Journal of Innovative and Creativity
Journal Homepage: https://joecy.org/index.php/joecy
e-mail: journaljoecy@gmail.com

Faculty of Education
University of Pahlawan Tuanku Tambusai
ISSN 2775-771X (Online)
ISSN 2962-570X (Printed)
6 (1) 2026, Page: 2092-2103

Pengembangan Aplikasi Pengelola File Berbasis Android Di Perangkat Mobile

Ali Hasim[1], Jimmie[2], Syarifah Aini[3]
Program Studi Teknologi Informasi, Universitas Muhammadiyah Palembang
Corresponding Author: Ali Hasim, e-mail: alihasyim227@gmail.com
Jimmie, e-mail: jimmie@um-palembang.ac.id
Syarifah Aini, e-mail: Syarifahaini-FT@U-Palembang.ac.id

Published: January, 2026

ABSTRAK

Tujuan penelitian ini adalah mengembangkan aplikasi pengelola file berbasis Android yang mampu membantu pengguna mengatur berkas secara lebih efisien pada penyimpanan lokal perangkat. Latar belakang penelitian ini berangkat dari meningkatnya kebutuhan pengguna terhadap manajemen file yang cepat, mudah, dan dapat digunakan tanpa koneksi internet, mengingat aplikasi bawaan Android belum sepenuhnya memenuhi kebutuhan tersebut. Aplikasi dirancang agar dapat menjalankan berbagai fungsi dasar seperti mengelola, mencari, mengelompokkan, serta mengoptimalkan penyimpanan file di perangkat mobile.

Bahan dan metode. Penelitian menggunakan metode pengembangan *System Development Life Cycle* (SDLC) model *Waterfall* melalui tahap analisis kebutuhan, perancangan, implementasi, hingga pengujian. Perancangan sistem dilakukan dengan diagram dan rancangan antarmuka, sedangkan proses pembuatan aplikasi menggunakan Android Studio. Pengujian dilakukan melalui *blackbox testing* untuk menilai fungsi aplikasi berdasarkan keluaran tanpa melihat proses internal sistem, serta *User Acceptance Testing* (UAT) untuk mengukur penerimaan pengguna

Hasil. Aplikasi yang dikembangkan berhasil menyediakan fitur utama seperti *delete*, *rename*, *copy*, *paste*, *share*, *zip*, *unzip*, pencarian file, pembuatan folder baru, serta melihat file tersembunyi. Seluruh fitur diuji dan berjalan sesuai kebutuhan pengguna. Hasil pengujian juga menunjukkan bahwa aplikasi mampu memberikan pengalaman penggunaan yang lebih sederhana dan efisien dibandingkan aplikasi pengelola file bawaan, serta lebih praktis daripada aplikasi berbasis *cloud* karena tidak memerlukan koneksi internet.

Kesimpulan. Aplikasi pengelola file berbasis Android ini terbukti memenuhi kebutuhan pengguna dalam mengatur berkas secara lokal melalui fitur yang lengkap, antarmuka yang sederhana, serta proses akses yang cepat. Aplikasi ini dapat menjadi solusi alternatif dalam pengelolaan file pada perangkat mobile dan berpotensi dikembangkan lebih lanjut dengan fitur lanjutan sesuai kebutuhan pengguna

Kata kunci: *Pengelola file, Android, aplikasi mobile, SDLC Waterfall, manajemen berkas.*

© The Author(s). 2021 Open Access This article is licensed under a Creative Commons Atibution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0

PENDAHULUAN

Perkembangan teknologi informasi sangatlah pesat, membawa perubahan signifikan dalam berbagai aspek kehidupan, termasuk cara seseorang mengelola dan mengakses informasi. Perangkat mobile, seperti smartphone dan tablet, telah menjadi kebutuhan pokok dalam kehidupan sehari-hari. Sebagai alat praktis untuk mencari informasi dan menyimpan data, perangkat ini sering digunakan untuk menyimpan berbagai jenis berkas seperti dokumen, gambar, video, dan file lainnya. Namun, jumlah berkas yang semakin banyak di perangkat seluler sering kali menimbulkan permasalahan, seperti kesulitan dalam mengatur, mencari, menghapus file, atau bahkan melihat file tersembunyi yang memakan kapasitas penyimpanan perangkat (Clarita et al., 2025).

Sistem operasi Android, sebagai salah satu *platform mobile* terpopuler, menyediakan kemampuan manajemen file bawaan. Namun, fitur tersebut sering kali dianggap kurang memadai oleh pengguna yang membutuhkan pengelolaan berkas yang lebih efisien. Keterbatasan ini

menciptakan peluang untuk mengembangkan aplikasi pengelola file berbasis Android yang mampu menawarkan solusi yang lebih cepat dan mudah digunakan.

Penelitian sebelum telah dilakukan oleh (Chai et al., 2019). Pada penelitian ini dilakukan pengembangan aplikasi pengelolaan file alternatif menggunakan teknologi *cloud computing,* tetapi kemampuan tersebut terbatas di penggunaan internet dan biaya tambahan untuk kapasitas yang lebih besar

Sebagai pengembangan dari penelitian tersebut, aplikasi pengelola file yang dibahas dalam penelitian ini menggunakan pendekatan berbasis penyimpanan mobile. Pendekatan ini tidak bergantung pada jaringan internet, sehingga lebih cocok untuk pengguna dengan keterbatasan akses jaringan atau preferensi terhadap pengelolaan data secara lokal. Aplikasi ini dirancang dengan beberapa fitur unggulan, seperti pengelompokan file berdasarkan jenis, pencarian file, penghapusan file, pengubahan nama file, pembagian file, pemindahan file, pembuatan folder baru, melihat file tersembunyi, serta melakukan kompresi dan ekstraksi file.

