

TAREAS MOVILES 

**INVESTIGACION**

**TAREA COMPLETA**

**Asignatura:**

Programación Visual

**Profesor:**

Torres Servín Emmanuel

**Alumnos:**

1. González Clímaco Bruno
2. Joaquín Espejel Jesús Elías
3. Juárez Galicia Karim Izahi

**Matriculas:**

1. 1321124272
2. 1321124258
3. 1321124269

**Grupo:**

4322IS

**Carrera:**

Ingeniería en Software

**INTRODUCCIÓN**

**En este documento se abarca el tema de Dispositivos móviles, en donde en esta primera tarea se encuentran temas como:**

* Tolerancia de fallos
* Persistencia de datos
* Manipulación de datos
* Modelado de objetos}

En cada punto se encuentran cosas útiles como:

* Características
* Conceptos
* Ventajas
* Desventajas
* Tipos
* Elementos
* Procesos
* Explicaciones
* Etc.

**Todo esto con el fin de obtener información útil que nos servirá en el desarrollo de programas para dispositivos móviles.**

**Tarea 1:** Realiza una investigación tomando en cuenta los siguientes temas y puntos.

* **Modelado de objetos de acceso a datos en dispositivos móviles Modelado de objetos en dispositivos móviles.**

**Características de los objetos de acceso a datos en dispositivos móviles.**

**Por lo general, las características esenciales de estos dispositivos son:**

**\***Son aparatos pequeños: la mayoría se pueden transportar en el bolsillo del propietario o en un pequeño bolso.

**\***Tienen capacidad de procesamiento.

**\***Tienen conexión permanente o intermitente a una red.

**\***Tienen memoria (RAM, tarjetas MicroSD, flash, etc.).

**\***Normalmente, se asocian al uso individual de una persona, tanto en posesión como en operación, la cual generalmente puede adaptarlos a su gusto.

**\***Tienen una alta capacidad de interacción mediante la pantalla o el teclado.

**\***Por lo tanto, en la mayoría de casos, un dispositivo móvil puede definirse con cuatro características que lo diferencian de otros dispositivos que, aunque pudieran parecer similares, carecen de algunas de las características de los verdaderos dispositivos móviles.

**Proceso de modelado de objetos de acceso a datos en dispositivos móviles.**

Es un componente de software que suministra una interfaz común entre la aplicación y uno o más dispositivos de almacenamiento de datos, tales como una Base de datos o un archivo. El término se aplica frecuentemente al Patrón de diseño Object.

Los Objetos de Acceso a Datos son un Patrón de Diseño Core J2EE y considerados una buena práctica. La ventaja de usar objetos de acceso a datos es que cualquier objeto de negocio (aquel que contiene detalles específicos de operación o aplicación) no requiere conocimiento directo del destino final de la información que manipula.

**Proceso de programación de objetos de acceso a datos en dispositivos móviles.**

Un Objeto de Acceso a Datos o Data Access Object (DAO) son una serie de objetos que le permiten tener acceso y manipular datos mediante programación en bases de datos locales o remotos. Puede utilizar DAO para administrar bases de datos, así como sus objetos y su estructura.

* **Manipulación de datos en dispositivos móviles**

**Reconocer el concepto de conexión a bases de datos.**

Una conexión de base de datos le permite trabajar con tablas de bases de datos directamente en Insights. Se pueden crear conexiones de base de datos para las bases de datos compatibles y bases de datos relacionales adicionales que utilizan el controlador Java Database Connectivity (JDBC).

**Describir la conexión a bases de datos estáticos, dinámicos, web y locales en dispositivos móviles.**

* Estáticos:

Una base de datos estática es aquella cuya función principal es el almacenamiento y registro de datos fijos. Es decir, guarda información que no se va a modificar ni editar con el tiempo. Se trata de un tipo de bases de datos de solo lectura.

* Dinámicos:

Una base de datos dinámica es aquella en la que se almacenan datos que pueden variar con el paso del tiempo. Para adecuarse a estos datos cambiantes, las bases dinámicas permiten realizar operaciones de edición, actualización o borrado de información.

* Web:

Base de Datos Web es una herramienta que organiza y administra la información de forma sencilla a través de una interfaz Web.

* Locales:

Una base de datos en local utiliza una red local (LAN), de manera que la infraestructura y la gestión de dicha base de datos se realiza en la propia organización. Solo pueden acceder a la información los equipos que estén conectados a la red local.

