

# **APLIKASI CHATROOM BERBASIS WEB**

Tugas Akhir disusun untuk memenuhi syarat  
Mencapai gelar Kesarjanaan Komputer pada  
Program Studi Teknik Informatika  
Jenjang Program Strata-1



Oleh :

**PRIMADITYA SURYA MAHARDIKA**

**08.01.53.0073**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS STIKUBANK (UNISBANK)**

**SEMARANG**

**2013**

## HALAMAN PENGESAHAN

Telah dipertahankan di depan tim dosen penguji Tugas Akhir Fakultas Teknologi Informasi UNIVERSITAS STIKUBANK (UNISBANK) Semarang dan diterima sebagai salah satu syarat guna menyelesaikan Program Strata 1, Program Studi Teknik Informatika Fakultas Teknologi Informasi.

Semarang, Februari 2013

Ketua

**(Setyawan Wibisono, S.Kom, M.Cs)**

Sekretaris

**(Dwi Budi Santoso, S.Kom)**

Anggota

**(Felix Andreas Sutanto, S.Kom, M.Cs)**

MENGETAHUI

**UNIVERSITAS STIKUBANK (UNISBANK) SEMARANG**

Dekan

Fakultas Teknologi Informasi

**(Dwi Agus Diartono, S.Kom, M.Kom)**

## **PERNYATAAN KESIAPAN UJIAN TUGAS AKHIR**

Saya, Primaditya Surya Mahardika, dengan ini menyatakan bahwa Laporan Tugas Akhir yang berjudul :

### **APLIKASI CHATROOM BERBASIS WEB**

adalah benar hasil karya saya dan belum pernah diajukan sebagai karya ilmiah, sebagian atau seluruhnya, atas nama saya atau pihak lain.

**Primaditya Surya Mahardika**

NIM : 08.01.53.0073

Disetujui oleh pembimbing

Kami setuju Laporan tersebut diajukan untuk Ujian Tugas Akhir

Semarang, Februari 2013

( **Setyawan Wibisono, S.Kom, M.Cs** )

Pembimbing I

Semarang, Februari 2013

( **Dwi Budi Santoso, S. Kom** )

Pembimbing II

## **MOTO DAN PERSEMBAHAN**

*Keluarga adalah tempat berteduh untuk menghadapi hari esok*

*Jika kamu tidak yakin kalau Allah akan meringankan masalahmu, kamu akan menderita.*

**Laporan tugas akhir ini saya persembahkan kepada :**

- Bapakku dan Ibuku.
- Bapak dan Ibu dari Bapakku dan Ibuku.

**FAKULTAS TEKNOLOGI INFORMASI**  
**UNIVERSITAS STIKUBANK (UNISBANK) SEMARANG**

Program Studi : Teknik Informatika  
Tugas Akhir Sarja Komputer

**”APLIKASI CHATROOM BERBASIS WEB”**

**PRIMADITYA SURYA MAHARDIKA**  
**NIM : (08.01.53.0073)**

***Abstrak***

Saat ini masih ada orang-orang yang belum memiliki email. Karena mereka tidak peduli atau tidak menganggap penting. Dan juga melakukan proses pendaftaran email pun terasa tidak mudah untuk sebagian orang. Namun bagaimanakah bila orang-orang ini ingin berbagi file dengan temannya?

Bagi seseorang yang belum memiliki email akan sulit untuk berbagi file kepada temannya. Untuk itulah diperlukan sebuah aplikasi web chatting yang cepat dan mudah digunakan yang disertai dengan fungsi upload file. Sehingga bisa berbagi file tanpa menggunakan email.

Aplikasi web ini adalah sebuah sarana dimana banyak orang sekaligus bisa melakukan percakapan bersama satu sama lain. Disamping digunakan untuk bercakap-cakap, aplikasi ini juga bisa digunakan untuk berbagi file.

Penggunaan aplikasi ini sangat mudah, pengguna tinggal melakukan registrasi singkat dan bisa langsung menggunakannya. Didalamnya terdapat fungsi untuk menambahkan chatroom sendiri. Dan disamping tombol pengiriman pesan terdapat fungsi upload file. Saat pengguna selesai mengupload, download link akan muncul pada kotak percakapan sehingga bisa diunduh pengguna lain.

Berdasarkan hasil analisa yang dilakukan dapat diambil kesimpulan bahwa untuk pengguna yang belum memiliki email, pengiriman file menjadi lebih efektif daripada menggunakan email.

***Kata kunci***

*Chatting, Chatroom, upload, download, web*

Semarang, Februari 2013

Mengetahui,

Pembimbing I

Pembimbing II

**(Setyawan Wibisono, S.Kom, M.Cs)**

**(Dwi Budi Santoso, S.Kom)**

## **KATA PENGANTAR**

Puji syukur dipanjatkan kehadirat Allah SWT atas berkah, rahmat, hidayah dan karunia-Nya sehingga dapat diselesaikannya Tugas Akhir. Skripsi ini disusun guna melengkapi salah satu syarat untuk menyelesaikan program studi strata satu jurusan Teknik Informatika di UNISBANK Semarang.

Menyadari keterbatasan kemampuan dalam menyelesaikan Tugas Akhir ini banyak mendapat bantuan dari semua pihak yang telah membantu, maka tidak lupa untuk menyampaikan ucapan terimakasih kepada :

1. Allah SWT.
2. Bapak Dr. Bambang Suko Priyono, M.M selaku Rektor UNISBANK Semarang.
3. Bapak Dwi Agus Diartono, S.Kom, M.Kom selaku Dekan Fakultas Teknologi Informasi UNISBANK Semarang.
4. Ibu Dewi Handayani UN, S.Kom, M.Kom selaku ketua Program Studi Teknik Informatika UNISBANK Semarang.
5. Bapak Setyawan Wibisono, S.Kom, M.Cs selaku pembimbing pertama dan Bapak Dwi Budi Santoso, S.Kom selaku pembimbing.
6. Bapak dan Ibu tercinta yang telah memberikan doa dan dukungan baik material maupun spiritual selama Tugas Akhir hingga selesai laporan ini.
7. Mentor dan sahabat saya b'gax.
8. DREAMTHEATER, yang membuat semangat saya membara untuk mengetik skripsi ini.

Disadari adanya keterbatasan yang dimiliki dalam pembuatan skripsi ini dan masih jauh dari kesempurnaan. Akhir kata, semoga tugas akhir ini dapat bermanfaat.

Semarang, Februari 2013

Penulis

( **Primaditya Surya M.** )

## DAFTAR ISI

<b>HALAMAN JUDUL .....</b>	<b>i</b>
<b>HALAMAN PERSETUJUAN .....</b>	<b>ii</b>
<b>HALAMAN PENGESAHAN .....</b>	<b>iii</b>
<b>HALAMAN MOTTO DAN PERSEMBAHAN .....</b>	<b>iv</b>
<b>ABSTRAKSI.....</b>	<b>v</b>
<b>KATA PENGANTAR.....</b>	<b>vi</b>
<b>DAFTAR ISI.....</b>	<b>viii</b>
<b>DAFTAR GAMBAR.....</b>	<b>xi</b>
<b>DAFTAR TABEL .....</b>	<b>xii</b>
<b>BAB I PENDAHULUAN</b>	
1.1 Latar Belakang .....	1
1.2 Perumusan Masalah .....	1
1.3 Batasan Masalah .....	1
1.4 Tujuan dan Manfaat .....	2
1.5 Metodologi Penelitian.....	2
1.5.1 Metode Pengumpulan Data .....	2
1.5.2 Metode Pengembangan <i>System</i> .....	2
1.6 Sistematika Penulisan .....	3
<b>BAB II LANDASAN TEORI</b>	
2.1 Konsep Dasar <i>System</i> .....	4
2.1.1 Pengertian <i>System</i> .....	4



2.1.2	Karakteristik <i>System</i> .....	4
2.2	Konsep <i>Web</i> .....	5
2.2.1	Pengertian <i>Web</i> .....	5
2.2.2	<i>Browser Web</i> .....	5
2.2.3	HTML.....	6
2.3	PHP .....	6
2.4	MySQL .....	7
2.5	Teori Dasar Perancangan <i>System</i> .....	7
2.5.1	Analisa <i>System</i> .....	7
2.5.2	Pengertian Perancangan <i>System</i> .....	8
2.5.3	Alat Bantu Dalam Perancangan <i>System</i> .....	8

### **BAB III PERANCANGAN SYSTEM**

3.1	Deskripsi <i>System</i> .....	22
3.2	Rancangan Aplikasi .....	22
3.3	Pemodelan Fungsional .....	25
3.3.1	<i>Data Flow Diagram Level 0</i> .....	25
3.3.2	<i>Data Flow Diagram Level 1</i> .....	26
3.3.3	<i>Flow Chart</i> .....	27
3.3.4	Perancangan <i>Database</i> .....	34
3.3.5	Pemilihan Kunci <i>Record</i> .....	34

### **BAB IV IMPLEMENTASI**

4.1	Implementasi Perangkat Keras .....	36
4.2	Implementasi Program .....	37

4.2.1	Halaman Registrasi .....	37
4.2.2	Halaman <i>Login</i> .....	38
4.2.3	Halaman <i>Chat room</i> .....	39
4.2.4	Halaman <i>Setting</i> Dan Daftar <i>User</i> .....	41
4.2.5	Halaman <i>Upload File</i> .....	43

## **BAB V PENUTUP**

5.1	Kesimpulan .....	45
5.2	Saran .....	46

## **DAFTAR PUSTAKA**

## **LAMPIRAN**

## DAFTAR GAMBAR

Gambar 3.1 Rancangan <i>System</i> .....	22
Gambar 3.2 Rancangan menu <i>Chat room</i> .....	23
Gambar 3.3 Data Flow Diagram Aplikasi Chatroom Level 0.....	26
Gambar 3.4 Data Flow Diagram Aplikasi Chatroom Level 1.....	27
Gambar 3.5 <i>Flow Chart</i> proses Registrasi.....	28
Gambar 3.6 <i>Flow Chart</i> proses <i>login</i> .....	29
Gambar 3.7 <i>Flow Chart</i> kirim pesan .....	30
Gambar 3.8 <i>Flow Chart</i> proses upload file.....	31
Gambar 3.9 <i>Flow Chart</i> tambah chatroom .....	32
Gambar 3.10 <i>Flow Chart</i> hapus pengguna .....	32
Gambar 3.11 <i>Flow Chart</i> ubah password .....	33
Gambar 3.12 <i>Flow Chart private chat</i> .....	34
Gambar 4.1 Tampilan halaman registrasi .....	38
Gambar 4.2 Tampilan halaman <i>login</i> .....	39
Gambar 4.3 Halaman <i>chatroom</i> .....	41
Gambar 4.4 Halaman Setting dan daftar <i>user</i> .....	43
Gambar 4.5 Halaman upload file.....	44

## DAFTAR TABEL

Tabel 2.1 Diagram <i>Context</i> .....	11
Tabel 2.2 DFD <i>Leveled</i> .....	12
Tabel 2.3 <i>Symbol Data Dictionary</i> .....	14
Tabel 2.4 <i>Symbol Entity Relationship Diagram</i> .....	15
Tabel 2.5 <i>Symbol Flow of Document</i> .....	21
Tabel 3.1 Struktur tabel <i>user</i> .....	34
Tabel 3.2 Struktur tabel <i>chatting</i> .....	34
Tabel 3.3 Struktur tabel <i>chatroom</i> .....	35
Tabel 3.4 Struktur tabel <i>private</i> .....	35

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Kemajuan teknologi berkembang sangat pesat. Salah satunya yaitu teknologi komputer. Dengan hadirnya komputer, maka akan semakin mempermudah dan mempersingkat pekerjaan manusia.

Kebanyakan orang saat ini terlalu sibuk dan sulit menentukan waktu untuk berdiskusi dengan keluarga mereka. Kadang kala mereka kehabisan waktu untuk saling bertemu secara langsung. Maka dengan aplikasi ini diharapkan siapa pun dapat melakukan percakapan *online* bersama secara mudah dan lebih cepat. Aplikasi *chat room* ini adalah, sebuah sarana yang digunakan untuk melakukan percakapan secara *online* dengan lebih mudah dan lebih cepat dibanding menggunakan *email*. Dikarenakan saat ini pendaftaran *email* tidak mudah dan membutuhkan waktu yang cukup lama.

### **1.2 Perumusan Masalah**

Berdasarkan latar belakang masalah diatas, maka dirumuskan masalah bagaimana merancang sebuah aplikasi *web* dengan fungsi *chat* dan berbagi *file*, yang mudah digunakan untuk semua kalangan pengguna.

### **1.3 Batasan Masalah**

Agar tidak menyimpang dari tujuan penulisan skripsi, maka hanya dibatasi masalah pada:

1. *Chat room* yang dibuat bisa dimasuki semua orang.
2. Bersifat multi *user*, bisa digunakan siapa saja.

#### **1.4 Tujuan dan Manfaat Penelitian**

##### **1. Tujuan**

Tujuan yang ingin dicapai adalah menciptakan aplikasi *chat room* berbasis *web* yang mudah digunakan oleh semua kalangan.

##### **2. Manfaat**

Fungsi utama aplikasi ini adalah setiap pengguna dapat berbagi *file* ke *user* lain dengan cepat, dan dapat melakukan *private chat*.

#### **1.5 Metodologi Penelitian**

Kegiatan penelitian pada umumnya bertujuan untuk menemukan, mengembangkan dan menguji kebenaran suatu ilmu pengetahuan. Oleh karena itu diperlukan adanya suatu metodologi penelitian yang sesuai dengan jenis penelitian itu sendiri.

##### **1.5.1 Metode Pengumpulan Data**

Menggunakan metode studi pustaka, yaitu pengumpulan data dengan membaca literatur-literatur atau buku-buku yang berhubungan dengan pemrograman aplikasi *chatting*, bahasa pemrograman *php*, *javascript* dan *jquery*.

##### **1.5.2 Metode Pengembangan System**

Metode yang digunakan membuat atau mengembangkan piranti lunak ini adalah model *prototype* (Pressman, 1992). Metode

ini merupakan metode pengembangan *system* dimana hasil analisa perbagian langsung diterapkan kedalam sebuah model tanpa harus menunggu *system* selesai dianalisa.

## 1.6 Sistematika Penulisan

Sistematika penulisan tugas akhir ini adalah sebagai berikut :

### BAB I PENDAHULUAN

Dalam bab ini akan dibahas mengenai latar belakang, perumusan masalah, tujuan dan manfaat, sistematika penulisan.

### BAB II LANDASAN TEORI

Bab ini membahas mengenai pengertian konsep dasar *system*, sekilas tentang *web* beserta komponen-komponen pendukungnya.

### BAB III PERANCANGAN *SYSTEM*

Dalam bab ini membahas mengenai pembuatan aplikasi *chat room* berbasis *web*.

### BAB IV IMPLEMENTASI *SYSTEM*

Menjelaskan tampilan *system* yang sudah jadi.

### BAB V PENUTUP

Dalam bab ini menjelaskan mengenai kesimpulan dari hasil penulisan tugas akhir ini.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Konsep Dasar *System***

##### **2.1.1. Pengertian *System***

Terdapat dua kelompok pendekatan di dalam mendefinisikan *system*, yaitu menekankan pada prosedurnya dan menekankan pada komponen atau elemennya.

1. Pendekatan *system* yang lebih menekankan pada prosedur :

*System* adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk menekankan suatu kegiatan atau menjelaskan suatu sasaran tertentu (Jogiyanto, 2005).

