

**Assignment Report**

Aneeb Ur Rehman

221322

Air University

Operating System

Miss Warda Aslam

29/12/2024

[**Comparative Analysis of Android and macOS Operating Systems 3**](#_nze96zoligf1)

[Introduction 3](#_1fob9te)

[Architecture 3](#_i7f7pn98fe8h)

[Pie Chart of OS: 4](#_vu86ci8x62wo)

[Bar Chart of Evaluation Metrics: 4](#_e7yjeeddl76m)

[Objective of Research paper 5](#_3znysh7)

[Process Management 5](#_1t3h5sf)

[Android OS 5](#_4d34og8)

[MAC OS 6](#_i1ytbohhalgr)

[Memory Management 6](#_v8vu6bt7qycw)

[Android OS 6](#_l4bpn3bw6bgt)

[MAC OS 6](#_sz81fd3hvrwu)

[File System 6](#_e8zd7l6chnsw)

[Android OS 6](#_e70nihstjroy)

[MAC OS 6](#_nf0w11ygwev)

[Security 7](#_9uqeqmz5xrlm)

[Android OS 7](#_6g1ga838elgf)

[MAC OS 7](#_kqf43sl2t68e)

[Scheduling 7](#_lg6jo7cvxkcb)

[Android OS 7](#_w6t1cawfetdv)

[MAC OS 7](#_a1srjh1x1rci)

[Creative Analogy 8](#_g2kmauipd08e)

[Comparison of OS concepts(Android & MAC) 8](#_5p8yygjba3wx)

[Comparison Table: 8](#_o003541uj5zv)

[Bar Chart of Comparison table: 9](#_hm0s1vyiobkb)

[Conclusion 9](#_2bn6wsx)

[References 10](#_3as4poj)

[● github link: 10](#_af31db1n2um)

# Comparative Analysis of Android and macOS Operating Systems

## Introduction

Operating systems (OS) manage computer hardware and software resources, enabling programs to run smoothly. Android and macOS are two widely used operating systems, serving different platforms and user needs. Android powers most smartphones and tablets, while macOS is used in Apple’s desktops and laptops, known for seamless integration with Apple’s ecosystem.

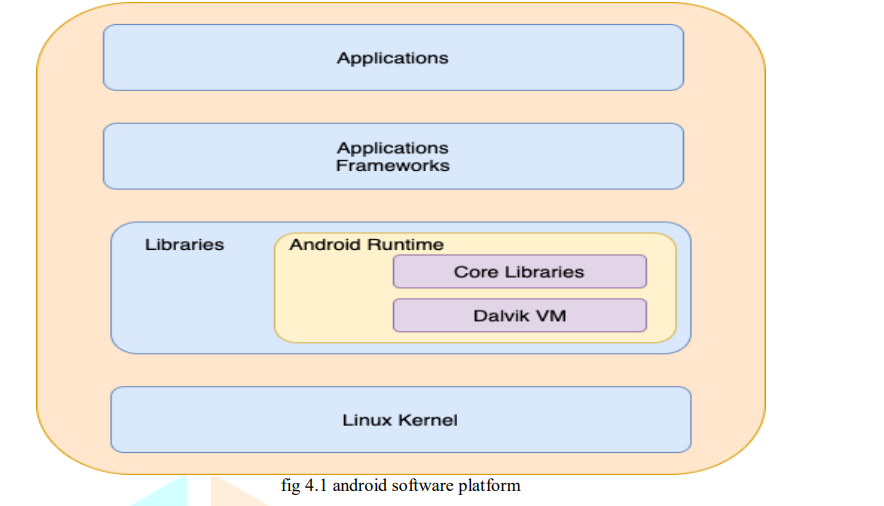
**Android**, developed by Google, is an open-source OS based on the Linux kernel. It offers extensive customization, supports a broad range of devices, and is characterized by its layered architecture that includes the Linux kernel, libraries, Android runtime (ART), and application framework