**Explicar el proceso de programación de conexión a bases de datos estáticos en dispositivos móviles.**

Una base de datos estática es aquella cuya función principal es el almacenamiento y registro de datos fijos. Es decir, guarda información que no se va a modificar ni editar con el tiempo.

Se trata de un tipo de bases de datos de solo lectura. Su implementación se suele realizar con el objetivo de registrar datos históricos para poder comparar su evolución a lo largo del tiempo.

**Explicar el proceso de programación de conexión a bases de datos dinámicos en dispositivos móviles.**

La estructura de una base de datos dinámica puede ser la misma que la de cualquier base de datos relacional. Es decir, la información se organiza en tablas que a su vez contienen diversas filas y columnas. Las filas corresponden a cada uno de los registros de las bases de datos, mientras que cada columna corresponde a un campo o atributo de cada registro.

La única característica especial es que en las bases de datos dinámicas los campos deben ser editables, ya sea manualmente o a través de formularios, para poder mantener la información siempre actualizada.

**Explicar el proceso de programación de conexión a bases de datos locales en dispositivos móviles.**

La base de datos es la forma más común de almacenar y administrar datos. Desde hace bastante tiempo, las bases de datos se manejan en el lado del servidor o en la nube y los dispositivos móviles solo se comunican con ellos a través de la red.

Sin embargo, para hacer que las aplicaciones sean más receptivas y menos dependientes de la conectividad de la red, la tendencia del uso fuera de línea o la menor dependencia de la red está ganando popularidad.

* **Persistencia de datos en los dispositivos móviles.**

**Concepto de persistencia en dispositivos móviles.**

El caso más típico es cuando necesitamos almacenar un conjunto de pares "clave/valor", para los que podemos usar property lists o bien el sistema de preferencias de iOS. Este es el mecanismo habitual para guardar las preferencias de una aplicación (colores, tipos de letra, datos básicos del usuario, etc.).

Si necesitamos almacenar objetos de clases propias cualesquiera podemos archivarlas, que sería lo que en otros lenguajes se denomina serializar. Acabaremos el capítulo viendo cómo se pueden archivar objetos en iOS.

**Retos de la persistencia en los dispositivos móviles.**

En la mayoría de las aplicaciones podemos configurar una serie de parámetros que las adaptan a las preferencias o necesidades del usuario. Son elementos tales como colores, tipos de fuentes, unidades de medida, nombres de usuarios y contraseñas de servicios de terceros, etc.

iOS nos ofrece un mecanismo estándar para almacenar estas preferencias de modo permanente. El API es bastante sencillo y nos permite establecer unos valores por defecto, modificarlos según lo que indique el usuario y leer los valores previamente fijados.

Hay que destacar que iOS solo nos da el API para leer/almacenar las preferencias, pero no el interfaz de usuario para mostrarlas/modificarlas, que es nuestra responsabilidad. Hay una excepción: si queremos que nuestras preferencias aparezcan dentro de las del sistema, el propio iOS se encargará de la interfaz.

**Formas de persistencia en los sistemas operativos de los dispositivos móviles: preferencias, almacenamiento de archivos, datos estructurados.**

* Preferencias:

Aunque hay algunos más, los tres sistemas operativos mayoritarios son Android, iOS de Apple, y Windows Phone.

* Almacenamiento de Archivos:

La mayoría de los archivos del sistema operativo “Windows” se almacena en la carpeta “C:\Windows”, especialmente en tales subcarpetas como “/System32” y “/SysWOW64”.

* Datos estructurados:

El sistema operativo se divide en partes o niveles, cada uno perfectamente definido y con un claro interface (comunicación) con el resto de los elementos. De esta forma, las zonas más internas del sistema operativo o núcleo del sistema estarán más protegidas de accesos indeseados desde las capas más externas.

**Tipos de persistencia: local, remota y de Cacheo/Hoarding en dispositivos móviles.**

* Local:

En programación, la persistencia es la acción de preservar la información de un objeto de forma permanente (guardado), pero a su vez también se refiere a poder recuperar la información del mismo (leerlo) para que pueda ser nuevamente utilizado.

* Remota:

Para conectarse a un servidor por SSH, se necesitan los siguientes datos:

* Puerto (normalmente 22, pero en algunos casos puede ser diferente)
* Dirección IP
* Usuario
* Archivo de par de claves SSH

Además, se hará uso de una librería JSCH (Java Secure Channel)

* Cacheo/Hoarding:

Para que nuestra aplicación pueda utilizar esta librería, el archivo oscache.jar debe estar en la carpeta WEB-INF/lib de la aplicación web. Toda la configuración de OSCache recae en el archivo oscache. properties el cual debe encontrarse en el classpath de la aplicación.

