# Introduction to mobile Operating System Platforms (OSPs): Symbian, Android, iOS, Blackberry and Widows


**Lecturer contact :**

Dr. Drake Patrick Mirembe, PhD

E:dpmirembe@gmail.com

Website: www.drakemirembe.org

# Presentation Overview

1. What is an Operating System?
2. Generic architecture of a typical mobile Operating System
3. Operating system differences
   - Architecture
   - Design
4. Background on mobile OS market share
   - Why each is popular in the different parts of the world

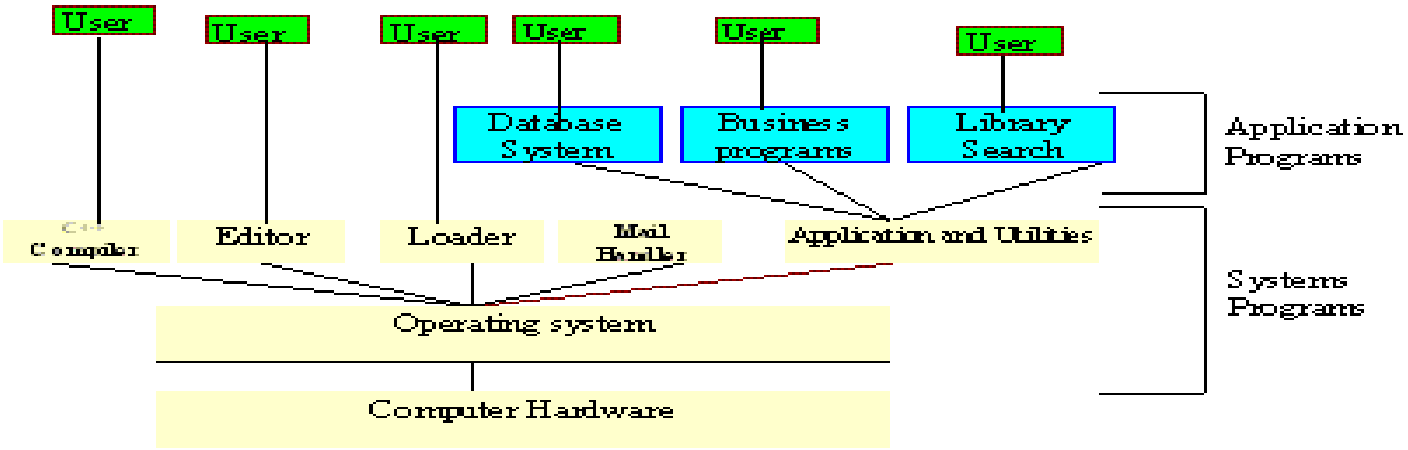
# Computer system components

- Application programs

  Database systems, business programs
- Systems programs

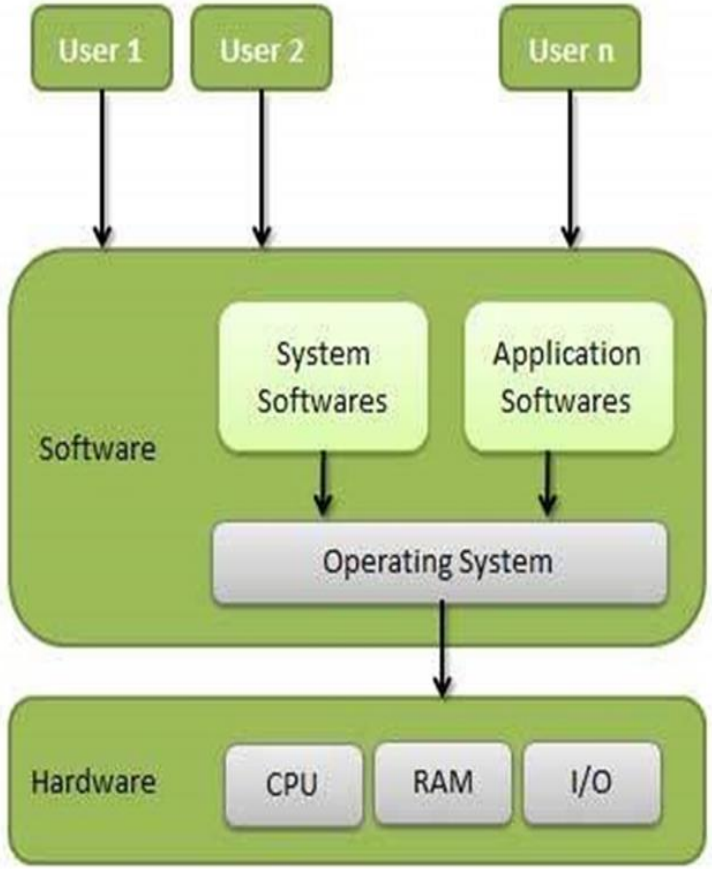
  Operating system, compilers, editors, loaders, utilities
- Hardware

  Memory, CPU, arithmetic-logic unit, various bulk storage, I/O, peripheral devices



# What is an Operating System?

An operating system is a program that acts as an interface between the user and the computer hardware and controls the execution of all kinds of programs




OS high level components

- Applications
- Middleware
- Kernel

# What is an Operating System?

## Functions of an OS

- Memory management
  - Keeps tracks of primary memory - what part of it are in use by whom, what part are not in use.
  - Decides which process will get memory when and how much.
  - Allocates the memory when a process requests it to do so.
  - De-allocates the memory when a process no longer needs it or is terminated.
- Processor management (process scheduling)
  - Keeps track of the processor and status of processes. The program responsible for this task is known as **traffic controller**.
  - Allocates the processor (CPU) to a process.
  - De-allocates processor when a process is no longer required.

# What is an Operating System?

## Functions of an OS

- Device management through drivers
  - Keeps tracks of all devices through the **I/O controller**.
  - Decides which process gets the device when and for how much time.
  - Allocates the device in the efficient way.
  - De-allocates devices.
- File management
  - Keeps track of information, location, uses, status etc. The information is organized in directories called a **file system** for easy navigation. The file system has files and other directories.
  - Decides who gets the resources.
  - Allocates the resources.
  - De-allocates the resources.

# What is an Operating System?

**Functions cont.**

- Security

  It prevents unauthorized access to programs and data.

- Control over system performance

  Records delays between a request for a service and the response from the system.

- Job accounting

  Keeps track of time and resources used by various jobs and users.

- Error detection

  Produces dumps, traces, error messages, and other debugging and error detecting aids.

# What is an Operating System?

**Functions cont.**

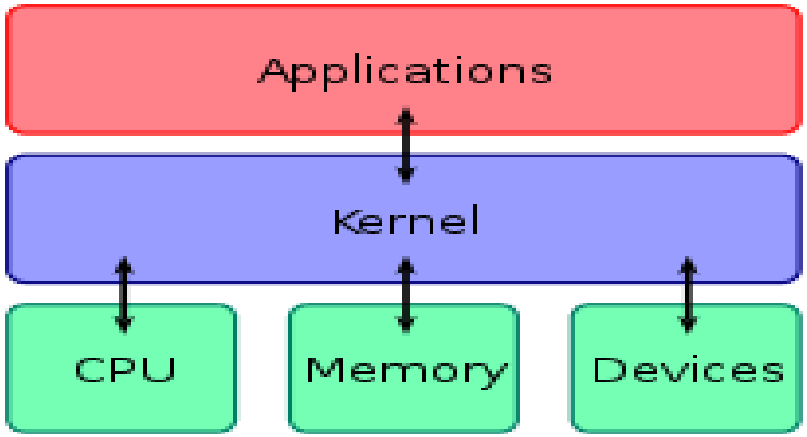
- Coordination between other software and users

  Coordination and assignment of compilers, interpreters, assemblers and other software to the various users of the computer systems.

- Establishes a user interface
  - GUI
  - Terminal
- Executes and provides services for application software
- Allows convenient usage – abstracts the complexities
- It is a control/management tool

# What is an Operating System?

**Kernel**




- Is a computer program at the core of a computer's operating system that has complete control over everything in the system
- The kernel code is always resident in memory and facilitates interactions between hardware and software components.

