BAB I
PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi informasi dan aktivitas digital yang pesat memicu peningkatan volume data secara signifikan. Kondisi ini menuntut adanya solusi *cloud storage* yang efisien, fleksibel, dan berkapasitas besar. Saat ini, layanan *public cloud storage* komersial seperti Google Drive, Dropbox, dan OneDrive menawarkan kemudahan akses layaknya *virtual file manager*. Namun, layanan tersebut memiliki batasan kapasitas gratis yang ketat umumnya 5 sampai 15 GB dan mengenakan biaya langganan yang mahal untuk kebutuhan penyimpanan berskala besar. Menurut (Waranggani, 2023), *International Data Corporation* (IDC), sebuah perusahaan riset pasar berskala global, memperkirakan bahwa nilai pasar sektor penyimpanan *cloud virtual* di Indonesia telah mencapai 933,63 juta dolar AS pada tahun 2023. Angka ini menunjukkan kenaikan sebesar 25% dibandingkan dengan 747,15 juta dolar AS pada tahun 2022. Di sisi lain, membangun infrastruktur *server* penyimpanan secara mandiri juga bukan solusi ideal bagi pengguna berskala kecil karena membutuhkan investasi perangkat keras dan biaya pemeliharaan yang tinggi.

Guna mengatasi permasalahan tersebut secara komprehensif, penelitian ini mengusulkan perancangan sistem *Private Cloud storage* yang mengintegrasikan antarmuka *Web UI*, protokol MTProto, dan algoritma kriptografi AES-256. Penelitian (Tandy et al., 2020) mengemukakan bahwa teknologi *cloud* sangat bermanfaat bagi pengguna karena menghilangkan kebutuhan untuk menginvestasikan dana besar dalam perangkat lunak dan aplikasi data, serta pemeliharaan perangkat. *Web UI* dirancang sebagai *virtual file manager* agar pengguna dapat mengelola berkas secara intuitif tanpa berinteraksi dengan antarmuka *chat* Telegram. Menurut (*Telegram Users by Country*, 2026), jumlah unduhan aplikasi Telegram pada tahun 2023 mencapai 257,8 juta unduhan di

beberapa negara utama, dengan India menempati posisi pertama sebanyak 83,8 juta unduhan, disusul Rusia 35,1 juta, Amerika Serikat 29,9 juta, dan Indonesia 24,3 juta unduhan.

Kendala batasan ukuran *file* sebesar 50 MB diatasi menggunakan *library* Pyrogram, yaitu sebuah *library* Python yang berkomunikasi langsung dengan *data center* Telegram melalui protokol biner *native* MTProto. Pendekatan ini secara drastis meningkatkan batas unggahan menjadi 2 GB per berkas secara efisien, tanpa memerlukan infrastruktur *server* penyimpanan tambahan, karena seluruh beban penyimpanan data dialihkan sepenuhnya ke pusat data Telegram. Sementara itu, untuk berkas yang ukurannya melampaui batas operasional sistem yang di atur 1,8 GB untuk memberikan margin enkripsi, sistem secara otomatis menerapkan teknik *file chunking* di sisi *backend* untuk memecah dan menyatukan berkas tanpa campur tangan pengguna. Sebagai catatan arsitektural, sistem ini berinteraksi dengan infrastruktur Telegram menggunakan identitas Akun Bot. Namun, penelitian ini secara tegas tidak menggunakan antarmuka *Telegram HTTP Bot API* konvensional yang memiliki batasan ukuran unggah maksimal 50 MB, melainkan bertindak sebagai *MTProto Client* menggunakan pustaka Pyrogram. Pendekatan teknis ini memungkinkan sistem untuk membangun *direct socket connection* ke pusat data Telegram guna memanfaatkan infrastruktur penyimpanan dengan batasan unggah yang jauh lebih besar.