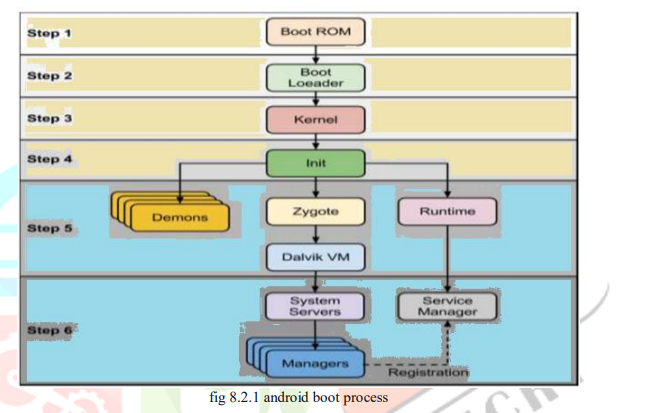
**macOS**, developed by Apple, is a proprietary OS built on a Unix foundation, prioritizing stability, security, and user experience. Its architecture comprises the Darwin kernel, core services, application services, and the user interface

## Architecture

**Android Architecture:** Android's architecture consists of the following layers.

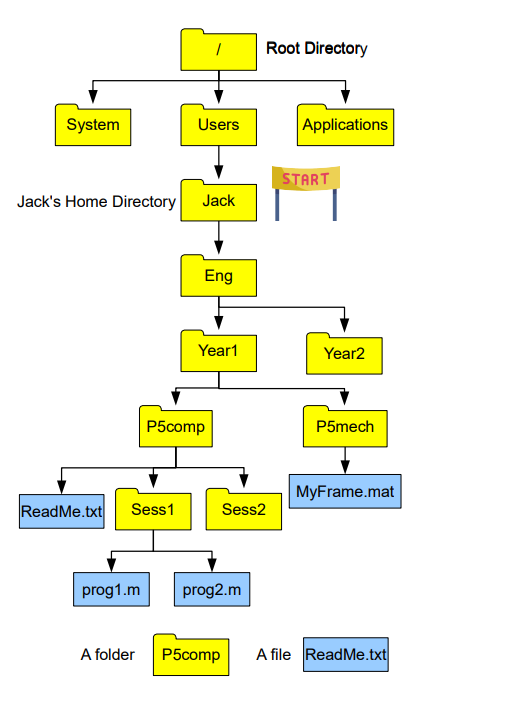
1. **Linux Kernel**: Serves as the foundation, managing core system services such as security, memory, and drivers.
2. **Libraries**: Includes native C/C++ libraries used by various components of the Android system.
3. **Android Runtime (ART)**: Provides a managed environment for running applications, replacing the older Dalvik Virtual Machine.
4. **Application Framework**: Facilitates app development by providing APIs for managing UI components, resources, and lifecycle.
5. **Applications Layer**: Contains system and user-installed apps that interact with the lower layers through the framework.





**macOS Architecture:** macOS architecture comprises the following components

1. **Darwin Kernel**: A Unix-based core providing fundamental OS services like file systems, networking, and process management.
2. **Core Services**: Includes essential APIs for managing features like database access and networking.
3. **Application Services**: Facilitates the development of Mac applications through frameworks like Cocoa.
4. **Aqua User Interface**: Offers the graphical interface, allowing users to interact with applications through intuitive design and visual elements.

