College of Science and Technology

Royal University of Bhutan

Unit I: Introduction to Mobile App Development

CTE308-AS2024

Tutor: Pema Galey

#17682761

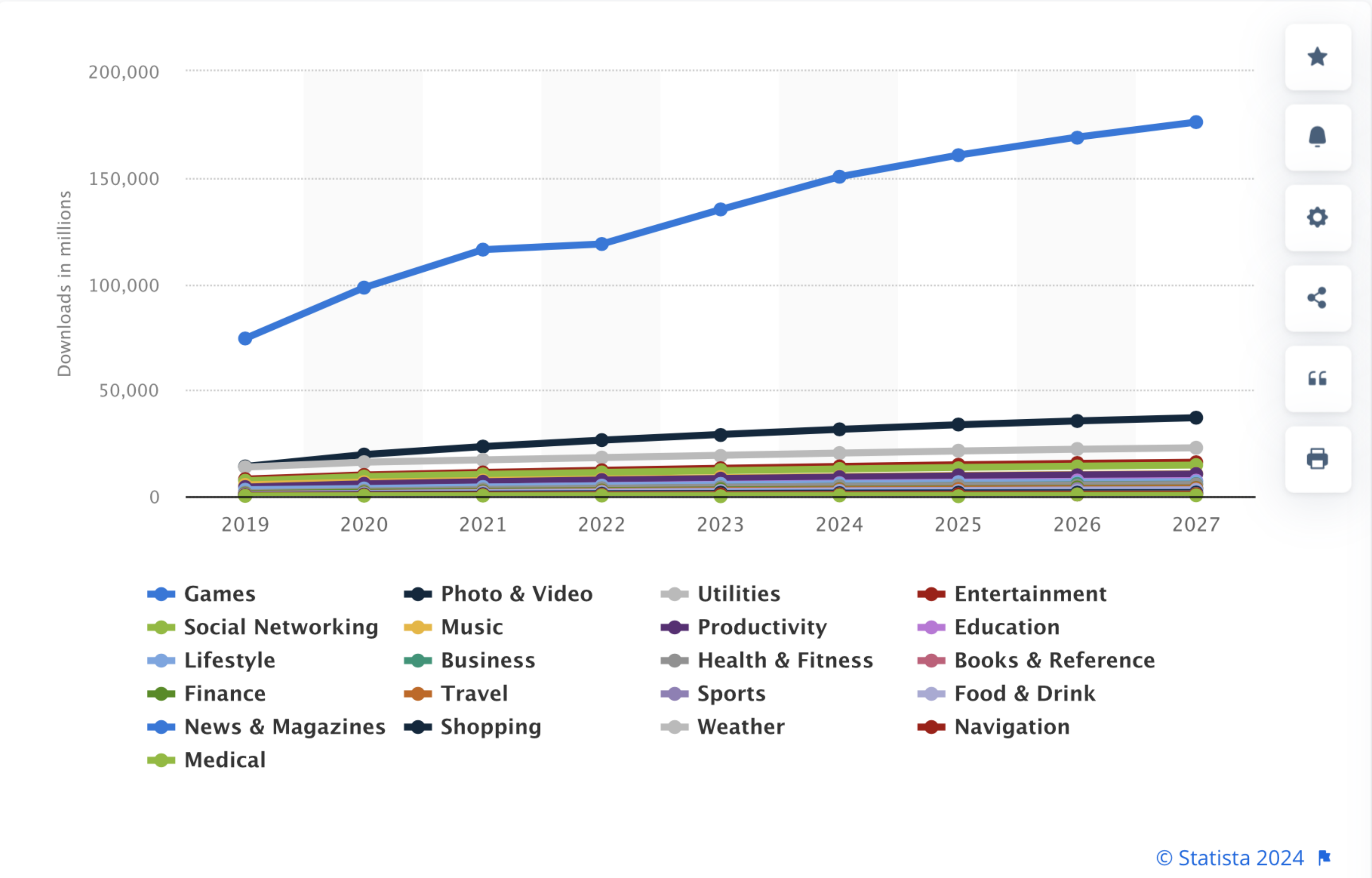
Overview

- Overview of Mobile Development
- Introduction to mobile platforms (Android, iOS).
- Key differences between native and cross-platform development.
- Introduction to cross-platform frameworks.

Introduction

- Mobile development is the process of creating software applications that run on mobile devices such as smartphones and tablets. These applications can be pre-installed on devices during manufacturing or delivered as web applications using server-side or client-side processing.
- The goal is to utilize the device's features, such as GPS, camera, and touch functionality, to provide a rich user experience.

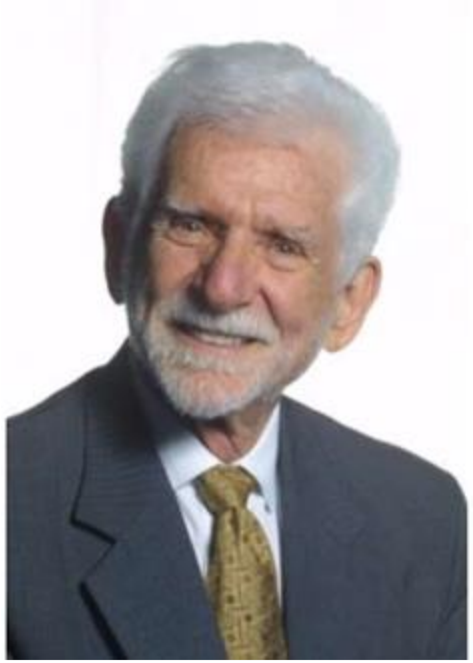
- Number of mobile app downloads worldwide from 2019 to 2027, by segment (in million downloads)
- The indicator 'Downloads' is forecast to experience significant growth in all segments in 2027.



© Statista 2024

History

- Martin Cooper – Father of the mobiles (cell phone)
- Born in December 26, 1928 in Chicago, United States.
- IBM released PDA on 16th Aug. 1994 and Stopped production in 1995 Feb. which weighs about 1.4 kilograms



Operating Systems

- An interface between hardware and user.
- It manages hardware and software resources of the systems.
- The OS which controls mobile devices is called Mobile OS.
  - ✓ They are simple and deal with the wireless versions of broadband and local connectivity.

Mobile Operating Systems

- It is an operating system that is specifically designed to run on mobile devices such as:
  - ✓ Mobile phones
  - ✓ Smartphones
  - ✓ PDAs
  - ✓ Tablet computers and
  - ✓ other handheld devices.

