AWS “Describir solo con las caracteristicas escenciales”

Lambda funciones

API Gateway

Dynamo Bases de datos no relacionales

MARCO TEORICO

1. MaTraGra

Bajo la premisa de construcción de la metodología ágil en ambientes contextualizados por los trabajos de grado en el espació académico en Ingeniería, dispuestos por Instituciones de Educación superior, para el cumplimiento de una práctica integral y consecuentes con la generalidad de los trabajos de grado, se disponen de dos roles:

* **Estudiante**: adquiere la máxima responsabilidad académica en el proceso del trabajo de grado, derivada de la responsabilidad en el cumplimiento de lo pactado en el anteproyecto, relacionada con el desarrollo del aplicativo y la documentación en torno a él y al proyecto abordado. Puede considerarse en forma individual o en equipo como se disponga.
* **Director**: asume roles de líder encargado de direccionar el proceso integralmente, con la máxima responsabilidad de validar ingenierilmente los productos entregables y la aplicación final, asimismo, la documentación de apoyo al proceso. Implícitamente se advierte en él, el papel de representación informal del usuario de la aplicación, en acciones de requerimientos y validación. Productor de las órdenes o certificaciones finales de cumplimiento.

**DOCUMENTACIÓN:** se dispone como parte del proceso metodológico y para efectos de sustentación documental del trabajo abordado, el cumplimiento de los compromisos adquiridos aprobados y registrados en el anteproyecto, de dos formas o categorías concurrentes de documentación:

* **Registro de actividades**: en el cual se consignan, las actividades propuestas y realizadas durante todo el proceso, con el control de fechas correspondientes y las observaciones pertinentes -figura 1-. Además, puede usarse como instrumento de planeación de agenda de actividades y registro sustentable del proceso.

Amazon Web Services (AWS abreviado) es una colección de servicios de computación en la nube pública (también llamados servicios web) que en conjunto forman una plataforma de computación en la nube, ofrecidas a través de Internet por Amazon.com.

1. Aplicación web

Una aplicación web, es un programa que está almacenado en un servidor remoto y desplegado en internet a través de la interfaz de los navegadores web. Estás son consideradas como servicios web y la gran mayoría, aunque no todas las paginas web, son aplicaciones web. Para que una página web sea considerada aplicación web, esta debe ejecutar alguna función en tiempo real por su usuario (llenar un formulario, crear información nueva, etc.)

Las aplicaciones web son diseñadas para una gran variedad de usos y pueden ser usadas por cualquier persona con acceso a un navegador web. Los tipos de aplicaciones web más comunes son: aplicaciones e-mail (Gmail, Outlook, Yahoo!), calculadoras online o tiendas e-commerce. Algunas aplicaciones web solo pueden ser accedidas desde navegadores web específicos, sin embargo, la mayoría son accesibles desde todos los navegadores web.

1. Ingeniería de software

Las mejores definiciones de ingeniería de software pueden ser:

* Según la IEEE: “La aplicación tecnológica y científica de conocimiento, procesos y experiencia para diseñar, implementar, probar y documentar software” (The Bureau of Labor Statistics—IEEE Systems and software engineering)
* Según la IEEE: “La aplicación de métodos sistemáticos, disciplinados y cuantificables para el desarrollo, la operación y el mantenimiento de software” (IEEE Standard Glossary of Software Engineering Terminology)
* Según Ian Sommerville: “Una disciplina ingenieril que se ocupa de todos los aspectos relacionados con producción de software”
* Según Fritz Bauer: “El uso de principios de ingeniería para económicamente obtener software de confianza y que funciona eficientemente en computadores reales”
* Según Merriam Webster: “Una rama de la ciencia de la computación que se encarga del diseño, la implementación y el mantenimiento de complejos programas de computación”
* Según Google: “La ingeniería de software, no se refiere solamente solamente el acto de escribir código, también se refiere a todas las herramientas y procesos que una organización utiliza para para crear y mantener ese código durante todo su ciclo de vida. Ingeniería de software puede ser pensada como la programación integrada a través del tiempo”

Para concluir, la ingeniería de software es el proceso de examinar y determinar las necesidades de los usuarios. Esto se hace desarrollando y diseñando, y al final probando el software para saber si suple las necesidades del usuario final. La ingeniería de software es usada para sistemas más grandes y complejos, los cuales son creados por organizaciones y negocios. Hay diferentes tipos de ingeniería de software. Esta el front-end, donde la parte visual de la aplicación es construida y Esta el back-end, en donde los ingenieros crean las partes que se encargan de recibir, almacenar y tratar la información.

1. Computación en la nube

La computación en la nube es la disponibilidad a pedido de los recursos del sistema informático, especialmente el almacenamiento de datos y la capacidad de cómputo, sin una gestión activa directa por parte del usuario. El término se usa generalmente para describir los centros de datos disponibles desde cualquier lugar para muchos usuarios a través de Internet desde cualquier dispositivo móvil o fijo.