2. *System* yang menekankan pada elemen atau komponennya mendefinisikan *system* sebagai berikut :

*System* adalah kumpulan dari elemen-elemen yang terintegrasi untuk mencapai suatu tujuan tertentu (Jogiyanto, 2005).

##### **2.1.2. Karakteristik *System***

Suatu *system* mempunyai karakteristik atau sifat-sifat yang tertentu, yaitu mempunyai komponen-komponen (*components*), batas *system* (*boundary*), lingkungan luar *system* (*environments*), penghubung (*interface*), *input*, *output*, pengolah (*process*) dan sasaran (*objectives*) atau tujuan (*goal*) (Jogiyanto, 2005).



## **2.2. Konsep Web**

### **2.2.1. Pengertian Web**

*Website* adalah kumpulan dari halaman-halaman situs, yang biasanya terangkum dalam sebuah domain atau subdomain, yang tempatnya berada di dalam *World Wide Web* (WWW) di *Internet*. Sebuah halaman *web* adalah dokumen yang ditulis dalam format HTML (*Hyper Text Markup Language*), yang hampir selalu bisa diakses melalui HTTP, yaitu protokol yang menyampaikan informasi dari *server website* untuk ditampilkan kepada para pemakai melalui *web browser*. Semua publikasi dari *website-website* tersebut dapat membentuk sebuah jaringan informasi yang sangat besar (Kurniawan, 2001).

### **2.2.2. Browser Web**

*Browser web* adalah *software* yang digunakan untuk menampilkan informasi dari *server web*. *Software* ini kini telah dikembangkan dengan menggunakan grafis *user interface*, sehingga pemakai dapat dengan mudah melakukan 'point' dan 'click' untuk pindah antar dokumen.

*Browser* adalah program komputer yang difungsikan untuk mengakses layanan *web*. Setelah ada koneksi dengan *Internet*, pengguna dapat mengunjungi alamat situs *web* yang ingin dituju melalui *system* ini. Jendela *web browser* akan menampilkan halaman *web* tujuan, jika susunan alamat yang dituju sudah benar.

Melalui *web browser*, dapat dipindah ke halaman *web* lain, menelusuri informasi, atau menyimpan informasi yang tersaji dalam suatu situs *web*.

*Web browser* yang populer dan banyak digunakan dewasa ini antara lain *Internet Explorer*, *Netscape Navigator*, *Opera*, dan *Mozilla Firefox* (Kurniawan, 2001).

### 2.2.3. HTML (*Hyper Text Markup Language*)

HTML (*Hypertext Markup Language*) adalah suatu protokol yang menentukan aturan yang perlu diikuti oleh *web browser* dalam meminta atau mengambil suatu dokumen, dan oleh *web server* dalam menyediakan dokumen yang diminta *web browser*.

Protokol ini merupakan protokol standar yang digunakan untuk mengakses dokumen HTML. Sebagai contoh jika pengguna sedang menjelajahi *web* dan melihat tulisan seperti <http://www.unpad.ac.id>, ini merupakan salah satu penggunaan protokol HTTP dalam *web*. Dan dapat dipastikan bahwa pengguna akan berhubungan banyak dengan protokol ini jika sedang mengakses *web* (Kurniawan, 2001).

## 2.3. PHP

Menurut dokumen resmi, PHP merupakan singkatan dari PHP *Hypertext Preprocessor*. Ia merupakan bahasa berbentuk skrip yang

ditempatkan dalam *server* dan diproses di *server*. Hasilnyalah yang dikirim ke klien tempat pemakai menggunakan *browser*.

Secara khusus, PHP dirancang untuk membentuk aplikasi *web* dinamis. Artinya ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, bisa menampilkan isi *database* ke halaman *web*. Pada prinsipnya PHP mempunyai fungsi yang sama dengan skrip-skrip seperti ASP(*Active Server Page*)*Cold Fusion*, ataupun *Perl*. Namun, perlu diketahui bahwa PHP sebenarnya bisa dipakai secara *command line*. Artinya, skrip PHP dapat dijalankan tanpa melibatkan *web server* maupun *browser* (Kadir, 2008).

## **2.4. MySQL**

MySQL (baca : mai-se-kyu-el) merupakan *software* yang tergolong sebagai DBMS (*Database Management System*) yang bersifat *Open Source*. *Open Source* menyatakan bahwa *software* ini dilengkapi dengan *source code* (kode yang dipakai untuk membuat MySQL), selain tentu saja bentuk *executable*-nya atau kode yang dapat dijalankan secara langsung dalam *system* operasi, dan bisa diperoleh dengan cara men-*download* (mengunduh) di *Internet* secara gratis (Kadir, 2008).

## **2.5. Teori Dasar Perancangan System**

### **2.5.1. Analisa System**

Analisa *system* adalah teknik pemecahan masalah yang menguraikan bagian-bagian komponen dengan mempelajari seberapa bagus bagian-bagian komponen tersebut bekerja dan berinteraksi untuk mencapai tujuan mereka.

### **2.5.2. Pengertian Perancangan *System***

Perancangan *System* adalah merancang atau mendesain suatu *system* yang baik, yang isinya adalah langkah-langkah operasi dan proses pengolahan data dan prosedur untuk mendukung operasi *system* (Kristanto, 2001).

Langkah-langkah dalam perancangan *system* meliputi :

#### **1. *Physical System***

*Physical System* berupa bagan alir *system* ataupun bagan alir dokumen.

#### **2. *Logical Model***

Dapat digambarkan dengan menggunakan diagram arus data atau DFD, DFD digunakan untuk menggunakan *system* yang telah ada.

### **2.5.3 Alat Bantu Dalam Perancangan *System***

#### **a. Normalisasi**

Normalisasi merupakan proses pengelompokan data elemen menjadi tabel-tabel yang menunjukkan *entity* relasinya. Pada proses normalisasi selalu diuji pada beberapa kondisi, apakah ada kesulitan pada saat menambah,

menghapus, mengubah ataupun membaca pada suatu *database*. Bila ada kesulitan pada pengujian tersebut, maka relasi tersebut dipecahkan pada beberapa tabel lagi atau dengan kata lain perancangan belumlah mendapat *database* yang optimal.

Beberapa hal yang perlu diketahui dari normalisasi adalah sebagai berikut : (Jogiyanto, 2001)

1. *Candidate Key* (Kunci Kandidat)

Kunci kandidat adalah satu atribut atau satu set minimal atribut yang mengidentifikasi secara unik suatu kejadian spesifik dari *entity*.

2. *Primary Key* (Kunci Primer)

*Primary Key* adalah suatu atribut atau satu set minimal atribut yang tidak hanya mengidentifikasi secara unik suatu kejadian spesifik, tapi juga dapat mewakili setiap kejadian dari suatu *entity*.

3. *Alternate Key* (Kunci Alternatif)

Kunci Alternatif adalah kunci kandidat yang tidak dipakai sebagai *primary key*.

4. *Foreign Key* (Kunci Tamu)

Kunci tamu adalah satu atribut (atau satu set atribut) yang melengkapi satu *relationship* (hubungan) yang menunjukkan ke induknya. Kunci tamu ditempatkan

pada *entity* anak dan sama dengan kunci primer induk direlasikan. Hubungan antara *entity* induk dengan anak adalah hubungan lawan banyak.

b. Bentuk – bentuk Normalisasi

1. *Unnormalized Form* (Bentuk Tidak Normal)

Bentuk tidak normal, kumpulan data yang direkam yaitu berbentuk apa adanya.

2. *First Normal Form*

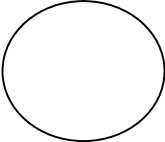
Sebelumnya setiap data dibentuk dalam *file* setelah itu pembentukan data dilakukan secara *record* demi *record*.

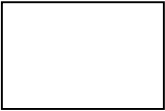
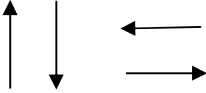
Suatu relasi dikatakan telah memenuhi bentuk normal kesatu jika :

- a) Tidak ada atribut berulang
- b) Harus *atomic*

c. Diagram *Context*

Diagram *context* adalah kasus khusus yang berfungsi memetakan model lingkaran yang dipresentasikan lingkaran tanggal yang mewakili keseluruhan *system* data. Adapun *symbol* yang digunakan adalah sebagai berikut :

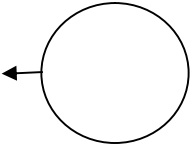
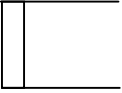
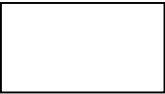
No	Gambar	Nama <i>Symbol</i>
1.		<i>Symbol System</i> Digunakan untuk menunjukkan <i>system</i> yang sedang beroperasi

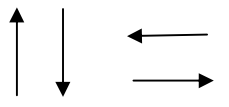
2.		<i>Symbol Terminator</i> Digunakan untuk memberikan <i>input</i> atau menerima <i>output</i> dari <i>system</i>
3.		<i>Symbol Alir</i> Digunakan untuk menunjukkan arus dari proses

Tabel 2.1. Diagram *Context*

d. DFD Levelled

DFD *Levelled* adalah model yang menggambarkan *system* sehingga jaringan kerja antara fungsi yang berhubungan satu sama lain dengan alir penyesuaian data. Adapun *symbol* yang digunakan adalah sebagai berikut :

No	Gambar	Nama <i>Symbol</i>
1.		<i>Symbol Proses</i> Digunakan untuk menunjukkan transformasi dari <i>input</i> menjadi <i>output</i>
2.		<i>Symbol Penyimpanan</i> Digunakan untuk memodelkan kumpulan data atau paket data dari hasil proses maupun penyediaan data untuk proses
3.		<i>Symbol Terminator</i> Digunakan untuk mewakili <i>entity</i> luar, dimana <i>system</i> berkomunikasi

4.		<i>Symbol Alir</i> Digunakan untuk menunjukkan arus dari proses
----	---	--

Tabel 2.2. DFD *Leveled*

e. *Data Dictionary*

*Data Dictionary* adalah *catalog* fakta tentang data dan kebutuhan informasi dari suatu *system* informasi data. *Data dictionary* digunakan untuk membuat pemakai *system* mengerti informasi secara detail dan mengorganisasikan semua elemen data yang digunakan sehingga pemakai dan analisa *system* mempunyai dasar pengertian *system* yang sama tentang *input* dan *output* dari proses.

*Data dictionary* antara lain :

1. Nama Arus Data

Digunakan untuk membaca arus data yang mengalir di program alir data.

2. Alir

Digunakan untuk menuliskan data yang sama tetapi mempunyai nama yang berbeda untuk departemen satu dengan yang lain.

3. Bentuk Data

Digunakan untuk mengelompokkan data *dictionary* ke dalam program data, kegunaannya sewaktu perancangan *system* data.



4. Arus Data

Digunakan untuk mewujudkan dari mana data mengalir dan kemana data akan menuju

5. Penjelasan

Digunakan untuk lebih memperjelas lagi tentang makna arus data yang dicatat di data *dictionary*.

6. Periode

Digunakan untuk menunjukkan kumpulan terjadinya arus data ini.

7. *Volume*

Digunakan untuk mengidentifikasi besarnya simpanan luar yang akan digunakan, kepastian dan jumlah dari alat *input*, alat pemroses dan alat *output*.

f. Struktur Data

Digunakan untuk mengorganisasikan *file* dan untuk menghilangkan *group* elemen yang berulang-ulang. *Symbol* yang digunakan dalam *data dictionary* antara lain sebagai berikut :

<i>Symbol</i>	Keterangan
=	mendefinisikan, menjadi
+	Dan
()	Opsional
{}	Perulangan

□	Memilih
**	Komentar
@	Identifikasi atribut

Tabel 2.3. *Symbol Data Dictionary*


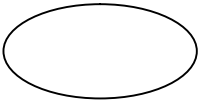
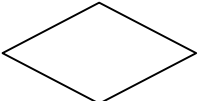
g. *Entity Relationship Diagram (ERD)*


*Entity relationship diagram* merupakan konseptual yang mendefinisikan hubungan antara penyimpanan data (*file data*). *Entity* adalah orang, tempat, kejadian atau konsep yang informasinya dapat direkam.

Fungsi dari ERD adalah antara lain :

1. Untuk memodelkan struktur data dan hubungan antar data.
2. Untuk menguji dalam proses yang harus dilakukan.

*Symbol-symbol* yang digunakan seperti pada tabel berikut :

No	Gambar	Nama <i>Symbol</i>
1.		<i>Symbol Entity</i> Digunakan untuk menggambarkan objek yang dapat didefinisikan dalam lingkungan pemakai.
2.		<i>Symbol Atribut</i> Digunakan untuk menggambarkan elemen-elemen dari suatu <i>entity</i> , yang menggambarkan suatu karakter <i>entity</i> .
3.		Hubungan Digunakan untuk menghubungkan satu sama lain dari suatu <i>entity</i> , hubungan ini disebut <i>relationship</i> .

4.		Garis Digunakan untuk menghubungkan <i>entity</i> dengan atribut
----	---	--

Tabel 2.4. *Symbol Entity Relationship Diagram*

Relasi menurut Harianto Kristanto adalah suatu hubungan antara dua *file* (Himpunan Data) atau lebih yang saling berkaitan.

Relasi juga dapat digambarkan sebagai hubungan antar *entity*.

Macam-macam relasi dapat dibedakan menjadi tiga yaitu :

(Jogiyanto, 2001)

1. Relasi Satu Lawan Satu (*One to One Relationship*).

Jumlah anggota *entity* yang berhubungan dengan *entity* yang lain satu lawan satu.

Contoh : seorang guru mengajar seorang siswa, seorang siswa diajar oleh seorang guru



2. Relasi Satu Lawan Banyak (*One - To - Many Relationship*).

Jumlah anggota *entity* yang berelasi atau jumlah anggota *entity* A yang terlibat dari suatu relasi dengan *entity* B adalah satu, sedangkan jumlah yang terlibat dengan *entity* B adalah banyak.

Contoh :Satu bagian mempekerjakan banyak pegawai, satu pegawai kerja dalam satu bagian.



3. Relasi Banyak Lawan Banyak (*Many – To – Many Relationship*).

Jumlah anggota *entity* yang berelasi dengan yang lain adalah banyak ke banyak.

Contoh : Satu mahasiswa mengambil banyak matakuliah dan satu matakuliah diambil banyak mahasiswa.



h. Desain *Database*

*Database* adalah kumpulan *file* yang mempunyai kaitan antara satu *file* dengan *file* yang lain sehingga membentuk satu bangunan data untuk menginformasikan satu perusahaan atau instansi dalam batasan tertentu.

Untuk tahap mendesain *database* secara umum yang perlu dilakukan adalah mengidentifikasi terlebih *file* yang diperlukan oleh *system* informasi. *File database* yang diperlukan oleh *system* dapat dibaca pada desain model yang digambarkan

dalam bentuk diagram atau data. Adapun langkah-langkah desain dibatasi secara umum yaitu :

1. Menambah kebutuhan *file database* untuk *system* baru.
2. Menentukan *file* dari *database*.

Setelah *file* yang dibuat telah dapat ditentukan maka *parameter* dari *file* selanjutnya juga dapat ditentukan, *parameter* ini meliputi :

1. Tipe dari *file*.
2. Media *file*.
3. *File* kunci dari kunci.

Menurut Harianto Kristanto dalam bukunya Jogiyanto H.M. yang berjudul Analisa dan Desain *System* Informasi tahun 2001, *database* terbentuk dari kumpulan *file* di dalam pemrosesan aplikasi dapat dikategorikan kedalam beberapa tipe diantaranya :

1. *File* Induk (*Master File*)

Dalam aplikasi *file* ini merupakan *file* yang penting, *file* ini tetap ada dalam *system* informasi.

