BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sistem bertujuan untuk mengidentifikasi permasalahan permasalahan yang ada pada sistem. Analisa ini diperlukan sebagai dasar untuk tahapan perancangan sistem. Analisis sistem meliputi identifikasi masalah, spesifikasi aplikasi, spesifikasi pengguna dan lingkungan operasi.

3.1.1 Analisis Masalah

Masalah yang dihadapi dalam membangun aplikasi ini adalah sebagai berikut:

a. Bagaimana membangun SIG berdasarkan data spasial dan data non-spasial yang diperoleh.
b. Bagaimana cara menentukan titik koordinat lokasi wisata secara akurat.
c. Bagaimana supaya sistem yang akan di bangun dapat dimengerti oleh pengguna, sehingga informasi yang diberikan memberikan hasil yang optimal.

3.1.2 Ruang Lingkup Masalah

Ruang lingkup masalah yang dibahas dalam membangun aplikasi ini yaitu memberikan layanan informasi kepada pengguna yang membutuhkan informasi lokasi dan informasi penting lainnya mengenai tempat pariwisata di Tasikmalaya. Fitur-fitur utama aplikasi ini adalah sebagai berikut:

a. Pencarian lokasi pariwisata

b. Menampilkan informasi-informasi penting suatu tempat pariwisata
c. Memberikan informasi rute kendaraan menuju objek wisata
d. Pengolahan data pariwisata

3.1.3 Analisis Data

Data yang digunakan dalam GIS berupa data spasial dan data atribut. Oleh karena itu penulis mengelompokkannya berdasarkan kategori tersebut.

3.1.3.1 Data Spasial

3.1.3.1.1 Analis Google Maps API

Google Maps adalah layanan Peta gratis dari perusahaan Google Inc, Google memngizinkan siapapun yang dapat menambahkan fitur Google Maps dalam web mereka sendiri dengan *Google Maps API*.

Dalam hal kecepatan pencarian data, google sangat diunggulkan oleh semua pemakai internet, maka google maps sangat optimal dan handal dalam hal pencarian data.

Elemen – elemen penting pada Google Map API adalah :

1. *Loading Google Maps API*

   Koneksi *script* yang kita buat ke *server* Google Map API dengan menggunakan *key* yang anda dapatkan pada saat anda mendaftar ke Google Maps API.

2. *Map DOM Elements*

   Untuk menampilkan *map* pada halaman web, umumnya menggunakan div tag untuk menampungnya. Diharuskan membuat sebuah *div tag*

dengan nama "*map_canvas*", dan mereferensikan *element* ini ke *Document Object Model* (DOM)

3. GMap2 - *Elementary Object*

   Class javascript yang membuat peta itu ada adalah *class* GMap2, *Object* dari *class* ini akan menyediakan sebuah peta di halaman web, *Variable map* akan berikan nilai sebuah *object* dari class GMap2. Fungsi GMap2 adalah sebagai *constructor* dan definisinya.

4. Inisialisasi *Map*

   nisialisasi di proses dengan menggunakan *method setCenter*(). *Method setCenter*() membutuhkan GetLatLng koordinat dan *zoom level*, dan *method* ini harus segera dikirim sebelum ada pengoperasian lain pada peta, termasuk seting atribut peta itu sendiri

5. *Loading Map*

   Ketika halaman HTML di *render*, *document object model* (DOM) sudah bisa digunakan, dan semua gambar *external* dan *script* diterima oleh *object document*. Untuk memastikan peta kita di muat sesudah halaman selesai dimuat oleh *browser*

6. Fungsi *GUnload*

   adalah *utility* untuk menangani *memory leak*

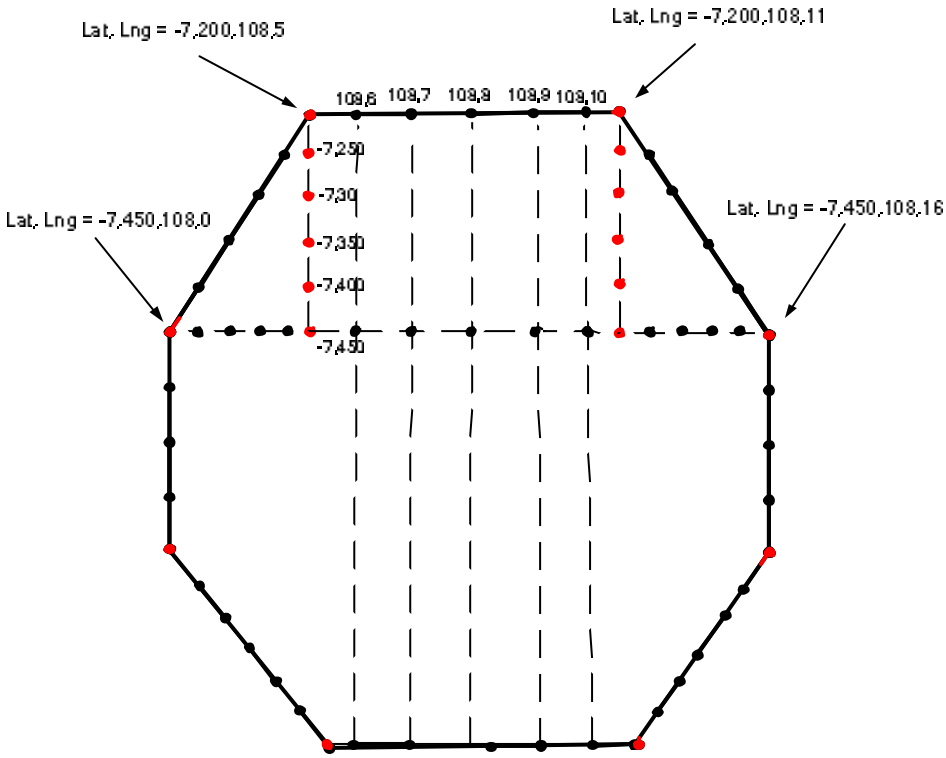
7. Fungsi *GDirections*

   Adalah *utility* yang menangani masalah pencarian rute

8. *GMarker*

   Marker adalah simbol yang menandakan suatu lokasi wisata pada peta yang ditampilkan Google Maps

Google Maps API adalah library JavaScript. Dengan menggunakan Google Maps API kita dapat menghemat waktu dan biaya untuk membangun aplikasi peta digital yang handal, sehingga dapat fokus hanya pada data-data non spasial, Maps yang dipakai digoogle maps adalah maps dunia, maka diperlukan sebuah pembatas yaitu *polygon*, *polygon* adalah shape yang digunakan untuk menandakan suatu daerah atau area. *Polygon* yaitu terdiri dari kumpulan titik koordinat. Daerah Tasikmalaya di tandai dengan *polygon* sebagai pembatas dari daerah sebelahnya agar daerah lokalisir Tasikmalaya bisa di lihat dan bisa di tentukan titik – titik koordinat yang di dalam daerah Kota Tasikmalaya atau berada diluar kota tasikmalaya, di gunakan rumus yaitu mencari 1 titik diantara dua titik, untuk langkah – langkahnya adalah sebagai berikut



Gambar 3.1 Teknik Penyelesaian Lokalisir Daerah

1. Menentukan titik/koordinat utama dari *polygon* atau batas daerah,
2. Membuat garis dari titik – titik utama disepanjang *polygon*,

3. Mencari titik – titik dari garis yang dibuat dengan jarak yang sama,
4. Dari perhitungan yang dilakukan lewat penentuan garis yang telah dibuat didapatkan batas kiri dan batas kanan dari sebuah *polygon* atau daerah lokalisir,
5. Batas kanan dan batas kiri dibuat menjadi *array* agar bisa di pakai untuk validasi data koordinat yang akan dimasukkan.
6. Membandingkan koordinat yang akan di inputkan apakah terdapat didalam *polygon* atau berada diluar *polygon*.

Dan untuk menandakan posisi potensi wisata di gunakan *marker*, sebagai penanda pada satu titik koordinat, dan sebagai penunjuk arah atau rutenya menggunakan API GDirections, GDirections menyediakan rute dari satu tempat ketempat lain.

3.1.3.2 Data Atribut/ Non Spasial

Data atribut merupakan data pendukung, dalam hal ini data wisata, seperti alamat, kode pos, gambar, keterangan dan sebagainya.

3.1.4 Analisis Prosedur

Analisis prosedur dilakukan untuk menetapkan proses apa saja yang dilakukan sistem. Dalam proses pelaksanaannya, pengolahan data harus dilaksanakan berdasarkan prosedur-prosedur yang telah ditetapkan, seperti berikut:

1. Data wisata dimasukan dari *user* yang telah terdaftar (*Member*).

2. Data wisata yang didapat kemudian diolah oleh Admin, diperiksa apakah data tersebut layak atau benar untuk di tampilkan, jika data tersebut layak untuk ditampilkan maka Admin memasukan *Latitude* dan *Longitude* (Koordinat) pada data tersebut agar kemudian tampilkan,
3. kemudian menghasilkan data yang siap disajikan kepada pengunjung. Pengunjung dapat melihat data wisata tersebut dan rute perjalanan menuju lokasi wisata tersebut.

3.1.5 Kebutuhan Perangkat Lunak

Berikut akan dijelaskan tentang beberapa kebutuhan spesifikasi perangkat lunak yang menunjang dalam pembuatan aplikasi ini, diantaranya sebagai berikut:

1. Sistem Operasi : Windows XP SP2.
2. *Database* : MySQL versi 5.0.
3. Pemrograman : PHP versi 5, JavaScript.
4. Aplikasi : Apache Web Server 2.2.

Aplikasi web yang akan dibangun memiliki kemampuan sebagai berikut:

a. Dapat menampilkan lokasi wisata beserta informasi yang ada pada suatu objek wisata dan rute kendaraan menuju lokasi wisata tersebut.
b. Memiliki kemampuan navigasi peta seperti *zoom in*, *zoom out*, *recenter*,*index*, *indeks* dan *calebar*.
c. Memiliki fitur pencarian, untuk mencari lokasi dan informasi suatu wisata.
d. Memberikan informasi rute perjalanan menuju suatu objek wisata.

3.1.6 Perangkat Keras

Perangkat keras merupakan salah satu hal yang harus kita perhatikan, karena tanpa perangkat keras yang memenuhi syarat, aplikasi yang akan kita bangun tidak akan bisa berjalan dengan baik. Berikut spesifikasi perangkat keras yang digunakan untuk membangun aplikasi ini:

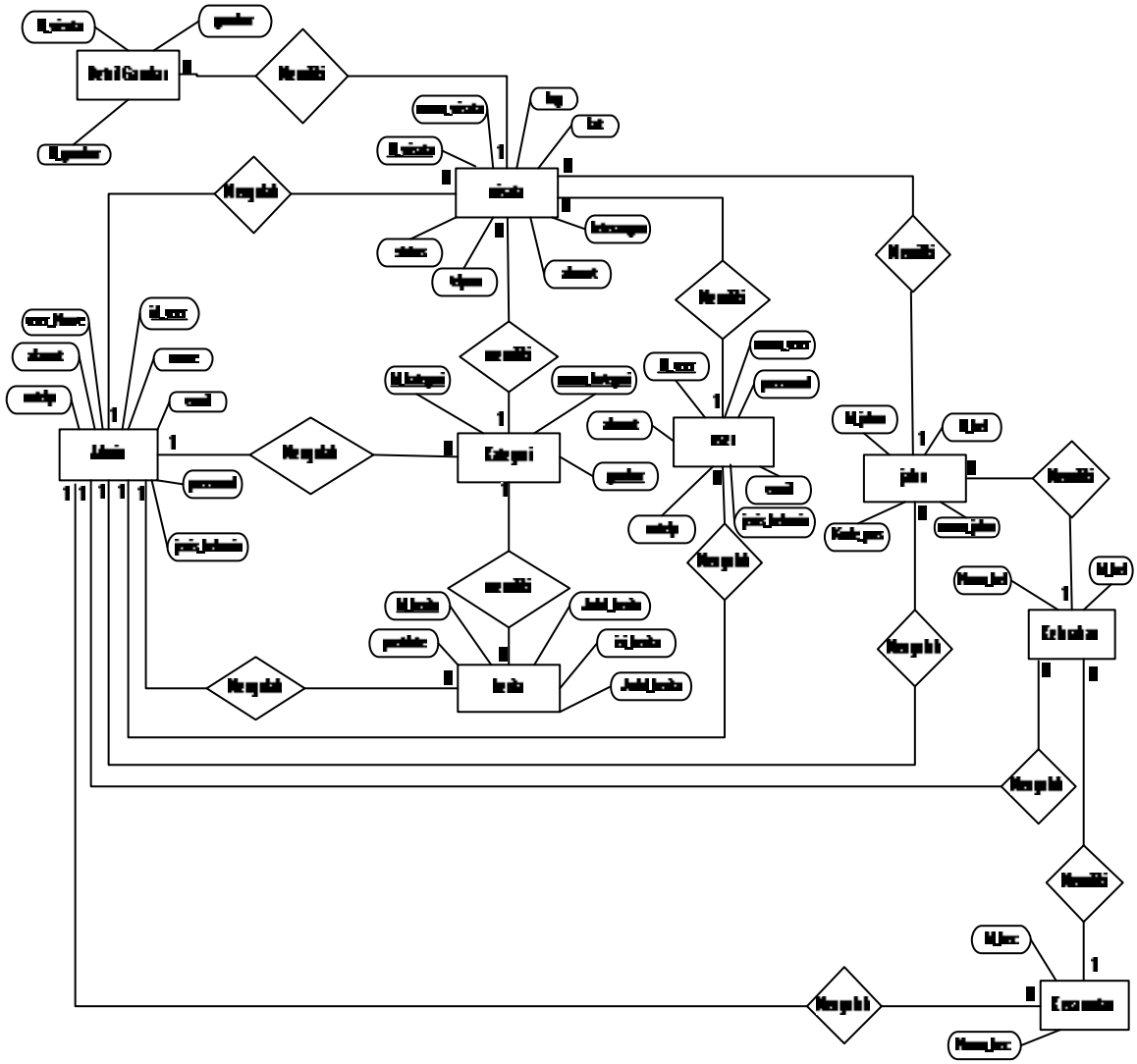
1. *Memory* 1 GB
2. *Processor Intel Pentium* IV 2.4GHz
3. *Harddisk* 80GB
4. *Video Card* 128MB
5. *Monitor* 1024 x 768 dengan 256 warna

3.1.7 Pengguna Sistem

Aplikasi ini dapat digunakan oleh pengunjung yang ingin mengetahui lokasi wisata dikota Tasikmalaya, pengunjung hanya bisa melihat data wisata yang telah di olah. Pengunjung bisa mendaftarkan sabagai *member* di website ini , maka pengunjung tersebut di beri otoritas untuk bisa memasukan data wisata, dan mengubah data yang telah di masukan oleh *user* tersebut, dan yang mempunyai akses penuh atas *WebGIS* ini adalah admin, admin disini adalah pegawai bagian IT di Dinas Perhubungan Kota Tasikmalaya, admin diharuskan mempunyai pengetahuan di bidang geografis, khususnya untuk wilayah kota Tasikmalaya. Dan admin menggunakan GPS untuk menentukan *Latitude* dan *Longitude* yang akurat.

3.2 Analisis Basis Data

E-R merupakan cara untuk mengorganisasikan data, dimana diagram ini akan memperlihatkan hubungan entitas yang terdapat dalam sistem. E-R yang diusulkan untuk sistem yang akan dibangun dapat dilihat pada gambar 3.2 dibawah ini.



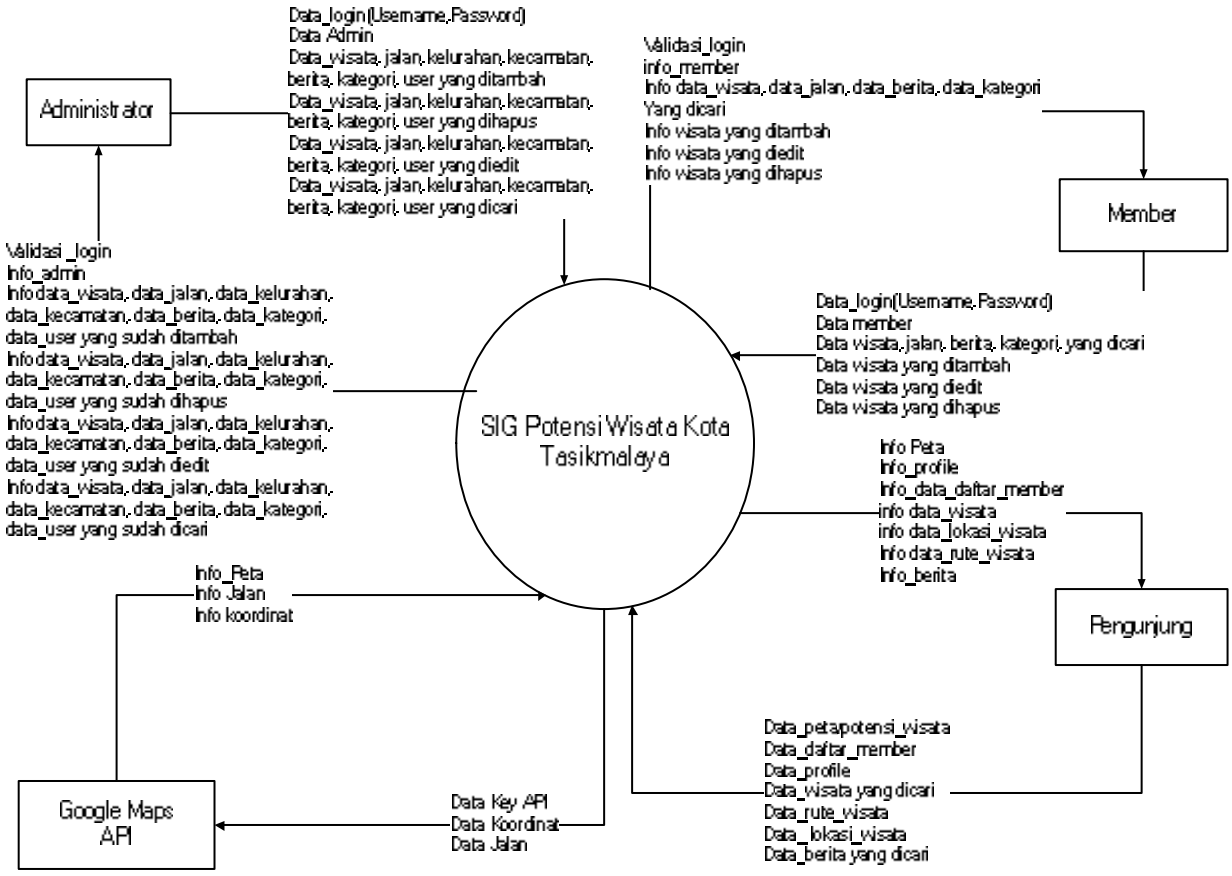
Gambar 3.2 ERD *WebGIS* Pariwisata Kota Tasikmalaya

3.3 Analisis Kebutuhan Fungsional

Dalam langkah ini dilakukan penentuan entitas-entitas, data-data yang mengalir serta prosedur-prosedur yang bisa dilakukan oleh masing-masing entitas.

3.3.1 Diagram Konteks (*Context Diagram*)

Diagram konteks adalah alur data yang berfungsi untuk menggambarkan keterkaitan aliran-aliran data antara sistem dengan bagian-bagian luar. Adapun diagram konteks untuk sistem yang akan dibangun dapat dilihat pada gambar 3.3 dibawah ini.



Gambar 3.3 Diagram konteks webSIG Pariwisata Kota Tasikmalaya

Diagram konteks menggambarkan hubungan antara entitas-entitas yang terdapat diluar sistem dengan sistem dan masukan serta keluaran dari sistem. Dalam sistem informasi ini, terdapat 4 entitas yaitu :

a. Pengunjung

   Pengunjung adalah siapa saja yang mengakses web site SIG ini tanpa harus *login* terlebih dahulu, dan aksesnya terbatas hanya fitur-fitur tertentu.

b. Member

   member adalah siapa saja yang yang mendaftar pada aplikasi ini dan mempunyai akses untuk menambah, menghapus, dan merubah data wisata yang telah mereka *input*kan.

c. Admin

   Admin adalah user yang memiliki hak akses penuh atas sistem ini yang dapat melakukan *update* data dan *manajemen* data.

