

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Unit Pelaksana Teknis Pusat Komputer Universitas Gadjah Mada khususnya bagian Jaringan, memberikan layanan *e-mail* dan *webhosting* dibawah domain *gamanet.ugm.ac.id*.

Khusus untuk pengguna dari layanan *webhosting* diatas selama ini menggunakan *telnet* dan *command line FTP (File Transfer Protocol) client* sebagai satu-satunya fasilitas yang ada. Sebagai fasilitas satu-satunya *telnet* dan *command line FTP client* tidak memberikan kemudahan bagi pengguna layanan diatas karena masih menggunakan perintah-perintah yang susah untuk diingat (perlu dihafalkan), padahal pengguna biasanya menginginkan penggunaan secara mudah.

Sehubungan dengan tidak nyamannya penggunaan fasilitas diatas maka diperlukan penambahan fasilitas yang lebih memberikan kemudahan bagi pengguna. Salah satu alternatif adalah fasilitas yang berbasis WWW (*World Wide Web*), sehingga pengguna dapat mengakses fasilitas tersebut tanpa berfikir harus menjalankan perintah yang rumit. Aplikasi *webhosting* akan membuat perintah-perintah yang diperlukan sesuai dengan keinginan pengguna.

1.2 Perumusan Masalah

Dari uraian diatas dapat dirumuskan beberapa masalah yang dihadapi oleh pengguna layanan *webhosting* di Unit Pelaksana Teknis Pusat Komputer Universitas Gadjah Mada seperti berikut:

1. Diperlukan suatu sistem baru yang dapat meningkatkan penggunaan layanan *webhosting*.
2. Diperlukan penyusunan sistem yang mengintegrasikan *telnet* dan *command line FTP*.
3. Diperlukan sistem yang berbasis web yang dapat dilihat dan digunakan dengan mudah dan cepat.

1.3 Batasan Masalah

Melihat luasnya cakupan yang dapat dibuat dalam layanan *webhosting* dan memperhatikan kebutuhan yang diperlukan oleh pengguna layanan *webhosting* pada domain *gamanet.ugm.ac.id*, maka tugas akhir ini hanya membahas pembuatan rancangan dan implementasi aplikasi *webhosting* untuk domain *gamanet.ugm.ac.id* yang mencakup pemrograman dengan bahasa PHP(*PHP Hypertext Preprocessor*), penggunaan *web server* Apache, basis data MySQL, FTP dan pengintegrasian hal-hal tersebut.

1.4 Maksud dan Tujuan

1. Untuk memberikan kemudahan kepada pengguna dalam menggunakan layanan *webhosting* untuk domain *gamanet.ugm.ac.id*.
2. Menghasilkan aplikasi *webhosting* yang berbasis *world wide web*.

1.5 Metode Penulisan

Metode penulisan yang dipakai dalam penyusunan tugas akhir ini adalah:

1. Studi literatur.

Studi literatur mengenai konsep *web server (Apache)*, *FTP (File Transfer Protocol)*, *multi-user* pada sistem operasi linux dan juga konsep bahasa *PHP (Php Hypertext Preprocessor)*.

2. Observasi.

Observasi dilakukan terhadap domain *gamanet.ugm.ac.id*. Observasi meliputi kebutuhan sistem, kebutuhan pengguna, dan kompatibilitas sistem.

1.6 Sistematika Penulisan Laporan Tugas Akhir

Untuk memberikan gambaran secara menyeluruh masalah yang akan dibahas dalam laporan tugas akhir ini, maka sistematika penulisan laporan ini dibagi dalam beberapa bab.

Bab I menjelaskan mengenai latar belakang masalah, perumusan masalah, maksud dan tujuan, batasan masalah, metode penulisan dan sistematika penulisan laporan tugas akhir.

Bab II menjelaskan tentang tinjauan secara umum terhadap Unit Pelaksana Teknis Pusat Komputer Universitas Gadjah Mada yang meliputi gambaran singkat mengenai Unit Pelaksana Teknis Pusat Komputer dan struktur organisasi.

Bab III akan dijelaskan mengenai konsep dasar dari *webserver*, *webbrowser*, situs, *FTP (File Transfer Protocol)*, *PHP (Php Hypertext Preprocessor)* sebagai compiler, *MySQL* sebagai server basis data, Normalisasi, diagram alir data (DAD). Juga akan dijelaskan mengenai diagram alir sistem (*flowchart*).

Bab IV akan dijelaskan tahap-tahap perancangan sistem. Dalam bab ini juga akan dijelaskan perancangan program aplikasi seperti desain tampilan antar muka, *flowchart* program serta penjelasannya, dan rancangan pengujian aplikasi.

Bab V dijelaskan tentang implementasi dari sistem ke dalam program aplikasi, seperti tampilan dialog antar muka serta proses input output program. Dan juga akan dijelaskan mengenai pengujian aplikasi.

Bab VI berisikan kesimpulan dari uraian dan pembahasan yang ada dalam bab-bab sebelumnya serta saran-saran untuk peningkatan dan pengembangan sistem lebih lanjut dimasa yang akan datang.

BAB II
UNIT PELAKSANA TEKNIS PUSAT KOMPUTER
UNIVERSITAS GADJAH MADA

2.1 Gambaran Singkat Unit Pelaksana Teknis Pusat Komputer Universitas Gadjah Mada

Unit Pelaksana Teknis Pusat Komputer Universitas Gadjah Mada terbentuk tanggal 18 Februari 1978. Unit ini bertanggung jawab kepada Rektor Universitas Gadjah Mada dan pembinaan sehari-hari dilakukan oleh Pembantu Rektor I.

Tujuan didirikannya Unit Pelaksana Teknis Pusat Komputer Universitas Gadjah Mada antara lain adalah:

- a. Memberikan pelayanan penggunaan komputer untuk mendukung pelaksanaan Tri Darma Perguruan Tinggi Universitas Gadjah Mada.
- b. Memberikan pelayanan penggunaan komputer untuk mendukung sistem administrasi akademik maupun administrasi umum.

2.2 Tugas Pokok dan Fungsi Unit Pelaksana Teknis Pusat Komputer Universitas Gadjah Mada.

Tugas pokok dan fungsi Unit Pelaksana Teknis Pusat Komputer diatur dalam Surat Keputusan Menteri Pendidikan dan Kebudayaan Republik Indonesia nomor: 0132/O/1992 tentang Organisasi dan Tata Kerja Universitas Gadjah Mada.

2.2.1 Tugas Unit Pelaksana Teknis Pusat Komputer

Unit Pelaksana Teknis Pusat Komputer mempunyai tugas mengumpulkan, mengolah, menyajikan, menyimpan data dan informasi serta memberikan layanan untuk program-program pendidikan, penelitian dan pengabdian kepada masyarakat.

2.2.2 Fungsi Unit Pelaksana Teknis Pusat Komputer

Untuk melaksanakan tugas pokok tersebut diatas maka Unit Pelaksana Teknis Pusat Komputer mempunyai fungsi-fungsi sebagai berikut:

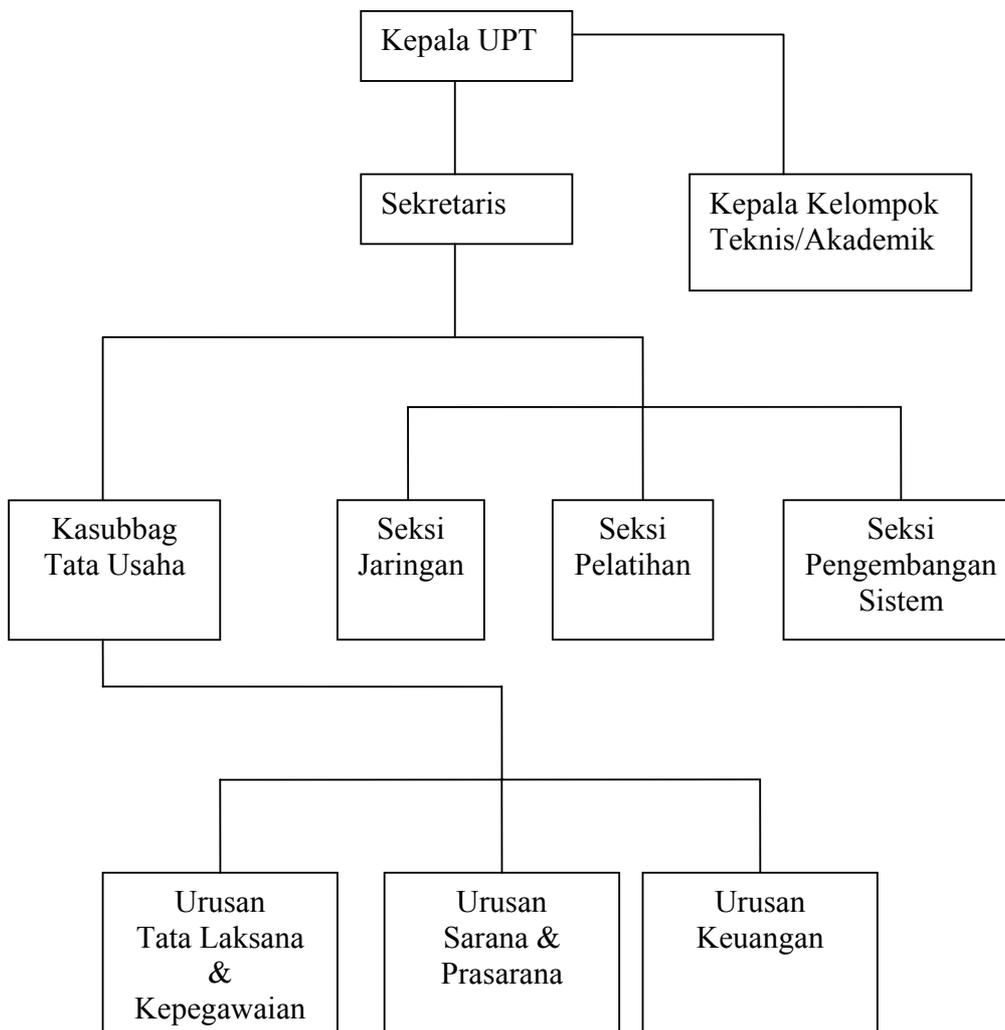
- a. Mengumpulkan, mengolah data dan informasi.
- b. Menyajikan, menyimpan data dan informasi.
- c. Melakukan urusan tata usaha Pusat Komputer.

2.3 Struktur Organisasi Unit Pelaksana Teknis Pusat Komputer Universitas Gadjah Mada 2002

Struktur Organisasi Unit Pelaksana Teknis Pusat Komputer Universitas Gadjah Mada mempunyai dasar yaitu Keputusan Kepala Unit Pelaksana Teknis Pusat Komputer Universitas Gadjah Mada No. 09/J01.KOM/UM.01.41/2001 dan ditetapkan tanggal 22 Januari 2001.

Struktur organisasi Unit Pelaksana Teknis Pusat Komputer ini terdiri atas Kepala Unit Pelaksana Teknis Pusat Komputer yang bertanggung jawab atas semua unit-unit pelaksana yang berada dibawahnya. Unit-unit pelaksana tersebut meliputi: Sekretaris, Kepala Kelompok Teknis/Akademik. Untuk sekretaris membawahi sub

unit Kepala Sub Bagian Tata Usaha, Seksi Jaringan, Seksi Pelatihan, Seksi Pengembangan Sistem, Koordinator Ketua Program. Dibawah Kepala Sub Bagian Tata Usaha terdapat lagi sub-sub unit yaitu: Urusan Tata Laksana dan Kepegawaian, Urusan Sarana dan Prasarana, Urusan Keuangan.



Gambar 2.1 Struktur Organisasi

Unit Pelaksana Teknis Pusat Komputer Universitas Gadjah Mada Yogyakarta

2.4 GAMANET.UGM.AC.ID

Gamanet.ugm.ac.id merupakan domain yang berada dibawah ugm.ac.id. Gamanet.ugm.ac.id digunakan untuk memberikan layanan *e-mail* kepada mahasiswa, pegawai dan dosen dari Universitas Gadjah Mada yang selanjutnya disebut dengan pengguna.

Selain memberikan layanan e-mail kepada pengguna pihak pengelola dari gamanet.ugm.ac.id juga memberikan layanan tambahan berupa layanan *webhosting*. Layanan webhosting diberikan secara gratis kepada pengguna. Layanan webhosting ini adalah layanan yang memungkinkan pengguna untuk membuat situs pribadi dengan alamat http://gamanet.ugm.ac.id/~nama_pengguna.

Pengelolaan gamanet.ugm.ac.id diberikan kepada seksi jaringan Unit Pelaksana Teknis Pusat Komputer. Pengelola menyediakan fasilitas telnet dan command line FTP untuk menggunakan layanan tersebut.

Untuk menjadi pengguna dari gamanet.ugm.ac.id mahasiswa, pegawai dan dosen Universitas Gadjah Mada harus melakukan registrasi. Formulir registrasi dapat diperoleh langsung di seksi jaringan Pelaksana Teknis Pusat Komputer. Proses registrasi sampai saat ini belum dapat dilakukan secara online. Hal ini dikarenakan adanya biaya pendaftaran yang harus diserahkan langsung kepada petugas.

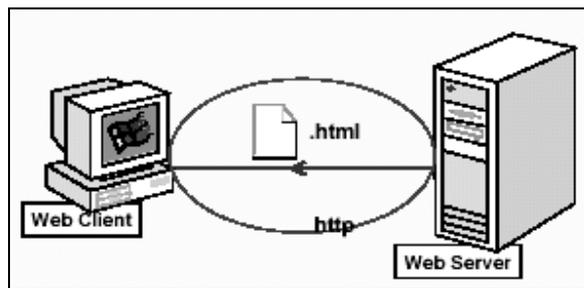
Berdasarkan pengamatan terakhir jumlah pengguna gamanet.ugm.ac.id adalah 107 pengguna. Dengan spesifikasi dari mahasiswa 78 pengguna, organisasi 9 pengguna, pegawai/dosen 20 pengguna. Dari 107 pengguna tersebut yang memanfaatkan layanan *webhosting* hanya 5 orang.

BAB III

LANDASAN TEORI

3.1 *World Wide Web*

World Wide Web adalah suatu sarana pembagian informasi antar pengguna jaringan komputer. Dengan adanya *world wide web* maka informasi (berupa tulisan, gambar, suara, atau video) dapat diberikan kepada orang lain sehingga memungkinkan terjadinya pertukaran pengetahuan atau pembagian pekerjaan.



Gambar 3. 1 Konsep *World Wide Web*

World Wide Web (biasa disingkat *WWW*) adalah salah satu dari sekian banyak layanan yang ada di internet. Layanan ini paling banyak digunakan di internet untuk menyampaikan informasi karena sifatnya mendukung multimedia. Artinya informasi tidak hanya disampaikan melalui teks, tapi juga gambar, video dan suara. *Web server* merupakan server internet yang mampu melayani koneksi transfer data dalam protocol HTTP. *Web server* merupakan hal yang terpenting dari server di internet dibandingkan server lainnya seperti *e-mail server*, *ftp server* ataupun *news server* (Onno W. Purbo, 2000). Hal ini di sebabkan *web server* telah dirancang untuk dapat

melayani beragam jenis data, dari teks sampai grafis tiga dimensi. Kemampuan ini telah menyebabkan berbagai institusi seperti universitas maupun perusahaan dapat menerima kehadirannya dan juga sekaligus menggunakannya sebagai sarana di internet.

Web server juga dapat menggabungkan dengan dunia mobile *wireless* internet atau yang sering disebut sebagai *WAP (Wireless Access Protocol)* yang banyak digunakan sebagai sarana *handphone* yang memiliki fitur WAP. Dalam kondisi ini, *webservice* tidak lagi melayani data file HTML tetapi telah melayani *WML (wireless Markup Language)*.

Web client adalah komputer yang tergabung dalam jaringan atau internet yang meminta informasi. Untuk dapat mengakses *web server*, *web client* menggunakan aplikasi yang disebut *Web browser*. *Web browser* meminta dan menerima data dari *web server* melalui suatu protokol yang disebut *http (hypertext transfer protocol)*. Protokol ini bertugas untuk mengirimkan perintah dari *web browser* ke *web server* serta mengirimkan file/data dari *web server* ke *web browser*. File yang dikirim dalam layanan web ini berekstensi ***.htm**, ***.html**, ***.php**, ***.php3** dan lain-sebagainya. HTML merupakan singkatan dari *hypertext markup language*, yaitu satu bahasa yang digunakan untuk mendefinisikan susunan informasi dalam file *hypertext*. *Hypertext* sendiri adalah suatu struktur penyampaian informasi dimana satu atau beberapa kata pada suatu file dapat di-*link* untuk mengeluarkan file baru yang biasanya berisi informasi detail tentang kata tersebut. Untuk mengakses informasi yang disediakan

oleh suatu komputer pada Internet, diperlukan alamat *URL (Uniform Resource Locator)* informasi tersebut berada.

3.2 Apache

Salah satu perangkat lunak yang biasa digunakan untuk *web server* oleh banyak web master di dunia adalah apache. perangkat lunak tersebut dapat kita *download* secara gratis dari web resmi apache yaitu <http://www.apache.org>. Apache merupakan perangkat lunak *open source* yang sekarang ini sudah merebut pasar dunia lebih dari 50%. *Web server* ini fleksibel terhadap berbagai system operasi seperti windows9x/Me/NT/2000/XP ataupun unix/linux. Dalam penggunaannya. Apache merupakan turunan dari *webserver* yang dikeluarkan oleh NCSA yaitu NCSA HTTPd pada sekitar tahun 1995. Kelebihan *web server* Apache :

1. *Freeware* (perangkat lunak yang gratis)
2. Mudah diinstall.
3. Mampu beroperasi pada berbagai *platform* sistem operasi .
4. Mudah mengonfigurasinya.
5. Apache *Web server* mudah dalam menambahkan periferal lainnya ke dalam *platform web server*nya, misalnya : untuk menambahkan modul, cukup hanya mengeset file konfigurasinya agar mengikutsertakan modul itu kedalam kumpulan modul lain yang sudah dioperasikan.
6. Dapat dijadikan pengganti bagi NCSA *web server*.
7. Perbaikan terhadap kerusakan dan error pada NCSA 1.3 dan 1.4

8. Merespon client lebih cepat daripada server NCSA.
9. Mampu di *kompilasi* sesuai dengan spesifikasi HTTP yang sekarang.
10. Menyediakan feature untuk *multihomed* dan *virtual server*.
11. Kita dapat mengeset respon error yang akan dikirim *web server* dengan menggunakan file atau skrip.
12. Secara otomatis menjalankan file *index.html*, halaman utamanya, untuk ditampilkan secara otomatis pada clientnya.
13. Lebih aman karena memiliki level-level pengamanan
14. Apache mempunyai komponen dasar terbanyak di antara *webserver-webserver* lain, yang berarti bahwa *webserver* Apache termasuk salah satu dari *webserver* yang lengkap.
15. Performansi dan konsumsi sumberdaya (*resource*) dari *webserver* apache tidak terlalu banyak, hanya sekitar 20 MB untuk file-file dasarnya dan setiap *daemonnya* hanya memerlukan sekitar 950 KB memory per-*child*.
16. Mendukung transaksi yang aman (*secure transaction*) menggunakan SSL (*Secure Socket Layer*).
17. Mempunyai dukungan teknis melalui web.
18. Mempunyai kompatibilitas *platform* yang tinggi.
19. Mendukung *third party* berupa modul-modul tambahan.

3.3 PHP (PHP Hypertext Preprocessor)

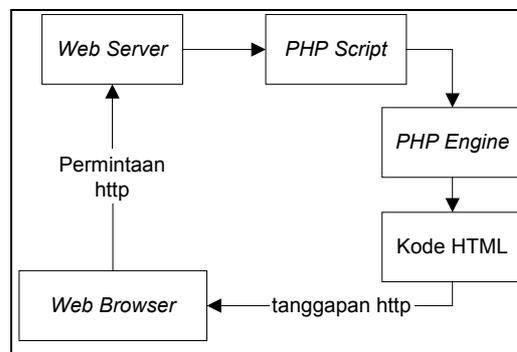
PHP merupakan singkatan dari **PHP Hypertext Preprocessor** adalah sebuah bahasa *scripting* yang terpasang pada HTML. Sebagian besar sintaks mirip dengan bahasa C, Java dan Perl, ditambah beberapa fungsi PHP yang spesifik. Tujuan utama bahasa ini adalah untuk memungkinkan perancang web menulis halaman web dinamik dengan cepat.

3.3.1 Pengertian PHP

Halaman web biasanya disusun dari kode-kode html yang disimpan dalam sebuah file berekstensi .html. File html ini dikirimkan oleh server (atau file) ke browser, kemudian browser menerjemahkan kode-kode tersebut sehingga menghasilkan suatu tampilan yang indah. Lain halnya dengan program php, program ini harus diterjemahkan oleh *webserver* sehingga menghasilkan kode html yang dikirim ke browser agar dapat ditampilkan. Program ini dapat berdiri sendiri ataupun disisipkan di antara kode-kode html sehingga dapat langsung ditampilkan bersama dengan kode-kode html tersebut. Program php dapat ditambahkan dengan menggapit program tersebut di antara tanda `<? dan ?>`. Tanda-tanda tersebut biasanya disebut tanda untuk *escaping* (kabur) dari kode html. File html yang telah dibubuhi program php harus diganti ekstensi-nya menjadi *.php* atau *.php3*.

PHP merupakan bahasa pemrograman web yang bersifat *server-side HTML=embedded scripting*, di mana script-nya menyatu dengan HTML dan berada

di server. Artinya adalah sintaks dan perintah-perintah yang kita berikan akan sepenuhnya dijalankan di server tetapi disertakan HTML biasa. PHP dikenal sebagai bahasa scripting yang menyatu dengan tag HTML, dieksekusi di server dan digunakan untuk membuat halaman web yang dinamis seperti *ASP (active Server Pages)* dan *JSP (Java Server Pages)*.



Gambar 3. 2 Konsep PHP

PHP pertama kali dibuat oleh Rasmus Lerdroft, seorang programmer C. Semula PHP digunakannya untuk menghitung jumlah pengunjung di dalam webnya. Kemudian ia mengeluarkan *Personal Home Page Tools* versi 1.0 secara gratis. Versi ini pertama kali keluar pada tahun 1995. Isinya adalah sekumpulan script PERL yang dibuatnya untuk membuat halaman webnya menjadi dinamis. Kemudian pada tahun 1996 ia mengeluarkan PHP versi 2.0 yang kemampuannya telah mampu mengakses basis data dan dapat terintegrasi dengan HTML.

Pada tahun 1998 tepatnya pada tanggal 6 Juni 1998 keluarlah PHP versi 3.0 yang dikeluarkan oleh Rasmus sendiri bersama kelompok pengembang softwarena..

Versi terbaru yaitu PHP 4.0 keluar pada tanggal 22 Mei 2000 merupakan versi yang lebih lengkap lagi dibandingkan dengan versi sebelumnya. Perubahan yang paling mendasar pada PHP 4.0 adalah terintegrasinya Zend Engine yang dibuat oleh Zend Suraski dan Andi Gutmans yang merupakan penyempurnaan dari PHP3 scripting engine. Yang lainnya adalah build in HTTP session, tidak lagi menggunakan library tambahan seperti pada PHP3. Tujuan dari bahasa scripting ini adalah untuk membuat aplikasi-aplikasi yang dijalankan di atas teknologi web. Dalam hal ini, aplikasi pada umumnya akan memberikan hasil pada *web browser*, tetapi prosesnya secara keseluruhan dijalankan *web server*.

3.3.2 Kelebihan PHP

Ketika *e-commerce* semakin berkembang, situs-situs yang statis pun semakin ditinggalkan karena dianggap sudah tidak memenuhi keinginan pasar karena situs tersebut harus tetap dinamis setiap hari. Pada saat ini bahasa PERL dan CGI sudah jauh ketinggalan jaman sehingga sebagian besar desainer web banyak beralih ke bahasa *server-side scripting* yang lebih dinamis seperti PHP.

