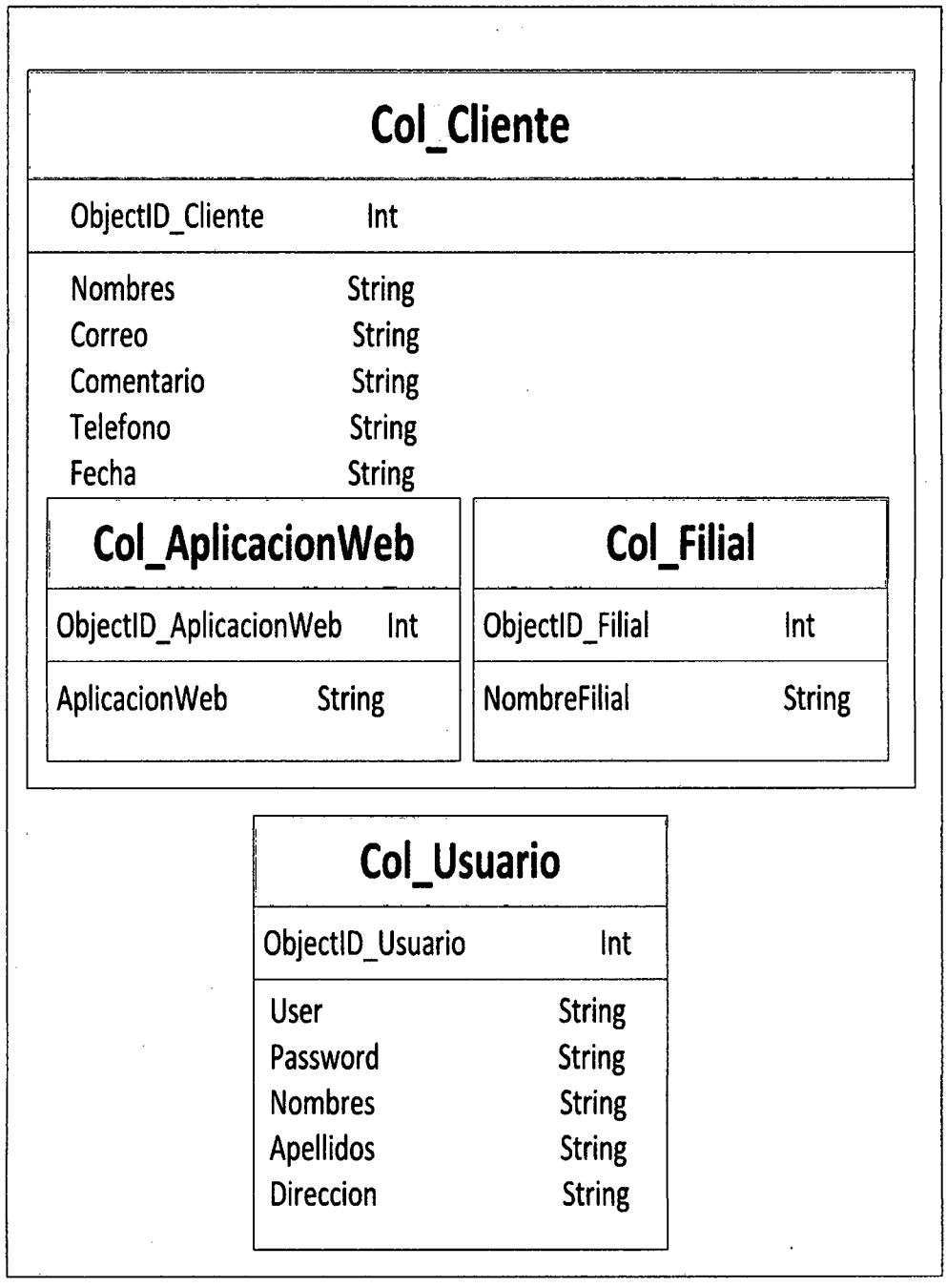


Figura 16. Modelo Anidado o Embebido de la Base de datos NoSQL
(Por el Autor)

Ahora representaremos con un ejemplo las colecciones en forma JSON:

Tabla 11. Colección "Cliente" en forma JSON

```
Cliente = {
        Id_: "ObjectId(2f602d787945c344bb4bda5)",
// El Id_ es generado por el motor de MongoDB

      Nombres: "Wilson Quiroz Peña",
      Correo: "wquiroz@gmail.com",
      Comentario: "Diplomado en Estructuras",
      Telefono: "976924873",
      Fecha: "10/03/2014",
}
```

Tabla 12. Colección "AplicacionWeb" en forma JSON

```
AplicacionWeb = {
        Id_: "ObjectId(4b682as5245c399sb4bos4)",
// El Id_ es generado por el motor de MongoDB

      AplicacionWeb: [
                      "Facebook",
                       "Página Web"
                        ]
}
```

Tabla 13. Colección "Filial" en forma JSON

```
Filial = {
        Id_: "ObjectId(9y8scd78794cd8s1bb45s8a)",
// El Id_ es generado por el motor de MongoDB

      NombreFilial: "Piura" }
```

Tabla 14. Colección "Usuario" en forma JSON

```
Usuario = {
    Id_: "ObjectId(4pd585s4d6s9787945sd47b)",
// El Id_ es generado por el motor de MongoDB

    User: "lmchavez",
    Password:"lm1234",
    Nombres: "Luis Miguel",
    Apellidos: "Chávez Quispe",
    Dirección:"Psje. Atahualpa 155"
}
```

Tabla 15. Modelo Anidado o Embebido de la Base de datos en forma JSON

```
Cliente = {
      Id_: "ObjectId(2f602d787945c344bb4bda5)",
// El Id_ es generado por el motor de MongoDB

   Nombres: "Wilson Quiroz Peña",
   Correo: "wquiroz@gmail.com",
   Comentario: "Diplomado en Estructuras",
   Telefono: "976924873",
   Fecha: "10/03/2014",


   AplicacionWeb: [
              {
              AplicacionWeb: "Página Web",

              }
               ],
  Filial: [
            {
             NombreFilial: "Piura" ,
            }
```

```
            ]
}
Usuario = {
    Id_: "ObjectId(4pd585sjf6s9787945sd47b)",
// El Id_ es generado por el motor de MongoDB

    User: "lmchavez",
    Password:"lm1234",
    Nombres: "Luis Miguel",
    Apellidos: "Chávez Quispe",
    Dirección:"Psje. Atahualpa 155"
}
```

3.4.3.4. Administración de Base de datos con MongoDB

3.4.3.4.1. Creación de la Base de datos

Para crearla basta con poner el siguiente comando con el nombre de la base de datos, como se muestra en el Shell de MongoDB.

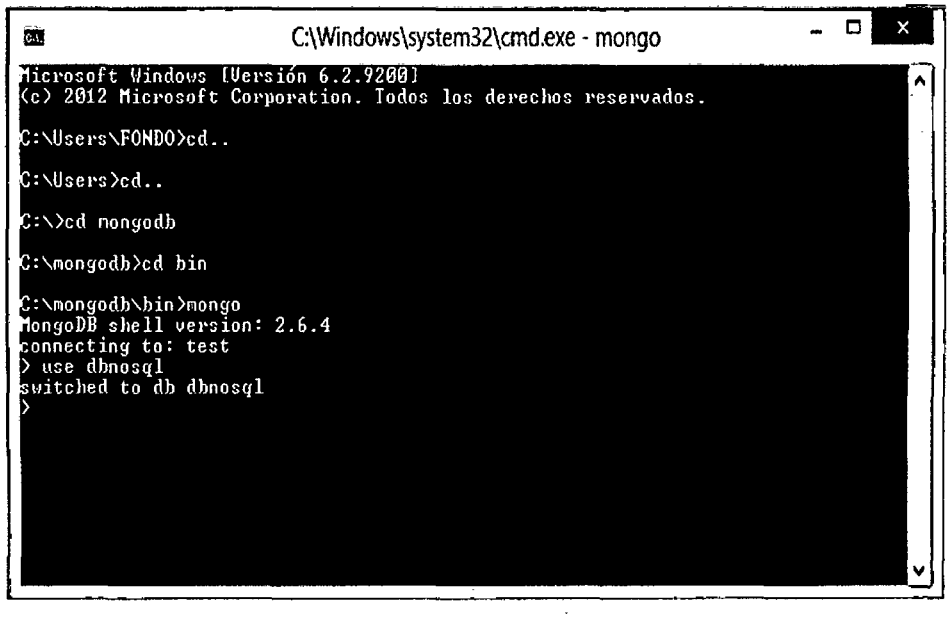


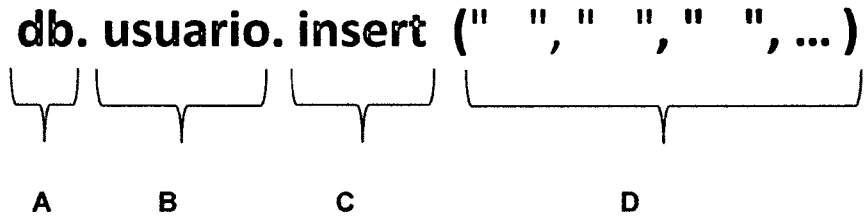
Figura 17. Creación de Base de Datos NoSQL- Shell MongoDB
(Por el autor)

3.4.3.4.2. Manipulación de Datos

Se refiere a las operaciones básicas como insertar, actualizar, eliminar, buscar, etc.

Es importante mencionar que las colecciones se crean cuando vamos a ejecutar la operación Insertar (.insert)

3.4.3.4.2.1. Insertar (.insert)



donde:

A: hace referencia la base de datos creada

B: crea la colección (nombre)

C: el comando insertar

D: son los llaves/valores a insertar

Veamos un ejemplo. Dentro de la ventana del Shell escribimos el siguiente comando:

Tabla 16. Comando Insert

```
> db.usuario.insert({
        user     : 'lmchavezq',
        password  : 'lmiguel',
        apellidos  : 'Chávez Quispe',
        nombres  : 'Luis Miguel',
        direccion  : 'Psj. Atahualpa 155'
});
```

3.4.3.4.2.2. Buscar (.find())

db. usuario. find( );

Comando "buscar"

Veamos un ejemplo. Si insertamos más documentos a la colección *usuario*, el comando buscar funcionará así:

Tabla 17. Comando find()

```
> db.usuario.find();
        { "_id" : ObjectId("5232344a2ad290346881464a"),
          "user" : "lmchavezq",
          "password" : "lmiguel",

          "nombre" : "Luis Miguel",
          "apellido" : "Chávez Quispe",
         "dirección" : "Psj. Atahualpa 155"
        }
        { "_id" : ObjectId("9751244a2ad2294465716744d"),
          "user" : "vcasasm",
          "password" : "ccaluaz",
          "nombre" : "Victor Jesús",
          "apellido" : "Casas Calua",
         "dirección" : "Jr. Chepen 178"
        }
```

Notemos que la búsqueda nos arroja los objetos resultantes, en este caso los documentos de los 2 usuarios que se han insertado acompañados del identificador único que crea MongoDB, este campo _id se toma además como índice por defecto.

3.4.3.4.2.3. Filtros:

Digamos que ahora queremos hacer la búsqueda pero filtrada por los algún parámetro. Para esto sólo debemos pasar el filtro deseado a la función **find()**, busquemos a un usuario por su user : "vcasasm"

Tabla 18. Filtros en NoSQL

```
> db.usuario.find({ user: ´ vcasasm ´});
        {
        "_id": ObjectId("9751244a2ad2294465716744d"),
        "user" : "vcasasm",
        "password" : "ccaluaz",
        "nombre" : "Victor Jesús",
        "apellido" : "Casas Calua",
        "dirección" : "Jr. Chepen 178"
        }
```

3.4.4. Construcción

Siguiendo con la metodología propuesta ahora nos centramos en la etapa de la Construcción la cual se subdivide en 3 sub- etapas: Extracción, Carga y Generalismo.

3.4.4.1. Extracción

El Centro de Actualización Profesional para Ingenierías – CAPI, actualmente cuenta con diferentes aplicaciones web en la cuales se almacenan tipos de datos, los cuales pueden ser importantes para almacenarlos en una base de datos NoSQL y que posteriormente ayuden en la generación de información *relevante para el proceso de* segmentación de clientes.

Estos datos se encuentran de manera "digital" en las aplicaciones web, y a los cuales es necesario darles una estructura adecuada que soporten las características de las bases de datos NoSQL.

En el presente trabajo de investigación, la extracción de los datos se aplicará a la red social Facebook y a la Portal Web de CAPI.

- **Extracción de Facebook**

  En el caso de la cuenta de la red social Facebook, CAPI cuenta con más de 5000 seguidores en ésta red social. En la presente investigación los datos *a extraer son: comentarios*, los cuales servirán para la identificación de palabras clave; éstos comentarios se generan de parte de los seguidores de Facebook específicamente a raíz de las constantes publicaciones y/o post que los administradores del área de Ventas y Marketing hacen. Cabe resaltar que éstas publicaciones son promociones de servicios que CAPI ofrece, como son diplomados y cursos de software en las distintas ramas de la Ingeniería.