Dengan menggunakan Android Studio sebagai alat pengembang, aplikasi ini diharapkan mampu memberikan solusi efektif dalam mengelola berkas pada perangkat mobile, meningkatkan pengalaman pengguna, serta memaksimalkan produktivitas mereka. Selain itu, pengembangan aplikasi ini diharapkan menjadi kontribusi nyata dalam memanfaatkan teknologi untuk mengatasi permasalahan pengelolaan data di perangkat mobile, sekaligus menjadi referensi bagi pengembang aplikasi lain dalam menciptakan inovasi serupa di masa depan (Tri Putra Aditya et al., 2024).

METODE

Metode pengujian dalam penelitian ini menggunakan *blackbox testing*, yaitu pengujian fungsional yang menilai aplikasi berdasarkan hasil eksekusi dari data uji tanpa melihat proses internal. Pengujian dilakukan oleh pengembang dan pengguna dengan mencoba berbagai fitur utama, termasuk pembatasan file, pembuatan laporan bulanan, serta pemberian notifikasi file. Sementara itu, kerangka penelitian disusun untuk menggambarkan hubungan antarvariabel secara visual dan menjadi dasar penyusunan langkah-langkah penelitian.

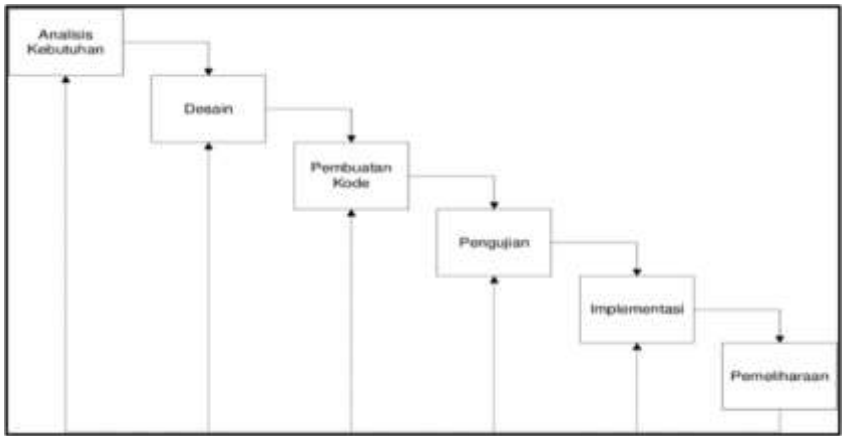
Proses penelitian dimulai dari pengumpulan data mengenai kebutuhan pengguna dalam mengelola file pada perangkat *mobile*, dilanjutkan dengan analisis fitur dan keterbatasan aplikasi bawaan Android. Tahap perancangan mencakup pembuatan UI/UX serta diagram sistem seperti *use case* dan a*ctivity diagram*. Aplikasi kemudian dikembangkan menggunakan Android Studio dengan mengimplementasikan fitur inti, diuji melalui metode *User Acceptance Testing* (UAT), dan akhirnya diimplementasikan pada perangkat Android untuk dievaluasi berdasarkan pengalaman pengguna sebagai dasar pengembangan selanjutnya.

Alat dan Bahan

Alat penelitian yang digunakan berupa laptop ASUS Vivobook M1402IA dengan prosesor AMD Ryzen 7 4800HS, RAM 8 GB, sistem operasi 64-bit, SSD 475 GB, serta perangkat pendukung berupa handphone OPPO Reno 7 dengan Android 13, Snapdragon 695 5G, RAM 8 GB, dan penyimpanan 128 GB. Perangkat lunak yang digunakan meliputi Android Studio, Google Chrome, StarUML, Draw.io, dan Java Development Kit (JDK). Bahan penelitian diperoleh melalui observasi, penelitian terdahulu, jurnal, serta referensi dari aplikasi sejenis yang digunakan untuk mendukung proses pengembangan aplikasi.

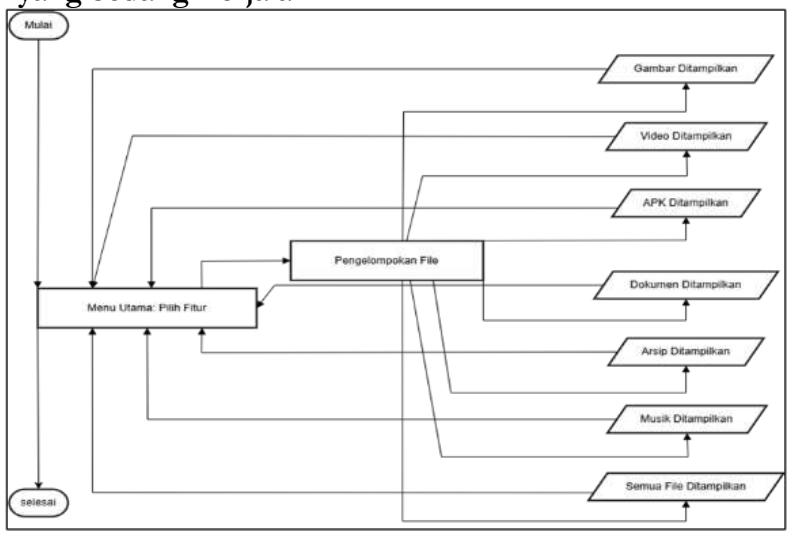
Metode Pengembangan Sistem

System *Development Life Cycle* (SDLC) merupakan proses pengembangan sistem informasi yang mencakup pembuatan, perubahan, dan penyempurnaan sistem melalui tahapan analisis, desain, pembuatan kode, pengujian, implementasi, dan pemeliharaan. Salah satu model SDLC yang paling umum digunakan adalah metode *waterfall* atau *Linear Sequential Model*, yaitu pendekatan pengembangan perangkat lunak yang sistematis, berurutan, dan bersifat linear, di mana setiap tahap harus diselesaikan sebelum melanjutkan ke tahap berikutnya tanpa kembali ke tahap sebelumnya. Model yang diperkenalkan oleh Winston Royce pada tahun 1970 ini tetap banyak digunakan karena memberikan alur kerja yang jelas dan terstruktur dalam proses pengembangan sistem (A. A. Wahid, 2020). Gambar 1 berikut merupakan bagan metode *waterfall*.