Seluruh aplikasi berbasis web dapat dibuat dengan PHP. Namun kekuatan yang paling utama PHP adalah pada konektivitasnya dengan system basis data di dalam web. Sistem basis data yang dapat didukung oleh PHP adalah : *Oracle*, *MySQL*, *Sybase*, *PostgreSQL*, dan lainnya .

PHP dapat berjalan di berbagai system operasi seperti windows 98/NT, UNIX/LINUX, solaris maupun macintosh. PHP merupakan perangkat lunak yang *open source* yang dapat anda download secara gratis dari situs resminya yaitu <http://www.php.net>, ataupun dari situs-situs yang menyediakan perangkat lunak tersebut seperti di <ftp://gerbang.che.itb.ac.id>. Perangkat lunak ini juga dapat berjalan pada *web server* seperti PWS (*Personal Web Server*), Apache, IIS, AOLServer, fhttpd, phttpd dan sebagainya. PHP juga merupakan bahasa pemrograman yang dapat kita kembangkan sendiri seperti menambah fungsi-fungsi baru.

Keunggulan lainnya dari PHP adalah PHP juga mendukung komunikasi dengan layanan seperti protocol IMAP, SNMP, NNTP, POP3 bahkan HTTP. PHP dapat diinstal sebagai bagian atau modul dari apache web server atau sebagai CGI script yang mandiri. Banyak keuntungan yang dapat diperoleh jika menggunakan PHP sebagai modul dari apache di antaranya adalah:

1. Tingkat keamanan yang cukup tinggi
2. Waktu eksekusi yang cepat.
3. Akses ke system basis data yang fleksibel. seperti MySQL, Interbase, PosgreSQL dan lainnya.

3.4 MySQL

SQL (Structured Query Language) adalah bahasa standar yang digunakan untuk mengakses server basis data . Sejak tahun 70-an bahasa ini telah dikembangkan oleh

IBM, yang kemudian diikuti dengan adanya Oracle, Informix dan Sybase. Dengan menggunakan SQL, proses akses basis data menjadi lebih *user-friendly* dibandingkan dengan misalnya dBase ataupun Clipper yang masih menggunakan perintah-perintah pemrograman murni.

Selain MySQL, ada beberapa jenis pemrograman yang berorientasi basis data yang dapat digunakan untuk aplikasi di web seperti Oracle. Oracle merupakan sebuah perusahaan besar di dunia yang cakupan bisnis salah satunya adalah penjualan perangkat lunak dan pembuatan perangkat lunak basis data yang diperuntukkan bagi perusahaan-perusahaan besar di dunia. Karena softwarena tidak bebas dipasaran atau tidak *free* perangkat lunak maka sebagian besar perusahaan kecil atau menengah atau programmer web masih menggunakan basis data MySQL sebagai perangkat lunak basis data perusahaan atau webnya.

MySQL adalah sebuah server basis data SQL *multiuser* dan *multi-threaded*. SQL sendiri adalah salah satu bahasa basis data yang paling populer di dunia. Implementasi program server basis data ini adalah program daemon **mysqld** dan beberapa program lain serta beberapa pustaka.

MySQL dibuat oleh TcX dan telah dipercaya mengelola sistem dengan 40 buah basis data berisi 10,000 tabel dan 500 di antaranya memiliki 7 juta baris (kira-kira 100 gigabyte data). Basis data ini dibuat untuk keperluan sistem basis data yang cepat, handal dan mudah digunakan. Walaupun memiliki kemampuan yang cukup baik, MySQL untuk sistem operasi Unix bersifat freeware, dan terdapat versi shareware untuk sistem operasi windows.

Sebagaimana basis data sistem yang lain, dalam SQL juga dikenal hierarki server dengan basis data-basis data. Tiap-tiap basis data memiliki tabel-tabel. Tiap-tiap tabel memiliki field-field.

Umumnya informasi tersimpan dalam tabel-tabel yang secara logik merupakan struktur 2 dimensi terdiri atas baris dan kolom. Field-field tersebut dapat berupa data seperti int, realm char, date, time dan lainnya. SQL tidak memiliki fasilitas pemrograman yang lengkap, tidak ada looping ataupun percabangan ,misalnya. Sehingga untuk menutupi kelemahan ini perlu digabung dengan bahasa pemrograman seperti C.

3.4.1 Keunggulan MySQL

1. MySQL merupakan program yang *multi-threaded*, sehingga dapat dipasang pada server yang memiliki multi-CPU.
2. Didukung program-program umum seperti C, C++, Java, Perl, PHP, Python, TCL APIs dls.
3. Bekerja pada berbagai platform. (tersedia berbagai versi untuk berbagai sistem operasi).
4. Memiliki jenis kolom yang cukup banyak sehingga memudahkan konfigurasi sistem basis data.
5. Memiliki sistem keamanan yang baik dengan verifikasi host.
6. Mendukung ODBC untuk sistem operasi Microsoft Windows.

7. Mendukung record yang memiliki kolom dengan panjang tetap atau panjang bervariasi. dan masih banyak keunggulan lainnya (lihat manual mysql_3.22)
8. MySQL merupakan perangkat lunak yang gratis, dan bisa di download di www.mysql.com. Sedangkan perangkat lunak basis data lainnya seperti ORACLE merupakan perangkat lunak yang harus di beli.
9. MySQL dan PHP saling terintegrasi. Maksudnya adalah pembuatan basis data dengan menggunakan sintak PHP dapat di buat. Sedangkan input yang di masukkan melalui aplikasi web yang menggunakan script *server-side* seperti PHP dapat langsung dimasukkan ke basis data MySQL yang ada di server dan tentunya web tersebut berada di sebuah *web server*.

3.4.2 Sistem Server Basis data MySQL

Sistem basis data MySQL memiliki sistem sekuritas dengan tiga verifikasi yaitu *user(name)*, *password* dan *host*. Verifikasi *host* memungkinkan untuk membuka sekuriti di **localhost**, tetapi tertutup bagi host lain (bekerja di lokal komputer). Sistem sekuriti ini ada di dalam basis data mysql dan pada tabel user. Proteksi juga dapat dilakukan terhadap basis data, tabel, hingga kolom secara terpisah.

3.5 Konsep Dasar Basis Data

Sistem basis data adalah suatu sistem yang mempunyai fungsi untuk mengorganisir data dan menyediakan informasi pada saat diperlukan. C.J , Date (1988) mendefinisikan Basis Data sebagai suatu kumpulan data operasional yang tersimpan dan dapat digunakan oleh sistem aplikasi–aplikasi dalam sebuah organisasi. Dengan kata lain, basis data merupakan gudang atau tempat penyimpanan dari berkas file yang terkomputerisasi.

Di antara basis data secara fisik dan pemakai terdapat suatu lapisan yang disebut *DataBase Manager* (DB Manager), atau lebih sering disebut dengan *DataBase Management System* (DBMS). Semua permintaan pemakai untuk mengakses database ditangani oleh DBMS ini.

Secara umum bahasa basis data terbagi dalam dua komponen, yaitu:

1. *Data Definition Language* (DDL).

DDL adalah bahasa yang digunakan untuk mendefinisikan skema ke DBMS. Skema adalah deskripsi lengkap tentang medan, rekaman, dan hubungan data pada basis data. Bahasa ini mendukung pembuatan tabel baru, pembuatan indeks, dan perubahan tabel.

2. *Data Manipulation Language* (DML).

DML adalah perintah-perintah yang digunakan untuk memanipulasi dan mengambil data pada suatu basis data. Manipulasi data misalnya:

1. Penyisipan atau penambahan data baru ke suatu basis data.
2. Penghapusan data dari suatu basis data.

3. Perubahan data di suatu basis data.

DML merupakan bahasa yang berfungsi memudahkan pemakai untuk mengakses data.

3.5.1 Normalisasi

Normalisasi adalah *suatu proses untuk mengubah suatu relasi yang memiliki masalah tertentu ke dalam dua buah relasi atau lebih yang tidak memiliki masalah tersebut* (Abdul Kadir, 1999). Normalisasi memberikan panduan yang sangat membantu bagi pengembang untuk mencegah penciptaan struktur tabel yang kurang fleksibel atau mengurangi ketidakefisienan.

Anomali adalah proses dalam basis data yang memberikan efek samping yang tidak diharapkan (misalnya menyebabkan ketidakkonsistenan data atau membuat suatu data menjadi hilang ketika data lain dihapus). Proses normalisasi diharapkan dapat menghilangkan anomali-anomali yang ditemukan.

Aturan-aturan normalisasi dinyatakan dalam istilah bentuk normal. Bentuk normal adalah suatu aturan yang dikenakan pada relasi-relasi dalam basis data dan harus dipenuhi oleh relasi-relasi tersebut pada level-level normalisasi. Suatu relasi dikatakan berada dalam bentuk normal tertentu jika memenuhi kondisi-kondisi tertentu.

Menurut Gerald V. Post proses normalisasi data melalui tahap-tahap sebagai berikut:

1. Bentuk Tidak Normal (0NF)

Yaitu bentuk file yang masih terdapat grup-grup berulang (*repeating group*), ketergantungan data secara parsial (*partial dependency*) atau ketergantungan data secara transitif (*transitif dependency*).

2. Bentuk Normal 1 (1NF)

Semua grup-grup yang berulang harus di pecah ke dalam tabel-tabel baru. Pastikan bahwa tabel baru yang dibuat menyertakan salinan kunci dari tabel asal. Tabel yang memegang grup yang berulang harus mempunyai kunci gabungan (*composit keys*) sehingga data dapat dikombinasi kembali dalam *query*.

3. Bentuk Normal Kedua (2NF)

Bentuk normal kedua menyatakan bahwa tabel harus memenuhi bentuk normal pertama. Setiap field bukan kunci harus tergantung pada kuncinya.

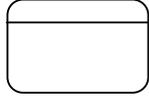
4. Bentuk Normal Ketiga (3NF)

Syarat bentuk normal ketiga adalah tabel harus memenuhi bentuk normal kedua. Syarat lain adalah setiap field bukan kunci harus tergantung hanya pada kuncinya. Sehingga ketergantungan tersembunyi dapat dihilangkan.

3.5.2 Diagram Arus Data (DAD)

Diagram Arus Data (DAD) merupakan *alat yang biasa dipakai untuk mendokumentasikan proses dalam sistem*, cara menggunakan informasi yang tersimpan dan memindahkan informasi antar fungsi di dalam sistem (Abdul Kadir, 1999). DAD dapat digunakan untuk menggambarkan arus data di dalam sistem dengan terstruktur dan jelas. Diagram Konteks adalah *DAD yang memperlihatkan sistem sebagai sebuah proses* (Abdul Kadir, 1999). Tujuannya adalah memberikan pandangan umum sistem. Diagram konteks memperlihatkan sebuah proses berinteraksi dengan lingkungannya. Ada pihak luar atau lingkungan yang memberi masukan dan ada pihak yang menerima keluaran sistem. Dalam hal ini pihak luar dapat berupa sistem lain, suatu perangkat keras, orang, atau organisasi.

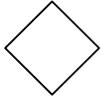
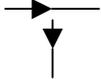
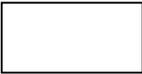
Tabel 3. 1 Tabel Simbol-Simbol Diagram Arus Data

Simbol	Arti
	Terminator
	Proses data
	Aliran data
	Penyimpanan data

3.5.3 Diagram alir (*Flowchart*)

Diagram alir adalah diagram yang menjelaskan urutan kerja dari program yaitu proses pembacaan data, pemrosesan data, pengambilalihan keputusan terhadap data dan penyajian hasil pemrosesan data. Diagram alir digunakan untuk mempermudah pemahaman akan algoritma dari sebuah program dan untuk proses dokumentasi. Diagram alir akan memberikan gambaran yang jelas akan logika dari program yang akan dibuat. Diagram alir memiliki simbol-simbol standar yang mempunyai arti sendiri-sendiri, yaitu:

Tabel 3. 2 Simbol-Simbol Diagram Alir

Simbol	Arti
	Keputusan
	Awal atau akhir program
	Penghubung
	Masukan atau keluaran
	Arah aliran
	Proses

BAB IV

DESAIN APLIKASI WEBHOSTING

4.1 Analisis Sistem Lama

Layanan *e-mail* yang dikelola oleh pihak UPT PUSKOM Universitas Gadjah Mada mempunyai fasilitas tambahan yaitu layanan *webhosting*. Fasilitas *webhosting* digunakan untuk menampilkan situs seorang pengguna. Pembuatan situs seorang pengguna tersebut diperlukan proses *upload* dan *download* file, dan manajemen file. Fasilitas lain yang dapat diletakkan pada situs tersebut antara lain buku tamu dan pencatat kunjungan. Untuk menggunakan fasilitas di atas pihak pengelola tidak menyediakan sistem khusus. Pihak pengelola hanya menyediakan fasilitas *telnet server* (manajemen file), *ftp server* (*upload* dan *download*) dan *web server* (menampilkan situs). Untuk menggunakan layanan *webhosting* maka pengguna dapat memakai sistem standar untuk *telnet* (misal *putty*, *telnet dos*), *ftp* (misal *ftp dos*, *ws ftp*, *cute ftp*) dan *web* (*internet explorer*, *netscape*, *mozilla*, *konqueror*).

Dengan melihat kenyataan di atas, maka diperlukan sebuah sistem yang dapat menggabungkan *telnet*, *ftp*, dan *web*. Sistem tersebut dinamakan aplikasi ***webhosting***. Sebelum dibuat aplikasi *webhosting* pengguna direpotkan dengan banyaknya sistem yang harus dipasang pada komputer. Selain itu kebanyakan sistem masih menggunakan *command line* (pengguna perlu mengetikkan perintah-perintah secara langsung) dan tidak mudah untuk digunakan.

Aplikasi *webhosting* ini diharapkan mampu menangani empat proses yang utama, pertama proses *upload* dan *download*. Dengan adanya proses ini maka pengguna dapat mengirimkan file berupa ke server dan juga dapat mengambil file dari server. Hal kedua adalah manajemen file pada server yang dapat menangani penghapusan file, pengubahan nama file dan pengubahan hak akses pada file. Untuk proses pertama dan kedua didalam aplikasi akan dibuat dalam satu modul. Proses ketiga adalah otomasi pembuatan halaman web yang hanya mengisi data-data sehingga pengguna tidak perlu mempunyai pengetahuan tentang pembuatan halaman web. Proses terakhir adalah pemberian fasilitas buku tamu dan pencacah kunjungan untuk situs pengguna.

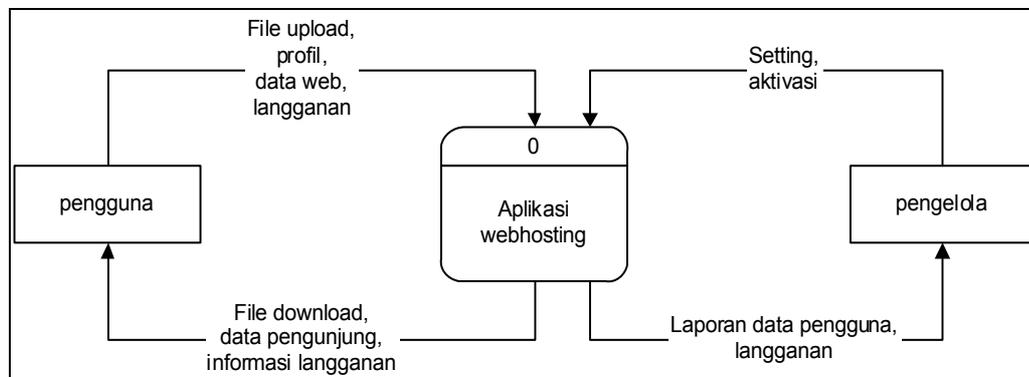
4.2 Desain Model

Desain model pada sistem ini disajikan dalam bentuk *logical model* dan *physical model*. Bentuk *logical model* digambarkan dengan Diagram Arus Data (DAD). Bentuk *physical model* digambarkan dalam diagram alir (*flowchart*).

4.2.1 Diagram Arus Data (DAD)

4.2.1.1 Diagram Konteks

Diagram konteks menggambarkan sumber, proses dan hasil dari informasi umum. Ada pihak luar atau lingkungan yang memberi masukan dan ada pihak yang menerima keluaran aplikasi. Dalam hal ini pihak luar disebut dengan entitas. Secara detail diagram konteks aplikasi *webhosting* sebagai berikut:



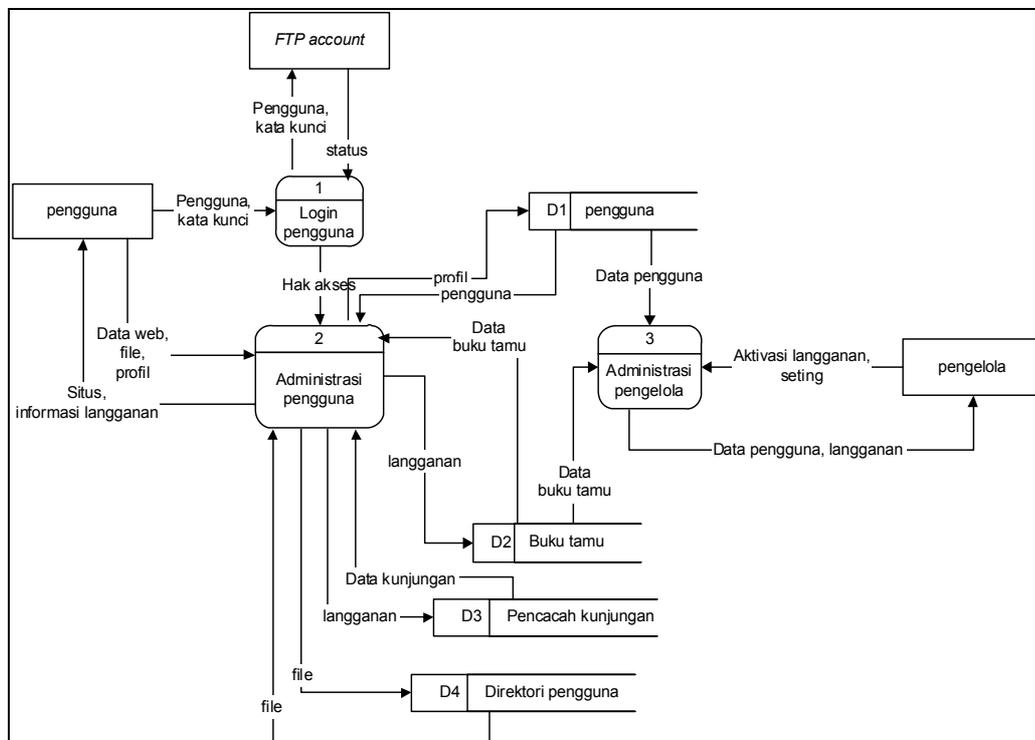
Gambar 4. 1 Diagram Konteks pencacahan dan buku tamu

Diagram konteks diatas memberikan gambaran bahwa aplikasi berinteraksi dengan dua entitas, yaitu pengguna dan pengelola. Pengguna adalah seseorang menjadi anggota dari layanan *webhosting*. Pengelola adalah petugas khusus yang diberi tugas untuk mengelola aplikasi oleh UPT PUSKOM Universitas Gadjah Mada.

Data dari pengguna berupa file, profil pengguna, data-data web dan data langganan akan diterima oleh aplikasi *webhosting*. File, data pengunjung situs, data kunjungan, dan informasi langganan akan diberikan oleh aplikasi *webhosting* kepada pengguna. Selain itu aplikasi juga menerima data seting web dan aktivasi langganan dari pengelola. Data yang diberikan oleh aplikasi kepada pengelola adalah laporan data pengguna, laporan data pelanggan.

4.2.1.2 Diagram Arus Data Level 0

Diagram arus data level 0 merupakan penjabaran yang lebih detail dari diagram konteks. Dibawah ini merupakan diagram arus data level 0 dari aplikasi *webhosting*.



Gambar 4. 2 Diagram Arus Data Level 0

Diagram arus data level 0 diatas memperlihatkan bahwa aplikasi *webhosting* terdiri atas tiga proses, yaitu proses login pengguna, proses administrasi pengguna, dan proses administrasi pengelola. Data pengguna dan kata kunci diperlukan untuk melakukan proses login pengguna. Pada pencocokan pengguna dan kata kunci diperlukan sistem lain yang terintegrasi dalam server yaitu *ftp server*. Sistem *ftp server* akan memberikan nilai balik *true* jika pengguna dan kata kunci yang diberikan cocok dengan pengguna dan kata yang ada di *server*. Proses login pengguna diperlukan untuk menentukan lokasi direktori pengguna, data buku tamu, data pencacah kunjungan. Proses login akan memberikan hak akses kepada pengguna

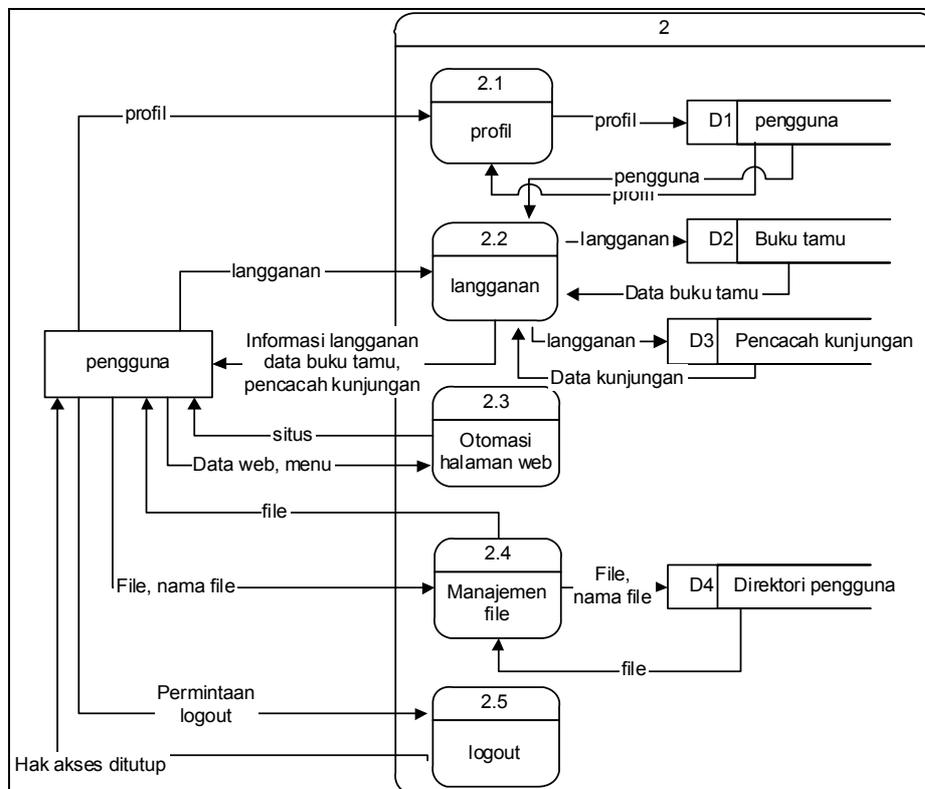
untuk melakukan administrasi. Jadi proses administrasi pengguna sangat tergantung kepada proses login pengguna.

Pada proses administrasi pengguna data yang diberikan kepada pengguna adalah file, informasi langganan, dan situs. Data dari pengguna berupa file, profil dan data web. Data berupa data pengguna, data langganan diberikan oleh proses administrasi pengelola kepada pengelola. Data yang diberikan oleh pengelola kepada proses administrasi pengelola berupa seting dan aktivasi langganan.