El archivo de muestra que obtenemos al descargar la librería es bastante autoexplicatorio (dentro de la carpeta etc.).

**Proceso de programación de persistencia en dispositivos móviles.**

La persistencia de datos es un medio mediante el cual una aplicación puede recuperar información desde un sistema de almacenamiento no volátil y hacer que esta persista. La persistencia de datos es vital en las aplicaciones empresariales debido al acceso necesario a las bases de datos relacionales.

* **Mecanismos de tolerancia a fallos**

**Elementos para tomar en cuenta en el desarrollo de aplicaciones orientadas a móviles:**

* + - Lentitud en las consultas:

Haga clic con el botón derecho en Consultas de consumo de recursos importantes y principales y seleccione “Ver consultas de consumo de recursos principales”, esta acción lo direccionara a la ventana en la que se muestran estas consultas que consumen muchos recursos.

* + - Alto consumo de batería:

Para controlar las aplicaciones que permanecen inactivas o activas, debemos entrar en el apartado "Ajustes" del teléfono y buscar "Mantenimiento del dispositivo" para luego pulsar en "Batería".

Una vez en batería, debemos entrar en el "Modo ahorro de batería" donde podremos ver las apps que permanecen activas e inactivas y añadir a cada listado la que más nos interese.

* + - Dificultad de sincronización:
* Compruebe que su dispositivo está conectado a Internet
* Vuelva a enviar un correo electrónico a través de la web
* Vuelva a enviar un correo electrónico desde otra aplicación o dispositivo
* Compruebe con su proveedor de correo electrónico el estado de su servidor de correo electrónico
  + - Modo off-line:

Una de las maneras en que las empresas pueden aprovechar las ventajas que la tecnología nos ofrece, es mediante el uso de aplicaciones móviles (utilizando las plataformas como iOS y Android), ya que mediante el uso de las apps móviles denominadas como de productividad, han ayudado a generar ventajas competitivas con las cuales las empresas han podido mejorar en cuestión de productividad y rentabilidad.

Es por ello que las empresas se han dado a la tarea de implementar y/o adoptar nuevos sistemas que les permitan agilizar y automatizar muchos de sus procesos, pero hacer que sus equipos de trabajo estén siempre de manera online puede llegar a representar un costo elevado.

* + - Recuperación de conexión:

Un pool de conexiones es un conjunto limitado de conexiones a una base de datos, que es manejado por un servidor de aplicaciones de forma tal, que dichas conexiones pueden ser reutilizadas por los diferentes usuarios. Este pool es administrado por un servidor de aplicaciones que va asignando las conexiones a medida que los clientes van solicitando consultas o actualizaciones de datos.

**Proceso de selección de los mecanismos de tolerancia a fallos en el desarrollo de aplicaciones de dispositivos móviles.**

La disponibilidad se puede definir como la propiedad de que un sistema esté listo para ser utilizado de inmediato, que el sistema este operando correctamente en cualquier momento dado y se encuentre disponible para realizar sus funciones. En otros términos, es que, muy probablemente funcionará en un instante dado.

Confiabilidad se refiere a la propiedad de que un sistema sea capaz de funcionar de manera continua sin fallar. En otros términos, es un sistema que muy probablemente continuará funcionando sin interrupción durante un lapso de tiempo relativamente largo. Por contraste con la disponibilidad, la confiabilidad se define en función de un intervalo de tiempo en lugar de un instante en el tiempo.

**Proceso de programación de los mecanismos de tolerancia a fallos en el desarrollo de aplicaciones de dispositivos móviles.**

La redundancia es la técnica fundamental requerida para lograr la tolerancia a fallas. Cuando se aplica a procesos, la noción de grupos de procesos se torna importante. Un grupo de procesos se compone de varios procesos que cooperan estrechamente para proporcionar servicios.

**CONCLUCIÓN**

En este documento se abarcaron temas interesantes que podrán servirnos para el desarrollo de nuevos programas especialmente centrados en los Dispositivos Móviles, entre algunos de estos temas encontramos cosas como:

* Modelado de objetos
* Objetos de acceso
* Conexión de bases de datos
* Bases de datos estáticos, dinámicos, web, locales
* Formas de persistencia
* Tolerancia de fallos
* Etc.

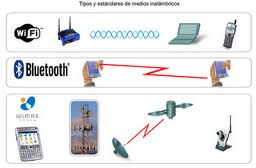
En algún determinado momento esta información podría sernos de utilidad, pues se abarcaron temas complejos eh importantes para la programación.

