

**LEMBAR PERNYATAAN KEASLIAN  
HASIL TUGAS AKHIR**

Saya yang bertandatangan di bawah ini,

Nama : **Ridla Kumara Hadi**  
No.Mahasiswa : **02523064**

Menyatakan bahwa seluruh komponen dan isi dalam Laporan Tugas Akhir dengan judul **APLIKASI MULTIPLAYER ROLE PLAYING GAME MENGGUNAKAN KOMUNIKASI JARINGAN BERBASIS CLIENT-SERVER** yang diajukan untuk diuji pada tanggal Desember 2007 adalah hasil karya saya sendiri.

Apabila di kemudian hari terbukti bahwa ada beberapa bagian dari karya ini adalah bukan hasil karya saya sendiri, maka saya siap menanggung resiko dan konsekuensi apapun.

Demikian Pernyataan ini saya buat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, Desember 2007

**Ridla Kumara Hadi**  
( NIM 02 523 064 )

**LEMBAR PENGESAHAN PEMBIMBING  
APLIKASI MULTIPLAYER ROLE PLAYING GAME  
MENGUNAKAN KOMUNIKASI JARINGAN  
BERBASIS CLIENT-SERVER**



Pembimbing Payung,

Yudi Prayudi, S.Si., M.Kom.

Pembimbing Pelaksana,

Syarif Hidayat, S.Kom.

## LEMBAR PENGESAHAN PENGUJI

*Aplikasi Multiplayer Role Playing Game Menggunakan Komunikasi Jaringan Berbasis Client-Server*

Tugas Akhir

Disusun oleh :

Nama : Ridla Kumara Hadi  
No. Mahasiswa : 02 523 064

Telah Dipertahankan di Depan Sidang Penguji sebagai Salah Satu Syarat untuk Memperoleh Gelar Sarjana Jurusan Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, Januari 2008

Tim penguji :

Yudi Prayudi, Ssi., M.Kom.

Ketua

Syarif Hidayat, S.Kom.

Anggota I

Lizda Iswari, ST.

Anggota II

Mengetahui,



Dekan Fakultas Teknologi Industri  
Universitas Islam Indonesia

*Fathul Wahid*  
Fathul Wahid, ST., MSc.

**MOTTO :**

"Senyummu dihadapan saudaramu adalah shadaqahmu (untuknya). Perintahmu kepada hal yang ma'ruf dan laranganmu kepada yang mungkar juga menjadi shadaqahmu. Petunjukmu bagi seseorang yang tersesat disuatu tempat juga menjadi shadaqahmu. Kamu menyingkirkan batu, duri dan tulang dari jalanan juga menjadi shadaqah bagimu"  
(Hadist Riwayat al-Bukhari)

"Dunia hanya berjalan tiga hari, yaitu : Kemarin, yang kita tidak berpengharapan apa-apa lagi darinya. Hari ini, yang harus kita peroleh kebaikan dan kesuksesannya. Dan esok hari, yang tidak kita ketahui apakah kita termasuk yang masih hidup atau yang tergolong sudah meninggal"  
(Al Hasan Al Bashri)

## Kata Pengantar

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

*Assalamu'alaikum Wr. Wb*

Alhamdulillah, segala puji syukur hanyalah kepada Allah SWT dan semoga sholawat serta salam dilimpahkan kepada junjungan kita Nabi Muhammad SAW, keluarga, sahabat, dan pengikut-pengikut beliau (amin). Sehingga penulisan laporan tugas akhir yang berjudul : **Aplikasi Multiplayer Role Playing Game Menggunakan Komunikasi Jaringan Berbasis Client-Server** dapat penulis selesaikan dengan baik.

Laporan tugas akhir ini disusun untuk melengkapi salah satu syarat guna memperoleh gelar Sarjana Teknik Informatika pada Universitas Islam Indonesia dan atas apa yang telah diajarkan selama perkuliahan baik teori maupun praktek, disamping laporan itu sendiri yang merupakan rangkaian kegiatan yang harus dilakukan setelah tugas akhir ini selesai.

Untuk itu saya menyampaikan ucapan terimakasih dan penghargaan yang setinggi-tingginya kepada :

1. Bapak Prof. Dr. Edi Suandi Hamid, selaku Rektor Universitas Islam Indonesia.
2. Bapak Fathul Wahid, ST., M.Sc. selaku Dekan Fakultas Teknologi Industri Universitas Islam Indonesia.

3. Bapak Yudi Prayudi. S.Si, M.Kom. selaku Ketua Jurusan Teknik Informatika serta Pembimbing Payung Tugas Akhir.
4. Bapak Syarif Hidayat, ST selaku Pembimbing Pelaksana Tugas Akhir dan seluruh dosen pengajar Jurusan Teknik Informatika. Terimakasih atas segala bantuan, dukungan, semangat, dan pengetahuannya, serta kemudahan yang telah diberikan.
5. Teman-teman Jurusan Teknik Informatika angkatan 2002 dan seluruh pihak yang tidak dapat saya sebutkan satu persatu.

Semoga apa yang telah mereka berikan dengan keikhlasan, mendapat pahala yang setimpal dari Allah SWT. Penulis menyadari dalam penulisan laporan tugas akhir ini masih jauh dari sempurna, karena keterbatasan kemampuan dan pengalaman. Penulis mengharapkan saran dan kritik yang bersifat membangun untuk memperbaiki tugas akhir ini semoga dapat bermanfaat bagi penulis khususnya dan pembaca pada umumnya.

*Wassalamu'alaikum Wr. Wb*

Yogyakarta, Desember 2007

Penulis

## SARI

Teknologi jaringan kini telah memasuki perkembangan yang semakin pesat. Banyak kebutuhan harian masyarakat perindustrian membutuhkan teknologi jaringan, bahkan untuk orang awam sekalipun. Berbagai fasilitasnya telah banyak digunakan oleh masyarakat, bahkan telah menjadi bagian kehidupan masyarakat yang tak terpisahkan. Sehingga pada akhirnya, kebutuhan terhadap teknologi jaringan menjadi menu pokok dalam keseharian.

*Game* digital adalah salah satu aktivitas yang menghibur dan dapat dinikmati untuk semua usia, tergantung pada jenis game yang sesuai dengan kemampuan nalar seseorang. Dengan adanya perkembangan teknologi *game engineering* yang begitu pesat, *game* digital saat ini tidak hanya dimainkan pada *standalone console device*, tetapi juga dapat dimainkan secara *multiplayer* dengan menggunakan komunikasi jaringan. Dengan sifat *multiplayer* memungkinkan suatu *game* dapat dimainkan oleh banyak orang pada waktu yang bersamaan. Untuk memungkinkan pemain berkomunikasi dengan pemain lain diperlukan suatu jaringan yang menghubungkan setiap *device* dengan *game server* yang berada pada satu jaringan.

Pengembangan Aplikasi *Multiplayer Role Playing Game* Menggunakan Komunikasi Jaringan Berbasis *Client-Server* ini didasari oleh semakin maraknya *game* digital yang masuk kedalam masyarakat. Penelitian ini menggunakan VbGORE sebagai *game engine*. Aplikasi *game* yang dihasilkan adalah *Game Kahriman*, yang bertemakan pertarungan manusia melawan monster dengan konsep RPG yang membutuhkan nilai untuk menaikkan level tokoh karakter dan penyelesaian *quest* untuk mendapatkan skill.

\*kata kunci : *game, multiplayer, client-server, RPG*

## TAKARIR

|                               |  |
|-------------------------------|--|
| <i>account</i>                | akun                                     |
| <i>address</i>                | alamat                                   |
| <i>admin</i>                  | pengelola                                |
| <i>artificial intelegence</i> | kecerdasan buatan                        |
| <i>avatar</i>                 | wujud grafis tokoh karakter              |
| <i>bandwith</i>               | kapasitas transmisi sambungan elektronik |
| <i>bit</i>                    | unit terkecil dari informasi             |
| <i>browser</i>                | aplikasi penjelajah internet             |
| <i>carrier</i>                | sinyal                                   |
| <i>character</i>              | karakter tokoh pada permainan            |
| <i>class</i>                  | kelas tipe karakter                      |
| <i>client</i>                 | aplikasi akses layanan                   |
| <i>database</i>               | basis data                               |
| <i>entertainment</i>          | hiburan                                  |
| <i>fiber optic</i>            | kabel penghantar sinar                   |
| <i>frame</i>                  | kerangka                                 |
| <i>framework</i>              | ruang bekerja                            |
| <i>game</i>                   | permainan                                |
| <i>game engine</i>            | mesin pengembangan permainan             |
| <i>game engineering</i>       | teknologi pengembangan permainan         |
| <i>game programming</i>       | pemrograman permainan                    |

|                                  |                                       |
|----------------------------------|---------------------------------------|
| <i>gameplay</i>                  | cara bermain                          |
| <i>Graphical User Interfaces</i> | antarmuka grafis untuk pengguna       |
| <i>hardware</i>                  | perangkat keras                       |
| <i>host</i>                      | sebuah komputer terhubung ke jaringan |
| <i>input</i>                     | masukan                               |
| <i>interface</i>                 | antarmuka                             |
| <i>layer</i>                     | lapisan                               |
| <i>level</i>                     | tingkatan                             |
| <i>login</i>                     | masuk ke dalam sistem                 |
| <i>logout</i>                    | keluar dari sistem                    |
| <i>multiplayer</i>               | banyak pemain                         |
| <i>object oriented</i>           | berorientasi objek                    |
| <i>online</i>                    | komunikasi terbuka                    |
| <i>password</i>                  | sandi                                 |
| <i>peer to peer</i>              | komunikasi dua arah tanpa server      |
| <i>platform</i>                  | dasar penopang sistem kerja           |
| <i>player</i>                    | pemain                                |
| <i>posting</i>                   | mengirimkan                           |
| <i>quest</i>                     | tugas, misi                           |
| <i>realtime</i>                  | waktu nyata                           |
| <i>role</i>                      | peran, tugas                          |
| <i>script</i>                    | kode bahasa pemrograman               |
| <i>server</i>                    | aplikasi penyedia layanan             |

|                          |                                      |
|--------------------------|--------------------------------------|
| <i>skill</i>             | kemampuan                            |
| <i>software</i>          | perangkat lunak                      |
| <i>standalone</i>        | berdiri sendiri                      |
| <i>user</i>              | pengguna                             |
| <i>username</i>          | nama pengguna                        |
| <i>user friendly</i>     | mudah dipahami                       |
| <i>wide area network</i> | jaringan komputer meliputi area luas |



## DAFTAR ISI

|   |          |
|---|----------|
| LEMBAR JUDUL.....                       | i        |
| LEMBAR PERNYATAAN KEASLIAN.....         | ii       |
| LEMBAR PENGESAHAN DOSEN PEMBIMBING..... | iii      |
| LEMBAR PENGESAHAN PENGUJI.....          | iv       |
| KATA PENGANTAR.....                     | v        |
| SARI.....                               | vi       |
| TAKARIR.....                            | vii      |
| DAFTAR ISI.....                         | viii     |
| DAFTAR TABEL.....                       | xiii     |
| DAFTAR GAMBAR.....                      | xlv      |
| <b>BAB I    PENDAHULUAN.....</b>        | <b>1</b> |
| 1.1 Latar Belakang Masalah.....         | 1        |
| 1.2 Rumusan Masalah.....                | 2        |
| 1.3 Batasan Masalah.....                | 3        |
| 1.4 Tujuan Penelitian.....              | 3        |
| 1.5 Manfaat Penelitian.....             | 3        |
| 1.6 Metodologi Penelitian.....          | 4        |
| 1.6.1. Pengumpulan Data.....            | 4        |

|  |           |
|--|-----------|
| 1.6.2. Pengembangan Perangkat Lunak..... | 4         |
| 1.7 Sistematika Penulisan.....           | 6         |
| <b>BAB II LANDASAN TEORI.....</b>        | <b>8</b>  |
| 2.1 Jaringan Komputer.....               | 8         |
| 2.1.1. Pengertian Jaringan Komputer..... | 8         |
| 2.1.2. Cara Kerja Jaringan Komputer..... | 8         |
| 2.2 Multiplayer Role Playing Game.....   | 14        |
| 2.2.1. Dasar-dasar MRPG.....             | 15        |
| 2.2.2. Perekonomian dalam MRPG.....      | 16        |
| 2.3 Client-Server.....                   | 16        |
| 2.3.1. Client.....                       | 17        |
| 2.3.2. Server.....                       | 17        |
| 2.3.3. Cara Kerja Client-Server.....     | 17        |
| 2.4 vbGORE.....                          | 18        |
| 2.5 Visual Basic.....                    | 19        |
| 2.6 Database MySQL.....                  | 19        |
| <b>BAB III METODOLOGI.....</b>           | <b>21</b> |
| 3.1 Metode Analisis.....                 | 21        |
| 3.2 Analisis Masalah.....                | 21        |
| 3.3 Hasil Analisis.....                  | 22        |
| 3.3.1. Kebutuhan Client.....             | 22        |

|  |           |
|--|-----------|
| 3.3.1.1. Kebutuhan Masukan.....  | 22        |
| 3.3.1.2. Kebutuhan Fungsi.....   | 24        |
| 3.3.1.3. Keluaran.....   | 24        |
| 3.3.2. Kebutuhan Server.....   | 24        |
| 3.3.2.1. Kebutuhan Masukan.....  | 24        |
| 3.3.2.2. Kebutuhan Fungsi.....   | 25        |
| 3.3.2.3. Keluaran.....   | 25        |
| 3.4 Antarmuka Sistem.....  | 26        |
| 3.5 Metode Perancangan.....  | 28        |
| 3.6 Hasil Perancangan.....   | 28        |
| 3.6.1. Perancangan model proses dengan menggunakan State<br>Diagram..... | 28        |
| 3.7 Perancangan Struktur Basis Data.....                                 | 35        |
| 3.8 Relasi Antar Tabel.....  | 44        |
| 3.9 Perancangan Antar Muka .....   | 46        |
| <b>BAB IV HASIL DAN PEMBAHASAN.....</b>                                  | <b>50</b> |
| 4.1 Implementasi Secara Umum.....  | 50        |
| 4.2 Perangkat Lunak Yang Dibutuhkan.....                                 | 50        |
| 4.3 Perangkat Keras Yang Dibutuhkan.....                                 | 51        |
| 4.4 Tahapan Pembuatan Perangkat Lunak.....                               | 51        |
| 4.5 Implementasi Hasil Program.....                                      | 52        |

|  |            |
|--|------------|
| 4.5.1. Langkah-langkah Penggunaan Aplikasi vbGore..... | 52         |
| 4.5.1.1. Penambahan Citra.....                         | 52         |
| 4.5.1.2. Aplikasi Map Editor.....                      | 54         |
| 4.5.2. Aplikasi Server.....                            | 66         |
| 4.5.3. Aplikasi Client.....                            | 67         |
| 4.5.4. Aplikasi Konfigurasi Permainan.....             | 84         |
| 4.6 Analisis Kinerja Perangkat Lunak.....              | 85         |
| 4.6.1. Pengujian Normal.....                           | 85         |
| 4.6.2. Pengujian Tidak Normal.....                     | 129        |
| 4.7 Pembahasan.....                                    | 166        |
| 4.7.1. Gameplay.....                                   | 167        |
| 4.7.2. Antarmuka.....                                  | 167        |
| 4.7.3. Quest.....                                      | 167        |
| 4.7.5. Pengembangan Karakter.....                      | 168        |
| 4.7.6. Sistem Job Class.....                           | 168        |
| 4.7.7. Pemain Melawan Pemain.....                      | 169        |
| 4.7.8. Perang Emperium.....                            | 169        |
| 4.7.9. Sistem Point untuk Guild.....                   | 169        |
| <b>BAB V KESIMPULAN DAN SARAN.....</b>                 | <b>170</b> |
| 5.1 Kesimpulan.....                                    | 170        |
| 5.2 Saran.....   | 171        |

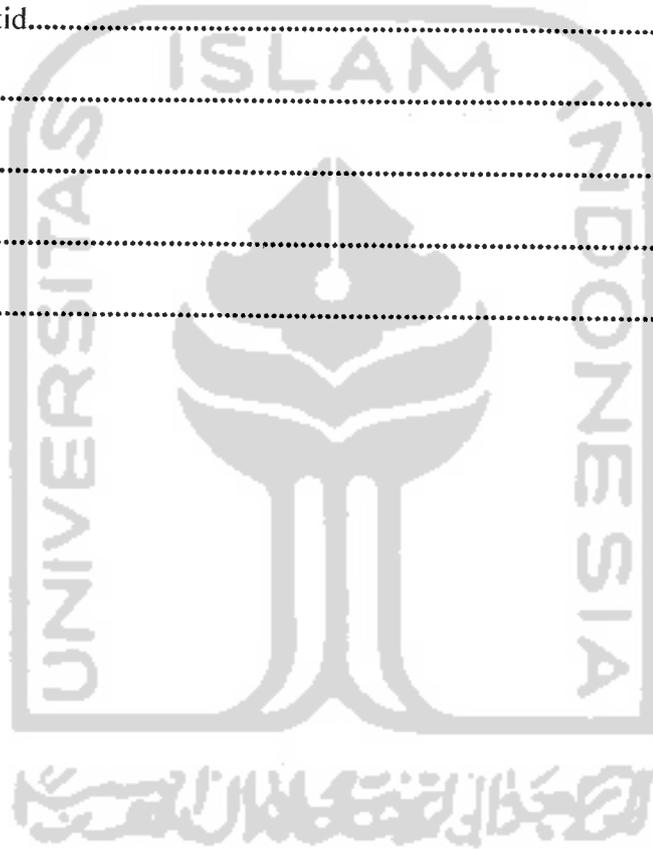
**DAFTAR PUSTAKA**

**LAMPIRAN**



## DAFTAR TABEL

|                                  |    |
|----------------------------------|----|
| Tabel 3.1 Tabel banned_ip.....   | 47 |
| Tabel 3.2 Tabel mail.....        | 47 |
| Tabel 3.3 Tabel mail_lastid..... | 48 |
| Tabel 3.4 Tabel npcs.....        | 48 |
| Tabel 3.5 Tabel objects.....     | 50 |
| Tabel 3.6 Tabel quests.....      | 52 |
| Tabel 3.7 Tabel users.....       | 54 |



## DAFTAR GAMBAR

|            |  |    |
|------------|--|----|
| Gambar 2.1 | Gambar Referensi model OSI dan proses encapsulation.....                                 | 10 |
| Gambar 2.2 | Gambar cara kerja sistem client-server.....  | 17 |
| Gambar 3.1 | State Diagram Aplikasi Multiplayer Role Playing Game<br>Berbasis Client-Server.....      | 29 |
| Gambar 3.2 | Relasi antar tabel Aplikasi Multiplayer Role Playing Game<br>berbasis client-server..... | 45 |
| Gambar 3.3 | Rancangan Menu Login.....  | 46 |
| Gambar 3.4 | Rancangan Menu Pembuatan Akun Baru.....  | 47 |
| Gambar 3.5 | Rancangan Layar Permainan.....   | 48 |
| Gambar 3.6 | Rancangan Konfigurasi Permainan.....   | 49 |
| Gambar 4.1 | Aplikasi Map Editor pada vbGore.....   | 54 |
| Gambar 4.2 | Search bar.....  | 55 |
| Gambar 4.3 | Hasil pencarian.....   | 55 |
| Gambar 4.4 | Jendela View Sheet.....  | 56 |
| Gambar 4.5 | Menu Map Editor.....   | 56 |
| Gambar 4.6 | Set Tile.....  | 59 |
| Gambar 4.7 | Tile Blocks.....   | 60 |
| Gambar 4.8 | Blokade gerakan dan blokade serangan jarak jauh pada tile.....                           | 60 |

|             |   |    |
|-------------|---|----|
| Gambar 4.9  | Map Floods.....   | 60 |
| Gambar 4.10 | Tile Information.....                                       | 61 |
| Gambar 4.11 | Jendela Exits.....  | 62 |
| Gambar 4.12 | Jendela NPC.....  | 63 |
| Gambar 4.13 | Particle effects.....                                       | 64 |
| Gambar 4.14 | Jendela sound effects.....                                  | 64 |
| Gambar 4.15 | Jendela map info.....                                       | 65 |
| Gambar 4.16 | Aplikasi server berjalan.....                               | 67 |
| Gambar 4.17 | Menu shut down aplikasi server.....                         | 67 |
| Gambar 4.18 | Menu login.....   | 68 |
| Gambar 4.19 | Menu pembuatan akun baru.....                               | 69 |
| Gambar 4.20 | Layar permainan.....  | 70 |
| Gambar 4.21 | Layar permainan dengan quickbar.....                        | 71 |
| Gambar 4.22 | Layar permainan dengan jendela chat.....                    | 72 |
| Gambar 4.23 | Layar permainan dengan jendela inventory.....               | 73 |
| Gambar 4.24 | Layar permainan dengan jendela statistik.....               | 74 |
| Gambar 4.25 | Layar permainan dengan jendela transaksi antar PC.....      | 75 |
| Gambar 4.26 | Layar Permainan dengan jendela transaksi PC dengan NPC..... | 76 |
| Gambar 4.27 | Layar permainan dengan jendela konfirmasi jumlah.....       | 77 |
| Gambar 4.28 | Layar permainan dengan daftar skill.....                    | 78 |
| Gambar 4.29 | Layar permainan dengan jendela mail box.....                | 79 |

|             |  |    |
|-------------|--|----|
| Gambar 4.30 | Layar permainan dengan jendela baca mail.....  | 80 |
| Gambar 4.31 | Layar permainan dengan tulis mail.....   | 81 |
| Gambar 4.32 | Layar permainan dengan jendela gudang penyimpanan.....   | 82 |
| Gambar 4.33 | Layar permainan dengan mini map.....   | 83 |
| Gambar 4.34 | Layar permainan kontrol pertarungan.....   | 84 |
| Gambar 4.35 | Aplikasi Konfigurasi Permainan.....  | 85 |
| Gambar 4.36 | Jendela chat informasi membunuh monster.....   | 86 |
| Gambar 4.37 | Jendela chat informasi terbunuh monster.....   | 87 |
| Gambar 4.38 | Jendela chat informasi naik level.....   | 88 |
| Gambar 4.39 | Jendela chat menerima pembicaraan tertutup.....  | 89 |
| Gambar 4.40 | Jendela chat berbicara tertutup.....   | 90 |
| Gambar 4.41 | Jendela chat informasi pemain bergabung bermain.....   | 91 |
| Gambar 4.42 | Jendela chat informasi pemain keluar permainan.....  | 92 |
| Gambar 4.43 | Jendela chat informasi jumlah pemain bermain.....  | 93 |
| Gambar 4.44 | Jendela chat informasi menerima <i>quest</i> dengan kebutuhan<br>membunuh monster.....             | 94 |
| Gambar 4.45 | Jendela chat informasi menerima <i>quest</i> dengan kebutuhan<br>membawakan barang.....            | 95 |
| Gambar 4.46 | Jendela chat informasi kebutuhan <i>quest</i> dan daftar <i>quest</i> yang<br>sedang dijalani..... | 96 |
| Gambar 4.47 | Jendela chat informasi menyelesaikan <i>quest</i> skill.....                                       | 97 |

|             |  |     |
|-------------|--|-----|
| Gambar 4.48 | Jendela chat informasi menyelesaikan quest non-skill.....                            | 98  |
| Gambar 4.49 | Jendela chat membatalkan quest yang belum selesai.....                               | 99  |
| Gambar 4.50 | Jendela chat informasi tidak ada mail pada saat login.....                           | 100 |
| Gambar 4.51 | Jendela chat informasi ada 1 mail pada saat login.....                               | 101 |
| Gambar 4.52 | Jendela chat informasi ada beberapa mail pada saat login.....                        | 102 |
| Gambar 4.53 | Jendela chat informasi ada mail baru pada saat bermain.....                          | 103 |
| Gambar 4.54 | Jendela chat informasi mail terkirim dan biaya pengiriman.....                       | 104 |
| Gambar 4.55 | Jendela chat informasi membunuh pemain lain.....                                     | 105 |
| Gambar 4.56 | Jendela chat informasi terbunuh pemain lain.....                                     | 106 |
| Gambar 4.57 | Jendela chat informasi melihat dan mengambil item dari tanah....                     | 107 |
| Gambar 4.58 | Jendela chat informasi melihat deskripsi NPC.....                                    | 108 |
| Gambar 4.59 | Jendela chat informasi melihat papan atau plang pengumuman....                       | 109 |
| Gambar 4.60 | Jendela chat informasi membeli barang dari NPC shop.....                             | 110 |
| Gambar 4.61 | Jendela chat informasi menjual barang ke NPC shop.....                               | 111 |
| Gambar 4.62 | Jendela chat informasi untuk bertransaksi dengan pemain lain....                     | 112 |
| Gambar 4.63 | Jendela chat informasi untuk membalas ajakan bertransaksi<br>dengan pemain lain..... | 113 |
| Gambar 4.64 | Jendela chat informasi transaksi sukses (membeli item<br>mengurangi gold) .....      | 114 |
| Gambar 4.65 | Jendela chat informasi transaksi sukses (menjual item<br>mendapatkan gold) .....     | 115 |

|             |  |     |
|-------------|--|-----|
| Gambar 4.66 | Jendela chat informasi membuat group.....  | 116 |
| Gambar 4.67 | Jendela chat informasi ajakan bergabung dengan group.....                                    | 117 |
| Gambar 4.68 | Jendela chat informasi pemain bergabung dengan group (pemain baru bergabung) .....           | 118 |
| Gambar 4.69 | Jendela chat informasi pemain bergabung dengan group (pemain dalam group) .....              | 119 |
| Gambar 4.70 | Jendela chat informasi pemain keluar dari group (pemain keluar)                              | 120 |
| Gambar 4.71 | Jendela chat informasi pemain keluar dari group (pemain dalam group yang ditinggalkan) ..... | 121 |
| Gambar 4.72 | Jendela chat informasi GM merubah pemain lain menjadi GM....                                 | 122 |
| Gambar 4.73 | Jendela chat informasi pemain diberi experience oleh GM.....                                 | 123 |
| Gambar 4.74 | Jendela chat informasi GM mengeluarkan pemain dari permainan secara paksa (Kick) .....       | 124 |
| Gambar 4.75 | Jendela chat informasi GM memblokade alamat IP.....  | 125 |
| Gambar 4.76 | Jendela chat informasi GM membuka blokade alamat IP.....                                     | 126 |
| Gambar 4.77 | Jendela chat informasi GM melihat daftar blokade alamat IP.....                              | 127 |
| Gambar 4.78 | Pesan peringatan untuk load settingan terakhir.....  | 127 |
| Gambar 4.79 | Pesan peringatan untuk menyimpan setting saat ini.....                                       | 128 |
| Gambar 4.80 | Pesan peringatan untuk mengembalikan ke settingan default.....                               | 128 |
| Gambar 4.81 | Pesan untuk keluar aplikasi tanpa menyimpan settingan.....                                   | 129 |

|             |  |     |
|-------------|--|-----|
| Gambar 4.82 | Pesan kesalahan pengisian nilai pemain dan password untuk koneksi MySQL pada konfigurasi server.ini..... | 130 |
| Gambar 4.83 | Pesan kesalahan pengisian nilai database untuk koneksi MySQL pada konfigurasi server.ini.....            | 131 |
| Gambar 4.84 | Pesan kesalahan pengisian nilai alamat ip untuk koneksi MySQL pada konfigurasi server.ini.....           | 131 |
| Gambar 4.85 | Pesan kesalahan tidak mendeteksi driver ODBC untuk koneksi MySQL pada konfigurasi server.ini.....        | 132 |
| Gambar 4.86 | Pesan kesalahan tabel tidak ditemukan pada database untuk koneksi MySQL pada konfigurasi server.ini..... | 132 |
| Gambar 4.87 | Pesan kesalahan Login dengan akun yang sedang bermain.....   | 133 |
| Gambar 4.88 | Pesan kesalahan Login dengan kata sandi salah.....   | 133 |
| Gambar 4.89 | Pesan kesalahan login dengan pemain yang belum terdaftar.....  | 134 |
| Gambar 4.90 | Pesan kesalahan membuat akun baru menggunakan nama yang sudah terdaftar.....                             | 135 |
| Gambar 4.91 | Pesan kesalahan membuat akun baru dengan panjang nama kurang dari 4 karakter.....                        | 136 |
| Gambar 4.92 | Pesan kesalahan membuat akun baru dengan panjang nama lebih dari 10 karakter.....                        | 136 |

|              |   |     |
|--------------|---|-----|
| Gambar 4.93  | Pesan kesalahan membuat akun baru dengan panjang kata sandi kurang dari 4 karakter.....                   | 136 |
| Gambar 4.94  | Pesan kesalahan membuat akun baru dengan panjang kata sandi lebih dari 10 karakter.....                   | 136 |
| Gambar 4.95  | Pesan kesalahan pengisian alamat ip server pada saat login.....   | 137 |
| Gambar 4.96  | Pesan kesalahan ada perbedaan versi map client dengan server....  | 137 |
| Gambar 4.97  | Pesan peringatan alamat ip telah diblok.....  | 138 |
| Gambar 4.98  | Pesan dalam permainan tidak dapat mengulang quest yang tidak bisa diulang.....                            | 139 |
| Gambar 4.99  | Pesan dalam permainan tidak dapat menjalani quest lebih dari satu.....                                    | 140 |
| Gambar 4.100 | Pesan dalam permainan tidak dapat menerima quest yang bukan untuk classnya.....                           | 141 |
| Gambar 4.101 | Pesan dalam permainan tidak ada quest yang sedang dijalani.....   | 142 |
| Gambar 4.102 | Pesan dalam permainan harus berada didekat mailbox untuk mengakses mailbox.....                           | 143 |
| Gambar 4.103 | Pesan dalam permainan tidak dapat mengirimkan mail karena nama penerima tidak terdaftar.....              | 144 |
| Gambar 4.104 | Pesan dalam permainan tidak dapat mengirimkan mail karena tidak mampu membayar biaya pengiriman mail..... | 145 |

|              |  |     |
|--------------|--|-----|
| Gambar 4.115 | Pesan dalam permainan Tidak memenuhi persyaratan statistik untuk menggunakan item.....                                   | 156 |
| Gambar 4.116 | Pesan dalam permainan tidak bisa memanggil summon lebih dari satu.....   | 157 |
| Gambar 4.117 | Pesan dalam permainan transaksi batal karena pemain yang diajak berransaksi tidak memiliki ruang pada inventory-nya..... | 158 |
| Gambar 4.118 | Pesan dalam permainan transaksi batal karena tidak memiliki ruang pada inventory-nya.....                                | 159 |
| Gambar 4.119 | Pesan dalam permainan tidak dapat bertransaksi dengan diri sendiri.....  | 160 |
| Gambar 4.120 | Pesan dalam permainan memblokade alamat IP dengan memasukan susunan alamat IP yang salah.....                            | 161 |
| Gambar 4.121 | Pesan dalam permainan memblokade alamat IP tanpa alasan.....   | 162 |
| Gambar 4.122 | Pesan dalam permainan memblokade alamat IP yang sedang diblokade.....  | 163 |
| Gambar 4.123 | Pesan dalam permainan membuka blokade alamat IP yang tidak sedang diblokade.....   | 164 |
| Gambar 4.124 | Pesan dalam permainan mengatur pemain yang tidak terdaftar menjadi GM.....   | 165 |
| Gambar 4.125 | Pesan dalam permainan gagal memunculkan monster.....   | 166 |



## BAB I

### PENDAHULUAN

#### 1.1 Latar Belakang Masalah

Perkembangan teknologi informasi yang begitu cepat telah mempengaruhi segala aspek kehidupan manusia, kemajuan teknologi ini sudah sangat dirasakan, dimana teknologi ini dapat dimanfaatkan dalam bentuk komunikasi jaringan lokal maupun *wide area* yang dapat membantu di segala aspek kehidupan. Berbagai fasilitas yang ditawarkan pada komunikasi jaringan ini telah menjadi bagian dari kehidupan sehari-hari baik yang sifatnya industrial maupun rumah tangga. Sehingga kebutuhan terhadap aplikasi jaringan akan mempermudah dalam kebutuhan informasi dan komunikasi sehari-hari.

Salah satu aplikasi yang biasanya digunakan pada suatu sistem jaringan adalah aplikasi berbasis *Client-Server*. Aplikasi sistem *Client-Server* merupakan sistem yang paling baik untuk digunakan pada sistem jaringan. Sistem berbasis *Client-Server* ini mampu menghasilkan aplikasi database yang tangguh dalam hal keamanan, serta mampu mengurangi kepadatan lalu lintas jaringan. Aplikasi *Client-Server* ini mampu berjalan baik pada jaringan lokal, *wide area* ataupun pada internet. Dengan demikian, fleksibilitas aplikasi ini dapat disesuaikan dengan kondisi jaringan yang diaplikasikan guna efektifitas *bandwidth* yang berakibat pada naiknya kecepatan pertukaran data.

Dalam perkembangannya teknologi jaringan kini telah merambah media *entertainment* dan multimedia. Telah muncul berbagai macam *game* digital

dengan fitur-fitur menarik yang lebih menarik daripada permainan tradisional. Dengan pertimbangan masalah sosial kini banyak berkembang *game multiplayer* yang mengharuskan pemain berinteraksi dengan pemain lainnya dibandingkan *game standalone*. Jenis *game multiplayer* lebih digemari karena lebih menyenangkan untuk dimainkan secara bersama-sama.

Model game multiplayer ada bermacam-macam, salah satunya adalah *Role Playing Game* (RPG) yang sedang populer sekarang ini. Dalam RPG para pemain dapat memilih sendiri perannya dalam permainan. Seperti halnya dalam kehidupan sehari-hari, dalam *Multiplayer* RPG setiap *character* berkembang dengan kenaikan *level* dan memperoleh *skill* yang digunakan untuk meningkatkan nilai sosialnya. Selain berinteraksi dengan pemain lainnya melalui chatting, pemain juga berinteraksi dengan *Non-Playing Character* (NPC) untuk melakukan *quest*. Pemain juga dapat melakukan transaksi dengan pemain lainnya maupun dengan NPC.

## 1.2 Rumusan Masalah

Bagaimana mengembangkan sebuah aplikasi *multiplayer* RPG yang dapat dimainkan banyak pemain dengan masing-masing komputernya sebagai client yang menerima masukan dari pemain dan terkoneksi pada suatu jaringan dimana pada jaringan tersebut terdapat sebuah komputer *server* yang berisi *Data Base Management System* (DBMS) serta *database* dan menjadi pusat seluruh kegiatan untuk menangani permintaan-permintaan yang dilakukan oleh banyak *client* secara *realtime* dengan menggunakan framework vbGORE, dan pemrograman menggunakan Visual Basic dan database MySQL.

### 1.3 Batasan Masalah

Dalam hal ini, batasan masalah yang harus diperhatikan adalah :

1. Perangkat lunak yang akan digunakan untuk aplikasi ini menggunakan *script basic*, SQL dan dijalankan di *platform Windows XP*.
2. Aplikasi yang dibangun menggunakan *vbcore* sebagai *game engine* atau *framework*.
3. Metode komunikasi yang digunakan untuk *multiplayer* adalah *Client-Server*.
4. *Game* dengan konsep *Role Playing Game* dimana *game* tidak memiliki alur cerita yang ditentukan.
5. *Game* adalah pemain melawan *Non-Playing Character* yang memiliki *artificial intelligence*.
6. Aplikasi *game multiplayer* berjalan pada setiap client di suatu jaringan yang sudah ada dan terkoneksi dengan aplikasi *server* yang telah berjalan.

### 1.4 Tujuan Penelitian

Tujuan diadakannya penelitian ini adalah sebagai berikut :

1. Membuat *multiplayer RPG* dengan *game engineering* menggunakan aplikasi *vbGORE* yang memakai sistem khusus untuk permainan *ORPG*.
2. Merancang dan mengimplementasikan sebuah aplikasi *game multiplayer* yang dapat dijadikan sebagai hiburan dan simulasi untuk membangun rasa kebersamaan dan kerjasama tim.

## 1.5 Manfaat Penelitian

1. Mengoptimalkan fungsi jaringan sebagai teknologi yang dibutuhkan pada setiap aspek kehidupan khususnya dunia *entertainment*.
2. Aplikasi game multiplayer yang berorientasi pada kerjasama tim dan sistem RPG yang berorientasi pada pengembangan individual sebagai prestasi yang patut dicontoh pada kehidupan keseharian.
3. Sebagai proses pembelajaran penulis dalam penerapan *script* basic dan SQL dalam pembuatan aplikasi ini.

## 1.6 Metodologi Penelitian

### 1.6.1 Pengumpulan Data

Pada penelitian ini, metode pengumpulan data yang digunakan adalah sebagai berikut :

a. Metode studi pustaka

Metode studi pustaka, yaitu metode mengumpulkan data dengan mencari referensi dari buku tentang konsep dan algoritma pemrograman dengan visual basic maupun MySQL .

b. Metode browsing

Mengumpulkan materi-materi dengan cara mengikuti forum-forum yang berkaitan dengan konsep-konsep *game programming* maupun Multiplayer RPG di website dan melakukan *posting* topik-topik yang akan ditanyakan kepada forum.

### 1.6.2 Pengembangan Perangkat Lunak

Metode pembuatan aplikasi ini disusun berdasarkan hasil dari data yang telah diperoleh . Metode ini meliputi :

a. Analisis masalah *Multiplayer* RPG

Tahap ini dilakukan untuk mengolah data yang diperoleh dan mengelompokkannya sesuai dengan kebutuhan perancangan

b. Desain

Tahap ini merupakan tahap penerjemahan kebutuhan atau hasil analisis ke dalam bentuk yang mudah dimengerti oleh pemain (*user*) dan perancangan grafis, *user interface* dan *gameplay* .

c. Implementasi

Tahap ini dilakukan untuk menerapkan hasil dari rancangan kedalam bentuk *game*.

d. Pengujian

Setelah aplikasi selesai dibuat, maka pada tahap ini merupakan tahap pengujian terhadap aplikasi tersebut. Pengujian ini dapat dilakukan dengan melakukan percobaan-percobaan dengan kondisi yang berbeda-beda. Dalam aplikasi ini, pengujian dilakukan secara *standalone* (bukan jaringan) dengan menggunakan localhost server, dimana *server* dan *client* berjalan pada satu mesin yang sama. Pengujian juga dilakukan dengan jaringan *peer to peer* antara 2 komputer. Sehingga diharapkan dapat menciptakan suatu permainan yang mudah dan menyenangkan bagi pemainnya.

## 1.7 Sistematika Penulisan

Sistematika yang digunakan dalam penyusunan laporan tugas akhir ini adalah sebagai berikut :

### Bab I Pendahuluan

Memuat latar belakang penelitian, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penelitian.

### Bab II Landasan Teori

Membahas dasar-dasar teori yang digunakan dalam merancang dan membangun aplikasi *Multiplayer Role Playing Game* beserta tinjauan pustaka.

### Bab III Metodologi

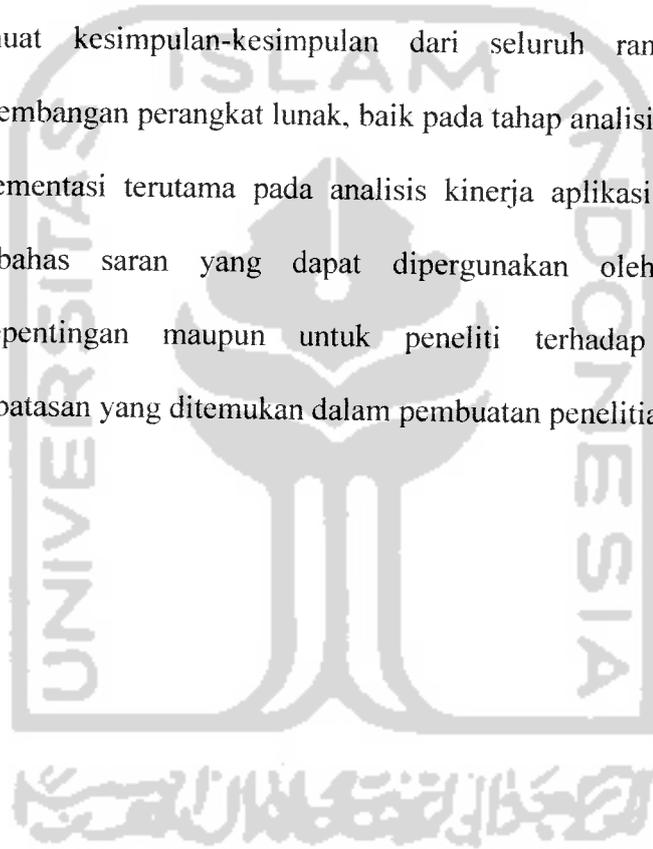
Memuat uraian tentang metode analisis, analisis masalah, analisis sistem dan hasil analisis mengenai semua perangkat lunak yang diperlukan untuk membangun sistem serta menghasilkan pemilahan kebutuhan sistem yang diwujudkan dari perangkat lunak, meliputi kebutuhan masukan dan keluaran sistem, fungsi-fungsi yang dibutuhkan, dan antarmuka yang diinginkan. Berikut dengan ini juga dicantumkan proses desain, implementasi dan pengujian aplikasi.

#### Bab IV Hasil Dan Pembahasan

Memuat dokumentasi hasil pengujian terhadap aplikasi yang dibandingkan kesesuaiannya dengan analisis dan perancangan yang telah dilakukan sebelumnya apakah sesuai dengan kebutuhan pengguna dan memiliki kelayakan sebagai suatu aplikasi yang baik

#### Bab V Simpulan Dan Saran

Memuat kesimpulan-kesimpulan dari seluruh rangkaian proses pengembangan perangkat lunak, baik pada tahap analisis, perancangan, implementasi terutama pada analisis kinerja aplikasi. Bab ini juga membahas saran yang dapat dipergunakan oleh pihak yang berkepentingan maupun untuk peneliti terhadap keterbatasan-keterbatasan yang ditemukan dalam pembuatan penelitian ini.



## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Jaringan Komputer**

##### **2.1.1 Pengertian Jaringan Komputer**

Jaringan komputer adalah sekelompok komputer otonom yang saling berhubungan satu dengan yang lainnya menggunakan protokol komunikasi melalui media transmisi, sehingga dapat saling berbagi menggunakan sumber daya yang ada dan berkomunikasi [WAH06a].

##### **2.1.2 Cara Kerja Jaringan Komputer**

Bagian utama dalam jaringan komputer, adalah sistem komunikasi data. Sistem komunikasi data membutuhkan media sebagai pembawa sinyal (*carrier*), yang dapat berupa kabel, *fiber optic* dan juga melalui udara (sinyal gelombang radio). Untuk menerima dan mengirimkan sinyal dari media perantara tersebut menggunakan NIC (*Network Interface Card*) yang terdapat pada komputer. Setiap NIC memiliki alamat khusus yang disebut sebagai MAC (*Media Access Control address*) yang terdiri dari kode heksadesimal 48 bit yang berbeda antara NIC satu dengan yang lainnya. *MAC address* menghindarkan data bertabrakan ketika berkomunikasi pada jaringan. Jika ada paket data yang akan dikirimkan maka sebelumnya akan melihat apakah jaringan sedang mengirimkan paket data atau tidak. Jika jaringan tidak mengirimkan paket data maka node akan mengirimkan paket data.

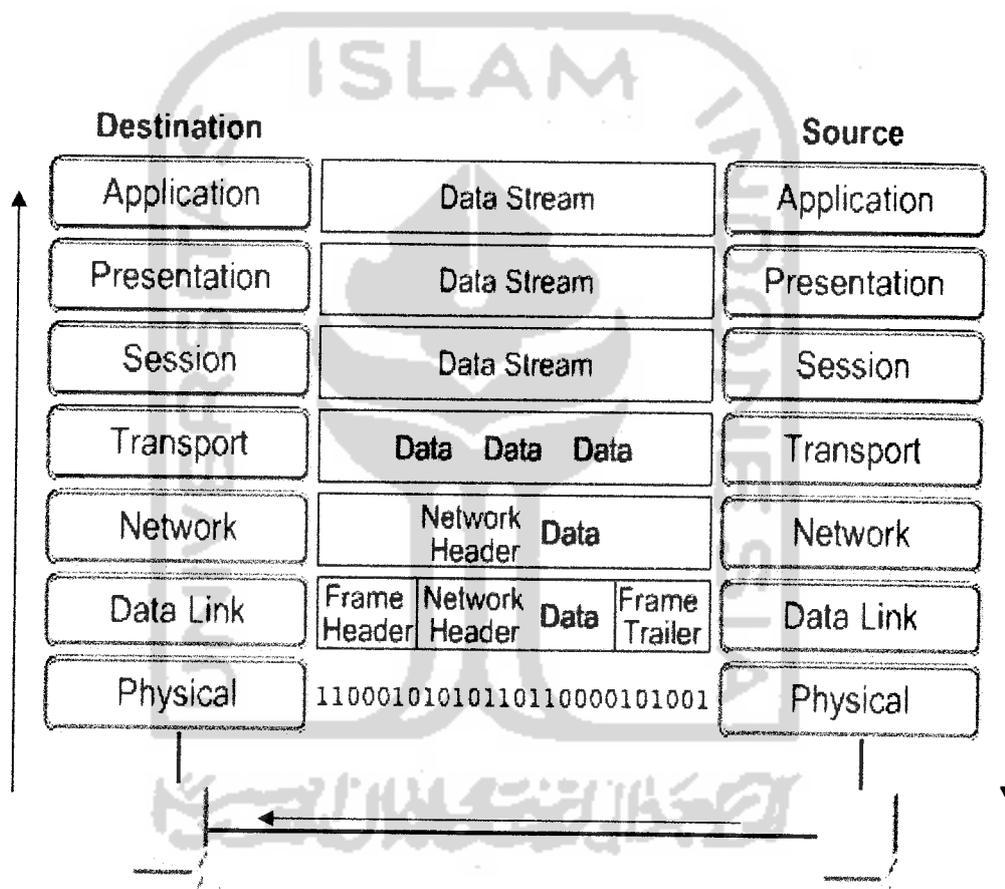
Pada jaringan komputer informasi yang disampaikan dari komputer satu ke komputer lainnya informasi tersebut harus dikirimkan lewat jaringan dan mengalami proses yang panjang melalui berbagai lapisan jaringan. Pertama-tama, informasi tersebut diolah menjadi data, kemudian diolah menjadi segmen-segmen, diolah menjadi paket-paket, menjadi *frame* dan terakhir menjadi *bit* yang dapat dikirimkan lewat kabel jaringan untuk disampaikan ke komputer lain, untuk diproses balik untuk mendapatkan informasi asal. Agar berfungsi dengan baik maka jaringan komputer tersebut memerlukan aturan dan definisi standar untuk berkomunikasi dengan interpreter yang dimengerti oleh seluruh peralatan dari berbagai perusahaan pemroduksi alat yang ada.

