



UNIVERSIDAD POLITÉCNICA
DE TECÁMAC

TAREAS MOVILES

Balbuena Martínez Yael

1321124280

Guerrero Gallardo Eduardo

1321124332

Gutierrez Hernández Jesús Fernando

1321124801



Materia:

Programación
Visual.

Profesor:

Torres Servín
Emmanuel

Grupo:

4322IS

Carrera:

Ingeniería en
Software

TAREA 1

INTRODUCCIÓN

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.

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.

La flexibilidad tiene un precio. Cuando se añaden DAOs a una aplicación, la complejidad adicional de usar otra capa de persistencia incrementa la cantidad de código ejecutado durante tiempo de ejecución. La configuración de las capas de persistencia requiere en la mayoría de los casos mucho trabajo.

Para tener acceso y manipular datos mediante programa debe comprender la jerarquía de DAO. El orden de los objetos en DAO se conoce como su modelo de objetos. El modelo de objetos de DAO le permite escribir código que puede aprovechar la funcionalidad de la base de datos.

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.

Modelado de objetos de acceso de datos en dispositivos móviles modelado de objetos en dispositivos móviles

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

- Los objetos de Acceso a datos son un patrón de Diseño CORE J2EE y considerados buena práctica
- Los objetos de acceso a datos pueden usarse en java para aislar una aplicación de la tecnología de persistencia java subyacente la cual podría ser JDBC que persistencia controlada por el contenedor o cualquier otra tecnología de persistencia.
- DAQ es un objeto que proporciona una interfaz a algún tipo de base de datos u otro mecanismo de persistencia. DAQ se puede utilizar en un gran porcentaje de las aplicaciones, que requieren almacenamiento de datos u otro mecanismo de persistencia.
- Proporciona algunas operaciones de datos específicos sin exponer a los detalles de la base de datos. Como resultado de ello, los mecanismos de acceso a los datos se pueden cambiar.

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

Son una serie de objetos que le permiten tener acceso y manipular datos mediante programación en bases de datos locales o remotos.

Utilizado para administrar bases de datos, así como sus objetos y su estructura.

Un 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.

La 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 móviles que puede ser un ordenador portátil, PDA u otro dispositivo de acceso a internet, es decir, los dispositivos móviles en cuestión.

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

- En 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 corporativos y una serie de estaciones móviles que acceden a los datos a través de las estaciones base

Manipulación de datos en dispositivos móviles

1. Reconocer el concepto de conexión a bases de datos

- Es un archivo de configuración donde se especifica los detalles físicos de una base de datos como por ejemplo el tipo de base de datos y la versión, y los parámetros que permiten una conexión JDBC desde el IBM
- Bases de datos las gestiona internamente el nodo de integración y, por consiguiente, las opciones de agrupación de conexiones configurables disponibles en el controlador ODBC no se deben utilizar.

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

- Estáticos: 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.
- Dinámicos: 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: le permite trabajar con tablas de base de datos directamente. Se pueden crear conexiones de base de datos para las bases de datos compatibles y bases de datos relacionales adicionales que utilizan el controlador.

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

- Son base de datos de solo lectura. Es decir, están diseñadas para agregar datos fijos que no se pueden modificar con el tiempo
- Se utilizan fundamentalmente para almacenar datos históricos o hechos invariables
- Se suelen combinar diferentes bases de datos estáticas realizadas en diferentes periodos para analizar la evolución de los datos en el tiempo
- Por eso son muy usadas para hacer estudios de mercado, investigaciones estadísticas y otros proyectos relacionados con el business.

