

## Research Article

**Aplikasi LAN Messenger Menggunakan Java Servlet**Priyo Nugroho Adi<sup>1</sup>, Ir. Kodrat Iman Satoto<sup>2</sup>, Maman Somantri<sup>2</sup>

1. Mahasiswa Jurusan Teknik Elektro Fakultas Teknik Universitas Diponegoro Semarang
2. Dosen Jurusan Teknik Elektro Fakultas Teknik Universitas Diponegoro Semarang

**Abstract**

Aplikasi LAN messenger dengan JAVA Servlet yang dibangun pada Penelitian ini berusaha memecahkan masalah pembatasan platform yang biasa terjadi pada aplikasi LAN messenger pada umumnya. Walaupun memungkinkan pada pengembangan tahap selanjutnya aplikasi ini mampu beroperasi pada jaringan yang berbeda pula ( karena berjalan pada protokol http ). Pembangunan aplikasi pada Penelitian ini menggunakan Servlet, yaitu salah satu kelas yang tersedia pada bahasa Pemrograman Java yang memungkinkan untuk Java me-request dan merespon protokol http. Dengan pedoman inilah aplikasi yang dibangun pada Penelitian ini dapat memberikan solusi pada pembatasan platform yang umumnya terjadi pada LAN messenger.

**Keyword :** LAN Messenger**1. PENDAHULUAN****1.1 Latar Belakang**

Penggunaan aplikasi *instant messenger* semakin populer seiring dengan menduniannya sistem operasi Windows pada akhir tahun 1990an. Para pengguna menyukai fitur *real time messaging* yang dimiliki aplikasi ini, namun masih terbatas pada *platform* dan jaringan yang sama. Seiring berjalannya waktu, *instant messenger* dikembangkan menjadi *online instant messenger* yang dapat berkomunikasi dengan *platform* dan jaringan yang berbeda melalui akses internet dan *server instant messenger*. Namun bukannya tanpa kelemahan, *online instant messenger* yang bekerja melalui *port-port* khusus ternyata rawan disusupi *spam* maupun virus jaringan. Atas alasan-alasan tersebut, beberapa perusahaan seperti Toyota Astra Motor dan Brataco Chemical Group tetap menggunakan LAN *instant messenger* walaupun dibatasi jenis *platform* dan jaringan. Aplikasi LAN Messenger dengan Java Servlet yang dibangun pada Penelitian ini berusaha memecahkan masalah pembatasan *platform* walaupun memungkinkan pada pengembangan tahap selanjutnya aplikasi ini mampu beroperasi pada jaringan yang berbeda pula.

**1.2 Tujuan**

Tujuan Penelitian ini adalah :  
Memberikan alternatif solusi pada masalah pembatasan *platform* yang biasa terjadi pada aplikasi LAN Messenger pada umumnya.

**1.3 Pembatasan masalah**

Berkaitan dengan keterbatasan yang penulis dalam Penelitian ini penulis memberikan beberapa batasan-batasan masalah sebagai berikut :

1. Pengujian terhadap sistem menggunakan jaringan lokal komputer tingkat sederhana (hanya *server* dan klien).
2. Pengujian pembuktian multi *platform* pada klien menggunakan sistem operasi Windows dan Linux Ubuntu.
3. Antarmuka *user* dengan sistem menggunakan antarmuka berbasis *web*.

4. Fasilitas yang akan diakomodasi pada LAN Messenger yang dibangun ini sebatas pada pengiriman pesan teks, baik secara privat, maupun *broadcast*.

**2. DASAR TEORI****2.1 LAN Messenger**

LAN Messenger adalah piranti lunak penyampai pesan *instant* (dalam hal ini dalam pembatasan pesan *instant* komputer) yang didesain untuk penggunaan dalam satu area jaringan lokal saja. Sebenarnya tiap sistem operasi telah memiliki fungsi bawaan untuk saling bertukar pesan, namun tetap saja masih dibatasi pada sistem operasi sejenis maupun yang masih satu pengembang[4]. Beberapa kelebihan LAN Messenger adalah :