Model referensi OSI (*Open System Interconnection*) merupakan salah satu arsitektur jaringan komputer yang dibuat oleh ISO (*International Standardization Organization*) untuk memecahkan permasalahan kompatibilitas *device* antar vendor. Model referensi OSI mengidentifikasi semua proses yang dibutuhkan untuk melakukan komunikasi dan membaginya kedalam kelompok secara logika yang disebut *layer*. OSI merupakan petunjuk bagi para pengembang aplikasi dalam membuat dan mengimplementasikan aplikasinya berjalan pada sebuah jaringan.

OSI terdiri dari tujuh *layer*, yang secara umum terbagi dalam dua kelompok, yaitu *layer* atas (*Application Layer*) dan *layer* bawah (*Data Transport Layer*). Lapisan yang tergolong dalam *layer* atas mendefinisikan bagaimana aplikasi pada sebuah *host* akan berkomunikasi dengan *host* lainnya. Sedangkan

layer bawah mendefinisikan bagaimana data terkirim dari satu *host* ke *host* lainnya[WAH06b].

Proses pengelolaan data dari lapisan ke lapisan lain disebut *data encapsulation*, dimana terdapat penambahan informasi depan (*header information*) data pada masing-masing lapisan yang disebut *Protocol Data Unit* [WIJ04a].



**Gambar 2.1** Gambar Referensi model OSI dan proses encapsulation

Berikut ini penjelasan dari gambar 2.1 Susunan Layer Referensi model OSI :

a. Lapisan Application

Lapisan *Application* memberikan layanan ke jaringan komputer untuk aplikasi-aplikasi *user* dan mengadakan komunikasi dari perangkat lunak ke perangkat lunak.

b. Lapisan Presentation

Lapisan ini memberikan layanan untuk konversi, sintaks, format dan enkripsi data, dengan menampilkan informasi dalam bentuk teks maupun grafis.

c. Lapisan Session

Lapisan ini membuka, mengatur dan menutup *session* antara aplikasi aplikasi. Protokol yang berfungsi pada lapisan ini antara lain NFS, NETBEUI, RPC, SQL, X Windows System, Apple Talk Session Protocol (ASP), dan *Digital Network Architecture Session Control Program*[WIJ04b].

Lapisan ini juga menentukan apakah informasi yang diminta oleh user berasal dari komputer lokal atau komputer lain dalam jaringan. Informasi yang berasal dari komputer lain akan dimintai *authentication* untuk memulai *session* pada komputer lokal.

d. Lapisan Transport

Lapisan transport menjaga keutuhan transmisi data. Pada lapisan ini data diubah menjadi segmen atau data *stream*.

Lapisan ini terdiri dari dua jenis koneksi yaitu *connection-oriented* dan *connectionless-oriented*.

Pada jenis *connection-oriented* memiliki koneksi yang reliable, pada setiap sessionnya bergaransi dengan didukung *Transmission Control Protocol* (TCP) dengan menggunakan port 6. *Connection-oriented* mampu mengirim ulang paket yang tidak diterima dan paket yang telah terkirim mendapatkan tanda terima (*acknowledge*) dari penerima. *Connection-oriented* menggunakan fasilitas yang dinamakan *three-way handshake* untuk mengadakan koneksi. Koneksi ini banyak dipakai untuk aplikasi menyalurkan data audio dan video yang tidak memiliki toleransi terhadap kemacetan jaringan. Untuk penggunaan yang didedikasikan pada suatu hubungan tertentu, maka penggunaan *bandwith* kurang efektif, karena jalur yang digunakan harus dengan jalur yang telah ditentukan maka jika terjadi *collision* pada jalur hubungan akan terputus.

Untuk menghindari agar pengiriman data tidak melampaui kapasitas penerimaan yang dapat mengakibatkan kemacetan lalu lintas jaringan, dipergunakan teknik *flow control*.

*Flow control* terdiri dari tiga jenis, yaitu *buffering*, *source quench message* dan *windowing*. *Buffering* adalah menyimpan data yang akan dikirim secara sementara di memori jika penerima sedang sibuk. Sedangkan *source quench message* mengirimkan sinyal agar pengirim memperlambat tempo pengiriman. Pada *windowing* penerima akan mengirimkan tanda terima. Jika pengirim tidak menerima setelah beberapa

saat, pengirim akan mengirim ulang dengan tempo diperlambat. Selain itu masih ada metode *sliding windows* untuk menggunakan *bandwith* secara efektif, jalur diisi terus dengan paket-paket data sampai penuh.

Lain halnya dengan *connectionless-oriented* yang menggunakan protokol User Datagram Protocol (UDP) mempunyai hubungan yang tidak reliable dan tidak bergaransi. Penerima tidak mengirimkan tanda terima dan paket-paket tidak diurut kembali seperti asalnya.

*Connectionless-oriented* menggunakan *bandwith* yang efektif karena semua jalur yang tersedia dapat digunakan oleh user lain. Jalur yang digunakan bergantung paket per paket, jika terjadi kemacetan paket dapat disalurkan ke jalur lain.

e. Lapisan Network

Lapisan network menyediakan topologi logika jaringan yang memungkinkan penggunaan alamat logika seperti IP *address*. Dengan IP *address* lapisan network berfungsi meneruskan paket-paket dari satu node ke node yang lain dalam jaringan dengan memilih jalur yang terbaik dalam meneruskan paket di jaringan. Pada lapisan ini segmen-segmen diubah menjadi paket-paket dengan menambah informasi mengenai alamat logika yang dituju dan alamat asal paket.

f. Lapisan Data Link

Lapisan data-link mengatur topologi jaringan, *error notification* dan *flow control*. Lapisan ini menyediakan fasilitas alamat perangkat keras dan mengolah paket dari lapisan *network* menjadi *frame* dengan

menambahkan informasi mengenai MAC *address* yang dituju dan alamat asal.

g. Lapisan Physical

Lapisan physical menentukan spesifikasi koneksi fisik jaringan seperti tipe kabel, tipe konektor, hubungan pin konektor dengan kabel dan tipe antarmuka suatu peralatan jaringan. Lapisan ini bertanggung jawab untuk mengaktifkan dan mengatur antarmuka fisik dari jaringan komputer.

## 2.2 Multiplayer Role Playing Game

MMORPG (*Massively Multiplayer Online Role Playing Game*) adalah sebuah genre dari permainan komputer role playing online dimana ada banyak jumlah pemain yang bermain dan berinteraksi satu dengan lainnya didalam sebuah dunia maya. Seperti halnya dalam permainan *role playing* biasanya, pemain mengasumsikan peran dari sebuah karakter fiksi dan memainkan aksi-aksi dari karakter tersebut. MMORPG dibedakan dari permainan single-player ataupun multiplayer kecil RPG dari jumlah pemainnya, dan dari dunia maya permainan yang persisten, yang terus menerus ada dan berjalan walaupun ketika tidak ada pemain yang bermain [WIK07a].

Sedangkan Multiplayer RPG adalah MMORPG yang disederhanakan jangkauan jaringan nya sebatas pada jaringan area lokal dengan tidak mengurangi sifat *massive* nya sehingga bisa dimainkan oleh orang banyak secara bersamaan.

MRPG memiliki dua sisi perangkat lunak yang harus di eksekusi yaitu sisi *client* dan sisi *server*. *Client* akan mengakses ke *server* pada jaringan dan *server* akan melayani nya bersamaan dengan *client-client* lainnya.

### 2.2.1 Dasar-dasar MRPG

Seperti halnya dengan RPG, MRPG memiliki fitur dasar yang wajib ada, seperti berikut:

- a. Gaya permainan pertarungan dengan banyak *dungeon* dan cerita-cerita legenda , monster-monster, termasuk *quest* (misi) dan *loot* (*item drop*).
- b. Sistem untuk karakter dapat berkembang, biasanya memakai kenaikan level dan nilai *experience*.
- c. *Job* atau *class* tiap karakter yang memiliki keunggulan dan kelemahan masing-masing, termasuk juga *job* atau *class skill* yang juga memiliki nilai *experience* untuk berkembang.
- d. Perekonomian, berdasarkan pada perdagangan dan transaksi item-item seperti senjata dan pakaian, dan memiliki mata uang tetap.
- e. *Guild* atau *clan*, adalah organisasi dari para pemain, dengan atau tidak dengan secara aktif didukung oleh game.
- f. Game Moderator atau Game Master (sering disebut GM), adalah individu yang mengawasi dunia yang berjalan pada game apakah terjadi kesalahan atau *bug* yang harus diperbaiki.
- g. Model client-server, dimana didalamnya terdapat dunia yang berjalan secara terus-menerus dalam server yang sedang berjalan, dan pemain terkoneksi dengan server melalui client.
- h. Komunitas pemain dan jaringan sosial.

### 2.2.2 Perekonomian dalam MRPG

Kehidupan ekonomi pada MRPG meliputi barang dan mata uang yang harus didapatkan melalui permainan dan nilainya yang terbatas bagi pemain. Ekonomi dunia maya ini terkadang memiliki pengaruh terhadap ekonomi di kehidupan nyata.

Dalam bukunya *Synthetic Worlds: The Business and Culture of Online Games*, Edward Castronova meneliti fitur-fitur pada MMORPG dan pengaruhnya pada kehidupan nyata, menyatakan bahwa kebutuhan dan persediaan pasar ada pada dunia maya dan hal itu menyeberang dan mempengaruhi hingga ke dunia nyata. Untuk itu yang dibutuhkan pada permainan MMORPG adalah:

- a. Kemampuan pemain untuk menjual barang ke pemain lain untuk mendapatkan mata uang maya didalam game.
- b. Kemampuan bertukar barang dengan barang atau nilai mata uang yang setara antara pemain dengan pemain lainnya.
- c. Pembelian barang virtual dengan mata uang nyata.
- d. Pertukaran mata uang maya dengan mata uang nyata [WIK07b].

### 2.3 Client-Server

Sistem client-server untuk saat ini merupakan sistem yang paling baik untuk digunakan karena mampu menghasilkan aplikasi database yang tangguh dalam hal sekuritas dan juga mampu mengurangi kepadatan lalu lintas jaringan. Pada jenis sistem ini terdapat dua aplikasi terpisah yang saling berhubungan, yaitu aplikasi client dan aplikasi server.

### 2.3.1 Client

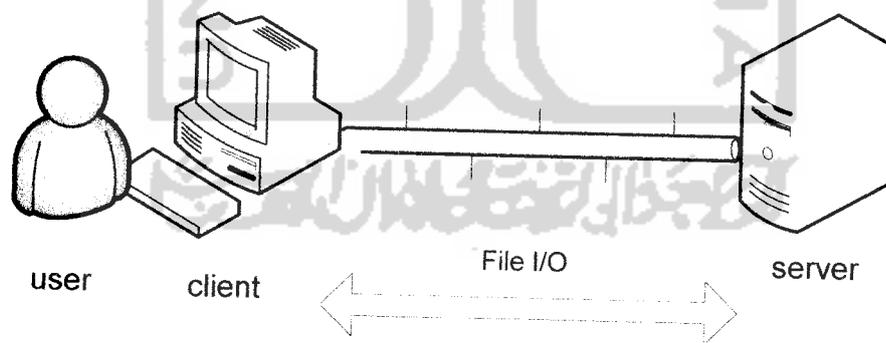
Client merupakan sebuah komputer yang berisi aplikasi, dan memiliki tanggung jawab untuk melakukan interaksi dengan user. Setiap input yang dimasukkan user akan ditangani oleh client yang nantinya akan diproses menuju server. Data yang telah diproses akan dikembalikan dan ditampilkan oleh client sebagai output kepada user.

### 2.3.2 Server

Server merupakan komputer yang berisi DBMS serta database, dan menjadi pusat seluruh kegiatan untuk menangani permintaan-permintaan yang diminta client.

### 2.3.3 Cara Kerja Client-Server

Sistem ini bekerja melalui permintaan yang dilakukan client kemudian server merespon permintaan. Dengan adanya dua aplikasi yang terpisah ini maka server harus terlebih dahulu dijalankan sebelum client dapat terhubung ke server.



**Gambar 2.2** Gambar cara kerja sistem client-server

Mekanisme keamanan data diterapkan pada sistem *client-server* sangat tinggi, sehingga aplikasi *client* tidak dapat membuka file-file data yang berada pada *server* secara langsung. Pada dasarnya *client* harus melakukan *login* dengan

memasukkan *user id* dan *password* yang telah ditentukan. Begitu proses *login* berhasil maka aplikasi *server* akan memberikan layanan untuk membuka data yang diminta oleh *client*. Dalam hal pengaksesan data ditentukan dengan hak akses sehingga tidak semua servis dapat dilakukan oleh *client*, tetapi bergantung pada hak akses yang diberikan.

Proses *query* pada sistem database *client-server* sangat sederhana, dimana aplikasi *client* cukup mengirimkan pernyataan-pernyataan SQL ke *server*. Perintah yang dikirim kemudian akan direspon oleh *server* dengan kembali mengirimkan hasil yang diminta tersebut.

Model sistem database *client-server* seperti ini memungkinkan kuantitas data yang ditransmisikan dapat direduksi, sehingga waktu akses yang diperlukan menjadi sangat singkat.

#### 2.4 vbGORE

vbGORE adalah sebuah *open source Online Role Playing Game engine*, dibangun memakai Visual Basic 6 dengan penggunaan dukungan dari DirectX 8 dan Winsock API *libraries* [VBG07].

Dengan penggunaan DirectX 8, didapatkan efek tiga dimensi, sedangkan vbGORE adalah engine dua dimensi. Kemampuannya meliputi *alphablending* dan rotasi hingga penskalaan dan pencahayaan. Pada vbGORE juga terdapat sistem pengiriman paket biner dan sistem pembungkusan paket Sox Winsock API, yang mendukung koneksi terhadap 80 pemain dengan *bandwith* 40 kbps minimum, dengan alokasi 0,5 kbps per koneksi *clientnya*.

## 2.5 Visual Basic

Microsoft Visual Basic adalah bahasa pemrograman yang digunakan untuk membuat aplikasi Windows yang berbasis grafis (GUI- *Graphical User Interface*).

Visual Basic merupakan *event-driven programming* (pemrograman terkendali kejadian) artinya program menunggu sampai adanya respon dari pemakai berupa kejadian tertentu. Ketika *event* terdeteksi, kode yang berhubungan dengan *event* akan dijalankan[KUS00].

## 2.6 Database MySQL

MySQL merupakan database yang dikembangkan dari bahasa SQL (*Structured Query Language*). SQL sendiri merupakan bahasa terstruktur yang digunakan untuk interaksi antara *script* program dengan database *server* dalam hal pengolahan data. Dengan SQL, dapat dibuat tabel yang nantinya akan diisi data, memanipulasi data (menambah data, menghapus data dan memperbaharui data), serta membuat suatu perhitungan dengan berdasarkan data yang ditemukan.

SQL tidak hanya terbatas digunakan untuk mendapatkan suatu tampilan database yang statis, saat ini sedang dikembangkan standar baru yaitu SQL3, yang direncanakan menjadi bahasa yang mendekati mesin turing, semisal *computable query* atau *recursive query* [PRA04a].

MySQL adalah database yang memiliki kecepatan yang tinggi dalam melakukan pemrosesan data, dapat diandalkan, dan mudah digunakan dan dipelajari. MySQL mendukung banyak bahasa pemrograman seperti C,C++, Perl, Python, Java dan PHP. Bahasa pemrograman tersebut dapat digunakan untuk

berinteraksi maupun berkomunikasi dengan MySQL server, atau dapat digunakan sebagai pembentuk komponen antarmuka dari suatu database MySQL. Selain itu, dengan bantuan *driver* ODBC, MySQL juga mampu berinteraksi dengan berbagai pemrograman visual seperti Delphi, Visual Basic dan sebagainya.

MySQL dapat melakukan koneksi dengan *client* menggunakan protokol TCP/IP, Unix socket (Unix), atau Named Pipes (NP). Dalam hal kemampuan menampung data, MySQL dapat menangani database dengan skala besar dengan *record* mencapai lebih dari lima puluh juta *record*. dapat menampung enam puluh ribu tabel dan juga menampung lima milyar baris data. Selain itu batas indeks pada tiap tabel dapat menampung mencapai tiga puluh dua indeks [PRA04b].

Dalam hal relasi antar tabel pada suatu database, MySQL menerapkan metode yang sangat cepat yaitu metode *one-sweep multijoin*. MySQL juga bersifat multiuser, yaitu dapat diakses oleh beberapa user dalam waktu bersamaan tanpa mengalami konflik pada sistem. Untuk sekuritas MySQL memiliki lapisan level *subnetmask* , nama *host* dan ijin akses user dengan sistem perijinan khusus serta kata sandi yang dimiliki setiap user dalam bentuk terenkripsi.

## **BAB III**

### **METODOLOGI**

#### **3.1 Metode Analisis**

Analisis kebutuhan perangkat lunak dilakukan untuk mengetahui semua permasalahan serta kebutuhan yang diperlukan dalam pengembangan aplikasi. Analisis dilakukan dengan mencari dan menentukan permasalahan yang dihadapi serta semua kebutuhan seperti analisis masalah, analisis sistem, masukan dan keluaran sistem, antarmuka sistem, dan fungsi-fungsi yang dibutuhkan.

Metode analisis yang digunakan dalam mengembangkan perangkat lunak ini adalah metode analisis berorientasi objek dengan menggunakan standar dari VBGame yang digunakan sebagai *game engine*.

Aplikasi yang akan dibuat adalah suatu sistem permainan Multiplayer Role Playing Game berbasis *client-server* sebagai teknologi komunikasi dalam pengolahan data.

#### **3.2 Analisis Masalah**

Analisis sistem adalah prosedur yang dilakukan untuk membuat spesifikasi sistem *gameplay* yang akan dikembangkan. Analisis sistem dilakukan dengan tujuan untuk mengidentifikasi objek-objek yang berperan dalam sistem.

Setelah objek-objek tersebut telah diidentifikasi maka dapat dikembangkan dan dirinci lebih dalam hasil yang telah didapat dari segmen sebelumnya.

### 3.3 Hasil Analisis

#### 3.3.1 Kebutuhan Client

##### 3.3.1.1 Kebutuhan Masukan

Kebutuhan masukan dari *Client* dibutuhkan untuk beberapa keperluan yang berbeda, yaitu:

1. *Login* digunakan untuk membedakan dan membatasi akses *user* ke dalam sistem sesuai level aksesnya masing-masing. Fitur yang diberikan oleh sistem kepada *user* akan berbeda, sesuai dengan level akses masing-masing.

Kebutuhan masukan untuk *login* adalah :

- a. *Username*
  - b. *Password*
  - c. *Account Baru*
2. Kebutuhan masukan untuk *user* dengan hak akses level Admin atau Game Master :
    - a. Hak main seperti pemain lainnya  
*Game Master* dapat melakukan permainan seperti layaknya pemain lain *non-Game Master*.
    - b. Perintah-perintah Game Master

Untuk pengembangan permainan, *Game Master* dapat melakukan perintah untuk menaikkan level pemain lain *non-Game Master*, memanggil monster, membunuh monster dalam satu layar, memperingatkan atau memblokade alamat ip dan mengembalikan status blok..

3. Kebutuhan masukan dengan hak akses level Pemain :

a. Pembuatan tokoh

Data *input* pada pembuatan account baru berisi : nama, password, class.

b. Kontrol pergerakan

*Input* pada kontrol pergerakan dapat dirubah pemain melalui konfigurasi permainan, tombol menganut sistem *cartesian* dua dimensi.

c. Kontrol pertarungan

Sama halnya dengan kontrol pergerakan , kontrol ini dapat dirubah pemain melalui konfigurasi permainan.

d. Perintah-perintah dasar

Perintah yang digunakan untuk melakukan interaksi maupun menyatakan ekspresi.

e. Interaksi dengan Non Playing Character

Melakukan pembicaraan yang menjurus dan dapat melakukan quest maupun bertransaksi dengan *Non Playing Character* yang memiliki kecerdasan buatan.

f. Interaksi dengan Playing Character

Melakukan pembicaraan bebas dan membentuk kelompok maupun saling bertransaksi dengan *Playing Character* yang dimainkan pemain lainnya.

g. Interaksi dengan lingkungan

Pemain dapat mengetahui keberadaannya dan kondisi wilayah map setempat dengan menjelajah dan dengan perintah-perintah dasar.

4. *Log Out* digunakan untuk keluar dari permainan dan sekaligus melakukan permintaan ke server untuk menyimpan data terakhir dari tokoh .
5. Client Data adalah data yang disusun dan ditampilkan oleh client setelah menerima informasi dari server. Data ini berupa data grafis permainan dan data multimedia lainnya.

### **3.3.1.2 Kebutuhan Fungsi**

Aplikasi *game client* menggunakan dua fungsi utama, yaitu: fungsi-fungsi untuk mengirimkan masukan menjadi bit-bit permintaan pada server dan fungsi-fungsi untuk menerima bit-bit data dari server yang akan diterjemahkan menjadi informasi untuk ditampilkan ke user.

### **3.3.1.3 Keluaran**

Adapun keluaran yang dihasilkan berupa bit-bit permintaan pada server, dan keluaran *gameplay* yang merupakan informasi dari data yang dikirimkan server dari proses pemasukan database maupun dari proses permintaan client.

## **3.3.2 Kebutuhan Server**

### **3.3.2.1 Kebutuhan Masukan**

Pada Sistem berbasis *client-server* maka baik pada sisi server juga membutuhkan masukan untuk pengolahan proses. Kebutuhan masukan tersebut adalah:

1. Data Server berupa konfigurasi cara kerja server dan konfigurasi koneksi jaringan dan koneksi database. Data pada server ini menentukan bagaimana *gameplay* akan berjalan dengan aturan-aturan yang ada.

2. Permintaan client pada server berupa bit-bit yang diterjemahkan menjadi data yang akan diproses sesuai aturan dan permintaan. Data permintaan ini berasal dari beberapa client yang harus diproses bersamaan dan dikirimkan kembali secara *realtime*.
3. Database menjadi masukan yang diambil menurut kebutuhan server atas kebutuhan informasi lingkungan dan olah data permintaan user.

#### **3.3.2.2 Kebutuhan Fungsi**

Aplikasi server menggunakan tiga fungsi utama, yaitu: fungsi-fungsi untuk mengirimkan keluaran menjadi bit-bit informasi pada client, fungsi-fungsi untuk menerima bit-bit data dari client yang akan diterjemahkan dan diproses menurut fungsi penanganan *gameplay* yang ada. Sedangkan fungsi penanganan *gameplay* adalah proses olah data dan pengambilan nilai atau data dari database yang hasilnya akan dikirimkan oleh server pada client.

#### **3.3.2.3 Keluaran**

Keluaran sistem server ini mempunyai dua keluaran tujuan, yakni:

1. Keluaran menuju client berupa data yang telah diolah dan hasilnya dikirimkan menjadi bit-bit data yang akan diterima client.
2. Keluaran menuju database yang meng-update database sesuai dengan perkembangan hasil proses yang dilakukan olah data pada server dan sudah dikeluarkan oleh client menjadi informasi kepada user.

### 3.4 Antarmuka Sistem

Antarmuka sistem dibuat sedemikian rupa sehingga permainan menjadi menarik dan sesuai dengan standar kebutuhan permainan. Antarmuka permainan terdiri dari *Graphical User Interfaces* yang interaktif, lingkungan atau map yang lengkap dengan banyak faktor pendukung, grafis game yang sederhana dengan keragaman jenis dan fungsi item. Salah satu keunggulan permainan multiplayer Role Playing Game adalah penampilan dari masing-masing tokoh yang dapat dirubah dengan menggunakan *avatar* yang unik dan beragam sesuai dengan kemampuan ekonomi masing-masing.

1. Graphical User Interfaces:
  - a. Menu *login*, yang terdiri dari masukan nama, kata sandi, tombol Connect, tombol account baru dan tombol exit.
  - b. Menu *Create Account* Baru, yang terdiri dari masukan nama, kata sandi dan kreasi tokoh pada pemilihan class.
  - c. Layar permainan, yaitu tampilan tokoh dengan lingkungan disekitarnya.
  - d. Jendela informasi dan percakapan, jendela ini menampilkan informasi yang didapatkan melalui proses-proses *gameplay* baik dengan aksi interaksi, perintah-perintah maupun percakapan dan transaksi yang sudah dilakukan.
  - e. Jendela informasi inventory, pada jendela ini terdapat informasi barang-barang yang dibawa, tokoh hanya dapat memakai barang yang ada pada bawaan.

- f. Jendela Statistik, pada jendela ini terdapat informasi dasar tentang kemampuan tokoh . Informasi meliputi statistik (*strength, agility, dexterity, vitality, magic, luck*) yang dapat diupdate user jika memiliki bonus *points*, informasi *hitpoints, manapoints, gold*, bonus *points* dan statistik lain yang mempengaruhi kemampuan dasar tokoh .
- g. Jendela *Quickbar*, adalah jendela yang dapat diakses melalui tombol shortcut (F1 sampai dengan F9). Pada jendela ini dapat dimodifikasi user untuk diakses pada item atau skill yang dibutuhkan.
- h. Jendela transaksi NPC, pada jendela ini disinkronisasikan dengan jendela inventory user, difungsikan untuk transaksi dengan cara *drag and drop*.
- i. Jendela transaksi PC ke PC, jendela ini terbuka ketika terjadi transaksi antara Playing Character dengan Playing Character dan terbuka pada kedua belah pihak yang bertransaksi. Untuk kolom barang dapat diisikan melalui jendela inventory dengan cara *drag and drop* sedangkan pada kolom gold diisi dengan mengetikkan sejumlah gold yang tidak melebihi gold yang dipunyai. Tombol konfirmasi aktif setelah kedua belah pihak menyetujui jumlah dan menekan tombol *accept*.
- j. Kolom skill, berisi tentang semua *skill* yang ada dan informasi *skill* yang telah dipelajari atau belum dipelajari.
- k. Jendela Storage, jendela ini didapat dengan cara mengakses pada NPC yang ditunjuk untuk menyimpan barang bawaan yang sekiranya belum dibutuhkan dan tidak perlu dibawa-bawa.

- l. Jendela konfirmasi jumlah, jendela ini keluar setelah melakukan transaksi, atau akan menjatuhkan barang, atau akan menyimpan barang di gudang dengan mengisikan jumlah barang.
- m. Menu *logout*, tekan *log out* untuk keluar dari permainan.

### 3.5 Metode Perancangan

Metode perancangan yang digunakan untuk mengembangkan aplikasi Multiplayer Role Playing Game ini menggunakan metode perancangan pemrograman untuk reaksi kejadian (*Event Driven*) dengan menggunakan diagram keadaan (*State Diagram*).

State Diagram merupakan sistem penggambaran dan alur keadaan, objek yang berinteraksi, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.

Selain menggunakan State Diagram, pada perancangan sistem ini juga menggunakan relasi antar tabel, *database* terstruktur, dan juga perancangan *interface* untuk mempermudah dalam penggunaannya

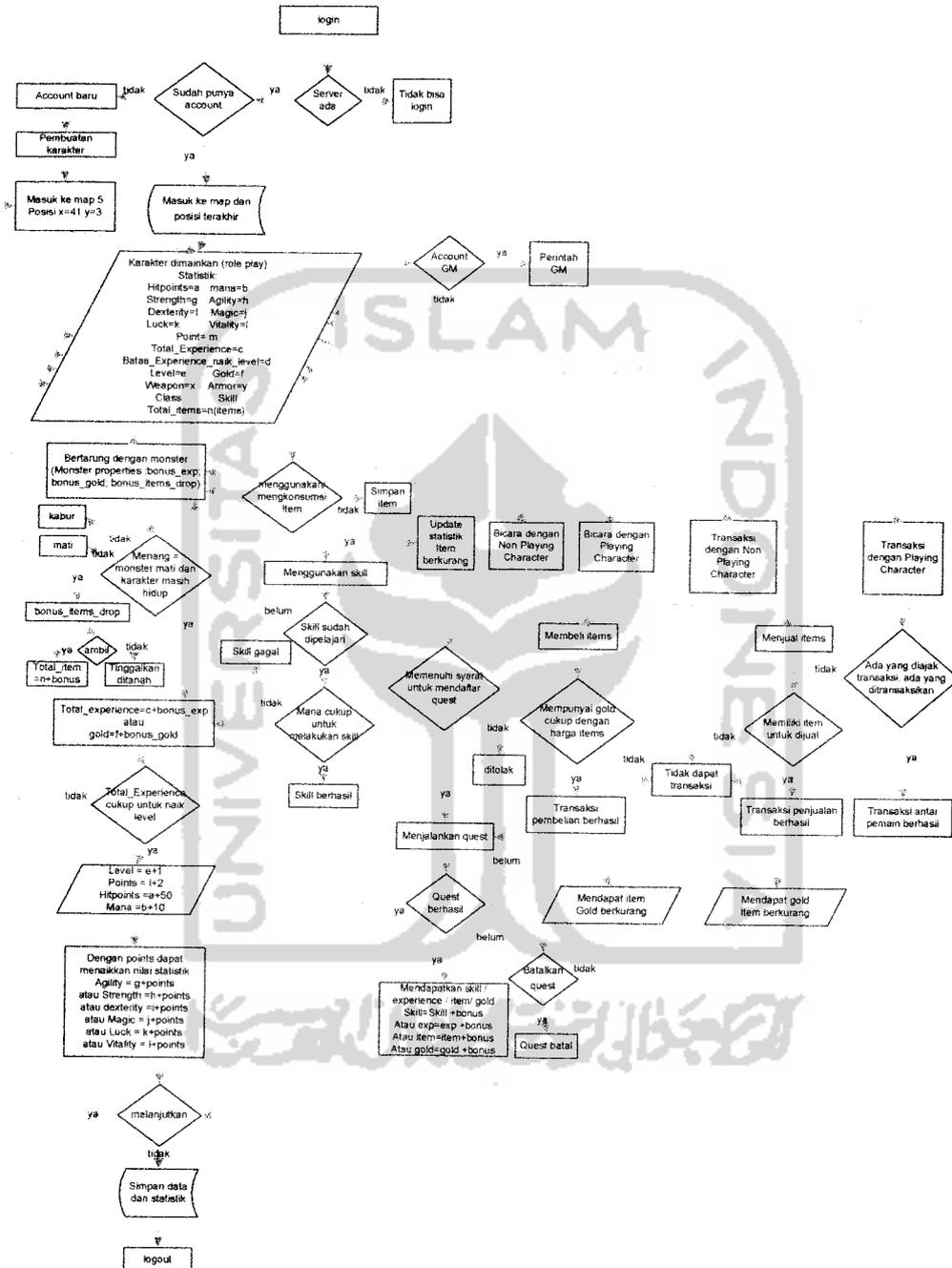
### 3.6 Hasil Perancangan

Hasil perancangan sistem ini dibedakan menjadi beberapa bagian, sesuai dengan tahapan yang digunakan pada metode perancangan, yaitu :

#### 3.6.1 Perancangan model proses dengan menggunakan State Diagram

State Diagram digunakan untuk menggambarkan sistem berawal dari login, proses-proses yang terjadi dan aksi-reaksi yang dimainkan, hingga logout, yang menjadi dalam satu kesatuan sistem berbasis *client-server*.

Seluruh keputusan diambil oleh user dengan keterikatan dan keterbatasan sistem, seperti digambarkan pada gambar 3.1



Gambar 3.1 State Diagram Aplikasi Multiplayer Role Playing Game Berbasis Client-Server

### Keterangan State Diagram :

#### 1. Keterangan gambar :

-  = Proses atau keadaan
-  = Fungsi percabangan
-  = Manipulasi data
-  = Pengambilan atau penyimpanan database
-  = Alur proses

#### 2. Proses Login

Pada proses login, user melakukan pengisian data berupa nama dan kata sandi yang nantinya akan di teruskan ke server untuk dimintai validasinya. Jika data tidak *valid* maka user diminta memperbaiki data atau menuju pada pembuatan akun baru.

#### 3. Proses pembuatan account baru

Pada pembuatan account baru user akan diminta untuk memasukkan nama user, dimana dalam hal ini nama user adalah juga nama tokoh , sedangkan pada isian kata sandi user akan memasukan kata sandi dua kali guna memverifikasi kata sandi tersebut. Pada proses ini user juga akan diminta untuk memilih jenis *class* dari tokoh yang akan dibuat.

#### 4. Proses memainkan tokoh

Pada proses permainan kontrol tokoh menjadi peranan penting agar tokoh dapat melakukan aksi-reaksi sesuai dengan input dari user. Dalam hal ini input terdiri dari dua alat yaitu *mouse* dan *keyboard*.

Sedangkan tokoh adalah kumpulan informasi data yang memiliki bentuk visual dan dimanipulasi hingga dihasilkan tampilan keluaran yang beraksi-interaksi dengan lingkungan game.

#### 5. Proses bertarung dengan monster

Dalam proses pertarungan dengan monster terjadi proses manipulasi antara dua data atau lebih (jika terjadi pengeroyokan), manipulasi ini berupa hitungan serangan yang mengakibatkan kerusakan dengan perhitungan dasar-dasar logika dan operator dasar aritmatika.

Monster memiliki statistik layaknya tokoh dan setiap jenis monster memiliki bonus *experience* yang berbeda-beda dan akan ditambahkan pada *experience* tokoh jika monster mati dan tokoh masih hidup. Dalam hal pertarungan ini jika tokoh mengalami kekalahan dan hampir mati maka user dapat memilih untuk kabur tanpa mendapat *experience* bonus.

Keadaan monster mati juga dapat mengakibatkan bonus item yang dijatuhkan oleh monster. Tokoh dapat mengambil item tersebut atau tetap meninggalkannya di tanah. Jika item diambil maka akan langsung masuk ke dalam barang bawaan (inventory). Item dapat digunakan jika memang dapat dikonsumsi atau dipakai, jika tidak maka bisa disimpan atau dijual.

#### 6. Proses penambahan experience

Proses penambahan *experience* ini terjadi jika monster mati dibunuh tokoh. *Experience* akan ditambahkan ke dalam total jumlah *experience*

tokoh yang telah dikumpulkan. Jika tokoh sudah memenuhi jumlah total *experience* untuk naik level maka tokoh berhak atas kenaikan level dengan sejumlah *point* yang dapat digunakan untuk memanipulasi statistik tokoh. Jika jumlah total belum mencukupi untuk kenaikan level maka user dapat terus menambahkan *experience* melalui pertarungan ataupun melalui *quest* dengan bonus *experience*.

#### 7. Proses manipulasi statistik

Proses manipulasi statistik ini dapat dilakukan ketika tokoh memiliki *point* untuk menaikkan nilai statistiknya. Untuk melakukan kenaikan nilai statistiknya dapat dilakukan pada jendela statistik.

#### 8. Proses menggunakan skill

Untuk menggunakan *skill*, tokoh harus memiliki pengetahuan *skill* yang akan digunakan, jika *skill* belum dapat dipelajari maka *skill* tidak dapat digunakan. Sedangkan kebutuhan *manapoints* untuk melakukan *skill* disesuaikan dengan *skill* yang akan digunakan, jika *manapoints* tidak mencukupi kebutuhan *skill* maka *skill* gagal dilakukan.

Ada dua macam jenis *skill* yang bisa digunakan tokoh yaitu *skill* offensif dan *skill* supportif. *Skill* offensif adalah *skill* yang digunakan pada monster untuk melakukan kerusakan pada monster, sedangkan *skill* supportif adalah *skill* yang digunakan untuk melindungi diri atau tokoh pemain lain atau untuk menaikkan status yang bersifat sementara pada tokoh atau tokoh pemain lain.

#### 9. Proses berbicara dengan NPC

NPC memiliki kemampuan berbicara dan memberikan informasi juga memberikan *quest* pada tokoh yang memenuhi syarat untuk melaksanakan *quest* yang diminta oleh NPC tersebut. NPC akan memberikan hadiah atas *quest* yang berhasil diselesaikan. Beberapa *quest* NPC tidak akan memberikan *quest* yang sama dua kali pada orang yang sama. NPC juga tidak mengizinkan tokoh untuk menjalankan beberapa *quest* secara bersamaan. Maksimum satu *quest* yang sedang berjalan untuk satu tokoh .

Hadiah akan diberikan kepada tokoh setelah *quest* selesai dan tokoh berbicara kembali dengan NPC pemberi *quest*. Hadiah bisa berupa sejumlah *gold*, *experience*, *item* atau *skill*.

#### 10. Proses berbicara dengan PC

Pembicaraan yang dimaksud adalah chatting dengan cara mengetikkan kata-kata yang dapat dilihat oleh tokoh lain yang terlihat pada layar. Kata-kata tersebut diketikkan dan ditampilkan dalam jendela informasi dan percakapan, selain itu juga muncul *chat bubble* yang keluar diatas kepala tokoh .

#### 11. Proses transaksi dengan NPC

Selain kemampuan untuk berbicara ada juga NPC yang memiliki kemampuan untuk berdagang. Hanya NPC yang ditunjuk yang dapat melakukan perdagangan jual dan beli item dengan PC.

## 12. Proses pembelian item

Sama halnya dengan perdagangan di dunia nyata, transaksi pembelian barang di dunia *game* juga memiliki syarat yang sama, yaitu ada dua orang yang melakukan transaksi dan ada barang yang ditukarkan. Sedangkan pada permainan ini dipersempit lagi dengan ada uang ada barang. PC yang memiliki jumlah gold yang sesuai dengan harga barang yang akan dibeli (harga NPC adalah *fixed*, tidak bisa ditawar) dapat membeli barang dari NPC. Barang itu akan langsung masuk kedalam barang bawaan (*inventory*).

## 13. Proses penjualan item

Sama halnya dengan proses pembelian item maka proses penjualan item adalah kebalikan dari pembelian. Dalam permainan ini barang yang dijual berharga dua kali harga belinya. Hal ini terjadi karena diperlukan dalam pembentukan ekonomi dunia *game* ini. Faktor lainnya yang mendukung fitur ini adalah NPC memiliki gold tidak terbatas. Sehingga sebanyak apapun PC menjual item, NPC tetap akan membayar sejumlah item dikali setengah harga belinya.

## 14. Proses transaksi dengan PC

Ketika terjadi transaksi antara *Playing Character* dengan *Playing Character* persetujuan kedua belah pihak yang bertransaksi, maka terjadilah transaksi antar PC dengan PC. Transaksi dapat terjadi sepihak jika diinginkan, sebagai contoh tokoh dapat bertransaksi tanpa menukarkan barangnya sedangkan tokoh lain menukarkan dengan

gold. Sifatnya lebih cenderung memberi, asalkan terjadi kesepakatan sebelumnya.

#### 15. Proses Logout

Proses ini mengeluarkan *user* dari sistem (dalam hal ini tokoh *offline*), dan menyimpan data terakhir tokoh pada *database* yang nantinya akan *diload* sewaktu-waktu *user* login kedalam sistem.

### 3.7 Perancangan Struktur Basis Data

Pembuatan aplikasi diawali dengan mengimplementasikan hasil rancangan sistem ke dalam tabel-tabel yang diintegrasikan menjadi sebuah *database*. Perangkat lunak yang digunakan adalah MySQL.