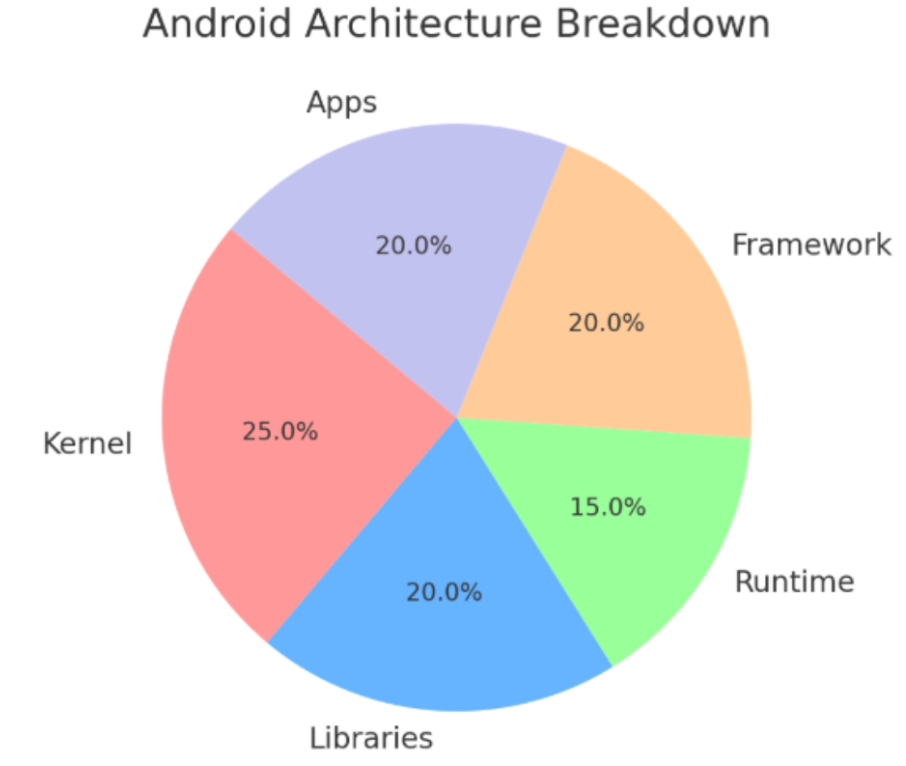


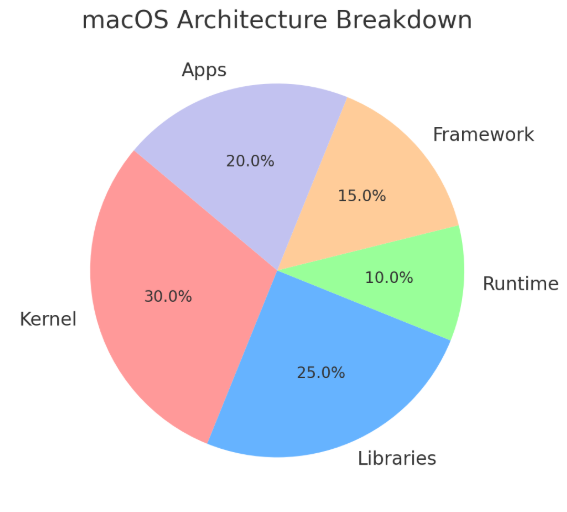
### 

### 

### 

### Pie Chart of OS:

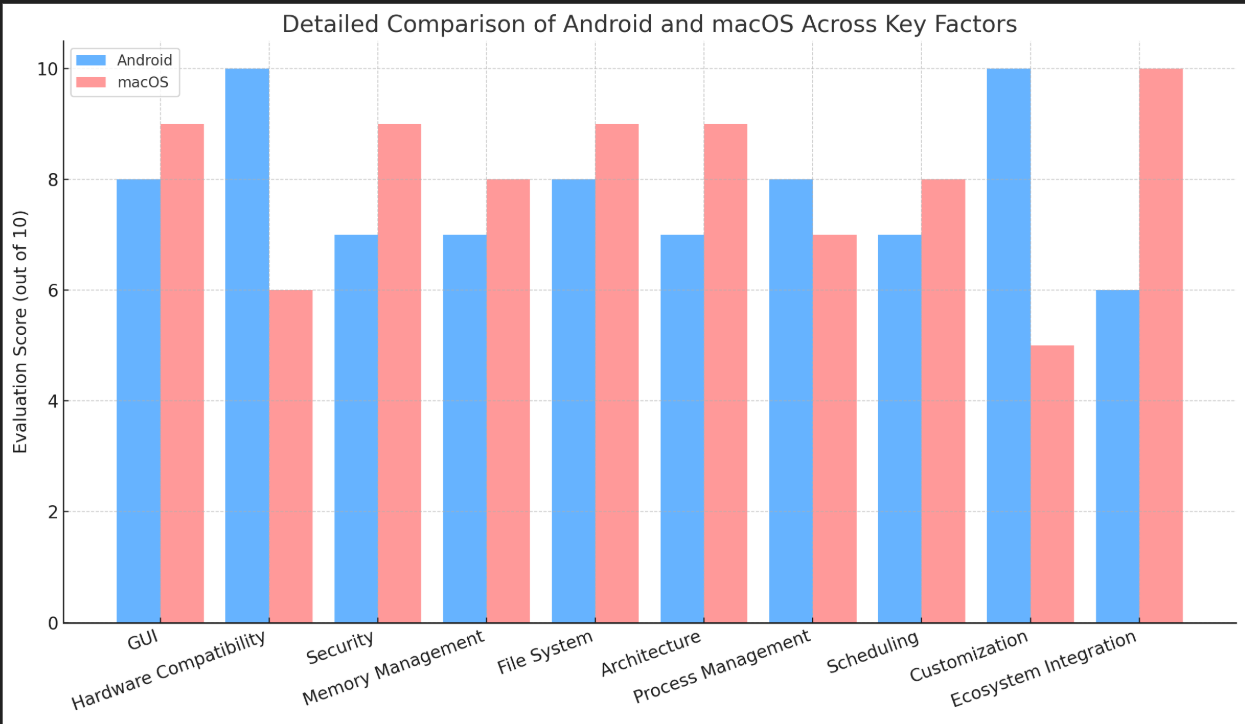




The pie charts represent the architectural breakdown of Android and macOS, segmented by components such as Kernel, Libraries, Runtime, Framework, and Applications.

### 

### Bar Chart of Evaluation Metrics:



The bar chart compares Android and macOS across key factors like GUI, hardware compatibility, security, memory management, file systems, architecture, process management, scheduling, customization, and ecosystem integration.

## Objective of Research paper

The primary objective of the referenced research paper is to evaluate and compare the quality attributes of various operating systems, with a particular focus on performance, security, and usability. The paper investigates how different design choices impact the efficiency, reliability, and user experience of operating systems, highlighting critical factors that contribute to their success in different environments.

**Key Objectives:**

* To analyze and compare the architectural designs of Android and macOS.
* To explore process and memory management techniques in both operating systems.
* To examine file system structures, focusing on how data is organized and accessed.
* To evaluate security mechanisms, including encryption, permissions, and authentication.
* To assess scheduling algorithms and real-time processing capabilities.

## Process Management

### Android OS

* **Process Creation**: Uses the Zygote process, which reduces startup time by cloning pre-initialized app processes.
* **Scheduling**: Implements the Completely Fair Scheduler (CFS) to ensure equal CPU distribution.
* **Multitasking**: Supports preemptive multitasking; background apps can be paused or terminated to save resources.
* **Inter-Process Communication (IPC):** Relies on the Binder framework for efficient IPC between apps and system services.

### MAC OS

* **Process Creation:** Uses UNIX-based fork() and exec() methods to spawn processes.
* **Scheduling**: Employs multilevel feedback queues, prioritizing foreground apps while managing background tasks.
