

BAB IV

DESKRIPSI PEKERJAAN

4.1 Membuat Prosedur Kerja

Dinas Komunikasi Dan Informatika Surabaya saat ini belum memiliki media penyebaran informasi seperti Dinas Pemerintahan yang lain. Selama ini penyebaran informasi masih menggunakan cara selebaran ataupun melalui surat-surat ke kecamatan-kecamatan yang ada di Surabaya, hal ini mengakibatkan proses bisnis Komunikasi Dan Informatika Surabaya kurang maksimal karena keterbatasan media penyebarluasan informasi terhadap masyarakat Surabaya.

Kerja praktek ini dilakukan selama 180 jam dengan pembagian waktu dalam satu hari kerja sebanyak 8 jam kerja sehari. Dalam kerja praktek ini diharuskan menemukan permasalahan yang ada, mempelajari permasalahan serta memberi solusi kepada masalah yang ada.

Untuk mengatasi permasalahan tersebut diperlukan langkah-langkah pembagian kerja sebagai berikut :

- a. Minggu 1, melakukan survey dan mengumpulkan data.
- b. Minggu 2, analisa kebutuhan dan merancang sistem.
- c. Minggu 3, mendesain dan membuat sistem.
- d. Minggu 4, *teting* dan implementasi sistem.

keempat langkah tersebut, dilakukan agar dapat menemukan solusi dari permasalahan yang ada. Lebih jelasnya dipaparkan pada sub bab dibawah ini.

4.2 Melakukan Survey dan Mengumpulkan Data

Survey dan pengumpulan data merupakan langkah awal dalam membuat suatu sistem baru, yaitu dengan melakukan wawancara secara langsung. Wawancara dilakukan oleh satu orang dengan bagian Aplikasi dan Telematika (Aptel) . Bagian Aptel memberikan informasi tentang mekanisme yang diperlukan dalam pembuatan sistem informasi *Company Profile* untuk Dinas Komunikasi Dan Informatika Surabaya.

4.3 Analisa Kebutuhan Dan Merancang Sistem

Dari wawancara diatas, maka dapat diketahui sistem apa yang dibutuhkan oleh Dinas Komunikasi Dan Informatika Surabaya untuk mengatasi permasalahan yang ada. Analisa kebutuhan sistem diambil berdasarkan data yang diperoleh pada saat survey ke bagian Aplikasi Dan Telematika.

4.4 Mendesain dan Membuat Sistem

Dalam mendesain dan membuat sistem informasi ini menggunakan permodelan sistem antara lain *Use Case Diagram*, *Activity Diagram* , dan *Class Diagram* sesuai dengan analisa kebutuhan dan rancangan sistem.

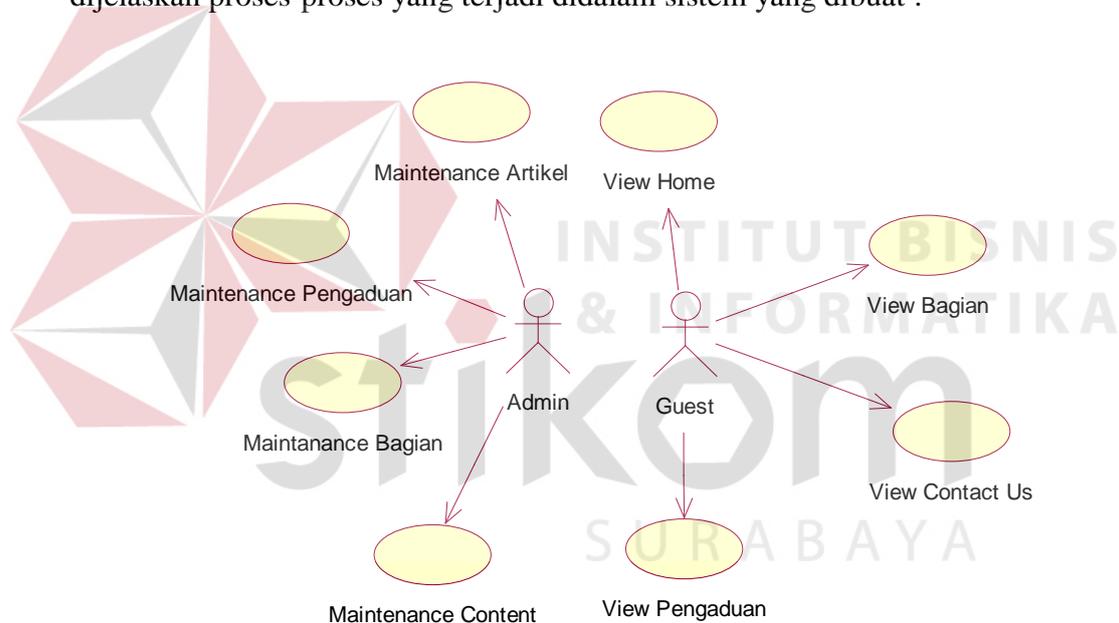
Tahapan yang dilakukan dalam mendesain dan membuat sistem informasi berbasis *object-oriented* adalah sebagai berikut :

1. Membuat *Use Case Diagram* sistem informasi yang menggambarkan hubungan antara aktor dengan sistem.

2. Membuat *Activity Diagram* yang menggambarkan logika prosedural, proses bisnis, dan jalur kerja dari *use case* yang telah di tampilkan sebelumnya.
3. Membuat *Class Diagram* yang menggambarkan hubungan objek-objek yang digunakan dalam sistem ini.

4.4.1 Membuat Use Case Diagram

Pada gambar *use case diagram* sistem informasi berbasis *web* ini dijelaskan proses-proses yang terjadi didalam sistem yang dibuat :



Gambar 4.1 Use Case Diagram Berbasis Web

Dari gambar *use case diagram* diatas terdapat dua aktor yang terlibat di dalam sistem. Kedua aktor tersebut masing-masing berperan sebagai *admin* dan *guest*. *Guest* bertindak sebagai orang yang dapat melakukan aktifitas seperti melihat informasi dari pengaduan, *contact us*, bagian, *content* dan artikel serta

dapat menghubungi pihak Dinas Komunikasi Dan Informatika Surabaya melalui halaman pengduan jika terjadi masalah yang berkaitan dengan Dinas Komunikasi Dan Informatika Surabaya. *Administrator/Admin* sistem adalah pihak yang bertanggung jawab atas *maintenance* data dan informasi pada sistem. Pada gambar *use case diagram* diatas juga terdapat 8 *use case* yaitu :

1. View Content.

Merupakan halaman yang berisi tentang Dinas Komunikasi Dan Informatika Surabaya seperti visi dan misi, maupun informasi lainnya yang *up to date*.

2. View Arikel

Menampilkan aritekkel-artikel berita yang berkaitan dengan Dinas Komunikasi Dan Informatika Surabaya yang *up to date*.

3. View Bagian

Merupakan halaman yang berisikan tentang detail bagian-bagian yang terdapat di Dinas Komunikasi Dan Informatika Surabaya.

4. View Pengaduan

Merupakan halaman yang berisikan tempat pengaduan masyarakat tentang masalah-masalah yang berkaitan dengan Dinas Komunikasi Dan Informatika Surabaya.

5. Maintenance Content

Use Case ini digunakan untuk meng*update content* yang berkaitan dengan Dinas Komunikasi Dan Informatika Surabaya.

6. Maintenance Artikel

Use Case ini digunakan untuk *mengupdate* artikel/berita yang berkaitan dengan Dinas Komunikasi Dan Informatika Surabaya.

7. Maintenance Bagian

Use case ini digunakan untuk *mengupdate* bagian yang ada di Dinas Komunikasi Dan Informatika Surabaya seperti *mengupdate* kepala bagian, keterangan tentang bagian yang ada.

8. Maintenance Pengaduan

Use Case ini digunakan untuk *mengupdate* sistem pengaduan masyarakat yang ada pada *web* ini, proses ini dilakukan oleh *admin* .

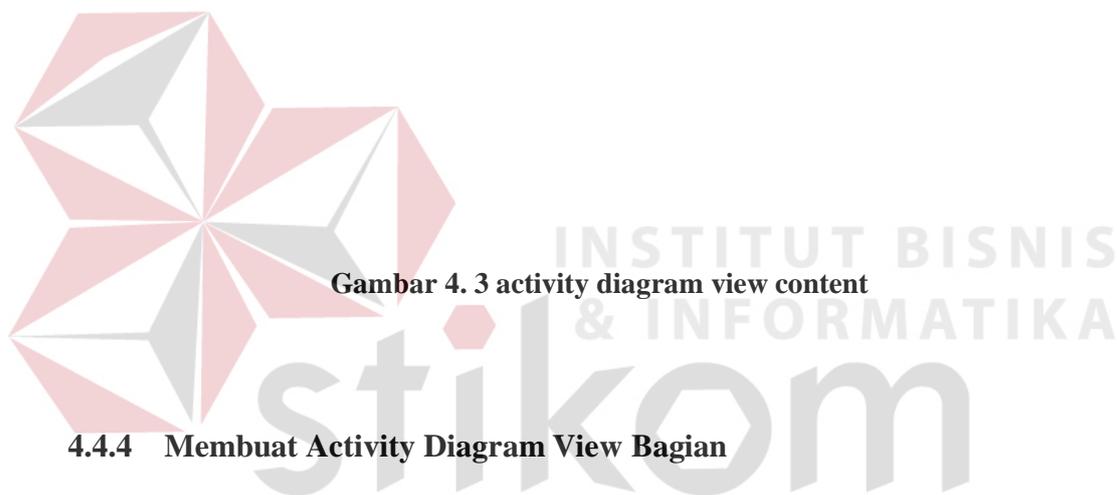
4.4.2 Membuat Activity Diagram View Artikel

Activity diagram view artikel berasal dari guest memilih berita artikel yang ada di sebelah kanan di tampilan *home*. Yang kemudian untuk membaca lebih rinci dapat memilih *more*.

Gambar 4. 2 activity diagram view Artikel

4.4.3 Membuat Activity Diagram View Content

Activity diagram view content berawal dari guest memilih content yang tersedia di halaman *Home*, setelah itu *guest* memilih icon *more* yang kemudian sistem akan menampilkan rincian *detail content*.



Gambar 4. 3 activity diagram view content

4.4.4 Membuat Activity Diagram View Bagian

Activity diagram view bagian berawal dari guest yang memilih menu bagian, dihalaman bagian terdapat bagan yang berisikan bagan struktur bagian yang ada di Dinas Komunikasi Dan Informatika Surabaya, setelah itu *guest* memilih bagian yang dikehendaki lalu sistem akan menampilkan detail bagian yang ditelaah dipilih *guest*.