  El proceso de captura de datos en el caso Facebook se hará mediante un software Online gratuito ***Social Mention,*** el cual permite la extracción de datos tanto de Facebook (palabras más usadas, likes por comentarios, número de seguidores, etc). Todos estos datos son extraídos mediante reportes a archivos Excel (.CSV), los cuales servirán para la administración de la base de datos NoSQL.

- **Extracción de Portal Web**

  En la actualidad CAPI cuenta con un Portal Web (www.capi.com.pe) para ofrecer sus servicios al público, el cual está construido en el gestor de contenidos WordPress. En dicho portal web se muestra la información detallada de cada diplomado y/o curso que se ofrece como: filiales donde están disponibles, plan de estudios, inversión, docentes, etc. Además de todo esto se indica la fecha de inicio de los diplomados y/o cursos de software en cada filial. En función de esto, el portal web también permite al usuario dejar comentarios u opiniones, los cuales se envía como correo electrónico a una cuenta de email específica uno por uno.

Para el proceso de extracción de éstos comentarios, se utilizará un plugin de WordPress llamado Contact Form to Database, el cual permite almacenar dichos comentarios en un archivo .CSV.

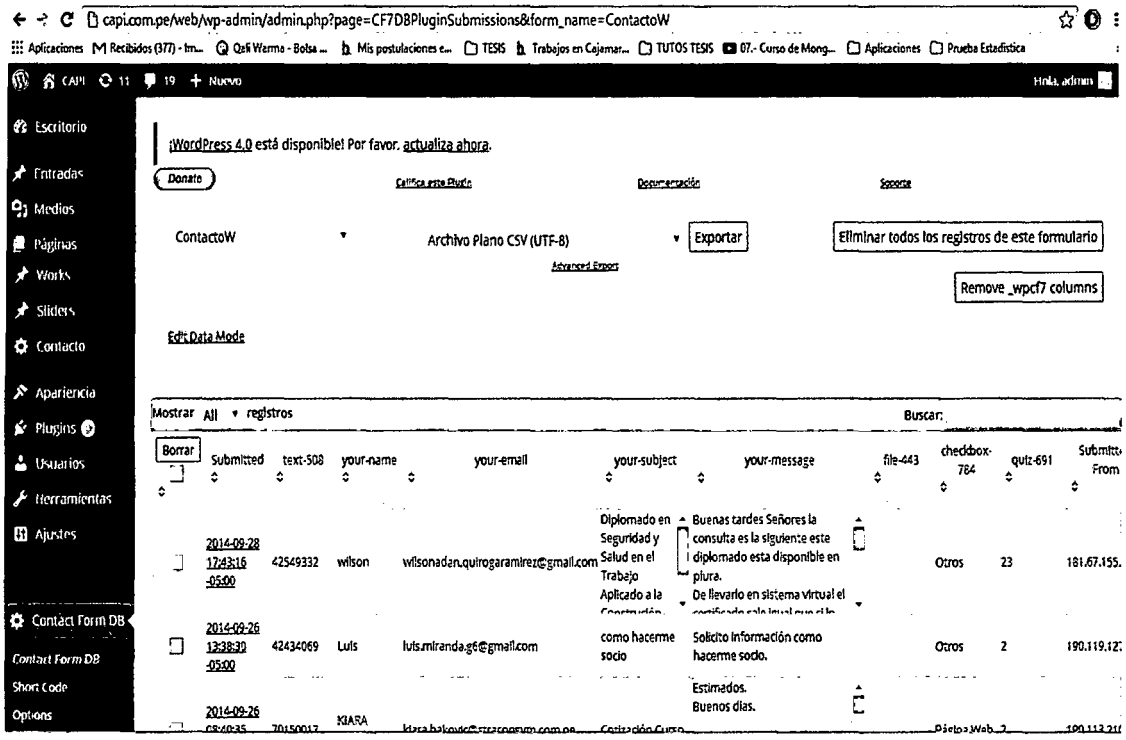


Figura 18. Plugin Contac FormBD WordPress
(Por el autor)

Los datos extraídos de ambas aplicaciones web en los formatos .CSV serán almacenados en la base de datos NoSQL MongoDB en la siguiente sub-etapa.

3.4.4.2. Carga

Una vez obtenido el archivo integrado con los datos de ambas aplicaciones procederemos a la carga de los mismos hacia la base de datos. Como mencionamos en la metodología la carga se puede efectuar de forma manual, pero dado el caso que son muchos registros nos conviene hacerlo de forma masiva.

Para éste proceso se procederá a cargar el archivo .CSV con los comandos propios de MongoDB para la importación de datos a través de su shell de administración.

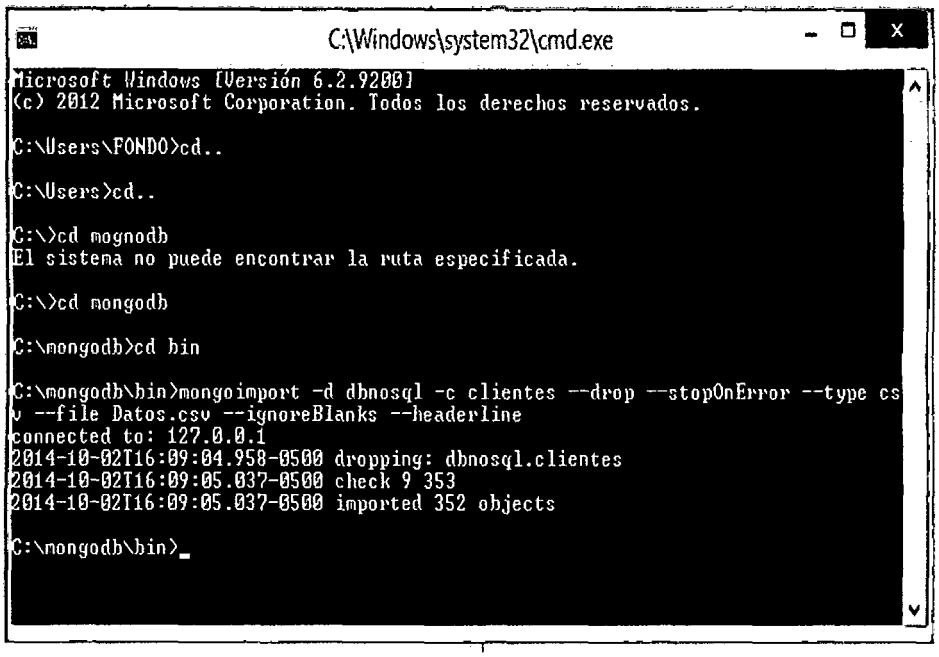


Figura 19. Comando para Carga de datos de .CSV
(Por el autor)

Dónde:

Tabla 19. Descripción de Comando para carga desde .CSV

Comando	Interpretación
-d	Es la base de datos
-c	Es la colección a insertar los datos
--drop	Es el comando para eliminar la colección cada vez que se haga una importación
--stopError	Es el comando para el manejo de errores
--type	Es el tipo de archivo de origen
--file	Es la ubicación de archivo
--ignoreBlanks	Es el comando para que ignore los atributos que no existen
--headerline	Para que mapee las cabeceras de los atributos que existen en el archivo

Es importante mencionar que nuestro archivo .CSV debe estar en formato UTF8

Para comprobar que los datos ya están almacenados en la base de datos basta con hacer lo siguiente:

```
> db.clientes.find();
```

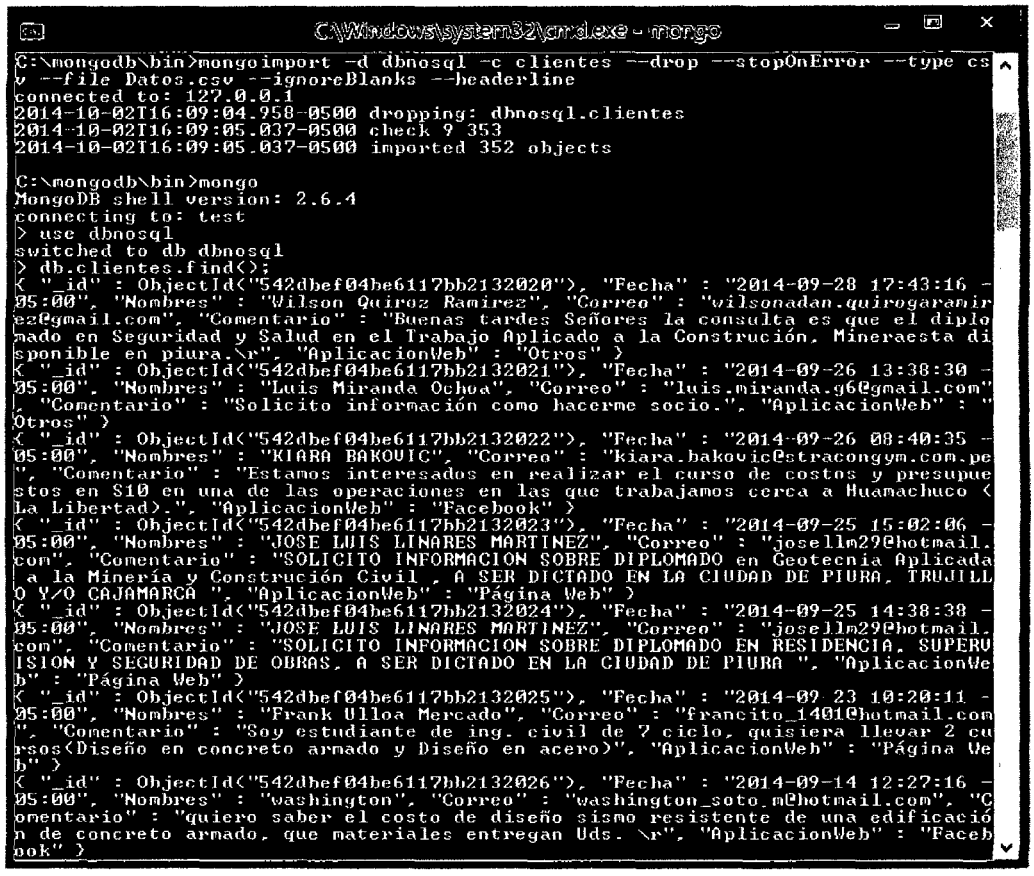


Figura 20. Búsqueda de datos cargados en MongoDB
(Por el autor)

3.4.4.3. Generalismo

En ésta sub etapa vamos a hacer el reconocimiento de patrones de palabras clave en base a los servicios que la empresa ofrece (diplomados y/o cursos de software). Para llevar a cabo este proceso construiremos una aplicación web la cual tendrá conexión con la base de datos NoSQL MongoDB.

Las herramientas necesarias a utilizar son:

- ✓ Base de datos NoSQL: MongoDB Versión 2.6.4.
- ✓ Lenguaje de Programación: PHP Versión 5.5.3.
- ✓ Driver de Conexión: php_mongo-1.5.1-5.5-vc11.dll

- ✓ XAMPP: Versión 1.8.3.
- ✓ Sublime Text 3
- ✓ MapReduce

En importante indicar que las versiones mencionadas con totalmente compatibles, si se usan otras versiones se tendrán que verificar su compatibilidad. Por ejemplo la última versión del driver de MongoDB 1.5.7 para PHP Version 5.5.15 aún no es compatible.

