

Modelado de objetos de acceso a datos en dispositivos móviles.

Modelado de objetos en dispositivos móviles.

GRUPO: 4322-IS

CUATRIMESTRE: MAYO-AGOSTO

NOMBRE DEL PROFESOR: TORRES SERVIN EMMANUEL

INTEGRANTES:

MARQUEZ RODRIGUEZ JOCELYN

SANCHEZ AVILA EDRIC YAMIL

GAMEZ LEDO IRVING JONATHAN



INTRODUCCIÓN

En este artículo se realiza un análisis de fuentes de información referentes a referentes a Bases de Datos Móviles con el objetivo de obtener sus características características principales e identificar dónde reside la complejidad de la arquitectura de estos sistemas, teniendo en cuenta que se han convertido en un producto estratégico de importante orden, al tener una gran demanda en el incremento de las necesidades de acceder a la información, especialmente en lo relativo a la toma de decisiones.


Se analizan conceptos para el entendimiento del tema, referenciando antecedentes históricos que marcaron el surgimiento de este tipo de Bases de Datos.

Finalmente se hace referencia a la importancia del uso de los agentes móviles, detallando algunos problemas que ayudan a solucionar.

Características de los objetos de acceso a datos en dispositivos móviles:

Una gran cantidad de dispositivos electrónicos se clasifican actualmente como dispositivos móviles, desde teléfonos hasta tablets, pasando por dispositivos como lectores de RFID.

Con tanta tecnología clasificada como móvil, puede resultar complicado determinar cuáles son las características de los dispositivos móviles.

- Son aparatos pequeños.
 - La mayoría de estos aparatos 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 puede adaptarlos a su gusto.
- Tienen una alta capacidad de interacción mediante la pantalla o el teclado.

En la mayoría de los 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.

Estas cuatro características son:

1. **Movilidad.** La característica más evidente de un dispositivo móvil es, precisamente, que es móvil. Se entiende por movilidad la cualidad de un dispositivo para ser transportado o movido con frecuencia y facilidad. Por tanto, el concepto de movilidad es una característica básica.

Los dispositivos móviles son aquéllos que son lo suficientemente pequeños como para ser transportados y utilizados durante su transporte.

2. **Reducido tamaño.** Se entiende por tamaño reducido la cualidad de un dispositivo móvil de ser fácilmente usado con una o dos manos sin necesidad de ninguna ayuda o soporte externo.

El tamaño reducido también permite transportar el dispositivo cómodamente por parte de una persona.

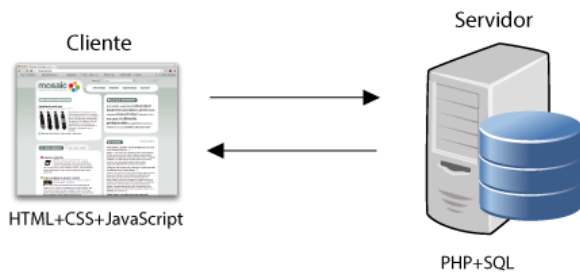
3. **Capacidad de comunicación inalámbrica.** Otro concepto importante es el término inalámbrico. Por comunicación inalámbrica se entiende la capacidad que tiene un dispositivo de enviar o recibir datos sin la necesidad de un enlace cableado. Por lo tanto, un dispositivo inalámbrico es aquél capaz de comunicarse o de acceder a una red sin cables (por ejemplo, un teléfono móvil o una PDA).
4. **Capacidad de interacción con las personas.** Se entiende por interacción el proceso de uso que establece un usuario con un dispositivo. Entre otros factores, en el diseño de la interacción intervienen disciplinas como la usabilidad y la ergonomía. Como hemos podido comprobar, la diversidad de términos, definiciones y características asociadas a los dispositivos móviles aumenta y cambia cada día, lo cual es propio de las tecnologías que están en continua evolución y desarrollo.

Proceso de modelado de objetos de acceso a datos en dispositivos móviles Y Proceso de programación de objetos de acceso a datos en dispositivos móviles:

Todas las bases de datos móviles tienen una arquitectura similar, donde debemos distinguir una serie de elementos principales característicos de este tipo de sistemas:

Servidor de base de datos corporativo y SMBD que gestiona y almacena los datos corporativos y proporciona aplicaciones corporativas. Se correspondería con el SMBD y el servidor que se utiliza en la organización.