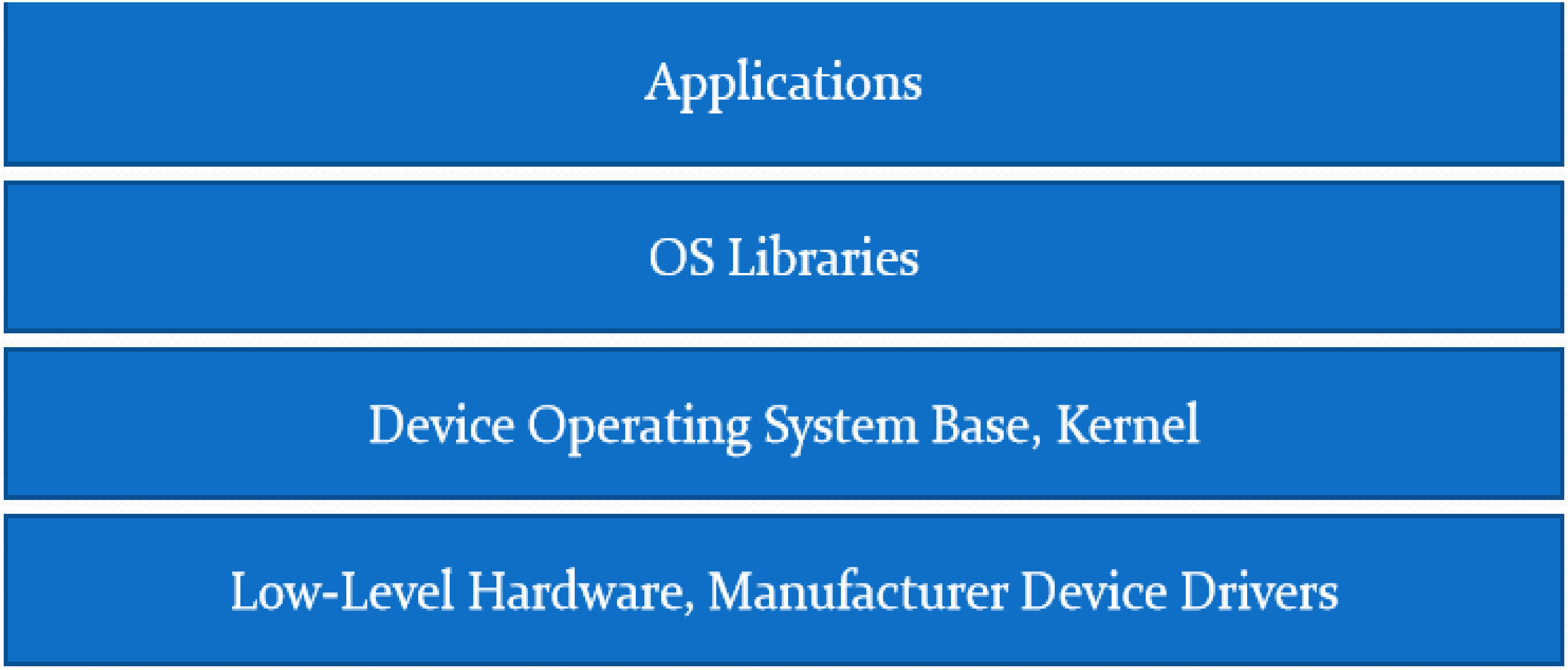
Mobile Operating Systems

- Design and capabilities of a Mobile OS is very different than a general purpose OS running on desktop machines:
  - ✓ Mobile devices have constraints and restrictions on their physical characteristic such as screen size, memory, processing power etc.
  - ✓ Scarce availability of battery power.
  - ✓ Limited amount of computing and communication capabilities.

Mobile Operating Systems

- Therefore, they need different types of operating systems depending on the capabilities they support.
  - ❖ Example: PDA OS is different from a Smartphone OS.
- Operating System is a piece of software responsible for management of operations, control, coordinate the use of the hardware among the various application programs, and sharing the resources of a device