#### 1. Tabel banned\_ip

Tabel yang memuat daftar dari ip address yang diblok. Struktur tabel banned\_ip dapat dilihat pada tabel 3.1

**Tabel 3.1** Tabel banned\_ip

| Field  | Type         | Null | Key | Comment                  |
|--------|--------------|------|-----|--------------------------|
| ip     | varchar(255) | NO   | PRI | Alamat ip yang di blok   |
| reason | varchar(255) | NO   |     | Alasan kenapa ip di blok |

#### 2. Tabel mail

Tabel yang memuat informasi mail yang dikirimkan user-user. Dalam hal ini tabel mail adalah mailbox atau kotak surat itu sendiri. Struktur tabel mail dapat dilihat pada tabel 3.2

**Tabel 3.2** Tabel mail

| Field | Type    | Null | Key | Comment      |
|-------|---------|------|-----|--------------|
| id    | int(11) | NO   | PRI | ID dari mail |

|      |                        |    |  |  |
|------|------------------------|----|--|--|
| sub  | varchar(255)           | NO |  | Subyek mail  |
| by   | varchar(255)           | NO |  | Nama penulis   |
| date | date                   | NO |  | Tanggal diterima   |
| msg  | text                   | NO |  | Isi dari mail  |
| new  | tinyint(3)<br>unsigned | NO |  | Jika mail adalah baru (1 = ya, 0 = tidak)                  |
| objs | mediumtext             | NO |  | Item yang termasuk dalam mail (index dari item dan jumlah) |

### 3. Tabel mail\_lastid

Tabel ini memuat informasi id terakhir dari mail terakhir yang dikirimkan dalam permainan. Struktur tabel mail\_lastid dapat dilihat pada tabel 3.3

**Tabel 3.3** Tabel mail\_lastid

| Field  | Type    | Null | Key | Comment          |
|--------|---------|------|-----|------------------|
| lastid | int(11) | NO   | PRI | ID mail terakhir |

### 4. Tabel npcs

Tabel ini memuat data dari Non Playing Character, dalam hal ini NPC adalah memiliki Artificial Intelligence yang dapat memfungsikan NPC sebagai pelayanan, informan, pedagang maupun monster. Struktur tabel npcs dapat dilihat pada tabel 3.4

**Tabel 3.4** Tabel npcs

| Field       | Type                   | Null | Key | Comment  |
|-------------|------------------------|------|-----|--|
| id          | smallint(6)            | NO   | PRI | Identifier dari NPC                                    |
| name        | varchar(255)           | NO   |     | Nama   |
| deser       | varchar(255)           | NO   |     | Deskripsi  |
| ai          | tinyint(3)<br>unsigned | NO   |     | Algoritma AI yang digunakan                            |
| chat        | tinyint(3)<br>unsigned | NO   |     | Indeks dari NPC chat dari file NPC Chat                |
| respawnwait | int(11)                | NO   |     | Waktu yang digunakan untuk respawn (dalam miliseconds) |

|                           |                        |    |   |
|---------------------------|------------------------|----|---|
| attackable                | tinyint(3)<br>unsigned | NO | Jikalau NPC dapat diserang (1 = ya, 0 = tidak)  |
| attackgrh                 | int(11)                | NO | Grh yang digunakan ketika NPC menyerang   |
| attackrange               | tinyint(3)<br>unsigned | NO | Jikalau NPC dapat melakukan serangan jarak jauh (0 atau 1 untuk serangan hand-to-hand ) |
| attacksfx                 | tinyint(3)<br>unsigned | NO | Indeks suara serangan   |
| Projectile<br>rotatespeed | tinyint(3)<br>unsigned | NO | Jika serangan jarak jauh, seberapa cepat rotasi dari projectile                         |
| hostile                   | tinyint(3)<br>unsigned | NO | NPC menyerang dahulu(1 = ya, 0 = tidak)   |
| quest                     | smallint(6)            | NO | ID dari quest NPC   |
| drops                     | mediumtext             | NO | daftar dari NPC item drops  |
| give_exp                  | int(11)                | NO | Experience diberikan ketika membunuh NPC  |
| give_gold                 | int(11)                | NO | Gold diberikan ketika membunuh NPC  |
| objs_shop                 | mediumtext             | NO | Item yang dijual shopkeeper/vendor  |
| char_hair                 | smallint(6)            | NO | Paperdolling hair ID  |
| char_head                 | smallint(6)            | NO | Paperdolling head ID  |
| char_body                 | smallint(6)            | NO | Paperdolling body ID  |
| char_weapon               | smallint(6)            | NO | Paperdolling weapon ID  |
| char_wings                | smallint(6)            | NO | Paperdolling wings ID   |
| char_heading              | tinyint(3)<br>unsigned | NO | Arah body menghadap terakhir save   |
| char_head heading         | tinyint(3)<br>unsigned | NO | Arah head menghadap ketika terakhir save  |
| stat_hitrate              | int(11)                | NO | Kesempatan serangan mengenai sasaran  |
| stat_mag                  | int(11)                | NO | Magic   |
| stat_def                  | int(11)                | NO | Defence   |
| stat_speed                | int(11)                | NO | Kecepatan berjalan  |
| stat_hit_min              | int(11)                | NO | Minimum hit   |
| stat_hit_max              | int(11)                | NO | Maximum hit   |
| stat_hp                   | int(11)                | NO | Health points   |
| stat_mp                   | int(11)                | NO | Mana points   |
| stat_sp                   | int(11)                | NO | Stamina points  |

## 5. Tabel objects

Tabel ini memuat data dari obyek atau item-item yang eksis pada permainan. Setiap obyek memiliki kegunaan yang berbeda tiap id jenisnya sesuai dengan data yang ada dalam tabel. Struktur tabel objects dapat dilihat pada tabel 3.5

**Tabel 3.5** Tabel objects

| Field                     | Type                    | Null | Key | Comment  |
|---------------------------|-------------------------|------|-----|--|
| Id                        | smallint(6)             | NO   | PRI | Identifier dari barang   |
| name                      | varchar(255)            | NO   |     | Nama   |
| price                     | int(10)<br>unsigned     | NO   |     | Harga beli barang  |
| objtype                   | tinyint(3)<br>unsigned  | NO   |     | Tipe barang  |
| weapontype                | tinyint(3)<br>unsigned  | NO   |     | Tipe senjata (valid hanya jika objtype=weapon)   |
| weaponrange               | tinyint(3)<br>unsigned  | NO   |     | Jarak serangan dari senjata (jika tipe senjata berjarak jauh)                          |
| classreq                  | smallint(5)<br>unsigned | NO   |     | Hanya mengizinkan class tertentu untuk menggunakan item (0 untuk semuanya)             |
| grhindex                  | int(10)<br>unsigned     | NO   |     | Indeks dari grafis barang (dari nilai Grh)   |
| usegrh                    | int(10)<br>unsigned     | NO   |     | Grh dari serangan senjata  |
| usesfx                    | tinyint(3)<br>unsigned  | NO   |     | Efek suara dimainkan ketika obyek digunakan (0 untuk silent)                           |
| Projectile<br>rotatespeed | tinyint(3)<br>unsigned  | NO   |     | Jika serangan jarak jauh, seberapa cepat rotasi dari projectile (0 untuk tanpa rotasi) |
| stacking                  | smallint(6)             | NO   |     | Jumlah item yang bisa ditumpuk (< 1 untuk server default)                              |
| sprite_body               | smallint(6)             | NO   |     | Paperdolling body berubah ketika digunakan (-1 untuk tanpa perubahan)                  |
| sprite_<br>weapon         | smallint(6)             | NO   |     | Paperdolling senjata berubah ketika digunakan (-1 untuk tanpa perubahan)               |

|                      |                     |    |   |
|----------------------|---------------------|----|---|
| sprite_hair          | smallint(6)         | NO | Paperdolling rambut berubah ketika digunakan (-1 untuk tanpa perubahan) |
| sprite_head          | smallint(6)         | NO | Paperdolling head berubah ketika digunakan (-1 untuk tanpa perubahan)   |
| sprite_wings         | smallint(6)         | NO | Paperdolling wings berubah ketika digunakan (-1 untuk tanpa perubahan)  |
| replenish_hp         | int(11)             | NO | Jumlah dari HP ditambahkan ketika digunakan                             |
| replenish_mp         | int(11)             | NO | Jumlah dari MP ditambahkan ketika digunakan                             |
| replenish_sp         | int(11)             | NO | Jumlah dari SP ditambahkan ketika digunakan                             |
| replenish_hp_percent | int(11)             | NO | Jumlah dari HP ditambahkan ketika digunakan                             |
| replenish_mp_percent | int(11)             | NO | Jumlah dari MP ditambahkan ketika digunakan                             |
| replenish_sp_percent | int(11)             | NO | Jumlah dari SP ditambahkan ketika digunakan                             |
| stat_str             | int(10)<br>unsigned | NO | Strength dinaikkan ketika digunakan                                     |
| stat_agi             | int(10)<br>unsigned | NO | Agility dinaikkan ketika digunakan                                      |
| stat_dex             | int(10)<br>unsigned | NO | Dexterity dinaikkan ketika digunakan                                    |
| stat_vit             | int(10)<br>unsigned | NO | Vitality dinaikkan ketika digunakan                                     |
| stat_luk             | int(10)<br>unsigned | NO | Luck dinaikkan ketika digunakan   |
| stat_mag             | int(10)<br>unsigned | NO | Magic dinaikkan ketika digunakan  |
| stat_def             | int(10)<br>unsigned | NO | Defence dinaikkan ketika digunakan                                      |
| stat_speed           | int(10)<br>unsigned | NO | Walk speed dinaikkan ketika digunakan                                   |
| stat_hit_min         | int(10)<br>unsigned | NO | Minimum hit dinaikkan ketika digunakan                                  |
| stat_hit_max         | int(10)<br>unsigned | NO | Maximum hit dinaikkan ketika digunakan                                  |
| stat_hp              | int(10)<br>unsigned | NO | Health dinaikkan ketika digunakan                                       |
| stat_mp              | int(10)<br>unsigned | NO | Mana dinaikkan ketika digunakan   |

|         |                     |    |  |  |
|---------|---------------------|----|--|--|
| stat_sp | int(10)<br>unsigned | NO |  | Stamina dinaikkan ketika digunakan                         |
| req_str | int(10)<br>unsigned | NO |  | Nilai minimum strength dibutuhkan ketika menggunakan item  |
| req_agi | int(10)<br>unsigned | NO |  | Nilai minimum agility dibutuhkan ketika menggunakan item   |
| req_dex | int(10)<br>unsigned | NO |  | Nilai minimum dexterity dibutuhkan ketika menggunakan item |
| req_vit | int(10)<br>unsigned | NO |  | Nilai minimum vitality dibutuhkan ketika menggunakan item  |
| req_luk | int(10)<br>unsigned | NO |  | Nilai minimum luck dibutuhkan ketika menggunakan item      |
| req_mag | int(10)<br>unsigned | NO |  | Nilai minimum magic dibutuhkan ketika menggunakan item     |
| req_lvl | int(10)<br>unsigned | NO |  | Nilai minimum level dibutuhkan ketika menggunakan item     |

#### 6. Tabel quests

Tabel ini memuat data quest-quest yang ada dalam permainan. Struktur tabel quests dapat dilihat pada tabel 3.6

**Tabel 3.6** Tabel quests

| Field           | Type                   | Null | Key | Comment   |
|-----------------|------------------------|------|-----|---|
| id              | smallint(6)            | NO   | PRI | Identifier dari quest   |
| name            | varchar(255)           | NO   |     | Nama  |
| redoable        | tinyint(3)<br>unsigned | NO   |     | Jika quest dapat dilakukan lagi (1 = ya, 0 = tidak)                 |
| text_start      | varchar(255)           | NO   |     | Text dikatakan saat memulai quest                                   |
| text_accept     | varchar(255)           | NO   |     | Text dikatakan saat menerima quest                                  |
| text_incomplete | varchar(255)           | NO   |     | Text dikatakan ketika sedang melakukan quest (reqs. belum dipenuhi) |
| text_finish     | varchar(255)           | NO   |     | Text dikatakan ketika menyelesaikan quest                           |

|                           |                         |    |   |
|---------------------------|-------------------------|----|---|
|                           |                         |    | (requirements dipenuhi)   |
| text_info                 | text                    | NO | Seluruh info quest yang ditampilkan pada client                                     |
| accept_req_class          | Int(10)<br>unsigned     | NO | Class yang dapat menerima quest ini   |
| accept_req_level          | int(10)<br>unsigned     | NO | kebutuhan minimum level untuk menerima quest  |
| accept_req_obj            | smallint(5)<br>unsigned | NO | kebutuhan objek untuk menerima quest (object ID)                                    |
| accept_req_objamount      | smallint(5)<br>unsigned | NO | kebutuhan jumlah objek untuk menerima quest (jika $\text{accept\_req\_obj} > 0$ )   |
| accept_reward_exp         | int(10)<br>unsigned     | NO | Experience bonus ketika menerima quest  |
| accept_reward_gold        | int(10)<br>unsigned     | NO | Gold bonus ketika menerima quest  |
| accept_reward_obj         | smallint(5)<br>unsigned | NO | Object bonus ketika menerima quest  |
| accept_reward_objamount   | smallint(5)<br>unsigned | NO | Jumlah Object bonus ketika menerima quest ( $\text{accept\_reward\_obj} > 0$ )      |
| accept_reward_learn_skill | tinyint(4)              | NO | Skill dipelajari ketika menerima quest  |
| finish_req_obj            | smallint(5)<br>unsigned | NO | Kebutuhan object untuk selesai quest (object ID)                                    |
| finish_req_objamount      | smallint(5)<br>unsigned | NO | Kebutuhan jumlah object untuk selesai ( $\text{finish\_req\_obj} > 0$ )             |
| finish_req_killnpc        | smallint(5)<br>unsigned | NO | Indeks dari NPC untuk dibunuh untuk memenuhi kebutuhan selesai                      |
| finish_req_kill_npcamount | smallint(5)<br>unsigned | NO | Jumlah dari NPCs untuk dibunuh ( $\text{killnpc} > 0$ ) untuk menyelesaikan quest   |
| finish_reward_exp         | int(10)<br>unsigned     | NO | Experience bonus ketika menyelesaikan quest   |
| finish_reward_gold        | int(10)<br>unsigned     | NO | Gold bonus ketika menyelesaikan quest   |
| finish_reward_obj         | smallint(5)<br>unsigned | NO | Object bonus ketika menyelesaikan quest   |
| finish_reward_objamount   | smallint(5)<br>unsigned | NO | Jumlah object bonus ketika menyelesaikan quest ( $\text{finish\_reward\_obj} > 0$ ) |

|                          |                        |    |  |   |
|--------------------------|------------------------|----|--|---|
| finish_reward_learnskill | tinyint(3)<br>unsigned | NO |  | Skill dipelajari ketika menyelesaikan quest |
|--------------------------|------------------------|----|--|---|

### 7. Tabel users

Tabel ini memuat data user (dalam hal ini tokoh yang dimainkan user). Data statistik, item, titik terakhir keberadaan, quest, skill dan lain-lain. Struktur tabel users dapat dilihat pada tabel 3.7

**Tabel 3.7** Tabel users

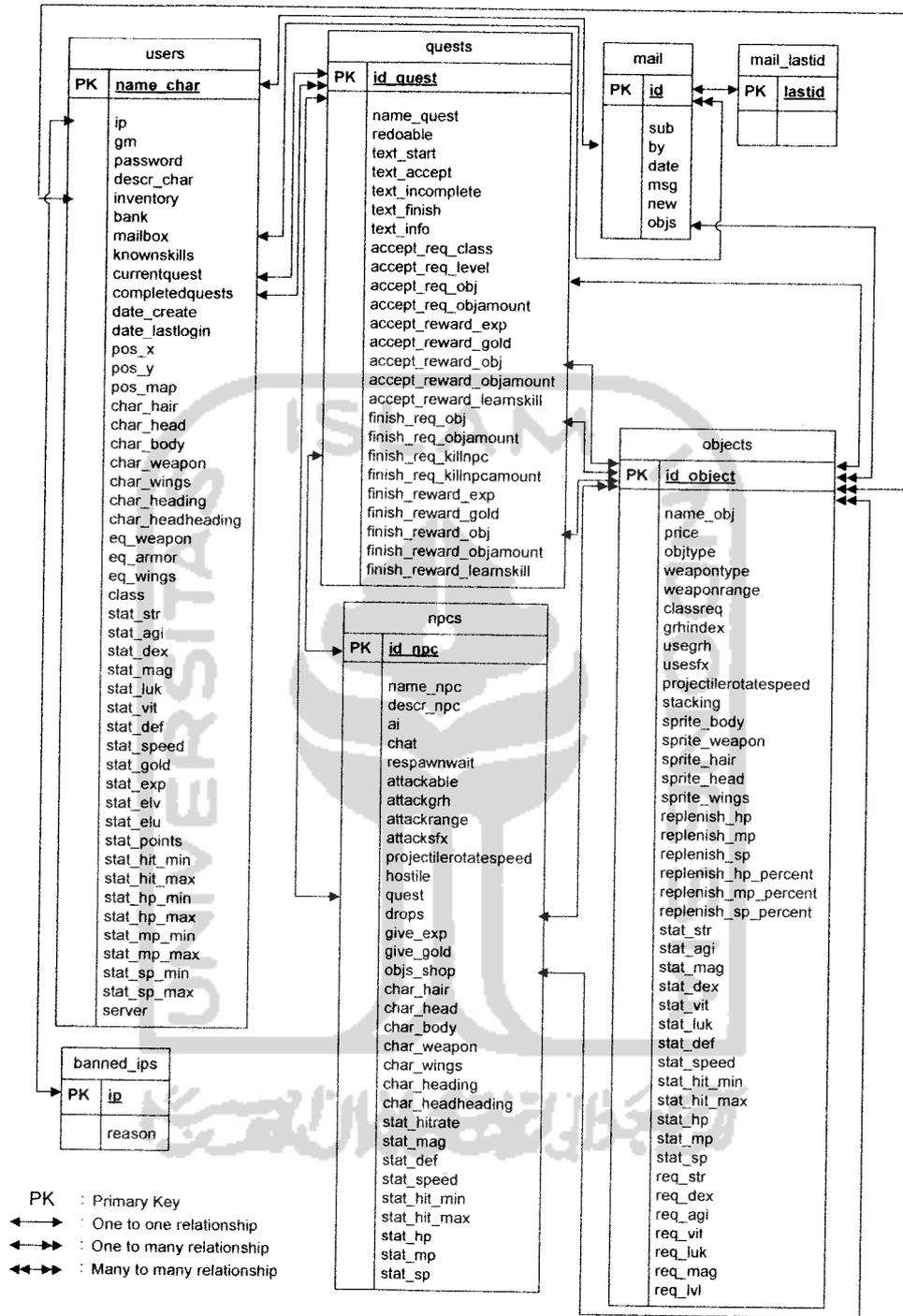
| Field           | Type                   | Null | Key | Comment   |
|-----------------|------------------------|------|-----|---|
| name            | varchar(255)           | NO   | PRI | Nama  |
| ip              | varchar(255)           | NO   |     | Menyimpan 10 ip address terakhir yang mengkoneksikan user dengan account bersangkutan |
| gm              | tinyint(3)<br>unsigned | NO   |     | Default user =0, untuk GM adalah 1  |
| password        | varchar(255)           | NO   |     | Password  |
| descr           | varchar(255)           | NO   |     | Deskripsi   |
| inventory       | text                   | NO   |     | Data Object ID dan jumlahnya yang berada inventory user                               |
| bank            | text                   | NO   |     | Daftar dari item user yang disimpan pada storage                                      |
| mail            | text                   | NO   |     | Mail ID dari mail yang tersimpan pada mailbox   |
| knownskills     | text                   | NO   |     | Skill yang diketahui user (1 = diketahui, 0 = belum)                                  |
| completedquests | text                   | NO   |     | Data quest yang telah diselesaikan (tidak direkomendasikan untuk diedit)              |
| currentquest    | text                   | NO   |     | Quest yang sedang dijalani user (tidak dapat diedit)                                  |
| date_create     | date                   | NO   |     | Tanggal account dibuat  |
| date_lastlogin  | date                   | NO   |     | Tanggal terakhir kali user Login  |
| pos_x           | tinyint(3)<br>unsigned | NO   |     | Posisi X  |
| pos_y           | tinyint(3)<br>unsigned | NO   |     | Posisi Y  |

|                  |                         |    |   |
|------------------|-------------------------|----|---|
| pos_map          | smallint(6)             | NO | Posisi Map  |
| char_hair        | smallint(6)             | NO | Paperdolling hair                                   |
| char_head        | smallint(6)             | NO | Paperdolling kepala                                 |
| char_body        | smallint(6)             | NO | Paperdolling badan                                  |
| char_weapon      | smallint(6)             | NO | Paperdolling senjata                                |
| char_wings       | smallint(6)             | NO | Paperdolling wings                                  |
| char_heading     | tinyint(3)<br>unsigned  | NO | Arah body menghadap terakhir save                   |
| char_headheading | tinyint(3)<br>unsigned  | NO | Arah head menghadap ketika terakhir save            |
| eq_weapon        | tinyint(3)<br>unsigned  | NO | Slot senjata yang dipakai                           |
| eq_armor         | tinyint(3)<br>unsigned  | NO | Slot armor yang dipakai                             |
| eq_wings         | tinyint(3)<br>unsigned  | NO | Slot wings yang dipakai                             |
| class            | smallint(5)<br>unsigned | NO | class dari tokoh berdasarkan ClassID                |
| stat_str         | int(10)<br>unsigned     | NO | Kekuatan fisik                                      |
| stat_agi         | int(10)<br>unsigned     | NO | Kelincahan  |
| stat_dex         | int(10)<br>unsigned     | NO | Ketepatan serangan                                  |
| stat_mag         | int(10)<br>unsigned     | NO | Kekuatan sihir                                      |
| stat_vit         | int(10)<br>unsigned     | NO | Ketahanan/kekuatan fisik                            |
| stat_luk         | int(10)<br>unsigned     | NO | Keberuntungan (nilai tambah dalam perhitungan acak) |
| stat_def         | int(10)<br>unsigned     | NO | Pertahanan  |
| stat_speed       | int(10)<br>unsigned     | NO | Kecepatan berjalan                                  |
| stat_gold        | int(10)<br>unsigned     | NO | Uang yang dimiliki                                  |
| stat_exp         | int(10)<br>unsigned     | NO | Nilai pengalaman                                    |
| stat_elv         | int(10)<br>unsigned     | NO | Level   |
| stat_elu         | int(10)<br>unsigned     | NO | Experience dibutuhkan untuk naik level              |
| stat_points      | int(10)<br>unsigned     | NO | Points yang ada dalam antrian dan belum digunakan   |

|              |                        |    |  |  |
|--------------|------------------------|----|--|--|
| stat_hit_min | int(10)<br>unsigned    | NO |  | Kerusakan serangan minimum                                     |
| stat_hit_max | int(10)<br>unsigned    | NO |  | Kerusakan serangan maximum                                     |
| stat_hp_min  | int(10)<br>unsigned    | NO |  | Health/hitpoints sebenarnya                                    |
| stat_hp_max  | int(10)<br>unsigned    | NO |  | Health/hitpoints penuh   |
| stat_mp_min  | int(10)<br>unsigned    | NO |  | Manapoints sebenarnya  |
| stat_mp_max  | int(10)<br>unsigned    | NO |  | Manapoints penuh   |
| stat_sp_min  | int(10)<br>unsigned    | NO |  | Stamina sebenarnya   |
| stat_sp_max  | int(10)<br>unsigned    | NO |  | Stamina penuh  |
| server       | tinyint(4)<br>unsigned | NO |  | Mengindikasikan dalam server<br>mana user berada (0 = offline) |

### 3.8 Relasi Antar Tabel

Basis data merupakan tabel-tabel yang saling terkait. Pada model data relasional, hubungan antar tabel direlasikan dengan kunci relasi (*relation key*) yang merupakan kunci utama dari masing-masing tabel. Perencanaan *database* yang tepat akan membuat paket program relasional akan bekerja dengan optimal. Relasi antar tabel pada sistem ini dapat dilihat pada gambar 3.2



Gambar 3.2 Relasi antar tabel Aplikasi Multiplayer Role Playing Game berbasis client-server

### 3.9 Perancangan Antar Muka (*interface*)

Perancangan *interface* menggambarkan desain tampilan dari sistem, ilustrasi dari rancangan *interface* terhadap sistem yang akan diaplikasikan. Rancangan dan karakteristik dari aplikasi client yang akan dibuat terdiri dari beberapa jendela dengan fasilitas yang dapat diberikan kepada pengguna sistem, yaitu :

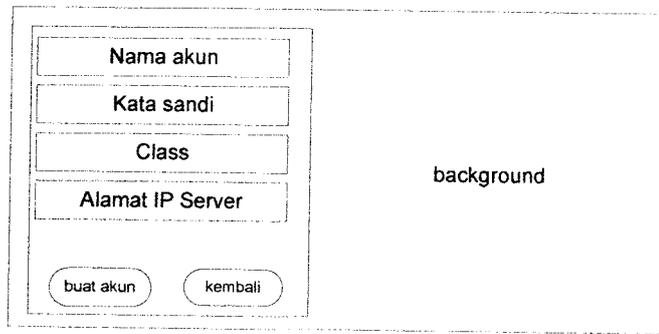
#### 1. Jendela Login

Menu ini digunakan untuk login akun yang telah terdaftar pada database, baik *end user* maupun game administrator. Pada menu ini terdapat pilihan isian alamat ip server yang dituju. Seperti yang akan ditampilkan pada gambar 3.3.

**Gambar 3.3** Rancangan Menu Login

#### 2. Jendela Pembuatan Akun Baru

Menu ini dapat diakses melalui menu login untuk pembuatan akun yang belum terdaftar pada *database*. Pada menu ini terdapat pilihan untuk kembali ke menu login. Seperti pada gambar 3.4.

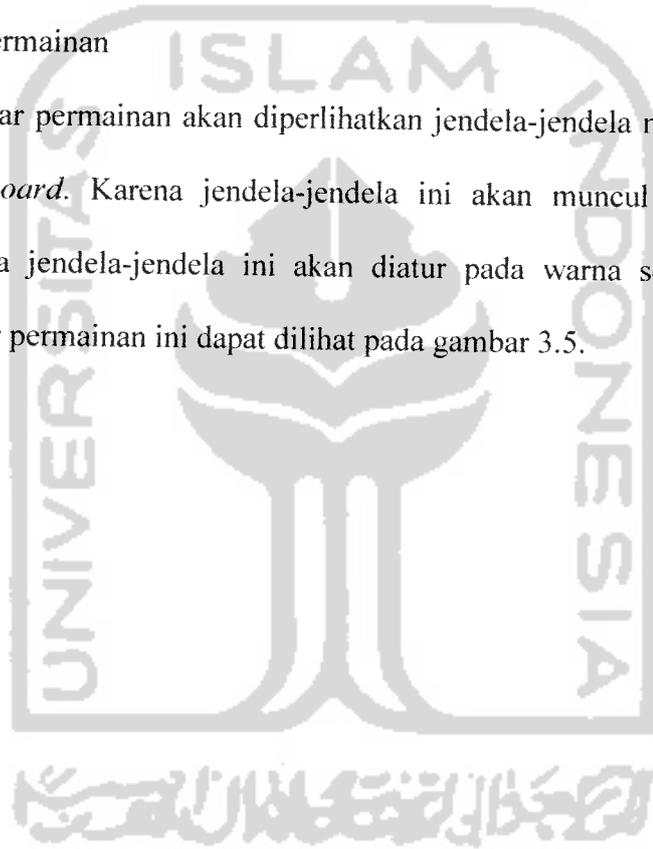


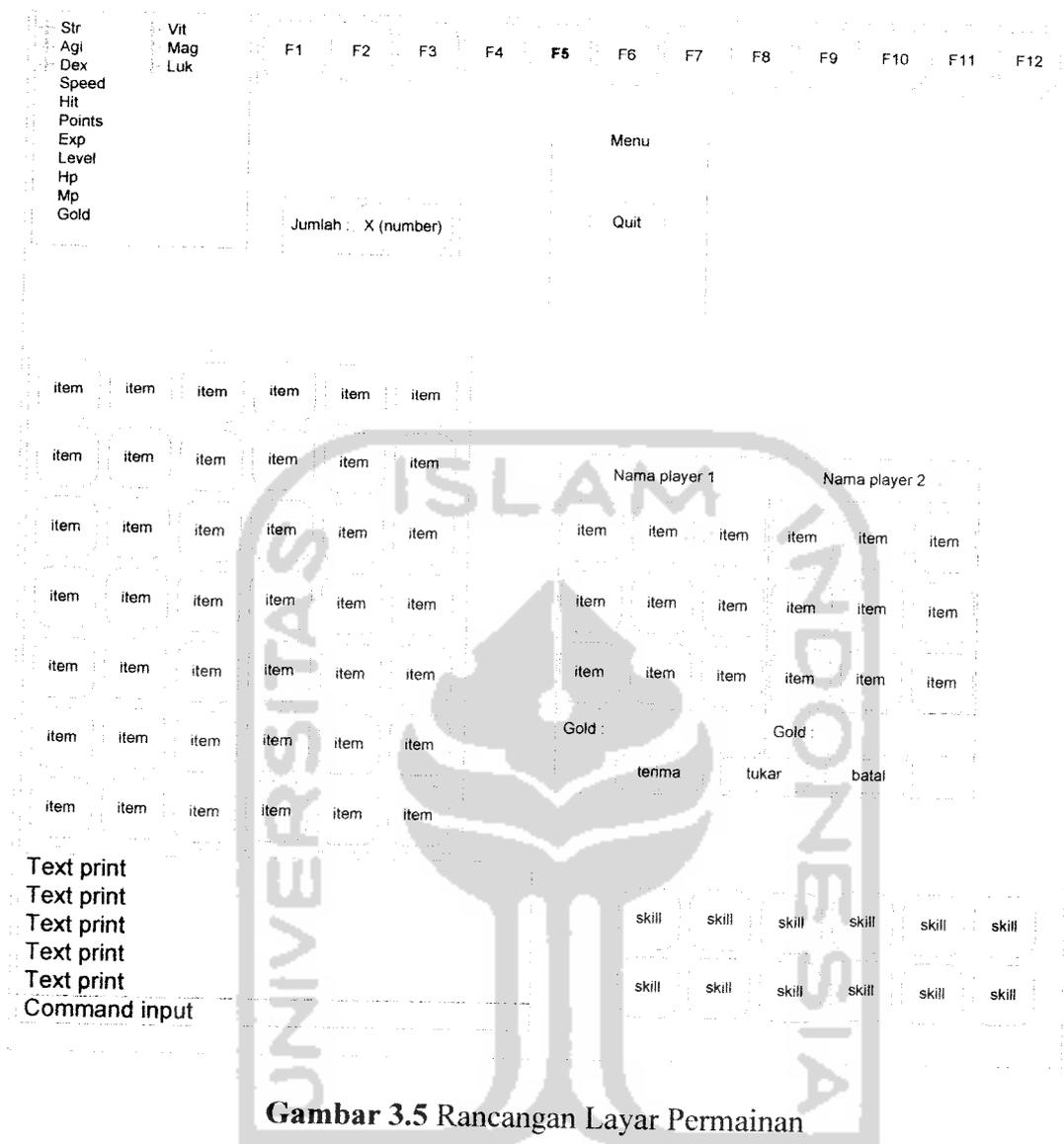
|                  |
|------------------|
| Nama akun        |
| Kata sandi       |
| Class            |
| Alamat IP Server |
| buat akun        |
| kembali          |

**Gambar 3.4** Rancangan Menu Pembuatan Akun Baru

### 3. Layar Permainan

Pada layar permainan akan diperlihatkan jendela-jendela melalui perintah kombinasi *keyboard*. Karena jendela-jendela ini akan muncul didepan layar permainan maka jendela-jendela ini akan diatur pada warna semi-transparan. Rancangan layar permainan ini dapat dilihat pada gambar 3.5.

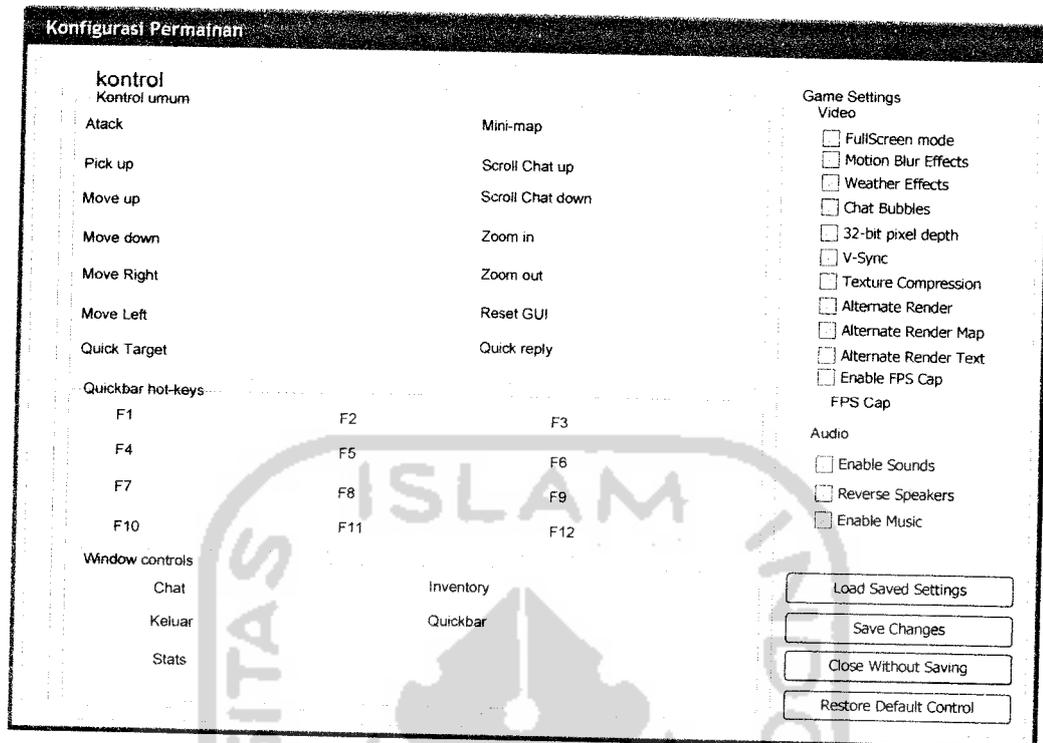




**Gambar 3.5** Rancangan Layar Permainan

#### 4. Konfigurasi Permainan

Pada konfigurasi permainan akan diperlihatkan pilihan-pilihan untuk pengaturan tombol kontrol pada permainan, pengaturan tampilan *video* pada permainan dan pengaturan *audio* pada permainan. Rancangan konfigurasi permainan dapat dilihat pada gambar 3.6.



Gambar 3.6 Rancangan Konfigurasi Permainan

## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 Implementasi Secara Umum

Implementasi sistem merupakan tahap dimana sistem mampu diaplikasikan dalam keadaan yang sesungguhnya. Dari implementasi ini akan diketahui apakah sistem yang dibuat dapat berjalan dengan baik atau tidak dan menghasilkan *output* yang sesuai dengan perancangan yang ada.

#### 4.2 Perangkat Lunak Yang Dibutuhkan

Perangkat lunak penunjang yang dibutuhkan untuk pengembangan implementasi perangkat lunak Aplikasi *Multiplayer Role Playing Game* Berbasis *Client-Server* adalah:

1. Sistem operasi Windows XP
2. Microsoft Visual Basic 6.0
3. Microsoft .NET Framework 2.0
4. Database MySQL versi 5.0.27
5. MySQL Connector /ODBC 5.31.12
6. SQLyog Community 5.21
7. VB Gore Open Source Online Role Playing Game Maker versi 1.0.12
8. VBruntime60
9. Adobe Photoshop CS2

10. DirectX 8.0 atau lebih tinggi

#### **4.3 Perangkat Keras Yang Dibutuhkan**

Perangkat Keras penunjang yang dibutuhkan untuk pengembangan implementasi perangkat lunak Aplikasi *Multiplayer Role Playing Game* Berbasis *Client-Server*, adalah:

1. 2 buah komputer dengan spesifikasi seperti disebut dibawah
2. Prosesor minimal Pentium 500 MHz atau yang lebih tinggi
3. RAM minimal 128 MB
4. VGA 16 MB dengan dukungan Direct3D 8
5. *Harddisk* dengan kapasitas 2 GB atau lebih
6. *Mouse*
7. *Keyboard*
8. Network Interface Card
9. UTP CAT 5 dengan konektor RJ 45 tersusun crossover terpasang pada NIC

#### **4.4 Tahapan Pembuatan Perangkat Lunak**

Dalam pembuatan perangkat lunak ini terdapat beberapa tahapan yang dapat dibedakan menjadi tiga tahap, yaitu:

1. Tahap Pembuatan Gambar dan Pemrograman Animasi

Citra yang memenuhi persyaratan seperti yang disebutkan kemudian dimasukkan kedalam *folder* \Grh dan diberi nama sesuai urutan nomor yang belum ada.

Untuk dapat diproses dalam pembuatan permainan, citra tersebut harus dipilah-pilah perbagian dengan ukuran 16x16, 32x32, 64x64 atau 128x128 *pixels*. Lebih dari ukuran tersebut akan mengakibatkan beban yang berlebihan pada kartu grafis komputer.

Data tersebut dimasukkan pada file *\Data2\GrhRaw.txt* dengan aturan penomoran yang unik tidak boleh ada yang sama. Aturan penomoran grh adalah sebagai berikut :

1. Satu frame

Grh# = Frames(=1) – Nomor Citra – AwalX – AwalY – lebar – tinggi

Penomoran diisikan pada lambang #. Untuk data hanya dengan satu *frame* (bukan animasi), nilai *frames* akan selalu satu. Nomor Citra diisi dengan nama file citra dimana frame berada. StartX dan StartY mewakili posisi pixel awal *frame* pada citra. Lebar dan tinggi merupakan besarnya *frame* yang dimaksudkan dalam satuan *pixel*. Untuk penulisan tidak menggunakan spasi.

2. Multi frame

Grh# = Frames(>1) - #-#-#-#-#... – kecepatan animasi

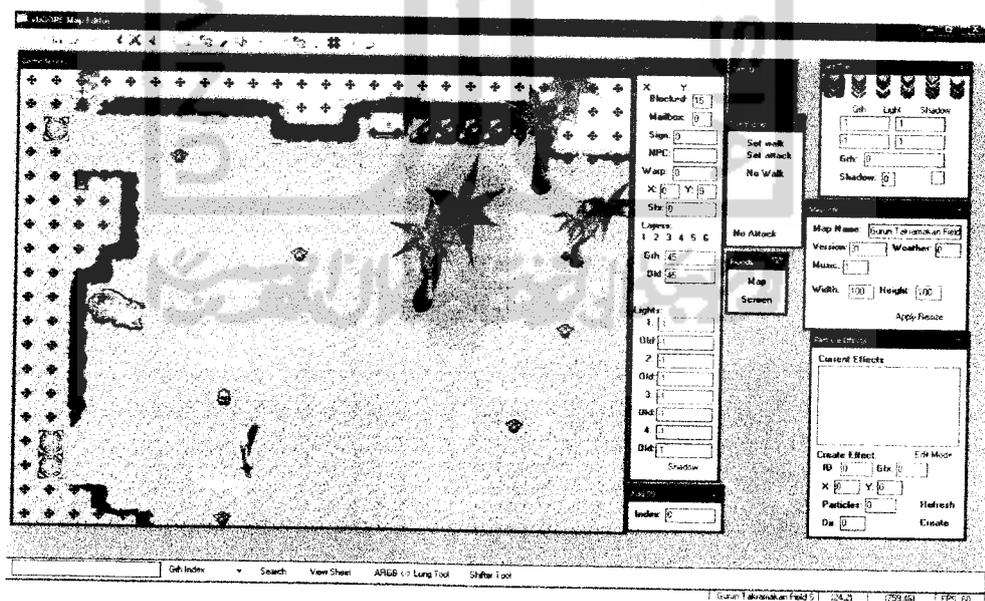
Penomoran diisikan pada lambang #. Sedangkan nilai *frames* sesuai dengan banyaknya *frame* yang akan digunakan pada animasi. Nilai #

selanjutnya adalah urutan nomor *frame* yang akan digunakan pada animasi. Kecepatan animasi bernilai banyaknya pergantian *frame* dalam satu detik.

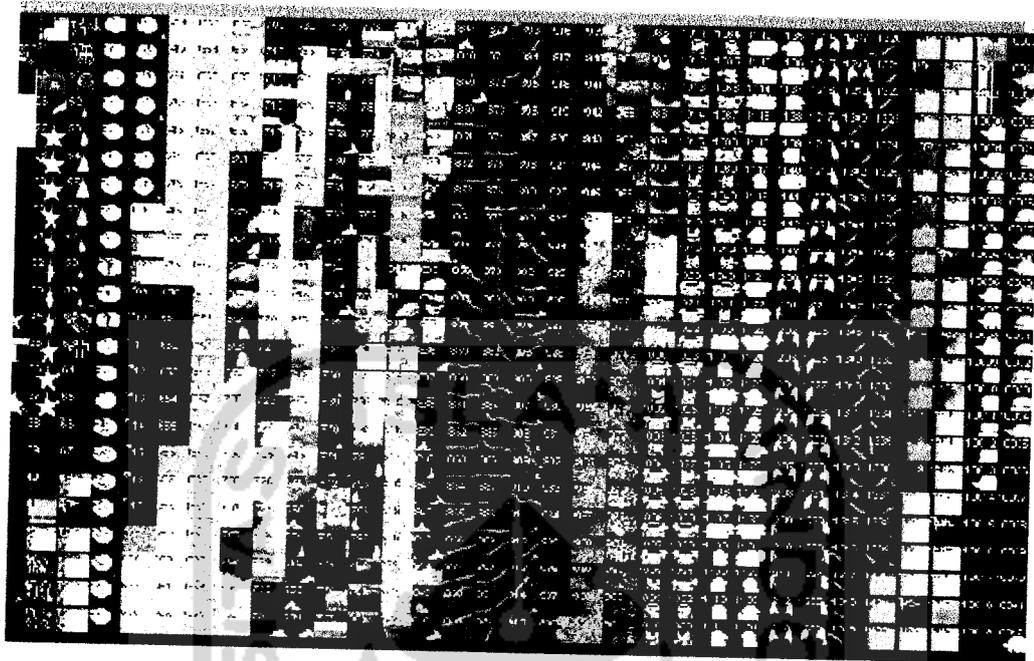
Data yang telah tersimpan belum dapat digunakan dalam proses pembuatan permainan. Harus dilakukan proses penyusunan data dengan menggunakan aplikasi GrhDatMaker.exe yang akan menyusun file GrhRaw.txt menjadi Grh.dat dan Grh.ini. Proses penambahan citra selesai dan siap digunakan pada aplikasi map editor.

#### 4.5.1.2 Aplikasi Map Editor

Aplikasi pada vbGore ini adalah aplikasi untuk pembuatan dunia pada permainan. Aplikasi ini menggunakan data yang disusun pada proses penambahan citra dan dari basis data permainan. Gambar berikut adalah tampilan *map editor*.



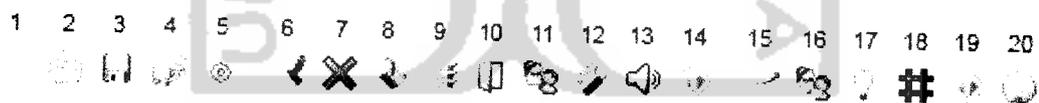
Gambar 4.1 Aplikasi Map Editor pada vbGore



Gambar 4.4 Jendela View Sheet

## 2. Menu Map Editor

Pada bagian atas *map editor* terdapat menu yang diwakilkan tombol-tombol sebagaimana diperlihatkan pada gambar 4.5.



Gambar 4.5 Menu Map Editor

Berikut adalah penjelasan tentang kegunaan menu-menu dan cara penggunaannya sesuai urutan nomor yang terlihat pada gambar 4.5 :

### a. New Map

Tombol ini digunakan untuk membuat sebuah *map* baru. Akan tampil *map* baru yang masih kosong.

b. Load Map

Menampilkan *map* yang telah disimpan sebelumnya. *Map* yang telah disimpan dapat dirubah dan disimpan lagi untuk digunakan selanjutnya.

c. Save

Tombol *save* digunakan untuk menyimpan *map* yang dibuat ataupun dirubah sebelumnya.

d. Save As

Tombol *save as* digunakan untuk menyimpan *map* dengan nama lain. *Map* yang tersimpan sebelumnya tidak akan berubah jika terjadi perubahan.

e. Optimize

Untuk mengoptimalkan kerja *map* pada jaringan dapat dilakukan optimasi dengan tombol *optimize*. *Map* yang telah dioptimalkan akan lebih cepat pengambilan datanya dan ukuran filenya diperkecil.

f. Set Tiles

*Set tiles* digunakan untuk menempatkan tekstur kedalam *map*.

Gambar 4.6 adalah jendela *set tiles* dengan nomor-nomor bagian untuk keterangan berikut :

### 1) Layer Section

Aplikasi vbGore menggunakan enam layer tekstur. Layer satu sampai dengan tiga adalah layer dibawah karakter. Sedangkan layer empat sampai enam ditampilkan diatas karakter.

### 2) Grh Set Checkbox

*Checkbox* ini untuk mengaktifkan penempatan tekstur. Klik kiri *mouse* untuk menempatkan tekstur dan untuk menghapus tekstur dengan menekan tombol ctrl dan klik kiri *mouse*.

### 3) Light Set Checkbox

*Checkbox* ini untuk mengaktifkan pencahayaan pada suatu tekstur.

### 4) Light Values

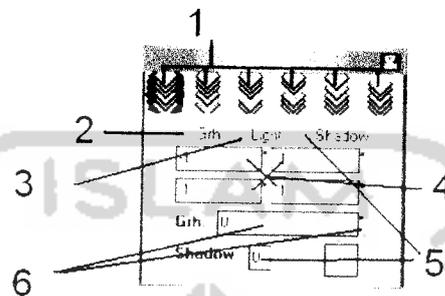
Nilai pencahayaan pada suatu tekstur dapat diatur disini. Terdapat empat kolom nilai pencahayaan. Setiap kolom mewakili setiap sudut pada tekstur.

### 5) Shadows

*Checkbox* ini untuk memberikan bayangan pada tekstur yang dimaksud. Untuk memberikan bayangan pada tekstur dengan memberi nilai satu pada kolom *shadow*, sedangkan untuk menghilangkan bayangan dengan memberi nilai nol pada kolom *shadow*.

## 6) Grh number

Kolom ini berisi nomor tekstur yang akan dipakai pada *map editor*. Nilai nol akan menghapus tekstur dari *map*.



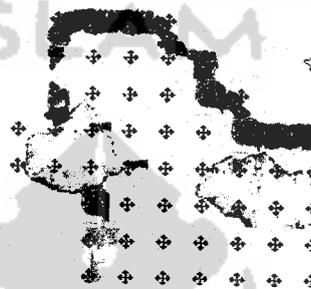
Gambar 4.6 Set Tile

## g. Blocks

Tombol *blocks* digunakan untuk memblokade tile dari gerakan maupun serangan jarak jauh. Jendela *blocks* akan terlihat ketika tombol ditekan, seperti terlihat pada gambar 4.7. Karakter dan NPC tidak akan melewati *tile* yang diblokade. Untuk memblokade gerakan dengan mengaktifkan *set walk checkbox* dan memilih arah blokade pada *no walk checkbox*. Untuk menghilangkan blokade gerakan dengan mengkosongkan arah pada *no walk checkbox*. *Tile* yang telah diblokade akan terlihat memiliki tanda panah merah sesuai dengan arah yang diblokade. Sama halnya dengan blokade gerakan, blokade serangan jarak jauh menandai *tile* dengan tanda silang berwarna hijau. Tanda blokade dapat dilihat pada gambar 4.8.



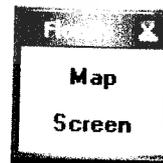
Gambar 4.7 Tile Blocks



Gambar 4.8 Blokade gerakan dan blokade serangan jarak jauh pada tile

h. Map Floods

Tombol ini berguna untuk memenuhi *map* dengan fitur-fitur lain yang tersedia. Penggunaan tombol ini digabungkan dengan tombol fitur lainnya yang sebelumnya telah ditekan. Terdapat dua pilihan untuk memenuhi layar ataupun untuk memenuhi seluruh *map*, seperti terlihat pada gambar 4.9.



Gambar 4.9 Map Floods

i. Tile Information

Tombol ini akan menampilkan gambar seperti terlihat pada gambar 4.10. Informasi tentang *tile* dapat dilihat dengan klik kanan pada *tile* yang dimaksud. Perubahan seketika dapat dilakukan dengan memasukkan nilai yang diinginkan pada kolom informasi yang dimaksud. Untuk menandai *tile* sebagai *mailbox* dilakukan dengan memberikan nilai satu pada kolom *mailbox*. Untuk menghapus fungsi *mailbox* pada *tile* dengan memberikan nilai nol.

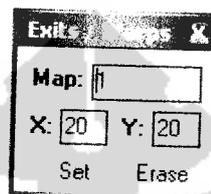
The image shows a 'Tile' information dialog box with the following fields and options:

- X: [ ] Y: [ ]
- Blocked:
- Mailbox:
- Sign: [0]
- NPC: [ ]
- Warp: [0]
- X: [ ] Y: [ ]
- Sfx: [0]
- Layers: 1 2 3 4 5 6
- Grh: [0]
- Old: [0]
- Lights:
  - 1: [0]
  - Old: [0]
  - 2: [0]
  - Old: [0]
  - 3: [0]
  - Old: [0]
  - 4: [0]
  - Old: [0]
- Shadow

Gambar 4.10 Tile Information

j. Exits

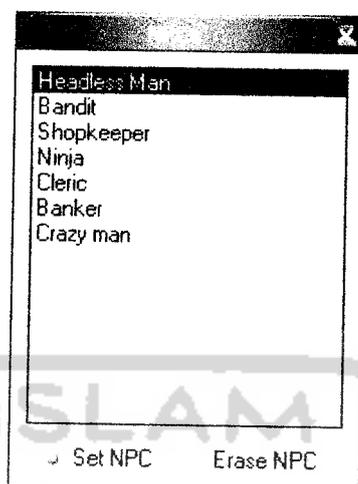
Tombol ini berguna untuk menandai *tile* sebagai *warp point* atau pintu keluar masuk pada suatu *map*. Jendela *exits* atau *warps* dapat dilihat pada gambar 4.11. Untuk menandai menggunakan *set* dan untuk menghapus menggunakan *erase*. Kolom *map* adalah ID *map* dimana pintu itu menghubungkan. Sedangkan X dan Y adalah koordinat *tile* tempat muncul berada pada *map* yang dimaksudkan.



Gambar 4.11 Jendela Exits

k. Jendela NPC

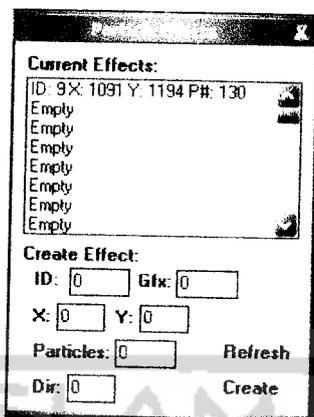
Tombol ini akan menampilkan jendela NPC untuk menempatkan NPC pada *map*. Untuk menempatkan NPC dengan *set* NPC dan menghapus NPC dengan *erase* NPC. Untuk menambahkan NPC dilakukan pada *database* tabel NPC. Jendela NPC dapat dilihat seperti pada gambar 4.12.



**Gambar 4.12** Jendela NPC

#### I. Particle Effects

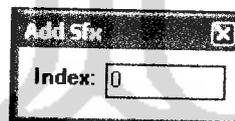
*Particle effects* digunakan untuk membuat efek animasi pada map. Efek ini akan berada pada map dan terlihat oleh seluruh client. Jendela *particle effects* dapat dilihat pada gambar 4.13. ID adalah ID dari *particle effect* yang digunakan. Sedangkan Gfx adalah tekstur yang digunakan untuk membentuk *particle effect*. X dan Y adalah posisi efek dalam pixel. *Particles* adalah jumlah *particles* yang digunakan. Dan dir adalah arah dari animasi *particle*. Sampai saat ini fitur ini masih dalam pengembangan sehingga masih belum digunakan dalam pembuatan *map*.