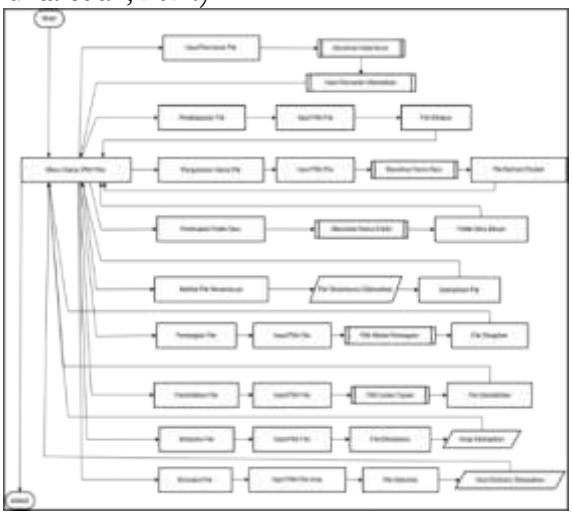
Gambar 1. Metode *Waterfall*

***Flowchart* Sistem yang Sedang Berjalan**



Gambar 2. *Flowchart* Pengelompokan File

Gambar 2 *Flowchart* ini memberikan gambaran yang sangat terperinci tentang langkah-langkah yang dilakukan dalam sistem pengelolaan file. Setiap fitur yang tersedia dirancang untuk membantu pengguna mengelola file dengan lebih efisien dan terorganisir. Mulai dari fitur dasar seperti pengelompokan file berdasarkan jenis, pencarian file dengan kata kunci, dan penghapusan file, hingga fitur yang lebih kompleks seperti pengubahan nama file, pembagian file melalui media tertentu, dan pemindahan file ke lokasi yang diinginkan, semuanya disusun dalam alur yang jelas dan mudah dipahami (Siska Narulita et al., 2024).

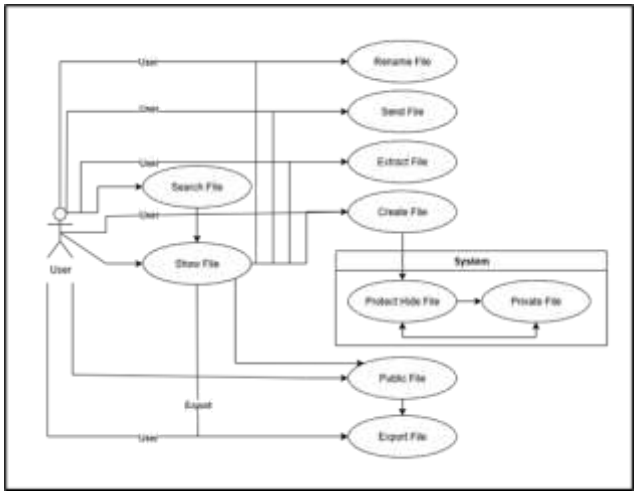


Gambar 3. *Flowchart* Fitur Aplikasi

Menurut Gambar 3, *flowchart* ini juga mencakup fitur tambahan yang sangat berguna, seperti pembuatan folder baru, yang memungkinkan pengguna untuk mengatur file mereka dalam struktur

yang lebih rapi. Ada pula fitur untuk melihat file tersembunyi, yang mempermudah pengguna untuk menemukan file yang tidak terlihat dalam tampilan standar. Fitur kompresi file dan ekstraksi file juga menjadi bagian penting dari sistem ini, memberikan solusi praktis untuk mengurangi ukuran file atau mengakses konten dari file arsip.

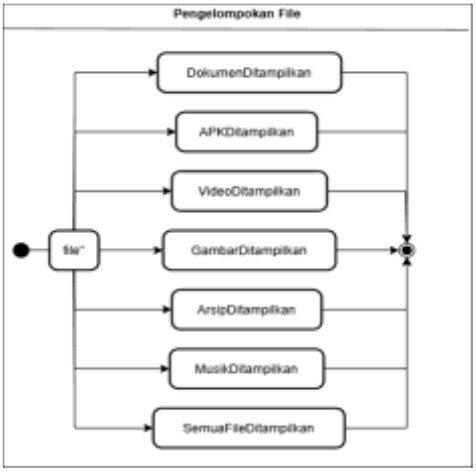
Use Case Diagram



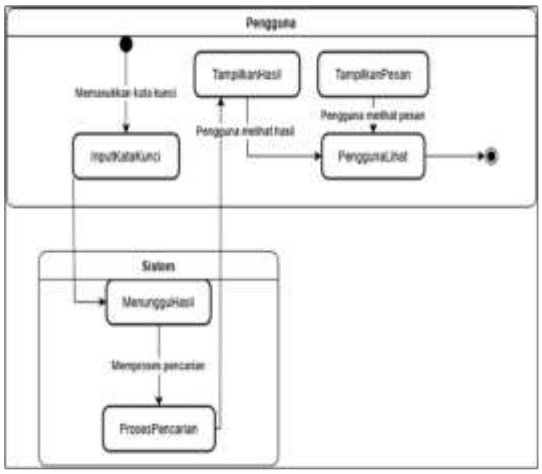
Gambar 4. *Use Case* Diagram Aplikasi

Diagram *use case* pada Gambar 4 menggambarkan fitur-fitur utama yang dapat diakses pengguna dalam sistem pengelolaan file, mulai dari fungsi dasar seperti pencarian, penghapusan, dan pemindahan file hingga fungsi lanjutan seperti kompresi dan ekstraksi. Diagram ini menunjukkan alur interaksi pengguna dengan sistem serta keterkaitan antarfitur yang dirancang untuk memudahkan perpindahan antaraktivitas secara efisien. Selain memberikan gambaran hubungan antara pengguna dan setiap fitur melalui antarmuka yang ramah pengguna, diagram ini juga menjadi panduan bagi pengembang dalam merancang dan mengimplementasikan fungsi-fungsi yang mampu memenuhi kebutuhan pengguna, baik pemula maupun profesional.