Base de datos remota y SMBD que gestiona y almacena los datos móviles. Son las bases de datos que deben estar implementadas en los dispositivos móviles.



Plataforma de base de datos móvil, que puede ser un ordenador portátil, PDA u otro dispositivo de acceso a Internet, es decir, los dispositivos móviles en cuestión.

Enlaces de comunicación bidireccionales entre el SMD corporativo y el SMD móvil. Que pueden ser redes inalámbricas de distinta naturaleza, comunicaciones vía satélite, etc.

Arquitectura BDMóviles

Si nos fijamos en la figura anterior, podemos observar dónde reside la complejidad de este tipo de sistemas que coinciden con las partes no comunes con respecto a las bases de datos tradicionales. Nos referimos al sistema gestor de base de datos móvil (SMD móvil) y al enlace de comunicaciones.

En una base de datos móvil, la comunicación entre los dispositivos es una parte importante, ya que es imprescindible una buena comunicación para el acceso a los datos.

La arquitectura de comunicaciones más utilizada consiste en tener una o varias estaciones base en contacto con la base de datos corporativa y una serie de estaciones móviles que acceden a los datos a través de las estaciones base.

Por tanto, podemos encontrarnos con alguna cuestión importante relacionada con el acceso o la localización, como puede ser localizar una estación móvil que contenga los datos que necesitamos en un momento determinado.

Manipulación de datos en dispositivos móviles:

En una aplicación web, como un gestor de contenidos o una tienda en línea o cualquier otra web que haga uso de bases de datos, podemos encontrar hasta cuatro tipos de desarrollo:

1. Desarrollo de HTML + CSS
2. Desarrollo JavaScript
3. Desarrollo en PHP
4. Consultas SQL

De las cuatro, las dos primeras se ejecutan en el navegador y las dos segundas, en el servidor, aproximadamente de la siguiente manera:

- En el servidor, un archivo .php contiene el programa PHP con las instrucciones SQL para acceder a los datos, con el código HTML y CSS necesario para poderlos visualizar correctamente y con el JavaScript necesario para interactuar con el usuario.
- El servidor ejecuta las instrucciones PHP y las consultas SQL, obtiene los datos y envía al cliente un archivo con el HTML, el CSS, el JavaScript y los datos obtenidos.
- El navegador presenta los datos al usuario y ejecuta las instrucciones JavaScript.
- El navegador envía al servidor los datos que proporciona el usuario.
- El servidor recibe esa información y la guarda en la base de datos.

En la práctica, PHP, SQL y HTML suelen ir en un archivo, mientras que el CSS y el JavaScript van en archivos aparte, con lo que el navegador recibe un archivo HTML con los datos, un archivo CSS con el diseño y un archivo JavaScript con el programa que se va a ejecutar.

En general, las diferentes tecnologías que podemos encontrar en una aplicación web requieren de varias personas con diferentes competencias. Volviendo a la base de datos, su diseño no es una tarea trivial.

Dependiendo de lo bien o mal hecha que esté, el número de transacciones (consultas o modificaciones) que deberán hacerse puede variar de modo considerable.

En aplicaciones sencillas o para sitios con poco movimiento, la importancia puede ser relativa, pero en aplicaciones complejas y, en especial, en sitios con muchas visitas, reducir el número de consultas a la base de datos puede reducir considerablemente el coste en hardware del sitio. Y es que, en sitios web con muchas visitas, la importancia de la base de datos es tal que suele situarse en un servidor aparte del servidor de la aplicación.

Normalmente, una base de datos requiere un programa de software de bases de datos completo, conocido como sistema de gestión de bases de datos (DBMS). Un DBMS sirve como interfaz entre la base de datos y sus programas o usuarios finales, lo que permite a los usuarios recuperar, actualizar y gestionar cómo se organiza y se optimiza la información. Un DBMS también facilita la supervisión y el control de las bases de datos, lo que permite una variedad de operaciones

administrativas como la supervisión del rendimiento, el ajuste, la copia de seguridad y la recuperación.