2. *File* Transaksi (*Transacsion file*)

*File* transaksi disebut juga *file input*, digunakan untuk mencatat atau merekam data hasil transaksi yang terjadi.

3. *File* Laporan (*Report File*)

*File* laporan disebut juga *file output*, digunakan untuk menyimpan informasi yang akan ditampilkan dalam laporan.

#### 4. *File Sejarah (History File)*

*File* sejarah disebut juga *file* arsip, digunakan untuk menyimpan informasi yang akan ditampilkan dalam laporan.

#### 5. *File Cadangan (Backup File)*

*File* cadangan berisi salinan *file* basis data yang masih aktif, *file* ini digunakan sebagai cadangan atau pelindung bila *file* utama dari basis data rusak atau tidak bisa diakses.

#### 6. *File Kerja (Working File)*

*File* kerja disebut juga *file* sementara, digunakan sementara oleh program bila memori tidak mencukupi. Setelah *system* selesai dijalankan maka *file* ini akan dihapus kembali.

Data yang terdapat pada perancangan *database* antara lain sebagai berikut :

##### 1. Karakter

Merupakan bagian dari data yang terkecil dapat berupa *numeric*, huruf ataupun karakter-karakter khusus yang membentuk suatu *item* data.

##### 2. *Field*

Menggunakan suatu atribut dari *record* yang menunjukkan suatu *item* data seperti nama, alamat dan sebagainya.

Kumpulan dari *field* membentuk suatu *record*.

### 3. Record

Merupakan kumpulan dari beberapa *field* yang menggunakan suatu *unit* data inti tertentu.

### 4. File




Merupakan kumpulan dari beberapa *record* yang menggambarkan suatu kesatuan tertentu.

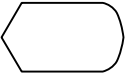
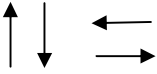


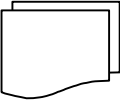
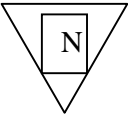
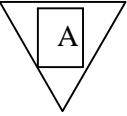
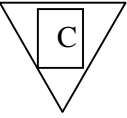
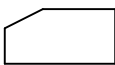
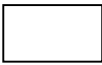
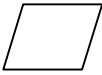
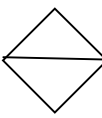
### 5. Database

Merupakan kumpulan dari *file* yang membentuk suatu *database*.


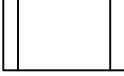
#### i. System Procedure

*System Procedure* digunakan untuk mendefinisikan hubungan antara tiap bagian, proses dan aliran data (dalam bentuk *output* dan *input*). Adapun *symbol* yang digunakan adalah sebagai berikut :

<i>Symbol</i>	Keterangan
	Menunjukkan <i>input / output</i> menggunakan <i>hardisk (database)</i>
	Menunjukkan <i>input / output</i> menggunakan <i>disket (stored data)</i>
	Menunjukkan <i>input / output</i> menggunakan pita kertas berlubang ( <i>papertape</i> )
.....	Menunjukkan <i>input</i> yang menggunakan <i>line keyboard (manual input)</i>

	Menunjukkan <i>output</i> yang ditampilkan dimonitor ( <i>display</i> )
	Menunjukkan arus proses
	Menunjukkan penghubung ke halaman yang sama ( <i>connector</i> )
	Menunjukkan penghubung ke halaman yang berbeda ( <i>off-page connector</i> )
	Menunjukkan dokumen <i>input / output</i> baik proses <i>manual</i> , mekanik atau komputer ( <i>document</i> )
	<i>File</i> non-komputer yang diarsip secara urut angka atau <i>numeric</i>
	<i>File</i> non-komputer yang diarsip secara urut
	<i>File</i> non-komputer yang diarsip secara urut dengan tanggal
	Menunjukkan <i>input / output</i> yang menggunakan kartu plong ( <i>punched card</i> )
	Menunjukkan kegiatan proses dan operasi <i>system computer</i> (proses)
	Menunjukkan data <i>system computer</i> (data)
	Menunjukkan proses pengurutan data diluar <i>computer</i> ( <i>sort</i> )



	Menunjukkan <i>input / output</i> menggunakan pita <i>magnetic</i>
	Menunjukkan proses yang tidak didefinisikan

Tabel 2.5. *Symbol Flow of Document*

### BAB III

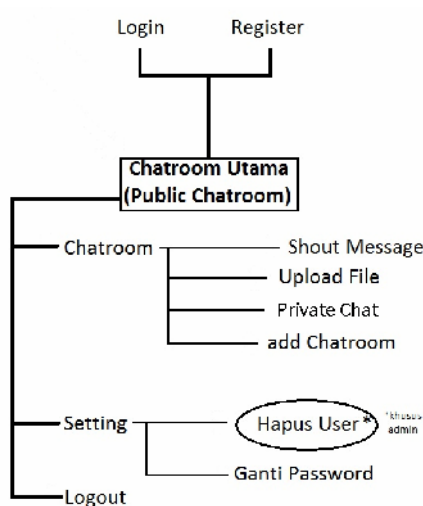
## PERANCANGAN SYSTEM

### 3.1. Deskripsi System

Merupakan aplikasi yang dapat melakukan komunikasi lebih dari dua arah menggunakan media *browser* dengan bentuk pesan tulisan. Untuk menggunakannya *user* harus melakukan registrasi singkat. Setelah itu *user* bisa memasuki aplikasi *chat room*. *Chat room* adalah sebuah ruang maya untuk bercakap-cakap, sedangkan nantinya masing-masing *user* dapat membuat *chat room* sendiri jadi akan terdapat banyak ruang maya yang akan digunakan. Untuk memperjelas, digambarkan dengan bagan berikut penjelasannya.

### 3.2. Rancangan Aplikasi

Berikut adalah gambaran *system* :



Gambar 3.1. Rancangan System

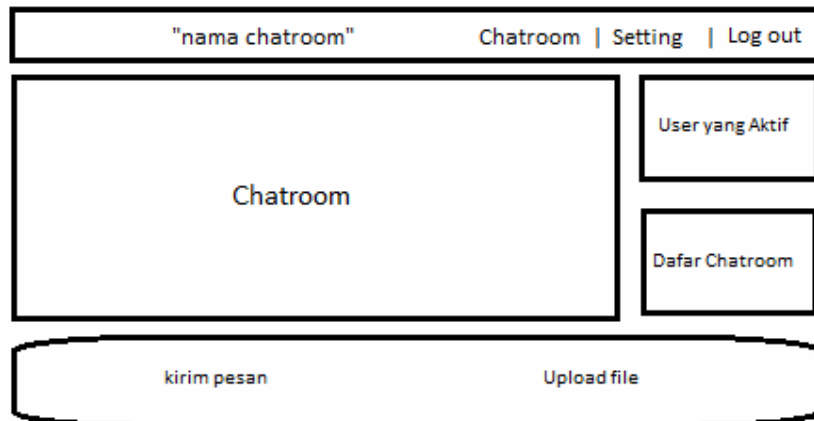
Yang pertama adalah *home page*. Di halaman awal ini terdapat 2 menu, yakni :

1. *Register*

Ini adalah menu untuk mendaftar agar bisa masuk ke aplikasi. Yang harus diisi *user* adalah *username* untuk *login*, *nickname chatting* (nama samaran untuk bercakap-cakap), dan *password*. *Password* diketik ganda untuk memantapkan *user*.

2. *Login*

Adalah menu untuk memasuki aplikasi, setelah *user* melakukan registrasi. Jika *user* merasa sudah pernah mendaftar maka bisa langsung *login*. Setelah *login*, *user* akan langsung menuju *chat room*.



Gambar 3.2. Rancangan menu *Chat room*

Setelah memasuki *chat room* terdapat berbagai menu yang terdapat diatas, tengah, dan bawah, yaitu:

1. Dibagian atas terdapat “Nama *Chat room*”, karena ini adalah awal *user login*, *user* memasuki *chat room* utama bernama *Public Chat room* maka namanya menjadi *PUBLIC CHAT ROOM*. Disebelahnya terdapat menu “*Chat room*” menu untuk kembali ke *chat room* utama. Sebelahnya lagi adalah menu “*Setting*” untuk mengganti *password* dan menghapus pengguna (khusus admin). Dan yang terakhir adalah menu “*Logout*” untuk keluar.
2. Di tengah terdapat kolom yang menampilkan total pengguna yang sedang aktif, jika diklik akan menampilkan *private chat*. Dan kolom dibawahnya merupakan menu untuk membuat *chat room*, dan menampilkan *chat room* terkini.
3. Di bawah terdapat menu kirim pesan. Saat pengguna menulis pesan akan ditampilkan di kolom “*Chat room*” yang terdapat dibagian tengah. *User* juga dapat meng-*upload file* ke dalam *chat room* agar bisa diunduh oleh pengguna lain dengan klik menu “*Upload File*”. Setelah proses kirim *file* selesai maka akan ditampilkan *link download* pada *chat room* agar bisa diunduh pengguna lain.

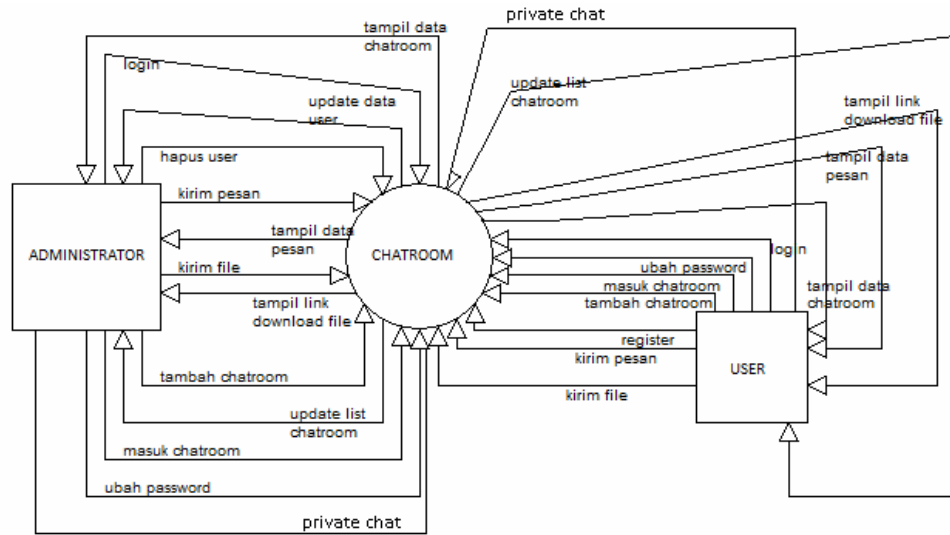
Untuk Admin, fungsi yang bisa dijalankan hampir sama dengan *user* biasa hanya ditambahkan fungsi menghapus pengguna. Fungsi ini bertujuan untuk mengendalikan kondisi *chat room* jika sewaktu-waktu ada *user* yang menimbulkan ketidaknyamanan terhadap *user* lain. Untuk itu dalam bab ini akan dibahas cara kerja *system* melalui DCD (*Data Context Diagram*), DFD (*Data Flow Diagram*) dan *Flow Chart*.

### 3.3. Pemodelan Fungsional

Untuk memodelkan seluruh fungsi yang tercakup dalam aplikasi ini, digunakan *Data Context Diagram* (DCD), *Data Flow Diagram* (DFD) serta *Flow Chart*. DCD dapat juga dikatakan sebagai *DFD Level 0*. Untuk DFD merupakan penjabaran lebih lanjut dari DCD. DFD berguna untuk menggambarkan fungsi-fungsi yang mentransformasikan data, serta berguna untuk menggambarkan bagaimana data ditransformasikan pada perangkat lunak. *Flow Chart* untuk memudahkan melihat proses operasi dan pengolahan data.

#### 3.3.1. *Data Flow Diagram Level 0*

Ada 2 *entity* luar dalam aplikasi ini, yaitu admin, pihak yang bertanggung jawab atas semua *system* operasi informasi. Meliputi seleksi *input* data yang diperoleh dari pihak yang dipercaya / informan, sampai memastikan seluruh *system* dapat bekerja secara akurat. Untuk dapat menggunakan aplikasi, admin diharuskan melakukan *login* terlebih dahulu, untuk menjamin sebagai pihak yang berwenang. Data-data yang dikirimkan saat *login* adalah data nama pengguna dan sandi. Apabila proses login gagal, maka aplikasi akan mengembalikan pesan *error*. Setelah berhasil *login*, admin dapat menjalankan aplikasi.

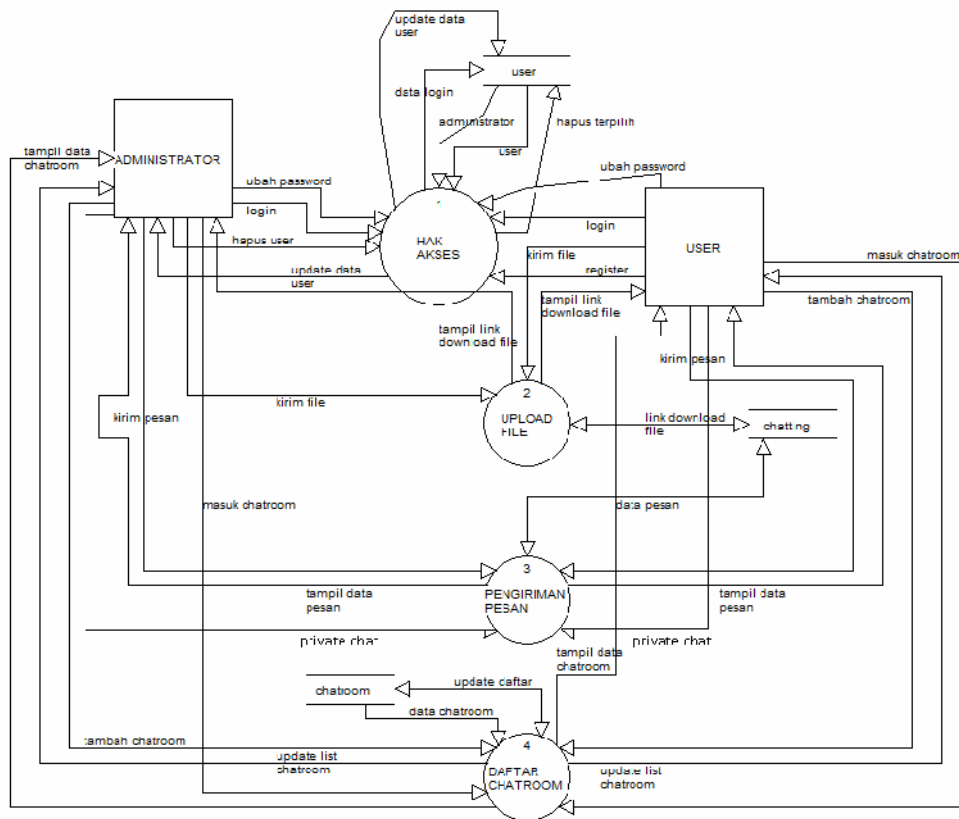


Gambar 3.3. Data Flow Diagram Aplikasi Chat room Level 0

### 3.3.2. Data Flow Diagram Level 1

Dalam DFD level 1 ini, aplikasi terdiri dari 2 proses, yaitu :