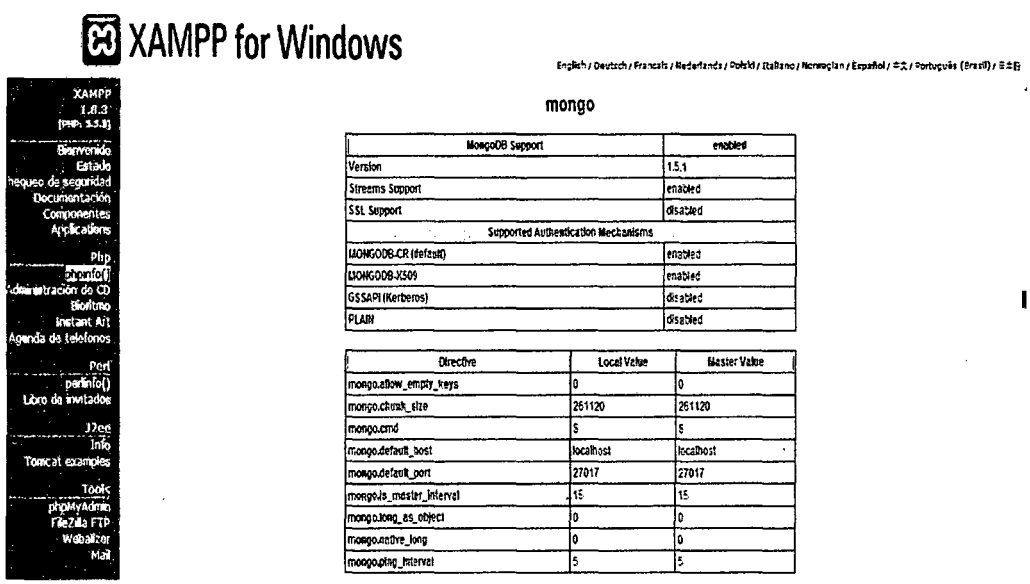
❖ **Conexión de PHP con MongoDB**

Para lograr una conexión exitosa es necesario contar con el driver de conexión de PHP para MongoDB. Este driver es un archivo .dll el cual se puede descargar de la página oficial de MongoDB. Una vez obtenido el driver se lo copia en el archivo php.ini que se encuentra dentro de XAMPP.

```
extension=php_mongo-1.5.1-5.5-vc11.dll
```

Figura 21. Driver de MongoDB para PHP
(Por el autor)

Para comprobar que éste proceso está correcto en la información de PHP de aparecer Mongo como se muestra en la figura siguiente:



MongoDB Support	enabled
Version	1.5.1
Streams Support	enabled
SSL Support	disabled
Supported Authentication Mechanisms	
MONGODB-CR (default)	enabled
MONGODB-X509	enabled
GSSAPI (Kerberos)	disabled
PLAIN	disabled

Directive	Local Value	Master Value
mongo.allow_empty_keys	0	0
mongo.chunk_size	261120	261120
mongo.cmd	$	$
mongo.default_host	localhost	localhost
mongo.default_port	27017	27017
mongo.is_master_interval	15	15
mongo.long_as_object	0	0
mongo.native_long	0	0
mongo.ping_interval	5	5

Figura 22. Configuración de MongoDB en XAMPP
(Por el autor)

Para comprobar la conexión a través de un navegador web utilizando php hacemos lo siguiente :

conexion.php

```php
<?php

try {
    $mongo = new Mongo();
    $databases = $mongo->listDBs();
    echo '<pre>';

    print_r($databases);
    $mongo->close();

} catch (MongoConnectionsException $e) {

    die($e->getMessage());
}
```

Figura 23. Código PHP de conexión a MongoDB
(Por el autor)

El código php anterior sirve para listar las bases de datos existentes dentro de MongoDB, como muestra la siguiente imagen:

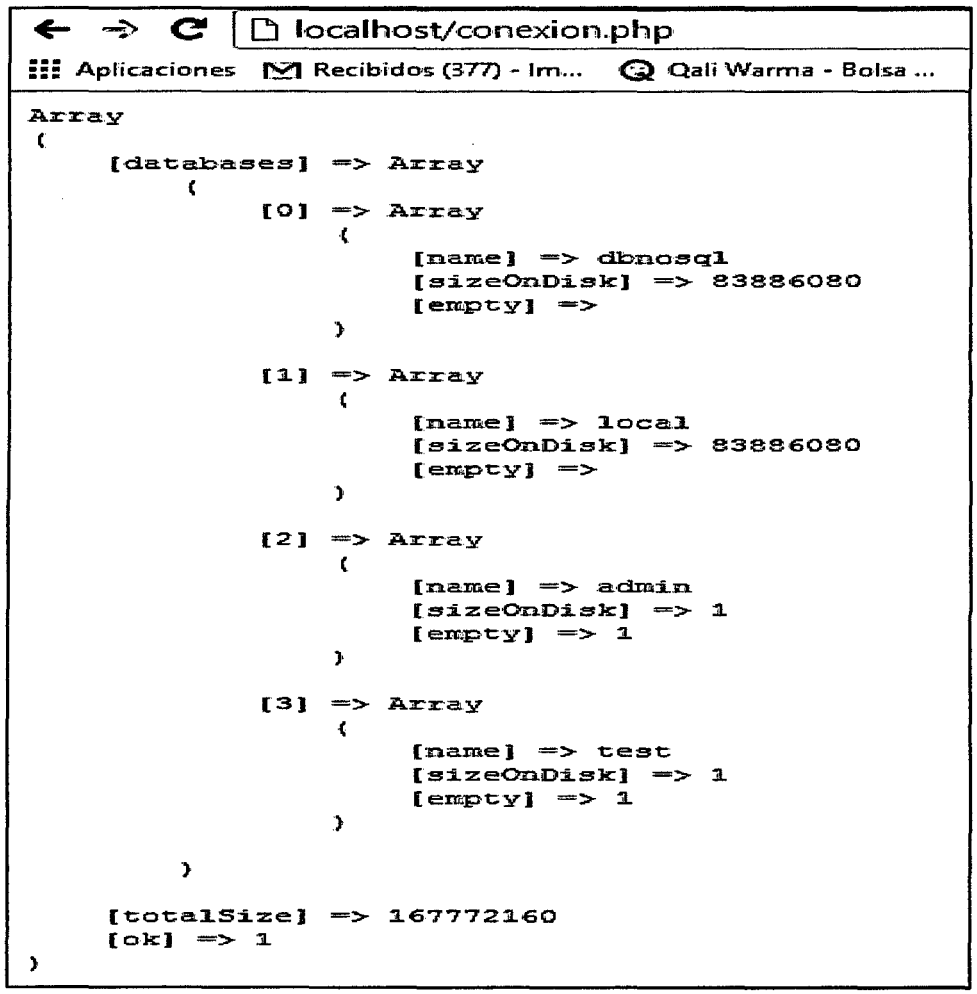


Figura 24. Listado de Bases de Datos existentes en MongoDB
(Por el autor)

En la lista podemos observar que la conexión es exitosa, ya que en ella aparece nuestra base de datos llamada ***dbnosql*** que creamos anteriormente en la etapa de diseño.

- **Desarrollo de la Aplicación**

  Aplicaremos Map-Reduce de acuerdo al diseño de nuestra base de datos. En el mundo no relacional Map-Reduce es *importante* para consultas completas, en consecuencia es necesario para el desarrollo de nuestra aplicación.

```
$map = new MongoCode("function() {".
            "for (i = 0; i < this.Comentario.length; i++) {".
                "emit(this.Comentario[i], 1);".
            "}".
        "}");

$reduce = new MongoCode("function(key, values) {".
                "var count = 0;".
                "for (var i = 0; i < values.length; i++){".
                    "count += values[i];".
                "}".
                "return count;".
            "}");

$command = array(
                    'mapreduce' => 'clientes',
                    'map' => $map,
                    'reduce' => $reduce,
                    'out' => 'comentariocount'
                );
```

Figura 25. Aplicación de Map-Reduce en PHP
(Por el autor)

La figura anterior nos muestra la aplicación de Map-Reduce desde PHP, pero también se puede aplicar directamente en el Shell de MongoDB. Sin embargo para efectos de nuestra aplicación web es necesario aplicarlo desde PHP.

La siguiente figura muestra el Panel Principal de Administración de la Aplicación:

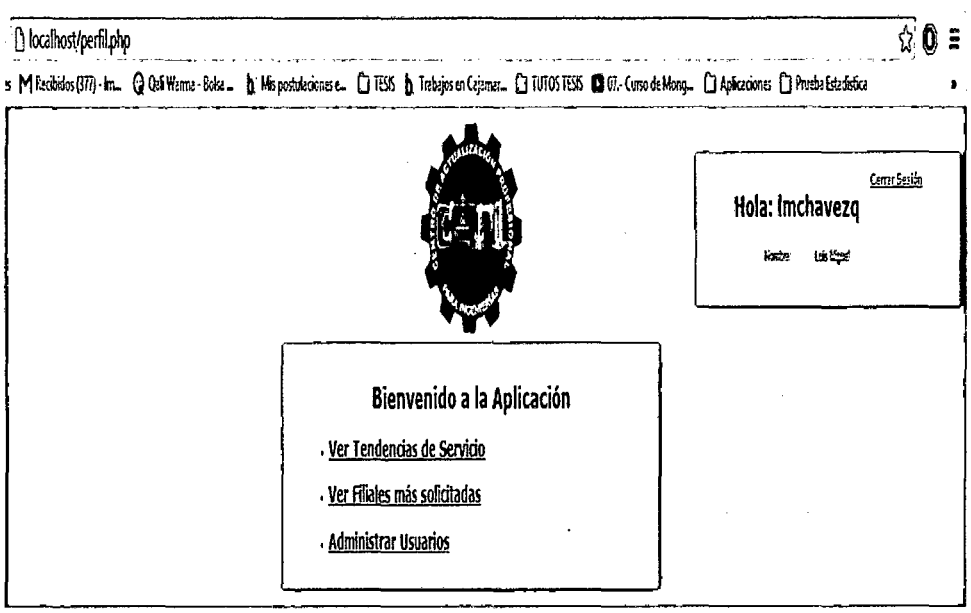


Figura 26. Formulario Web: Panel Principal de Administración
(Por el autor)

La primera opción "*Ver Tendencias de Servicio*", nos muestra las palabras claves más frecuentes que los usuarios externos suelen digitar en las aplicaciones web (Página web y Facebook). La información que se muestra proviene de los datos cargados en la base de datos NoSQL creada anteriormente. Para obtener ésta información se utilizó un trozo de código basado en el modelo de programación Map-Reduce que se muestra en la Figura 25.

La aplicación muestra con mayor énfasis a aquellos servicios que tienen mayor tendencia por los potenciales clientes y con menor énfasis aquellos servicios que tiene poca tendencia.

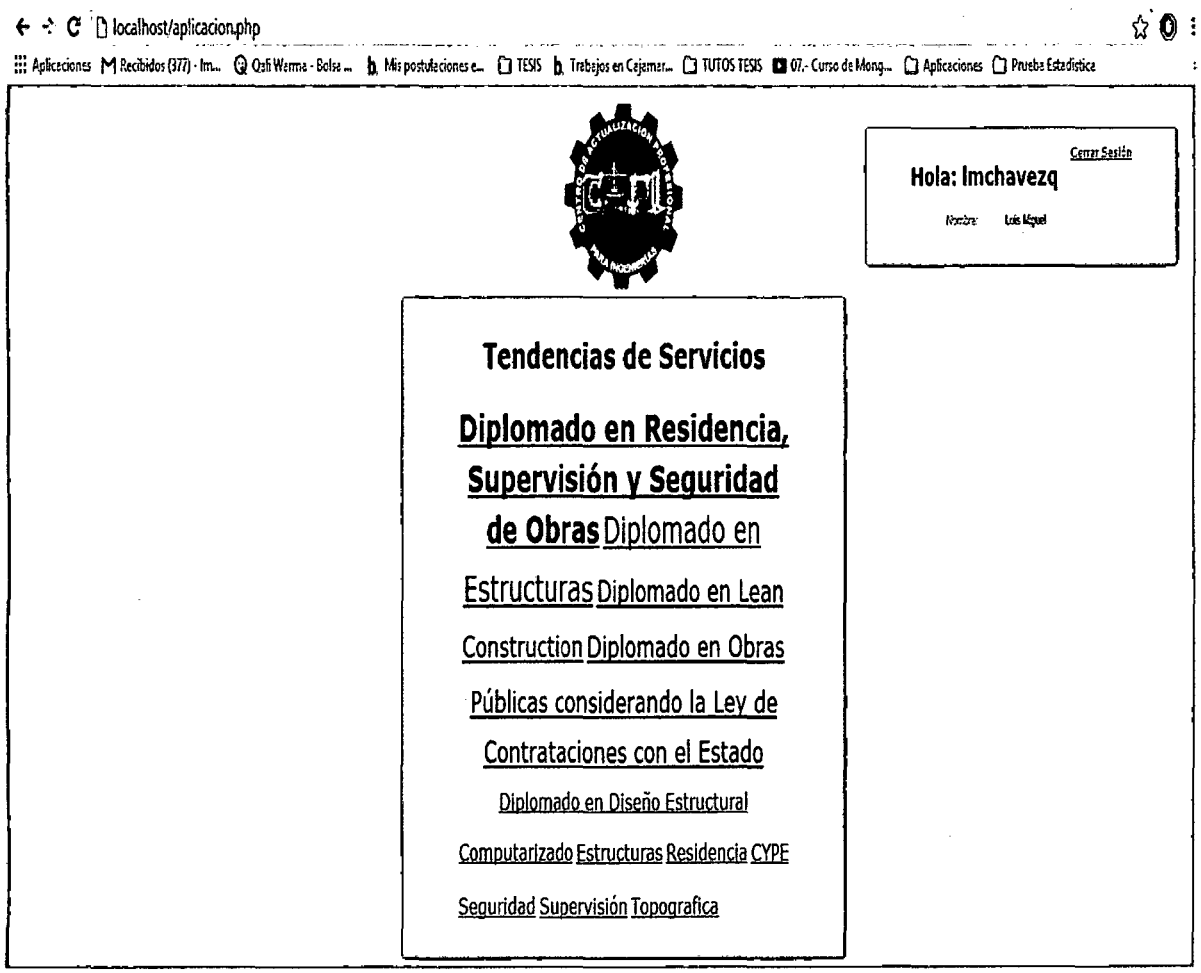


Figura 27. Formulario Web: Tendencias de Servicios
(Por el autor)

La segunda opción "Ver Filiales más Solicitadas", nos muestra las Filiales en que los potenciales clientes desean se les brinde el servicio. La filial en blanco aparece debido que no siempre un cliente menciona la filial en donde quiere el servicio.