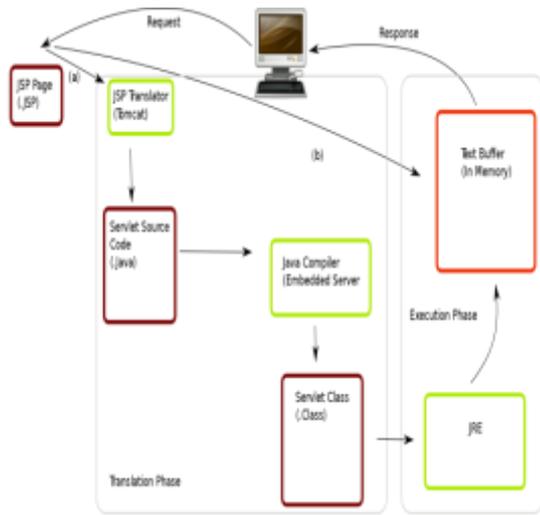
1. Tidak memerlukan akses internet untuk saling berkomunikasi.
2. Pengguna sistem ini umumnya ada didalam *firewall*, sehingga perlindungan *firewall* dapat diatur ke tingkat yang lebih optimum.
3. Komunikasi terjadi didalam jaringan lokal saja dan tidak meninggalkan *gateway*, sehingga kerahasiaan data dapat terjamin di dalam jaringan lokal.

**2.2 JAVA**

Java adalah bahasa pemrograman yang dikembangkan oleh James Gosling di bawah naungan Sun Microsystem dan resmi dirilis pertama kali pada tahun 1995. Bahasa pemrograman java memiliki karakteristik *general purpose*, berbasis kelas, berorientasi objek dan termodularisasi.[5]

**2.3 Servlet**

Servlet adalah sebuah kelas dalam bahasa pemrograman Java yang memungkinkan Java dapat merequest dan merespon protokol http. Jadi, dengan menggunakan *servlet*, pengembang program dapat membuat konten dinamis dalam *web server* dengan menggunakan *platform* Java dan menjalankannya.[6]



Gambar 2.2 Siklus dari Java Server Pages

### 2.4 Java Server Pages (JSP)

JSP adalah suatu teknologi web berbasis bahasa pemrograman Java dan berjalan di platform Java, serta merupakan bagian dari teknologi J2EE (*Java 2 Enterprise Edition*). Teknologi JSP menyediakan cara yang lebih mudah dan cepat untuk membuat halaman-halaman web yang menampilkan isi secara dinamis. Teknologi JSP didesain untuk mempermudah dan mempercepat dalam pembuatan aplikasi berbasis web yang bekerja dengan berbagai macam web server, application server, browser dan development tool. JSP juga merupakan bahasa scripting untuk web programming yang bersifat server side seperti halnya PHP dan ASP. JSP dapat berupa gabungan antara baris HTML dan fungsi-fungsi dari JSP itu sendiri[2].

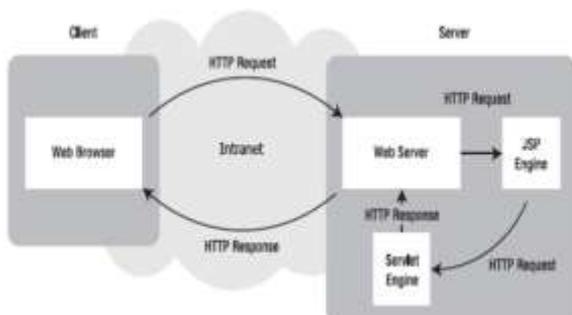
### 2.5 MySQL

MySQL adalah sebuah sistem manajemen database relasi (*relational database management system*) yang bersifat “terbuka” (*open source*). Terbuka maksudnya MySQL boleh di download dan dikembangkan oleh siapa saja

## 3. PERANCANGAN SISTEM

### 3.1 Gambaran Umum JSP

Sistem yang dibangun pada penelitian ini adalah LAN Messenger menggunakan Java Servlet, yang kemudian akan disebut LAN Messenger saja. Sistem ini memungkinkan beberapa user saling berinteraksi pesan melalui web browser. Karena sistem ini dibangun dengan berbasis Web Java (JSP), sistem ini memungkinkan interaksi user antar sistem operasi yang berbeda yang tentu saja mendukung Java. Gambar 3.1 menunjukkan gambaran umum bagaimana arsitektur JSP memungkinkan sistem LAN Messenger dapat berjalan[1].