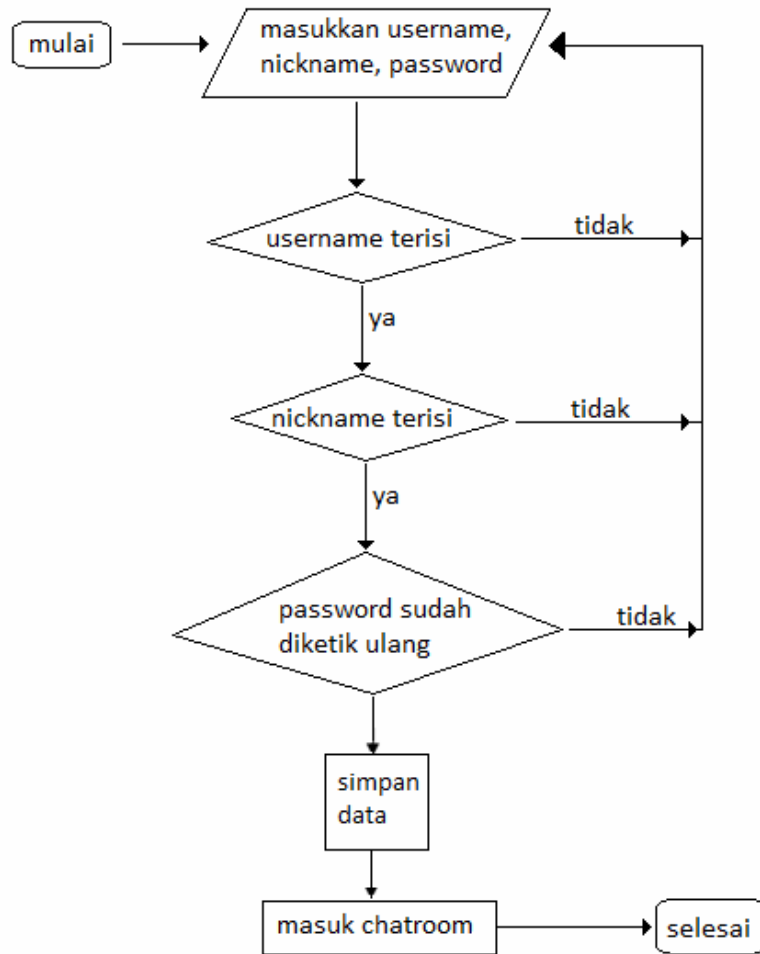
- a. *Sub proses* otentikasi *Administrator* menangani otentikasi. Melibatkan data *store* berupa nama dan sandi. Alur data yang masuk ke proses ini adalah data admin dan alur data yang keluar dari proses ini yaitu informasi mengenai *login*.
- b. *Sub proses* untuk mengirim pesan. Merupakan proses pengiriman pesan *chatting*. Melibatkan data pesan.
- c. *Sub proses* untuk mengirim *file*. Merupakan proses *upload file*. Melibatkan data *file*.
- d. *Sub proses* untuk menghapus *user*. Merupakan proses penghapusan data *user* tertentu, sehingga *user* tidak bisa *login*. Melibatkan data *user*.



Gambar 3.4. Data Flow Diagram Aplikasi Chat room Level 1

### 3.3.3. Flow Chart

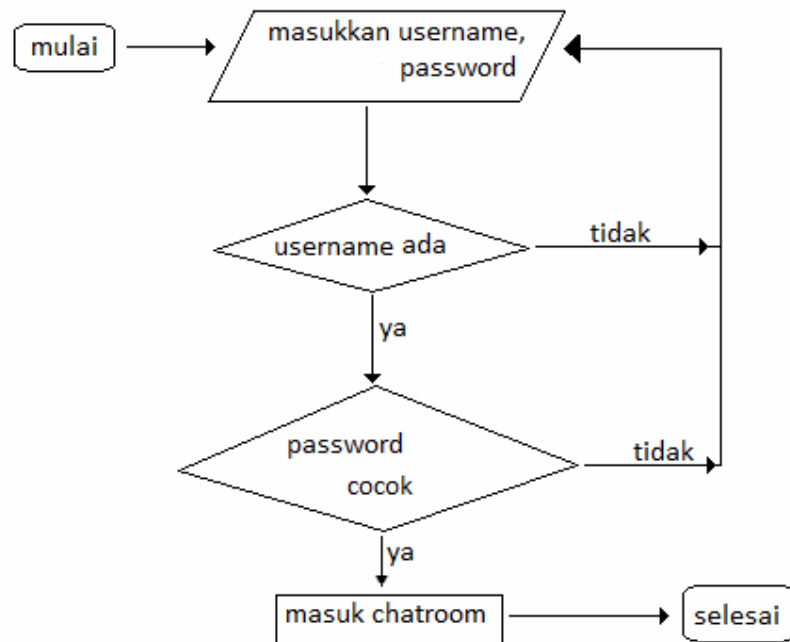
Untuk memudahkan melihat langkah-langkah atau proses dari fungsi-fungsi yang dijalankan pada aplikasi, dapat dilihat pada *Flow Chart* berikut:



Gambar 3.5. *Flow Chart* proses Registrasi

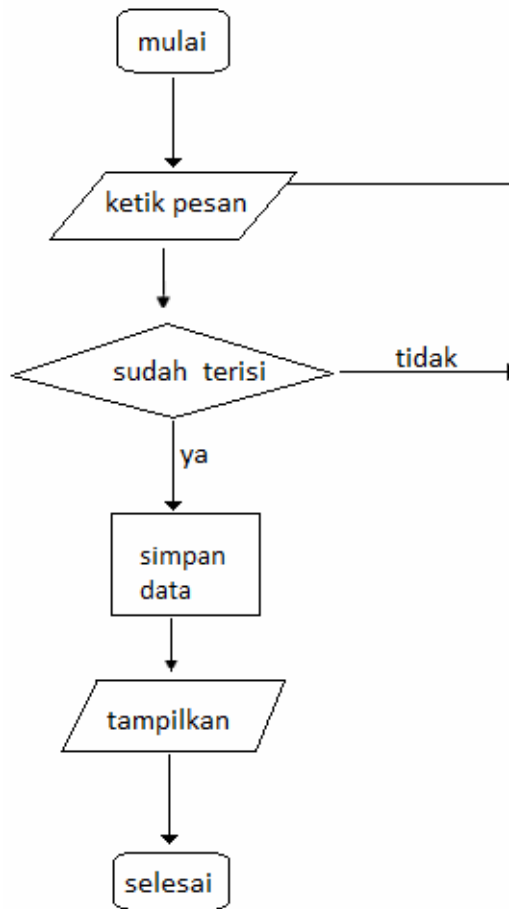
Isi *username, nickname, password* kemudian ketika *username, nickname, password*, kosong maka tidak akan melakukan apapun, namun jika sudah terisi semua maka data akan tersimpan dan pengguna masuk ke aplikasi.





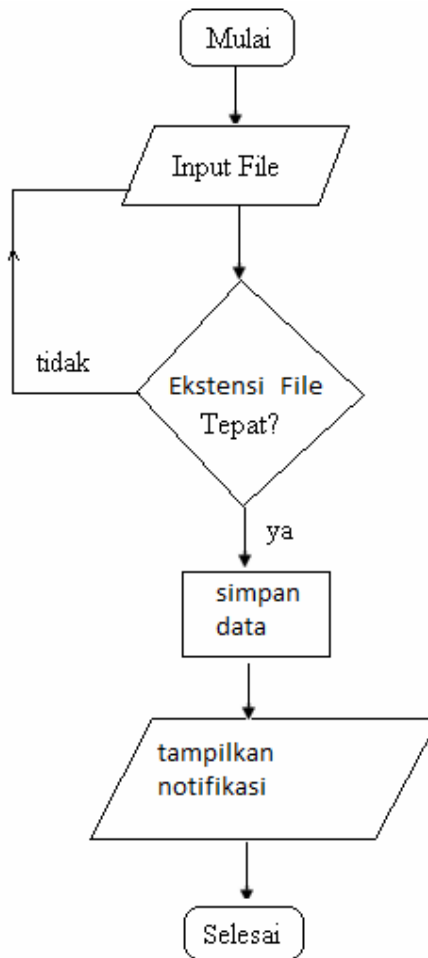
Gambar 3.6. *Flow Chart* proses login

Pertama masukan data *username* dan *password*, jika *username* dan *password* kosong tidak melakukan apapun, jika terisi data *login* akan dicocokkan dengan *database* yang ada, bila data benar pengguna menuju *chat room*.



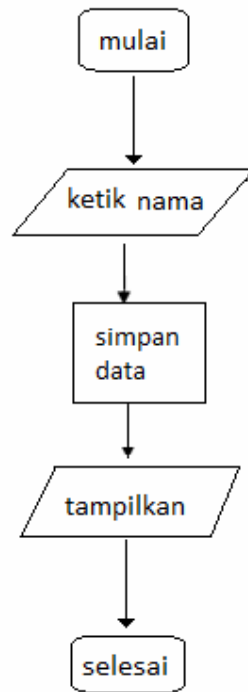
Gambar 3.7. *Flow Chart* kirim pesan

Masukan pesan, jika masih kosong tidak melakukan apapun, jika sudah terisi pesan akan disimpan di *database*, kemudian ditampilkan di pengguna.



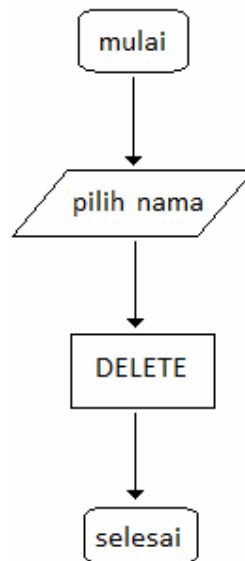
Gambar 3.8. *Flow Chart* proses *upload file*

Pertama pilih *file* yang di-*upload*, jika ekstensi *file* tidak tepat maka tidak akan meng-*upload*, jika tepat, maka *file* akan di-*upload* ke *server* dan ditampilkan *download link file* tersebut.



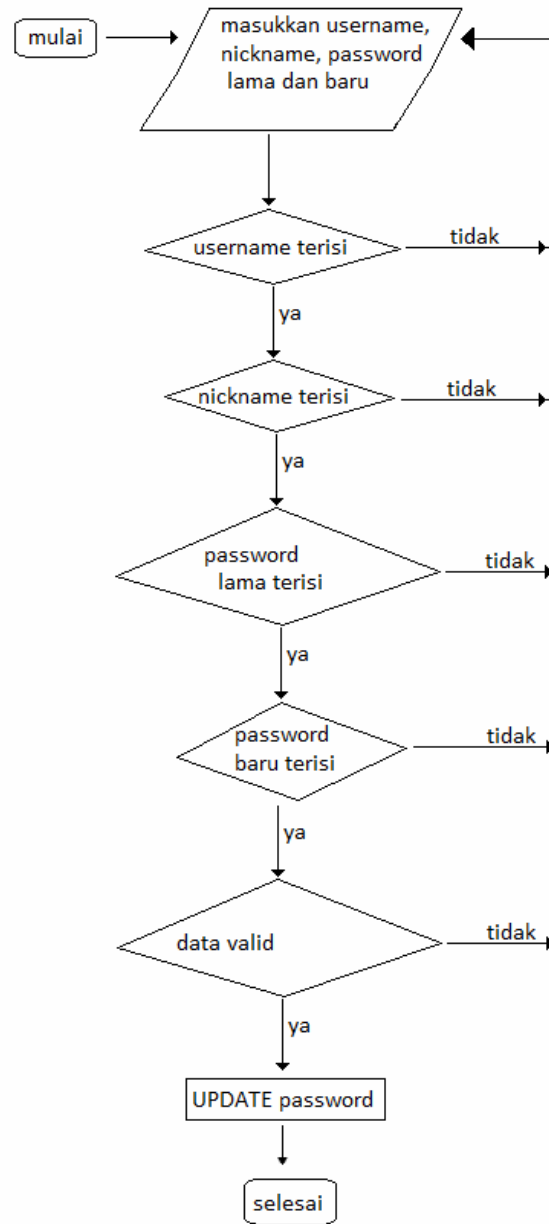
Gambar 3.9. *Flow Chart* tambah chat room

Masukan nama, tekan ok, maka data tersimpan, tampilkan



Gambar 3.10. *Flow Chart* hapus pengguna

Klik pada *username*, maka pengguna tersebut akan terhapus.



Gambar 3.11. *Flow Chart* ubah *password*

Masukkan *username*, *nickname*, *password* lama dan baru, jika semuanya belum terisi tidak akan melakukan apapun, jika sudah terisi semuanya, data dicocokkan, bila tidak cocok tidak akan melakukan apapun, bila cocok data akan dirubah.



Gambar 3.12. Flow Chart private chat

Untuk memulai percakapan privat, klik nama pengguna, maka akan menampilkan halaman baru percakapan hanya dua orang.

### 3.3.4. Perancangan Database

Database adalah suatu tempat untuk menyimpan berbagai informasi atau data dalam jumlah yang sangat banyak sehingga dapat diakses dan ditampilkan secara fleksibel.

#	Column	Type	Collation	Attributes	Null	Default	Extra
1	Uid*(primary key)	int(9)			No	None	AUTO_INCREMENT
2	uname	varchar(20)	latin1_swedish_ci		No	None	
3	nickname	varchar(10)	latin1_swedish_ci		No	None	
4	password	varchar(32)	latin1_swedish_ci		No	None	
5	isadmin	enum("0","1")	latin1_swedish_ci		No	0	
6	olstatus	enum("0","1")	latin1_swedish_ci		No	0	
7	log	Timestamp			No	CURRENT_TIMESTAMP	

Tabel 3.1. Struktur tabel user

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	Nomor*(primary key)	int(9)			no	none	AUTO_INCREMENT

2	chatterid	int(9)			no	none	
3	roomid	int(9)			no	none	
4	postedtext	varchar(160)	latin1_swedish_ci		no	none	
5	tanggal	datetime			no	none	
6	log	timestamp			no	CURRENT_TIMESTAMP	

Tabel 3.2. Struktur tabel *chatting*

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	crid*(primary key)	int(9)			no	none	AUTO_INCREMENT
2	Creatorid	int(9)			no	none	
3	crname	varchar(160)	latin1_swedish_ci		no	none	
4	createdat	timestamp			no	CURRENT_TIMESTAMP	

Tabel 3.3. Struktur tabel *chatroom*

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	id*(primary key)	int(11)			no	none	AUTO_INCREMENT
2	From	int(11)			no	none	
3	To	int(11)			no	none	
4	Msg	Text	latin1_swedish_ci		no	none	
5	Time	Timestamp		on update CURRENT_TIMESTAMP	no	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP
6	seen	Enum('0','1')	latin1_swedish_ci		no	0	

Tabel 3.4. Struktur table *private*

### 3.3.5. Pemilihan Kunci *Record*

Dari tabel-tabel diatas, kemudian dilakukan pemilihan *field-field* yang akan dipergunakan sebagai kunci *record* atau *primary key* yang nantinya dipergunakan sebagai basis relasi antar tabel. Kunci *record* merupakan *record* di dalam tabel yang bersifat unik (*unique*), artinya di dalam satu *field* yang memuat kunci *record* hanya memuat satu *record* tertentu saja. Dalam tabel-tabel di atas, dapat ditentukan kunci *record* sebagai berikut:

- a. uid
- b. nomor
- c. crid
- d. id

## **BAB IV IMPLEMENTASI**

Setelah aplikasi dirancang dan diimplementasikan ke dalam pemrograman yang sesungguhnya, maka langkah berikutnya adalah melakukan pengujian. Hal ini untuk membuktikan tingkat keberhasilan program.