El campo "*Filial*" de la colección "*clientes*" dentro de la base de datos no aparece en aquellas colecciones que no contengan dicho campo; esto es una de las características resaltantes de las bases de datos NoSQL, no relaciones o *no estructuradas*, ya que *como* su mismo nombre los dice no es necesario tener la misma estructura en cada documento de una colección. En consecuencia a esto el sistema reconoce dicho campo y lo muestra como en blanco.

Para obtener ésta información se utilizó el comando *Group* de MongoDB para PHP

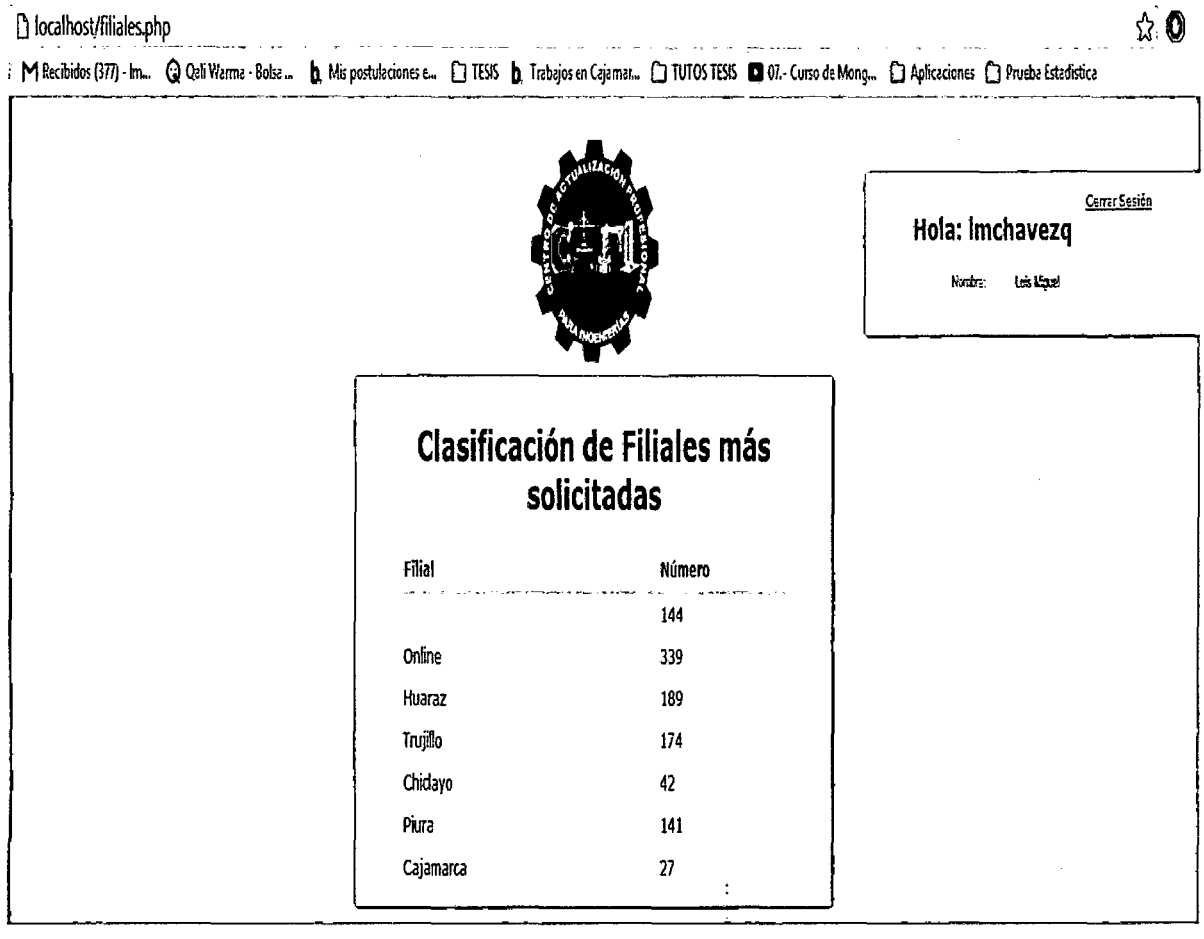


Figura 28. Formulario Web: Clasificación de Filiales más solicitadas
(Por el autor)

La tercera opción "Administrar Usuarios", nos permite gestionar los usuarios para el manejo de sesiones y loggeo de la aplicación. Ésta información proviene de la colección "usuario" de la base de datos NoSQL creada anteriormente.

Para administrar los usuarios, se creó un CRUD, teniendo en cuenta comandos de MongoDB para PHP.

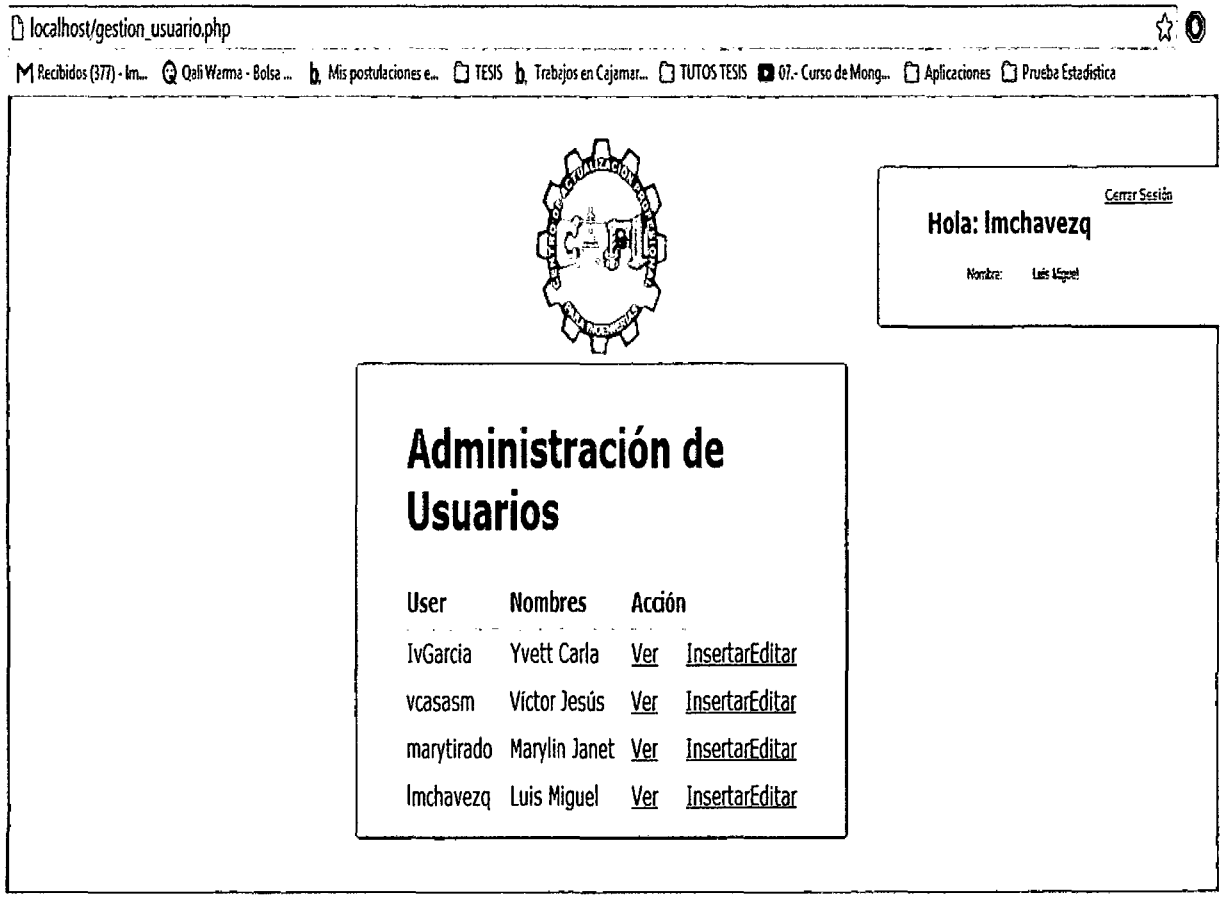


Figura 29. Formulario Web: Administración de Usuarios
(Por el autor)

3.4.4.4. Análisis

En esta última sub etapa, analizamos los datos obtenidos de la aplicación, concluyendo que en cuanto a los servicios con más tendencia son los diplomados en Residencia, Supervisión y Seguridad en Obras seguido del diplomado en estructuras. También se pueden apreciar otros diplomas que tienen menor tendencia. Por otro lado vemos que las filiales más solicitadas son

Huaraz y Trujillo, pero también hay una gran demanda de manera Online.

b) Análisis, tratamiento de datos y presentación de resultados

La aplicación desarrollada con la base de datos NoSQL fue presentado a los colaboradores del Área de Ventas y Marketing de las filiales del CAPI, siendo un total de 12 personas. Posterior a su uso, se aplicó un cuestionario tipo Likert (ver Anexo 1) para medir el nivel de satisfacción del usuario y un segundo cuestionario para medir el nivel de automatización de acceso a la información (ver Anexo 02). Además se usó una ficha de información (ver Anexo 03) para medir el tiempo de respuesta en el acceso a la información para la toma de decisiones. De la misma manera, se aplicaron los cuestionario (ver Anexo 1 y Anexo 02) y la ficha de observación (ver Anexo 03) antes de implementar la aplicación con la base de datos NoSQL, de modo que nos permita medir el impacto del uso de la base de datos en la toma de decisiones en comparación con el proceso actual.

Los cuestionarios aplicados, cuenta con 07 preguntas tipo Likert cada uno. Para el procesamiento de los datos recogidos de cada cuestionario, para el caso de los indicadores *Nivel de Satisfacción* del usuario (Ver Tabla 20) y *Nivel de Automatización del acceso la información* (Ver Tabla 21) se usaron el número de **frecuencias** de cada respuesta por cada encuestado, luego se obtuvo el total por respuesta (Muy Malo, Malo, Regular, Bueno, Muy Bueno).
A continuación se muestran los resultados obtenidos antes (Pre Test) y después (Post Test)

3.4.5. Pre Test

- ✓ **Indicador: Nivel de Satisfacción del Usuario**

  Éste indicador se refiere al usuario interno que interactúa con la aplicación.

  De acuerdo a los datos obtenidos, se obtienen los siguientes resultados:

Tabla 20. Resumen Nivel de Satisfacción-Proceso Actual

	PROCESO ACTUAL	
NIVEL DE SATISFACCIÓN	CANTIDAD DE FRECUENCIA (fa)	PORCENTAJE (%)
Muy Bueno (5)	0	0%
Bueno (4)	0	0%
Regular (3)	5	6%
Malo (2)	27	32%
Muy Malo (1)	52	62%
TOTAL	**84**	**100%**

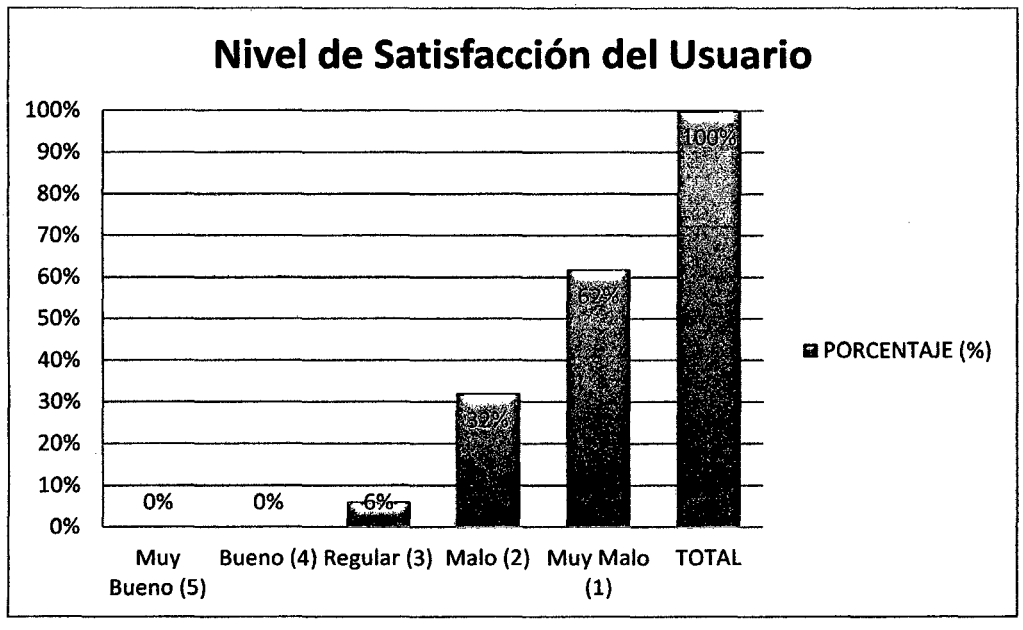


Gráfico 01. Nivel de Satisfacción- Proceso Actual

Con los resultados obtenidos en el grafico anterior (Ver Gráfico 01), podemos apreciar que para el proceso actual, el nivel de **satisfacción del usuario** en el acceso a la información en el proceso de segmentación de clientes es considerado Muy Malo en un 62%, Malo en un 32% y apenas Regular con un 6%.