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

- Son bases de datos relacionales, es decir, en ellas se establecen relaciones entre los registros y sus campos
- Están orientadas al almacenamiento de información que podría cambiar en el tiempo
- Permiten añadir, modificar o eliminar la información presente en los campos de la base de datos
- Se opone a las bases de datos estáticas, las cuáles no permiten editar la información

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

- Es la habilidad de recuperar la información de los sistemas de computación y/o repositorios de información sobre los dispositivos móviles en cualquier momento en cualquier lugar.
 - Este es un sistema distribuido que soporta conectividad móvil posee todas las capacidades de un sistema de base de datos y permiten a las unidades móviles
- Se debe tomar en cuenta
- Desconexiones

- Movilidad
- Errores
- Fallas en el dispositivo móvil
- Se debe mantener la autonomía y la consistencia local del SMBD

Persistencia de datos en los dispositivos móviles

1. Concepto de persistencia en dispositivos móviles

Esta consiste en tres tipos de almacenamientos con un propósito específico

- Preferencias Compartidas o 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.

2. Retos de la persistencia en los dispositivos móviles

- En estos se crea una instancia del contenedor de datos para PROTO DATA STORE y antes hay que definir como se transforma el dato a guardar según el esquema definido y como ese dato se recupera.
- Este forma de guardar datos es algo diferente ya que como veíamos antes, hay que definir un esquema de datos con anterioridad y se tiene en cuenta el tipo del dato que vayamos a guardar y que definimos en el esquema

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

- Almacenamiento específico: Almacena archivos diseñados solo para tu app, ya sea en directorios dedicados dentro de un volumen de almacenamiento interno o en directorios dedicados diferentes dentro del almacenamiento externo
- Almacenamientos compartido: Almacena archivos que tu app pretenda compartir con otras apps, incluidos archivos multimedia, documentos y otros
- Preferencias: Almacena datos primitivos y privados en pares clave-valor
- Base de datos: Almacena datos estructurados en una base de datos privado mediante la biblioteca de persistencias.

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

- El modelo de persistencia de objetos proporciona un conjunto de atributos para mapear las clases del lado del cliente a tablas y las propiedades/campos atributos de tabla
- El modelo de persistencia de objetos no proporciona una api para crear actualizar o eliminar tablas

5. Proceso de programación de persistencia en dispositivos móviles.

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

6. Mecanismos de tolerancia a fallos

- Elementos para tomar en cuenta en el desarrollo de aplicaciones orientadas a móviles:
- Conectividad: una solución móvil en una empresa jamás es una solución aislada. Normalmente es una extensión de los sistemas empresariales existentes como ERPs o CRMs por lo tanto, es fundamental entender las opciones de conectividad disponibles en el mercado y el impacto que tiene en nuestra
- Plataforma: más común para desarrollo de aplicaciones móviles este para el desarrollo de aplicaciones corporativas, he elegido especialmente este
- Herramientas de desarrollo: las opciones de IDE dependen de la selección de plataforma

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

- Este es una propiedad que le permite a un sistema seguir funcionando correctamente en caso de fallo de uno o varios de sus componentes. Si disminuye su calidad de funcionamiento, la disminución.
- Este es un sistema que está capacitado para continuar su funcionamiento cuando algún componente del sistema posiblemente a un nivel más reducido, lo que es

mejor a que el sistema falle completamente. El termino es comúnmente usado para describir sistemas basados en computadoras diseñados para continuar en mayor o menor medida las operaciones que realiza con, a lo mejor, una reducción de su rendimiento o un incremento de los tiempos de respuesta en las componentes.

CONCLUSIÓN

Estos procedimientos almacenados con la integridad referencial donde una base de datos móvil, la comunicación entre los dispositivos es una parte importante, ya que es imprescindible a una buena comunicación para el acceso a los datos.

Este es un proceso mediante el cual las aplicaciones son desarrolladas para dispositivos de mano para teléfonos móviles.

Los Objetos de Acceso a Datos son un Patrón de los subordinados 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.

Los Objetos de Acceso a Datos pueden usarse en Java para aislar a una aplicación de la tecnología de persistencia Java subyacente (API de Persistencia Java), la cual podría ser JDBC, JDO, Enterprise JavaBeans, TopLink, EclipseLink, Hibernate, iBATIS, o cualquier otra tecnología de persistencia.