# What is an Operating System?

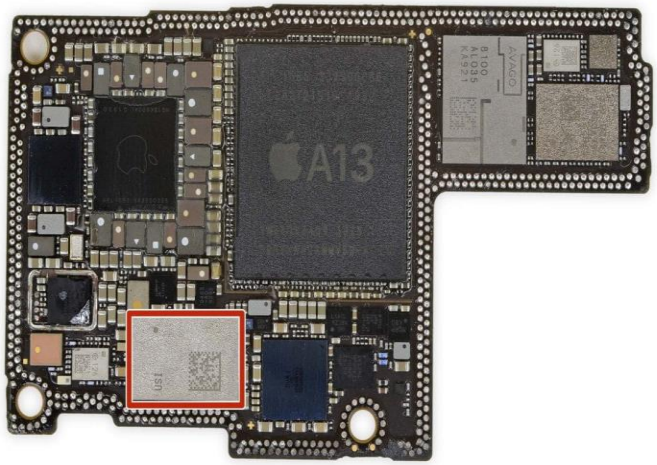
**Kernel**

- First program of the OS loaded, remains in the main memory and handles the rest of the startup
- Handles memory, peripherals, file, storage disks and input/output (I/O) requests from software translating them into data-processing instructions for the central processing unit.

# What is an Operating System?

**OS location**

- ROM chip in hand held/mobile devices
- Hard disk in most computers




# What is an Operating System?

## OS Evolution

Serial Systems

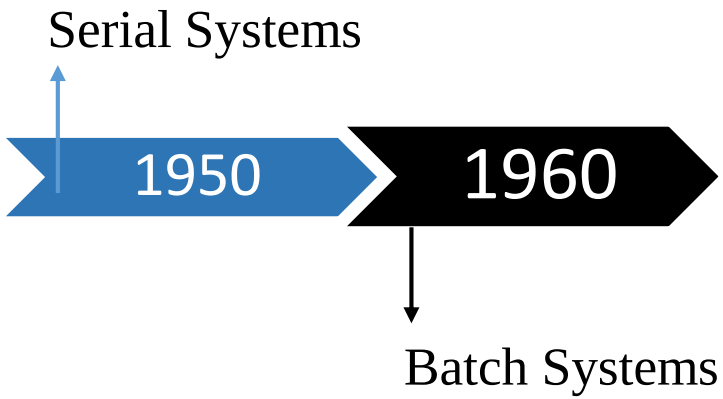
1950

Serial Processing

- Type the program or use a punched card
- Convert the punched card to a card reader
- Submit to the computing machine
- Take printouts from a printer
- Programmer ready for the next program

# What is an Operating System?

## OS Evolution

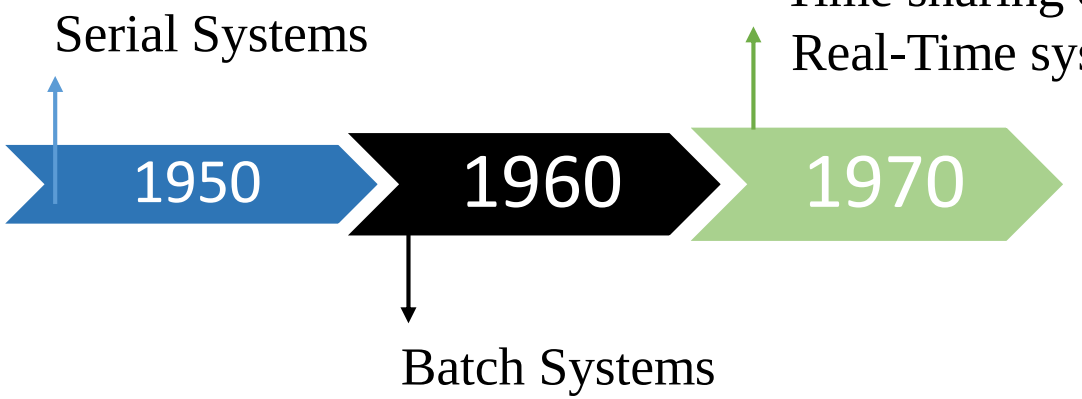



Batch processing

- Process programs in a batch
- Carrier carried the batches between the rooms
  (card reader, execution and printer)

# What is an Operating System?

## OS Evolution




### Multi programming

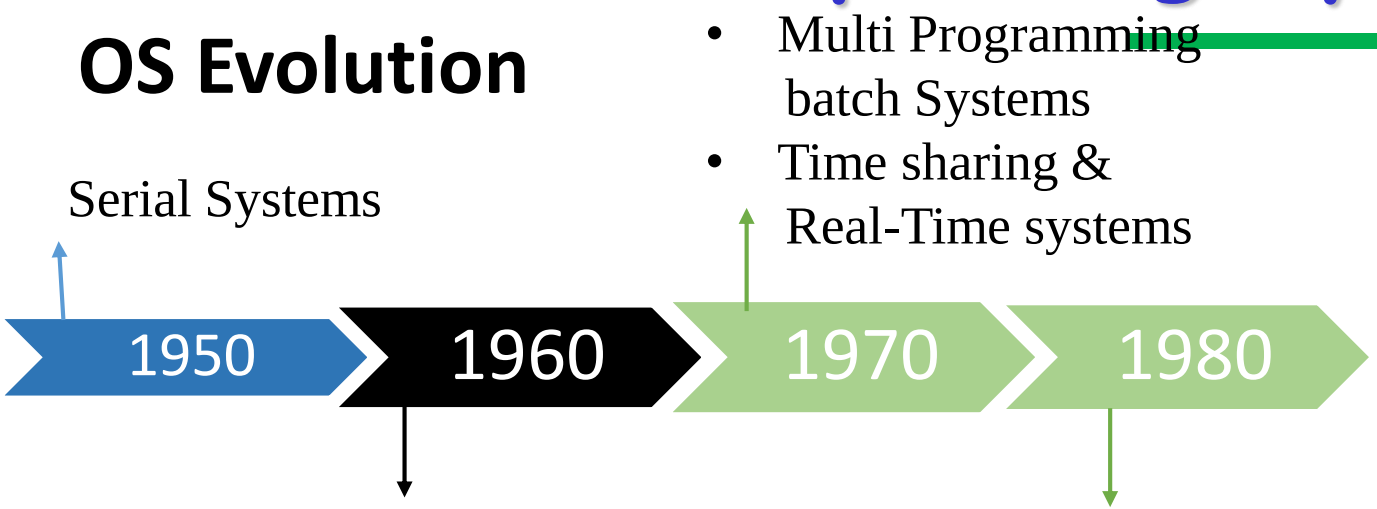
- Execute several programs simultaneously on a single processor

### Time sharing(Multi tasking)

- CPU is shared by different processes hence “Time sharing” the time slots are defined by the system
- The scheduler selects a job in a queue and switches the CPU to that job
- When the time slot expires the CPU switches to another job

# What is an Operating System?

## OS Evolution




### Multi processor (Tightly coupled)

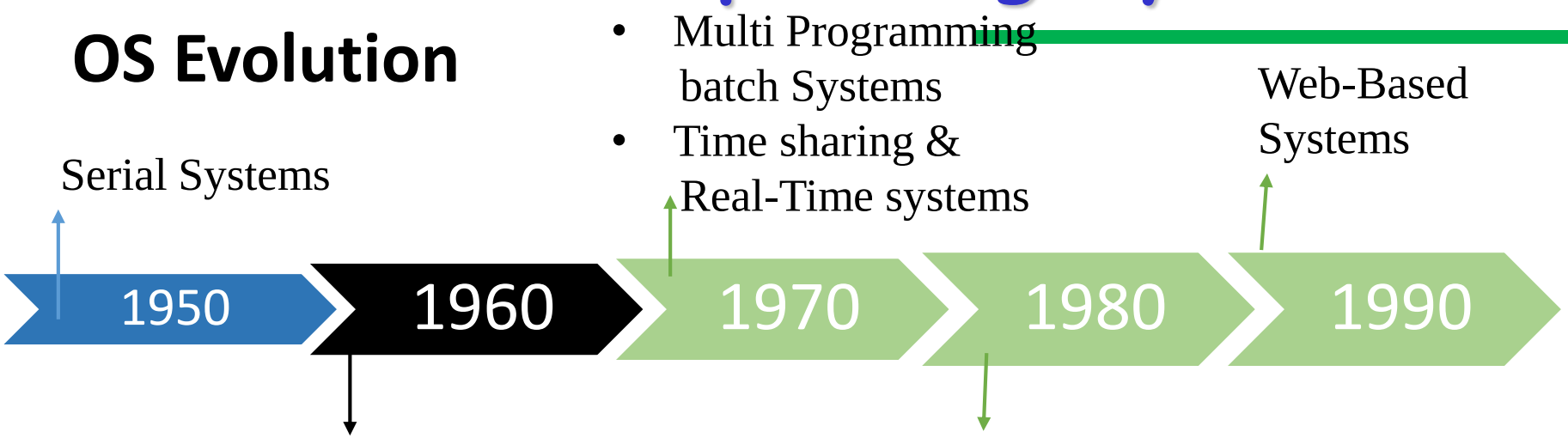
- A number of processors executing their jobs in parallel
- Shares the computer Bus, clock, memory and peripherals

