

UNIVERSIDAD AUTONOMA GABRIEL RENE MORENO
FACULTAD DE INGENIERIA EN CIENCIAS DE LA COMPUTACION Y
TELECOMUNICACIONES



SISTEMA DE INFORMACION PARA UNA EMPRESA DE
TELECOMUNICACIONES

ESTUDIANTE: TORREZ CONDARCO CARLOS EDUARDO

MATERIA: SISTEMA DE INFORMACION II

GRUPO: SB

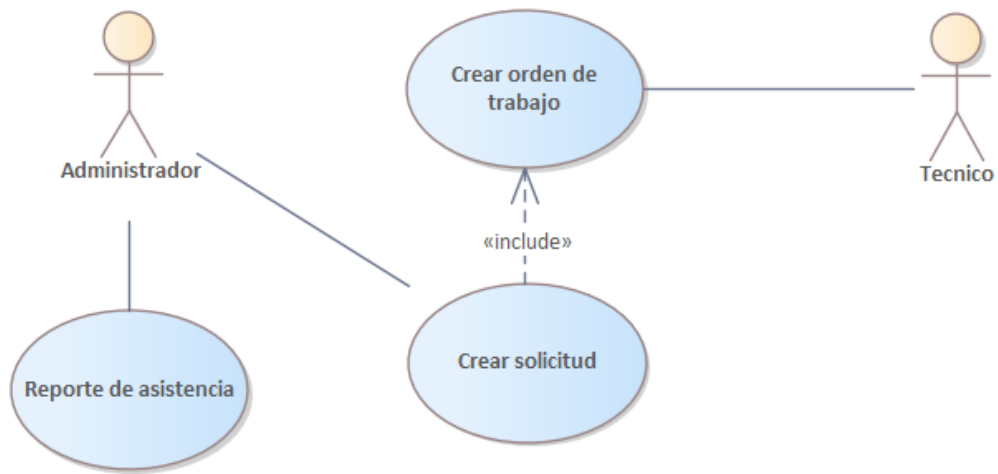
DOCENTE: ING. MARTINEZ CANEDO ROLANDO ANTONIO

