

Capítulo 3: UI CON FRAGMENTOS

Capítulo 4: **Services y Procesos Asincrónicos en Android**

Capítulo 5: Content Provider y Notificaciones



División de Alta Tecnología

4

Services y Procesos Asincrónicos en Android

Android Application Developer – Nivel Avanzado

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.




Objetivos

Al finalizar el capítulo, el alumno logrará:

- Comprender el beneficio de Services para procesos background.
- Comprender el uso de notificaciones.
- Comprender la diferencia del uso de Thread, Handler, AsyncTask para tareas asincrónicas.

3 - 2

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Agenda

- Services.
 - ¿Qué son los services?
 - Escenarios de uso.
- Técnicas de Implementación de Services.
 - AsyncTask, Handlers, Threads
- Broadcast y Receivers.
 - Patrón Publisher
 - Patrón Subscribe
 - Gestionando las notificaciones del teléfono
 - Boot, SMS, Battery Low, Air Plane, Call IN

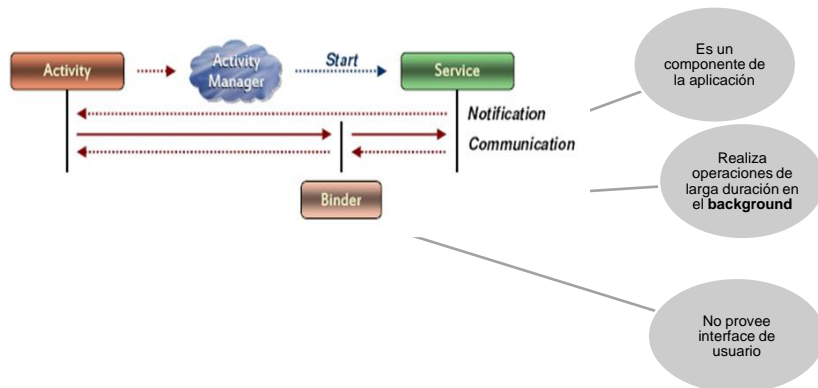
3 - 3

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Services

- ¿Qué son los services?



3 - 4

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Services

- **Escenarios de uso**

- El servicio se ejecutará en el main thread por defecto, lo recomendado es crear otro hilo para realizar operaciones de larga duración. Utilizar `IntentServices` , sino puede darse el caso de ANR (Application Not Responding).
- Es recomendable utilizarlos, solo cuando se tiene la necesidad de que siga ejecutándose aun cuando no haya interacción con el usuario, caso contrario se puede optar por un thread adicional.

3 - 5

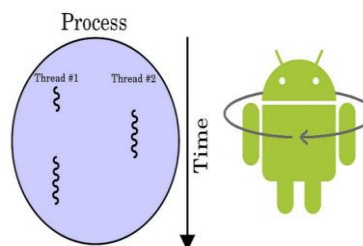
Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Técnicas de Implementación de Services

- **Thread, Handler, AsyncTask**

- Android modifica la UI y maneja los eventos con un thread, este es llamado main thread.
- Si no se usara concurrencia todo el código de Android se ejecutaría en un solo hilo, lo que conlleva que las sentencias se ejecuten una detrás de otra.
- Para proveer una buena experiencia al usuario, se debe tomar en cuenta las operaciones potenciales que podrían tomar un tiempo considerable, estas deberán ser asincrónicas.



3 - 6

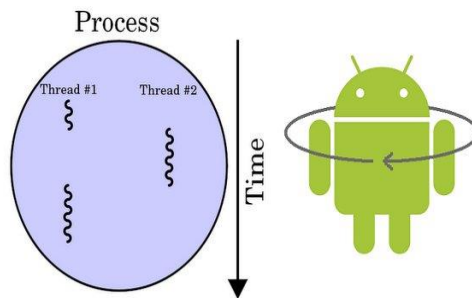
Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Técnicas de Implementación de Services

• Thread

- Android soporta el uso de la clase thread para realizar procesamiento asincrónico.
- Si necesitas actualizar la interface de usuario desde un nuevo hilo, se debe sincronizar con el main thread.