4.2.1.3 Diagram Arus Data Level 1

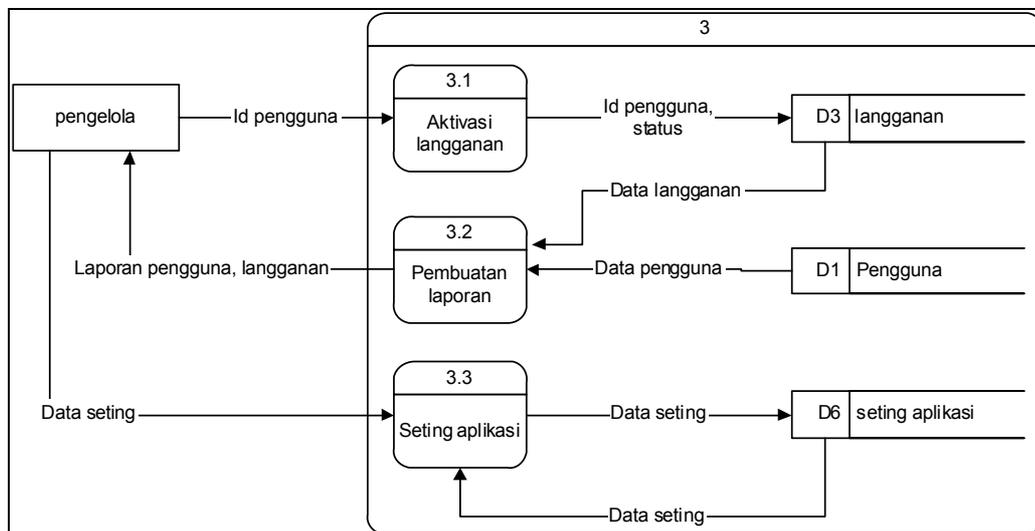
Diagram arus data level 0 diatas dibagi kedalam ke dalam diagram arus data level 1. Penjabaran dilakukan pada proses 2 dan proses 3 karena pada proses-proses tersebut masih mengandung sub-sub proses didalamnya.

Proses 2 pada DAD level 1 ini terdiri dari lima subproses, yaitu profil, langganan, otomasi halaman web, manajemen file, logout. Data berupa detail profil dari pengguna diperlukan pada proses profil. Data dari proses profil disimpan di dalam penyimpanan pengguna. Data berupa konfigurasi berlangganan diberikan oleh pengguna yang akan berlangganan fasilitas tambahan. Data hasil konfigurasi disimpan ke dalam masing-masing jenis langganan.



Gambar 4. 3 DAD Level 1 Proses Pengguna

Detail web, detail menu diperlukan dalam proses otomasi halaman web. Data tersebut akan disimpan oleh proses ke dalam file dan juga ke dalam basis data. File dan nama file diperlukan untuk proses manajemen file. Proses ini membutuhkan file yang diambil dari direktori pengguna. Proses logout akan memberikan data berupa penghapusan hak akses terhadap halaman tersebut.

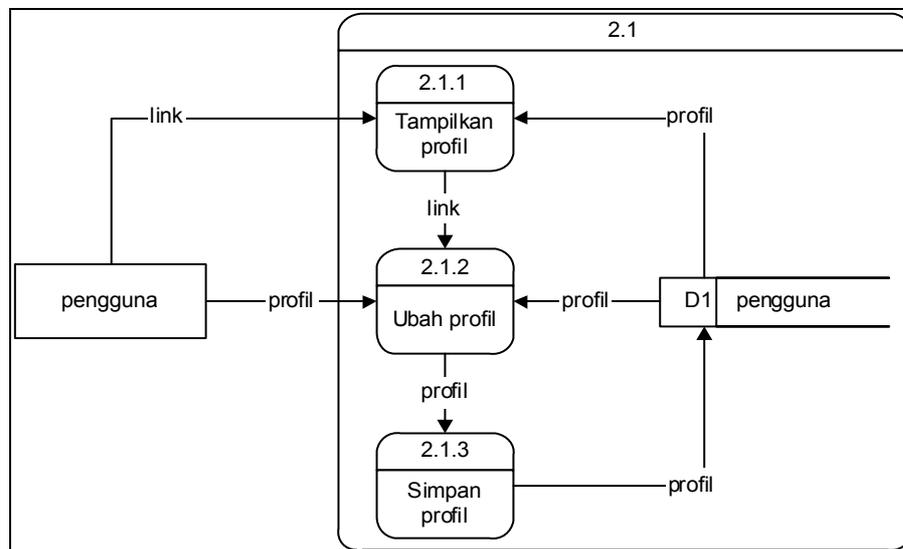


Gambar 4. 4 DAD Level 1 Proses Administrasi Pengelola

Proses 3 terdiri dari tiga sub proses, yaitu proses aktivasi langganan, proses pembuatan laporan, dan proses seting aplikasi. Id pengguna digunakan sebagai kata kunci untuk proses aktivasi langganan. Dengan id pengguna tersebut proses aktivasi langganan akan mengubah status dari langganan pengguna. Proses pembuatan laporan mengambil data langganan dari simpanan langganan. Data tersebut pada proses pembuatan laporan akan ditampilkan kepada pengelola. Data berupa username, host, password, database diperlukan untuk proses seting web. Data tersebut akan digunakan untuk melakukan koneksi ke basis data.

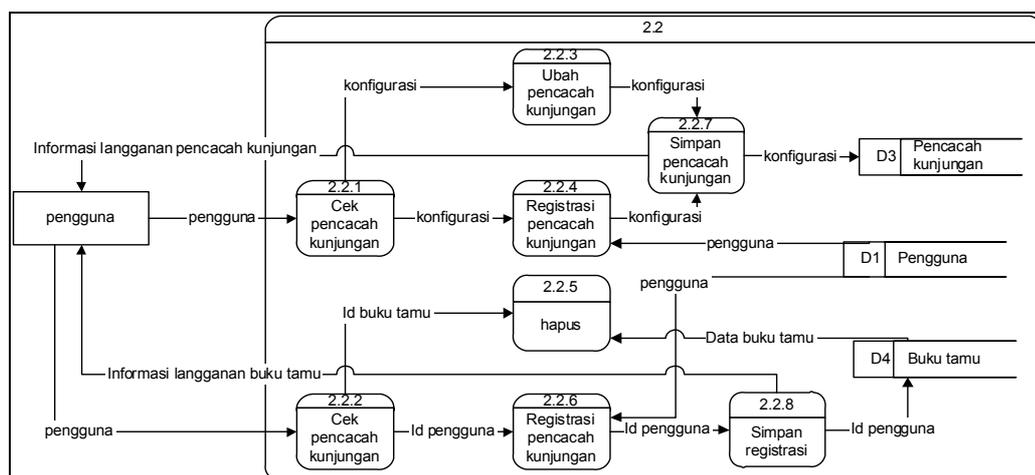
4.2.1.4 Diagram Arus Data Level 2

Pada DAD level 1 banyak terlihat proses-proses yang masih mempunyai sub proses. Untuk proses 2.1 dapat dipecah lagi menjadi DAD level 2 sebagai berikut:



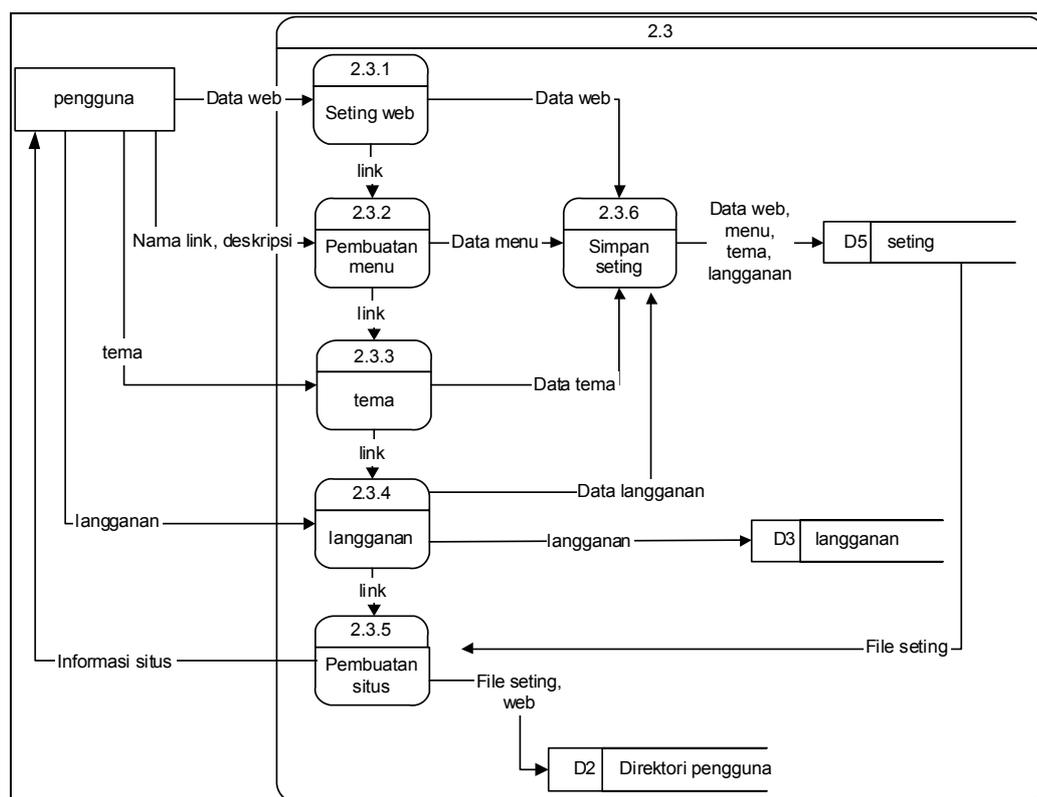
Gambar 4. 5 DAD Level 2 Proses Profil

Proses tampilkan profil mengambil data detail profil dari simpanan. Proses ubah profil mengambil data profil dari simpanan dan profil diganti datanya oleh pengguna. Data yang telah diubah tersebut kemudian di simpan ke dalam simpanan.



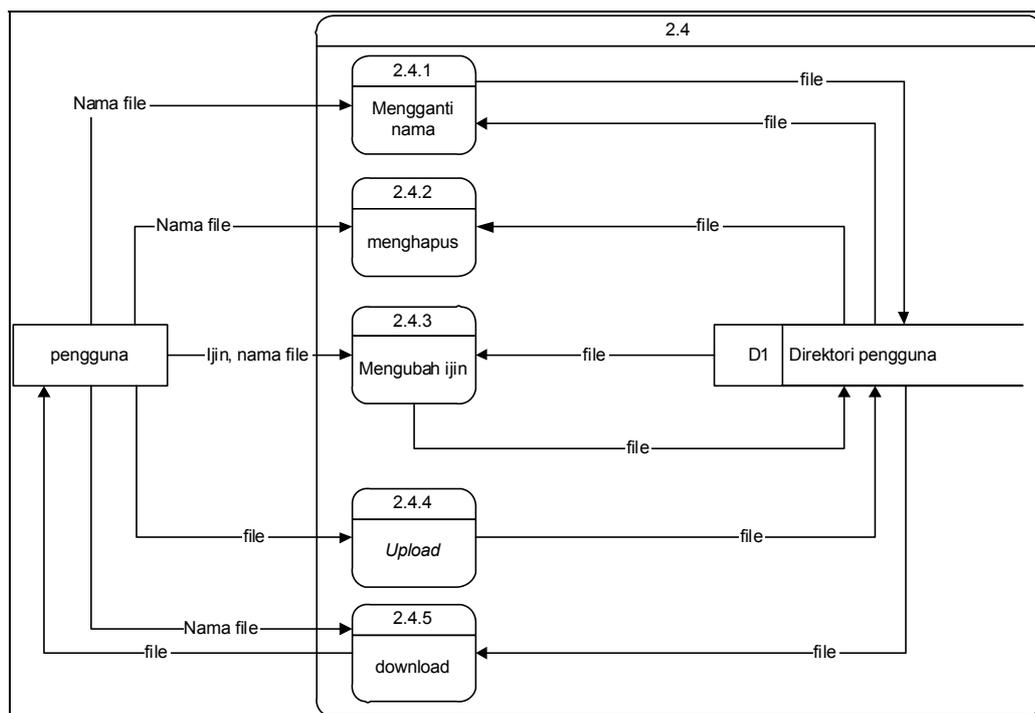
Gambar 4. 6 DAD Level 2 Proses Langganan

Proses langganan terdiri dari dua bagian yang terpisah, bagian pertama adalah langganan pencacah kunjungan dan yang kedua adalah langganan buku tamu. Proses pengecekan pencacah kunjungan untuk pengguna akan dilakukan. Jika sudah terdaftar maka pengguna akan dapat mengubah konfigurasi sebelumnya. Proses registrasi pencacah kunjungan akan dilakukan jika pengguna belum terdaftar. Data yang dimasukkan berupa konfigurasi nilai awal kunjungan dan tipe tampilan pencacah kunjungan. Bagian kedua adalah bagian langganan buku tamu. Proses pengecekan



Gambar 4. 7 DAD Level Otomasi Halaman Web

Data web berupa bagian atas, kaki, deskripsi diperlukan untuk proses seting web, data tersebut disimpan ke file seting. Proses pembuatan menu meminta data berupa link dan deskripsi menu untuk tiap-tiap link. Data tersebut disimpan ke dalam file seting. Data untuk proses tema hanya berupa nama tema yang akan disimpan ke file seting. Proses langganan memberikan data *default* kepada simpanan proses ini sama seperti proses 2.2. Proses pembuatan situs membutuhkan data file seting dan data tema setelah itu dikirim ke direktori pengguna



Gambar 4. 8 DAD Level 2 Proses Manajemen File

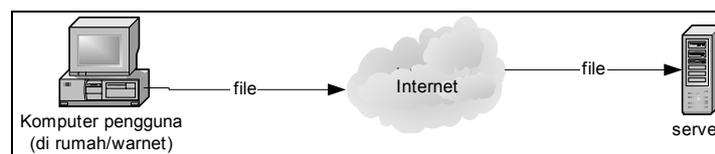
Proses manajemen file terdiri dari lima proses, yaitu proses mengganti nama, proses menghapus, proses mengubah ijin, proses upload dan proses download. Proses penggantian nama memerlukan data berupa nama file yang baru. Nama file

diperlukan pada proses menghapus file/direktori yang berasal dari direktori pengguna. Ijin baru merupakan data yang diberikan oleh pengguna pada proses mengubah ijin. Untuk proses *upload* file sistem memerlukan masukan berupa file lengkap dengan letaknya. Pada proses ini file akan dikirimkan ke direktori pengguna yang berada di server. Proses *download* merupakan mengambil data file yang akan di *download* dan file tersebut dikirim ke direktori pengguna di komputer lokal.

4.2.2 Diagram Alir

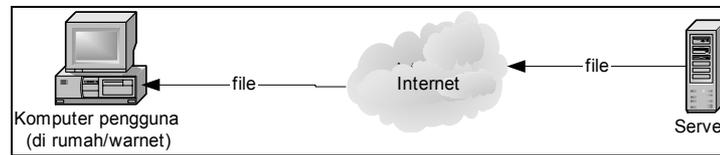
4.2.2.1 Upload dan Download

Proses *upload* dan *download* merupakan proses utama dalam aplikasi *webhosting* ini. Proses *upload* adalah proses pengiriman file dari direktori lokal komputer (komputer pribadi dirumah/diwarnet) ke direktori pengguna di server.

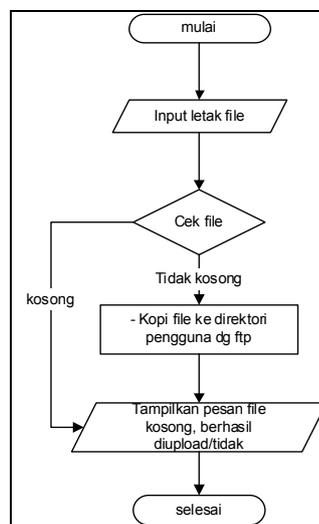


Gambar 4. 9 Proses *Upload File*

Proses *download* adalah proses penyalinan file dari direktori pengguna/direktori yang telah disediakan di server ke direktori lokal pengguna. File di server yang *download* tidak mengalami perubahan sama sekali, sedangkan di direktori lokal pengguna akan bertambah satu buah file.

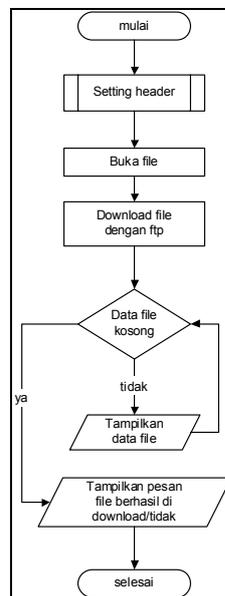


Gambar 4. 10 Proses *Download* File



Gambar 4. 11 Bagan Alir Proses *Upload* File

Diagram alir system (*flowchart system*) diatas menggambarkan secara jelas tentang proses *upload* file. Pada proses *upload* akan dilakukan pengecekan tentang file yang akan di*upload*. Jika file tersebut kosong maka file tersebut tidak di*upload*. Pengiriman file ke server menggunakan koneksi ftp (*file transfer protocol*).



Gambar 4. 12 Diagram Alir Proses *Download* File

Untuk melakukan proses *download* file aplikasi perlu menanyakan telah file yang *download* akan disimpan dimana. Setelah itu maka dilakukanlah proses *download* file dengan cara membuka file tersebut di server dan mengirimkannya berdasarkan *record-record* dari file tersebut menggunakan koneksi ftp.

Cepat atau lambatnya proses *upload* dan proses *download* tergantung pada banyak aspek. Aspek yang paling utama adalah koneksi ke internet dan spesifikasi server. Jika pengguna menggunakan koneksi modem sebesar 33 kbps maka proses diatas akan lebih lambat daripada menggunakan koneksi modem sebesar 56 kbps. Selain itu spesifikasi dari komputer yang digunakan juga mempengaruhi hal tersebut.

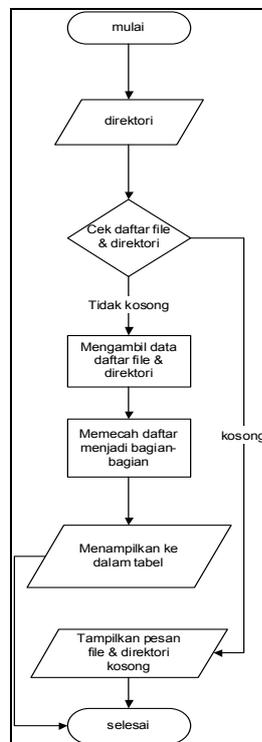
4.2.2.2 Manajemen File

Manajemen file adalah kumpulan proses-proses untuk memanipulasi file. Sebelum dilakukan manipulasi maka perlu untuk ditampilkan daftar file atau direktori dari pengguna secara lengkap (nama, tanggal, ukuran, pemilik, grup, ijin). Pemanipulasian file meliputi: pembuatan direktori, pengubahan nama (*rename*), hapus (*delete*), ijin (*chmod*), dan menuju direktori tertentu. Selain itu di dalam manajemen file terintegrasi proses *upload* dan *download*. Integrasi antara manajemen file dengan proses *upload* dan *download* dikarenakan keduanya sama-sama berhubungan dengan struktur direktori dan file yang ada di server. Pada proses manajemen file maupun *upload* dan *download* digunakan koneksi ftp.

4.2.2.2.1 Daftar File/Direktori

Di dalam manajemen file sangat diperlukan daftar file dan direktori didalam suatu direktori yang ditampilkan secara lengkap. Untuk menampilkan daftar file dan direktori diperlukan masukan berupa direktori yang akan ditampilkan. Setelah direktori yang akan ditampilkan diketahui maka dilakukan proses pengambilan daftar file dan direktori. Hasil dari pengambilan tersebut di pecah menjadi delapan bagian (UNIX) atau tiga bagian (windows).

Daftar file dan direktori ditampilkan berdasarkan data file dan direktori yang telah di pecah tersebut ke dalam tabel. Sehingga untuk tiap file dan direktori mempunyai informasi yang lengkap tetapi terpisah-pisah untuk masing-masing bagian. Setelah data ditampilkan maka proses manipulasi akan dapat dilaksanakan.



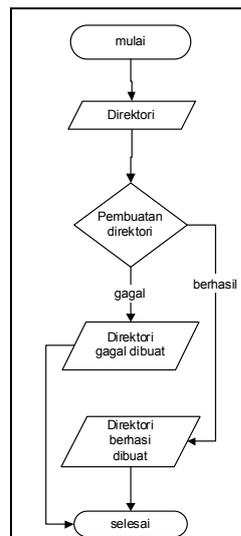
Gambar 4. 13 Diagram Alir Menampilkan Daftar File dan Direktori

4.2.2.2 Pembuatan Direktori

Direktori adalah tempat untuk dimana didalamnya terdapat satu atau lebih file atau direktori. Pada manajemen file ini direktori dapat dibuat jika pengguna berada dalam direktorinya sendiri atau di dalam direktori yang diberi ijin untuk ditulis.

Proses pembuatan direktori memerlukan masukan dari pengguna. Masukan dari pengguna berupa nama dari direktori yang akan dibuat. Aturan pemberian nama untuk direktori sama seperti aturan untuk pembuatan file. Aturan tersebut seperti tidak boleh menggunakan karakter khusus (~!@#\$\$%^&*()_+=|\/?.,). Pemberian ijin

secara *default* untuk direktori yang dibuat bernilai 755. Artinya pengguna dapat menulis, membaca dan mengeksekusinya, selain dari pengguna hanya dapat melihat dan mengeksekusi.

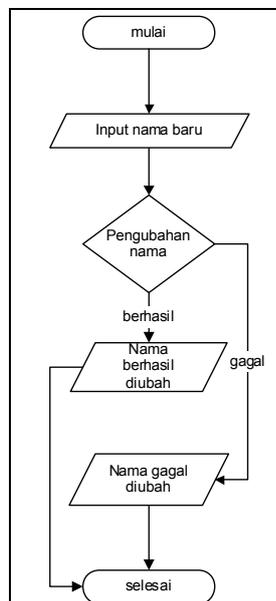


Gambar 4. 14 Diagram Alir Proses Pembuatan Direktori

4.2.2.2.3 Pengubahan Nama (*rename*)

Pada proses ini nama yang dapat diubah adalah nama file dan nama direktori. Proses pengubahan nama adalah proses dimana nama file atau direktori yang lama akan diganti dengan nama yang baru. Pada proses ini yang berubah hanya namanya saja sedangkan isi dan format dari file atau direktori tersebut tidak ikut berubah.

Untuk mengubah nama diperlukan masukan dari pengguna berupa nama baru dari file atau direktori. Aturan pemberian nama baru sama dengan aturan untuk membuat direktori.

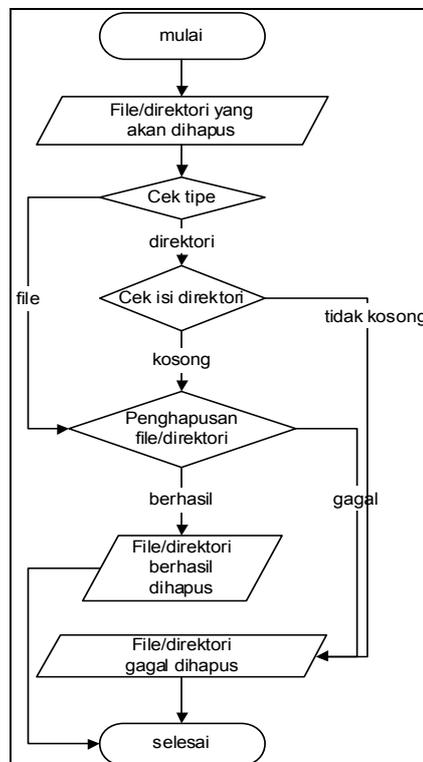


Gambar 4. 15 Diagram Alir Pengubahan Nama

4.2.2.2.4 Hapus (*delete*)

Proses penghapusan dapat berlaku untuk file dan direktori. Penghapusan file atau direktori adalah proses menghilangkan berkas file atau direktori dari media penyimpanan. Setelah penghapusan maka pengguna tidak dapat melakukan manipulasi terhadap file atau direktori tersebut.