Gambar 4.13 Particle effects

m. Sound Effects

*Sound effects* berguna untuk menambahkan efek suara pada suatu *map*. Untuk menambahkan sound effect pada *map* dengan menuliskan nomor dari *sound effect* yang tersimpan pada \Sfx. Penambahan file efek suara dilakukan dengan penomoran sederhana. Jendela *sound effects* dapat dilihat pada gambar 4.14.

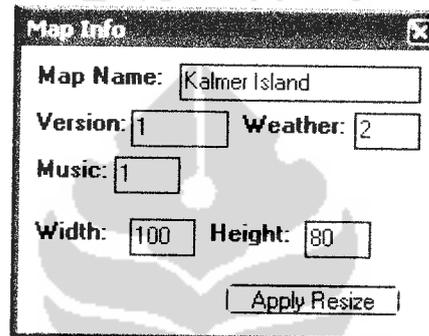


Gambar 4.14 Jendela sound effects

n. Map Info

Tombol ini akan menampilkan jendela *map info*. *Map info* berisi informasi tentang *map* yang bersangkutan. Terdapat nama *map* yang dapat dirubah sewaktu-waktu. Versi map jika ada update dan jika versi pada *client* tidak cocok dengan *server* maka dapat dilakukan

*update*. Kolom *music* untuk diisi dengan nomor ID musik yang dimainkan ketika karakter berada pada *map* tersebut. *Weather* adalah efek cuaca yang terjadi pada *map*. Sedangkan *width* dan *height* untuk pengaturan panjang dan lebar *map* dengan maksimal 100x100, untuk perubahan menggunakan tombol *apply resize*. Jendela *map info* dapat dilihat pada gambar 4.15.



Gambar 4.15 Jendela *map info*

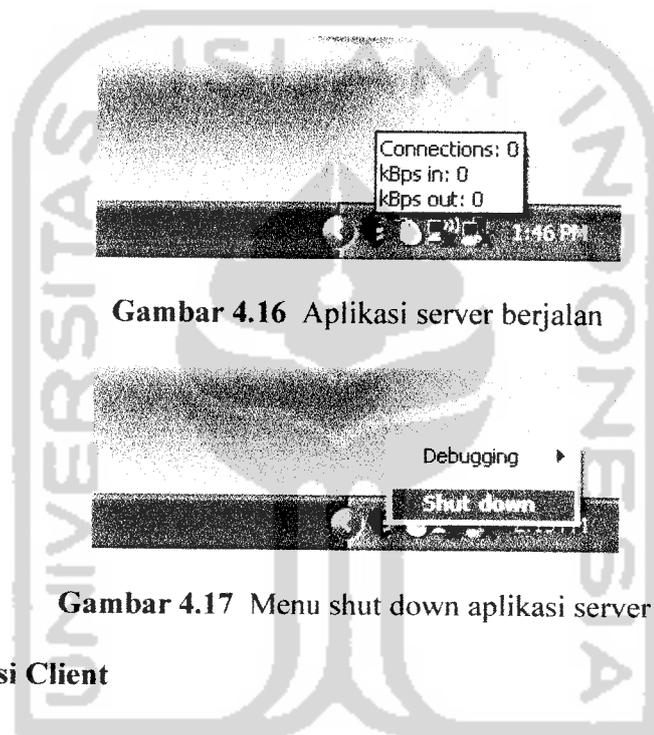
o. Hide/Show Weather

Tombol ini memperlihatkan atau menyembunyikan efek cuaca pada *map*. Biasanya untuk perancangan *map* dengan mengaktifkan efek cuaca akan sangat mengganggu.

p. Hide/Show NPCs

Tombol ini memperlihatkan atau menyembunyikan NPC yang ada pada *map*.

4.16. Informasi dasar akan terlihat pada *event mouse over* yang memperlihatkan jumlah koneksi *client* yang sedang terjadi, *request* masuk dan data dikirimkan dalam *kBps*. Untuk mematikan aplikasi server menggunakan tombol klik kanan *mouse* yang akan menampilkan pilihan untuk mematikan server seperti terlihat pada gambar 4.17.



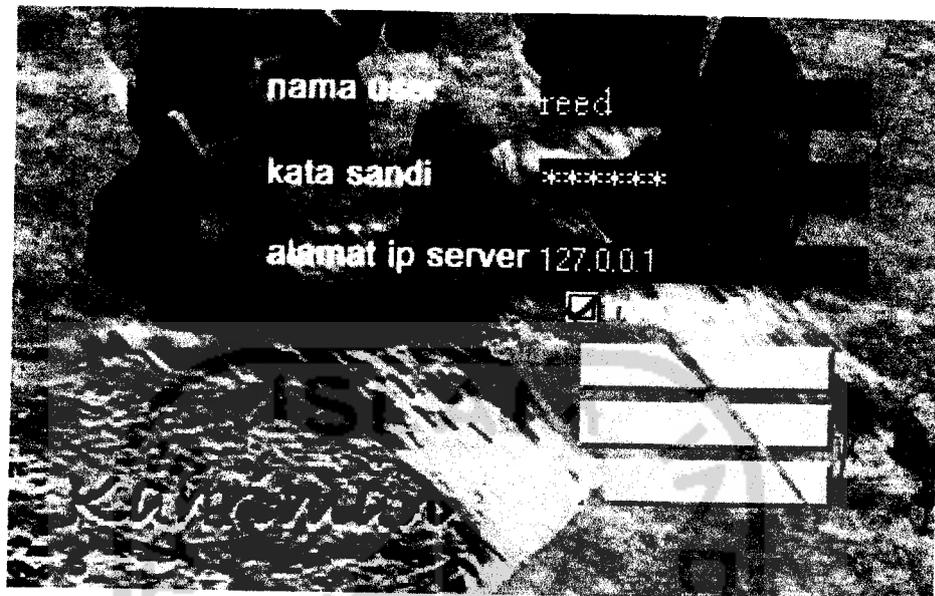
Gambar 4.16 Aplikasi server berjalan

Gambar 4.17 Menu shut down aplikasi server

### 4.5.3 Aplikasi Client

#### 1. Menu Login

Pada aplikasi *client* permainan pertama kali dibuka akan ditampilkan menu *login* seperti terlihat pada gambar 4.18. Pada menu ini terdapat isian nama pemain, kata sandi, dan alamat IP server. Terdapat pilihan untuk menyimpan informasi *login* dan pilihan untuk mengkonfirmasi hubungan ke server, pilihan membuat akun baru dan pilihan keluar dari permainan.



Gambar 4.18 Menu login

## 2. Menu Pembuatan Akun Baru

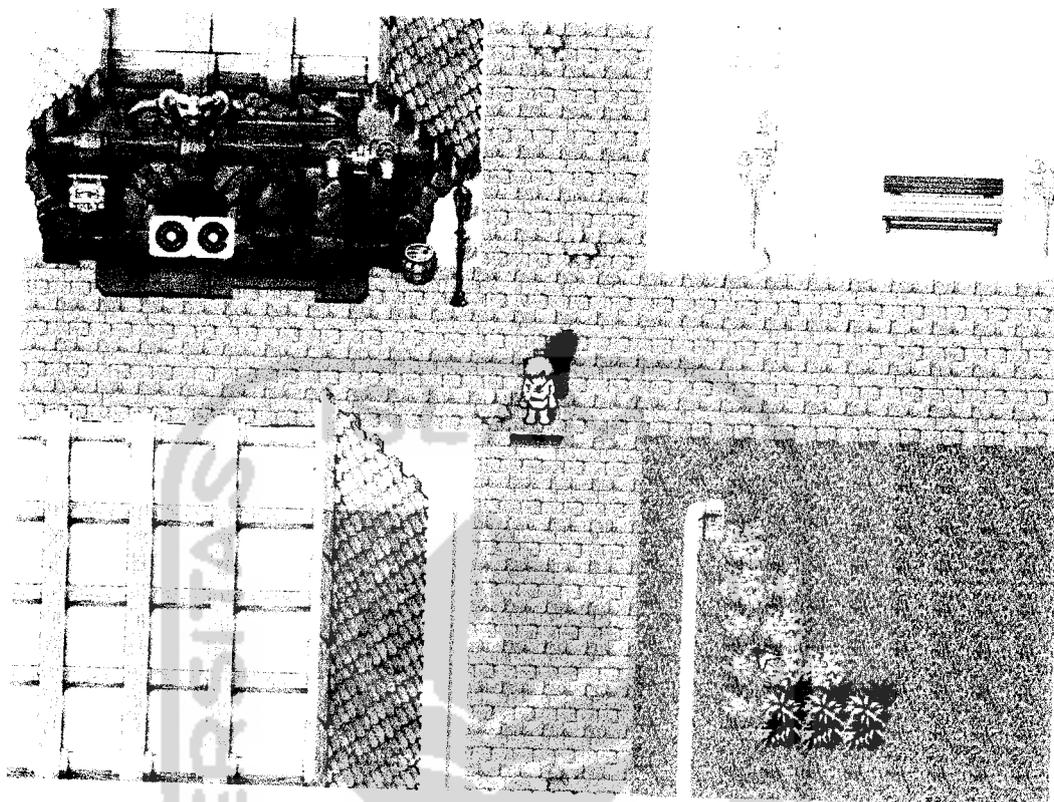
Untuk pemilihan pembuatan akun baru pada menu *login* akan merujuk pada menu pembuatan akun baru. Pada menu ini terdapat isian nama *pemain*, kata sandi, *class* yang diinginkan pada permainan meliputi 4 pilihan *class*. Terdapat isian alamat ip server untuk menghubungkan pada server yang dituju jika akun siap untuk dibuat, jika tidak terdapat pilihan untuk kembali ke menu login. Menu pembuatan akun baru dapat dilihat pada gambar 4.19.



Gambar 4.19 Menu pembuatan akun baru

### 3. Layar Permainan

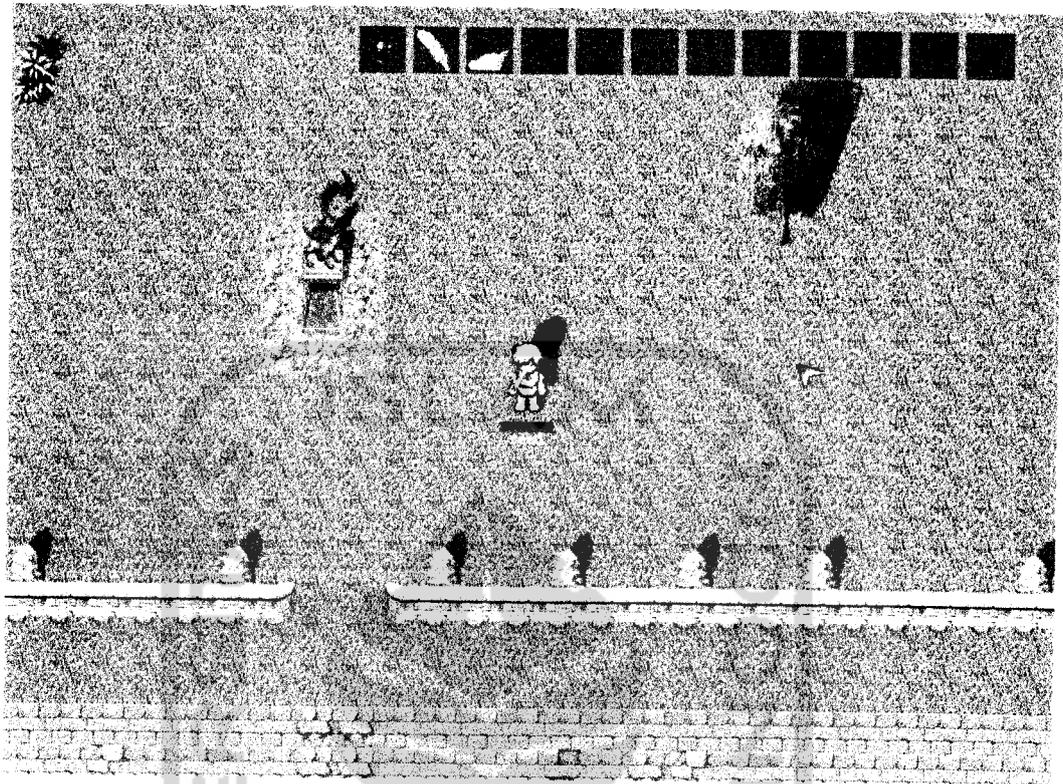
Pada aplikasi *client* yang sudah terhubung dengan server permainan akan menampilkan layar permainan. Pada layar permainan akan terlihat tokoh sebagai *Player Character (PC)* yang dikontrol menggunakan *keyboard* dan *mouse*. Tokoh yang dimainkan pemain akan berada pada tengah-tengah layar permainan pada suatu *map* yang telah disiapkan server. Informasi dasar tokoh meliputi nama tokoh, *healthpoints* dan *manapoints* dalam bentuk *bar* berada dibawah karakter tokoh. Layar Permainan dapat dilihat seperti pada gambar 4.20.



Gambar 4.20 Layar permainan

#### 4. Jendela Quickbar

Pada layar permainan dapat ditampilkan Jendela *Quickbar* melalui kontrol *keyboard* yang telah didefinisikan oleh pemain seperti terlihat pada gambar 4.21. Terdapat dua belas *shortcut* pada jendela *quickbar* ini dan pemain dapat mengatur jalan pintas terhadap penggunaan suatu item dari *inventory* tokoh ataupun jalan pintas terhadap penggunaan suatu *skill* dari daftar *skill* yang diketahui oleh tokoh.

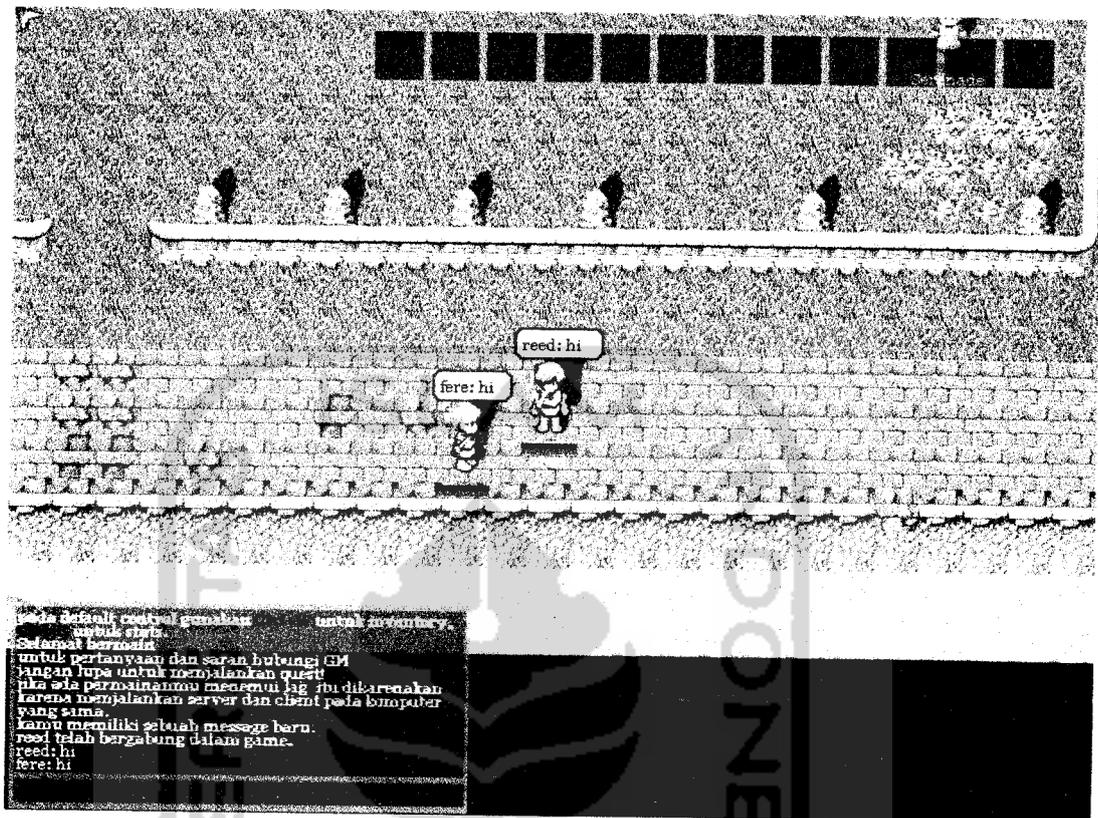


**Gambar 4.21** Layar permainan dengan quickbar

#### 5. Jendela Chat

Informasi terhadap kejadian yang terjadi pada permainan dapat terlihat pada jendela *chat*. Pada jendela ini setiap pembicaraan, interaksi ataupun perintah-perintah yang dilakukan pemain akan tercatat pada jendela *chat*. Jendela *chat* seperti yang terlihat pada gambar 4.22, juga dapat dikontrol untuk melihat informasi pembicaraan yang telah berlalu dengan menggunakan kontrol yang telah didefinisikan pemain pada konfigurasi permainan.

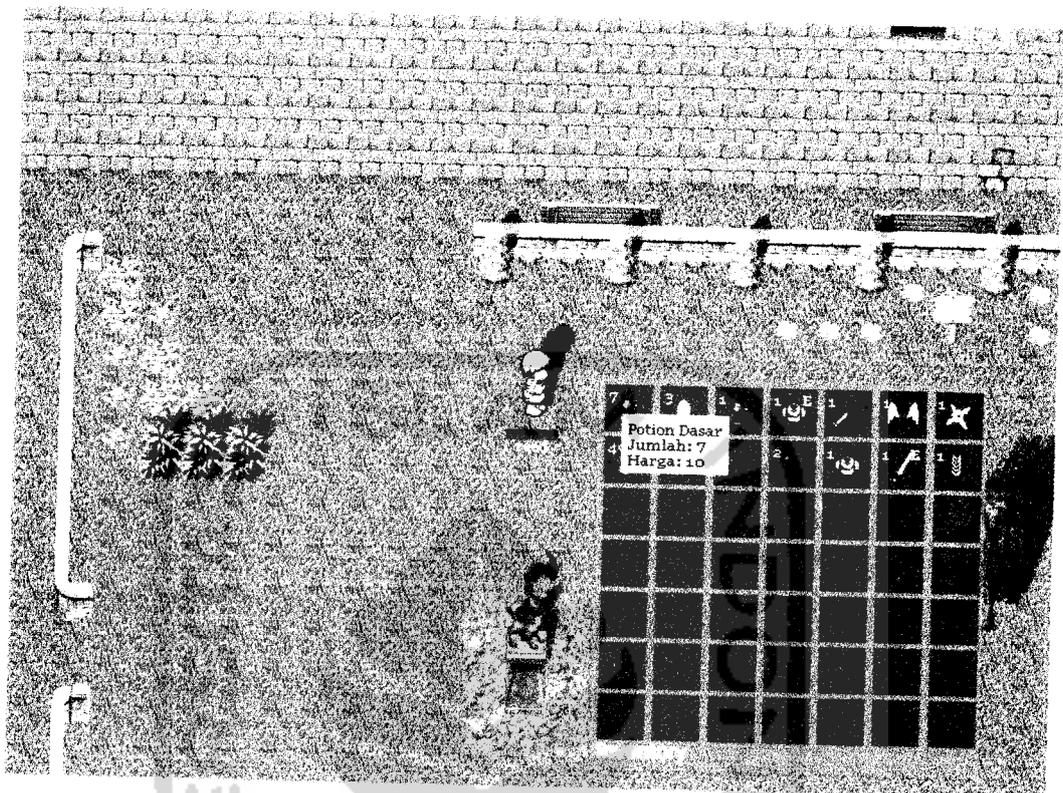




**Gambar 4.22** Layar permainan dengan jendela chat

#### 6. Jendela *Inventory*

Pada jendela *inventory* terdapat informasi barang-barang bawaan tokoh seperti yang terlihat pada gambar 4.23. Informasi setiap barang dapat dilihat dengan menggunakan tombol klik kanan pada *mouse* , dan untuk menggunakan barang dengan tombol klik kiri pada *mouse* . Organisasi barang bawaan dapat dilakukan dengan menahan tombol klik kanan dan *drag* ke slot lain maupun membuang ke tanah.

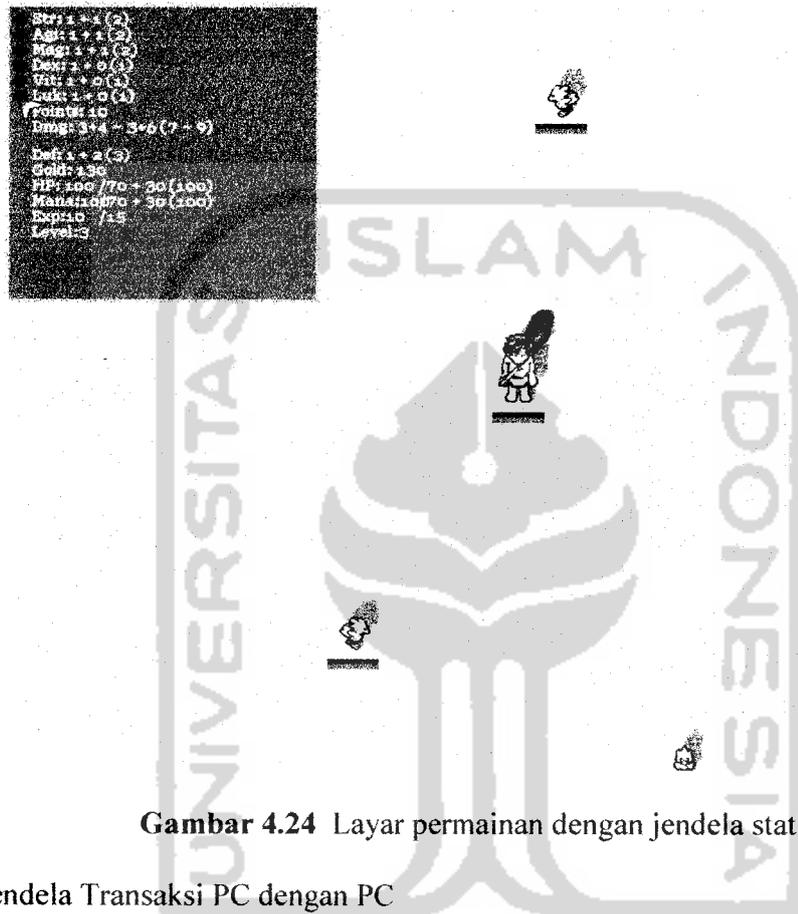


**Gambar 4.23** Layar permainan dengan jendela inventory

#### 7. Jendela Statistik

Informasi lengkap keadaan dan statistik tokoh dalam digit dapat dilihat pada jendela statistik. Jendela statistik dapat ditampilkan melalui kontrol yang telah didefinisikan pemain pada konfigurasi permainan. Informasi yang terlihat adalah statistik dan informasi kondisi yang sedang terjadi pada tokoh. Pada statistik tokoh dapat ditambahkan sesuai dengan kebutuhan pemain jika memiliki points. *Point* didapat dari kenaikan *level*, satu *point* hanya dapat digunakan untuk menaikkan satu nilai dari salah satu atribut statistik. Atribut statistik yang bisa dinaikkan nilainya akan menampilkan tanda plus (+) berwarna

merah pada sebelah kiri atribut statistik tersebut seperti yang terlihat pada gambar 4.24.

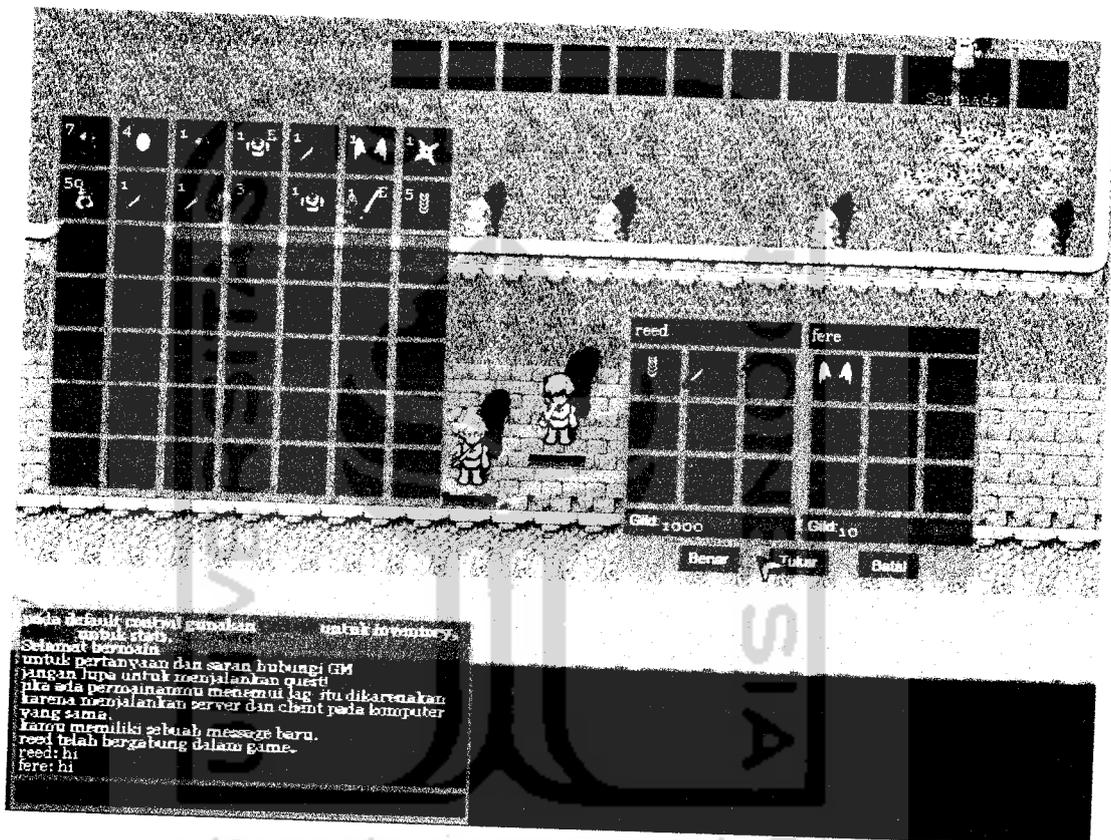


**Gambar 4.24** Layar permainan dengan jendela statistik

#### 8. Jendela Transaksi PC dengan PC

Jendela transaksi antar *player character* yang dimainkan pemain yang berbeda terpicu jika terjadi suatu transaksi dengan mengetikkan perintah `/trade` pada jendela *chat* pada tiap-tiap pemain yang akan melakukan transaksi. Pada jendela ini terdapat dua informasi yang berbeda tentang kedua pemain yang akan melakukan transaksi. Terdapat dua kolom berbeda untuk masing-masing pemain, yaitu nama, barang yang akan ditukarkan dan jumlah

gold yang ditukarkan. Terdapat dua pilihan yaitu konfirmasi barang yang akan ditukarkan dan pembatalan transaksi. Pilihan penukaran akan aktif jika konfirmasi dari kedua belah pihak telah diajukan. Jendela transaksi dapat dilihat pada gambar 4.25.

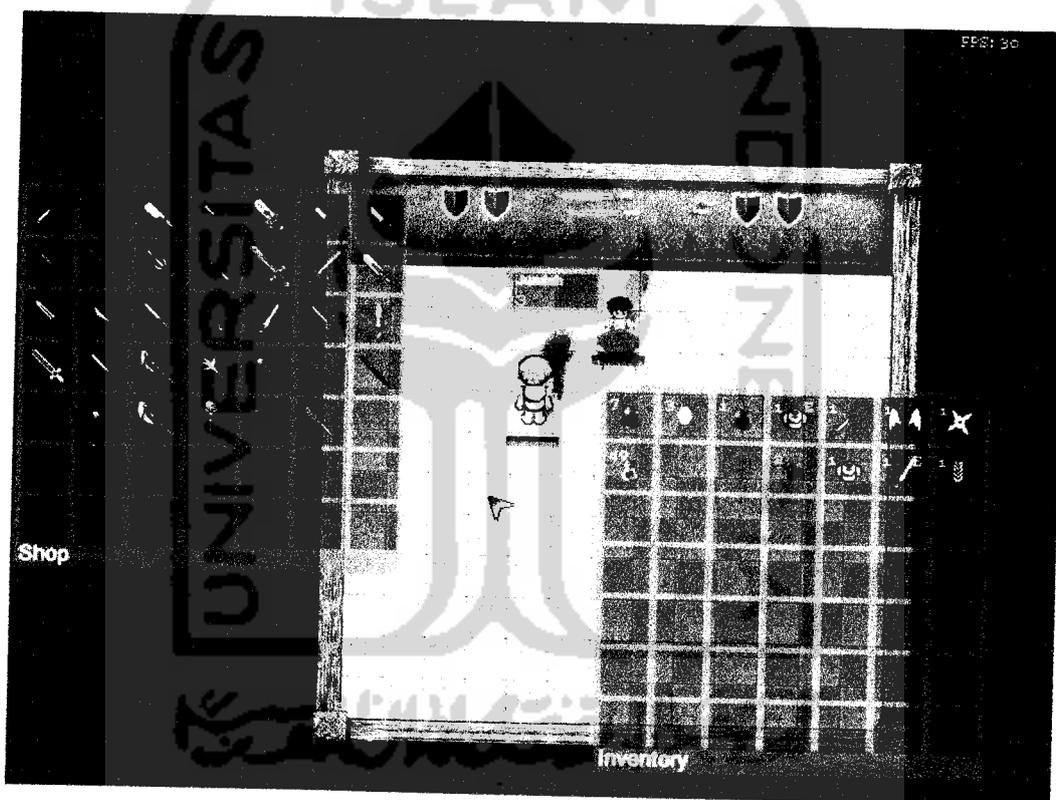


Gambar 4.25 Layar permainan dengan jendela transaksi antar PC

#### 9. Jendela Transaksi PC dengan NPC Shop

Pada permainan terdapat *non playing character* (NPC) yang berfungsi sebagai toko untuk membeli barang-barang. Jendela NPC *shop* dapat diakses dengan tombol klik kanan pada *mouse* pada NPC yang telah ditunjuk sebagai NPC *shop*. Untuk melakukan transaksi pembelian dengan menahan tombol klik

kanan pada *mouse* dan *drag* pada barang yang diperlukan dari jendela NPC *shop* ke jendela *inventory* . Untuk melakukan transaksi penjualan dengan menahan tombol klik kanan pada *mouse* dan *drag* pada barang yang diperlukan dari jendela *inventory* ke jendela NPC *shop* . Jendela konfirmasi jumlah akan muncul dan pemain diminta untuk memasukkan jumlah yang diinginkan. Jendela NPC *shop* ini dapat dilihat pada gambar 4.26.

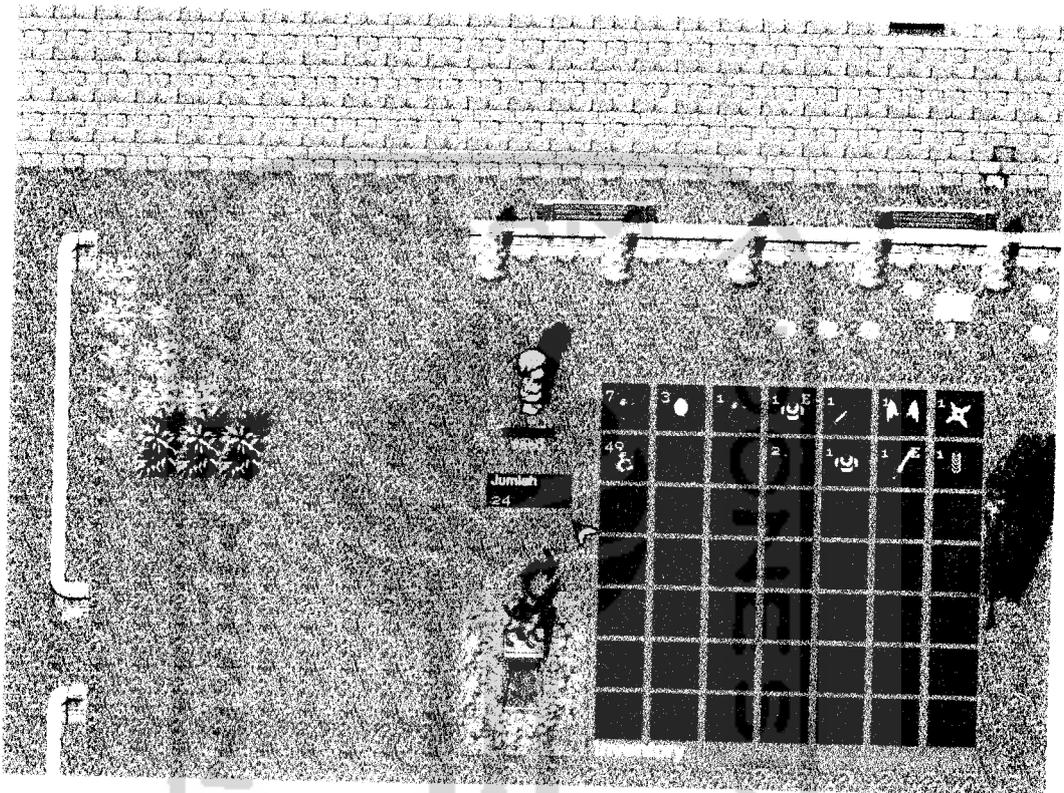


**Gambar 4.26** Layar Permainan dengan jendela transaksi PC dengan NPC

#### 10. Jendela Konfirmasi Jumlah

Jendela konfirmasi jumlah pada permainan dapat terpicu pada lima kejadian, yaitu ketika terjadi transaksi dengan NPC *shop* , transaksi antar PC,

menjatuhkan barang ketanah, menyimpan barang dari *inventory* ke gudang dan ketika mengirimkan barang melalui *mail* . Pemain akan diminta untuk memasukkan jumlah yang diinginkan seperti yang terlihat pada gambar 4.27.

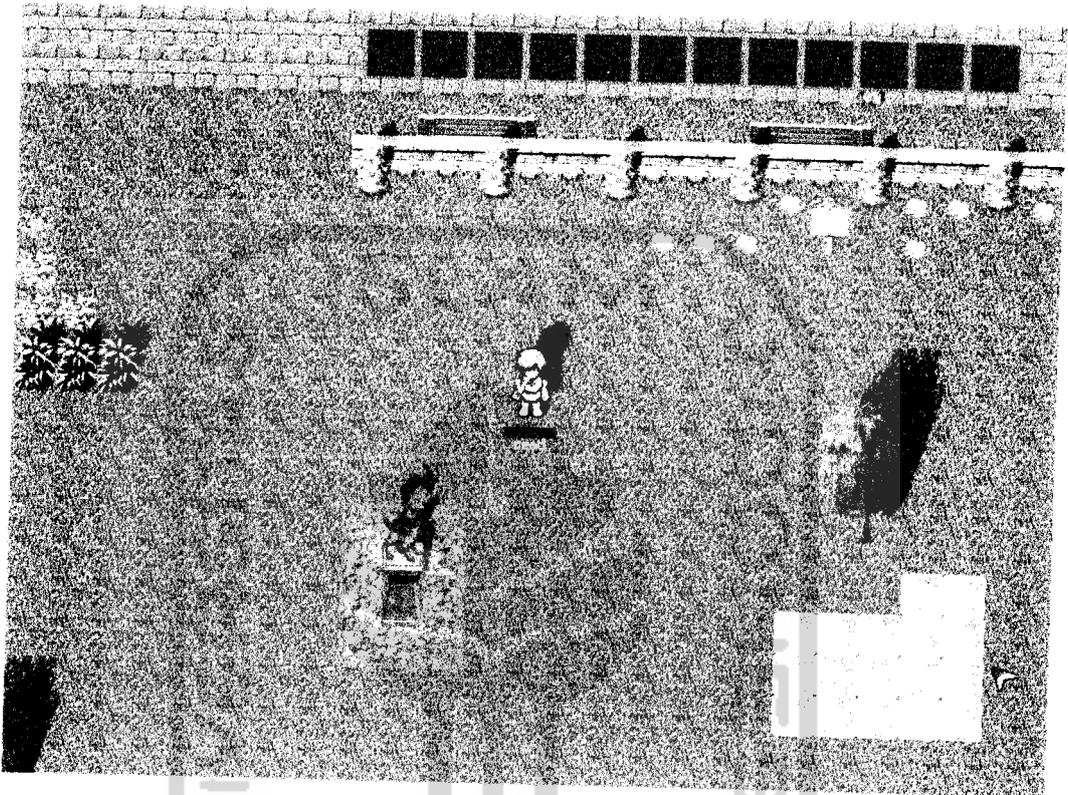


**Gambar 4.27** Layar permainan dengan jendela konfirmasi jumlah

## 11. Daftar Skill

Tokoh pada permainan dapat menguasai kemampuan yang berbeda-beda sesuai dengan class masing-masing. Setiap *skill* yang dikuasai akan diwakili dengan gambar ikon *skill* yang bersangkutan. Daftar ikon *skill* tersebut akan disusun pada pojok kanan bawah layar permainan dan dapat

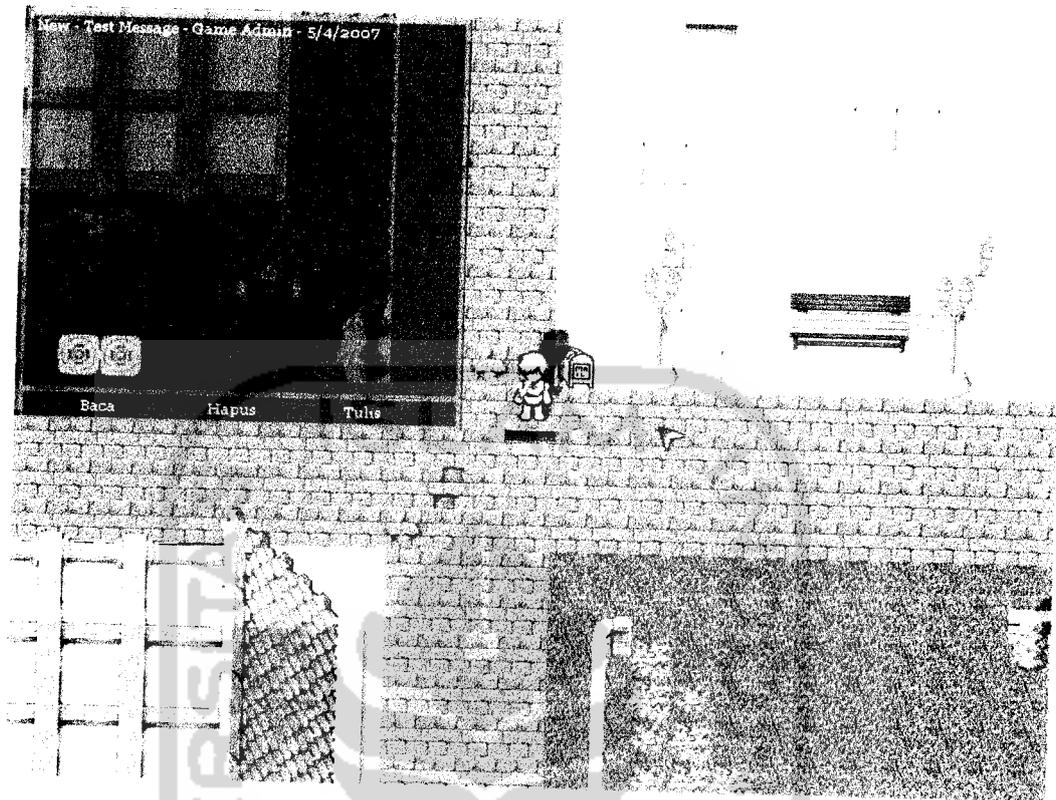
diakses melalui tombol shift dan klik kiri *mouse* pada jendela *quickbar*. Daftar *skill* bisa dilihat seperti pada gambar 4.28.



**Gambar 4.28** Layar permainan dengan daftar skill

## 12. Jendela Mailbox

Jendela *Mailbox* hanya dapat diakses melalui tombol klik kanan *mouse* pada kotak *mail* yang telah ditentukan oleh permainan seperti terlihat pada gambar 4.29. Pemain hanya dapat mengakses jendela *mailbox* jika berada didekat *mailbox*. Pada jendela *mailbox* terdapat daftar *mail* yang diterima baik yang sudah terbaca ataupun yang belum terbaca. Terdapat tiga pilihan pada jendela *mailbox* ini yaitu membaca *mail*, menulis *mail* atau menghapus *mail*.



**Gambar 4.29** Layar permainan dengan jendela mail box

### 13. Jendela Baca Mail

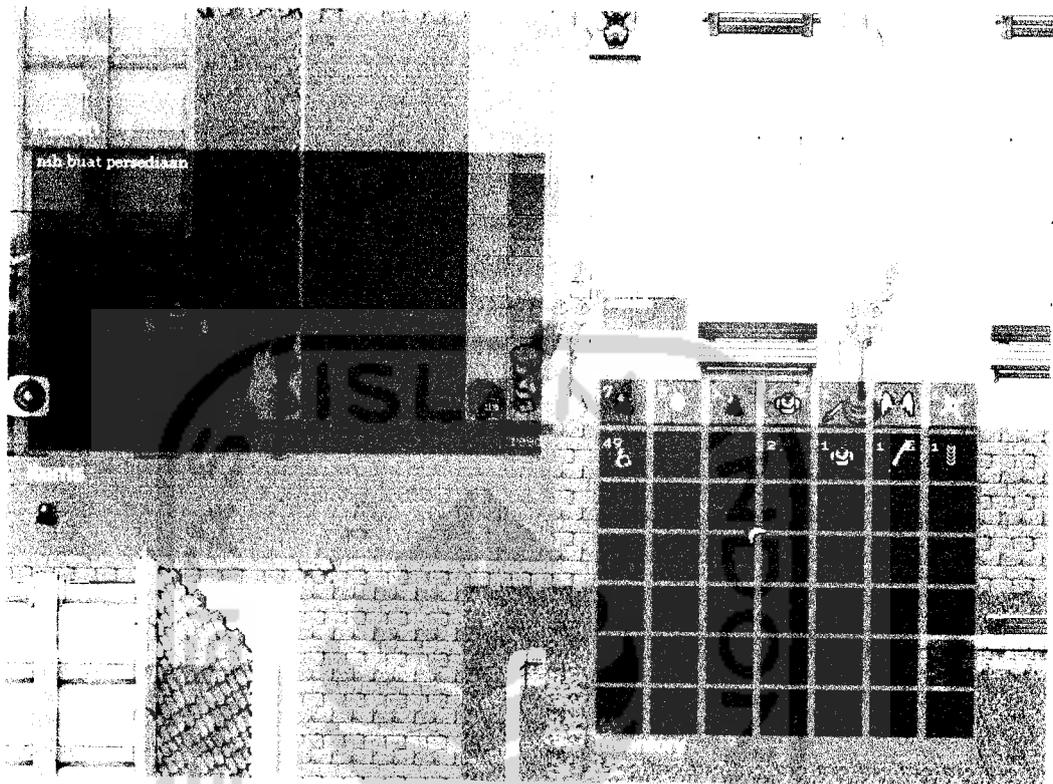
*Mail* yang berada pada jendela *mailbox* dapat dibaca dan diambil item yang disertakan didalamnya. Pada *mail* yang dibaca terdapat nama pengirim *mail*, topik *mail*, isi *mail* dan item yang disertakan bersama *mail* jika ada. Jendela baca *mail* dapat dilihat seperti pada gambar 4.30.



**Gambar 4.30** Layar permainan dengan jendela baca mail

#### 14. Jendela Tulis Mail

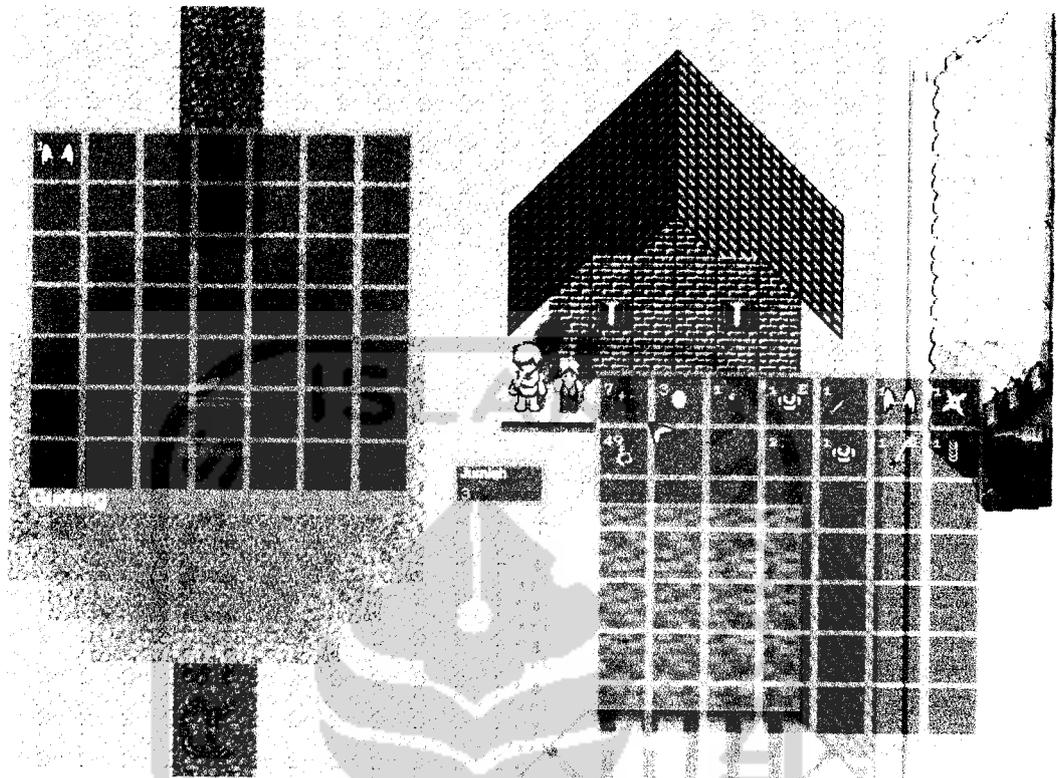
Pada pilihan untuk menulis *mail* pada jendela *mailbox* akan menampilkan jendela tulis *mail* seperti terlihat pada gambar 4.31. Pemain dapat menulis *mail* dan mengirimnya selama masih memiliki biaya untuk pengiriman yang besarnya telah ditentukan. Pada jendela tulis *mail* terdapat isian tujuan atau penerima *mail* , topik *mail* yang akan ditulis, dan isi *mail* yang dikehendaki. Pemain dapat menyertakan item untuk dikirim bersama *mail*.