### Distributed (Loosely coupled)

- Each processor has its own local memory
- Processors communicate through with each other through various lines eg. High speed buses

# What is an Operating System?

## OS Evolution




Web based

- Allows users/devices access applications and information stored remotely or in a different part of the web
- Are interfaces to distributed computing systems like cloud and utility computing systems

# mobile OS

The mobile Operating System is a newer concept which has been built on what the Computer operating systems have accomplished over the years.

Enabling factors

- Hardware
- Software
- Internet

# Wearables OS

- Wearable OS is designed or refactored to satisfy the requirements of wearable computers
- These wearable devices vary significantly from one to the other and are also different from the requirements of both desktop and mobile devices

Operational modes:

- Standalone mode
- Need a wireless connection to an Internet-connected device, typically a smartphone

# Wearables OS

- Wear OS (previously Android wear)
- Apple watchOS
- Xiaomi
- Fitbit OS
- Garmin OS
- Tizen for wearables

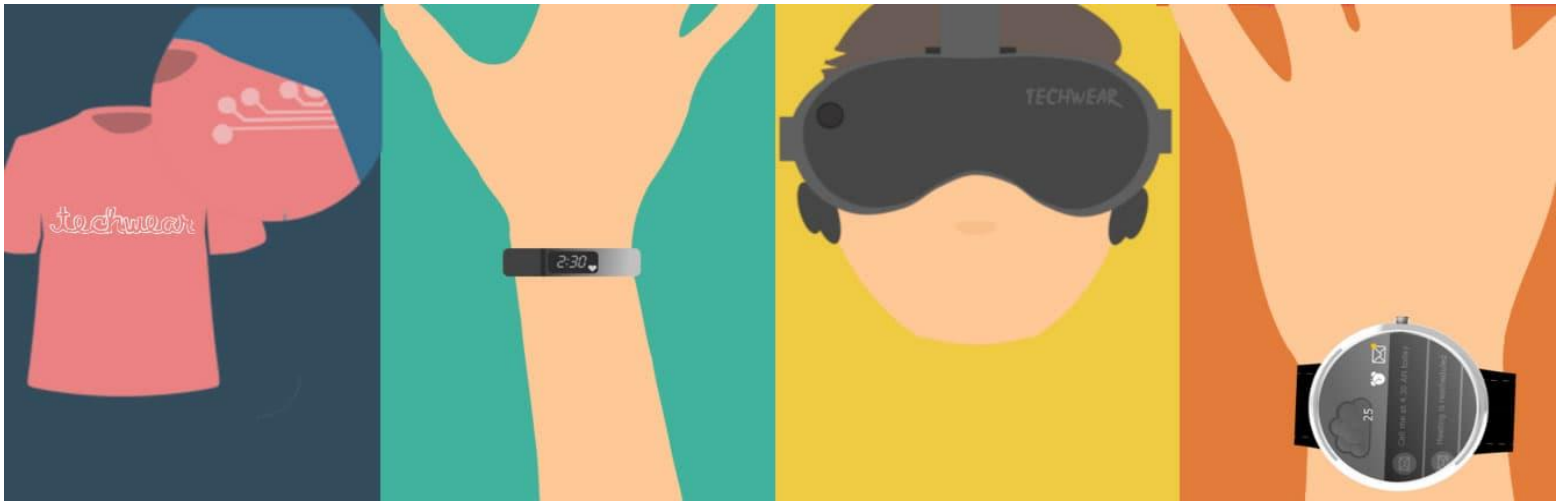
Properties

- Support Bluetooth, WiFi, 3G and LTE
- Hardware manufacturers Asus, Broadcom, Fossil, HTC, Intel, LG, MediaTek, Imagination Technologies, Motorola, New Balance, Qualcomm, Samsung, Huawei, Skagen, Polar, TAG Heuer, Suunto, and Mobvoi.

# Wearable technology

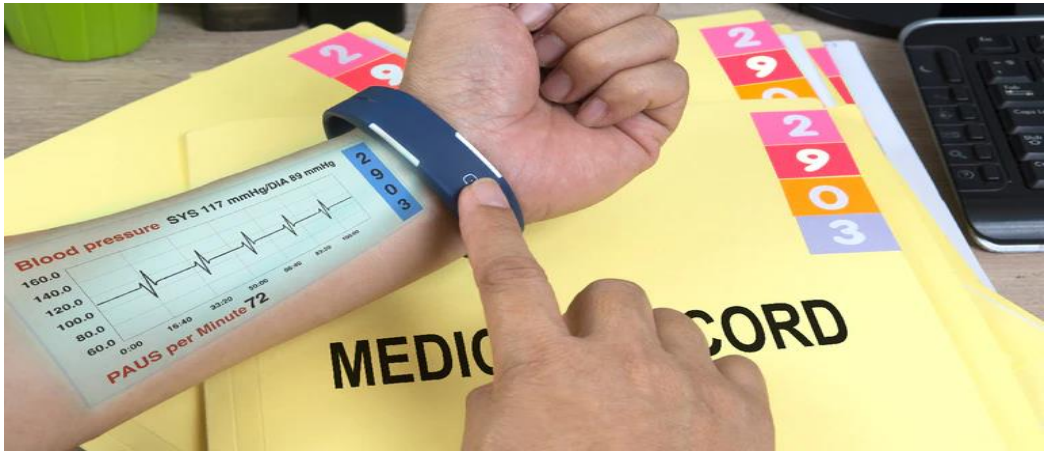

- Heart rate
- Blood oxygen levels
- Breathing rate
- Muscle electrical activity
- Stress/emotion
- Cognitive function
- Movement patterns
- Sweat analysis
- Sleep

# Forms of wearable technology




- Smart watches
- Smart jewelry
- Fitness trackers
- Implantables
- Head mounted displays