Contenido

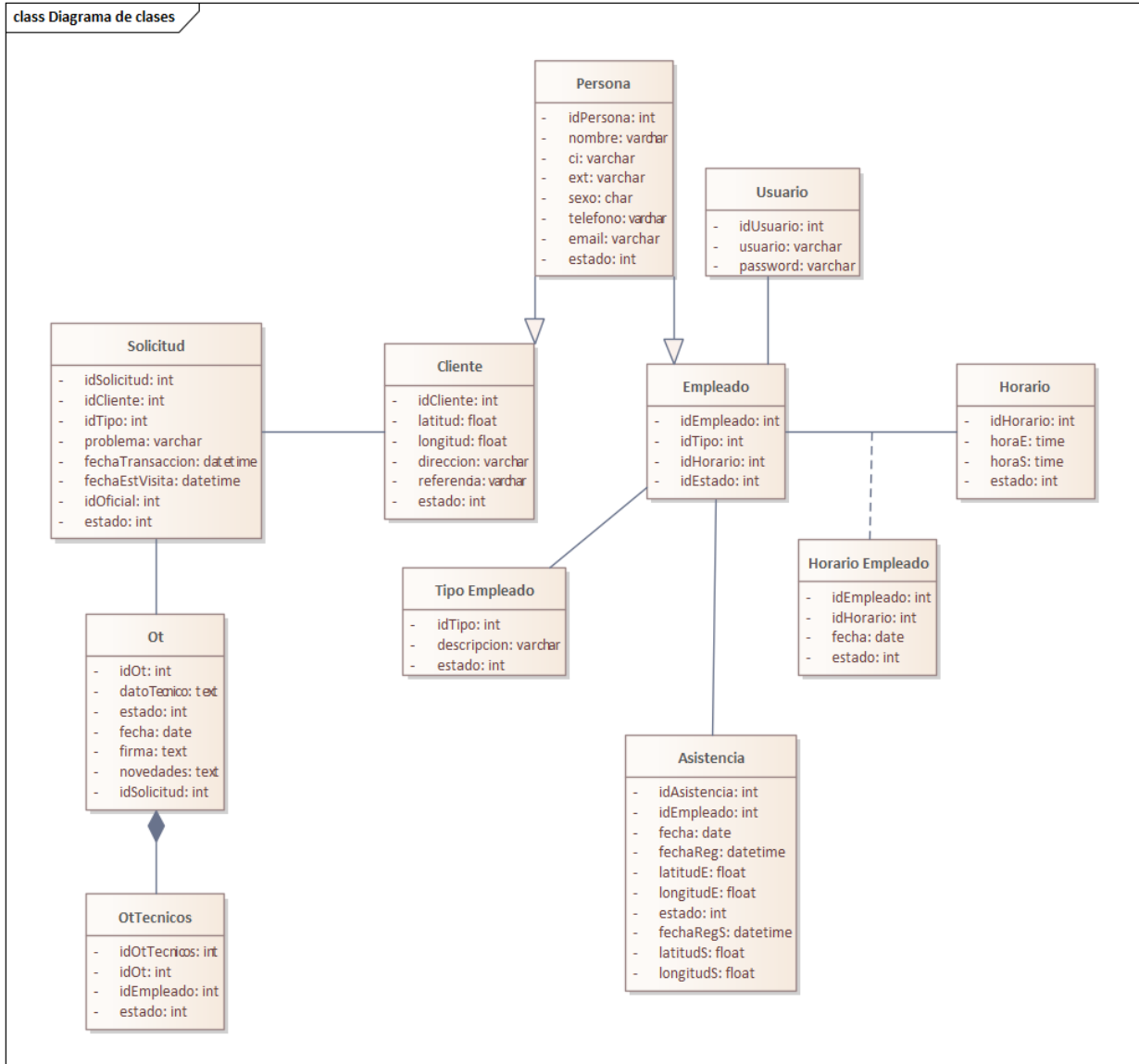
1. Diagrama general de casos de uso	3
2. Diagrama de clases	4
3. Diagrama de despliegue.....	5
4. Computación en la nube	5
4.1. Características	5
4.2. Métodos de entrega.....	6
4.2.1. Software como servicio	7
4.2.2. Plataforma como servicio.....	7
4.2.3. Infraestructura como servicio	7
4.2.4. Tipos de PaaS	7
5. Ventajas y desventajas de computación en la nube	8
5.1. Ventajas	8
5.2. Desventajas	9
6. Sistemas de gestión empresarial.....	9
6.1. Estructura de los sistemas de gestion empresarial	10
6.2. Funciones de un sistema de gestion empresarial.....	10
7. Como optimizar los servicios de ubicación para ahorrar batería	11
7.1. Quita actualizaciones de ubicación	11
7.2. Establece tiempos de espera	11
7.3. Actualizaciones de ubicación pasiva	11
8. Cuadro de comparaciones	11
8.1. PHP vs Python vs Node js	11
8.2. React Native vs Flutter vs Android vs Swift	13
8.3. SQL Server vs MySQL vs PostgreSQL	14