Activity Diagram



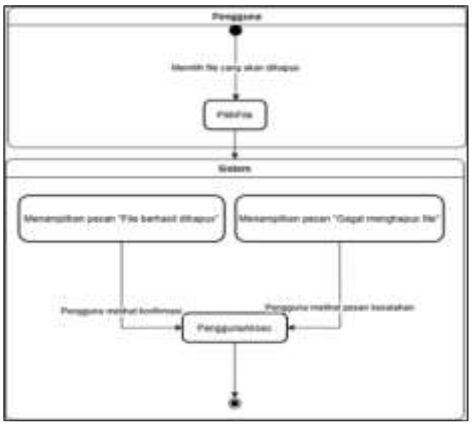
Gambar 5. *Activity Diagram* Pengelompokan File



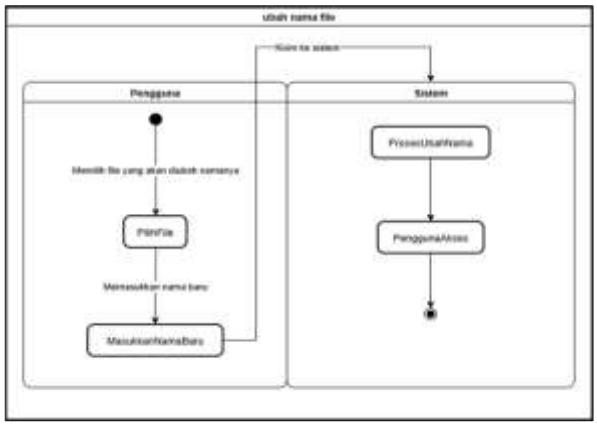
Gambar 6. *Activity Diagram* Pencarian File

Menurut Gambar 5*,* diagram ini menyoroti fitur pengelompokan file yang intuitif dan fleksibel, di mana pengguna dapat dengan mudah melihat file berdasarkan kategori yang relevan. Desain alur memungkinkan pengguna untuk dengan cepat berpindah antara kategori file yang berbeda, sambil memberikan opsi untuk menampilkan semua file jika diperlukan. Fitur ini sangat berguna untuk mengorganisir file dan membantu pengguna menemukan file tertentu dengan lebih cepat.

Gambar 6, diagram ini menunjukkan bahwa proses pencarian file sederhana dan efisien. Dengan hanya memasukkan kata kunci, pengguna dapat langsung mendapatkan hasil pencarian tanpa langkah tambahan. Hal ini menandakan kemudahan penggunaan dan fokus sistem pada penyediaan hasil yang cepat dan akurat



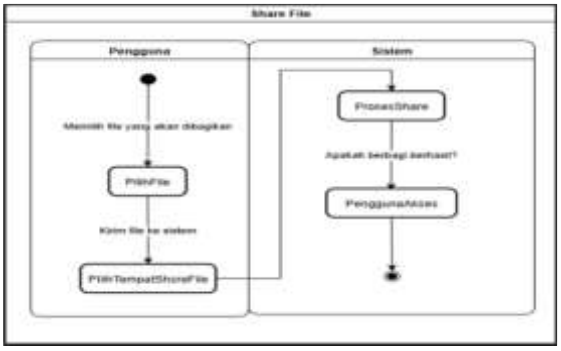
Gambar 7. *Activity Diagram* Penghapusan File



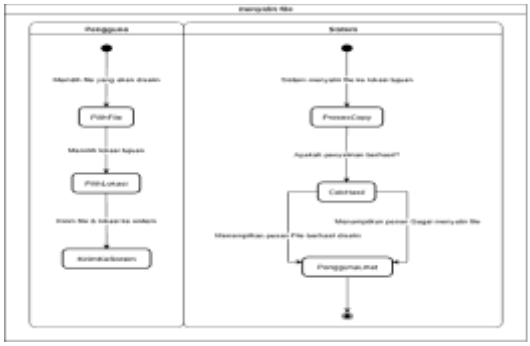
Gambar 8. *Activity Diagram* Ubah Nama File

Menurut Gambar 7, diagram ini menggambarkan alur yang sederhana namun efektif untuk proses penghapusan file, dengan menambahkan langkah konfirmasi untuk mencegah kesalahan pengguna. Proses ini mencerminkan prinsip-prinsip dasar dalam desain sistem yang user-friendly dan aman, dengan fokus pada pencegahan kesalahan dan pengambilan keputusan yang disengaja oleh pengguna

Menurut Gambar 8, proses ini mencerminkan pendekatan *user-friendly* dalam desain sistem, yang memberikan kontrol kepada pengguna dengan tetap mempertahankan keandalan dan keamanan dalam mengelola file. Diagram ini relevan untuk berbagai aplikasi, baik di lingkungan pribadi maupun profesional, di mana pengorganisasian file menjadi hal yang krusial.