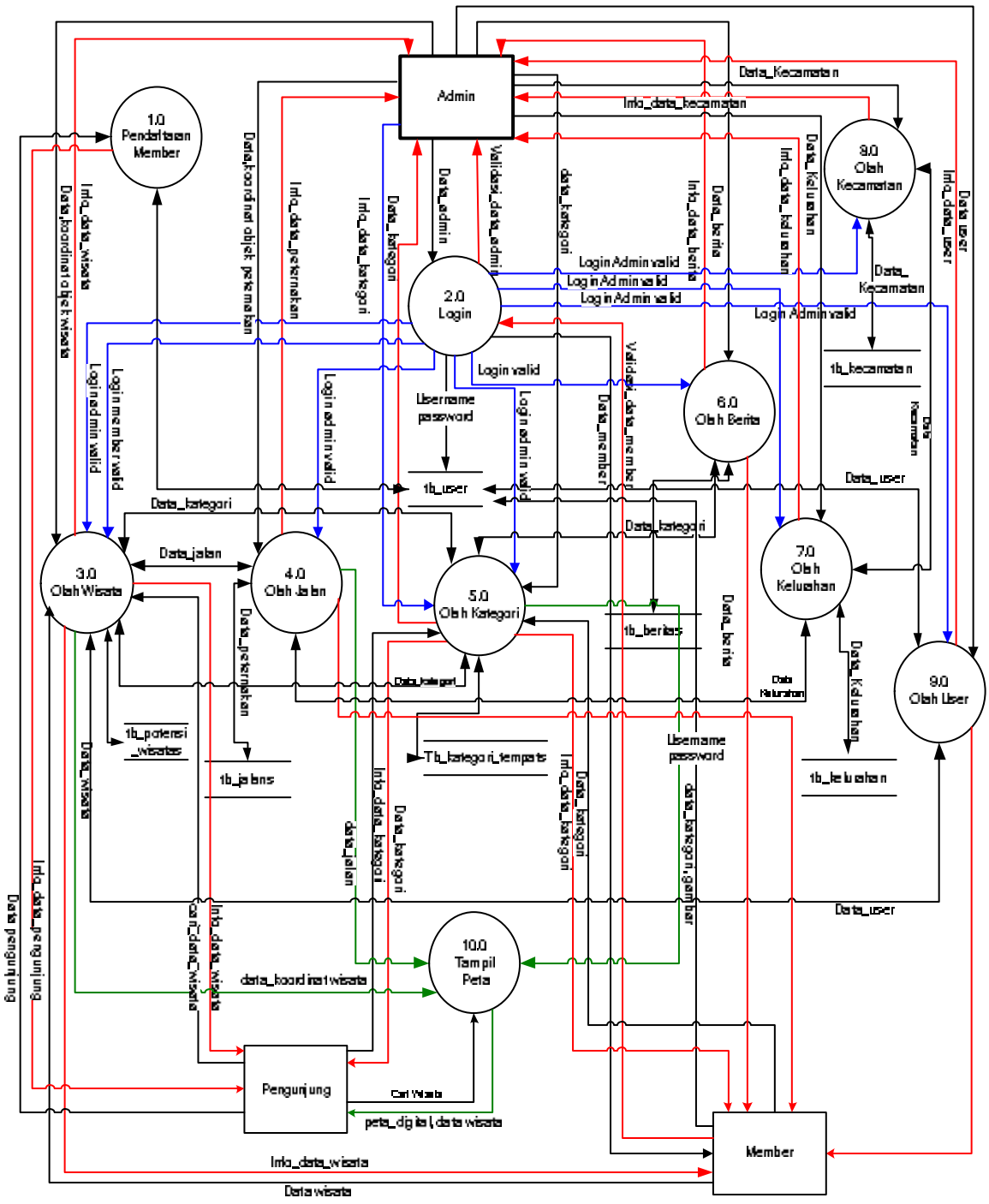
d. Google Maps API

   Google Maps Api adalah *tool* atau *library* dari Google untuk menampilkan peta digital di website, penggunaan *Google Maps* di website dapat melalui pemrogramman.

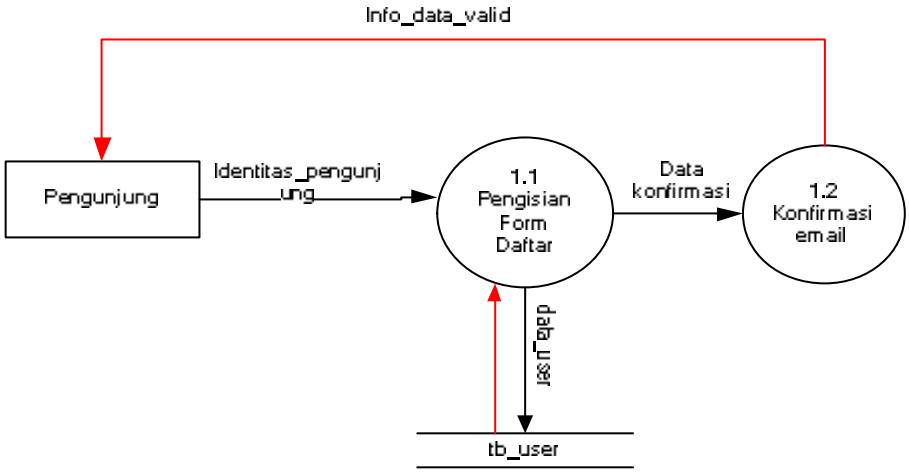
3.3.2 Diagram Aliran Data (*Data Flow Diagram*)

DFD adalah sebuah teknik yang menggambarkan aliran data dan transformasi yang digunakan sebagai perjalanan data dari masukan menuju keluaran..

Dibawah ini adalah DFD Level 1 dimana terdapat sepuluh proses yaitu proses login merupakan proses pertama yang harus dilakukan admin agar bisa memasuki proses selanjutnya yaitu pendaftaran, login, olah wisata, olah jalan, olah kategori, olah berita, olah kelurahan, olah kecamatan dan olah *user*, dan proses pertama yang harus di lakukan proses pendaftaran pengunjung menjadi *member*, agar bisa melakukan proses tambah, ubah dan hapus wisata, pada gambar 3.4 dijelaskan bagaimana seluruh usulan sistem yanga kan di bangun...

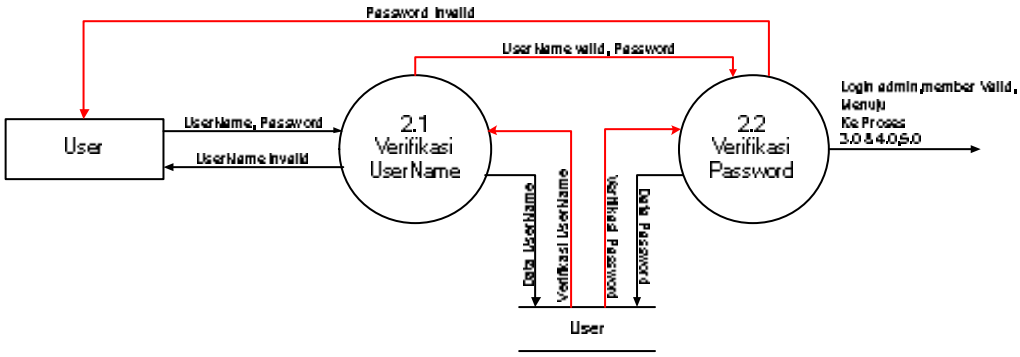


Gambar 3.4 DFD Level 1 WebGIS Pariwisata Kota Tasikmalaya

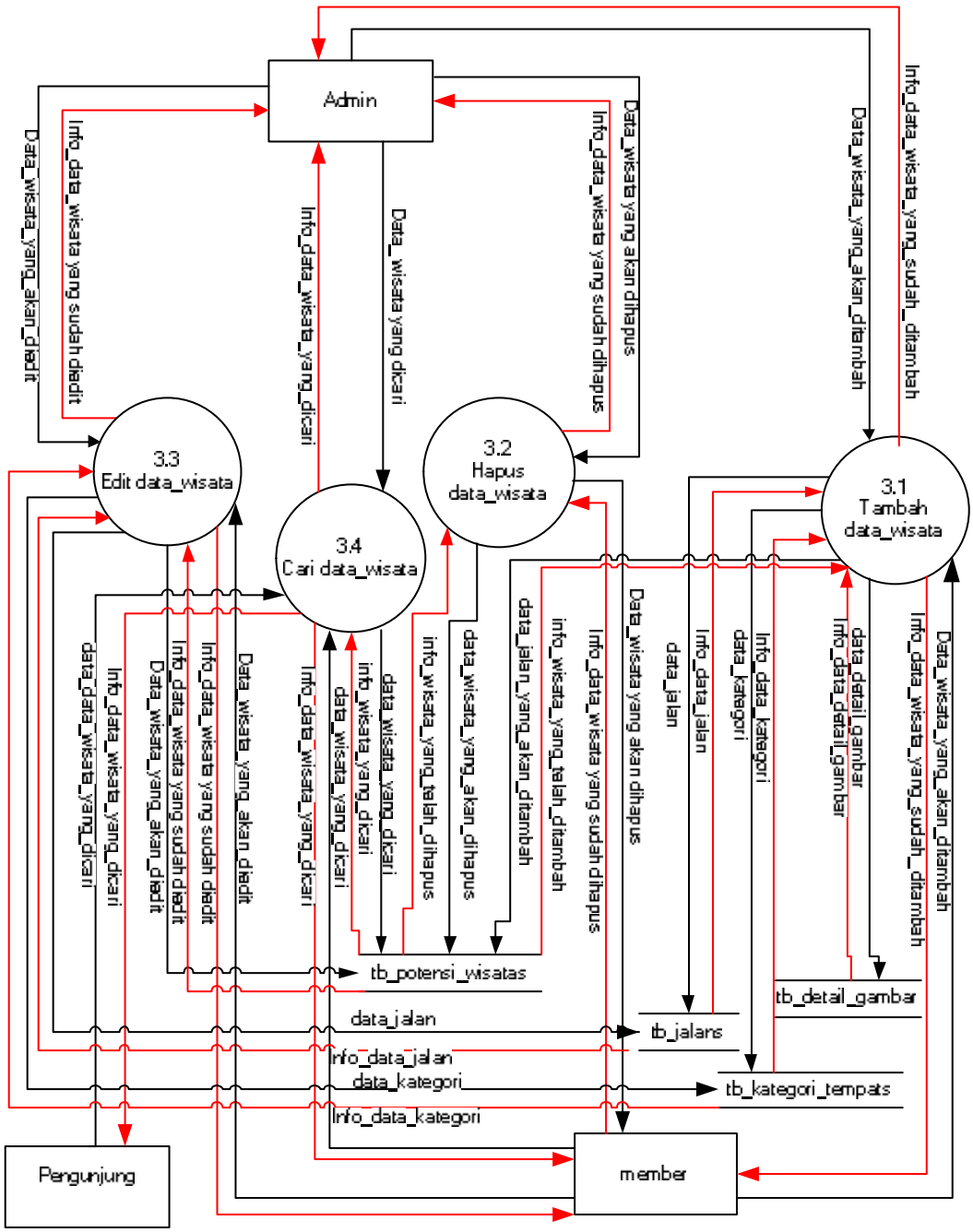


Gambar 3.5 DFD Level 2 Proses 1 Pendaftaran *Member*

DFD Level 1 menjelaskan proses pendaftaran *member* oleh pengunjung.

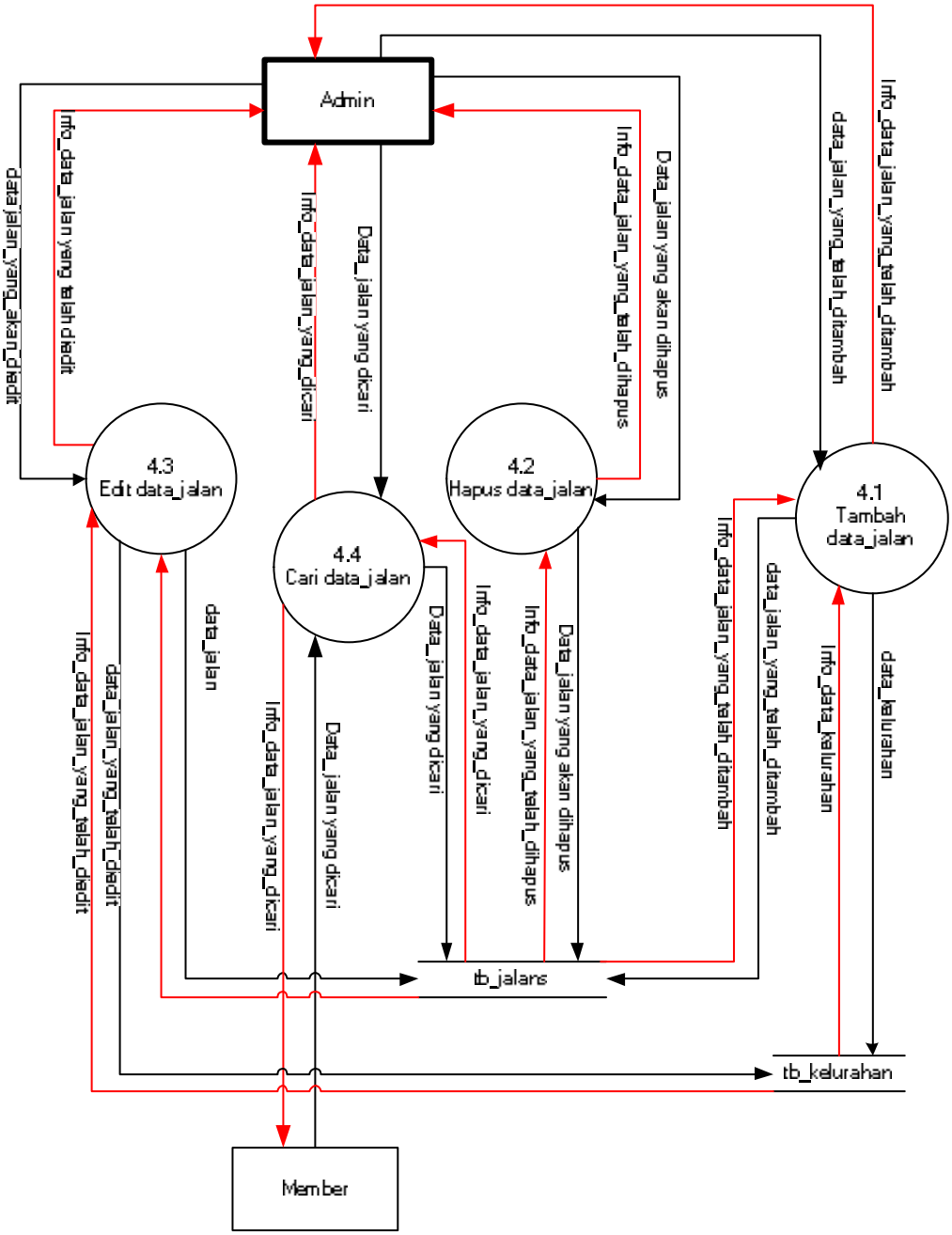


Gambar 3.6 DFD Level 2 Proses 2 Login *User*



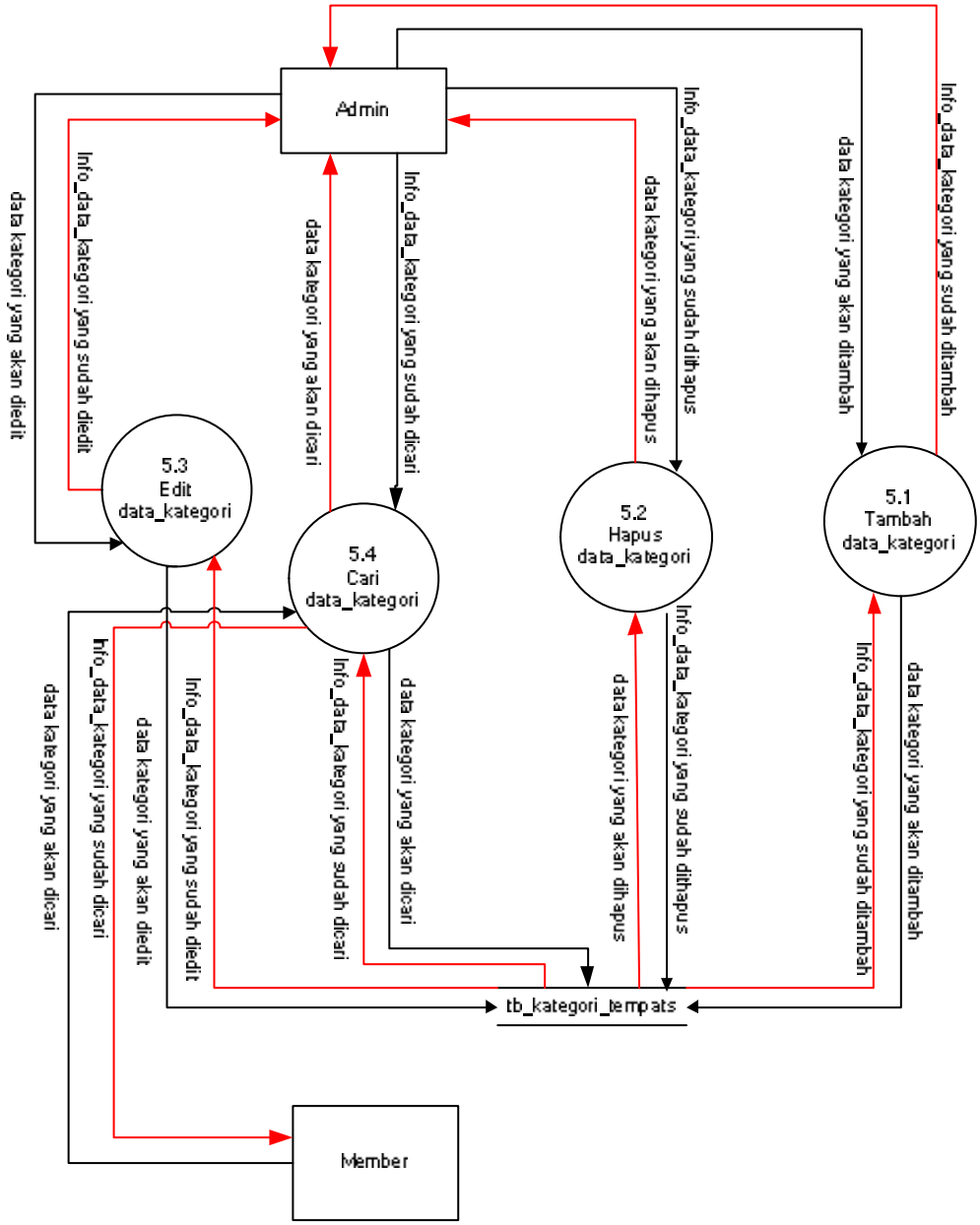
Gambar 3.7 DFD Level 2 Proses Olah Wisata

Pada gambar 3.7 Pada DFD Level 2 proses olah wisata ini terdiri atas proses tambah,hapus, *edit*, dan cari data wisata.



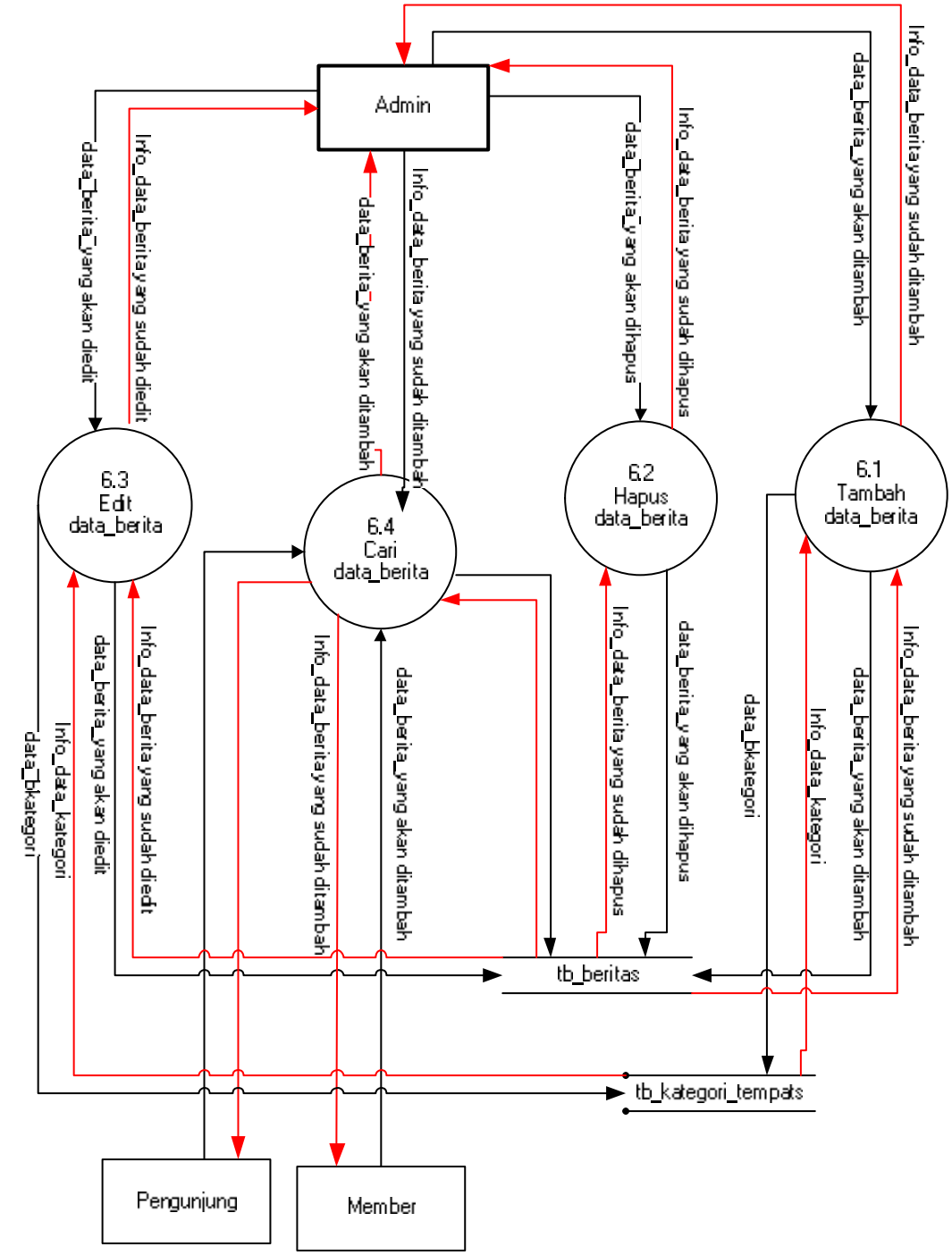
Gambar 3.8 DFD Level 2 Proses Olah Jalan

Pada gambar 3.8 Pada DFD Level 2 proses olah jalan ini terdiri atas proses tambah,hapus, *edit*, dan cari data jalan.



