

COMPARATIVO DE BASE DE DATOS RELACIONALES Y NO RELACIONALES PARA LA TOMA DE DECISIONES DE USO

DR. VALENTÍN ÁLVAREZ HILARIO*

Facultad de Ingeniería, Universidad Autónoma de Guerrero, Chilpancingo, Guerrero, México, valentin_ah@uagro.mx

C. EUSTOLIA GÓMEZ SANTANA†

Facultad de Ingeniería, Universidad Autónoma de Guerrero, Chilpancingo, Guerrero, México, email@email.com

DR. EDGARDO SOLIS CARMONA

Facultad de Ingeniería, Universidad Autónoma de Guerrero, Chilpancingo, Guerrero, México, 09302@uagro.mx

MC. JORGE VAZQUEZ GALARCE

Facultad de Ingeniería, Universidad Autónoma de Guerrero, Chilpancingo, Guerrero, México, 13216@uagro.mx

* Profesor investigador de tiempo completo de la Universidad Autónoma de Guerrero.

† Egresada de la Facultad de Ingeniería.

RESUMEN

Este artículo es un análisis descriptivo de los modelos de bases de datos relacionales y no relacionales, pero también brinda un método de acuerdo a las características de cada modelo, para que el lector tenga un argumento preliminar para tomar una decisión al momento de implementar el sistema. Por otro lado, la comparación de modelos se realiza desde el punto de vista del motor de la base de datos, sin tener en cuenta las características del motor desarrollado por su proveedor. Finalmente, se propone un conjunto de criterios basados en la experiencia de los autores con el modelo. El propósito de este documento no es realizar un estudio comparativo de modelos relacionales versus no relacionales, sino proporcionar una herramienta para ayudar a los programadores a tomar decisiones al ejecutar proyectos de software.

ABSTRACT

This article is a descriptive analysis of relational and non-relational database models, but it also provides a method according to the characteristics of each model, so that the reader has a preliminary argument to make a decision when implementing the system. On the other hand, the comparison of models is made from the point of view of the database engine, without taking into account the characteristics of the engine developed by its supplier. Finally, a set of criteria based on the authors' experience with the model is proposed. The purpose of this paper is not to perform a comparative study of relational versus non-relational models, but to provide a tool to help programmers make decisions when executing software projects.

PALABRAS RESERVADAS

Base de Datos, Modelo Relacional, No SQL

KEYWORDS

Database - Information - Model - SQL - Relational - Non-Relational - No SQL

INTRODUCCIÓN

Hoy en día, cuando se trata de grandes cantidades de información almacenada en bases de datos, la pregunta es qué sistema de administración de bases de datos usar y, lo que es más importante, qué modelo usar. Esto se debe al aumento de información y datos de las aplicaciones actuales como Facebook, Twitter, etc. Este artículo proporciona un análisis descriptivo de los modelos de bases de datos relacionales y no relacionales para comprender cómo se almacenan los datos en su estructura, cómo se relacionan los datos almacenados en ella y de la misma manera proporciona una guía para que pueda elegir la base de datos del sistema de gestión de datos. Hay varias formas de elegir la mejor opción.

Parte de la encuesta es indagar el nivel de conocimiento de los estudiantes de séptimo, quinto y grado de la Escuela de Ingeniería de Sistemas ESPOCH sobre el manejo de modelos de bases de datos no relacionales, para conocer su nivel de conocimiento sobre nuevas bases de datos teóricas, con el fin de proponer alternativas, tales como nuevas tecnologías de la información. A modo de comparación, se realizará un balance entre los DBMS, exponiendo primero las características de cada modelo, determinando así su calidad para resaltar sus fortalezas y debilidades.

La información obtenida de cada modelo y la aplicación de la encuesta a los estudiantes de la FIE (Facultad de Informática y Electrónica) en ocupaciones de sistemas, se refiere a las diferentes bases de datos que implementan y manejan al momento de analizar los criterios de selección de la implementación. , sugerirá la mejor solución al elegir relacional o no relacional. Al detectar el uso limitado de bases de datos no relacionales, se propone un análisis de dos modelos de bases de datos para brindar orientación para cerrar la brecha entre las bases de datos relacionales bien conocidas y las bases de datos no relacionales innovadoras.