**Gambar 4.31** Layar permainan dengan tulis mail

#### 15. Jendela Gudang Penyimpanan

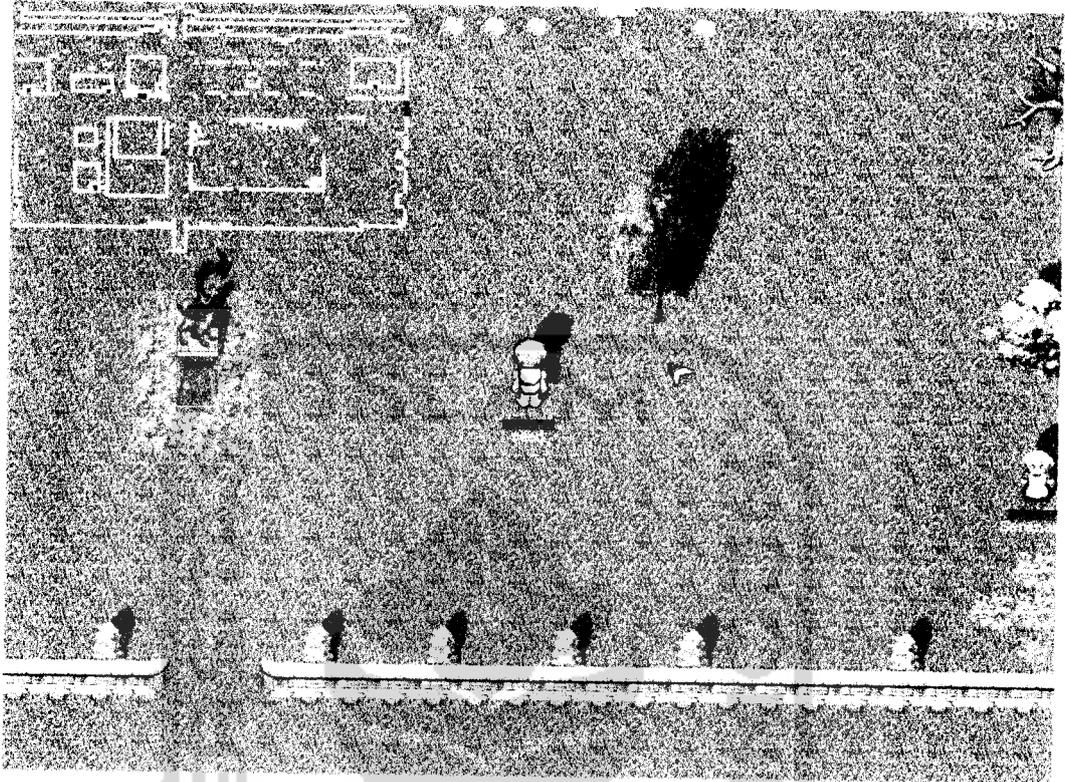
Gudang penyimpanan pada permainan dapat diakses melalui NPC Banker yang telah ditentukan. Jendela gudang penyimpanan berfungsi untuk menampung barang-barang yang tidak diperlukan dalam *inventory*. Untuk menyimpan atau mengambil barang dapat dilakukan dengan menahan tombol klik kanan *mouse* dan *drag* item dari jendela *inventory* atau jendela gudang. Untuk item yang dapat ditumpuk akan menampilkan jendela konfirmasi jumlah yang harus diisi pemain. Jendela gudang penyimpanan ini dapat dilihat seperti pada gambar 4.32.



**Gambar 4.32** Layar permainan dengan jendela gudang penyimpanan

#### 16. Mini Map

*Mini map* ditampilkan melalui tombol Tab pada keyboard. *Mini map* berfungsi untuk mewakili keadaan map yang sebenarnya. Informasi yang dapat ditampilkan melalui kode warna sebaagai berikut; putih adalah daerah map, merah adalah pintu menuju map lain, biru adalah NPC, hijau adalah posisi tokoh pada map yang bersangkutan. *Mini map* akan ditampilkan pada pojok kiri atas layar permainan, seperti terlihat pada gambar 4.33.

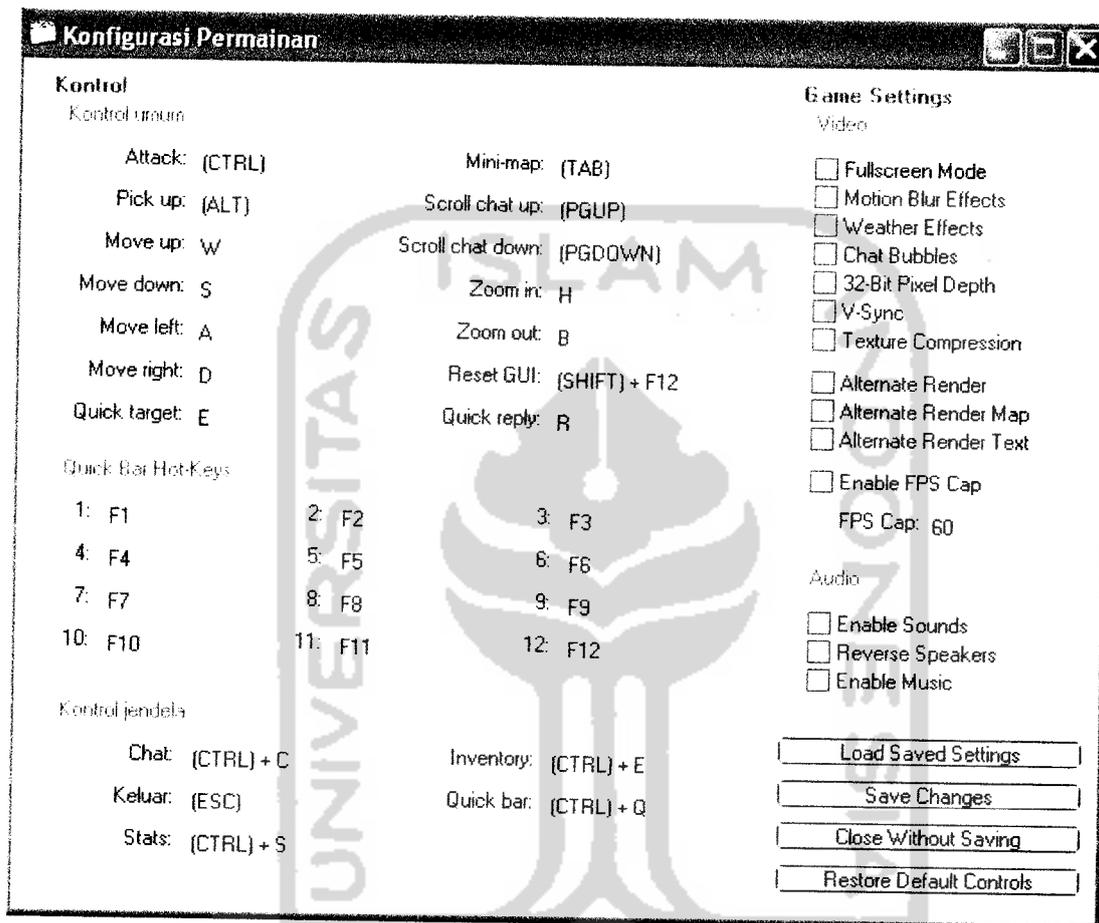


**Gambar 4.33** Layar permainan dengan mini map

#### 17. Kontrol Pertarungan

Inti dari permainan adalah pertarungan dengan monster-monster. Pemain dapat mengontrol tokohnya sesuka hati dengan batasan-batasan yang ada pada permainan. Untuk bertarung dengan monster, pemain diharuskan menjalankan tokohnya ke daerah dengan monster dan menyerang monster sesuai kemampuan tokoh. Jika merasa kurang yakin bisa mengalahkan monster pemain dapat melarikan tokohnya ketempat aman. Pada pertarungan diperlihatkan informasi *damage* yang didapatkan tokoh dan monster yang saling menyerang. *Damage* yang diterima tokoh berwarna merah dan *damage* yang

mengembalikan settingan konfigurasi ke default yang disarankan permainan. Gambar konfigurasi permainan dapat dilihat pada gambar 4.35.



Gambar 4.35 Aplikasi Konfigurasi Permainan

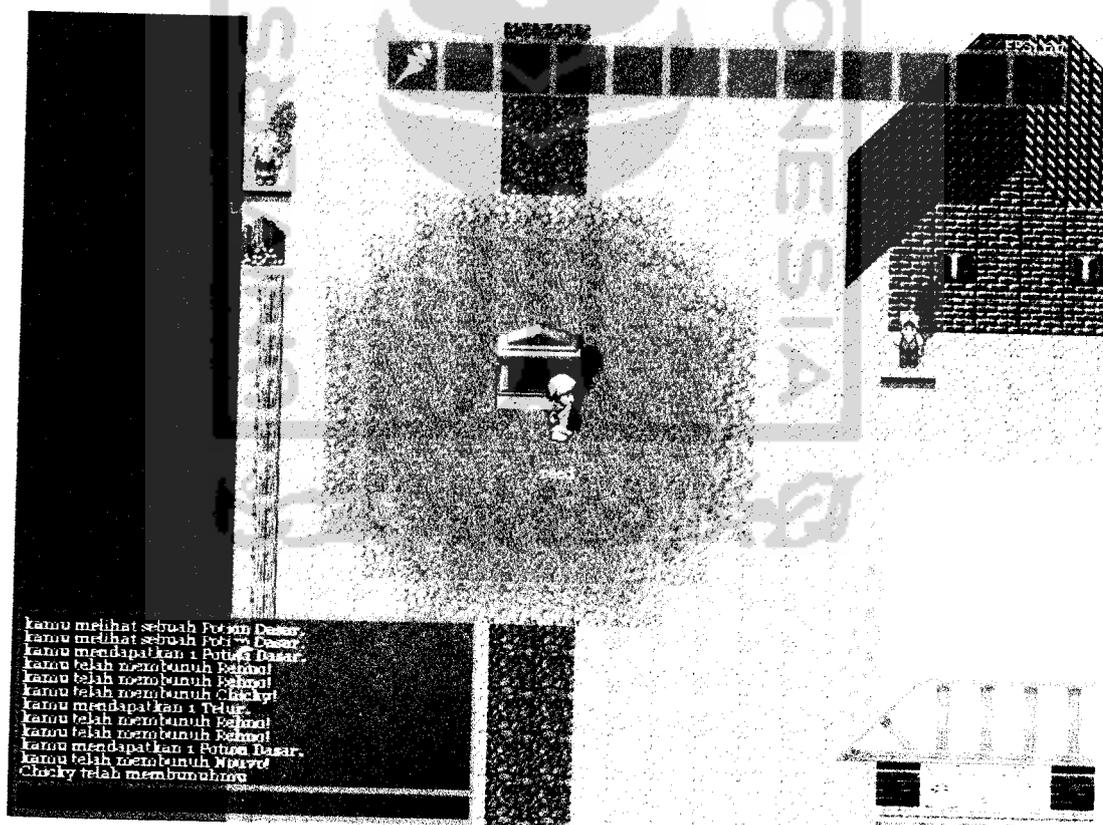
## 4.6 Analisis Kinerja Perangkat Lunak

### 4.6.1 Pengujian Normal

Pengujian normal dilakukan dengan memberikan masukan data sesuai dengan yang dibutuhkan.

## 2. Pemain Terbunuh Oleh Monster

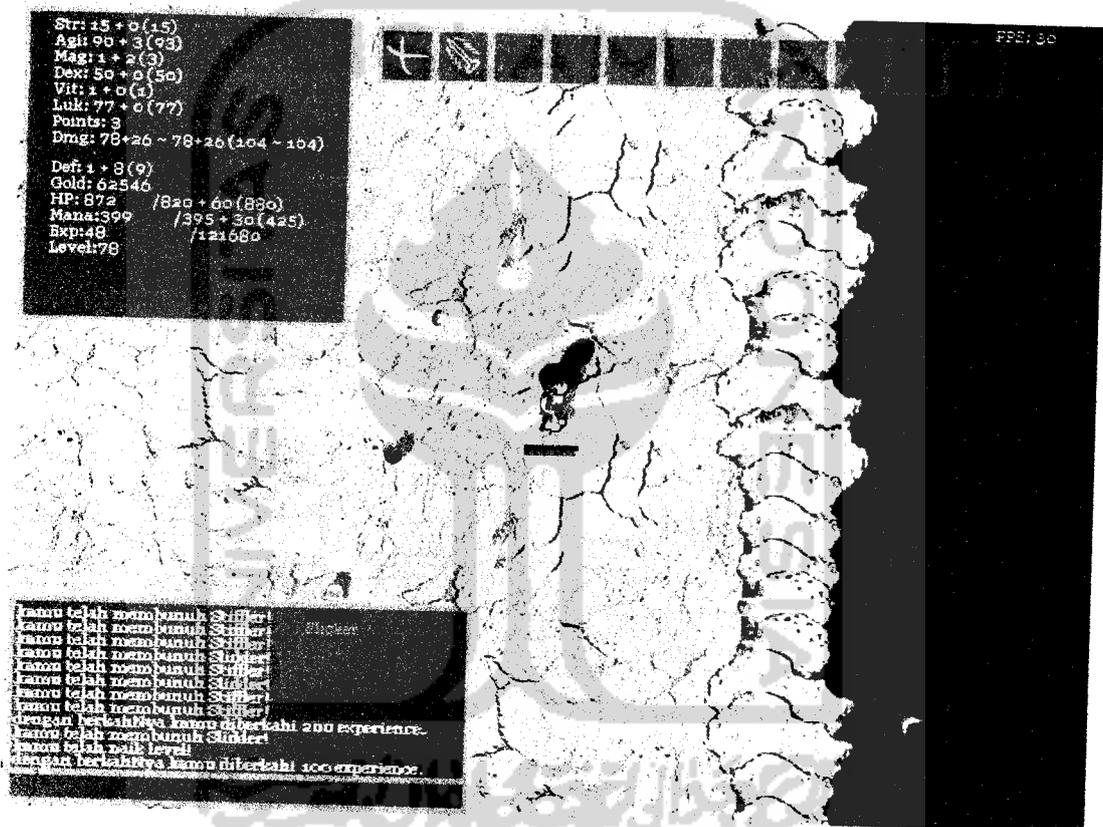
Kemungkinan yang terjadi pada pertarungan dengan monster adalah tokoh kalah bertarung dan mati. Pemain akan mendapatkan informasi nama monster yang membunuhnya. Seperti halnya permainan MRPG lainnya, pada permainan ini tokoh hidup kembali dengan hitpoints dan manapoints kosong sehingga membutuhkan waktu untuk kembali penuh. Pemain yang terbunuh terkena penalti pengurangan satu persen *experience* untuk naik level dari *experience*. Titik kehidupan kembali dan informasi nama monster yang membunuh pemain dapat dilihat seperti pada gambar 4.37.



Gambar 4.37 Jendela chat informasi terbunuh monster

### 3. Naik Level

Pada permainan jika seorang pemain telah mencapai *experience* yang dibutuhkan untuk naik level maka tokohnya akan mengalami kenaikan level yang diinformasikan melalui jendela chat. Informasi kenaikan level dapat dilihat seperti gambar 4.38.

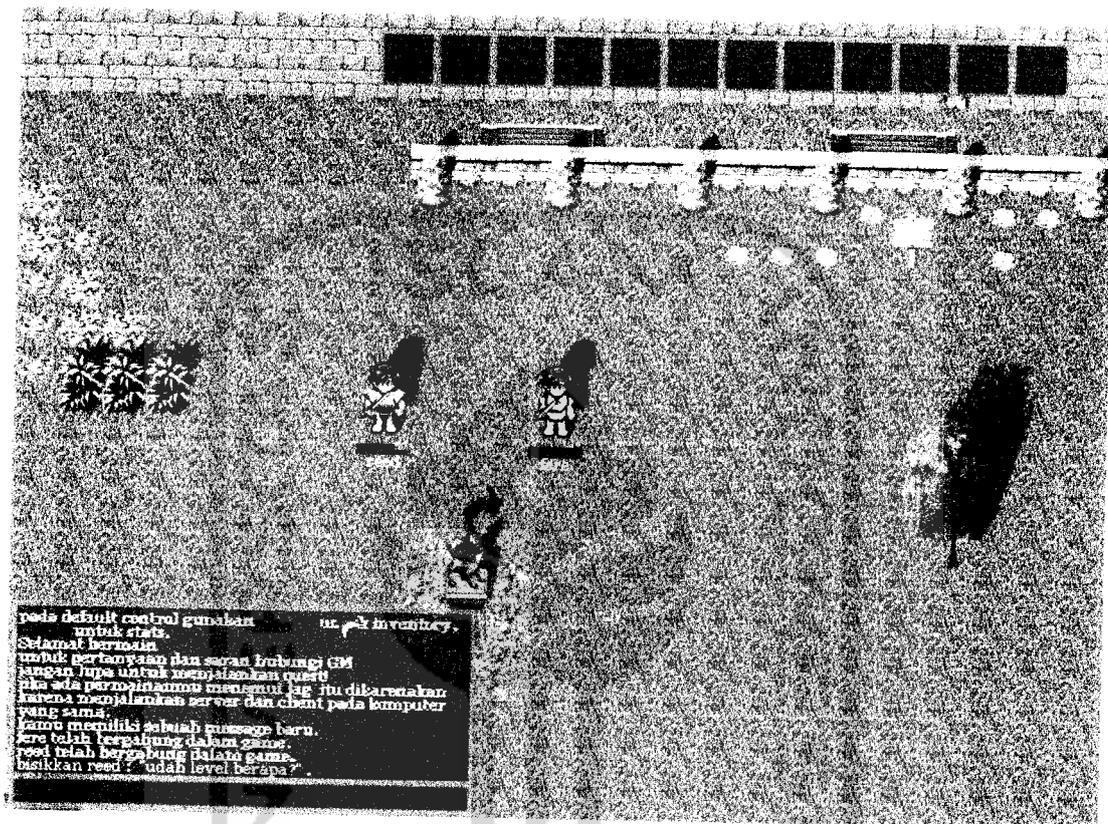


Gambar 4.38 Jendela chat informasi naik level

### 4. Pembicaraan Tertutup Antara Dua Pemain

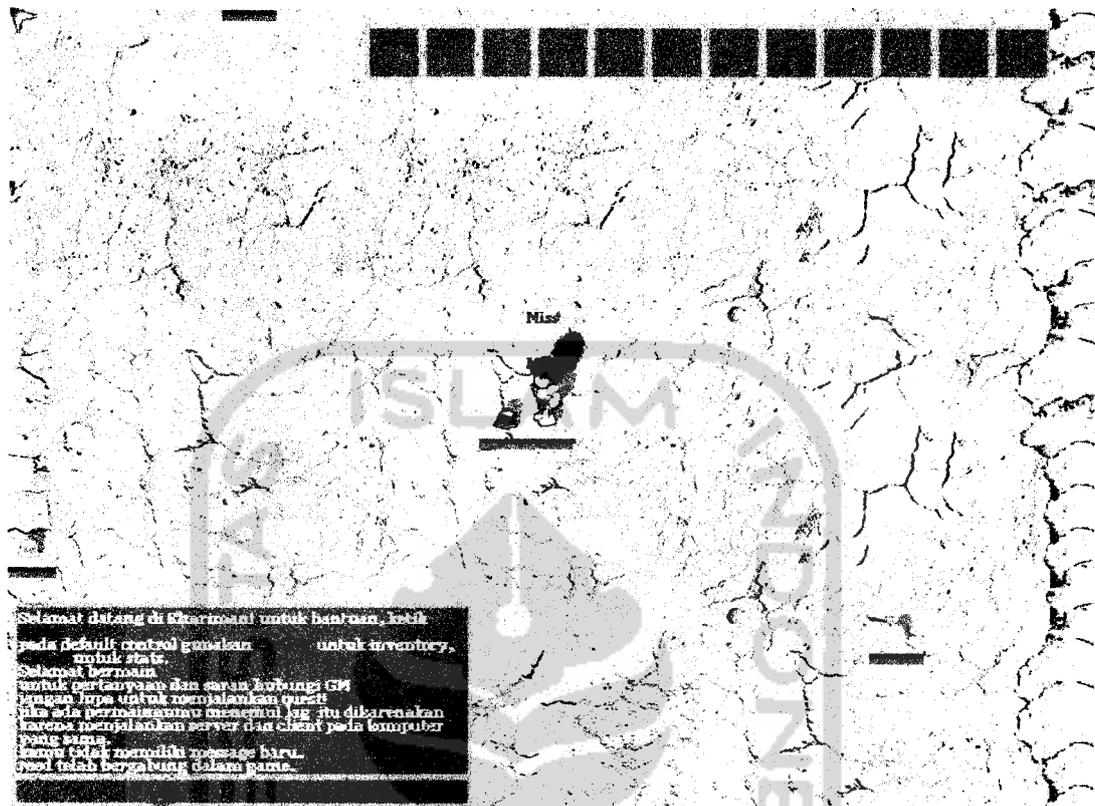
Interaksi antara dua pemain dapat dilakukan secara tertutup, pembicaraan yang dirahasiakan dapat tidak diperlihatkan secara umum dengan perintah `/tell nama_pemain pembicaraan`. Pembicaraan ini dapat dilihat

pada gambar 4.39 untuk pemain yang dituju, sedangkan untuk pemain yang berbicara dapat dilihat pada gambar 4.40.



Gambar 4.39 Jendela chat menerima pembicaraan tertutup



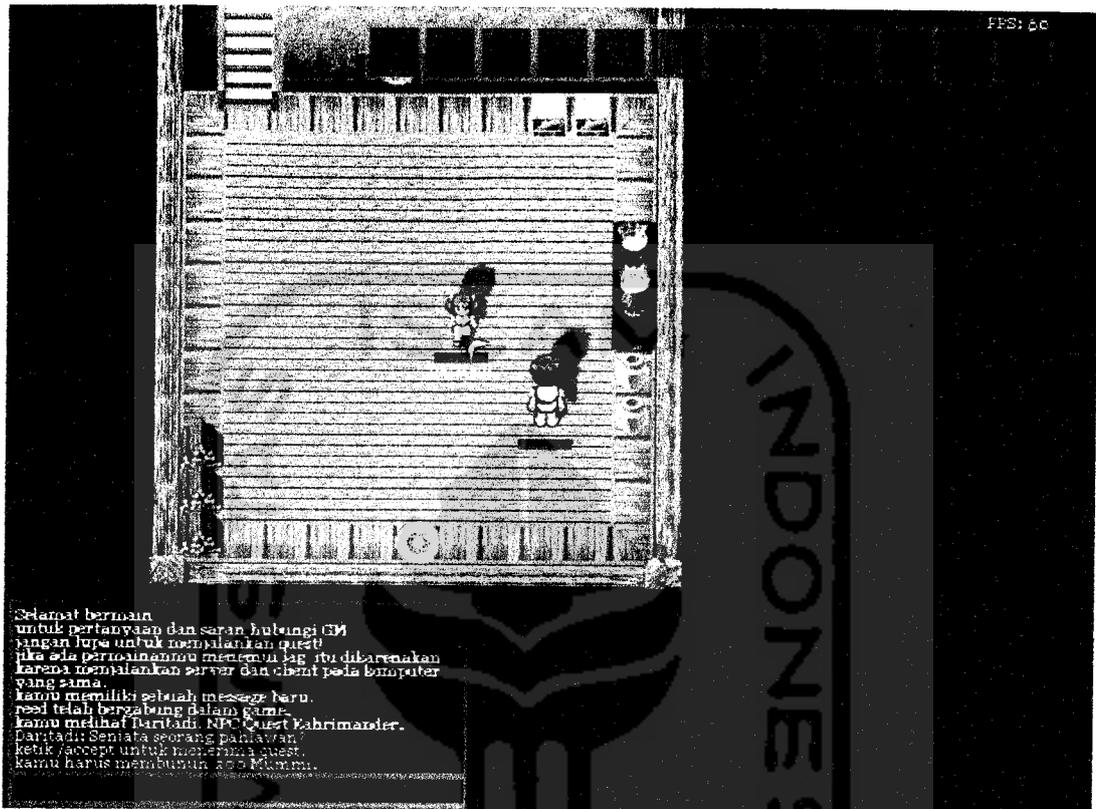


**Gambar 4.41** Jendela chat informasi pemain bergabung bermain

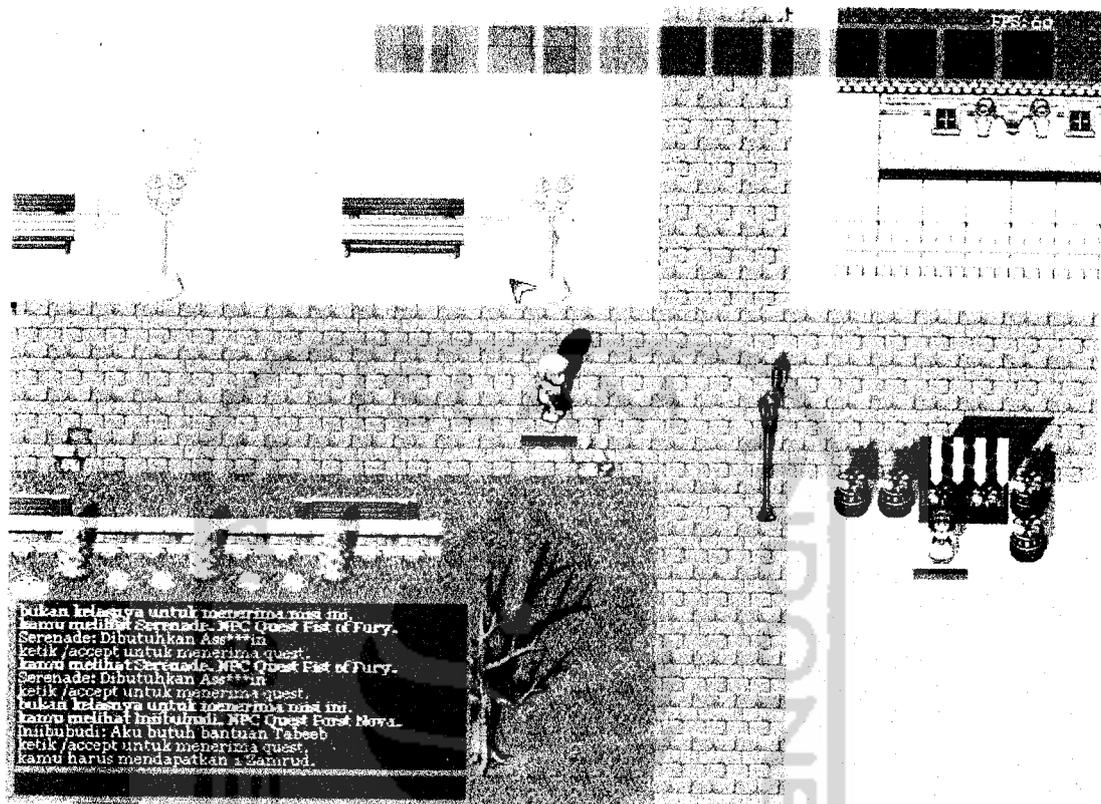
#### 6. Informasi Pemain Keluar Permainan

Seperti halnya dengan bergabungnya pemain, maka pemain yang keluar dari permainan akan diinformasikan pada jendela *chat* seperti terlihat pada gambar 4.42.





**Gambar 4.44** Jendela chat informasi menerima *quest* dengan kebutuhan membunuh monster



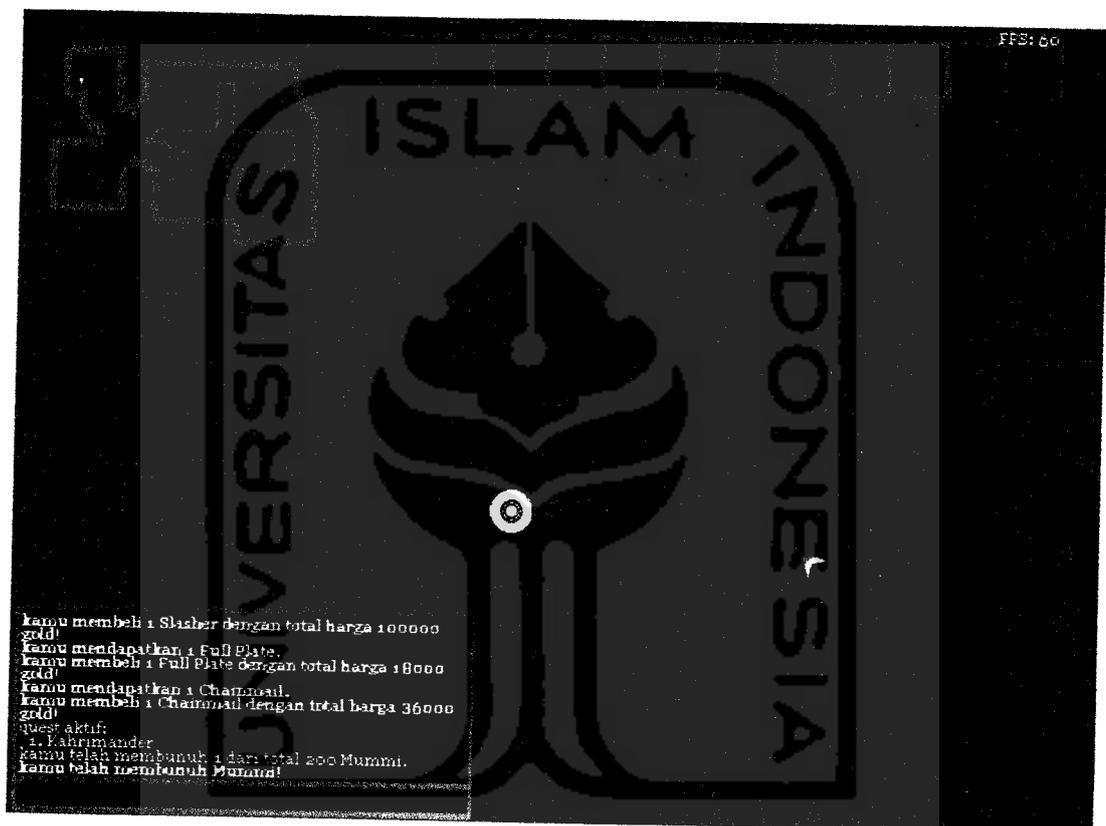
**Gambar 4.45** Jendela chat informasi menerima quest dengan kebutuhan membawakan barang

## 10. Informasi Kebutuhan Quest

Ketika menjalani *quest* dengan kebutuhan membunuh monster, pemain akan diinformasikan sejauh mana *quest* yang dijalani telah ditempuh. Pada jendela *chat* akan diinformasikan jumlah monster yang dibunuh dari sejumlah kebutuhan untuk selesai *quest*. Informasi kebutuhan *quest* ini dapat dilihat pada gambar 4.46.

## 11. Daftar Quest Yang Sedang Dijalani

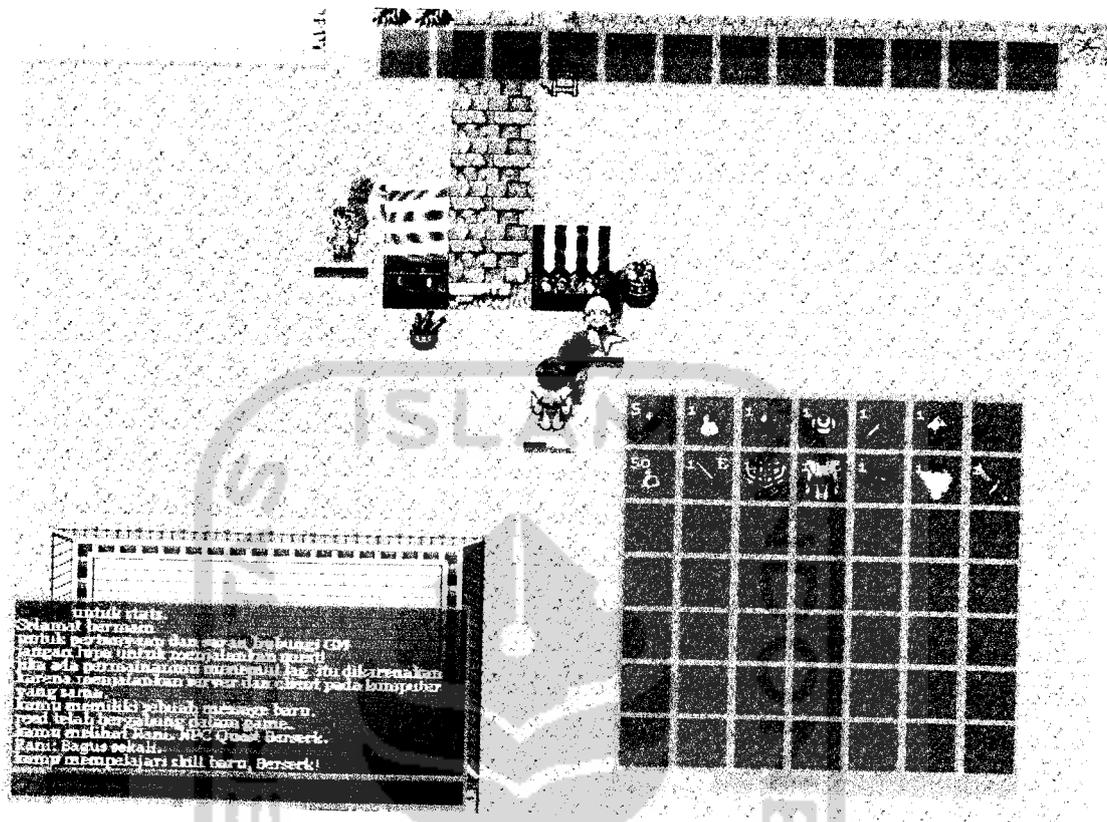
Pemain dapat mengetahui *quest* apa yang sedang dijalannya dengan perintah `/quest`. Pada jendela *chat* akan terlihat informasi daftar *quest* yang sedang dijalani, seperti terlihat pada gambar 4.46.



**Gambar 4.46** Jendela chat informasi kebutuhan *quest* dan daftar *quest* yang sedang dijalani

## 12. Menyelesaikan Quest Skill

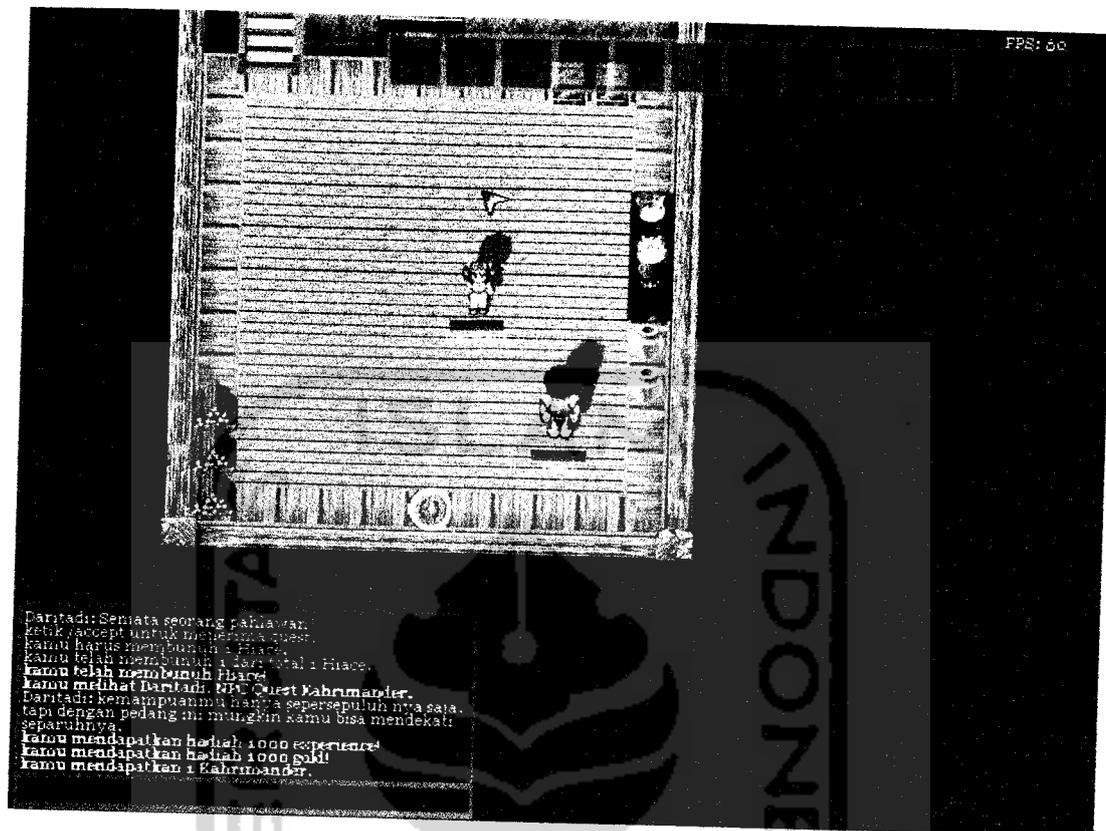
Pemain yang menyelesaikan *quest skill* akan mendapatkan *skill* sesuai dengan *quest skill* yang dijalannya. Informasi *skill* yang didapatkan diperlihatkan seperti pada gambar 4.47.



Gambar 4.47 Jendela chat informasi menyelesaikan quest skill

### 13. Menyelesaikan Quest Non-Skill

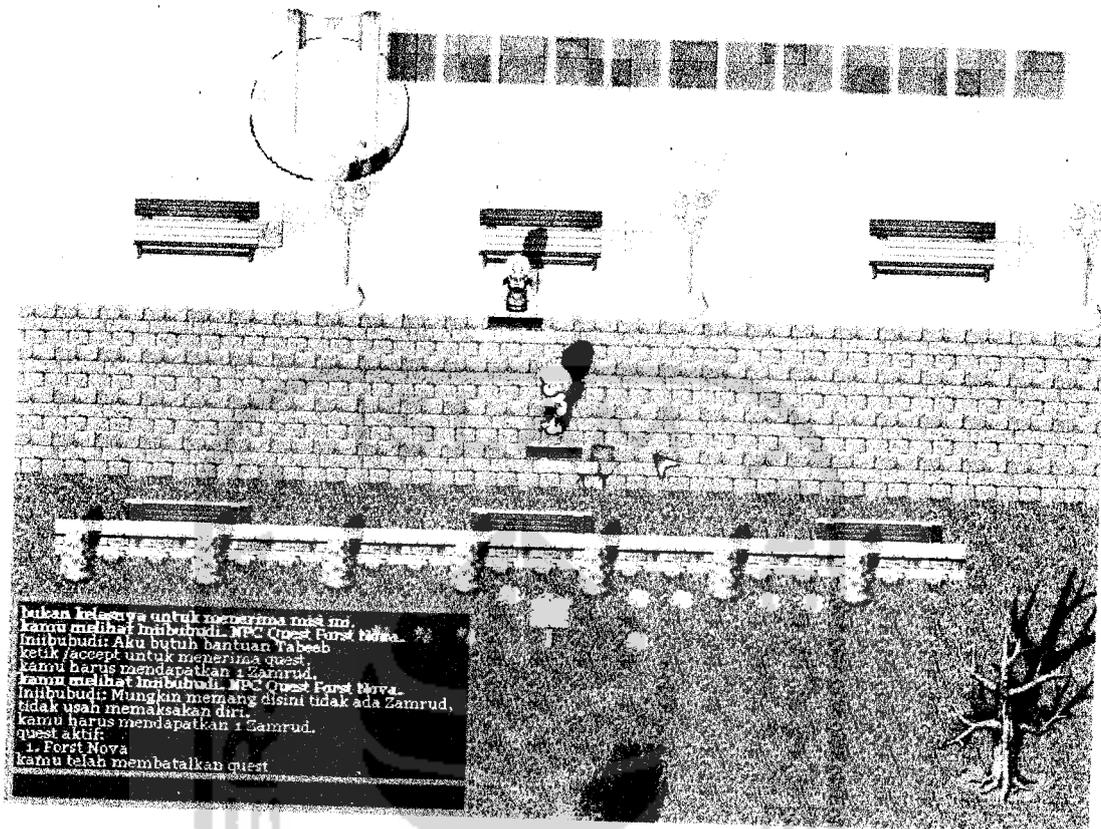
Pemain yang menyelesaikan *quest non-skill* akan mendapatkan hadiah sesuai dengan *quest* yang dijalaninya. Hadiah bisa berupa item, *experience* ataupun gold. Informasi Hadiah yang didapatkan diperlihatkan seperti pada gambar 4.48.



**Gambar 4.48** Jendela chat informasi menyelesaikan quest non-skill

#### 14. Membatalkan Quest

Pemain yang sedang menjalankan *quest* dapat membatalkan *quest* jika merasa *quest* terlalu sulit untuk dijalankan sehingga dapat menjalankan *quest* yang lain. Untuk membatalkan *quest* menggunakan perintah `/cancelquest 1`. Perintah ini hanya bisa digunakan jika ada *quest* yang sedang dijalani. Informasi pembatalan *quest* diperlihatkan seperti pada gambar 4.49.



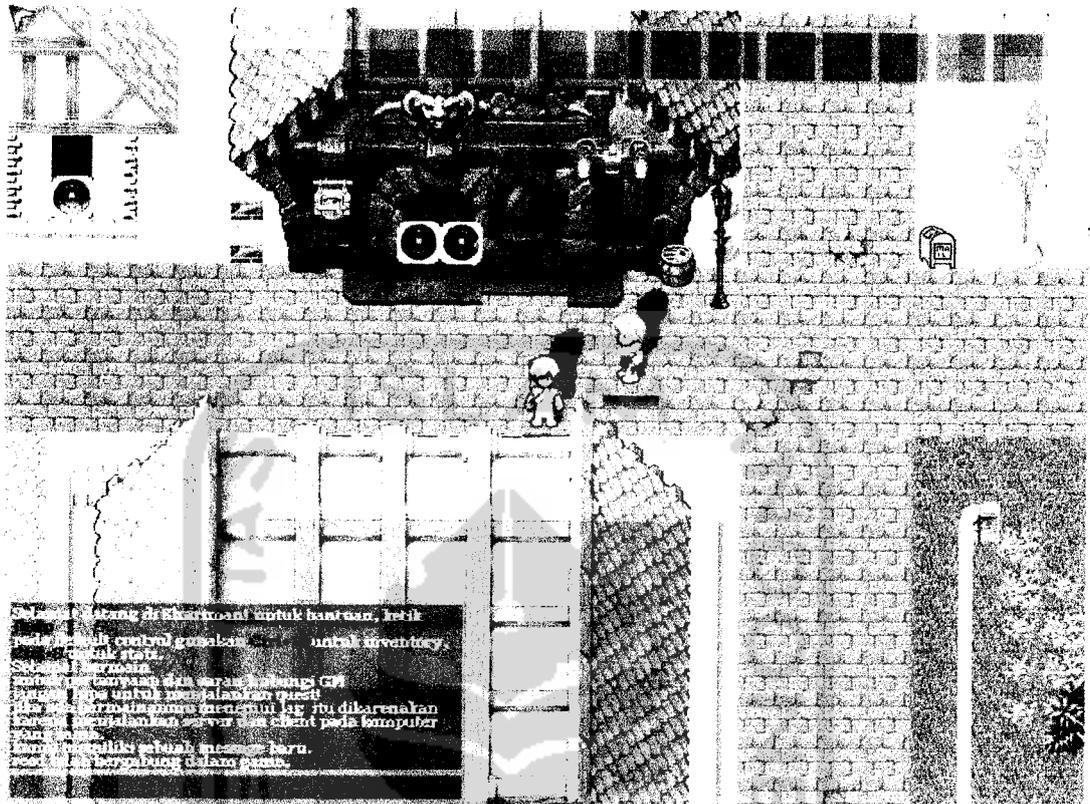
**Gambar 4.49** Jendela chat membatalkan quest yang belum selesai

#### 15. Informasi *Mail* Pada Saat Login

Setiap kali pemain melakukan *login*, server akan memberitahukan status *mailbox* pemain. Pada jendela *chat* akan diperlihatkan informasi *mail* baru pada *mail box*. Pada gambar 4.50 terlihat informasi pemain tidak memiliki *mail* baru pada *mail box*nya. Gambar 4.51 memperlihatkan ada satu *mail* baru pada *mailbox* pemain, sedangkan pada gambar 4.52 memperlihatkan ada beberapa *mail* baru pada *mailbox* pemain.