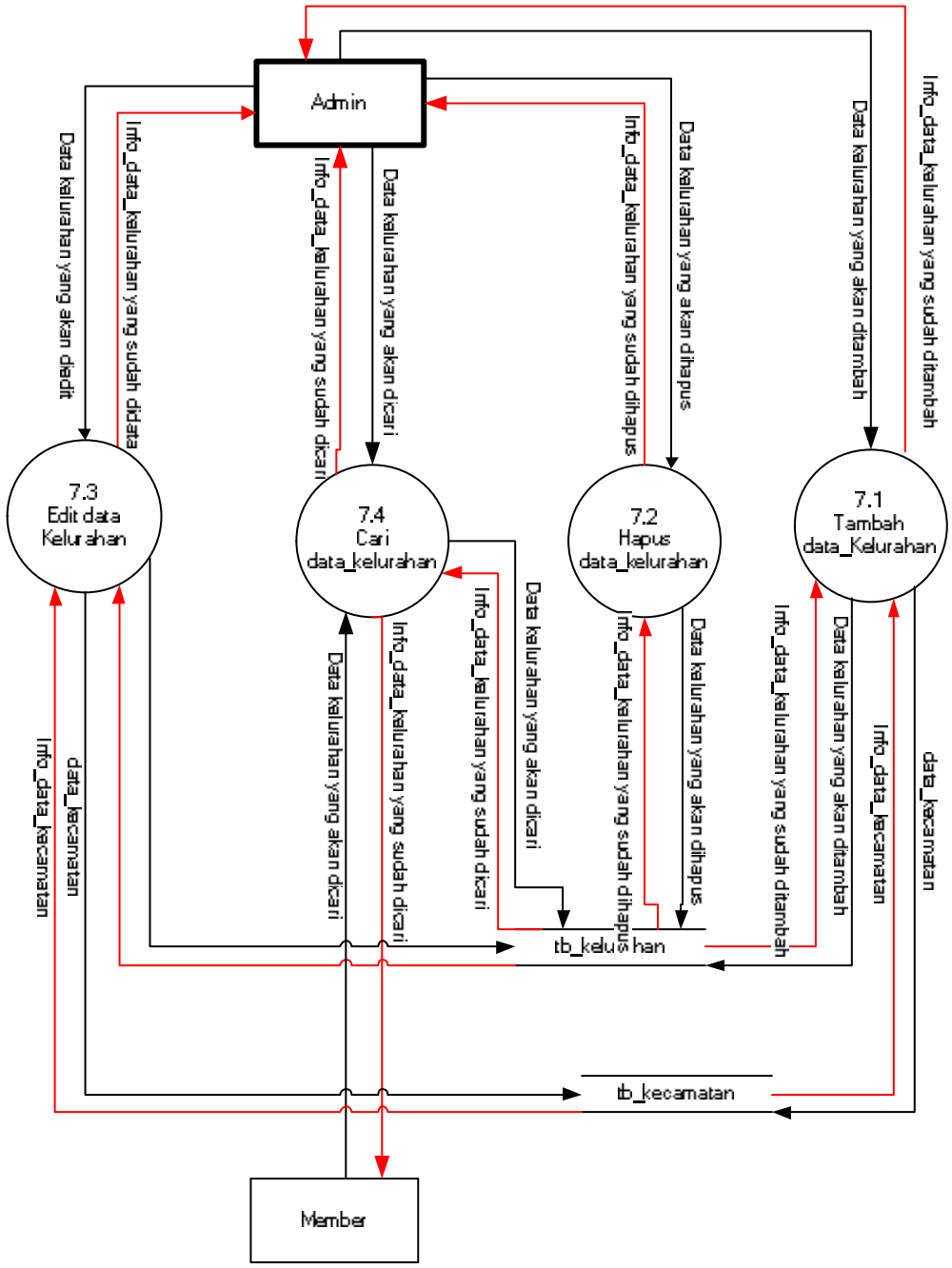
Gambar 3.9 DFD Level 2 Proses Olah Kategori

P Pada gambar 3.09 ada DFD Level 2 proses olah Kategori ini terdiri atas proses tambah,hapus, edit, dan cari data Kategori.



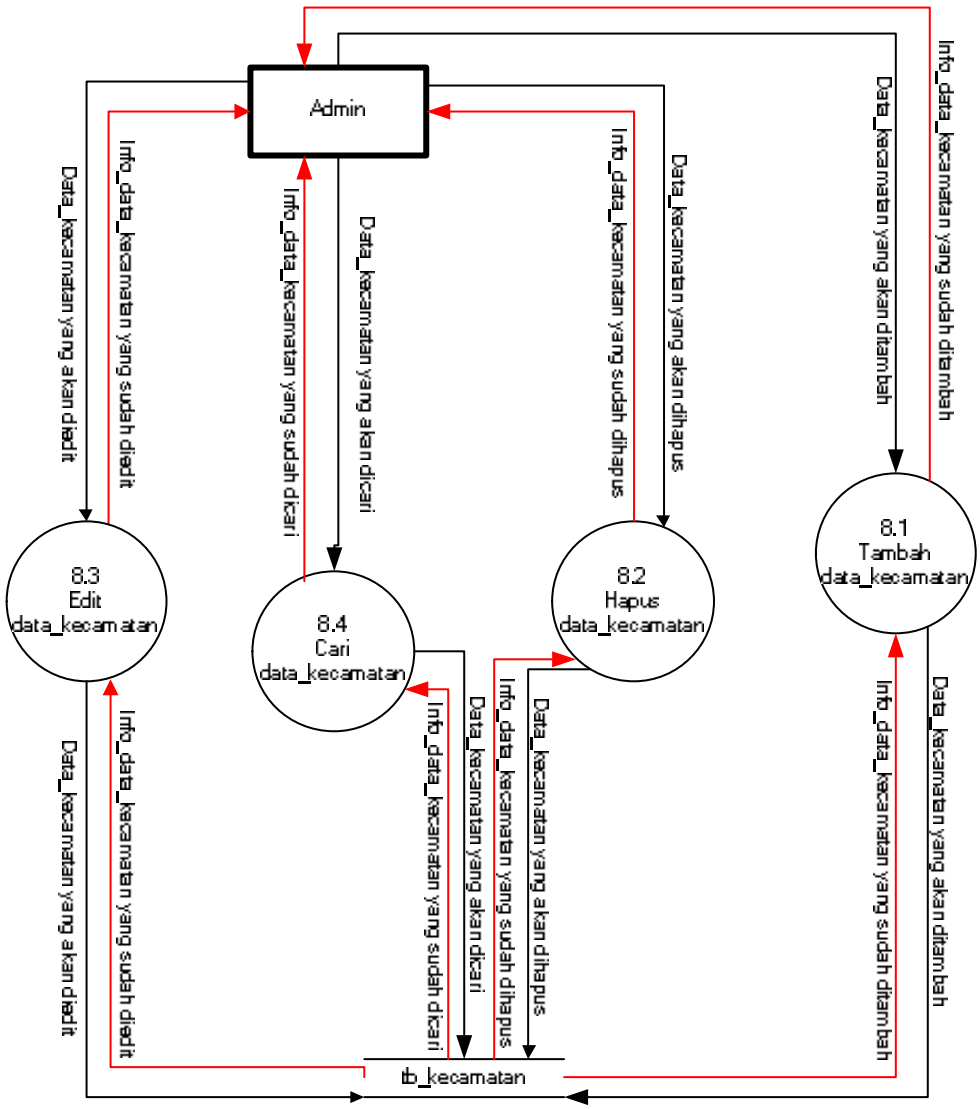
Gambar 3.10 DFD Level 2 Proses Olah Berita

Pada gambar 3.10 Pada DFD Level 2 proses olah berita ini terdiri atas proses tambah,hapus, edit, dan cari data berita.



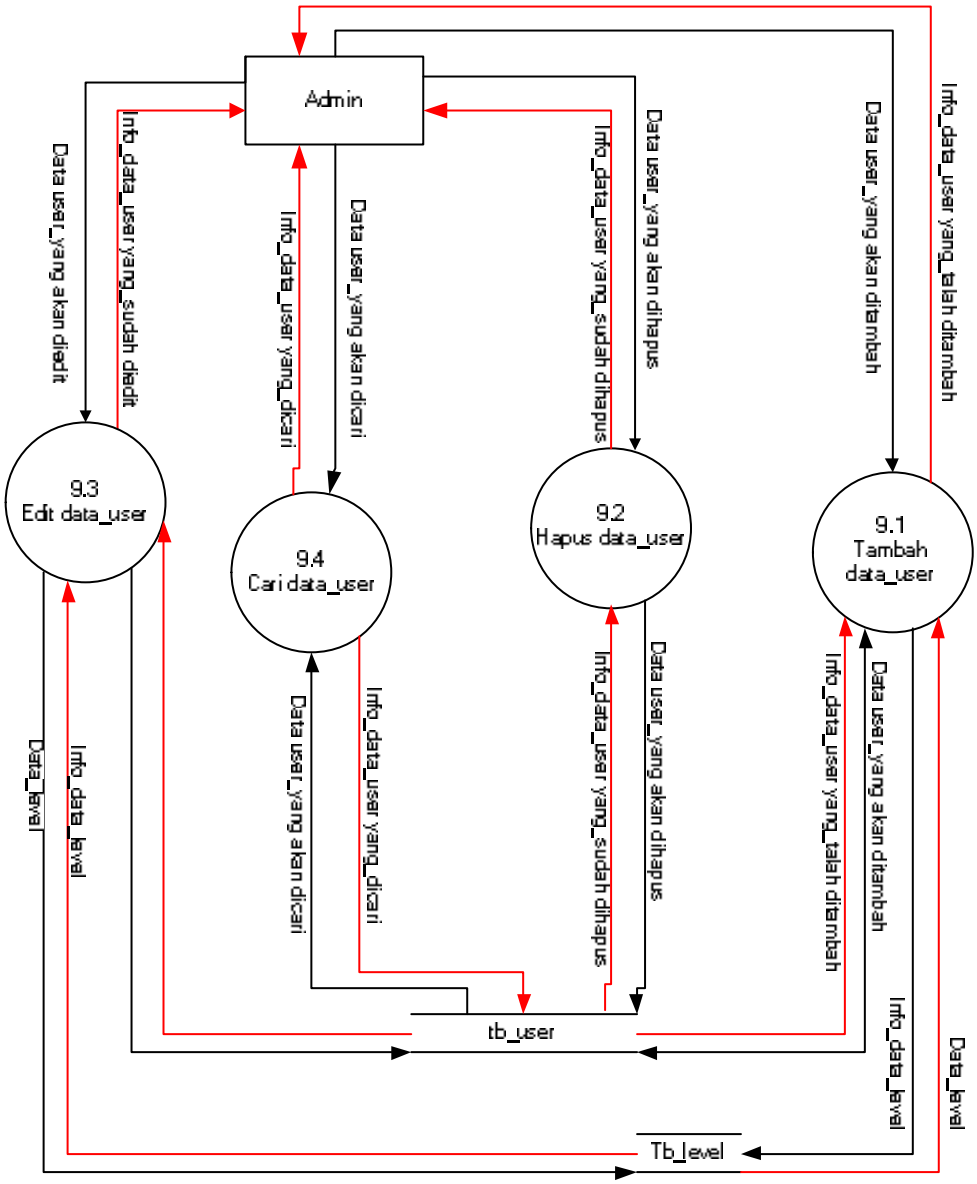
Gambar 3.11 DFD Level 2 Proses Olah Kelurahan

Pada gambar 3.11 Pada DFD Level 2 proses olah kelurahan ini terdiri atas proses tambah,hapus, edit, dan cari data kelurahan.



Gambar 3.12 DFD Level 2 Proses Olah Kecamatan

Pada gambar 3.12 DFD Level 2 proses olah kelurahan ini terdiri atas proses tambah,hapus, edit, dan cari data kecamatan.



Gambar 3.13 DFD Level 2 Proses Olah User

Pada gambar 3.13 DFD Level 2 proses olah user ini terdiri atas proses tambah,hapus, *edit*, dan cari data user.

3.3.3 Spesifikasi Proses

Proses-proses yang terdapat pada *Data Flow Diagram* akan dijelaskan lebih terperinci pada spesifikasi proses sebagai berikut :

Tabel 3.1. Spesifikasi Proses

weNO	PROSES	KETERANGAN
1	**No Proses**	1.1
	Nama Proses	Daftar *Member*
	Source (sumber)	Pengunjung
	Input	Identitas Pengunjung
	Output	Info Identitas Pengunjung
	Destination (tujuan)	Pengunjung
	Logika Proses	Begin {pengunjung melakukan daftar *member* dengan memasukkan data pengunjung} end
2	**No Proses**	1.2
	Nama Proses	Konfirmasi
	Source (sumber)	Pengunjung
	Input	Data daftar
	Output	Pendaftaran Invalid
	Destination (tujuan)	pengunjung

	Logika Proses	Begin
		{pengunjung *input* identitas pengujung ke *database*}
		if validasi benar
		Then pendaftaran berhasil, kirim kode_konfirmasi ke email pengunjung
		else
		Pendaftaran invalid
		endif
		end

3	**No Proses**	2.1
	Nama Proses	Login
	Source (sumber)	Admin
	Input	-UserName -Password
	Output	-UserName Invalid -Password Invalid
	Destination	Admin

	(tujuan)	

	Logika Proses	Begin
		{Admin input UserName, Password ke database}
		if UserName ada dan Password benar
		Then UserName, Password valid menuju proses 2.0
		else
		Tampil info UserName, Password invalid
		endif
		end

4	**No Proses**	2.2
	Nama Proses	Verifikasi Username
	***Source* (sumber)**	Admin
	Input	Username
	Output	Username Invalid
	Destination **(tujuan)**	Admin

	Logika Proses	begin
		{Admin memasukan Username ke database}
		if Username ada
		then UserName valid menuju ke proses 1.2
		else
		Tampil info login invalid
		endif
		end

3	**No Proses**	2.3
	Nama Proses	Verifikasi Password
	***Source* (sumber)**	Admin
	Input	Password
	Output	Password Invalid
	Destination **(tujuan)**	Admin

	Logika Proses	begin
		{Admin memasukan Password ke database}
		if Password ada
		then Password valid menuju ke proses 2.0
		else
		Tampil info login invalid
		endif
		end

4	**No Proses**	3.1
	Nama Proses	Tambah Data Wisata
	***Source* (sumber)**	Admin, *Member*
	Input	Data wisata yang akan ditambah
	Output	Informasi Data wisata yang sudah ditambah
	***Destination* (tujuan)**	Admin, *Member*

	Logika Proses	Begin
		{Admin/*Member* manambah data wisata ke *database*}
		If data wisata benar
		Then
		Menampilkan data berhasil ditambah
		Else
		Menampilkan pesan *error* data tidak berhasil ditambah
		End

5	**No Proses**	3.2
	Nama Proses	Hapus Data Wisata
	***Source* (sumber)**	Admin
	Input	Data Wisata Yang akan di hapus
	Output	Info Data wisata yang sudah dihapus
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin/*Member* manghapus data wisata ke *database*}
		If data wisata ada
		Then
		Menampilkan data berhasil dihapus
		Else
		Menampilkan pesan *error* data tidak berhasil dihapus
		End

6	**No Proses**	3.3
	Nama Proses	*Edit* Data Wisata
	***Source* (sumber)**	Admin, *Member*
	Input	Data Wisata yang akan diubah
	Output	Informasi Wisata yang diubah
	***Destination* (tujuan)**	Admin, *Member*, dan pengunjung
	Logika Proses	Begin {Admin/*Member* mangubah data wisata ke

		database} If data wisata benar Then Menampilkan data berhasil diedit Else Menampilkan pesan *error* data tidak berhasil di-*edit* End
7	**No Proses**	3.4
	Nama Proses	Cari Data Wisata
	***Source* (sumber)**	Admin, *Member* dan pengunjung
	Input	Data_wisata_yang_dicari
	Output	Info_data_wisata_yang_dicari
	***Destination* (tujuan)**	Admin, *member*, dan pengunjung

	Logika Proses	Begin
		{Admin/*Member*/pengunjung mancari data wisata ke *database*}
		If data wisata ada
		Then
		Menampilkan data berhasil ditemukan,dan data ditampilkan
		Else
		Menampilkan pesan *error* data tidak ditemukan
		End

8	**No Proses**	4.1
	Nama Proses	Tambah Data Jalan
	***Source* (sumber)**	Admin
	Input	Data jalan yang akan ditambah
	Output	Informasi Data jalan yang sudah ditambah
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin manambah data jalan ke *database*}
		If data jalan benar
		Then
		Menampilkan data jalan berhasil ditambah
		Else
		Menampilkan pesan *error* data tidak berhasil ditambah
		End

9	**No Proses**	4.2
	Nama Proses	Hapus Data Jalan
	***Source* (sumber)**	Admin

	Input	Data Jalan Yang akan di hapus
	Output	Info Data jalan yang sudah dihapus
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin menghapus data jalan ke *database*}
		If data wisata ada
		Then
		Menampilkan data berhasil dihapus
		Else
		Menampilkan pesan *error* data tidak berhasil dihapus
		End

10	**No Proses**	4.3
	Nama Proses	*Edit* Data Jalan
	***Source* (sumber)**	Admin
	Input	Data jalan yang akan diubah
	Output	Informasi jalan yang diubah
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin mengubah data jalan ke *database*}
		If data data jalan benar
		Then
		Menampilkan data berhasil di-*edit*
		Else
		Menampilkan pesan *error* data tidak berhasil di-*edit*
		End

11	**No Proses**	4.4
	Nama Proses	Cari Data Jalan
	***Source* (sumber)**	Admin, *Member*
	Input	Data_jalan_yang_dicari
	Output	Info_data_jalan_yang_dicari
	***Destination* (tujuan)**	Admin, *member*

	Logika Proses	Begin
		{Admin/*Member* mencari data jalan ke *database*}
		If data jalan ada
		Then
		Menampilkan data berhasil ditemukan,dan data ditampilkan
		Else
		Menampilkan pesan *error* data tidak ditemukan
		End

12	**No Proses**	5.1
	Nama Proses	Tambah Data kategori
	***Source* (sumber)**	Admin
	Input	Data kategori yang akan ditambah
	Output	Informasi Data kategori yang sudah ditambah
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin manambah datakategori ke *database*}
		If data kategori benar
		Then
		Menampilkan data kategori berhasil ditambah
		Else
		Menampilkan pesan *error* data tidak berhasil ditambah
		End

13	**No Proses**	5.2
	Nama Proses	Hapus Data kategori
	***Source* (sumber)**	Admin
	Input	Data kategori yang akan di hapus
	Output	Info Data kategori yang sudah dihapus
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin menghapus data kategori ke *database*}
		If data kategori ada
		Then
		Menampilkan data berhasil dihapus
		Else
		Menampilkan pesan *error* data tidak berhasil dihapus
		End

14	**No Proses**	5.3
	Nama Proses	*Edit* Data kategori
	***Source* (sumber)**	Admin
	Input	Data kategori yang akan diubah
	Output	Informasi kategori yang diubah
	***Destination* (tujuan)**	Admin
	Logika Proses	Begin {Admin mengubah data kategori ke *database*} If data data kategori benar Then

		Menampilkan data berhasil di-*edit* Else Menampilkan pesan *error* data tidak berhasil di-*edit* End
15	**No Proses**	5.4
	Nama Proses	Cari Data kategori
	***Source* (sumber)**	Admin, *Member*
	Input	Data_kategori_yang_dicari
	Output	Info_data_kategori_yang_dicari
	***Destination* (tujuan)**	Admin, *member*

	Logika Proses	Begin
		{Admin/*Member* mencari data jalan ke *database*}
		If data kategori ada
		Then
		Menampilkan data berhasil ditemukan,dan data ditampilkan
		Else
		Menampilkan pesan *error* data tidak ditemukan
		End

16	**No Proses**	6.1
	Nama Proses	Tambah Data Berita
	***Source* (sumber)**	Admin
	Input	Data barita yang akan ditambah
	Output	Informasi Data berita yang sudah ditambah
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin manambah data berita ke *database*}
		If data berita benar
		Then
		Menampilkan data berita berhasil ditambah
		Else
		Menampilkan pesan *error* data tidak berhasil ditambah
		End

	No Proses	6.2
	Nama Proses	Hapus Data berita
	***Source* (sumber)**	Admin
	Input	Data berita yang akan di hapus
	Output	Info Data berita yang sudah dihapus

7		
	Destination **(tujuan)**	Admin

	Logika Proses	Begin
		{Admin menghapus data berita ke *database*}
		If data berita ada
		Then
		Menampilkan data berhasil dihapus
		Else
		Menampilkan pesan *error* data tidak berhasil dihapus
		End

18	**No Proses**	6.3
	Nama Proses	*Edit* Data kategori
	Source **(sumber)**	Admin
	Input	Data berita yang akan diubah
	Output	Informasi berita yang diubah
	Destination **(tujuan)**	Admin

	Logika Proses	Begin
		{Admin mengubah data kategori ke *database*}
		If data data kategori benar
		Then
		Menampilkan data berhasil di-*edit*
		Else
		Menampilkan pesan *error* data tidak berhasil di-*edit*
		End

19	**No Proses**	6.4
	Nama Proses	Cari Data kategori
	Source **(sumber)**	Admin, *Member*
	Input	Data_kategori_yang_dicari
	Output	Info_data_kategori_yang_dicari
	Destination **(tujuan)**	Admin, *member*

	Logika Proses	Begin
		{Admin/*Member* mencari data berita ke *database*}
		If data kategori ada
		Then
		Menampilkan data berhasil ditemukan,dan data ditampilkan
		Else
		Menampilkan pesan *error* data tidak ditemukan
		End

20	**No Proses**	7.1

	Nama Proses	Tambah Data Kelurahan
	***Source* (sumber)**	Admin
	Input	Data kelurahan yang akan ditambah
	Output	Informasi Data kelurahan yang sudah ditambah
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin manambah data kelurahan ke *database*}
		If data kelurahan benar
		Then
		Menampilkan data kelurahan berhasil ditambah
		Else
		Menampilkan pesan *error* data tidak berhasil ditambah
		End

21	**No Proses**	7.2
	Nama Proses	Hapus Data kelurahan
	***Source* (sumber)**	Admin
	Input	Data kelurahan yang akan di hapus
	Output	Info Data kelurahan yang sudah dihapus
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin menghapus data kelurahan ke *database*}
		If data kelurahan ada
		Then
		Menampilkan data berhasil dihapus
		Else
		Menampilkan pesan *error* data tidak berhasil dihapus
		End