Mobile OS Structure



Applications

OS Libraries

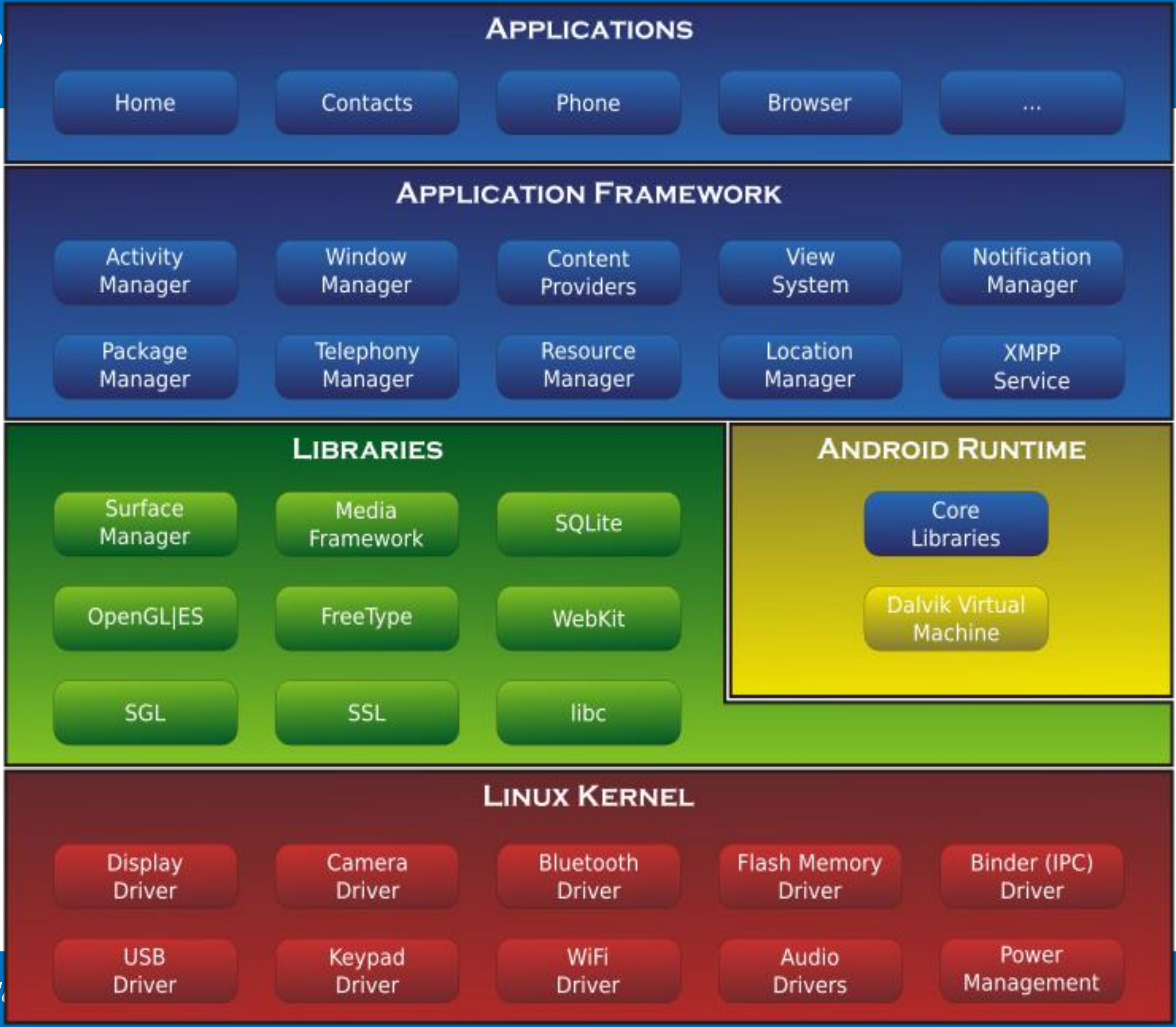
Device Operating System Base, Kernel

Low-Level Hardware, Manufacturer Device Drivers

Types of Mobile OS

- There are many Mobile Operating Systems. The most popular Mobile OS are:
  - ✓ Android OS
  - ✓ iPhone OS/iOS
  - ✓ HarmonyOS
  - ✓ Java ME Platform
  - ✓ BlackBerry
  - ✓ Symbian OS
  - ✓ Windows Mobile