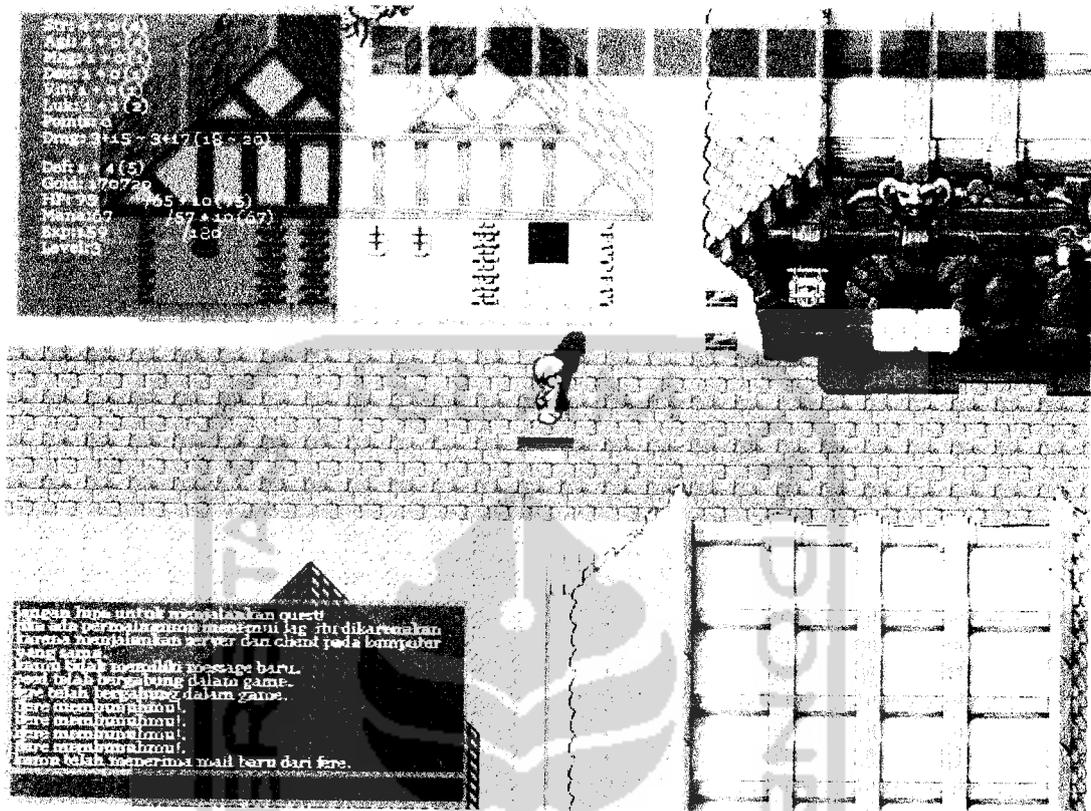


Gambar 4.50 Jendela chat informasi tidak ada mail pada saat login



Gambar 4.51 Jendela chat informasi ada 1 mail pada saat login





**Gambar 4.53** Jendela chat informasi ada mail baru pada saat bermain

#### 17. Mail Terkirim

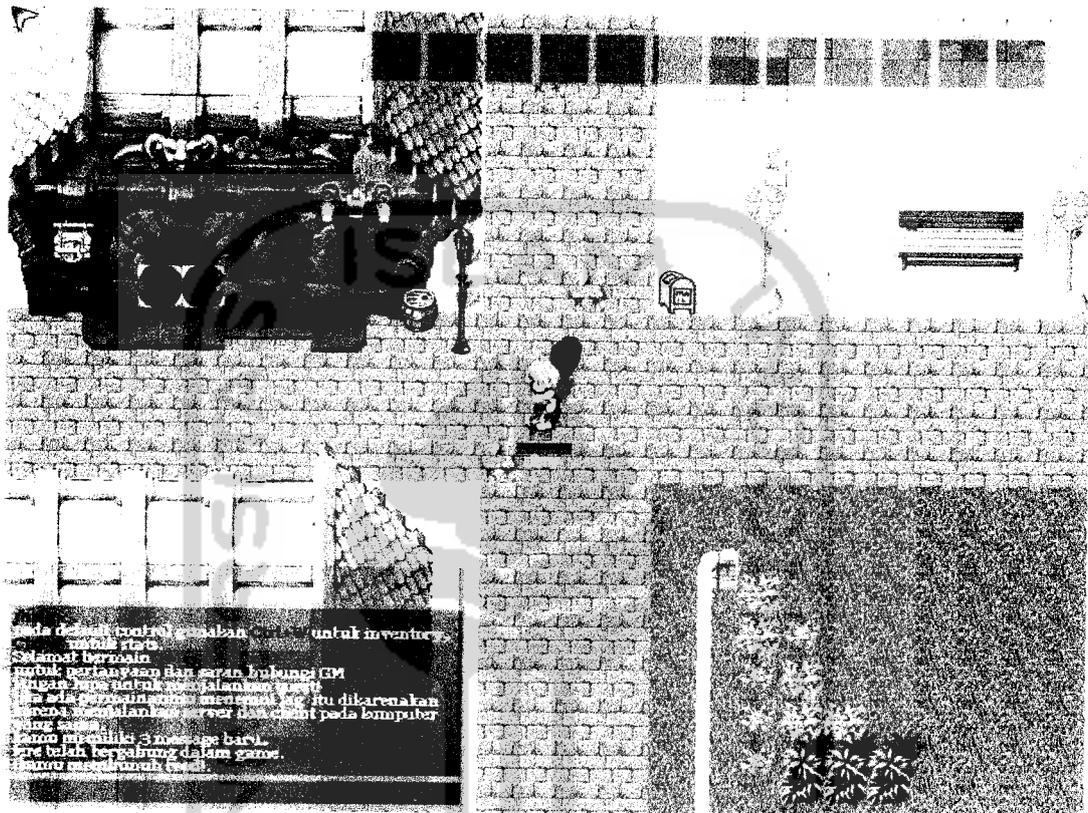
Untuk menuliskan *mail* pemain harus berada dekat *mail box*. Jika penulisan *mail* selesai dan *mail* telah terkirim akan disertai informasi bahwa *mail* telah terkirim dengan sukses. Informasi *mail* telah terkirim dapat dilihat pada gambar 4.54.

#### 18. Informasi Biaya Pengiriman Mail

Untuk setiap *mail* yang terkirim akan dikenakan biaya 20 gold sebagai biaya kirim dan langsung dikurangkan dari jumlah gold yang dibawa



dapat dilihat pada gambar 4.55 dan informasi pemain terbunuh pemain lain pada gambar 4.56.



Gambar 4.55 Jendela chat informasi membunuh pemain lain

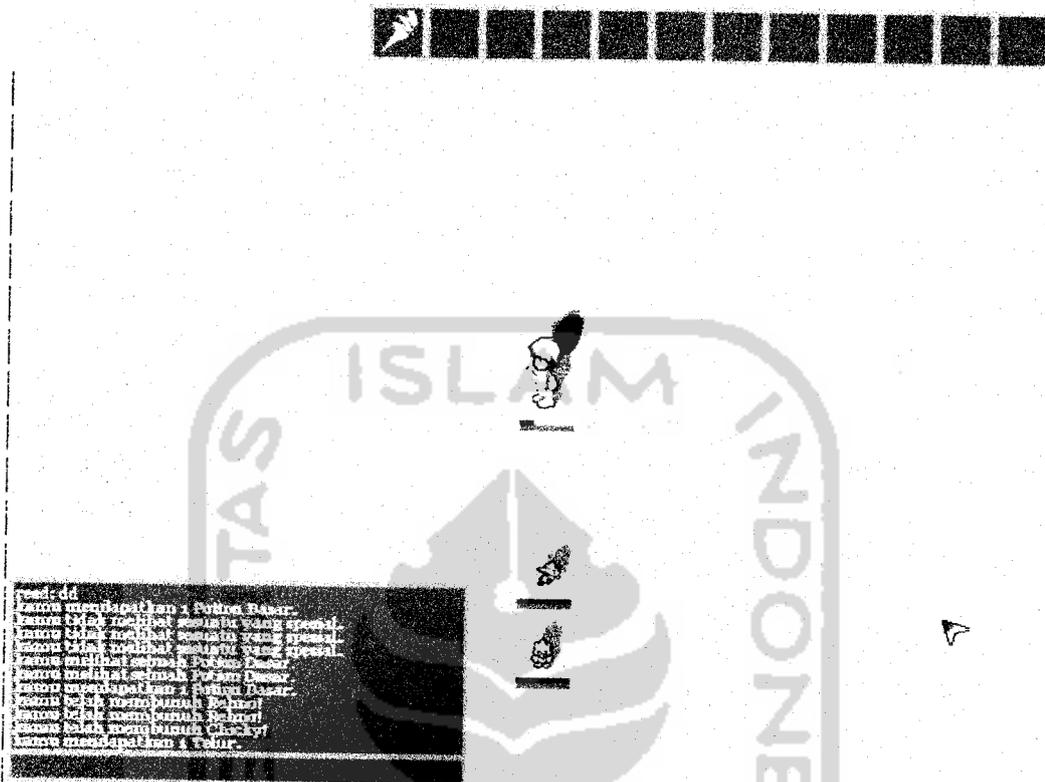


**Gambar 4.56** Jendela chat informasi terbunuh pemain lain

## 20. Mendapatkan Item Dari Tanah

Ketika monster mati maka akan menjatuhkan item ke tanah dengan kemungkinan yang telah ditentukan sever. Item yang terjatuh diatas tanah dapat dilihat deskripsinya dengan klik kanan tombol *mouse* seperti terlihat pada gambar 4.57. Untuk mengambil item tokoh harus berada diatas item dan pemain menekan tombol alt. Pada jendela *chat* akan terlihat informasi pengambilan item seperti terlihat pada gambar 4.57.



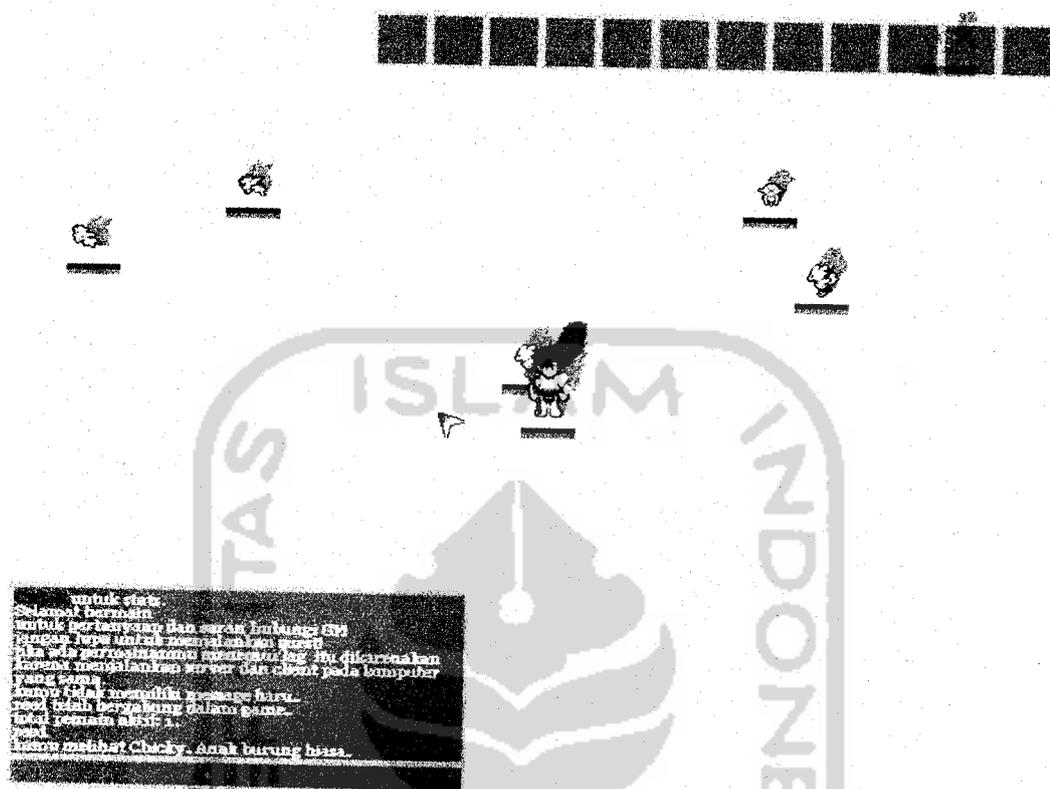


**Gambar 4.57** Jendela chat informasi melihat dan mengambil item dari tanah

#### 21. Melihat Deskripsi NPC

Pemain dapat melihat deskripsi NPC dengan klik kanan tombol *mouse*

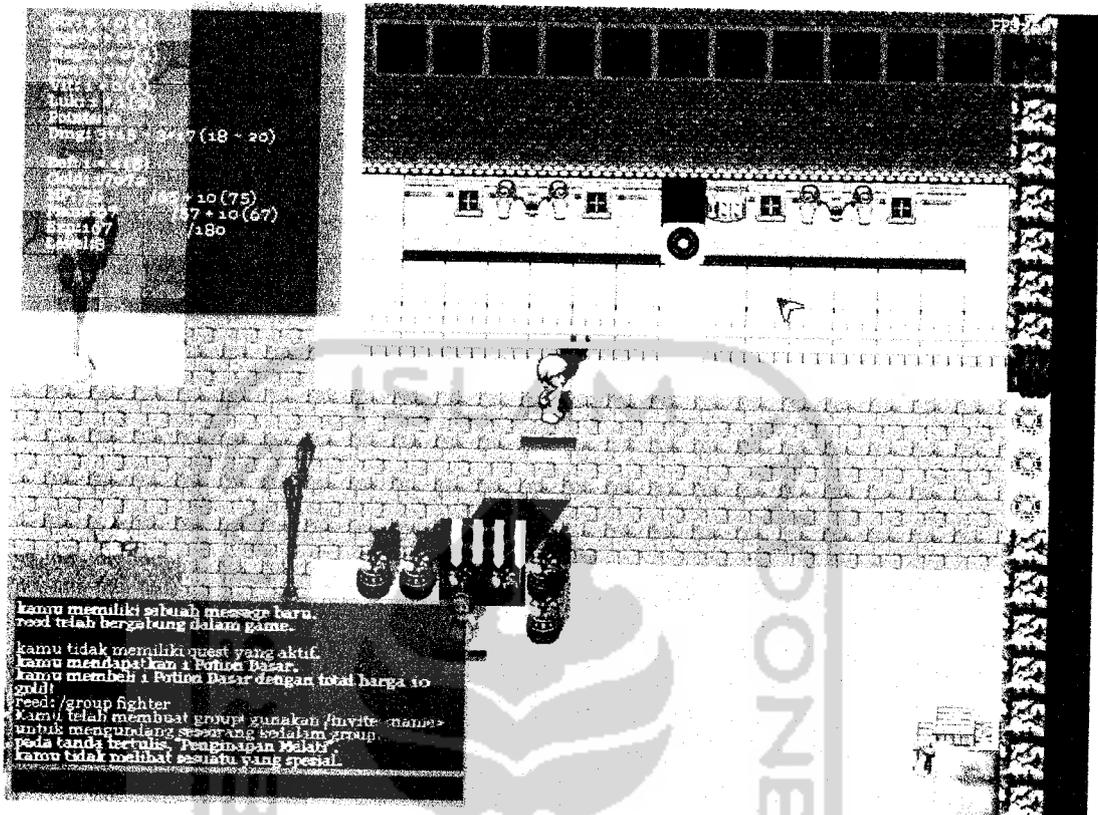
. Deskripsi NPC akan terlihat pada jendela *chat* seperti terlihat pada gambar 4.58.



**Gambar 4.58** Jendela chat informasi melihat deskripsi NPC

## 22. Melihat Tanda Pengumuman

Pemain dapat melihat informasi pada papan pengumuman ataupun pada plang dengan klik kanan tombol *mouse* . Deskripsi papan pengumuman atau plang akan terlihat pada jendela *chat* seperti terlihat pada gambar 4.59.

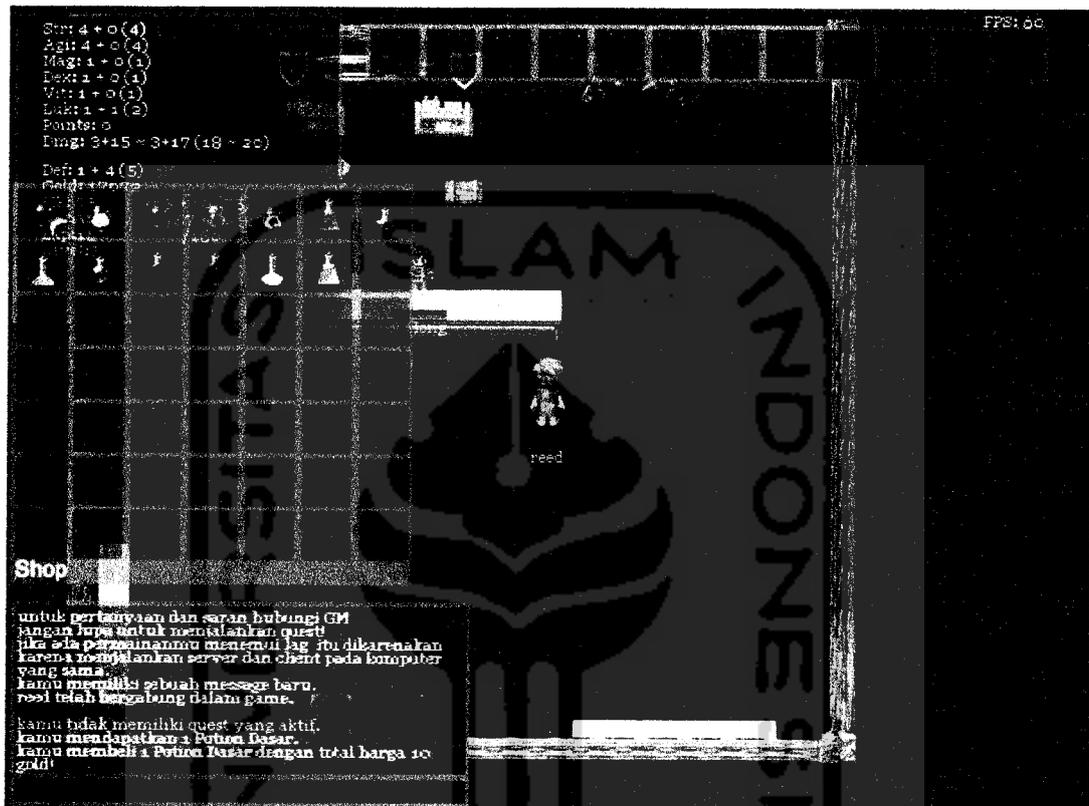


**Gambar 4.59** Jendela chat informasi melihat papan atau plang pengumuman

### 23. Membeli Barang Dari NPC Shop

Pemain yang membutuhkan item untuk digunakan pada tokohnya dapat membeli item pada *NPC shop* yang telah ditentukan. *NPC shop* berada pada toko ataupun ditempat-tempat lain yang telah ditentukan. *NPC shop* tidak bergerak. Pemain dapat mengakses jendela *NPC shop* dengan klik kanan tombol *mouse* pada *NPC shop*. Pembelian dapat dilakukan dengan klik kiri tombol *mouse* pada item yang dimaksud, ataupun menahan klik kanan tombol *mouse* dan *drag* pada item menuju jendela *inventory* pemain untuk pembelian lebih dari satu dengan memasukkan jumlah yang diinginkan. Transaksi

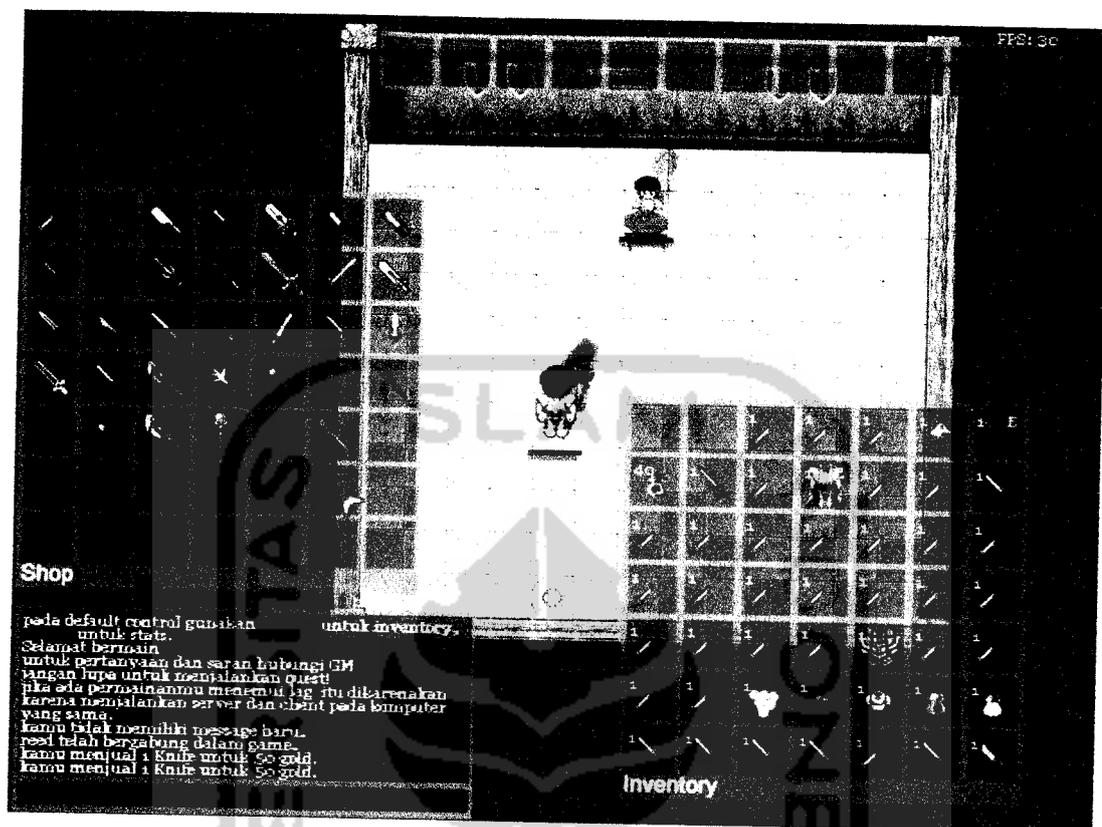
pembelian akan diinformasikan pada jendela *chat* seperti terlihat pada gambar 4.60.



**Gambar 4.60** Jendela chat informasi membeli barang dari NPC shop

#### 24. Menjual Barang Pada Npc Shop

Proses transaksi penjualan item identik dengan proses transaksi pembelian dengan membalik tujuan dari jendela *inventory* pemain menuju jendela NPC *shop*. Proses transaksi penjualan ini dapat dilihat pada gambar 4.61.



Gambar 4.61 Jendela chat informasi menjual barang ke NPC shop

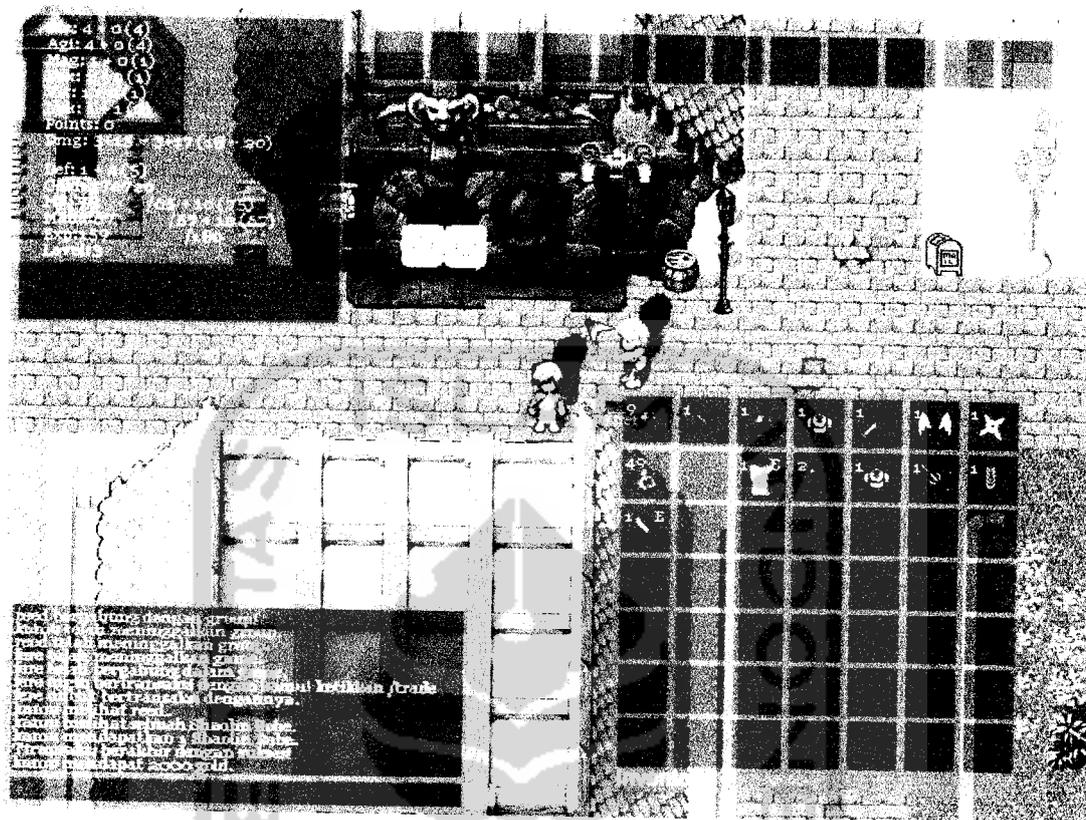
## 25. Bertransaksi Dengan Pemain Lain

Pemain dapat bertransaksi antar player character dengan mengetikkan perintah `/trade nama_yang_diajak_bertransaksi` pada jendela *chat* pada tiap-tiap pemain yang akan melakukan transaksi. Informasi untuk bertransaksi dengan pemain lain dapat dilihat pada gambar 4.62. Sedangkan informasi untuk membalas ajakan transaksi pada pemain yang diajak bertransaksi pada gambar 4.63.









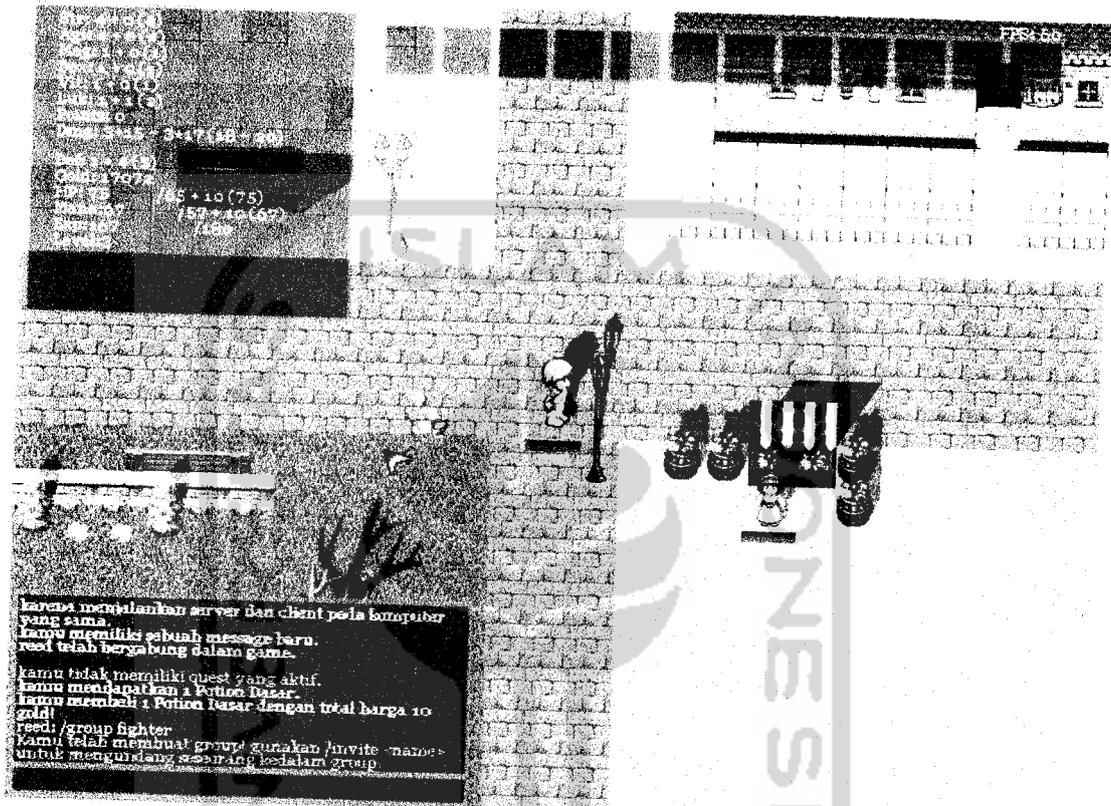
**Gambar 4.65** Jendela chat informasi transaksi sukses (menjual item mendapatkan gold)

## 27. Membuat Group

Pemain yang menginginkan untuk membuat group dapat menggunakan perintah `/creategroup nama_group`. Satu group hanya bisa berisi delapan pemain, sebagai ketua adalah pembuat group. Pembagian *experience* perorangan dalam group adalah *experience* yang didapatkan dari membunuh monster dibagi jumlah pemain satu grup dalam jarak dua layar ditambah bonus satu *experience*. Untuk mengajak pemain lain bergabung

kedalam group dengan menggunakan perintah `/invite nama_pemain`.

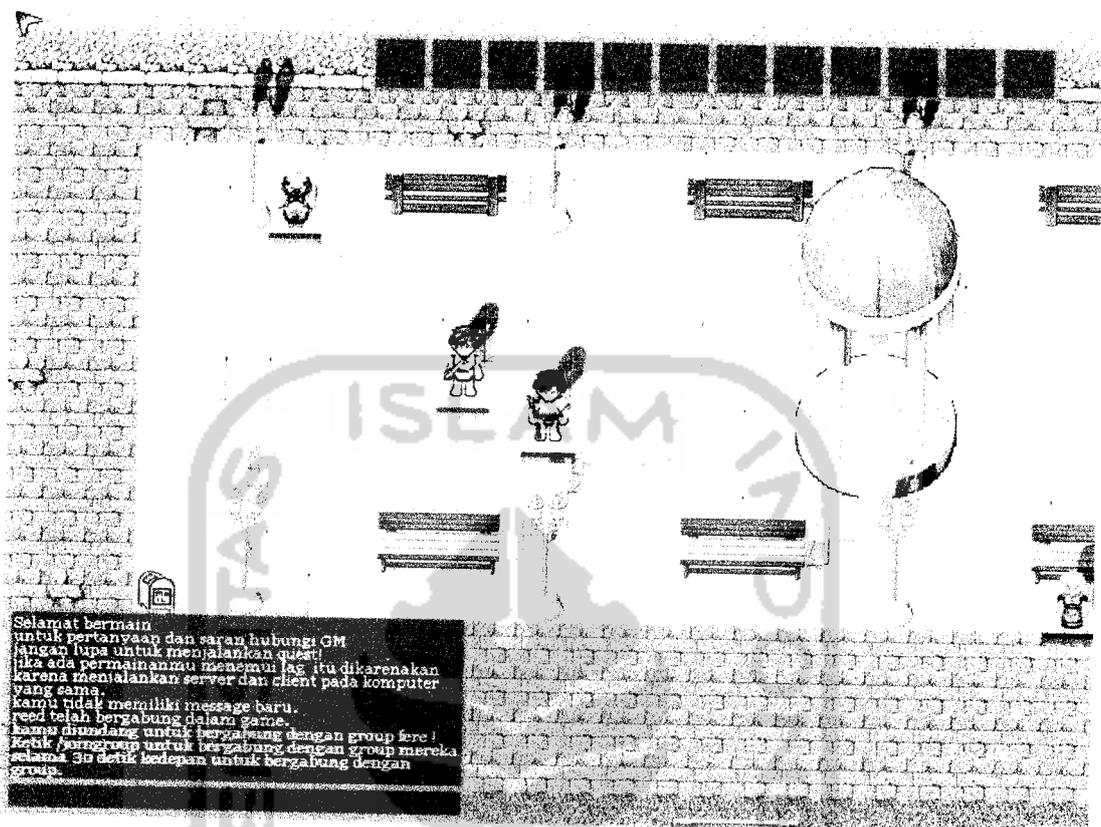
Informasi pembuatan group dapat dilihat pada gambar 4.66.



**Gambar 4.66** Jendela chat informasi membuat group

## 28. Ajakan Bergabung Dalam Group

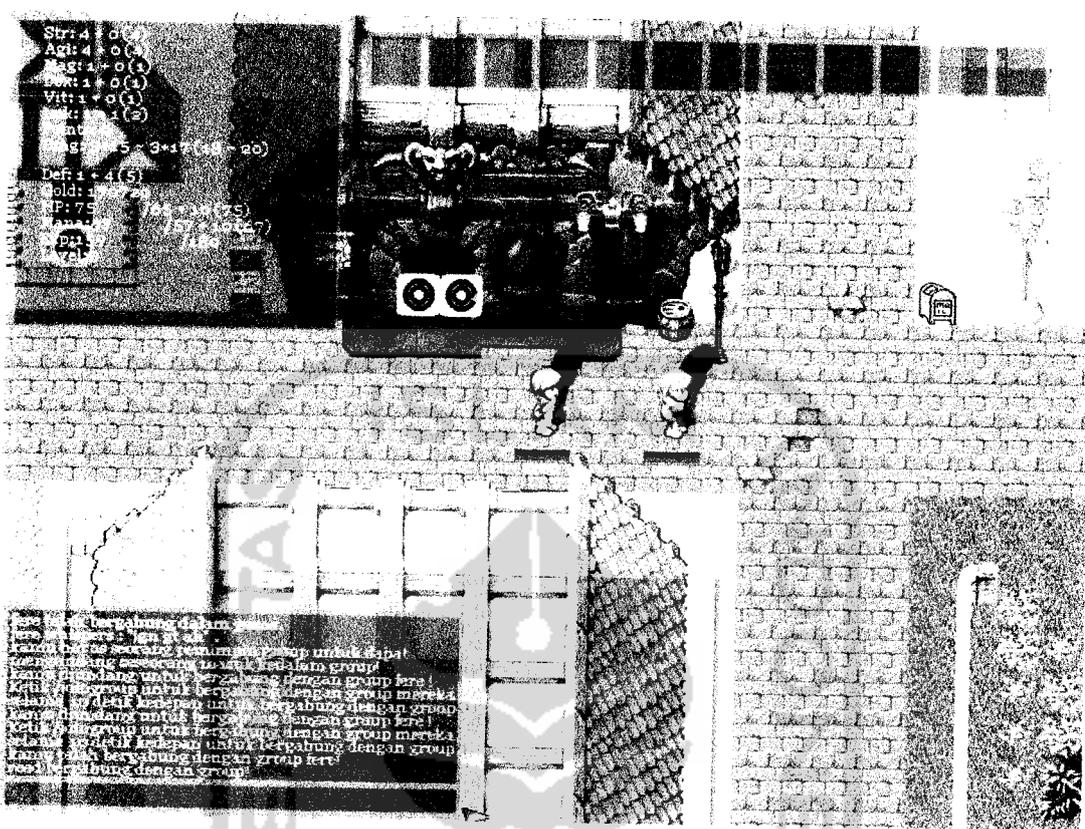
Pemain yang diajak bergabung kedalam suatu group akan menerima informasi seperti terlihat pada gambar 4.67. Untuk menerima ajakan bergabung dengan perintah `/joingroup` dalam jarak waktu 30 detik sejak diperlihatkannya informasi ajakan bergabung dengan group pada jendela chat.



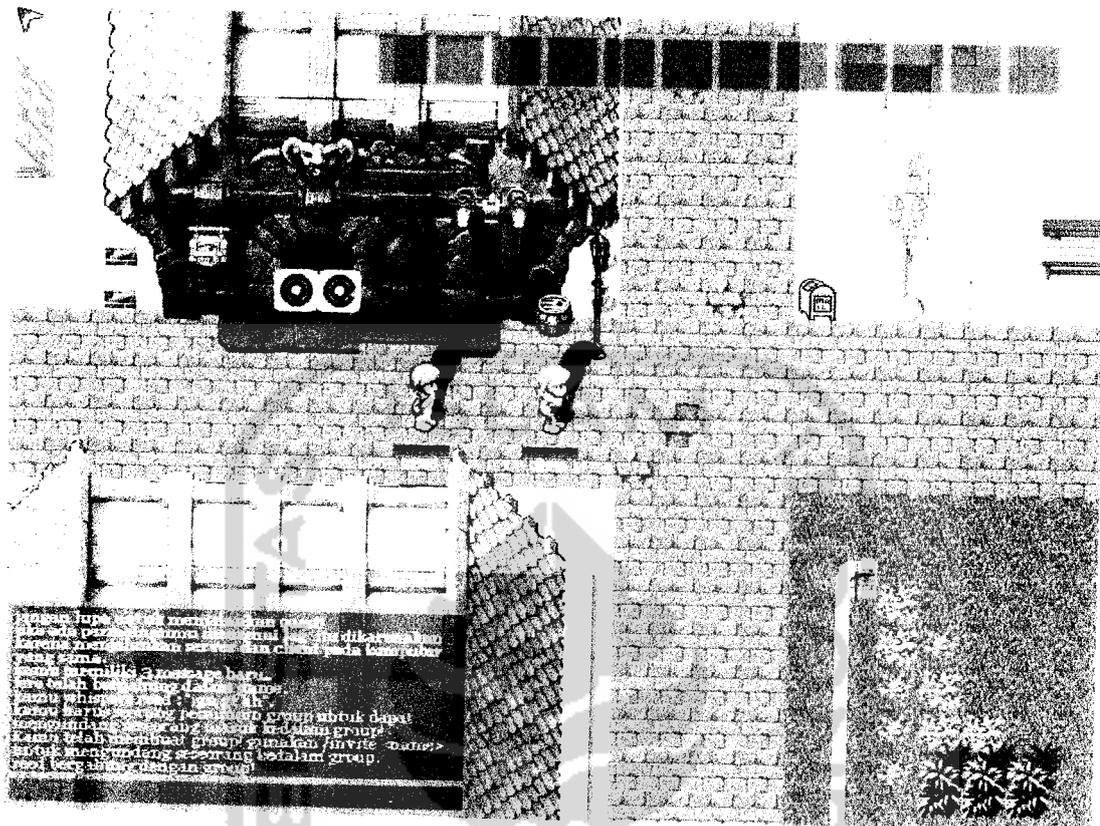
**Gambar 4.67** Jendela chat informasi ajakan bergabung dengan group

## 29. Pemain Bergabung Dalam Group

Pemain yang menerima ajakan bergabung akan mendapatkan informasi bahwa pemain telah bergabung dalam group. Lihat pada gambar 4.68. Sedangkan bagi pemain yang sebelumnya telah bergabung dengan group tersebut akan mendapatkan informasi bahwa ada pemain baru yang bergabung dengan group. Seperti terlihat pada gambar 4.69.



Gambar 4.68 Jendela chat informasi pemain bergabung dengan group (pemain baru bergabung)

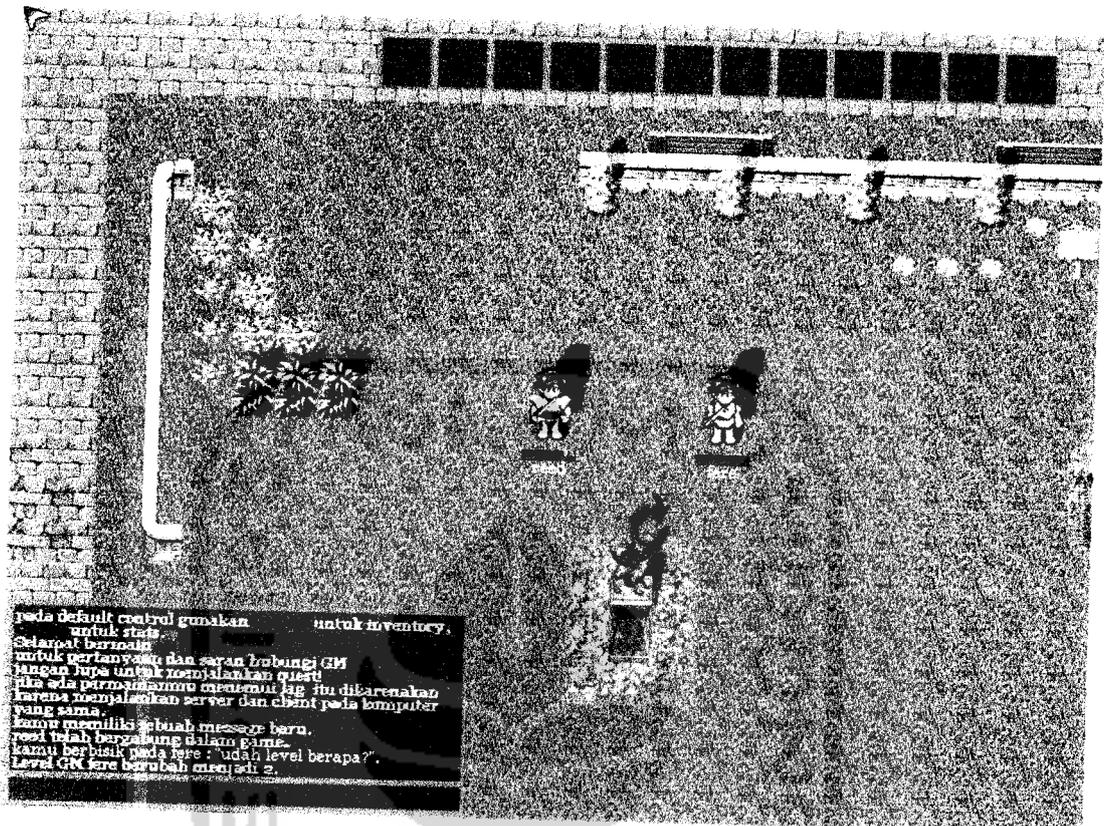


**Gambar 4.69** Jendela chat informasi pemain bergabung dengan group (pemain dalam group)

### 30. Pemain Keluar Dari Group

Pemain yang menginginkan keluar dari group dapat menggunakan perintah `/leavegroup` atau keluar dari permainan akan secara otomatis keluar dari group. Jika ketua group menggunakan perintah `/leavegroup` ataupun keluar dari permainan maka group secara otomatis akan bubar. Informasi yang diperlihatkan kepada pemain yang keluar group dapat dilihat pada gambar 4.70. Sedangkan pada pemain yang masih berada dalam group akan mendapatkan informasi seperti terlihat pada gambar 4.71.

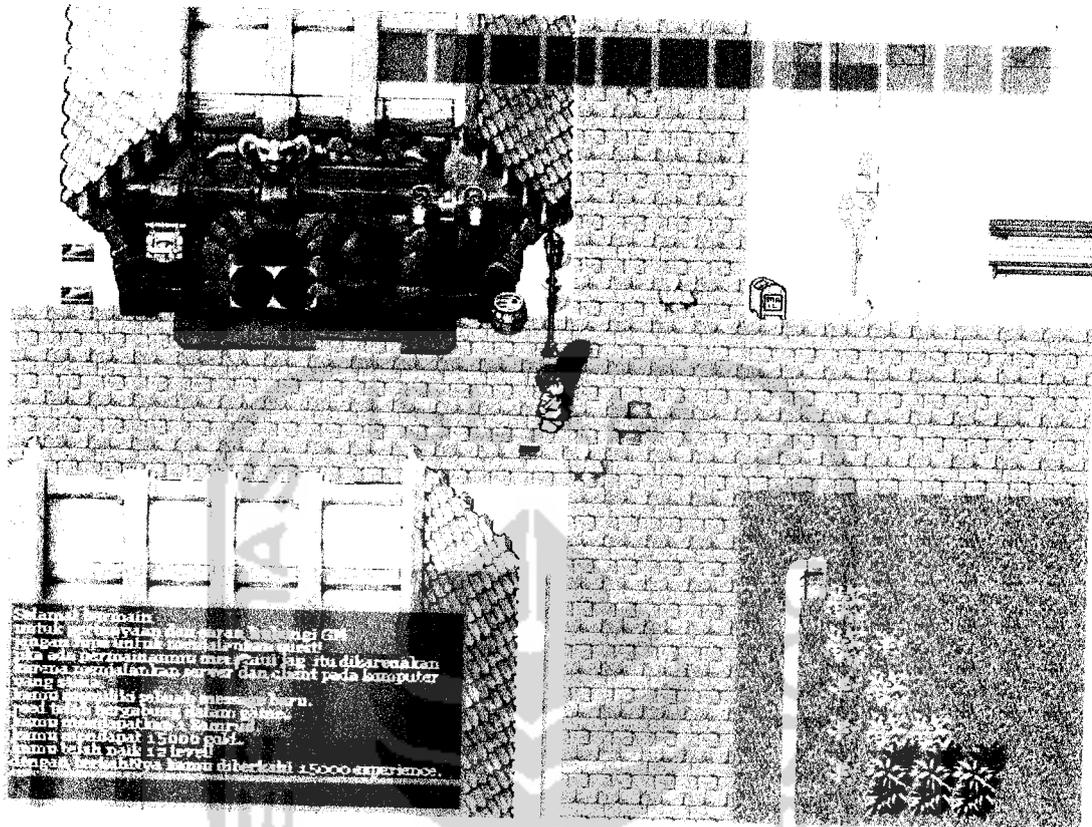




**Gambar 4.72** Jendela chat informasi GM merubah pemain lain menjadi GM

### 32. Pemain Diberikan *Experience* Oleh GM

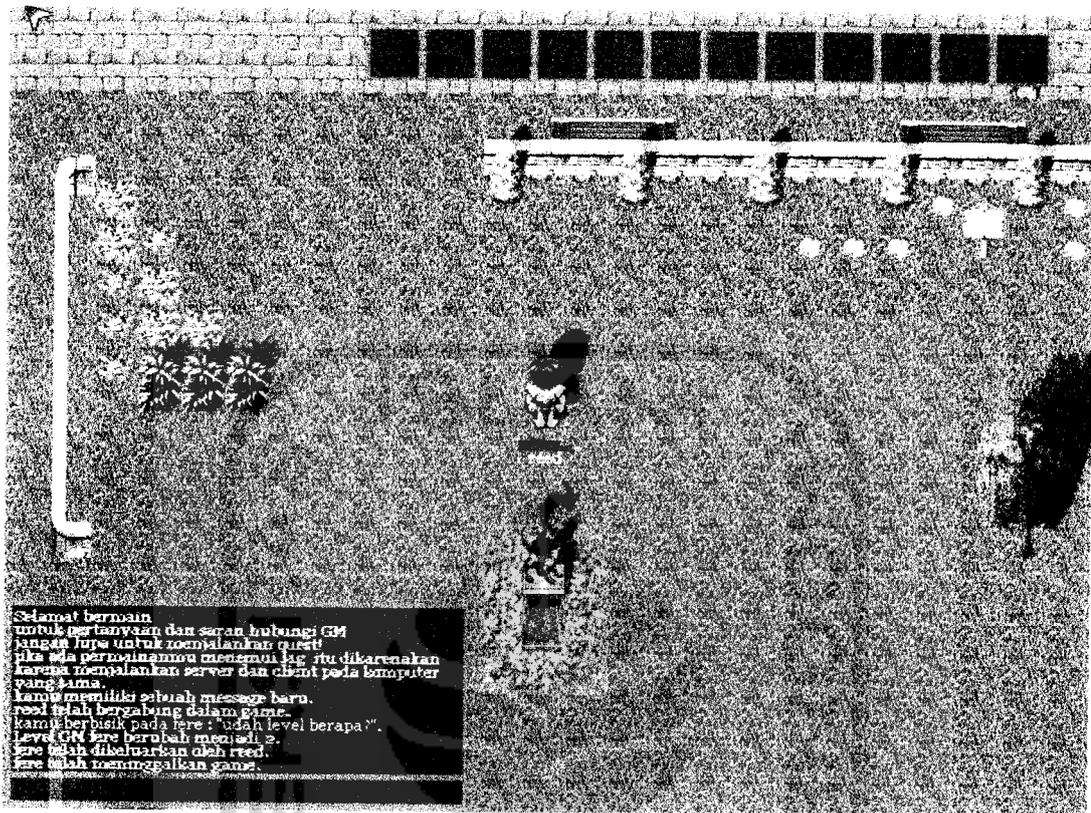
GM berhak untuk memberikan *experience* kepada pemain lain dengan menggunakan perintah `/raise nama_pemain experience` yang diberikan. Informasi pemberian *experience* ini dapat dilihat pada jendela *chat* pemain yang dinaikkan *experience* nya seperti terlihat pada gambar 4.73.