La flexibilidad tiene un precio. Cuando se añaden DAOs a una aplicación, la complejidad adicional de usar otra capa de persistencia incrementa la cantidad de código ejecutado durante tiempo de ejecución. La configuración de las capas de persistencia requiere en la mayoría de los casos mucho trabajo.

Las aplicaciones críticas con el rendimiento no deberían usar DAOs.

BIBLIOGRAFÍA

Desarrollo Libre, (s / f), Persistencia en Android Developer,

<https://www.desarrollolibre.net/blog/android/persistencia-en-android-developer>

Henry Curi, (s / f), Administración de Datos en Dispositivos Móviles,

<https://pdfcoffee.com/administracion-de-datos-en-dispositivos-moviles-3-pdf-free.html>

IBM Integration Bus, 2015, IBM Dispositivos Móviles,

<https://www.ibm.com/docs/es/integration-bus/10.0?topic=overview-database-connections>

Modelos de BD, 2012, Base de Datos Móviles,

<https://modelosbd2012t1.wordpress.com/2012/03/15/base-de-datos-moviles-3/>

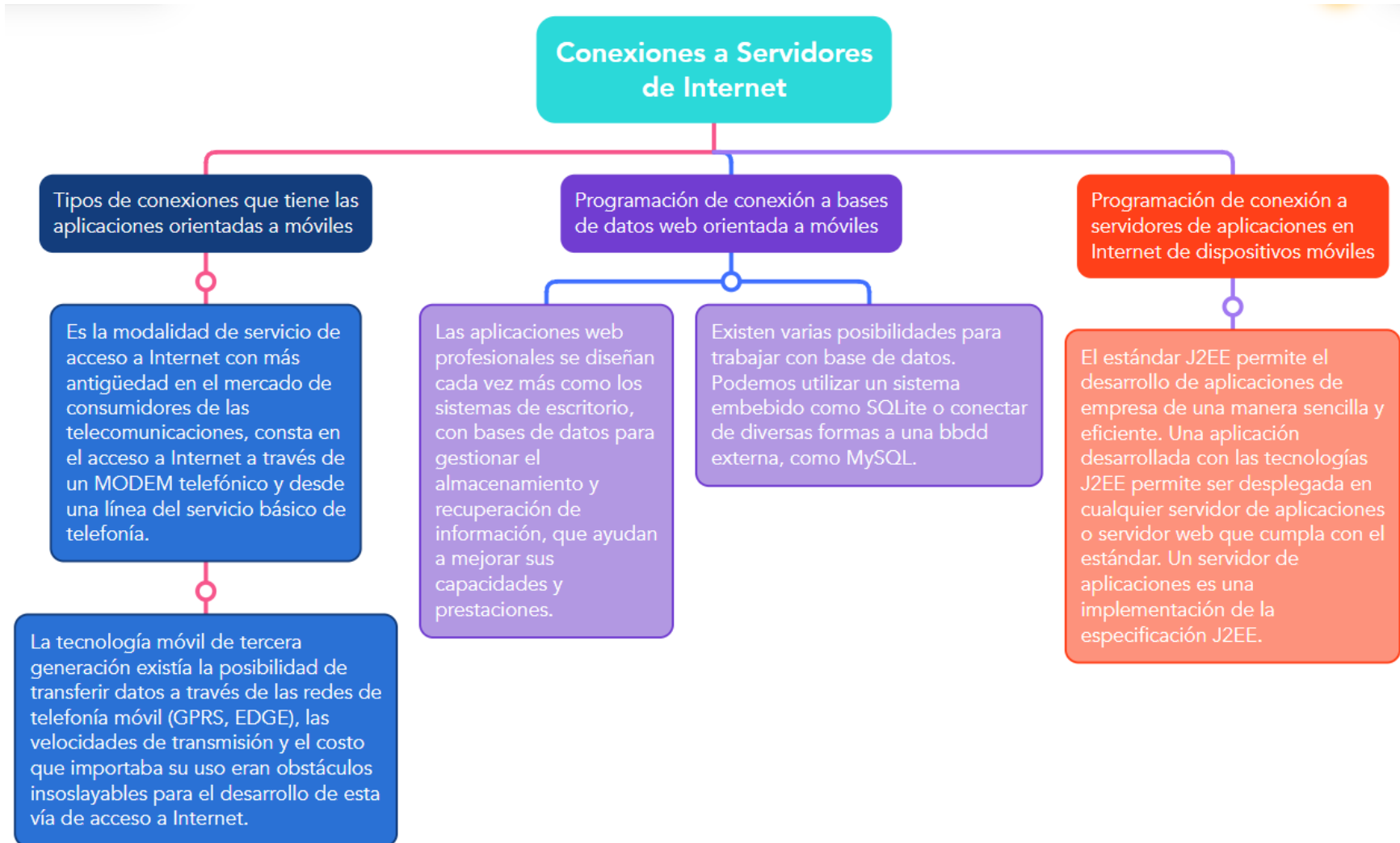
Prezi, 2015, 19 Noviembre, MODELO DE OBJETOS DE ACCESO DE DATOS,