PLANTEAMIENTO DEL PROBLEMA

Hoy en día, cuando se trata de grandes cantidades de información almacenada en bases de datos, la pregunta es qué sistema de administración de bases de datos usar y, lo que es más importante, qué modelo se recomienda utilizar, esa es una problemática que vamos a analizar para estar preparados en las ventajas de una y otra alternativa.

MÉTODO

El uso de modelos de bases de datos y en base a las respuestas proporcionadas por el universo (población) de los estudiantes de la Facultad de ingeniería, se obtiene la recopilación de información, correspondiente a los estudiantes de la carrera de Ingeniero en Computación, quienes respondieron a la encuesta utilizando preguntas cerradas.

1.- MARCO TEÓRICO

El estudio es de carácter descriptivo ya que categoriza y describe cada base de datos y utiliza como técnica la encuesta.

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su uso posterior. En este sentido, una biblioteca puede ser pensada como una base de datos compuesta principalmente por documentos y textos impresos en papel e indexados para su consulta (Llanos Diego, 2010).

Una base de datos es una gran cantidad de información almacenada en registros para permitir una mayor eficiencia al ingresar, buscar, actualizar o eliminar información. En algunos casos, la información debe estar relacionada entre sí para evitar la duplicación de información y para organizar mejor la información; en otros casos, para mejorar el rendimiento, algunos desarrolladores no imponen relaciones de datos.

Las bases de datos han existido desde que las personas sintieron la necesidad de organizar la información que generan todos los días. Fue así como diseñaron los archivadores. Es una forma de almacenamiento de datos donde se organizan y relacionan los archivos. Pero los datos de estas bases de datos no están informatizados. No fue hasta 1963, en un simposio realizado en California, que en Estados Unidos se introdujo el término base de datos informática, cuya lógica estaba muy relacionada con la de base de datos no informática (Archivador), pero cuyo concepto era diferente, muy interesante y ambicioso en la época.

Actualmente, el procesamiento de grandes cantidades de información y las transacciones se realizan cada vez más de forma simultánea, como la toma de decisiones a la hora de elegir un sistema de gestión de bases de datos. Para lograr el objetivo de determinar qué base de datos es la mejor opción al elegir una base de datos relacional o no relacional, se siguió el siguiente proceso.

Se realizó una encuesta de cada sistema comparando el uso de cada sistema al realizar acceso relacional frente al realizar MongoDB no relacional, lo que arrojó la siguiente información.

Uno de los DBMS es sin duda Access, o Microsoft Access, un software que permite gestionar bases de datos. Este programa forma parte de Microsoft Office, que es un paquete de aplicaciones que te permite realizar tareas de oficina. (Pérez Julián, Porto y Merino María, 2017).

Por otro lado, el término base de datos no relacional se introdujo en 1998, pero Johan Oskarsson no tuvo conocimiento de la tendencia no relacional o NoSQL (Not Only SQL) hasta el año 2000.

Como base de datos no relacional tenemos:

Clave-valor: son altamente divisibles, lo que permite escalar horizontalmente, lo que permite que las aplicaciones adapten el rendimiento a medida que aumentan los usuarios. En situaciones como los juegos, la tecnología publicitaria funciona bien en este modo. Por ejemplo, DynamoDB y MongoDB están diseñados para ofrecer latencias de milisegundos consistentes de un solo dígito para cargas de trabajo de cualquier tamaño. Carece de consistencia a nivel de base de datos, lo que permite que las aplicaciones brinden mayor confiabilidad. Codifique los datos almacenados como XML, JSON o BSON.

Documentación: Es un modelo gratuito para filas y columnas desnormalizadas. Por lo general, en el nivel de la aplicación, los datos se representan como documentos JSON porque es más intuitivo para los desarrolladores ver sus modelos de datos como documentos. Las bases de datos de documentos son cada vez más populares porque los desarrolladores pueden almacenar datos en la base de datos usando el mismo formato de modelo de documento que usan en su código de aplicación.