Gambar 3.1 Gambaran umum arsitektur Java Server Pages.

Gambar 3.1 menjelaskan beberapa langkah bagaimana server web menciptakan sebuah halaman web. Langkah-langkah tersebut adalah :

- 1) Web browser mengirimkan HTTP request ke web server. HTTP request untuk JSP tidak berbeda dengan request html umumnya, meski halaman yang di request berekstensi.jsp.
- 2) Web server merupakan Java Server, dengan ekstensi yang diperlukan untuk mengidentifikasi dan menangani Java Servlet. Saat web server mengenali bahwa HTTP request merupakan permintaan halaman JSP, web server akan menyampaikan request tersebut ke JSP engine.
- 3) JSP engine akan memuat halaman JSP dari disk dan mengubahnya menjadi Java Servlet. Dari titik ini, servlet dari JSP tak dapat terbedakan dari servlet lain yang dikembangkan langsung dari Java, meskipun kode Java yang dibuat secara otomatis sebuah servlet JSP sulit untuk dibaca, dan tidak dimungkinkan untuk mengubahnya begitu saja.
- 4) JSP engine mengkompilasi servlet menjadi sebuah kelas yang dapat dieksekusi dan meneruskan permintaan asli ke sebuah servlet engine. Perlu diperhatikan bahwa JSP engine hanya mengkonversi halaman JSP ke Java dan merekompilasi ulang servlet jika menemukan bahwa halaman JSP telah berubah sejak permintaan terakhir. Hal inilah yang membuat proses JSP lebih efisien dibandingkan dengan bahasa scripting lainnya (seperti PHP) dan sebab itu juga JSP memiliki respon yang lebih cepat.
- 5) Sebuah bagian dari web server disebut servlet engine memuat kelas servlet dan mengeksekusinya. Selama eksekusi, servlet menghasilkan keluaran dalam format HTML, yang diteruskan kepada web server dalam sebuah respon HTTP.
- 6) Web server meneruskan HTTP respon ke browser client.

Web browser client menangani halaman HTML yang dihasilkan secara dinamis di dalam HTTP respon tepat seperti seolah halaman tersebut adalah halaman statis.

### 3.2 Diagram Konteks LAN Messenger

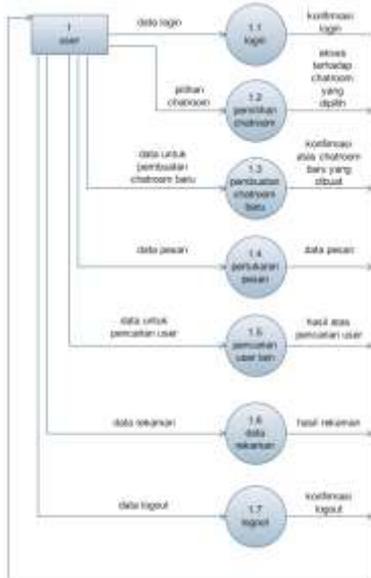
Diagram konteks merupakan level tertinggi dalam Diagram Alir Data. Diagram konteks memperlihatkan aliran data sistem secara utuh dalam proses. Gambar 3.5 adalah diagram konteks Aplikasi LAN Messenger yang menggunakan Java Servlet.



Gambar 3.2 Gambaran umum arsitektur Java Server Pages.