A continuación se presentan las preguntas incluido en el cuestionario de ***Nivel de Satisfacción del usuario con respecto al acceso a la información en el proceso de segmentación de clientes***. Para poder observar el cuestionario completo Ver Anexo 01.

Tabla 21. Preguntas Cuestionario 01- Nivel de Satisfacción del Usuario

Ítem	Pregunta
1.	¿Cómo califica el procedimiento de obtención de información utilizada en el proceso de segmentación de clientes?
2.	¿Cómo considera la organización y almacenamiento de la información utilizada en el proceso de segmentación de clientes?
3.	¿Cuál es su opinión sobre los medios utilizados para acceder a la información necesaria para el proceso de segmentación?
4.	¿Cómo califica el aporte de la información utilizada en el proceso de segmentación de clientes?
5.	¿Cómo califica el aporte de la información resultado de la segmentación de clientes en la propuesta de servicios de capacitación?
6.	¿Cómo considera el actual proceso de segmentación de clientes?
7.	¿Cómo califica la información disponible y relacionada con los requerimientos y preferencias de capacitación demandados?

- ✓ **Indicador: Nivel de Automatización de Acceso a la Información**

  De acuerdo a los datos obtenidos, se obtienen los siguientes resultados:

Tabla 22. Resumen Nivel de Automatización-Proceso Actual

	PROCESO ACTUAL	
NIVEL DE AUTOMATIZACION	CANTIDAD DE FRECUENCIA (fa)	PORCENTAJE (%)
Muy Bueno (5)	0	0%
Bueno (4)	0	0%
Regular (3)	7	8%
Malo (2)	16	19%
Muy Malo (1)	61	73%
Total	**84**	**100%**

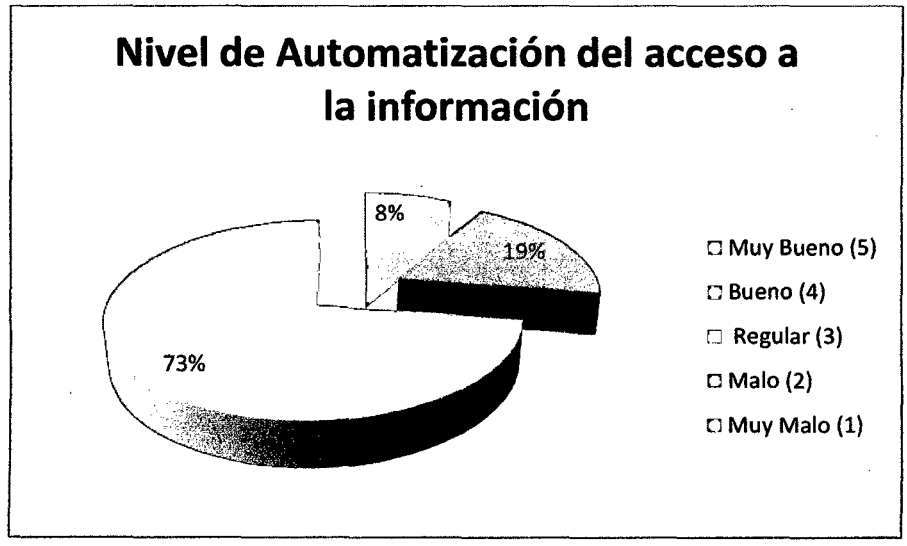


Gráfico 02. Nivel de Automatización de acceso a la información-Proceso Actual

En cuanto al nivel de **automatización del acceso a la información**, en el gráfico anterior (Ver Gráfico 02) podemos apreciar que actualmente es considerada Muy Malo en una 73%, Malo en un 19% y apenas Regular con un 8%.

A continuación se presentan las preguntas incluido en el cuestionario de Nivel de ***Automatización del acceso a la información en el proceso de segmentación de clientes***. Para poder observar el cuestionario completo Ver Anexo 02.

Tabla 23. Preguntas Cuestionario 02- Nivel de Automatización de acceso a la información

Ítem	Pregunta
1.	¿Cómo califica el procedimiento para tener acceso a la información necesaria para el proceso de segmentación de clientes?
2.	¿Cómo califica los medios de almacenamiento de información utilizada en el proceso de segmentación de clientes?
3.	¿Cuál es su opinión sobre el nivel de seguridad en el almacenamiento y acceso a la información a utilizar en el proceso de segmentación de clientes?
4.	¿Cómo califica la herramienta de almacenamiento y reporte de información necesaria en el proceso de segmentación de clientes?
5.	¿Cómo califica la herramienta que le permita registrar y conocer las preferencias de capacitación?
6.	¿Cómo considera al rango de tiempo que le lleva a reunir información útil, para el proceso de segmentación de clientes?
7.	¿Cómo califica el tiempo que le toma realizar el proceso de segmentación de clientes?

- ✓ **Indicador: Tiempo de acceso a la información**

  Para éste indicador usamos una guía de observación para medir los tiempos que toman los siguientes 3 procesos, obteniendo los siguientes resultados antes de la aplicación de la base de datos NoSQL

Tabla 24. Resumen Tiempo de Acceso a la información-Proceso Actual

PROCESO	ANTES (minutos)
P1: Extracción de información de filiales más solicitadas	**10.34**
P2:Consulta de diplomados y/o cursos con más tendencia	**14.47**
P3:Extracción de información para campañas de marketing	**7.18**

Cabe resaltar que los minutos obtenidos en la guía de observación (Ver Tabla 22), fueron obtenidos en una sola toma, debido a que los procesos considerados no se dan de forma frecuente y repetitiva, si no cada cierto tiempo considerable. Por ésta razón fue complicado aplicar varias tomas para un mismo proceso.

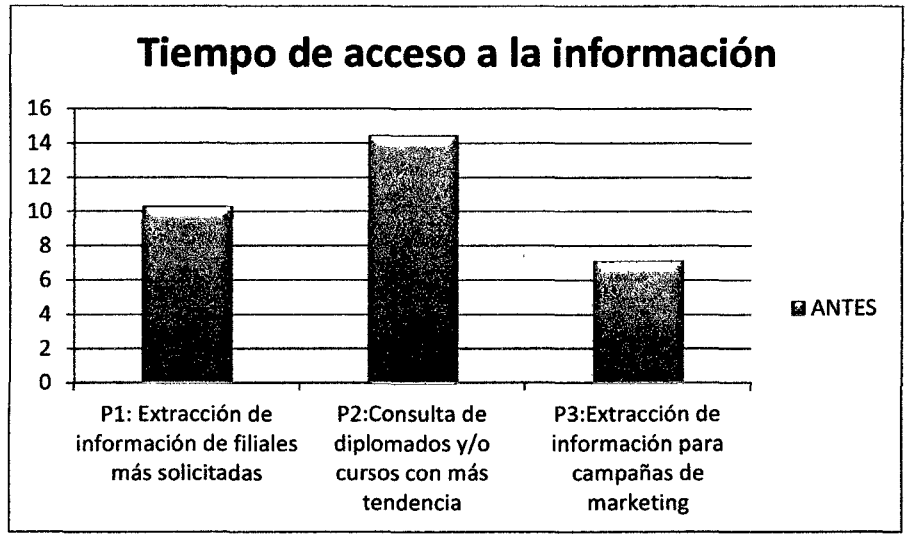


Gráfico 03. Tiempo de Acceso a la información-Proceso Actual

En la gráfica anterior (Ver Gráfico 03), podemos apreciar que los procesos considerados para el indicador ***tiempo de acceso a la información*** toman de entre 07 a 15 minutos aproximadamente.

3.4.6. Post Test

- ✓ **Indicador: Nivel de Satisfacción del usuario**

  De acuerdo a los datos obtenidos, se obtienen los siguientes resultados:

Tabla 25. Resumen Nivel de Satisfacción-Base de Datos NoSQL

	BASE DE DATOS NOSQL	
NIVEL DE SATISFACCIÓN	CANTIDAD DE FRECUENCIA (fa)	PORCENTAJE (%)
Muy Bueno (5)	48	57%
Bueno (4)	28	33%
Regular (3)	8	10%
Malo (2)	0	0%
Muy Malo (1)	0	0%
TOTAL	**84**	**100%**

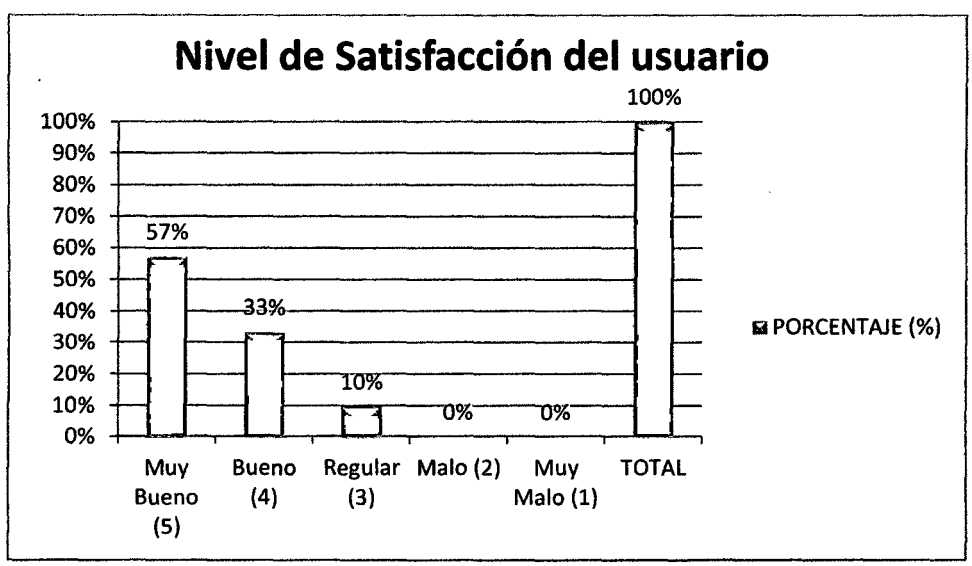


Gráfico 04. Nivel de Satisfacción del Usuario- Base de Datos NoSQL

Después de aplicar la base de datos NoSQL, podemos apreciar (Ver Gráfico 04), que los resultados obtenidos han mejorado en cuanto al nivel de **Satisfacción del usuario**: el 57% considera Muy bueno, el 33% considera Bueno y el 10% considera Regular.

Cabe mencionar que las preguntas aplicadas fueron las mismas de la Tabla 21 mostrada en la parte del Pre Test.

✓ **Indicador: Nivel de Satisfacción del Acceso a la información**

Tabla 26. Resumen Nivel de Automatización-Base de Datos NoSQL

	BASE DE DATOS NOSQL	
NIVEL DE AUTOMATIZACION	CANTIDAD DE FRECUENCIA (fa)	PORCENTAJE (%)
Muy Bueno (5)	52	62%
Bueno (4)	28	33%
Regular (3)	4	5%
Malo (2)	0	0%
Muy Malo (1)	0	0%
Total	**84**	**100%**

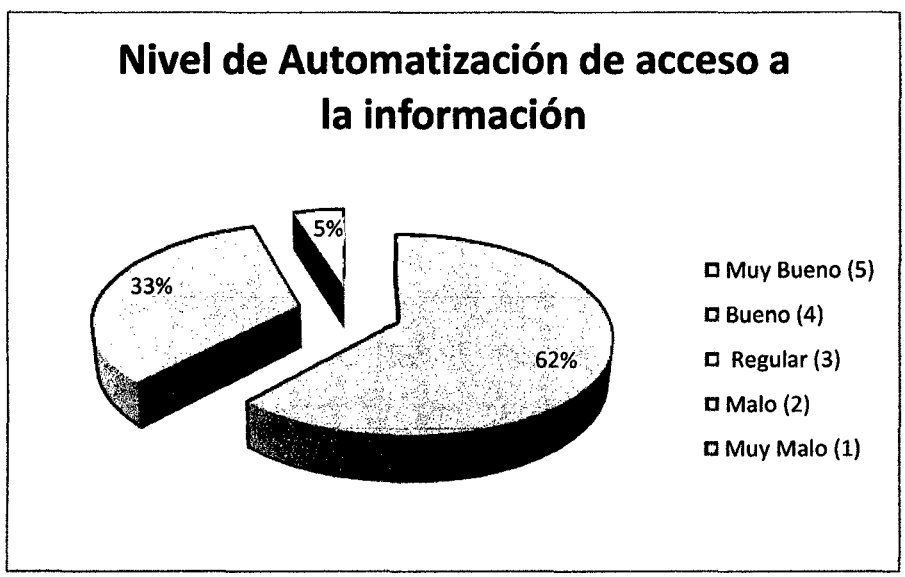


Gráfico 05. Nivel de Automatización de acceso a la información-Base de Datos NoSQL

En cuanto al nivel de **automatización del acceso a la información** después de implementarse la base de datos NoSQL (Ver Gráfica 05), obtenemos los siguientes resultados: Muy Bueno con 62%, Bueno con 33% y Regular con 5%.

