

Perancangan Program Pengiriman Berkas dengan Implementasi Java Socket

Fajar Rahmat¹, Rahman Hidayah², Ranto Anjasmara Marpaung³,
Salman Al Haritsi⁴, Linna Oktaviana Sari⁵

^{1,2,3,4,5}Jurusan Teknik Elektro Program Studi Teknik Informatika Universitas Riau

e-mail: fajar.rahmat2738@student.unri.ac.id

Abstrak

Pertukaran berkas melalui jaringan telah menjadi bagian penting dari kehidupan sehari-hari. Untuk memenuhi kebutuhan ini, penelitian ini merancang serta mengimplementasikan program pengiriman berkas berbasis Java Socket. Dengan berfokus pada efisiensi dan keamanan, penelitian ini menggunakan pendekatan pengembangan waterfall yang telah dimodifikasi untuk mencakup analisis kebutuhan sistem, perancangan aplikasi, dan uji coba aplikasi. Arsitektur aplikasi memungkinkan pengguna saling terhubung melalui server, dengan username sebagai pengenal. Dengan IDE Apache Netbeans, program ini menggunakan Java Socket untuk implementasi skema client-server. Penulis menggunakan metode pengujian blackbox untuk uji coba aplikasi, menghasilkan hasil positif pada skenario seperti pengaktifan server, koneksi pengguna, pengiriman dan pengunduhan berkas, serta manajemen koneksi oleh admin. Hasil dari penelitian ini menunjukkan bahwa aplikasi ini dapat menjadi solusi efisien untuk pertukaran berkas pada lingkungan jaringan internal, dengan keamanan yang ditingkatkan. Meskipun demikian, masih diperlukan pengelolaan jaringan yang cermat untuk memastikan akses terbatas pada pengguna internal. Saran untuk pengembangan lebih lanjut mencakup otomatisasi pembuatan folder saat pengguna terhubung ke server. Dengan demikian aplikasi ini dapat memberikan kontribusi dalam memenuhi kebutuhan pertukaran berkas dengan cara yang efisien dan aman di lingkungan jaringan tertentu.

Kata kunci: *Pertukaran berkas, jaringan, Java Socket.*

Abstract

File exchange over networks has become an important part of everyday life. To meet this need, this study designed and implemented a Java Socket-based file delivery program. With a focus on efficiency and security, this research uses a modified waterfall development approach to cover system requirements analysis, application design, and application testing. The application architecture allows users to connect to each other through the server, with a username as an identifier. Using the Apache Netbeans IDE, the program uses Java Socket to implement the client-server scheme. The author used the blackbox testing method to test the application, producing positive results in scenarios such as server startup, user connection, file sending and downloading, and connection management by the admin. The results of this study show that this application can be an efficient solution for file exchange in an internal network environment, with enhanced security. However, it still requires careful network management to ensure access is limited to internal users. Suggestions for further development include automating folder creation when users connect to the server. This application can thus contribute to meeting the needs of file exchange in an efficient and secure manner in certain network environments.

Keywords : *File exchange, network, Java Socket.*

PENDAHULUAN

Dalam era modern seperti sekarang ini, pertukaran berkas melalui jaringan telah menjadi bagian penting dari kehidupan sehari-hari. Banyak organisasi dan individu mengandalkan layanan pengiriman berkas untuk berbagi data dengan mudah dan aman. Dalam konteks ini, perancangan dan implementasi program pengiriman berkas yang efisien sangat diperlukan.

Semua kemajuan dalam teknologi jaringan mendorong transfer data yang cepat dan mudah. Saat ini telah terjadi perkembangan yang luar biasa dan berkelanjutan. Komputer pribadi lebih efisien dan bertenaga daripada super komputer sepuluh tahun yang lalu berkat kecepatan jaringan dan kapasitas stasiun kerja (Goscinski, 2019). Dengan berkembangnya teknologi jaringan, manajemen informasi dan aplikasi bisnis berkembang dalam dua arah. Pertama, arsitektur client server tradisional dengan mode struktur sistem web multi-layer. Kedua, lingkungan jaringan beralih dari pemrosesan terkonsentrasi ke terdistribusi (Xia, 2019).