Gambar 4. 4 activity diagram view bagian

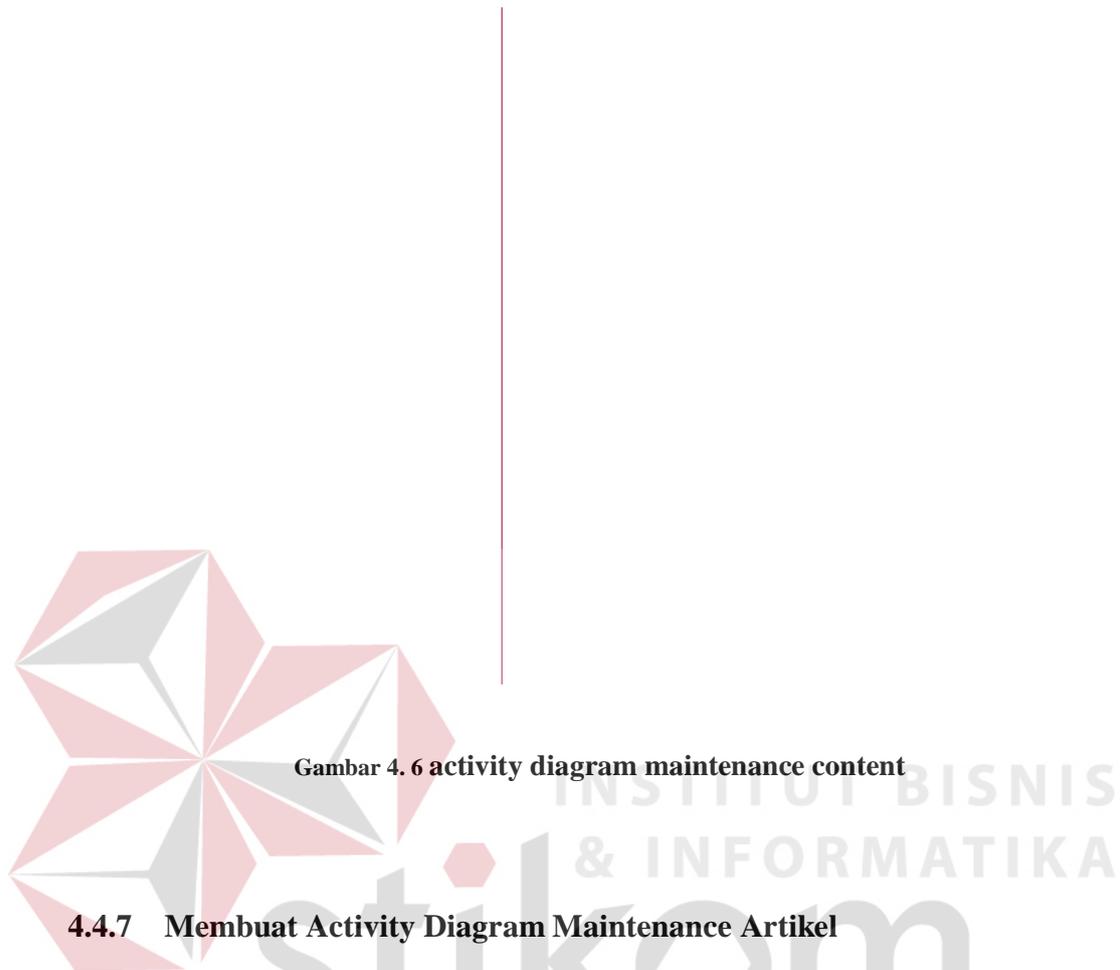
4.4.5 Membuat Activity Diagram View Pengaduan

Activity diagram view pengaduan berawal dari *guest* memilih menu pengaduan, kemudian sistem akan menampilkan data pengaduan, dihalaman tersebut *guest* diminta untuk mengisikan nama dan alamat *email* yang digunakan untuk pendataan, yang selanjutnya *guest* diharuskan mengisi keluhan yang ingin dilaporkan. Kemudian sistem akan menyimpan apa yang sudah diinputkan oleh *guest* tersebut.

Gambar 4. 5 activity diagram view pengaduan

4.4.6 Membuat Activity Diagram Maintenance Content

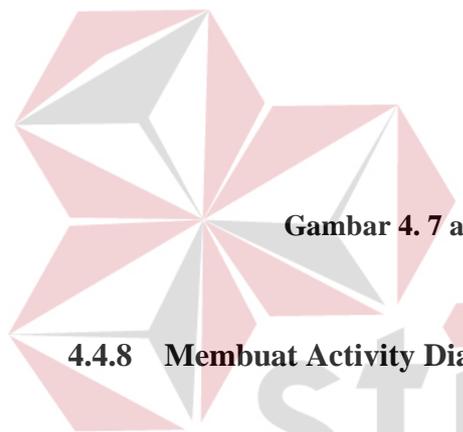
Acitivity diagram maintenance content berawal dari *admin* mengakses *page admin* pada *website*, kemudian *admin* harus melakukan verifikasi *login* terlebih dahulu yang selanjutnya *admin* memilih menu *view data*, setelah itu sistem akan muncul menu memilih data *content*. Setelah *admin* melakukan proses *update* maka sistem akan melakukan penyimpanan yang dimana akan secara otomatis dilakukan *update* pada *database* sistem.



Gambar 4.6 activity diagram maintenance content

4.4.7 Membuat Activity Diagram Maintenance Artikel

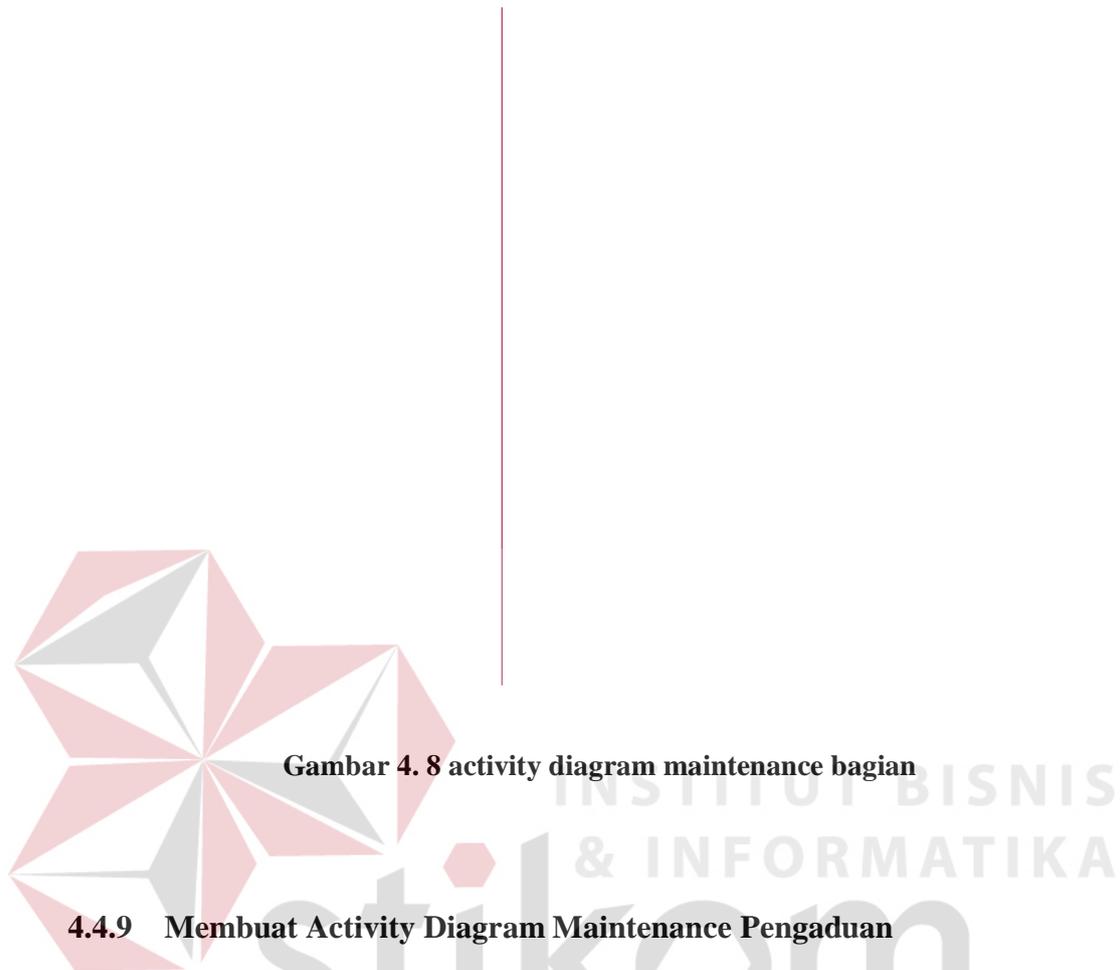
Acitivity diagram maintenance artikel berawal dari *admin* mengakses *page admin* pada *website*, kemudian *admin* harus melakukan verifikasi *login* terlebih dahulu yang selanjutnya *admin* memilih menu *view data*, setelah itu sistem akan muncul menu memilih data artikel. Setelah *admin* melakukan proses *update* maka sistem akan melakukan penyimpanan yang dimana akan secara otomatis dilakukan *update* pada *database* sistem.



Gambar 4. 7 activity diagram maintenance artikel

4.4.8 Membuat Activity Diagram Maintenance Bagian

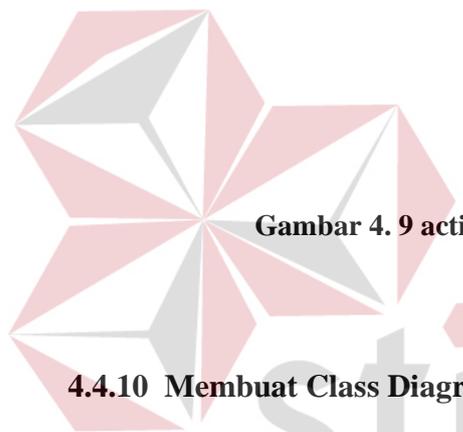
Acitivity diagram maintenance bagian berawal dari *admin* mengakses page *admin* pada *website*, kemudian *admin* harus melakukan verifikasi *login* terlebih dahulu yang selanjutnya *admin* memilih menu *view* data, setelah itu sistem akan muncul menu memilih data bagian. Setelah *admin* melakukan proses *update* maka sistem akan melakukan penyimpanan yang dimana akan secara otomatis dilakukan *update* pada *database* sistem.



Gambar 4. 8 activity diagram maintenance bagian

4.4.9 Membuat Activity Diagram Maintenance Pengaduan

Acitivity diagram maintenance pengaduan berawal dari *admin* mengakses page *admin* pada *website*, kemudian *admin* harus melakukan verifikasi *login* terlebih dahulu yang selanjutnya *admin* memilih menu *view* data, setelah itu sistem akan muncul menu memilih pengaduan. Setelah *admin* melakukan proses *update* maka sistem akan melakukan penyimpanan yang dimana akan secara otomatis dilakukan *update* pada *database* sistem.