# Future of wearable technology

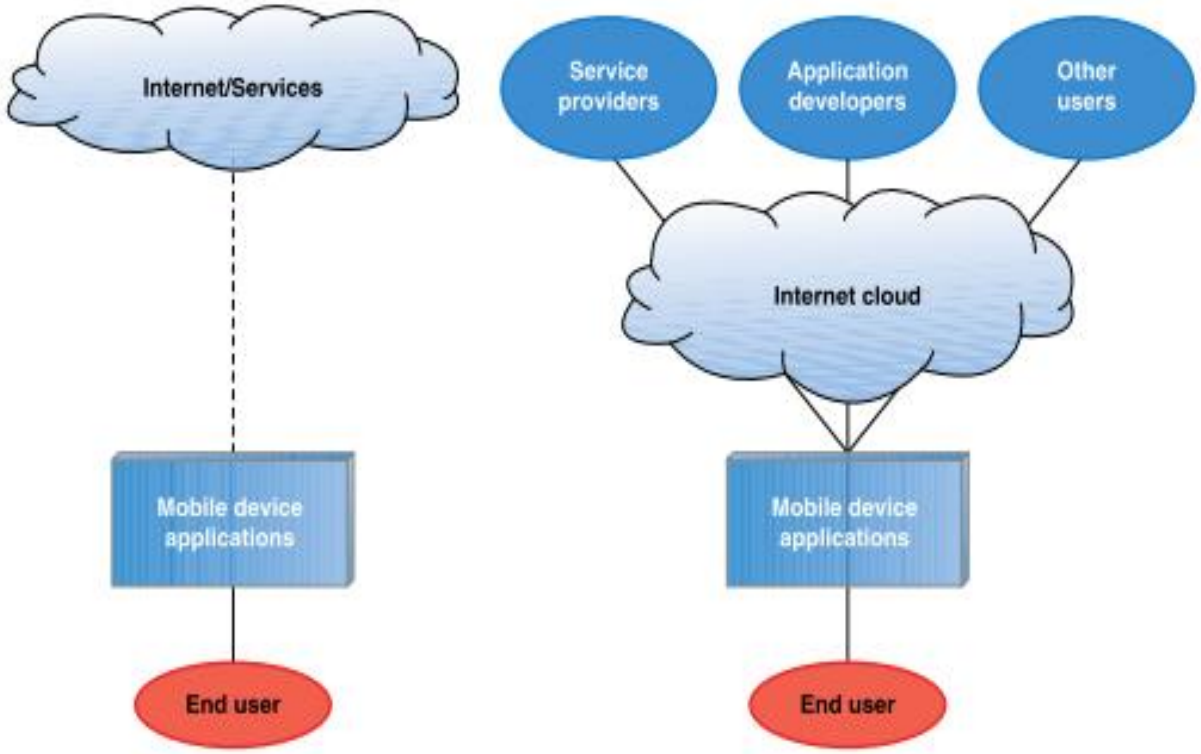



# Generic architecture of a typical mobile OS

Layered architecture

1. Applications
   - Native
   - Extended
2. Middleware
   - Interface between applications and the kernel
   - Enables communication and data management
3. Kernel
   - Facilitates interaction between hardware and middleware

# Generic architecture of a typical mobile OS



# Operating system differences: Overview

| | | | | |
|---|---|---|---|---|
| Symbian Ltd | Google, Open Handset Alliance | Apple Inc | Blackberry Ltd | Microsoft |
| June 1997 | September 2008 | June 2007 | January 1999 | November 2010 |
| Nokia Belle 2012 | Android 12 | iOS 14 | Blackberry OS 10.2 | Windows Mobile 6.5 |
| Proprietary | Free and Open source | Proprietary | Proprietary | Proprietary |
| C++ | C, C++, Java, Kotlin | C, C++, Objective C & Swift | C++ | C#, C++, C |
| Nokia Ovi Store | Google Play | App Store | Blackberry World | Microsoft Store |
| CodeWarrior, Carbide | Android Studio, Intellij IDEA, Eclipse, Visual Studio | Xcode, Appcode, Atom | Momentics, RIMs JDE, RIMs JDE plugin for Eclipse | Visual Studio |
| .SIS, .SISX | .APK | .IPA | .ALX, .COD | .XAP, .APPX |
| | SQLite | SQLite, Core Data, Firebase, Realm | SQLite | SQLCE, SQL Azure, Perst |

# Mobile Operating System differences: Architecture

- User experience and battery life
- Cloud readiness HTML 5 capability, Web Apps accessed through APIs, Cross platform capability, Performance and Cloud integration
- Mobile device Fragmentation
- Openness to players of the mobile ecosystem
  - Manufacturers OEM make/sell the devices
  - Service providers – Networks and Value Added services
  - End users (customers)
  - ISV develop commercial applications
  - Developer communities – Develop apps and contribute to the development and evolution of the operating system if it is open sourced

# Mobile Operating System differences: Architecture

OS share more similarities than differences

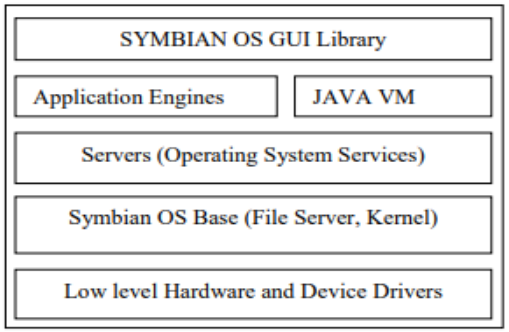
- Development language, IDE and SDK
- App stores
- Packaging
- Persistence storage and Database support
- Multi tasking & 3D support (Touch screens & sensors a must)
- Device based payments (Google Pay, Apple Pay, Microsoft Pay)
- Security

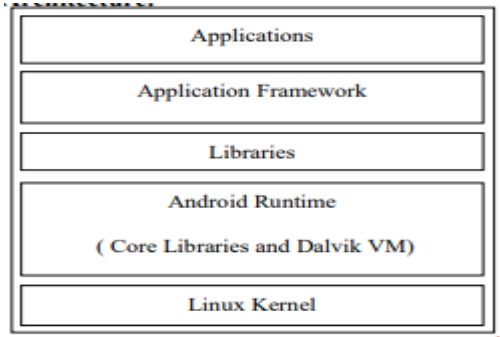
Solutions

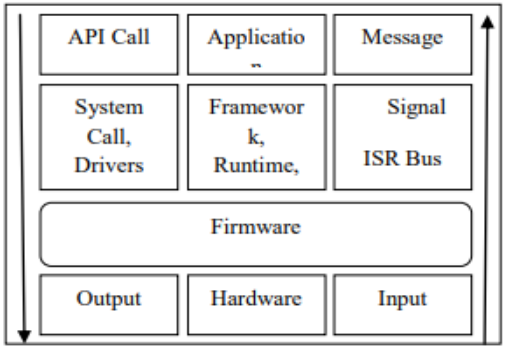
- Hybrid model cross platform development– Build once run everywhere

# Mobile Operating System differences: Architecture

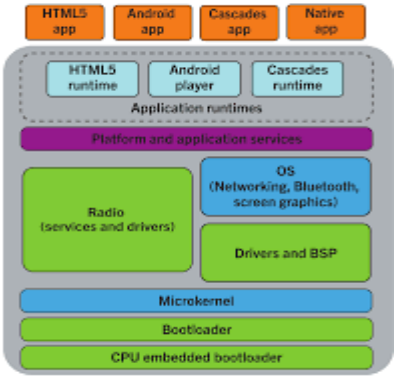
symbian



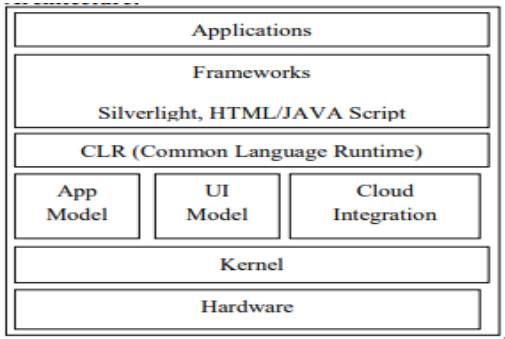




BlackBerry




BlackBerry 10 platform




Eight Tech Consults
People. Innovation. Technology. Services

# Mobile Operating System differences: Design


UI Design focus

- Responsive design
- Smooth UI/UX
- Consistent network access
- Mobility – eg Apple continuity

# Mobile Operating System differences: Design

UI Design closer look at Android and iOS

Top of screen navigation

**iOS**

"Back" action on left. Main action on right. Title starts left-aligned, but becomes centered when scrolled.


**Android**

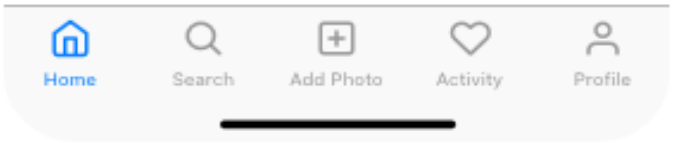
"Back" and title left-aligned. Action(s) on right.

# Mobile Operating System differences: Design

## Primary navigation destinations

**iOS**

Primary destinations displayed as 2-5 icons across bottom of screen.




**Android**

Primary nav destinations shown in a variety of different places.