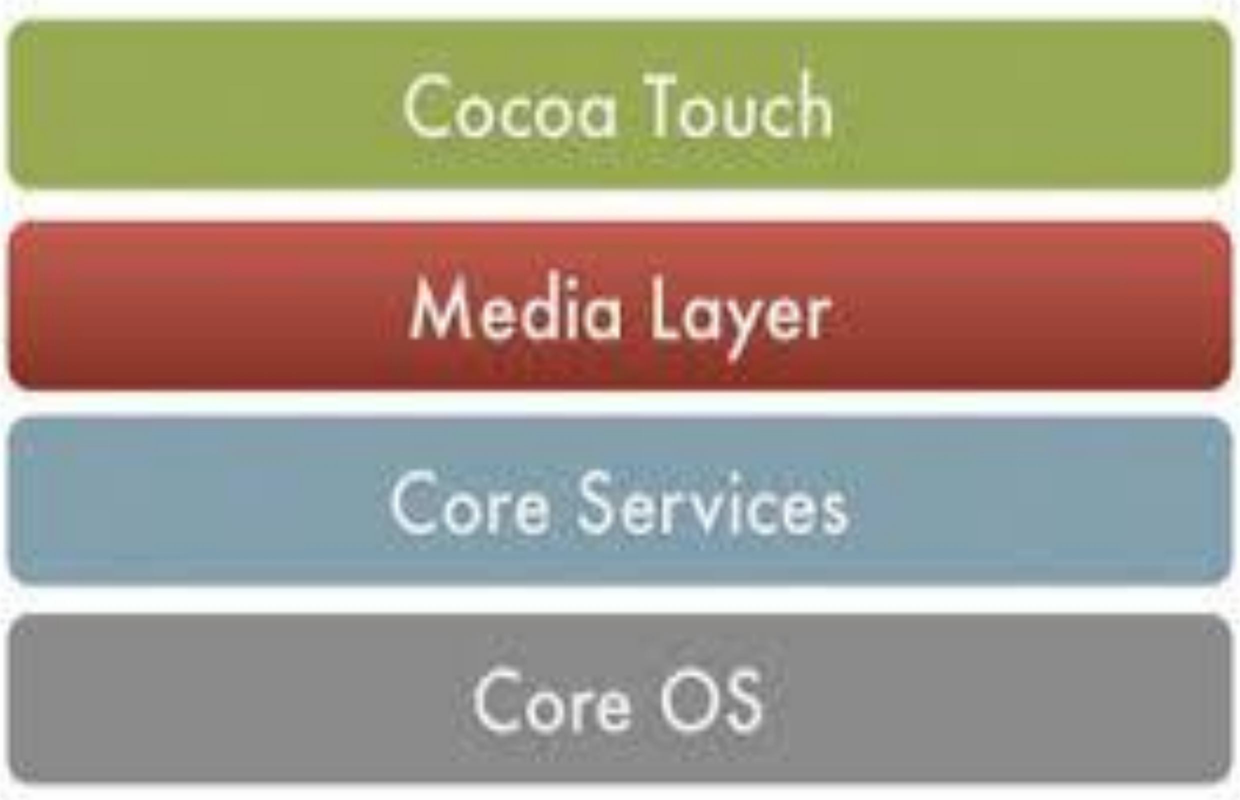
Android
OS Structure



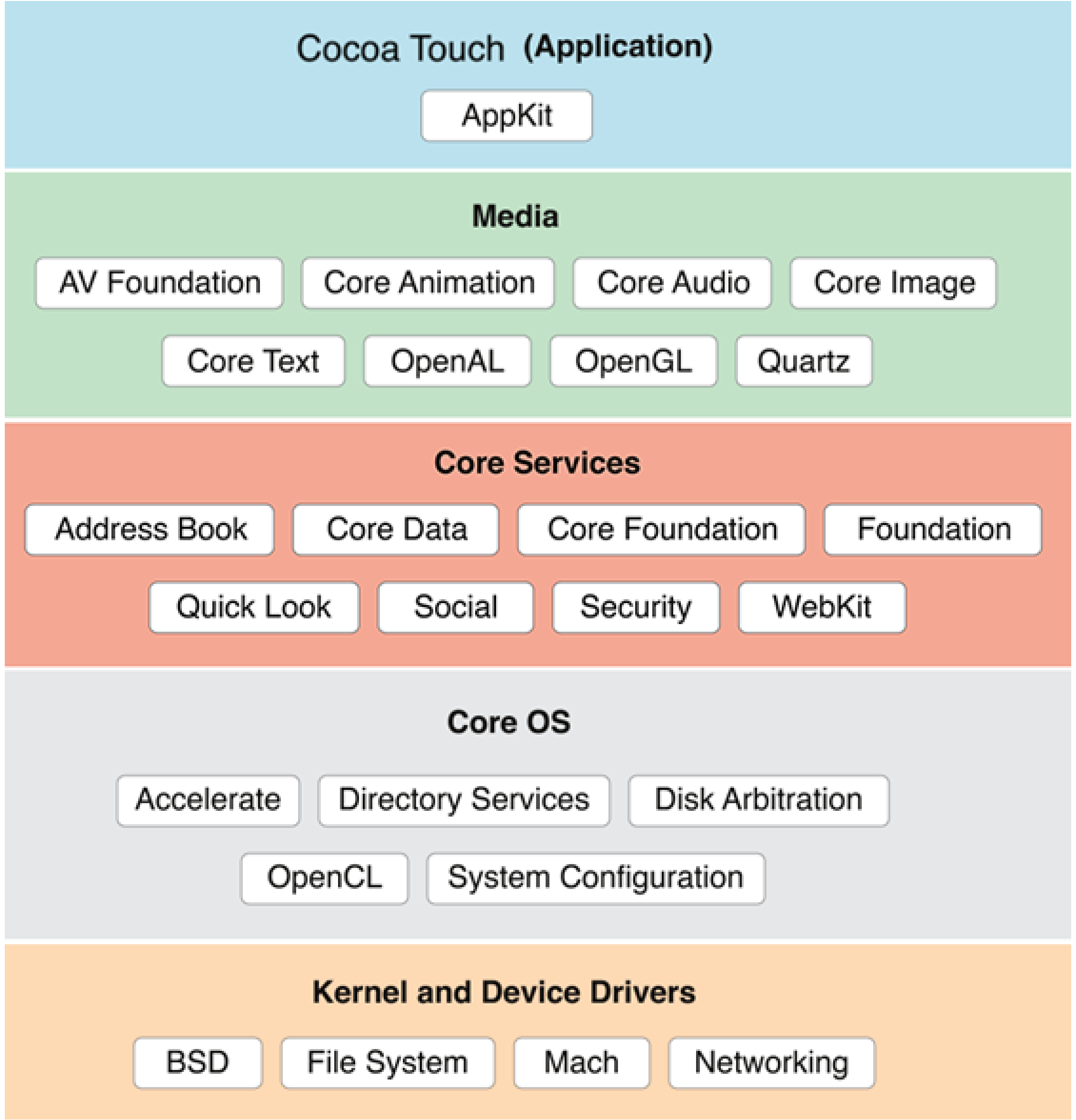
iPhone OS/iOS (Apple)

- Apple's iPhone OS was originally developed for use on its iPhone devices.
- Now, the mobile operating system is referred to as iOS and is supported on a number of Apple devices including the iPhone, iPad, and iPod Touch.
- The iOS mobile operating system is available only on Apple's own manufactured devices as the company does not license the OS for third-party hardware.
- Apple iOS is derived from Apple's Mac OS X operating system

iPhone OS/iOS Architecture



iPhone OS/iOS Architecture



Cocoa Touch (Application)

AppKit

Media

AV Foundation, Core Animation, Core Audio, Core Image, Core Text, OpenAL, OpenGL, Quartz

Core Services

Address Book, Core Data, Core Foundation, Foundation, Quick Look, Social, Security, WebKit

Core OS

Accelerate, Directory Services, Disk Arbitration, OpenCL, System Configuration

Kernel and Device Drivers

BSD, File System, Mach, Networking

Android OS (Google)

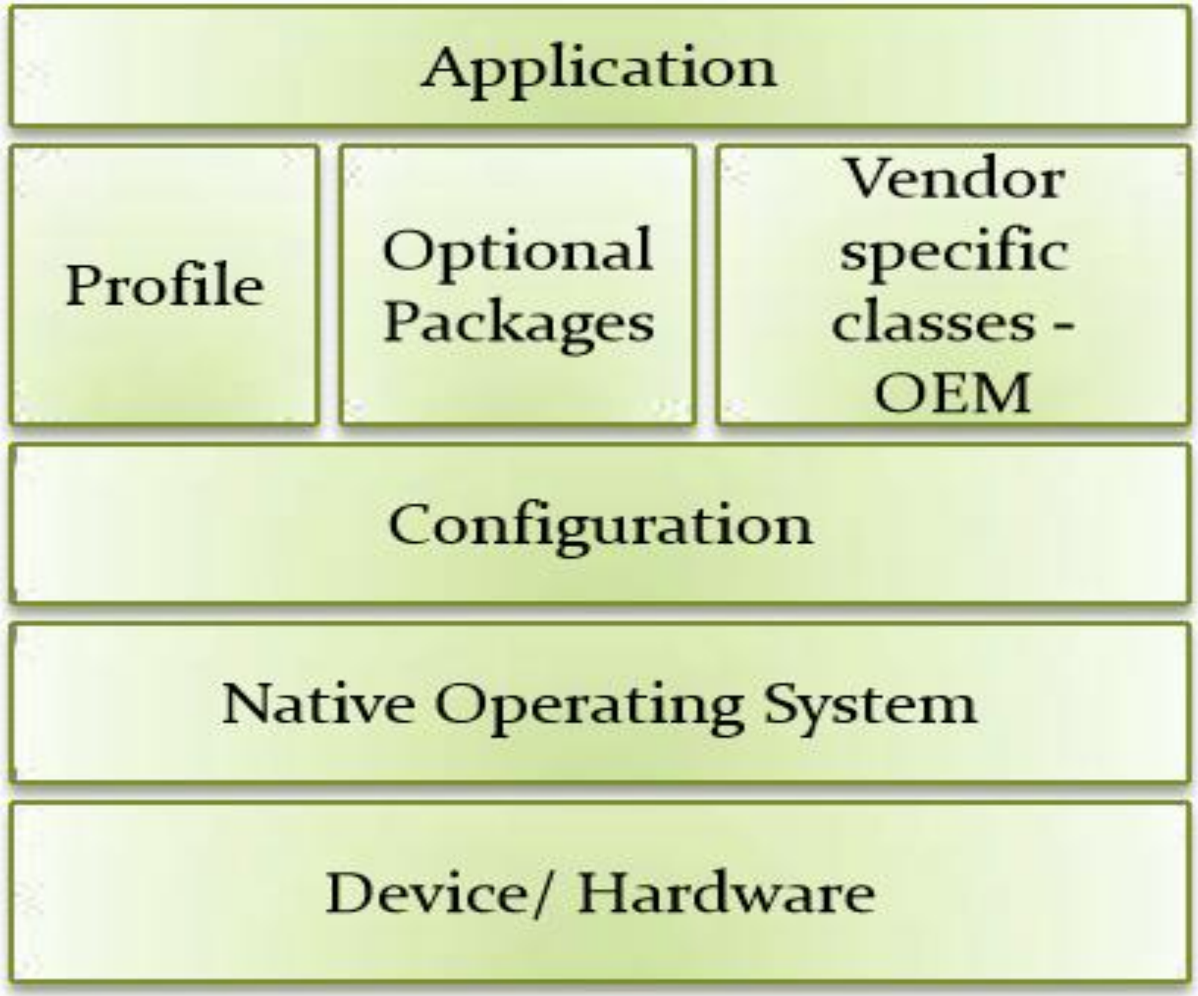
- Google’s Open Source and Free Software Stack
- It consists of OS, middleware, & key applications for use on mobile devices, including smartphones.
- Updates have been developed under “dessert-inspired” version names.
  - ✓ *Cupcake, Donut, Éclair, Gingerbread, Honeycomb,Ice Cream Sandwich, Jelly Bean, Kitkat, Lollipop, Marshmallon, Nougat, Oreo, Pie, 10.0Q(Giving away the name of dessert)*
  - ✓ Each new version until Android 9 (Pie) is in alphabetical order with enhancements and improvements.
  - ✓ Latest Android OS is Android 16 (Baklava)

