INSTITUTO TECNOLÓGICO SUPERIOR DE JEREZ

4to Semestre

Ingeniería en Sistemas Computacionales

Tópicos Avanzados de Programación

Actividad: Mapa conceptual

Docente: I.S.C. Salvador Acevedo Sandoval

Alumno: Alan Osvaldo Guzmán Caldera

Correo Electrónico: alanosvaldo88@gmail.com

No. Control: S17070164

Jerez De García Salinas, Zac.

03/05/2019.

Cuestionario

1. **¿Cuál es el sufijo para las aplicaciones que se instalan en Android?**
   Un archivo de APK incluye todos los contenidos de una aplicación de Android y es el archivo que usan los dispositivos con tecnología Android para instalar la aplicación.

2. **¿Cuáles son los 4 componentes que forman a una aplicación Android?**
   Actividades: Representan una pantalla con interfaz de usuario, se implementa como una subclase de Activity.

   Servicios: Son componentes que se ejecutan en segundo plano para realizar operaciones prolongadas o tareas para procesos remotos. No proporciona una interfaz de usuario. Se implementa como una subclase de Service.

   Proveedores de contenido: Administran un conjunto de datos de la app. A través del proveedor de contenido, otras aplicaciones pueden consultar o incluso modificar los datos (si el proveedor de contenido lo permite). También son útiles para leer y escribir datos privados de tu aplicación y que no se comparten. Se implementa como una subclase de ContentProvider.

   Receptores de mensajes: Son componentes que responden a los anuncios de mensajes en todo el sistema. Comúnmente, un receptor de mensajes es simplemente una “puerta de enlace” a otros componentes y está destinado a realizar una cantidad mínima de trabajo. Se implementa como una subclase de BroadcastReceiver y cada receptor de mensajes se proporciona como un objeto Intent.

3. **¿Cómo se "activan" dichos componentes?**
   Actividades, servicios y receptores de mensajes se activan mediante un mensaje asincrónico llamado intent.
   El proveedor de contenido, no se activa mediante intents, sino a través de solicitudes de un ContentResolver.

4. **¿Qué es el archivo MANIFEST y para qué sirve?**
   Es un archivo de configuración donde podemos aplicar las configuraciones básicas de nuestra app. Sirve para:
   *Identificar los permisos de usuario que requiere la aplicación.
   *Declarar el nivel de API mínimo requerido por la aplicación en función de las API que usa la aplicación.
   *Declarar características de hardware y software que la aplicación usa o exige.

*Declarar los componentes de la aplicación.

5. **¿Cuáles son los estados en los que se puede encontrar una app?**
   Activa (Running): Está encima de la pila, lo que quiere decir que es visible y tiene el foco
   Visible (Paused): La actividad es visible pero no tiene el foco. Se alcanza este estado cuando pasa a activa otra actividad con alguna parte trasparente o que no ocupa toda la pantalla. Cuando una actividad está tapada por completo, pasa a estar parada.
   Parada (Stopped): Cuando la actividad no es visible. El programador debe guardar el estado de la interfaz de usuario, preferencias, etc.
   Destruida (Destroyed): Cuando la actividad termina al invocarse el método finish(), o es matada por el sistema.

6. **¿Cuáles son los métodos que permiten manipular dichos estados?**
   onCreate(Bundle): Se llama en la creación de la actividad. Se utiliza para realizar todo tipo de inicializaciones. Puede recibir información de estado de la actividad, por si se reanuda desde una actividad que ha sido destruida y vuelta a crear.
   onStart(): Nos indica que la actividad está a punto de ser mostrada al usuario.
   onResume(): Se llama cuando la actividad va a comenzar a interactuar con el usuario. Es un buen lugar para lanzar las animaciones y la música.
   onPause(): Indica que la actividad está a punto de ser lanzada a segundo plano. Es el lugar adecuado para detener animaciones, música o almacenar los datos que estaban en edición
   onStop(): La actividad ya no va a ser visible para el usuario.
   onRestart(): Indica que la actividad va a volver a ser representada después de haber pasado por onStop().
   onDestroy(): Se llama antes de que la actividad sea totalmente destruida

7. **¿Qué es y para qué sirve MATERIAL DESIGN?**
   Es una guía integral para el diseño visual, de movimientos y de interacción en distintas plataformas y dispositivos. Sirve para crear:
   - Un tema nuevo.
   - Nuevos widgets para vistas complejas.
   - Nuevas API para sombras y animaciones personalizadas.