1. Diagrama general de casos de uso

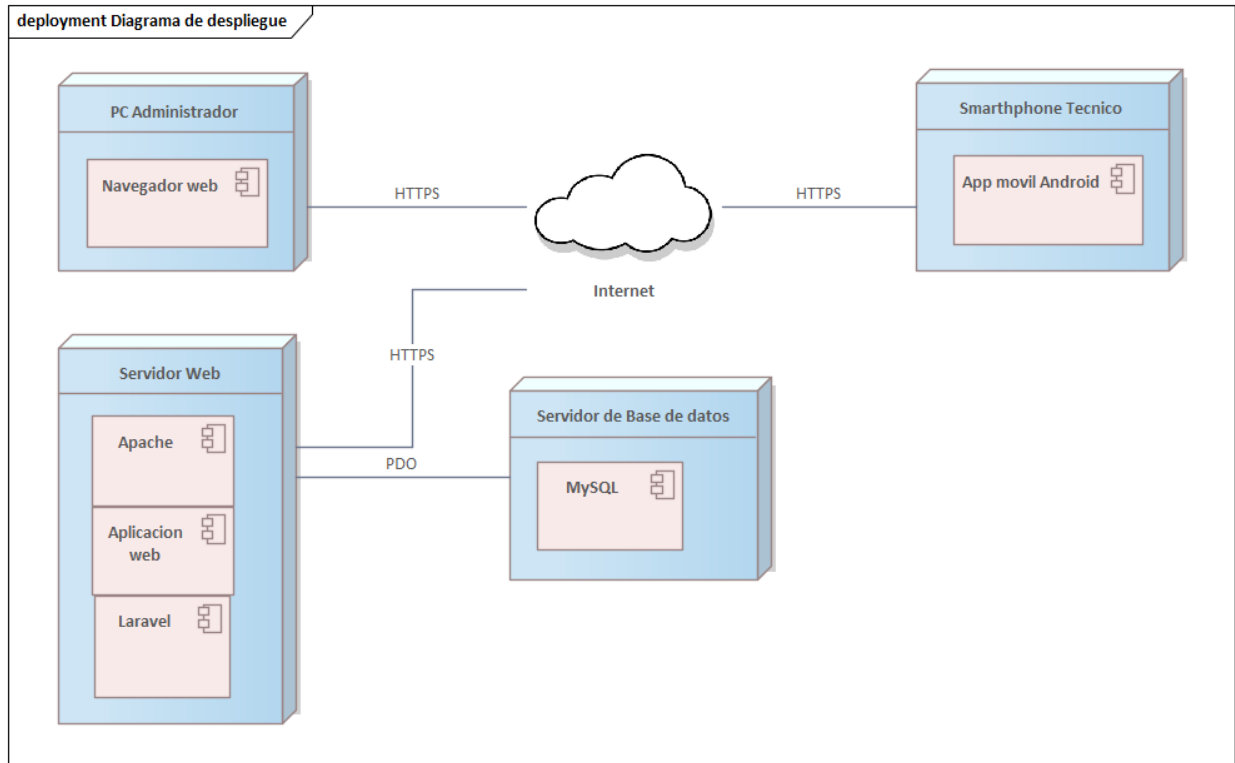
uc Diagrama de casos de uso general



2. Diagrama de clases



3. Diagrama de despliegue



4. Computación en la nube

conocida también como servicios en la nube, informática en la nube, nube de cómputo o simplemente «la nube», es un paradigma que permite ofrecer servicios de computación a través de una red, que usualmente es internet.

4.1. Características

Existen varias características de la computación en la nube:

- **Autoservicio bajo demanda:** un consumidor puede aprovisionar por sus propios medios capacidades informáticas, como cómputo, almacenamiento y redes, en forma automática sin requerir la interacción humana del proveedor de servicios.
- **Acceso amplio y ubicuo a toda la red:** todas las capacidades están disponibles a través de la red y se accede a ellas a través de mecanismos estándares y plataformas heterogéneas como, por ejemplo: teléfonos móviles, tabletas, computadoras portátiles y estaciones de trabajo.
- **Ubicación transparente y agrupación de recursos:** los recursos informáticos del proveedor de servicios se agrupan para brindar servicio a múltiples consumidores, con diferentes recursos físicos virtualizados que se asignan y reasignan dinámicamente de acuerdo con la demanda.
- **Rápida elasticidad:** los recursos se pueden aprovisionar y liberar rápidamente según la demanda. Para el consumidor, las capacidades disponibles para el aprovisionamiento a menudo parecen ser ilimitadas y pueden ser apropiadas en cualquier cantidad en cualquier momento.

- Servicio medido: quizás una de las características determinantes. Los sistemas en la nube tienen mecanismos de medición en alguno de los niveles de abstracción para el tipo de servicio (por ejemplo, almacenamiento, procesamiento, ancho de banda y cuentas de usuario activas).

En un sentido más ampliado y basándose en varias fuentes, la computación en la nube presenta las siguientes características clave:

