

KOMPARASI STRUKTUR DIREKTORI PADA SISTEM OPERASI MOBILE ANDROID DAN IOS**Anggita Risqi Nur Clarita¹, Faizah Via Fadhillah², Zahra Nurhaliza³, Elkin Rilvani⁴**Program Studi Teknik Informatika, Fakultas Teknik
Universitas Pelita Bangsa

Jl. Inspeksi Kalimalang Tegal Danas N0.9, Cikarang Pusat, Kab.Bekasi, Indonesia

anggitarisqi312210450@mhs.pelitabangsa.ac.id¹,faizahviafadhillah.312210460@mhs.pelitabangsa.ac.id²,zahranurhaliza.312210364@mhs.pelitabangsa.ac.id³, elkin.rilvani@pelitabangsa.ac.id⁴**Abstract**

This research aims to compare directory structures on Android and iOS mobile operating systems using a literature study approach. In this study, an in-depth analysis of previous research results related to the directory structure of the two operating systems was conducted, including hierarchy, function, and flexibility of use. The results show that Android has a more open and flexible directory structure than iOS, which tends to be more closed and tightly controlled by the system. The findings provide new insights into the fundamental differences between Android and iOS in file management and data security. In conclusion, this research confirms the importance of understanding the directory structure to support optimized usage and application development on both platforms.

Abstrak

Penelitian ini bertujuan untuk membandingkan struktur direktori pada sistem operasi mobile Android dan iOS dengan menggunakan pendekatan studi literatur. Dalam studi ini, dilakukan analisis mendalam terhadap hasil penelitian terdahulu terkait struktur direktori kedua sistem operasi, meliputi hierarki, fungsi, dan fleksibilitas penggunaannya. Hasil penelitian menunjukkan bahwa Android memiliki struktur direktori yang lebih terbuka dan fleksibel dibandingkan iOS, yang cenderung lebih tertutup dan terkontrol ketat oleh sistem. Temuan tersebut memberikan wawasan baru mengenai perbedaan mendasar antara Android dan iOS dalam pengelolaan file dan keamanan data. Kesimpulannya, penelitian ini menegaskan pentingnya memahami struktur direktori untuk mendukung optimalisasi penggunaan dan pengembangan aplikasi pada kedua platform.

Article History

Submitted: 29 Desember 2024

Accepted: 1 Januari 2025

Published: 6 Januari 2025

Key Words

Android, iOS, Mobile Storage Management, Structured Directory

Sejarah Artikel

Submitted: 29 Desember 2024

Accepted: 1 Januari 2025

Published: 6 Januari 2025

Kata Kunci

Android, iOS, Manajemen Penyimpanan Mobile, Struktur Direktori.

PENDAHULUAN

Sistem operasi adalah otak di balik layar yang membuat perangkat digital dapat bekerja dengan lancar dan baik, mulai dari mengelola memori, mengatur proses, hingga mengatur data dalam struktur yang tertata rapi. Salah satu aspek menarik dari sistem operasi adalah bagaimana mereka menyusun struktur direktori untuk mengelola *file* dan data (Umar, 2023). Direktori, sebagai wadah yang digunakan untuk menampung folder dan berkas, memainkan peran penting dalam pengaturan data. Mereka mengatur berkas dan folder secara hierarkis, mirip dengan cara kita menyusun dokumen di dalam folder. Dengan struktur ini, sistem operasi dapat melacak dan mengelola berkas secara efisien, yang mempermudah pencarian dan pengelolaan data. Struktur direktori ini lebih dari sekadar tumpukan folder yang terlihat biasa, mereka adalah fondasi dari cara kerja sebuah perangkat (GeeksforGeeks, n.d.).

Secara umum, struktur direktori dalam sistem operasi memiliki 4 skema utama yang sering digunakan untuk mendefinisikan logika penyusunan direktori, yaitu *single-level directory*, *two-level directory*, *tree-structured directories*, dan *acyclic-graph directories* (Trivedi, n.d.). Setiap skema memiliki kelebihan dan kekurangan masing-masing, tergantung

pada kebutuhan sistem yang menerapkannya. Pemahaman tentang skema ini membantu menjelaskan perbedaan dalam pendekatan yang digunakan oleh berbagai sistem operasi.