A menudo, el término «computación en la nube» se lo relaciona con una reducción de costos, disminución de vulnerabilidades y garantía de disponibilidad. Asimismo, la computación en la nube se la relaciona con un modelo de pago por uso. No obstante, el concepto de pago no puede ser solo relacionado con erogación económica dado que solo aplica en caso de proveedores externos, y en muchos casos hace referencia a poder medir el consumo aplicándose a centros de costos de la propia empresa.

La computación en la nube es un nuevo modelo de prestación de servicios tecnológicos que impacta sin lugar a dudas en diversos negocios. Este modelo se apoya en infraestructuras tecnológicas dinámicas, caracterizados por la virtualización de recursos, un alto grado de automatización, una elevada capacidad de adaptación para atender demandas variables.

El concepto de «nube informática» es muy amplio, y abarca casi todos los posibles tipos de servicio en línea, pero cuando las empresas predican ofrecer un utilitario alojado en la nube, por lo general se refieren a alguna de estas tres modalidades: el software como servicio (SaaS, por sus siglas en inglés), plataforma como servicio (PaaS) e infraestructura como servicio (IaaS).

El software como servicio es un modelo de distribución de software en el que las aplicaciones están alojadas por una compañía o proveedor de servicio y puestas a disposición de los usuarios a través de una red, generalmente internet. La plataforma como servicio es un conjunto de utilitarios para abastecer al usuario de sistemas operativos y servicios asociados a través de Internet sin necesidad de descargas o instalación alguna. Infraestructura como servicio se refiere a la tercerización de los equipos utilizados para apoyar las operaciones, incluido el almacenamiento, hardware, servidores y componentes de red.

1. Bases de datos NoSQL

(https://www.grapheverywhere.com/bases-de-datos-nosql-marcas-tipos-ventajas/)

El termino NoSQL se refiere a la denominación en inglés Not Only SQL. Plantea modelos de datos específicos de esquemas flexibles que se adaptan a los requisitos de las aplicaciones más modernas.

(https://www.acens.com/wp-content/images/2014/02/bbdd-nosql-wp-acens.pdf)

Las bases de datos NoSQL son sistemas de almacenamiento de información que no cumplen con el esquema entidad–relación. Tampoco utilizan una estructura de datos en forma de tabla donde se van almacenando los datos, sino que para el almacenamiento hacen uso de otros formatos

Esta forma de almacenar la información ofrece ciertas ventajas sobre los modelos relacionales. Entre las ventajas más significativas podemos destacar:

* **Se ejecutan en máquinas con pocos recursos:** Estos sistemas, a diferencia de los sistemas basados en SQL, no requieren de apenas computación, por lo que se pueden montar en máquinas de un coste más reducido.
* **Escalabilidad horizontal:** Para mejorar el rendimiento de estos sistemas simplemente se consigue añadiendo más nodos, con la única operación de indicar al sistema cuáles son los nodos que están disponibles.
* **Pueden manejar gran cantidad de datos:** Esto es debido a que utiliza una estructura distribuida, en muchos casos mediante tablas Hash.
* **No genera cuellos de botella:** El principal problema de los sistemas SQL es que necesitan transcribir cada sentencia para poder ser ejecutada, y cada sentencia compleja requiere además de un nivel de ejecución aún más complejo, lo que constituye un punto de entrada en común, que ante muchas peticiones puede ralentizar el sistema.

Algunas de las diferencias más destacables que nos podemos encontrar entre los sistemas NoSQL y los sistemas SQL están:

* **No utilizan SQL como lenguaje de consultas.** La mayoría de las bases de datos NoSQL evitan utilizar este tipo de lenguaje o lo utilizan como un lenguaje de apoyo. Por poner algunos ejemplos, Cassandra utiliza el lenguaje CQL, MongoDB utiliza JSON o BigTable hace uso de GQL.
* **No utilizan estructuras fijas como tablas para el almacenamiento de los datos.** Permiten hacer uso de otros tipos de modelos de almacenamiento de información como sistemas de clave–valor, objetos o grafos.
* **No suelen permitir operaciones JOIN.** Al disponer de un volumen de datos tan extremadamente grande suele resultar deseable evitar los JOIN. Esto se debe a que, cuando la operación no es la búsqueda de una clave, la sobrecarga puede llegar a ser muy costosa. Las soluciones más directas consisten en desnormalizar los datos, o bien realizar el JOIN mediante software, en la capa de aplicación.
* **Arquitectura distribuida.** Las bases de datos relacionales suelen estar centralizadas en una única máquina o bien en una estructura máster–esclavo, sin embargo, en los casos NoSQL la información puede estar compartida en varias máquinas mediante mecanismos de tablas Hash distribuidas.

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. Algunos ejemplos de este tipo son Cassandra, BigTable o HBase.
* **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 en grafo:** 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.

* **Bases 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.