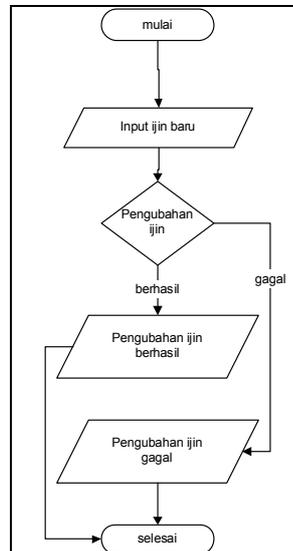
Penghapusan file dan direktori mempunyai prinsip yang berbeda tetapi tetap sama-sama menghilangkan berkas. Pada file proses penghilangan berkas akan langsung dikerjakan. Sedangkan pada direktori proses penghapusan akan dilakukan jika di dalam direktori tersebut tidak terdapat file ataupun direktori.



Gambar 4. 16 Diagram Alir Proses Penghapusan File/Direktori

4.2.2.2.5 Ijin (*chmod*)

Proses perijinan pada aplikasi ini dapat berlaku dengan baik untuk sistem operasi yang berbasis UNIX. Pada sistem operasi yang berbasis UNIX ijin dari file dan direktori dibagi menjadi tiga bagian yaitu pengguna, grup dan lainnya. Untuk tiap bagian mempunyai tiga buah ijin yaitu baca (4), tulis (2) dan eksekusi (1). Dengan adanya ijin tersebut maka file atau direktori akan lebih aman karena akses dibatasi.

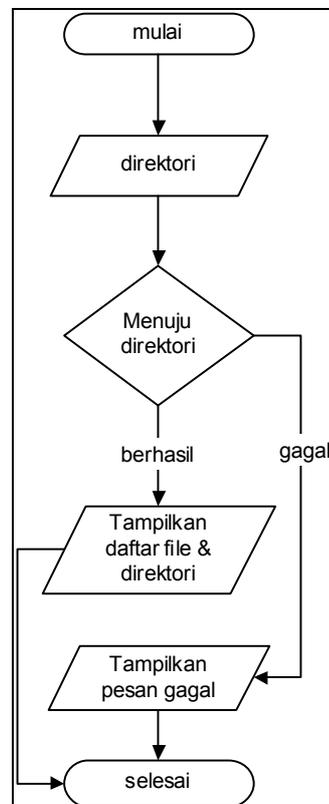


Gambar 4. 17 Diagram Alir Pengubahan Ijin

4.2.2.2.6 Menuju Direktori tertentu

Seringkali pengguna menginginkan pergi ke direktori tertentu tanpa harus melewati direktori-direktori sebelumnya. Untuk mengatasi hal tersebut maka dibuat fasilitas tambahan pada manajemen file yang mengakomodasi keperluan tersebut.

Prinsip dari proses tersebut adalah langsung menuju ke direktori yang dituju dengan memberikan alamat direktori secara lengkap. Setelah menuju direktori tersebut maka ditampilkan file atau direktori yang ada pada direktori tujuan.



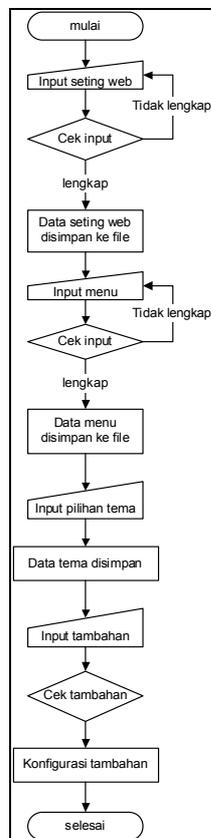
Gambar 4. 18 Proses Menuju Direktori tertentu

4.2.2.3 Otomasi Halaman Web

Otomasi halaman web adalah suatu proses pembuatan halaman-halaman web untuk situs yang berada dibawah domain gamanet.ugm.ac.id yang dibangun secara langsung oleh aplikasi. Pengguna hanya mengisikan data-data untuk kelengkapan situs yang akan dibuat dengan sistem *wizard* yang mudah dimengerti bagi pemula. Dengan fasilitas ini pengguna tidak perlu memiliki pengetahuan tentang pembuatan web, dan bagi pengguna yang sibuk tidak perlu susah-susah meluangkan waktunya untuk membuat web dan meng-*upload*-nya.

Data-data yang dimasukkan oleh pengguna dibagi menjadi empat kategori yaitu seting web, pembuatan menu, tema, tambahan. Kategori seting web meminta pengguna untuk mengisikan empat jenis isian, yaitu logo, bagian kepala, deskripsi web, dan bagian kaki. Pada kategori ini akan menentukan tulisan yang akan muncul pertama kali jika situs tersebut dikunjungi. Kategori selanjutnya adalah pembuatan menu. Jumlah menu yang akan dibuat ditentukan pada pengisian kategori sebelumnya. Pada tiap-tiap menu terdapat nama *link* dan deskripsi untuk tiap-tiap *link*. Kategori ketiga adalah pemilihan tema dari situs. Tema ini akan menentukan tampilan dari situs secara keseluruhan. Kategori keempat adalah kategori tambahan. Pada kategori ini pengguna dipersilahkan untuk memilih menu tambahan yang diberikan. Jika salah satu atau semua item dipilih maka akan ditampilkan halaman konfigurasi untuk masing-masing pilihan.

Setelah semua konfigurasi selesai maka aplikasi akan membuat halaman-halaman web kemudian dikirimkan ke direktori *public_html* dari pengguna menggunakan koneksi ftp. Fasilitas ini akan berjalan dengan baik jika menggunakan apache sebagai *webserver*.



Gambar 4. 19 Diagram Alir Otomasi Halaman Web

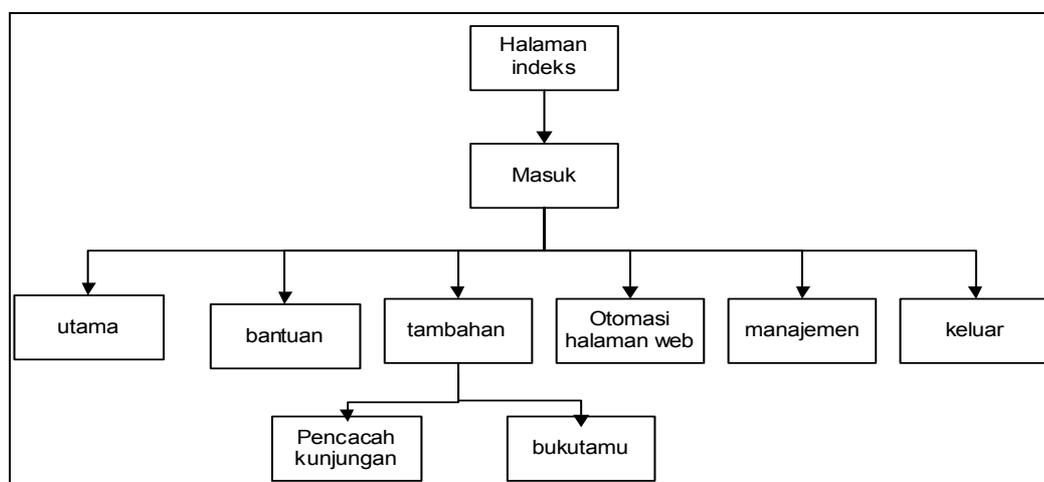
4.3 Desain Input/Output

Proses input dalam aplikasi *webhosting* yang paling utama pada tahap pemasukan data pada bagian otomasi halaman web. Tahap pemasukan data diperlukan untuk menentukan proses otomasi halaman web. Proses ini dilakukan oleh pengguna yang akan membuat web. Data yang dimasukkan ada 4 tahap:

1. Data pribadi pengguna, data ini memuat nama pengguna, nama lengkap, alamat, kota dan kode pos.

2. Kelengkapan *website*, yaitu judul *website*, deskripsi, logo, bagian kepala, bagian kaki.
3. Pembuatan menu, dalam pembuatan menu kita dapat menentukan jumlah menu yang akan dibuat.
4. Fasilitas yang dipakai, pengguna dapat memilih fasilitas yang akan ditampilkan. Fasilitas yang ditawarkan adalah buku tamu, tema, kounter.

Output dari aplikasi *webhosting* berupa daftar file-file yang berada pada direktori pengguna, halaman web dari pengguna dan berupa file. Desain menu aplikasi *webhosting* secara garis besar adalah sebagai berikut:

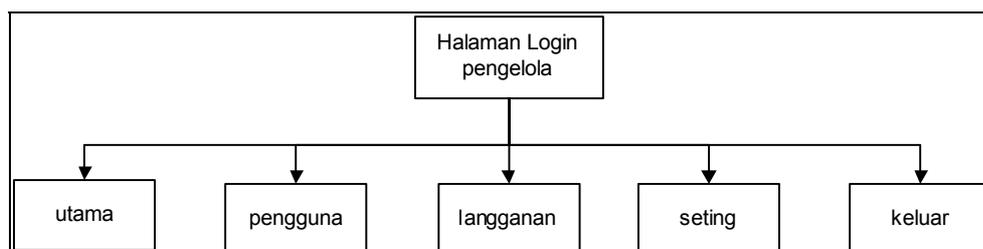


Gambar 4. 20 Desain Menu Pengguna

Menu pengguna dari aplikasi *webhosting* ini terdiri dari enam bagian utama. Untuk masuk ke menu utama pengguna harus melakukan login menggunakan *userid* dan *password* seperti pada waktu pendaftaran.

Bagian utama akan menampilkan profil dari pengguna, jika pengguna belum mengisi profil maka akan muncul tampilan untuk mengisi profil. Selain itu akan dideteksi apakah pengguna sudah membuat *website*, jika belum maka akan muncul *link* untuk membuat *website* menggunakan fasilitas otomasi halaman web. Bantuan akan memberikan panduan penggunaan dari *website* ini. Bagian tambahan akan menampilkan dua buah link untuk membuat kounter dan buku tamu. Otomasi halaman web akan memberikan kemudahan untuk membuat halaman-halaman web hanya dengan mengisi data-data. Manajemen digunakan untuk pengelolaan file dan direktori pengguna. Selain itu digunakan untuk melakukan upload dan download file. Keluar digunakan untuk menutup sesi dari pengguna.

Pengelola mempunyai menu yang tersendiri. Antara pengguna dengan pengelola mempunyai mekanisme login yang berbeda dan menu yang ditampilkan juga berbeda. Berikut menu untuk pengelola:



Gambar 4. 21 Desain Menu Pengelola

Menu untuk pengelola dibagi menjadi lima buah menu. Sebelum pengelola dapat menggunakan menu-menu tersebut, pengelola diwajibkan untuk melakukan login terlebih dahulu. Menu utama merupakan tampilan pertama kali jika pengelola

telah login. Menu pengguna digunakan untuk menampilkan daftar pengguna lengkap dengan profilnya. Menu langganan digunakan untuk menampilkan pengguna dari fasilitas buku tamu dan pencacah kunjungan. Menu seting untuk menampilkan seting yang dapat diubah oleh pengelola. Menu keluar untuk menutup akses terhadap menu pengelola.

4.3.1 Desain Input

4.3.1.1 Desain Input Data Pribadi (Profil)

Bagian kepala	
utama bantuan tambahan website manajemen keluar	
<i>Profil</i>	
Nama pengguna	<input type="text"/>
Nama lengkap	<input type="text"/>
Alamat	<input type="text"/>
Kota	<input type="text"/>
Kode pos	<input type="text"/>
<input type="button" value="Simpan profil"/>	
Bagian kaki	

Gambar 4. 22 Desain Input Data Pribadi (Profil)

Input data pribadi/profil dari pengguna akan muncul jika pengguna belum terdaftar pada basisdata. Bagian ini meminta pengguna untuk memasukkan data-data nama lengkap, alamat, kota dan kode pos. Nama pengguna akan langsung ditampilkan pada label dengan mengambil sesi nama pengguna pada saat login.

Pengisian profil ini digunakan sebagai syarat untuk dapat menggunakan fasilitas kounter dan buku tamu.

Profil dapat diubah jika suatu saat data yang telah dimasukkan ternyata sudah berubah/tidak benar lagi. Profil akan selalu ditampilkan dihalaman utama dari aplikasi *webhosting* untuk setiap pengguna.

4.3.1.2 Desain Input Otomasi Halaman Web

The image shows a web form with the following structure:

- Bagian kepala** (Header): A box containing navigation links: | utama | bantuan | tambahan | website | manajemen | keluar |
- langkah 1 - seting web** (Step 1 - Website Settings):
 - Logo: A text input field followed by a "Browse" button.
 - Bagian kepala: A text input field.
 - Bagian kaki: A text input field.
 - Deskripsi web: A text input field.
- pre langkah 2 - pembuatan menu** (Pre Step 2 - Menu Creation):
 - Menu: A dropdown menu.
 - lanjut >>: A button to proceed to the next step.
- Bagian kaki** (Footer): A box at the bottom of the form.

Gambar 4. 23 Desain Input Seting Web

Langkah pertama pada proses otomasi halaman web adalah seting web. Bagian ini akan menentukan tampilan bagian kepala/atas dari *website* dan juga bagian

kaki dari *website*. Selain itu deskripsi web diperlukan sebagai tampilan utama dari *website*. Logo akan ditampilkan pada bagian atas dari *website*.

Untuk lebih memudahkan dan mempersingkat proses konfigurasi menu dari halaman web tersebut maka ditentukan jumlah menu yang akan dibuat. Pada aplikasi ini ditentukan jumlah maksimum dari menu adalah sepuluh. Jumlah ini ditentukan dengan pertimbangan tampilan untuk menu dan banyaknya data yang dapat ditampilkan.

Selanjutnya adalah tahapan pembuatan menu. Desain dari pembuatan menu seperti dibawah ini:

The image shows a web application interface for creating a menu. It is enclosed in a rectangular border and divided into three main sections:

- Bagian kepala (Header):** Contains a horizontal navigation bar with the text: | utama | bantuan | tambahan | website | manajemen | keluar |
- Langkah 2 - pembuatan menu (Main Content):**
 - Section: **Menu 1**
 - Field: **Judul** (Title) with a text input box.
 - Field: **Deskripsi (akan muncul jika link di klik)** (Description) with a larger text area.
 - Button: **lanjut >>** (Next >>)
- Bagian kaki (Footer):** A simple horizontal bar at the bottom.

Gambar 4. 24 Desain Input Menu

Langkah kedua ini untuk pembuatan menu dengan jumlah menu telah ditentukan pada langkah sebelumnya. Judul dari tiap-tiap menu akan berperan juga

sebagai nama link pada *website*. Deskripsi akan ditampilkan setiap kali pemakai melakukan klik pada menu yang telah dibuat.

The screenshot shows a web form with a header section labeled 'Bagian kepala' containing a navigation menu: '| utama | bantuan | tambahan | website | manajemen | keluar |'. Below the header is a title bar 'Langkah 3 - Tema'. The main content area contains three radio button options: 'Tema 1', 'Tema 2', and 'Tema 3'. A 'lanjut >>' button is positioned below the options. The footer section is labeled 'Bagian kaki'.

Gambar 4. 25 Desain Input Tema

Langkah ketiga adalah pemilihan tema dari *website*. Tema dibuat oleh *webmaster*. Pemilihan tema ini akan menentukan jenis tampilan dari *website* pengguna.

The screenshot shows a web form with a header section labeled 'Bagian kepala' containing a navigation menu: '| utama | bantuan | tambahan | website | manajemen | keluar |'. Below the header is a title bar 'langkah 4 - tambahan'. The main content area contains two rows of radio button options: 'Pencacah kunjungan' with 'dipilih' and 'tidak' options, and 'Buku tamu' with 'dipilih' and 'tidak' options. A 'lanjut >>' button is positioned below the options. The footer section is labeled 'Bagian kaki'.

Gambar 4. 26 Desain Input Tambahan

Bagian ini meminta pengguna untuk memilih dua fasilitas tambahan, yaitu pencacah kunjungan dan buku tamu. Jika pengguna tidak ingin menggunakan salah

satu atau kedua fasilitas tersebut maka pengguna harus memilih bagian tidak. Pengguna akan diberikan tampilan konfigurasi lanjutan untuk pencacah kunjungan jika pengguna memilih pencacah kunjungan. Tetapi jika pengguna tidak memilih pencacah kunjungan maka proses otomatisasi halaman web akan memberikan konfirmasi bahwa proses akan segera selesai jika pengguna menekan tombol selesai.

The image shows a web form titled "Bagian kepala" (Header) with a navigation menu: "| utama | bantuan | tambahan | website | manajemen | keluar |". Below the header is a section titled "langkah 5 - Pencacah Kunjungan" (Step 5 - Visitor Counter). Inside this section is a box labeled "Pencacah Kunjungan" (Visitor Counter) containing three radio button options: "Tipe 1", "Tipe 2", and "Tipe 3". Below the options is a text input field labeled "kunjungan mulai" (start visitor) and a "lanjut >>" (next) button. The form is enclosed in a box labeled "Bagian kaki" (Footer).

Gambar 4. 27 Desain Input Pencacah Kunjungan

Konfigurasi pencacah kunjungan dilakukan untuk menentukan tipe dari pencacah kunjungan dan juga nilai awal dari pencacah kunjungan. Tipe dari pencacah kunjungan diambil dari basis data. Setelah diambil dari basis data maka akan ditampilkan gambar angka lengkap berdasarkan tipe tersebut.

4.3.1.3 Desain Input Upload File

The wireframe shows a rectangular container with a header bar labeled "Bagian kepala" and a footer bar labeled "Bagian kaki". Between these bars is a navigation menu with links: "| utama | bantuan | tambahan | website | manajemen | keluar |". Below the navigation is a section titled "Upload file" in a light gray box. This section contains three rows, each with a label "File 1", "File 2", and "File 3" respectively. Each label is followed by a text input field and a "Browse" button. At the bottom of the "Upload file" section is a central "upload" button.

Gambar 4. 28 Desain Input Upload File

Proses *upload* file membutuhkan masukan berupa file yang akan dikirim ke server. Proses pengiriman file tersebut menggunakan input khusus dengan tipe file. Maksimum jumlah file yang dapat dikirim adalah sepuluh dengan masing-masing file maksimum berukuran 2 MB. Jumlah file yang dapat dikirim ditentukan pada bagian manajemen file.

4.3.1.4 Desain Input Seting Pengelola

Bagian kepala

| utama | pengguna | langganan | seting | keluar |

Judul web

Bagian kaki

Ftp server

Web server

Public html

database

hostname

username

password

Simpan

Bagian kaki

Gambar 4. 29 Desain Input Seting Pengelola

Desain input seting pengelola terdiri dari tiga bagian, yaitu bagian seting web, seting manajemen file dan basis data. Untuk seting web judul web akan ditampilkan pada bagian atas dari *web browser*. Bagian kaki akan ditampilkan pada setiap halaman di bagian paling bawah. Untuk seting manajemen file, *ftp server* digunakan untuk menentukan koneksi ftp dari file manajemen file. *Web server* diperlukan untuk menampilkan situs dari pengguna. *Public html* merupakan lokasi dari situs pengguna. Untuk melakukan koneksi ke basis data diperlukan *nama host* dari server, *nama database*, *username* dan *password*.

4.3.2 Desain Output

4.3.2.1 Desain Output Manajemen File

Bagian kepala

| utama | bantuan | tambahan | website | manajemen | keluar |

Manajemen file

Lingkungan kerja

Sistem operasi:
 direktori sekarang :

<input type="checkbox"/> File 1	ukuran	tanggal	ijin	pemilik	grup	<input type="checkbox"/> hapus	<input type="checkbox"/> ganti	<input type="checkbox"/> Chmod
<input type="checkbox"/> File 2	ukuran	tanggal	ijin	pemilik	grup	<input type="checkbox"/> hapus	<input type="checkbox"/> ganti	<input type="checkbox"/> Chmod
<input type="checkbox"/> File 3	ukuran	tanggal	ijin	pemilik	grup	<input type="checkbox"/> hapus	<input type="checkbox"/> ganti	<input type="checkbox"/> Chmod
<input type="checkbox"/> File 4	ukuran	tanggal	ijin	pemilik	grup	<input type="checkbox"/> hapus	<input type="checkbox"/> ganti	<input type="checkbox"/> Chmod

>>

manipulasi

Menuju direktori:

Pembuatan direktori:

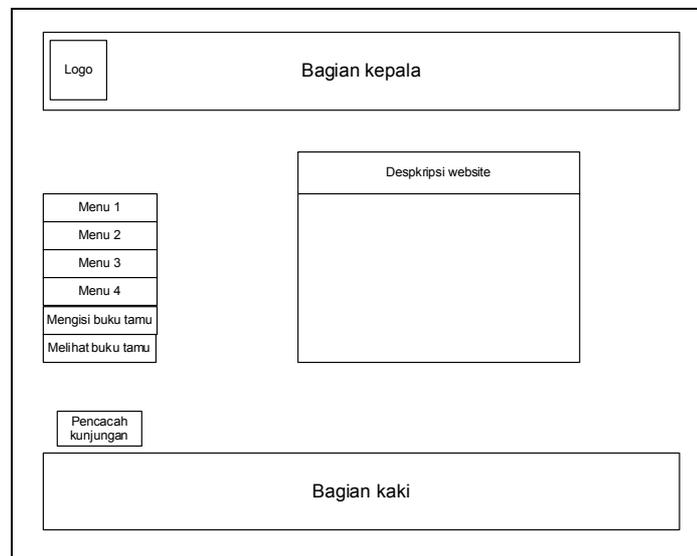
Upload File :

Bagian kaki

Gambar 4. 30 Desain Output Manajemen File

Manajemen file merupakan bagian paling penting. Disini pengguna dapat melakukan perubahan nama file, penghapusan file, perubahan ijin untuk file, *upload* dan *download* file. Selain itu pengguna diberikan panduan letak direktori yang sedang aktif. Pembuatan direktori juga dapat dilakukan pada bagian ini.

4.3.2.2 Desain Output Otomasi Halaman Web



Gambar 4. 31 Desain Output Otomasi Halaman Web

Tampilan dari output dari otomasi halaman web dapat dilihat pada *website* pengguna (terdapat *link* di utama). Jenis tampilan ditentukan ketika pengguna memilih tema pada saat konfigurasi. *Link* mengisi buku tamu dan melihat buku tamu akan ditampilkan jika pengguna memilih buku tamu saat konfigurasi. Begitu juga untuk pencacah kunjungan akan muncul jika saat konfigurasi pengguna memilih fasilitas tersebut.

4.3.2.3 Desain Output Pengguna pada Menu Pengelola

Data yang ditampilkan adalah userid, nama, nama lengkap, alamat, kota dan kode pos. Bagian ini hanya akan menampilkan data pengguna yang pernah masuk ke

menu pengguna. Jika pengguna belum pernah masuk ke menu pengguna maka pengguna tersebut tidak muncul dalam daftar.

Bagian kepala					
utama pengguna langganan seting keluar					
userid	Nama	Nama lengkap	alamat	Kota	Kode pos
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Bagian kaki					

Gambar 4. 32 Desain Output Pengguna pada Menu Pengelola

4.3.2.4 Desain Output langganan pada Menu Pengelola

Bagian kepala			
utama pengguna langganan seting keluar			
userid	registrasi	Jumlah komentar	status
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
userid	registrasi	cacah kunjungan	status
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Bagian kaki			

Gambar 4. 33 Desain Output Langganan pada Menu Pengelola

Pada tampilan ini dibagi menjadi dua bagian, bagian pertama menampilkan pengguna dari fasilitas buku tamu. Data yang ditampilkan adalah userid, tanggal registrasi, jumlah komentar, dan status dari pengguna. Status dari pengguna jika bernilai satu maka penggunaan fasilitas dinyatakan aktif.

Bagian kedua akan menampilkan pencacah kunjungan. Data yang ditampilkan userid, tanggal registrasi, cacah kunjungan dan status pengguna. Sama seperti bagian buku tamu, jika status bernilai 1 maka penggunaan fasilitas dinyatakan aktif.