Pada aspek privasi dan integritas data, sistem menerapkan algoritma kriptografi AES-256 dan fungsi *hash SHA-256*. Penelitian (Surya et al., 2025) menunjukkan bahwa AES-256 memberikan waktu enkripsi atau dekripsi yang lebih efisien dibandingkan RSA dan ECC pada *file* berukuran besar. Setiap berkas atau segmen *chunk* dienkripsi secara lokal di sisi *backend* menggunakan algoritma AES-256-CBC, sementara aliran transmisi datanya menuju infrastruktur Telegram diakselerasi kinerjanya oleh *library* TgCrypto khusus pada lapisan protokol jaringan MTProto. Sebelum dikirimkan ke Telegram. Dengan demikian, data yang tersimpan di infrastruktur platform selalu dalam kondisi terenkripsi. Basis data lokal berbasis SQLite hanya difungsikan untuk menyimpan *metadata* berkas dan

parameter enkripsi, sehingga jejak data sensitif di sisi *server* aplikasi dapat diminimalkan. AES-256 dipilih karena merupakan bagian dari AES yang ditetapkan sebagai standar kriptografi oleh NIST untuk melindungi data elektronik, dan diakui secara luas sebagai algoritma enkripsi simetris yang paling kokoh terhadap ancaman kriptanalisis saat ini (NIST et al., 2001).

Melalui pendekatan pengembangan perangkat lunak metode *Prototyping* yang iteratif, sistem yang dihasilkan diharapkan mampu menghadirkan solusi penyimpanan *file* berkapasitas besar dan berbiaya infrastruktur rendah. Penelitian yang berjudul "Penerapan Metode Prototyping Pada Perancangan Private *Cloud storage* Berbasis Telegram Bot Melalui Protokol MTProto Dengan Algoritma Kriptografi AES-256" ini bertujuan menciptakan sistem yang mudah dikelola sekaligus memberikan jaminan keamanan privasi data yang kokoh di setiap lapisan penyimpanan.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, terdapat beberapa persoalan teknis yang harus diselesaikan. Oleh karena itu, rumusan masalah dalam penelitian ini dirumuskan melalui pertanyaan berikut:

1. Bagaimana merancang arsitektur sistem *Private Cloud storage* yang mengintegrasikan *Web UI* sebagai *virtual file manager* dengan protokol MTProto melalui *library* Pyrogram sebagai mekanisme komunikasi utama ke *backend* penyimpanan Telegram menggunakan metode Prototyping?

2. Bagaimana menerapkan algoritma AES-256 untuk enkripsi dan fungsi *hash SHA-256* untuk verifikasi integritas berkas secara lokal di sisi *backend* sebelum data diunggah ke infrastruktur Telegram guna meningkatkan keamanan data pengguna?

3. Bagaimana mengatasi batasan ukuran unggah maksimal pada *server* Telegram dengan mengimplementasikan teknik *file chunking* melalui protokol MTProto?

1.3. Tujuan Penelitian

Sebagai upaya untuk menjawab berbagai tantangan yang telah dirumuskan pada bagian sebelumnya, perancangan dan pengembangan sistem dalam penelitian ini memiliki beberapa tujuan spesifik, yaitu:

1. Merancang dan membangun arsitektur sistem Private *Cloud storage* yang mengintegrasikan *Web UI* sebagai *virtual file manager* dengan protokol MTProto melalui *library* Pyrogram sebagai mekanisme komunikasi utama ke *backend* penyimpanan Telegram menggunakan metode Prototyping sebagai pendekatan pengembangan yang iteratif.
2. Menerapkan algoritma AES-256 pada mekanisme enkripsi dan dekripsi berkas serta fungsi *hash SHA-256* untuk verifikasi integritas file secara lokal di sisi *backend* sebelum dan sesudah data dikirimkan ke infrastruktur Telegram.
3. Mengoptimalkan kemampuan sistem dalam menangani berkas berukuran besar secara efisien melalui pemanfaatan protokol MTProto via *library* Pyrogram dan penerapan teknik *file chunking* otomatis untuk berkas yang melampaui batas yang telah ditentukan, sehingga sistem mampu mengelola berkas dengan ukuran besar jauh melampaui batas layanan *cloud storage* konvensional.