Graph: Es una base de datos de gráficos que facilita la creación y ejecución de aplicaciones que trabajan con conjuntos de datos altamente conectados, una "estructura gráfica dirigida" utilizada para representar datos. Un gráfico consta de aristas y nodos. Ejemplos típicos de estas bases de datos son: redes sociales, motores de recomendación, detección de fraude y gráficos de conocimiento. Neptune es un servicio de base de datos de gráficos que admite tanto el modelo de gráfico de propiedades como el marco de descripción de recursos (RDF).

En memoria: las aplicaciones de juegos y tecnología publicitaria tienen casos de uso como tablas de clasificación, almacenamiento de sesiones y análisis en tiempo real que requieren tiempos de respuesta de microsegundos y pueden experimentar grandes picos de tráfico en cualquier momento. Por ejemplo, Redis es muy eficiente para atender cargas de trabajo de baja latencia y alto rendimiento, como McDonald's, que no se pueden atender mediante almacenes de datos basados en disco.

El tipo no relacional es un tipo de sistema de gestión de base de datos que se diferencia del modelo clásico de sistema de gestión de base de datos relacional, principalmente en los siguientes aspectos: no utiliza SQL como lenguaje de consulta principal, los datos almacenados no requieren estructuras fijas como como tablas, por lo general no admite operaciones JOIN y ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad) no está totalmente garantizado y, por lo general, escala bien horizontalmente (Adam Lith, Mattson Jakob, 2010).

Otro aspecto importante a considerar cuando se analiza un modelo es el término escalabilidad.

La escalabilidad vertical se refiere al crecimiento del hardware ya que la información se tiene que pasar a otro servidor más potente, o por el contrario tener una imagen espejo en otro servidor para que si el servidor principal falla, la copia de seguridad se utilice de inmediato hasta que se resuelva el problema y se coloque. de vuelta Entonces eso significa costo.

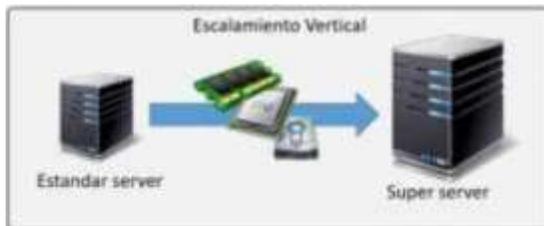


Imagen 1. Escalabilidad vertical.

Fuente: Blancarte, O.(2017): "Software Architect" [online]. Mexico, 2017. Disponible en: <https://www.oscarblancarteblog.com/2017/03/07/escalabilidad-horizantal-y-vertical>.

La escalabilidad horizontal se refiere al uso de nodos, lo que significa que el trabajo se distribuye, como si tuviera diferentes espejos de bases de datos distribuidos en diferentes servidores, si un nodo falla, los otros nodos seguirán funcionando.

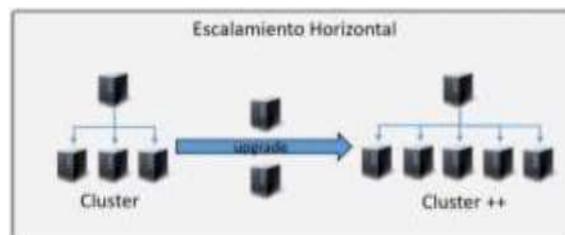


Imagen 2 Escalabilidad horizontal.

Fuente: Blancarte, O.(2017): "Software Architect" [online]. Mexico, 2017.Disponible en:
<https://www.oscarblancarteblog.com/2017/03/07/escalabilidad-horizantal-y-vertical>

Las bases de datos relacionales utilizan los términos ACID que se explican a continuación.

Atomicidad. La transacción debe ejecutarse en su totalidad o no podrá ejecutarse.

Consistencia. Es capaz de pasar de un movimiento efectivo a otro, siempre obedeciendo las reglas establecidas.

