BAB IV

HASIL DAN PEMBAHASAN

4.1. Perancangan Sistem

Perancangan sistem adalah tahapan setelah analisis dari siklus pengembangan sistem yang didefinisikan dari kebutuhan-kebutuhan fungsional dan persiapan untuk rancang bangun implementasi yang menggambarkan bagaimana suatu sistem dibentuk, yang dapat berupa penggambaran, perancangan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi juga menyangkut konfigurasi dari komponen-komponen perangkat keras dan perangkat lunak.

4.1.1. Tujuan Perancangan Sistem

Sistem berdasarkan hasil analisis secara menyeluruh, maka dirancanglah suatu *website* yang merupakan salah satu solusi yang ditawarkan untuk memperbaiki dan mengembangkan sistem yang sedang berjalan di Laskar Buah Batu Corps Indonesia (LBI-15) saat ini. Dengan dirancangnya sistem berbasis web berdasarkan kebutuhan informasi dari masing-masing pihak maka diharapkan sistem ini dapat berguna dan membantu dalam penyampaian informasi yang dibutuhkan dengan cepat dan mudah dalam pengaksesannya sesuai dengan kebutuhan dari user.

4.1.2. Gambaran Umum Sistem Yang Diusulkan

Website adalah suatu halaman yang saling berhubungan yang umumnya berada pada peladen yang sama berisikan kumpulan informasi yang disediakan secara perorangan, kelompok, atau organisasi sebagai media yang terhubung melalui web server dan dapat diakses melalui internet.

Pada *website* yang dirancang untuk organisasi Laskar BBC Indonesia 15 terdapat prosedur-prosedur yang telah ditetapkan, sebagai berikut:

Tabel 4.1 Prosedur Yang Diusulkan

Kategori Pengguna	Tugas	Hak
Administrator	Mengelola Web	- Menyetujui pendaftaran calon anggota - Menambah, mengubah, dan menghapus seluruh isi konten *website* - Mencetak kartu anggota, laporan

User	Mengakses Web	- Melihat isi website - Memberi komentar - Melakukan pendaftaran sebagai anggota - Menerima informasi dari sistem

4.1.3. Perancangan Prosedur Yang Diusulkan

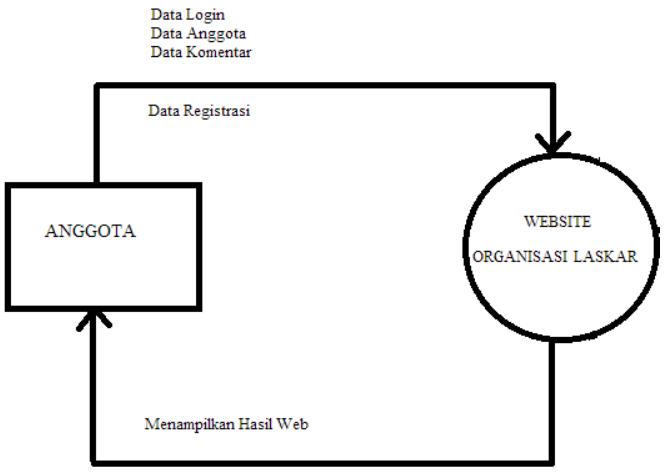
Perancangan proses dalam suatu sistem dilakukan untuk memudahkan dalam pengaliran suatu data dalam program. Sehingga memudahkan seseorang dalam pembuatan sistem agar sistem dapat dengan mudah dimengerti oleh orang yang menggunakan sistem tersebut. Perbedaan antara sistem lama dengan yang baru adalah dengan sistem yang baru proses penyajian informasi yang dibutuhkan oleh pelanggan dapat disajikan secara cepat, tepat, akurat sehingga dapat meningkatkan efektifitas kinerja perusahaan. Pada tahap perancangan proses secara umum komponen-komponen yang terkait terdiri dari:

1. Perancangan Diagram Konteks
2. Perancangan Data Flow Diagram
3. Kamus Data

4.1.3.1 Diagram Konteks

Diagram Konteks yaitu suatu diagram yang menggambarkan keseluruhan sistem. Dimana tahapan-tahapan dari DFD yang mempresentasikan keseluruhan proses perangkat lunak sebagai satu buah lingkungan dengan input dan output data yang diindikasikan dengan tanda panah masuk atau keluar dari satu atau lebih entitas (individu suatu objek yang mewakili sesuatu yang nyata). Dalam diagram konteks hanya membuat suatu proses yang mewakili keseluruhan proses yang ada di dalam sistem.

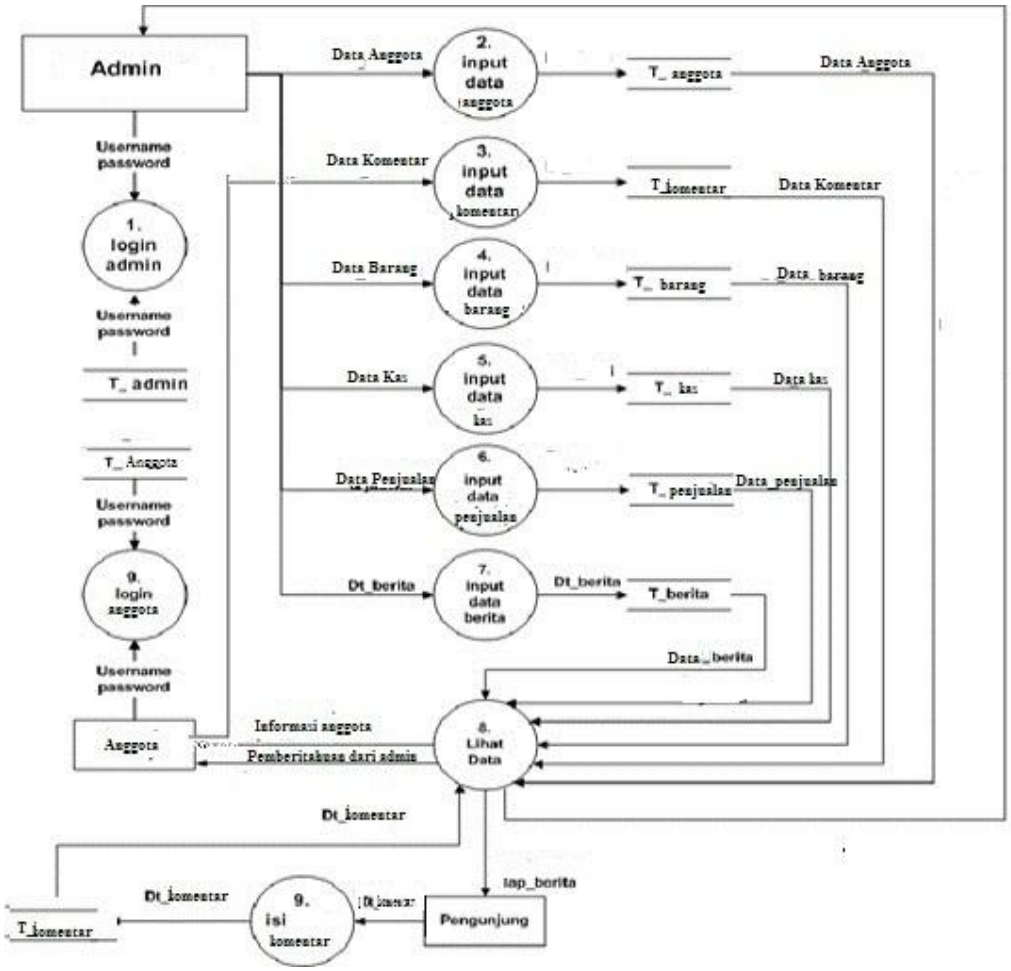
Sesuai dengan permasalahan yang dibahas, penulis dapat menggambarkan Diagram kontek untuk sistem ini seperti terlihat pada gambar berikut :



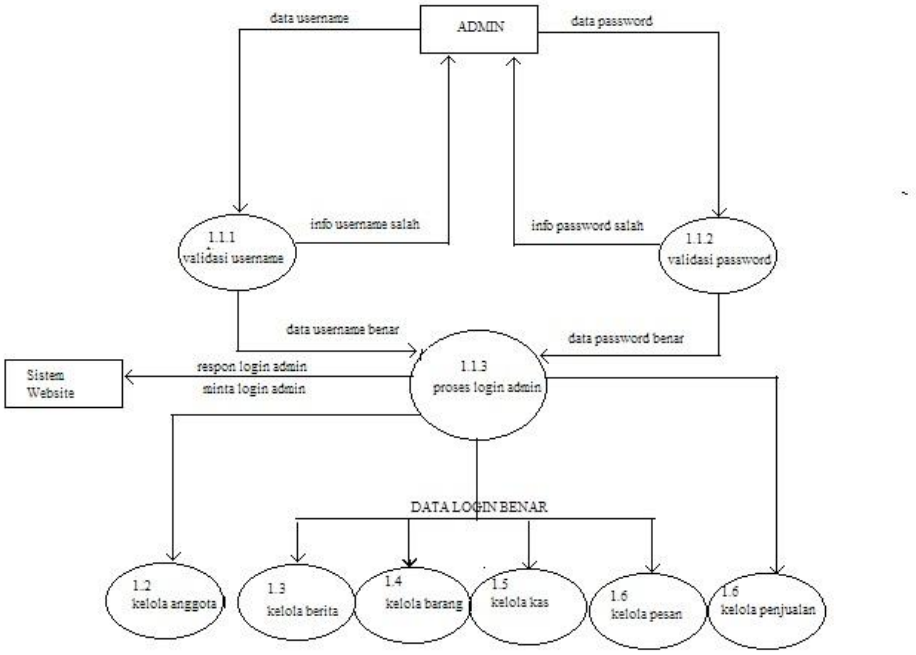
Gambar 4.1 Diagram Konteks Yang Diusulkan

4.1.3.2. Data Flow Diagram

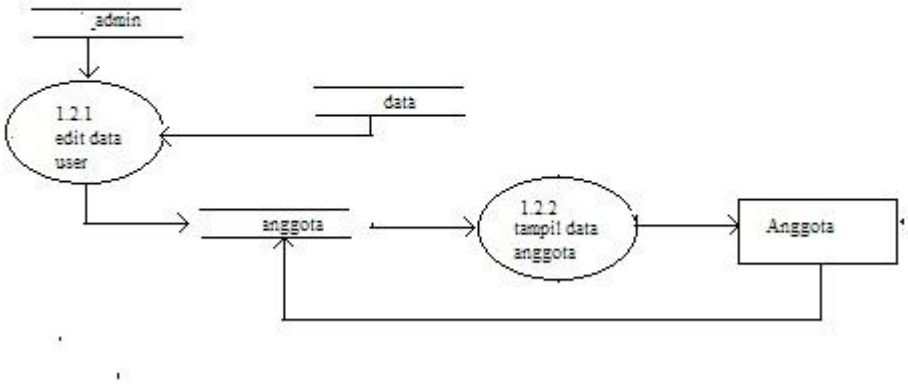
Data Flow Diagram (DFD) adalah representasi grafik yang menggambarkan arus data dari suatu sistem. Data Flow Diagram ini merupakan alat bantu dalam berkomunikasi dengan pemakai sistem untuk memahami secara logika tanpa memperhitungkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan.