### **4.1. Implementasi Perangkat Keras**

Sebelum melakukan pengujian, perlu dipersiapkan alat yang mendukung baik perangkat keras maupun perangkat lunak. Perangkat keras yang digunakan adalah sebagai berikut:

1. *Laptop/Komputer minimal mempunyai 2 core*
2. *RAM minimal 1 GB pada windows 7*
3. *Harddisk*
4. *VGA*
5. *LCD Monitor*
6. *Keyboard dan Mouse/Touchpad laptop standart*

Sedangkan perangkat lunak yang digunakan adalah sebagai berikut:

1. *Windows 7*
2. *Browser Mozilla Firefox*
3. *phpMyAdmin*
4. *Database Server MySQL*
5. *Web Server*

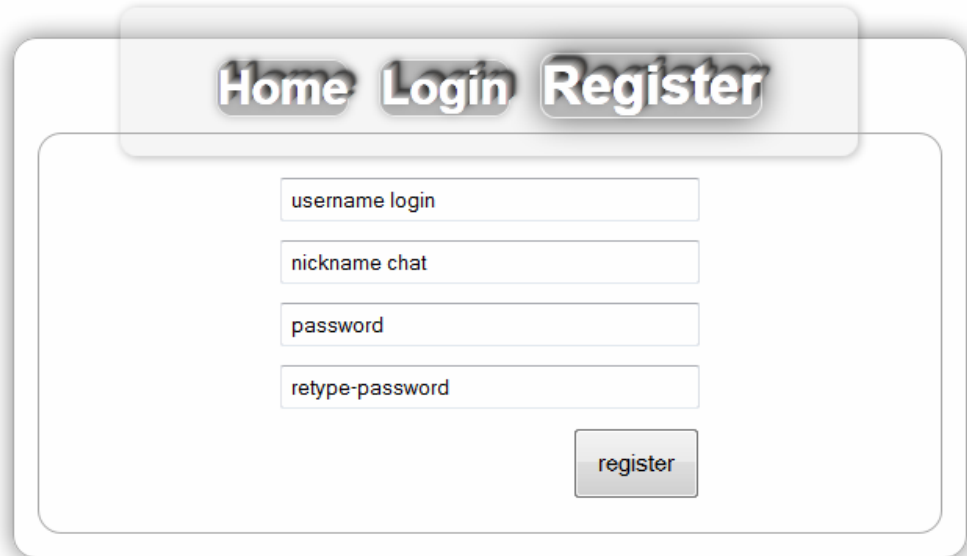


## 4.2. Implementasi Program

Antarmuka (*interface*) aplikasi untuk *system* yang dibuat berbasis *web*. Pada aplikasi ini, admin dapat memasukkan (*input*) pesan, mengupload *file*, membuat *chat room*, mengedit *password* dan menghapus *user* tertentu. Adapun hasil pengujian *system* adalah sebagai berikut:

### 4.2.1. Halaman Registrasi

Ketika aplikasi pertama kali dipanggil atau dijalankan, maka yang akan tampil adalah halaman *login* admin. Namun apabila pengguna belum mempunyai akun maka dapat melakukan registrasi singkat.



The image shows a registration form with a header containing three buttons: 'Home', 'Login', and 'Register'. Below the header, there are four input fields: 'username login', 'nickname chat', 'password', and 'retype-password'. A 'register' button is located at the bottom right of the form.

Gambar 4.1. Tampilan halaman registrasi

```
function kirim()
{
    var un = $('#textbusername').val();
    var nc = $('#textbnickname').val();
```

```

var pw = $('#textbpassword').val();
var pw2 = $('#textbrepass').val();
if (un != "" && pw != "" && nc != "" && pw2
!= "")
{
$('##notif').load('login/ENGINE.php',{newuser:u
n,newnick:nc,newpass:pw,newpass2:pw2});
}
}

```

*function kirim()* berfungsi untuk pendaftaran, jika *un/username*, *nc/nickname*, *pw/password*, *pw2* tidak sama dengan kosong maka program lanjut berjalan ke proses *login*. Jika tidak maka tidak melakukan apapun.

Jika tombol “*register*” sudah diklik maka akan melakukan fungsi *kirim()*

```

$('##registerbtn').click(function(){
    kirim();
});

```

#### 4.2.2. Halaman *Login*

Pada halaman *login*, seorang admin diharuskan untuk mengisi *username* dan *password* untuk dapat menjalankan aplikasi lebih lanjut. Pada Aplikasi ini admin *login* menggunakan *username* root dan *password* pwn3d. Halaman *login* admin dapat dilihat pada Gambar 4.2.



Gambar 4.2. Tampilan halaman *login*

```

function kirim()
{

```

```
var un = $('#tbusername').val();
var pw = $('#tbpassword').val();
if (un != "" && pw != "")
{
$('#notif').load('login/ENGINE.php',{uname:un,
passwd:pw});
}
}
```

Jika *un/username* dan *pw/password* tidak kosong maka program akan berjalan, jika kosong tidak melakukan apapun.

function kirimOnreturn(evt)

```
{
if (evt.keyCode == 13 )
{
kirim();
}
}
```

Dalam hal ini, *evt.keyCode == 13* adalah kode dari *enter*, dengan demikian jika pengguna menekan *enter* akan melakukan fungsi *kirim()* , melanjutkan proses *login* yakni menuju *chat room* utama.

#### 4.2.3. Halaman *Chat room*

Apabila *username* dan *password* yang diisikan oleh admin sudah benar, maka akan tampil halaman antarmuka *chatting* dan menu pilihan sebagai tampilan awal aplikasi. Admin bisa mengirim pesan melalui *message bar* dan menekan tombol *shout*, admin juga bisa *upload file* ke dalam *chat room*. Halaman *chat room* dapat dilihat pada Gambar 4.3.



Gambar 4.3. Halaman *chat room*

```
<?php
if ($_GET['room'] != "")
{
    $scrid = $_GET['room'];
}
else
{
    $scrid = 0;
}
?>
```

Ini fungsi yang memunculkan nama *chat room* yang sedang digunakan, dalam hal ini adalah *PUBLIC CHAT ROOM*. Yaitu adalah *room* utama yang pertama dimasuki *user* setelah *login* berhasil.

```
function kirim()
{
    var txbx = $('#textbox').val();
    if (txbx != "")
    {
        $('#chatting').load('chat
room/chatting.php?room=<?php echo $scrid;
?>', {textcht:txbx},function(){
        $('#textbox').val("");
    });
    }
}
```

*function kirim()* diatas adalah fungsi untuk mengirim pesan ke dalam *chat room*.

```

<?php $timestamp = time();?>
var usertarget = 0;
$('#usersendto').change(function(){
    usertarget = $(this).val();
    //alert(usertarget);
});
$('#newcrbtn').click(function(){
    var cname = $('#newcrtxt').val();
    $('#crlst').load('chat
room/crlst.php',{ newcname:cname },function()

```

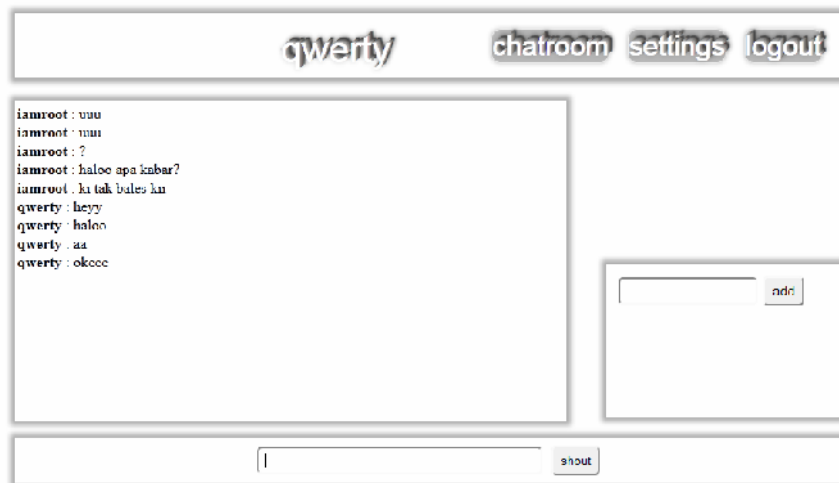
Diatas adalah fungsi menambahkan *chat room* dan menampilkan nama-nama *chat room* yang ada.

```

$('#olusers').load('chat room/olusers.php');
$('#btnpost').click(function(){
    kirim();
});
setInterval(function(){
    $('#chatting').load('chat
room/chatting.php?room=<?php echo $scrid; ?>');
    $('#olusers').load('chat room/olusers.php');

```

Diatas adalah fungsi menampilkan total *user* yang sedang *online*.



Gambar 4.4. Halaman *Private Chat*

```

function showPrivateConv($from,$to,$n=15)
{
    $q = "SELECT c.msg,u1.nickname as dari,u2.nickname as ke from
(SELECT * FROM private pv WHERE (pv.`from`='$from' and
pv.`to`='$to') or (pv.`from`='$to' and pv.`to`='$from') order by time desc

```

```

limit 0,$n) as c join user u1 on c.`from` = u1.uid join user u2 on
c.`to`=u2.uid group by c.id order by c.time asc";
//echo $q;
$xq = mysql_query($q) or die(mysql_error());
while($isi = mysql_fetch_array($xq)
{
    echo
"<strong>".htmlentities(trim($isi['dari']),ENT_QUOTES)."</strong> :
".htmlentities(trim($isi['msg']),ENT_QUOTES)."<br>";
}
}

```

Fungsi diatas adalah fungsi memulai *private chat* saat pengguna mengklik nama pengguna lain, kemudian menampilkan pesan. Pada halaman ini fungsi daftar pengguna yang sedang *online* dihilangkan.

```

unction isReplied($currentUser,$chatwith)
{
    $q = "select `from` from private pv where pv.`from`=$chatwith'
and pv.to=$currentUser' and seen='0' order by time desc limit 0,1";
    $xq = mysql_query($q) or die(mysql_error());
    $isi = mysql_fetch_array($xq);
    $jml = mysql_num_rows($xq);
    if ($isi[0] != $currentUser and $jml>0)
    {
        return "*";
    }
    else
    {
        return "";
    }
}

```

Ketika ada pesan baru, maka nama pada pengguna pengirim pesan pribadi akan muncul tanda bintang “\*”.

```

function iveSeen($me,$someonetryingtalktome)
{
    mysql_query("update private set seen='1' where
`from`='$someonetryingtalktome' and `to`='$me'") or
die(mysql_error());
}

```

Ketika pesan pribadi yang terkirim sudah dilihat maka tanda bintangnya akan hilang.

#### 4.2.4. Halaman Setting Dan Daftar User

Halaman Setting merupakan pilihan untuk mengganti *password*. Sedangkan halaman daftar *user* merupakan halaman untuk melihat *user* yang telah terdaftar dan admin dapat menghapusnya dari keanggotaan, dengan menekan tombol *delete* pada *user* yang dikehendaki.

The screenshot shows a web interface with a navigation bar at the top containing 'chatroom', 'settings', and 'logout' buttons. Below the navigation bar is a form for updating user information. The form contains four input fields: 'username', 'nickname', 'new password', and 'old password'. Below these fields is an 'update' button. Underneath the form is a table listing users with columns for 'username', 'nickname', and 'delete?'. The table contains four rows of user data.

username	nickname	delete?
information	information	<a href="#">delete</a>
jackie.chan	jackie	<a href="#">delete</a>
joker	joker_batman	<a href="#">delete</a>
n00bz	n00bee	<a href="#">delete</a>

Gambar 4.5. Halaman *Setting* dan daftar *user*

```
$('#notif').load('chat
room/ENGINE.php',{newuser:un,newnick:nc,ne
wpass:pw,oldpass:pw2});
}
});
$('#deluser').click(function(){
var uid = $(this).attr('href').replace('#delete-
');
$('#crcontent').empty();
```

Diatas adalah jika variabel *nickname*, *password* lama, *password* tidak kosong dan *user* mengklik tombol “*update*” serta data yang diisikan cocok maka data *password* lama akan berubah diganti dengan *password* baru. Dan dibawah ini adalah fungsi menghapus *user* pada admin.

```
$('#notif').load('chat
room/ENGINE.php',{newuser:un,newnick:nc,ne
wpass:pw,oldpass:pw2});
```

```

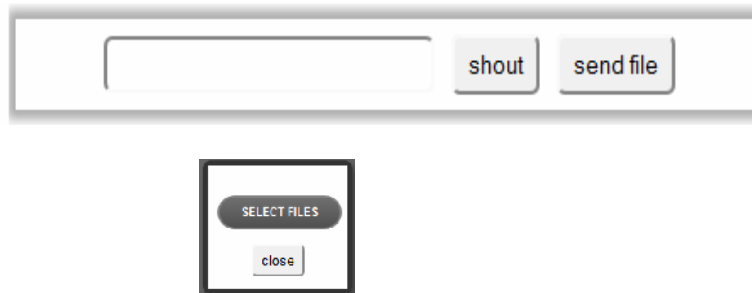
$(.deluser').click(function(){
    var uid = $(this).attr('href').replace('#delete-
    ');
    $('#crcontent').empty();

```

Jika nama *user* yang dikehendaki diklik maka akan melakukan fungsi *\$uid* pada *modules/functions.php* yaitu menghapus pengguna dari *database*

#### 4.2.5. Halaman *Upload File*

Halaman *upload file* merupakan, halaman untuk mengirim atau meng-*upload file* ke dalam *chat room* dengan menekan tombol *send file*, dan memilih *file* yang akan di-*upload*. Pada fungsi *upload file* ini menggunakan *plugin flash uploadify*.



Gambar 4.6. Halaman *upload file*

```

ob_start();
session_start();
include_once '.././../modules/functions.php';
$targetFolder = '/www/crutcrud/uploads';

```

Menentukan letak dimana *file* akan di-*upload*.

```

$fileTypes =
array('jpg','jpeg','gif','png','bmp','doc','docx','
xls','xlsx','ppt','pptx','rar','exe');

```

Menentukan *file* dengan ekstensi apa saja yang boleh di-*upload*.



## **BAB V PENUTUP**

### **5.1. Kesimpulan**

Dari pembuatan aplikasi *chat room* berbasis *web* ini, serta pengujian terhadap aplikasi yang dibuat dapat disimpulkan secara rinci sebagai berikut:

- a. Berbagi *file* lebih mudah dan cepat daripada menggunakan *email*. Untuk pengguna yang ingin mengirimkan *file* kepada seseorang, terutama yang belum mempunyai *email* akan lebih mudah dan cepat menggunakan aplikasi *chat room* ini.
- b. Terdapat dua jenis hak akses pada *system*, yaitu sisi *User* dan sisi *Administrator* yang terdapat fungsi menghapus *user*. Fungsi ini dimaksudkan, bila terdapat seorang pengguna yang liar. Dalam arti berbuat onar seperti menghina pengguna lain, maka *Administrator* dapat menghapus pengguna (*kick*) tersebut dari dalam *chat room* agar tidak bisa menggunakannya. Sehingga kondisi *chat room* dapat dikendalikan.
- c. Saat memasuki *chat room* yang lain dan ingin kembali ke *chat room* utama (*public chat room*) harus menggunakan tab yang sebelumnya, jika tidak akan terjadi kesalahan tampilan yaitu tulisan yang berkedip-kedip dan fungsi kirim *file* akan tampak tanpa diklik.
- d. Karena batasan pengguna yang tidak dibatasi maka dapat menyebabkan aplikasi berjalan lambat saat pengguna berjumlah sangat banyak.
- e. Kadang daftar *user* dan daftar *chat room* agak lama termuat.