**Hamburger menu** for app destinations

Search via **contextual action**

**Tabs** for views into similar data

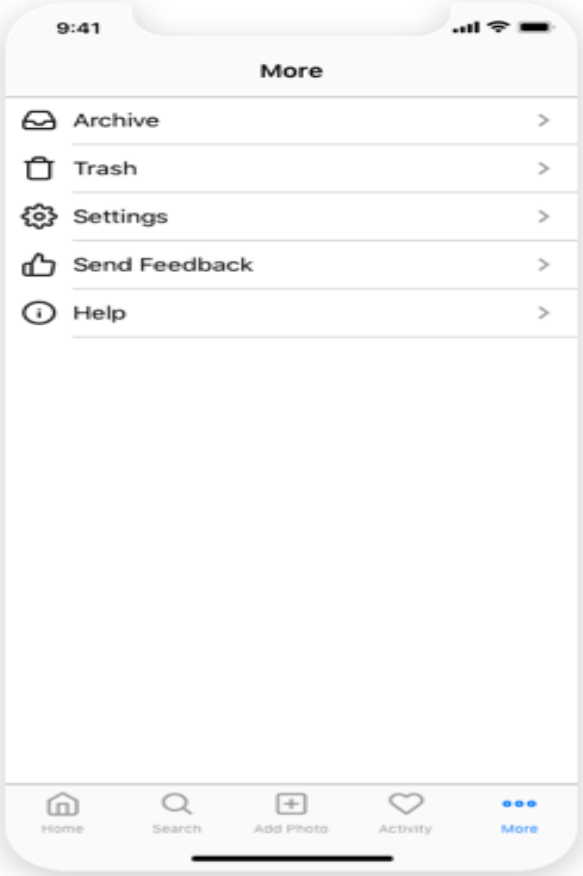
**Floating Action Button** for screen's primary action

# Mobile Operating System differences: Design

## Secondary navigation destinations

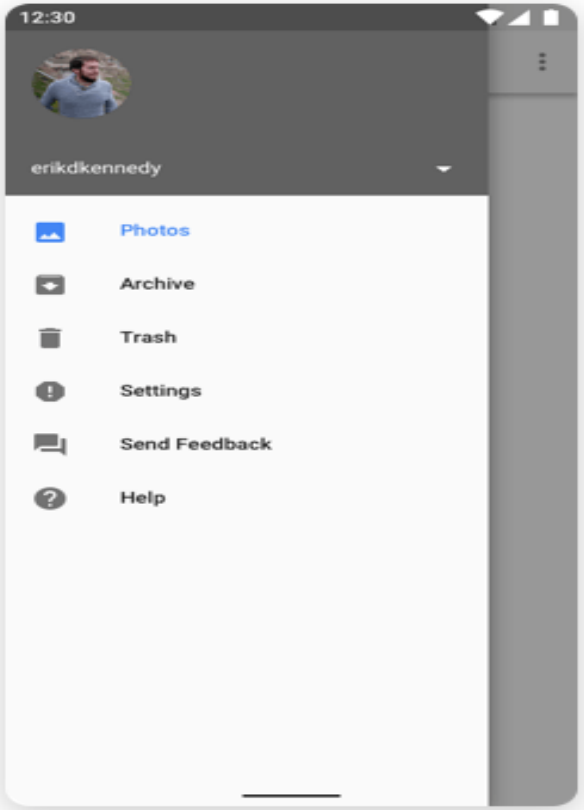
**iOS**

Nav destinations that don't fit in the bottom tab bar can be placed in a "More" tab.




**Android**

Secondary nav destinations are shown in a sidebar that's shown when the user taps the "hamburger" menu icon.

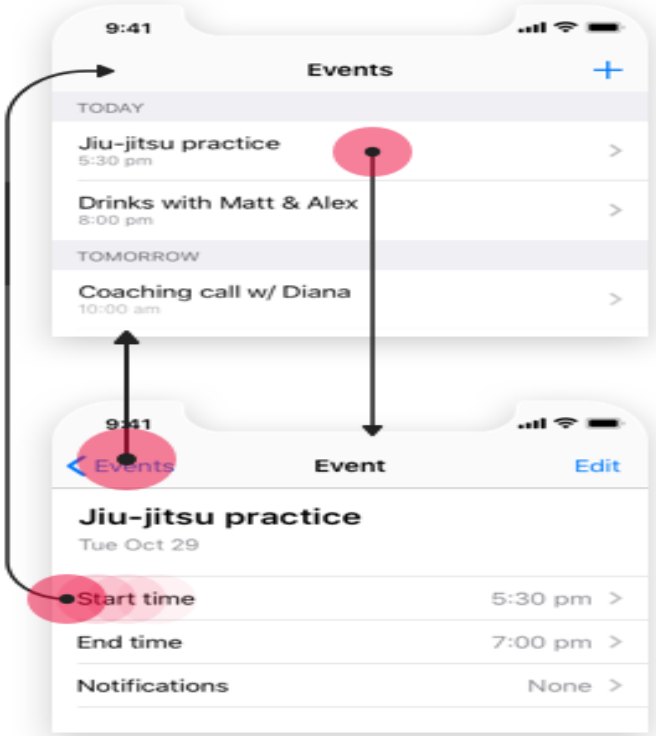


# Mobile Operating System differences: Design

## Back pattern

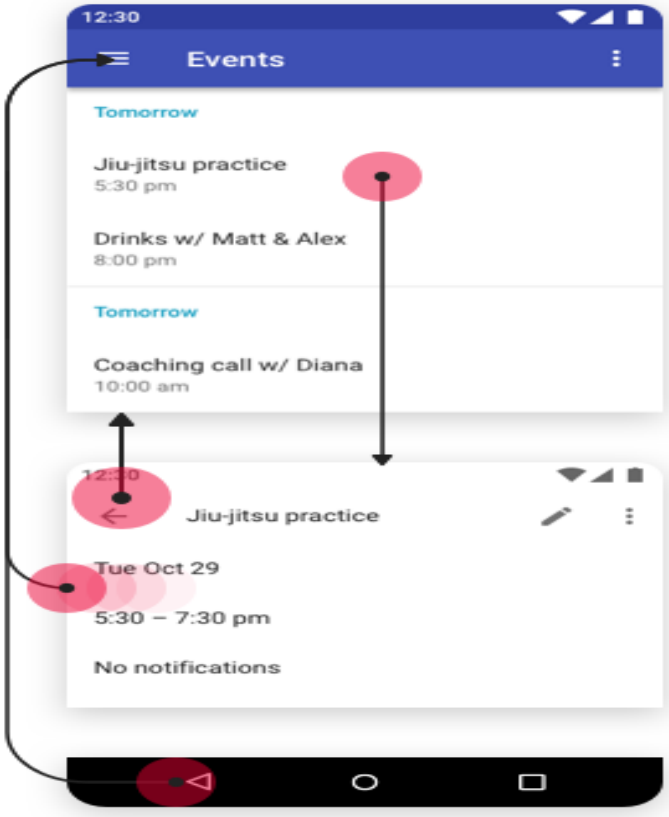
**iOS**

Navigate back using (1) a **"Back"-like action** in the upper-left. If that's present, you can also (2) **swipe right from the left edge.**




**Android**

Back buttons, if present, are in the upper-left. Android 9 (and older) phones also have a **permanent back button** below the screen. On Android 10 (and newer), you can navigate back by **swiping right from the left edge.**

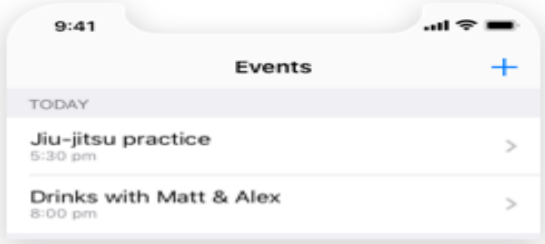


# Mobile Operating System differences: Design

## Call to action button

**iOS**

Primary action is usually in the upper-right.

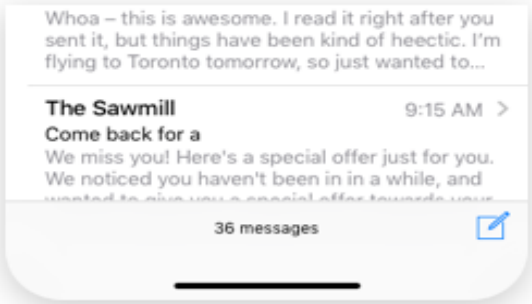



**Android**

Primary action floats at bottom-right of screen - the "floating action button".