Gambar 4. 9 activity diagram maintenance Pengaduan

4.4.10 Membuat Class Diagram Login (*Admin*)

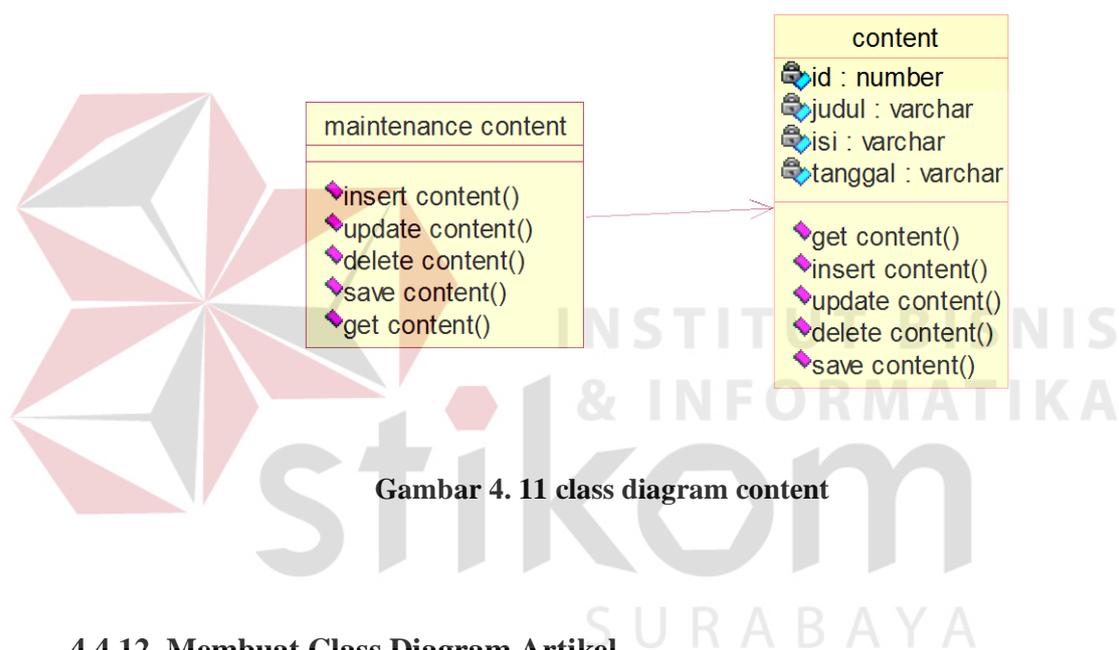
Class diagram login menggambarkan *class login verifcator* yang digunakan untuk melakukan verifikasi atau memeriksa *inputan user* pada saat *login*.

Gambar 4. 10 class diagram Login

4.4.11 Membuat Class Diagram Content

Class diagram digunakan untuk menggambarkan desain statis dari sistem yang sedang dibangun. Berikut ini terdapat beberapa *class diagram* yang digunakan untuk menyusun sistem informasi berbasis *web*.

Class diagram content menggambarkan hubungan antara beberapa *class* yang terdapat dalam satu *package content*.

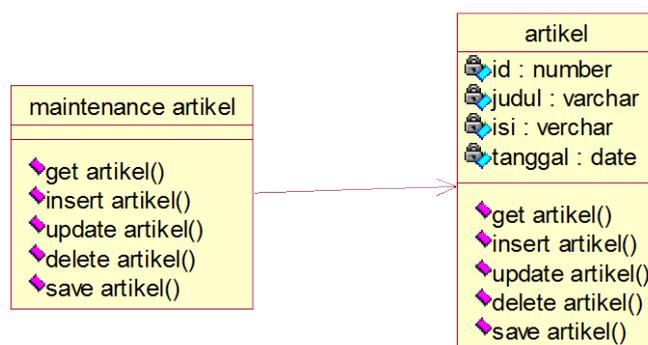


Gambar 4. 11 class diagram content

4.4.12 Membuat Class Diagram Artikel

Class diagram digunakan untuk menggambarkan desain statis dari sistem yang sedang dibangun. Berikut ini terdapat beberapa *class diagram* yang digunakan untuk menyusun sistem informasi berbasis *web*.

Class diagram Artikel menggambarkan hubungan antara beberapa *class* yang terdapat dalam satu *package* artikel.

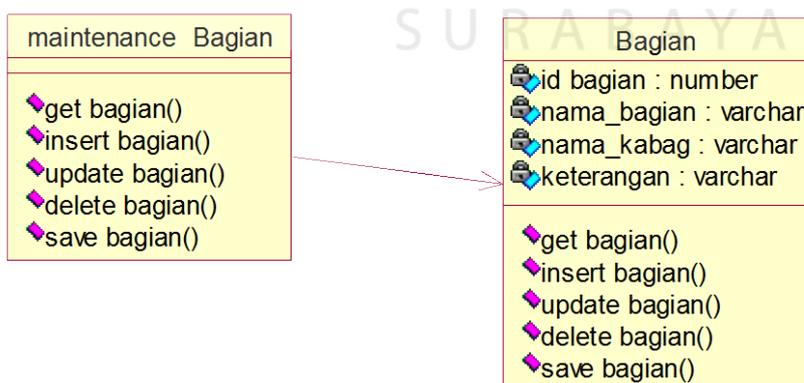


Gambar 4. 12 class diagram artikel

4.4.13 Membuat Class Diagram Bagian

Class diagram digunakan untuk menggambarkan desain statis dari sistem yang sedang dibangun. Berikut ini terdapat beberapa *class diagram* yang digunakan untuk menyusun sistem informasi berbasis *web*.

Class diagram bagian menggambarkan hubungan antara beberapa class yang terdapat dalam satu *package* bagian.



Gambar 4. 13 class diagram bagian

4.5 Struktur Tabel

Perancangan tabel yang digunakan berdasarkan *class diagram* yang telah dibuat adalah sebagai berikut :

1. Tabel Content

Nama Tabel : Content

Fungsi : Menyimpan data content pada *website*

Primary key : Id

Foreign key : -

Tabel 1 Table Content

NO	NAMA FIELD	TIPE	PANJANG	KETERANGAN
1	ID	NUMBER		Id content
2	JUDUL	VARCHAR	50	Judul Content
3	ISI	VARCHAR	4000	Isi Content
4	TGL	DATE	-	Tanggal diposting

2. Tabel Artikel

Nama Tabel : Artikel

Fungsi : Menyimpan data artikel pada *website*

Primary key : Id

Foreign key : -

Tabel 2 Tabel Artikel

NO	NAMA FIELD	TIPE	PANJANG	KETERANGAN
1	ID	NUMBER		Id Artikel
2	JUDUL	VARCHAR	50	Judul Artikel
3	ISI	VARCHAR	4000	Isi Artikel
4	TGL	DATE	-	Tanggal diposting

c) Tabel Bagian

Nama Tabel : Bagian

Fungsi : Menyimpan data Bagian pada *website*

Primary key : Id_bagian

Foreign key : -

Tabel 3 Tabel Bagian

NO	NAMA FIELD	TIPE	PANJANG	KETERANGAN
1	ID_BAGIAN	NUMBER		Id Bagian
2	NAMA_BAGIAN	VARCHAR	50	Judul Bagian
3	NAMA_KABAG	VARCHAR	50	Isi Bagian
4	KETERANGAN	VARCHAR	4000	KETERANGAN BAGIAN

d) Tabel Login

Nama Tabel : Login

Fungsi : Menyimpan data Login pada *website*

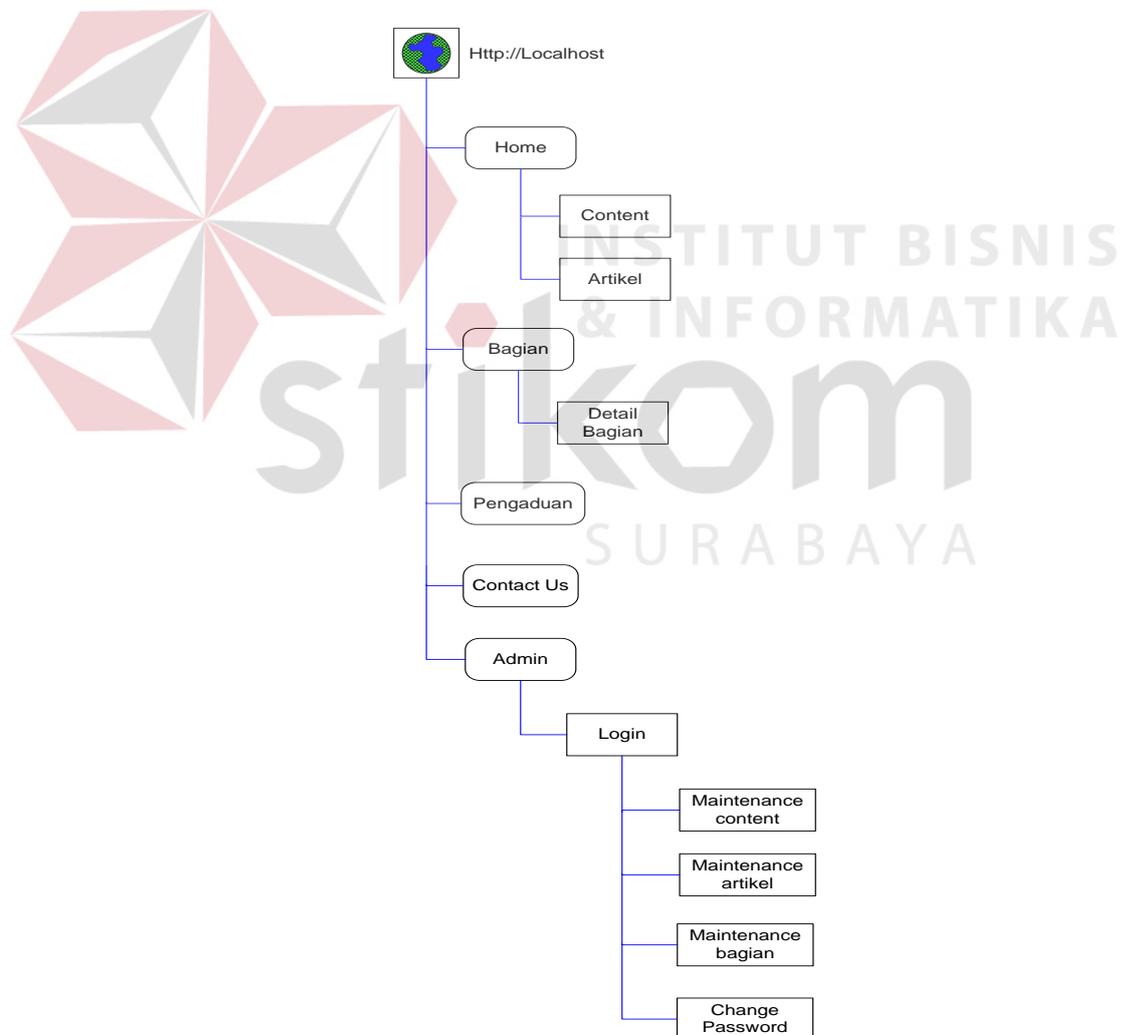
Primary key : username

Foreign key :-

Tabel 4 Tabel login

NO	NAMA FIELD	TIPE	PANJANG	KETERANGAN
1	USERNAME	VARCHAR	25	username
2	PASSWORD	VARCHAR	25	Password

4.6 Site Map



Gambar 4. 14 site Map

4.7 Desain Input/Output

Desain *input/output* dari Sistem Informasi *Web Company Profile Dinas Komunikasi Dan Informatika Surabaya* adalah sebagai berikut :

a. Halaman Login

Halaman *Login* merupakan halaman *web* yang didesain untuk melakukan proses autentifikasi *admin* pada *web site* .