* **Multitasking**: Provides seamless multitasking with robust memory management to ensure smooth operation.
* **IPC**: Utilizes Mach messages and XPC for secure communication between system components.

## 

## Memory Management

### Android OS

* **Memory Allocation**: Managed through Dalvik and ART runtimes for efficient memory usage.
* **Virtual Memory**: Uses zRAM to compress memory and create swap space, enhancing performance.
* **Caching and Protection**: Sandboxes apps to limit access to sensitive data and memory.

### MAC OS

* **Memory Allocation**: Dynamic memory allocation with malloc() and VM paging.
* **Virtual Memory**: Supports memory compression and swap space to handle heavy workloads.
* **Caching and Protection**: FileVault encrypts data, while SIP prevents unauthorized system modifications.

## 

## File System

### Android OS

* **Structure**: Ext4 file system by default, with F2FS for flash storage.
* **Organization**: Divides storage into partitions for system, user data, and cache.
* Supports external SD cards formatted in FAT32 or exFAT.

### MAC OS

* **Structure**: Uses APFS (Apple File System), optimized for SSDs.
* **Organization**: Enables snapshots, encryption, and space sharing.
* APFS improves file integrity with copy-on-write and crash protection features.

## 

## Security

### Android OS

* **Permissions**: Uses SELinux and app-level permissions to restrict access.
* **Encryption**: Supports file-based encryption (FBE) and full-disk encryption.
* **Authentication**: Biometric options (fingerprint, facial recognition) and two-factor authentication.

### MAC OS

* **Permissions**: Gatekeeper and app notarization prevent unauthorized software.
* **Encryption**: FileVault encrypts entire disks to protect sensitive data.
* **Authentication**: Touch ID and system-wide two-factor authentication.

## 

## Scheduling

### Android OS

* **Algorithm**: Completely Fair Scheduler (CFS) ensures equal CPU access.
* **Real**-**Time** **Processing**: Limited real-time processing, focusing on responsiveness.

.

### MAC OS

* **Algorithm**: Uses multilevel feedback queues.
* **Real**-**Time** **Processing**: Capable of handling real-time tasks, benefiting multimedia applications.