22	**No Proses**	7.3
	Nama Proses	*Edit* Data keluran
	***Source* (sumber)**	Admin
	Input	Data kelurahan yang akan diubah
	Output	Informasi berita yang diubah
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin mengubah data kelurahan ke *database*}
		If data data kelurahan benar
		Then
		Menampilkan data berhasil di-*edit*
		Else

		Menampilkan pesan *error* data tidak berhasil di-*edit* End
23	**No Proses**	7.4
	Nama Proses	Cari Data kelurarahn
	***Source* (sumber)**	Admin
	Input	Data_kelurahan_yang_dicari
	Output	Info_data_kelurahan_yang_dicari
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin mencari data kelurahan ke *database*}
		If data kelurahan ada
		Then
		Menampilkan data berhasil ditemukan,dan data ditampilkan
		Else
		Menampilkan pesan *error* data tidak ditemukan
		End

24	**No Proses**	8.1
	Nama Proses	Tambah Data Kecamatan
	***Source* (sumber)**	Admin
	Input	Data Kecamatan yang akan ditambah
	Output	Informasi Data kecamatan yang sudah ditambah
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin manambah data Kecamatan ke *database*}
		If data Kecamatan benar
		Then
		Menampilkan data Kecamatan berhasil ditambah
		Else
		Menampilkan pesan *error* data tidak berhasil ditambah
		End

25	**No Proses**	8.2
	Nama Proses	Hapus Data Kecamatan
	***Source* (sumber)**	Admin
	Input	Data Kecamatan yang akan di hapus
	Output	Info Data kecamatan yang sudah dihapus
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin menghapus data Kecamatan ke-*database*}
		If data kecamatan ada
		Then
		Menampilkan data berhasil dihapus
		Else
		Menampilkan pesan *error* data tidak berhasil dihapus
		End

26	**No Proses**	8.3
	Nama Proses	*Edit* Data kategori
	***Source* (sumber)**	Admin
	Input	Data Kecamatan yang akan diubah
	Output	Informasi Kecamatan yang diubah
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin mengubah data Kecamatan ke *database*}
		If data data Kecamatan benar
		Then
		Menampilkan data berhasil di-*edit*
		Else
		Menampilkan pesan *error* data tidak berhasil di-*edit*
		End

27	**No Proses**	8.4
	Nama Proses	Cari Data Kecamatan
	***Source* (sumber)**	Admin,
	Input	Data_kecamatan_yang_dicari
	Output	Info_data_kecamatam_yang_dicari
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin mencari kecamatan berita ke *database*}
		If data kecamatan ada
		Then
		Menampilkan data berhasil ditemukan,dan data ditampilkan
		Else
		Menampilkan pesan *error* data tidak ditemukan
		End

28	**No Proses**	9.1
	Nama Proses	Tambah Data User
	***Source* (sumber)**	Admin
	Input	Data user yang akan ditambah

	Output	Informasi Data user yang sudah ditambah
	Destination **(tujuan)**	Admin

	Logika Proses	Begin
		{Admin manambah data user ke *database*}
		If data User benar
		Then
		Menampilkan data User berhasil ditambah
		Else
		Menampilkan pesan *error* data tidak berhasil ditambah
		End

29	**No Proses**	9.2
	Nama Proses	Hapus Data User
	Source **(sumber)**	Admin
	Input	Data User yang akan di hapus
	Output	Info Data user yang sudah dihapus
	Destination **(tujuan)**	Admin

	Logika Proses	Begin
		{Admin menghapus data User ke-*database*}
		If data User ada
		Then
		Menampilkan data berhasil dihapus
		Else
		Menampilkan pesan *error* data tidak berhasil dihapus
		End

30	**No Proses**	9.3
	Nama Proses	*Edit* Data User
	Source **(sumber)**	Admin
	Input	Data User yang akan diubah
	Output	Informasi Useryang diubah
	Destination **(tujuan)**	Admin

	Logika Proses	Begin
		{Admin mengubah data User ke *database*}
		If data data User benar
		Then
		Menampilkan data berhasil di-*edit*
		Else
		Menampilkan pesan *error* data tidak berhasil di-*edit*
		End

31	**No Proses**	9.4

	Nama Proses	Cari Data User
	***Source* (sumber)**	Admin,
	Input	Data_User_yang_dicari
	Output	Info_data_user_yang_dicari
	***Destination* (tujuan)**	Admin

	Logika Proses	Begin
		{Admin mencari kecamatan User ke *database*}
		If data user ada
		Then
		Menampilkan data berhasil ditemukan,dan data ditampilkan
		Else
		Menampilkan pesan *error* data tidak ditemukan
		End

30	**No Proses**	10.0
	Nama Proses	Informasi Peta
	***Source* (sumber)**	Pengunjung
	Input	Data_lokasi_wisata, nama_lokasi_wisata
	Output	Info data lokasi wisata, info nama lokasi
	***Destination* (tujuan)**	User

	Logika Proses	Begin
		{pengunjung mencari lokasi wisata dipeta}
		If data wisata ada
		Then
		Menampilkan lokasi wisata dipeta
		Else
		Menampilkan pesan *error* data tidak ditemukan
		End

3.3.4 Kamus Data

Kamus data adalah kumpulan dari elemen-elemen atau simbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap *field* atau *file* didalam sistem. Kamus data merupakan deskripsi formal mengenai seluruh elemen yang mencakup DFD.

Tabel 3.9 Kamus Data

No	Kamus	Keterangan
1	Nama alir data	Data Pendaftaran
	Where used/how used	pengunjung-proses 1 (input)
	Deskripsi	Dilakukan oleh pengunjung yang mendaftar
	Struktur data	username+password+name+email+jenis_kelamin+no_telp+alamat
	username	[A-Z | a-z]
	password	[A-Z | a-z | 0-9]
	name	[A-Z | a-z]
	email	[A-Z | a-z | 0-9]
	jenis_kelamin	[A-Z | a-z]
	no_telp	[0-9]
	alamat	[A-Z | a-z | 0-9]
2	Nama alir data	Data Login
	Where used/how used	admin-proses 1 (input)
	Deskripsi	Dilakukan oleh admin yang mengelola
	Struktur data	username+password
	username	[A-Z | a-z]
	password	[A-Z | a-z | 0-9]
3	Nama alir data	Data wisata
	Where used / used	Admin,member –proses 2.0(input),proses 2.0

		(output)
	Deskripsi	Pengolahan data wisata
	Struktur data	id_wisata+ nama_wisata+alamat+no_telp +keterangan+longitude+latitude+id_kategori+ +status+id_user
	id_wisata	[0-9]
	nama_wisata	[A-Z | a-z]
	alamat	[A-Z | a-z]
	no_telp	[0-9]
	keterangan	[[A-Z | a-z] [0-9]
	lat	[0-9]
	lng	[0-9]
	id_kategori	[0-9]
	status	[[A-Z | a-z]
	id_user	[0-9]
4	Nama alir data	Data jalan
	Where used / used	Admin,proses 3.0(input),proses 3.0 (output)
	Deskripsi	Pengolahan data jalan
	Struktur data	Id_jalan+ nama_jalan+kode_pos+id_kel
	id_jalan	[0-9]
	nama_jalan	[[A-Z | a-z]
	kode_pos	[0-9]
	id	[0-9]

5	Nama alir data	Data kategori
	Where used / used	Admin,pengunjung –proses 4.0(input),proses 4.0 (output)
	Deskripsi	Pengolahan data kategori
	Struktur data	id_kategori+ nama_kategori+gambar
	id_kategori	[0-9]
	nama_kategori	[A-Z | a-z]
	gambar	[A-Z | a-z]
6	Nama alir data	Data berita
	Where used / used	Admin–proses 5.0(input),proses 5.0 (output)
	Deskripsi	Pengolahan data berita
	Struktur data	id_berita+id_kategori+judul+isi_berita+pengirim+postdate
	id_berita	[0-9]
	id_kategori	[A-Z | a-z]
	judul	[A-Z | a-z]
	isi_berita	[A-Z | a-z]
	pengirim	[A-Z | a-z]
	postdate	[0-9]
7	Nama alir data	Data kelurahan
	Where used / used	Admin–proses 7.0(input),proses 7.0 (output)
	Deskripsi	Pengolahan data kelurahan
	Struktur data	id_ kel+nama_kel+id_kec

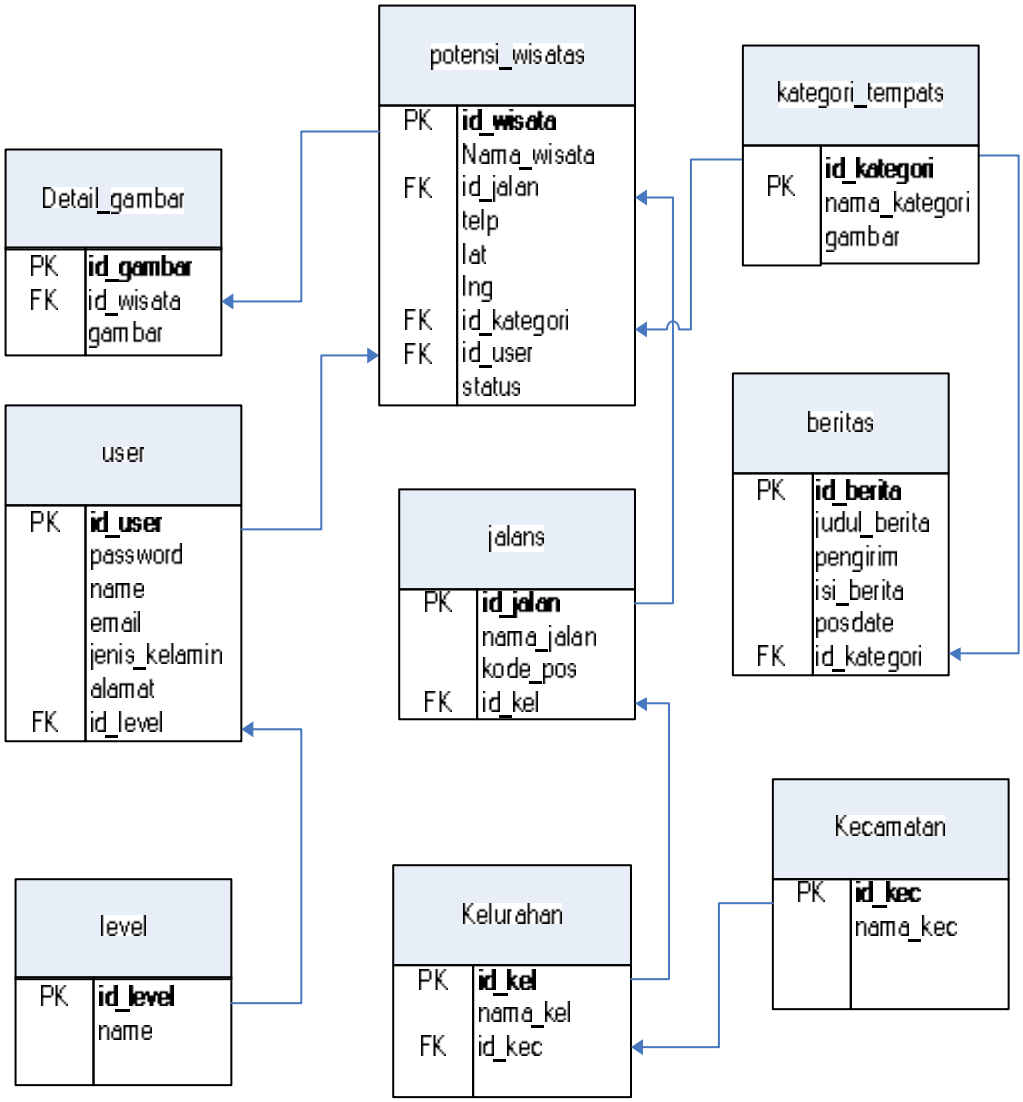
	id_kel	[0-9]
	nama_kel	[A-Z | a-z]
	id_kec	[0-9]
8	Nama alir data	Data Kecamatan
	Where used / used	Admin–proses 8.0(input),proses 8.0 (output)
	Deskripsi	Pengolahan data kecamatan
	Struktur data	id_kec+nama_kec
	id_kec	[0-9]
	nama_kec	[A-Z | a-z]
9	Nama alir data	Data user
	Where used / used	Admin, proses 6.0(input),proses 6.0 (output)
	Deskripsi	Pengolahan user
	Struktur data	id_user+username+password+name +email+jenis_kelamin+notelp+alamat+id_level
	id_user	[0-9]
	username	[A-Z | a-z]
	password	[A-Z | a-z]
	name	[A-Z | a-z]
	email	[0-9]
	jenis_kelamin	[A-Z | a-z]
	notelp	[A-Z | a-z]
	alamat	[0-9]
	id_level	[0-9]

3.4 Perancangan Sistem

Perancangan sistem merupakan bagian dari metedologi penelitian dan pengembangan suatu perangkat lunak yang dilakukan setelah tahap analisis yang bertujuan untuk memberikan gambaran secara terperinci.

3.4.1 Skema Relasi

Dibawah ini adalah skema relasi yang terjadi pada databse hasil konversi. Meskipun hasil konversi, dalam database ini kita boleh manambahkan beberapa atribut yang sesuai dengan kasus yang akan kita tangani.



Gambar 3.14 Skema Relasi WebGIS Pariwisata Kota Tasikmalaya

3.4.2 Struktur Table

Struktur tabel digunakan untuk menggambarkan tabel-tabel beserta field-field yang digunakan secara lebih detail, dimana tipe data dan ukuran suatu field serta kunci-kunci (*primary key* atau *foreign key*) disajikan. Tabel-tabel yang digunakan sebagai berikut:

1. Tabel Potensi_wisatas

Digunakan untuk menampung data wisata beserta koordinat.

Tabel 3.10 Struktur Tabel Potensi_Wisatas

Nama field	Tipe Data	Ukuran Data	Kunci
id_wisata	Integer	11	PK
nama_wisata	Varchar	100	
alamat	Varchar	255	FK Dari tabel jalans
telpon	Varchar	50	
keterangan	Varchar	1000	
lat	Varchar	30	
lng	Varchar	30	
id_kategori	Integer	11	FK Dari tabel kategori_tempats
status	Varchar	20	
id_user	Integer	11	FK Dari tabel user

2. Tabel Jalans

Tabel jalans digunakan untuk menampung jalan.

Tabel 3.11 Struktur Tabel Jalans

Nama field	Tipe Data	Ukuran Data	Kunci
id_jalan	Integer	11	PK
nama_jalan	Varchar	255	
kode_pos	Integer	8	
id_kel	integer	11	FK Dari tabel kelurahan

3. Tabel Kelurahan

Tabel kelurahan digunakan untuk menampung data kelurahan.

Tabel 3.12 Struktur Tabel Kelurahan

Nama field	Tipe Data	Ukuran Data	Kunci
id_kel	Integer	11	PK
nama_kel	Varchar	255	
id_kec	integer	11	FK Dari tabel kecamatan

4. Tabel Kecamatan

Tabel kecamatan digunakan untuk menampung data kecamatan.

Tabel 3.13 Struktur Tabel Kecamatan

Nama field	Tipe Data	Ukuran Data	Kunci
id_kec	Integer	11	PK
nama_kec	Varchar	255	

5. Tabel Kategori_tempats

Tabel kategori_tempats di gunakan menyimpan data kategori wisata dan kategori berita.

Tabel 3.14 Struktur Tabel Kategori_tempats

Nama field	**Tipe Data**	**Ukuran Data**	**Kunci**
id_kategori	Integer	11	PK
name	varchar	255	
gambar	varchar	255	

6. Tabel Beritas

Tabel beritas untuk menyimpan data berita yang akan di posting.

Tabel.15 Struktur Tabel Beritas

Nama field	**Tipe Data**	**Ukuran Data**	**Kunci**
id_berita	Integer	11	PK
judul_berita	varchar	255	
isi_berita	Text		
pengirim	Varchar	29	
postdate	Date		
id_kategori	Integer	11	FK Dari tabel Kategori_tempats

7. Tabel User

Table user menyimpan data user.

Tabel 3.16 Struktur Tabel User

Nama field	Tipe Data	Ukuran Data	Kunci
id_user	Integer	11	PK
username	Varchar	255	
password	Varchar	255	
name	Varchar	50	
email	Varchar	255	
notelp	Varchar	50	
alamat	Varchar	50	
id_level	Integer	11	FK Dari tabel level
lastlogin	Date		

8. Tabel Level

Tabel level menyimpan data level dari user.

Tabel 3.17 Struktur Tabel Level

Nama field	Tipe Data	Ukuran Data	Kunci
id_level	Integer	11	PK
name	varchar	255	

9. Tabel Detail_gambar

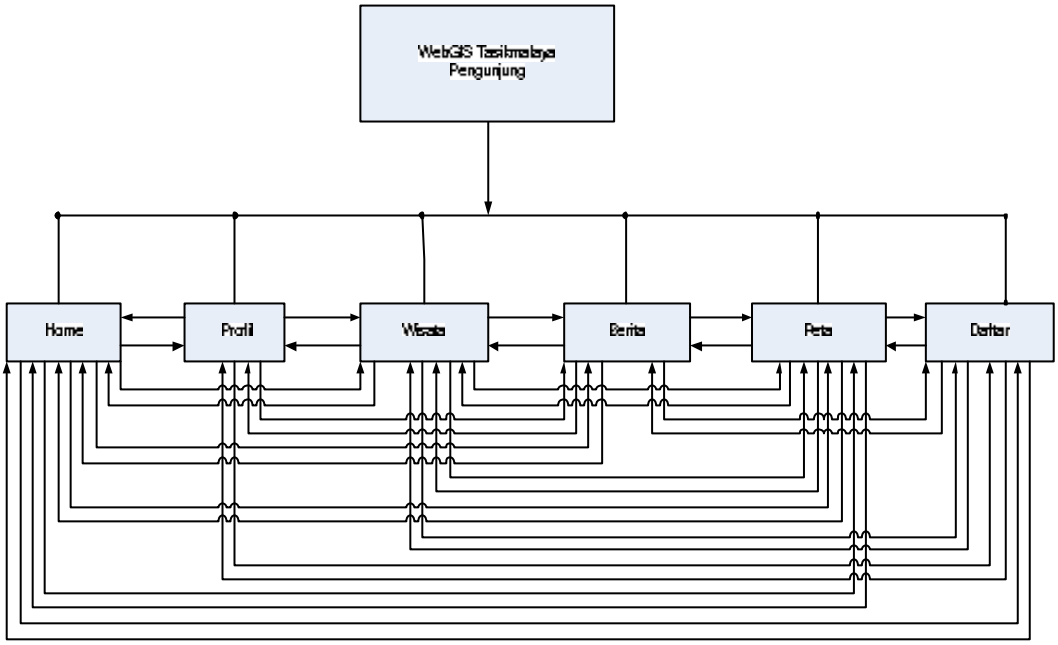
Tabel detail_gambar menyimpan data gambar dari wisata yang di inputkan.

Tabel 3.18 Struktur Tabel Detail_gambar

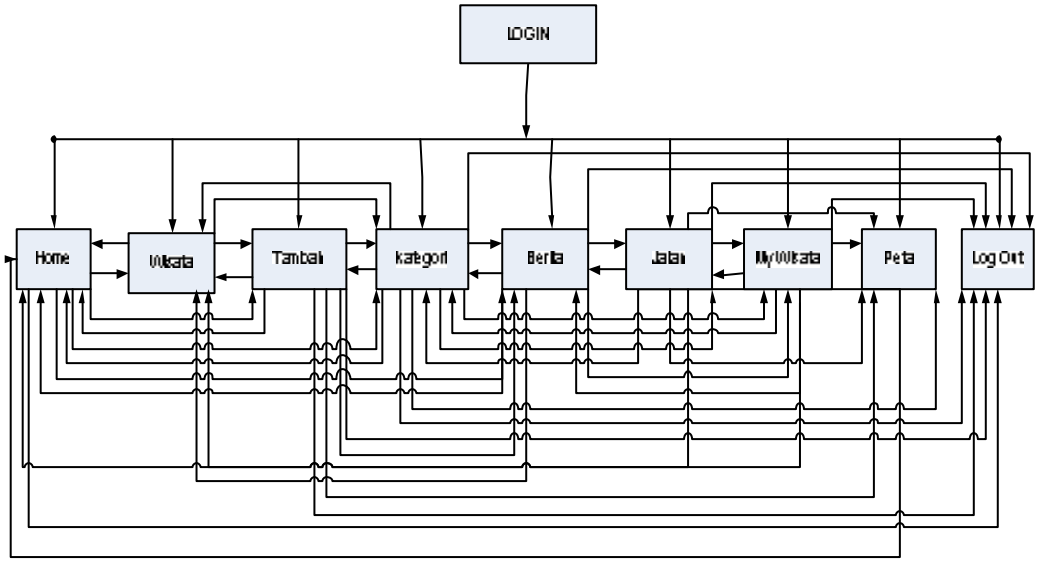
Nama field	Tipe Data	Ukuran Data	Kunci
id_gambar	Integer	11	PK
id_wisata	integer	11	FK Dari tabel potensi_wisatas
gambar	varchar	255	

3.4.3 Perancangan Struktur Menu

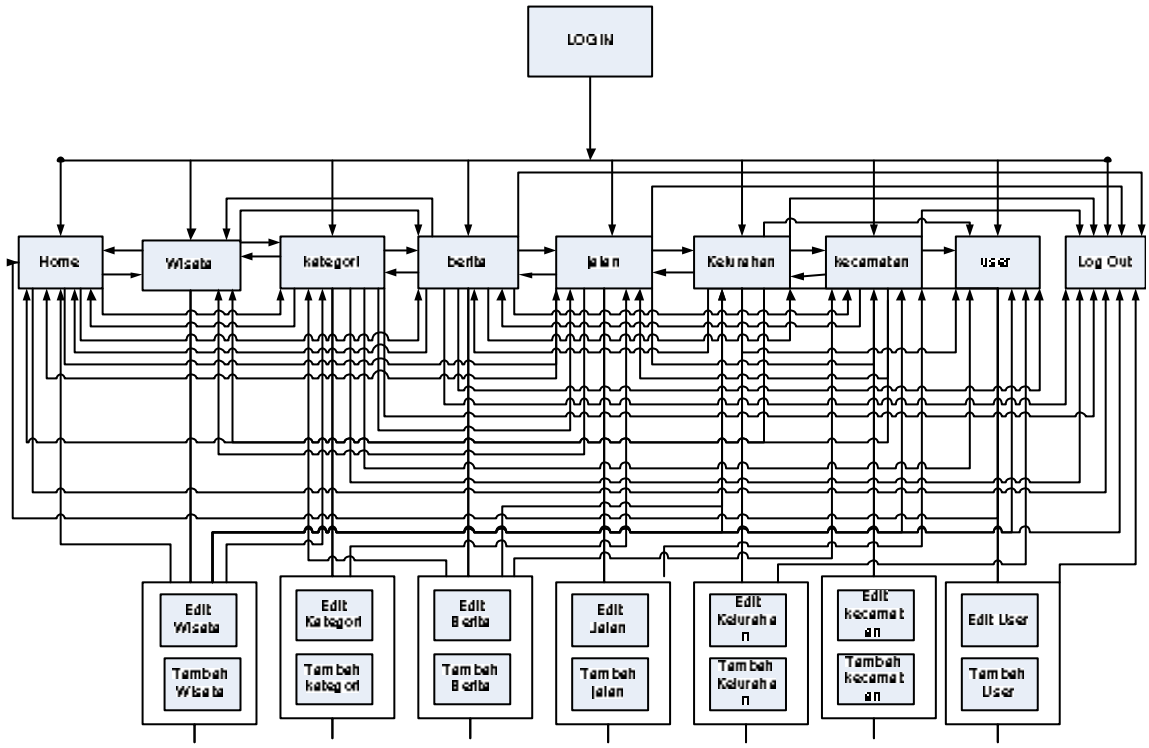
Berikut ini adalah perancangan struktur menu yang akan diterapkan pada program dan dirancang seefisien mungkin untuk mengurangi kompleksitas sistem.