4.4 Rancangan Umum Sistem

Proses-proses yang ada di dalam sistem secara umum digolongkan dalam beberapa modul. Modul-modul tersebut akan terintegrasi ke dalam sebuah aplikasi.

a. Modul Manajemen File

Modul manajemen file berfungsi untuk melakukan kegiatan yang berhubungan dengan manajemen file. Di dalam modul ini sebetulnya terdapat sebuah modul yang terintegrasi yaitu modul *upload & download* file. Modul ini selalu berhubungan dengan file yang ada di server. Untuk menjalankan modul ini digunakan koneksi ftp.

b. Modul Otomasi Halaman Web

Modul ini akan membuat suatu kesatuan situs yang berupa kumpulan file-file html, php, gambar. Setelah selesai melakukan konfigurasi maka file-file tersebut akan di *upload* ke direktori pengguna yang ada di server.

c. Modul Pencacah kunjungan

Modul pencacah kunjungan digunakan untuk melakukan perhitungan jumlah kunjungan pada suatu *website*. Untuk memudahkan dalam penggunaan fasilitas pencacah kunjungan maka digunakan basis data yang menampung data jumlah kunjungan, tipe pencacah kunjungan, kategori pencacah kunjungan. Tipe pencacah kunjungan menentukan tampilan gambar angka pada *website*. Modul ini dapat digunakan untuk registrasi dan *request* untuk menampilkan pencacah kunjungan.

d. Modul Buku tamu

Modul ini digunakan untuk melakukan proses penyimpanan dan penampilan tamu yang mengisi buku tamu pada website pengguna. Pada modul ini digunakan basis data yang berfungsi untuk menyimpan data tamu yang mengisi buku tamu.

e. Modul Pengelola

Modul ini digunakan untuk melakukan pengaturan pencacah kunjungan dan buku tamu dari pengguna. Selain itu digunakan untuk mengatur seting dari aplikasi seperti koneksi basis data, seting web dan seting manajemen file.

4.5 Rancangan Basis Data

Proses perancangan basis data dimulai dengan pendefinisian entitas dan relasi sistem berdasarkan proses langganan. Pada aplikasi *webhosting* terdiri dari dua buah langganan, yaitu langganan pencacah kunjungan dan langganan buku tamu.

4.5.1 Pendefinisian Entitas dan relasi pada langganan pencacah kunjungan

Tabel 4. 1 Tabel Langganan Pencacah Buku Tamu

userid	Nama	nama_lengkap	alamat	kota	kodepos	...
1	sevens	Sevens Spirits	NG 5	Jogja	55283	...
6	clef	Clef IT Community	Nomaden	Jogja	55281	...
2	anno	-	-	-	-	...

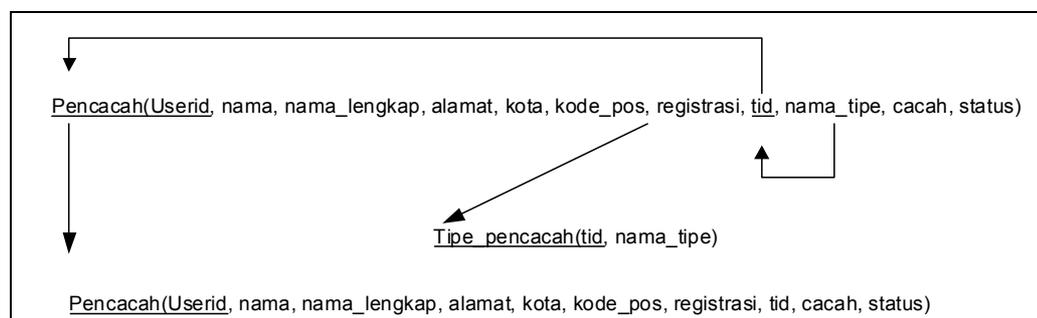
...	Registrasi	tid	Nama tipe	cacah	status
...	2002-01-26 02:20:29	3	roman	55	ya
...	2002-01-28 22:40:29	1	green	1	ya
...	2003-01-11 21:00:14	3	Roman	15	tidak

4.5.1.1 Bentuk Normal Pertama (1NF)

Pada tabel 4.1 sudah memenuhi bentuk normal pertama. Karena setiap atribut bernilai tunggal untuk setiap baris.

4.5.1.2 Bentuk Normal Kedua (2NF)

Tabel 4.1 memperlihatkan pada pada *nama_tipe* terjadi duplikasi data . Hal itu terjadi karena beberapa atribut pada tabel tersebut memiliki ketergantungan parsial.



Gambar 4. 34 Dekomposisi Pencacah Kunjungan Bentuk Normal Kedua (2NF)

Dari gambar 4.34 menunjukkan bahwa tabel 4.1 dipecah menjadi dua bagian yaitu tabel tipe_pencacah dan tabel pencacah.

Tabel 4. 2 Pencacah Bentuk Normal Kedua (2NF)

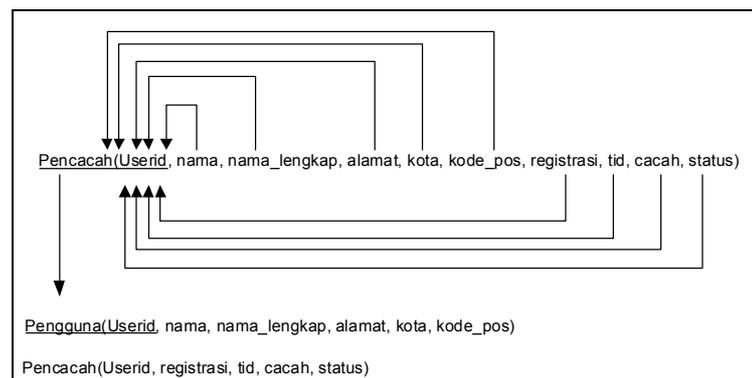
userid	nama	nama lengkap	alamat	kota	kodepos	Registrasi	tid	cacah	Status
1	sevens	Sevens Spirits	NG 5	Jogja	55283	2002-01-26 02:20:29	3	55	ya
6	clef	Clef IT Community	Nomaden	Jogja	55281	2002-01-28 22:40:29	1	1	ya
2	anno	-	-	-	-	2003-01-11 21:00:14	3	15	Tidak

Tabel 4. 3 Tipe_pencacah Bentuk Normal Kedua (2NF)

tid	Nama tipe
1	Green
3	Roman

4.5.1.3 Bentuk Normal Ketiga (3NF)

Tabel 4.2 tidak memenuhi bentuk normal ketiga karena memiliki ketergantungan yang tersembunyi. Pada tabel 4.2 akan terjadi masalah jika pengguna tidak berlangganan pencacah kunjungan. Pada tabel 4.2 tersebut tidak dapat menampilkan pengguna tersebut.



Gambar 4. 35 Dekomposisi Bentuk Normal Ketiga (3NF) Pencacah

Berdasarkan gambar 4.35 maka tabel 4.2 dipecah dua buah tabel baru. Tabel-tabel tersebut adalah tabel pengguna dan tabel pencacah.

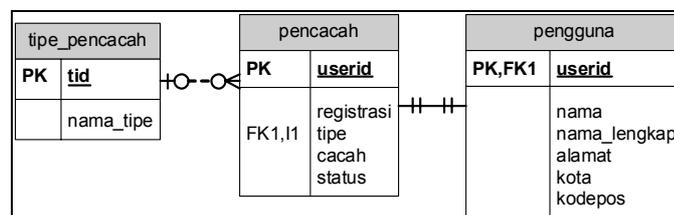
Tabel 4. 4 Tabel Pengguna Bentuk Normal Ketiga (3NF)

userid	nama	nama_lengkap	alamat	kota	Kodepos
1	sevens	Sevens Spirits	NG 5	Jogja	55283
6	clef	Clef IT Community	Nomaden	Jogja	55281
2	anno	-	-	-	-

Tabel 4. 5 Tabel Pencacah Bentuk Normal Ketiga (3NF)

Userid	Registrasi	tid	cacah	Status
1	2002-01-26 02:20:29	3	55	ya
6	2002-01-28 22:40:29	1	1	ya
2	2003-01-11 21:00:14	3	15	Tidak

Dari hasil pendekomposisian yang telah dilakukan mulai dari bentuk normal pertama sampai bentuk normal ketiga diperoleh relasi antar tabel sebagai berikut:



Gambar 4. 36 Relasi Antar Tabel Langgan Pencacah Kunjungan

4.5.2 Pendefinisian Entitas dan relasi pada transaksi buku tamu

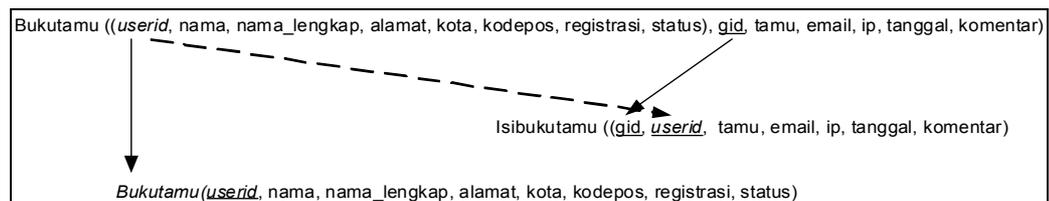
Tabel 4. 6 Tabel Langgan Buku Tamu Bentuk Normal Kedua (2NF)

userid	nama	nama_lengkap	alamat	Kota	kodepos	registrasi	...
1	sevens	Sevens Spirits	Jogja	Jogja	55283	2002-01-27 00:33:56	...
1	sevens	Sevens Spirits	Jogja	Jogja	55283	2002-01-27 00:33:56	...
1	sevens	Sevens Spirits	Jogja	Jogja	55283	2002-01-27 00:33:56	...
6	clef	Clef IT Community	Nomaden	Jogja	55281	2002-01-28 22:40:29	...
6	clef	Clef IT Community	Nomaden	Jogja	55281	2002-01-28 22:40:29	...

...	status	gid	tamu	email	ip	tanggal	Komentar
...	ya	2	Guest	guest@oke.com	192.168.1.4	2002-01-28 22:33:06	just visiting
...	ya	3	Aris	aris@gamanet.ugm.ac.id	192.168.1.3	2002-01-28 22:33:38	nice site
...	ya	4	Totok	prince@site.com	192.168.1.2	2002-01-28 22:34:29	okey
...	ya	5	okey	smart@okey.com	192.168.1.3	2002-01-28 22:41:06	yesss
...	ya	6	aris	aris@gamanet.ugm.ac.id	192.168.1.5	2002-01-28 22:41:17	testing aja

4.5.2.1 Bentuk Normal Pertama (1NF)

Tabel 4.6 menunjukkan adanya perulangan. Perulangan terjadi pada *userid*, *nama*, *nama_lengkap*, *alamat*, *kota*, *kodepos*, *registrasi*, *status*. Karena itu maka tabel perlu dipecah seperti ditunjukkan pada gambar 4.37.



Gambar 4. 37 Dekomposisi Buku Tamu Bentuk Normal Pertama (1NF)

Dari hasil pemecahan pada gambar 4.37 maka diperoleh hasil yang ditunjukkan pada tabel 4.7 dan tabel 4.8.

Tabel 4. 7 Tabel Buku Tamu Bentuk Normal Kedua (2NF)

userid	nama	nama_lengkap	alamat	kota	kodepos	registrasi	status
1	sevens	Sevens Spirits	Jogja	Jogja	55283	2002-01-27 00:33:56	ya
6	clef	Clef IT Community	Nomaden	Jogja	55281	2002-01-28 22:40:29	ya

Tabel 4. 8 Tabel IsiBukutamu Bentuk Normal Kedua (2NF)

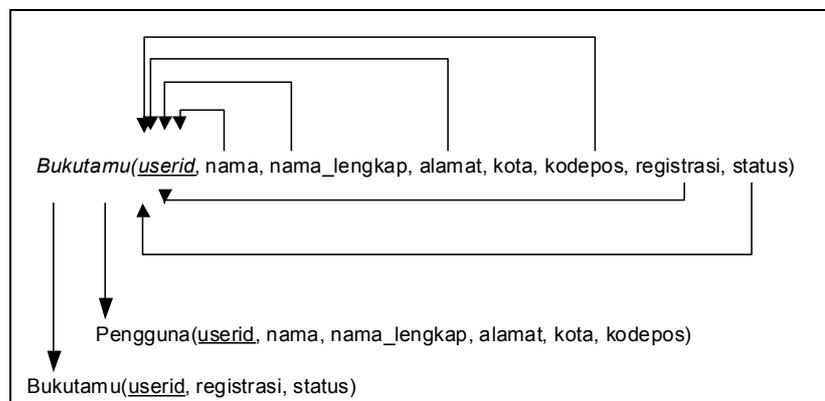
gid	userid	tamu	email	Ip	tanggal	Komentar
2	1	Guest	guest@oke.com	192.168.1.4	2002-01-28 22:33:06	just visiting
3	1	Aris	aris@gamanet.ugm.ac.id	192.168.1.3	2002-01-28 22:33:38	nice site
4	1	Totok	prince@site.com	192.168.1.2	2002-01-28 22:34:29	okey
5	6	okey	smart@okey.com	192.168.1.3	2002-01-28 22:41:06	yesss
6	6	aris	aris@gamanet.ugm.ac.id	192.168.1.5	2002-01-28 22:41:17	testing aja

4.5.2.2 Bentuk Normal Kedua (2NF)

Dari hasil dekomposisi sebelumnya diketahui bahwa tabel yang terbentuk sudah memenuhi bentuk normal kedua. Karena tidak ada kolom yang mempunyai ketergantungan parsial.

4.5.2.3 Bentuk Normal Ketiga (3NF)

Permasalahan akan muncul jika ternyata pengguna yang berlangganan buku tamu belum ada tamu yang memberikan komentarnya. Jika terjadi hal demikian maka pengguna tidak dapat ditampilkan. Hal tersebut dapat terjadi karena *nama*, *nama_lengkap*, *alamat*, *kota*, *kodepos* mempunyai ketergantungan tersembunyi terhadap *userid*. Gambar 4.38 akan menunjukkan hal tersebut.



Gambar 4. 38 Dekomposisi Bukutamu Bentuk Normal Ketiga (3NF)

Tabel yang dihasilkan dari proses dekomposisi terlihat pada tabel 4.9 dan tabel 4.10. Kedua tabel tersebut merupakan akhir dari proses normalisasi.

Tabel 4. 9 Pengguna Bentuk Normal Ketiga (3NF)

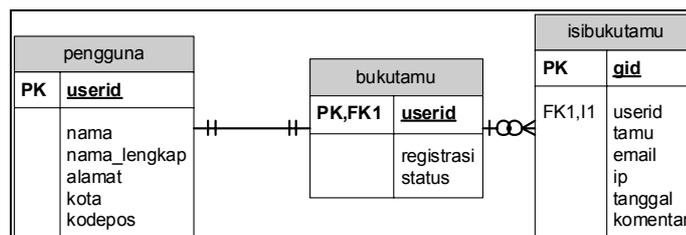
userid	nama	nama_lengkap	alamat	kota	kodepos
1	sevens	Sevens Spirits	Jogja	Jogja	55283
6	clef	Clef IT Community	Nomaden	Jogja	55281

Tabel 4. 10 Buku Tamu Bentuk Normal Ketiga (3NF)

userid	registrasi	status
1	2002-01-27 00:33:56	ya
6	2002-01-28 22:40:29	ya

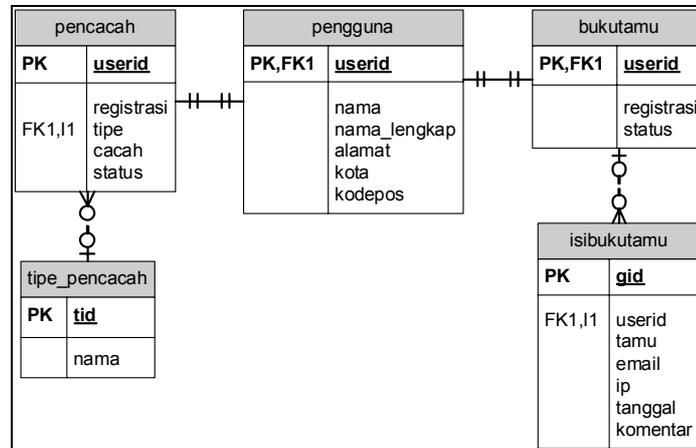
Dari hasil pendekomposisian tersebut diatas diperoleh relasi antar tabel.

Gambar 4.39 menunjukkan relasi tersebut.



Gambar 4. 39 Relasi Langgan Buku Tamu

Relasi pada gambar 4.36 dan gambar 4.39 dijadikan sebuah relasi utuh. Relasi tersebut digunakan untuk menyatukan dua buah formulir langganan. Relasi lengkapnya ditunjukkan pada gambar 4.40.



Gambar 4. 40 Relasi Lengkap Aplikasi *Webhosting*

4.6 Rancangan Pengujian Aplikasi

Pengujian aplikasi digunakan untuk menentukan tingkat kebutuhan aplikasi. Proses pengujian aplikasi akan dilakukan pada tiga kategori, yaitu komputer lokal, jaringan lokal, internet.

Sebelum dilakukan pengujian akan dicatat spesifikasi dari komputer yang digunakan beserta sistem operasi yang digunakan. Tabel 4.11 adalah spesifikasi komputer.

Tabel 4. 11 Spesifikasi Komputer

No.	Peripherals	Spesifikasi
1	Processor	...
2	VGA	...
3	Harddisk	...
4	Memori	...
5	Ethernet Card	...
6	Sistem Operasi	...
7	Web Browser	...

Untuk masing-masing komputer harus mempunyai *web browser*. Tiap komputer harus terhubung dengan jaringan lokal untuk pengujian kategori kedua. Untuk pengujian kategori ketiga, komputer yang digunakan harus terhubung dengan internet.

Hal-hal yang harus diperhatikan dalam pengujian aplikasi adalah:

1. Waktu yang diperlukan untuk menampilkan halaman utama.
2. Waktu yang diperlukan untuk menampilkan kembali (*refresh*) halaman utama.
3. Waktu yang diperlukan untuk login.
4. Waktu yang diperlukan untuk menampilkan halaman manajemen file.
5. Waktu yang diperlukan untuk menampilkan kembali(*refresh*) halaman manajemen file.
6. Waktu yang diperlukan untuk melakukan proses *upload* file.
7. Waktu yang diperlukan untuk melakukan proses *download* file.

Tabel 4. 12 Tabel Pengujian Aplikasi

titik uji	Komputer ...		Komputer ...	
	1	2	1	2
Index				
<i>Refresh</i> index				
Login				
Manajemen file				
<i>Refresh</i> manajemen file				
Upload(... MB)				
Download(... MB)				

BAB V

IMPLEMENTASI APLIKASI WEBHOSTING

Setelah tahap desain aplikasi webhosting dapat diselesaikan, tahap selanjutnya adalah implementasi desain sistem tersebut agar siap untuk dioperasikan. Pada tahap ini melibatkan perangkat lunak *Microsoft Internet Explorer, Konqueror, Mozilla, MySQL, HAPedit, Quanta+, Apache, PHP, PHPMyAdmin*.

Tahap implementasi ini dibagi menjadi beberapa bagian, yaitu Implementasi basis data dan pembuatan aplikasi *webhosting*. Untuk pembuatan aplikasi *webhosting* dibagi lagi menjadi beberapa bagian, yaitu login, menu utama, bantuan, langganan, manajemen file, otomatisasi halaman web dan administrasi pengelola.

5.1 Implementasi Basis Data

Aplikasi *webhosting* ini menggunakan MySQL sebagai *Database Management System(DBMS)*. Untuk mempermudah dalam melakukan konfigurasi basis data digunakan perangkat lunak bantuan yaitu *PHPMyAdmin*. Pembuatan tabel-tabel yang telah dirancang sebelumnya menggunakan perintah *Data Definition Language (DDL)*. Gambar 5.2 menampilkan perintah pembuatan tabel-tabel tersebut secara lengkap.

```

CREATE TABLE pengguna (
    userid bigint(20) DEFAULT '0' NOT NULL,
    nama varchar(25),
    nama_lengkap varchar(50),
    alamat varchar(100),
    kota varchar(30),
    kodepos int(5) DEFAULT '00000' NOT NULL,
    PRIMARY KEY (userid),
    UNIQUE nama (nama)
);

CREATE TABLE tipe_pencacah (
    tid int(4) DEFAULT '0' NOT NULL,
    nama varchar(20) NOT NULL,
    PRIMARY KEY (tid)
);

CREATE TABLE bukutamu (
    userid bigint(20) DEFAULT '0' NOT NULL
    REFERENCES pengguna(userid) ON UPDATE CASCADE,
    registrasi datetime DEFAULT '0000-00-00 00:00:00' NOT NULL,
    status enum('ya','tidak') DEFAULT 'ya' NOT NULL,
    PRIMARY KEY (userid)
);

CREATE TABLE isibukutamu (
    gid bigint(20) NOT NULL auto_increment,
    userid bigint(20) DEFAULT '0' NOT NULL
    REFERENCES pengguna(userid) ON UPDATE CASCADE,
    tamu varchar(25) NOT NULL,
    email varchar(50) NOT NULL,
    ip varchar(15) NOT NULL,
    tanggal datetime DEFAULT '0000-00-00 00:00:00' NOT NULL,
    komentar varchar(250) NOT NULL,
    PRIMARY KEY (gid)
);

CREATE TABLE pencacah (
    userid bigint(20) DEFAULT '0' NOT NULL,
    registrasi datetime DEFAULT '0000-00-00 00:00:00' NOT NULL,
    tipe tinyint(4) DEFAULT '1' NOT NULL
    REFERENCES tipe_pencacah ON UPDATE CASCADE,
    cacah bigint(20) DEFAULT '0' NOT NULL,
    status enum('ya','tidak') DEFAULT 'ya' NOT NULL,
    PRIMARY KEY (userid)
);

```

Gambar 5. 1 Perintah Pembuatan Tabel

5.2 Implementasi Aplikasi *Webhosting*

Aplikasi adalah sarana penghubung antara komputer dengan pengguna. Untuk membuat sebuah aplikasi dibutuhkan satu atau lebih perangkat lunak. Aplikasi webhosting ini menggunakan lebih dari satu perangkat lunak. *Microsoft Internet Explorer* (windows), *Konqueror*, *Mozilla* (linux) adalah *web browser* yang digunakan untuk menjalankan aplikasi. *HAPedit* (windows), *Quanta+* (linux) merupakan editor untuk membuat aplikasi. *Apache* sebagai *web server*. *PHP* (*PHP Hypertext Preprocessor*) berfungsi sebagai *compiler*.

Aplikasi webhosting ini dapat digunakan untuk lintas platform, artinya bahwa aplikasi dapat dipasang pada hampir semua sistem operasi yang mempunyai fasilitas *webservice* (menggunakan *apache*), *database server* (MySQL), *ftp server*, dan dapat *multi-user*. Selain itu aplikasi ini dapat diakses dari mana saja dan menggunakan sistem operasi apa saja dengan syarat *server* yang digunakan terhubung dengan internet dan sistem operasi tersebut mempunyai *web browser*.

Aplikasi ini dibagi menjadi dua bagian, yaitu bagian administrasi pengguna dan bagian administrasi pengelola. Cara mengakses kedua bagian tersebut berlainan. Alamat yang digunakan untuk mengakses bagian administrasi pengguna adalah <http://gamanet.ugm.ac.id> dan alamat untuk mengakses bagian administrasi pengelola adalah <http://gamanet.ugm.ac.id/admin.php>.