✓ **Indicador: Tiempo de acceso a la información**

Tabla 27. Resumen Tiempo de Acceso a la Información- Base de Datos NoSQL

PROCESO	DESPUÉS
P1: Extracción de información de filiales más solicitadas	1.12
P2:Consulta de diplomados y/o cursos con más tendencia	0.45
P3:Extracción de información para campañas de marketing	1.44

Cabe resaltar que los minutos obtenidos en la guía de observación (Ver Tabla 22), fueron obtenidos en una sola toma, debido a que los procesos considerados no se dan de forma frecuente y repetitiva, si no cada cierto tiempo considerable. Por ésta razón fue complicado aplicar varias tomas para un mismo proceso.

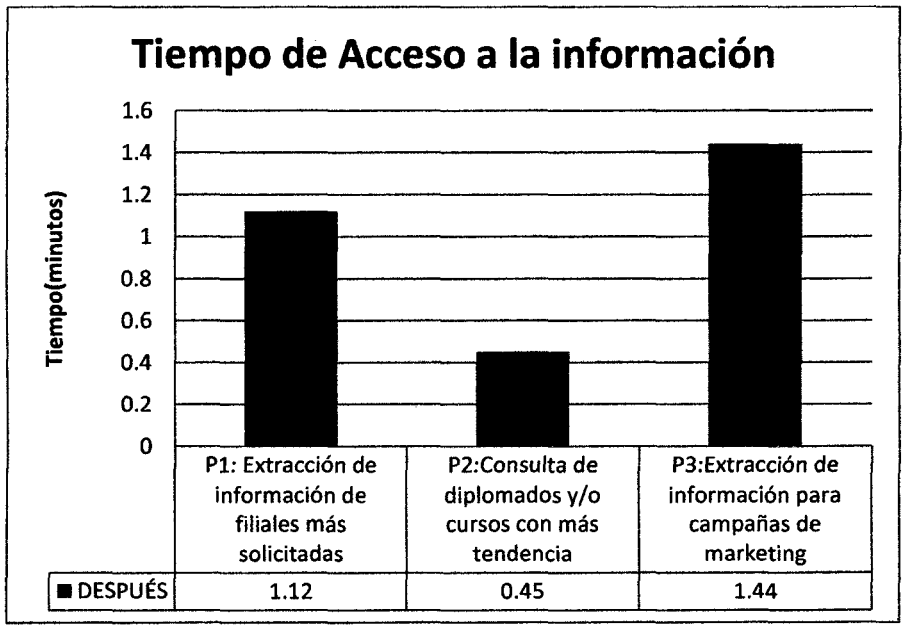


Gráfico 06. Tiempo de Acceso a la Información-Base de Datos NoSQL

En el gráfico anterior (Ver Gráfico 06) se puede observar que hay una notable diferencia después de la aplicación de la base de datos NoSQL, en cuanto a tiempos de acceso a la información de los procesos considerados, éstos ahora no superan más de 2 minutos.

CAPITULO IV. ANÁLISIS Y DISCUSIÓN DE RESULTADOS

Con los resultados obtenidos de las encuestas y de las fichas de observación, mostradas en el capítulo anterior, se realizarán las pruebas de hipótesis correspondientes con el fin de analizar los indicadores de **nivel de satisfacción de los usuarios, nivel de automatización en el acceso a la información** y el **tiempo de respuesta en el acceso a la información para la toma de decisiones**, las cuales medirán el efecto de la variable dependiente (**Acceso al información en el proceso de segmentación de clientes**) tras la manipulación de la variable independiente (**Base de datos NoSQL**).

4.1. Análisis de resultados

4.1.1. Prueba de hipótesis para el primer indicador: Nivel de Satisfacción del usuario

a. Formulación de las hipótesis

Hipótesis Nula:

H_0: La aplicación de una base de datos NoSQL no aumenta el nivel de satisfacción de los usuarios con respecto al acceso de información en el proceso de segmentación de clientes.

Hipótesis Alterna:

H_a: La aplicación de una base de datos NoSQL aumenta el nivel de satisfacción de los usuarios con respecto al acceso de información en el proceso de segmentación de clientes.

b. Elección del nivel de significancia o confianza

El nivel de significancia será del 5%, $\alpha = 0.05$.

c. ***Elección del estadístico de prueba***

Por tener una muestra igual a la cantidad de la población que son 12 encuestados y al ser esta una muestra $n < 30$, se aplicará la prueba estadística *t*-student para muestras emparejadas, utilizada para medir muestras medidas en más de un tiempo. Para este caso mediremos el nivel de satisfacción de los usuarios con respecto al acceso a la información en el proceso de segmentación de clientes con el uso del proceso actual (Pre-test) y el nivel de satisfacción de los usuarios con respecto al acceso a la información en el proceso de segmentación de clientes con el uso de una base de datos NoSQL (Post-test).

Tabla 28. Análisis-Resultado Nivel de Satisfacción

NIVEL DE SATISFACCIÓN	ANTES	DESPUÉS	DIFERENCIA
Entrevistado 1	11	31	**-20**
Entrevistado 2	8	31	**-23**
Entrevistado 3	7	29	**-22**
Entrevistado 4	11	33	**-22**
Entrevistado 5	10	34	**-24**
Entrevistado 6	9	28	**-19**
Entrevistado 7	11	31	**-20**
Entrevistado 8	8	33	**-25**
Entrevistado 9	18	32	**-14**
Entrevistado 10	9	31	**-22**
Entrevistado 11	8	31	**-23**
Entrevistado 12	11	32	**-21**

Luego se aplica un análisis de estadística descriptiva de la columna **DIFERENCIA** para hallar la ***Media*** y la ***Desviación Estándar***

Tabla 29. Análisis Estadístico Descriptivo- Nivel de Satisfacción del Usuario

	DIFERENCIA
Media	-21.25
Error típico	0.826868866
Mediana	-22
Moda	-22
Desviación estándar	2.864357773

Varianza de la muestra	8.204545455
Coeficiente de asimetría	1.443087188
Rango	11
Mínimo	-25
Máximo	-14
Suma	-255
Cuenta	12
Nivel de confianza (95.0%)	1.819926103

Luego, para obtener el estadístico de prueba, se aplicará la siguiente fórmula.

$$t = \frac{X_d - \mu_d}{\frac{s_d}{\sqrt{n}}} = -25.6993$$

Cálculo estadístico de la prueba t para dos muestra emparejadas:

Tabla 30. Prueba t para dos muestras -Nivel Satisfacción de Usuario

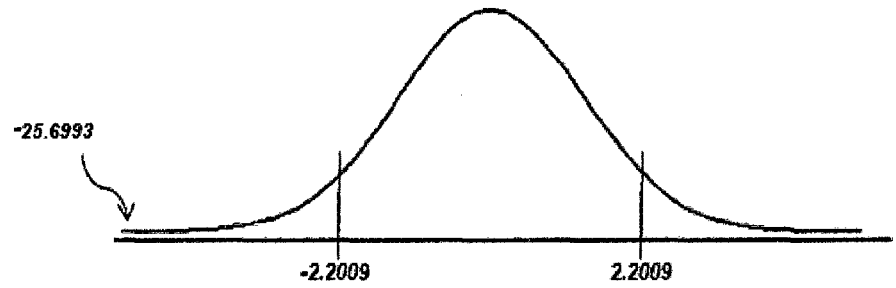
	Antes	*Después*
Media	10.08333333	31.3333333
Varianza	8.265151515	2.78787879
Observaciones	12	12
Coeficiente de correlación de Pearson	0.296702745	
Diferencia hipotética de las medias	0	
Grados de libertad	11	
Estadístico t	-25.69935921	
P(T<=t) una cola	1.78833E-11	
Valor crítico de t (una cola)	1.795884819	
P(T<=t) dos colas	3.57666E-11	
Valor crítico de t (dos colas)	2.20098516	

d. Cálculo del valor crítico de la estadística de prueba

De acuerdo a los datos obtenidos, aplicando la prueba *t* para dos medidas de muestras emparejadas y usando $\alpha = 0.05$, tenemos como valor crítico:

$$-t_{3.5766E^{-11}} \leq -2.2009 \text{ y } t_{3.5766E^{-11}} \geq 2.2009$$

e. Definición de la regla de decisión



Hipótesis	H_0: $\mu = \mu_0$ H_a: $\mu \neq \mu_0$
Estadístico de prueba	$t = \frac{X_d - \mu_d}{\frac{s_d}{\sqrt{n}}}$
Regla de rechazo: **Método *p* - value**	Rechazar H_0 si *p* - value ≤ α
Regla de rechazo: **Método del valor crítico**	Rechazar H_0 si $t \leq$ - 2.2009 o Rechazar H_0 si $t \geq$ 2.2009

f. Toma de decisión de aceptar o rechazar Ho

Como el *p* - value = 3.5766E-11 es menor que el nivel de significancia α = 0.05, y como el valor del estadístico de prueba *t* = -25.69935921 es menor que el valor crítico = - 2.2009. Se tiene evidencia suficiente para rechazar la hipótesis nula.

Por lo tanto, se acepta la hipótesis alterna H_a: **La aplicación de la base de datos NoSQL aumenta el nivel de satisfacción de los usuarios respecto al acceso a la información en el proceso de segmentación de clientes.**

4.1.2. Prueba de hipótesis para el segundo indicador: Nivel de Automatización del acceso a la información

a. Formulación de las hipótesis

Hipótesis Nula:

H_0: La aplicación de una base de datos NoSQL no aumenta el nivel de automatización en el acceso a la información en el proceso de segmentación de clientes.

Hipótesis Alterna:

H_a: La aplicación de una base de datos NoSQL aumenta el nivel de automatización en el acceso a la información en el proceso de segmentación de clientes.

b. Elección del nivel de significancia o confianza

El nivel de significancia será del 5%, $\alpha = 0.05$.

c. Elección del estadístico de prueba

Por tener una muestra igual a la cantidad de la población que son 12 encuestados y al ser esta una muestra $n < 30$, se aplicará la prueba estadística *t*-student para muestras emparejadas, utilizada para medir muestras medidas en más de un tiempo. Para este caso mediremos el nivel de automatización en el acceso a la información en el proceso de segmentación de clientes con el uso del proceso actual (Pre-test) y el nivel de automatización en el acceso a la información en el proceso de segmentación de clientes con el uso de una base de datos NoSQL (Post-test).

Tabla 31. Análisis-Resultado Nivel de Automatización

NIVEL DE AUTOMATIZACIÓN	ANTES	DESPUES	DIFERENCIA
Entrevistado 1	8	33	**-25**
Entrevistado 2	8	32	**-24**
Entrevistado 3	18	32	**-14**
Entrevistado 4	8	33	**-25**

Entrevistado 5	8	32	**-24**
Entrevistado 6	9	31	**-22**
Entrevistado 7	12	34	**-22**
Entrevistado 8	8	30	**-22**
Entrevistado 9	8	31	**-23**
Entrevistado 10	12	32	**-20**
Entrevistado 11	9	30	**-21**
Entrevistado 12	7	34	**-27**

Luego se aplica un análisis de estadística descriptiva de la columna **DIFERENCIA** para hallar la ***Media*** y la ***Desviación Estándar***

Tabla 32. Análisis Estadístico Descriptivo- Nivel de Automatización del Acceso al información

DIFERENCIA	
Media	-22.41666667
Error típico	0.949149007
Mediana	-22.5
Moda	-22
Desviación estándar	3.28794861
Varianza de la muestra	10.81060606
Coeficiente de asimetría	1.456338822
Rango	13
Mínimo	-27
Máximo	-14
Suma	-269
Cuenta	12
Nivel de confianza(95.0%)	2.08906288

Luego, para obtener el estadístico de prueba, se aplicará la siguiente fórmula.

$$t = \frac{X_d - \mu_d}{\frac{s_d}{\sqrt{n}}} = -23.6176$$

Cálculo estadístico de la prueba t para dos muestra emparejadas:

Tabla 33. Prueba t para dos muestras emparejadas. Nivel de Automatización

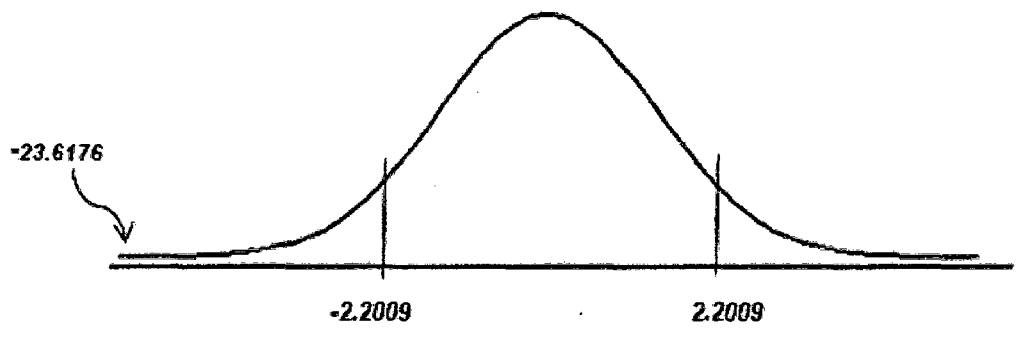
	Antes	*Después*
Media	9.583333333	32
Varianza	9.537878788	1.818181818
Observaciones	12	12
Coeficiente de correlación de Pearson	0.065491361	
Diferencia hipotética de las medias	0	
Grados de libertad	11	
Estadístico t	-23.61764748	
P(T<=t) una cola	4.46017E-11	
Valor crítico de t (una cola)	1.795884819	
P(T<=t) dos colas	8.92033E-11	
Valor crítico de t (dos colas)	2.20098516	

d. Cálculo del valor crítico de la estadística de prueba

De acuerdo a los datos obtenidos, aplicando la prueba *t* para dos medidas de muestras emparejadas y usando $\alpha = 0.05$, tenemos como valor crítico:

$$- t_{8.92033E^{-11}} \leq -2.2009 \text{ y } t_{8.92033^{-11}} \geq 2.2009$$

e. Definición de la regla de decisión



Hipótesis	H_0: $\mu = \mu_0$ H_a: $\mu \neq \mu_0$
Estadístico de prueba	$t = \frac{X_d - \mu_d}{\frac{s_d}{\sqrt{n}}}$
Regla de rechazo: **Método *p* - value**	Rechazar H_0 si *p* - value ≤ α
Regla de rechazo: **Método del valor crítico**	Rechazar H_0 si $t \leq$ - 2.2009 o Rechazar H_0 si $t \geq$ 2.2009

f. Toma de decisión de aceptar o rechazar Ho

Como el *p* - value = 8.92033E-11 es menor que el nivel de significancia α = 0.05, y como el valor del estadístico de prueba *t* = - 23.61764748 es menor que el valor crítico = - 2.2009. Se tiene evidencia suficiente para rechazar la hipótesis nula.