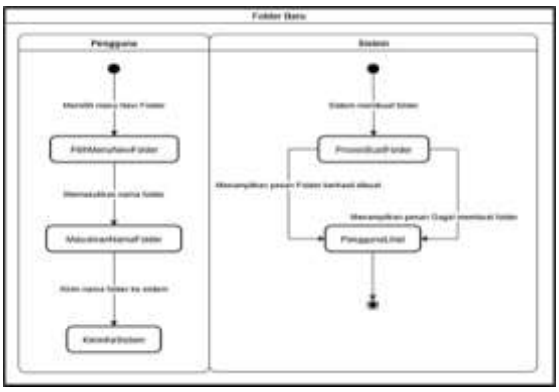
Gambar 9. *Activity Diagram* Membagikan File



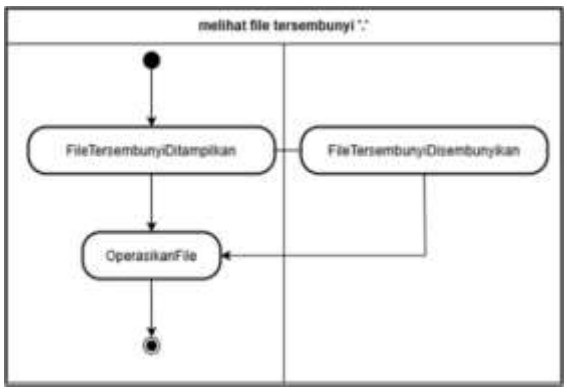
Gambar 10. *Activity Diagram* Menyalin dan Menempel File

Menurut Gambar 9, proses ini dirancang untuk memastikan efisiensi, kemudahan, dan kejelasan bagi pengguna dalam menjalankan tindakan berbagi file. Setiap langkah disusun secara terstruktur untuk meminimalkan kebingungan dan memberikan panduan yang jelas dalam memilih file, menentukan media pembagian, dan menyelesaikan proses berbagi. Fokus utama terletak pada menciptakan pengalaman pengguna (*user experience*) yang intuitif, di mana pengguna dapat dengan mudah memahami alur kerja tanpa memerlukan panduan tambahan.

Menurut Gambar 10, diagram ini relevan dalam menggambarkan fungsi dasar yang sering digunakan dalam sistem operasi atau aplikasi pengelolaan file. Proses ini tidak hanya meningkatkan efisiensi kerja, tetapi juga memastikan keandalan dengan memberikan konfirmasi bahwa file berhasil ditempel di lokasi tujuan. Dengan desain yang sederhana namun efektif, diagram ini menjadi representasi penting untuk memahami bagaimana sistem mendukung pengelolaan file secara digital dalam berbagai konteks



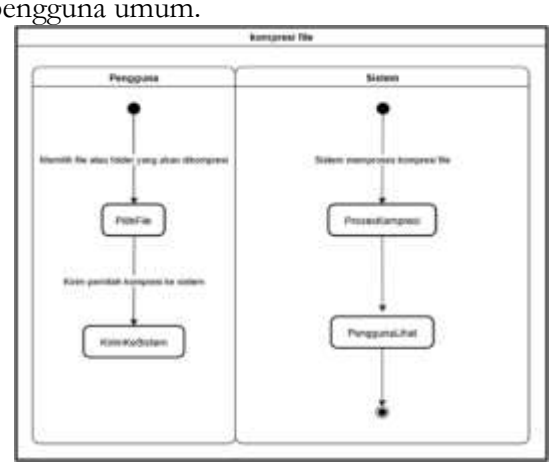
Gambar 11. *Activity Diagram* New Folder



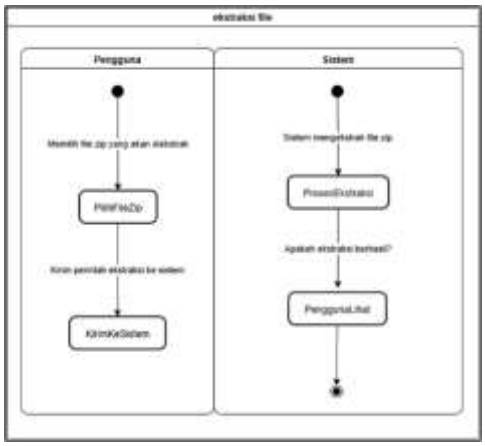
Gambar 12. *Activity Diagram* File Tersembunyi

Menurut Gambar 11, proses ini sangat penting dalam pengelolaan file dan folder, terutama untuk membantu pengguna menciptakan struktur penyimpanan yang lebih rapi dan terorganisir. Dengan fitur membuat folder baru, pengguna dapat dengan mudah mengelompokkan file berdasarkan kategori, proyek, atau kebutuhan spesifik lainnya, baik untuk keperluan pribadi seperti menyimpan foto dan dokumen

Menurut Gambar 12 , proses ini menjadi bagian penting dari aplikasi pengelolaan file atau sistem operasi karena memberikan fleksibilitas kepada pengguna untuk mengakses file tersembunyi yang biasanya tidak terlihat dalam kondisi normal. File tersembunyi sering kali digunakan untuk menyimpan konfigurasi, pengaturan sistem, atau data penting lainnya yang tidak perlu diakses oleh pengguna umum.



Gambar 13. *Activity Diagram* Kompresi File



Gambar 14. *Activity Diagram* Ekstraksi File

Menurut Gambar 13, *activity diagram* di atas menggambarkan proses kompresi file, yaitu langkah penting dalam pengelolaan file yang bertujuan untuk mengurangi ukuran data atau menggabungkan beberapa file menjadi satu arsip yang lebih mudah dikelola. Proses ini tidak hanya relevan dalam pengelolaan file sehari-hari, tetapi juga memiliki aplikasi luas dalam transfer data, penyimpanan *cloud*, dan pengarsipan file jangka panjang

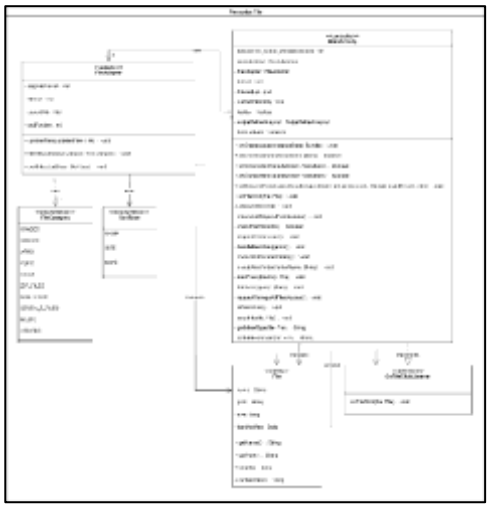
Menurut Gambar 14, *activity diagram* di atas menggambarkan proses ekstraksi file, yaitu langkah untuk membuka atau menguraikan file yang telah dikompresi sebelumnya ke dalam bentuk aslinya. Proses ini bertujuan untuk memungkinkan pengguna mengakses konten yang ada di dalam arsip dengan cara yang mudah dan efisien.