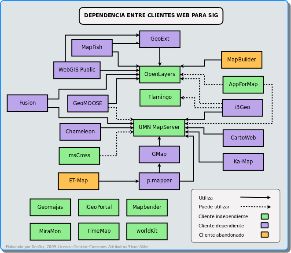
**MENTAL DEL TEMA CLIENTES LIGEROS**

**Características de los clientes ligeros en aplicaciones orientadas a móviles.**

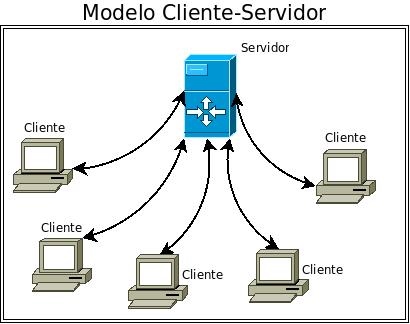
**cliente y de servidor.**

**Clientes Ligeros**

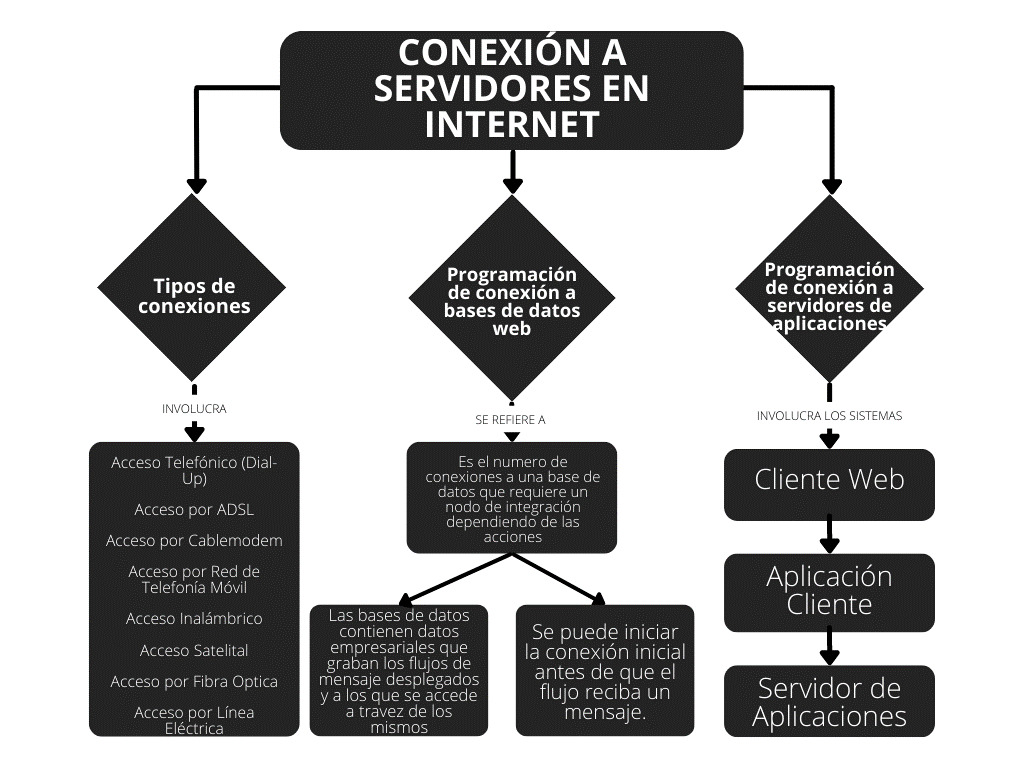




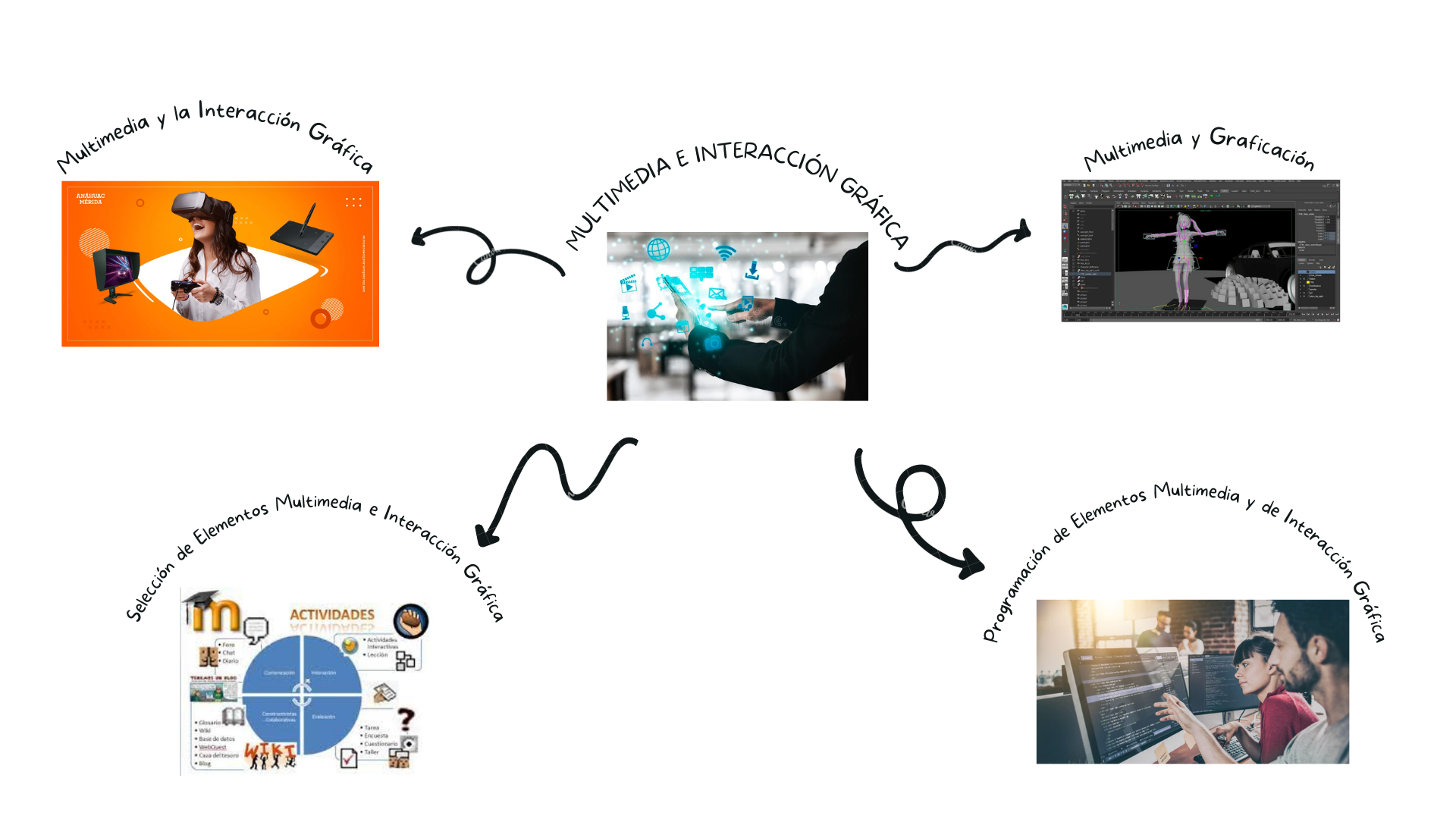
**Proceso de programación**



**MAPA CONCEPTUAL DEL TEMA CONEXIÓN A SERVIDORES EN INTERNET**



**MAPA CONCEPTUAL DEL TEMA CONEXIÓN A SERVIDORES EN INTERNET**



**INTRODUCCIÓN**

Este trabajo trata sobre las diversas características usos, elementos y procesos de los dispositivos y aplicaciones móviles, con el fin de darle conocer al lector sobre los puntos antes mencionados, logrando que el lector este informado sobre este tema.

Al realizar esta investigación se encontraron dificultades, al encontrar ciertos temas y ejemplos, por lo que tuvimos que hacer una investigación exhaustiva, lo que genero encontrar la información correcta.

Este documento fue completado gracias a la cooperación de nuestros compañeros, los que ayudaron a buscar información para completar esta investigación.

**Algunos de los puntos que se encontraran en esta investigación son los siguientes:**

• Características y elementos de las aplicaciones dependientes de contexto en los móviles.

• Usos de la geolocalización en las aplicaciones de los móviles.

• Proceso de selección de los elementos que permitan aplicaciones dependientes de

contexto.

• Proceso de programación de aplicaciones dependientes de contexto en los móviles.

Sensores embebidos en los dispositivos móviles

• Elementos y características de los sensores embebidos en los dispositivos móviles.

• Proceso de programación de aplicaciones de acuerdo con sensores embebidos de móviles.