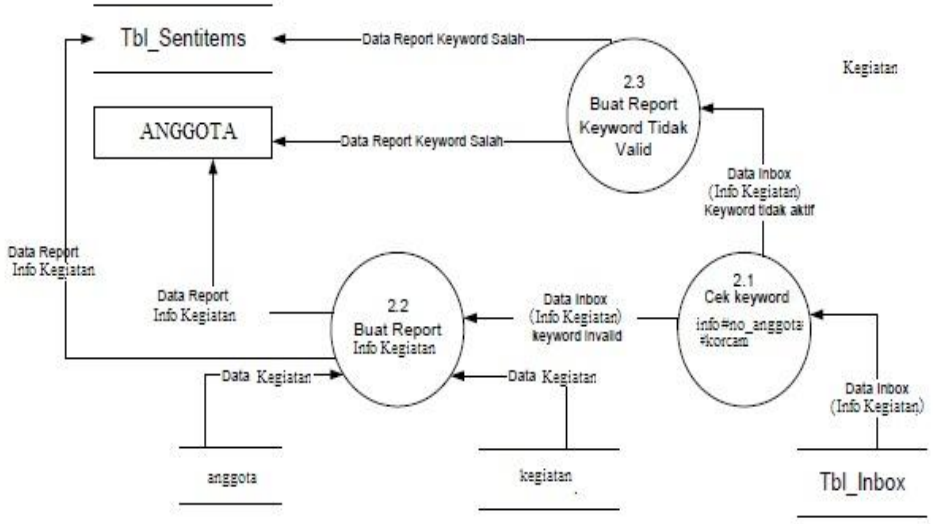
Gambar 4.2 DFD Lv 0 Diusulkan



Gambar 4.3 DFD Lv 1 proses 1



Gambar 4.4 DFD Lv 1 proses 2



Gambar 4.5 DFD Lv 2 proses 2

4.1.3.3. Kamus Data

Kamus data merupakan bagian dari perancangan sistem yang berisi field-field yang diperlukan oleh suatu database untuk menjalankan aplikasi program yang telah dibuat.Kamus data dapat dikatakan sebagai penjelasan dari field-field dalam table database.

Dengan menggunakan kamus data, analisis sistem dapat mendefinisikan data yang mengalir di sistem dengan lengkap. Kamus data dibuat berdasarkan arus data yang ada di *Data Flow Diagram*. Arus data dan *flow* diagram sifatnya global hanya ditujukan nama arus datanya saja. Data-data yang terkait pada website Laskar Buah Batu Corps Indonesia 15 adalah sebagai berikut :

Tabel 4.2 Tabel Arus Data

1	Nama Arus Data	Data User
	Deskripsi	Data yang di isi saat pendaftaran anggota
	Aliran Data	Dari anggota ke halaman proses daftar
	Periode	Setiap User mendaftar sebagai anggota
	Struktur File	user, pass , nama ,email, telpon, alamat, korcam, tgl lahir, jkelamin, foto, pekerjaan, marital, agama
2	Nama Arus Data	Informasi login anggota
	Deskripsi	Data user untuk login valid.
	Aliran Data	Dari proses halaman login ke tabel anggota
	Periode	Setiap melakukan login
	Struktur File	user, pass
3	Nama Arus Data	Data login admin
	Deskripsi	Data ketika admin melakukan login valid
	Aliran Data	Dari admin ke proses 1.1
	Periode	Setiap Login admin
	Struktur File	username, password
4	Nama Arus Data	Data kas
	Deskripsi	Data ketika adminn menginputkan kas anggota
	Aliran Data	Daro proses 1.0 ke ke proses 1.5

	Periode	Setiap ada pemasukan uang
	Struktur File	no_kas, tgl_pem, no_anggota, keterangan, jumlah

4.1.4. Perancangan Basis Data

Perancangan basis data adalah suatu kegiatan pembuatan dan penyimpanan data- data yang terdapat didalam sistem kedalam suatu database. Perancangan basis data bertujuan untuk mengorganisasikan dan menyusun data-data agar tidak terjadi kesalahan dalam pengolahan data dalam sebuah sistem.

4.1.4.1. Normalisasi

Normalisasi adalah suatu proses pengelompokan data elemen menjadi tabel-tabel yang menunjukan entity dan relasinya yang berfungsi untuk menghilangkan redudansi data dan atau, menentukan *key* yang unik untuk mengakses data item yang merupakan pembentukan sebuah *database* relation sedemikian rupa sehingga database tersebut menjadi modul modifikasi.

1. Bentuk Unnormal (UNF)

   Bentuk Tidak Normal atau *Un Normalized Form (UNF)* merupakan kumpulan data yang akan direkam. Data akan dikumpulkan sesuai apa adanya berdasarkan kedatangannya. Berikut ini merupakan bentuk tidak normal atau *Un Normalized Form (UNF)* : { id, username, password,

level, nama, no_anggota, no_pendaftaran, jabatan, tgl_masuk, bayar, kd_barang, nama_barang, harga, id_berita, judul, isi, gambar, kategori, tgl, no_pendaftaran, tgl_pendaftaran, nama, tempat_lahir, tgl_lahir,j_kelamin,alamat, email, agama, pekerjaan, status, telepon, korcam, foto, marital, user, pass, no_penj, kd_barang, jml_beli, total_bayar, no_kas, tgl_pem, no_anggota, keterangan, jumlah, id, judul, isi, korcam, tgl, kd_barang, jumlah, id_session, id, nama, email, komen, date, id_berita, url_berita, no_penj, tgl_pnj }

2. Bentuk normal pertama (1NF)

   Suatu relasi dikatakan mempunyai bentuk *normal form* pertama atau *First Norm Form* (1NF) bila semua *domain* adalah sederhana (*anomatik*). Artinya setiap atribut mempunyai domain tunggal. Bentuk normal pertama atau *First Norm Form* (1NF): { id, username, password, level, nama, no_anggota, no_pendaftaran, jabatan, tgl_masuk, bayar, kd_barang, nama_barang, harga, id_berita, judul, isi, gambar, kategori, tgl, no_pendaftaran, tgl_pendaftaran, nama, tempat_lahir, tgl_lahir,j_kelamin,alamat, email, agama, pekerjaan, status, telepon, korcam, foto, marital, user, pass, no_penj, kd_barang, jml_beli, total_bayar, no_kas, tgl_pem, no_anggota, keterangan, jumlah, id, judul, isi, korcam, tgl, kd_barang, jumlah, id_session, id, nama, email, komen, date, d_berita, url_berita, no_penj, tgl_pnj }

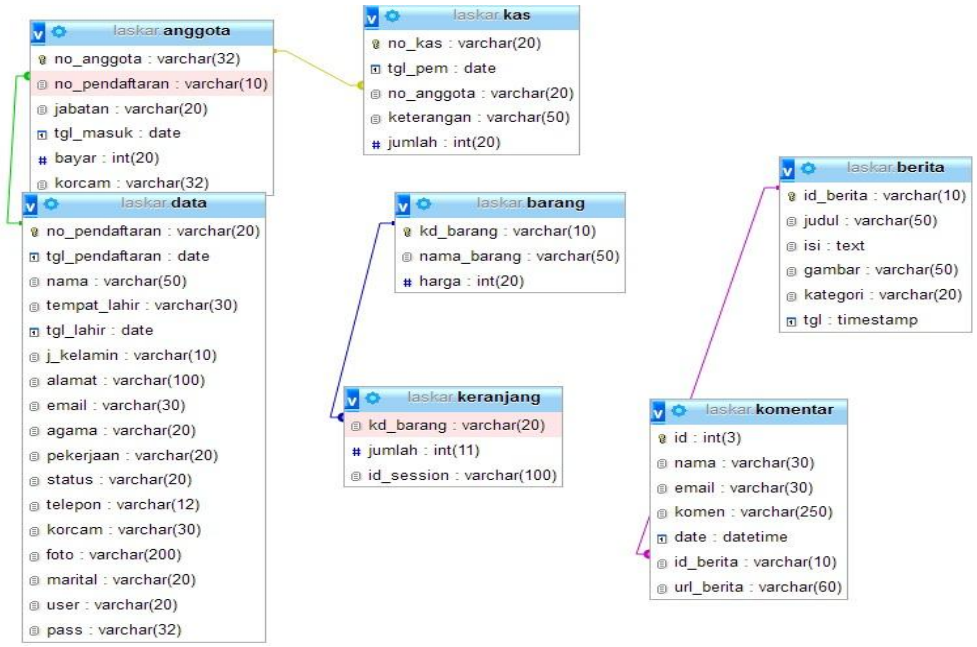
3. Aturan normal kedua atau *Second Norm Form* (2NF), menyatakan bahwa setiap field yang tidak termasuk dalam key primer memiliki ketergantungan fungsional pada key primer secara utuh. Adapun normal keduanya adalah :

    a) Tabel admin : {id*, username, password, level, nama}
    b) Tabel anggota : {no_anggota*, no_pendaftaran**, jabatan, tgl_masuk, bayar}
    c) Tabel barang : {kd_barang*, nama_barang, harga}
    d) Tabel berita : {id_berita*, judul, isi, gambar, kategori, tgl}
    e) Tabel data :{no_pendaftaran*, tgl_pendaftaran, nama, tempat_lahir, tgl_lahir,j_kelamin,alamat, email, agama, pekerjaan, status, telepon, korcam, foto, marital, user, pass}
    f) Tabel detail penjualan : {no_penj*, kd_barang, jml_beli, total_bayar}
    g) Tabel kas : {no_kas*, tgl_pem, no_anggota**, keterangan, jumlah}
    h) Tabel kegiatan : {id*, judul, isi, korcam, tgl}
    i) Tabel keranjang : {kd_barang**, jumlah, id_session}
    j) Tabel komentar : {id*, nama, email, komen, date, id_berita**, url_berita}

k) Tabel penjualan : {no_anggota**, no_penj**, tgl_pnj}

4.1.4.2. Relasi Tabel

Relasi tabel merupakan model data yang berupa notasi grafis dalam pemodelan data konseptual yang menggambarkan hubungan antara penyimpanan. Model data merupakan sekumpulan cara, peralatan untuk mendeskripsikan data data yang ada hubungannya satu sama lain, semantiknya serta batasan konsistensi.