## 

## Creative Analogy

Imagine Android as a sprawling metropolis, bustling with various districts (apps) that operate semi-independently. The Binder framework acts as the city’s infrastructure, allowing communication between districts. The open nature of the city attracts a diverse range of developers, fostering rapid innovation but also requiring careful oversight to ensure security and efficiency.

In contrast, macOS resembles a fortified castle, meticulously designed and guarded. Each app is a resident within the castle walls, adhering to strict guidelines enforced by gatekeepers (security features like Gatekeeper and SIP). The castle’s design emphasizes security, stability, and seamless integration, creating a cohesive environment at the expense of some flexibility.

## 

**Insights and personal Observations on OS difference**

* Android’s flexibility allows it to operate across various devices, making it adaptable but complex.
* macOS provides seamless performance through tight integration with Apple hardware.
* Android’s open-source model encourages innovation but increases fragmentation.
* macOS offers better security features but limits customization.
* Android excels in app diversity and availability, while macOS prioritizes stability and user experience.
* macOS’s file system (APFS) is optimized for SSDs, while Android’s ext4 focuses on compatibility.
* Both systems utilize advanced encryption but apply different security philosophies.

## 

## 

## 

## 

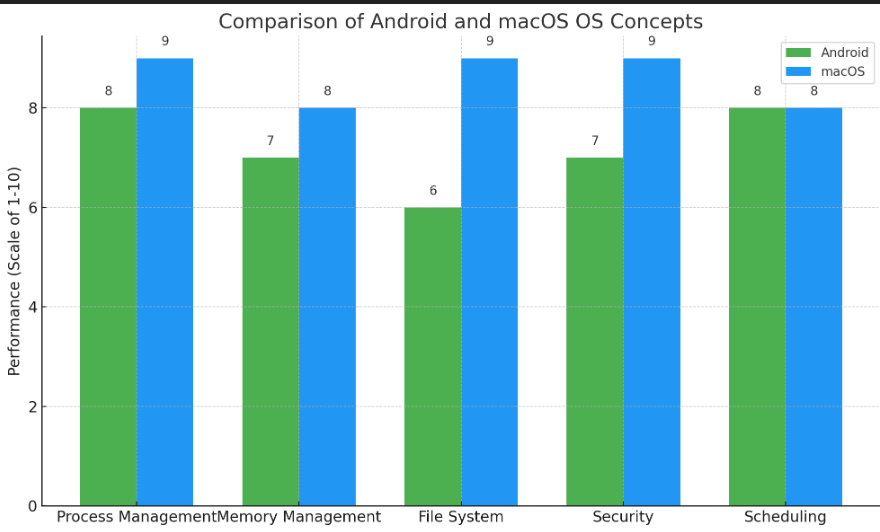
## 

## Comparison of OS concepts(Android & MAC)

### Comparison Table:

| **OS** | **Android** | **MAC** |
| --- | --- | --- |
| **Process Management** | Uses Zygote process for fast app creation, CFS for scheduling | Based on UNIX used fork() and exec() for process creation |
| **Process Management** | Dynamic memory allocation with zRAM and paging | Dynamic memory allocation with VM compression and paging |
| **File System** | Ext4, F2FS, FAT32 | APFS, HFS+ |
| **Security** | SELinux, FDE, runtime permissions | SIP, FileVault, Gatekeeper |
| **Scheduling** | CFS Algorithm, Fair distrubution | Multilevel feedback queue, supports real-time tasks |

### Bar Chart of Comparison table:

**

Here's a visual comparison chart of Android and macOS based on key OS concepts.

## 

## Conclusion

Android and macOS are exemplary operating systems that thrive in different domains. Android's open-source nature and adaptability make it the leading mobile OS globally, supporting a wide range of devices and fostering innovation. In contrast, macOS prioritizes security, stability, and performance, making it ideal for professional environments and creative industries.

The comparative analysis reveals that while Android offers greater flexibility and customization, macOS stands out for its security and seamless integration. Each system reflects the goals and philosophies of their respective developers, highlighting the diverse approaches to operating system design and functionality. Understanding these differences allows users and developers to make informed decisions when selecting an OS that best meets their needs.

## 

## References

* Mear, I., & Peasley, E. *Operating System Basics for Mac*. University of Oxford.
* Jaiswal, M. *Android the Mobile Operating System and Architecture*. IJCRT.

### github link: https://github.com/Aneeb-ur-Rehman/OS-Assignment-