- Autorreparable: en caso de desperfecto, los proveedores posibilitan procesos de respaldo de información, que consiste en crear una copia de seguridad automática de la última reservación de datos, por lo tanto, es prácticamente imposible que exista una pérdida de información.⁵
- Agilidad: capacidad de mejora para ofrecer recursos tecnológicos al usuario por parte del proveedor.
- Costo: los recursos en la nube suelen tener costos menores a los que un aprovisionamiento físico local podría representar. Generalmente, la inversión inicial que representaría tener un aprovisionamiento local se ve anulada debido a la naturaleza bajo demanda de la nube.
- Escalabilidad y elasticidad: aprovisionamiento de recursos sobre una base de autoservicio casi en tiempo real, sin que los usuarios necesiten cargas de alta duración.
- Independencia entre el dispositivo y la ubicación: permite a los usuarios acceder a los sistemas utilizando un navegador web, independientemente de su ubicación o del dispositivo que utilice (por ejemplo, PC, teléfono móvil).
- Virtualización: la tecnología de virtualización permite compartir servidores y dispositivos de almacenamiento y una mayor utilización.
- Disponibilidad de la información: el usuario no tendrá inconvenientes para acceder a la información, debido a que esta permanecerá en Internet y su acceso se permite desde cualquier dispositivo conectado en la red (previa autorización); de manera que no es necesario almacenar copias de documentos en dispositivos de almacenamiento físico.⁷
- Rendimiento: los sistemas en la nube controlan y optimizan el uso de los recursos de manera automática, dicha característica permite un seguimiento, control y notificación del mismo. Esta capacidad aporta transparencia tanto para el consumidor o el proveedor de servicio.
- Seguridad: puede mejorar debido a la centralización de los datos. La seguridad es a menudo tan buena o mejor que otros sistemas tradicionales, en parte porque los proveedores son capaces de dedicar recursos a la solución de los problemas de seguridad que muchos clientes no pueden permitirse el lujo de abordar. El usuario de la nube es responsable de la seguridad a nivel de aplicación. El proveedor de la nube es responsable de la seguridad física.⁸
- Mantenimiento: en el caso de las aplicaciones de computación en la nube, es más sencillo, ya que no necesitan ser instalados en el ordenador de cada usuario y se puede acceder desde diferentes lugares.

4.2. Métodos de entrega

La computación en la nube establece su arquitectura a partir de una fragmentación entre aplicación informática, plataforma y hardware, dando como resultado según NIST⁴ los siguientes métodos de entrega: software como servicio, plataforma como servicio e infraestructura como servicio

4.2.1. Software como servicio

El software como servicio se encuentra en la capa más alta y caracteriza una aplicación completa ofrecida como un servicio bajo demanda a través de multitenencia —una sola instancia del software que corre en la infraestructura del proveedor sirve a múltiples organizaciones de clientes—. Las aplicaciones que suministran este modelo de servicio son accesibles a través de un navegador web —o de cualquier aplicación diseñada para tal efecto— y el usuario no tiene control sobre ellas, aunque en algunos casos se le permite realizar algunas configuraciones. Esto le elimina la necesidad al cliente de instalar la aplicación en sus propios computadores, evitando asumir los costos de soporte y el mantenimiento de hardware y software.

4.2.2. Plataforma como servicio

La capa del medio, la plataforma como servicio, es la encapsulación de una abstracción de un ambiente de desarrollo y el empaquetamiento de una serie de módulos o complementos que proporcionan, normalmente, una funcionalidad horizontal (persistencia de datos, autenticación, mensajería, etc.). De esta forma, un arquetipo de plataforma como servicio podría consistir en un entorno conteniendo una pila básica de sistemas, componentes o APIs preconfiguradas y listas para integrarse sobre una tecnología concreta de desarrollo (por ejemplo, un sistema Linux, un servidor web, y un ambiente de programación como Perl o Ruby). Las ofertas de plataformas como servicio pueden servir a todas las fases del ciclo de desarrollo y pruebas del software, o pueden estar especializadas en cualquier área en particular, tal como la administración del contenido.

Ejemplos comerciales son Google App Engine, que sirve aplicaciones de la infraestructura Google; Microsoft Azure, una plataforma en la nube que permite el desarrollo y ejecución de aplicaciones codificadas en varios lenguajes y tecnologías como .NET, Java, Go y PHP o la Plataforma G, desarrollada en Perl. Servicios PaaS como estos permiten gran flexibilidad, pero puede ser restringida por las capacidades disponibles a través del proveedor. En este modelo de servicio al usuario se le ofrece la plataforma de desarrollo y las herramientas de programación por lo que puede desarrollar aplicaciones propias y controlar la aplicación, pero no controla la infraestructura.

4.2.3. Infraestructura como servicio

La infraestructura como servicio —también llamada en algunos casos hardware as a service, HaaS) se encuentra en la capa inferior y es un medio de entregar almacenamiento básico y capacidades de cómputo como servicios estandarizados en la red. Servidores, sistemas de almacenamiento, conexiones, enrutadores, y otros sistemas se concentran (por ejemplo, a través de la tecnología de virtualización) para manejar tipos específicos de cargas de trabajo —desde procesamiento en lotes (“batch”) hasta aumento de servidor/almacenamiento durante las cargas pico—. El ejemplo comercial mejor conocido es Amazon Web Services, cuyos servicios EC2 y S3 ofrecen cómputo y servicios de almacenamiento esenciales (respectivamente).