8. **¿Cuáles son las 6 grandes pautas que especifica MATERIAL DESIGN para un buen diseño de apps?**

   Tema material: Te ofrece un nuevo estilo para tu aplicación, widgets del sistema que te permiten configurar la paleta de colores y animaciones predeterminadas para información táctil y transacciones de actividades.

   Listas y tarjetas: Android proporciona dos widgets para mostrar listas y tarjetas con estilos y animaciones de Material Design:

   - **RecyclerView:** Es una versión más acoplable de ListView que admite diferentes tipos de diseños y proporciona mejoras en el rendimiento.
   - **CardView:** Te permite mostrar extractos de información importante dentro de tarjetas que tienen apariencia y estilo coherentes.

   Visualización de sombras: Además de las propiedades X e Y, las vistas de Android poseen una propiedad Z.

   Animaciones: Las nuevas API de animaciones te permiten crear animaciones personalizadas para la información táctil en los controles de IU, además de realizar cambios en el estado de las vistas y transiciones entre actividades.

   Elementos de diseño: Permiten implementar aplicaciones de Material Design

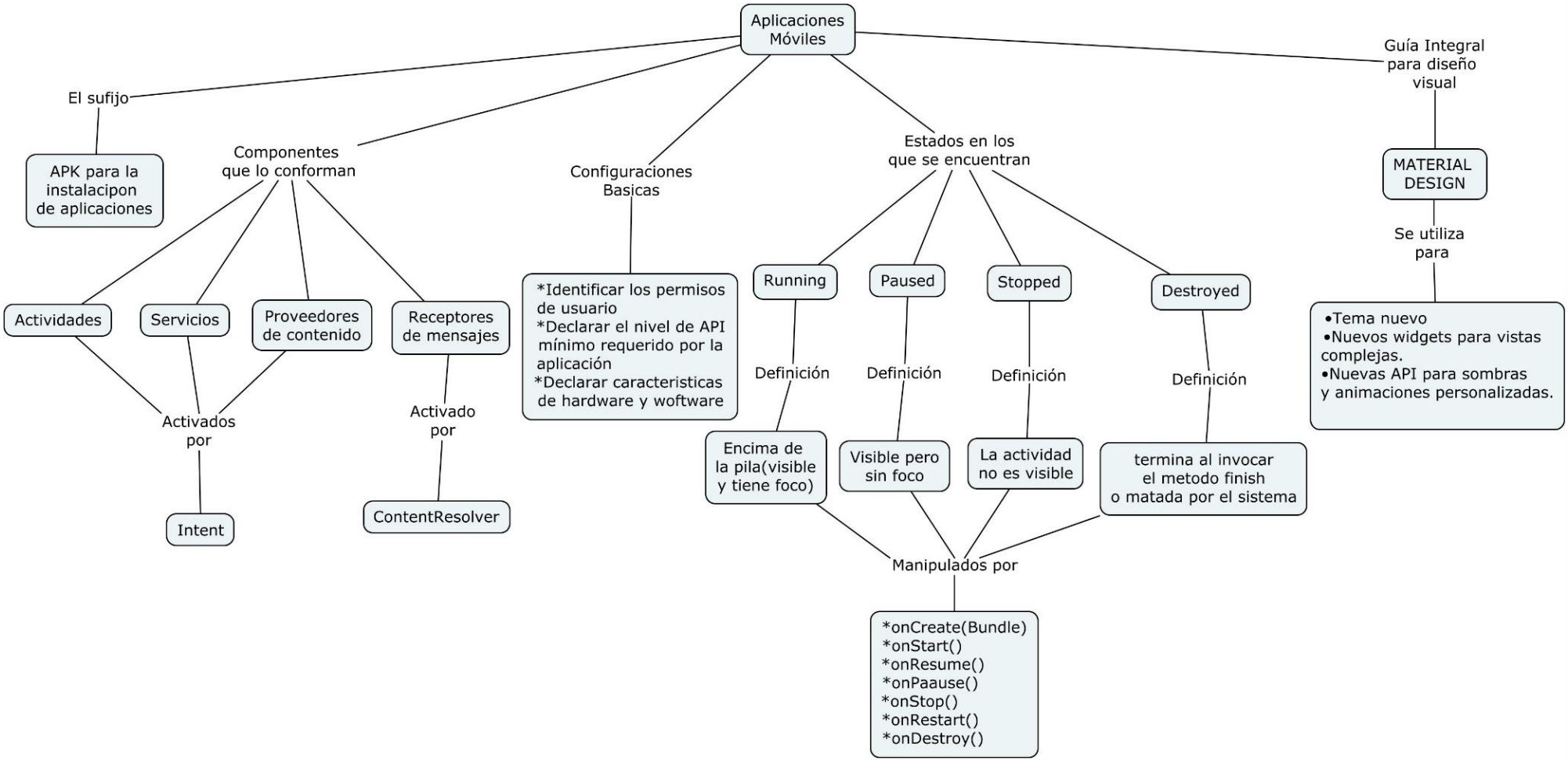
   - **Dibujables en vector.**
   - **Teñido de los dibujables.**
   - **Extracción de color.**