◆ Saat ini terdapat beberapa sistem operasi pada komputer dan *Smartphone* yaitu, *Windows*, *macOs*, *Linux*, *Android* dan *iOS* (Rahadian, n.d.). Setiap sistem operasi ini memiliki karakteristik unik, termasuk dalam cara mereka mengatur struktur direktori. Misalnya, *Android* dan *iOS*, dua sistem operasi paling populer di dunia ponsel pintar, memiliki pendekatan yang berbeda dalam mengatur struktur direktori mereka. Ini bukan hanya soal preferensi desain, tetapi juga berkaitan erat dengan filosofi keamanan, fleksibilitas, dan pengalaman pengguna yang ingin dicapai oleh masing-masing platform (Abdualrahman & Alzahrani, 2024).

Android, sebagai sistem operasi *open source* yang paling banyak digunakan, memiliki struktur direktori yang berbasis *Linux* dan dapat diakses oleh pengguna yang memiliki perangkat yang di-root (Kapetanios, Patsakis, Alepis, & Polyzos, 2020). Di sisi lain, *iOS* memiliki pendekatan yang lebih terkontrol terhadap akses direktori, mencerminkan filosofi keamanan *Apple* yang lebih ketat. Perbedaan ini memiliki implikasi signifikan terhadap pengembangan aplikasi, manajemen file, dan keamanan data pengguna.

Sistem operasi *Android* dan *iOS* menghadapi sejumlah tantangan dalam mengelola struktur direktori, termasuk kapasitas penyimpanan, kecepatan akses, dan keandalan data. Hal ini menimbulkan pertanyaan tentang bagaimana perbedaan struktur direktori antara kedua sistem operasi tersebut mempengaruhi keamanan, fleksibilitas, dan pengalaman pengguna. Bagaimana tantangan dalam manajemen kapasitas penyimpanan dan kecepatan akses diatasi oleh *Android* dan *iOS* melalui struktur direktori masing-masing. Selain itu, bagaimana kedua sistem operasi ini mengelola struktur direktori untuk menjamin bahwa data tetap aman dan dapat diakses dengan cepat.

Pembahasan dalam penelitian ini akan difokuskan pada versi sistem operasi *Android 15* dan *iOS 15*. Dengan memilih kedua versi ini, penelitian diharapkan dapat mencakup teknologi dan fitur terbaru yang mempengaruhi pengelolaan struktur direktori, terutama terkait keamanan dan efisiensi akses data. *Android 15* dan *iOS 15* membawa sejumlah peningkatan dibandingkan versi sebelumnya, khususnya dalam aspek keamanan data dan manajemen penyimpanan yang lebih responsif terhadap kebutuhan pengguna modern.

Penelitian ini bertujuan untuk memahami pendekatan masing-masing sistem operasi dalam mengelola struktur direktori dan menghadapi tantangan kapasitas penyimpanan serta kecepatan akses. Dengan menganalisis perbedaan antara *Android* dan *iOS*, penelitian ini diharapkan dapat memberikan gambaran mengenai cara keduanya menangani masalah keamanan, fleksibilitas akses, dan integritas data. Adapun manfaat dari penelitian ini adalah memberikan wawasan kepada pengembang aplikasi untuk merancang aplikasi yang lebih aman dan efisien dengan mempertimbangkan karakteristik tiap sistem operasi. Penelitian ini juga dapat membantu pengguna untuk lebih memahami pengelolaan data pada perangkat mereka, serta menjadi referensi bagi penelitian lebih lanjut di bidang sistem operasi *mobile*, khususnya dalam pengembangan manajemen struktur direktori yang lebih baik.

PENELITIAN RELEVAN

Penelitian ini didukung oleh beberapa penelitian yang membahas perbedaan antara struktur direktori pada *Android* dan *iOS*. Penelitian yang telah dilakukan oleh Ruqiya et al. (2020) menunjukkan bahwa *Android* memiliki struktur direktori yang fleksibel dan memungkinkan akses pengguna yang lebih luas, sehingga memudahkan penelusuran data dan pengaturan file (Abdualrahman & Alzahrani, 2024). Sebaliknya, menurut penelitian yang dilakukan oleh Erika G. Llabres dan Esli Joy N. Fernando (2023), *iOS* mengutamakan keamanan dan privasi dengan struktur tertutup yang membatasi akses aplikasi terhadap direktori, meningkatkan perlindungan data pengguna (Llabres & Fernando, 2023).