Aislamiento. Se pueden realizar múltiples transacciones contra el mismo registro, y ambas son independientes.

Durabilidad. Una vez ejecutada la transacción, conserva los cambios realizados.

Las bases de datos no relacionales utilizan el teorema CAP, lo que significa:

Consistencia. Indica que la actualización o modificación realizada debe mostrarse a cada acceso realizado como actualización.

Disponibilidad. Siempre debe tener acceso rápido a la información.

Tolerancia de partición. El sistema debería funcionar siempre, incluso si algunas partes del mismo son actualmente inasequibles.

Una vez implementadas las prácticas de gestión en cada software, a continuación se muestra una tabla con información relevante para cada sistema.

Tabla 1. Características de cada Bases de datos.

Base de Datos Relacional (SQL)	Base de Datos No Relacional (No SQL)
Si maneja esquemas	No es necesario realizar un esquema
En muchos casos se debe normalizar	No es necesario la normalizacion
Maneja tablas con registros	Maneja colecciones con objetos y arreglos a lo que se denomina documentos
Siempre se debe crear la clave primaria	No necesariamente crear la primaria pues el sistema internamente le identifica con una
Muy util al manejar transacciones	No maneja transacciones
Se desperdicia espacio en memoria al tener campos vacios	Existe desperdicio de espacio en memoria al repetir informacion
Una vez disenada la base de datos es complicada realizarle modificaciones	Una vez creado el objeto es sencillo realizarle cambios
Crecimiento vertical	Crecimiento horizontal
Prioridad consistencia	Prioridad disponibilidad

Las bases de datos relacionales son atómicas, es decir, puedes cambiar varias tablas al mismo tiempo o ninguna de ellas, esto no es adecuado para bases de datos no relacionales, ya que si quieres cambiar la información de varios objetos, necesitas agregar varios cambios. a los objetos según sea necesario.

No tiene sentido tratar de adaptar un esquema relacional a un esquema no relacional, es mejor aplicar una base de datos no relacional desde el principio, y si tiene muchos datos relacionados, no tiene sentido usar una base de datos relacional.

Si la base de datos necesita realizar transacciones, entonces se debe usar una base de datos relacional.

En una base de datos relacional al aumentar un campo a la tabla este campo es para todos los registros de la misma, mientras que en un no relacional si se tiene que aumentar un campo a un objeto solo se lo puede realizar específicamente a ese objeto sin necesidad de aumentarlo en los demás lo que disminuye espacio de memoria. Si se realiza una búsqueda en una relación de uno a varios se tiene que realizar consultas en la tabla padre para discernir en la tabla hija, mientras que en una no relacional al realizar una consulta se obtiene toda la información del objeto directamente debido a que en el objeto se tiene toda su información y si

se desea expresar por ejemplo una relación de muchos se la puede hacer por medio de vectores donde se ingresa toda la multiplicidad de información que exista.

En una base de datos relacional este mal uso de espacio en blanco en campo de algunos de sus registros se lo puede solucionar al realizar normalización, pero no en su totalidad.

La decisión para utilizar una base de datos relacional o una no relacional siempre debe estar enfocada en lo que el programador necesita priorizar, como rapidez de consulta, flexibilidad de arquitectura, rendimiento etc.

En muchos casos las bases de datos están diseñadas para organizar la información que genera una aplicación, son bases de datos que permiten almacenar datos de forma permanente, cada tipo de base de datos es importante dependiendo del proyecto, al momento de planificar un proyecto de software es importante considerar los datos ¿Qué son? las prioridades; por ejemplo, la velocidad de búsqueda, la solidez, el rendimiento, la integridad de los datos, etc. Los diseñadores quieren dar más peso, las decisiones relacionales o no relacionales favorecerán una técnica u otra.

2.- APLICACIÓN DE LA ENCUESTA

A continuación, se muestran las preguntas más representativas de la encuesta aplicada a los estudiantes de la facultad.

Qué modelo manejador de bases de datos ha utilizado?