Algunos ejemplos de software de bases de datos o DBMS populares incluyen MySQL, Microsoft Access, Microsoft SQL Server, FileMaker Pro, Oracle Database y dBASE.

Las bases de datos de autogestión son la ola del futuro y ofrecen una posibilidad fascinante para las organizaciones que deseen utilizar la mejor tecnología de bases de datos disponible sin los problemas que supone ejecutar y utilizar esa tecnología.

Las bases de datos de autogestión utilizan una tecnología basada en la nube y el machine Learning para automatizar muchas de las tareas rutinarias necesarias para gestionar bases de datos, como el ajuste, la seguridad, las copias de seguridad, las actualizaciones y otras tareas de gestión rutinarias.

Al automatizar estas tediosas tareas, los administradores de bases de datos se quedan libres para realizar un trabajo más estratégico.

Las funciones de autogestión, autoprotección y autorreparación de las bases de datos de autogestión están preparadas para revolucionar la forma de gestionar y proteger los datos de las empresas, lo que permite obtener ventajas de rendimiento, reducir costes y mejorar la seguridad.

Persistencia de datos en los dispositivos móviles:

Aparte de la Base de Datos SQLite en Android existen otros medios para almacenar información de la aplicación e inclusive información de preferencias del usuario; en esta primera entrega, veremos algunas de ellas para luego adentrarnos un poco más en cada una.

Antes de hablar de los tipos de medios de persistencia que existen en Android es importante refrescar un poco el concepto como tal de persistencia en informática, nuestro campo de interés.

La persistencia en el ámbito de una aplicación indiferentemente si es una aplicación Android o de cualquier otro tipo consiste en que los datos manipulados por la aplicación "sobrevivan" a la ejecución de la misma en el tiempo; en otras palabras; consiste en almacenar los datos en un medio secundario, no volátil para posterior reconstrucción y utilización; por lo tanto son independientes en el tiempo del proceso que los creó.

Coloquialmente hablando... consiste en que los datos no se borren luego de que la aplicación se cierre.

La Persistencia en Android

La persistencia en Android consiste en tres tipos de almacenamientos con un propósito muy específico.

Persistencia en Android: Preferencias Compartidas o Shared Preferences

Con Shared Preferences podemos almacenar y recuperar en el formato clave-valor información como texto, booleanos y números; lo que lo convierte en potencial para almacenar configuraciones del usuario como: estilos, preferencias, etc.

Shared Preferences en la práctica

los modos de acceso posibles son:

MODE_PRIVATE: Sólo nuestra aplicación tiene acceso a estas preferencias.

MODE_WORLD_READABLE: Todas las aplicaciones pueden leer estas preferencias, pero sólo la nuestra puede modificarlas (deprecated desde el API 17).

MODE_WORLD_WRITEABLE: Todas las aplicaciones pueden leer y modificar estas preferencias (deprecated desde el API 17).

Mecanismos de tolerancia a fallos:

En los sistemas distribuidos, como en todos los sistemas diseñados para prestar un servicio continuo, la disponibilidad es un parámetro clave para medir su

desempeño; por lo tanto, es importante que este tipo de sistemas sea capaz de recuperarse ante la ocurrencia de fallos.

Por otra parte, dado que en la actualidad la causa más frecuente por la cual un sistema suspende involuntariamente su operación es algún fallo debido al software y que la mayoría de estos fallos tienen una naturaleza transitoria, se ha tratado de mejorar la fiabilidad de los sistemas desde dos enfoques: la Ingeniería de Software y la Tolerancia a Fallos.

Dentro de este último enfoque, los protocolos de recuperación basados en rollback son de particular interés, pues permiten implementar una abstracción de sistema en la cual el fallo de un proceso se traduce simplemente en una falta de disponibilidad transitoria, no requieren de hardware especial para su implementación y, por lo tanto, representan una solución asequible para cualquier aplicación distribuida.

Un diseño tolerante a fallos es un sistema que está capacitado para continuar su funcionamiento cuando algún componente del sistema falla, posiblemente a un nivel más reducido, lo que es mejor a que el sistema falle completamente.