Class Diagram

Class diagram adalah salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan struktur statis sistem. Diagram ini memodelkan kelas-kelas yang ada dalam sistem, beserta atribut, operasi (metode), dan hubungan antar kelas.



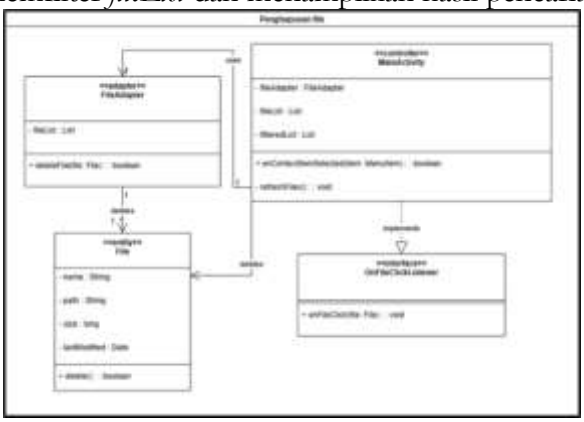
Gambar 15. Pengelompokan File



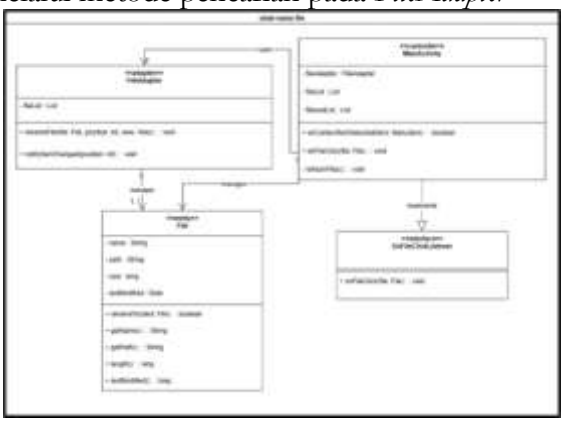
Gambar 16. Pencarian File

Gambar 15 merupakan *Class Diagram* Pengelompokan File yang menggambarkan bagaimana sistem mengelola berbagai kategori file seperti gambar, video, musik, dokumen, arsip, dan lainnya. Kelas *MainActivity* berperan sebagai pengendali utama yang berinteraksi dengan *FileAdapter* untuk menampilkan file ke pengguna berdasarkan kategori yang dipilih

Gambar 16 merupakan *Class Diagram* Pencarian File yang menunjukkan alur pencarian file berdasarkan kata kunci. Dalam diagram ini, *MainActivity* bekerja sama dengan *FileAdapter* untuk memfilter *fileList* dan menampilkan hasil pencarian melalui metode pencarian pada *FileAdapter*



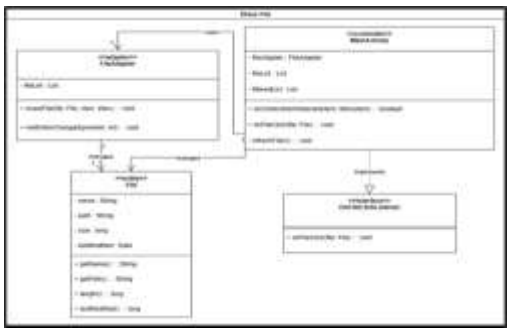
Gambar 17. Penghapusan File



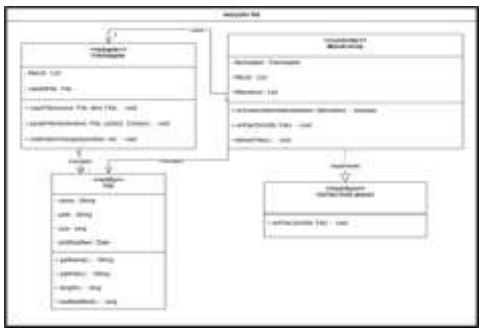
Gambar 18. Ubah Nama File

Gambar 17 merupakan *Class Diagram* Penghapusan File yang menunjukkan proses penghapusan file yang dilakukan melalui *MainActivity* dengan memanggil metode pada *FileAdapter* untuk menghapus file dari penyimpanan dan memperbarui tampilan daftar file.

Gambar 18 merupakan *Class Diagram* Ubah Nama File. Diagram ini menunjukkan bahwa FileAdapter memiliki metode *renameFile()* yang bertanggung jawab untuk mengganti nama file. Proses rename diawali dari aksi pada *MainActivity* dan diteruskan ke File Adapter.



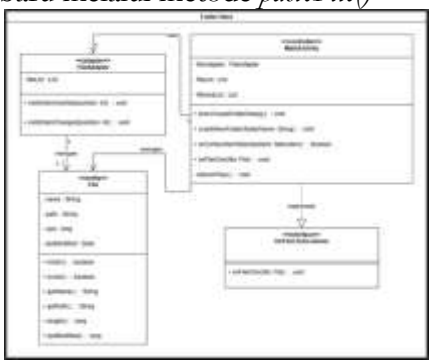
Gambar 19. Membagikan File



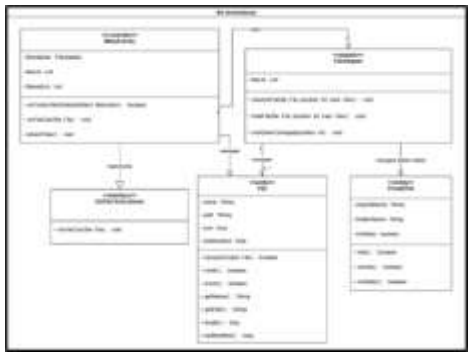
Gambar 20. Menyalin dan Menempel File

Gambar 19 merupakan *Class Diagram* Membagikan File yang menggambarkan fitur berbagi file melalui intent Android. Diagram menunjukkan bahwa *FileAdapter* memiliki metode *shareFile()* untuk membagikan file kepada aplikasi lain, dengan *MainActivity* sebagai pengendali utama.