Gambar 3.15 Perancangan Menu Pengunjung



Gambar 3.16 Perancangan Menu Member

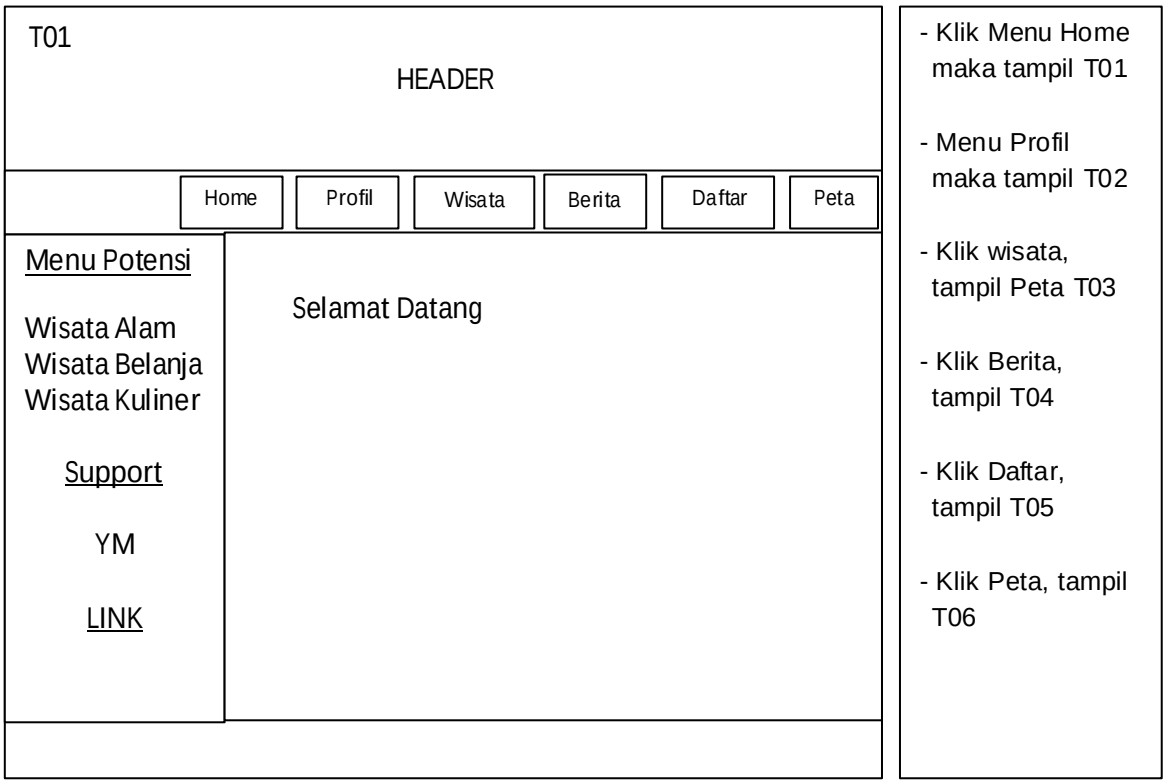


Gambar 3.17 Perancangan Menu Admin

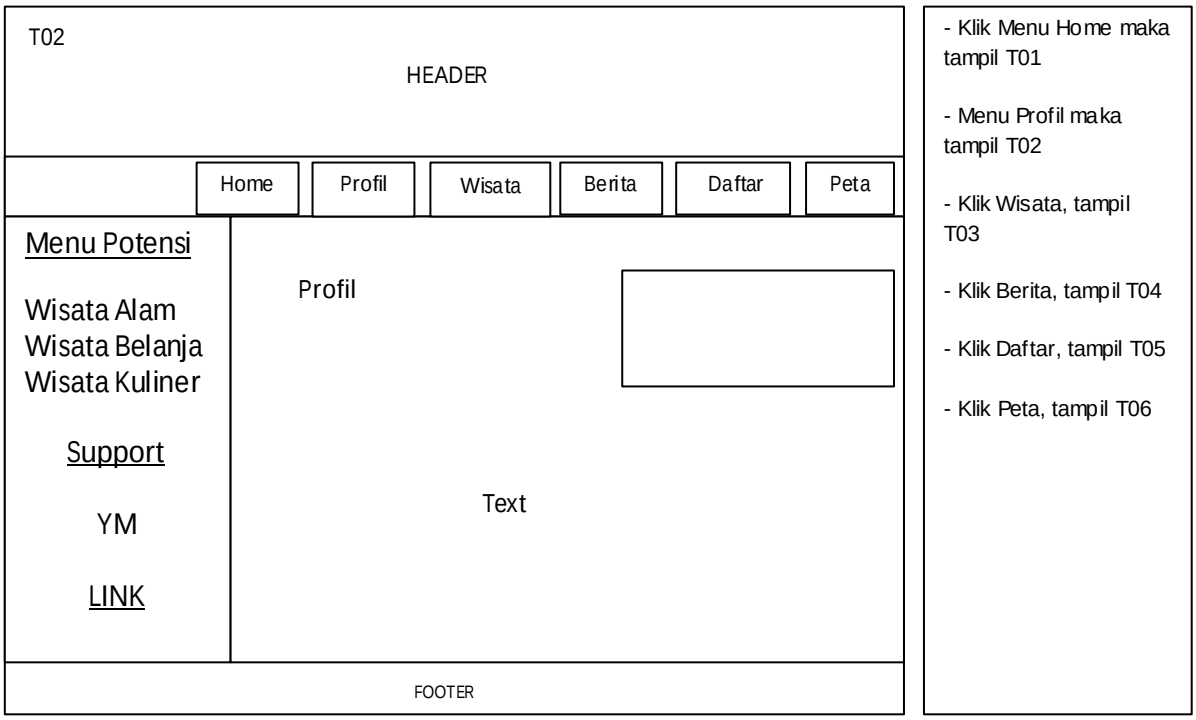
3.4.4 Perancangan Antarmuka

Merancang antarmuka merupakan bagian yang paling penting dari merancang sistem. Biasanya hal tersebut juga merupakan bagian yang paling sulit, karena dalam merancang antarmuka harus memenuhi tiga persyaratan yaitu sebuah antarmuka harus sederhana, sebuah antarmuka harus lengkap, dan sebuah antarmuka harus memilki kinerja yang cepat.

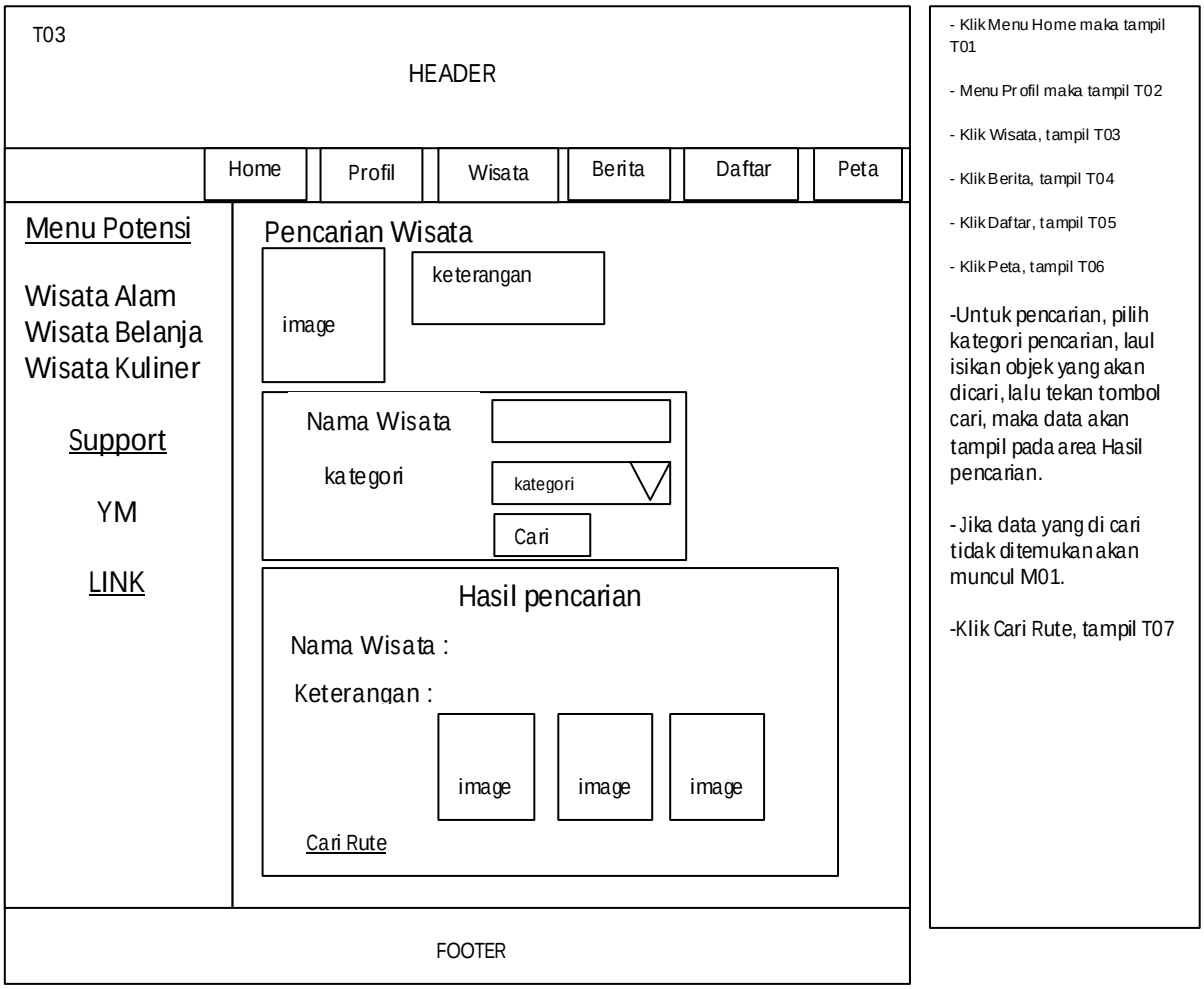
3.4.4.1 Perancangan Antarmuka Pengunjung



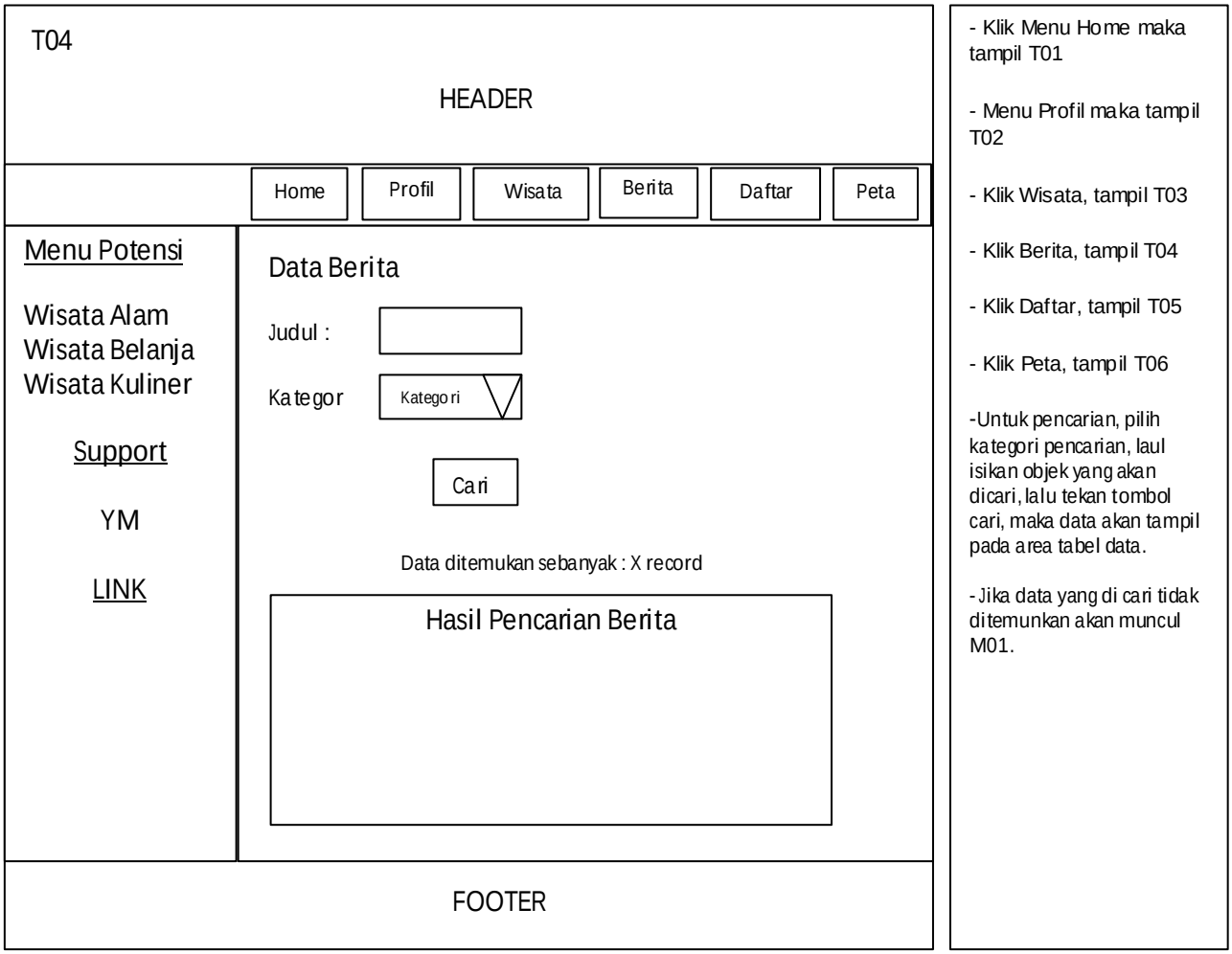
Gambar 3.18 Tampilan Menu Utama - Pengunjung



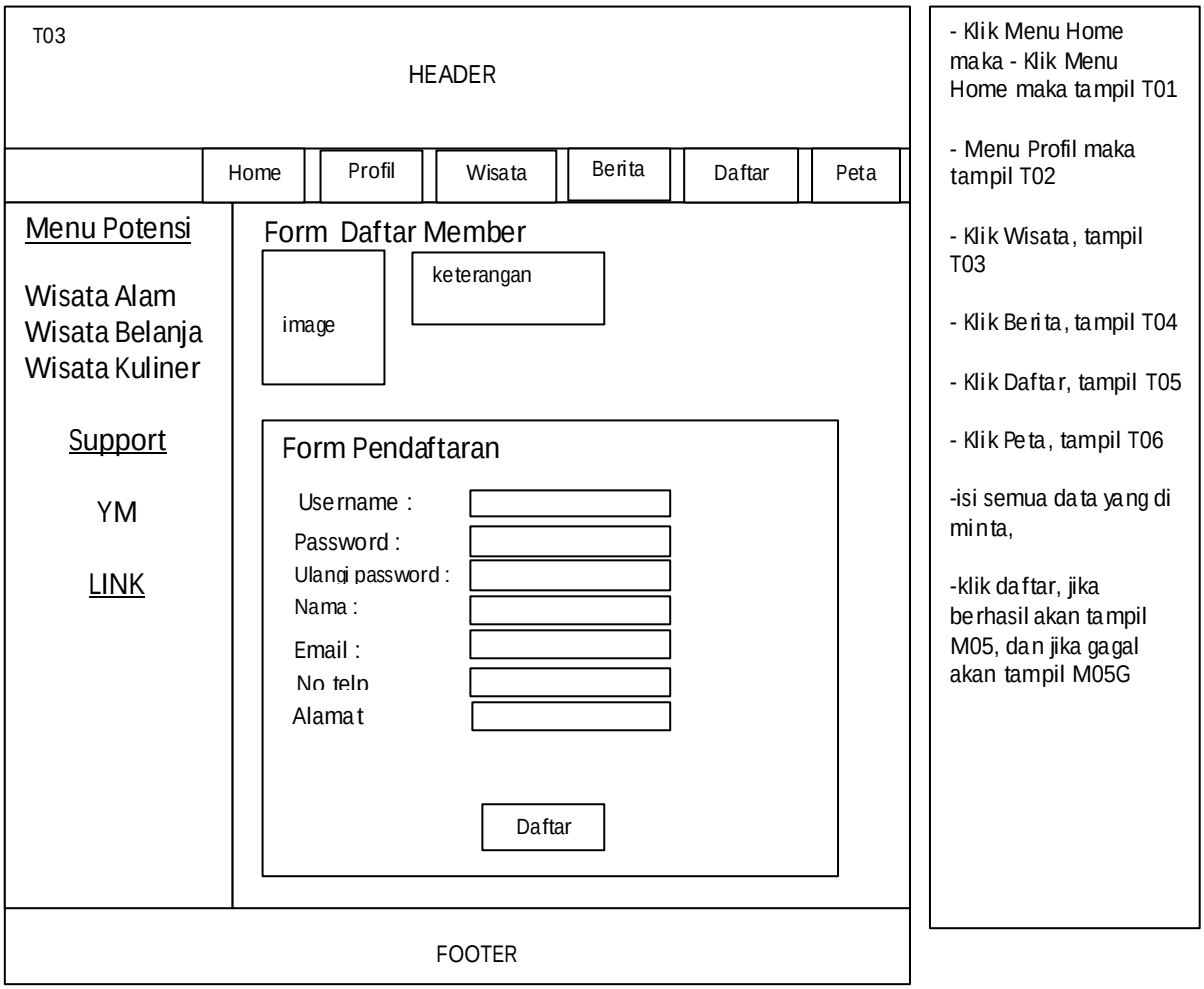
Gambar 3.19 Tampilan Halaman Profil



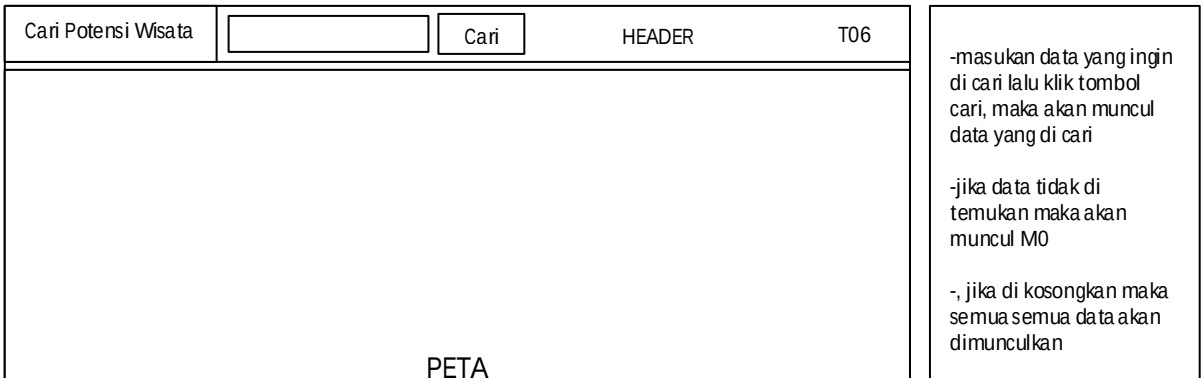
Gambar 3.20 Tampilan Halaman Data Wisata