Gambar 4. 15 desain Halaman Login

b. Halaman Insert Content

Halaman *Insert content* merupakan halaman *web* yang didesain untuk melakukan proses penambahan *content* baru pada *website*.



Insert Content

ID

JUDUL

ISI

Tanggal

Gambar 4. 16 Form insert Content

c. Halaman Insert Artikel

Halaman *Insert Artikel* merupakan halaman *web* yang didesain untuk melakukan proses penambahan Artikel baru pada *website*.



Insert Artikel

ID

JUDUL

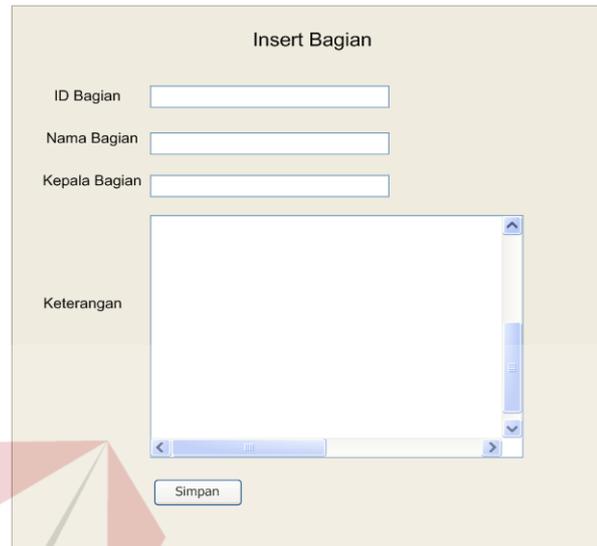
ISI

Tanggal

Gambar 4. 17 Form Insert Artikel

d. Halaman Insert Bagian

Halaman *Insert Bagian* merupakan halaman *web* yang didesain untuk melakukan proses penambahan Bagian baru pada *website*.



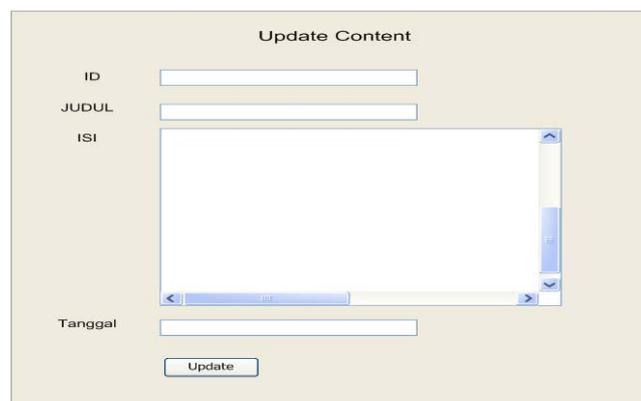
The screenshot shows a web form titled "Insert Bagian". It contains the following fields:

- ID Bagian: A single-line text input field.
- Nama Bagian: A single-line text input field.
- Kepala Bagian: A single-line text input field.
- Keterangan: A large multi-line text area with a vertical scrollbar on the right and a horizontal scrollbar at the bottom.
- Simpan: A button located below the text area.

Gambar 4. 18 Form Insert Bagian

e. Halaman Maintenance Content

Halaman *Maintenance content* merupakan halaman *web* yang didesain untuk melakukan proses penambahan *Content* baru pada *website*.



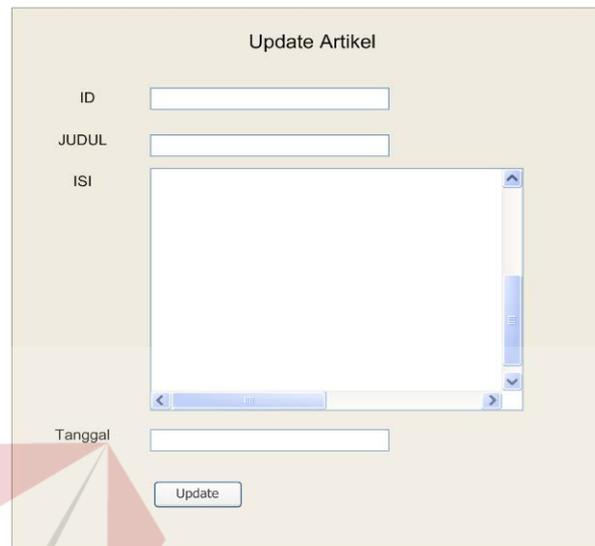
The screenshot shows a web form titled "Update Content". It contains the following fields:

- ID: A single-line text input field.
- JUDUL: A single-line text input field.
- ISI: A large multi-line text area with a vertical scrollbar on the right and a horizontal scrollbar at the bottom.
- Tanggal: A single-line text input field.
- Update: A button located below the text area.

Gambar 4. 19 Maintenance Content

f. Halaman Maintenance Artikel

Halaman *Maintenance* artikel merupakan halaman *web* yang didesain untuk melakukan proses penambahan Artikel baru pada *website*.



The image shows a web form titled "Update Artikel". It contains the following fields and elements:

- ID**: A single-line text input field.
- JUDUL**: A single-line text input field.
- ISI**: A large multi-line text area with a vertical scrollbar on the right side.
- Tanggal**: A single-line text input field.
- Update**: A button located below the "Tanggal" field.

Gambar 4. 20 Form Maintenance Artikel

f. Halaman Maintenance Bagian

Halaman *Maintenance* Bagian merupakan halaman *web* yang didesain untuk melakukan proses penambahan Bagian baru pada *website*.

Update Bagian

ID Bagian

Nama Bagian

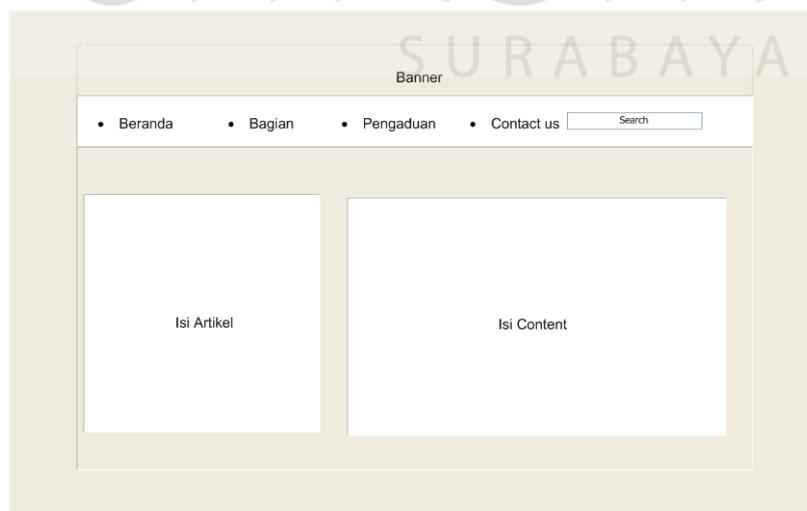
Kepala Bagian

Keterangan

Gambar 4. 21 Form Maintenance Bagian

g. Halaman Home

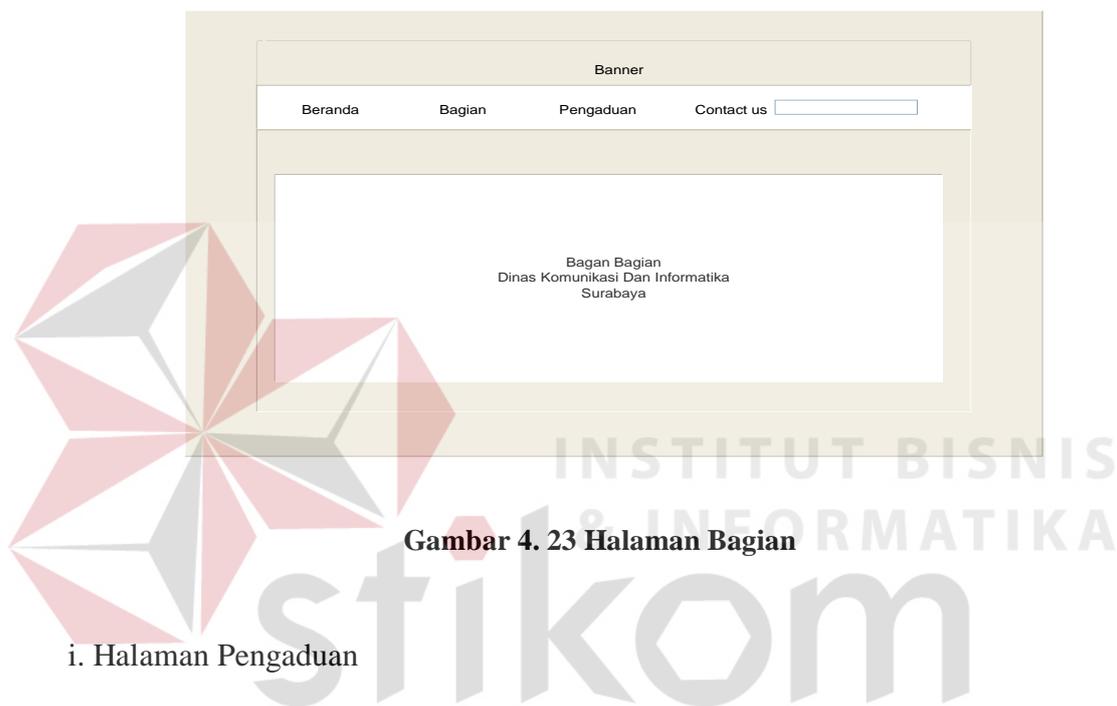
Halaman *Home* merupakan Halaman *web* yang didesain untuk menampilkan halaman utama yang berisi tentang *content* dan artikel yang berkaitan dengan Dinas Komunikasi Dan Informatika Surabaya.