<https://prezi.com/qt4zw0gspoez/52-modelo-de-objetos-de-acceso-de-datos/>

TAREA 2



TAREA 3



TAREA 4



TAREA 5

INTRODUCCIÓN

Los temas que son mencionados en este documento es para el gran interés que nosotros estudiantes o en el ámbito laboral se ha preguntado se ha preguntado sobre las aplicaciones móviles dependientes del contexto , sabemos bien que muchos ya lo saben pero no la mayoría pues aún tenemos una espinita en nuestra mente de cómo son y para qué son las aplicaciones móviles pues en contexto les puedo decir que el conocimiento de programación de aplicaciones para móviles ha pasado de manera muy rápida de ser un conocimiento más, a ser una necesidad debido a la rápida implantación y evolución de las plataformas móviles.

Esta rápida evolución crea incertidumbre sobre que tecnologías son las más adecuadas para la programación de móviles. Una de las arquitecturas más implantada es la proporcionada por el sistema Android. Este pequeño documento será una introducción a la programación de sistemas Android, pero también se hará énfasis, en plataformas de desarrollo multi-plataforma. Para poder dar herramientas de decisión sobre cómo sería necesario programar un cierto tipo de aplicaciones. El objetivo principal de esta acción formativa es el aprendizaje del desarrollo de aplicaciones Android y la interacción con los recursos del teléfono móvil.

También debemos de nosotros están conscientes que con IBM MobileFirst Platform Foundation, puede desarrollar aplicaciones móviles utilizando puede desarrollar aplicaciones móviles utilizando cualquiera de los cuatro enfoques: desarrollo web, desarrollo híbrido, desarrollo híbrido mixto o desarrollo nativo. Para no hacerlo tan exagerado IBM es una reconocida empresa multinacional estadounidense de tecnología y consultoría, pues esta empresa proporciona prestaciones que le ayudarán a responder al desarrollo acelerado de dispositivos móviles. Esta estructura flexible le proporciona más opciones en el momento de la implantación del canal de comunicación móvil o cuando libere una nueva versión de la aplicación, tanto si es una aplicación híbrida o nativa.

Esperemos que en este documento pueda resolver las dudas que tenemos, y como anexo no dejemos de actualizarnos ya que el mundo sigue en movimiento y cada día existe una nueva actualización.

Aplicaciones móviles dependientes del contexto

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

Contenido multimedia

El uso de formatos multimedia da lugar a un feed más atractivo y acogedor para el usuario. Las imágenes y vídeos como es obvio resaltar, deberán ir acordes a nuestro negocio al igual que la calidad deberá de ser impecable. En este caso lo que queremos lograr es que el usuario se encuentre cómodo dentro de nuestra app y que no se encuentre saturado ante tanto texto, al igual que tampoco debemos de sobrecargarlo con muchas imágenes y vídeos sin sentido.