Penelitian yang dilakukan oleh Rajat Imam Nivade et al. (2022), memperkuat hal ini dengan menggarisbawahi perbedaan aksesibilitas dan keamanan antara kedua sistem (Imam Nivade Rajat & Sanjay Ajinkya, 2022). Penelitian ini tidak hanya membandingkan struktur direktori secara teknis, tetapi juga melihat bagaimana hal tersebut memengaruhi pengalaman pengguna. Dengan menggabungkan faktor keamanan dan aksesibilitas, penelitian ini bertujuan memberikan gambaran menyeluruh tentang keunggulan masing-masing OS. Diharapkan, penelitian ini dapat memberi wawasan lebih dalam tentang bagaimana sistem manajemen direktori mempengaruhi penggunaan sehari-hari, terutama dalam hal fleksibilitas dan keamanan. Selain itu, peneliti juga akan menemukan referensi-referensi penting untuk mendalami perbandingan kedua sistem operasi tersebut.

METODE PENELITIAN

Dalam penelitian ini, peneliti menggunakan tiga metode utama, yaitu Studi Pustaka, Studi Literatur, dan Analisis Data. Setiap metode memiliki peran penting dalam pengumpulan dan pengolahan data untuk mencapai tujuan penelitian.

Studi Pustaka

Studi pustaka adalah metode yang digunakan untuk mengumpulkan informasi dari berbagai sumber tertulis yang relevan dengan topik penelitian. Studi pustaka mencakup kegiatan membaca, mencatat, dan mengolah bahan dari jurnal, buku, *e-book*, dan artikel (Fia Anggita Zalsabila, 2024). Melalui studi pustaka ini, peneliti mendapatkan pemahaman dasar mengenai struktur direktori pada sistem operasi *Android* dan *iOS*, yang menjadi landasan utama dalam penelitian ini.

1. Struktur Direktori

Struktur direktori berfungsi sebagai tabel simbol yang menerjemahkan nama *file* ke dalam entri direktori. Pengorganisasian direktori harus memungkinkan proses seperti penambahan, penghapusan, pencarian entri bernama, serta pembuatan daftar semua entri direktori (SILBERSCHATZ ABRAHAM et al., 2012). Struktur ini menjadi elemen penting dalam manajemen *file* pada sistem operasi, termasuk *Android* dan *iOS*.

2. *Android*

Android adalah sistem operasi berbasis *Linux* yang dirancang khusus untuk perangkat seluler seperti *smartphone* dan tablet. Sistem ini menyediakan lapisan perangkat lunak lengkap, mulai dari kernel hingga aplikasi. Salah satu keunggulan utama *Android* adalah fleksibilitasnya dalam mendukung berbagai jenis perangkat keras melalui penggunaan *Hardware Abstraction Layer (HAL)*. *HAL* berfungsi sebagai antarmuka antara perangkat keras dan *framework Android*, memungkinkan sistem operasi untuk berkomunikasi dengan perangkat keras tanpa bergantung langsung pada implementasi spesifiknya (Ye, 2017).

3. *iOS*

iOS adalah sistem operasi *mobile* yang dikembangkan oleh *Apple Inc.*, dirancang khusus untuk perangkat seperti *iPhone*, *iPad*, dan *iPod touch*. Berbasis pada kernel *Unix*, *iOS* menawarkan pengalaman pengguna yang intuitif dengan antarmuka grafis yang sederhana dan dukungan aplikasi yang ketat melalui *App Store* (Keur & Hillegass, 2016). Keunggulan utama dari *iOS* adalah tingkat keamanannya yang tinggi serta integrasi yang mulus dengan ekosistem perangkat *Apple* lainnya.

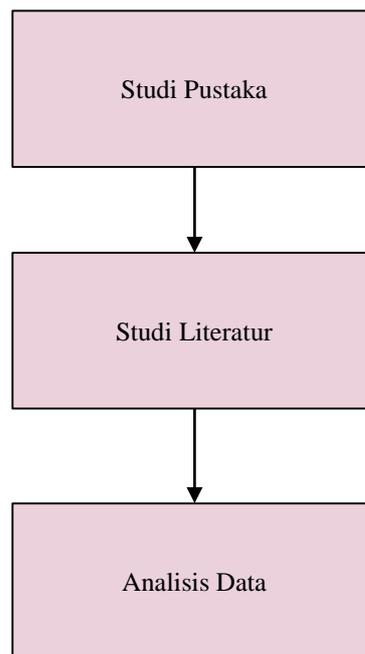
Studi Literatur

Penelitian sebelumnya menunjukkan bahwa *Android* memiliki struktur direktori yang fleksibel, memungkinkan akses pengguna yang lebih luas untuk memudahkan penelusuran data dan pengaturan file (Ruqiya et al., 2020). Sebaliknya, *iOS* lebih menekankan pada keamanan dan privasi dengan struktur direktori tertutup yang membatasi akses aplikasi terhadap data

pengguna (Llabres & Fernando, 2023). Selain itu, terdapat perbedaan signifikan antara aksesibilitas dan keamanan pada kedua sistem operasi ini (Imam Nivade Rajat & Sanjay Ajinkya, 2022).