Gambar 4. 22 Halaman Home

h. Halaman Bagian

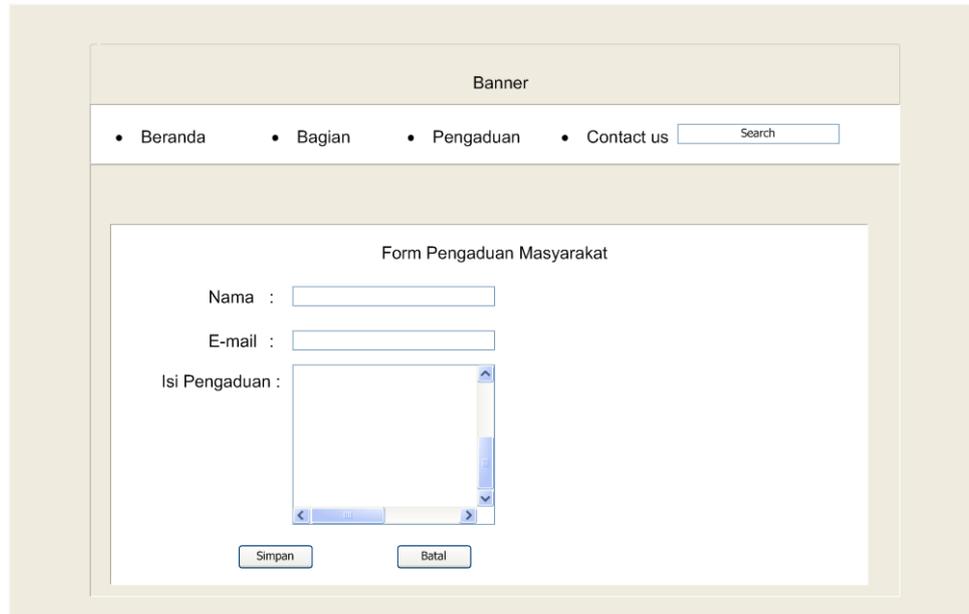
Halaman Bagian merupakan Halaman *web* yang didesain untuk menampilkan bagian yang di Dinas Komunikasi Dan Informatika Surabaya. Tampilannya berupa Bagan yang akan menampilkan Kabag perbagian dan jika di klik akan muncul detail Bagiannya.



Gambar 4. 23 Halaman Bagian

i. Halaman Pengaduan

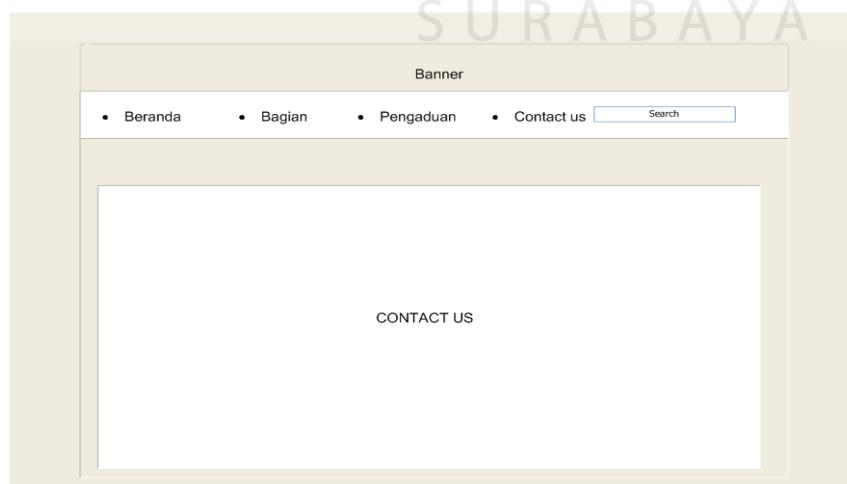
Halaman Pengaduan merupakan Halaman *web* yang didesain untuk menampung pengaduan masalah dari masyarakat Surabaya yang berkaitan dengan Dinas Komunikasi Dan Informatika Surabaya.



Gambar 4. 24 Halaman Pengaduan Masyarakat

j. Halaman *Contact Us*

Halaman Pengaduan merupakan Halaman *web* yang didesain untuk menampung pengaduan masalah dari masyarakat Surabaya yang berkaitan dengan Dinas Komunikasi Dan Informatika Surabaya.



Gambar 4. 25 Halaman Contact Us

4.8 Mengimplementasi sistem

Mengimplementasikan sistem merupakan tahap pengujian dimana desain sistem dapat berjalan dengan baik. Implementasi dilakukan oleh dua orang yaitu penganalisa sistem dan pendesain *input output*. Desain *page* yang telah dibuat oleh pendesaian *input output* cukup sesuai untuk mengimplementasikan sistem, sehingga tidak membutuhkan banyak perubahan.

4.8.1 Kebutuhan sistem

Pada tahap ini dijelaskan mengenai implementasi dari perangkat lunak yang harus disiapkan oleh pengguna.

Adapun perangkat lunak yang digunakan, yaitu :

1. Windows XP Profesional Edition.
2. Net Framework 2.0.
3. Oracle Express edition.

Untuk perangkat keras, minimal pengguna harus mempersiapkan spesifikasi sebagai berikut :

1. Microprocessor Dual Core 2,6 GHz.
2. RAM minimal 256 Mb , disarankan 512 Mb.
3. Kapasitas bebas hardisk minimum adalah 1,8 Gb.
4. VGA monitor.
5. Keyboard.
6. Mouse atau *device* yang kompetibel.
7. Drive CD ROM atau DVD.

8. Peralatan jaringan (*Ethernet Card*, kabel UTP, *Modem* dan *Switch* atau *Hub*).

4.9 Hasil Implementasi

Hasil implementasi yang dilakukan oleh penganalisa sistem dengan pendesain *input output* menghasilkan desain program. Listing program dapat dilihat pada lampiran.

4.10 Melakukan Pembahasan Hasil Implementasi Sistem

4.10.1 Instalasi Program

Dalam tahap ini, pengguna baru harus memperhatikan dengan benar terhadap penginstalan perangkat lunak.

Berikut adalah perangkat lunak yang harus diinstall terlebih dahulu :

1. Net Framework 2.0.
2. Oracle express Edition.

4.10.2 Penjelasan Pemakaian

Tahap ini merupakan langkah-langkah dari pemakaian *Company Profile* Dinas Komunikasi Dan Informatika Surabaya. Berikut sub-sub pembahasan pemakaian :

1. Home Page

“Home Page” merupakan tampilan awal dari *website* ini. Pada halaman ini terdapat menu navigasi yang digunakan untuk menuju kehalaman lain *website* seperti “Bagian”, “Pengaduan”, “Contact Us”. Pada halaman ini juga berisikan artikel serta *content* yang berguna untuk pengunjung *website* ini.



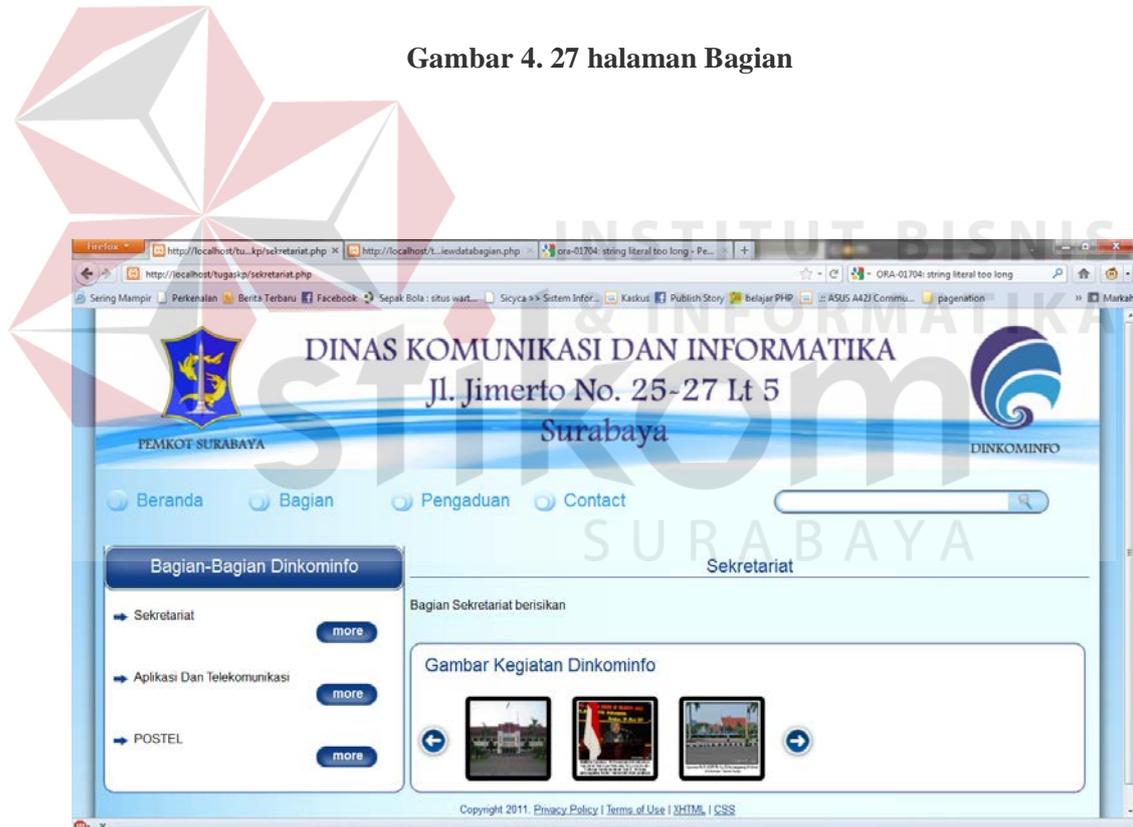
Gambar 4. 26 Halaman Home

2. Halaman Bagian

Halaman “Bagian” berisikan tentang bagan struktur organisasi Dinas Komunikasi dan Informatika Surabaya. Pada halaman ini diberikan *java script* dimana jika pointer diarahkan pada salah satu gambar bagan maka akan muncul siapa kabag yang memimpin bagian tersebut. Bagi yang ingin tahu lebih detail tentang informasi serta susunan pengurus/anggota bagian itu dapat mengklik pada bagan yang diinginkan.