El 66% de los estudiantes encuestados tienen una notable inclinación en el uso de las base de datos relacionales, esto puede ser porque en el transcurso de su carrera los conocimientos adquiridos con respecto a las bases de datos se inclinaron más por los modelos relacionales, apenas un 2% de los estudiantes han tenido al menos una experiencia con los modelos no racionales y un 32% han realizado trabajos diversos utilizando los dos modelos en diferentes ámbitos o alcances de los sistemas que han desarrollados ya sea para proyectos de fin de semestre o trabajos particulares.

En que se basa para la selección de un modelo manejador de base de datos?

El sesenta y dos por ciento de los estudiantes encuestados dijeron que eligieron un modelo de base de datos para un proyecto en función de la escalabilidad que aportaría al proyecto, mientras que el 26 por ciento eligió

un modelo de base de datos en función de la facilidad de implementación del proyecto. El modelo a utilizar. Solo el 12% indicó que preferiría elegir el modelo más conocido en función de la cantidad de información posible para tener las herramientas para implementar el modelo elegido.

Que escalabilidad considera que es la mejor?

El 54% de los estudiantes encuestados sintieron que la escalabilidad vertical les dio mejores resultados al implementar un proyecto final o un emprendimiento personal, el 30% sintieron que se sentían más cómodos usando la escalabilidad horizontal, el 16% las personas no conocen el término.

La escalabilidad vertical hace referencia al modelo de base de datos relacional y se considera la opción con mayor porcentaje cuando se utiliza con mayor frecuencia.

Considera que es un problema la duplicidad en información?

El 94% de los estudiantes encuestados no está de acuerdo con la posible presencia de duplicación constante de datos en sus modelos de base de datos, ya que cree que es un problema al implementar proyectos de software, ya que pueden ocurrir errores al usar datos de la capa de aplicación; el 6% no piensa que la duplicación de información es un problema.

El modelo relacional es muy estricto en este término, ya que para mayor seguridad se aplica normalización y se compensan cuestiones al tener esto en cuenta. Una no-relación no entra en conflicto con este término.

Para la implementación de un sistema informático con base de datos a que se le debe dar prioridad?

El 80% de los estudiantes encuestados cree que al elegir un modelo de base de datos se debe priorizar el crecimiento que logrará porque los datos siempre deben estar registrados y almacenados en un motor potente y eficiente; el 20% dijo que la prioridad debe ser la velocidad porque los usuarios actuales necesitan para generar tareas de manera más rápida y ágil y eficiente.

Los resultados obtenidos reafirman el apoyo a las características del modelo no relacional.

Con su experiencia como programador considera que se puede realizar un sistema académico en un SGBD No relacional?

El 72% de los estudiantes encuestados piensa que puede usar un modelo no relacional para los sistemas académicos, mientras que el 28% piensa que no es recomendable crear un sistema académico con un modelo no relacional.

Teniendo en cuenta los resultados anteriores sobre el crecimiento, se puede suponer que el grupo de estudiantes votó por este nuevo sistema.

Considera importante el uso de procedimientos almacenados, vistas, triggers?

El 100% de los estudiantes entrevistados considera importante el uso de procedimientos almacenados, vistas, disparadores, al momento de implementar sus proyectos de software.

Este es un término que aporta gran valor a los sistemas de bases de datos relacionales, pero en los sistemas de bases de datos no relacionales no es necesario realizar consultas complejas, por razones en aplicaciones donde una mayor familiaridad con los modelos relacionales tiende a favorecer los conceptos en ellas.

Cuando sale al mercado una nueva base de datos, la adquiere con facilidad?

El 82% de los encuestados respondió que era difícil obtener nuevas versiones y/o motores de base de datos actualizados para ejecutar sus proyectos, ya sea por los altos costos de licencia o por la dificultad para acceder a la ubicación y/o ubicaciones de descarga; el 18% respondió que tiene acceso a los canales de distribución para las nuevas versiones del motor de base de datos.

Los estudiantes insistieron en que la conocida era mejor, porque sentían que la nueva base de datos era inestable y no estaba suficientemente documentada.

Considera que las Bases de Datos Relacionales son la mejor opción para todos los tipos de almacenamiento?