Gambar 4.6 Relasi Antar Tabel

4.1.4.2. Struktur Tabel

Tabel merupakan tempat penyimpanan informasi dari sebuah aliran data dalam sebuah sistem. Berikut ini merupakan struktur dari beberapa tabel sistem yang akan dibangun.

Tabel 4.3 Tabel Admin

Nam field	Type	Panjang	Kunci
Id	Int	10	Primary Key
Username	Varchar	32	
Password	Varchar	32	
Level	Int	1	
Nama	Varchar	50	

Tabel 4.4 Tabel Anggota

Nam field	Type	Panjang	Kunci
No_anggota	Int	20	Primary Key
No_pendaftaran	Varchar	10	
Jabatan	Varchar	20	
Tgl_masuk	date		
Bayar	int	20	
Nama	Varchar	50	

Tabel 4.5 Tabel Barang

Nam field	Type	Panjang	Kunci
kd_barang	Varchar	10	Primary Key
nama_barang	Varchar	50	
harga	Int	20	

Tabel 4.6 Tabel Berita

Nam field	Type	Panjang	Kunci
Id	Int	10	Primary Key
Judul	Varchar	50	
Isi	Varchar	500	
Gambar	date		
Kategori	Varchar	20	

Tabel 4.7 Tabel Data

Nam field	Type	Panjang	Kunci
No_pendaftaran	Varchar	20	Primary Key
Tgl_pendaftaran	Date		
Nama	Varchar	50	
Tempat_lahir	Varchar	30	

Tgl_lahir	Date		
J_kelamin	Varchar	10	
Alamat	Varchar	30	
Email	Varchar	20	
Agama	Varchar	20	
Pekerjaan	Varchar	20	
Status	Varchar	20	
Telepon	Varchar	12	
Korcam	Varchar	30	
Foto	Varchar	200	
Marital	Varchar	20	
User	Varchar	20	
Pass	Varchar	32	

Tabel 4.8 Tabel Detail_penj

Nam field	Type	Panjang	Kunci
No_penj	Varchar	20	Primary Key
Kd_barang	Varchar	10	
Jml_beli	Int	20	
Total_bayar	Int	20	

Tabel 4.9 Tabel Kas

Nam field	Type	Panjang	Kunci
No_kas	Varchar	20	Primary Key
Tgl_pem	Date		
No_anggota	Varchar	20	
Keterangan	Varchar	50	
Jumlah	Int	20	

Tabel 4.10 Tabel Keranjang

Nam field	Type	Panjang	Kunci
Kd_barang	Varchar	20	
Jumlah	Int	11	
Id_session	Varchar	100	

Tabel 4.11 Tabel Komentar

Nam field	Type	Panjang	Kunci
Nama	Varchar	20	
Tgl_komen	Date		
Deskripsi	Varchar	250	
Id_komentar	Int	11	Primary Key

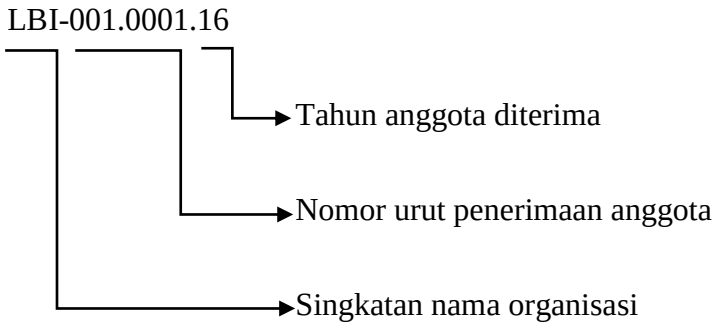
Tabel 4.12 Tabel Penjualan

Nam field	Type	Panjang	Kunci
No_anggota	Varchar	20	Primary Key
Tgl_pnj	Date		

4.1.4.3. Kodifikasi

Pengkodean dibuat untuk menidentifikasi suatu objek secara lebih singkat dan dapat mengurangi kesalahan dalam mengidentifikasi objek serta berguna untuk mengelompokan data. Pada sistem informasi di Laskar BBC Indonesia 15 ini terdapat kode-kode sebagai berikut:

a. Nomor Anggota



b. Nomor pendaftaran

P001

Nomor urut pendaftaran

"P" diambil dari kata Pendaftaran

c. Kode barang

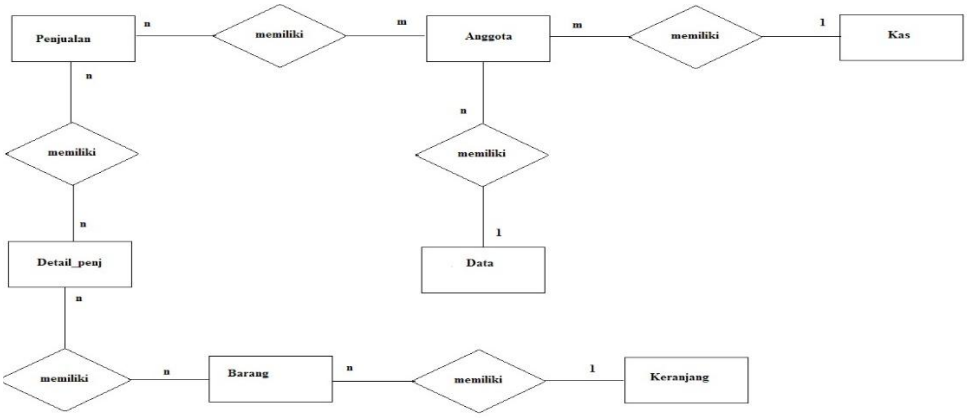
BRG-01

Nomor urut inputan barang

"BRG" diambil dari kata Barang

4.1.4.4 Entity Relationship Diagram

Entity Relation Diagram (ERD) adalah pengekspresian dari keadaan sebenarnya ke dalam kumpulan objek-objek dasar yang disebut entitas melalui relasi diantara entitas-entitas tersebut. Adapun diagram ERD pada *website* Laskar yang dirancang adalah sebagai berikut :



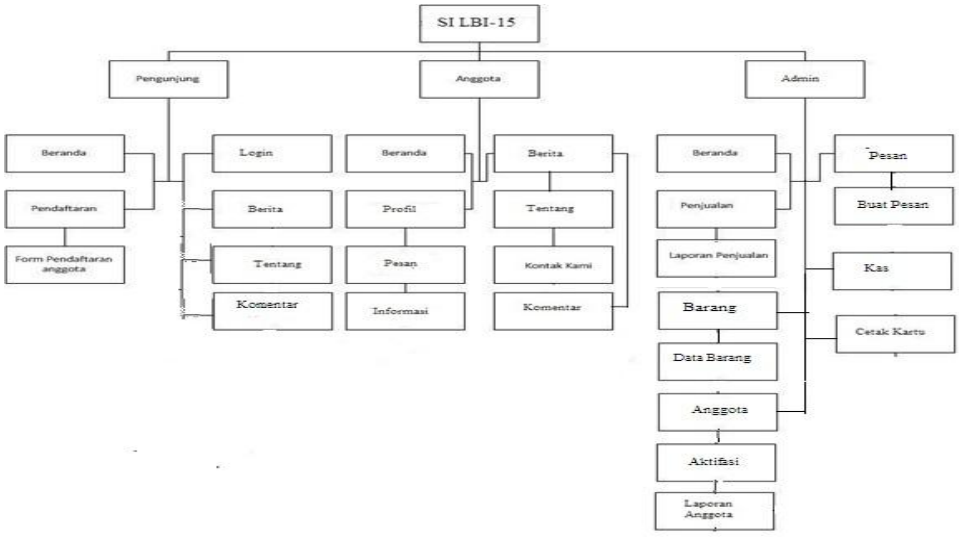
Gambar 4.7 ERD

4.2. Perancangan Antar Muka

Perancangan antarmuka merupakan tahap yang harus dikerjakan pada proses pembuatan sebuah program atau aplikasi. Antarmuka atau *interface* adalah suatu bagian yang berhubungan langsung dengan pengguna aplikasi. Perancangan antarmuka bertujuan agar program atau aplikasi yang dihasilkan terlihat lebih menarik dan mudah dimengerti pada saat dioperasikan oleh pengunjung , anggota maupun admin itu sendiri.

4.2.1. Struktur Menu

Struktur menu merupakan bentuk utama dari suatu rancangan program yang berfungsi untuk memudahkan dalam menjalankan suatu program sesuai dengan kebutuhannya. Berikut ini adalah struktur menu yang dirancang untuk Laskar BBC Indonesia 15.