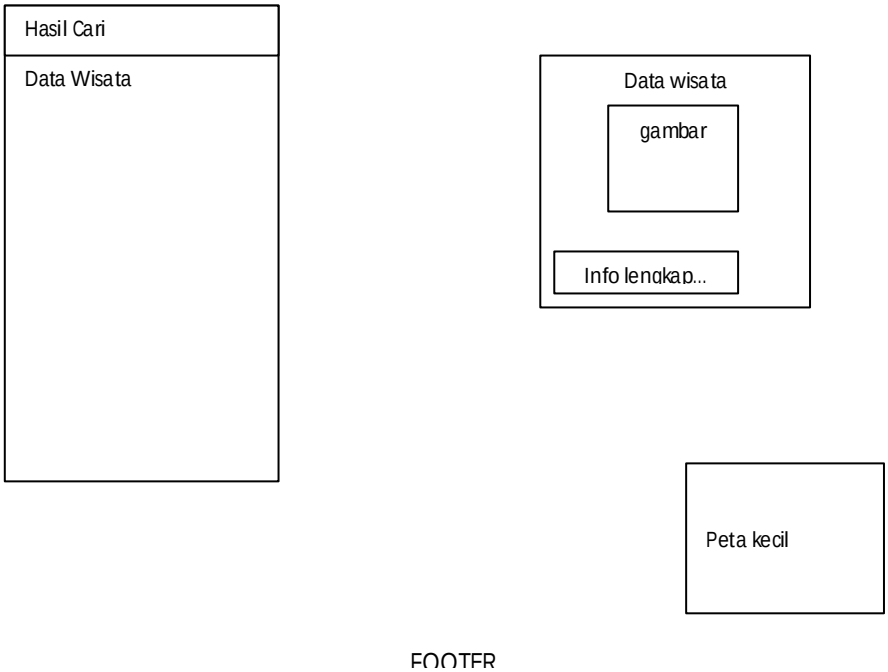


Gambar 3.21 Tampilan Halaman Data Berita

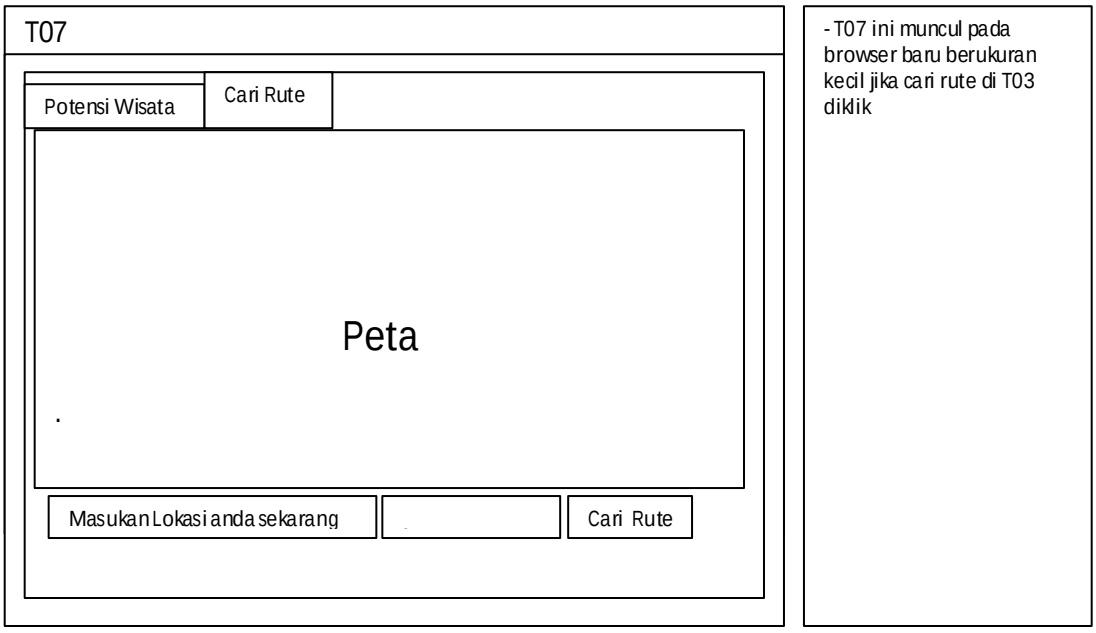


Gambar 3.22 Tampilan Halaman Daftar



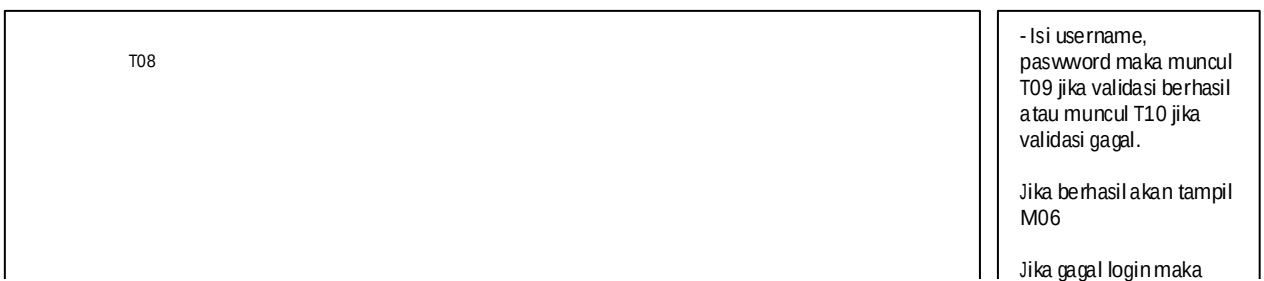


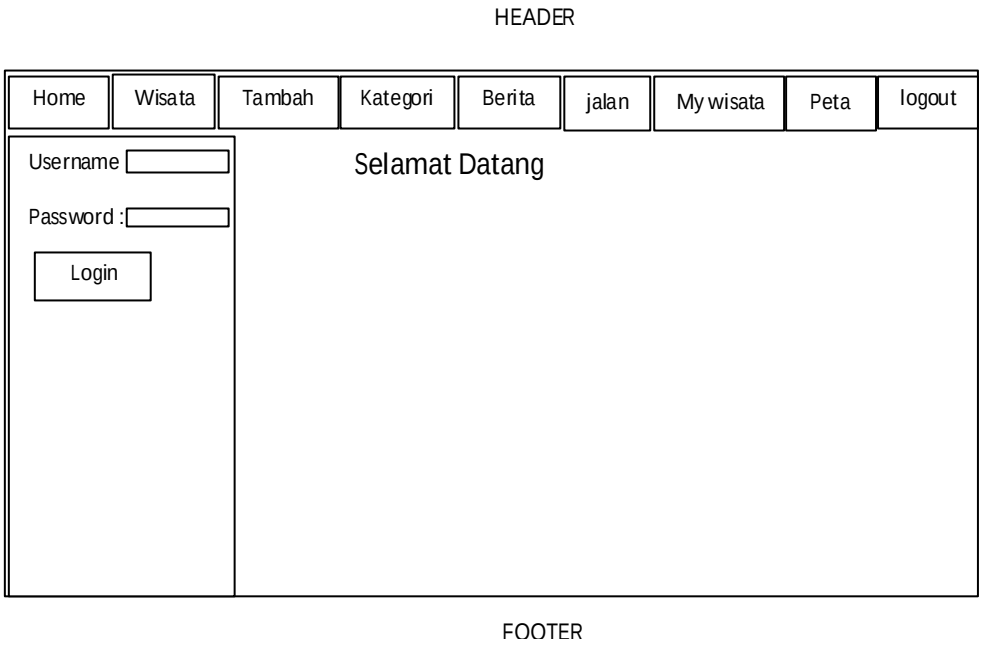
Gambar 3.23 Tampilan Halaman peta



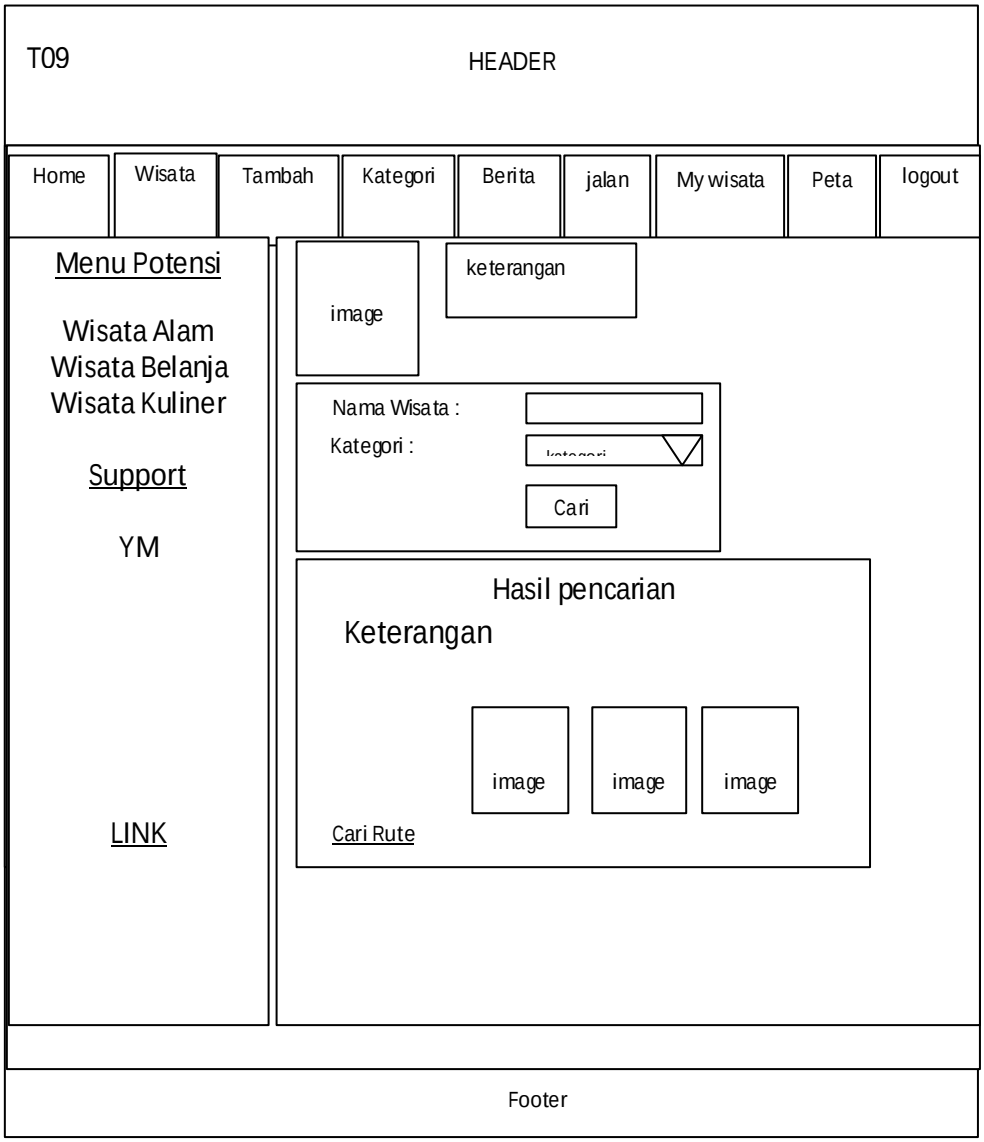
Gambar 3.24 Tampilan Halaman Cari Rute

1.2.4.2 Perancangan Antarmuka Member





Gambar 3.25 Tampilan Login Member



- Isi username, paswword maka muncul T09 jika validasi berhasil atau muncul T10 jika validasi gagal.
-klik Logout untuk keluar dari pengolahan data dan akan muncul T08.

-Klik wisata, maka muncul T09

- Klik Tambah, maka muncul T10.

- Klik Kategori, maka muncul T11.

- Klik Berita, maka muncul T12.

- Klik Jalan, maka muncul T13.

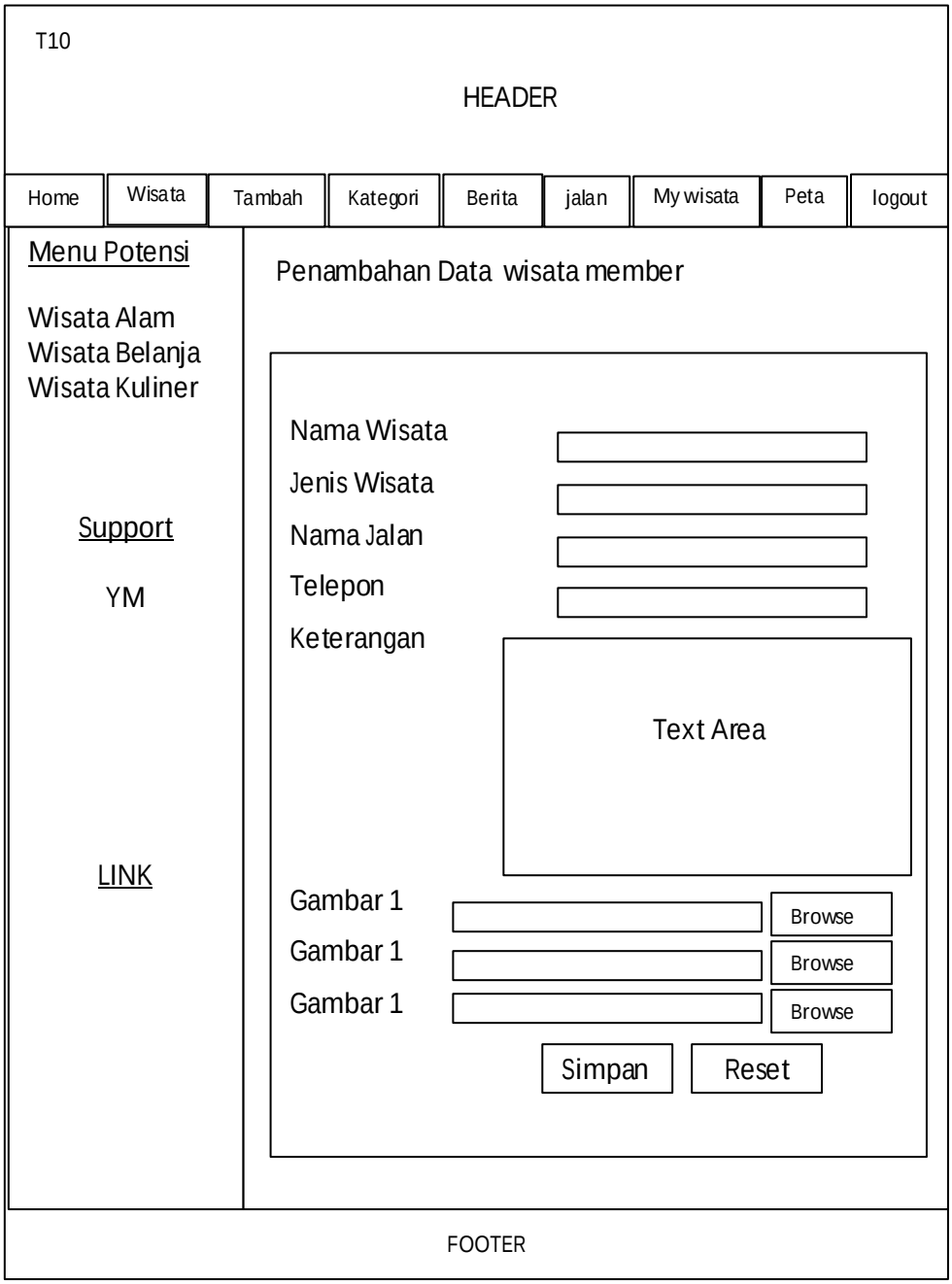
- Klik My Wisata, maka muncul T14.

- Klik Peta, maka muncul T06.

-masukan data yang ingin dicari,dan pilih kategori untuk memilih kategori lalu klik tombol cari, maka data akan dimunculkan

-Jika data tidak ketemu maka akan tampil M01

Gambar 3.26 Tampilan Menu Member



-Klik wisata, maka muncul T09

- Klik Tambah, maka muncul T10.

- Klik Kategori, maka muncul T11.

- Klik Berita, maka muncul T12.

- Klik Jalan, maka muncul T13.

- Klik My Wisata, maka muncul T14.

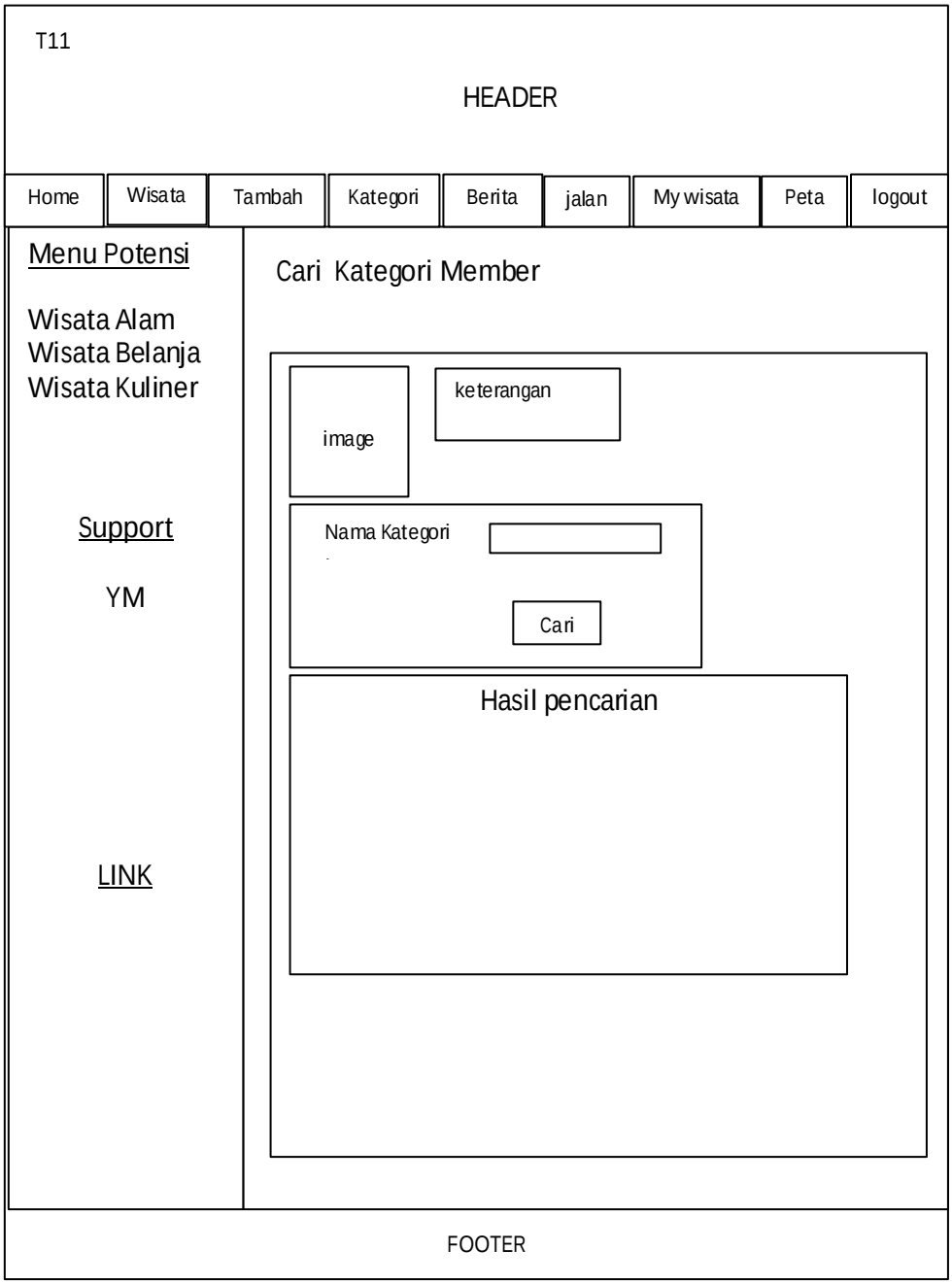
- Klik Peta, maka muncul T06

-klik Logout untuk keluar dari pengolahan data dan akan muncul T08.

-isi semua data yang tersedia, dan klik simpan untuk menyimpan data

-jika penyimpanan data berhasil maka akan muncul M02

Gambar 3.27 Tampilan Tambah Data Wisata Member



-Klik wisata, maka muncul T09

- Klik Tambah, maka muncul T10.

- Klik Kategori, maka muncul T11.

- Klik Berita, maka muncul T12.

- Klik Jalan, maka muncul T13.

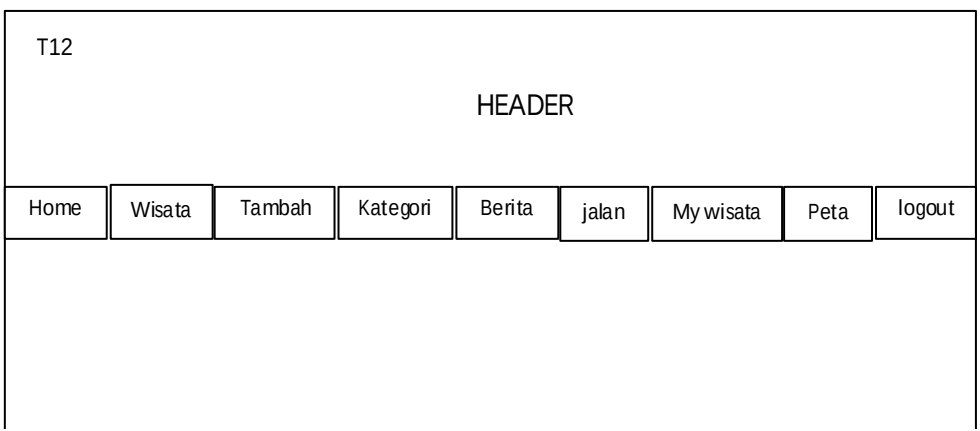
- Klik My Wisata, maka muncul T14.

- Klik Peta, maka muncul T06.