Java ME

- J2ME platform is a set of technologies, specifications and libraries developed for small devices like:
  - ✓ Mobile phones (Feature phones)
  - ✓ Pagers
  - ✓ Personal organizers.

- Java ME was designed by Sun Microsystems.

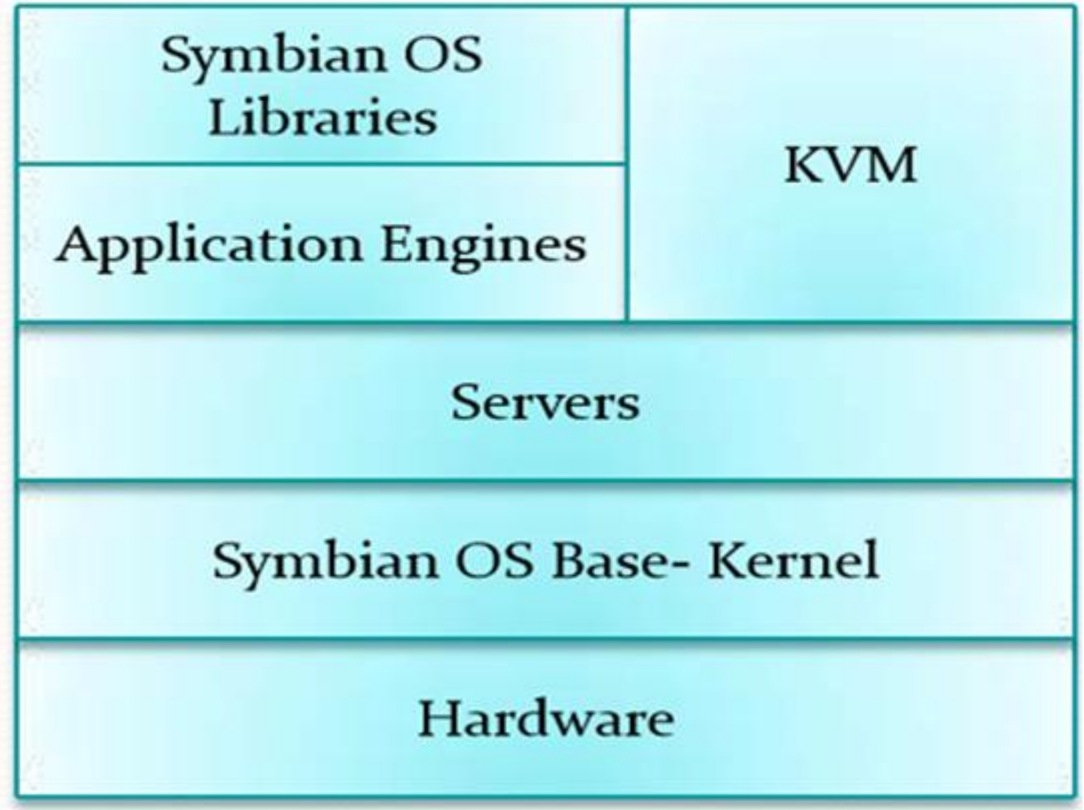
J2ME Structure



Symbian OS (Nokia)

- Symbian is a mobile operating system targeted at mobile phones that offers a high-level of integration with communication and personal information management (PIM) functionality.
- Nokia has made the Symbian platform available under an alternative, open and direct model, to work with some OEMs and the small community of platform development collaborators.
- Nokia does not maintain Symbian as an open source development project.

Symbian OS Architecture



Symbian OS Libraries

Application Engines

KVM

Servers

Symbian OS Base- Kernel

Hardware

Windows Mobile

- Windows Mobile is Microsoft's mobile operating system used in smartphones and mobile devices – with or without touchscreens.
- The Mobile OS is based on the Windows CE 5.2 kernel.
- In 2010 Microsoft announced a new smartphone platform called Windows Phone 7
- As of December 10, 2019, Windows 10 Mobile users are no longer eligible to receive new security updates, non-security hotfixes, free assisted support options, or online technical content updates from Microsoft for free.
- Microsoft recommended Windows Phone user to switch to Android or iOS.

Mobile OS Constraints & Restrictions

- Smaller Screen Size
- One Screen appears at a time
- Shorter battery life
- Wireless network Connections
- Slower Processor Speeds
- Less Available Memory

Mobile OS Constraints & Restrictions

- **Smaller Screen Size:**
  - ❖ Stay focused on the user's immediate task.
  - ❖ Display only the information that users need at any given moment.
  - ❖ For example, a customer relationship management system can provide a massive amount of information, but users only require a small amount of that information at one time.
  - ❖ Design the UI so that users can perform tasks easily and access information quickly.