**3.3 Diagram Alir Data Tingkat 1 LAN Messenger**

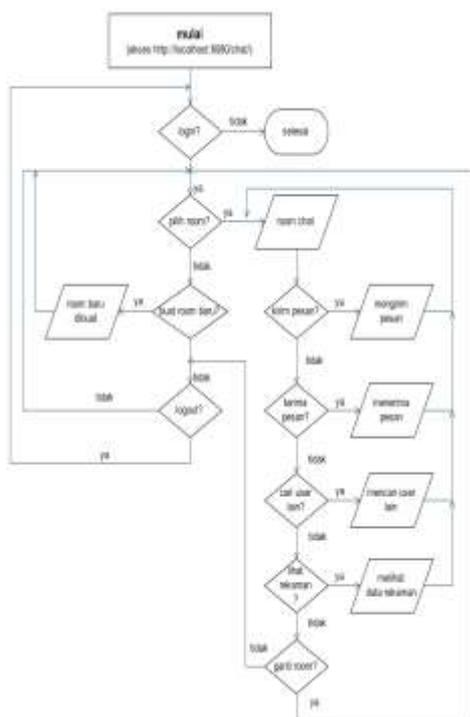
Diagram Alir Data tingkat 1 *user LAN Messenger* memberikan gambaran yang lebih jelas mengenai sub-proses yang terjadi dalam interaksi *user* dan aplikasi *LAN Messenger*. Diagram Alir Data tingkat 1 untuk *user* ini terlihat bahwa sub-proses *user* terdiri dari tujuh kemungkinan.



Gambar 3.3 DFD level 1 User LAN Messenger

**3.4 FlowChart**

Dalam analisis perancangan aplikasi *LAN Messenger*, *flowchart* digunakan untuk menggambarkan alur kemungkinan proses dari *user* dalam mengakses ke dalam aplikasi *LAN Messenger*. Proses yang dilakukan *user* dalam aplikasi yaitu : *login*, pemilihan *chatroom*, pembuatan *chatroom* baru, pertukaran pesan, pencarian *user*, dan *logout* dari aplikasi. Di halaman berikut adalah gambar *flowchart user* untuk aplikasi *LAN Messenger*.



Gambar 3.4 Flowchart User LAN Messenger.

**4. IMPLEMENTASI DAN HASIL PENGUJIAN SISTEM**

**4.1 Piranti lunak yang digunakan**

Dalam membangun aplikasi ini, piranti lunak yang digunakan adalah :

1. Macromedia Dreamweaver sebagai editor JSP
2. Apache Tomcat versi 7.0, sebagai container dan web server JSP
3. MySQL, sebagai aplikasi database

**4.2 Tampilan Aplikasi LAN Messenger**

Berikut adalah tampilan – tampilan antarmuka aplikasi LAN Messenger.



Gambar 4.1 Tampilan Halaman Login

Daftar Room	
<input type="radio"/> Akunting	Keterangan
<input type="radio"/> Marketing	Keterangan
<input type="radio"/> Gudang	Keterangan
<input type="radio"/> Invoice	Keterangan
<input type="radio"/> brataco	Keterangan
<input type="radio"/> IT	Keterangan
<input type="radio"/> All	Keterangan
<input type="radio"/> Ekspedisi	Keterangan
Buat room baru	Start

keluar daftar room

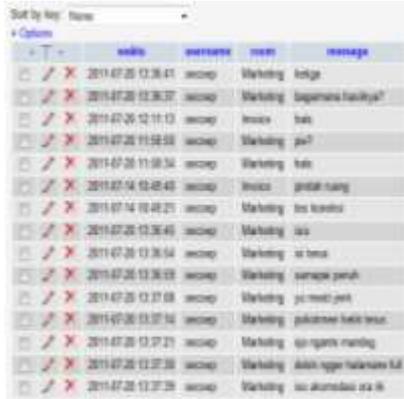
Gambar 4.2 Tampilan Memilih Room



Gambar 4.3 Tampilan Room Pertukaran Pesan



Gambar 4.4 Tampilan Halaman Pencarian User



Gambar 4.5 Tampilan Isi Seluruh Database



Gambar 4.6 Tampilan Halaman Logout

4.3 Tabel Hasil Pengujian Aplikasi LAN Messenger

No	Nama uji	Tujuan	Indikator	Hasil
1.	Uji tampilan aplikasi	Tampilan aplikasi yang tampil ketika aplikasi LAN Messenger diakses pada web server adalah tampilan aplikasi yang sesuai.	Ketika kita mengakses alamat web server, tampilan welcome page yang sesuai akan tampil.	sesuai
2.	Uji login	Kesesuaian data user dengan akses user serta mencegah duplikasi nama user.	User akan mengakses chatroom dengan data nama yang dimintanya, serta nama user yang terduplikasi tidak akan diberi akses. User lain juga akan diberi informasi jika ada user lain yang bergabung di chatroom.	sesuai
3.	Uji	Untuk	Akses	sesuai