Por lo tanto, se acepta la hipótesis alterna H_a: **La aplicación de la base de datos NoSQL aumenta el nivel de automatización en el acceso a la información en el proceso de segmentación de clientes.**

4.1.3. Prueba de hipótesis para el tercer indicador: Tiempo de acceso a la información

a. Formulación de las hipótesis

Hipótesis Nula:

H_0: La aplicación de una base de datos NoSQL no disminuye el tiempo de acceso a la información en el proceso de segmentación de clientes.

Hipótesis Alterna:

H_a: La aplicación de una base de datos NoSQL disminuye el tiempo de acceso a la información en el proceso de segmentación de clientes.

b. *Elección del nivel de significancia o confianza*

El nivel de significancia será del 5%, $\alpha = 0.05$.

c. *Elección del estadístico de prueba*

Por tener una muestra igual a la cantidad de la población que son 12 encuestados y al ser esta una muestra $n < 30$, se aplicará la prueba estadística *t*-student para muestras emparejadas, utilizada para medir muestras medidas en más de un tiempo. Para este caso mediremos el nivel de automatización en el acceso a la información en el proceso de segmentación de clientes con el uso del proceso actual (Pre-test) y el nivel de automatización en el acceso a la información en el proceso de segmentación de clientes con el uso de una base de datos NoSQL (Post-test).

Tabla 34. Análisis-Resultado Tiempo de acceso a la información

PROCESO	ANTES	DESPUÉS	DIFERENCIA	Tiempo Reducido
P1: Extracción de información de filiales más solicitadas	10.34	1.12	**9.22**	**89%**
P2:Consulta de diplomados y/o cursos con más tendencia	14.47	0.45	**14.02**	**97%**
P3:Extracción de información para campañas de marketing	7.18	1.44	**5.74**	**80%**

Luego se aplica un análisis de estadística descriptiva de la columna **DIFERENCIA** para hallar la ***Media*** y la ***Desviación Estándar***

Tabla 35. Análisis Estadístico Descriptivo- Tiempo de Acceso a la información

DIFERENCIA	
Media	7.766666667
Error típico	1.354564301
Mediana	9.02
Desviación estándar	2.346174191
Varianza de la muestra	5.504533333
Coeficiente de asimetría	-1.717901876
Rango	4.16
Suma	23.3
Cuenta	3

Nivel de confianza(95.0%)	5.828219786

Luego, para obtener el estadístico de prueba, se aplicará la siguiente fórmula.

$$t = \frac{X_d - \mu_d}{\frac{s_d}{\sqrt{n}}} = 5.7337$$

Cálculo estadístico de la prueba t para dos muestra emparejadas:

Tabla 36. Prueba t para dos muestras emparejadas- Tiempo de acceso a la información

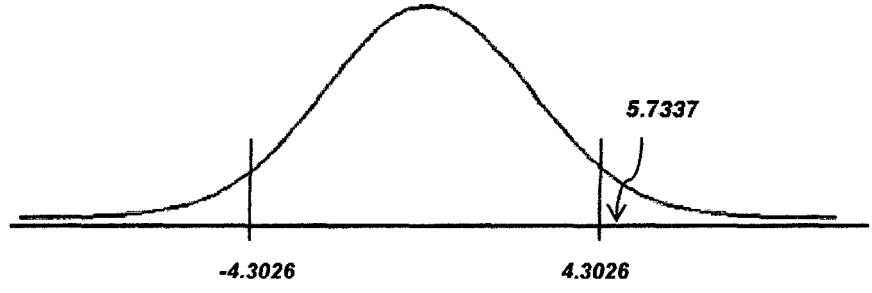
	ANTES	*DESPUES*
Media	8.77	1.003333333
Varianza	4.0539	0.255233333
Observaciones	3	3
Coeficiente de correlación de Pearson	-0.587594935	
Diferencia hipotética de las medias	0	
Grados de libertad	2	
Estadístico t	5.733700987	
P(T<=t) una cola	0.01454844	
Valor crítico de t (una cola)	2.91998558	
P(T<=t) dos colas	0.02909688	
Valor crítico de t (dos colas)	4.30265273	

d. Cálculo del valor crítico de la estadística de prueba

De acuerdo a los datos obtenidos, aplicando la prueba *t* para dos medidas de muestras emparejadas y usando $\alpha = 0.05$, tenemos como valor crítico:

$$-t_{0.02909} \leq -4.3026 \text{ y } t_{0.02909} \geq 4.3026$$

e. Definición de la regla de decisión



Hipótesis	H_0: $\mu \geq \mu_0$ H_a: $\mu < \mu_0$
Estadístico de prueba	$t = \frac{\bar{X}_d - \mu_d}{\frac{s_d}{\sqrt{n}}}$
Regla de rechazo: **Método *p* - value**	Rechazar H_0 si p - value $\leq \alpha$
Regla de rechazo: **Método del valor crítico**	Rechazar H_0 si $t \leq$ - 4.3026 o Rechazar H_0 si $t \geq$ 4.3026

f. Toma de decisión de aceptar o rechazar Ho

Como el p - value = 0.02909 es menor que el nivel de significancia α = 0.05, y como el valor del estadístico de prueba t = 5.7337009 es mayor que el valor critico = 4.3026.Se tiene evidencia suficiente para rechazar la hipótesis nula.

Por lo tanto, se acepta la hipótesis alternativa H_a: **La aplicación de la base de datos NoSQL disminuye el tiempo en el acceso a la información en el proceso de segmentación de clientes.**

4.2. Discusión de resultados

- ✓ Con la aplicación del examen de diagnóstico Pre Test en el proceso actual, se puede notar que existe un bajo nivel de satisfacción del usuario interno obteniendo resultados de entre 30% y 60% que lo consideran *Malo* y *Muy Malo*. De la misma manera se puede apreciar que el nivel de automatización en el acceso a la información es considerado *Muy Malo* y *Malo* con resultados de 73% y 19% del total respectivamente. En cuanto al tiempo de acceso a la información en el proceso actual, se tienen resultados de rangos de tiempos con una diferencia significativa en comparación al examen de diagnóstico PostTest en los procesos considerados en la guía de observación. Estos resultados nos dan a conocer que actualmente no se cuentan con herramientas y/o tecnologías que ayuden a almacenar, organizar y analizar información importante en el proceso de segmentación de clientes; por lo que se está de acuerdo con Maureen, L., Espinoza, O. y Núñez, K. [10], quienes afirman que para conocer, analizar y evaluar las necesidades, preferencias, motivaciones y comportamiento de compra de los clientes ha comenzado a ser un tema relevante para las empresas, especialmente, porque se tienen que gestionar grandes volúmenes de información con relación a su cartera de clientes. Para ello, la utilización de nuevas tecnologías de la información y las comunicaciones son un factor clave para el desarrollo de los procesos de gestión del conocimiento y de la relación que mantienen las empresas con sus clientes. Finalmente ante el análisis anterior se comprueba que se cumple con el primer objetivo planteado: **describir el proceso actual de acceso a la información para la segmentación de clientes**.

- ✓ Para el cumplimiento del segundo objetivo específico: **examinar datos no estructurados de diferentes aplicaciones web para tomarlos en cuenta en la aplicación de bases de datos no relacionales NoSQL**, lo que se efectúo fue la consulta de las diferentes aplicaciones web que el CAPI utiliza o maneja actualmente para identificar los distintos tipos de datos potenciales e importantes que sirvan para el respectivo análisis y gestión en una base de datos NoSQL. Dentro de esa información se encontraron diferentes aplicaciones web que se utilizan como Twitter, Youtube, Facebook, Página Web, Linkedin; siendo las más utilizadas Facebook, *Youtube* y *Página web*, de las cuales se tomaron en cuentan para el desarrollo de éste trabajo sólo

Facebook y Página Web debido al proceso de nuestro caso y a los tipos de datos que el futuro motor de base de datos podría almacenar.

- ✓ Con la aplicación del examen de diagnóstico Post Test, se puede apreciar que existe una notable diferencia de mejora en los diferentes indicadores analizados en comparación con el PreTest. Para el indicador Nivel de satisfacción de usuario, después de aplicar la base de datos NoSQL, podemos apreciar que los resultados obtenidos han mejorado: el 57% lo considera Muy bueno, el 33% lo considera Bueno. Para el indicador Nivel de Automatización en el acceso a la información notamos una gran mejora con resultados de 62% y 33% que se consideran *Muy Bueno* y *Bueno* respectivamente. Los resultados de este indicador nos permiten evidenciar que nuestra propuesta es sin duda un gran aporte para mejor tiempos de accesos y consultas a información importante que ayuden en el proceso de segmentación de clientes. En cuanto al indicador tiempo de acceso a la información se puede apreciar que los tiempos disminuyen notablemente en los procesos medidos, teniendo resultados de no mayores a 2 minutos de cada proceso. Si comparamos éstos dos últimos indicadores estamos de totalmente de acuerdo con lo que menciona Barragán, A. y Forero, A. [2] en su trabajo de investigación: las bases de datos no relacionales proporcionan tiempos de respuesta mucho más bajos que las bases tradicionales, además de sus ventajas en escalabilidad y disponibilidad del sistema. También coincidimos con Ballón, J. [12], quien concluye que se reduce en un 60% el tiempo de elaboración de propuestas en base al registro de información histórica en bases de datos NoSQL. Finalmente ante lo anterior se comprueba que se cumple con el tercer objetivo específico planteado: **Aplicar bases de datos no relacionales, para mejorar el acceso a la información en el proceso de segmentación de clientes en el Centro de Actualización Profesional para Ingenierías CAPI.**

CAPITULO V. CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

- Con los resultados obtenidos en el capítulo anterior se demuestra que el sí se cumple el objetivo principal el cual es la mejora del acceso a la información en el proceso de segmentación de clientes en el Centro de Actualización Profesional para Ingenierías CAPI aplicando bases de datos NoSQL.

- La aplicación de una base de datos NoSQL, mejora el acceso a la información en el proceso de segmentación de clientes en el Centro de Actualización Profesional para Ingenierías CAPI, teniendo un tiempo de mejora de 89% que con el proceso actual; lo cual fue probado estadísticamente analizando los indicadores de la variable dependiente (Acceso a la información), los mismos que muestran una diferencia significativa entre el uso del proceso actual (pre-test) y la aplicación de la base de datos NoSQL (post-test).

- La etapa de Planeación y Definición de Requerimientos dentro de la Metodología propuesta permitió describir el proceso actual de acceso a la información para la segmentación de clientes, por lo que se concluye que se cumple con el primer objetivo específico planteado.

- La sub etapa de Extracción dentro de etapa de Construcción de la metodología propuesta permitió examinar datos no estructurados de diferentes aplicaciones web para tomarlos en cuenta en la aplicación de bases de datos no relacionales NoSQL, por lo que se concluye que se cumple con el segundo objetivo específico planteado

- Por todo lo anterior, se concluye que éste trabajo de investigación incluye varios resultados importantes, el primero de estos es la propuesta de una metodología propia para el desarrollo de éste trabajo, la cual se planteó con el fin de cumplir con el objetivo principal, enfocado siempre a la hipótesis planteada, siendo esto

un logro debido a la dificultad encontrada en documentación y experimentación para ajustarse al tema principal del caso de estudio.

- El segundo resultado es que para la utilización de la base de datos NoSQL, se hizo un previo de análisis de los diferentes tipos de motores de estos tipos de base de datos para que permita la implementación de éste caso específico, puesto que actualmente, el mercado ofrece una gran variedad de estos motores, dependiendo de las necesidades de almacenamiento que se busque. Además, esta propuesta es novedosa, puesto que involucra diversas herramientas intermediaras entre el usuario como PHP y el Shell del motor seleccionado MongoDB.