Gambar 4.8 Rancangan Struktur Menu Website

4.2.2. Perancangan Input

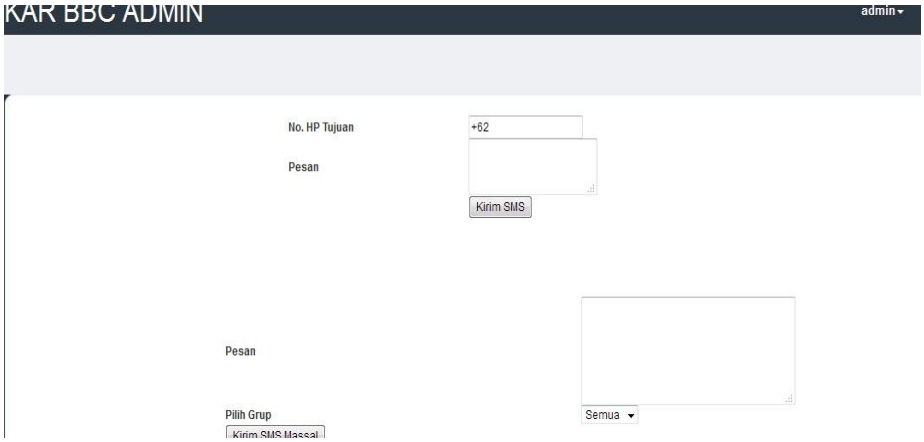
Perancangan input digunakan sebagai rancangan bentuk *input* (masukan) berisikan tentang data-data yang harus diinput user untuk dieksekusi oleh program atau website ini. Seperti gambar di bawah ini, perancangan yang telah di rancang sebagai berikut:

1. Form Input Pendaftaran



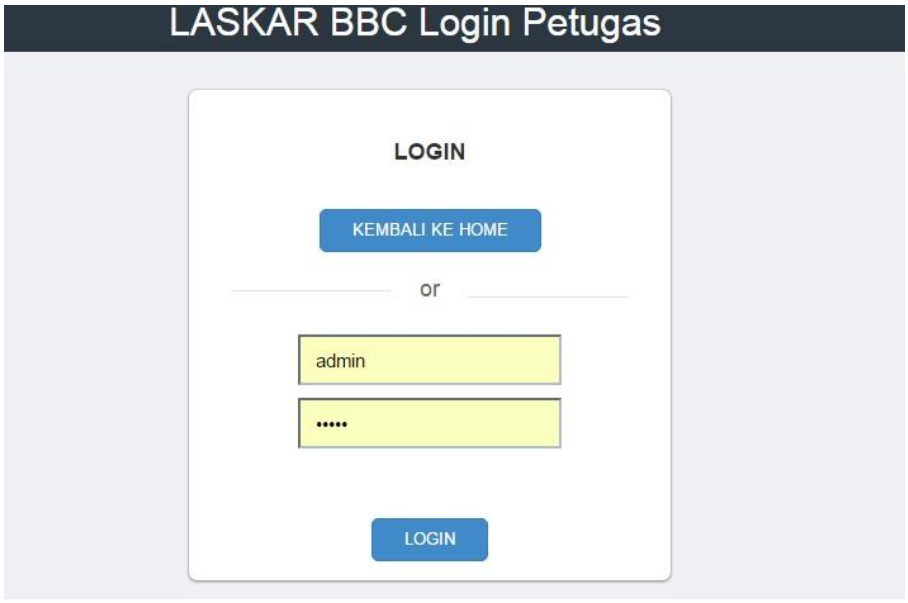
Gambar 4.9 Rancangan Form Pendaftaran

2. Kirim SMS



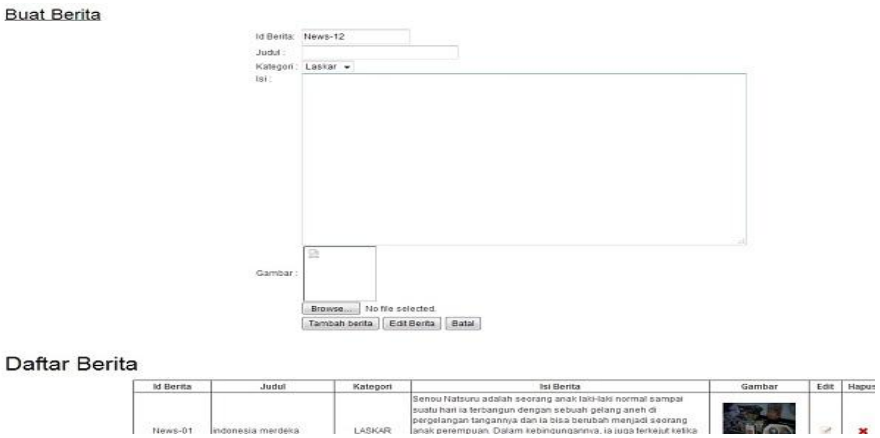
Gambar 4.10 Rancangan Kirim SMS

3. Form Login Admin



Gambar 4.11 Rancangan Login Admin

4. Form Input Berita



Gambar 4.12 Rancangan Input Berita

5. Penulisan format SMS *Keyword*

   a. SMS Keyword : LBI

      Contoh : LBI

      SMS balasan yang diterima apabila format benar adalah "Ketik INFO/KAS#NO ANGGOTA#KORCAM Untuk mendapatkan informasi"

   b. SMS Info Kegiatan : INFO#NO ANGGOTA#KORCAM

      Contoh : INFO#LBI-001.0003.17#Coblong

      SMS balasan yang diterima apabila format benar adalah : "Rapat. Rapat diadakan pada tanggal 5 Februari di Sekretariat Korcam Coblong Dago"

c. SMS Kas Anggota : KAS#NO ANGGOTA#KORCAM

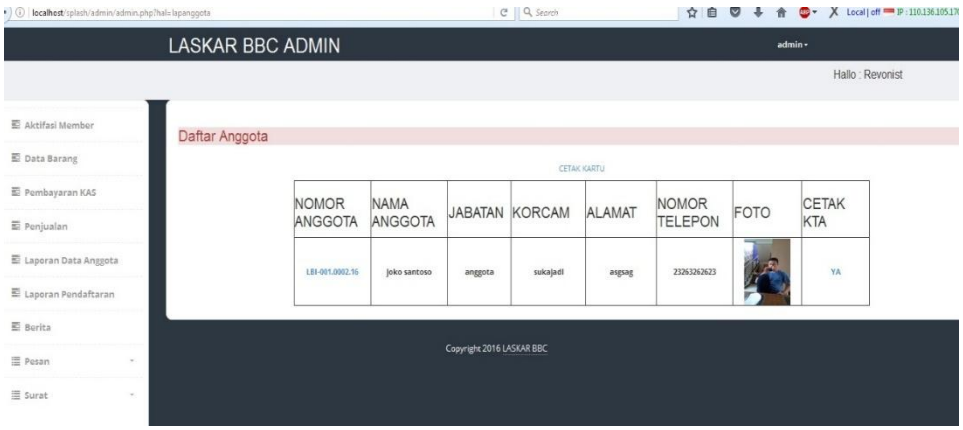
Contoh : KAS#LBI-001.0003.17#Coblong

SMS balasan yang diterima apabila format benar adalah : “Pembayaran Kas terakhir anda pada bulan Januari”

4.2.3. Perancangan Output

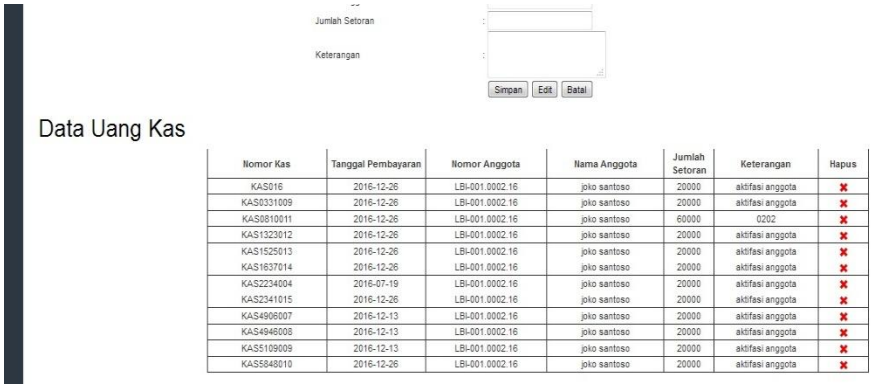
Seperti halnya rancangan input, rancangan output juga merupakan rencana pembuatan antarmuka untuk keluaran sebuah program. Berikut ini adalah output atau keluaran yang dirancang untuk *website* Laskar Buah Batu Corps Indonesia 15:

1. Laporan Anggota



Gambar 4.13 Rancangan Laporan Anggota

2. Laporan Uang Kas



Gambar 4.14 Rancangan Laporan Uang Kas

3. Laporan Penjualan Atribut Organisasi

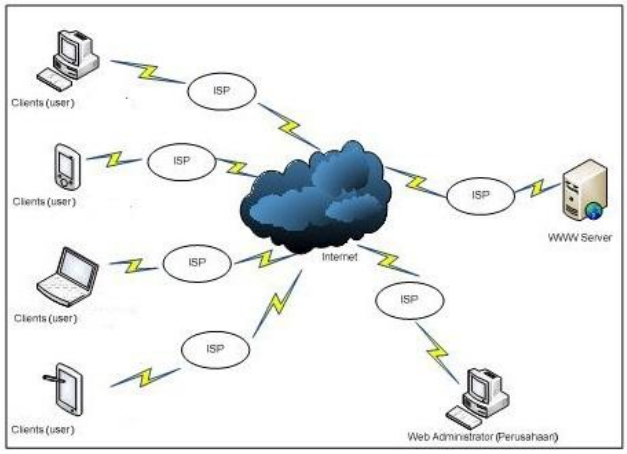


Gambar 4.15 Rancangan Laporan Penjualan Atribut Organisasi

4.3. Perancangan Arsitektur Jaringan

Arsitektur jaringan merupakan suatu pemilihan topologi jaringan dan protokol yang tepat agar kinerja perangkat lunak yang dibangun berjalan

maksimal. Pada aplikasi *dekstop* arsitektur jaringan sangat mempengaruhi kinerja perangkat lunak. Sedangkan untuk aplikasi berbasis *website*, arsitektur jaringan merupakan sesuatu yang optional karena *website* dapat dijalankan selama memiliki koneksi *internet* dan *internet* browser.