1.4. Batasan Masalah

Agar arah dan fokus penelitian tetap terpusat pada pemecahan masalah utama, serta untuk menjaga ruang lingkup pengembangan sistem agar tetap terukur dan terarah secara teknis, maka batasan masalah dalam penelitian ini ditetapkan sebagai berikut:

1. Antarmuka yang disediakan kepada pengguna akhir dibangun sepenuhnya berbasis *Web UI* dan dirancang menyerupai antarmuka *virtual file manager*, tidak dikembangkan aplikasi *mobile native* atau antarmuka berbasis desktop yang terpisah.
2. Media penyimpanan fisik memanfaatkan sepenuhnya infrastruktur *server* Telegram melalui *library* Pyrogram yang berkomunikasi langsung dengan

server Telegram via protokol MTProto, bukan melalui Public Telegram Bot API dengan *endpoint* HTTP yang memiliki batas unggahan 50 MB per berkas.

3. Aspek keamanan data difokuskan pada proses enkripsi dan dekripsi berkas menggunakan AES-256 serta verifikasi integritas *file* menggunakan SHA-256 pada lapisan *backend*. Keamanan jaringan dan aspek di luar *application layer* tidak dibahas secara mendalam.
4. Teknik *file chunking* otomatis diterapkan pada berkas yang melampaui ukuran 1,8 GB dengan memecahnya menjadi beberapa *chunks* berukuran maksimal 1,8 GB. Pemotongan pada angka tersebut berfungsi sebagai batas aman guna mengakomodasi *overhead* enkripsi AES dan *metadata*, sehingga berkas tidak membentur batas absolut 2 GB dari protokol MTProto. Berkas dengan ukuran di bawah 1,8 GB akan diunggah secara utuh.
5. Basis data lokal hanya digunakan untuk menyimpan *metadata* berkas, meliputi nama *file*, ukuran, jumlah dan urutan *chunk*, serta parameter kriptografi yang diperlukan untuk proses dekripsi, konten berkas tidak disimpan di basis data lokal.
6. Sistem ini dirancang dengan arsitektur Multi-User *terdesentralisasi*, di mana setiap pengguna diwajibkan memiliki dan menginput kredensial API Telegram serta Chat ID pribadi milik mereka sendiri. Sistem tidak menyediakan infrastruktur API pusat untuk mencegah kebocoran privasi antar pengguna.

1.5 Manfaat Penelitian

Penelitian ini diharapkan memberikan manfaat yang signifikan bagi dua kelompok utama, yaitu:

1.5.1. Manfaat Teoritis

a. Memberikan kontribusi ilmiah pada bidang Teknik Informatika dalam sub-ranah keamanan data khususnya dalam kriptografi AES-256, arsitektur sistem penyimpanan terdistribusi berbasis Bot API, serta rekayasa perangkat lunak dengan metode Prototyping.
b. Memperkaya referensi akademis mengenai arsitektur pemanfaatan infrastruktur *platform chatting* khususnya melalui protokol MTProto via *library*

Pyrogram sebagai *backend* penyimpanan alternatif berbiaya rendah dengan kapasitas besar.

c. Menyediakan kajian teknis mengenai penggabungan dua mekanisme penanganan file besar komunikasi langsung via Pyrogram MTProto yang mendukung unggahan hingga 2 GB per berkas dan teknik *file chunking* dalam satu arsitektur sistem yang kohesif dan terintegrasi.
d. Menjadi referensi bagi penelitian lanjutan yang dapat mengembangkan sistem serupa, misalnya dengan menambahkan fitur sinkronisasi *real-time*, berbagi berkas terenkripsi antar pengguna, atau penggunaan algoritma enkripsi asimetris untuk manajemen kunci.