**Realiza una investigación tomando en cuenta los siguientes temas y puntos.**

**Aplicaciones móviles dependientes del contexto**

**• Características y elementos de las aplicaciones dependientes de contexto en los móviles.**

Cuando hablamos de este concepto sobre el escenario de las Apps móviles nos referimos al conjunto de circunstancias que nos rodean (clientes, tecnología, necesidades, objetivos, etc.) y sin las que no se puede dar una situación determinada con éxito. En las aplicaciones corporativas entendemos este dato como parte del desarrollo.

Por tanto, el contexto de las aplicaciones implica tener en cuenta el tipo de dispositivo desde el que se va a utilizar, las funcionalidades, la movilidad… Pensamos en ello desde el principio porque, en función de estas características, nuestra aplicación será de una forma u otra. Eso sí, siempre buscando su adaptación a la línea del negocio y a la del cliente, sin perder de vista los objetivos corporativos y la filosofía empresarial.

Las aplicaciones ya forman parte de nuestro día a día, aunque solo sea para enviar un mensaje vía WhatsApp. Con ellas, también ha cambiado el contexto, pues las Apps de hace 5 años, incluso menos, se unían a los primeros desarrollos, de manear que la novedad del canal se convertía en una barrera.

Gracias a sus funciones, los usuarios han ido tomando conciencia de las ventajas que aportan de cara a su comodidad, a sus necesidades y a su contacto con las empresas. Sin embargo, cuando se crea una aplicación, no se hace pensando en un público general, sino en su público objetivo y potencial.

• **Usos de la geolocalización en las aplicaciones de los móviles.**

La geolocalización básicamente consiste en conocer la posición geográfica o ubicación (o coordenadas) de algún objeto: sea un teléfono, tablet, computadora portátil, auto, etc…a través de coordenadas que generalmente vienen de satélites, aunque también se pueden obtener por otros medios.

Hoy en día, la mayoría de las personas poseen al menos un dispositivo que proporciona esta información: el teléfono móvil; ya que cualquier tipo de teléfono móvil, sea o no un smartphone, puede ser localizado, ya que existen varias maneras de ser geolocalizados.

Ahora bien, la geolocalización a través de GPS o Sistema de Posicionamiento Global (Global Positioning System) no es la única que existe, también tenemos la geolocalización a través de GSM o Sistema Global para las Comunicaciones Móviles (Global System for Mobile Communications) y la geolocalización por Wi-Fi (Wireless Fidelity).

Un ejemplo muy claro y sencillo de la geolocalización en nuestra vida cotidiana es cuando utilizamos los datos de nuestra ubicación desde nuestro smartphone y puedes encontrar comercios cercanos, restaurantes, cines, gasolineras, cafeterías, etc.

aplicaciones como Google Maps, Apple Maps, Wazeo Google Earth, ¿no?, pues justamente estas apps están basadas en servicios de geolocalización y además llevan años siendo de grandísima utilidad para millones de usuarios al rededor del mundo, ofreciendo información de comercios cercanos, sobre el tráfico de la ciudad, las mejores rutas y los distintos transportes públicos para llegar, etc.

Sin lugar a duda, la geolocalizaciónes una tecnología que nos permite mejorar muchos aspectos de las actividades que realizamos día con día, como podrás darte cuenta, sin ella resultaría mucho más complicado realizar algunas actividades como:

**\***Obtener la ruta más corta a nuestro destino.

**\***Saber que restaurantes o lugares se encuentran cercanos a mi.

**\***Conocer el punto exacto donde viene nuestro taxi o nuestra comida o,

**\***Conocer el punto dónde se encuentra el paquete que compraste por Internet.

**• Proceso de selección de los elementos que permitan aplicaciones dependientes de contexto.**

**Elementos y características de los sensores embebidos en los dispositivos móviles.**

Los sistemas embebidos son el núcleo de muchos productos, máquinas y operaciones inteligentes, como las aplicaciones de aprendizaje automático e inteligencia artificial. Como las aplicaciones de los sistemas embebidos aparecen hoy en día en todas las industrias y sectores, los dispositivos y el software embebidos desempeñan un papel crucial en el funcionamiento de los coches, los electrodomésticos, los dispositivos médicos, los quioscos interactivos y otros equipos que utilizamos en nuestra vida cotidiana. En este artículo, hemos proporcionado ejemplos de sistemas embebidos con explicaciones para ayudarle a aprender cómo esta tecnología está impactando en todas las facetas de la vida moderna.