Gambar 4.16 Rancangan Arsitektur Jaringan

4.4. Pengujian

Pengujian program ini menggunakan metode Blackbox. Pengujian Black-Box merupakan pengujian program berdasarkan fungsi dari program. Tujuan dari metode Black Box ini adalah untuk menemukan kesalahan fungsi pada program. Pengujian dengan metode blackbox dilakukan dengan cara memberikan sejumlah input pada program aplikasi yang kemudian diproses sesuai dengan kebutuhan fungsionalnya untuk melihat apakah program aplikasi menghasilkan output yang diinginkan dan sesuai dengan fungsi dari program tersebut. Apabila dari input yang diberikan proses menghasilkan output yang

sesuai dengan kebutuhan fungsionalnya, maka program aplikasi yang bersangkutan telah benar, tetapi jika output yang dihasilkan tidak sesuai dengan kebutuhan fungsionalnya, maka masih terdapat kesalahan pada program aplikasi.

4.4.1. Rencana Pengujian

Rencana pengujian untuk sistem yang telah dibangun di tuangkan dalam tabel di bawah ini.

Tabel 4.13 Rencana Pengujian Admin

Menu Yang Diuji	**Butir Uji**	**Jenis Pengujian**
Login	Melakukan Proses Uji Login Admin	Blackbox
Beranda	Tampilan Link Beranda	Blackbox
Master	1. Pengolahan Data Anggota 2. Pengolahan Data Barang	Blackbox

	3. Pengolahan Data Berita 4. Pengolahan Pesan	
Laporan	1. Pengolahan Laporan Anggota 2. Pengolahan Laporan Transaksi	Blackbox
Lain-lain	Pengolahan Komentar	Blackbox

Tabel 4.14 Rencana Pengujian Anggota

Menu Yang Diuji	**Butir Uji**	**Jenis Pengujian**
Login	Melakukan Proses Login Anggota	Blackbox
Beranda	Tampilan Link Beranda	Blackbox
Pendaftaran	Melakukan Pengisian Form Pendaftaran	Blackbox
Kontak	Tampilan Link Kontak	Blackbox
Lain-lain	1. Posting Komentar	Blackbox

	2. Pemberitahuan	

4.4.2. Kasus dan Hasil Pengujian

Pengujian akan dilakukan satu persatu sesuai dengan rencana pengujian yang telah di rencanakan. Berikut adalah hasil pengujian.

4.4.2.1 Pengujian Login

Pengujian login dibagi menjadi 2 bagian yaitu login admin yang mempunyai hak akses penuh dan login user yang mempunyai hak akses dalam proses pendaftaran. Pengujian login ini dilakukan dengan menggunakan validasi terhadap data username

password dan hak akses yang dimasukan, apabila data masukan valid maka login sukses tetapi apabila data masukan tidak valid maka login gagal.

Tabel 4.15 Pengujian Login Admin

Kasus dan Hasil Uji Login Admin (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**

Username, dan Password yang dimasukan benar (valid)	Dapat masuk ke halaman utama administrator	User name dan password sesuai dengan hak akses.	[x] Diterima [ ] Ditolak
Kasus dan Hasil Uji Login Admin (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username dan Password yang dimasukan salah (tidak valid)	Tidak dapat login sebagai admin, dan menampilkan pesan "Username atau Password yang dimasukan salah"	User name dan password tidak sesuai dengan hak akses	[x] Diterima [ ] Ditolak

Tabel 4.16 Pengujian Login User

Kasus dan Hasil Uji Login User (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**

Username, dan Password yang dimasukan benar (valid)	Dapat masuk ke halaman profil anggota	User name dan password sesuai dengan hak akses	[x] Diterima [ ] Ditolak
Kasus dan Hasil Uji Login User (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username dan Password yang dimasukan salah (tidak valid)	Tidak dapat login sebagai user, dan menampilkan pesan "Username atau Password yang dimasukan salah"	User name dan password tidak sesuai dengan hak akses	[x] Diterima [ ] Ditolak

Tabel 4.17 Pengujian Input Pendaftaran Anggota

Kasus dan Hasil Uji Pendaftaran User (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**

Isi data formulir pendaftaran	Dapat masuk ke halaman data anggota	Pengisian sesuai dengan yang diharapkan	[x] Diterima [ ] Ditolak
Klik Tombol Daftar	Data tersimpan ke dalam database dan muncul pesan pendaftaran berhasil	Tombol daftar sesuai dengan yang diharapkan	[x] Diterima [ ] Ditolak
Klik Tombol Reset	Data yang telah diinputkan pada textfield akan menjadi kosong	Tombol reset sesuai dengan yang diharapkan	[x] Diterima [ ] Ditolak

4.4.2.2. Pengujian Aktifasi Anggota Baru

Pengujian ini adalah pengujian yang berisi data user sementara yang diinputkan dari formulir pendaftaran pada *website* yang selanjutnya akan divalidasi oleh admin untuk menjadi data anggota yang telah disetujui.

Tabel 4.18 Pengujian Aktifasi Anggota Baru

Kasus dan Hasil Uji Aktifasi Anggota Baru (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pilih user yang telah terdaftar untuk diaktifkan	Data tersimpan ke dalam database	Tombol simpan sesuai dengan yang diharapkan	[x] Diterima [ ] Ditolak
Kasus dan Hasil Uji Aktifasi Anggota Baru (Data Tidak Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pilih user yang telah terdaftar untuk diaktifkan	Data user dapat dipilih kembali	Tombol simpan sebagai aktifasi anggota baru tidak sesuai dengan yang diharapkan	[ ] Diterima [x] Ditolak

4.4.3. Kesimpulan dan Hasil Pengujian

Hasil pengujian yang telah dilakukan menunjukan bahwa aplikasi yang dibangun sudah memenuhi persyaratan fungsional. Akan tetapi pada prosesnya masih memungkinkan untuk terjadi kesalahan, namun frekuensi kesalahannya relative kecil dikarenakan dalam perancangan telah dirancang sistem untuk

menangani kemungkinan kesalahan proses yang terjadi. Secara fungsional aplikasi yang telah dibangun dinyatakan layak dan telah menghasilkan keluaran sesuai dengan yang diharapkan.

4.5. Implementasi

Implementasi adalah suatu bentuk kegiatan yang merupakan rangkaian lanjutan dari kegiatan perancangan perangkat lunak. Implementasi dimaksudkan sebagai usaha untuk mewujudkan hasil dari perancangan perangkat lunak. Hasil ini berarti hasil dari kegiatan rangkaian implementasi perancangan perangkat lunak adalah proses aplikasi yang telah dibuat dan sudah dapat digunakan serta berfungsi dengan baik.

Jadi secara garis besar implementasi merupakan proses penerapan rancangan program yang telah dibuat pada bab sebelumnya atau pada tahapan sebelumnya aplikasi dalam melaksanakan sistem informasi pemograman yang dibuat.

4.5.1. Implementasi Perangkat Lunak

Adapun beberapa perangkat lunak yang harus disiapkan dalam pembangunan *website* ini dengan menggunakan beberapa *development tools*, antara lain :

1. XAMPP

Dengar menginstal *XAMPP* tidak perlu lagi menginstal *MySQL, Apache* secara manual, karena *XAMPP* telah menyediakan fitur tersebut. Versi yang di gunakan penulis saat ini adalah *XAMPP* 1.8.1 [PHP: 5.4.7] menggunakan *localhost* sebagai server dan sudah tersedia *Apache/2.4.3 (Win32) OpenSSL/1.0.1c PHP/5.4.7* sebagai web servernya.

2. Adobe Dreamweaver CS5

   Penulis menggunakan *Adobe Dreamweaver CS5* Versi 11.0 sebagai media untuk penulisan script PHP serta desainnya

3. GAMMU

   Mobile Management : GNU All Mobile Management Utilities (GAMMU) sebagai tambahan fitur *sms gateway*

4. Mozila Firefox

   Sebagai media untuk menampilkan *website*

4.5.2. Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan kebutuhan minimum yang harus terpenuhi antara lain :

1. Untuk Admin
   a) *Processor Intel Pentium Dual Core* dan sekelasnya atau keatas
   b) RAM minimal 512 Mb
   c) *Hard Drive*, minimal 160 Gb sebagai media penyimpanan data
   d) *Mouse, Keyboard,* dan *Monitor* sebagai peralatan antarmuka

