



Programación de aplicaciones Android

Dia 4

Diálogos	3
AlertDialog	3
DatePickerDialog o TimePickerDialog	3
ProgressDialog	3
Lectura de códigos QR	4
Google Mobile Vision	4
ZXing	4
Volley	5
Creación de Requests	5
Referencias	8

I. Diálogos

Un diálogo es una pequeña ventana que le pide al usuario que tome una decisión o ingrese información adicional. Un cuadro de diálogo no llena la pantalla y normalmente se utiliza para eventos modales que requieren que los usuarios realicen una acción antes de que puedan continuar.

La clase **Dialog** es la clase base de los diálogos, pero se debe evitar instanciar **Dialog** directamente. En su lugar, se debe utilizar una de las siguientes subclases:

A. AlertDialog

Un cuadro de diálogo que puede mostrar un título, hasta tres botones, una lista de elementos seleccionables o un diseño personalizado.

B. DatePickerDialog o TimePickerDialog

Un diálogo con una interfaz de usuario predefinida que permite al usuario seleccionar una fecha o hora.

C. ProgressDialog

Google recomienda evitar el ProgressDialog y agregar un progress bar dentro del layout (ver <https://material.google.com/components/progress-activity.html>) en su lugar.

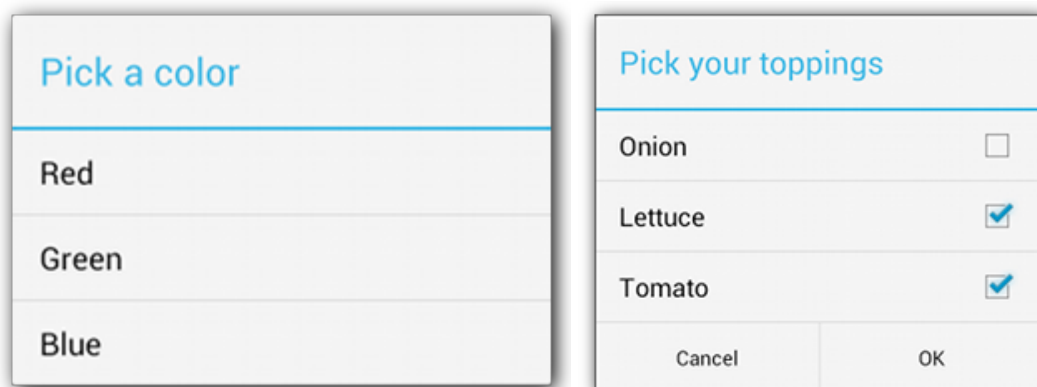


Figura 1. Ejemplos de diálogos.

Estas clases definen el estilo y la estructura de un diálogo, pero se debe utilizar un **DialogFragment** como un contenedor para el diálogo. La clase **DialogFragment** proporciona todos los controles que se necesitan para crear el diálogo y administrar su ciclo de vida.

II. Lectura de códigos QR

Android proporciona distintas opciones (bibliotecas) para leer códigos de barras o QR.

A. Google Mobile Vision

Biblioteca desarrollada por Google que provee herramientas de visión por computadora como detección de caras, lector de códigos QR, entre otros. Disponible en: <https://developers.google.com/vision/>

B. ZXing

Biblioteca de terceros para leer códigos de barras y QR. Por **sencillez** se va a utilizar está en este taller. Licencia Apache. Disponible en: <https://github.com/dm77/barcodescanner>

III. Volley

Para Android existen distintas opciones que permiten acceder a servicios REST. Entre las más usadas se encuentra Volley.

Volley es una biblioteca HTTP que facilita el acceso a redes en las aplicaciones de Android de una forma eficiente.

Volley ofrece los siguientes beneficios:

- Manejo automático de requests.
- Múltiples conexiones de red simultáneas.
- Priorización de requests.
- API para cancelación de requests.
- Herramientas de depuración y rastreo.
- Manejo automático de caché.

Entre otras opciones se encuentran:

- OkHttp (<http://square.github.io/okhttp/>)
- Spring (<http://projects.spring.io/spring-android/>)

A. Creación de Requests

En alto nivel, Volley se utiliza creando una cola (**RequestQueue**) y agregando requests (**Request**). Un **RequestQueue** crea y gestiona distintos hilos separados del hilo principal (o hilo de interfaz gráfica) para ejecutar las operaciones de red, leer y escribir en el caché y analizar las respuestas. Los **Request** hacen el análisis de las respuestas y Volley se encarga de enviar las respuestas analizadas al hilo principal.