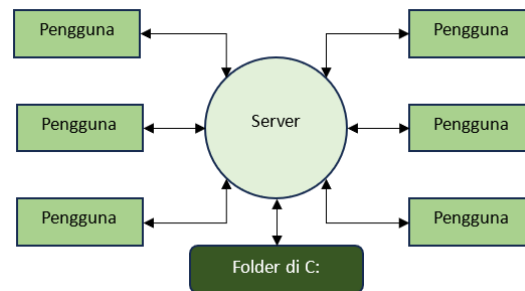
File data menjadi persyaratan yang sangat diperlukan dalam lingkungan saat ini, jika pada lingkungan kecil seperti 5 hingga 50 komputer, maka permasalahannya adalah mengelola transfer file tersebut dengan biaya yang rendah dan kompleksitas yang lebih sedikit (Arun Singh, 2020). Jaringan Internet tidak dapat digunakan karena kurangnya keamanan data rahasia dan ancaman Penyusup. Pembentukan dan pengelolaan protokol tersebut sekali lagi merupakan tugas khusus. Jadi, diperlukan aplikasi Client Server pribadi, dengan kemampuan untuk mendukung jenis file yang banyak digunakan seperti txt, doc, pdf, jpeg, mp3, mp4, flv dll, dengan keamanan khusus di lingkungan pribadi, dan jawabannya adalah pengiriman berkas terpusat pada server menggunakan Java Sockets.

METODE

Penulis menggunakan metode pengembangan waterfall yang sedikit dimodifikasi dalam penelitian ini (Pridarani, 2014). Metodologi waterfall merupakan metodologi pengembangan perangkat lunak yang menawarkan pendekatan sistematis dan berurutan terhadap perangkat lunak, dimulai dari tingkat pengembangan sistem melalui analisis, desain, kode, pengujian dan pemeliharaan sistem (Yogantara, 2018). Sedangkan pada penelitian ini penulis menggunakan metode yang melibatkan beberapa tahapan, seperti: analisa kebutuhan sistem, perancangan aplikasi, uji coba aplikasi. Berikut ini adalah penjelasan dari setiap tahapan (Adlini, 2022):

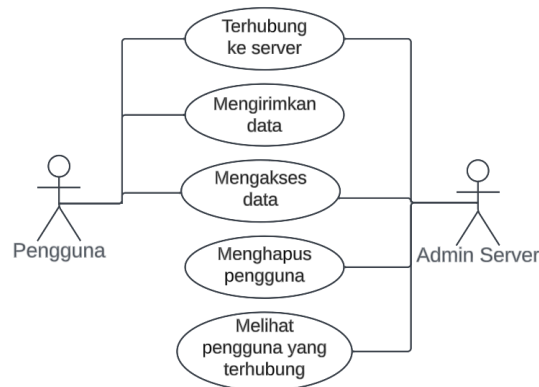
Analisa Kebutuhan Sistem

Pada tahap ini ditentukan apa saja yang menjadi kebutuhan sistem, termasuk fitur-fitur apa saja yang diperlukan agar sistem dapat digunakan dan berjalan dengan baik. Pada dasarnya, aplikasi pengiriman berkas ini akan menghubungkan banyak pengguna di satu jaringan, dan ini menjadi salah satu syarat agar pengguna dapat saling mengirimkan berkas di aplikasi. Terdapat satu server yang menjadi penghubung antar pengguna, sebelum server diaktifkan, pengguna tidak bisa terhubung ke server tersebut. Selain itu pengguna juga harus memiliki satu folder di penyimpanan C dengan nama folder yang sudah ditentukan, folder ini akan menjadi tempat terkumpulnya pengiriman data dari berbagai pengguna. Berikut adalah tampilan arsitektur dari aplikasi pengiriman berkas ini (Siswanto, 2010).