9. **Menciona 5 "mejores prácticas" indicadas por Google para el "desempeño" (performance) de la aplicación**

   - Identificar criterios de aceptación del performance.
   - Identificar el entorno de prueba
   - Plan de pruebas de performances.
   - Diseño de pruebas de performance.
   - Scripting.
   - Configuración del entorno de prueba.
   - Ejecutar pruebas de performance.
   - Analizar, afinar y volver a probar.

10. **Menciona 5 "mejores prácticas" indicadas por Google para la "Crear apps para miles de usuarios"**

    - Conectividad, contenido y capacidad.
    - Conectividad fluida en Android y en la Web.
    - Contenido correcto para el contexto indicado.
    - Optimizaciones para dispositivos.
    - Compilaciones eficientes y ligeras.
    - Reduce el consumo de batería.
    - Conserva el uso de datos.


Aplicaciones Móviles
El sufijo
APK para la instalacipon de aplicaciones
Componentes que lo conforman
Actividades
Servicios
Proveedores de contenido
Receptores de mensajes
Activados por
Intent
Activado por
ContentResolver
Configuraciones Basicas
*Identificar los permisos de usuario
*Declarar el nivel de API mínimo requerido por la aplicación
*Declarar caracteristicas de hardware y woftware
Estados en los que se encuentran
Running
Paused
Stopped
Destroyed
Definición
Definición
Definición
Definición
Encima de la pila(visible y tiene foco)
Visible pero sin foco
La actividad no es visible
termina al invocar el metodo finish o matada por el sistema
Manipulados por
*onCreate(Bundle)
*onStart()
*onResume()
*onPaause()
*onStop()
*onRestart()
*onDestroy()
Guía Integral para diseño visual
MATERIAL DESIGN
Se utiliza para
•Tema nuevo
•Nuevos widgets para vistas complejas.
•Nuevas API para sombras y animaciones personalizadas.

BIBLIOGRAFÍA

ander_gs. (2013). TuProgramacion. Recuperado el 01 de 05 de 2019, de TuProgramacion: http://www.tuprogramacion.com/glosario/que-es-el-android-manifest/

Developers. (s.f.). Aspectos fundamentales de la aplicación. Recuperado el 01 de 05 de 2019, de Aspectos fundamentales de la aplicación: https://developer.android.com/guide/components/fundamentals?hl=es-419

Developers, G. (s.f.). Cómo crear compilaciones para miles de millones de usuarios. Recuperado el 01 de 05 de 2019, de Cómo crear compilaciones para miles de millones de usuarios: https://developers.google.com/billions/?hl=es-419

Echeverria, D. (s.f.). Tis para mejorar el desempeño de aplicaciones Web. Recuperado el 01 de 05 de 2019, de Tis para mejorar el desempeño de aplicaciones Web: https://sg.com.mx/revista/58/tips-para-mejorar-el-desempeno-de-tus-aplicacionesweb

Valencia, U. P. (2017). Ciclo de vida de una actividad. Recuperado el 01 de 05 de 2019, de Ciclo de vida de una actividad: http://www.androidcurso.com/index.php/tutorialesandroid/37-unidad-6-multimedia-y-ciclo-de-vida/158-ciclo-de-vida-de-unaactividad