-masukan data yang ingin dicari,dan pilih kategori untuk memilih kategori lalu klik tombol cari, maka data akan cimunculkan

-Jika data tidak ketemu maka akan tampil M01

Gambar 3.28 Tampilan Kategori Member

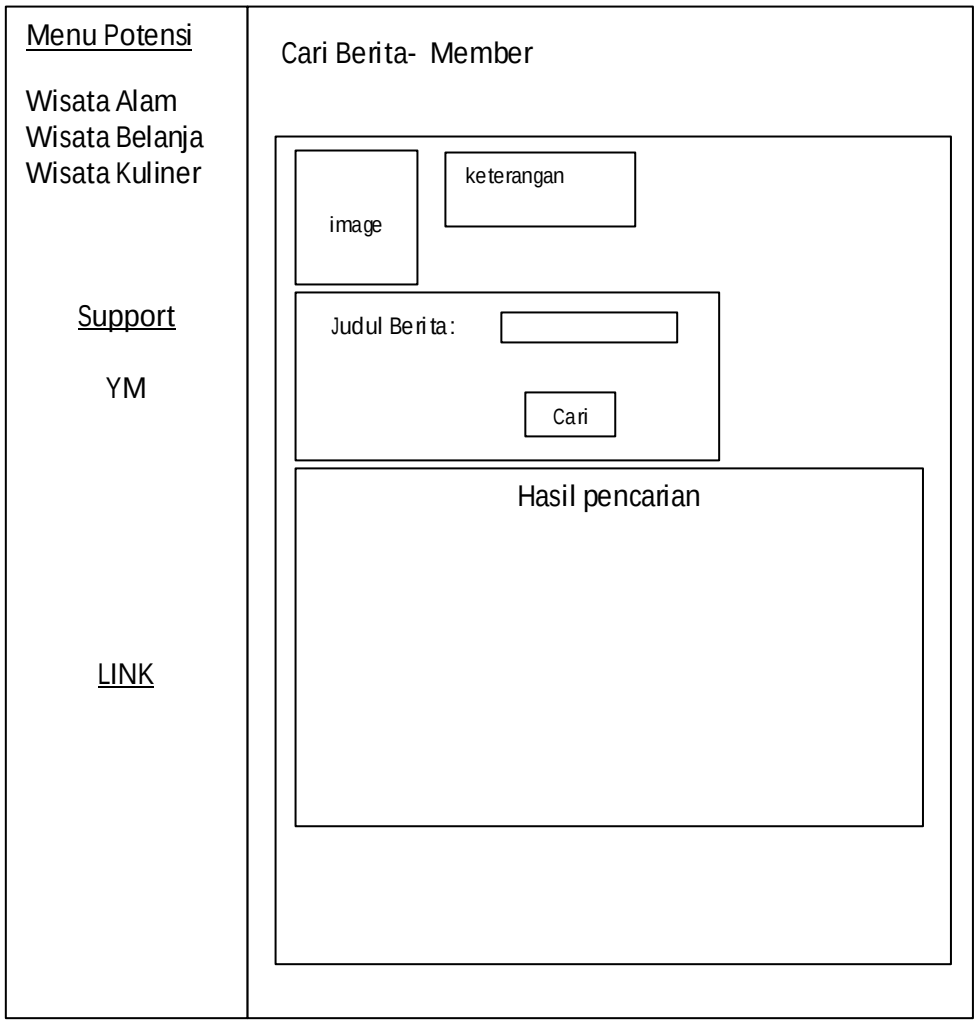


-Klik wisata, maka muncul T09

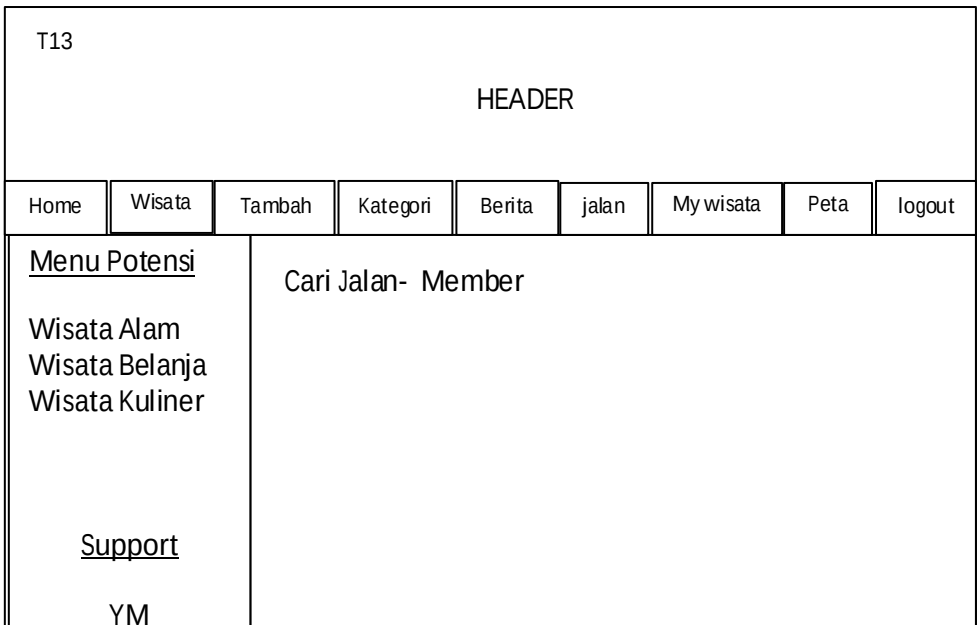
- Klik Tambah, maka muncul T10.

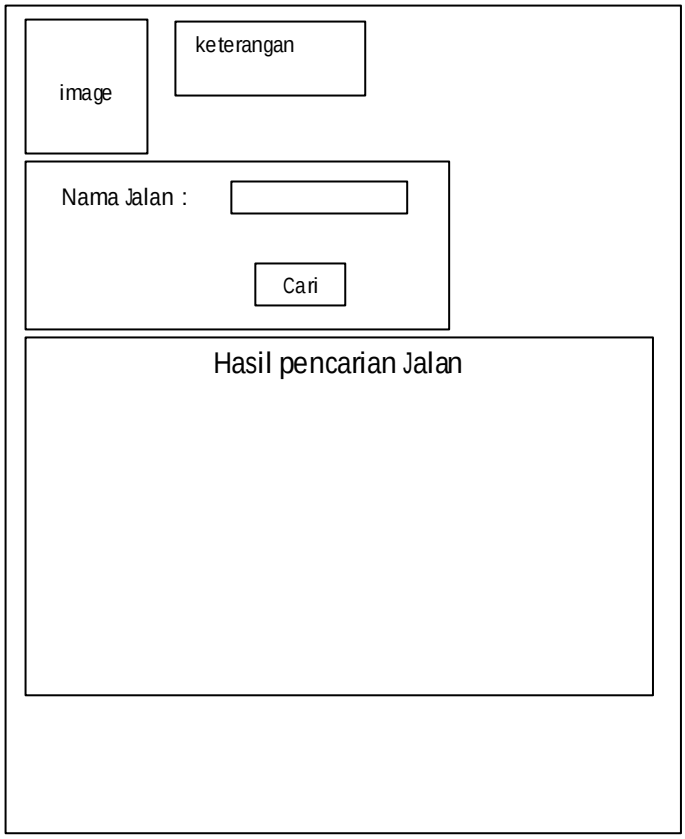
- Klik Kategori, maka muncul T11.

- Klik Berita, maka muncul T12.