**Gambar 4.73** Jendela chat informasi pemain diberi experience oleh GM

### 33. GM Mengeluarkan Pemain Dari Permainan Dengan Paksa

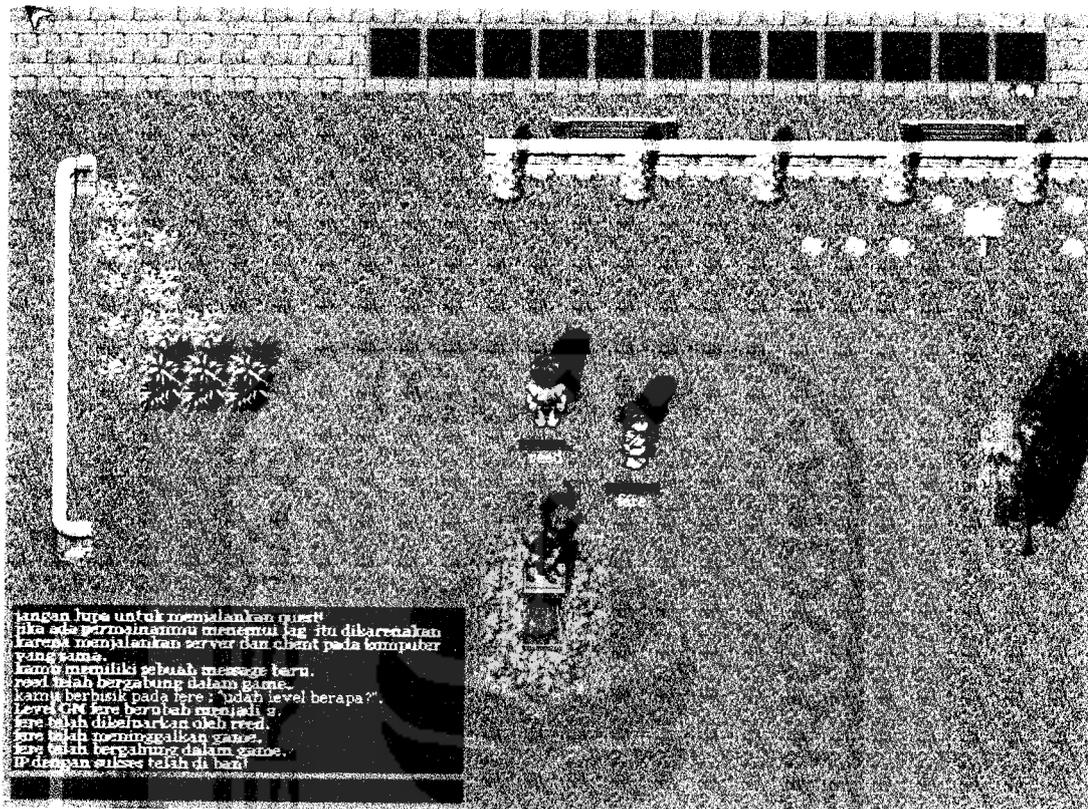
GM memiliki hak untuk mengeluarkan pemain dari permainan secara sepihak dengan menggunakan perintah `/kick nama_pemain`. Pemain yang dikeluarkan masih bisa bergabung kembali dengan permainan selama GM belum memblokir alamat IP komputernya. Informasi pemain telah dikeluarkan oleh GM akan muncul pada jendela *chat* GM seperti pada gambar 4.74.



**Gambar 4.74** Jendela chat informasi GM mengeluarkan pemain dari permainan secara paksa (Kick)

#### 34. GM Memblokade Alamat IP

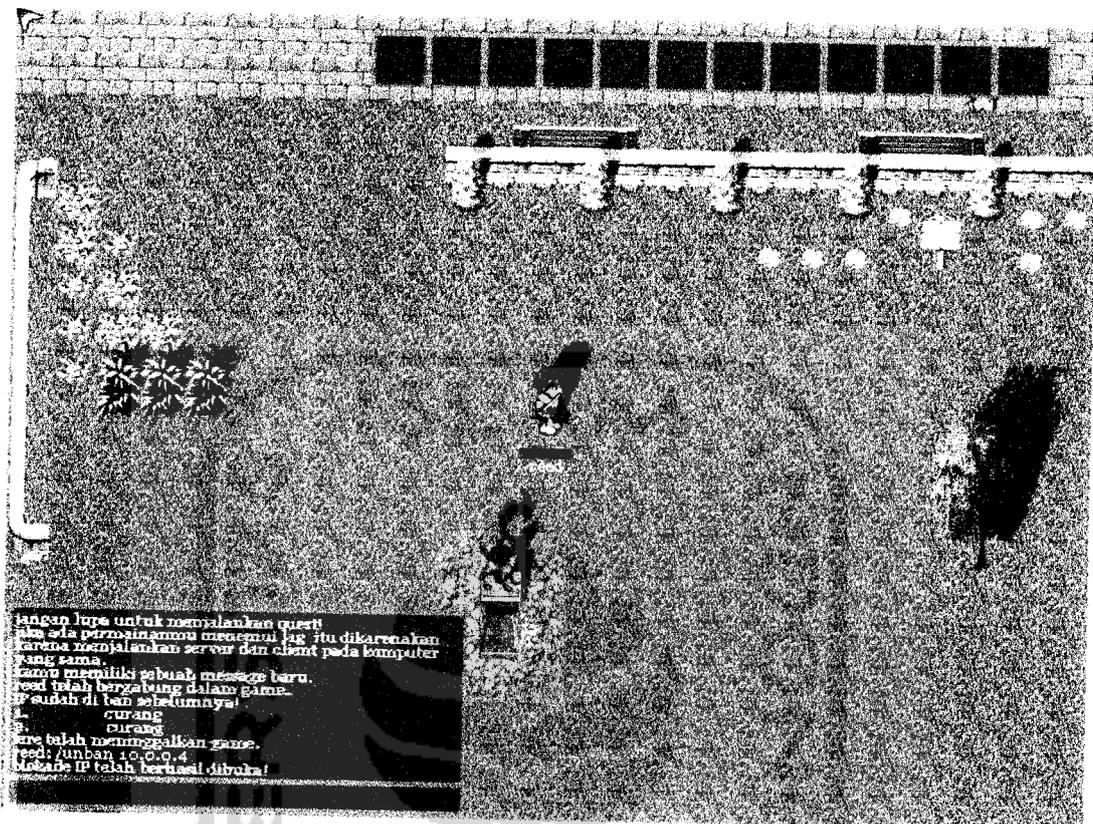
GM memiliki hak untuk memblokade alamat IP yang memiliki masalah ketika bermain. Untuk memblokade alamat IP menggunakan perintah `/banip alamat_IP alasan`. Informasi blokade alamat IP berhasil akan ditampilkan pada jendela *chat* GM seperti terlihat pada gambar 4.75.



**Gambar 4.75** Jendela chat informasi GM memblokade alamat IP

### 35. GM Membuka Blokade Alamat IP

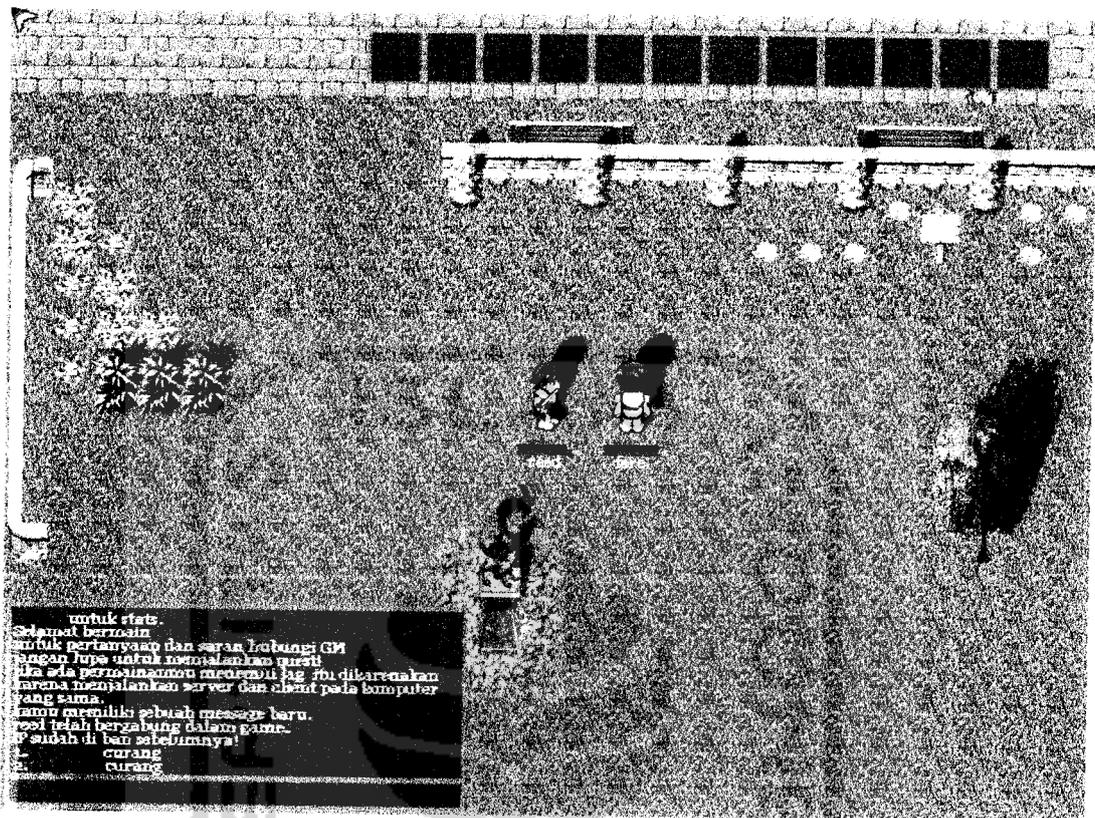
Untuk membuka blokade alamat IP menggunakan perintah `/unbanip` alamat\_IP. Informasi blokade IP berhasil dibuka akan terlihat pada jendela chat GM seperti terlihat pada gambar 4.76.



**Gambar 4.76** Jendela chat informasi GM membuka blokade alamat IP

### 36. Informasi Alamat IP Yang Diblokade

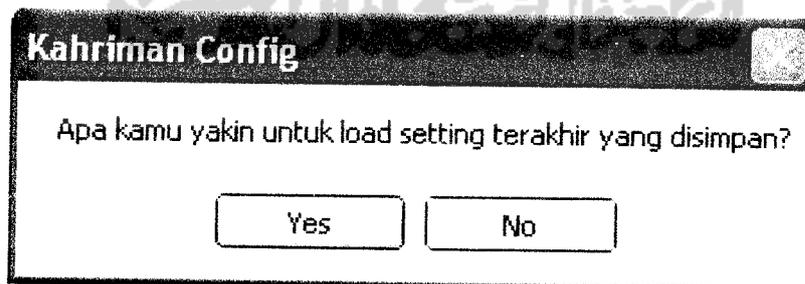
Untuk melihat alamat IP mana saja yang sedang diblokade dapat menggunakan perintah `/banlist`. Daftar informasi alamat IP dan alasan blokade dapat dilihat pada gambar 4.77.



**Gambar 4.77** Jendela chat informasi GM melihat daftar blokade alamat IP

### 37. Load Settingan Konfigurasi Terakhir

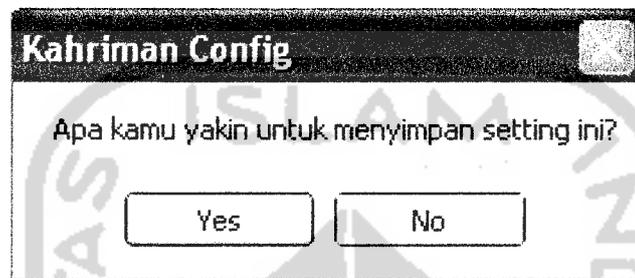
Pada aplikasi konfigurasi permainan jika akan load settingan terakhir yang disimpan akan memunculkan pesan seperti gambar 4.78.



**Gambar 4.78** Pesan peringatan untuk load settingan terakhir

### 38. Menyimpan Settingan Konfigurasi

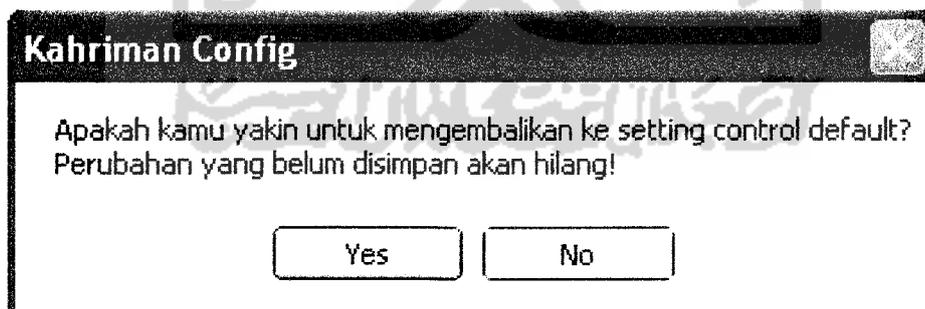
Pada aplikasi konfigurasi permainan jika akan menyimpan settingan yang telah dirubah akan memunculkan pesan seperti gambar 4.79.



**Gambar 4.79** Pesan peringatan untuk menyimpan setting saat ini

### 39. Load Settingan Konfigurasi Default

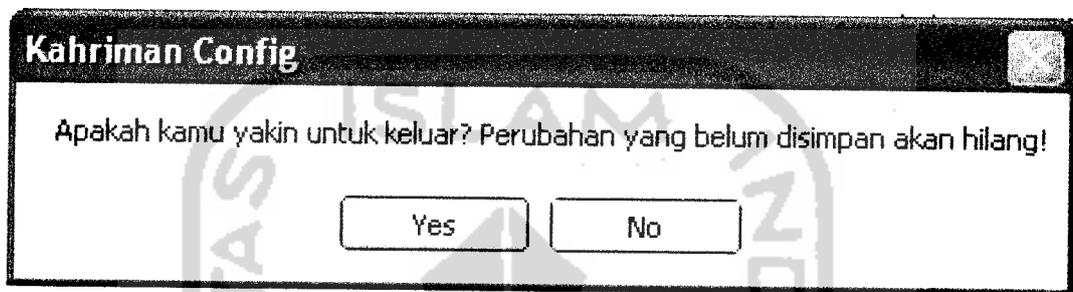
Pada aplikasi konfigurasi permainan jika akan load settingan default yang disarankan permainan akan memunculkan pesan seperti gambar 4.80. settingan yang berubah sebelumnya dan belum disimpan akan hilang jika load settingan default disetujui.



**Gambar 4.80** Pesan peringatan untuk mengembalikan ke settingan default

#### 40. Keluar Tanpa Menyimpan

Jika akan keluar dari aplikasi permainan tanpa menyimpan perubahan settingan akan memunculkan pesan seperti pada gambar 4.81.



**Gambar 4.81** Pesan untuk keluar aplikasi tanpa menyimpan settingan

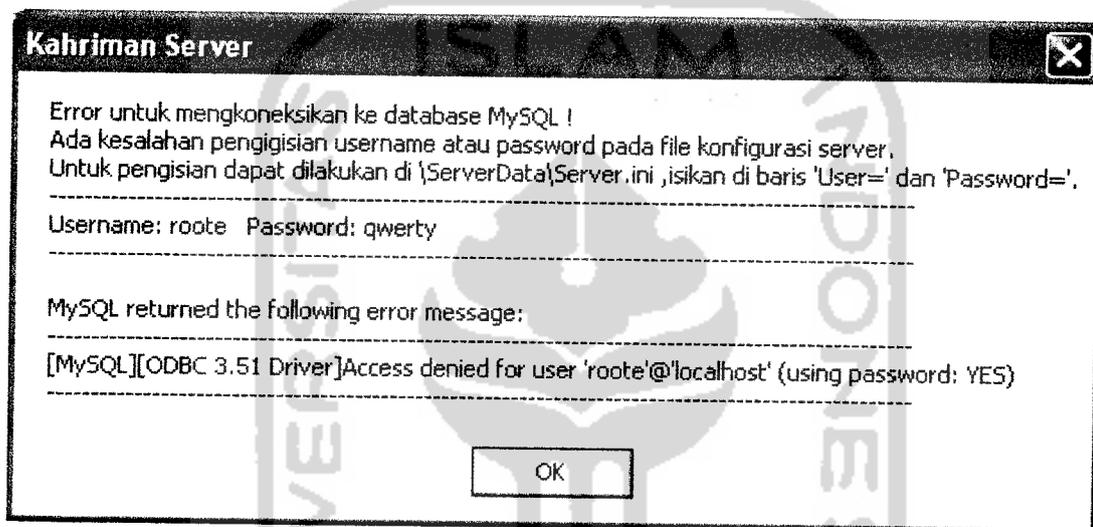
#### 4.6.2 Pengujian Tidak Normal

Pengujian tidak normal (*robust testing*) dilakukan dengan memberikan masukan yang menurut spesifikasi awal tidak diijinkan.

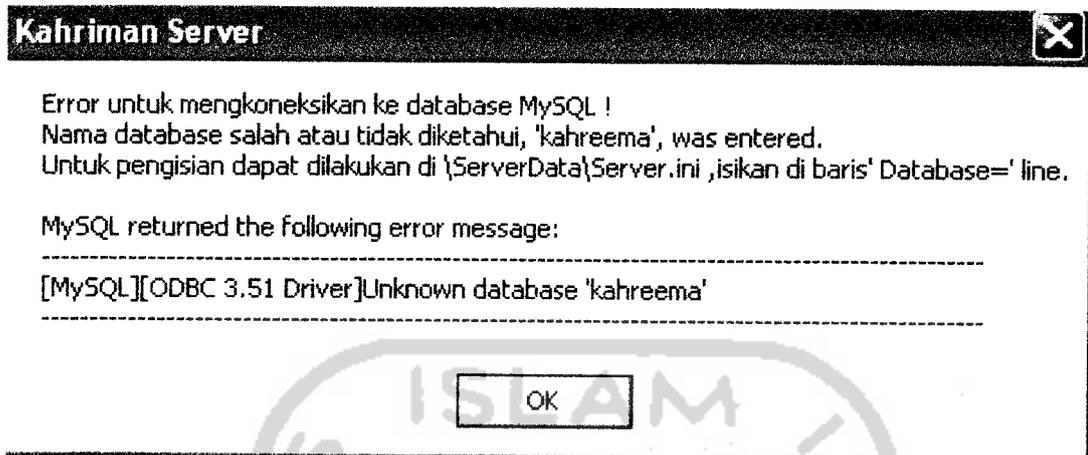
##### 1. Aplikasi Server Gagal Mengkoneksikan Dengan Sever MySQL

Pengujian tidak normal terhadap aplikasi server dilakukan pada uji konektivitas terhadap server MySQL. Pengujian dilakukan dengan lima kondisi yang berbeda. Kondisi pertama adalah kesalahan terhadap pengisian nilai pemain dan nilai password. Pesan kesalahan yang muncul dapat dilihat pada gambar 4.82. Kondisi kedua adalah kesalahan pengisian nama database dengan pesan kesalahan yang terlihat seperti gambar 4.83. Pada kondisi ketiga, pengujian dilakukan dengan kesalahan pada pengisian alamat ip server MySQL.

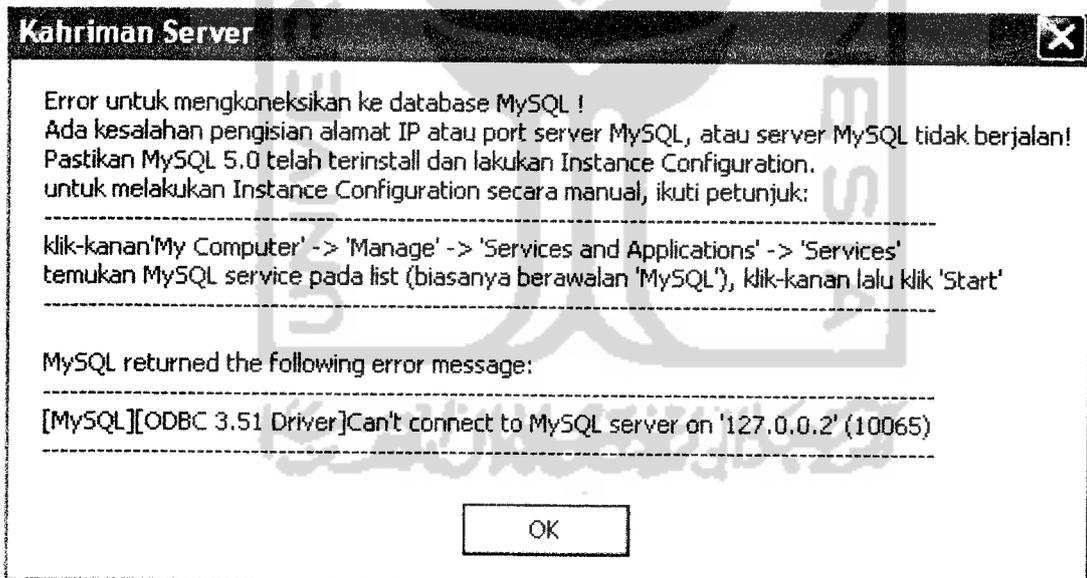
Pesan kesalahan akan muncul seperti gambar 4.84. Sedangkan kondisi keempat adalah menguji koneksi tanpa menggunakan driver ODBC versi 3.51, hasil uji ini memunculkan pesan kesalahan seperti pada gambar 4.85. Dan pengujian kelima menguji dengan database yang benar tetapi tidak memiliki tabel didalamnya, sehingga pesan kesalahan pada gambar 4.86 akan muncul.



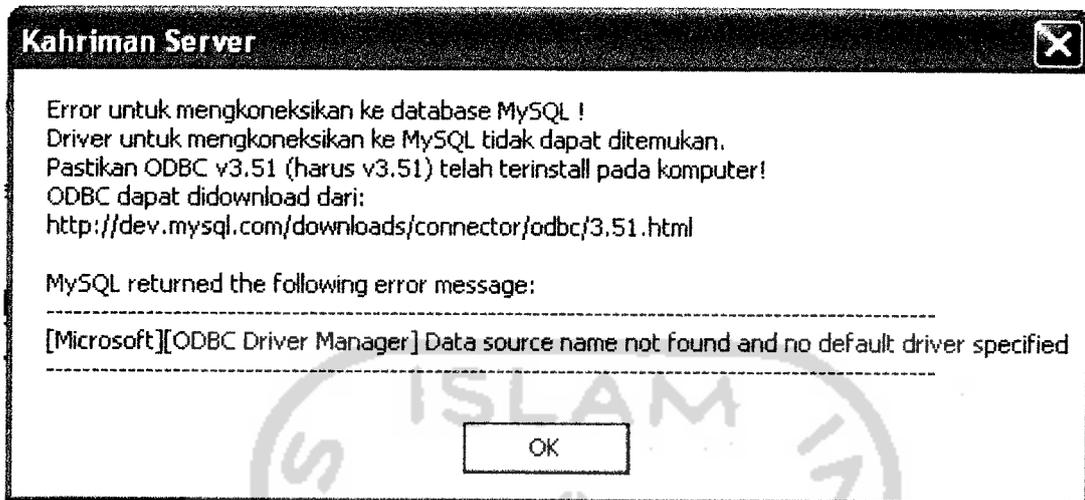
**Gambar 4.82** Pesan kesalahan pengisian nilai pemain dan password untuk koneksi MySQL pada konfigurasi server.ini



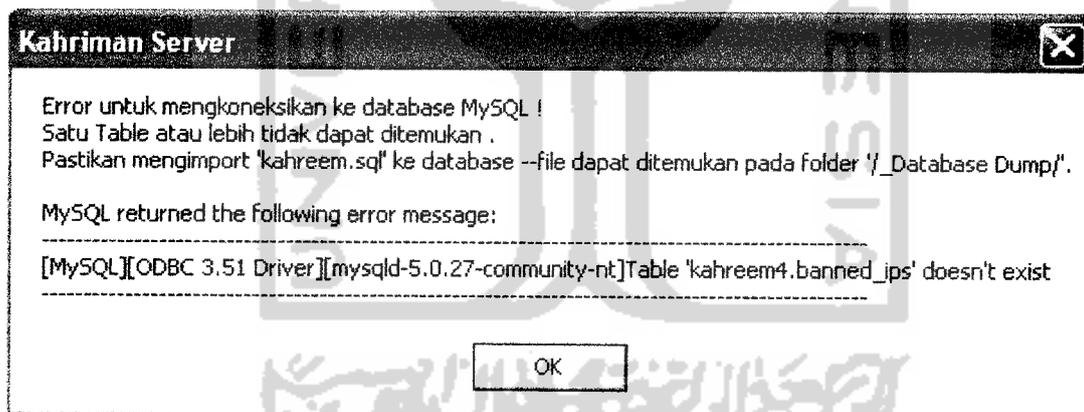
**Gambar 4.83** Pesan kesalahan pengisian nilai database untuk koneksi MySQL pada konfigurasi server.ini



**Gambar 4.84** Pesan kesalahan pengisian nilai alamat ip untuk koneksi MySQL pada konfigurasi server.ini



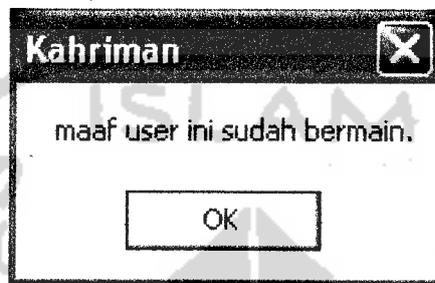
**Gambar 4.85** Pesan kesalahan tidak mendeteksi driver ODBC untuk koneksi MySQL pada konfigurasi server.ini



**Gambar 4.86** Pesan kesalahan tabel tidak ditemukan pada database untuk koneksi MySQL pada konfigurasi server.ini

## 2. Login Dengan Akun Yang Sedang Bermain

Pengujian pada aplikasi client permainan dilakukan mulai dari menu login. Pada menu login diujikan dengan memasuki permainan dengan akun yang sedang bermain. Hasilnya akan muncul pesan seperti pada gambar 4.87.



**Gambar 4.87** Pesan kesalahan Login dengan akun yang sedang bermain

## 3. Login Dengan Kata Sandi Yang Salah

Pengujian menu login selanjutnya adalah login dengan akun yang benar tetapi kesalahan ada pada kata sandi. Pesan yang keluar seperti yang terlihat pada gambar 4.88.



**Gambar 4.88** Pesan kesalahan Login dengan kata sandi salah

## 4. Login Dengan Akun Yang Tidak Terdaftar

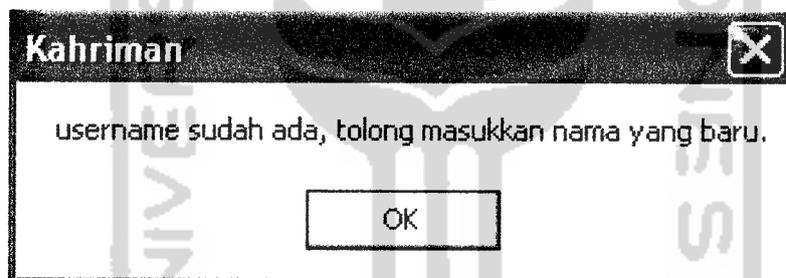
Pengujian dilanjutkan dengan login menggunakan akun yang salah. Akan muncul pesan seperti pada gambar 4.89.



**Gambar 4.89** Pesan kesalahan login dengan pemain yang belum terdaftar

5. Membuat Akun Baru Dengan Nama Yang Telah Terdaftar

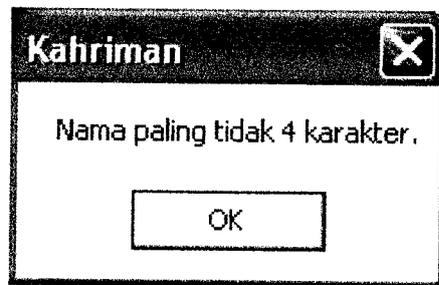
Pada menu pembuatan akun baru dilakukan pengujian dengan membuat akun baru dengan nama pemain yang sudah ada dan telah terdaftar sebelumnya. Pesan kesalahan yang keluar seperti pada gambar 4.90.



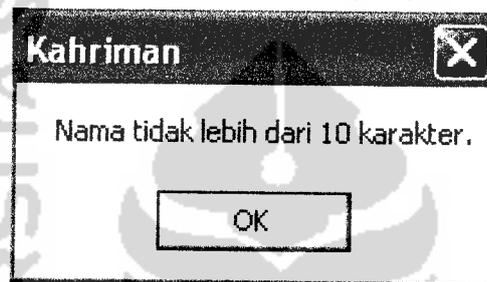
**Gambar 4.90** Pesan kesalahan membuat akun baru menggunakan nama yang sudah terdaftar

6. Membuat Akun Baru Dengan Nama Tidak Sesuai Aturan

Jika melakukan pembuatan akun baru dengan pengisian nama pemain yang tidak memenuhi persyaratan akan memunculkan pesan seperti gambar 4.91 untuk pengisian nama pemain kurang dari empat karakter dan jika melebihi sepuluh karakter akan muncul pesan kesalahan seperti gambar 4.92.



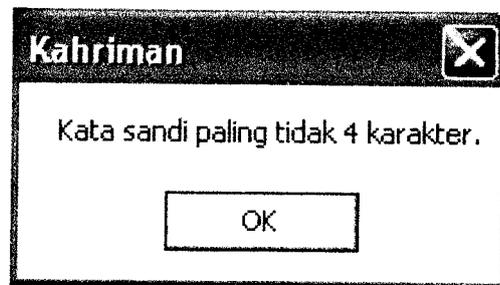
**Gambar 4.91** Pesan kesalahan membuat akun baru dengan panjang nama kurang dari 4 karakter



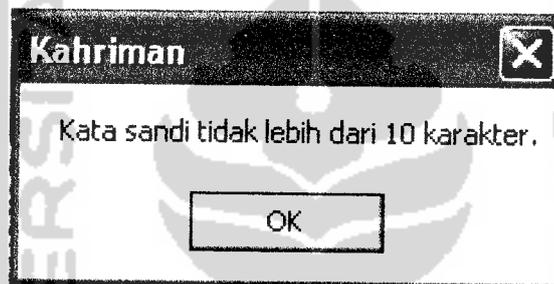
**Gambar 4.92** Pesan kesalahan membuat akun baru dengan panjang nama lebih dari 10 karakter

#### 7. Membuat Akun Baru Dengan Kata Sandi Tidak Sesuai Aturan

Pembuatan akun baru dengan pengisian kata sandi yang tidak memenuhi persyaratan akan memunculkan pesan seperti gambar 4.93 untuk pengisian kata sandi kurang dari empat karakter dan jika melebihi sepuluh karakter akan muncul pesan kesalahan seperti gambar 4.94.



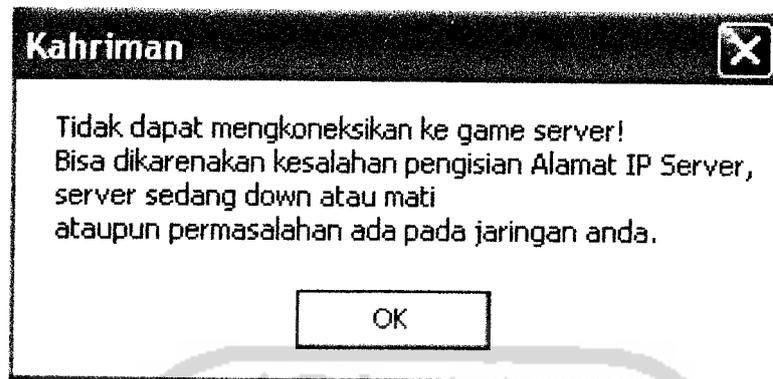
**Gambar 4.93** Pesan kesalahan membuat akun baru dengan panjang kata sandi kurang dari 4 karakter



**Gambar 4.94** Pesan kesalahan membuat akun baru dengan panjang kata sandi lebih dari 10 karakter

#### 8. Login Dengan Format Alamat Ip Server Yang Salah

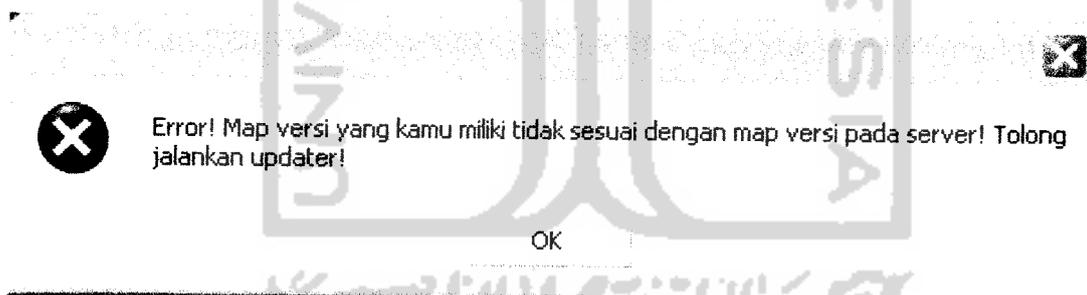
Login dengan tidak mengisikan alamat IP server atau salah mengisikan alamat server akan memunculkan pesan kesalahan seperti gambar 4.95. Pesan ini berlaku juga untuk kondisi server yang tidak berjalan tetapi pemain mencoba untuk memasuki permainan dan kondisi jika ada permasalahan jaringan yang mengakibatkan client tidak dapat berkomunikasi dengan server.



**Gambar 4.95** Pesan kesalahan pengisian alamat ip server pada saat login

#### 9. Versi Map Tidak Sama Dengan Versi Map Pada Server

Pengujian tahap berikut menggunakan aplikasi client versi pengembangan yang mencoba dihubungkan dengan aplikasi server versi lengkap. Karena perbedaan versi map akan memunculkan pesan kesalahan seperti pada gambar 4.96.

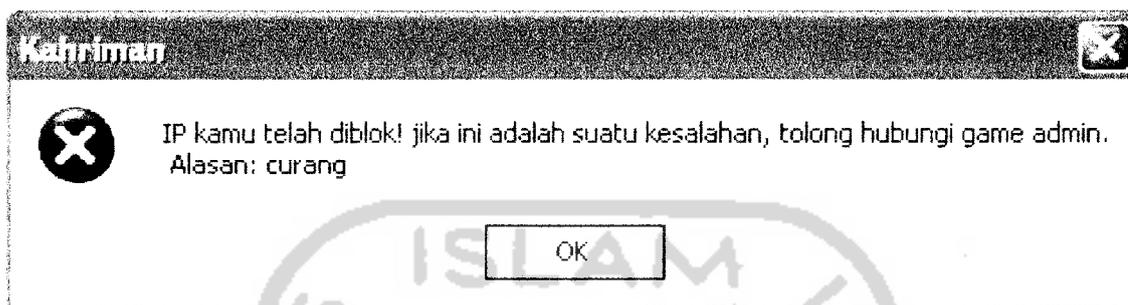


**Gambar 4.96** Pesan kesalahan ada perbedaan versi map client dengan server

#### 10. Alamat IP Yang Sedang Diblokade Mencoba Memasuki Permainan

Pengujian selanjutnya adalah mencoba menghubungkan komputer client yang telah diblokade alamat IP-nya oleh GM. Akan muncul pesan bahwa

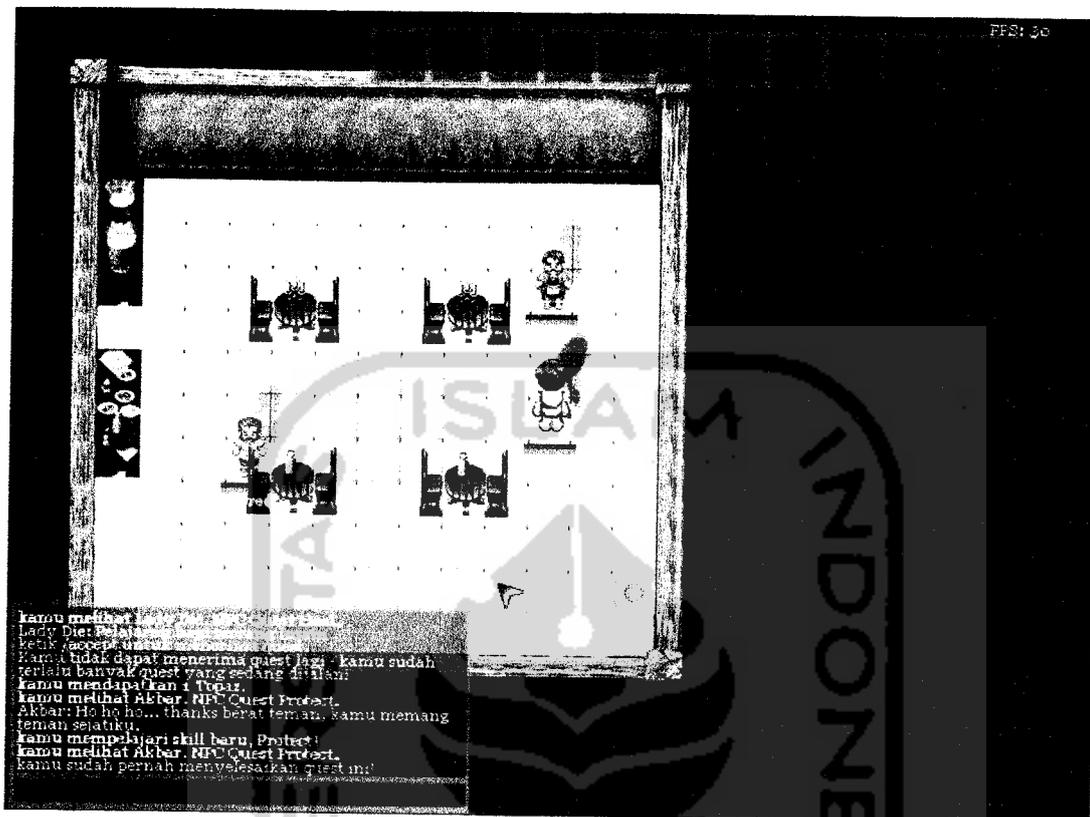
alamat IP telah diblokade berikut dengan alasannya seperti terlihat pada gambar 4.97.



**Gambar 4.97** Pesan peringatan alamat ip telah diblok

#### 11. Quest Yang Sudah Pernah Diselesaikan Tidak Bisa Dilakukan Lagi

Pengujian selanjutnya dilakukan pada permainan. Pesan kesalahan pada permainan akan ditampilkan pada jendela chat. Seperti terlihat pada gambar 4.98 terlihat pengujian dilakukan terhadap *quest* yang tidak dapat diulang dan sudah pernah diselesaikan. Pengujian dilakukan dengan mencoba menerima *quest* yang tidak dapat diulang.



**Gambar 4.98** Pesan dalam permainan tidak dapat mengulang quest yang tidak bisa diulang

## 12. Tidak Bisa Menjalankan Lebih Dari Satu Quest

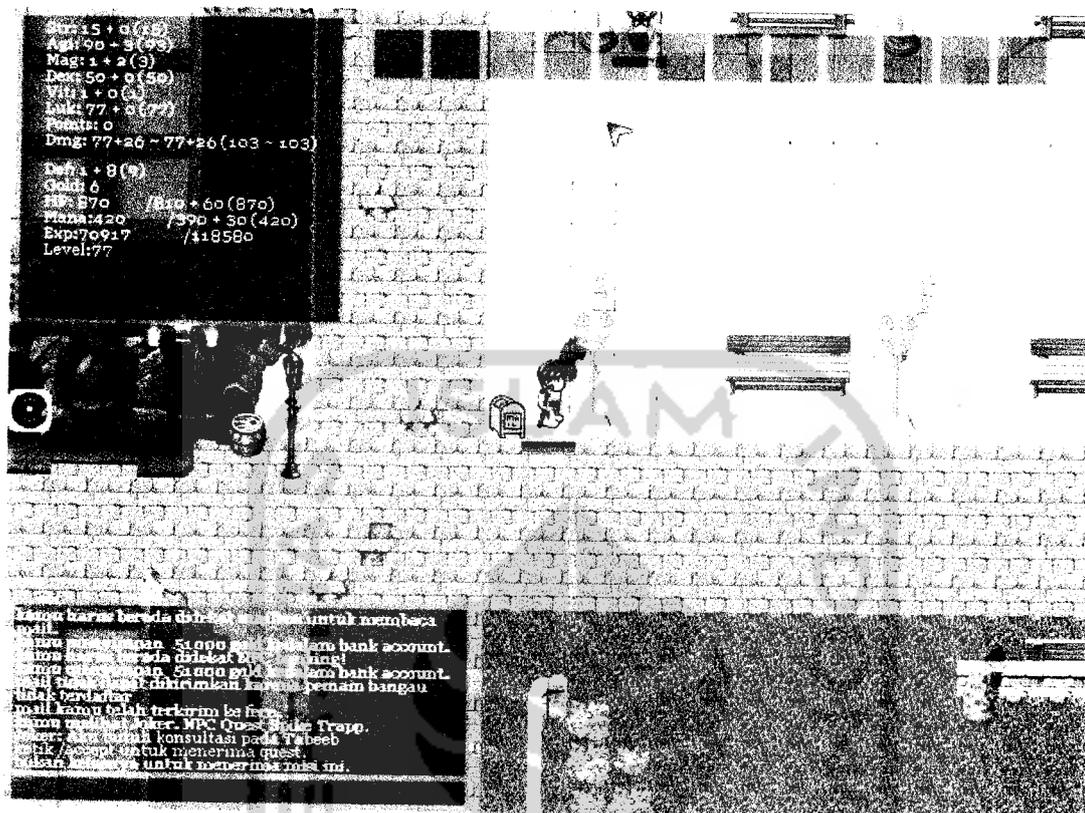
Selanjutnya pengujian dilakukan dengan menerima satu *quest* dan mencoba untuk menerima *quest* lainnya. Hasilnya dapat dilihat pada jendela *chat* pada gambar 4.99.



**Gambar 4.99** Pesan dalam permainan tidak dapat menjalani quest lebih dari satu

### 13. Class Berbeda Dengan Class Quest Yang Ditawarkan

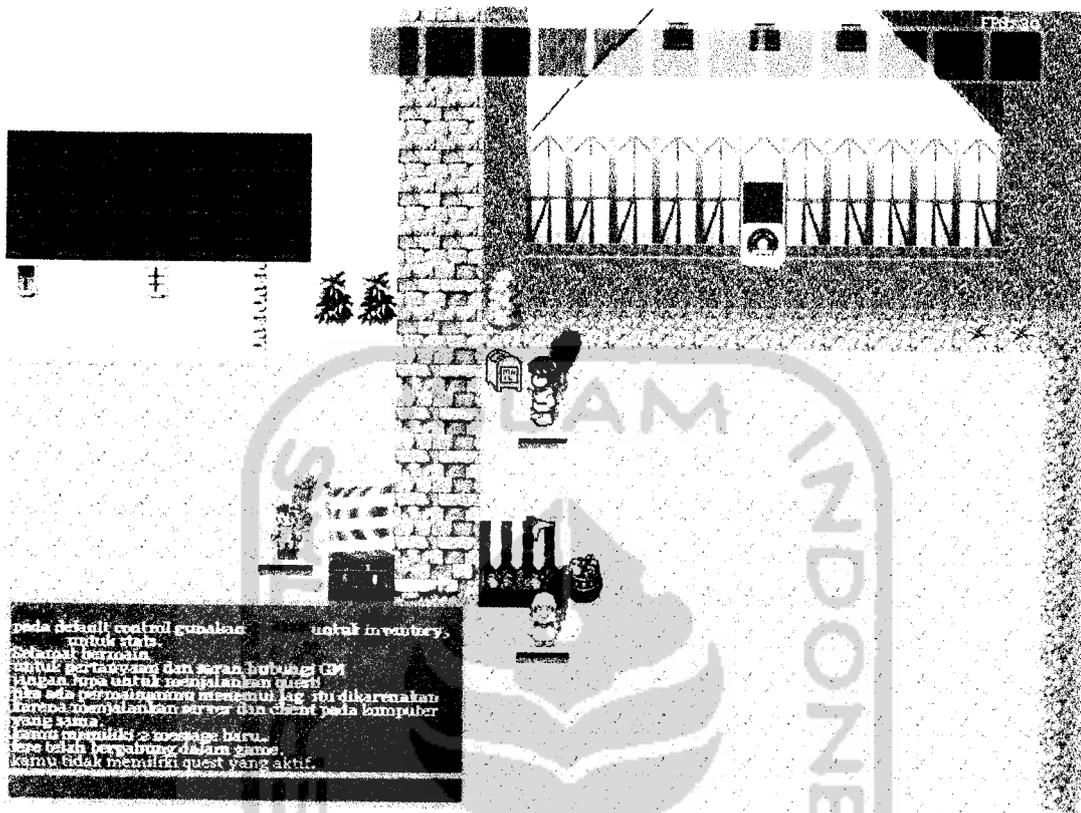
Pada permainan ini setiap satu *quest* hanya bisa dijalani oleh satu class tertentu seperti yang sudah diatur. Jika pemain yang berbeda class-nya mencoba untuk menerima *quest* yang bukan diperuntukkan class-nya maka akan muncul pesan pada jendela chat. Pesan dapat dilihat pada gambar 4.100.