Gambar 1. Tampilan Arsitektur Aplikasi Pengiriman Berkas

Untuk menghubungkan pengguna ke server, pengguna harus memasukkan username. Hal ini bertujuan agar pengirim data dapat dikenali. Pengguna lain dapat mengetahui siapa saja yang sedang terhubung ke server. Setiap pengguna dapat di keluarkan atau di hapus melalui server oleh admin aplikasi. Berikut adalah tampilan dari use case diagram dari aplikasi pengiriman berkas.



Gambar 2. Use Case Diagram Aplikasi Pengiriman Berkas

Perancangan Aplikasi

Dalam perancangan aplikasi pengiriman berkas ini penulis menggunakan Integrated Development Environment (IDE) Apache NetBeans dan mengimplementasikan bahasa pemrograman Java. NetBeans adalah salah satu program yang paling banyak digunakan oleh programmer atau developer. Netbeans adalah IDE atau lingkungan pengembangan integrasi berbasis Java yang berjalan di atas Swing. Dengan kata lain, Swing adalah teknologi yang memungkinkan pengembangan aplikasi desktop dan berjalan di berbagai platform, seperti Windows, Mac OS, Linux, dan Solaris (Ertie, 2022).

Java merupakan bahasa pemrograman tingkat tinggi yang berorientasi objek, dan program Java terdiri dari bagian-bagian yang disebut kelas. Kelas terdiri dari metode-metode yang melakukan pekerjaan dan mengembalikan informasi setelah melakukan tugasnya. Pemrograman java menggunakan sekumpulan kelas di perpustakaan kelas Java yang disebut Java Application Programming Interface (API). Kelas-kelas diorganisasikan ke dalam kelompok-kelompok yang disebut paket. Antarmuka Pemrograman Aplikasi Java menyediakan fitur yang cukup untuk membuat applet dan aplikasi tingkat lanjut (Hidayat, 2018).

Untuk membuat skema client server dalam aplikasi penulis mengimplementasikan Java Socket dalam programnya. Socket adalah mekanisme komunikasi yang memungkinkan pertukaran data antar program atau proses dalam satu mesin dan antar mesin. Kebanyakan socket dibuat untuk memungkinkan dua program berkomunikasi satu sama lain dalam jaringan. Mampu menangani banyak klien sekaligus adalah kelebihan lain dari komunikasi socket.

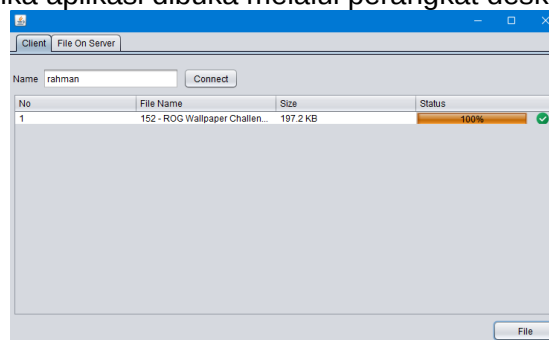
Uji Coba Aplikasi

Salah satu tahapan penting dalam pembuatan aplikasi adalah uji coba aplikasi dengan tujuan menemukan kesalahan atau bug sebelum aplikasi digunakan oleh pengguna akhir. Untuk tujuan ini, penulis menggunakan metode pengujian blackbox. Metode Blackbox Testing adalah salah satu pendekatan yang mudah digunakan karena hanya memerlukan penentuan batas bawah dan batas atas dari data yang diharapkan. Banyak jumlah data yang diuji dapat dihitung berdasarkan jumlah field data entri yang akan diuji, aturan entri yang harus dipatuhi, serta skenario batas atas dan batas bawah yang sesuai. Dengan menggunakan pendekatan ini, dapat ditemukan apakah fungsionalitas dapat menerima masukan data yang tidak sesuai, yang dapat mengakibatkan penyimpanan data yang kurang valid (W. N. Cholifah, 2018).