	pemilihan chatroom	memastikan kesesuaian akses chatroom dengan pilihan user.	chatroom sesuai dengan pilihan chatroom user.	
4.	Uji pembuatan chatroom baru	Untuk memastikan chatroom baru yang dibentuk sesuai dengan data dan keterangan yang dimasukkan user.	Chatroom baru yang dibentuk sesuai dengan data dan keterangan yang dimasukkan user.	sesuai
5.	Uji pertukaran pesan	Untuk memvalidasi kesesuaian dan kecepatan tampilan pesan dalam chatroom dibandingkan dengan pengiriman pesan oleh user.	Tampilan pesan dalam chatroom dapat sesuai dan tidak terlalu lambat dibandingkan dengan pengiriman pesan oleh user.	sesuai
6.	Uji pencarian user	Untuk memastikan fungsi pencarian user di chatroom-chatroom yang aktif.	Hasil pencarian harus sesuai dengan kondisi user yang aktif di chatroom dan sesuai dengan data nama user yang dicari.	sesuai
7.	Uji data rekaman	Untuk memastikan data terekam dan dapat ditampilkan	Isi database harus sesuai dengan percakapan yang dilakukan	sesuai
8.	Uji logout	Untuk memastikan user mendapat akses keluar dari aplikasi LAN Messenger.	Jika tombol logout dipilih maka user akan kehilangan akses terhadap chatroom dan dibawa ke halaman logout dari aplikasi LAN Messenger. User lain juga akan mendapat konfirmasi bahwa ada user yang keluar dari chatroom.	sesuai

## 5. KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Dari Penelitian Aplikasi LAN Messenger, Laporan Penelitian dan pengujian – pengujian yang telah dilakukan pada aplikasi, penulis dapat menyimpulkan bahwa :

1. LAN messenger yang telah dibuat telah memenuhi syarat dasar messenger, yaitu[9]:
  - a. Mampu meregistrasi user baru
  - b. Mampu melakukan validasi login
  - c. Mampu menyediakan informasi yang *uptodate*
  - d. Mampu melakukan pertukaran pesan
  - e. Mampu melakukan interaksi antar user secara *custom*
2. Aplikasi LAN messenger dapat menjadi alternatif solusi pada masalah pembatasan *platform* yang umum terjadi pada aplikasi LAN Messenger.
3. JSP *engine* hanya mengkonversi halaman JSP ke Java dan merekompilasi ulang *servlet* jika menemukan bahwa halaman JSP telah berubah sejak permintaan terakhir. Itulah sebab mengapa JSP memiliki respon lebih cepat dan memakan lebih sedikit sumber daya komputer dibanding PHP.

### 5.2 Saran

1. Untuk pengembangan selanjutnya, dapat dikembangkan fasilitas *Voice Messenger* untuk melengkapi fasilitas LAN messenger yang sudah ada.
2. Untuk pemakaian skala yang lebih besar, ada baiknya bila *Cascade Style Sheet* dapat dirancang lebih profesional dan *User Friendly*.
3. Penampilan database dapat dibuat lebih interaktif dengan user jika nantinya dibutuhkan fitur yang lebih lanjut.

## 6. Referensi

- [1] Giulio Zambon, *Beginning JSP™, JSF™, and Tomcat™ Web Development*, Apress, 2007
- [2] Marty Hall, *Core Servlets and Java Server Pages*, 2<sup>nd</sup> ed., Sun Microsystem, >1998
- [3] IBM, *Servlet and JSP Programming*, IBM RedBook, 2000
- [4] [http://en.wikipedia.org/wiki/LAN\\_messenger](http://en.wikipedia.org/wiki/LAN_messenger), April 2011.
- [5] [http://en.wikipedia.org/wiki/Java\\_\(programming\\_language\)](http://en.wikipedia.org/wiki/Java_(programming_language)), April 2011
- [6] ---, <http://en.wikipedia.org/wiki/Servlet>, April 2011
- [7] ---, <http://id.wikipedia.org/wiki/HTTP>, April 2011
- [8] Sukwinder Singh, *Learning Servlet – Sukwinder Singh Project*, 2003
- [9] Zecho, *Software Development*, TechArena Online Publication, 2010