Gambar 20 merupakan *Class Diagram* Menyalin Dan Menempel File. Dalam diagram ini terdapat atribut tambahan *copiedFile* pada *FileAdapter* yang menyimpan file sementara sebelum ditempel ke lokasi baru melalui metode *pasteFile()*



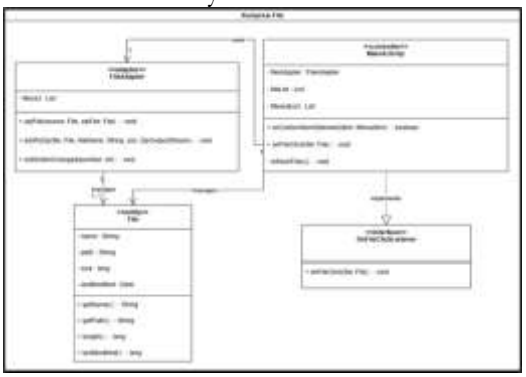
Gambar 21. Foldel Baru



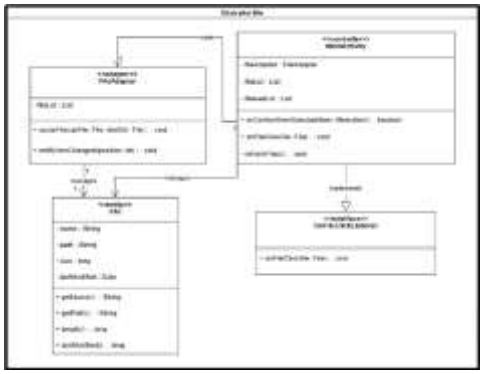
Gambar 22. File Tersembunyi

Gambar 21 merupakan *Class Diagram* Pembuatan Folder Baru. Diagram ini menunjukkan bahwa *MainActivity* memiliki metode *showCreateFolderDialog()* untuk menampilkan dialog input nama folder dan metode *createNewFolder()* yang membuat folder baru dengan memanfaatkan metode mkdir() pada File

Gambar 22 merupakan *Class Diagram* File Tersembunyi. Diagram ini memperkenalkan kelas tambahan *PrivateFile* untuk mengelola status file yang disembunyikan dengan cara mengganti nama file menjadi awalan titik. Kelas ini memiliki metode *hide(), unhide(), dan isHidden()* untuk mengelola status file tersembunyi.



Gambar 23. Kompresi File



Gambar 24. Ekstraksi File

Gambar 23 merupakan *Class Diagram* Kompresi File. Dalam diagram ini, File Adapter dilengkapi metode *zipFile()* untuk membuat file zip dan metode *addToZip()* untuk menambahkan file ke dalam arsip zip. Proses ini dipicu dari aksi pengguna melalui *MainActivity.*

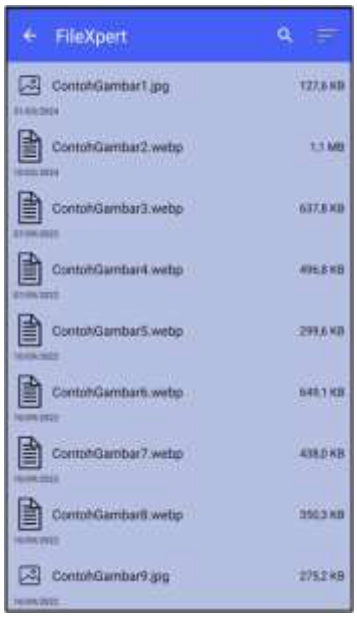
Gambar 24 merupakan *Class Diagram* Kompresi File. Dalam diagram ini, File *Adapter* dilengkapi metode *zipFile()* untuk membuat file zip dan metode *addToZip()* untuk menambahkan file ke dalam arsip zip. Proses ini dipicu dari aksi pengguna melalui *MainActivity.*

HASIL DAN PEMBAHASAN

Desain *Interface*



Gambar 25. Halaman Utama



Gambar 26. Halaman Gambar

Menurut gambar 25, halaman ini memiliki 7 menu yaitu halaman gambar, video, APK, PDF/DOCX, ZIP/RAR/7Z, musik, dan Semua file.

Menurut gambar 26, halaman ini berfungsi untuk menampilkan file gambar-gambar yang ada dalam penyimpanan android.



Gambar 27. Halaman Video



Gambar 28. Halaman Apk

Menurut gambar 27 halaman ini menampilkan file video yang berada di penyimpanan android.

Menurut gambar 28, halaman ini menampilkan file aplikasi yang belum terinstal yang berada dalam penyimpanan android.



Gambar 29. Halaman PDF dan DOCX



Gambar 30. Halaman ZIP, RAR, dan 7Z

Menurut gambar 29 halaman ini menampilkan berkas-berkas dokumen PDF dan DOCX yang ada dalam penyimpanan android.

Menurut gambar 30 halaman ini menampilkan file-file ZIP, RAR, dan 7Z yang ada dalam penyimpanan android.



Gambar 31. Halaman Musik



Gambar 32. Halaman Semua File

Menurut gambar 31 halaman ini menampilkan musik-musik yang ada dalam penyimpanan file android.