Analisis Data

Metode ini digunakan untuk menganalisis data yang telah diperoleh dari hasil studi pustaka dan studi literatur. Data ini kemudian dibandingkan untuk melihat perbedaan dan persamaan dalam pengelolaan struktur direktori antara *Android* dan *iOS*, terutama dalam hal keamanan, kecepatan akses, dan keandalan penyimpanan data. Melalui ketiga metode ini, peneliti dapat memberikan gambaran yang lebih komprehensif mengenai peran struktur direktori pada kedua sistem operasi.



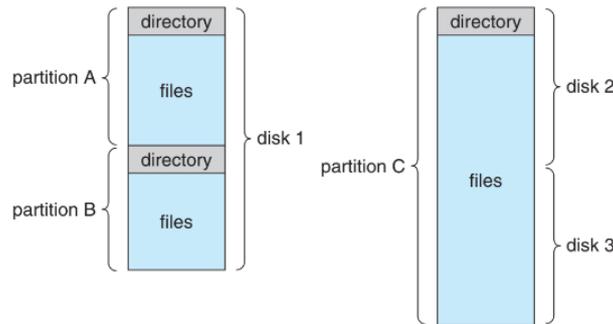
Gambar 1. Alur Metode Penelitian

HASIL DAN PEMBAHASAN

Struktur Direktori dalam OS

Struktur direktori adalah komponen penting dalam sistem operasi yang berfungsi untuk mengorganisir *file* dan folder di dalam sistem. Direktori berperan sebagai tabel simbol yang menghubungkan nama *file* dengan lokasi fisik mereka, memungkinkan pengguna dan sistem untuk mengelola data secara efisien. Dalam pembahasan ini, kami akan mengeksplorasi bagaimana struktur direktori pada sistem operasi *Android* dan *iOS* mempengaruhi pengalaman pengguna, termasuk faktor kemudahan akses dan keamanan.

Organisasi sistem *file* menggambarkan bagaimana *file* dan direktori diatur di dalam sistem operasi. Struktur ini sangat penting untuk memastikan bahwa data dapat diakses dengan efisien dan aman.



Gambar 2. Organisasi Sistem File
(Sumber: Abraham Silberschatz, 2012)

Sistem *file* Solaris menawarkan metode pengorganisasian *file* yang berbeda, memberikan keunggulan dalam pengelolaan data dalam skala besar, dengan mempertimbangkan fleksibilitas dan keamanan.

/	ufs
/devices	devfs
/dev	dev
/system/contract	ctfs
/proc	proc
/etc/mnttab	mntfs
/etc/svc/volatile	tmpfs
/system/object	objfs
/lib/libc.so.1	lofs
/dev/fd	fd
/var	ufs
/tmp	tmpfs
/var/run	tmpfs
/opt	ufs
/zpbge	zfs
/zpbge/backup	zfs
/export/home	zfs
/var/mail	zfs
/var/spool/mqueue	zfs
/zpbg	zfs
/zpbg/zones	zfs

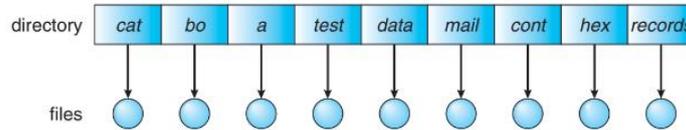
Gambar 3. Sistem *File* Solaris
(Sumber: Abraham Silberschatz, 2012)

Struktur direktori dalam sistem operasi memiliki empat skema utama yang digunakan untuk mendefinisikan logika penyusunan direktori. Keempat skema ini *single-level directory*, *two-level directory*, *tree-structured directories*, dan *acyclic-graph directories* mewakili cara-cara berbeda dalam mengorganisir *file* dan direktori agar dapat diakses dengan mudah. Setiap skema memiliki kelebihan dan kelemahan yang mempengaruhi efisiensi dan fleksibilitas dalam pengelolaan *file* dalam sistem operasi.