## 5.2. Saran

Setelah dilakukan pengujian terhadap aplikasi ini terdapat beberapa *bug* dan kekurangan, dijelaskan sebagai berikut:

1. Karena tidak terdapat opsi pengiriman *file* secara privat, hendaknya terdapat opsi kirim *file* dengan cara mengklik pada nama *user* yang dituju.
2. Jumlah pengguna harus dibatasi dalam jumlah yang ditentukan, agar tidak lambat.
3. Perlu dibenahi fungsi *logout* agar berhasil dengan satu kali klik. Serta perlu dibenahi tampilan opsi kirim *file* agar tidak muncul dengan sendirinya.

## DAFTAR PUSTAKA

- Jogiyanto, HM. (2005), *Analisis dan Desain Sistem Informasi*, ANDI Offset, Yogyakarta.
- Kadir, Abdul. (2002), *Dasar Pemrograman WEB Dinamis Menggunakan PHP*, ANDI Offset, Yogyakarta.
- Kadir, Abdul. (2002), *Penuntun Praktis Belajar SQL*, ANDI Offset, Yogyakarta.
- Kadir, Abdul. (2009), *Membuat Aplikasi Web dengan PHP + Database MySQL*, ANDI Offset, Yogyakarta.
- McFarland, David Sawyer. (2013), **CSS3: The Missing Manual, Third Edition**, O'Reilly Media, California.
- Supriyanto, Aji. (2005), *Pengantar Teknologi Informasi*, Salemba Infotek, Jakarta.
- W, A.S. (2011), *Website Super Canggih dengan Plugin jQuery Terbaik*, Mediakita, Jakarta.

## LAMPIRAN

Index.php

```
<?php
//INI ADALAH HALAMAN LOGIN!!!!!!!!!!!!!!
session_start();
include_once 'modules/dbcon.php';
include_once 'modules/functions.php';
if(is_logged_in() === true)
{
    //header('location:"chatroom.php"');
    redirectTo('chatroom.php');
}
else
{
    //stay here
    echo '
    <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JACKIE</title>
    <script type="text/javascript" src="others/js/jquery.min.js"></script>
    <script type="text/javascript" src="others/js/functions.js"></script>
    <script type="text/javascript" src="others/js/login.js"></script>
    <link rel="stylesheet" type="text/css" href="others/css/style.css">
    <link rel="stylesheet" type="text/css" href="others/css/login.css">
</head> <body>
```

```

<center>
<div id="notif"></div>
<div id="mainwrapper">
  <div id="top" class="transbg">
    <!-- menu pakek <a ... -->
    <a href="#home" id="home" class="loginmenu">Home</a>
    <a href="#login" id="login" class="loginmenu">Login</a>
    <a href="#register" id="register" class="loginmenu">Register</a>
    <a href="#about" id="about" class="loginmenu">Petunjuk Penggunaan</a>
  </div>
  <div id="contentmenu" class="wrapper">;
    include_once 'login/home.php';
  echo '
      </div>
    </div>
  </center>
</body>
</html>
  ';
}??>

```

chatroom.php

```

<?php
session_start();
include_once 'modules/dbcon.php';
include_once 'modules/functions.php';
if(is_logged_in() === true)
{

```

```
//print_r($_SERVER);

    //ok

    $scrid = (int) $_GET['room'];

    echo '

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>

    <head>

        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

        <title>JackieC - '.getChatroomname($scrid).'
```

```

    <a id="judull" class="crmenu">'.getChatroomname($scrid).'

```

chatroom/chatroom.php

```

<script>
//----- fungsi lokal -----
<?php

```

```
if ($_GET['room'] != '')
{
    $scrid = $_GET['room'];
}
else
{
    $scrid = 0;
}
?>
function kirim()
{
    var txbx = $('#textbox').val();
    if (txbx != '')
    {
        $('#chatting').load('chatroom/chatting.php?room=<?php echo $scrid; ?>', {textcht:txbx}, function(){
            $('#textbox').val("");
        });
    }
}

function kirimOnreturn(evt)
{
    if (evt.keyCode == 13 )
    {
        kirim();
    }
}

//-----
```



```

$(document).ready(function(){

    <?php $timestamp = time();?>

    var usertarget = 0;

    $('#usersendto').change(function(){

        usertarget = $(this).val();

        //alert(usertarget);

    });

        $('#newcrbtn').click(function(){

            var crname = $('#newcrtxt').val();

            $('#crlst').load('chatroom/crlst.php',{newcrname:crname},function(){

                $('#newcrtxt').val('');

            });

        });

    $(function() {

        $('#file_upload').uploadify({

            'formData' : {

                'timestamp' : '<?php echo $timestamp;?>',

                'token' : '<?php echo md5('unique_salt' . $timestamp);?>'

            },

            'swf' : 'others/plugins/uploadify/uploadify.swf',

            'uploader' : 'others/plugins/uploadify/uploadify.php?myuid=<?php echo $_SESSION['uid'];

?>&room=<?php echo $crld; ?>',

                'myuid' : '<?php echo $_SESSION['uid']; ?>'

        });

    });

```

```

$('#chatting').load('chatroom/chatting.php?room=<?php echo $crid; ?>');
$('#olusers').load('chatroom/olusers.php');
$('#btnpost').click(function(){
    kirim();
});
setInterval(function(){
    $('#chatting').load('chatroom/chatting.php?room=<?php echo $crid; ?>');
    $('#olusers').load('chatroom/olusers.php');
    $('#crlst').load('chatroom/crlst.php');
},3000);
});
</script>
<div id="middle">
    <div id="chatting" class="wrapper"></div>
    <table id="details">
        <tr><td height="52%" style="vertical-align:top">
            <div id="olusers" class="wrapper"></div>
        </td></tr>
        <tr><td height="48%" style="vertical-align:bottom">
            <div id="chatrooms" class="wrapper">
                <div id="crcontrol">
                    <input type="text" id="newcrtxt" class="cpost"><input type="button"
id="newcrbtn" value="add" class="cpost">
                </div>
                <div id="crlst"></div>
            </div>
        </td></tr>
    </table>

```

```
</div>
<div id="bottom" class="wrapper">
  <input type="text" id="textbox" class="cpost" size="100%" onkeydown="kirimOnreturn(event)">
  <input type="button" id="btnpost" class="cpost" value="shout">
  <input type="button" id="btnsendfile" class="cpost" value="send file">
</div>
<div id="popup-wrapper" style="background-color:white">
  <br>
  <form>
    <div id="queue"></div>
    <input id="file_upload" name="file_upload" type="file" multiple="true">
  </form>
  <!-- To : <?php showComboboxUserlist(); ?> -->
  <input type="button" id="btnclose" class="cpost" value="close">
</div>
```

### Chatroom/chatting.php

```
<?php
session_start();
include_once '../modules/dbcon.php';
include_once '../modules/functions.php';

if (isset($_POST['textcht']))
{
  $uid = $_SESSION['uid'];
  $msg = $_POST['textcht'];
  $scrid= $_GET['room'];
```

```
    sendNewpost($uid,$msg,$crid);
}
showConv(20,$_GET['room']);
?>
```

#### Chatroom/crlist.php

```
<?php
session_start();
include_once '../modules/dbcon.php';
include_once '../modules/functions.php';

if (isset($_POST['newcrname']))
{
    $uid = $_SESSION['uid'];
    $crname = $_POST['newcrname'];
    addnewroom($crname,$uid);
}
showChatroomlist();
?>
```

#### Chatroom/ENGINE.php

```
<?php
//ini chatroom engine
session_start();

//include_once dbconn, fungsi2x, tar dipanggil disini
include_once '../modules/dbcon.php';
```

```
include_once '../modules/functions.php';
```

```
if(isset($_POST['menu']))
```

```
{
```

```
    $menu = $_POST['menu'];
```

```
    if($menu == 'chatroom')
```

```
    {
```

```
        include_once 'chatroom.php';
```

```
    }
```

```
    elseif ($menu == 'settings')
```

```
    {
```

```
        if($_POST['deluser'] != "")
```

```
        {
```

```
            $deluid = $_POST['deluser'];
```

```
            killUser($deluid);
```

```
        }
```

```
        include_once 'settings.php';
```

```
    }
```

```
    elseif ($menu == 'logout')
```

```
    {
```

```
        include_once 'logout.php';
```

```
    }
```

```
    else
```

```
    {
```

```
        include_once 'chatroom.php';
```

```
    }
```

```
}
```

```
//-----
```

```
if($_POST['newuser']!=') and ($_POST['newnick']!=') and ($_POST['newpass']!=') and  
$_POST['oldpass'])
```

```
{
```

```
    $newuser = $_POST['newuser'];
```

```
    $newnick = $_POST['newnick'];
```

```
    $newpass = $_POST['newpass'];
```

```
    $oldpass = $_POST['oldpass'];
```

```
    $minchr = 5;
```

```
    $maxchr = 15;
```

```
    if($newuser == 'username' or $newnick == 'nickname' or $newpass == 'new password' or $newpass2  
    == 'old password')
```

```
    {
```

```
        shownotif('isikan semua data');
```

```
    }
```

```
    elseif (strlen($newnick)<$minchr ){
```

```
        shownotif('jumlah huruf nickname minimal '.$minchr);
```

```
    }
```

```
    elseif(strlen($newnick)> $maxchr){
```

```
        shownotif('jumlah huruf nickname maximal '.$maxchr);
```

```
    }
```

```
    else {
```

```
        if(is_valid_user($_SESSION['username'],$oldpass)=== true)
```

```
        {
```

```
            $newuser = cegahXss($newuser);
```

```
$newnick = cegahXss($newnick);
$newpass = cegahXss($newpass);
$uid = $_SESSION['uid'];
updateUser($uid, $newuser, $newnick, $newpass);
$_SESSION['username'] = $newuser;
shownotif("ok");
}
else
{
    shownotif("password lama salah");
}
}
}
?>
```

#### Chatroom/logout.php

```
<?php
session_start();
//include_once dbconn, fungsi2x, tar dipanggil disini
include_once '../modules/dbcon.php';
include_once '../modules/functions.php';
$uid = $_SESSION['uid'];
iamoffline($uid);
session_destroy();
redirTo('index.php');
?>
```

### Chatroom/olusers.php

```
<?php
session_start();

include_once '../modules/dbcon.php';
include_once '../modules/functions.php';

showOlusers();

?>
```

### Chatroom/setting.php

```
<script type="text/javascript">
$(document).ready(function(){
    //----- js register script begins here -----
    var utxt = 'username';
    var utxtnick = 'nickname';
    var ptxt = 'new password';
    var oldpass = 'old password';
    $('#textbusername').val(utxt).focus(function(){
        if($(this).val() == utxt){
            $(this).val("");
        }
    }).blur(function(){
        if($(this).val() == ''){
            $(this).val(utxt);
        }
    });
    $('#textbnickname').val(utxtnick).focus(function(){
```



```
if($(this).val() == utxtnick){
    $(this).val("");
}
}).blur(function(){
    if($(this).val() == ""){
        $(this).val(utxtnick);
    }
});

$('#textbpassword').val(ptxt).focus(function(){
    if($(this).val() == ptxt){
        $(this).val("");
        this.type = 'password';
    }
}).blur(function(){
    if($(this).val() == ""){
        $(this).val(ptxt);
        this.type = 'text';
    }
});

$('#textboldpass').val(oldpass).focus(function(){
    if($(this).val() == oldpass){
        $(this).val("");
        this.type = 'password';
    }
}).blur(function(){
    if($(this).val() == ""){
        $(this).val(oldpass);
        this.type = 'text';
    }
});
```

```
    }
  });
  //-----
  $('#updatebtn').click(function(){
    var un = $('#textbusername').val();
    var nc = $('#textbnickname').val();
    var pw = $('#textbpassword').val();
    var pw2 = $('#textboldpass').val();
    if (un != "" && pw != "" && nc != "" && pw2 != "")
    {
      $('#notif').load('chatroom/ENGINE.php',{newuser:un,newnick:nc,newpass:pw,oldpass:pw2});
    }
  });
  $('#deluser').click(function(){
    var uid = $(this).attr('href').replace('#delete-', '');
    $('#crcontent').empty();
    $('#crcontent').load('chatroom/ENGINE.php',{menu:'settings',deluser:uid});
  });

  //----- end of js register script -----
});
</script>
```

```
<style>
```

```
#stgs {
```

```
-moz-box-shadow: 0 0 5px 5px #aaa;
```

```
-webkit-box-shadow: 0 0 5px 5px #aaa;
```

```
box-shadow: 0 0 5px 5px #aaa;
```

```

-moz-border-radius: 5px;
background-color: #fff;
}
</style>
<div align="center" style="margin: 20px;" id="stgs">
<table>
<!-- <input type="button" value="login" name="login" id="loginbtn" class="clogin">-->
<tr>
<td colspan="3" align="center"></td>
</tr>
<tr>
<td><input size="40" name="uname" value="username" id="textbusername"
class="csettings"></td>
</tr>
<tr>
<td><input size="40" name="nname" value="nickname" id="textbnickname"
class="csettings"></td>
</tr>
<tr>
<td><input size="40" name="passwrđ" value="new password" id="textbpassword"
class="csettings"></td>
</tr>
<tr>
<td><input size="40" name="rpasswrđ" value="old password" id="textboldpass"
class="csettings"></td>
</tr>
<tr>
<td><input type="button" value="update" name="register" id="updatebtn"
class="csettings" style="float: right;"></td>
</tr>

```

```
</table>
<?php
session_start();
//include_once dbconn, fungsi2x, tar dipanggil disini
include_once '../modules/dbcon.php';
include_once '../modules/functions.php';
$uid = $_SESSION['uid'];
if (is_admin($uid))
{
    showUserlist();
}
?>
</div>
```

### Login/about.php

```
<div align="center">
    <table align="center">
        <tr>
            <td>
                <p><br><br>
                <b>Petunjuk mengenai penggunaan aplikasi :</b> <br><br>
                    <b>1.</b> File yang dapat dikirim hanya file-file berekstensi
                    <u>'jpg','jpeg','gif','png','bmp','doc','docx','xls','xlsx','ppt','pptx','rar','exe'</u><br><br>
                    <b>2.</b> Chatroom yang ditampilkan hanya 10. Chatroom yang lama
                    akan tergusur chatroom yang baru, tapi tidak hilang. Jadi pengguna harus mencatat <i>link/url</i>
                    chatroom tersebut jika ingin dibuka sewaktu-waktu.<br><br>
                    <b>3.</b> Proses Logout dilakukan 2 kali.
                </p>
```

```
        </td>
    </tr>
</table>
</div>
```