Menú de navegación

Mediante un menú podremos agrupar la funcionalidad principal de la app, así como mejorar la eficiencia en la navegación de nuestro usuario. Este elemento se suele utilizar sobre todo cuando se tiene un perfil, para que pueda acceder a todo tipo de detalles.

Listas y detalle

Listas de productos, servicios o lugares es el componente principal que toda aplicación suele disponer ya que es lo que muestra el contenido de la misma. Pueden encontrarse en formato tradicional, es decir, cada elemento ocupa una línea o el tipo Grid, donde pueden aparecer varios en una misma línea. De igual importancia es la vista de detalle a la que accedemos a través de las listas y esta representa el feed de cada uno de los elementos mostrados con anterioridad, productos, servicios, etc. Es muy importante el nivel de personalización, ya que cuando un usuario llega hasta aquí es porque está más que interesado. Por ello, intentará buscar algo más que una simple descripción. Por ejemplo, en el caso de un producto o servicio querrá buscar cómo adquirirlo de una manera sencilla o en el caso de un bar le interesará ver su menú.

Splashscreen

Trata justo de la pantalla inicial que se abre cuando le damos a la app. No son muchos los que le dan importancia, pero si no se hace un uso correcto de la misma el desastre está asegurado. Este refleja la identidad visual de nuestra marca al igual que recuerda al usuario donde está. Como consejo, recomendamos que no dure más de 5 segundos, incluya o no animación, ya que si dura más puede provocar la salida del usuario. Al fin y al cabo, quienes entran van buscando algo concreto y no pararse a ver nuestra marca.

Servicio de atención al cliente

No por ir más abajo que el resto es menos importante, ni mucho menos. El servicio de atención al cliente es uno de los elementos principales que toda app debe tener, ya que, si en algún momento un usuario tiene algún problema o duda, debemos de saber resolverla de la mejor manera posible para lograr una óptima satisfacción.

Barra de herramientas

Esta permite hacer saber dónde nos encontramos en cada momento, así como facilitar la navegación hacia atrás y acceder a características comunes como son los ajustes. Esta barra no se debe de saturar con muchos elementos, tanto como por estética como por facilidad al usuario.

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.

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).

La geolocalización en dispositivos móviles

Cada vez son más los smartphones, wearables y múltiples dispositivos que ya incluyen GPS para poder determinar nuestra localización o ubicación sobre un mapa. De hecho, en la industria de las aplicaciones móviles existe una infinidad de posibilidades que ofrece la geolocalización y gracias a ello, hay muchas maneras en las que te puede ayudar esta funcionalidad desde un dispositivo móvil. 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. Sin lugar a duda, la geolocalización es 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 mí.
- 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.

Los componentes de la aplicación son bloques de creación esenciales de una aplicación para Android. Cada componente es un punto de entrada por el que el sistema o un usuario ingresan a tu aplicación. Algunos componentes dependen de otros.

Actividades

Es el punto de entrada de interacción con el usuario. Representa una pantalla individual con una interfaz de usuario. Por ejemplo, una aplicación de correo electrónico tiene una actividad que muestra una lista de los correos electrónicos nuevos, otra actividad para redactar el correo electrónico y otra actividad para leerlo.

Las actividades se implementan como subclases de la clase Activity. Para obtener más información sobre la clase Activity, consulta la guía para desarrolladores Actividades.

Servicios

Un servicio es un punto de entrada general que permite mantener la ejecución de una aplicación en segundo plano por diversos motivos. Es un componente que se ejecuta en segundo plano para realizar operaciones de ejecución prolongada o para realizar tareas de procesos remotos. Un servicio no proporciona una interfaz de usuario.

Receptores de emisiones

Un receptor de emisión es un componente que posibilita que el sistema entregue eventos a la aplicación fuera de un flujo de usuarios habitual, lo que permite que la aplicación responda a los anuncios de emisión de todo el sistema.

Un receptor de emisión se implementa como una subclase de BroadcastReceiver y cada receptor de emisión se entrega como un objeto Intent. Para obtener más información, consulta la clase BroadcastReceiver.