1. *Single-Level Directory*

Single-Level Directory adalah struktur direktori yang paling sederhana, di mana semua

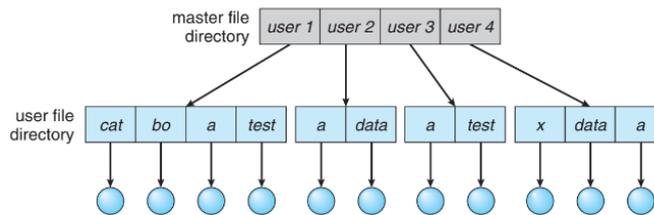
file disimpan dalam satu direktori yang sama. Struktur ini mudah dipahami dan dikelola, namun memiliki keterbatasan signifikan, terutama saat jumlah *file* meningkat atau ketika sistem digunakan oleh lebih dari satu pengguna. Setiap nama *file* harus unik karena semuanya berada dalam direktori yang sama, yang dapat menambah kesulitan dalam pengorganisasian data seiring waktu (SILBERSCHATZ ABRAHAM et al., 2012).



Gambar 4. Struktur *Single-Level Directory*
(Sumber: Abraham Silberschatz, 2012)

2. *Two-Level Directory*

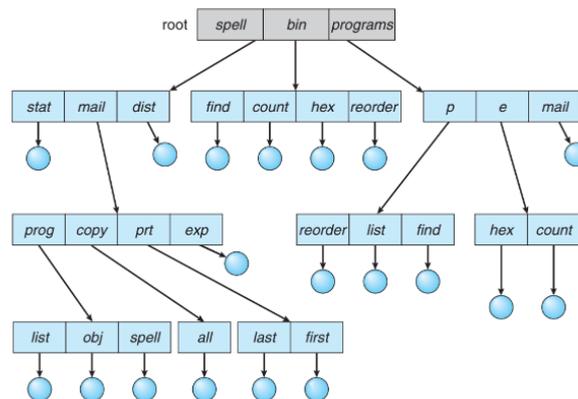
Two-Level Directory adalah struktur yang membagi direktori menjadi dua tingkat: direktori utama (*master directory*) dan subdirektori. Sistem ini memungkinkan setiap pengguna memiliki subdirektori sendiri, sehingga *file* milik masing-masing pengguna dapat diorganisir secara terpisah. Kelebihannya adalah tidak perlu memberikan nama *file* yang unik di seluruh sistem, cukup di dalam subdirektornya. Namun, interaksi antar direktori pengguna tidak didukung secara langsung, membatasi fleksibilitas (SILBERSCHATZ ABRAHAM et al., 2012).



Gambar 5. Struktur *Two-Level Directory*
(Sumber: Abraham Silberschatz, 2012)

3. *Tree-Structured Directories*

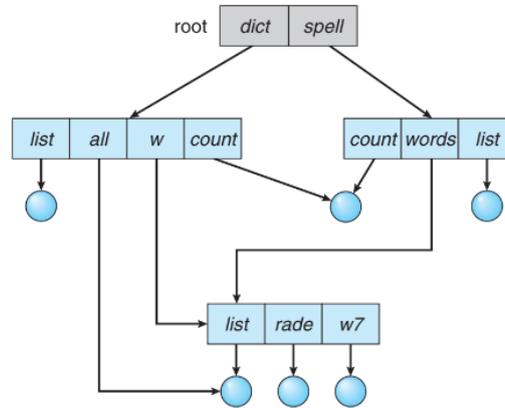
Tree-Structured Directories adalah hierarki direktori di mana direktori utama dapat memiliki subdirektori, dan subdirektori dapat memiliki subdirektori lainnya. Struktur ini mendukung pengelolaan *file* yang lebih kompleks tetapi memerlukan navigasi hierarki yang efisien (SILBERSCHATZ ABRAHAM et al., 2012). Selain memiliki direktori *root*, setiap *file* sistem memiliki jalan yang berbeda.



Gambar 6. *Tree-Structured Directories*
(Sumber: Abraham Silberschatz, 2012)

4. Acyclic-Graph Directories

Acyclic-Graph Directories memungkinkan file atau direktori diakses melalui lebih dari satu jalur tanpa membentuk siklus. Struktur ini ideal untuk berbagi file antar pengguna tetapi memerlukan sistem referensi untuk mencegah kesalahan pengelolaan file (SILBERSCHATZ ABRAHAM et al., 2012).



