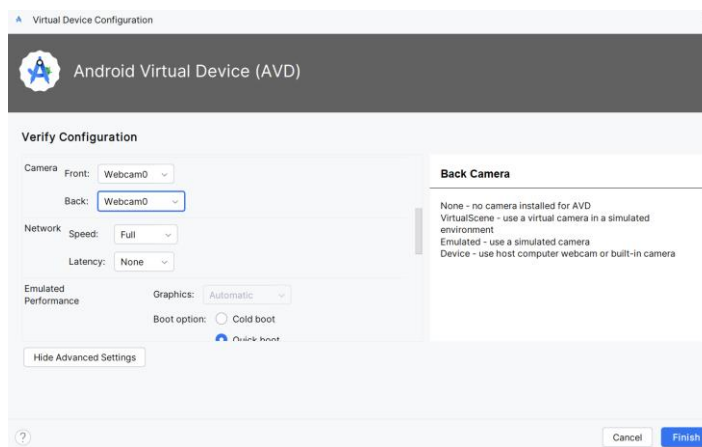


## HACER FOTOS

En esta guía aprenderemos a acceder a la cámara y a la galería desde una app en Android.

1. Crea un proyecto: File→New Project → Empty Views Activity , lo llamaremos, por ejemplo, **HacerFotos**
2. Lo primero que, recomiendo, es si estamos utilizando el emulador, es **Editar** sus propiedades y en **Show Advanced Settings→Camera**, marcar la opción **Webcam0**. Puedes configurar esto en la frontal o en la trasera.



3. Para acceder a la cámara se deben agregar los siguientes permisos en el **AndroidManifest.xml**.
  - CAMERA: Este permiso es esencial para acceder a la cámara del dispositivo y tomar fotos o videos.
  - WRITE\_EXTERNAL\_STORAGE (opcional): Si quieres guardar las fotos o videos capturados en el almacenamiento externo del dispositivo (como la tarjeta SD), necesitas este permiso. Sin embargo, si solo quieres mostrar la imagen o el video en tu app sin guardarlo, no es necesario

Debes agregar las siguientes líneas dentro del elemento `<manifest>` :

```
<uses-feature
    android:name="android.hardware.camera"
    android:required="false" />

<uses-permission android:name="android.permission.CAMERA" />
<uses-permission
    android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
```

Estos permisos que hemos añadido en el [AndroidManifest.xml](#) da permiso a la app, pero luego el usuario debe aceptarlas, siempre tiene la última palabra, esto significa que, al abrir la cámara, debes pedirle al usuario permiso para usar la cámara y el almacenamiento externo mientras la app se está ejecutando.

Por tanto, al pulsar el botón “Abrir” la primera vez se pedirá permiso para acceder a la cámara, si el permiso es concedido se abrirá la cámara, desde donde se hará una foto y esa foto será cargada en el ImageView.

```
//Activity para abrir la camara
val openCamera =
registerForActivityResult(ActivityResultContracts.StartActivityForResult()) { result ->
    if (result.resultCode == RESULT_OK) {
        val data = result.data!!
        bitmap = data.extras!!.get("data") as Bitmap
        binding.img.setImageBitmap(bitmap)
    }
}

//Activity para pedir permisos de cámara
//Usamos registerForActivityResult para recuperar los datos devueltos
val requestCameraPermissionLauncher =
registerForActivityResult(ActivityResultContracts.RequestPermission())
{
    isGranted: Boolean ->
    if (isGranted) {
        val intent = Intent(MediaStore.ACTION_IMAGE_CAPTURE)
        //si quisieras vídeo.
        //val intent = Intent(MediaStore.ACTION_VIDEO_CAPTURE)
        openCamera.launch(intent)
    } else {
        Log.e("SAR", "Permiso de cámara no concedido")
    }
}
binding.openCamera.setOnClickListener {

requestCameraPermissionLauncher.launch(android.Manifest.permission.CAMERA)
}
```

Al pulsar el botón “Guardar” se usará el nombre que el usuario ha puesto en el EditText, y se guardará en la galería de imágenes.

```
binding.saveToGallery.setOnClickListener {
    saveImageToGallery(binding.txtImagen.text.toString())
}

private fun saveImageToGallery(filename:String = "my_image.jpg"): Uri?
{
    val filepath = Environment.DIRECTORY_PICTURES
    val contentValues = ContentValues().apply {
        put(MediaStore.Images.Media.DISPLAY_NAME, filename)
        put(MediaStore.Images.Media.MIME_TYPE, "image/jpeg")
        put(MediaStore.Images.Media.RELATIVE_PATH, filepath)
    }
    val contentResolver = contentResolver
    val uri =
contentResolver.insert(MediaStore.Images.Media.EXTERNAL_CONTENT_URI,
contentValues)

    try {
        val outputStream = contentResolver.openOutputStream(uri!!)
        if (outputStream != null) {
            bitmap.compress(Bitmap.CompressFormat.JPEG, 100,
outputStream)
        }
        outputStream?.close()
    } catch (e: IOException) {
        e.printStackTrace()
        Toast.makeText(this, "Se ha producido algún error al almacenar
la imagen en la galería.",
            Toast.LENGTH_SHORT).show()
        return null
    }
    Toast.makeText(this, "Imagen almacenada en la galería",
Toast.LENGTH_SHORT).show()
    binding.txtImagen.setText("")
    return uri
}
```

Al pulsar en el botón “Galería”, se accede a la galería de imágenes. Al hacer clic sobre una imagen de la galería, se cierra la galería y la imagen se carga en el ImageView.

```
//Activity para lanzar la galería de imágenes.
//Usamos registerForActivityResult para recuperar los datos devueltos
val pickMedia =
registerForActivityResult(ActivityResultContracts.PickVisualMedia()) {
uri ->
    // Callback is invoked after the user selects a media item or
    closes the
    // photo picker.
    if (uri != null) {
        Log.d("SAR", "Selected URI: $uri")
        binding.img.setImageURI(uri)
        Log.d("SAR", "Cargada")
    } else {
        Log.d("SAR", "No media selected")
    }
}
binding.openGallery.setOnClickListener{

pickMedia.launch(PickVisualMediaRequest(ActivityResultContracts.PickVi
sualMedia.ImageOnly))
    //si quisieras que se vean imágenes y vídeos de la galería.
    //
pickMedia.launch(PickVisualMediaRequest(ActivityResultContracts.PickVi
sualMedia.ImageAndVideo))
}
```