HASIL DAN PEMBAHASAN

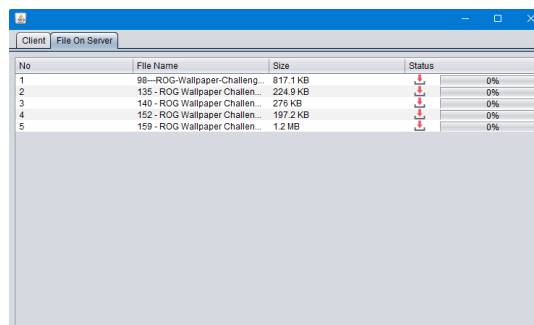
Tampilan Aplikasi Client

Berikut adalah tampilan dari sisi client ketika aplikasi pertama kali dibuka. Gambar berikut adalah tampilan jika aplikasi dibuka melalui perangkat desktop.



Gambar 3. Tampilan Aplikasi Pengguna Halaman Client

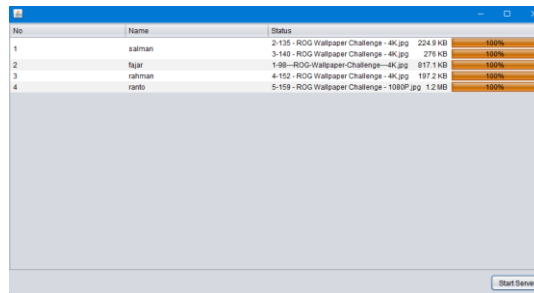
Terdapat 2 halaman di tampilan pengguna yaitu halaman client dan halaman file on server. Untuk halaman client diperlukan username pengguna untuk menyambungkan pengguna ke server. Terdapat sebuah tabel di halaman client yang nantinya akan berisi data apa saja yang sudah dikirim oleh pengguna tersebut. Untuk halaman file on server akan berisi semua data yang sudah berhasil terkirim dari berbagai pengguna yang terhubung dengan server. Pada halaman file on server pengguna bisa mendownload file dari server yang sudah dikirimkan oleh pengguna lain. Berikut adalah gambar tampilan halaman file on server.



Gambar 4. Tampilan Aplikasi Pengguna Halaman File On Server

Tampilan Aplikasi Server

Berikut adalah tampilan dari sisi server dari aplikasi pengiriman berkas. Gambar berikut adalah tampilan jika aplikasi dibuka melalui perangkat desktop.



Gambar 5. Tampilan Aplikasi Server

Pada tampilan server terdapat tombol start server yang berfungsi untuk menjalankan server. Sebelum server dijalankan, pengguna tidak dapat terhubung ke server. Pada gambar tampilan server diatas menunjukkan ada beberapa pengguna yang sedang terhubung ke server dan telah mengirimkan beberapa data yang berekstensi jpg. Semua data yang dikirim oleh pengguna yang terhubung ke server akan terdata di tabel yang ada pada server. Untuk memutuskan koneksi pengguna dengan server bisa klik kanan pada pengguna yang ingin di putuskan koneksinya, dan action tersebut hanya dapat dilakukan di sisi server saja.

Hasil Uji Coba Aplikasi

Berikut ini adalah hasil dari uji coba aplikasi pengiriman berkas dengan metode blackbox. Beberapa skenario pengujian akan dijelaskan dari tabel berikut.

Tabel 1. Skenario dan Hasil Pengujian Aplikasi

Skenario Pengujian	Hasil
Mengaktifkan server melalui tampilan server aplikasi	Berhasil
Pengguna mencoba terhubung ke server dengan memasukkan username dan mengklik tombol connect di halaman client	Berhasil
Pengguna mencoba mengirim data ke server dengan mengklik tombol file di halaman client	Berhasil
Pengguna melihat semua data yang ada pada server di halaman file on server	Berhasil
Pengguna mencoba mendownload data yang ada di server	Berhasil
Admin aplikasi melihat semua pengguna yang sedang terhubung ke server melalui tabel yang ada di tampilan aplikasi server	Berhasil
Admin aplikasi mencoba memutuskan koneksi pengguna yang sedang terhubung ke server dengan klik kanan pada pengguna yang akan diputuskan koneksinya	Berhasil