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.

CONCLUSIÓN

Para concluir con esta investigación: administración de datos en dispositivos móviles de Desarrollo de aplicaciones para dispositivos móviles, se puede mencionar que los datos son la base de toda aplicación ya sea móvil o de escritorio, ya que de manera significativa se usa en la mayoría de las aplicaciones ya sea para guardar datos o cualquier información que el usuario requiere

Y en muchos casos las aplicaciones móviles tienen su contrapartida en la web tradicional donde generalmente se realiza la carga masiva de datos, es decir se dispone de un sistema backend de administración y el frontend de la aplicación es el sistema móvil.

REFERENCIAS BIBLIOGRAFICAS:

Vera, J., Mamani, A. O., & Villalba, K. (2015). Modelo de sistema de recomendación de Objetos de Aprendizaje en dispositivos móviles, caso: Desarrollo del pensamiento computacional. In *XX Congreso Internacional de Informática Educativa, TISE 2015. Nuevas Ideas en Informática Educativa (Santiago, Chile, 1-3 de diciembre de 2015)* (pp. 730-734).

Ramírez-Sánchez, M., Suárez-Rivero, J. P., & Castellano-Hernández, M. Á. (2014). Epigrafía digital: tecnología 3D de bajo coste para la digitalización de inscripciones y su acceso desde ordenadores y dispositivos móviles. *Profesional de la Información*, 23(5), 467-474.

Basantes, A. V., Naranjo, M. E., Gallegos, M. C., & Benítez, N. M. (2017). Los dispositivos móviles en el proceso de aprendizaje de la Facultad de Educación Ciencia y Tecnología de la Universidad Técnica del Norte de Ecuador. *Formación universitaria*, 10(2), 79-88.

Tovar, L. C., Bohórquez, J. A., & Puello, P. (2014). Propuesta metodológica para la construcción de objetos virtuales de aprendizaje basados en realidad aumentada. *Formación universitaria*, 7(2), 11-20.

CLIENTES LIGEROS

Un cliente ligero o cliente delgado es una computadora cliente o un software de cliente en una arquitectura de red cliente-servidor que depende principalmente del servidor central

CLIENTES LIGEROS EN APLICACIONES ORIENTADAS A MÓVILES.



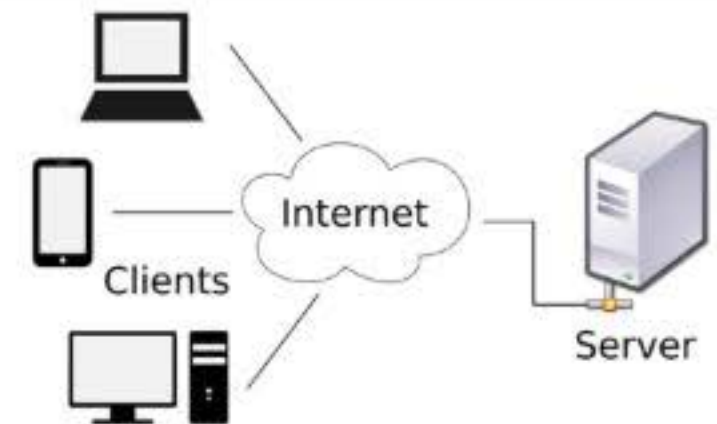
Los clientes ligeros se gestionan de manera centralizada y la mayor parte de sus memorias, aplicaciones y datos se almacenan en un servidor. Puede ser un servidor de nube o un servidor periférico en el sitio.

CLIENTE Y DE SERVIDOR.

Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.

PROCESO DE PROGRAMACIÓN DE CLIENTES LIGEROS DE APLICACIONES ORIENTADAS A MÓVILES.

El cliente liviano está cercanamente conectado con la interfaz de usuario. En un sistema cliente liviano-servidor, el único software que es instalado en el cliente liviano es la interfaz de usuario, algunas aplicaciones frecuentemente usadas.



CONEXIÓN A SERVIDORES EN INTERNET

TIPOS DE CONEXIONES QUE TIENEN LAS APLICACIONES ORIENTADAS A MÓVILES.