Mobile OS Constraints & Restrictions

- **Shorter Battery Life:**
  - ❖ Try to handle data transmission efficiently.
  - ❖ The less often the device needs to transmit data, the longer the battery lasts.
- **Wireless Network Connections:**
  - ❖ Try to simplify how your application creates network connections.
  - ❖ Compared with standard LANs, longer latency periods that are inherent in some wireless network connections can influence how quickly users receive information that is sent over the network.

Mobile OS Constraints & Restrictions

- **Slower Processor Speed:**
  - ❖ Avoid processor-intensive tasks wherever possible.
  - ❖ Slower processor speeds can affect how users perceive the responsiveness of an application.
- **Less Available Memory:**
  - ❖ Free up as much memory as possible.
  - ❖ For example, while an application is not being used, try to keep it from using memory

Mobile OS Features

- Multitasking
- Scheduling
- Memory Allocation
- File System Interface
- I/O Interface
- Protection and Security
- Multimedia feature

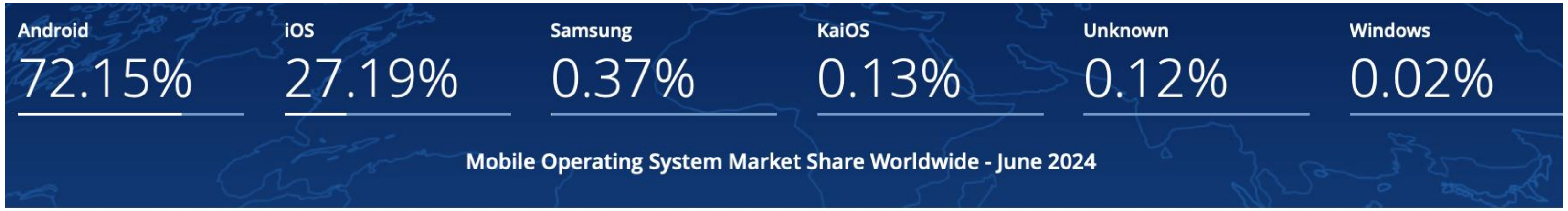
Importance and Impact of Mobile Applications

- **Ubiquity and Accessibility**: Mobile devices are ubiquitous, with billions of users worldwide.
- **Economic Impact**: The mobile app industry is a significant contributor to the global economy, generating billions of dollars in revenue through app sales, in-app purchases, and advertising.
- **Business Transformation**: Mobile apps have transformed how businesses operate, enabling new business models and revenue streams.
- **Innovation and Technological Advancements**: Mobile apps drive innovation by leveraging emerging technologies like AI, AR, VR, and IoT.
- **Social and Cultural Impact**: Mobile apps have a profound impact on society, influencing how people communicate, learn, and entertain themselves.
- **Healthcare and Well-being**: Mobile health apps assist in monitoring and managing health conditions, promoting healthy lifestyles, and providing remote healthcare services.

- Can you tell me more on the importance and impact of Mobile Applications?

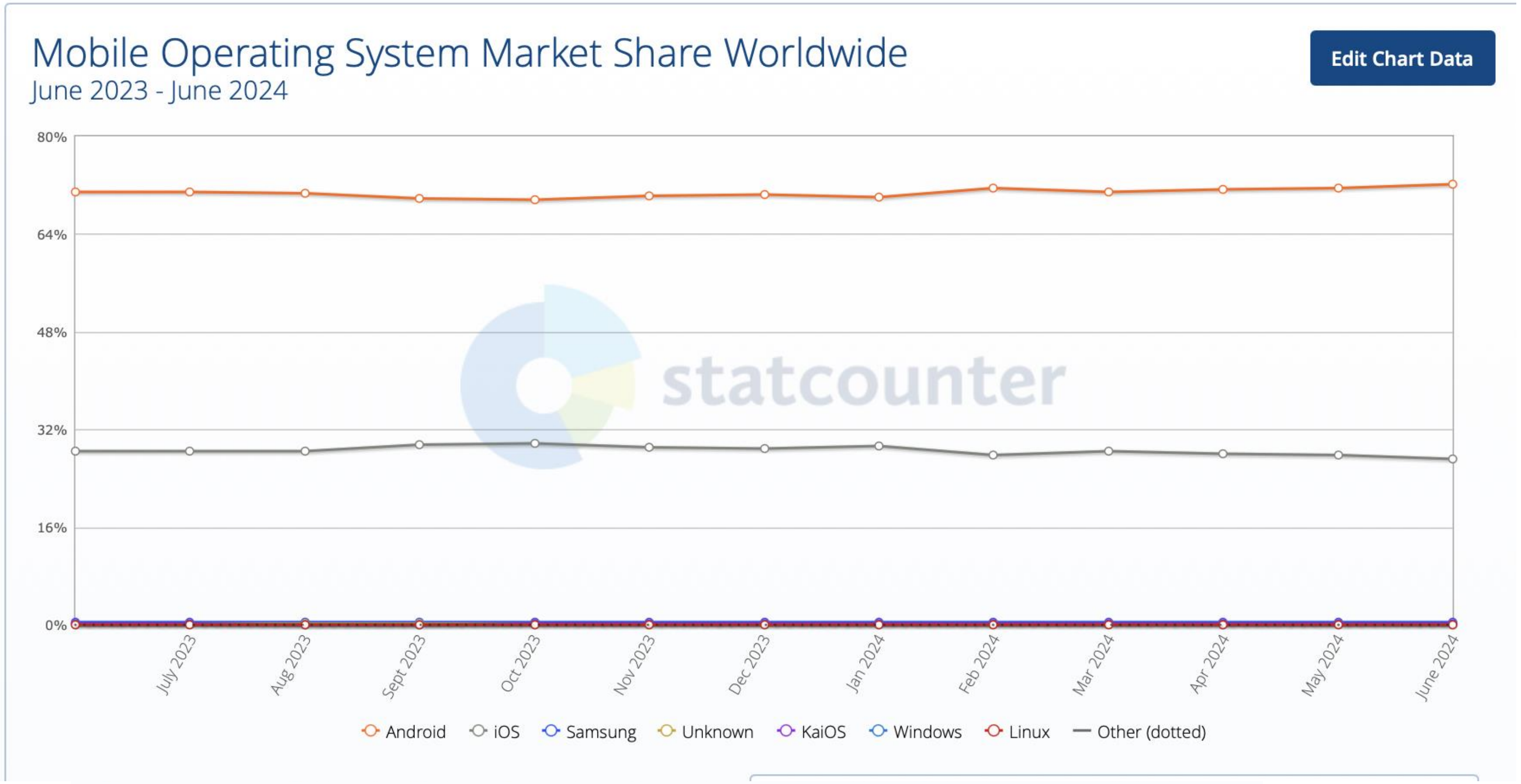
Market Scenario of Mobile Apps

The Android and iPhone OS have captured a huge market for mobile applications and services.