Menurut gambar 32, halaman ini mencakup semua file yang ada dalam penyimpanan file seperti melihat dokumen, gambar, file yang di download dan sebagainya.



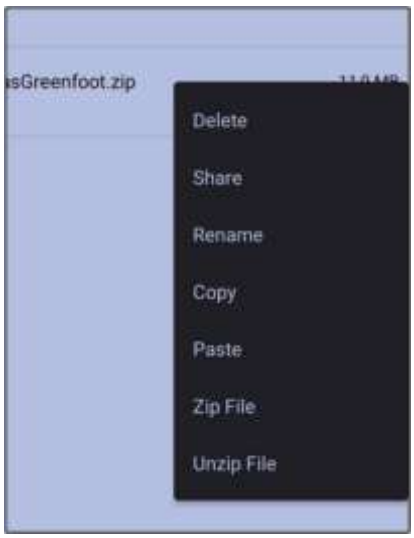
Gambar 33. Fitur Menu *Sort*, Folder Baru, Sembunyikan File



Gambar 34. Fitur Pencarian

Menurut gambar 33, pada fitur menu *sort* kita sebagai pengguna dapat mengurutkan file menurut nama, tanggal, dan ukuran. Kita dapat membuat folder baru di aplikasi ini. Kita dapat melihat file tersembunyi dalam penyimpanan kita.

Menurut gambar 34, fitur ini digunakan untuk mempermudah pencarian file atau folder dalam penyimpanan android.



Gambar 35. Fitur Utama

Menurut gambar 35, terdapat beberapa fitur utama dalam pengelolaan file ini yaitu delete, share, rename, copy, paste, zip file, dan unzip file. Fitur-fitur pada sistem memiliki fungsi yang berbeda untuk memastikan aplikasi berjalan dengan baik. Fitur *delete* digunakan untuk menghapus file atau folder dalam penyimpanan Android, sedangkan *share* memungkinkan pengguna membagikan file ke aplikasi lain seperti WhatsApp, Gmail, atau Messenger. Fitur *rename* berfungsi mengubah nama file atau folder, dan *copy* digunakan untuk menyalin file ke lokasi lain yang kemudian dapat ditempel menggunakan fitur *paste*. Selain itu, tersedia fitur *zip file* yang digunakan untuk mengompresi file menjadi format ZIP, serta *unzip file* yang berfungsi mengekstraksi kembali file ZIP tersebut.

HASIL PENELITIAN

Hasil penelitian menunjukkan bahwa aplikasi pengelola file berbasis Android ini mampu memberikan solusi yang lebih efisien dibandingkan dengan fitur bawaan Android. Aplikasi ini dirancang menggunakan metode *Waterfall* dan dikembangkan dengan Android Studio, serta dilengkapi fitur utama seperti pengelompokan file, pencarian file, penghapusan, kompresi, ekstraksi, dan pengelolaan file tersembunyi. Dibandingkan dengan solusi berbasis *cloud*, aplikasi ini lebih ramah bagi pengguna dengan keterbatasan akses internet karena mengandalkan penyimpanan lokal. Selain itu, desain antarmuka yang intuitif dan fitur-fitur lengkapnya meningkatkan pengalaman pengguna secara signifikan. Dalam implementasinya, pengguna dapat dengan mudah mengelola file berdasarkan kategori seperti gambar, video, dokumen, dan lainnya. Fitur tambahan, seperti kompresi dan ekstraksi file, juga memberikan nilai tambah yang penting, menjadikan aplikasi ini sebagai solusi efektif untuk pengelolaan berkas pada perangkat mobile.

KESIMPULAN

Perkembangan teknologi informasi, khususnya pada perangkat mobile, menghadirkan kebutuhan akan pengelolaan file yang lebih efisien. Sistem operasi Android menyediakan fitur manajemen file bawaan, tetapi sering kali dianggap kurang memadai bagi pengguna dengan kebutuhan pengelolaan yang kompleks. Penelitian ini mengembangkan aplikasi pengelola file berbasis Android dengan fitur unggulan seperti pengelompokan file berdasarkan jenis, pencarian, penghapusan, kompresi, ekstraksi, dan pengelolaan file tersembunyi. Berbeda dengan solusi berbasis *cloud*, aplikasi ini menggunakan pendekatan penyimpanan mobile lokal, yang lebih sesuai bagi pengguna dengan keterbatasan akses internet

REFERENSI

A. A. Wahid. (2020). "Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi," . *Jurnal Ilmu-Ilmu Informatika Dan Manajemen STMIK*, *1*(October).

Chai, D., Gajah, U., Simarmata, E. M., & Laia, Y. (2019). Aplikasi Penyimpanan File Alternatif Bagi Pengguna Smartphone Berbasis Android. *Jurnal Sistem Informasi Dan Ilmu Komputer Prima(JUSIKOM PRIMA)*, *3*(1), 35–42. https://doi.org/10.34012/jusikom.v3i1.555

Clarita, A. R. N., Faizah, V. F., Zahra, N., & Elkin, R. (2025). Komparasi Struktur Direktori Pada Sistem Operasi Mobile Android Dan Ios. *Scientica*, *3*, 86.

Siska Narulita, Ahmad Nugroho, & M. Zakki Abdillah. (2024). Diagram Unified Modelling Language (UML) untuk Perancangan Sistem Informasi Manajemen Penelitian dan Pengabdian Masyarakat (SIMLITABMAS). *Bridge : Jurnal Publikasi Sistem Informasi Dan Telekomunikasi*, *2*(3), 244–256. https://doi.org/10.62951/bridge.v2i3.174

Tri Putra Aditya, Aisyah Aisyah, & Rakhmadi Rahman. (2024). Strategi Manajemen Memori pada Sistem Operasi Android untuk Smartphone Kelas Menengah. *Router : Jurnal Teknik Informatika Dan Terapan*, *2*(3), 165–172. https://doi.org/10.62951/router.v2i3.165