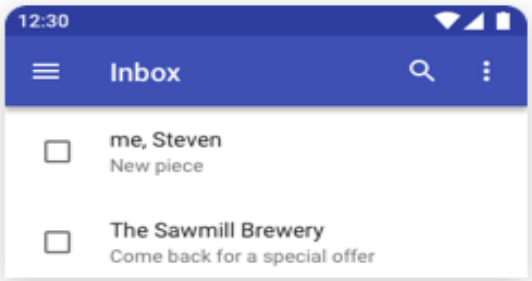

**iOS**

Important page actions will sometimes be shown in a bottom "toolbar".




**Android**

Other actions relevant to the current page are displayed at the top.



# Mobile Operating System differences: Design

## Search

**iOS**

Cancel search mode by pressing "Cancel", and clear any existing query by pressing "X".


**Android**

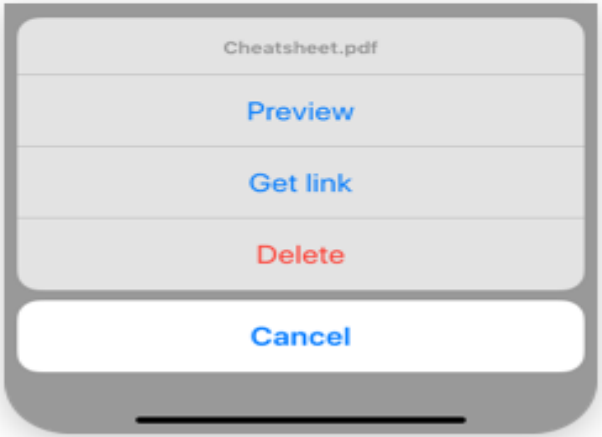
Cancel search mode by pressing the Back arrow, and clear any existing query by pressing "X".

# Mobile Operating System differences: Design

## Action menus

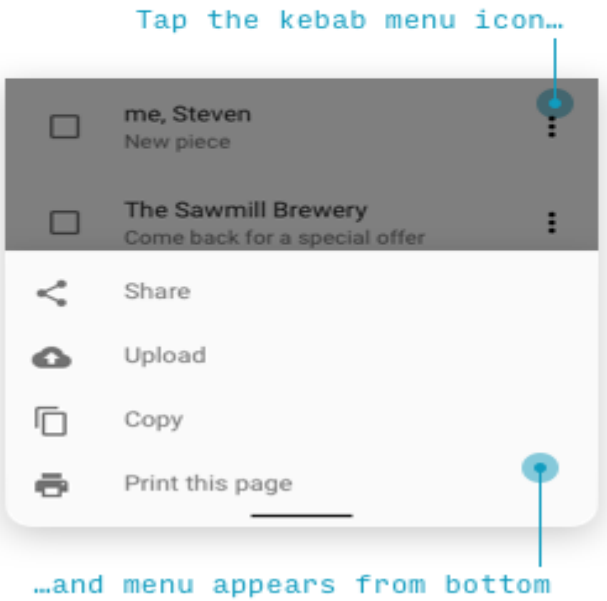
**iOS**

Action menus can be triggered from anything, and slide up from the bottom.




**Android**

Action menus are triggered by tapping the 3-dot "kebab menu". Large menus will slide up from the bottom.

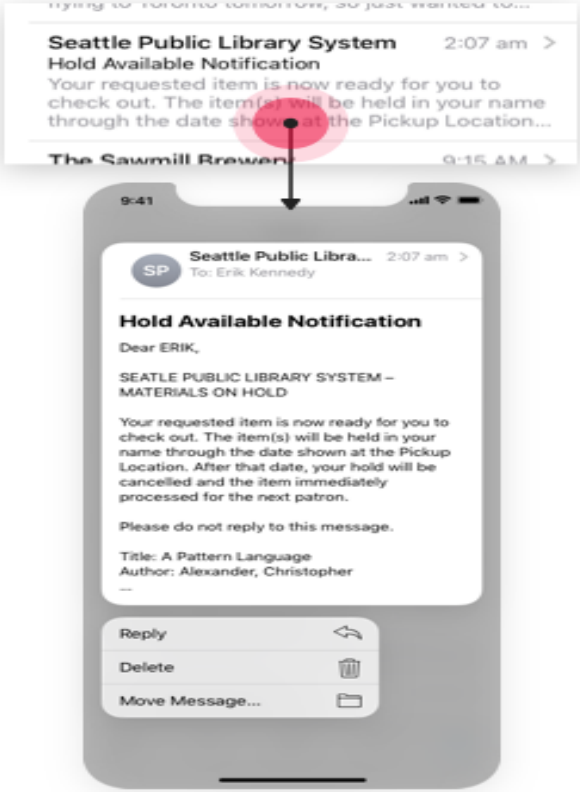


# Mobile Operating System differences: Design

## on - action menus

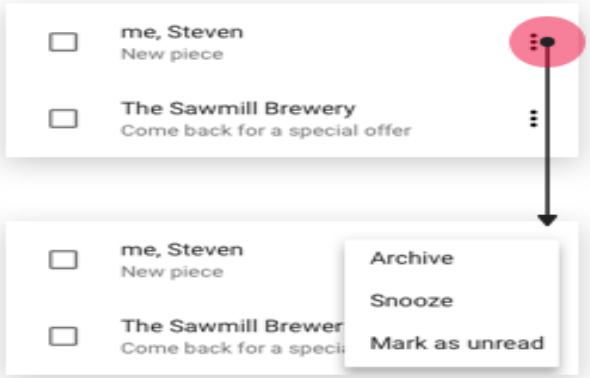
**iOS**

On-element menus ("context menus") can be triggered by **long-pressing** the element. There is no visual indicator of what will respond to a long-press.




**Android**

As always, the kebab menu triggers menus. Android on-element menus are typically shorter than bottom panels.

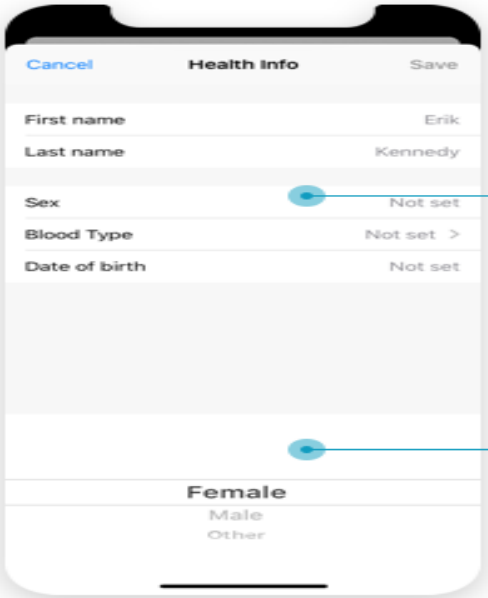


# Mobile Operating System differences: Design

## Selection controls

**iOS**

The "Picker" control allows the user to pan to the value they want. Difficult to use with 10+ values!

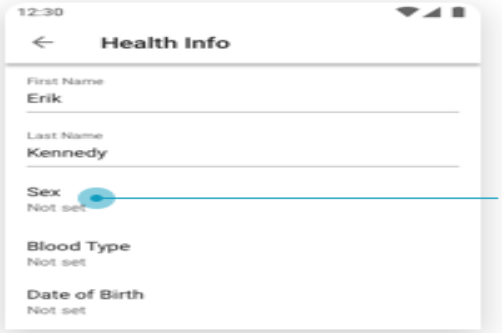



Tap here…

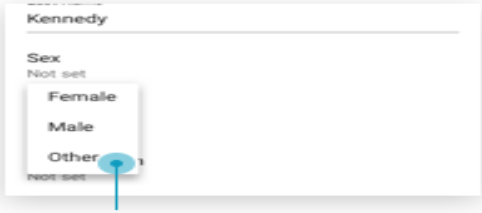
…and the picker control appears

**Android**

Display small choices with (a) an in-place dropdown menu or (b) a modal confirmation dialog.