SIMPULAN

Aplikasi pengiriman berkas ini dirancang dengan Integrated Development Environment (IDE) Apache NetBeans dan mengimplementasikan program menggunakan bahasa pemrograman Java. untuk membuat skema program client server, program menggunakan salah satu library Java yaitu Java Socket. Aplikasi pengiriman berkas ini bisa berjalan jika pengguna berada di dalam jaringan yang sama. Pengguna akan terhubung ke sebuah server, yang dimana server tersebut harus diaktifkan dahulu oleh admin server. Pengguna harus mengisi username agar dapat terhubung ke server.

Pada aplikasi pengiriman berkas ini pengguna dapat mengirimkan data ke server, dan pengguna lainnya juga dapat melihat data di server tersebut yang dikirim oleh pengguna lain. Pengguna diharuskan membuat sebuah folder dengan nama folder yang sudah ditentukan di penyimpanan local disk C. Admin server dapat memutuskan koneksi pengguna yang terhubung dengan server melalui tampilan aplikasi server.

Dengan aplikasi ini pengguna dapat melakukan pengiriman data lebih aman dari serangan siber, dikarenakan pengiriman data hanya dilakukan pada satu jaringan yang sama dan tidak dapat diakses oleh orang luar. Tetapi apabila jaringan dapat diakses oleh orang luar itu sama saja tidak aman, oleh karena itu dibutuhkannya pengelolaan jaringan yang hanya dapat digunakan oleh orang internal saja.

DAFTAR PUSTAKA

- Adlini, M. N. (2022). METODE PENELITIAN KUALITATIF STUDI PUSTAKA. *Jurnal Edumaspul*, 6 (1).
- Arun Singh, A. K. (2020). MultiClient MultilInstance Centralized File Server using TCP protocol in Java Sockets. *International Journal of Computer Applications*, Vol. 23, No.8, pp. 24-29, 2011.
- Ertie, N. H. (2022). Aplikasi Inventori Barang Menggunakan Java Dengan . *PHPMYAdmin. Cross-border*, Vol. 5, No. 1, pp. 601-610.
- Goscinski, J. R. (2019). The development of an efficient checkpointing facility exploiting operating systems services of the GENESIS cluster operating system. *Future Generation Computer Systems*, Vol. 20, No. 4, pp 523-538.
- Hidayat, A. d. (2018). Rancang Bangun Aplikasi Perhitungan Grafik dan Energi Aktivitas Kegempaan Gunungapi Berbasis Java dan MySql. Vol 5, No. 2, pp. 21-30.
- Pridarani, M. M. (2014). Penggunaan metode Research and Development dalam penelitian Bahasa Indonesia di perguruan tinggi. *Seminar Nasional dan Launching Asosiasi Dosen Bahasa dan Sastra Indonesia (ADOBSI)*, dated 25 April 2015, (pp. 555–559). Surakarta: ADOBSI.
- Siswanto, S. (2010). *Systematic review sebagai metode penelitian untuk mensintesis hasil-hasil penelitian (sebuah pengantar)*. *Buletin Penelitian Sistem Kesehatan*, 13(4), 21312.
- W. N. Cholifah, Y. Y. (2018). Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap, . *STRING (Satuan Tulisan Ris. dan Inov. Telknol.*, vol. 3, no. 2, p. 206.
- Xia, D. H. (2019). Design of Distributed Architecture based on Java Remote Method Invocation Technology. pp. 618, *2009 International Conference on Environmental Science and Information Application Technology (IEEE Computer Society)*.
- Yogantara, W. A. (2018). Perancangan Jaringan Wireless Local Area Network Pada Dinas Pemerintah Kota Semarang. Pp. 1-9.