Proveedores de contenido

Un proveedor de contenido administra un conjunto compartido de datos de la aplicación que puedes almacenar en el sistema de archivos, en una base de datos SQLite, en la Web o en cualquier otra ubicación de almacenamiento persistente a la que tenga acceso tu aplicación.

Proceso de programación de aplicaciones dependientes de contexto en los móviles

La idea: ¿Qué va a hacer la app?

En la segunda de las fases de desarrollo de una aplicación móvil, se plantean con exactitud la dimensión que se desea para la app. Se deben tener en cuenta las necesidades de los usuarios o los problemas que se quieren solucionar. Partiendo de esas características, se pasará a definir un concepto determinado y el valor añadido que puede aportar.

Elegir al proveedor de desarrollo

Al momento de seleccionar al proveedor y la tecnología de la app, siempre recomendamos trabajar con una empresa que transmita confianza desde el primer momento, con quien os sintáis cómodos y que respire experiencia en desarrollos tecnológicos. Además, existen beneficios adicionales a tener en cuenta, como si realiza entregas parciales de la app o proporciona el código fuente al finalizar el desarrollo.

Análisis de los requerimientos

Tras una investigación del sector y de la competencia, es momento de definir el alcance del proyecto. En un trabajo en equipo junto con los desarrolladores se concreta cada funcionalidad de la app y se analizan todos los casos de uso. Pasa de lo general a lo concreto, lo qué se va a ofrecer en cada una de las pantallas. Con las funcionalidades bien definidas, la fase el desarrollo será más fluido, se tendrá un calendario de entregas periódicas realista y se evitarán sorpresas en mitad de la programación.

Quinta fase de desarrollo de una aplicación móvil: La planificación

Cualquier proyecto requiere de una fase de planning en la que se establecerán las directrices. Un calendario de trabajo que especifica la lista de acciones a realizar hasta el cierre del proyecto: el tiempo de desarrollo, las campañas de marketing, la subida a las tiendas o el lanzamiento de la app. Es conveniente avisar al desarrollador sobre la fecha del lanzamiento deseada, para coordinar la fecha de subida a las tiendas.

diseño gráfico de la aplicación

En esta fase de desarrollo de una aplicación móvil se definirán el contenido y las interacciones de la app acorde a la experiencia de usuario. Se elabora la propuesta visual de las pantallas, siguiendo la imagen de marca y las tendencias del diseño de apps, siempre pensando en ofrecer la mejor experiencia para los usuarios. Estos diseños se crearán mediante wireframes y prototipos, que serán la base para el desarrollador.

Desarrollo del software y creación de la app final

El desarrollador mediante código comienza a construir la aplicación móvil con el lenguaje de programación y tecnología indicados en la fase de análisis. Como hemos mencionado, recomendamos la existencia de entregas parciales y periódicas para que se vea la evolución del producto. Gracias a ello, se pueden corregir a tiempo los fallos que surjan según que avanza el desarrollo.

Mejorar los detalles: Testing o revisión de la app

En esta etapa se realizan una serie de pruebas que aseguren la calidad de la plataforma. QA (Quality Assurance) es un proceso de evolución y mejora continua donde se realizan acciones para comprobar que todas las acciones que pueda hacer un usuario dentro de la app funcionen correctamente en todos los dispositivos.

Fase de lanzamiento en el desarrollo de una aplicación móvil

Una vez se garantice la calidad, es momento de ofrecer la app a los usuarios. Se puede subir en las tiendas de aplicaciones como App Store y Google Play, por lo que se recomienda tener en cuenta el tiempo de publicación y aprobación de ambas para la fecha de lanzamiento de la app. También se puede dar a conocer mediante acciones de marketing, con landing pages, banners y publicidad o mejorando el posicionamiento ASO en las stores para su popularización.