Tap here…




…and a dropdown menu appears

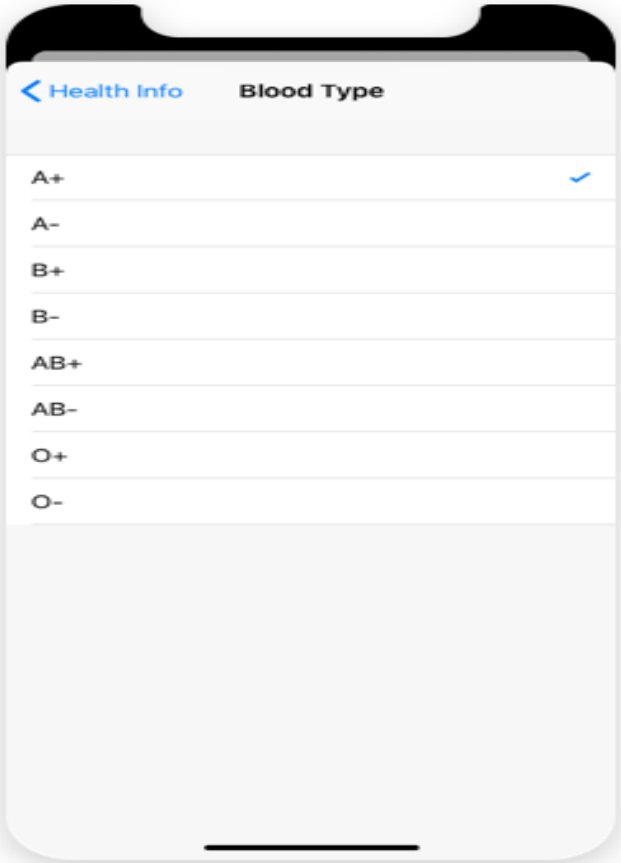

…or a modal, which can be used in the same situation.

# Mobile Operating System differences: Design

## Selection controls: single choice

**iOS**

Single-choice options are displayed with checkmarks. Multiple-choice options use toggle switches.




**Android**

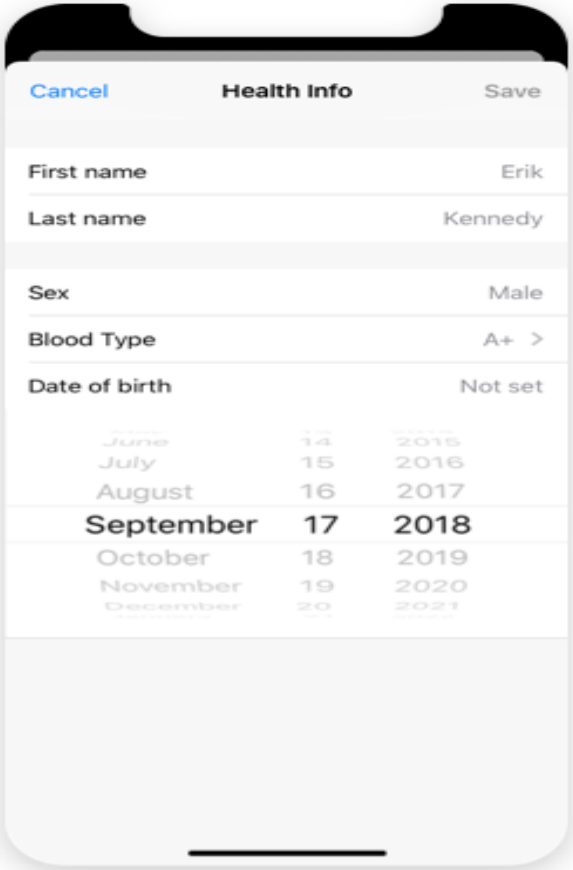
Single-choice options are displayed with radio buttons. Multiple-choice options use checkboxes or toggle switches.

# Mobile Operating System differences: Design

## Date picker

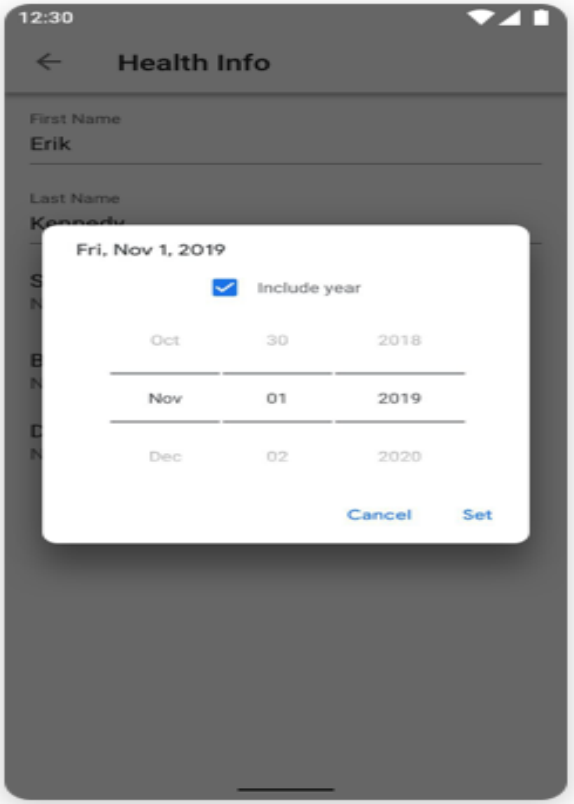
**iOS**

Date pickers are simply spinners that can be (a) anchored to the bottom or (b) displayed inline (as shown).




**Android**

A totally custom control is used for picking dates.

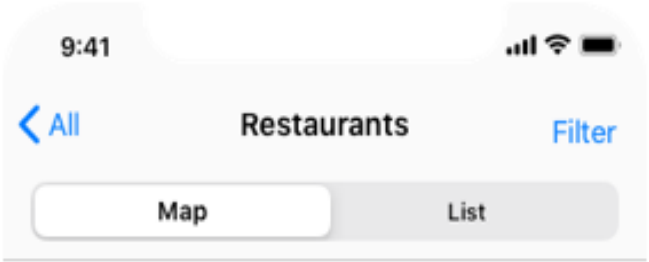


# Mobile Operating System differences: Design

## Tabs

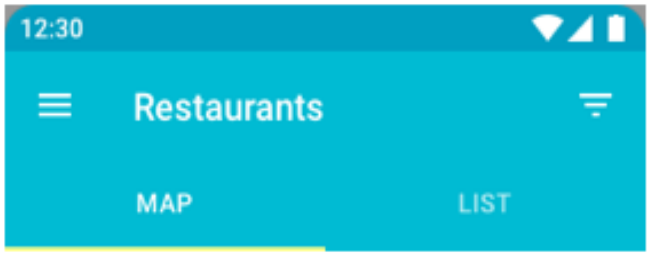
**iOS**

Use a segmented button instead of tabs.




**Android**

Use the default “flat design”-style tabs.



# Mobile Operating System differences: Design

## Default fonts

**iOS**

The primary font is SF.

**San Francisco 34**

San Francisco - 28

San Francisco - 22

San Francisco - 20

San Francisco - 17

San Francisco - 15

San Francisco - 13

San Franciso - 10

"SF Pro Display" used for sizes 20+

**Android**

The primary font is Roboto.

**Roboto - 34**

Roboto - 24

Roboto - 20

Roboto - 16

Roboto - 14

Roboto - 12

Roboto - 10

# Mobile Operating System differences: Design

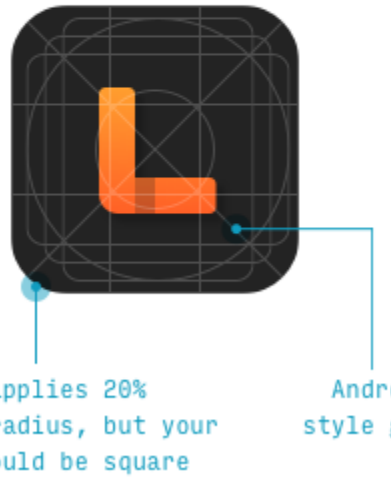
## App icon size and shape

**iOS**

Design your icon at 180x180 px first, and check/tweak/export larger sizes after that.