5.2.1 Administrasi Pengguna

5.2.1.1 Login

Proses pertama yang harus dilakukan agar dapat mengakses administrasi pengguna adalah melakukan login. Proses login ini berhubungan dengan server ftp, karena autentikasi pengguna menggunakan koneksi ftp. Potongan kode program untuk login ditunjukkan pada gambar 5.3.

```

$ftp_conn= @ftp_connect($ftp_server);
$login = @ftp_login($ftp_conn, $uid, $pwd);
if ($login)
{
    $connect = dbconnect($db_host, $db_user, $db_pass, $db_name);
    $sql = "select * from pengguna where nama='$uid'";
    $sql_res = doQuery($sql,$connect);
    $sql_row = getRow($sql_res);
    if ($sql_row < 1)
    {
        $sql = "select max(userid) from pengguna";
        $sql_res = doQuery($sql,$connect);
        $sql_data = getRecord($sql_res);
        $userid = $sql_data[0] + 1;
        $sql = "insert into pengguna(userid, nama) "
            ."values('$userid','$uid)";
        $sql_res = doQuery($sql, $connect);
    }
    else
    {
        $sql_data = getRecord($sql_res);
        $userid = $sql_data[0];
    }
    session_register("sesi_ftp_login");
    .
    .
    $sesi_ftp_login=1;
    .
    .
    $currentDir = @ftp_pwd($ftp_conn);
}
else
{
    message(_login_failed,_login,1);
}

```

Gambar 5. 2 Kode Program Proses Login

Kode program pada gambar 5.3 akan melakukan koneksi ke *ftp server* (`$ftp_conn= @ftp_connect($ftp_server)`) menggunakan *username* dan *password* yang telah pengguna masukkan (`$login = @ftp_login($ftp_conn, $uid, $pwd)`). Jika menghasilkan nilai *false* maka login pengguna dinyatakan gagal. Jika nilai kembalian *true*, maka dilanjutkan pengecekan pengguna. Aplikasi akan mendaftarkan secara otomatis pengguna ke dalam basis data jika pengguna baru pertama kali menggunakan aplikasi *webhosting* ini. Setelah pengecekan pengguna maka dilakukan registrasi sesi menggunakan perintah `session_register("sesi_ftp_login")`. Hasil dari registrasi sesi ini akan disimpan di server. Sesi tersebut digunakan untuk melakukan proses-proses selanjutnya. Akhir dari proses login adalah menampilkan menu utam dari administrasi pengguna.



Gambar 5. 3 Tampilan Login

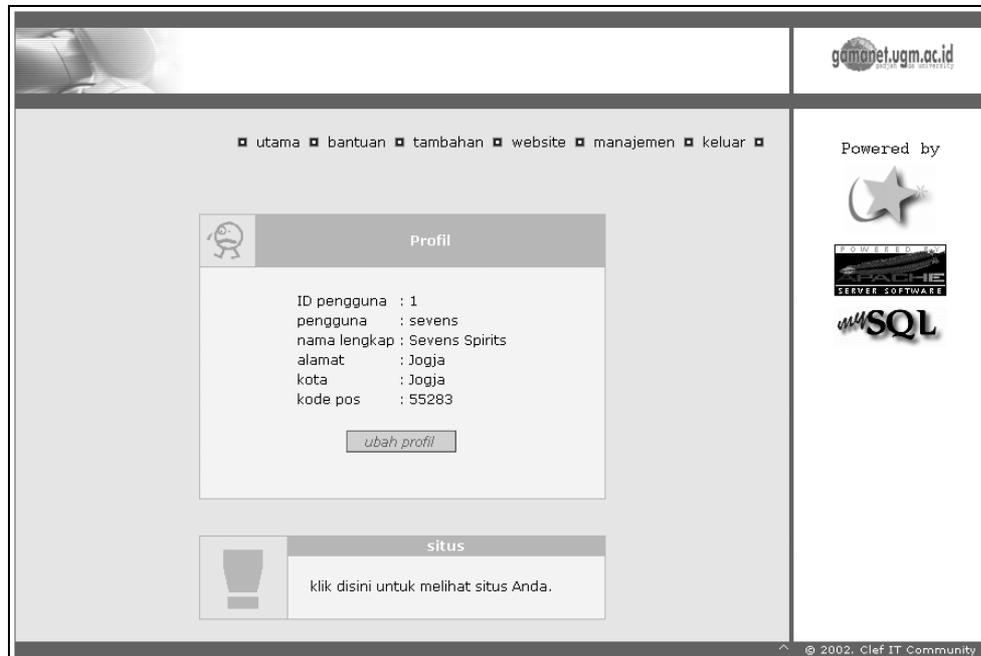
5.2.1.2 Menu Utama

Menu utama adalah tampilan yang pertama kali dilihat oleh pengguna setelah berhasil melakukan login. Kode program untuk menampilkan menu utama ditunjukkan dengan gambar 5.5.

```
function mainmenu()
{
    global $ftp_conn, $waktu, $public_dir, $currentDir,
        $sesi_ftp_username;
    //$ftp_conn = ftp_connect($ftp_server);
    showProfile();
    $public = $currentDir;
    $public .= $public_dir;
    if (@ftp_chdir($ftp_conn, $public))
    {
        echo "<br><br>";
        message("<a href=\""$web_server/~$sesi_ftp_username\""
            target=\""_blank\"">"._click_visit_web."</a>",_site);
    }
    else
    {
        echo "<br><br>";
        message(_didnt_configure."<br>
            <a href=\""?go=genpage&ctime=$waktu\"">
                ._click_create_web."</a>",_site);
    }
}
}
```

Gambar 5.4 Kode Program Menu Utama

Pada menu utama akan ditampilkan profil dari pengguna dengan fungsi `showProfile()`. Selain menampilkan profil dari pengguna, aplikasi juga akan melakukan deteksi situs dari pengguna. Untuk mendeteksi digunakan perintah `ftp_chdir($ftp_conn, $public)`. Jika menghasilkan nilai `true` maka ditampilkan link untuk menuju ke situs pengguna, tetapi jika pengguna belum mempunyai situs maka akan ditampilkan pesan bahwa pengguna belum mempunyai situs.



Gambar 5. 5 Tampilan Menu Utama

5.2.1.3 Bantuan

Bantuan akan membantu pengguna dalam menggunakan aplikasi ini. Bantuan yang ada tersedia dalam dua versi bahasa, yaitu bahasa indonesia dan bahasa inggris. Di dalam bantuan ini dijelaskan cara membuat situs, fasilitas yang dapat digunakan oleh pengguna dan beberapa aturan penggunaan aplikasi.



Gambar 5. 6 Tampilan Bantuan

5.2.1.4 Langganan

Menu Langganan mempunyai dua sub menu yaitu pencacah kunjungan dan buku tamu. Pada masing-masing menu, pengguna dapat berlangganan fasilitas tersebut untuk dipasang pada situs masing-masing. Pada sub menu tersebut akan dilakukan deteksi langganan dari pengguna. Jika pengguna belum berlangganan maka akan ditampilkan *link* untuk melakukan langganan.

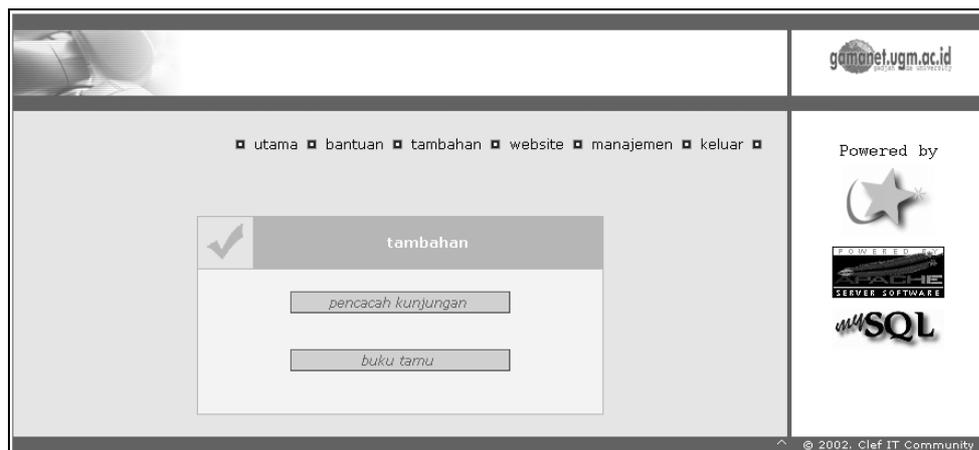
```

$sql = "select * from pencacah where "
      . "userid=\""$sesi_ftp_userid\"";
$sql_res = doQuery($sql,$connect);
$sql_row = getRow($sql_res);
if ($sql_row==1)
{
    $msg = _have_been_registered."<br>"
      . "<a href=\""?go=counter&request=change "
      . "&ctime=$waktu\">"
      . _click_change_counter."</a>";
    message($msg,_counter);
}
elseif ($sql_row<1)
{
    countRegForm("new");
}

```

Gambar 5. 7 Kode Program Pencacah Kunjungan

Kode program pada gambar 5.7 akan melakukan pengecekan terhadap langganan pencacah kunjungan. Jika pengguna sudah berlangganan maka pengguna dapat melakukan perubahan konfigurasi, tetapi jika pengguna belum terdaftar maka akan ditampilkan form registrasi pencacah kunjungan menggunakan perintah `countRegForm("new")`.



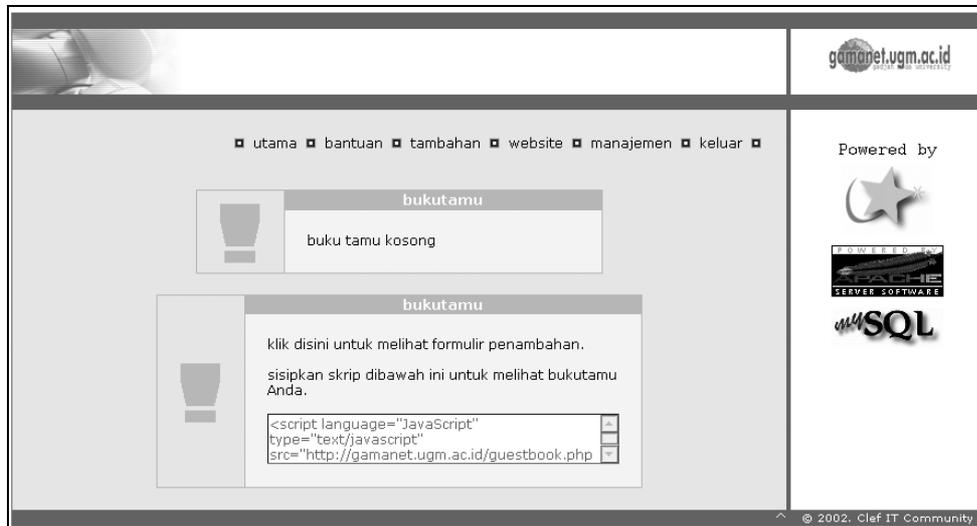
Gambar 5. 8 Tampilan Langganan Pencacah Kunjungan

Pada sub menu buku tamu akan dideteksi juga status dari pengguna. Jika belum berlangganan maka secara otomatis pengguna akan didaftarkan ke dalam basis data, jika sudah berlangganan pengguna dapat menghapus daftar pengunjung yang tidak diinginkan.

```
function gbregForm()
{
    global $connect, $sesi_ftp_userid, $web_server;
    $sql = "select * from bukutamu where "
        ."userid=\"$sesi_ftp_userid\"";
    $sql_res = doQuery($sql,$connect);
    $sql_rows = getRow($sql_res);
    if ($sql_rows < 1)
    {
        $sql = "insert into bukutamu(userid,registrasi,status) "
            ."values('$sesi_ftp_userid', now(''),'ya')";
        $sql_res = doQuery($sql, $connect);
    }
    elseif ($sql_rows==1)
    {
        $sql_data = getRecord($sql_res);
        $uid = $sql_data[0];
        listGB();
    }
    $msg = "<a href='guestbook.php?' "
        ."uname=$sesi_ftp_userid&act=sample'"
        ." target='_blank'>
        .
        .
        message($msg, _guestbook);
}
```

Gambar 5.9 Kode Program Langganan Buku Tamu

Kode program yang ditunjukkan oleh gambar 5.9 akan melakukan pengecekan langganan buku tamu dari pengguna. Setelah dilakukan pengecekan dan ternyata belum berlangganan maka akan didaftarkan secara langsung. Setelah pendaftaran maka akan ditampilkan cara untuk memasang skrip buku tamu pada situs pengguna. Jika pengguna sudah berlangganan akan ditampilkan daftar pengunjung yang memberi komentar pada situs pengguna (`listGB()`).



Gambar 5. 10 Tampilan Registrasi Buku Tamu

5.2.1.5 Otomasi Halaman Web

Pada menu *website* pengguna dapat membuat situs secara cepat. Cepat disini diartikan bahwa struktur dasar dari situs sudah dibuat hanya kurang data-data dari pengguna. Menu ini mempunyai enam bagian yang terintegrasi dalam *wizard* (berurutan).

Tampilan pada gambar 5.11 meminta pengguna untuk mengisikan data bagian kepala, bagian kaki dan deskripsi dari situs. Pengguna juga diharuskan mengirimkan gambar logo ke server. Data bagian kepala akan selalu ditampilkan setiap halaman situs pada bagian atas. Untuk data bagian kaki akan selalu ditampilkan setiap halaman pada bagian bawah. Logo akan ditampilkan pada bagian atas kiri di setiap halaman situs.

The screenshot shows a web application interface with a navigation menu at the top: [utama](#) [bantuan](#) [tambahan](#) [website](#) [manajemen](#) [keluar](#). The main content area is divided into two sections:

LANGKAH 1 - PENGATURAN SITUS

pengaturan situs

Logo:

bagian kepala

bagian kaki

deskripsi situs (muncul di halaman utama)

PRE LANGKAH 2 - PEMBUATAN MENU

Pre - pembuatan menu

jumlah menu

Powered by

SERVER SOFTWARE

© 2002. Clef IT Community

Gambar 5. 11 Tampilan 1 Otomasi Halaman Web

Untuk menentukan pembuatan menu yang akan dibuat pada tampilan 2 diperlukan jumlah menu yang akan dibuat. Pengguna akan diminta untuk menentukan jumlah menu yang akan dibuat. Aplikasi menyediakan pilihan pembuatan menu dari satu sampai sepuluh menu yang dapat dibuat. Pembatasan pembuatan menu dilakukan karena mempertimbangkan ukuran file konfigurasi yang akan dibuat dan kecepatan akses terhadap file.

```

$next_step=true;
.
.
check_entry($web_header, "header");
if ($next_step==false)
{
    show_page("1");
    break;
}
else
{
    if (file_exists($logo_pic))
    {
        $dirNow = @ftp_pwd($ftp_conn);
        if (@ftp_chdir($ftp_conn,"$dirNow.$public_dir"))
        {
            $dirNow = @ftp_pwd($ftp_conn);
            @ftp_put($ftp_conn, $dirNow .
                "/web_logo.gif", $logo_pic, 1);
        }
        else
        {
            @ftp_mkdir($ftp_conn,"$dirNow.$public_dir");
            $dirNow = @ftp_pwd($ftp_conn);
            $dirNow .= $public_dir;
            @ftp_put($ftp_conn, $dirNow .
                "/web_logo.gif", $logo_pic, 1);
        }
    }
    @unlink($logo_pic);

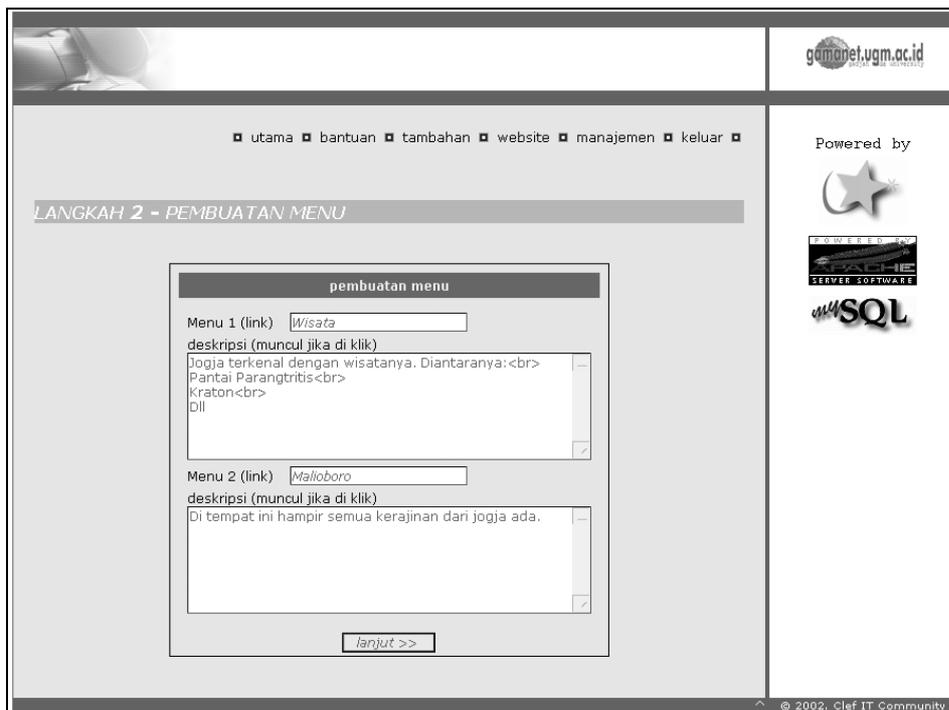
    $retval=chdir("tmp");
    $ffile=fopen($sesi_ftp_username."_setting.php","w+");
    fputs($ffile, "<?php\n");
    fputs($ffile, "//Generated by Sevens Spirits. \n");
    .
    .
    fclose($ffile);
    $retval=chdir("../");
    show_page(2, $sum_menu);
}

```

Gambar 5. 12 Kode Program Konfigurasi Tampilan 1 Otomasi Halaman Web

Data yang dimasukkan akan disimpan kedalam file pengguna_setting.php. data yang disimpan adalah *userid*, *username*, *web_server*, *ftp_server*, *header*, *footer*,

description. Selain itu juga dilakukan pengiriman logo ke direktori pengguna dengan perintah `ftp_put($ftp_conn, $dirNow . "/web_logo.gif", $logo_pic, 1)`.



Gambar 5. 13 Tampilan 2 Otomasi Halaman Web

Gambar 5.13 akan menampilkan konfigurasi menu dengan jumlah yang telah ditentukan sebelumnya. Pada masing-masing menu terdapat dua isian. Isian pertama meminta pengguna untuk mengisi nama *link*. Isian kedua adalah deskripsi. Deskripsi akan ditampilkan pada situs jika pengunjung memilih link yang sesuai.

Seperti langkah sebelumnya, data pada gambar 5.13 akan disimpan ke dalam file `pengguna_setting.php`. Gambar 5.14 adalah kode program untuk menyimpan data diatas.

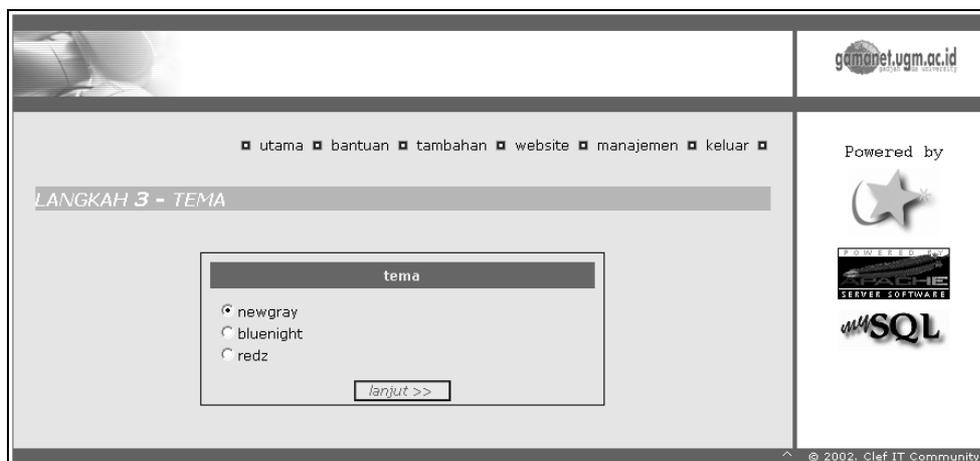
```

$retval=chdir("tmp");
$ffile=fopen($sesi_ftp_username."_setting.php","a+");
for ($i=1;$i<=$smenu;$i++)
{
    $tmenu="tmenu".$i;
    $txtmenu="txtmenu".$i;
    fputs($ffile, "\$link_menu$i=\"${$tmenu}\";\n");
    fputs($ffile, "\$text_menu$i=\"${$txtmenu}\";\n\n");
}
fclose($ffile);
$retval=chdir("../");
show_page("3");

```

Gambar 5. 14 Kode Program Konfigurasi Tampilan 2 Otomasi Halaman Web

Langkah kedua ini akan menyimpan data menu yang telah dibuat. Data disimpan berurutan dari data pertama sampai data terakhir. Setelah penyimpanan selesai maka akan ditampilkan langkah ketiga(`show_page("3")`).



Gambar 5. 15 Tampilan 3 Otomasi Halaman Web

Langkah ini mengharuskan pengguna memilih tema dari situs yang akan dibuat. Tema ini akan menentukan tampilan dari situs. Tiap tema mempunyai karakter berbeda-beda. Kode untuk konfigurasi tema sebagai ditunjukkan pada gambar 5.16.

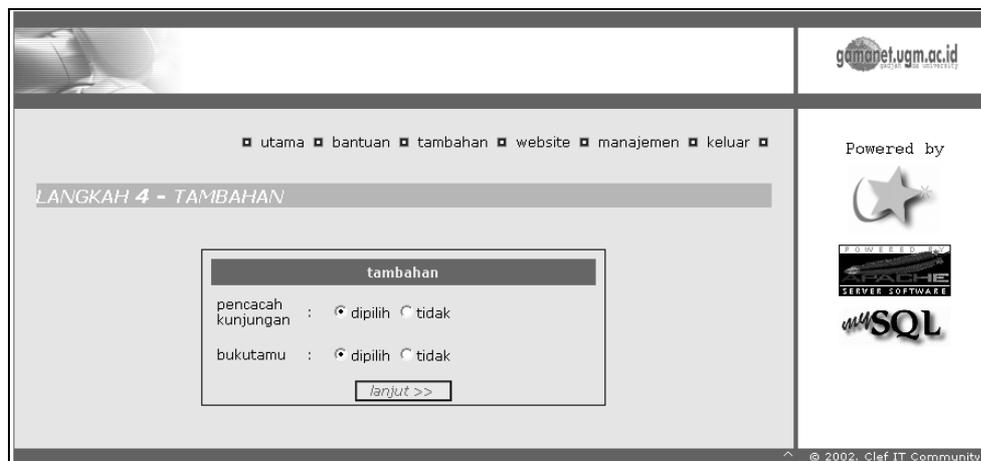
```

if (empty($themes))
    show_page("3");
else
{
    session_register("sesi_user_themes");
    $sesi_user_themes=$themes;
    show_page("4");
}

```

Gambar 5. 16 Kode Program Konfigurasi Tampilan 3 Otomasi Halaman Web

Jika pengguna tidak memilih tema maka akan ditampilkan lagi form pemilihan tema, sedangkan jika pengguna memilih salah satu tema maka aplikasi akan menandai tema yang dipilih dengan membuat sesi (`session_register("sesi_user_themes")`). Langkah selanjutnya adalah menampilkan form keempat.



Gambar 5. 17 Tampilan 4 Otomasi Halaman Web

Tampilan pada gambar 5.17 mengharuskan pengguna untuk memilih ya atau tidak untuk tiap jenis tambahan. Setelah dipilih maka akan dikonfigurasi ke dalam aplikasi.