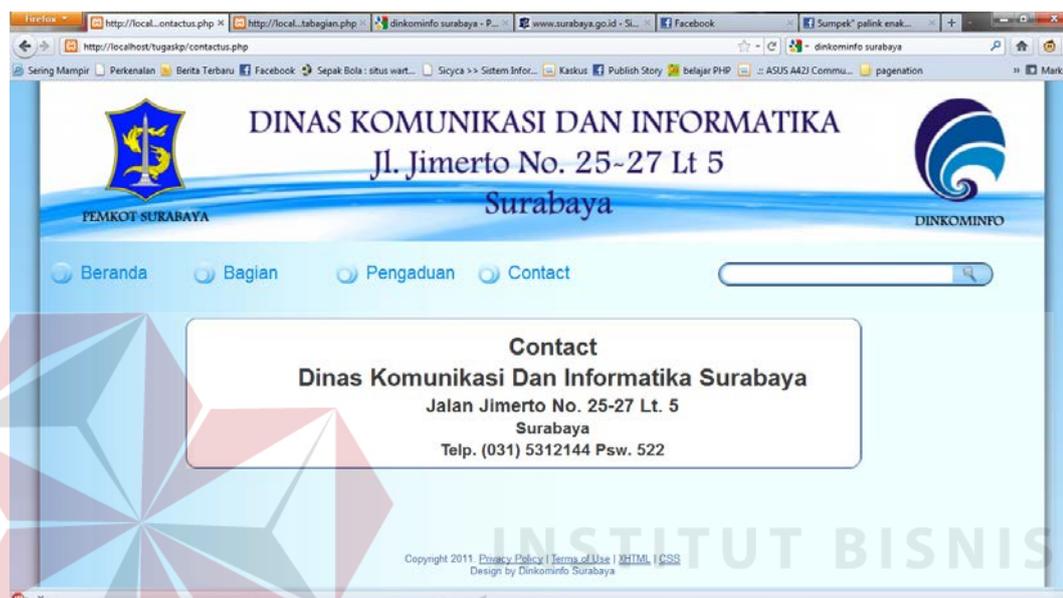
Gambar 4. 27 halaman Bagian



Gambar 4. 28 detail Halaman Bagian

3. Halaman Contact

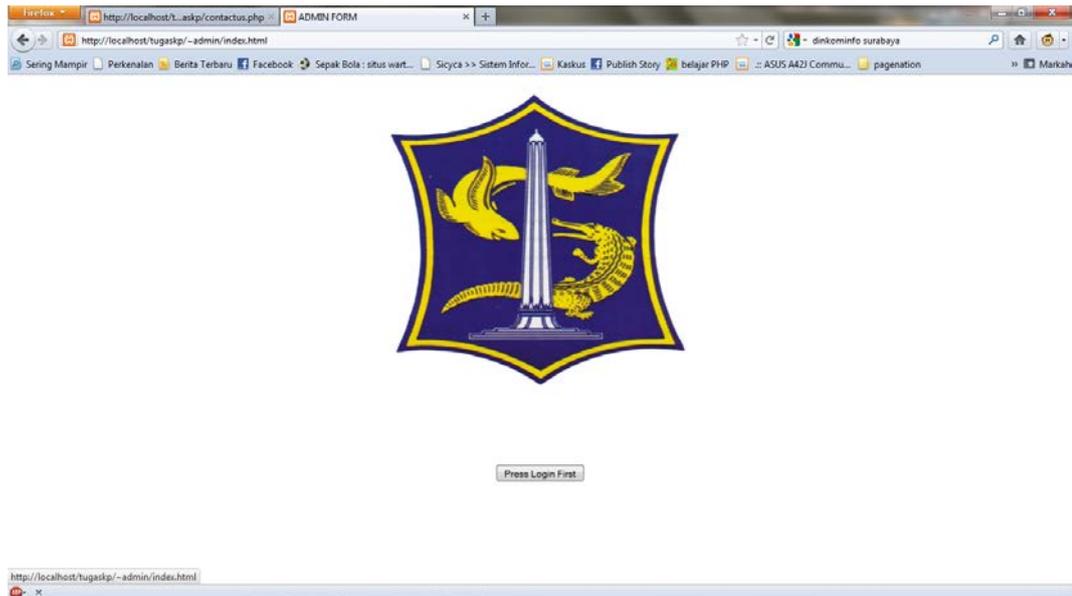
Halaman “Contact” berisikan tentang informasi keberadaan Dinas Komunikasi dan Informatika Surabaya dapat dihubungi. Di halaman ini terdapat pula alamat dan no telepon yang bisa dihubungi jika terjadi sesuatu.



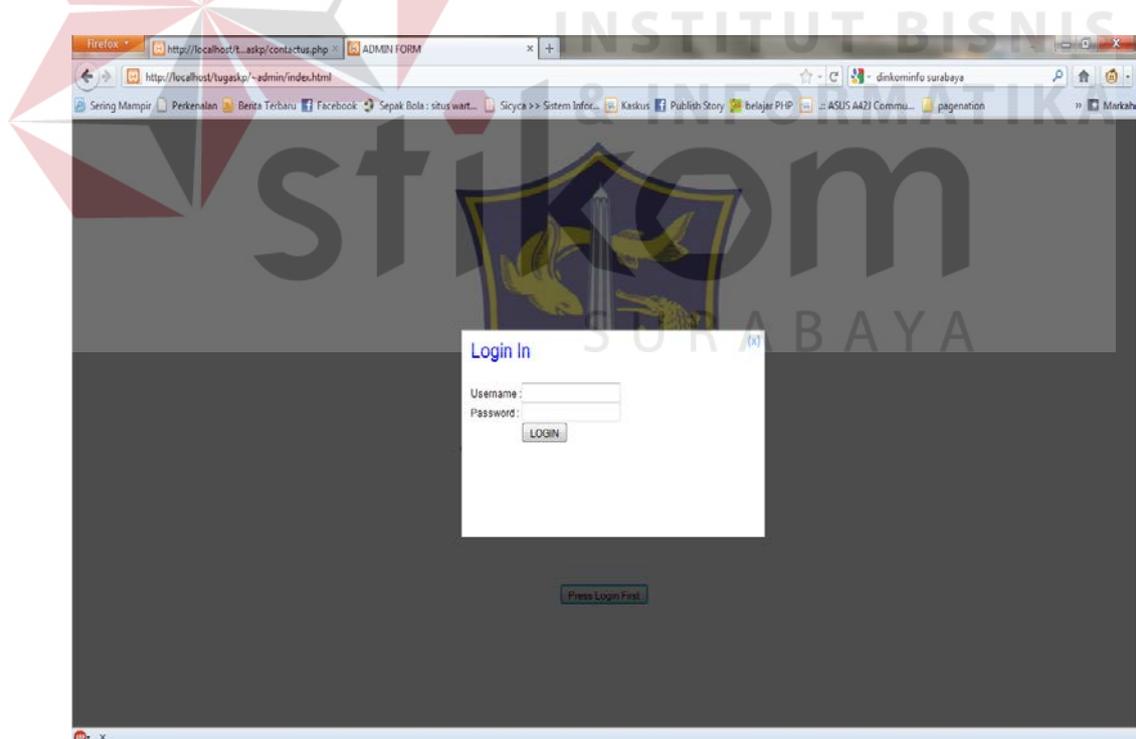
Gambar 4. 29 Halaman Contact

4. Halaman Login Admin Form

Halaman “Login Admin Form” dimana dalam form ini seorang *admin* haruslah melakukan proses login terlebih dahulu. Barulah dapat mengakses form *admin* yang lain.



Gambar 4. 30 Halaman Awal *Admin Form*



Gambar 4. 31 Halaman Login *Admin*

4. Halaman Maintenance Artikel

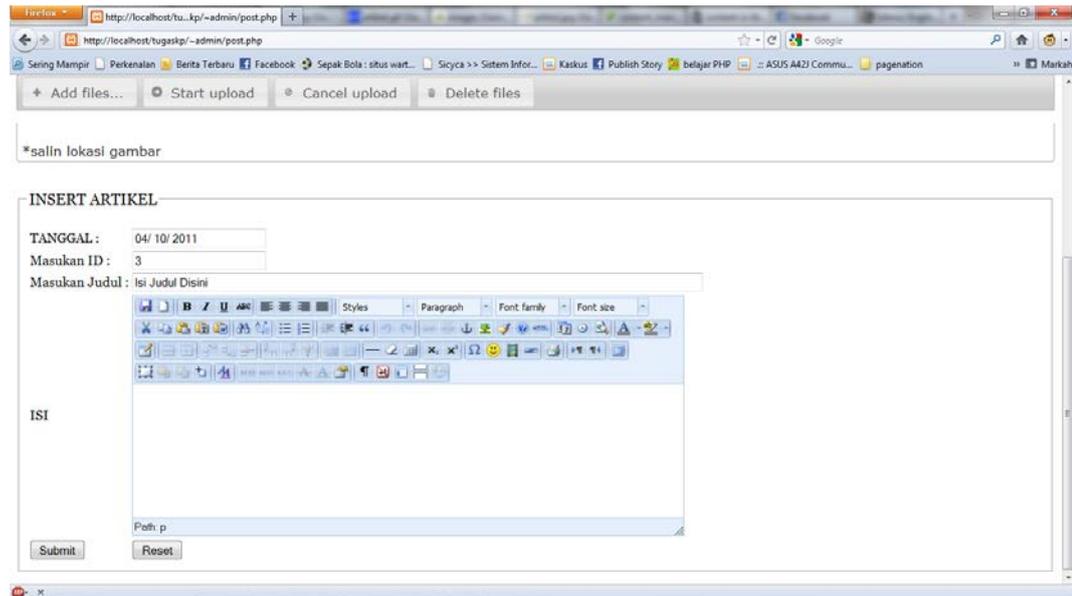
Halaman “Maintenance Artikel” merupakan halaman *web* yang berfungsi untuk melakukan manipulasi data yaitu *insert, update, delete*.

a. Insert

Untuk menambahkan artikel baru pada halaman *admin*, *admin* di haruskan memilih *insert* artikel seperti yang ada pada gambar 4.32. Setelah itu barulah terdapat pilihan untuk mengisi judul, tanggal dan ID yang akan di *generate* secara otomatis.



Gambar 4. 32 halaman pilihan insert



Gambar 4. 33 insert artikel

b. *Update*

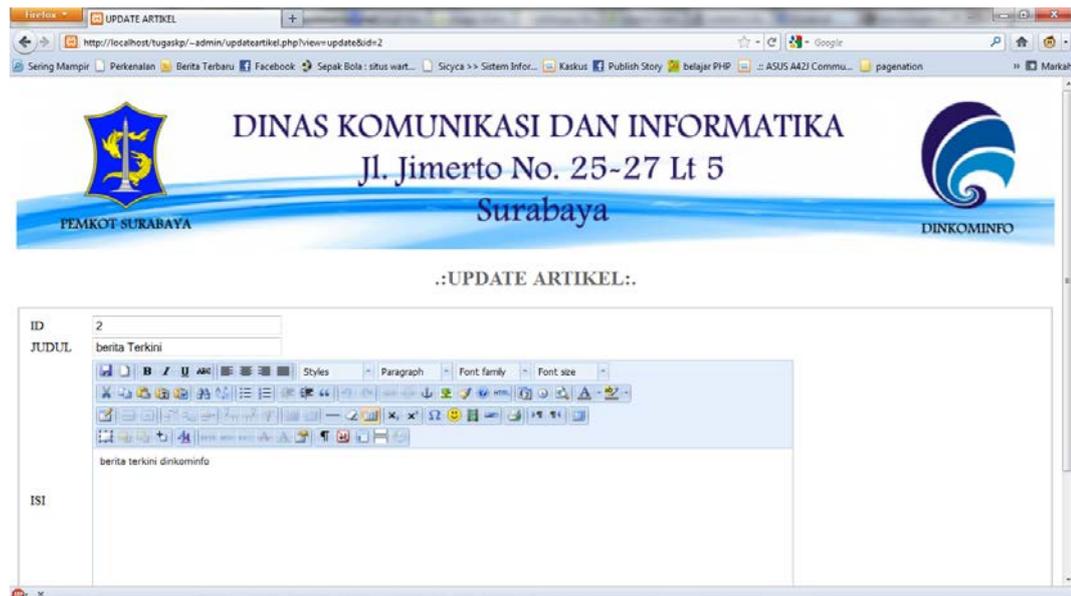
Untuk melakukan *update*, *admin* diminta untuk memilih tab “view your data”. Pada tab tersebut maka akan muncul pilihan data, kemudian pilih data artikel. Dan setelah itu akan ada 2 *button* diantaranya *button* , dan . Untuk melakukan *update* tekan tombol . Seperti pada gambar 4.34.