4.2.4. Tipos de PaaS

Públicos privados e híbridos

Existen varios tipos de plataformas como servicio, incluyendo públicos, privados e híbridos. PaaS fue originalmente pensado para las nubes públicas, antes de expandirse a las privadas e híbridas. Los PaaS públicos son derivados del software como servicio y está situado entre esta y la infraestructura como servicio.

Los PaaS privados son comúnmente descargados e instalados desde una infraestructura local de una empresa, o desde una nube pública.

Un PaaS híbrido es típicamente un despliegue consistente en una mezcla de despliegues públicos y privados.

5. Ventajas y desventajas de computación en la nube

5.1. Ventajas

Las principales ventajas de la computación en la nube son:

- Integración probada de servicios Red. Por su naturaleza, la tecnología de computación en nube se puede integrar con mucha mayor facilidad y rapidez con el resto de las aplicaciones empresariales (tanto software tradicional como Cloud Computing basado en infraestructuras), ya sean desarrolladas de manera interna o externa.
- Prestación de servicios a nivel mundial. Las infraestructuras de cloud computing proporcionan mayor capacidad de adaptación, recuperación completa de pérdida de datos (con copias de seguridad) y reducción al mínimo de los tiempos de inactividad.
- Una infraestructura 100 % de computación en la nube permite también al proveedor de contenidos o servicios en la nube prescindir de instalar cualquier tipo de software, ya que este es provisto por el proveedor de la infraestructura o la plataforma en la nube. Un gran beneficio de esta computación es la simplicidad y el hecho de que requiera mucha menor inversión para empezar a trabajar.
- Portabilidad de la información: en sus inicios, los proveedores de almacenamiento en la nube canalizaban sus servicios a usuarios corporativos, sin embargo, actualmente los usuarios particulares se han familiarizado con la computación en la nube y son quienes los utilizan con mayor frecuencia en teléfonos inteligentes y tabletas, entre otros.
- Implementación más rápida y con menos riesgos, ya que se comienza a trabajar más rápido y no es necesaria una gran inversión. Las aplicaciones de la computación en la nube suelen estar disponibles en cuestión de días u horas en lugar de semanas o meses, incluso con un nivel considerable de personalización o integración.
- Actualizaciones automáticas que no afectan negativamente a los recursos de TI. Al actualizar a la última versión de las aplicaciones, el usuario se ve obligado a dedicar tiempo y recursos para volver a personalizar e integrar la aplicación. Con la computación en la nube no hay que decidir entre actualizar y conservar el trabajo, dado que esas personalizaciones e integraciones se conservan automáticamente durante la actualización.
- Contribuye al uso eficiente de la energía. En este caso, a la energía requerida para el funcionamiento de la infraestructura. En los centros de datos tradicionales, los servidores consumen mucha más energía de la requerida realmente. En cambio, en las nubes, la energía consumida es solo la necesaria, reduciendo notablemente el desperdicio.

5.2. Desventajas

Las desventajas de la computación en la nube, generalmente, se encuentran relacionadas con la confiabilidad, disponibilidad de los servicios, acceso, seguridad y, sobre todo, con el almacenamiento. A continuación se detallan:

- La centralización de las aplicaciones y el almacenamiento de los datos origina una interdependencia de los proveedores de servicios.
- La disponibilidad de las aplicaciones está sujeta a la disponibilidad de acceso a Internet.
- La confiabilidad de los servicios depende de la "salud" tecnológica y financiera de los proveedores de servicios en nube. Empresas emergentes o alianzas entre empresas podrían crear un ambiente propicio para el monopolio y el crecimiento exagerado en los servicios.
- La disponibilidad de servicios altamente especializados podría tardar meses o incluso años para que sean factibles de ser desplegados en la red.
- La madurez funcional de las aplicaciones hace que continuamente estén modificando sus interfaces, por lo cual la curva de aprendizaje en empresas de orientación no tecnológica tenga unas pendientes significativas, así como su consumo automático por aplicaciones.
- Seguridad. La información de la empresa debe recorrer diferentes modos para llegar a su destino, cada uno de ellos (y sus canales) son un foco de inseguridad. Si se utilizan protocolos seguros, HTTPS por ejemplo, la velocidad total disminuye debido a la sobrecarga que estos requieren.
- Escalabilidad a largo plazo. A medida que más usuarios empiecen a compartir la infraestructura de la nube, la sobrecarga en los servidores de los proveedores aumentará, si la empresa no posee un esquema de crecimiento óptimo puede llevar a degradaciones en el servicio o altos niveles de fluctuación de retardo (jitter).

