

A close-up, slightly angled view of a smartphone screen displaying a home screen with various app icons. The icons include App Store, Notes, Clock, Reminders, Facebook, WhatsApp, Uber, Settings, Messenger, Google Maps, Gmail, Zoho Mail, Paytm, Calcbot, My Airtel, B'days, Day Stars, Music, Instagram, Buffer, Hangouts, and a dock with Phone, Chrome, and Messages. The background is a light blue gradient with a faint circular pattern.

# Curso de Introducción al Desarrollo Android





# Alberto Luengo

## Estudios

Técnico Superior en Administración de Sistemas Informáticos

Técnico Superior en Desarrollo de Aplicaciones Web

## Experiencia

Android (Kotlin y Java)

iOS (Swift)

MEAN

PHP

Flutter y React Native, algún día

## Intereses

Informática

Videojuegos

Deporte (lo justo)

Series y películas

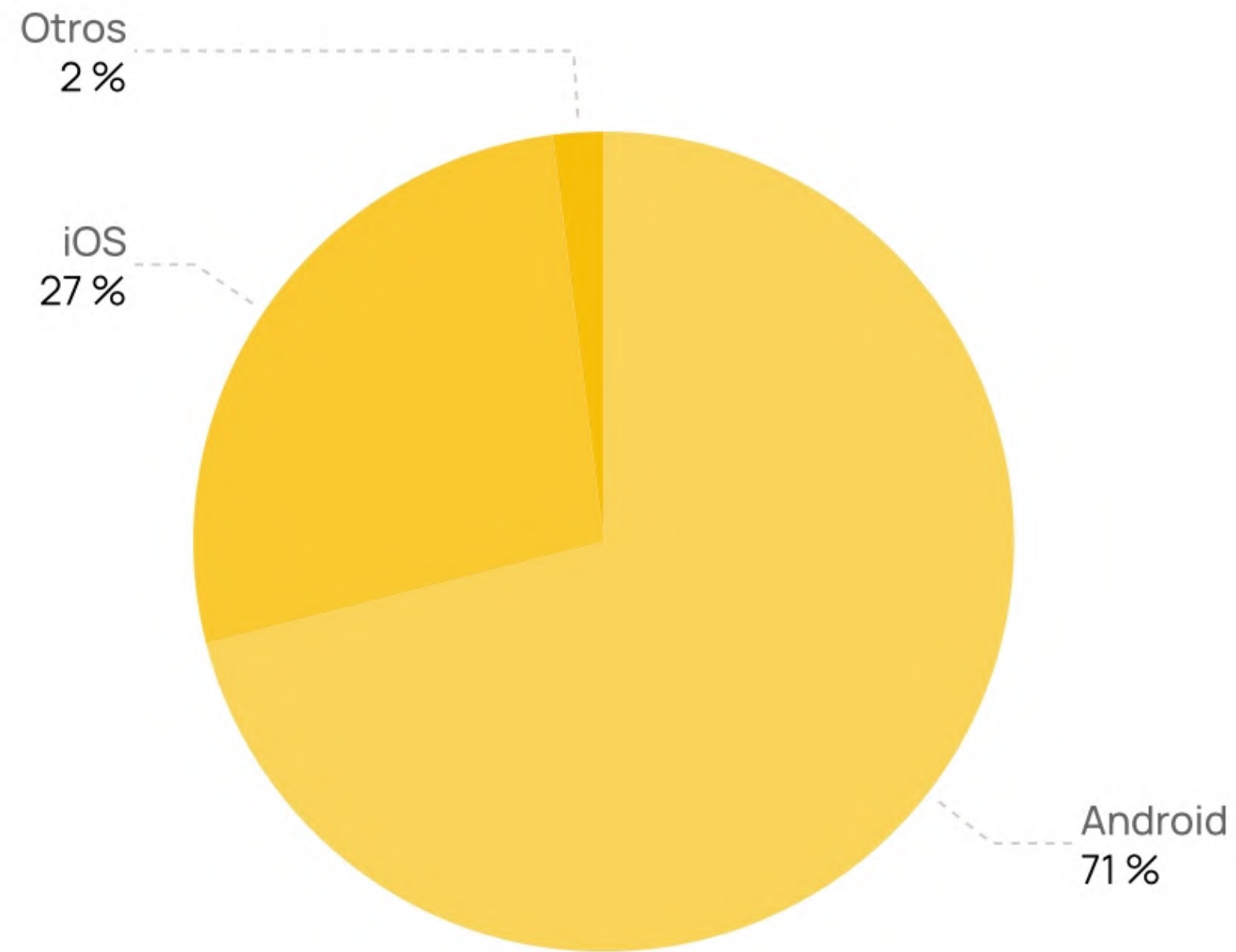
Música

Viajar (ya no tanto)