Android	iOS	Samsung	KaiOS	Unknown	Windows
72.15%	27.19%	0.37%	0.13%	0.12%	0.02%

Mobile Operating System Market Share Worldwide - June 2024



CHARACTERISTICS

If you examine the most successful apps, you can see six major characteristics they have in common.

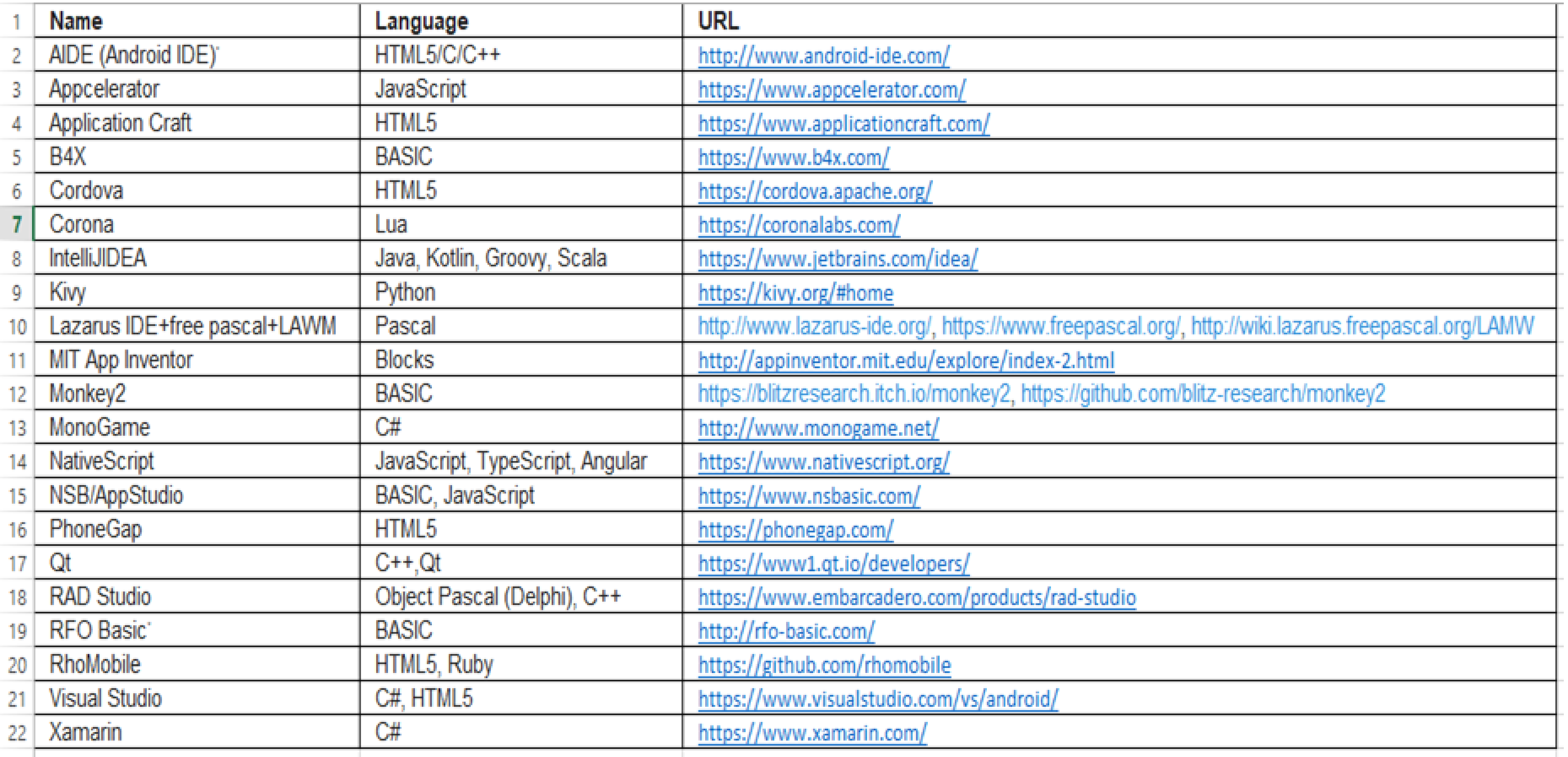
- Connectivity
- Convenience
- Localization
- Reachability
- Security
- Personalization

Development IDEs

- An Integrated Development Environment (IDE) is an all-in-one solution that allows an app developer to perform the software development cycle repeatedly and quickly.
  - ✓ It provides tools to design, write code, compile, test, debug and package the app software.

Development IDE: Android

- List of alternative Android App Development IDEs:



1	Name	Language	URL
2	AIDE (Android IDE)'	HTML5/C/C++	http://www.android-ide.com/
3	Appcelerator	JavaScript	https://www.appcelerator.com/
4	Application Craft	HTML5	https://www.applicationcraft.com/
5	B4X	BASIC	https://www.b4x.com/
6	Cordova	HTML5	https://cordova.apache.org/
7	Corona	Lua	https://coronalabs.com/
8	IntelliJIDEA	Java, Kotlin, Groovy, Scala	https://www.jetbrains.com/idea/
9	Kivy	Python	https://kivy.org/#home
10	Lazarus IDE+free pascal+LAWM	Pascal	http://www.lazarus-ide.org/, https://www.freepascal.org/, http://wiki.lazarus.freepascal.org/LAMW
11	MIT App Inventor	Blocks	http://appinventor.mit.edu/explore/index-2.html
12	Monkey2	BASIC	https://blitzresearch.itch.io/monkey2, https://github.com/blitz-research/monkey2
13	MonoGame	C#	http://www.monogame.net/
14	NativeScript	JavaScript, TypeScript, Angular	https://www.nativescript.org/
15	NSB/AppStudio	BASIC, JavaScript	https://www.nsbasic.com/
16	PhoneGap	HTML5	https://phonegap.com/
17	Qt	C++,Qt	https://www1.qt.io/developers/
18	RAD Studio	Object Pascal (Delphi), C++	https://www.embarcadero.com/products/rad-studio
19	RFO Basic'	BASIC	http://rfo-basic.com/
20	RhoMobile	HTML5, Ruby	https://github.com/rhomobile
21	Visual Studio	C#, HTML5	https://www.visualstudio.com/vs/android/
22	Xamarin	C#	https://www.xamarin.com/