Cuando se agrega un **Request** a un **RequestQueue** mediante el método *add()*, Volley ejecuta un hilo de procesamiento de caché y un grupo de hilos de distribución de red. Cuando se agrega un **Request** a la cola, éste es recogido por el hilo de caché y evaluado: si la solicitud puede ser atendida desde caché, la respuesta almacenada en caché se analiza en el subproceso de caché y se entrega en el hilo principal. Si el request no puede ser atendido desde caché, se coloca en la cola de la red. El primer hilo de red disponible toma la solicitud de la cola, realiza la transacción HTTP, analiza la respuesta en el subproceso de trabajo, escribe la respuesta en caché y envía la respuesta analizada al hilo principal.

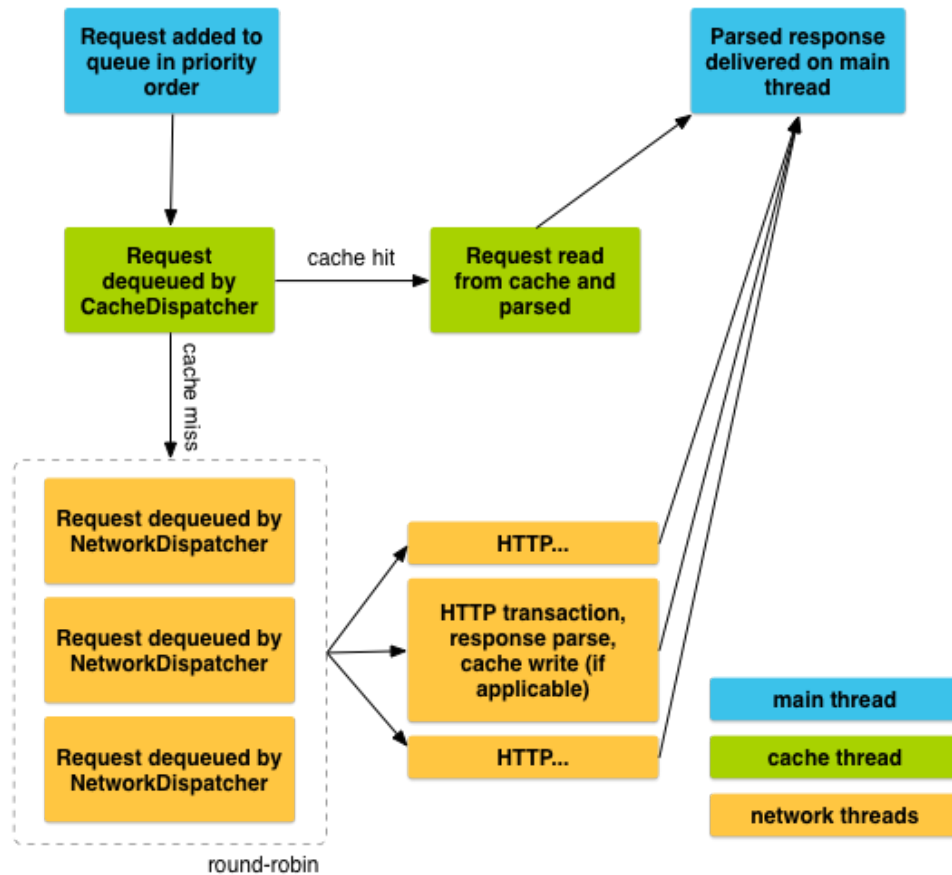


Figura 2. Proceso a alto nivel de un Request de Volley.

IV. Referencias

- *Portions of this page are reproduced from work created and [shared by the Android Open Source Project](#) and used according to terms described in the [Creative Commons 2.5 Attribution License](#).*
- *Portions of this page are modifications based on work created and [shared by the Android Open Source Project](#) and used according to terms described in the [Creative Commons 2.5 Attribution License](#).*
- *La lista completa de sitios se enlista a continuación:*
 - <https://developer.android.com>
 - <https://developer.android.com/guide/topics/ui/dialogs.html>
 - <https://developer.android.com/reference/android/app/DialogFragment.html>
 - <https://developer.android.com/training/volley/index.html>
 - <https://developer.android.com/training/volley/simple.html>