INVOLUCRA

APLICACIONES MÓVILES NATIVAS.
(NATIVE APP) SON AQUELLAS EN LAS QUE SE LLEVA A CABO ESPECÍFICAMENTE UN DESARROLLO PARA CADA UNA DE LAS PLATAFORMAS. EN EL CASO DE LOS MÓVILES: ANDROID, IOS...

APLICACIONES MÓVILES HÍBRIDAS O MULTIPLATAFORMA.
LAS APLICACIONES HÍBRIDAS (HÍBRID APP) O MULTIPLATAFORMAS (CROSS PLATFORM) SON AQUELLAS QUE NOS PERMITEN DESARROLLAR PARA VARIAS PLATAFORMAS A LA VEZ CON TAN SOLO REALIZAR UN DESARROLLO.

APLICACIONES MÓVILES WEB.
LAS APLICACIONES WEB (WEB APP) SON AQUELLAS EN LAS QUE SE REALIZA UN ÚNICO DESARROLLO PARA TODAS LAS PLATAFORMAS Y PODEMOS ACCEDER DESDE UN NAVEGADOR WEB A DICHAS APLICACIONES.

PROCESO DE PROGRAMACIÓN DE CONEXIÓN A BASES DE DATOS WEB

SE REFIERE A

El número de conexiones a una base de datos que requiere un nodo de integración depende de las acciones de los flujos de mensajes que acceden a la base de datos. Para cada hebra de flujo de mensajes, un nodo de integración que accede a una base de datos realiza una conexión para cada nombre de origen de datos (DSN).

LAS BASES DE DATOS CONTIENEN DATOS EMPRESARIALES QUE GRABAN LOS FLUJOS DE MENSAJE DESPLEGADOS Y A LOS QUE SE ACCEDERÍA A TRAVÉS DE LOS MISMOS. DEBE CREAR CONEXIONES DESDE EL NODO DE INTEGRACIÓN A LA BASE DE DATOS UTILIZANDO ODBC O JDBC.

NORMALMENTE, EL NODO DE INTEGRACIÓN REALIZA LAS CONEXIONES CUANDO NECESITA UTILIZARLAS EN EL FLUJO DE MENSAJES. NO OBSTANTE, EN EL CASO DE LAS CONEXIONES CON LOS ORIGENES DE DATOS BASADOS EN ESOL, PUEDE ESTABLECER LA CONEXIÓN INICIAL ANTES DE QUE EL FLUJO RECIBA UN MENSAJE.

PROCESO DE PROGRAMACIÓN DE CONEXIÓN A SERVIDORES DE APLICACIONES

SE REFIERE A

Una aplicación desarrollada con las tecnologías J2EE permite ser desplegada en cualquier servidor de aplicaciones o servidor web que cumpla con el estándar. Un servidor de aplicaciones es una implementación de la especificación J2EE.

INVOLUCRA LOS SISTEMAS

CLIENTE WEB (CONTENEDOR DE APPLETS): ES USUALMENTE UN NAVEGADOR E INTERACTÚA CON EL CONTENEDOR WEB HACIENDO USO DE HTTP. RECIBE PÁGINAS HTML O XML Y PUEDE EJECUTAR APPLETS Y CÓDIGO JAVASCRIPT.

APLICACIÓN CLIENTE: SON CLIENTES QUE NO SE EJECUTAN DENTRO DE UN NAVEGADOR Y PUEDEN UTILIZAR CUALQUIER TECNOLOGÍA PARA COMUNICARSE CON EL CONTENEDOR WEB O DIRECTAMENTE CON LA BASE DE DATOS.

SERVIDOR DE APLICACIONES: PROPORCIONA SERVICIOS QUE SOPORTAN LA EJECUCIÓN Y DISPONIBILIDAD DE LAS APLICACIONES DESPLEGADAS. ES EL CORAZÓN DE UN GRAN SISTEMA DISTRIBUIDO.

MULTIMEDIA E INTERACCIÓN GRÁFICA EN MÓVILES

MULTIMEDIA Y GRAFICACIÓN



Utilizados para representar esquemas, planos, dibujos lineales, etc.