Gambar 7. Acyclic-Graph Directories (Sumber: Abraham Silberschatz, 2012)

Perbandingan Struktur Direktori pada Android dan iOS

Android telah berkembang lebih cepat daripada iOS. Perkembangan cepat ini dapat dikaitkan dengan keterbukaan platform. Meskipun perbandingan antara Android dan iOS dapat dilakukan dari berbagai sudut pandang, pada akhirnya keduanya menghadapi tantangan yang serupa dalam memenuhi kebutuhan pengguna. Berikut tabel yang membahas beberapa fitur yang membedakan Android dan iOS (Imam Nivade Rajat & Sanjay Ajinkya, 2022).

Tabel 1. Perbandingan Android & iOS

Attributes	Android 15	iOS 15
Developer	Google	Apple
OS Family	Linux	OS X, Unix
Widgets	Yes	No, except in Notification Center
Programmed in	C, C++, Java	C, C++, objective-C
Messaging	Google Hangouts	iMessage
Internet browsing	Google Chrome (or Android Browser on older versions)	Mobile Safari
Source model	Open Source	Closed, with open source components
Video chat	Google Hangouts	Facetime
App Store	Google Play - 1,000,000+ Apps. Other app stores like Apple app store - 850,000+ Amazon and Getjar also Apps distribute Android apps.	
Device manufacturer	Google, LG, Samsung, HTC, Sony, ASUS, and many more	Apple Inc

Android, yang dikembangkan oleh Google, adalah sistem operasi berbasis Linux yang dimodelkan sebagai sumber terbuka. Hal ini memungkinkan komunitas di seluruh dunia untuk mengembangkan dan menyesuaikan aplikasi yang menawarkan fleksibilitas dan personalisasi yang tinggi. Sistem ini meningkatkan pengalaman pengguna dengan mendukung widget di layar utama. Aplikasi Android biasanya dibuat menggunakan bahasa pemrograman seperti C, C++, dan Java. Google Hangouts adalah aplikasi komunikasi bawaan Android, tetapi pengguna dapat menggantinya dengan aplikasi pesan lainnya. Salah satu browser yang paling umum digunakan adalah Google Chrome atau Android Browser dalam versi sebelumnya.

Sedangkan iOS yang dikembangkan oleh Apple, adalah sistem operasi bermodel sumber tertutup (closed source) berbasis OS X dan Unix. Ini memberikan Apple penuh kontrol atas pengembangan dan integrasi sistem. Widget awalnya hanya dapat diakses melalui Notification Center, tetapi fitur ini terus diperbarui dalam versi terbaru. Aplikasi yang dibuat untuk iOS dibuat menggunakan bahasa pemrograman seperti C++, Objective-C, dan lainnya. Salah satu aplikasi bawaan iOS adalah iMessage untuk berkomunikasi melalui pesan dan FaceTime untuk panggilan video, yang keduanya hanya tersedia untuk ekosistem Apple. Safari Mobile adalah browser iOS utama.

1. Struktur Direktori pada *Android*

Android merupakan, sistem operasi yang dikembangkan oleh *Google*, berbasis *Linux*. Untuk meningkatkan pengalaman pengguna, sistem ini mendukung penggunaan widget. Bahasa pemrograman seperti C, C++, dan Java biasanya digunakan untuk membuat aplikasi untuk *Android*. *Android* menggunakan *Google Hangouts* untuk kebutuhan pesan, tetapi pengguna dapat menggunakan aplikasi pesan lainnya. *Android* menggunakan model sumber terbuka (*open source*), yang memungkinkan komunitas di seluruh dunia untuk mengembangkan dan menyesuaikan aplikasinya, seperti *Google Chrome* atau *Android Browser* dalam versi lama. *Google Hangouts* adalah pilihan utama *Android* untuk komunikasi video (Imam Nivade Rajat & Sanjay Ajinkya, 2022).



Gambar 8. *Android 15 Version*

Struktur direktori pada *Android 15* didasarkan pada skema *tree-structured directories*, yang mengorganisir direktori secara hierarkis untuk mengelola *file* sistem, data aplikasi, dan *file* pengguna.