### Login/ENGINE.php

```
<?php
//the engine of login
session_start();
//include_once dbconn, fungsi2x, tar dipanggil disini
include_once '../modules/dbcon.php';
include_once '../modules/functions.php';
if(isset($_POST['menu']))
{
    $menu = $_POST['menu'];
    if($menu == 'home')
    {
        include_once 'home.php';
    }
    elseif ($menu == 'login')
    {
        include_once 'login.php';
    }
    elseif ($menu == 'register')
    {
        include_once 'register.php';
    }
}
```

```

elseif ($menu == 'about')
{
    include_once 'about.php';
}
else
{
    include_once 'home.php';
}
}
//----- login -----
if (isset ($_POST['uname']) and isset ($_POST['passwd']))
{
    $u = $_POST['uname'];
    $p = $_POST['passwd'];
    if (is_valid_user($u, $p) === true) //validasi user disini, pake is_valid_user()
    {
        //session buat uid
        $uid = get_userid($u);
        $_SESSION['uid'] = $uid;
        $_SESSION['username'] = $u;
        iamonline($uid);
        //session buat username
        redirectTo('chatroom.php'); //if user is valid, redirect to chatroom
    }
    else //user not valid
    {
        $wrongpwd = "invalid username or password";
        shownotif($wrongpwd);
    }
}

```

```
}
}
//----- register -----
if(($_POST['newuser']!= '') and ($_POST['newnick']!= '') and ($_POST['newpass']!= '') and
$_POST['newpass2']!= ''){
    $newuser = $_POST['newuser'];
    $newnick = $_POST['newnick'];
    $newpass = $_POST['newpass'];
    $newpass2 = $_POST['newpass2'];

    $minchr = 5;
    $maxchr = 15;

    if($newuser == 'username' or $newnick == 'nickname' or $newpass == 'password' or $newpass2 ==
'retype-password')
    {
        shownotif('isikan semua data');
    }
    elseif (strlen($newnick)< $minchr ){
        shownotif('jumlah huruf nickname minimal '.$minchr);
    }
    elseif(strlen($newnick)> $maxchr){
        shownotif('jumlah huruf nickname maksimal '.$maxchr);
    }
    elseif ($newpass != $newpass2)
    {
        shownotif('password tidak sama');
    }
}
```

```
else {  
    $newuser = cegahXss($newuser);  
    $newnick = cegahXss($newnick);  
    $newpass = cegahXss($newpass);  
    addNewuser($newuser, $newnick, $newpass);  
    $uid = get_userid($newuser);  
    $_SESSION['uid'] = $uid;  
    $_SESSION['username'] = $newuser;  
    iamonline($uid);  
    //session buat username  
    redirTo('chatroom.php'); //if user is valid, redirect to chatroom  
  
}  
}  
//----- end of register -----  
?>
```

### Login/home.php

```
<div align="center">  
    <table align="center" bgcolor="green">  
        <tr>  
            <td>  
                <p><br/>  
                <br/>  
                Selamat datang</p>  
                <p><br/>  
                <br/>
```



```
<br/>
<br/>
<br>
<br>
</p>
</td>
</tr>
</table>
</div>
<?php
?>
```

### Login/login.php

```
<script type="text/javascript">
//----- fungsi lokal -----
function kirim()
{
    var un = $('#tbusername').val();
    var pw = $('#tbpassword').val();
    if (un != "" && pw != "")
    {
        $('#notif').load('login/ENGINE.php',{uname:un,passwd:pw});
    }
}

function kirimOnreturn(evt)
{
```

```
if (evt.keyCode == 13 )
{
    kirim();
}
}
//-----
```

```
$(document).ready(function(){
    //----- js login script begins here -----
    var utxt = 'username';
    var ptxt = 'password';
    $('#tbusername').val(utxt).focus(function(){
        if($(this).val() == utxt){
            $(this).val("");
        }
    }).blur(function(){
        if($(this).val() == ''){
            $(this).val(utxt);
        }
    });
    $('#tbpassword').val(ptxt).focus(function(){
        if($(this).val() == ptxt){
            $(this).val("");
            this.type = 'password';
        }
    }).blur(function(){
        if($(this).val() == ''){
            $(this).val(ptxt);
        }
    });
});
```

```

        this.type = 'text';
    }
});

$('#loginbtn').click(function(){
    kirim();
});
//----- end of js login script -----
});
</script>

<div align="center" style="margin: 20px;">
    <table>
        <tr>
            <td><input size="40" name="username" value="username" id="tbusername" class="clogin"
onkeydown="kirimOnreturn(event)"></td>
        </tr>
        <tr>
            <td><input size="40" name="password" value="password" id="tbpassword" class="clogin"
onkeydown="kirimOnreturn(event)"></td>
        </tr>
        <tr>
            <td>
                <input type="button" value="login" name="login" id="loginbtn" class="clogin">
            </td>
        </tr>
    </table>
</div>

```

```
<?php
?>
```

### Login/register.php

```
<script type="text/javascript">
function kirim()
{
    var un = $('#textbusername').val();
    var nc = $('#textbnickname').val();
    var pw = $('#textbpassword').val();
    var pw2 = $('#textbrepass').val();
    if (un != "" && pw != "" && nc != "" && pw2 != "")
    {
        $('#notif').load('login/ENGINE.php',{newuser:un,newnick:nc,newpass:pw,newpass2:pw2});
    }
}

$(document).ready(function(){
    //----- js register script begins here -----
    var utxt = 'username login';
    var utxtnick = 'nickname chat';
    var ptxt = 'password';
    var ptxtrepass = 'retype-password';
    $('#textbusername').val(utxt).focus(function(){
        if($(this).val() == utxt){
            $(this).val("");
        }
    })
})
```

```
}).blur(function(){
    if($(this).val() == ""){
        $(this).val(utxt);
    }
});
$('#textbnickname').val(utxtnick).focus(function(){
    if($(this).val() == utxtnick){
        $(this).val("");
    }
}).blur(function(){
    if($(this).val() == ""){
        $(this).val(utxtnick);
    }
});
$('#textbpassword').val(ptxt).focus(function(){
    if($(this).val() == ptxt){
        $(this).val("");
        this.type = 'password';
    }
}).blur(function(){
    if($(this).val() == ""){
        $(this).val(ptxt);
        this.type = 'text';
    }
});
$('#textbrepass').val(ptxtrepass).focus(function(){
    if($(this).val() == ptxtrepass){
        $(this).val("");
    }
});
```

```

        this.type = 'password';
    }
}).blur(function(){
    if($(this).val() == ''){
        $(this).val(ptxtrepass);
        this.type = 'text';
    }
});
//-----
$('#registerbtn').click(function(){
    kirim();
});

//----- end of js register script -----
});
</script>
<div align="center" style="margin: 20px;">
<table>
<!--      <input type="button" value="login" name="login" id="loginbtn" class="clogin">-->
    <tr>
        <td colspan="3" align="center"></td>
    </tr>
    <tr>
        <td><input size="40" name="uname" value="username login" id="textusername"
class="cregister"></td>
    </tr>
    <tr>

```

```
        <td><input size="40" name="nname" value="nickname chat" id="textbnickname"
class="cregister"></td>
    </tr>
    <tr>
        <td><input size="40" name="passwd" value="password" id="textbpassword"
class="cregister"></td>
    </tr>
    <tr>
        <td><input size="40" name="rpasswd" value="retype-pass" id="textbrepass"
class="cregister"></td>
    </tr>
    <tr>
        <td><input type="button" value="register" name="register" id="registerbtn"
class="cregister"></td>
    </tr>
</table>
</div>
<?php
?>
```

Modules/dbcon.php

```
<?php
/*
$dbHost = 'sql207.byethost33.com';
$dbUser = 'b33_11479050';
$dbPass = '1q2w3e';
$dbName = 'b33_11479050_cruttcud';
*/
```

```
$dbHost = 'localhost';
$dbUser = 'root';
$dbPass = "";
$dbName = 'cruttcrud';
mysql_connect($dbHost, $dbUser, $dbPass);
mysql_select_db($dbName);
?>
```

### Modules/function.php

```
<?php
error_reporting(0);
session_start();
include_once 'dbcon.php';
function updateUser($uid,$newuser,$newnick,$newpass)
{
    mysql_query("UPDATE user SET uname = '".addslashes($newuser)."', nickname =
    '".addslashes($newnick)."', password = '".addslashes($newpass)."' WHERE uid = ".$uid." LIMIT 1 ;") or
    die(mysql_error());
}
function addNewuser($newuser,$newnick,$newpass)
{
    mysql_query("INSERT INTO user (uname,nickname,password,isadmin,olstatus) VALUES
    ('".addslashes($newuser)."', '".addslashes($newnick)."', '".addslashes($newpass)."', '0', '1');") or
    die(mysql_error());
}
function is_valid_user($user,$pass) //tested
{
    $q="select * from user where uname='".addslashes($user)."'";
```



```
$result=mysql_query($q);
if ($isi=mysql_fetch_array($result))
{
    $validpass=$isi['password'];
    if($pass==$validpass)
    {
        return true;
    }
    else
    {
        return false;
    }
}
else
{
    return false;
}
}
function get_userid($username) //tested
{
    $q="select * from user where uname='".addslashes($username)."'";
    $result=mysql_query($q);
    if ($isi=mysql_fetch_array($result))
    {
        return $isi['uid'];
    }
}
function is_logged_in() //tested
```

```
{
  if ($_SESSION['username'] != "" and $_SESSION['uid'] != "")
  {
    return true;
  }
  else
  {
    return false;
  }
}
function getsesUsername()
{
  return $_SESSION['username'];
}
function is_admin($uid)
{
  $uid = (int) $uid;
  $xq = mysql_query("SELECT isadmin FROM user WHERE uid='".$uid.'" LIMIT 1");
  if ($isi = mysql_fetch_array($xq))
  {
    $isadmin = $isi['isadmin'];
    if ($isadmin == 1)
    {
      return true;
    }
  }
  else
  {
    return false;
  }
}
```

```

    }
}
}
function getNickname($uid)
{
    $uid = (int) $uid;
    $xq = mysql_query("SELECT nickname FROM user WHERE uid='".$uid.'" LIMIT 1");
    if ($isi = mysql_fetch_array($xq))
    {
        $nick = $isi['nickname'];
        return $nick;
    }
}
function sendNewpost($uid,$msg,$scrid=0)
{
    $uid = (int) $uid;
    $msgx = cegahXss($msg);
    mysql_query("INSERT INTO chatting (chatterid,roomid,postedtext,tanggal) VALUES ('".$uid."',
    '".$scrid."', '".$msgx."', NOW())" or die(mysql_error()));
}
function showConv($n,$scrid=0)
{
    $xq = mysql_query("SELECT * from (SELECT * FROM chatting WHERE roomid='$scrid' order by
    tanggal desc limit 0,$n) as c join user on c.chatterid = user.uid order by c.tanggal asc");
    while($isi = mysql_fetch_array($xq))
    {
        if ($isi['uid']==0)
        {

```

```

        echo "<div class='info'><strong>".$isi['postedtext']."</strong></div>";
    }
    else
    {
        echo "<strong>".htmlentities(trim($isi['nickname']),ENT_QUOTES)."</strong> :
.htmlentities(trim($isi['postedtext']),ENT_QUOTES)."<br>";
    }
}
}
function showOlusers($n = null)
{
    $xq = mysql_query("select nickname from user where olstatus='1'");
    while ($isi = mysql_fetch_array($xq))
    {
        echo htmlentities($isi['nickname'],ENT_QUOTES).<br>;
    }
}
function showUserlist()
{
    $uid = $_SESSION['uid'];
    echo '<table border=1 style="margin:20px;" id="userlisttbl">
    <tr>
    <th>username</th><th>nickname</th><th>delete?</th></tr>';
    $xq = mysql_query("select * from user where uid <> '". $uid. "' order by uname asc");
    while($isi = mysql_fetch_array($xq))
    {
        echo '<tr><td style="padding-left:10px;padding-right:10px;">'. $isi['uname']. '</td><td
style="padding-left:10px;padding-right:10px;">'. $isi['nickname']. '</td><td style="padding-
left:10px;padding-right:10px;"><a href="#delete-' . $isi['uid']. '" class="deluser">delete</a></td></tr>';
    }
}

```

```

    }
    echo "</table>";
}
function showComboboxUserlist()
{
    $uid = $_SESSION['uid'];
    echo '<select name="usersendto" id="usersendto">
        <option value="0">ALL USER</option>';
    $xq = mysql_query("select * from user where uid <> '". $uid. "' order by uname asc");
    while($isi = mysql_fetch_array($xq))
    {
        echo '<option value="'. $isi['uid']. "'>'. $isi['nickname']. ' </option>';
    }
    echo "</select>";
}

function iamonline($uid)
{
    $uid = (int) $uid;
    mysql_query("UPDATE user SET olstatus = '1' WHERE uid='". $uid. "' LIMIT 1 ;") or die(mysql_error());
}

function iamoffline($uid)
{
    $uid = (int) $uid;
    mysql_query("UPDATE user SET olstatus = '0' WHERE uid='". $uid. "' LIMIT 1 ;") or die(mysql_error());
}

function killUser($uid)
{

```

```

    $uid = (int) $uid;
    mysql_query("delete from user where uid=".$uid." limit 1") or die(mysql_error());
}
//----- fungsi-fungsi pembantu -----
function cegahXss($str)
{
    // $str = trim($str); //hilangkan space yang b'lebih
    // $str = utf8_decode($str); //cegah bila terjadi masalah unicode codec
    // $str = htmlentities($str, ENT_NOQUOTES); //ganti karakter tertentu ke bentuk html entiti
    // $str = str_replace("#", "&#35;", $str);
    // $str = str_replace("%", "&#37;", $str);
    // return $str;
    return addslashes($str);
}
function redirTo($location)
{
    echo '<script>>window.location.replace("'.$location.'")</script>';
}

function msgbox($msg)
{
    echo '<script>alert("'.$msg.'")</script>';
}

function shownotif($notif)
{
    echo '<script>
        $(document).ready(function(){
            $("#notif").slideDown("slow").text("'.$notif.'").delay(2000).slideUp("slow");
        });
    </script>';
}

```

```

    });
</script>
;
}
function uploadFile($postedfile,$from,$to)
{
    if (!empty($postedfile) and $postedfile['size']>0)
    {
        $tmpfile = $postedfile['tmp_name'];
        if (!preg_match('/.php/i', $postedfile['name']))
        {
            $saveto = '/uploads/';
            $namafile = $postedfile['name'];

            if (substr($namafile,-1,1) == "/")
            {
                $namafile = substr($namafile,0,strlen($namafile)-1);
            }

            $destination = str_replace('//','/',$_SERVER['DOCUMENT_ROOT'].$saveto.$namafile);
            if((is_uploaded_file($tmpfile)) and (!file_exists($destination)))
            {
                move_uploaded_file($tmpfile,$destination);

                mysql_query("INSERT INTO uploads (nomor,filename,path,from,to) VALUES
(NULL,'$namafile','"$_SERVER['HTTP_HOST']."\\uploads\\$namafile','$from','$to')") or
shownotif(mysql_error());

                shownotif("upload berhasil");
            }
        }
    }
}

```

```

else
{
    shownotif("upload gagal");
}
}
else
{
    shownotif("upload gagal");
}
}
}
//----- 15.11.2012
function addnewroom($roomname,$creatorid)
{
    $uid = (int) $creatorid;
    $roomname = cegahXss($roomname);
    mysql_query("INSERT INTO chatroom (crid, creatorid, crname) VALUES (NULL, '$creatorid', '$roomname')") or die(mysql_error());
}

function showChatroomlist($n=10)
{
    $xq = mysql_query("SELECT * FROM chatroom where crid='0'");
    while($isi = mysql_fetch_array($xq))
    {
        echo "<a href='chatroom.php?room=$isi[crid]' target='_blank'>$isi[crname]</a><br>";
    }
}