Development IDE: iOS

- List of iOS App Development IDEs:
  1. Xcode [swift]
  2. Appcode [Dev. JetBrain, C++, C]
  3. Atom [Github]
  4. SublimeText3
  5. CodeRunner 2

Development IDEs

- List of IDEs for leading mobile apps:
  - Android Studio
  - Xcode
  - Visual Studio Code

TYPES of Mobile Apps

- A mobile application, most commonly referred to as an app, is a type of application software designed to run on a mobile device, such as a smartphone or tablet.
- There are three approaches for developing Mobile Apps:
  1. **Native Apps**
  2. **Web Apps**
  3. **Hybrid Apps**

TYPES : Native Apps

- In order to create true, native applications:
  - Java/Kotlin programming language must be used for Android
  - Objective C programming language for iOS

- Common, key characteristics of native applications are:
  - Applications have unhindered access to device hardware
  - Support all user interfaces
  - Interactions available in the respective mobile operating environment.

TYPES : Native Apps

- Apps are directly installed in the device.
- They do not need any data transfer to the server and works in the device without network as the data about the app is stored in the device itself.
- For example, Notes and Reminder in phones.

TYPES : Native Apps

- The are many apps on your mobile device are native apps.

- **Pros**
  - Fast and Responsive
  - Easy to tap into wider functionality
  - Push Notifications
  - Match UI/UX to platform conventions

- **Cons**
  - More than one codebase.
  - Native can cost more and take longer to build.
  - Take up device storage
  - App updates

TYPES : Native Apps

- Example:
  - ❖ Facebook
  - ❖ Google Maps
  - ❖ Spotify

TYPES : Web Apps

- An application that is accessed via a web browser over a network such as the Internet is called web app.
  - Apps are not installed in the device and can be accessed through native browser by hitting the URL of the web.
  - The device memory size is not imperative as the app data is not stored in the device.
  - It is completely dependent on the quality of the browser.
    - For example, www.google.com.

TYPES : Web Apps

- Dedicated mobile web applications are developed with a combination of **HTML5**, **JavaScript**, and **CSS**.
- Web apps don’t need to be downloaded like mobile apps do.
- Web apps load in browsers like Chrome, Safari, or Firefox and they don’t take up any memory or storage on the user’s device.

TYPES : Web Apps

- Pros and Cons of Web-Based Mobile Apps are:

- **Pros**
  - Lower development cost
  - No need for device storage
  - Access from almost any device

- **Cons**
  - Dependent on internet connection
  - Dependent on the browser
  - Limited access to APIs

TYPES : Hybrid Apps

- It is the combination of native app and web app.
- You install it like a native app, but it’s actually a web app on the inside.
- Hybrid apps, like web apps, are built with **Javascript**, **HTML**, and **CSS** and run in something called **Webview**, a simplified browser within your app.
- Example: Twitter, Gmail, Instagram, Uber

TYPES : Why Hybrid Apps?

- **A hybrid app offers a solution to the following problem:**
  - Say you have an idea for an app and you don’t know if people will like it or not.
  - Your goal is to put something usable into their hands as quickly as possible. You’re short on resources, so you need to create a simple version of your product that still provides value. In the start up world, this is called an MVP (minimum viable product).
  - Building a web app might be the truly minimal option, but it won’t really allow you to test whether people will download and use an app on their device.

TYPES : Hybrid Apps

- The Hybrid Apps are developed using multi-platform web technologies:

  - **Pros**
    - One codebase to manage
    - You save time and money
    - Easier to scale
    - You still have access to device features
    - Big number of APIs available

  - **Cons**
    - Lower Performance and Speed
    - Cross-platform is tough
    - The UX of the app will suffer

CHOOSE RIGHT TYPES?

- **Native App if your app:**
  - ✓ Will run only on a specific platform – for example, only on iOS or only on Android
  - ✓ Involves heavy graphics (like in games)
  - ✓ Requires full access to device functions and capabilities (camera, microphone, geolocation, and others) or full control over UI
  - ✓ Should be fast, stable and overall have a high-performance level.

CHOOSE RIGHT TYPES?

- **Hybrid App if your app:**
  - ✓ You have limited resources for development and maintenance
  - ✓ At the same time, you want your app act very similar to native apps
  - ✓ You want your users to be able to find your app in application stores (Play Stores for Android and App Stores for iOS)

CHOOSE RIGHT TYPES?

- **Web-Based Mobile App if your app:**
  - ✓ You need the app up and running as soon as possible
  - ✓ You have very limited resources for development and maintenance
  - ✓ At the same time, you want your app to reach a broad audience across platforms

Cross-Platform App Development

- Cross-platform app development refers to the practice of creating applications that can run on multiple operating systems (e.g., Android, iOS, MacOS, Windows, Web) using a single codebase.
- Developers use cross-platform frameworks and tools to write code once and deploy it across different platforms, ensuring consistent functionality and user experience.
- Example: Flutter, React Native, etc.

Impact of Cross-Platform App Development

- **Broader Market Reach**: By targeting multiple platforms, businesses can reach a wider audience, increasing their app's user base and potential revenue.
- **Cost Efficiency**: Developing a single codebase for multiple platforms reduces the overall development costs compared to creating separate native apps for each platform.
- **Faster Time to Market**: Cross-platform development accelerates the app development process, allowing businesses to launch their apps more quickly.
- **Simplified Maintenance**: Maintaining and updating a single codebase is simpler and more efficient than managing separate codebases for each platform.

Disadvantages of Cross-Platform App Development

- **Performance Limitations:** Cross-platform apps may not perform as efficiently as native apps, particularly in resource-intensive applications.
- **Limited Access to Native Features:** Cross-platform frameworks may not provide access to all native features and functionalities of each platform, limiting the app's capabilities and user experience.
- **Platform-Specific Issues:** Cross-platform apps might encounter issues or bugs specific to individual platforms.
- **Dependency on Frameworks**: Relying on third-party frameworks can introduce risks, such as dependency on framework updates and potential discontinuation of support.
- **User Experience Trade-Offs:** Achieving a truly native look and feel on each platform can be challenging with cross-platform development, potentially leading to a less polished user experience compared to native apps.

Advantages of Cross-Platform App Development

- Cost Savings
- Consistent User Experience
- Reusability of Code
- Faster Development Process
- Easier Maintenance
- Access to a Larger Talent Pool

Thank you!