Gambar 4. 34 update artikel



Gambar 4. 35 edit artikel



Gambar 4. 36 form edit artikel

c. Delete

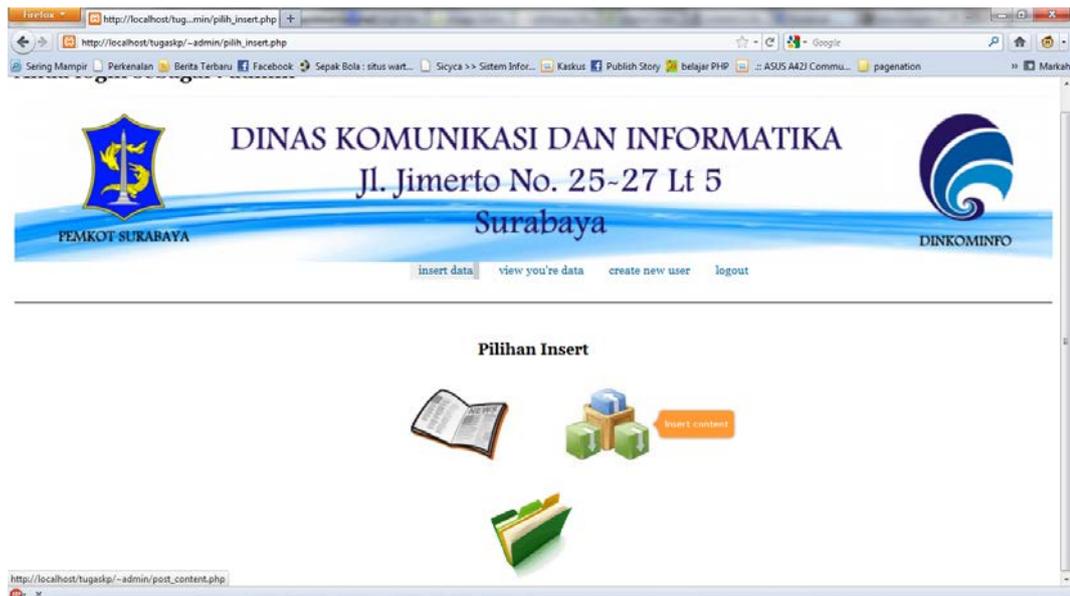
Untuk melakukan proses *delete admin* hanya cukup menekan *button* “✕”, kemudian data akan secara otomatis terhapus dari *database*.

5. Halaman Maintenance Content

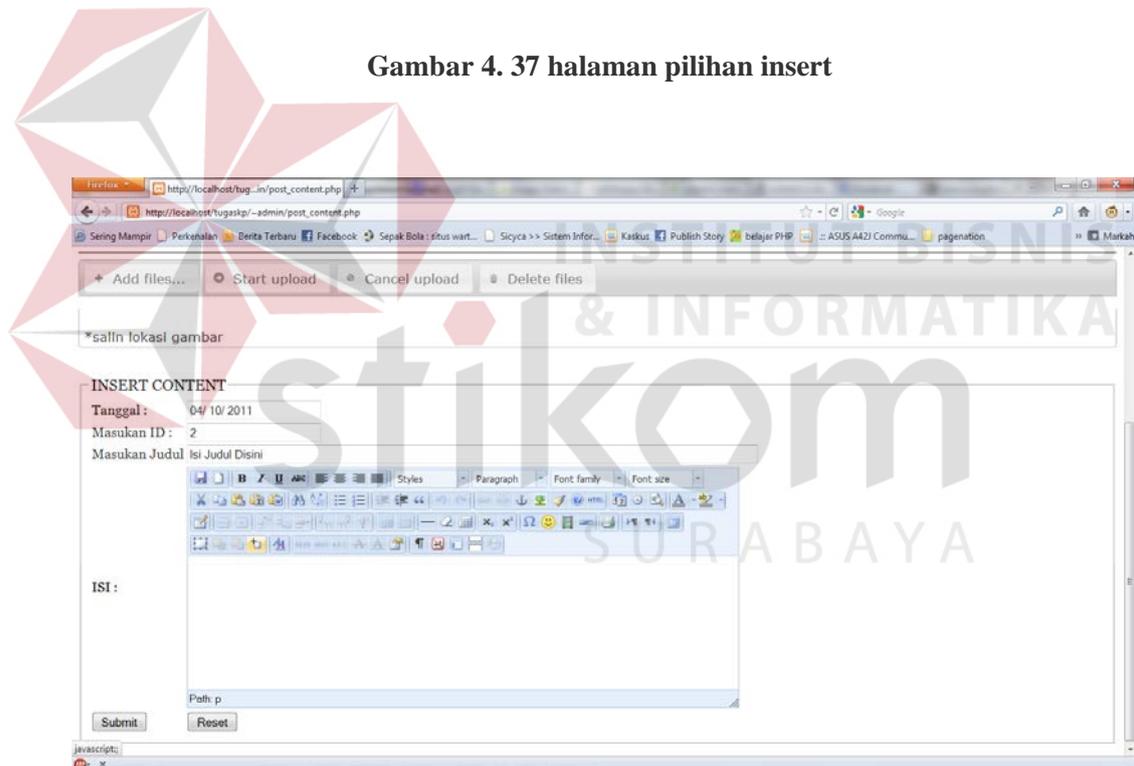
Halaman “Maintenance Content” merupakan halaman *web* yang berfungsi untuk melakukan manipulasi data yaitu *insert, update, delete*.

a. Insert

Untuk menambahkan artikel baru pada halaman *admin*, *admin* di haruskan memilih *insert content* seperti yang ada pada gambar 4.37. Setelah itu barulah terdapat pilihan untuk mengisi judul, tanggal dan ID yang akan di *generate* secara otomatis.



Gambar 4. 37 halaman pilihan insert



Gambar 4. 38 insert content

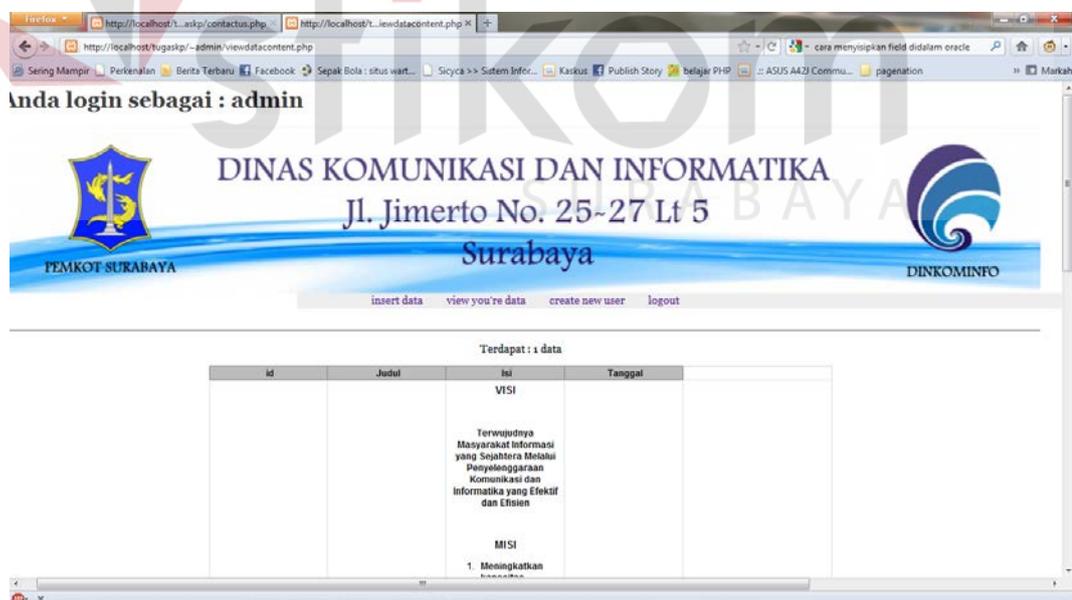
b. *Update*

Untuk melakukan *update*, *admin* diminta untuk memilih tab “view your data”. Pada tab tersebut maka akan muncul pilihan data, kemudian pilih data

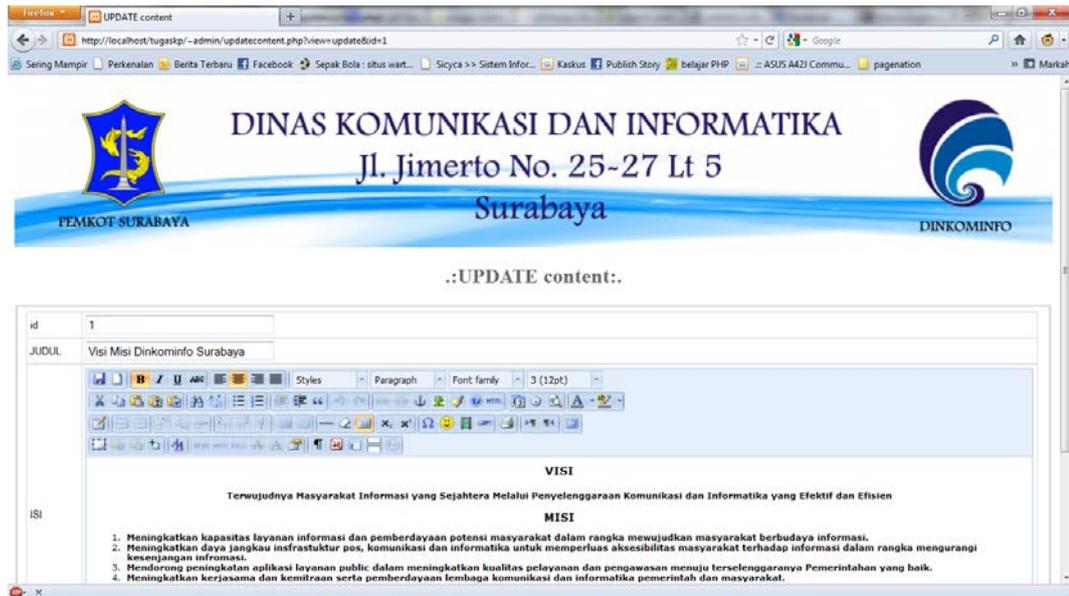
cotent. Dan setelah itu akan ada 2 *button* diantaranya *button* “✎”, dan “✖”. Untuk melakukan *update* tekan tombol “✎”. Seperti pada gambar 4.39.