3 - 7

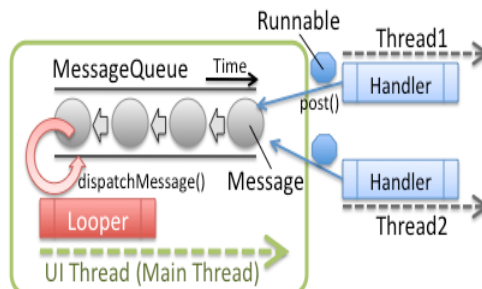
Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Técnicas de Implementación de Services

• Handler

- Android provee adicionales constructores para el manejo de concurrencia en comparación con el estándar de java.
- Se puede usar la clase android.os.Handler o la clase AsyncTask.
- La clase Handler se usa para registrar un hilo y brindar un canal para enviar data a este hilo.



3 - 8

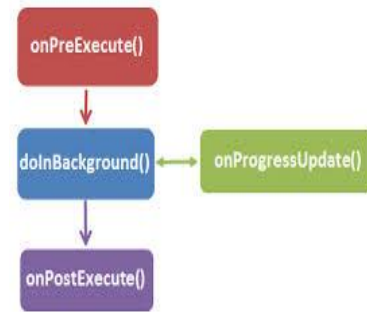
Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Técnicas de Implementación de Services

- **Asynctask**

- La clase Asynctask encapsula la creación de un proceso background y la sincronización con el main thread.
- Los parámetros son los siguientes: Asynctask <TypeOfVarArgsParams, ProgressValue, ResultValue>.
- Es inicializado por el método execute(), este llama a los métodos: doInBackground y onPostExecute()



3 - 9

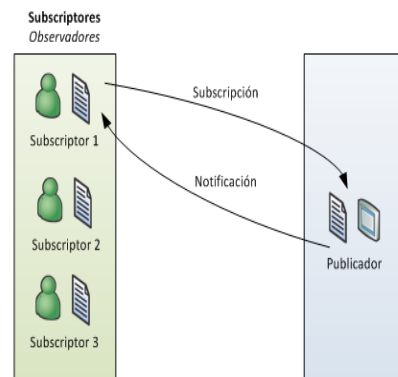
Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Broadcast y Receivers

- **Patron Publish/Subscribe**

- Broadcast receivers son la implementación de android del patrón de mensajería Publish/Subscribe, más precisamente es un patrón Observer.
- Aplicaciones (publisher) pueden generar broadcasts para enviar eventos, sin saber a quienes los envía.
- Receivers (subscribers).
- BroadcastReceiver es una pieza de código, con lo que una app se subscribe para ser notificada cuando un evento ocurre.



3 - 10

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Broadcast y Receivers

- **Gestionando las notificaciones del teléfono**
 - Un Broadcast Receiver es una especie de receptor de eventos que produce el sistema operativo android.
 - Algunos tipos de eventos que puede capturar:
 - Evento de Mensaje recibido:
`android.provider.Telephony.SMS_RECEIVED`
 - Evento de llamada recibida:
`android.intent.action.PHONE_STATE`
 - Evento de Modo Vuelo:
`android.intent.action.AIRPLANE_MODE`
 - Evento batería baja: `android.intent.action.BATTERY_LOW`
 - Evento de inicio de sistema operativo:
`android.intent.action.BOOT_COMPLETED`
 - Evento de desbloqueo de pantalla:
`android.intent.action.SCREEN_ON`

3 - 11

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Ejercicio N° 3: Utilizar los servicios del sistema para programar alarmas

Usar la clase `AlarmManager`.

Al finalizar el laboratorio, el alumno logrará:

- Usar la clase `AlarmManager`.

3 - 12

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Lecturas adicionales

Para obtener información adicional, puede consultar:

- Componentes Services
 - <http://developer.android.com/guide/components/services.html>
- Notificaciones y Alarmas
 - https://www.cs.umd.edu/class/fall2011/cmsc436/CMSC436/Lectures_Labs_files/NotificationsAndAlarms.pdf

3 - 13

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Resumen

En este capítulo, usted aprendió:

- Uso de Services para procesos background.
- Uso de la barra de notificaciones y sus diferentes componentes.
- Uso de handler y asyntask para procesamiento asincrónico.

3 - 14

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



Tarea N° 3: Realizar descarga de archivo de Google Drive (Agenda) desde aplicación Android

Realizar descarga de archivo de Google Drive (Agenda) desde aplicación Android.

Al finalizar la tarea, el alumno logrará:

- Implementar la tarea asíncrona más adecuada.