Mantenimiento a largo plazo

Todo software se actualiza, por ejemplo, se van publicando nuevos sistemas operativos, cambian las políticas de las tiendas o se venden versiones superiores de los modelos de smartphones. Por ello, hay que contar con un proveedor que tras su lanzamiento continúe optimizando y manteniendo la aplicación móvil en el largo plazo.

Sensores embebidos en los dispositivos móviles

Elementos y características de los sensores embebidos en los dispositivos móviles

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.

Los componentes de un sistema embebido, son los siguientes, en la parte central se encuentra el microprocesador, microcontrolador, DSP, etc. La CPU o unidad que aporta la capacidad de cómputo del sistema, pudiendo incluir memoria interna o externa, un micro con arquitectura específica según los requisitos.

Las características básicas de los sistemas embebidos son las siguientes: -

Deben ser confiables.

La confiabilidad, en inglés reliability $R(t)$, es la probabilidad de que el sistema trabaje correctamente dado que está funcionando en $t=0$.

La mantenibilidad, en inglés Maintainability $M(d)$, es la probabilidad de que el sistema vuelva a trabajar correctamente d unidades de tiempo después de un fallo.

La disponibilidad, en inglés Availability $A(t)$, es la probabilidad de que el sistema esté funcionando en el tiempo t .

La seguridad informática: consiste en disponer de una comunicación confidencial y autenticada.

La creación de un sistema confiable debe ser considerada desde un comienzo, no como una consideración posterior.

Deben ser eficientes en cuanto a la energía, al tamaño de código, al peso y al costo.

Están dedicados a ciertas aplicaciones.

Interfaces de usuario dedicadas (sin ratón, keyboard y pantalla).

CONCLUSIÓN

Se obtuvieron grandes conocimientos en el manejo de las tecnologías relacionadas con un proyecto de este tipo. Se aprendieron conceptos importantes para el desarrollo de software dentro de las limitaciones típicas de un dispositivo móvil. También se obtuvieron conocimientos avanzados en el uso del lenguaje de programación y un buen manejo de las diferentes APIs que hacen posible desarrollar aplicaciones bajo la plataforma Android. La aplicación móvil desarrollada está enfocada a facilitar el acceso al contenido informativo del AEIRNNR de manera más rápida, atractiva y sobre todo evitándonos pérdidas de tiempo al obtener esta información.

A medida que la aplicación fue tomando forma, se implementaron funciones adicionales al diseño original. Esto sucedió ya que se notaron algunas posibles mejoras que no representaban grandes inconvenientes desde el punto de vista de implementación, pero otorgaban considerables mejoras en la usabilidad general de la aplicación. La realidad aumentada que nosotros hemos empleado en la búsqueda de lugares de interés en el AEIRNNR, también podría ser utilizada en otros campos como en la educación, en medicina, entre otros, ya que esta tecnología proporciona un aporte de información muy útil.

Se presentaron con claridad las definiciones de los términos relacionados con la computación móvil y cada uno de sus tres componentes: dispositivos móviles, aplicaciones móviles y redes de comunicación; esto en forma general y contextualizándolos al entorno empresarial. Es posible que, debido a la rápida evolución de la tecnología, estas definiciones puedan llegar a presentar variaciones en un futuro no muy lejano para ser adaptadas a un contexto empresarial cada vez más dinámico.

BIBLIOGRAFÍA

Occam Agencia de inbound marketing Madrid, (s / f), La relación de las aplicaciones con el contexto, <https://www.occamagenciadigital.com/blog/la-relacion-de-las-aplicaciones-con-el-contexto>

Osiris Montero Ríos, 2017, 05 Mayo, Aplicación móvil basada en el contexto, <https://www.redalyc.org/journal/5122/512253718010/html/>

SitemaApps, (s / f), aplicaciones móviles, <https://www.sitemapps.com/noticias/detalle/mobile-apps-future-api>

Jorge Ruiz, 2015, APLICACIONES DEPENDIENTES DE CONTEXTO, <http://jorge-ruiz.porexpertos.es/category/aplicaciones-dependientes-de-contexto>