e) Modem atau perangkat lain agar terhubung pada jaringan *Internet*

f) Alat pendukung lainnya seperti *printer* dan *scanner*

2. Untuk User

a) *Processor Intel Pentium Dual Core* dan sekelasnya atau keatas

b) RAM minimal 512 Mb

c) *Hard Drive*, sebagai media penyimpanan data

d) *Mouse, Keybord,* dan *Monitor* sebagai peralatan antarmuka

e) Modem atau perangkat lain agar terhubung pada jaringan *Internet*

g) Alat pendukung lainnya seperti *printer* dan *scanner*

4.5.3. Implementasi Basis Data

Implementasi basis data dilakukan dengan menggunakan bahasa *SQL*, dimana aplikasi pemrograman yang digunakan adalah *MySQL,* implementasi basis datanya dalam bahasa *SQL* adalah sebagai berikut :

```
-- phpMyAdmin SQL Dump
-- version 3.5.2.2
-- http://www.phpmyadmin.net
--
-- Host: 127.0.0.1
-- Generation Time: Jan 07, 2017 at 07:12 AM
-- Server version: 5.5.27
```

```
-- PHP Version: 5.4.7

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";


/*!40101 SET
@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIEN
T */;
/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESU
LTS */;
/*!40101 SET
@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTI
ON */;
/*!40101 SET NAMES utf8 */;

--
-- Database: `laskar`
--

-- --------------------------------------------------------

--
-- Table structure for table `admin`
--

CREATE TABLE IF NOT EXISTS `admin` (
```

```
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `username` varchar(15) NOT NULL,
  `password` varchar(32) NOT NULL,
  `level` int(1) NOT NULL DEFAULT '0',
  `nama` varchar(50) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=2 ;

-- --------------------------------------------------------

--
-- Table structure for table `anggota`
--

CREATE TABLE IF NOT EXISTS `anggota` (
  `no_anggota` varchar(32) NOT NULL DEFAULT '',
  `no_pendaftaran` varchar(10) NOT NULL,
  `jabatan` varchar(20) NOT NULL,
  `tgl_masuk` date NOT NULL,
  `bayar` int(20) NOT NULL,
  PRIMARY KEY (`no_anggota`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--
-- Table structure for table `barang`
```

```
--

CREATE TABLE IF NOT EXISTS `barang` (
  `kd_barang` varchar(10) NOT NULL,
  `nama_barang` varchar(50) NOT NULL,
  `harga` int(20) NOT NULL,
  PRIMARY KEY (`kd_barang`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--
-- Table structure for table `berita`
--

CREATE TABLE IF NOT EXISTS `berita` (
  `id_berita` varchar(10) NOT NULL,
  `judul` varchar(50) NOT NULL,
  `isi` text NOT NULL,
  `gambar` varchar(50) NOT NULL,
  `kategori` varchar(20) NOT NULL,
  `tgl` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_berita`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------
```

```
--
-- Table structure for table `daemons`
--

CREATE TABLE IF NOT EXISTS `daemons` (
  `Start` text NOT NULL,
  `Info` text NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

-- --------------------------------------------------------

--
-- Table structure for table `data`
--

CREATE TABLE IF NOT EXISTS `data` (
  `no_pendaftaran` varchar(20) NOT NULL DEFAULT '',
  `tgl_pendaftaran` date NOT NULL,
  `nama` varchar(50) NOT NULL,
  `tempat_lahir` varchar(30) NOT NULL,
  `tgl_lahir` date NOT NULL,
  `j_kelamin` varchar(10) NOT NULL,
  `alamat` varchar(100) NOT NULL,
  `email` varchar(30) NOT NULL,
  `agama` varchar(20) NOT NULL,
  `pekerjaan` varchar(20) NOT NULL,
  `status` varchar(20) NOT NULL,
  `telepon` varchar(12) NOT NULL,
```

```
  `korcam` varchar(30) NOT NULL,
  `foto` varchar(200) NOT NULL,
  `marital` varchar(20) NOT NULL,
  `user` varchar(20) NOT NULL,
  `pass` varchar(32) NOT NULL,
  PRIMARY KEY (`no_pendaftaran`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--
-- Table structure for table `detail_penj`
--

CREATE TABLE IF NOT EXISTS `detail_penj` (
  `no_penj` varchar(20) NOT NULL,
  `kd_barang` varchar(10) NOT NULL,
  `jml_beli` int(20) NOT NULL,
  `total_bayar` int(20) NOT NULL,
  PRIMARY KEY (`no_penj`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--
-- Table structure for table `gammu`
--
```

```
CREATE TABLE IF NOT EXISTS `gammu` (
  `Version` int(11) NOT NULL DEFAULT '0'
) ENGINE=MyISAM DEFAULT CHARSET=utf8;


-- --------------------------------------------------------


--
-- Table structure for table `inbox`
--


CREATE TABLE IF NOT EXISTS `inbox` (
  `UpdatedInDB` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `ReceivingDateTime` timestamp NOT NULL DEFAULT '0000-00-00
00:00:00',
  `Text` text NOT NULL,
  `SenderNumber` varchar(20) NOT NULL DEFAULT '',
  `Coding`
enum('Default_No_Compression','Unicode_No_Compression','8bit','Default
_Compression','Unicode_Compression') NOT NULL DEFAULT
'Default_No_Compression',
  `UDH` text NOT NULL,
  `SMSCNumber` varchar(20) NOT NULL DEFAULT '',
  `Class` int(11) NOT NULL DEFAULT '-1',
  `TextDecoded` text NOT NULL,
  `ID` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `RecipientID` text NOT NULL,
  `Processed` enum('false','true') NOT NULL DEFAULT 'false',
```

```
  PRIMARY KEY (`ID`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8
AUTO_INCREMENT=16 ;


--
-- Triggers `inbox`
--
DROP TRIGGER IF EXISTS `inbox_timestamp`;
DELIMITER //
CREATE TRIGGER `inbox_timestamp` BEFORE INSERT ON `inbox`
 FOR EACH ROW BEGIN
   IF NEW.ReceivingDateTime = '0000-00-00 00:00:00' THEN
      SET NEW.ReceivingDateTime = CURRENT_TIMESTAMP();
   END IF;
END
//
DELIMITER ;


-- --------------------------------------------------------


--
-- Table structure for table `kas`
--


CREATE TABLE IF NOT EXISTS `kas` (
  `no_kas` varchar(20) NOT NULL,
  `tgl_pem` date NOT NULL,
  `no_anggota` varchar(20) NOT NULL,
```

```
  `keterangan` varchar(50) NOT NULL,
  `jumlah` int(20) NOT NULL,
  PRIMARY KEY (`no_kas`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--