El 64 % de los encuestados cree que las bases de datos relacionales son más eficientes para almacenar datos, mientras que el 36 % cree que las bases de datos relacionales no son eficientes para almacenar datos en sus registros.

Está claro que la gente sabe más sobre bases de datos relacionales, pero el término preferido es bases de datos no relacionales, aunque no se aplican con tanta frecuencia.

Usaría Bases de Datos No Relacional para su próximo proyecto?

El 48 % de los encuestados respondió que no usaría un modelo no relacional en su próximo proyecto de software; el 42 % pensó que usaría un modelo no relacional en su próximo proyecto; el 10 % respondió que lo haría cuando tomara una decisión.

Debido al desconocimiento de las nuevas tecnologías, los alumnos no utilizarán bases de datos no relacionales, prefiriendo seguir aplicando con soltura lo que ya saben.

CONCLUSIONES RESULTADOS

Una vez aplicada la encuesta y analizados los resultados se obtuvo la siguiente información: Los estudiantes no aplicaron bases de datos no relacionales por desconocimiento, el 90% dijo trabajar en Windows, el 10% trabajó en Windows y Linux, y también trabajaron en bases de datos de datos relacionales porque los conocen al 100%, pero en base a los resultados obtenidos les llamó la atención los términos de bases de datos no relacionales.

El 66% de los encuestados dijo que sabía de bases de datos relacionales, mientras que el 2% sabía de bases de datos no relacionales, y solo el 32% conocía ambas, pero eso no significa que las usaría, ya que el 4% mostró conocimiento de bases de datos no relacionales. Mientras que el 96% prefiere las bases de datos relacionales por su familiaridad, pero no se excluye el conocimiento del tema.

DISCUSIÓN

Una vez comparados los dos modelos, podemos proceder a la tabulación y análisis de la encuesta. La población serán todos los estudiantes de Ingeniería en computación según lo indicado por los estudiantes de quinto, séptimo y noveno grado de la carrera de Ingeniero en Computación.

Hipótesis: La práctica académica de bases de datos relacionales limita el uso de bases de datos no relacionales.
 H_0 = Los estudiantes conocen las bases de datos no relacionales.

H_1 = Los estudiantes desconocen las bases de datos no relacionales.

Para probar la hipótesis se utilizó una distribución chi-cuadrado por tratarse de una muestra pequeña y un nivel de significancia de 0.05 por tratarse de un proyecto de investigación.

BASE	Valor 1	Valor 2	TOTAL
Relacional	3	5	8
NO Relacional	1	1	2
	4	6	10
VE1=	3,2	VE2=	1,8
VE3=	0,8	VE4=	1,2
	0,0125	0,00833333	
	0,05	0,03333333	
	$\chi^2 =$	0,10416667	
	P <	0.05	

Estos valores se observan en la tabla y determinamos los valores asumidos. grados

de libertad = $(2-1)(2-1) = 1$

Usando grados de libertad y 0,05, buscamos en la tabla el valor de χ^2 , que es 0,004.

Cuando el valor χ^2 calculado > tabla χ^2 , rechace la hipótesis nula y acepte la hipótesis alternativa.

$0.104167 > 0.004$

Entonces, en nuestro caso, rechazamos H_0 vacío, por lo que los estudiantes en realidad no conocen las bases de datos no relacionales, por lo que rara vez se usan. Por lo tanto, necesitan aprender sobre nuevos modelos de sistemas de bases de datos y artículos que los animen y guíen en qué modelo aplicar.

CONCLUSIONES

Cuando el uso de transacciones no es tan importante, se recomienda utilizar la práctica de bases de datos relacionales.

Los objetos en las bases de datos no relacionales contienen toda la información, mientras que en las bases de datos relacionales se distribuye entre tablas.

No es posible decir con certeza que una base de datos es mejor que la otra, ya que ambas están diseñadas para mejorar el flujo de información entre las aplicaciones y los contenedores de datos.