ELEMENTOS MULTIMEDIA Y LA INTERACCIÓN GRÁFICA

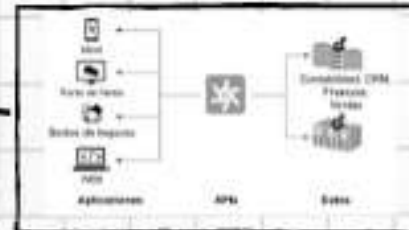


Son programas (o documentos) informáticos, que suelen estar almacenados en discos compactos (CD-ROM), o DVDs. También pueden estar disponibles en Internet (páginas Web).

ELEMENTOS MULTIMEDIA Y INTERACCIÓN GRÁFICA



Nos permiten interactuar con el ordenador utilizando diferentes códigos en la presentación de la información (texto, imagen, sonido, ...).



SELECCIÓN DE ELEMENTOS MULTIMEDIA Y DE INTERACCIONES GRÁFICAS



Diseño del programa desde el punto de vista técnico y estético. Diseño del programa desde el punto de vista didáctico.

Aplicaciones móviles dependientes del contexto

Y

Sensores embebidos en los dispositivos móviles

GRUPO: 4322-IS

CUATRIMESTRE: MAYO-AGOSTO

NOMBRE DEL PROFESOR: TORRES SERVIN EMMANUEL

INTEGRANTES:

MARQUEZ RODRIGUEZ JOCELYN

SANCHEZ AVILA EDRIC YAMIL

GAMEZ LEDO IRVING JONATHAN



INTRODUCCIÓN

En este proyecto se realiza un análisis en donde se presenta diversas aplicaciones desarrolladas basadas en el uso de uno o más sensores disponibles, abriendo un panorama amplio a nuevas formas de interactuar con los dispositivos.

Las interfaces físicas permiten interactuar de forma más natural e intuitiva, moviendo, girando o simplemente apoyando el dispositivo sobre una superficie.

Se podrá evidenciar como los desarrollos realizados proponen nuevas formas de interacción.

¿Qué es el contexto en las aplicaciones?

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, 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.

Características:

1. Ofrecer beneficios únicos

Sin duda es el aspecto más importante de todos, ofrecer la solución a un problema que pocos se hayan arriesgado a tocar y ofrecer beneficios que nadie más ofrece puede ser un gran impulso para una aplicación, especialmente cuando es una necesidad de un gran segmento de personas.

Esto debe venir acompañado de una buena implementación y de sencillez en su uso; pues de nada sirve ofrecer algo novedoso cuando es complicado de hacer o presenta muchos problemas.

Esta característica hará que una aplicación sume muchos puntos, pero para eso se deben también implementar algunas estrategias de posicionamiento, para que así más usuarios puedan descubrirla y disfrutarla.

2. Adaptación a varios sistemas operativos.

Suponiendo que contamos con una aplicación que ofrece como en el caso anterior beneficios únicos a sus usuarios, es importante conocer qué porcentaje de ellos hacen uso de cada uno de los sistemas operativos del mercado, pues de nada sirve que se desarrolle en un único sistema cuando la mitad del público se encuentra haciendo uso de otro.

3. Diseño agradable y capacidades de interacción

Otro de los aspectos en los que se debe enfocar una App es en ofrecer una interfaz visual agradable, simple, ordenada y funcional. Manteniendo siempre

coherencia con la identidad visual de la empresa y con el diseño de sus otras plataformas o sistemas, especialmente aquellas disponibles para sus clientes. Pero esto no es todo, también es importante que un buen diseño este acompañado de unas funcionalidades útiles, rapidez de navegación y por supuesto acciones que generen interacción y una experiencia agradable en los usuarios. Por ejemplo, agregar motores de búsqueda, medios de contacto, notificaciones o personalización por medio de perfiles de usuario.

4. Analítica

Si bien este no es un aspecto que le sirva directamente a los usuarios de la App, es una herramienta de gran utilidad para la compañía ya que esto le permite medir y rastrear las acciones que el usuario ejecuta dentro de la herramienta, por lo que la compañía puede conocer aquellos patrones de comportamiento por los que ellos se caracterizan, para así ofrecer mejoras de interés o eliminar aquello que de acuerdo con los datos no es funcional para ellos.