6. Sistemas de gestión empresarial

Los sistemas de gestión empresarial son el conjunto de políticas, prácticas, procedimientos y procesos utilizados en la elaboración e implementación de estrategias, su ejecución, y toda la actividad de gestión asociada. Las empresas utilizan actualmente sistemas de información a todos los niveles de operación para recoger, procesar y almacenar datos, optimizando las operaciones diarias del negocio.

Un sistema de gestión empresarial automatizado, al igual que uno manual, también basa su funcionamiento en datos.

Los recoge, los almacena, los procesa y los analiza. Sus ventajas son:

- Hacer que sea posible almacenar inmensas cantidades de información.
- Evitar los errores típicos de los registros manuales.
- Multiplicar la profundidad del análisis, mediante cálculos y comparaciones que serían prácticamente imposibles en un sistema manual.

La tecnología juega un importante papel en la gestión de las empresas hoy día y este tipo de sistemas son una buena prueba de ello; en especial desde la explosión big data, que ha puesto de manifiesto la necesidad de trabajar con grandes volúmenes de información para cualquier tipo de negocio, independientemente de su sector de actividad, tamaño o características individuales.

6.1. Estructura de los sistemas de gestión empresarial

El objetivo de los sistemas de gestión empresarial es proporcionar información de calidad a los usuarios en forma de informes y simulaciones de modelos matemáticos que les ayudan a predecir el futuro y decidir en consecuencia.

Para lograr sus metas, los sistemas de gestión empresarial automatizados deberían presentar las siguientes características:

- Orientación al negocio.
- Integración con los sistemas existentes.
- Flexibilidad y facilidad de uso.

La estructura de un sistema de este tipo, si se analiza en base a sus componentes físicos, se construiría en torno a seis elementos:

- Hardware: desde los equipos informáticos hasta cualquiera de sus accesorios, como impresoras o escáneres.
- Software: los sistemas y todas sus aplicaciones.
- Base de datos: data warehouse y repositorios.
- Procedimientos: los manuales son un ejemplo clásico.
- Usuarios: como los responsables de cuentas, analistas de negocio o los empleados del equipo de ventas.
- Productos: reporting o presupuestación, entre otros.

6.2. Funciones de un sistema de gestión empresarial

Las funciones de los sistemas de gestión empresarial son muy diversas, pero siempre han de estar correctamente contextualizadas para permitir:

- Gestionar las transacciones.
- Elaborar informes.
- Procesar consultas.

Sin embargo, una de sus funciones más importantes es el soporte a la toma de decisiones, especialmente cuando la decisión no es de rutina, por lo que el responsable de definir la iniciativa a aplicar debe proporcionar el juicio, la evaluación y conocimientos sobre la definición del problema. En esta labor, la aportación de los sistemas de gestión empresarial es insustituible ya que, sin ellos no sería posible evaluar todas las variables que pueden afectar a la resolución de un problema o tomar en

consideración todos los datos que se deben tener en cuenta para una visión completa y de calidad de la realidad.

7. Como optimizar los servicios de ubicación para ahorrar batería

Para optimizar el consumo de batería por GPS, existen practicas recomendadas para la ubicación como:

7.1. Quita actualizaciones de ubicación

Una fuente común de agotamiento innecesario de la batería es la falta de eliminación de las actualizaciones de ubicación cuando ya no son necesarias.

Se puede utilizar componentes que prioricen los ciclos de vida para administrar mejor el ciclo de vida de las actividades de una aplicación.

7.2. Establece tiempos de espera

Para protegerse contra el agotamiento de la batería, establece un tiempo de espera razonable cuando deban detenerse las actualizaciones de ubicación.

El tiempo de espera garantiza que las actualizaciones no continúen indefinidamente y protege la app en situaciones donde se solicitan actualizaciones

7.3. Actualizaciones de ubicación pasiva

En casos de uso en segundo plano, es una buena idea limitar las actualizaciones de ubicación.