Tabel 2. Direktori utama pada *Mobile Android* Versi 15

Direktori	Fungsi	Contoh File/Folder
<i>/system/</i>	Menyimpan file sistem operasi Android.	Kernel <i>Linux</i> , <i>library</i> , dan aplikasi sistem, seperti kontak dan pesan.
<i>/data/</i>	Menyimpan data aplikasi seperti cache dan database.	Folder <i>sandbox</i> sendiri yang terletak di bawah <i>/data/data/</i> . Misalnya, untuk menyimpan data <i>WhatsApp</i> ,

<i>/mnt/</i>	atau Penyimpanan file media	<i>folder sandbox</i> tersebut berada di <i>/data/data/com.whatsapp/</i> .
<i>/storage/</i>	pengguna.	Gambar, <i>video</i> , dan dokumen, menggunakan <i>/mnt/sdcard/</i> atau <i>/storage/emulated/0/</i> .
<i>/dev/</i>	Berisi file perangkat keras.	Kamera dan sensor.
<i>/etc/</i>	Menyimpan file konfigurasi sistem.	Pengaturan jaringan.

Akses ke direktori ini terbatas tanpa *rooting*. Pengguna biasa hanya dapat mengakses penyimpanan internal di */mnt/*, sedangkan direktori sistem seperti */system/* dan */data/* memerlukan hak akses *root*.

2. Struktur Direktori pada *iOS*

iOS adalah sistem operasi yang dibuat oleh *Apple* dan didasarkan pada OS X dan *Unix*. Pada awalnya, *widget* hanya dapat digunakan melalui *Notification Center* pada *iOS*, dan fitur ini terus diperbarui di versi terbaru. *iOS* membuat aplikasi menggunakan bahasa seperti C, C++, dan *Objective-C*. Untuk kebutuhan pesan, *iOS* menggunakan *iMessage* sebagai aplikasi bawaan yang eksklusif untuk ekosistem *Apple*. Selain itu, *browser* utama *iOS* adalah *Mobile Safari*. Meskipun beberapa komponen *iOS* bersifat *open source*, sistem operasi tersebut menggunakan model sumber tertutup. *FaceTime*, aplikasi bawaan untuk komunikasi *video*, tersedia di semua perangkat *Apple* (Imam Nivade Rajat & Sanjay Ajinkya, 2022).



Gambar 9. *iOS 15 Version*

Struktur direktori pada *iOS 15* juga menggunakan skema *tree-structured directories*, dengan penekanan kuat pada keamanan dan isolasi data aplikasi melalui sistem *sandboxing*.

Tabel 3. Direktori utama pada *iOS* Versi 15

<i>Directori</i>	<i>Fungsi</i>	<i>Contoh File/Folder</i>
<i>/System/</i>	Menyimpan <i>file</i> sistem operasi inti.	Kernel, <i>library</i> , dan aplikasi bawaan seperti <i>Safari</i> dan <i>Mail</i> . Baik pengguna maupun aplikasi pihak ketiga tidak dapat mengubahnya.
<i>/private/var/</i>	Menyimpan semua pengguna dan aplikasi.	<i>/private/var/mobile/Containers/Data/Aplication/</i> digunakan untuk folder <i>sandbox</i> aplikasi tertentu. <i>File</i> media pengguna seperti foto, musik, dan <i>video</i>

Jurnal Ilmiah Sain dan Teknologi

disimpan di
/private/var/mobile/Media/.

Berisi *file* aplikasi bawaan dan aplikasi pihak ketiga yang telah diinstal. Kamera, *Photo*, *WhatsApp*, dll.

Akses ke direktori ini sangat terbatas. Pengguna umum hanya dapat mengakses *file* media melalui aplikasi bawaan (misalnya *Files* atau *Photos*), sementara akses langsung ke direktori seperti */System/* memerlukan proses *jailbreak*.

SIMPULAN

Berdasarkan analisis di atas dapat disimpulkan perbedaan antara *Android* dan *iOS* untuk mengelola struktur direktori dan dampaknya terhadap keamanan, fleksibilitas, dan pengalaman pengguna. *Android* dan *iOS* memiliki pendekatan yang berbeda untuk struktur direktori, yang mencerminkan filosofi masing-masing sistem operasi dalam hal keamanan, fleksibilitas, dan pengalaman pengguna. *Android* mengutamakan fleksibilitas dengan struktur direktori yang lebih terbuka, seperti */data/* dan */storage/*, yang memungkinkan pengguna mengakses dan memodifikasi *file* dengan lebih mudah. Namun, pendekatan ini meningkatkan risiko keamanan, terutama pada perangkat tanpa perlindungan tambahan. Sebaliknya, *iOS* menawarkan tingkat keamanan dan privasi yang lebih tinggi melalui struktur direktori tertutup, seperti */private/var/* dan *sandboxing*, yang secara ketat membatasi akses aplikasi terhadap data pengguna, meskipun mengurangi fleksibilitas pengguna.

