

Desarrollo de Aplicaciones para Dispositivos Móviles

Ingeniero Walter Medina
walter.medina@correounivalle.edu.co



Técnicas básicas para depurar una app

El debugging es una habilidad esencial para los desarrolladores de software, ya que permite encontrar y solucionar problemas en el código, mejorando así la calidad y el rendimiento de las aplicaciones. Hay dos formas principales para hacer debugging en android:

- **Usando Breackpoints:** Puedes establecer puntos de interrupción en tu código, conocidos como "breakpoints" que permiten examinar el estado actual de las variables, la pila de llamadas y otros aspectos importantes del programa.

```

44
45 private fun saveInvetory(){
46     val name = binding.etName.text.toString() name: "mouse"
47     val price = binding.etPrice.text.toString().toInt() price: 45000
48     val quantity = binding.etQuantity.text.toString().toInt() quantity: 10 binding: FragmentAd
49     val inventory = Inventory(name = name, price = price, quantity = quantity) name: "mouse" p
50     inventoryViewModel.saveInventory(inventory) inventory: Inventory(id=0, name=mouse, price=4500
51     Toast.makeText(context, text: "Artículo guardado !!", Toast.LENGTH_SHORT).show()
52     findNavController().popBackStack()
53
54 }
55

```

- **Usando el logcat:** Cada vez que hay problemas en la app, es necesario ver el logcat, es ahí donde se muestra la mayoría de errores. Pero también sirve para rastrear los log que se hayan colocado en el código

```

private fun saveInventory(){
    val name = binding.etName.text.toString()
    val price = binding.etPrice.text.toString().toInt()
    val quantity = binding.etQuantity.text.toString().toInt()
    val inventory = Inventory(name = name, price = price, quantity = quantity)
    inventoryViewModel.saveInventory(inventory)
    Log.d( tag: "test",inventory.toString())
    Toast.makeText(context, text: "Artículo guardado !!", Toast.LENGTH_SHORT).show()
    findNavController().popBackStack()
}

```

test

```

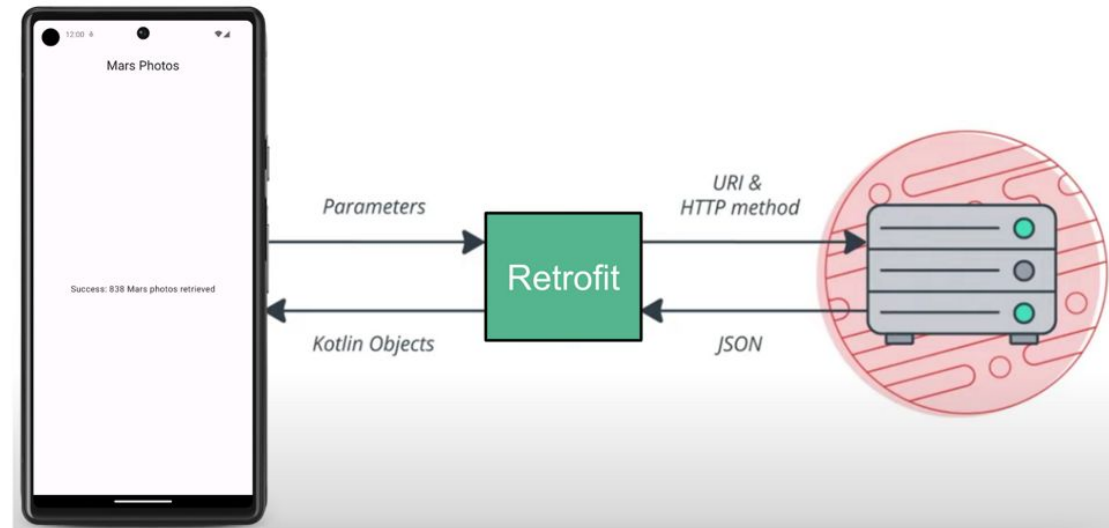
I [5483] wnc.h(68): Already at the latest configurations for experiment package com.google
I setting boot class path to /apex/com.android.runtime/javalib/core-oj.jar:/apex/com.and
D Inventory(id=0, name=zapatos , price=120000, quantity=12)
D handleUpdateState: mTargetEnable = falsemController.getBluetoothState() = 10
D handleUpdateState: mTargetEnable = falsemController.getBluetoothState() = 10
D handleUpdateState: mTargetEnable = falsemController.getBluetoothState() = 10

```



Consumo de API REST usando Retrofit

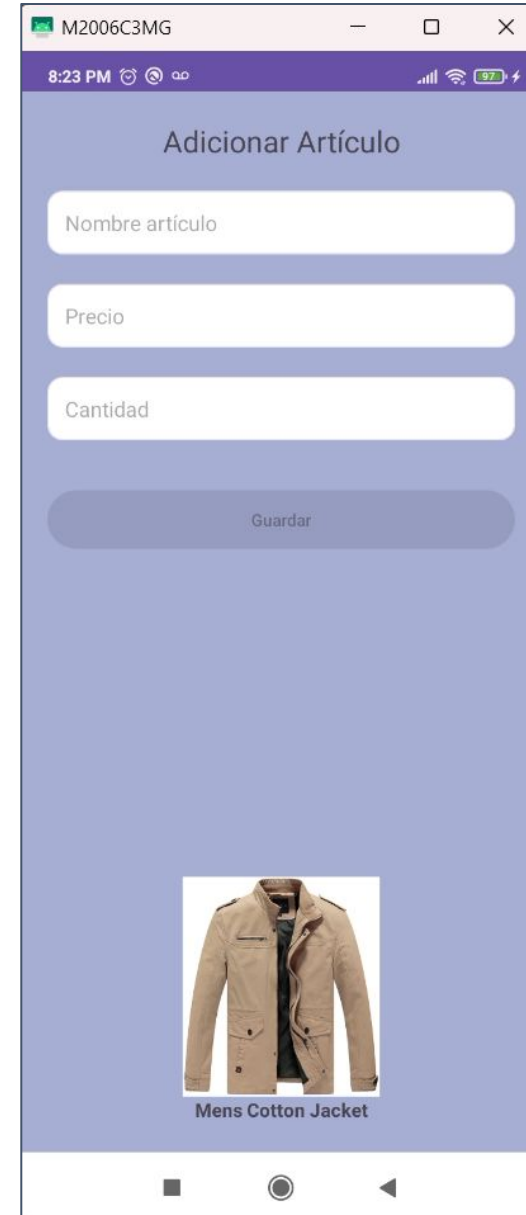
- **Arquitectura REST : Representational State Transfer** (transferencia de estado representacional). La mayoría de los servidores web ejecutan servicios web con este tipo de arquitectura.
- **Retrofit:** Es una biblioteca de Android que facilita el consumo de servicios web RESTful. Su principal objetivo es simplificar y agilizar el proceso de realizar solicitudes HTTP y procesar las respuestas en una aplicación Android



Ejercicio en Clase

Encuentra el error usando breakpoints al momento de consumir el servicio con retrofit, el cual permite mostrar la imagen del producto relacionado a continuación.

Actualmente si se lanza la aplicación se muestra el nombre del producto pero no su respectiva imagen.



GRACIAS !!

*“El esfuerzo vence al talento cuando el talento
no se está esforzando”.*
Anónimo



Bibliografía:

- ❖ <https://developer.android.com/studio/debug?hl=es-419>
- ❖ <https://square.github.io/retrofit/>
- ❖ <https://developer.android.com/codelabs/basic-android-kotlin-compose-getting-data-internet?hl=es-419#0>