1.5.2. Manfaat Praktis

a. Menyediakan solusi penyimpanan *file* berkapasitas masif dan berbiaya infrastruktur minimal dengan mengandalkan *server* Telegram sebagai medium penyimpanan, sehingga pengguna terbebas dari kebutuhan investasi perangkat keras penyimpanan berskala besar.
b. Memberikan antarmuka pengelolaan *file* yang intuitif dan terstruktur melalui *Web UI* berbasis *virtual file manager*, sehingga kompleksitas teknis interaksi dengan Telegram Bot API tersembunyi sepenuhnya dari pengguna akhir.
c. Menjamin privasi dan keamanan data pengguna secara signifikan melalui enkripsi AES-256 menyeluruh setiap berkas dan setiap *chunk* dienkripsi sebelum meninggalkan *server* aplikasi sehingga meminimalkan risiko akses tidak sah.
d. Menghilangkan hambatan teknis batasan ukuran file secara bertingkat, Pyrogram MTProto menangani berkas hingga 2 GB per berkas tanpa membebani kapasitas penyimpanan *server* lokal**,** sementara *file chunking* otomatis menangani berkas di atas batas tersebut sehingga pengguna dapat menyimpan berkas berukuran besar jauh melampaui layanan *cloud storage* komersial pada umumnya.

1.6. Jadwal Penelitian

Penelitian ini direncanakan berlangsung selama 6 bulan, mulai Februari 2026 hingga Juli 2026. Jadwal pelaksanaan setiap kegiatan penelitian disajikan pada tabel berikut.

Tabel 1. 1 Jadwal Kegiatan Penelitian.

No	Kegiatan Penelitian	Bulan Februari	Bulan Maret	Bulan April	Bulan Mei	Bulan Juni	Bulan Juli
1	Penyusunan Proposal & Studi Literatur	■	■				
2	Analisis Kebutuhan Sistem	■	■				
3	Perancangan Sistem *Prototype*		■	■			
4	Implementasi *Backend* & Pyrogram MTProto			■	■		
5	Implementasi Enkripsi AES-256 & SHA-256				■	■	
6	Implementasi *File chunking*				■	■	
7	Pengujian & Evaluasi *Prototype*				■	■	■
8	Revisi & Penyempurnaan Sistem				■	■	■
9	Penyusunan Laporan Akhir					■	■

Bagian yang di arsir menunjukkan kegiatan yang dilaksanakan pada bulan yang bersangkutan. Jadwal bersifat tentatif dan dapat disesuaikan dengan kebutuhan aktual selama proses penelitian berlangsung, sepanjang tidak melampaui batas waktu penelitian yang telah ditetapkan.

1.7 Sistematika Penulisan

Laporan tugas akhir ini disusun secara sistematis ke dalam lima bab yang saling berkesinambungan, dengan rincian sebagai berikut:

BAB I PENDAHULUAN

Menguraikan latar belakang masalah, alasan pemilihan judul, rumusan dan batasan masalah, tujuan dan manfaat penelitian, serta jadwal dan sistematika penulisan Tugas Akhir.

BAB II TINJAUAN PUSTAKA

Membahas kajian pustaka dan konsep-konsep teoritis yang menjadi fondasi utama dalam memecahkan masalah dan mendukung jalannya penelitian.

BAB III METODOLOGI PENELITIAN

Menjelaskan secara rinci kerangka berpikir, metode, dan prosedur penelitian yang diterapkan, beserta tahapan perancangan sistem yang akan dibangun.

BAB IV HASIL DAN PEMBAHASAN

Memaparkan hasil akhir dari penelitian yang telah dilakukan, khususnya yang berkaitan dengan implementasi dan proses pembuatan aplikasi.

BAB V KESIMPULAN DAN SARAN

Berisi kesimpulan akhir yang ditarik dari keseluruhan pembahasan penelitian, serta saran dan rekomendasi yang berguna untuk pengembangan aplikasi ke depannya.