5. Actualizaciones

Las actualizaciones de una App son una necesidad, pues estas permiten mejorar los errores, aquellas funcionalidades que quizá no están funcionando como deberían o que han recibido muchas critica parte de los usuarios.

Esta característica va muy de la mano con el punto anterior, pues la analítica permite conocer toda la información necesaria para implementar estrategias de mejoramiento de la aplicación y para darle a entender a los usuarios que son importantes y que sus comentarios son escuchados.

Lo más importante de todo esto es conocer muy bien a los clientes y entender lo que buscan y necesitan, así poder convertir la aplicación en todo lo que ellos esperan, en una herramienta de verdadero valor, que se posicione en sus mentes y que se convierta en una herramienta de uso constante.

Usos de la geolocalización en las aplicaciones de los móviles

- 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.

Cada aplicación de Android reside en su propia zona de pruebas de seguridad y está protegida por las siguientes características de seguridad de Android:

- El sistema operativo Android es un sistema Linux multiusuario en el que cada aplicación es un usuario diferente.
- De forma predeterminada, el sistema le asigna a cada aplicación un ID de usuario de Linux único (solo el sistema utiliza el ID y la aplicación lo desconoce). El sistema establece permisos para todos los archivos en una aplicación de modo que solo el ID de usuario asignado a esa aplicación pueda acceder a ellos.
- Cada proceso tiene su propia máquina virtual (VM), por lo que el código de una aplicación se ejecuta de forma independiente de otras aplicaciones.
- De forma predeterminada, cada aplicación ejecuta su propio proceso de Linux. El sistema Android inicia el proceso cuando se requiere la ejecución de alguno de los componentes de la aplicación y, luego, lo cierra cuando el proceso ya no es necesario o cuando el sistema debe recuperar memoria para otras aplicaciones.

De esta manera, el sistema Android implementa el principio de mínimo privilegio. Es decir, de forma predeterminada, cada aplicación tiene acceso solo a los componentes que necesita para llevar a cabo su trabajo y nada más.

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

Las fases de desarrollo de una aplicación móvil abarcan desde el presupuesto y concepción de la idea, hasta el mantenimiento de la app. En este artículo vamos a prestar atención a cada una de las partes de su proceso de desarrollo y comentaremos las siguientes etapas:

Elaboración del presupuesto

La idea

Elegir al proveedor

Análisis de los requerimientos

La planificación

UX y diseño de la aplicación

Desarrollo app

Testing

Lanzamiento

Mantenimiento

Sensores embebidos en los dispositivos móviles

Elementos y características de los sensores embebidos en los dispositivos móviles:

Se denomina desarrollo de aplicaciones móviles al conjunto de conocimientos, técnicas y procesos necesarios para crear software simple que pueda ser usado en dispositivos móviles como por ejemplo tabletas y celulares.

El objetivo de estas aplicaciones es crear valor para el usuario, ya sea por su utilidad o porque crea entretenimiento.

Cada vez más empresas crean apps para fidelizar a sus clientes.

Las etapas del proceso de desarrollo incluyen una investigación de mercado y tecnología, prototipo, diseño, desarrollo final, pruebas e implementación.

CARACTERÍSTICAS:

Las principales características de un sistema embebido son el bajo costo y consumo de potencia.

Dado que muchos sistemas embebidos son concebidos para ser producidos en miles o millones de unidades, el costo por unidad es un aspecto importante a tener en cuenta en la etapa de diseño.

Fiabilidad y seguridad: Un fallo en un sistema de control puede hacer que el sistema controlado se comporte de forma peligrosa o antieconómica.

Eficiencia: Gran parte de los sistemas de control deben responder con gran rapidez a los cambios en el sistema controlado

Interacción con dispositivos físicos: Los sistemas empujados interactúan con su entorno mediante diversos tipos de dispositivos que normalmente no son convencionales (teclados, impresoras): convertidores A/D y D/A, pwm, entradas y salidas digitales paralelo y serie, (interfases con sensores, actuadores, periféricos especiales).