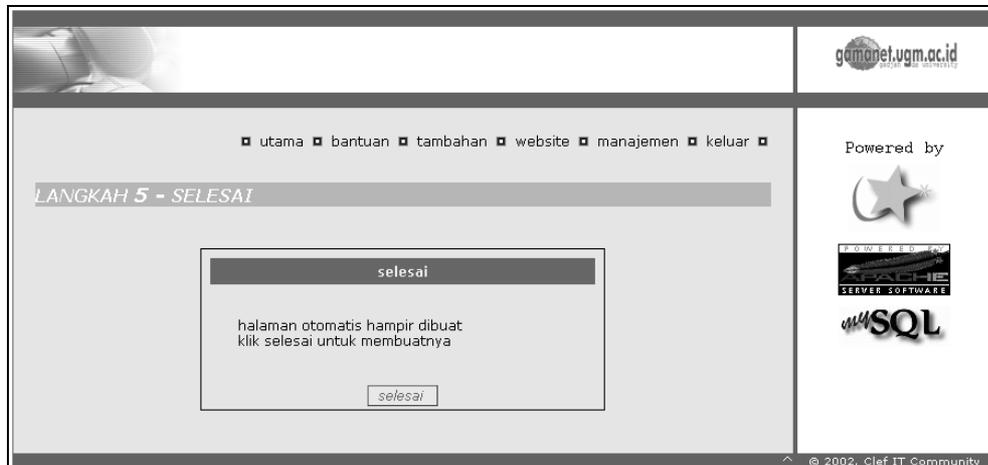
```

if ($fcounter==1)
{
    $sql = "select * from pencacah where userid='$sesi_ftp_userid'";
    $sql_res = doQuery($sql, $connect);
    $sql_row = getRow($sql_res);
    if ($sql_row < 1)
    {
        $sql = "insert into pencacah "
            ."values('$sesi_ftp_userid',now(''), 1, 0, 'ya')";
        $sql_res = doQuery($sql, $connect);
    }
}
else
    $fcounter = 0;
if ($fguestbook==1)
{
    $sql = "select * from bukutamu where userid='$sesi_ftp_userid'";
    $sql_res = doQuery($sql, $connect);
    $sql_row = getRow($sql_res);
    if ($sql_row < 1)
    {
        $sql = "insert into bukutamu "
            ."values('$sesi_ftp_userid',now(''), 'ya')";
        $sql_res = doQuery($sql, $connect);
    }
}
else
    $fguestbook = 0;
$retval=chdir("tmp");
$file=fopen($sesi_ftp_username."_setting.php","a+");
fputs($file, "// add Ons\n");
fputs($file, "\$counter=\"\$fcounter\";\n");
fputs($file, "\$guestbook=\"\$fguestbook\";\n");
fputs($file, "?>\n");
fclose($file);
$retval=chdir("../");
show_page("5");

```

Gambar 5. 18 Kode Program Konfigurasi Tampilan 4 Otomasi Halaman Web

Kode yang ditunjukkan pada gambar 5.18 akan mendaftarkan secara langsung masing-masing langganan jika pengguna belum melakukan langganan. Setelah dilakukan pendaftaran maka diteruskan dengan menyimpan konfigurasi dari pencacah kunjungan dan buku tamu ke dalam file pengguna_setting.php. kemudian akan ditampilkan form yang terakhir yaitu pembuatan situs.



Gambar 5. 19 Tampilan 5 Otomasi Halaman Web

Pada bagian terakhir ini aplikasi akan melakukan pembuatan situs, pengiriman file ke direktori pengguna, dan konfigurasi direktori. Untuk pembuatan situs akan dilakukan pengecekan direktori untuk meletakkan file-file konfigurasi menggunakan perintah `ftp_chdir($ftp_conn, "$public")`. Jika pengguna belum memiliki direktori tersebut maka akan dibuatkan direktori menggunakan perintah `ftp_mkdir($ftp_conn, "$public")`. Potongan kode program konfigurasi terakhir untuk proses otomasi halaman web ditunjukkan pada gambar 5.20.

```

$current = ftp_pwd($ftp_conn);
ftp_site($ftp_conn, "chmod 755 \"$current\"");
if (empty($sesi_user_themes))
{
    $sesi_user_themes="newgray";
}
$direct = opendir("template/$sesi_user_themes");
while ($upload=readdir($direct))
{
    if (($upload==".") || ($upload==".."))
    {
        continue;
    }
    ftp_put($ftp_conn, $current . "/" . $upload,
        "template/" . $sesi_user_themes . "/" . $upload, 1);
}
ftp_put($ftp_conn, $current . "/setting.php",
"tmp/" . $sesi_ftp_username . "_setting.php", 1);
ftp_site($ftp_conn, "chmod 777 \"$current/setting.php\"");
ftp_chdir($ftp_conn, "..");
$current = ftp_pwd($ftp_conn);
ftp_site($ftp_conn, "chmod 755 \"$current\"");
.
.
@unlink("tmp/" . $sesi_ftp_username . "_setting.php");
session_unregister("sesi_user_themes");
.
.
unset($sesi_counter);
.
.
show_page("6");

```

Gambar 5. 20 Kode Program Konfigurasi Tampilan 5 Otomasi Halaman Web

Pada direktori yang telah ditentukan akan disalin file-file web dari direktori aplikasi menggunakan perintah `ftp_put($ftp_conn, $current . "/setting.php", "tmp/" . $sesi_ftp_username . "_setting.php", 1)`. Jika semua file sudah dapat disalin maka dilakukan setting ijin direktori agar dapat dilihat oleh pengunjung situs (`ftp_site($ftp_conn, "chmod 755 \"$current\"")`).

Langkah terakhir adalah membersihkan aplikasi dari data-data yang telah dibuat oleh pengguna (`session_unregister("sesi_user_themes")`).

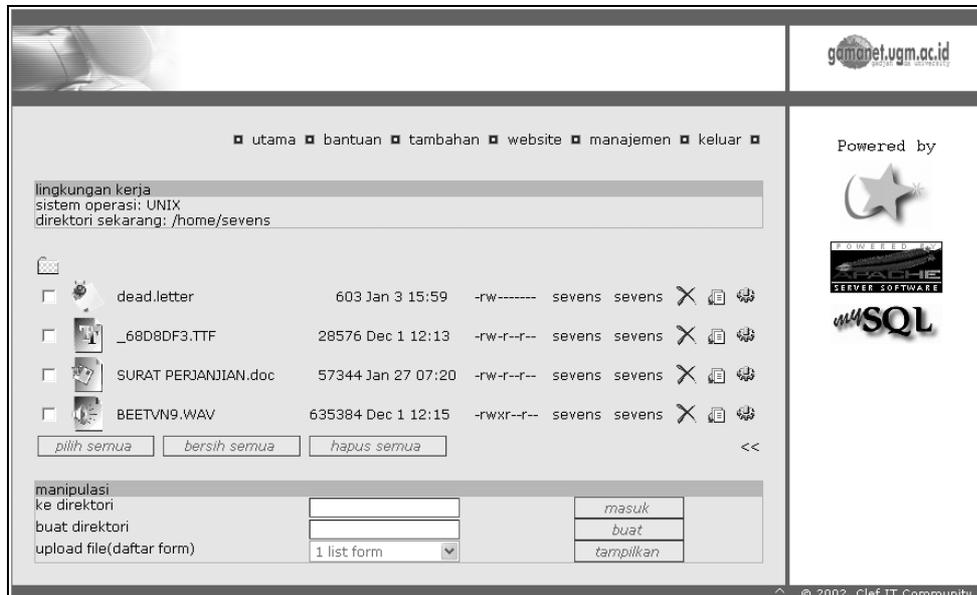
Hasil dari otomasi halaman web ini akan dapat dilihat pada situs pengguna dengan alamat `http://gamanet.ugm.ac.id/~nama_pengguna`. Gambar 5.21 adalah tampilan hasil pembuatan otomasi halaman web.



Gambar 5. 21 Tampilan Hasil Otomasi Halaman Web

5.2.1.6 Manajemen File

Manajemen file digunakan untuk memanipulasi file dan direktori yang ada di direktori pengguna. Manipulasi tersebut meliputi: membuat direktori, *rename*, *delete*, *chmod*. Proses lain yang terdapat dalam manajemen file adalah *upload* dan *download*.



Gambar 5. 22 Tampilan Manajemen File

Untuk dapat menampilkan seperti pada gambar 5.22 diperlukan sebuah fungsi yang dapat melakukan *generate*. Fungsi tersebut adalah fungsi `filemanager()`. Fungsi ini mengambil data dari server dengan fungsi `ftp_rawlist()` dan memecah-mecah data yang diperoleh ke dalam array menggunakan fungsi `parse_ftp_rawlist()`.

5.2.1.6.1 Rename

Proses *rename* digunakan untuk mengubah nama file atau direktori. Pada proses ini isi dari file tersebut tidak akan mengalami perubahan.



Gambar 5. 23 Dialog *Rename File*

Tampilan yang ditunjukkan oleh gambar 5.23 meminta pengguna untuk memberikan nama baru. Jika nama baru tidak diberikan maka file tersebut akan di *rename* ke dalam nama yang sama. Tombol cancel digunakan untuk membatalkan proses *rename*.

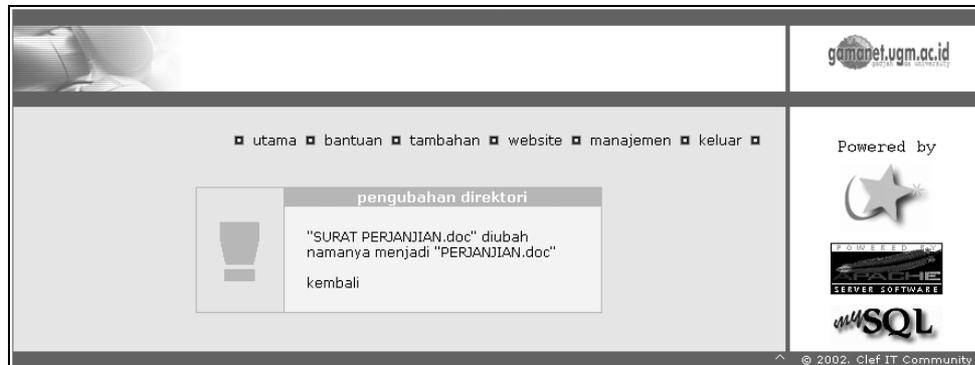
```

if(@ftp_rename($ftp_conn, "$file", "$file2"))
{
    $msg = "\"$file\" \"._rename_to.\" \"$file2\"";
}
else
{
    $msg = _couldn_rename.\" \"$file\" \"._to.\" \"$file2\"";
}
message($msg, rename_directory,2);

```

Gambar 5. 24 Kode Program Proses *Rename File*

Kode pada gambar 5.24 akan mengubah nama file atau direktori yang lama menjadi nama yang baru dengan perintah `ftp_rename($ftp_conn, "$file", "$file2")`. Jika tidak dapat diubah maka akan ditampilkan pesan tidak dapat diubah sedangkan jika file atau direktori dapat diubah maka akan ditampilkan pesan bahwa file dapat diubah (`message($msg, _rename_directory,2)`). Tampilan pesan tersebut sebagai berikut:



Gambar 5. 25 Tampilan Proses *Rename File*

5.2.1.6.2 Delete

Proses penghapusan file atau direktori ini dibagi menjadi dua bagian. Pertama menghapus file atau direktori satu-persatu. Dan yang kedua menghapus file atau direktori lebih dari satu sekaligus.



Gambar 5. 26 Tampilan Konfirmasi Hapus File

Proses penghapusan file satu-persatu akan dibedakan menjadi dua proses, yaitu penghapusan direktori dan penghapusan file. Kedua proses ini mempunyai tampilan yang sama tetapi proses yang dilakukan berbeda. Tampilan di atas meminta pengguna untuk menekan tombol *OK* atau *Cancel*. Jika tombol *OK* ditekan maka

proses penghapusan file akan dilakukan. Tetapi jika pengguna menekan tombol *cancel* maka proses penghapusan file tidak jadi dilakukan.

```
if(@ftp_delete($ftp_conn, "$file"))
{
    $msg = "$file "._deleted;
}
else
{
    $msg = _unable_deleted." $file";
}
message($msg, delete_file,2);
```

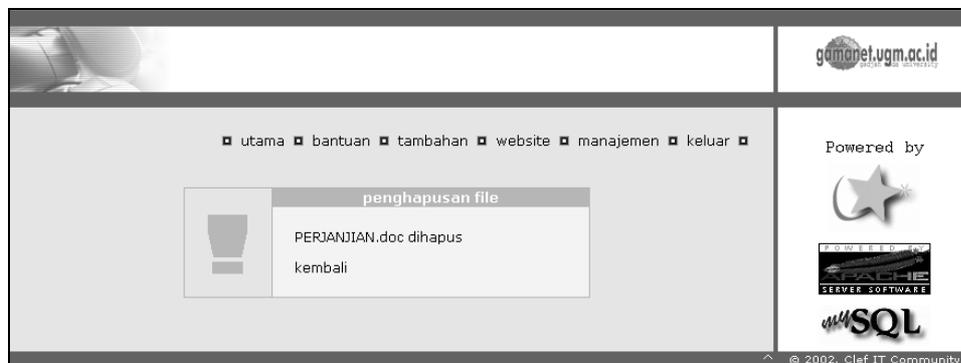
Gambar 5. 27 Kode Program Penghapusan File

Kode program yang ditunjukkan oleh gambar 5.27 akan melakukan penghapusan file dengan perintah `ftp_delete($ftp_conn, "$file")`. Perintah tersebut hanya dapat dilakukan untuk menghapus file saja. Perintah diatas akan memberikan nilai balik *true* atau *false*.

```
If(@ftp_rmdir($ftp_conn, "$file"))
{
    $msg = "$file "._deleted;
}
else
{
    $msg = _unable_deleted." $file";
}
message($msg, delete_directory,2);
```

Gambar 5. 28 Kode Program Penghapusan Direktori

Perintah `ftp_rmdir($ftp_conn, "$file")` digunakan untuk menghapus direktori. Perintah ini juga akan memberikan nilai balik *true* atau *false*.



Gambar 5. 29 Tampilan Hasil Penghapusan File

Penghapusan file lebih dari satu memerlukan daftar file-file yang akan dihapus. Untuk memilih semua file pengguna dapat menggunakan tombol pilih semua. Setelah file atau direktori yang akan dihapus dipilih, proses selanjutnya adalah menghapus pilihan dengan tombol hapus semua.

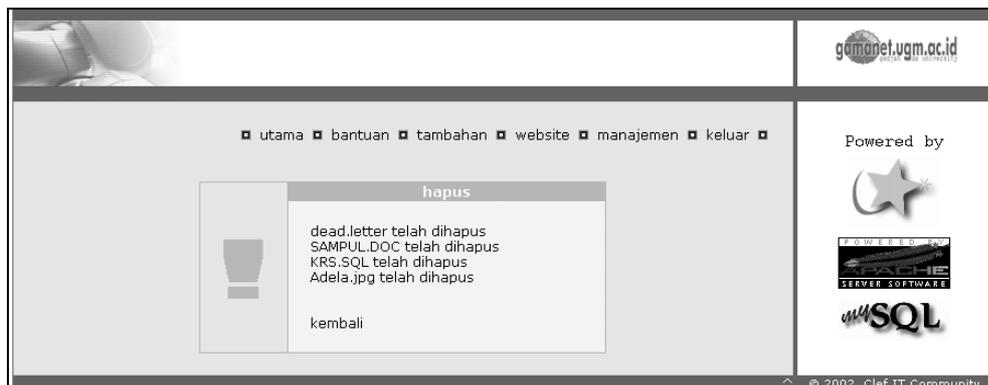
```

for ($i=1;$i<=$total;$i++)
{
    $tmp="dir".$i;
    if (isset($$tmp))
    {
        if (@ftp_rmdir($ftp_conn, "${$tmp}"))
            $msg.=${$tmp}." "._have_been_delete."<br>";
        else
            $msg.=${$tmp}." "._unable_deleted."<br>";
        continue;
    }
    .
    .
    $tmp="file".$i;
    if (isset($$tmp))
    {
        if (@ftp_delete($ftp_conn, "${$tmp}"))
            $msg.=${$tmp}." "._have_been_delete."<br>";
        else
            $msg.=${$tmp}." "._unable_deleted."<br>";
        continue;
    }
}

```

Gambar 5. 30 Kode Program Penghapusan File Lebih dari Satu

Kode yang ditunjukkan pada gambar 5.30 akan melakukan perulangan dari 0 sampai jumlah daftar file atau direktori. File atau direktori yang akan dihapus dibagi menjadi tiga kategori yaitu file, direktori, link. Untuk tiap kategori mempunyai penanganan yang berbeda. Untuk menghapus direktori dan link menggunakan fungsi `ftp_rmdir($ftp_conn, "${$tmp}")`. Proses menghapus file digunakan perintah `ftp_delete($ftp_conn, "${$tmp}")`. Tampilan hasil penghapusan lebih dari satu ditunjukkan pada gambar 5.31.



Gambar 5. 31 Tampilan Hasil Penghapusan

5.2.1.6.3 Chmod

Chmod adalah proses pengubahan ijin untuk file dan direktori. Proses ini akan menampilkan halaman baru dimana pengguna dapat mengubah konfigurasi ijin untuk file atau direktori tersebut.

pemilik	grup	lain
<input checked="" type="checkbox"/> baca	<input checked="" type="checkbox"/> baca	<input checked="" type="checkbox"/> baca
<input checked="" type="checkbox"/> tulis	<input type="checkbox"/> tulis	<input type="checkbox"/> tulis
<input type="checkbox"/> eksekusi	<input type="checkbox"/> eksekusi	<input type="checkbox"/> eksekusi
<input type="button" value="set ijin"/>		

Gambar 5. 32 Tampilan Konfigurasi Ijin

Pada proses pengubahan ijin file disediakan sembilan isian. Tiga isian untuk pemilik file(baca, tulis, eksekusi), tiga untuk grup(baca, tulis, eksekusi) dan tiga untuk lain (baca, tulis, eksekusi).

```

if(@ftp_site($ftp_conn, "chmod $permissions \"$file\"")
{
    $msg= _permission_for." $file "._changed_succeed;
}
else
{
    $msg= _coudn_change_for . $file;
}
message($msg, _permission,2);

```

Gambar 5. 33 Kode Program Ijin File

Kode program pada gambar 5.33 akan mengubah ijin dari file atau direktori yang dipilih sesuai dengan masukkan yang diberikan oleh pengguna. Perintah yang digunakan untuk mengubah ijin tersebut adalah `ftp_site($ftp_conn, "chmod $permissions \"$file\"")`.

5.2.1.6.4 Membuat Direktori

Direktori sering kali diperlukan untuk mengelompokkan file-file tertentu. Fasilitas pembuatan direktori selalu muncul pada manajemen file.

```

if(@ftp_mkdir($ftp_conn, "$dirname"))
{
    $msg = "$dirname "._created;
}
else
{
    $msg = _coudn_created." $dirname";
}
message($msg, create_directory,2);

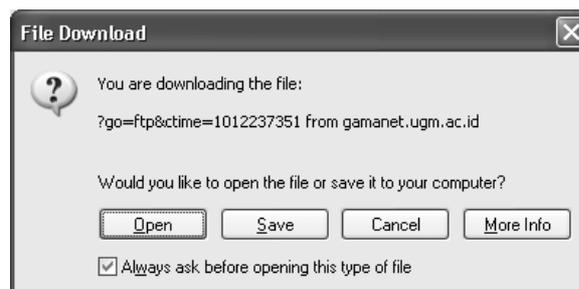
```

Gambar 5. 34 Kode Program Membuat Direktori

Pembuatan direktori memerlukan masukan berupa nama direktori yang akan dibuat. Perintah untuk membuat direktori adalah `ftp_mkdir($ftp_conn, "$dirname")`. Perintah tersebut akan memberikan nilai balik *true* atau *false*.

5.2.1.6.5 Download

Proses download akan melakukan penyalinan file dari server ke direktori pengguna di komputer lokal. Pengguna akan memilih file yang akan didownload kemudian akan ditampilkan dialog seperti pada gambar 5.35.



Gambar 5. 35 Tampilan Download File

Tombol *open* digunakan untuk membuka file yang akan didownload. *Save* digunakan untuk menyimpan file tersebut di direktori pengguna di komputer lokal. *Cancel* akan membatalkan proses *download*.

```

header("Content-disposition: attachment; filename=\"\$file\"");
header("Content-type: application/octetstream");
header("Pragma: ");
header("Cache-Control: cache");
header("Expires: 0");

$fp = fopen($downloadDir . $file, "w");
ftp_fget($ftp_conn,$fp,"$file",$mode) OR DIE(_error_download);
fclose($fp);
$data = readfile($downloadDir . $file);
$i=0;
while ($data[$i] != "")
{
    echo $data[$i];
    $i++;
}
unlink($downloadDir . $file);

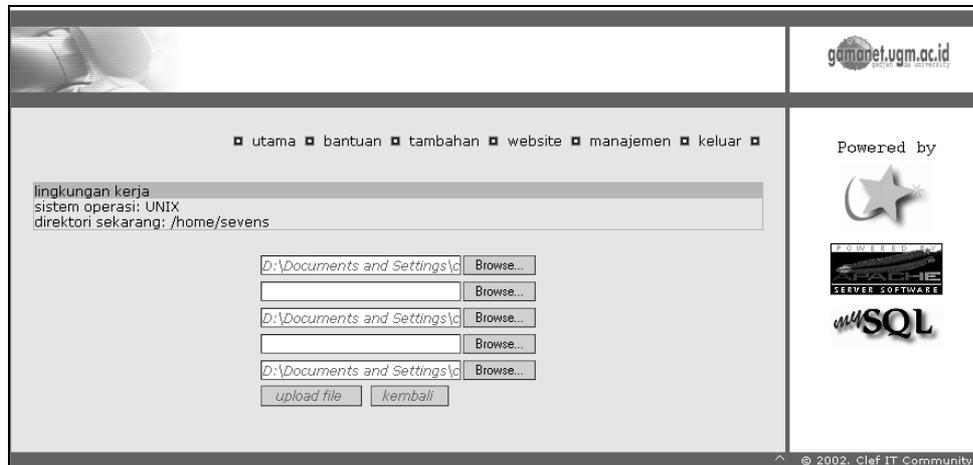
```

Gambar 5. 36 Kode Program *Download*

Untuk melakukan download file diperlukan seting header untuk mengirimkan data. Data dikirim menggunakan perintah `ftp_fget($ftp_conn, $fp, "$file", $mode)` dengan cara membuka file tersebut *record per record*.

5.2.1.6.6 Upload

Proses *upload* digunakan untuk menyalin file dari direktori lokal pengguna ke direktori pengguna di server. Proses ini memerlukan masukan berupa nama file beserta lokasi. Pada aplikasi ini disediakan maksimum sepuluh form isian. Pengisian nama file dan lokasinya tidak mengharuskan berurutan ataupun terisi semua. Pengguna dapat memilih form manapun untuk mengirimkan file.



Gambar 5. 37 Tampilan Form Isian *Upload*

Gambar 5.37 meminta pengguna untuk memberikan masukan nama file lengkap dengan lokasinya. Dari lima buah isian yang ditampilkan pengguna dapat mengisi sesuai kebutuhan dan tidak harus berurutan. File-file tersebut akan di *upload* ke direktori yang ditunjukkan diatas yaitu `/home/sevens`.