**En los sistemas móviles**

Estos sistemas son más pequeños y fáciles de usar. Aunque vienen con una memoria limitada, la gente los prefiere por su portabilidad y manejabilidad. He aquí algunos ejemplos de sistemas de control embebidos móviles:

Hay muchas cosas con sistemas embebidos incorporados en el Internet de las Cosas (IoT), así como en dispositivos máquina a máquina (M2M). Excepcionalmente versátiles y adaptables, los sistemas embebidos pueden encontrarse en todos los dispositivos inteligentes de hoy en día. Es difícil encontrar una sola parte de la vida moderna que no implique esta tecnología. Estos son algunos de los ejemplos de la vida real de las aplicaciones de los sistemas embebidos.

* Sistemas de calefacción central
* Sistemas GPS
* Rastreadores de fitness
* Dispositivos médicos
* Sistemas de automoción
* Tránsito y cobro de tarifas
* Cajeros automáticos
* Robots de fábrica
* Estaciones de recarga de vehículos eléctricos
* Quioscos interactivos

**1. Sistemas de calefacción central**

Sistema de calefacción doméstico conectado

Los sistemas de calefacción central convierten la energía química en energía térmica en una sala de calderas y transfieren esa energía en calor, que luego se suministra a numerosos espacios de un edificio. Es importante que estos sistemas cuenten con controles de termostato para ajustar la temperatura, lo que se consigue mediante un sistema integrado.

Si un sistema de calefacción central no dispone de controles de temperatura, puede provocar el sobrecalentamiento de una habitación y dejar fría otra. Los controles de termostato adecuados le permitirán ajustar la temperatura a un nivel confortable y ahorrar mucha energía.

Los ejemplos de sistemas empotrados en la calefacción central pueden encontrarse en una serie de estructuras que requieren el control de la temperatura, tanto para el confort como para la gestión de bienes sensibles a la temperatura.

* Algunos ejemplos son:
* Edificios de oficinas
* Fábricas
* Tiendas de comestibles
* Casas
* Escuelas
* Hospitales

**2. Sistemas GPS**

Sistema GPS para vehículos

El GPS es un sistema de navegación que utiliza satélites y receptores para sincronizar datos relacionados con la ubicación, el tiempo y la velocidad. El receptor o dispositivo que recibe los datos tiene un sistema integrado para facilitar la aplicación de un sistema de posicionamiento global. Los dispositivos GPS integrados permiten a las personas encontrar fácilmente su ubicación actual y su destino. Por ello, están cobrando un rápido impulso y se están convirtiendo en las herramientas de navegación más utilizadas en los automóviles.

Hoy en día, los sistemas GPS se utilizan generalmente en:

* Coches
* Dispositivos móviles
* Palmtop

**3. Rastreadores de fitness**

Rastreador de fitness

Los rastreadores de fitness son dispositivos portátiles que pueden controlar su salud y hacer un seguimiento de actividades como dormir, correr y caminar. Estos dispositivos utilizan sistemas integrados para recopilar datos relacionados con el ritmo cardíaco, la temperatura corporal y el número de pasos, que luego se envían a servidores a través de WAN como LTE o GPRS.

Los rastreadores de fitness se utilizan generalmente para:

* Seguimiento de la actividad personal
* Control médico
* Entrenamiento deportivo

**4. Productos sanitarios**

Dispositivo médico conectado

Los dispositivos médicos de los centros sanitarios llevan tiempo incorporando sistemas integrados. Una nueva clase de dispositivos médicos utiliza sistemas embebidos para ayudar a tratar a los pacientes que necesitan una monitorización frecuente y una atención constante en casa. Estos sistemas llevan incorporados sensores que recogen datos relacionados con la salud de los pacientes, como la frecuencia cardíaca, el pulso o las lecturas de los implantes, que se envían a una nube donde el médico puede revisar los datos del paciente en su dispositivo de forma inalámbrica. Los dispositivos médicos se han utilizado ampliamente para diagnosticar y tratar a los pacientes de forma eficiente, y algunos de sus ejemplos son:

* Marcapasos
* Desfibrilador
* Escáneres de ultrasonidos

**5. Sistemas de automoción**

Tecnología de vehículos conectados a bordo

Los sistemas embebidos de automoción se diseñan e instalan para mejorar la seguridad de los automóviles. Gracias a los sistemas de seguridad de los vehículos, la tasa de mortalidad en el tráfico ha caído en picado en los últimos años. Las industrias automovilísticas van más allá para reforzar los automóviles con sistemas y sensores de tecnología avanzada, lo que no es posible sin los sistemas embebidos.