Robustez: Embarcados en sistemas con movimiento o que pueden ser transportados, sujetos a vibraciones e incluso impactos (coches, robots, instrumentación portátil). No siempre trabajan en condiciones óptimas de temperatura, humedad, y limpieza.

Entre las plataformas de los sistemas operativos embebidos tenemos:

QNX es un sistema operativo embebido desarrollado por QNX Software Systems Ltd, para aplicaciones de dispositivos electrónicos, telecomunicaciones, sistemas automotrices, etc.; que necesitan una gran confiabilidad, desempeño, funcionalidades específicas, y una escalabilidad masiva.

Linux embebido es un sistema Linux para los dispositivos embebidos comenzó con el soporte del kernel y el compilador para los microprocesadores más populares de 32 bits: x86, ARM, PowerPC, MIPS y SH. Y luego continuó con la aparición de diferentes distribuciones de Linux con soporte para características específicas de los sistemas embebidos.

Windows XP para sistemas embebidos. Aunque Microsoft ha mantenido a los fabricantes de sistemas operativos alternativos a Windows contra la pared en lo que se refiere a equipos de sobremesa, su posición en la batalla por la supremacía en el terreno de los sistemas operativos para dispositivos que no fueran PC era vulnerable. De ahí que Microsoft da un paso más en este sector, lanzando la versión para dispositivos embebidos en la Conferencia para desarrolladores de este tipo de sistemas que se celebra en Las Vegas. Esta decisión parece ir en consonancia con las predicciones que apuntan a una progresiva inclusión de microprocesadores en casi cualquier tipo de aparatos (frigoríficos, coches, etc).

DOS está estrechamente ligado al progresivo desarrollo de los microprocesadores de arquitectura x86 de Intel. Y es que es un sistema operativo que hay que conocer porque sigue estando en vigor gracias a su estabilidad ya que se desarrollaba en paralelo con la tecnología de Intel e IBM y porque es un sistema operativo muy simple, monousuario y monotarea. En otras palabras, DOS sólo ejecuta un programa cada vez

Java para sistemas embebidos. La tecnología Java es cada vez más utilizada en sistemas embebidos avanzados.

Existen también plataformas desarrolladas por distintos fabricantes que proporcionan herramientas para el desarrollo y diseño de aplicaciones y prototipos con sistemas embebidos desde ambientes gráficos, algunos ejemplos de estas son: Arduino, mbed, Raspberry Pi, BeagleBone, etc.

. Conclusión

Como conclusión se tiene que el hardware de los dispositivos permite el desarrollo de aplicaciones móviles más complejas que aprovechan sus sensores en beneficio de sus usuarios.

Este proyecto presenta diferentes aplicaciones desarrolladas que se basan en el uso de uno o más sensores disponibles, abriendo un amplio panorama a nuevas formas de interactuar con los dispositivos.

Las interfaces físicas permiten interactuar de una forma más natural e intuitiva, moviendo, rotando o simplemente apoyando el dispositivo sobre una superficie.

Podrá demostrar cómo los desarrollos realizados proponen nuevas formas de interacción.

REFERENCIAS BIBLIOGRAFICAS

Enriquez, J. G., & Casas, S. I. (2013). Usabilidad en aplicaciones móviles. *Informes científicos técnicos-UNPA*, 5(2), 25-47.

Vaíllo Martín, J. (2013). *Desarrollo de una aplicación móvil de compra de entradas adaptativa según contexto para Android* (Master's thesis).

Soria Morillo, L. M. (2011). Middleware para el desarrollo de aplicaciones ubicuas en dispositivos móviles.

Flórez Aristizábal, L. (2015). *Diseño y desarrollo de una plataforma computacional sobre dispositivos móviles para emulación de sistemas dinámicos* (Doctoral dissertation, Universidad Autónoma de Occidente).

Barillaro, S., De Luca, G., Valiente, W., Carnuccio, E., García, G., Volker, M., ... & Pérez, M. (2016, May). Diseño de sistema IoT de monitoreo y alarma para personas mayores. In *XVIII Workshop de Investigadores en Ciencias de la Computación (WICC 2016, Entre Ríos, Argentina)*.

Velasquez Luis, R. *Diseño de un sistema embebido para el control de un invernadero* (Doctoral dissertation).