```



```

    $xq = mysql_query("SELECT * FROM chatroom where crid <> '0' order by createdat desc limit
0,$n");
    while($isi = mysql_fetch_array($xq))
    {
        echo "<a href='chatroom.php?room=$isi[crid]' target='_blank'>$isi[crname]</a><br>";
    }
}
function getChatroomname($crid=0)
{
    $xq = mysql_query("SELECT * FROM chatroom where crid='$crid'") or
die(mysql_error());
    $isi = mysql_fetch_array($xq);
    return $isi['crname'];
}??>

```

others\css\chatroom.css

```

root {
    display: block;
}
body {
    vertical-align: middle;
    background-image: url('../pics/bg.jpg');
}
#mainwrapper {
    border: 0px solid #a4a4a4;
    margin: 15px;
    -moz-box-shadow:10gpx 2px 5px #ecec;
    -moz-border-radius:5px 5px 5px 5px;
    padding: 10px;
}

```

```
width: 90%;
height: 90%;
}
.wrapper {
  -moz-box-shadow: 0 0 5px 5px #aaa;
  -webkit-box-shadow: 0 0 5px 5px #aaa;
  box-shadow: 0 0 5px 5px #aaa;
  -moz-border-radius: 5px;
  background-color: #fff;
}
#top {
  height: 70px;
}
#menubox {
  margin-right: 14px;
  margin-top: 30px;
  float: right;
}
#notif {
  top: 0px;
  position: absolute;
  height: 30px;
  z-index: 2;
  border: 1px solid #989898;
  width: 95%;
  margin-left: 20px;
  margin-right: 20px;
  /*-----*/
```

```
}  
.crmenu {  
    text-decoration: none;  
    background-color: #B7B7B7;  
    border: 1px solid #EEEEEE;  
    border-radius: 10px 10px 10px 10px;  
    box-shadow: 0 0 5px #BCBCBC;  
    color: #FFFFFF;  
    font-family: 'Arial Black',sans-serif;  
    font-size: 30px;  
    line-height: 0px;  
    outline: none;  
    margin: 8px 8px 15px;  
    text-shadow: 0 0 0 #CFCFCF, 1px -1px 0 #B3B3B3, 2px -2px 0 #989898, 3px -3px 0 #7C7C7C, 4px -4px 0  
#616161, 5px -5px 0 #464646, 6px -6px 5px rgba(0, 0, 0, 0.8), 6px -6px 1px rgba(0, 0, 0, 0.5), 0 0 5px  
rgba(0, 0, 0, 0.2);  
}  
#judull {  
    font-size: 40px;  
    float: left;  
    margin-right: 100px;  
    text-align: left;  
}  
#crcontent {  
}  
#middle {  
    margin-top: 25px;  
    height: 60%;
```

```
        margin-bottom: 25px;
    }
    #chatting {
        float: left;
        margin-right: 10px;
        width: 75%;
        height: 100%;
        text-align: left;
        overflow: auto;
        padding: 3px;
    }
    #details {
        float: right;
        width: 21%;
        margin-left: 10px;
        overflow: auto;
        padding-top: 5px;
        padding-left: 10px;
        padding-right: 10px;
        padding-bottom: 0px;
        text-align: left;
        height: 100%;
    }
    #olusers {
        height: 80%;
        margin-bottom: 1%;
        overflow: auto;
        padding: 10px;
```

```
        text-align: left;
    }
    #chatrooms {
        height: 90%;
        padding: 10px;
        text-align: left;
    }

        #clist {
            overflow: auto;
        }
#bottom {
    clear: both;
    padding: 5px;
    margin-bottom: 25px;
}
.cpost {
    -moz-border-radius: 5px;
    -webkit-border-radius: 5px;
    border-radius: 5px;
    padding: 5px;
    margin: 3px;
}
#textbox {
}
#btnpost {
    padding: 5px;
}
.csettings{
```

```
margin: 5px;
padding: 5px;
-moz-border-radius:5px;
-webkit-border-radius:5px;
}
#popup-wrapper {
padding: 10px;
}
.info {
color: gray;
padding: 5px;
}
```

#### others\css\login.css

```
root {
display: block;
}
body {
vertical-align: middle;
background-image: url('../pics/bg.jpg');
background-repeat: repeat;
}
#notif {
top: 0px;
position: absolute;
height: 30px;
z-index: 2;
```

```
border: 1px solid #989898;
width: 95%;
margin-left: 20px;
margin-right: 20px;
/*-----*/
}

#mainwrapper {
background-color: #FFFFFF;
border: 1px solid #A9A9A9;
border-radius: 15px 15px 15px 15px;
box-shadow: 0 0 15px grey;
font-family: Arial,Helvetica,Sans-Serif;
font-size: 12px;
font-weight: bold;
margin-bottom: 5%;
margin-top: 5%;
padding: 10px;
width: 80%;
}

#top {
border: 0 solid black;
border-radius: 10px 10px 10px 10px;
/* border-top-left-radius: 5px;
border-top-right-radius: 5px;*/
margin-top: -30px;
margin-bottom: -20px;
padding: 30px;
padding-top: 50px;
```

```
width: 70%;
}
#contentmenu {
border: 1px solid #A9A9A9;
border-radius: 15px 15px 15px 15px;
}
.wrapper {
margin: 5px;
}

.loginmenu {
text-decoration: none;
background-color: #B7B7B7;
border: 1px solid #EEEEEE;
border-radius: 10px 10px 10px 10px;
box-shadow: 0 0 5px #BCBCBC;
color: #FFFFFF;
font-family: 'Arial Black',sans-serif;
font-size: 30px;
line-height: 0px;
outline: none;
margin: 8px 8px 15px;

text-shadow: 0 0 0 #CFCFCF, 1px -1px 0 #B3B3B3, 2px -2px 0 #989898, 3px -3px 0 #7C7C7C, 4px -4px 0
#616161, 5px -5px 0 #464646, 6px -6px 5px rgba(0, 0, 0, 0.8), 6px -6px 1px rgba(0, 0, 0, 0.5), 0 0 5px
rgba(0, 0, 0, 0.2);
}

/*===== home =====*/
```



```
/*===== end of home =====*/
```

```
/*===== login =====*/
```

```
.clogin {
```

```
margin: 10px;
```

```
padding: 10px;
```

```
-moz-border-radius:5px;
```

```
-webkit-border-radius:5px;
```

```
}
```

```
#loginbtn {
```

```
float: right;
```

```
margin-bottom: 0px;
```

```
/* background-color: #CFCFCF;*/
```

```
padding: 10px;
```

```
font-size: 15px;
```

```
}
```

```
/*===== end of login =====*/
```

```
/*===== register =====*/
```

```
.cregister{
```

```
margin: 5px;
```

```
padding: 5px;
```

```
-moz-border-radius:5px;
```

```
-webkit-border-radius:5px;
```

```
}  
#registerbtn{  
    float: right;  
    margin-bottom: 0px;  
/* background-color: #CFCFCF;*/  
    padding: 10px;  
    font-size: 15px;  
}  
/*===== end of register =====*/  
/*===== about =====*/  
/*===== end of about =====*/
```

#### others\css\popup.css

```
.modalPopLite-mask  
{  
    position:fixed;  
    z-index:9000;  
    background-color:#000;  
    display:none;  
    top:0px;  
    left:0px;  
    width:100%;  
}  
.modalPopLite-wrapper  
{  
    position:fixed;  
    z-index:9999;
```

```
/*display:none;*/  
/*left:-10000px;*/  
-webkit-border-radius: .5em;  
    -moz-border-radius: .5em;  
    border-radius: .5em;  
-webkit-box-shadow: 0 0px 25px rgba(0,0,0,.9);  
-moz-box-shadow: 0 0px 25px rgba(0,0,0,.9);  
    box-shadow: 5px 5px 10px rgba(0, 0, 0, 0.4);  
border: 5px solid rgb(000, 0, 0);  
border: 5px solid rgba(000, 0, 0, .5);  
  
}
```

```
/* popup */  
.popBox-holder  
{  
    display:none;  
    position: absolute;  
    left: 0px;  
    top: 0px;  
    width:100%;  
    height:100%;  
    text-align:center;  
    z-index: 999;  
    background-color:#000;  
    filter:alpha(opacity=40);  
    opacity:0.5;
```

```
}
```

```
.popBox-container
```

```
{
```

```
display:none;
```

```
background-color: #fff;
```

```
border:4px solid #000;
```

```
padding:10px;
```

```
text-align:center;
```

```
z-index: 1000;
```

```
    -webkit-border-radius: .5em;
```

```
    -moz-border-radius: .5em;
```

```
    border-radius: .5em;
```

```
    -webkit-box-shadow: 0 1px 2px rgba(0,0,0,.2);
```

```
    -moz-box-shadow: 0 1px 2px rgba(0,0,0,.2);
```

```
    box-shadow: 0 1px 2px rgba(0,0,0,.2);
```

```
}
```

```
.popBox-container .done-button
```

```
{
```

```
margin-top:10px;
```

```
}
```

```
.popBox-container .button {
```

```
display: inline-block;
```

```
zoom: 1; /* zoom and *display = ie7 hack for display:inline-block */
```

```
display: inline;
```

```
vertical-align: baseline;
```

```
margin: 0 2px;
outline: none;
cursor: pointer;
text-align: center;
text-decoration: none;
font: 14px/100% Arial, Helvetica, sans-serif;
padding: .5em 2em .55em;
text-shadow: 0 1px 1px rgba(0,0,0,.3);
-webkit-border-radius: .5em;
-moz-border-radius: .5em;
border-radius: .5em;
-webkit-box-shadow: 0 1px 2px rgba(0,0,0,.2);
-moz-box-shadow: 0 1px 2px rgba(0,0,0,.2);
box-shadow: 0 1px 2px rgba(0,0,0,.2);
}
.popBox-container .button:hover {
    text-decoration: none;
}
.popBox-container .button:active {
    position: relative;
    top: 1px;
}

.popBox-container .small {
    font-size: 11px;
    padding: .2em 1em .275em;
}

.popBox-container .blue {
```

```
    color: #d9eef7;
    border: solid 1px #0076a3;
    background: #0095cd;
    background: -webkit-gradient(linear, left top, left bottom, from(#00adee), to(#0078a5));
    background: -moz-linear-gradient(top, #00adee, #0078a5);
    filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#00adee',
endColorstr='#0078a5');
}
.popBox-container .blue:hover {
    background: #007ead;
    background: -webkit-gradient(linear, left top, left bottom, from(#0095cc), to(#00678e));
    background: -moz-linear-gradient(top, #0095cc, #00678e);
    filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#0095cc',
endColorstr='#00678e');
}
.popBox-container .blue:active {
    color: #80bed6;
    background: -webkit-gradient(linear, left top, left bottom, from(#0078a5), to(#00adee));
    background: -moz-linear-gradient(top, #0078a5, #00adee);
    filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#0078a5',
endColorstr='#00adee');
}

.popBox-ajax-progress
{
    position: fixed;
    left: 0px;
    top: 0px;
    width:100%;
```

```
height:100%;
text-align:center;
z-index: 99999;
background-color:#000;
filter:alpha(opacity=40);
opacity:0.5;
background-image: url('../pics/ajax-loader.gif');
background-repeat:no-repeat;
background-position:center center;
}
/* end popup */
```

#### others\css\style.css

```
root {
    display: block;
}
table {
    border-collapse: collapse;
}
.transbg{
    overflow:hidden;
    -webkit-border-radius:10px;
    -moz-border-radius:10px;
    border-radius:10px;
    background:none repeat scroll 0 0 rgba(221, 221, 221, 0.3);
    -moz-box-shadow:0px 0px 1px rgba(0, 0, 0, 0.3), 0px 1px 6px rgba(0, 0, 0, .3);
    -webkit-box-shadow:0px 0px 1px rgba(255, 255, 255, .15), 1px 1px 6px rgba(0, 0, 0, .3);
```

```
        box-shadow:0px 0px 1px rgba(255, 255, 255, .15), 1px 1px 6px rgba(0, 0, 0, .3);
    }
}
```

#### others\js\chatroom.js

```
$(document).ready(function(){
    $('#textbox').focus();
    $('#notif').hide(0);
    $('.crmenu').click(function(){
        var lnk = $(this).attr('href').replace('#', '');
        $('#crcontent').empty();
        $('#crcontent').load('chatroom/ENGINE.php',{menu:lnk});
    });
    $('#popup-wrapper').modalPopLite({ openButton: '#btnsendfile', closeButton: '#btnclose', isModal:
true });
});
```

PLUGIN JQuery:

others\js\functions.js

others\js\jquery.min.js

others\js\popup.js

#### others\js\login.js

```
$(document).ready(function(){
    //===== effects =====
```



```

$('#notif').hide(0);
$('#home').css('box-shadow','0 0 20px #5B5B5B');
$('#home').css('font-size','35');
$('.loginmenu').click(function(){
    $(this).css('cursor','wait');
    var lnk = $(this).attr('href').replace('#','');
    $('#contentmenu').slideUp('slow',function(){
        $(this).empty();
        var currid = '#' + lnk;
        $('.loginmenu').css('box-shadow','0 0 5px #BCBCBC');
        $('.loginmenu').css('font-size','30');
        $(currid).css('box-shadow','0 0 20px #5B5B5B');
        $(currid).css('font-size','35');
        $(this).load('login/ENGINE.php',{menu:lnk},function(){
            $(this).slideDown('slow',function(){
                $('.loginmenu').css('cursor','pointer');
            });
        });
    });
});
//===== register =====
});

```

PLUGIN Uploadify:

[others\plugins\uploadify\check-exists.php](#)

[others\plugins\uploadify\jquery.uploadify.js](#)

[others\plugins\uploadify\jquery.uploadify.min.js](#)

others\plugins\uploadify\uploadify.css

others\plugins\uploadify\uploadify.swf

others\plugins\uploadify\uploadify-cancel.png

others\plugins\uploadify\uploadify.php

```
<?php
```

```
ob_start();
```

```
session_start();
```

```
include_once '../../modules/functions.php';
```

```
// Define a destination
```

```
$targetFolder = '/cuttcrud/uploads'; // Relative to the root
```

```
$verifyToken = md5('unique_salt' . $_POST['timestamp']);
```

```
$me = getNickname($_GET['myuid']);
```

```
$room = (int) $_GET['room'];
```

```
if (!empty($_FILES) && $_POST['token'] == $verifyToken) {
```

```
    $tempFile = $_FILES['filedata']['tmp_name'];
```

```
    $targetPath = $_SERVER['DOCUMENT_ROOT'] . $targetFolder;
```

```
    $targetFile = rtrim($targetPath, '/') . '/' . $_FILES['filedata']['name'];
```

```
    // Validate the file type
```

```
    $fileTypes =
```

```
array('jpg', 'jpeg', 'gif', 'png', 'bmp', 'doc', 'docx', 'xls', 'xlsx', 'ppt', 'pptx', 'rar', 'exe', 'pdf', 'zip'); // File extensions
```

```
    $fileParts = pathinfo($_FILES['filedata']['name']);
```

```
    if (in_array($fileParts['extension'], $fileTypes)) {
```

```
        move_uploaded_file($tempFile, $targetFile);
```

```
        $namafile = $_FILES['filedata']['name'];
```

```
        sendNewpost(0, "".$me."\" baru mengunggah file (<a color='gray'  
href='uploads/$namafile'>$namafile</a>)", $room); //just modified 150113 9:30AM  
    } else {  
        sendNewpost(0, $me." gagal mengunggah file", $room); //just added 150113 9:30AM  
    }  
}  
ob_flush();  
?>
```