-- Table structure for table `kegiatan`
--

CREATE TABLE IF NOT EXISTS `kegiatan` (
  `id` int(8) NOT NULL AUTO_INCREMENT,
  `judul` varchar(50) NOT NULL,
  `isi` text NOT NULL,
  `korcam` varchar(32) NOT NULL,
  `tgl` date NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=6 ;

-- --------------------------------------------------------

--
-- Table structure for table `keranjang`
--

CREATE TABLE IF NOT EXISTS `keranjang` (
```

```
  `kd_barang` varchar(20) NOT NULL,
  `jumlah` int(11) NOT NULL,
  `id_session` varchar(100) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--
-- Table structure for table `komentar`
--

CREATE TABLE IF NOT EXISTS `komentar` (
  `nama` varchar(20) NOT NULL,
  `tgl_komen` date NOT NULL,
  `deskripsi` text NOT NULL,
  `id_komentar` int(11) NOT NULL AUTO_INCREMENT,
  PRIMARY KEY (`id_komentar`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=1 ;

-- --------------------------------------------------------

--
-- Table structure for table `outbox`
--

CREATE TABLE IF NOT EXISTS `outbox` (
```

```
  `UpdatedInDB` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `InsertIntoDB` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  `SendingDateTime` timestamp NOT NULL DEFAULT '0000-00-00
00:00:00',
  `SendBefore` time NOT NULL DEFAULT '23:59:59',
  `SendAfter` time NOT NULL DEFAULT '00:00:00',
  `Text` text,
  `DestinationNumber` varchar(20) NOT NULL DEFAULT '',
  `Coding`
enum('Default_No_Compression','Unicode_No_Compression','8bit','Default
_Compression','Unicode_Compression') NOT NULL DEFAULT
'Default_No_Compression',
  `UDH` text,
  `Class` int(11) DEFAULT '-1',
  `TextDecoded` text NOT NULL,
  `ID` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `MultiPart` enum('false','true') DEFAULT 'false',
  `RelativeValidity` int(11) DEFAULT '-1',
  `SenderID` varchar(255) DEFAULT NULL,
  `SendingTimeOut` timestamp NULL DEFAULT '0000-00-00 00:00:00',
  `DeliveryReport` enum('default','yes','no') DEFAULT 'default',
  `CreatorID` text NOT NULL,
  PRIMARY KEY (`ID`),
  KEY `outbox_date` (`SendingDateTime`,`SendingTimeOut`),
  KEY `outbox_sender` (`SenderID`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8
AUTO_INCREMENT=62 ;
```

```
--
-- Triggers `outbox`
--
DROP TRIGGER IF EXISTS `outbox_timestamp`;
DELIMITER //
CREATE TRIGGER `outbox_timestamp` BEFORE INSERT ON `outbox`
 FOR EACH ROW BEGIN
   IF NEW.InsertIntoDB = '0000-00-00 00:00:00' THEN
      SET NEW.InsertIntoDB = CURRENT_TIMESTAMP();
   END IF;
   IF NEW.SendingDateTime = '0000-00-00 00:00:00' THEN
      SET NEW.SendingDateTime = CURRENT_TIMESTAMP();
   END IF;
   IF NEW.SendingTimeOut = '0000-00-00 00:00:00' THEN
      SET NEW.SendingTimeOut = CURRENT_TIMESTAMP();
   END IF;
END
//
DELIMITER ;

-- --------------------------------------------------------

--
-- Table structure for table `outbox_multipart`
--

CREATE TABLE IF NOT EXISTS `outbox_multipart` (
```

```
  `Text` text,
  `Coding`
enum('Default_No_Compression','Unicode_No_Compression','8bit','Default
_Compression','Unicode_Compression') NOT NULL DEFAULT
'Default_No_Compression',
  `UDH` text,
  `Class` int(11) DEFAULT '-1',
  `TextDecoded` text,
  `ID` int(10) unsigned NOT NULL DEFAULT '0',
  `SequencePosition` int(11) NOT NULL DEFAULT '1',
  PRIMARY KEY (`ID`,`SequencePosition`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

-- --------------------------------------------------------

--
-- Table structure for table `pbk`
--

CREATE TABLE IF NOT EXISTS `pbk` (
  `ID` int(11) NOT NULL AUTO_INCREMENT,
  `GroupID` int(11) NOT NULL DEFAULT '-1',
  `Name` text NOT NULL,
  `Number` text NOT NULL,
  PRIMARY KEY (`ID`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8
AUTO_INCREMENT=1 ;
```

```
-- --------------------------------------------------------

--
-- Table structure for table `pbk_groups`
--

CREATE TABLE IF NOT EXISTS `pbk_groups` (
  `Name` text NOT NULL,
  `ID` int(11) NOT NULL AUTO_INCREMENT,
  PRIMARY KEY (`ID`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8
AUTO_INCREMENT=1 ;

-- --------------------------------------------------------

--
-- Table structure for table `penjualan`
--

CREATE TABLE IF NOT EXISTS `penjualan` (
  `no_penj` varchar(20) NOT NULL,
  `tgl_pnj` date NOT NULL,
  PRIMARY KEY (`no_penj`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--
```

```
-- Table structure for table `phones`
--

CREATE TABLE IF NOT EXISTS `phones` (
  `ID` text NOT NULL,
  `UpdatedInDB` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `InsertIntoDB` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  `TimeOut` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  `Send` enum('yes','no') NOT NULL DEFAULT 'no',
  `Receive` enum('yes','no') NOT NULL DEFAULT 'no',
  `IMEI` varchar(35) NOT NULL,
  `Client` text NOT NULL,
  `Battery` int(11) NOT NULL DEFAULT '-1',
  `Signal` int(11) NOT NULL DEFAULT '-1',
  `Sent` int(11) NOT NULL DEFAULT '0',
  `Received` int(11) NOT NULL DEFAULT '0',
  PRIMARY KEY (`IMEI`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

--
-- Triggers `phones`
--
DROP TRIGGER IF EXISTS `phones_timestamp`;
DELIMITER //
CREATE TRIGGER `phones_timestamp` BEFORE INSERT ON `phones`
 FOR EACH ROW BEGIN
    IF NEW.InsertIntoDB = '0000-00-00 00:00:00' THEN
```

```
    SET NEW.InsertIntoDB = CURRENT_TIMESTAMP();
  END IF;
  IF NEW.TimeOut = '0000-00-00 00:00:00' THEN
    SET NEW.TimeOut = CURRENT_TIMESTAMP();
  END IF;
END
//
DELIMITER ;

-- --------------------------------------------------------

--
-- Table structure for table `sentitems`
--

CREATE TABLE IF NOT EXISTS `sentitems` (
 `UpdatedInDB` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
 `InsertIntoDB` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