Es probable que mientras tu app esté en segundo plano, otra aplicación solicite con frecuencia actualizaciones de ubicación en primer plano. Los servicios de ubicación ponen estas actualizaciones a disposición de tu app.

8. Cuadro de comparaciones

8.1. PHP vs Python vs Node js

	PHP	PHYTON	NODE JS
Acerca del lenguaje	Creado en 1994 por Rasmus Lerdorf Diseñado exclusivamente para desarrollo web Desarrollo rápido, escritura dinámica	Creado en 1991 por Guido Van Rossum Diseñado como un lenguaje de uso general Código abierto, Easy Deployment, mejoras continuas	Creado en 2009 por Ryan Dahl Diseñado con la idea de poder ejecutar JavaScript fuera del entorno del navegador Entorno en tiempo de ejecución
Pros	Funciona muy bien para el desarrollo web	Funciona bien para variedad de propósitos:	Usa JavaScript el cual es fácil de aprender

	<p>Soporta de forma nativa la mayoría de los softwares de servidores web</p> <p>Sintaxis derivada de C</p> <p>Amplias librerías y frameworks para desarrollo web</p>	<p>robótica, backend, gráficos, etc.</p> <p>preinstalado en la mayoría de las distribuciones GNU/LINUX</p> <p>Fácil de aprender</p> <p>Diversidad en cuestión de funcionalidades y usos</p>	<p>Es un tipo de servidor más veloz que otros</p> <p>Fácilmente escalable</p> <p>Comparte el mismo código tanto en el lado del cliente como el servidor</p> <p>Modulo de almacenamiento en cache</p>
Contras	<p>Puede ser difícil hacer debug</p>	<p>A menudo muy explicito e inelegante de leer</p>	<p>Es una tecnología muy joven</p> <p>Si no se cuida el código, puede ser complicado depurar a posteriori</p>
Porque elegir este lenguaje	<p>Se ejecuta en varias plataformas: Windows, Unix, Linux, Mac Os, etc.</p> <p>Compatible con Apache, IIS</p> <p>Se ejecuta eficientemente en el lado del servidor</p> <p>Gratis</p>	<p>Fácil de aprender y depurar</p> <p>Tiene muchas bibliotecas precompiladas que hacen el desarrollo más fácil</p> <p>Shell interactivo</p> <p>Admite aplicaciones GUI</p>	<p>Permite una escalabilidad alta</p> <p>Tiene buen rendimiento</p> <p>Ideal para aplicaciones de alto trafico</p> <p>Es la mejor opción del mercado para aplicaciones en tiempo real como chats online</p>
Diferencia Clave	<p>PHP es utilizado para desarrollar un sitio web dinámico o una aplicación web</p> <p>PHP es el lenguaje de elección para el desarrollo web</p> <p>PHP tiene muchos marcos</p> <p>PHP tiene una amplia gama de convenciones de nomenclatura y sintaxis</p> <p>Código abierto, fácil implementación, mejoras continuas</p>	<p>Python es un lenguaje de programación orientado a objetos de alto nivel utilizado para el desarrollo rápido de aplicaciones</p> <p>Python es ampliamente utilizado en la IA, la ciencia de datos y la comunidad científica</p> <p>Python tiene un pequeño número de marcos</p> <p>Python ofrece una sintaxis muy clara y concisa de códigos</p> <p>Desarrollo rápido, escritura dinámica y código fácil de entender</p>	<p>Nodejs brilla en aplicaciones web de tiempo real empleando la tecnología push a través de Websockets</p> <p>Tiene uno de los mejores rendimientos</p> <p>El código funciona tanto para backend como frontend</p> <p>Nodejs es perfecto para microservicios el cual es una solución popular en las aplicaciones empresariales</p>