# El Curso

# SISTEMAS OPERATIVOS, EL MERCADO ACTUAL

Datos aproximados según [statcounter.com](https://www.statcounter.com) en 2022





# ¿Por qué ser programador de apps?

- El 98% del mercado móvil utiliza Android o iOS
- Mucha demanda con diversas salidas (no solo programador)
- Buenos salarios y crecimiento continuo
- Posibilidad de trabajo en remoto
- Resolver problemas del día a día



# METODOLOGÍA

INTRODUCCIÓN A LA  
PROGRAMACIÓN

Aprenderás los conceptos básicos de la programación en cada lenguaje

TEORÍA A BASE DE  
PROYECTOS  
PRÁCTICOS

Conocerás la teoría a base del desarrollo de ejercicios prácticos

DESARROLLO DE  
PROYECTOS

Realizarás varios proyectos en clase, guiados y que servirán para tu portfolio y experiencia

DESARROLLO DE  
PROYECTOS

Programarás una app en Android como proyecto final para la obtención del título



# Android

EL SISTEMA OPERATIVO DE GOOGLE





ANDROID

**Android** – es el sistema operativo más importante actualmente, la mayoría de dispositivos móviles funcionan con **Android** y es desarrollado por **Google**.



# ¿Donde se utiliza?

Móviles y Tablets

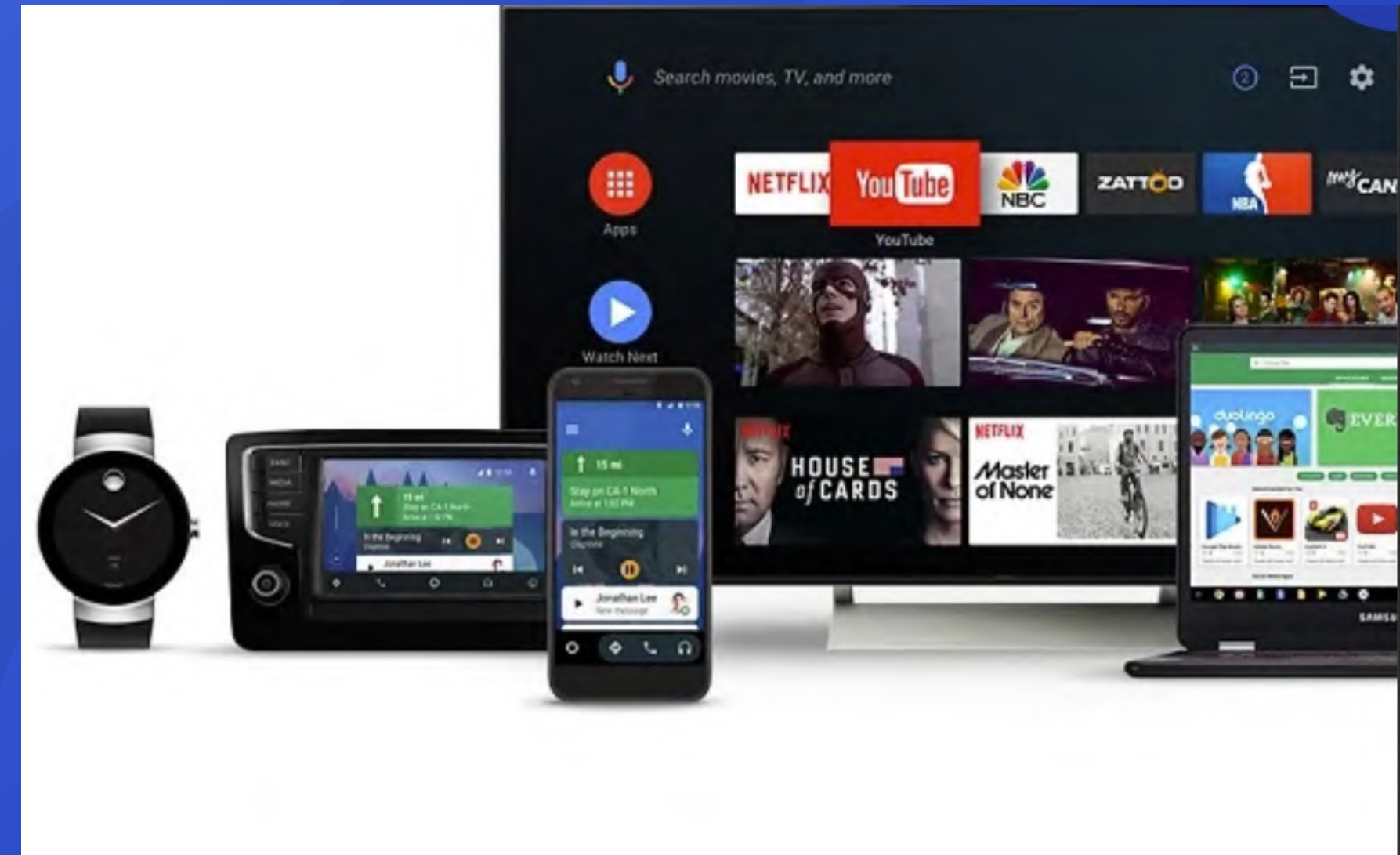
TV

Android Wear

Android Auto

Android Things

Chrome OS



# Versiones de Android

ANDROID PLATFORM VERSION	API LEVEL	CUMULATIVE DISTRIBUTION	S
4.1 Jelly Bean	16		<b>New features</b> Material You Redesigned widgets  Game Mode Rich content insertion AppSearch API Compatible media transcoding Improved notifications  <b>Behavior changes</b> Performance Classes Stretch overscroll App splash screens Restricted App Standby Bucket Improved refresh rate switching Passpoint updates  <b>Security and privacy</b> App hibernation Nearby device permissions Approximate location Bluetooth permissions Permission group lookup Clipboard access notifications Permission package visibility
4.2 Jelly Bean	17	99,9%	
4.3 Jelly Bean	18	99,7%	
4.4 KitKat	19	99,7%	
5.0 Lollipop	21	98,8%	
5.1 Lollipop	22	98,4%	
6.0 Marshmallow	23	96,2%	
7.0 Nougat	24	92,7%	
7.1 Nougat	25	90,4%	
8.0 Oreo	26	88,2%	
8.1 Oreo	27	85,2%	
9.0 Pie	28	77,3%	
10. Q	29	62,8%	
11. R	30	40,5%	
12. S	31	13,5%	