 `SendingDateTime` timestamp NOT NULL DEFAULT '0000-00-00
00:00:00',
 `DeliveryDateTime` timestamp NULL DEFAULT NULL,
 `Text` text NOT NULL,
 `DestinationNumber` varchar(20) NOT NULL DEFAULT '',
 `Coding`
enum('Default_No_Compression','Unicode_No_Compression','8bit','Default
_Compression','Unicode_Compression') NOT NULL DEFAULT
'Default_No_Compression',
```

```
  `UDH` text NOT NULL,
  `SMSCNumber` varchar(20) NOT NULL DEFAULT '',
  `Class` int(11) NOT NULL DEFAULT '-1',
  `TextDecoded` text NOT NULL,
  `ID` int(10) unsigned NOT NULL DEFAULT '0',
  `SenderID` varchar(255) NOT NULL,
  `SequencePosition` int(11) NOT NULL DEFAULT '1',
  `Status`
enum('SendingOK','SendingOKNoReport','SendingError','DeliveryOK','Del
iveryFailed','DeliveryPending','DeliveryUnknown','Error') NOT NULL
DEFAULT 'SendingOK',
  `StatusError` int(11) NOT NULL DEFAULT '-1',
  `TPMR` int(11) NOT NULL DEFAULT '-1',
  `RelativeValidity` int(11) NOT NULL DEFAULT '-1',
  `CreatorID` text NOT NULL,
  PRIMARY KEY (`ID`,`SequencePosition`),
  KEY `sentitems_date` (`DeliveryDateTime`),
  KEY `sentitems_tpmr` (`TPMR`),
  KEY `sentitems_dest` (`DestinationNumber`),
  KEY `sentitems_sender` (`SenderID`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;


--
-- Triggers `sentitems`
--
DROP TRIGGER IF EXISTS `sentitems_timestamp`;
DELIMITER //
```

```
CREATE TRIGGER `sentitems_timestamp` BEFORE INSERT ON `sentitems`
 FOR EACH ROW BEGIN
   IF NEW.InsertIntoDB = '0000-00-00 00:00:00' THEN
      SET NEW.InsertIntoDB = CURRENT_TIMESTAMP();
   END IF;
   IF NEW.SendingDateTime = '0000-00-00 00:00:00' THEN
      SET NEW.SendingDateTime = CURRENT_TIMESTAMP();
   END IF;
END
//
DELIMITER ;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

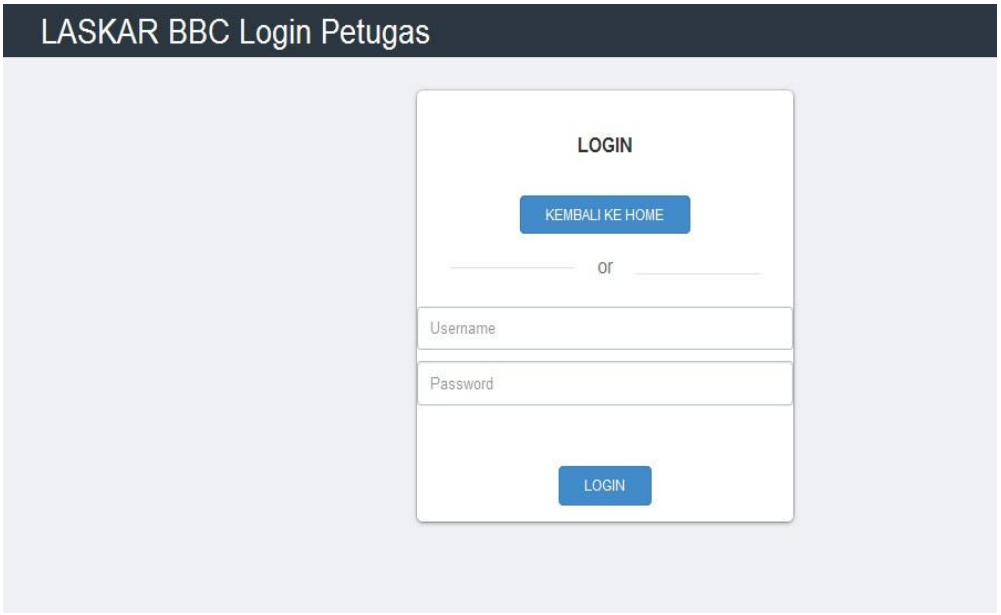
4.5.4. Implementasi Antar Muka

Implementasi antarmuka dilakukan dengan setiap halaman web yang dibuat dan pengkodeannya dalam bentuk file program berekstensi .php dan

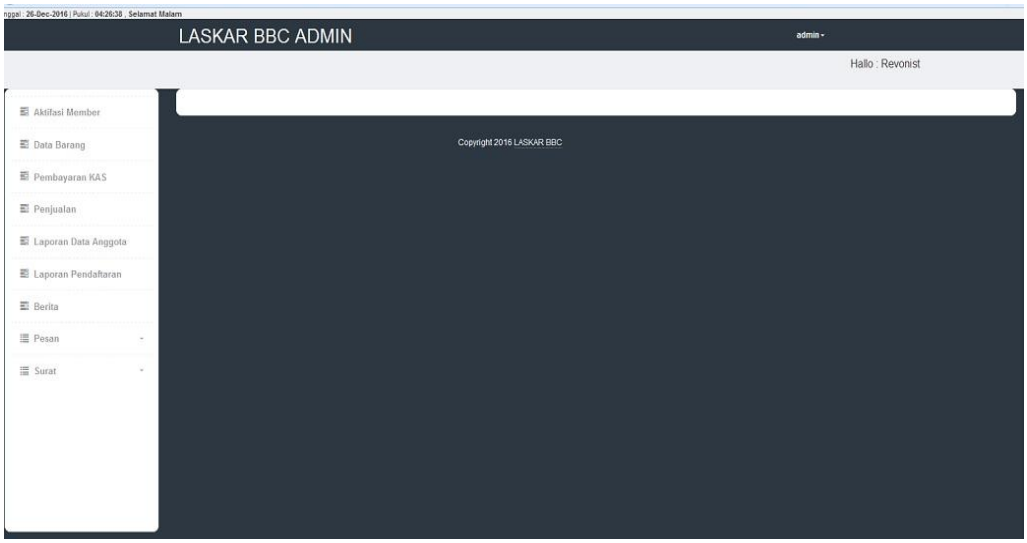
.html. Berikut ini adalah implementasi antarmuka yang dibuat dan dibedakan antara antarmuka untuk admin dan user.

4.5.4.1. Implementasi Halaman Admin

Berikut ini adalah tampilan halaman login admin dan halaman utama yang muncul setelah login sukses.



Gambar 4.17 Halaman Login Admin



Gambar 4.18 Halaman Utama Admin

4.5.4.2. Implementasi Halaman User

Berikut ini adalah tampilan dari halaman user saat mengakses *website*



Gambar 4.19 Halaman Login Anggota



Gambar 4.20 Halaman Anggota Setelah Login

4.5.5. Implementasi Instalasi Program

Implementasi instalasi program merupakan aspek awal untuk menjelaskan bagaimana penggunaan website ini dibangun untuk dapat dijalankan secara utuh didalam sistem komputer. Berikut langkah-langkah yang dilakukan dalam melakukan proses instalasi program agar berjalan pada kondisi offline.

1. Pastikan sudah memiliki software XAMPP untuk servernya
2. Lakukan konfigurasi Software XAMPP (*xampp-control.exe*), dengan mengaktifkan Apache dan MySQL

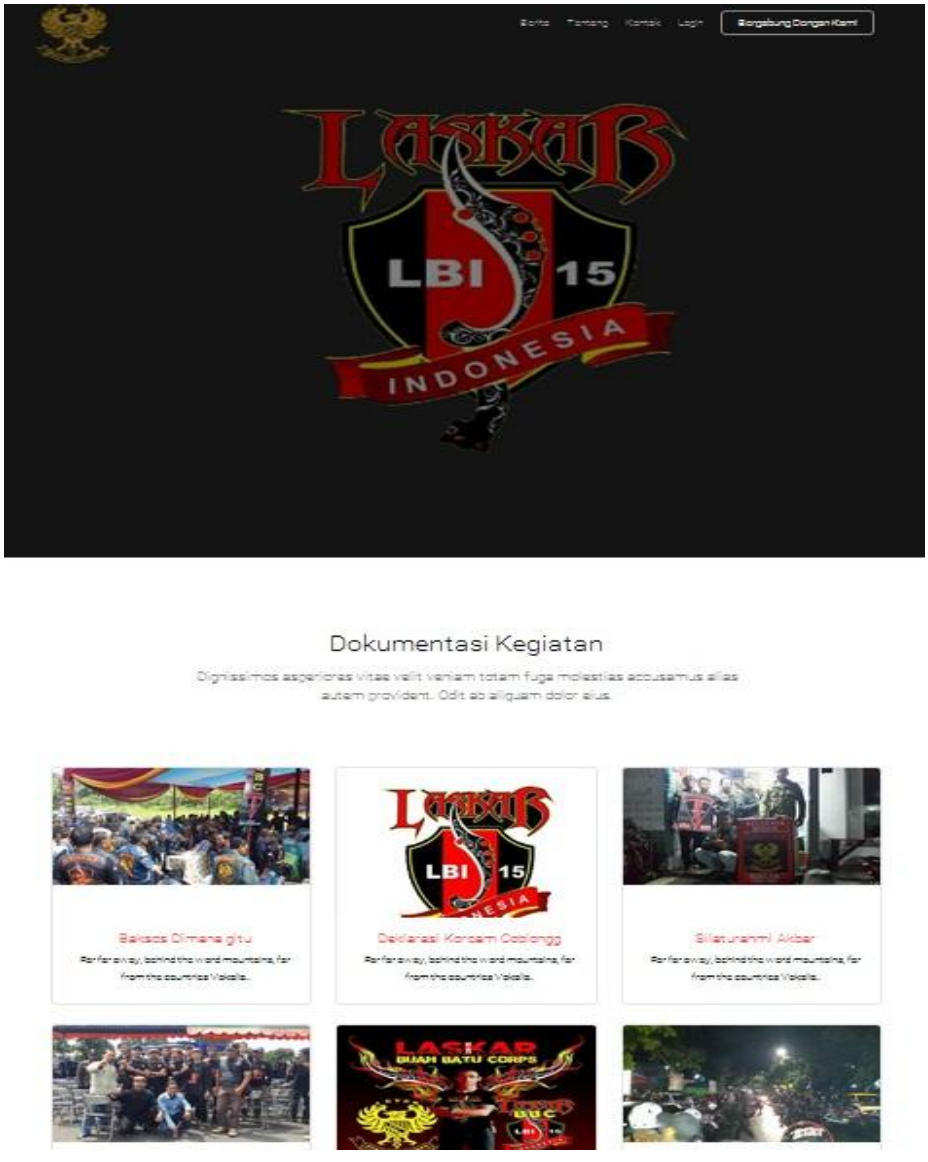
3. Konfigurasi database yang ada pada PhpMyAdmin, dengan menuju link, http://localhost/phpmyadmin/
4. Konfigurasi penyimpanan file data website Laskar Buah Batu Corps Indonesia secara offline, dengan menyimpan file pada folder *splash* di pada sub folder htdocs
5. Konfigurasi koneksi database, pastikan file database yang ada pada folder *splash* dengan nama data koneksinya adalah koneksi.php, harus memiliki sintak database yang sama.
6. Setelah proses instalasi selesai semua maka untuk mengakses menu utama website Laskar Buah Batu Corps Indonesia secara *offline* menggunakan alamat : http://localhost/splash dan untuk direktori admin http://localhost/splash/admin

4.5.6. Penggunaan Program

Pada tahapan ini penggunaan program bertujuan untuk mempermudah user dalam mendapatkan informasi khususnya untuk anggota Laskar itu sendiri, yaitu dengan mengikuti langkah – langkah yang akan di jelaskan. Akan dijelaskan dari tampilan utama ketika pengguna (anggota atau admin) mengakses *website* tersebut, kemudian melakukan pendaftaran, isi komentar, login user, melakukan pengolahan data hingga logout atau keluar dari *website* tersebut.

4.5.6.1 Tampilan Utama Website

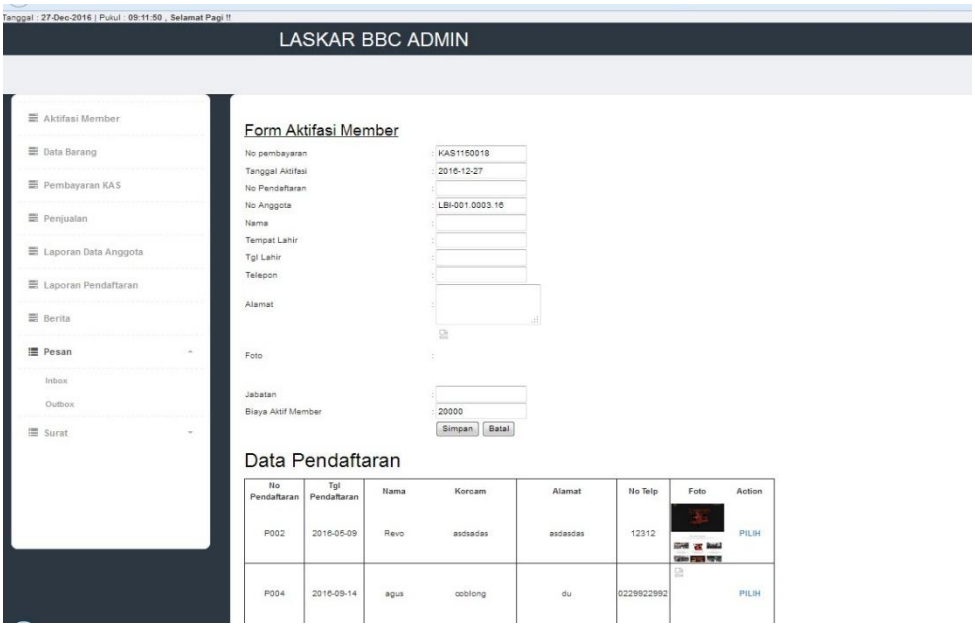
Ini adalah tampilan utama dari website yang akan dijumpai oleh *user* (pengguna) baik anggota maupun admin pada saat mengakses website Laskar Buah Batu Corps Indonesia.



Gambar 4.21 Tampilan Halam Utama Website

4.5.6.2. Tampilan Utama Halaman Admin

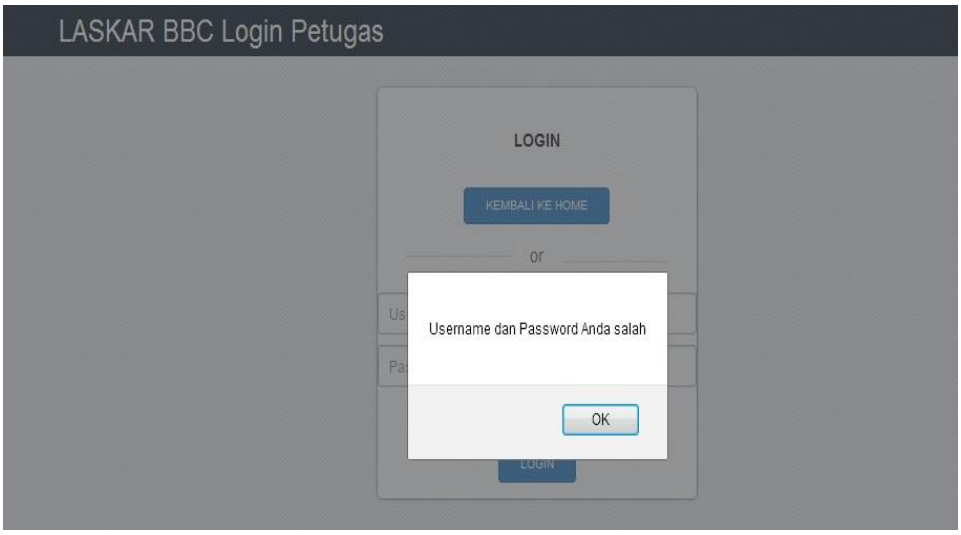
Ini adalah tampilan utama saat admin sukses melakukan login pada website, terdapat beberapa menu di panel sebelah kiri yang bisa digunakan admin sesuai dengan kebutuhannya.



Gambar 4.22 Tampilan Halaman Utama Admin

4.5.6.3. Tampilan Salah Input Password Admin

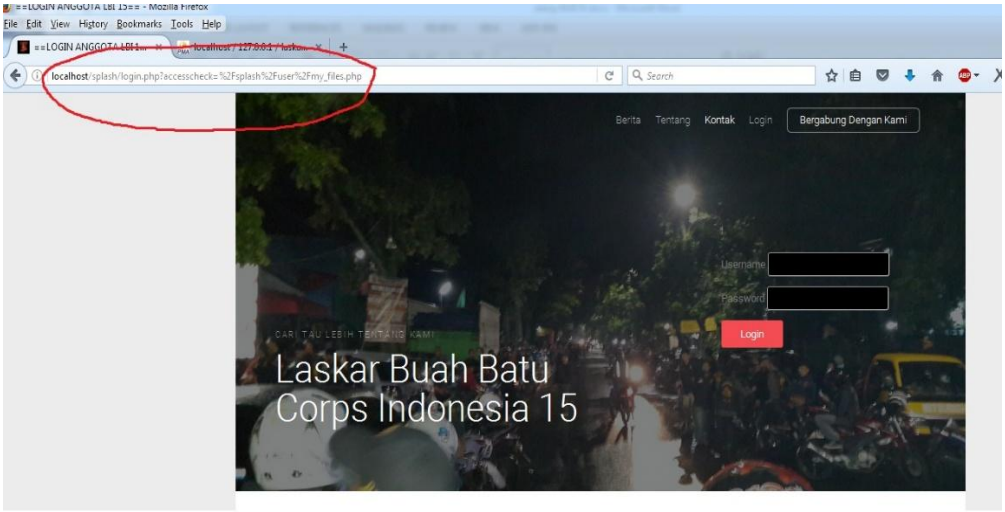
Halaman ini merupakan tampilan pada saat admin salah menginputkan username ataupun passwordnya yang akan memunculkan pemberitahuan gagal login.



Gambar 4.23 Tampilan Salah Input Password Admin

4.5.6.4. Tampilan Gagal Autentifikasi Login Anggota

Halaman ini adalah tampilan saat user gagal dalam melakukan autentifikasi yang mana menyebabkan tidak bisa masuk ke halaman home anggota dan dialihkan ke halaman login anggota dikarenakan hak akses dibatasi oleh sistem.



Gambar 4.24. Tampilan Salah Input Password Anggota

4.5.6.5 Halaman Pendaftaran Anggota

Tampilan ini adalah halaman dari pendaftaran user yang ingin bergabung ke dalam organisasi Laskar Buah Batu Corps Indonesia secara online, namun user yang ingin bergabung tersebut tetap harus menyerahkan formulir fisik yang disediakan oleh sekretariat tiap wilayah untuk ditandatangi langsung oleh calon anggota dan ketua di wilayah tersebut.



Gambar 4.25 Form Pendaftaran Anggota