8.2. React Native vs Flutter vs Android vs Swift

	React Native	Flutter	Android	Swift
Lenguaje de programacion	Java Script	Dart	Java, Kotlin	Swift
Biblioteca de componentes	Amplia variedad de componentes	Menor que React Native, pero va creciendo		
Curva de aprendizaje	Si ya conoces JavaScript React Native es fácil de aprender	Curva ligeramente inclinada si no sabes programar en Dart y no recuerdas sobre clases en C	Fácil de aprender, sabiendo java o kotlin	Su curva de aprendizaje es menor a otros lenguajes
Creador por	Facebook	Google	Google	Apple
Arquitectura principal	Flux y Redux	BLoc	Linux	LLVM
Pros	Ofrece varios elementos listos para aplicar Podemos recargar nuestra aplicación sin volver a compilar Uso directo de código nativo para optimizar la aplicación Codigo compartido en ambas plataformas, Android y iOS Desarrollo más rápido gracias a elementos pre-fabricados Accesos a funcionalidades nativas como cámara, acelerómetro UI móvil de alta calidad	Escritura de código más rápida y ayuda a los equipos a agregar funciones, corregir errores y experimentar más rápido Los desarrolladores solo tienen que escribir una base de código que cubre Android y iOS Es sencillo crear, personalizar widgets Ideal para MVC, ayudándonos a construir en menos tiempo	Compilación rápida Ejecución de la app en tiempo real gracias al emulador Ejecución de la app directamente desde el móvil Inteligente gestión de memoria Sistema operativo más demandado Sistema de código abierto	Alta compatibilidad Mayor simplicidad Mejor rendimiento Mayor seguridad Lenguaje amigable
Contras	La navegación de la aplicación no sería equivalente	No es compatible con plataformas de integración	Los requisitos son un poco elevados, tendrás que tener	Algunos programadores tienen problemas

	a la navegación nativa Las aplicaciones creadas con React Native son mas lentas que las aplicaciones nativas de Android con Kotlin o iOS con Swift	continua como TRAVIS o JENKINS Aun no es tan rico en desarrollo nativo	una buena PC para el desarrollo	al migrar de Objective C a Swift
--	-------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------	---------------------------------	----------------------------------

8.3. SQL Server vs MySQL vs PostgreSQL

	Microsoft SQL Server	Mysql	Postgresql
Descripción	DBMS relacionales de Microsoft	RDBMS de código abierto ampliamente utilizado	RDBMS de código abierto ampliamente utilizado
Modelo de base de datos principal	DBMS relacional	DBMS relacional	DBMS relacional
Desarrollador	Microsoft	Oracle	PostgreSQL Global Development Group
Lanzamiento inicial	1989	1995	1989
Licencia	Comercial	Código abierto	Código abierto
Ofertas DBaaS (enlaces patrocinados)		ScaleGrid para MySQL:alojamiento MySQL totalmente administrado en AWS, Azure y DigitalOcean con alta disponibilidad y acceso SSH en la #1 DBaaS multinube.	ScaleGrid para PostgreSQL:alojamiento de PostgreSQL totalmente administrado en AWS, Azure y DigitalOcean con alta disponibilidad y acceso SSH en la #1 DBaaS multinube.
Lenguaje de implementación	C++	C y C++	C
Sistemas operativos de servidor	Linux Windows	FreeBSD Linux OS X Solaris Windows	FreeBSD HP-UX Linux NetBSD OpenBSD OS X Solaris Unix Windows
Soporte XML	Sí	Sí	Sí
Sql	Sí	Sí	Sí

API y otros métodos de acceso	ADO.NET JDBC ODBC OLE DB Tabular Data Stream (TDS)	ADO.NET API nativa propietaria de ODBC JDBC	ADO.NET API de streaming ODBC de biblioteca nativa JDBC para objetos grandes
Lenguajes de programación compatibles	JavaScript JavaScript (nodo.js) PHP Python R Ruby Visual Basic	Ada C C# C++ D Delphi Eiffel Erlang Haskell Java JavaScript (Node.js) Objective-C OCaml Perl PHP Python Ruby Scheme Tcl	Java de .Net C C++ Delphi JavaScript (nodo.js) Perl PHP Python Tcl
Scripts del lado del servidor	Transact SQL, lenguajes .NET, R, Python y (con SQL Server 2019) Java	Sí	funciones definidas por el usuario
Triggers	Sí	Sí	Sí
Métodos de particionamiento	tablas se pueden distribuir entre varios archivos (partición horizontal); particionamiento a través de la federación	partición horizontal, particionamiento con MySQL Cluster o MySQL Fabric	particionamiento por rango, lista y (desde PostgreSQL 11) por hash
Conceptos de transacción	Ácid	Ácid	Ácid
Concurrencia	Sí	Sí	Sí
Durabilidad	Sí	Sí	Sí
Capacidades en memoria	Sí	Sí	No