<https://developer.android.com/about/versions/12>



# ¿Quien conoce Java?

# ¿Quien conoce Kotlin?



# Kotlin - Como lenguaje principal

Desde 2017, Kotlin ha sido el lenguaje oficial de Android por defecto.

Actualmente la mayoría de ofertas de empleo solamente piden Kotlin como lenguaje de programación para Android.

El lenguaje oficial anterior era Java, y por ello también Kotlin es compatible con funciones o variables de Java.

Es un lenguaje potente, rápido y ligero. Una curva de aprendizaje más rápida y sencilla que Java. Es interoperable, compatible con Java al 100%.

```

1  package es.sublimestudio.a0demo2
2
3  import ...
4
5
6
7  class MainActivity : AppCompatActivity() {
8
9      override fun onCreate(savedInstanceState: Bundle?) {
10         super.onCreate(savedInstanceState)
11         setContentView(R.layout.activity_main)
12     }
13 }
14
15
16

```



# C y C++ (NDK)



- Mejor rendimiento
- Más complejidad
- Acceso a hardware





# XML - Recursos, vistas

XML no es un lenguaje de programación como tal. Pero nos sirve para esquematizar las vistas y recursos de nuestra app

Ejemplo de textos traducibles

```
<resources>
  <string name="app_name">App de Prueba</string>
  <string name="welcome_message">Bienvenido</string>
</resources>
```



# Gradle (DSL)

Permite la automatización de compilación de código

Los script se escriben utilizando

Groovy o Kotlin DSL (Domain Specific Language)



# Gradle

```

1  plugins {
2      id 'com.android.application'
3      id 'org.jetbrains.kotlin.android'
4  }
5
6  android {
7      compileSdk 32
8
9      defaultConfig {
10         applicationId "es.sublinestudio.a0demo2"
11         minSdk 28
12         targetSdk 32
13         versionCode 1
14         versionName "1.0"
15
16         testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
17     }
18
19     buildTypes {
20         release {

```



# Donde obtener ayuda

Webs de ayuda que utilizan todos los programadores cuando se atascan:

[Kotlin en StackOverflow](#)

[Kotlin en Quora](#)

[Kotlin en reddit.](#)

[Kotlin en Medium](#)

[Kotlin Playground](#)

[Kotlin Examples](#)



# Gracias.

# ¿Preguntas?