**Gambar 4.100** Pesan dalam permainan tidak dapat menerima quest yang bukan untuk classnya

#### 14. Tidak Ada Quest Yang Sedang Dijalani

Seorang pemain yang sama sekali tidak menjalani *quest* dan mencoba untuk mendapatkan informasi *quest* yang sedang dijalaninya maka pemain akan menerima pesan pada jendela chatnya bahwa pemain tidak memiliki *quest* yang sedang dijalani. Pesan ini dapat dilihat seperti pada gambar 4.101.



**Gambar 4.101** Pesan dalam permainan tidak ada quest yang sedang dijalani

#### 15. Harus Berada Didekat Mailbox Jika Akan Menulis Mail

Pemain yang akan mengakses *mailbox* diharuskan berada didekat *mail* box. Jika pemain mencoba untuk mengakses *mailbox* dari jarak yang tidak diharuskan maka pemain akan menerima pesan seperti pada gambar 4.102.



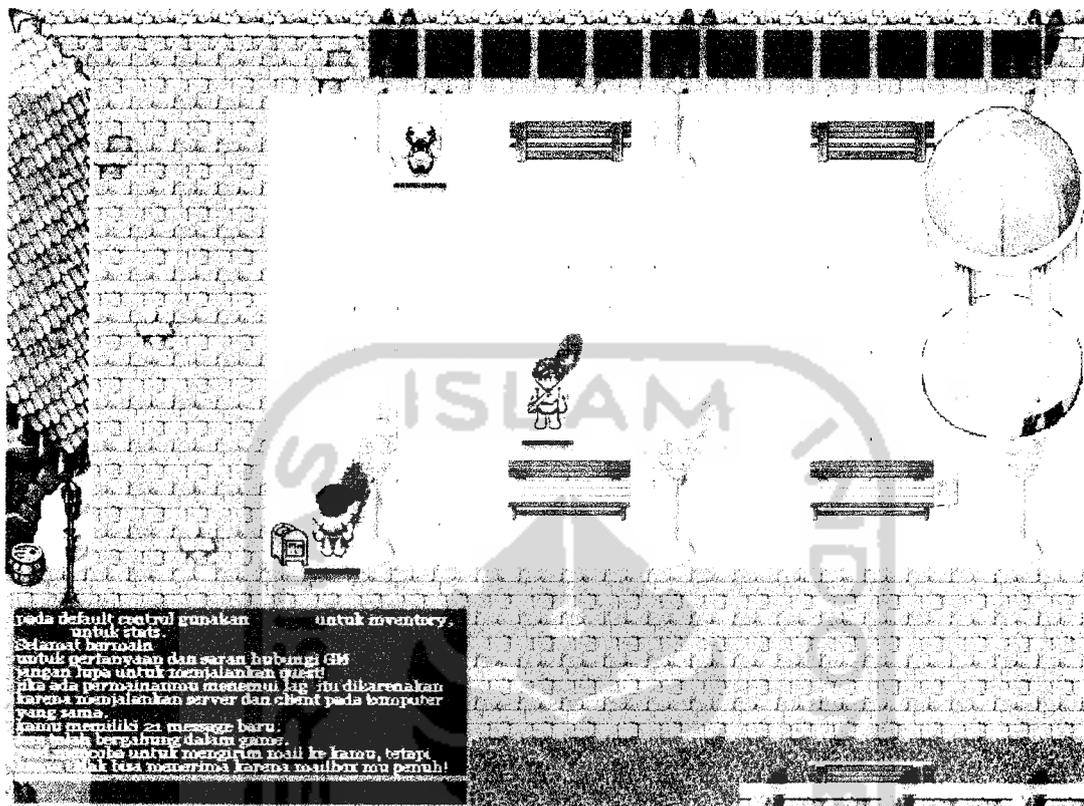
**Gambar 4.102** Pesan dalam permainan harus berada didekat mailbox untuk mengakses mailbox

#### 16. Mengirim Mail Kepada Nama Yang Tidak Terdaftar

Pengujian diteruskan dengan menguji pengiriman *mail* kepada pemain yang namanya tidak terdaftar pada database. Hasilnya akan muncul pesan kegagalan pengiriman *mail* dan informasi bahwa pemain yang dituju adalah tidak terdaftar. Pesan ini dapat dilihat seperti pada gambar 4.103.

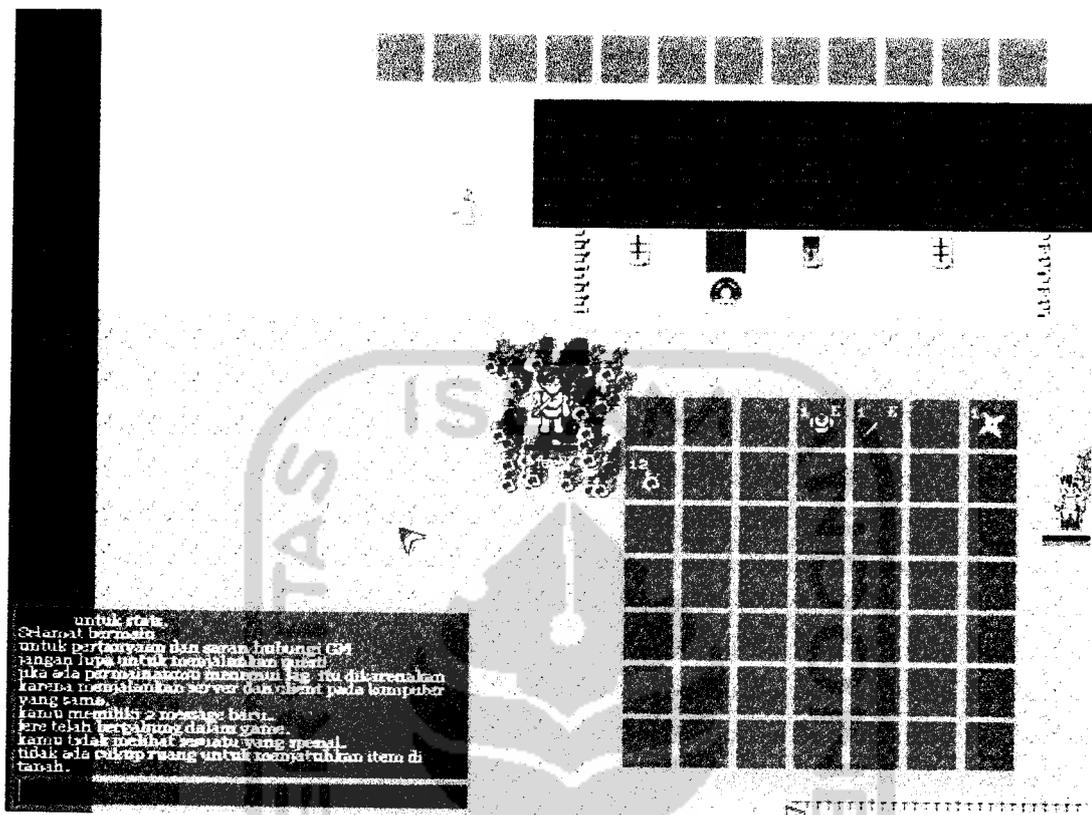






Gambar 4.105 Pesan dalam permainan tidak dapat menerima mail karena mailbox penuh

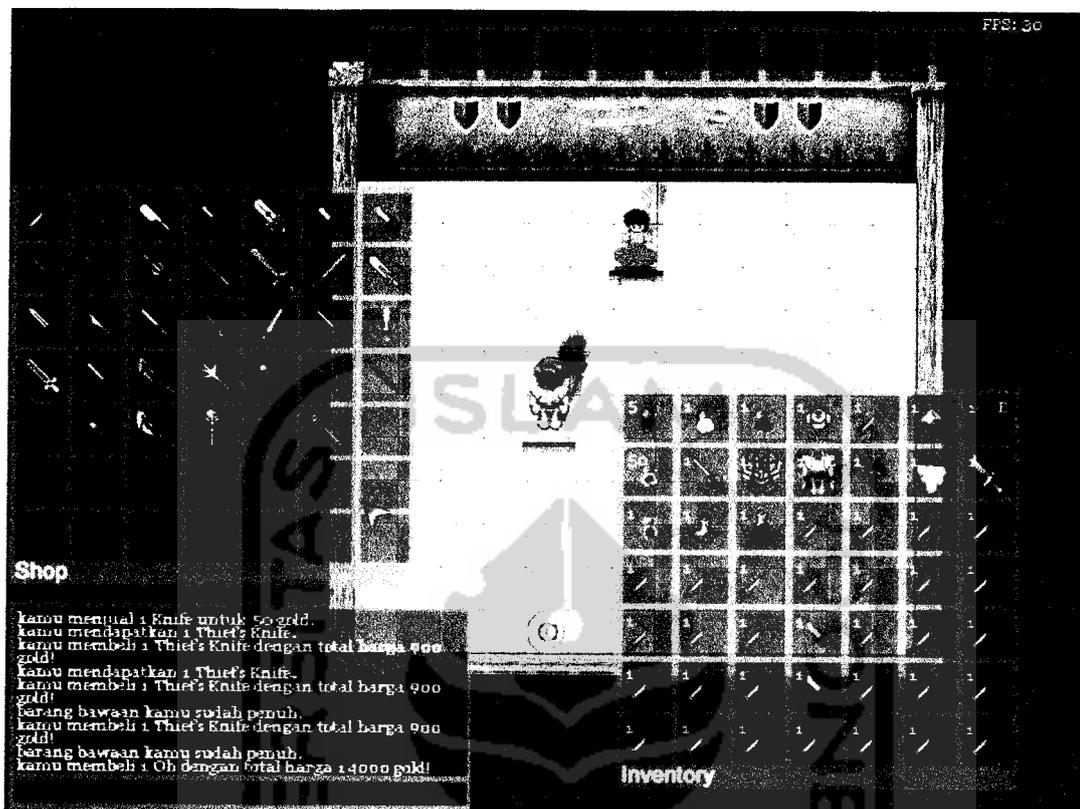




**Gambar 4.107** Pesan dalam permainan tidak dapat menjatuhkan item karena tidak ada cukup ruang

## 20. Inventory Penuh

Pengujian selanjutnya dengan mengisi jendela *inventory* dengan item hingga penuh dan tidak dapat lagi dimasuki item yang tidak dapat ditumpuk. Pada jendela *chat* akan muncul pesan bahwa *inventory* sudah penuh dan tidak bisa ditambahkan dengan item lain seperti terlihat pada gambar 4.108.



**Gambar 4.108** Pesan dalam permainan tidak bisa mendapatkan barang karena inventory penuh

## 21. Harus Berada Didekat NPC Shop Untuk Melakukan Transaksi

Pengujian pembelian barang pada NPC *shop* dilakukan dengan posisi tokoh yang berada jauh dari NPC *shop* sehingga pesan akan muncul seperti pada gambar 4.109.



**Gambar 4.109** Pesan dalam permainan tidak bisa bertransaksi karena terlalu jauh dengan NPC shop

## 22. Tidak Memiliki Cukup Gold Untuk Membeli Barang

Pemain yang membeli barang pada NPC *shop* secara otomatis akan berkurang gold yang dimilikinya sesuai dengan jumlah dan harga item yang dibelinya. Jika pemain memiliki jumlah gold dibawah jumlah dan harga item yang akan dibelinya maka pemain akan mendapatka pesan bahwa pemain tidak memiliki cukup gold untuk membeli item seperti pada gambar 4.110.

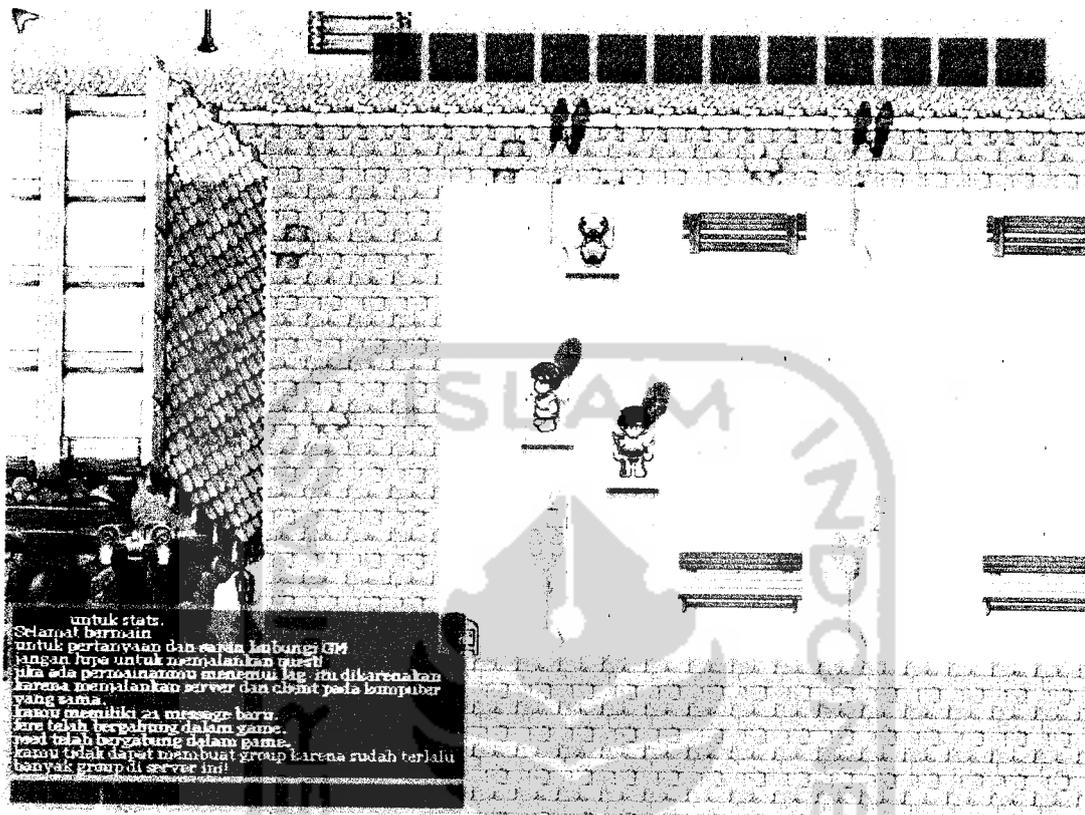


**Gambar 4.110** Pesan dalam permainan tidak bisa mendapatkan barang karena tidak memiliki cukup gold untuk membeli item

### 23. Menambahkan Item Pada Gudang Penyimpanan Yang Sudah Penuh

Pada gudang penyimpanan yang sudah penuh pemain tidak dapat memindahkan item dari *inventory* kecuali item yang dapat ditumpuk yang sudah ada jenisnya pada gudang penyimpanan. Jika pemain memindahkan item yang tidak bisa ditumpuk dari *inventory* menuju ke gudang maka akan muncul pesan seperti pada gambar 4.111.





**Gambar 4.112** Pesan dalam permainan tidak bisa membuat group karena jumlah maksimal group dalam satu server sudah terpenuhi

#### 25. Kapasitas Maksimal Jumlah Pemain Dalam Satu Group Tercapai

Dalam satu group hanya diperbolehkan maksimal delapan orang pemain, jika batas maksimal telah tercapai maka ketua group tidak dapat mengajak pemain lain untuk bergabung dalam group. Jika ketua group mencoba mengajak pemain lain akan mendapatkan pesan seperti pada gambar 4.113.



**Gambar 4.113** Pesan dalam permainan tidak bisa mengajak pemain bergabung dalam group karena batas maksimal pemain dalam group sudah terpenuhi

26. Mengajak Bergabung Pemain Yang Sudah Memiliki Group

Pemain yang sudah bergabung dengan group maka pemain harus keluar terlebih dahulu jika pemain ingin bergabung dengan group lain. Jika ketua group mengajak pemain lain yang telah memiliki group maka ketua group akan mendapatkan pesan seperti pada gambar 4.114.

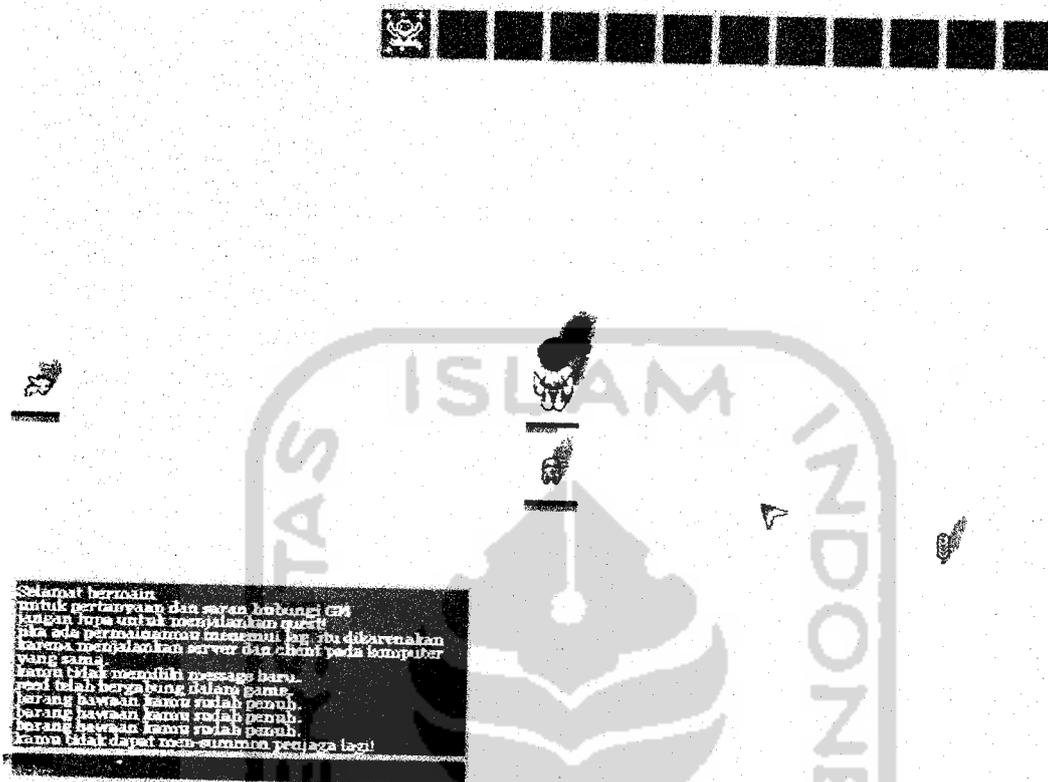




**Gambar 4.115** Pesan dalam permainan Tidak memenuhi persyaratan statistik untuk menggunakan item

## 28. Tidak Dapat Memanggil Summon Lebih Dari Satu

*Skill* Summon pada *class* Tabeeb pada permainan adalah kemampuan untuk memanggil bantuan monster. Monster akan datang dan membantu selama lima menit. Pemain hanya dapat memanggil satu bantuan monster saja. Untuk pemanggilan monster yang kedua dan seterusnya tidak dapat dilakukan. Pesan yang muncul karena mencoba memanggil bantuan monster lebih dari satu dapat dilihat pada gambar 4.116.



**Gambar 4.116** Pesan dalam permainan tidak bisa memanggil summon lebih dari satu

## 29. Bertransaksi Dengan Inventory Yang Sudah Penuh

Pengujian dilakukan dengan bertransaksi dengan pemain lain yang *inventory* -nya telah penuh. Transaksi akan secara otomatis dibatalkan karena pemain yang *inventory* -nya penuh tidak akan mendapatkan barang. Pesan kepada pemain yang *inventory* -nya penuh dapat dilihat pada gambar 4.117. dan pesan kepada pemain pihak lain dapat dilihat pada gambar 4.118.



**Gambar 4.117** Pesan dalam permainan transaksi batal karena pemain yang diajak  
 berransaksi tidak memiliki ruang pada inventory-nya



**Gambar 4.118** Pesan dalam permainan transaksi batal karena tidak memiliki ruang pada inventory-nya

### 30. Tidak Dapat Bertransaksi Dengan Diri Sendiri

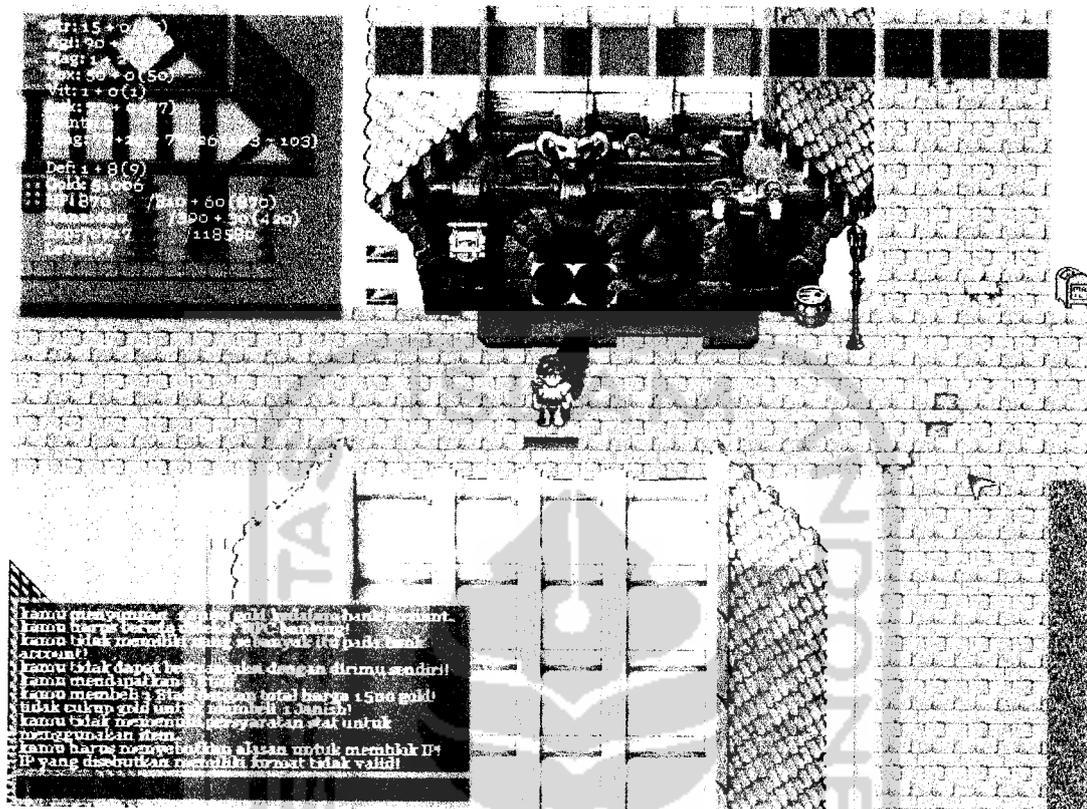
Pengujian selanjutnya adalah mencoba melakukan transaksi dengan diri sendiri. Pesan yang muncul dapat dilihat pada gambar 4.119.



**Gambar 4.119** Pesan dalam permainan tidak dapat bertransaksi dengan diri sendiri

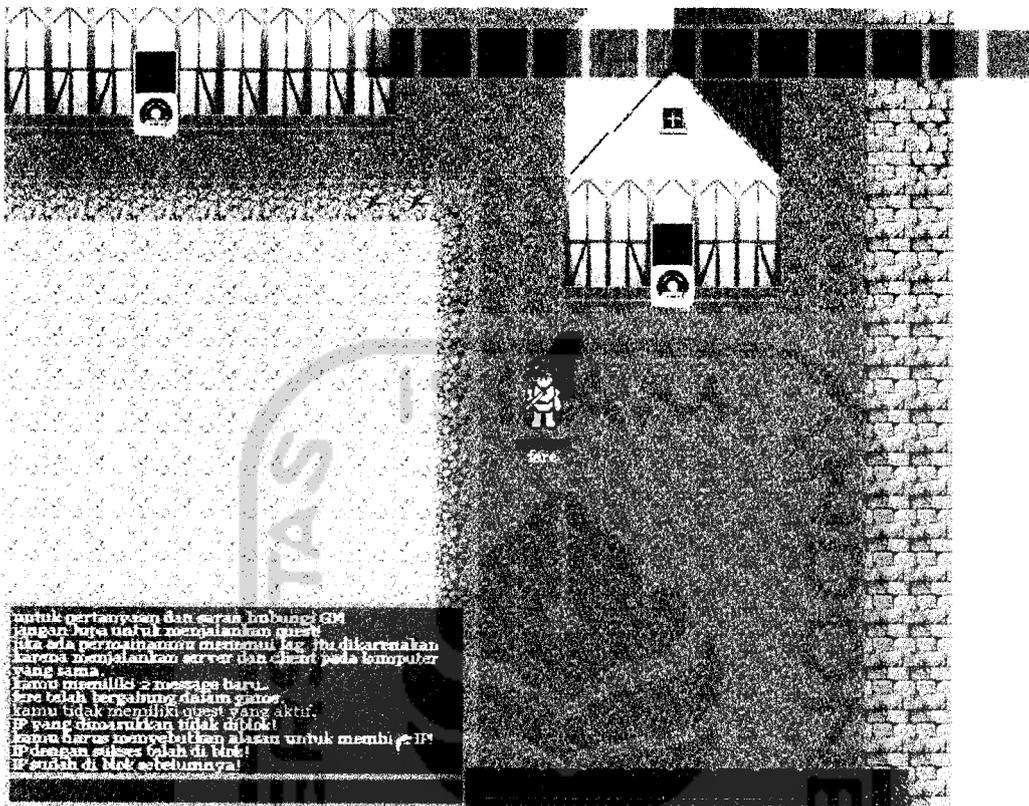
### 31. GM Memblokade Alamat IP Dengan Aturan Penulisan Salah

Pengujian berikut menggunakan perintah GM dengan kesalahan pada aturan penulisan perintah blokade alamat IP. Kesalahan akibat salah penulisan alamat IP akan memunculkan pesan kesalahan seperti pada gambar 4.120. Untuk kesalahan karena tidak menuliskan alasan dapat dilihat seperti pada gambar 4.121.



**Gambar 4.120** Pesan dalam permainan memblokade alamat IP dengan memasukan susunan alamat IP yang salah

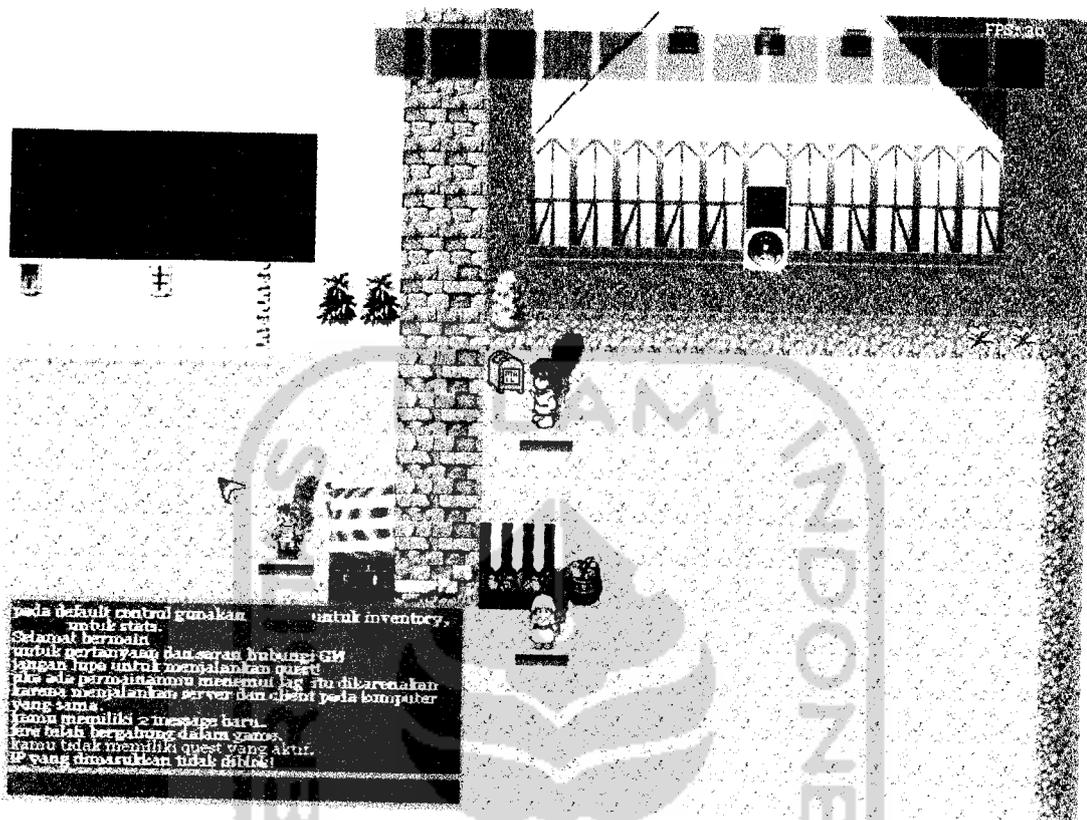




**Gambar 4.122** Pesan dalam permainan memblokade alamat IP yang sedang diblokade

### 33. GM Membuka Blokade Alamat IP Yang Tidak Sedang Diblokade

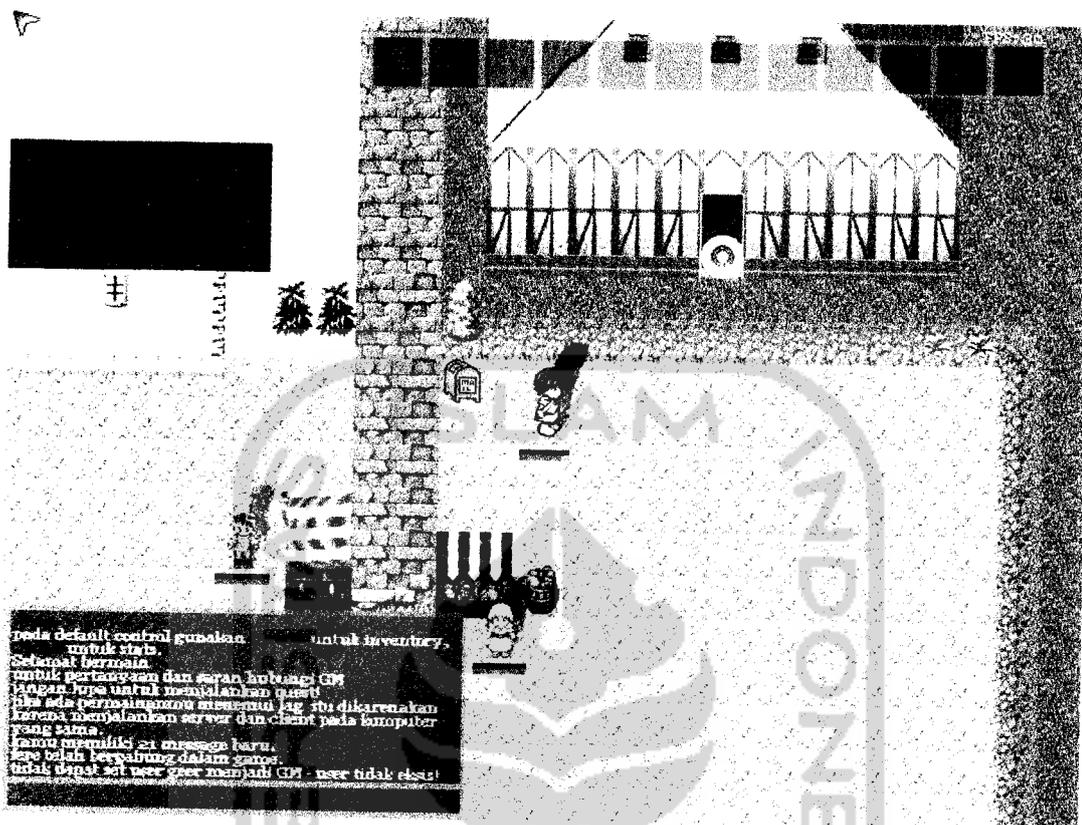
Pengujian selanjutnya adalah membuka blokade alamat IP yang tidak diblokade. Pesan yang muncul dapat dilihat pada gambar 4.123.



**Gambar 4.123** Pesan dalam permainan membuka blokade alamat IP yang tidak sedang diblokade

#### 34. GM Menjadikan Pemain Yang Tidak Terdaftar Sebagai GM

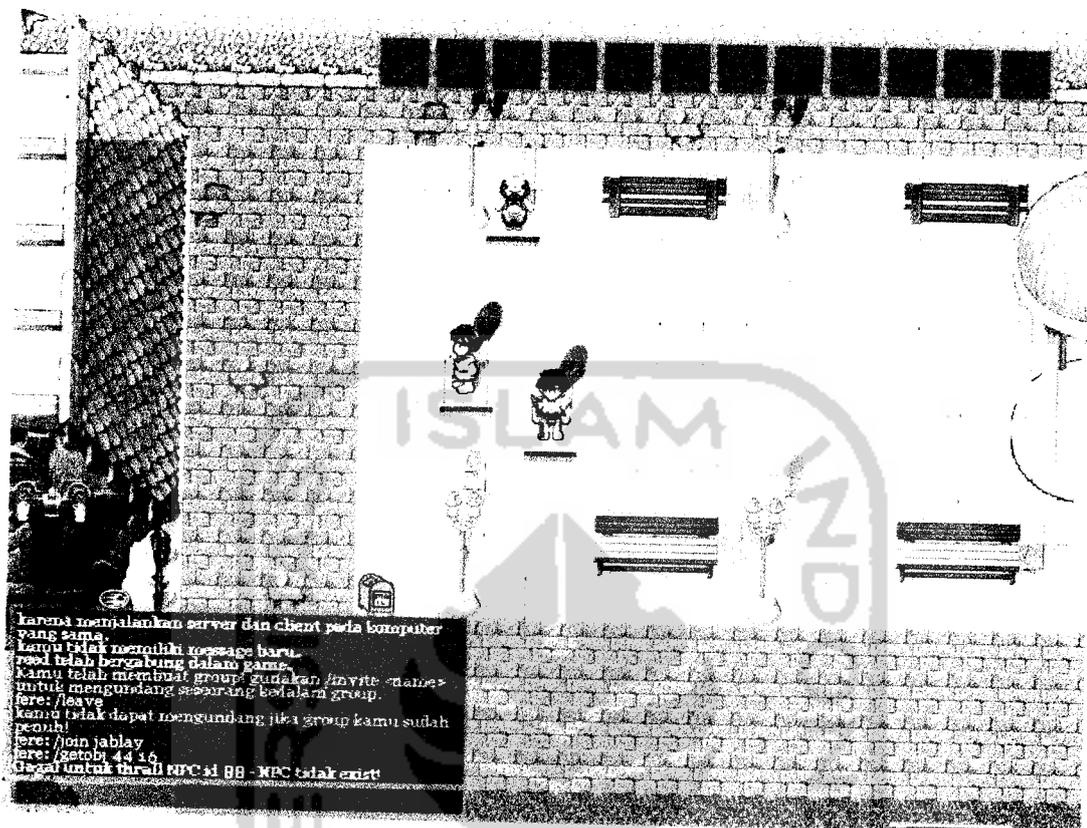
Pengujian selanjutnya adalah menggunakan perintah untuk menjadikan pemain lain menjadi GM tetapi nama pemain yang dimasukkan salah atau tidak diketahui. Sehingga muncul pesan tidak dapat mengubah pemain menjadi GM seperti terlihat pada gambar 4.124.



**Gambar 4.124** Pesan dalam permainan mengatur pemain yang tidak terdaftar menjadi GM

### 35. GM Gagal Memunculkan Monster

Pengujian selanjutnya adalah kesalahan pada pengisian ID monster ketika menggunakan perintah untuk memunculkan monster. Karena ID monster tidak diketahui maka monster tidak akan muncul. Pesan yang muncul dapat dilihat pada gambar 4.125.



Gambar 4.125 Pesan dalam permainan gagal memunculkan monster

#### 4.7 Pembahasan

Pada bagian pembahasan ini akan membandingkan aplikasi permainan dengan aplikasi permainan yang serupa. Untuk perbandingan diambil dari permainan Ragnarok Online yang dipasarkan semenjak 31 Agustus 2002 di Korea Selatan, oleh GRAVITY Co., Ltd.

Ragnarok Online adalah *massively multiplayer online role-playing game* yang dibuat berdasarkan komik Ragnarok oleh Lee Myung-Jin. Ragnarok memiliki cerita

berdasarkan mitologi Normandia dan berkultur Jepang, Afrika dan Kristen.  
[WIK07d]

#### 4.7.1 Gameplay

Ragnarok Online menggunakan kombinasi *sprite* isometrik dan grafik tiga dimensi untuk lingkungan pada permainan. Pada proses pendaftaran akun dilakukan pada situs resmi Ragnarok Online. Gender karakter adalah sama dengan gender akun ketika mendaftar. Setelah proses login terdapat pemilihan server untuk bermain. Setiap server diberikan tiga slot karakter yang berbeda untuk satu akun, masing-masing karakter dapat diberi nama atau dihapus.

#### 4.7.2 Antarmuka

Perbedaan utama antarmuka dengan Ragnarok Online adalah pergerakan karakter yang dikontrol melalui *mouse* yang lebih mudah. Selain itu Ragnarok Online menggunakan sistem jendela dinamis dengan *scroll*. Skills juga memakai sistem jendela. Dan untuk konfigurasi dapat diakses melalui permainan kecuali konfigurasi video. Selain hal-hal diatas kedua permainan memiliki antarmuka yang hampir serupa.

#### 4.7.3 Quest

Sama seperti MMORPG lainnya Ragnarok Online juga tidak memiliki alur tertentu, pemain bebas memainkan perannya. Terdapat juga NPC dengan fungsi sebagai penyedia quest maupun penyedia shop. Pada quest tidak dibatasi untuk mendapatkan skill, experience, gold maupun item, tetapi juga untuk merubah *job class*.

#### **4.7.5 Pengembangan Karakter**

Ragnarok Online memakai sistem experience yang sama, yaitu dengan mendapatkan dari membunuh monster. Setiap karakter memiliki dua level yang berbeda yaitu Base Level dan Job Level, untuk setiap kenaikan Base Level, karakter mendapatkan point untuk menaikkan atribut statistik, kenaikan atribut statistik memiliki harga point untuk dinaikkan per satu nilai. Sedangkan untuk kenaikan Job Level akan mendapatkan satu skill point yang digunakan untuk pemilihan skill yang tersedia pada job classnya. Skill yang tersedia memiliki persyaratan skill lain untuk dapat dipelajari.

#### **4.7.6 Sistem Job Class**

Total job class yang tersedia pada Ragnarok Online adalah tiga puluh sembilan job. Pada awalnya semua karakter adalah Novice, job class dasar yang memiliki sepuluh job level, setelah job level sepuluh maka Novice dapat mendaftar ke NPC job quest yang ditunjuk untuk menjalani quest. Job pertama terdiri sembilan job class yang berbeda. Jika karakter telah memaksimalkan job yang pertama ini maka karakter dapat mendaftar untuk menjadi job kedua yang lebih tinggi tingkatannya. Tiap-tiap karakter memiliki pilihan salah satu dari dua job class yang kedua sesuai dengan job pertamanya. Jika karakter telah memaksimalkan job kedua maka karakter dapat memilih untuk menjadi tingkatan tertinggi pada jobnya, job tingkat tinggi ini dapat dicapai dengan mengulang kembali karakter menjadi Novice dan menjalani job pertama dan kedua dengan skill minimalis. Jika pengulangan berhasil sampai maksimal maka dapat menjadi job tingkat tertinggi.

**KESIMPULAN****4.7.7 Pemain Melawan Pemain**

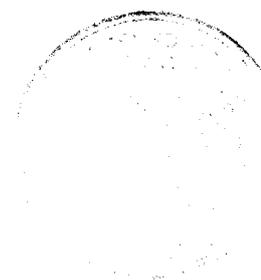
Pada Ragnarok Online juga terdapat fitur untuk pemain melawan pemain dengan mengakses pada NPC untuk memasuki arena *Player vs Player*. Pemain dapat memilih dua mode yang ada. Yaitu Yoyo-mode dan Nightmare-mode. Pada Yoyo-mode jika karakter mati maka tidak dikenakan penalti sedangkan Nightmare-mode jika pemain mati akan dikenakan penalti experience maupun penalti item.

**4.7.8 Perang Emperium**

Fitur ini hanya ada pada Ragnarok Online. Emperium adalah item yang dibutuhkan untuk membuat Guild. Guild adalah persatuan karakter-karakter yang diakui kerajaan pada Ragnarok Online. Pada perang ini Guild berperang melawan Guild lain untuk merebut kastil. Kastil ini menjadi basis pertemuan ataupun jalan menuju *dungeon* khusus yang berada didalam kastil.

**4.7.9 Sistem Point untuk Guild**

Fitur unik pada Ragnarok Online yang memberikan bonus point bagi Guild yang mampu mempertahankan kastil agar tidak direbut Guild lain. Point ini akan didata dan digunakan untuk peringkat regional (misalnya dalam satu negara). Setiap tiga bulan Guild peringkat pertama akan mendapatkan hadiah item dan juga lima peringkat teratas lainnya. Dan setiap enam bulan diadakan turnamen bagi peringkat pertama sampai enam pada tiap server, pemenangnya akan mendapatkan hadiah uang (mata uang nyata) dan berhak mendapatkan tiket untuk turnamen tingkat dunia yang diadakan tiap tahunnya.



## 5.2 Saran

Beberapa hal yang perlu diperhatikan bagi pengguna aplikasi ini sehubungan dengan kemungkinan pengembangan aplikasi lebih lanjut diantaranya:

1. Untuk input kontrol masih dirasakan sulit dan *command script* yang tidak *user friendly*.
2. Aplikasi ini hanya memvisualisasikan proses gerak secara dua dimensi dengan efek tiga dimensi. Sehingga untuk lebih lanjut masih dapat dikembangkan dengan real 3D animasi yang lebih menarik.
3. Fasilitas yang ada dalam aplikasi masih diperlukan banyak penambahan dan informasi yang ditampilkan aplikasi kurang informatif, sehingga masih membutuhkan untuk melihat panduan permainan diluar permainan.
4. Manajemen aplikasi server permainan tidak *user friendly*, sehingga membutuhkan administrator permainan yang menguasai script SQL.
5. Untuk pengembangan permainan sejenis dapat menggunakan perangkat lunak lain. Rekomendasi untuk penggunaan perangkat lunak lain adalah: Kaneva, Multiverse, Elusyum, RPGMakerXP, RPG Toolkit, xtremeworlds, debugged Konfuze dan Mirage. Kata kunci yang digunakan Online Role Playing Game Maker.

## DAFTAR PUSTAKA

- [BOE03] Boefe, *Introduction to Visual Basic Socket Programming*.  
<http://www.osix.net/modules/article/?id=25>, diakses 27 Maret 2007.
- [DAV05] Davison, Andrew, *Killer Game Programming in Java*. California: O'Reilly , 2005.
- [DYE05] Dyer, Russell, *MySQL in a Nutshell*, California: O'Reilly , 2005.
- [HAL07] Hall, Justin, *Passively Multiplayer Online Games*.  
<http://www.passivelymultiplayer.com/index.html>, diakses 13 Februari 2007.
- [MEL05] Melber, Derek ,dan Balter, Dan , *MCSE 70-270 Exam Cram 2: Windows XP Professional*, Second Edition, Indianapolis: Que, 2005.
- [MUL06] Multiverse, <http://update.multiverse.net/wiki/index.php>, diakses pada tanggal 29 September 2006.
- [PRA03] Prasetyo, Didik Dwi, *Belajar Sendiri: Administrasi Database Server MySQL*. Jakarta: Elex Media Komputindo, 2003.
- [PRA04] Prasetyo, Didik Dwi, *Belajar Sendiri: Aplikasi Database Client/Server Menggunakan Delphi dan MySQL*. Jakarta: Elex Media Komputindo, 2004.
- [PUT04] Putra, Indra, *Membuat Program Aplikasi Nyata dengan Visual Basic 6.0*. Yogyakarta: Andi Offset, 2004.
- [POS05] Posey, Brien M., *Firewall 101*.  
[http://www.windowsnetworking.com/articles\\_tutorials/Firewalls-101.html](http://www.windowsnetworking.com/articles_tutorials/Firewalls-101.html), diakses 25 November 2006.
- [RAH02] Rahardjo, Budi, *Memahami Teknologi Informasi*. Jakarta: Elex Media Komputindo, 2002.

- [SPO07] Spodi, *From vbGORE Visual Basic Online RPG Maker*.  
[http://www.vbgorecom/index.php\\_title=Main\\_Page.html](http://www.vbgorecom/index.php_title=Main_Page.html), diakses pada tanggal 20 Januari 2007.
- [SUR00] Suryo Kusumo, Ario, *Buku Latihan Microsoft Visual Basic 6.0*.  
Jakarta: Elex Media Komputindo, 2000.
- [WAH06] Wahana, *Menginstalasi Perangkat Jaringan Komputer*. Jakarta: Elex Media Komputindo, 2006.
- [WIJ04] Wijaya, Hendra, *Belajar Sendiri: Cisco Router*. Jakarta: Elex Media Komputindo, 2004.
- [WIK07a] Wikipedia, *Game Theory*.  
[http://www.reference.com/cite.html?qh=Game\\_theory&ia=wiki](http://www.reference.com/cite.html?qh=Game_theory&ia=wiki), diakses 13 Februari 2007.
- [WIK07b] Wikipedia, *MMORPG*. <http://en.wikipedia.org/wiki/MMORPG>, diakses 13 Februari 2007.
- [WIK07c] Wikipedia, *MMORPG terms and acronims*.  
[http://en.wikipedia.org/wiki/MMORPG\\_terms\\_and\\_acronims](http://en.wikipedia.org/wiki/MMORPG_terms_and_acronims), diakses 13 Februari 2007.
- [WIK07d] Wikipedia, *Ragnarok Online*.  
[http://en.wikipedia.org/wiki/Ragnarok\\_Online](http://en.wikipedia.org/wiki/Ragnarok_Online), diakses 13 Februari 2007.