Las bases de datos no relacionales son efectivas cuando los datos crecen en tiempo real y existe la necesidad de brindar información procesada a partir de grandes volúmenes de datos con una estructura horizontal.

Los requisitos cambiantes o dinámicos son un gran problema al diseñar bases de datos porque los usuarios siempre tienen algo que cambiar, y generalmente lo hacen cuando el diseño ya está ensamblado y en producción, lo cual es cierto para las bases de datos no relacionales es ventajoso porque es indiferente al solicitud inicial.

Por el lado de la extracción, las bases de datos relacionales están muy desarrolladas porque gestionan mejor las consultas gracias a operadores como Join.

Las facultades deben proporcionar un mecanismo para que los estudiantes amplíen sus horizontes intelectuales a otras tendencias, incentivando a los estudiantes a utilizar diferentes tendencias, modelos que existen actualmente.

La renovación del currículo de aula será determinante para que los estudiantes de las carreras de sistemas tengan la oportunidad de comprender ambos modelos y aplicarlos; de manera que en un momento dado tengan el conocimiento y la capacidad de decir cuál modelo de acuerdo al dominio en el que se encuentra el problema se desarrolla mejor.

El desconocimiento de los estudiantes en modelos no relacionales es una gran limitante para poder desarrollar mejor esta tendencia.

REFERENCIAS

- Adam Lith, Mattson Jakob. (2010): Investigating storage solutions for large data. Recuperado de:<http://publications.lib.chalmers.se/records/fulltext/123839.pdf> Consultado en 11/03/2019
- AWS(Sf).Amazon DocumentDB. Recuperado de: <https://aws.amazon.com/es/nosql/> Línea de Código Aprende a programar (2016). Anadir N elementos a un array en MongoDB. <http://lineadecodigo.com/mongodb/anadir-n-elementos-a-un-array-en-mongodb/> Consultado en 11/03/2019
- Blancarte, O.(2017): “Software Architect” [online]. Mexico, 2017. Disponible en: <https://www.oscarblancarteblog.com/2017/03/07/escalabilidad-horizonta-y-vertical>. Consultado en 21/01/2019
- Dos Ideas (2013): ACID en las bases de datos Recuperado de: <https://dosideas.com/noticias/base-de-datos/973-acid-en-las-bases-de-datos> Consultado en 11/03/2019
- Chaparro Vega, Jherson Jhair. (2016): Como se aplica ACID en las Bases de datos No SQL. Recuperado de: <https://prezi.com/scb0mofxixux/acid-en-nosql/><https://medium.com/@eugeniomendoza/c%C3%B3mo-saber-si-necesitas-una-base-de-datos-nosql-b6cfd5bb7d9b> Consultado en 11/03/2019

Genbeta. (2014): MongoDB: que es, como funciona y cuando podemos usarlo (o no). Recuperado de:
<https://rn.genbeta.com/desarrollo/mongodb-que-es-como-funciona-y-cuando-podemos-usarlo-o-no>
Consultado en 11/03/2019

Llanos Ferraris, Diego Rafael.(2010): Fundamentos de informatica y programacion en C.
Recuperadode:<https://books.google.es/books?id=FfEfCB-hXCgC&pg=PT297&dq=base+datos+relacional+codd&hl=es&sa=X&ved=0ahUKEwjixfLYxcPXAhXMWhQKHQImAfUQ6AEIQjAF#v=onepage&q=base%20datos%20relacional%20codd&f=false> Consultado en 30/01/2019

Perez Julian, Porto y Merino Maria. (2017): Definicion de Access. Recuperado de:
<https://definicion.de/access/>

Rubenza. (2013): Tutorial MongoDB. Operaciones de consulta. Recuperado de: <https://charlascylon.com/2013-06-26-tutorial-mongodb-operaciones-de-consulta>

Consultado en 11/03/2019

Yesica Lizeth. (2018): Que es el teorema CAP y como elegir la base de datos para tu proyecto. Recuperado de:
<https://platzi.com/blog/que-es-el-teorema-cap-y-como-elegir-la-base-de-datos-para-tu-proyecto/> Consultado en 11/03/2019