Gambar 4. 39 pilihan *update*



Gambar 4. 40 edit content



Gambar 4. 41 Form edit Content

c. Delete

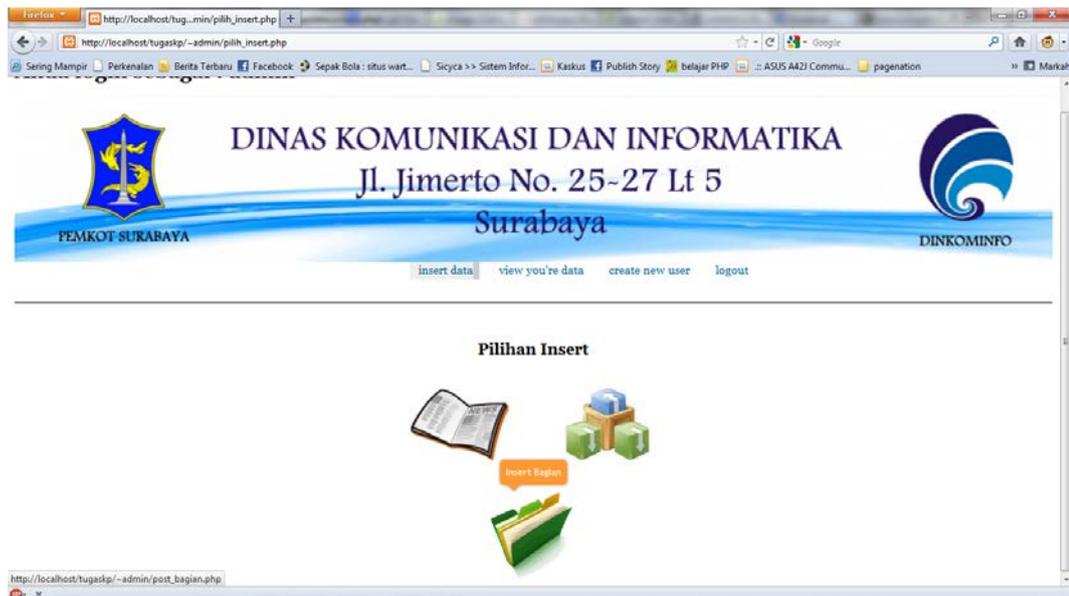
Untuk melakukan proses *delete admin* hanya cukup menekan *button* “✕”, kemudian data akan secara otomatis terhapus dari *database*.

6. Halaman Maintenance Bagian

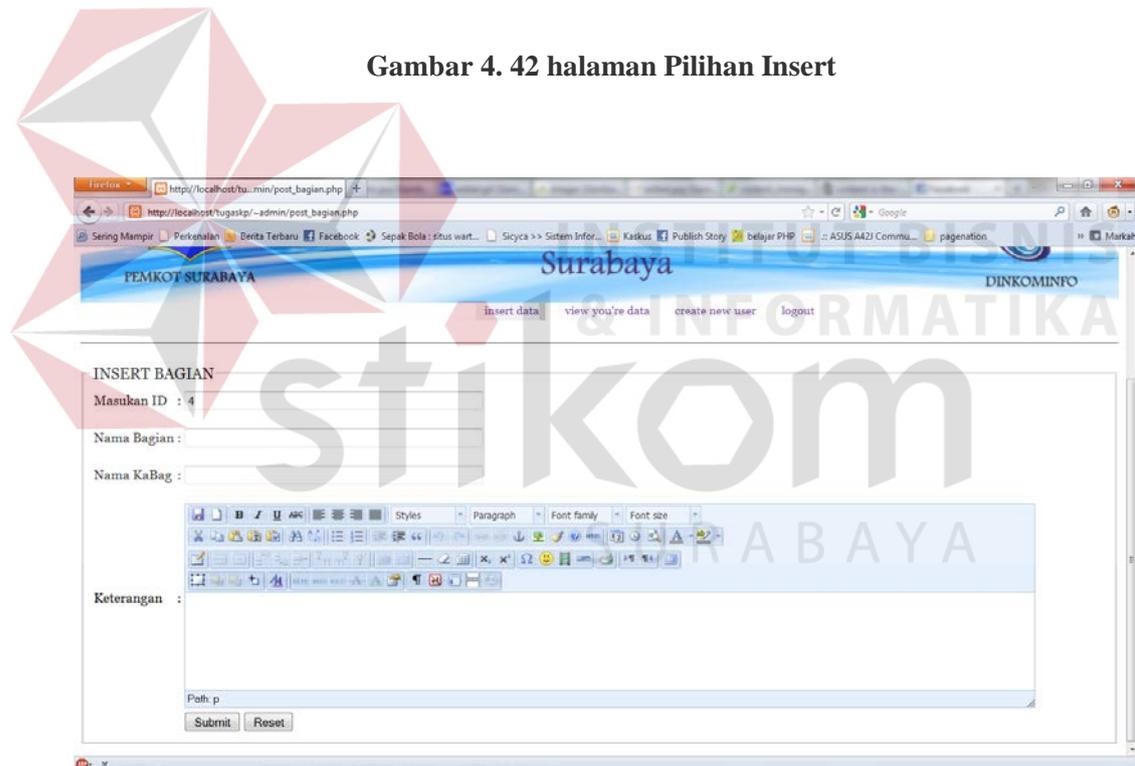
Halaman “Maintenance Bagian” merupakan halaman *web* yang berfungsi untuk melakukan manipulasi data yaitu *insert*, *update*, *delete*.

a. Insert

Untuk menambahkan artikel baru pada halaman *admin*, *admin* di haruskan memilih *insert* bagian seperti yang ada pada gambar 4.42. Setelah itu barulah terdapat pilihan untuk mengisi nama bagian, nama kabag, keterangan dan ID yang akan di *generate* secara otomatis.



Gambar 4. 42 halaman Pilihan Insert

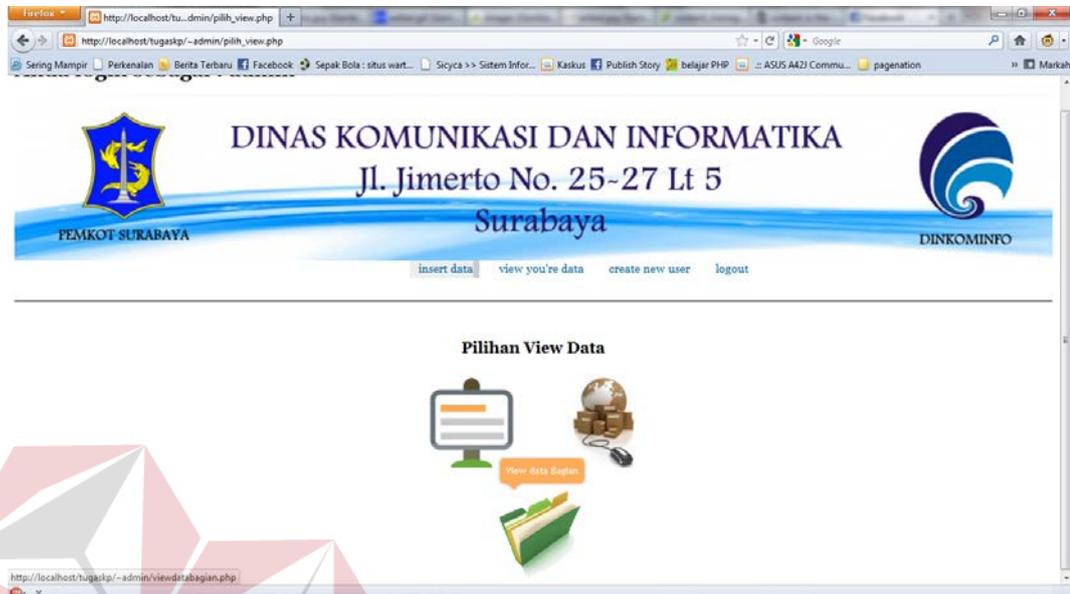


Gambar 4. 43 Insert Bagian

b. *Update*

Untuk melakukan *update*, *admin* diminta untuk memilih tab “view your data”. Pada tab tersebut maka akan muncul pilihan data, kemudian pilih data

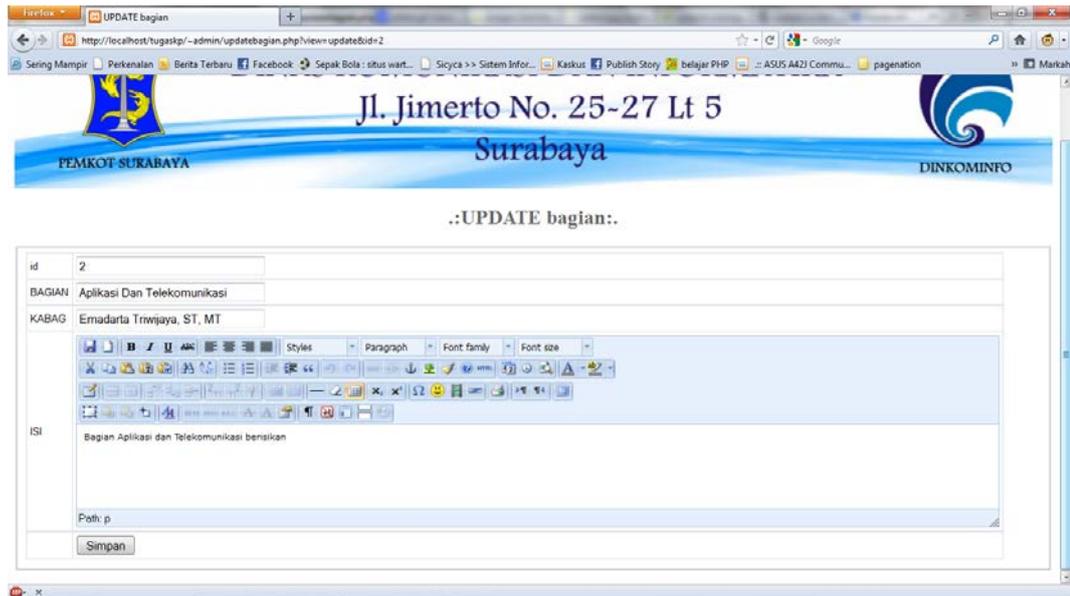
content. Dan setelah itu akan ada 2 *button* diantaranya *button* “✎”, dan “✖”. Untuk melakukan *update* tekan tombol “✎”. Seperti pada gambar 4. 44.



Gambar 4. 44 Pilihan *Update*



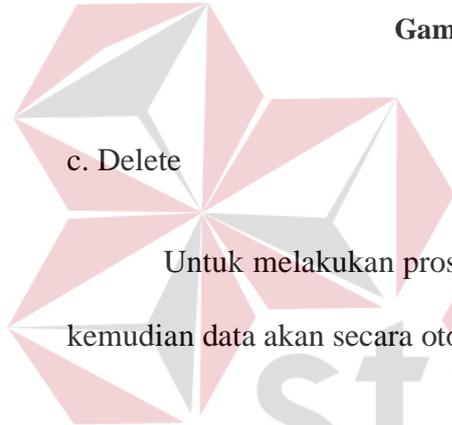
Gambar 4. 45 Edit Bagian



Gambar 4. 46 Form Edit Bagian

c. Delete

Untuk melakukan proses *delete admin* hanya cukup menekan *button* “✕”, kemudian data akan secara otomatis terhapus dari *database*.



INSTITUT BISNIS
& INFORMATIKA
stikom
SURABAYA