Dalam Menghadapi tantangan manajemen kapasitas penyimpanan dan kecepatan akses, *Android* menggunakan struktur direktori berbasis *Linux* seperti */data/* dan */storage/*, dan mendukung *file system ext4* untuk efisiensi dan pemulihan data. Di sisi lain, *iOS* menggunakan *Apple File System (APFS)*, yang dirancang untuk efisiensi penyimpanan kecepatan akses tinggi dan enkripsi otomatis sehingga memberikan pengalaman pengguna yang lebih aman dan responsif.

Untuk menjaga keamanan dan memastikan data tetap dapat diakses dengan cepat, *Android* menggunakan sistem izin aplikasi dan API standar untuk pengelolaan *file*, sementara *iOS* menggunakan enkripsi tingkat sistem dan pendekatan *sandboxing*, yang membatasi akses aplikasi terhadap data pengguna. Hasilnya, *Android* lebih cocok untuk pengguna yang membutuhkan fleksibilitas tinggi, sementara *iOS* lebih unggul dalam keamanan dan integrasi sistem.

Pendekatan yang digunakan oleh masing-masing sistem operasi untuk mengelola struktur direktori menunjukkan bahwa mereka memperhatikan berbagai kebutuhan pengguna. Pilihan antara *Android* dan *iOS* dapat disesuaikan dengan preferensi pengguna terhadap fleksibilitas atau keamanan.

DAFTAR PUSTAKA

- Abdualrahman, A., & Alzahrani, H. (2024). Usability, Reliability, Security, and Status: Unpacking the Factors Shaping User Choice between iOS and Android in Saudi Arabia. *European Journal of Information Technologies and Computer Science*, 4. <https://doi.org/10.24018/ejcompute.2024.4.3.130>
- Fia Anggita Zalsabila, N. (2024). Implementasi Clean Architecture Menggunakan Implementasi Clean Architecture Menggunakan Metode Agile pada Pengembangan Aplikasi Android: Studi Kasus Buang.in KATA KUNCI Agile. *Jurnal Ilmiah Informatika (JFI)*, 12(02), 120–125.

- GeeksforGeeks. (n.d.). *Structures of directory in operating system*. Retrieved November 2, 2024, from <https://www.geeksforgeeks.org/structures-of-directory-in-operating-system/>
- Imam Nivade Rajat, & Sanjay Ajinkya. (2022). Smartphone: IOS Vs Android. *INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT*, 06(06). <https://doi.org/10.55041/ijrem14475>
- Kapetanios, C., Patsakis, C., Alepis, E., & Polyzos, T. (2020). This is just metadata: From no communication content to user profiling, surveillance and exploitation. In *Springer* (pp. 277–302). https://doi.org/10.1007/978-3-030-41196-1_13
- Keur, Christia., & Hillegass, Aaron. (2016). *iOS Programming*.
- Llabres, E. G., & Fernando, E. J. N. (2023). A comparison: Mobile operating system Android vs iOS. *International Journal of Novel Research and Development (IJNRD)*, 8(11), 1–6.
- Rahadian, N. (n.d.). *Identifikasi mengenai sistem operasi Apple (iOS)*. Scribd. Retrieved November 2, 2024, from <https://www.scribd.com/document/405339193/Makalah-Sistem-Operasi-IOS-doc>
- Ruqiya, Islam, N., Rai, A., & Khan, N. (2020). Usability analysis of android and iOS operating systems. *International Journal of Engineering Trends and Technology*, 68(10), 105–111. <https://doi.org/10.14445/22315381/IJETT-V68I10P218>
- SILBERSCHATZ ABRAHAM, BAER GALVIN PETER, & GAGNE GREG. (2012). *OPERATING SYSTEM CONCEPTS NINTH EDITION*.
- Trivedi, S. (n.d.). *Directory structure in OS*. Scaler. Retrieved November 2, 2024, from <https://www.scaler.com/topics/directory-structure-in-os/>
- Umar, A. (2023). A comparative study of modern operating systems in terms of memory and security: A case study of Windows, iOS, and Android. *SLU Journal of Science and Technology*.
- Ye, Roger. (2017). *Android system programming: porting, customizing, and debugging Android HAL*. Packt Publishing.