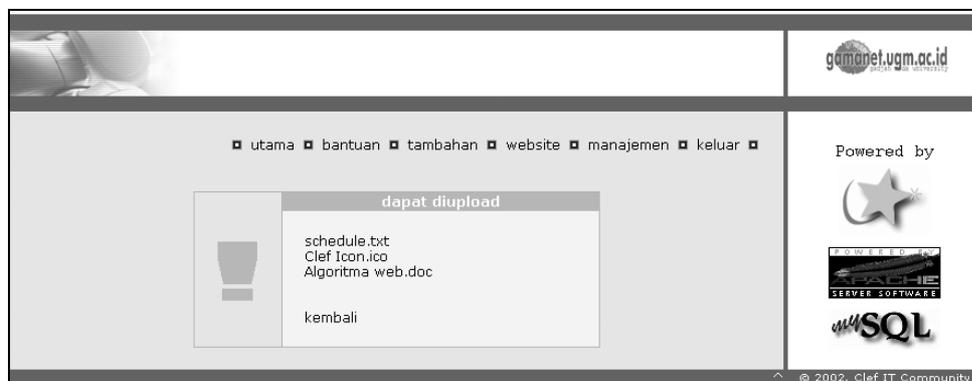
```

for ($i=0;$i<$shownum;$i++)
{
    if (file_exists($HTTP_POST_FILES[$f1]['tmp_name']))
    {
        .
        .
        if (!@ftp_put($ftp_conn, $currentDir . "/" .
            $HTTP_POST_FILES[$f1]['name'],
            $HTTP_POST_FILES[$f1]['tmp_name'], 1))
        {
            $msg .= $HTTP_POST_FILES[$f1]['name'] . "<br>";
        }
        else
        {
            $msgs .= $HTTP_POST_FILES[$f1]['name'] . "<br>";
        }
    }
    @unlink($HTTP_POST_FILES[$f1]['tmp_name']);
}

```

Gambar 5. 38 Kode Program *Upload File*

Kode program pada gambar 5.38 akan mengirimkan file yang dimasukkan ke server di direktori dimana pengguna mengidentifikasi. Perintah untuk melakukan *upload* adalah `ftp_put($ftp_conn, $currentDir . "/" . $HTTP_POST_FILES[$f1]['name'], $HTTP_POST_FILES[$f1]['tmp_name'], 0)`. Proses pengiriman berulang dari 1 sampai jumlah form isian. Setelah proses tersebut akan ditampilkan file yang dapat *diupload*, file yang gagal *diupload*.

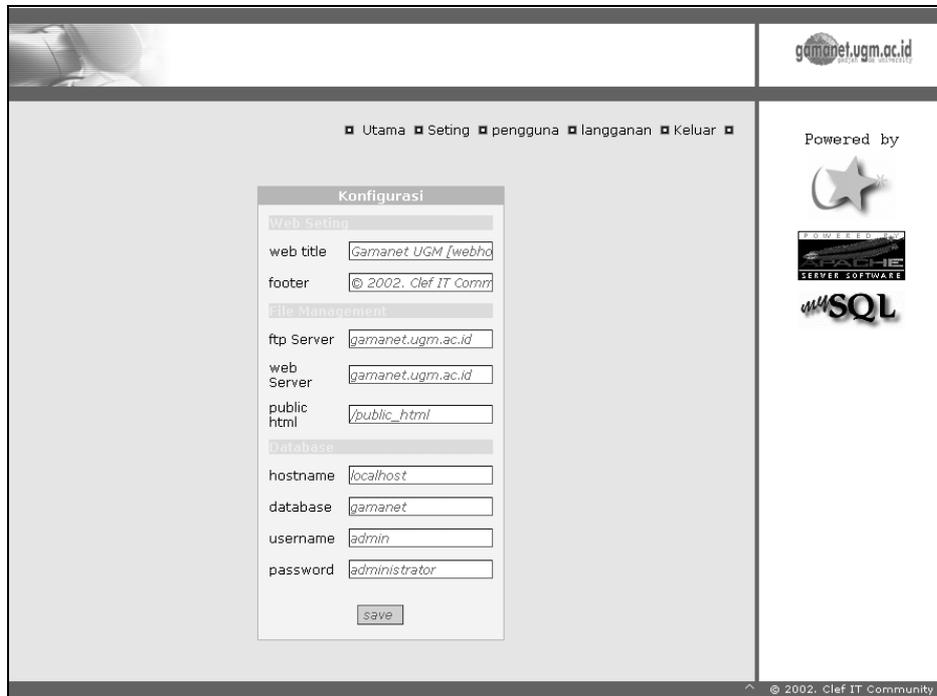


Gambar 5. 39 Tampilan Hasil Upload

5.2.2 Administrasi Pengelola

5.2.2.1 Konfigurasi Situs

Agar fungsi situs dapat berjalan sebagaimana mestinya diperlukan konfigurasi yang tepat. Konfigurasi yang diperlukan untuk melakukan proses koneksi ke basis data adalah *host*, *user*, *password*, *database*. Untuk proses manajemen file diperlukan konfigurasi *ftp server*, *web server*, dan *public html*. Untuk memberikan identitas pada situs diperlukan konfigurasi judul web dan *footer*.



Gambar 5. 40 Konfigurasi Situs

Gambar 5.40 menampilkan konfigurasi yang dapat diubah oleh pengelola. Secara *default* aplikasi akan menampilkan konfigurasi standar yang telah dibuat sebelumnya. Konfigurasi dibagi tiga yaitu web seting, manajemen file, dan *database*.

```

.
.
$file=fopen("variabel.php","w+");
fputs($file, "<?php\n");
fputs($file, "//Generated by Sevens Webhosting Application. \n");
fputs($file, "\$db_host=\"".trim($host)."\n");
.
.
fclose($file);

adminMenu();
$msg = "Konfigurasi berhasil disimpan";
message($msg, "Konfigurasi");
.
.

```

Gambar 5. 41 Kode Program Menyimpan Hasil Konfigurasi

Kode program pada gambar 5.41 akan melakukan penyimpanan hasil konfigurasi ke dalam file variabel.php. untuk menyimpan hasil tersebut menggunakan perintah `fputs($ffile, "\$db_host=\"".trim($host)."\";\n")`. Aplikasi akan menampilkan pesan berhasil jika konfigurasi dapat disimpan sedangkan tampilan gagal akan muncul jika *field* konfigurasi ada yang kosong atau belum diisi.

5.2.2.2 Pengguna

Bagian pengguna akan menampilkan daftar pengguna yang telah menggunakan aplikasi. Pengguna akan didaftarkan langsung ke basis data pada pertama kali login. Untuk selanjutnya pengguna dapat mengubah profil masing-masing. Hasil dari pengisian profil tersebut akan ditampilkan disini. Jika pengguna belum mengisi profil maka pada bagian ini akan ditampilkan userid dan nama. Untuk bagian yang lainnya akan dikosongkan.

uid	nama	nama lengkap	alamat	kota	kode pos
1	sevens	Sevens Spirits	Jogja	Jogja	55283
2	anno				
3	lank				
4	div				
5	tta				
6	clef	Clef IT Community	Nomaden	Jogja	55281

Gambar 5. 42 Daftar Pengguna

5.2.2.3 Langganan

Proses langganan yang diberikan dibagi menjadi dua macam, yaitu langganan pencacah kunjungan dan langganan buku tamu. Tampilan dari menu langganan seperti ditunjukkan pada gambar 5.43.

The screenshot shows a web application interface with a navigation menu at the top: [Utama](#) [Seting](#) [pengguna](#) [langganan](#) [Keluar](#). The main content area is divided into two sections:

Pencacah kunjungan

uid	nama	registrasi	cacah	perintah
1	sevens	2002-01-26 02:20:29	54	deactivate
6	clef	2002-01-28 22:40:29		1 deactivate

Buku Tamu

uid	nama	registrasi	pengunjung	perintah
1	sevens	2002-01-27 00:33:56		3 activate
2	anno	2002-01-26 02:15:33		0 activate
4	div	2002-01-28 23:08:16		0 activate
6	clef	2002-01-28 22:40:29		2 activate

The sidebar on the right contains the text "Powered by" above a star logo, a logo for "SERVER SOFTWARE", and the "SQL" logo. At the bottom right, there is a copyright notice: "© 2002, Clef IT Community".

Gambar 5. 43 Daftar Pengguna yang Berlangganan

Bagian atas akan menampilkan daftar pelanggan pencacah kunjungan. Data yang ditampilkan berupa userid, nama, tanggal berlangganan, dan cacah kunjungan. Bagian dibawahnya akan menampilkan daftar pelanggan buku tamu yang terdiri dari userid, nama, tanggal berlangganan, dan jumlah pengunjung yang mengisi buku tamu.

Pada kedua bagian tersebut status tiap pengguna yang berlangganan dapat dibuat aktif maupun tidak aktif. Jika status langganan pengguna aktif maka pengguna dapat menggunakan fasilitas tersebut, dan sebaliknya jika status langganan tidak aktif maka pengguna tidak dapat menggunakan fasilitas tersebut.

```

if (empty($userid))
{
    $msg = "userid tidak boleh kosong";
    message($msg, "Aktivasi");
    break;
}
if ($type=="counter")
{
    if ($status==0)
        $sql = "update pencacah set status='0' where userid='$userid'";
    else if ($status==1)
        $sql = "update pencacah set status='1' where userid='$userid'";
}
if ($type=="guestbook")
{
    if ($status==0)
        $sql = "update bukutamu set status='0' where userid='$userid'";
    else if ($status==1)
        $sql = "update bukutamu set status='1' where userid='$userid'";
}
$sql_res = doQuery($sql, $connect);
$sql_row = getAffected($connect);
if ($sql_row <1)
    $msg = "perubahan status gagal dilaksanakan";
else
    $msg = "perubahan status berhasil dilaksanakan";
adminMenu();
message($msg, "aktivasi");

```

Gambar 5. 44 Kode Program Aktivasi Langganan

Kode program pada gambar 5.44 akan mengaktifkan maupun menonaktifkan langganan berdasarkan status sebelumnya. Jika status sebelumnya aktif maka proses yang dilakukan adalah menonaktifkan langganan.

5.3 Instalasi Aplikasi

Untuk dapat menggunakan aplikasi dengan baik maka dilakukan instalasi. Instalasi dilakukan dalam dua bagian.

Bagian pertama akan mengharuskan seseorang yang akan melakukan instalasi (*instalator*) untuk menyalin file-file aplikasi kedalam direktori tempat untuk

menyimpan file-file situs. Untuk selanjutnya direktori tersebut akan disebut dengan direktori utama. Setelah menyalin file dilanjutkan dengan mengubah ijin dari direktori utama menjadi 755 (untuk linux), perintah yang digunakan `chmod 755 -R direktori_utama`. Dilanjutkan dengan perubahan pemilik dari direktori menjadi kepunyaan apache (`chown apache.apache -R direktori_utama`).

Bagian kedua adalah instalasi melalui web. Disini *instalator* harus mengisikan data-data situs untuk dapat menggunakan aplikasi *webhosting*. Data yang diperlukan untuk instalasi basis data adalah *basis data*, *host*, *administrator*, *password*. Data selanjutnya adalah nama admin dan password yang digunakan untuk pengelola. Langkah terakhir adalah data untuk setting web, terdiri dari *judul web*, *footer*, *ftp server*, *web server*, *basis data* (seperti pada langkah sebelumnya).

5.4 Struktur File

File-file yang dibuat pada sub bab 5.2 pada *web server* apache diletakkan pada direktori `var/www/html` untuk sistem operasi linux, `apache\htdocs` untuk sistem operasi windows atau direktori lain sesuai dengan konfigurasi *web server*. Aplikasi *webhosting* terdiri dari 313 buah file yang terdapat dalam 34 direktori.

5.4.1 Direktori Utama

Direktori utama ini merupakan direktori paling atas dari struktur direktori aplikasi *webhosting*. Terdiri dari 8 buah direktori dan 13 buah file. File-file yang terdapat dalam direktori utama ini adalah:

1. *install.php*, file ini digunakan untuk melakukan instalasi aplikasi webhosting.
2. *index.php*, file ini akan dijalankan pertama kali jika pengguna mengakses aplikasi *webhosting*. File *index.php* ini merupakan kunci dari aplikasi *webhosting*, karena semua *link* yang digunakan akan menuju ke file *index.php* untuk kemudian diteruskan ke file lain atau ke proses tertentu.
3. *login.php*, file ini akan menampilkan menu login untuk pengguna.
4. *header.php*, merupakan file yang akan menampilkan halaman web bagian atas.
5. *footer.php*, akan menampilkan halaman web bagian bawah. Jika file *header.php* dan file *footer.php* ke dalam suatu file baru maka akan tercipta tampilan halaman web yang utuh tetapi bagian tengah kosong.
6. *admin.php*, file ini digunakan untuk melakukan administrasi situs bagi pengelola.
7. *admin.inc*, digunakan untuk menyimpan informasi *username* dan *password* bagi pengelola.
8. *genpage.php*, digunakan untuk membuat otomatisasi halaman web.
9. *setting.php*, digunakan untuk menyimpan seting awal bagi aplikasi yang sifatnya tidak berubah.
10. *variabel.php*, digunakan untuk menyimpan seting awal bagi aplikasi yang sifatnya dapat berubah.

11. *counter.php*, digunakan untuk mengatur pencacah kunjungan. Pengaturan meliputi registrasi, menampilkan pencacah kunjungan dan perubahan konfigurasi.
12. *guestbook.php*, digunakan untuk mengatur proses bukutamu yang meliputi registrasi, *insert* buku tamu, menampilkan buku tamu menghapus buku tamu dan membuat contoh buku tamu.
13. *function.php*, digunakan untuk mengumpulkan fungsi-fungsi yang dibuat sendiri.
14. *setpermission.php*, digunakan untuk menampilkan konfigurasi ijin dari file atau direktori yang dipilih.

5.4.2 Direktori counter

Direktori ini digunakan untuk mengelompokkan tipe-tipe pencacah kunjungan. Tipe pencacah kunjungan yang ada adalah 14 buah. Setiap tipe mempunyai 11 buah file yang terdiri dari 10 file dengan nama *0.gif* sampai *9.gif* dan sebuah file yang mempunyai nama sesuai dengan nama tipe. Tipe yang tersedia adalah *cycle*, *digital*, *gold*, *green*, *grees*, *mage*, *metallic*, *normal*, *noskin*, *odometer*, *pump*, *redskin*, *roman*, *verda*.

5.4.3 Direktori help

Bantuan akan membantu pengguna dalam menggunakan aplikasi *webhosting* ini. Direktori ini akan menyimpan file-file help yang dibuat ke dalam dua buah bahasa yaitu bahasa indonesia dan bahasa inggris. Untuk bantuan dalam bahasa indonesia menggunakan file *id.php* dan untuk bahasa inggris digunakan file *en.php*.

5.4.4 Direktori img

Direktori img menyimpan gambar-gambar yang dibutuhkan untuk menampilkan halaman web yang sempurna. Terdiri dari 39 buah file gambar yang semuanya berformat gif.

5.4.5 Direktori language

Aplikasi *webhosting* dibuat dalam dua bahasa yaitu bahasa indonesia dan bahasa inggris. Untuk memudahkan dalam mengubah bahasa yang digunakan maka dibuat file yang berbeda untuk masing-masing bahasa. File-file tersebut berisi konstanta-konstanta yang diterjemahkan ke dalam bahasa indonesia(*id.php*) dan bahasa inggris(*en.php*).

5.4.6 Direktori script

Javascript diperlukan untuk melakukan proses manajemen file. Direktori script menyimpan file javascript untuk bahasa indonesia(*id.js*) dan bahasa inggris (*en.js*).

5.4.7 Direktori style

Direktori style digunakan untuk menampung file *cascading style sheet(cm.css)*. File ini digunakan untuk mengubah tampilan dari situs agar sesuai dengan *style* yang telah ditentukan.

5.4.8 Direktori template

Direktori ini digunakan untuk mengelompokkan *template* pada proses otomasi halaman web. File *template.inc* yang ada digunakan untuk mendaftarkan *template* yang ada. Tiap-tiap *template* mempunyai 10 buah file.

5.4.9 Direktori tmp

Direktori ini digunakan untuk menyimpan sementara konfigurasi pada proses otomasi halaman web. Nama file yang digunakan adalah nama pengguna *_setting.php*. setelah proses otomasi halaman web selesai maka file yang telah dibuat akan dihapus.

5.5 Pengujian Aplikasi

Pengujian aplikasi digunakan untuk menentukan kebutuhan-kebutuhan dari aplikasi. Pengujian dilakukan pada tiga lingkungan infra struktur komputer yang berbeda. Komputer yang digunakan untuk melakukan pengujian aplikasi terdiri dari 4 buah. Satu buah komputer sebagai server, dua buah komputer sebagai client pada jaringan lokal dan satu buah komputer yang terhubung dengan internet.

Tabel 5. 1 Spesifikasi Server Pada Jaringan Lokal

No.	<i>Peripherals</i>	Spesifikasi
1	Processor	Intel Pentium 233 MHz MMX
2	VGA	2 MB
3	Harddisk	30 GB 7200 RPM
4	Ethernet Card	10 Mbps
5	Memori	126 MB
6	Sistem Operasi	Linux Mandrake 8.0
7	Web Browser	Konqueror, Mozilla

Tabel 5. 2 Spesifikasi Client 1 Pada Jaringan Lokal

No.	<i>Peripherals</i>	Spesifikasi
1	Processor	Intel Pentium 4 1,6 GHz
2	VGA	32 MB
3	Harddisk	40 GB 5400 RPM
4	Memori	128 MB
5	Ethernet Card	10 Mbps
6	Sistem Operasi	Windows XP
7	Web Browser	Internet Explorer 6, Netscape 6

Tabel 5. 3 Spesifikasi Client 2 Pada Jaringan Lokal

No.	<i>Peripherals</i>	Spesifikasi
1	Processor	AMD Duron 750 MHz
2	VGA	16 MB
3	Harddisk	10 GB 7200 RPM
4	Memori	112 MB
5	Ethernet Card	10/100 Mbps
6	Sistem Operasi	Windows XP
7	Web Browser	Internet Explorer 6, Netscape 6

Tabel 5. 4 Spesifikasi Client Internet

No.	Peripherals	Spesifikasi
1	Processor	Intel Pentium 4 1,6 GHz
2	VGA	32 MB
3	Harddisk	20 GB 7200 RPM
4	Memori	128 MB
5	Ethernet Card	10/100 Mbps
6	Sistem Operasi	Windows XP
7	Web Browser	Internet Explorer 6

Pengujian aplikasi untuk masing-masing kategori dilakukan dengan cara menghitung waktu tampil untuk halaman-halaman yang telah ditentukan sebelumnya. Penghitungan tersebut dilakukan dua kali dan dilakukan pada dua *web browser* yang berbeda. Pengujian untuk kategori internet dilakukan pada dua situs yang berbeda dan menggunakan sebuah web browser. Pengujian tersebut juga dilakukan dua kali. Satuan dari hasil pengujian yang dilakukan menggunakan detik.

Tabel 5. 5 Hasil Pengujian Aplikasi Pada Server

titik uji	Konqueror		mozilla	
	1	2	1	2
Pengujian				
Index	0,06	0,104	0,08	0,071
refresh index	0,164	0,104	0,04	0,041
Login	1,047	0,911	0,528	0,596
file manager	0,247	0,393	0,504	0,605
refresh file manager	0,206	0,47	0,596	0,517
upload(1.03 MB)	3,601	3,562	5,746	5,41
download(1.03 MB)	26,42	14,83	7,182	7,978

Dari hasil pengujian pada komputer server yang ditunjukkan pada tabel 5.5 memberikan gambaran bahwa web browser yang handal untuk menjalankan aplikasi *webhosting* ini adalah mozilla. Hal tersebut dibuktikan dengan rata-rata waktu untuk

menampilkan halaman pada titik uji relatif lebih kecil. Untuk proses *upload* dan *download* mozilla sangat bagus. Waktu yang diperlukan antara mozilla dan konqueror terpaut jauh.

Tabel 5. 6 Tabel Hasil Pengujian Aplikasi Pada Jaringan Lokal

titik uji	client 1				client 2			
	1		2		1		2	
Browser	ie	Nets	ie	nets	Ie	Nets	ie	nets
Index	0,02	0.02	0.02	0,02	0,02	0.02	0,02	0,019
refresh index	0,016	0.018	0.016	0,016	0,02	0.018	0,016	0,016
Login	0,205	0.207	0.208	0,215	0,2	0.207	0,201	0,199
file manager	0,249	0.259	0.259	0,253	0,25	0.256	0,246	0,241
refresh file manager	0,255	0.256	0.257	0,256	0,25	0.251	0,249	0,248
upload(1.03 MB)	2,961	3014	2.97	2,776	2,76	3.01	2,713	2,768
download(1.03 MB)	10,47	-	8031	-	10,3	-	7,827	-

Hasil pengujian aplikasi pada jaringan lokal untuk tiap *client* dilakukan dua kali untuk tiap *web browser* yang digunakan. Pada *web browser* netscape 6 proses *download* file tidak dapat dilakukan karena netscape 6 tidak mendukung proses *download* file pada aplikasi *webhosting*. Untuk *web browser* internet explorer sudah dapat mendukung proses *download* file. Netscape lebih cepat dalam menampilkan halaman web daripada internet explorer. Tetapi untuk proses *upload* dan *download* internet explorer lebih unggul. Dari hasil pengujian pada jaringan lokal dapat ditentukan bahwa aplikasi *webhosting* dapat dijalankan dengan baik menggunakan *web browser* internet explorer.

Tabel 5. 7 Hasil Pengujian di Internet

titik uji	situs 1		situs 2	
	1	2	1	2
index	0,012	0,012	0,018	0,225
refresh index	0,002	0,002	0,016	0,167
login	0,142	0,023	0,184	0,532
file manager	2,755	1,623	3,314	2,526
refresh file manager	2,755	1,623	2,412	2,305
upload(1.03 m)	200,3	200,2	-	-
download(1.03 m)	129,4	137,7	-	-

Pengujian aplikasi *webhosting* di internet menggunakan sebuah komputer *client*. Aplikasi dipasang pada dua buah *webhosting* yang berbeda. Nama situs yang pertama adalah <http://sevens.goldgames-online.com> dan yang kedua adalah <http://sevens.host.sk>. Server dari situs pertama berada di Indonesia (<http://www.techscape.co.id>) dan server dari situs kedua berada di Slovakia (<http://www.host.sk>). Pengujian pada <http://gamanet.ugm.ac.id> tidak dapat dilakukan karena server yang akan digunakan belum siap untuk aplikasi *webhosting*, selain itu sampai sekarang situs <http://gamanet.ugm.ac.id> tidak dapat diakses dari jaringan yang berada di luar UGM. Koneksi internet yang digunakan sebesar 64 kbps.

Pengujian ini memberikan gambaran bahwa kondisi dari tiap server tidak sama. Pada situs pertama proses *upload* dan *download* dapat dilakukan sedangkan pada situs kedua proses tersebut tidak dapat dilakukan karena sistem administrator membatasi penggunaan skrip php. Hasil yang diperoleh menunjukkan bahwa situs pertama lebih cepat dalam menampilkan halaman web daripada situs kedua. Selisih waktu untuk menampilkan halaman web dari situs pertama dengan situs kedua tidak

terpaut jauh. Dari hasil ini dapat diambil kesimpulan bahwa jauh dekat rute antara server dengan *client* akan menentukan waktu untuk menampilkan halaman web disamping spesifikasi dari server.

Dari tiga pengujian yang telah dilakukan dapat diambil kesimpulan bahwa aplikasi webhosting dapat digunakan dengan baik menggunakan *web browser* internet explorer untuk sistem operasi berbasis windows dan mozilla untuk sistem operasi yang berbasis linux. Kecepatan untuk membuka halaman web tidak hanya tergantung pada spesifikasi server tetapi bergantung juga dengan rute antara server dengan client dan juga koneksi internet yang digunakan. Spesifikasi minimum dari *client* yang dapat digunakan untuk menjalankan aplikasi *webhosting* adalah komputer yang mampu untuk menjalankan web browser internet explorer 4.0 atau yang terbaru.

BAB VI

PENUTUP

6.1 Kesimpulan

1. Aplikasi *webhosting* ini dibuat dengan desain yang sederhana sehingga pengguna dapat dengan mudah dalam pengoperasiannya.
2. Pengguna tidak perlu menghafal perintah-perintah *FTP* dan *telnet* untuk melakukan konfigurasi situs. Perintah-perintah tersebut sudah dijalankan oleh aplikasi ketika pengguna melakukan proses klik.
3. Aplikasi *webhosting* menyediakan fasilitas pembuatan situs secara cepat dan praktis sehingga pengguna tidak perlu membuat sendiri situsnya. Aplikasi *webhosting* hanya membutuhkan data-data situs untuk pembuatannya.
4. Aplikasi *webhosting* memberikan fasilitas tambahan berupa buku tamu dan pencacah kunjungan untuk mempercantik situs pengguna.