- Para finalizar se concluye que las bases de datos NoSQL, pueden ser de gran utilidad en muchos sectores y áreas, actualmente están teniendo un auge en cuanto a las redes sociales, debido a la cantidad de información generada. Pero es importante resaltar que el trabajo con bases de datos NoSQL requieren, en la mayoría de los casos, conocer bien el negocio que se desea modelar para definir adecuadamente la estructura en la que se van a almacenar los datos. Un esquema de datos bien ajustado a un negocio muy específico permite optimizar los resultados de las consultas desde la etapa de diseño.

5.2. Recomendaciones

- Se debe profundizar en temas relacionados con el mundo NoSQL y gestión de datos no estructurados para estandarizar una metodología que sirva como base para otros trabajos futuros en éste tema.
- En la etapa de Construcción se debe apostar por incluir nuevos modelos de programación más complejos para obtener información más concisa y precisa como por ejemplo reportes con gráficos y KPIs.
- Continuar con la construcción y mejora de la aplicación, sobre todo en la sub etapa de Carga: la cual debería de estar integrada en la aplicación, con la finalidad que el usuario interactúe directamente con la importación de la data y no la aplique por consola del motor de base de datos NoSQL.
- No debemos dejar de lado también la posibilidad de implementar soluciones hibridas que usen bases de datos relacionales y NoSQL en conjunto, por ejemplo Facebook usa MySQL para ciertos datos y Cassandra para cubrir otros requerimientos.

Referencias Bibliográficas

- [1] D. Brito," Estudio del uso de MongoDB como alternativa a las bases de datos relacionales tradicionales en aplicaciones web que requieren rapidez de lectura/escritura de los datos almacenados". Tesis de Pregrado, Facultad De Sistema Informáticos, Universidad Tecnológica Israel, Cuenca, Ecuador. Diciembre 2011

- [2] A. Barragán and A. Forero," Implementación de una base de datos NoSQL para la generación de una matriz O/D". Trabajo de Grado, Facultad De Ingeniería, Programa de Ingeniería de Sistemas, Universidad Católica de Colombia, Bogotá D.C. Mayo 2013

- [3] S. Mancilla," Uso de Base de datos NoSQL documentales para crear sitios web de alto rendimiento". Trabajo de Graduación, Facultad De Ingeniería, Escuela de Ingeniería en Ciencias y Sistemas, Universidad de San Carlos de Guatemala. Agosto 2013.

- [4] C. López, "Análisis de las Bases de Datos NoSQL como alternativa a las bases de datos SQL". Tesis Pregrado. Escuela de Ingeniería de Antioquía, Universidad de Antioquía, Envigado, 2012.

- [5] J. Ropero, "Método general de Extracción de Información basado en el uso de Lógica Borrosa. Aplicación en portales web." MS Tesis. Escuela Superior Técnica de Informática. Departamento de Tecnología Electrónica. Universidad de Sevilla, Sevilla, 2009.

- [6] L. Gallardo, F. Bermeo y V. Cedeño, "Sistema de reportes y análisis sobre tendencias en la Web de la ESPOL usando Hadoop para el procesamiento masivo de los datos", Facultad de Ingeniería en Electricidad y Computación. Escuela Superior Politécnica del Litoral (ESPOL), Guayaquil.

- [7] F. De la Rosa," Sistemas de Inteligencia Web: Análisis de redes sociales", MS Tesis, Departamento de Lenguajes y Sistemas Informáticos, Universidad de Sevilla, Sevilla, Febrero 2012.

- [8] J. Manyika and C. Roxburgh, "Big data: The next frontier for innovation, competition, and productivity," USA, May, 2011.

- [9] D. Linthicum," Big Data Analytics Deep Dive". IBM, USA, 2012. Available:http://resources.idgenterprise.com/original/AST-0073561_big_data_ibm_v2.pdf

- [10] L. Maureen, O. Espinoza and K. Nuñez, "Segmentación Basada En El Valor Del Cliente. Caso Aplicado A D&S S.A.". Trabajo de Investigación. Departamento de Administración y Auditoría, Facultad de Cs. Empresariales, Universidad del Bío, Concepción, Chile.

- [11] L. Herrera, "Implementación de una solución Cloud Computing usando una base de datos NoSQL para le gestión de datos de paciente con Diabetes Melitus". Tesis Pregrado. Facultad de Ingeniería de Sistemas e Informática. Escuela Académico Profesional de Ingeniería de Sistemas. Universidad Nacional Mayor de San Marcos. Lima, 2013

- [12] J. Ballón, "Implementación de un Sistema de Propuestas de Proyectos de Software en Avantica Technologies". Tesis Pregrado. Facultad de Ingeniería y Arquitectura. Carrera Profesional de Ingeniería de Computación y Sistemas. Universidad San Martín de Porres. Lima, 2014

- [13] N. Coronel, "Diseño de un Datamart para seguros masivos". Tesis pregrado. Facultad de Ingeniería Industrial y Sistemas. Carrera Profesional de Ingeniería de Sistemas. Universidad Tecnológica del Perú. Lima, 2012.

- [14] P. Uceda, "Utilización de tecnologías DataWarehouse para mejorar la toma de decisiones del área de créditos de la ONG Afider"", Proyecto Profesional. Facultad de Ingeniería. Escuela Académico Profesional de Ingeniería de Sistemas. Universidad Nacional de Cajamarca.Cajamarca,2005.

- [15] A. Martin, S. Chávez and M. Murazzo," Bases de Datos NoSql en Cloud Computing". XV Workshop de Investigadores en Ciencias de la Computación. Departamento e Instituto de Informática. San Juan, 2013.

- [16] A. Torre and A. Illarramendi,"Diseño de un repositorio RDF basado en Tecnologías NOSQL". Tecnalia Research & Innovation. Departamento de

Lenguajes y Sistemas Informáticos, San Sebastián. Available: http://lbd.udc.es/jornadas2011/actas/JISBD/JISBD/S1/Emergentes/JISBD2011_articulo.pdf

- [17] Acens, "Bases de datos NoSQL: Tipos". AcensWhitePappers. Febrero 2014.
- [18] M. Méndez, Big Data: ¿humo o reto corporativo? e- Penteo. Barcelona. 2012. Available: http://www.angelmendez.es/wp-content/big-data_humo-o-reto-corporativo_final.pdf
- [19] A. Augsbuger, ""Paralelización de un algoritmo para la detección de cúmulos de galaxias". MS Tesis. Facultad de Ciencia Físicas y Matemáticas. Departamentos de Ciencias de la Computación. Universidad de Chile. Santiago, 2012.
- [20] C. Díaz, "Evaluación de la herramienta de código libre Apache Hadoop". Trabajo de Investigación. Escuela Politécnica Superior. Universidad Carlos III de Madrid. Madrid, 2011.
- [21] O. Etzioni, "The World-Wide Web. Quagmire or Gold Mine" Communications of the ACM, November 1996, Vol. 39, no.11.
- [22] M. Scotto, A. Sillitti, Succi, T. Vernazza. "Managing Web-Based Information", International Conference on Enterprise Information Systems (ICEIS 2004), Porto, Portugal, April 2004. Page 1-3.
- [23] S. K. Pal, V. Talwar and P. Mitra, "Web Mining in Soft Computing Framework: Relevance, State of the Art and Future Directions". IEEE Transactions on Neural Networks Vol. 13, No. 5 pp 1163-1177, September 2002.
- [24] Y.H. Tao, T.P. Hong y Y.M. Su, "Web usage mining with intentional browsing data".Expert Systems Applications. 34(3): 1893-1904. 2008.
- [25] R. Kosala y H. Blockeel. "Web Mining Research: A Survey" ACM SIGKDD Explorations, Newsletter of the ACM Special Interest Group on Knowledge Discovery and Data Mining, June 2000, Vol. 2, nº1, pp. 1-15.

- [26] N. Kushmerick. "Gleaning answers from the Web". Proceedings of the AAAI Spring Symposium on Mining Answers from Texts and Knowledge Bases, Palo Alto, 43-45. 2002.

- [27] P. Nuñez Fuentes, "Segmentación de Clientes de una Cadena de Supermercados en base a estilos de vida". Memoria Doctoral. Facultad de Ciencias Físicas y Matemáticas. Universidad de Chile. Santiago, 2010.

- [28] C. Martínez Cruz, "Sistema de Gestión de Base de Datos Relaciones Difusas Multipropósito. Una Ontología para la Representación de conocimiento difuso". Tesis Doctoral. E.T.S. de Ingeniería Informática. Universidad de Granada. España, 2008.

- [29] E. Velásquez, "Inteligencia de Negocios". Tecnología PYME. Enero 2009. Available: http://www.tecnologiapyme.com/administracion-electronica/que-es-la-inteligencia-de-negocios-business-intelligence

- [30] E. Sevilla, "Guía Metodológica para la definición y desarrollo de un Datawarehouse". Tesis PreGrado. Facultad de Ingeniería. Universidad Americana. Nicaragua, 2003.

- [31] E. Ilbay, "Propuesta metodológica para aplicar Bussiness Intelligence. Caso Práctico"COHERVI S.A."". Tesis Pregrado. Facultad de Informática y Electrónica. Escuela de Ingeniería de Sistemas. Escuela Politécnica de Chimborazo. Riobamba, 2009.

- [32] E. Wayne and C. White. "Evaluating ETL and Data Integration Platforms".The Data Warehousing Institute. 101communications LLC. Seatle. Available: http://roberto-espinosa.es/doc/etlreport.pdf

ANEXOS

ANEXO 01:

CUESTIONARIO 01

Objetivo

Objetivo de la Encuesta: La presente encuesta tiene por objetivo determinar el Nivel de satisfacción de los usuarios en el acceso a la información en el proceso de segmentación de clientes.

Instrucciones

Lea atentamente cada una de las preguntas, revise todas las opciones y encierre en un círculo la respuesta que usted considere.

5	4	3	2	1
Muy Bueno	**Bueno**	**Regular**	**Malo**	**Muy Malo**

Ítem	Pregunta	Respuesta				
1.	¿Cómo califica el procedimiento de obtención de información utilizada en el proceso de segmentación de clientes?	1	2	3	4	5
2.	¿Cómo considera la organización y almacenamiento de la información utilizada en el proceso de segmentación de clientes?	1	2	3	4	5
3.	¿Cuál es su opinión sobre los medios utilizados para acceder a la información necesaria para el proceso de segmentación?	1	2	3	4	5
4.	¿Cómo califica el aporte de la información utilizada en el proceso de segmentación de clientes?	1	2	3	4	5
5.	¿Cómo califica el aporte de la información resultado de la segmentación de clientes en la propuesta de servicios de capacitación?	1	2	3	4	5
6.	¿Cómo considera el actual proceso de segmentación de clientes?	1	2	3	4	5
7.	¿Cómo califica la información disponible y relacionada con los requerimientos y preferencias de capacitación demandados?	1	2	3	4	5

¡Muchas Gracias!

Fecha: / /

ANEXO 02:

ENCUESTA 02

Objetivo

Objetivo de la Encuesta: La presente encuesta tiene por objetivo determinar Nivel de automatización en el acceso a la información, el tiempo de acceso a la información para campañas de segmentación.

Instrucciones

Lea atentamente cada una de las preguntas, revise todas las opciones y encierre en un círculo la respuesta que usted considere.

5	4	3	2	1
Muy Bueno	**Bueno**	**Regular**	**Malo**	**Muy Malo**

Ítem	Pregunta	Respuesta				
1.	¿Cómo califica el procedimiento para tener acceso a la información necesaria para el proceso de segmentación de clientes?	1	2	3	4	5
2.	¿Cómo califica los medios de almacenamiento de información utilizada en el proceso de segmentación de clientes?	1	2	3	4	5
3.	¿Cuál es su opinión sobre el nivel de seguridad en el almacenamiento y acceso a la información a utilizar en el proceso de segmentación de clientes?	1	2	3	4	5
4.	¿Cómo califica la herramienta de almacenamiento y reporte de información necesaria en el proceso de segmentación de clientes?	1	2	3	4	5
5.	¿Cómo califica la herramienta que le permita registrar y conocer las preferencias de capacitación?	1	2	3	4	5
6.	¿Cómo considera al rango de tiempo que le lleva a reunir información útil, para el proceso de segmentación de clientes?	1	2	3	4	5
7.	¿Cómo califica el tiempo que le toma realizar el proceso de segmentación de clientes?	1	2	3	4	5

¡Muchas Gracias!

Fecha: / /

Anexo 03:

GUIA DE OBSERVACIÓN

Objetivo
Objetivo de la Guía de Observación: La presente guía de observación tiene por objetivo el desempeño de tiempo en el acceso a la información en el proceso de segmentación clientes.

Ítem	Proceso	Tiempo (min)
P1.	Extracción de información de filiales más solicitadas	
P2.	Consulta de diplomados y/o cursos con más tendencia	
P3.	Extracción de información para campañas de marketing	

Fecha: / /