Algunos ejemplos clave de un sistema de seguridad activa son el control de velocidad adaptativo, la advertencia de avería del coche, el reconocimiento de peatones, la asistencia a la incorporación, los airbags, etc. Estas son algunas de las características que se prevé que mitiguen el riesgo de accidentes y fomenten la demanda de sistemas integrados en todo el mundo.

Algunos ejemplos más de sistemas integrados para automóviles son:

* Sistema de navegación del coche
* Sistema antibloqueo de frenos
* Sistema de entretenimiento del vehículo

**6. Tránsito y cobro de tarifas**

Recaudación automática de billetes de transporte

El cobro automático de billetes (AFC) es un sistema de emisión de billetes que permite a los pasajeros pagarlos a través de máquinas expendedoras o servicios en línea. Estos sistemas se originaron con monedas y fichas, pero han sido sustituidos por tarjetas de banda magnética o tarjetas inteligentes. Un AFC es un dispositivo básico de la estación que comprende una máquina expendedora de billetes, una máquina automática de acceso y una máquina de comprobación de billetes. Estos componentes son sistemas integrados que garantizan transacciones más rápidas, operaciones fluidas y un cobro más eficiente.

**•Proceso de programación de aplicaciones de acuerdo con sensores embebidos de móviles**

El desarrollador de software embebido necesita dominar el lenguaje de programación C. Este lenguaje está considerado como uno de los lenguajes más eficientes e independientemente del campo o empresa en la que se trabaje, es muy probable que se trabaje con C y C++.

El **software embebido** es la primera capa de código que se ejecuta en un dispositivo. Los **ingenieros de software embebido** escriben código, pero a diferencia de otros ingenieros de software, necesitan una **comprensión profunda del hardware** en el que el código se ejecutará.

Para programar sistemas embebidos, los desarrolladores pueden programar directamente en assembly del microprocesador, o utilizar lenguajes como C, C++ o incluso Java.

Además de estos lenguajes, existen plataformas que tienen herramientas específicas para el diseño de apps y prototipos con sistemas embebidos:

* Arduino
* Raspberry Pi
* BeagleBone
* Mbed

**Tareas del Ingeniero de Software en Sistemas Embebidos**

* Diseñar e implementar software embebido para configurar los componentes de HW.
* Participar en varias fases del ciclo de desarrollo del producto: definición del producto, diseño, planificación, ejecución y prueba.
* Revisar los requisitos del producto.
* Crear prototipos y evaluar soluciones alternativas en hardware de destino.
* Apoyar el análisis de la arquitectura de software integrado, software FTA y DFMEA.
* Entregar documentos detallados de especificación funcional y diseño técnico para las arquitecturas y componentes de software embebidos.
* Mantener y mejorar los entornos de construcción embebidos, y los entornos de pruebas automatizadas.

**CONCLUSIÓN**

Como conclusión nosotros logramos finalizar nuestra investigación con éxito, ya que logramos recaudar bastante información sobre las aplicaciones móviles dependientes del contexto y los sensores embebidos en los dispositivos móviles.

Esto nos permitió mejorar nuestro conocimiento sobre este tema, ayudándonos a conocer más sobre sus características y elementos de los dispositivos móviles.

Al realizar esta investigación se nos permitió conocer más sobre la implementación y programación de los sensores embebidos en los dispositivos móviles, lo que nos ayudó a comprender mejor su funcionamiento y sus distintos usos y funciones en la vida cotidiana.

Es de gran importancia para los estudiantes el manejo de sistemas y computadores embebidos que permitan con facilidad establecer firmemente las habilidades de programación estrechamente relacionadas con los conceptos y nociones de control.

**FUENTES BIBLIOGRAFICAS:**

* <https://es.slideshare.net/JavierMF/mecanismos-de-persistencia-en-android>
* <https://sites.google.com/site/mrtripus/home/sistemas-operativos-2/3-6-tolerancia-a-fallos-nivel-proceso-nivel-almacenamiento>
* <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/218>
* <https://inmediatum.com/blog/innovacion1/funcionalidades-offline-en-apps/>
* <https://jairogarciarincon.com/clase/interfaces-de-usuario-con-javafx/libretadirecciones-persistencia-con-base-de-datos-con-mysql-remota>
* <https://dspace.ups.edu.ec/bitstream/123456789/11409/1/UPS-GT001580.pdf>
* <https://es.digi.com/blog/post/examples-of-embedded-systems>
* https://www.freelancermap.com/blog/es/que-hace-desarrollador-sistemas-embebidos/