**Android**

Design your app icon at 48x48 px first, and check/tweak/export larger sizes after that.




| PLATFORM | MINIMUM TAP TARGET SIZE |
|---|---|
| iOS | 44x44pt |
| Android | 48x48dp |

# Mobile Operating System differences: Design


Software design

- User experience and battery life
- Cloud readiness Mobile device Fragmentation
- Openness to players of the mobile ecosystem

# Mobile Operating System differences: Security

Mobile devices frequently contain or have access to sensitive information and must be protected.

Common threats

- Physical attacks – Device loss, Theft and Disposal
- Logical attacks – through network or other communication devices
- Logical attacks on the devices – Rootkits, malicious software and configurations

# Mobile Operating System differences: Security

Android

Has 5 security layers

- Security at the OS level (Linux kernel)

  User based permission model

  Secure mechanism for Inter Process Communication

  Process isolation & ability to clear any unnecessary insecure parts of the kernel

- Mandatory application sandbox

  Uses a user based protection to create an application sandbox that assigns a unique user Id to each app which runs its own process

- Secure inter process communications

  Each app is run at a different process level through the kernel which does not allow apps to interact with each other and only assigns them limited access to the Android OS

# Mobile Operating System differences: Security

Android

- Application signing

  Provides user permission based access control and provides a list of permissions on the first page of the APK so the app may use them installation

- Application defined and user granted permissions

  This feature gives a set of file system permissions so that each app has its own files and except a developer explicitly exhibits files to another Android app, files generated by one app cannotbe read or changed by another one

  If an app needs to access data for another app it must be given access custom permissions

# Mobile Operating System differences: Security

iOS

- Provides APIs to perform security features for developers
- iOS applies Common Data Security Architecture CDSA to perform security features like desktop counterpart and file access permissions on low level properties which by the BSD kernel (Unix OS based kernel)
- CDSA handles higher level functionality eg encryption, security data storage and authentication
- Users have no control on permissions access required by an app for doing its job
- Has a sandbox with limited permissions required by third party apps to run and in the sandbox each app runs separately from other apps on the iOS

# Mobile Operating System differences: Security

Windows

- Utilizes the same security mechanisms of Windows 10 OS
- Windows Hello for business – this provides an identity and access control features that only authorized users could access data and resources. It has a secure multi factor authentication MFA deployment and employs a companion device offering the PIN and biometric authentication methods
- Windows information protection – this technology enables an automatic data separation for preserving corporate information when they are being shared with personal data and apps

# Mobile Operating System differences: Security

Windows

- Malware resistance – this technology applies multi layered protections such as start up processes, hardware devices and app platforms for reducing the threats of malware.

# Mobile Operating System differences: Security

Blackberry

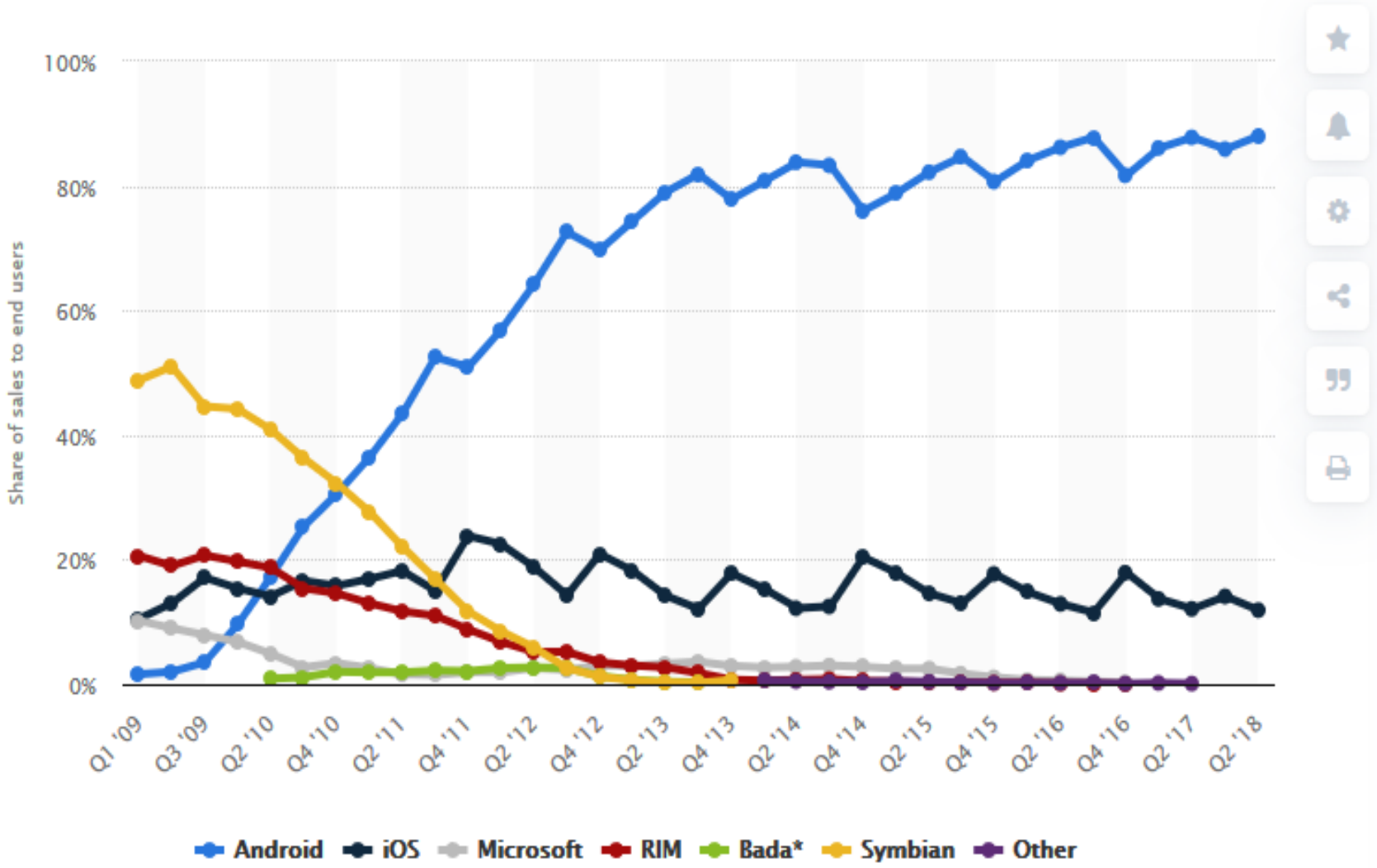
- Platform security – verifies authenticity of the blackberry OS and its apps when the OS boots up. This provides resilience and security protection against tampering, malware and data leakage
- Secure device management – provides the highest level of security for users that can use a specific space for their personal data usage without sacrificing their security needs. It also permits easy access to all the personal accounts and maximizes productivity while seamlessly securing the data.

# Mobile Operating System differences: Security

Blackberry

- Data in transit security - BlackBerry supports a full range of encryption and authentication approaches, allowing the users to safely connect their devices to networks using the BlackBerry infrastructure, VPN, and Wi-Fi.
- App security - This technology assigns to all apps in their own sandboxes for securing against data leakage and malware

# Background on mobile OS market share



# Background on mobile OS market share

Android

- Largest number of devices to choose from
- Frequently updated
- Large number of application available
- Excellent UI
- Multi-tasking
- Free developer tools
- Expandable memory
- Affordable
- No restrictions on applications
- Phones are available from every service provider
- Large community
- Cloud storage (Google cloud)
- Open source

# Background on mobile OS market share

iOS

- Excellent UI
- Apple validates applications
- Consistent UI across devices
- Easy app development due to low fragmentation
- Less heat generation due to effective battery usage
- Easy communication better apple devices
- Multi tasking
- Security – apps are validated
- Device ecosystem

# Background on mobile OS market share

Blackberry

- Secure and excellent Integration with company email systems Windows
- Built in support for Windows Office suite
- Excellent home screen status

# Background on mobile OS market share

Windows

- Quality Build Phones With Exceptional Design For Everyone
- Seamless Email And Social Media Integration
- Microsoft Integration and Support
- Consistency Across All Devices
- Offline Apps eg. Maps
- Expandable storage

# Background on mobile OS market share

Symbian

- Allowed impressive battery life.
- Required lower hardware requirements.
- Low power consumption and high processing performance.
- Easy corporate email integration and highly secure communication