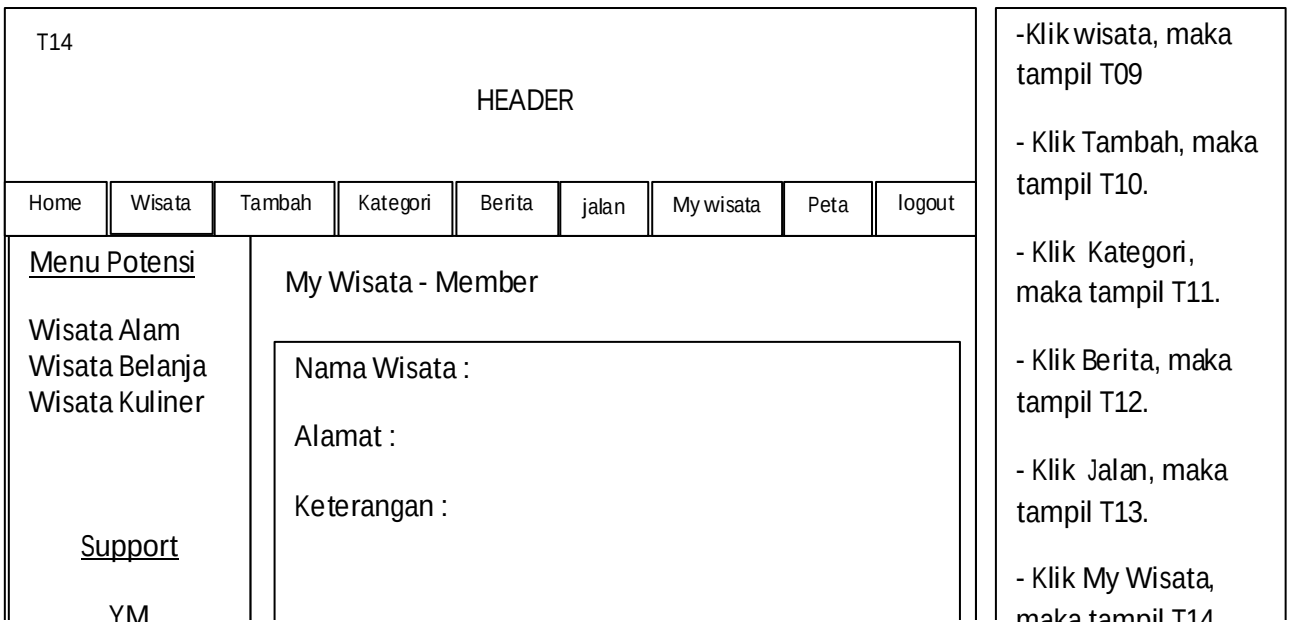


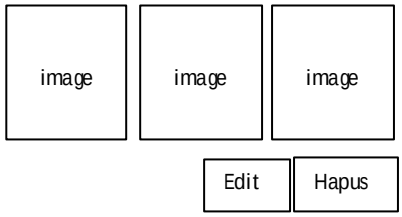
Gambar 3.29 Tampilan Berita Member





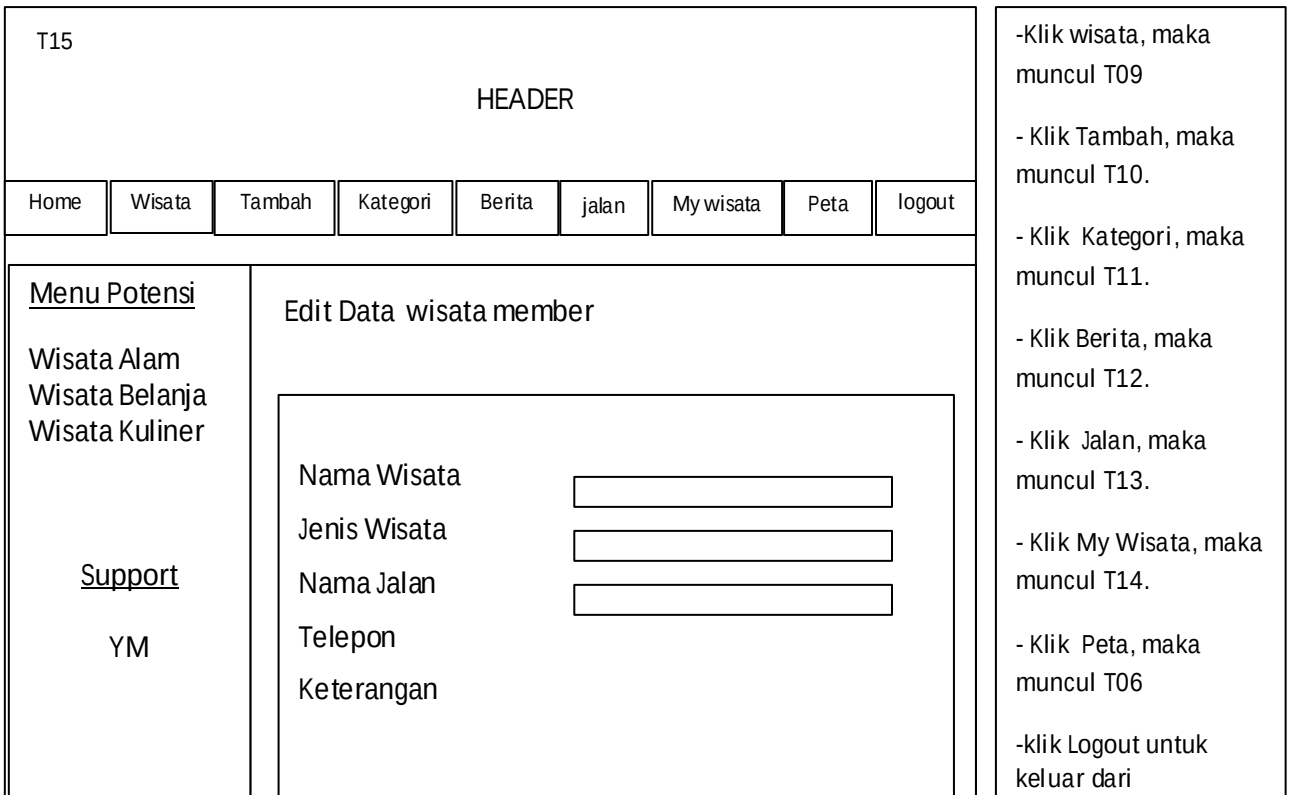
Gambar 3.30 Tampilan Jalan Member

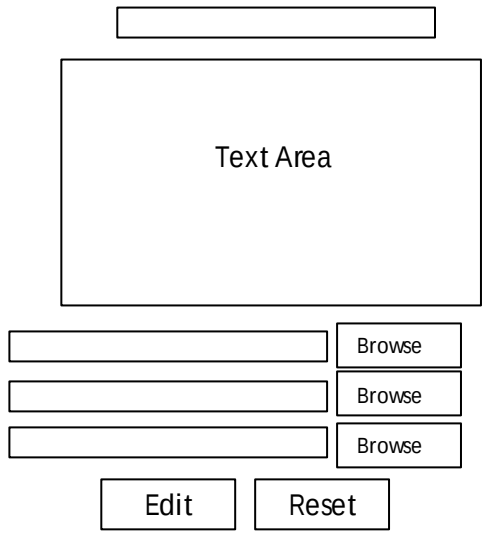




FOOTER

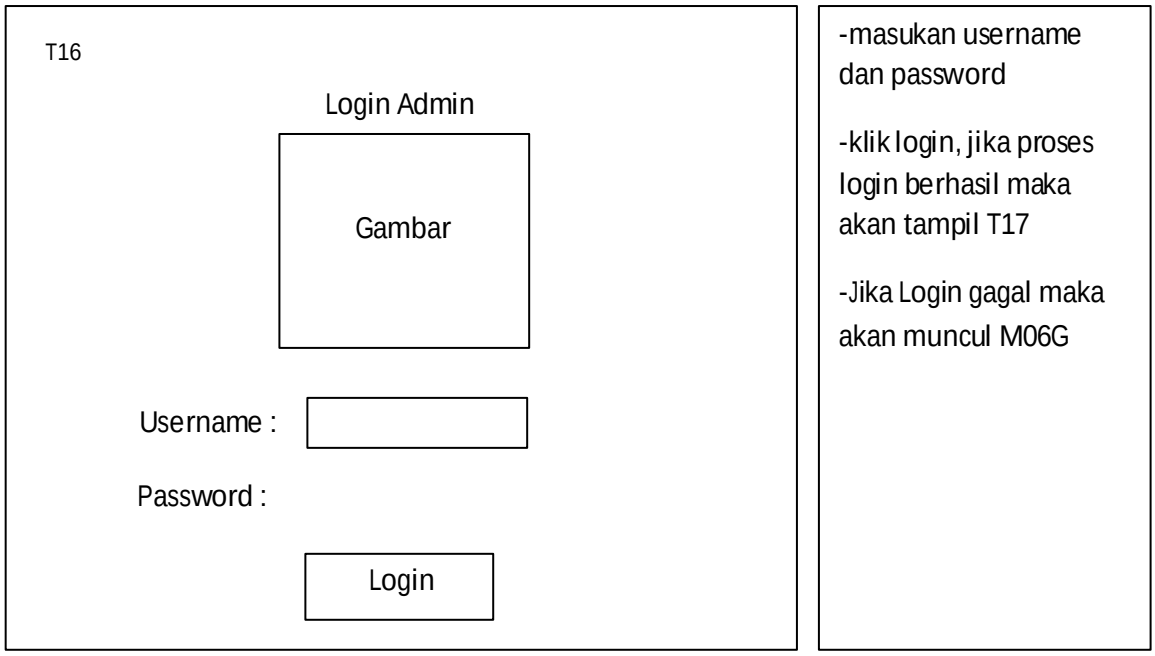
Gambar 3.31 Tampilan My wisata Member

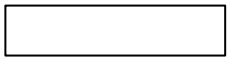




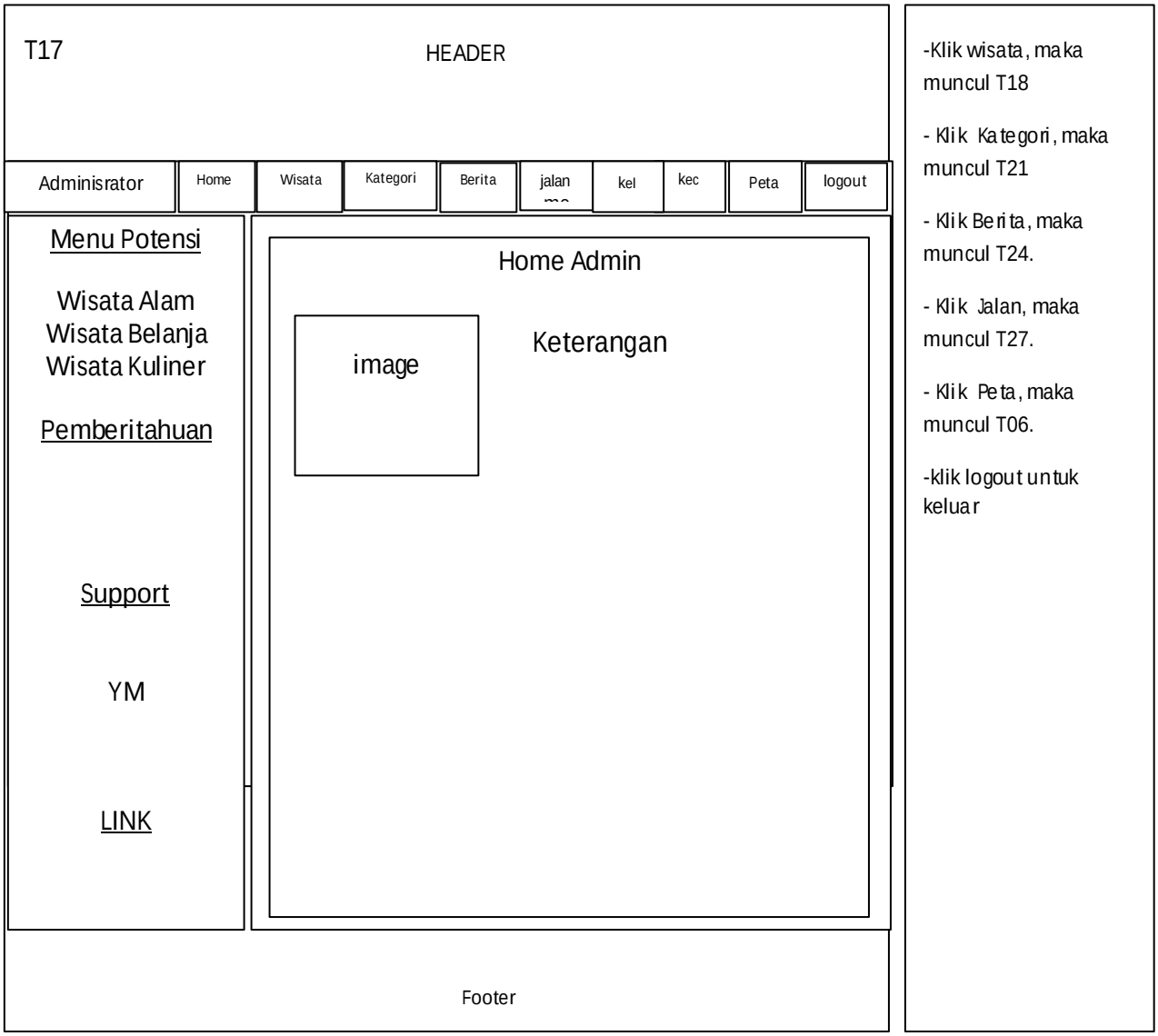
Gambar 3.32 Tampilan Edit Data - Member

1.1.1.3 Perancangan Antarmuka Admin

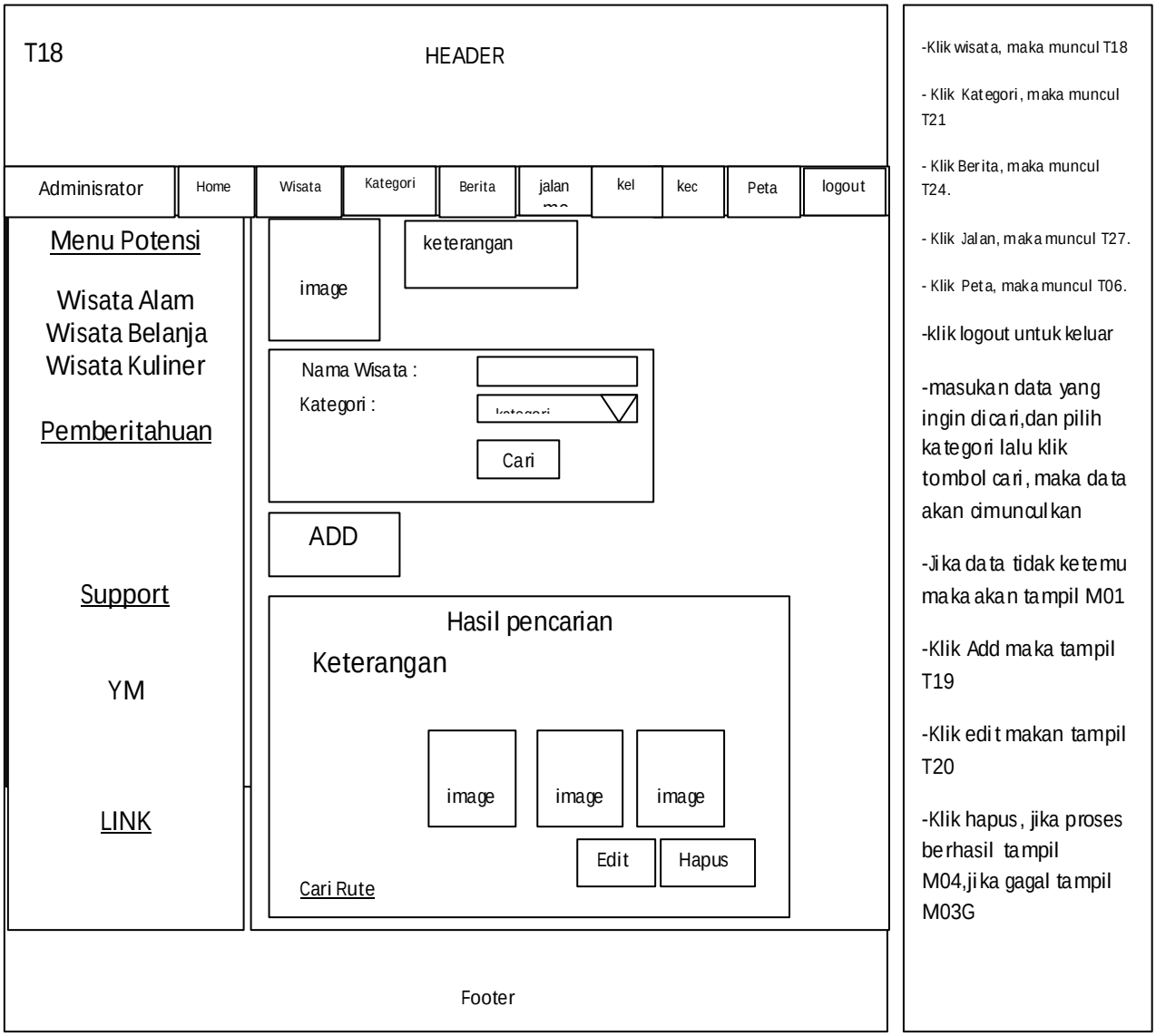




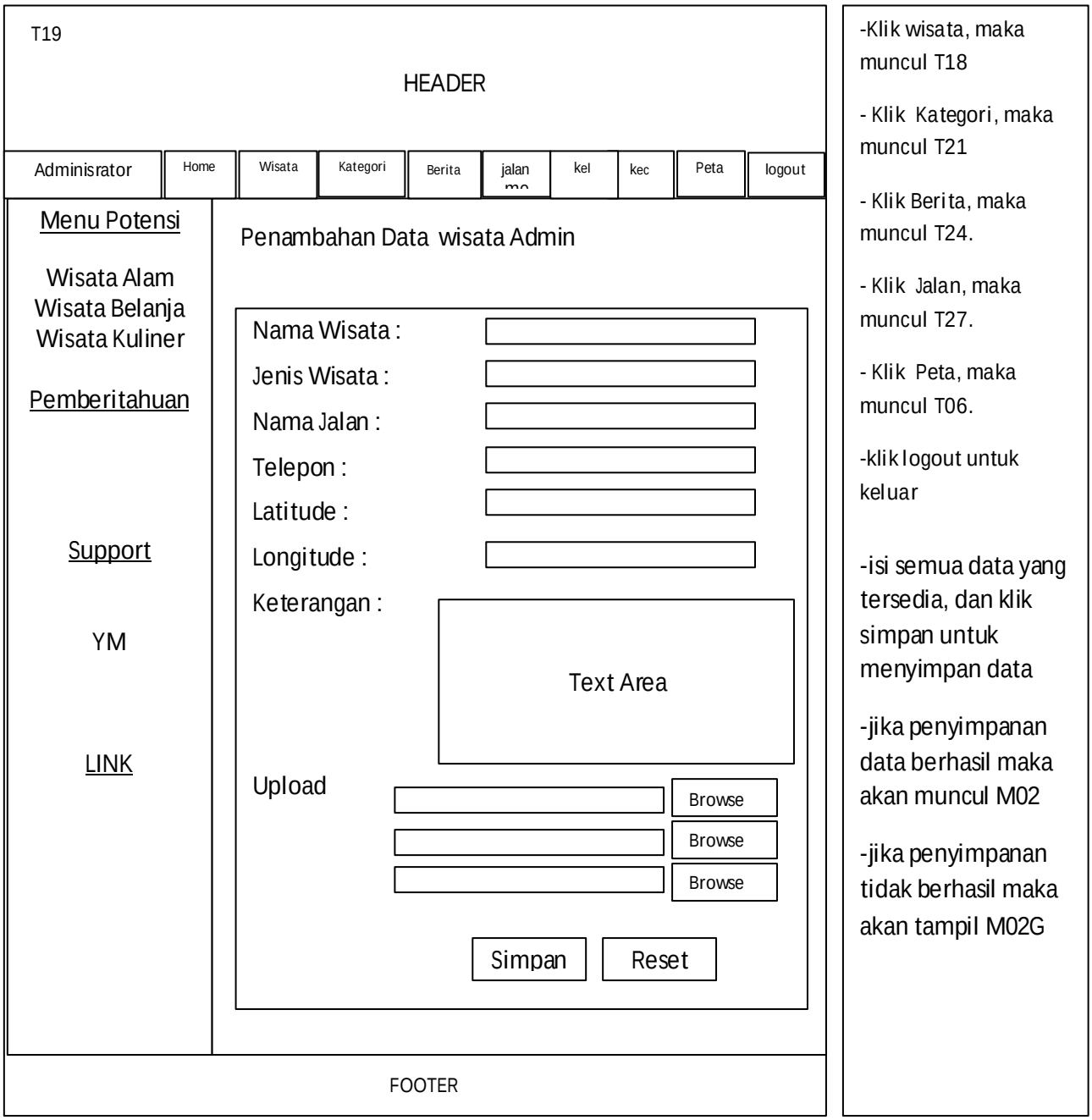
Gambar 3.33 Tampilan Login Admin



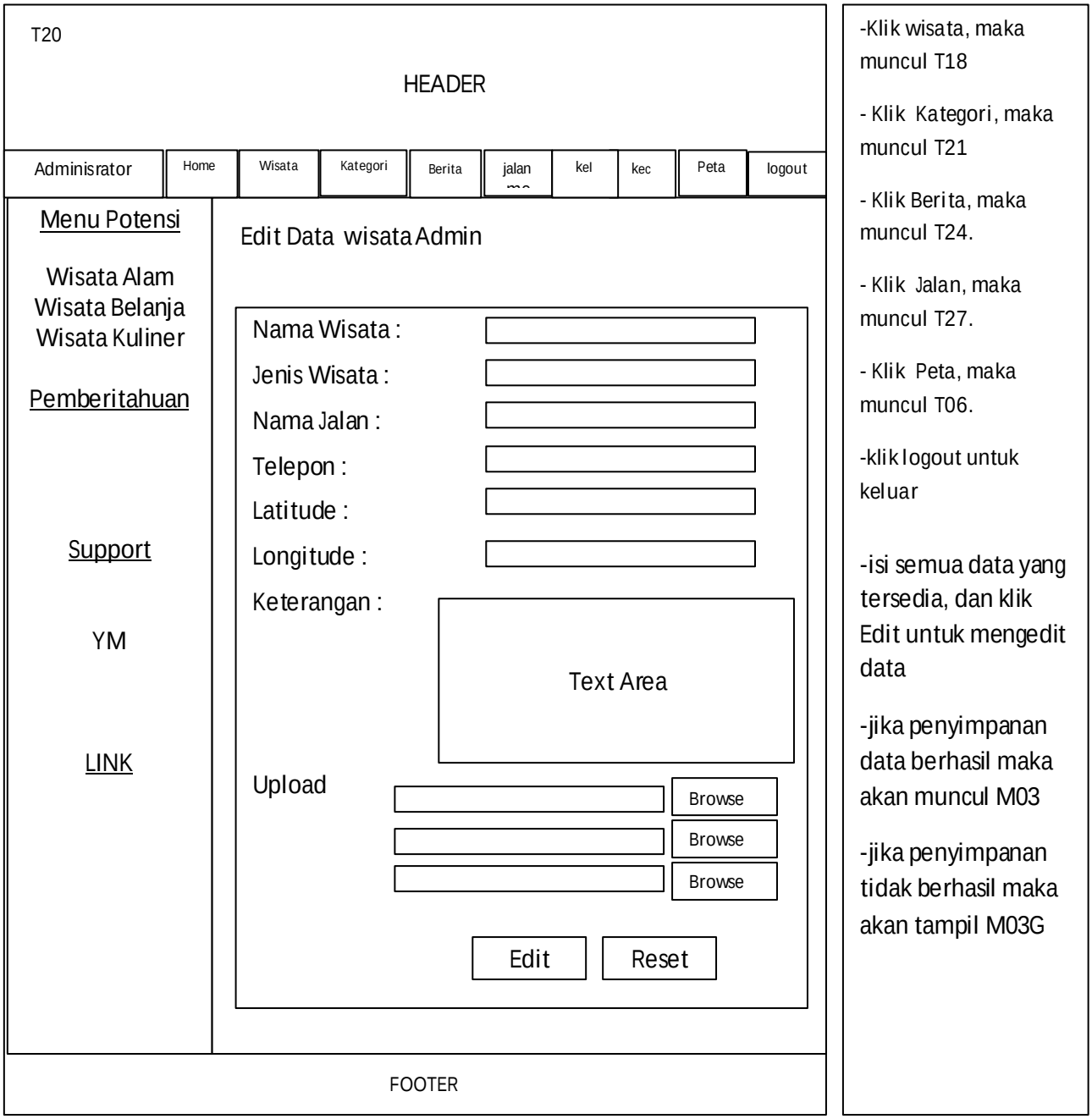
Gambar 3.34 Tampilan Home Admin



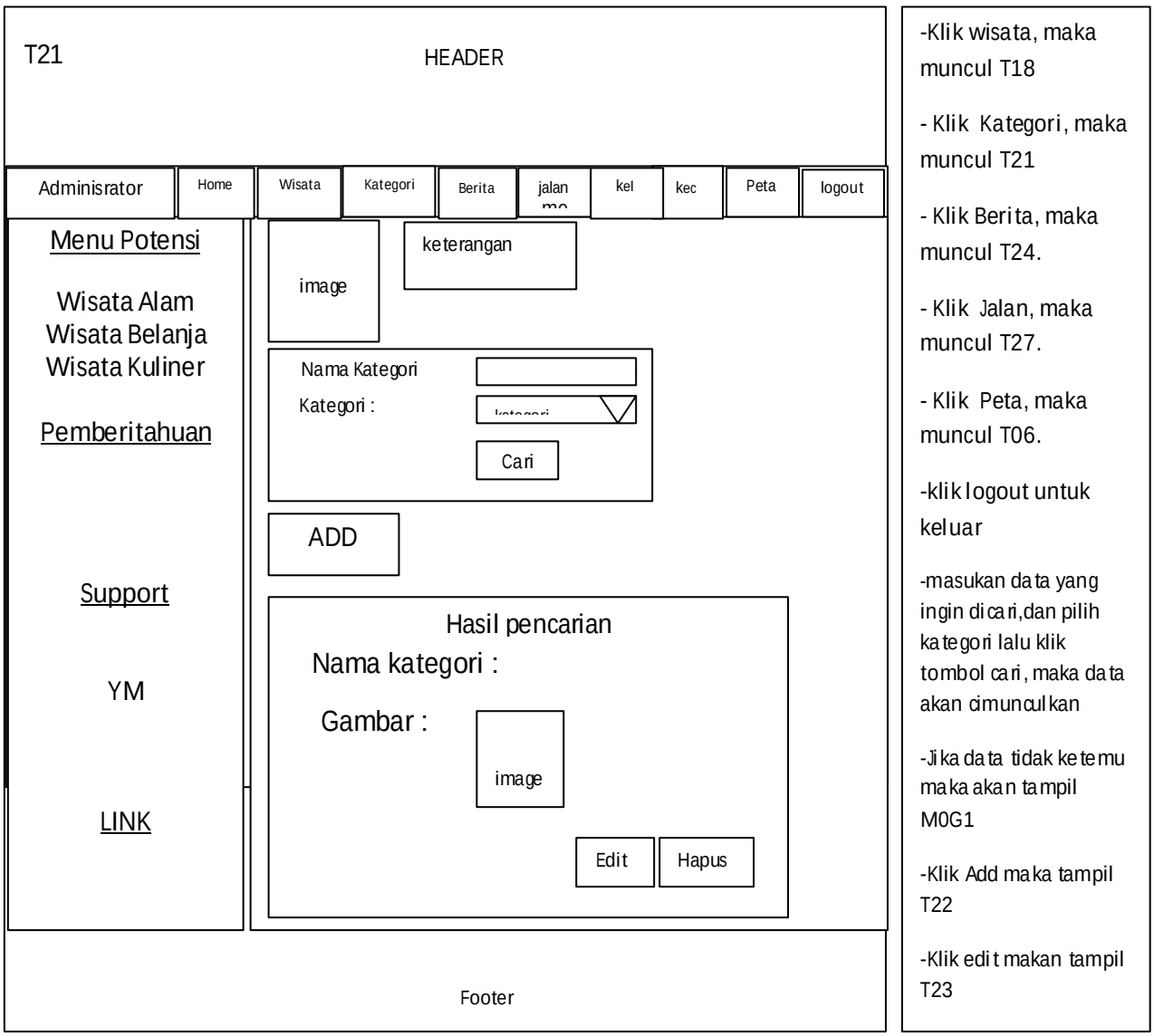
Gambar 3.35 Tampilan Menu Admin



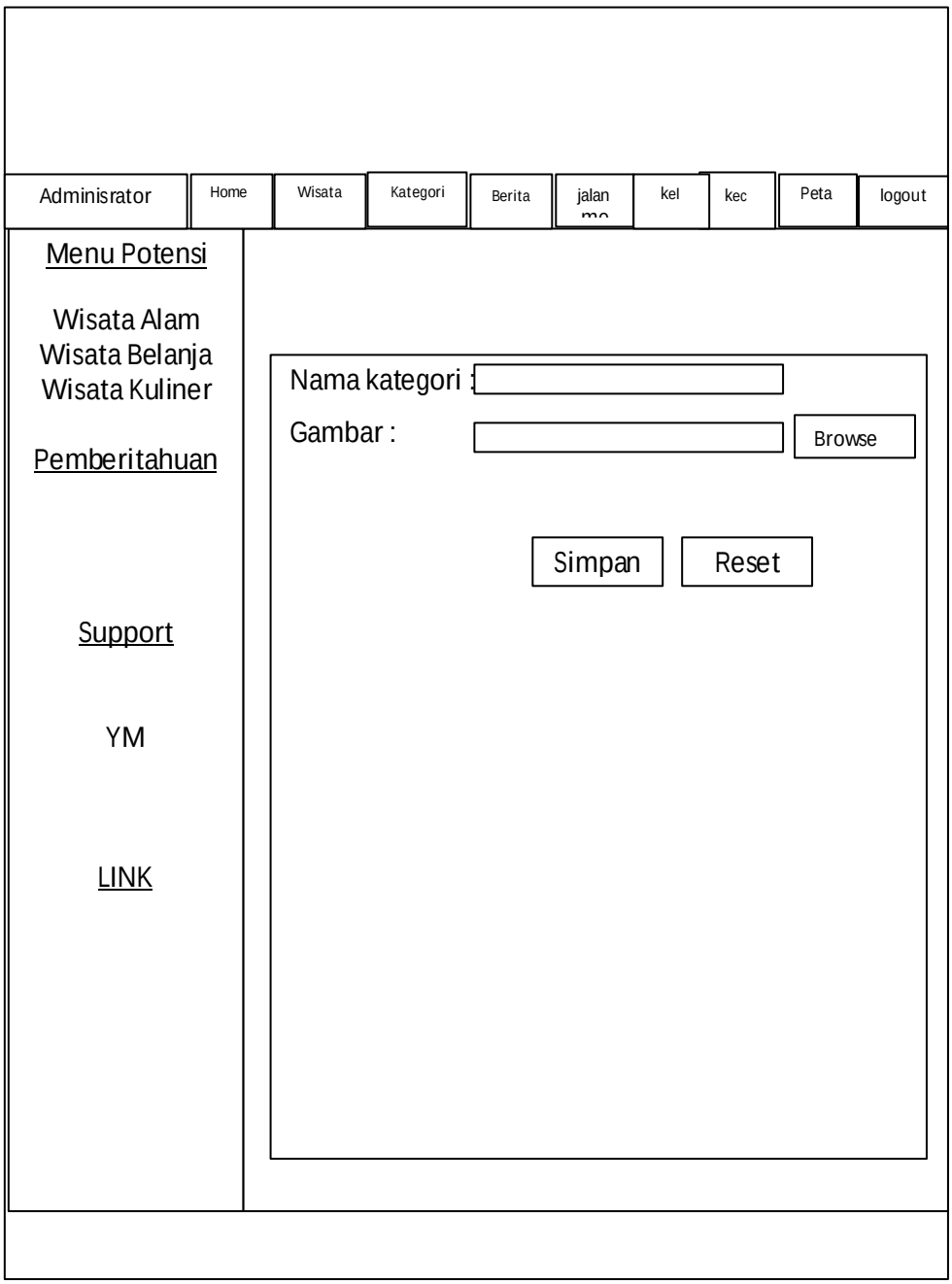
Gambar 3.36 Tampilan Tambah Data Wisata Admin



Gambar 3.37 Tampilan Edit Data Wisata Admin



Gambar 3.38 Tampilan Data Kategori Admin



-Klik wisata, maka muncul T18

- Klik Kategori, maka muncul T21

- Klik Berita, maka muncul T24.

- Klik Jalan, maka muncul T27.

- Klik Peta, maka muncul T06.

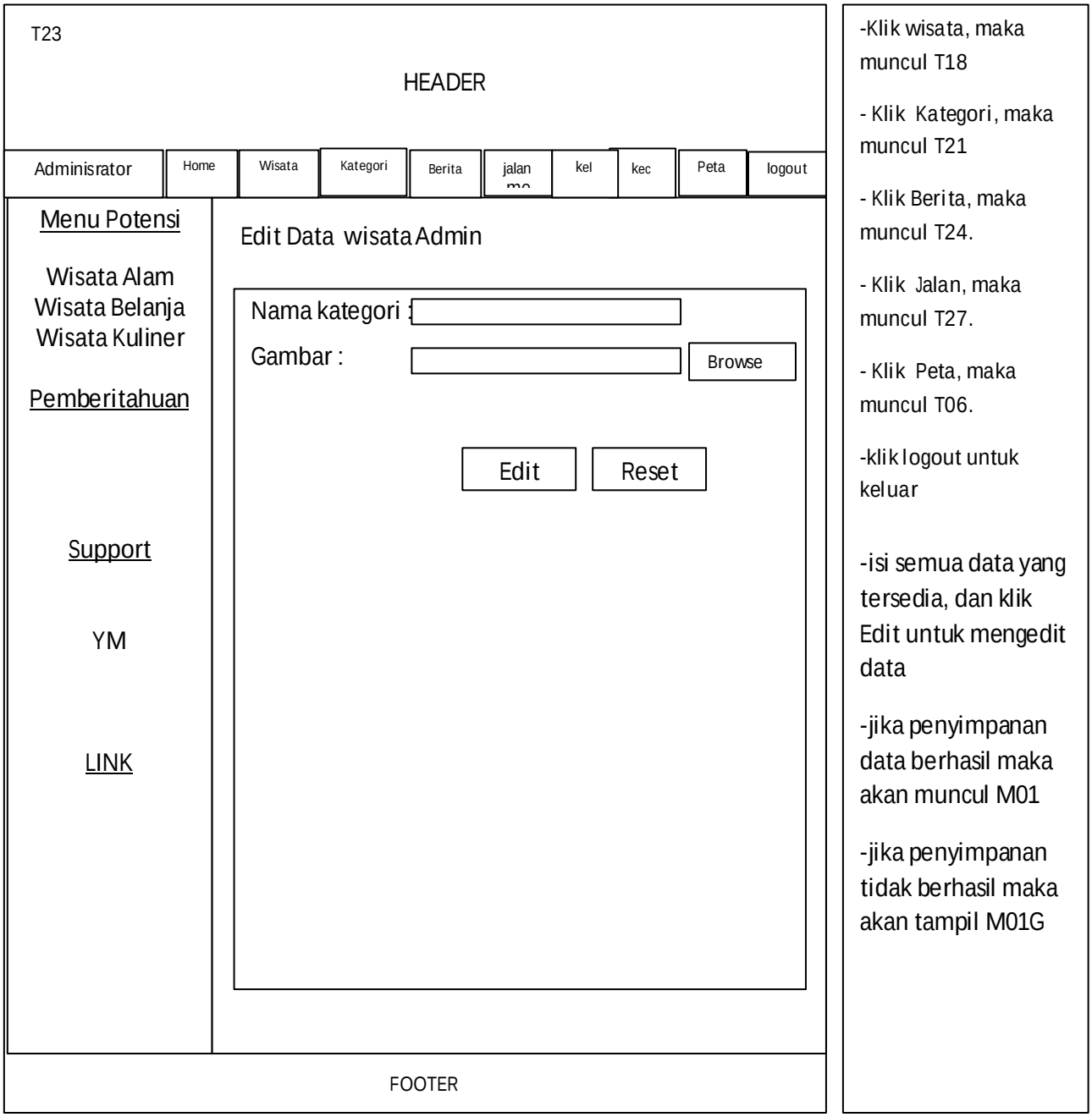
-klik logout untuk keluar

-isi semua data yang tersedia, dan klik simpan untuk menyimpan data

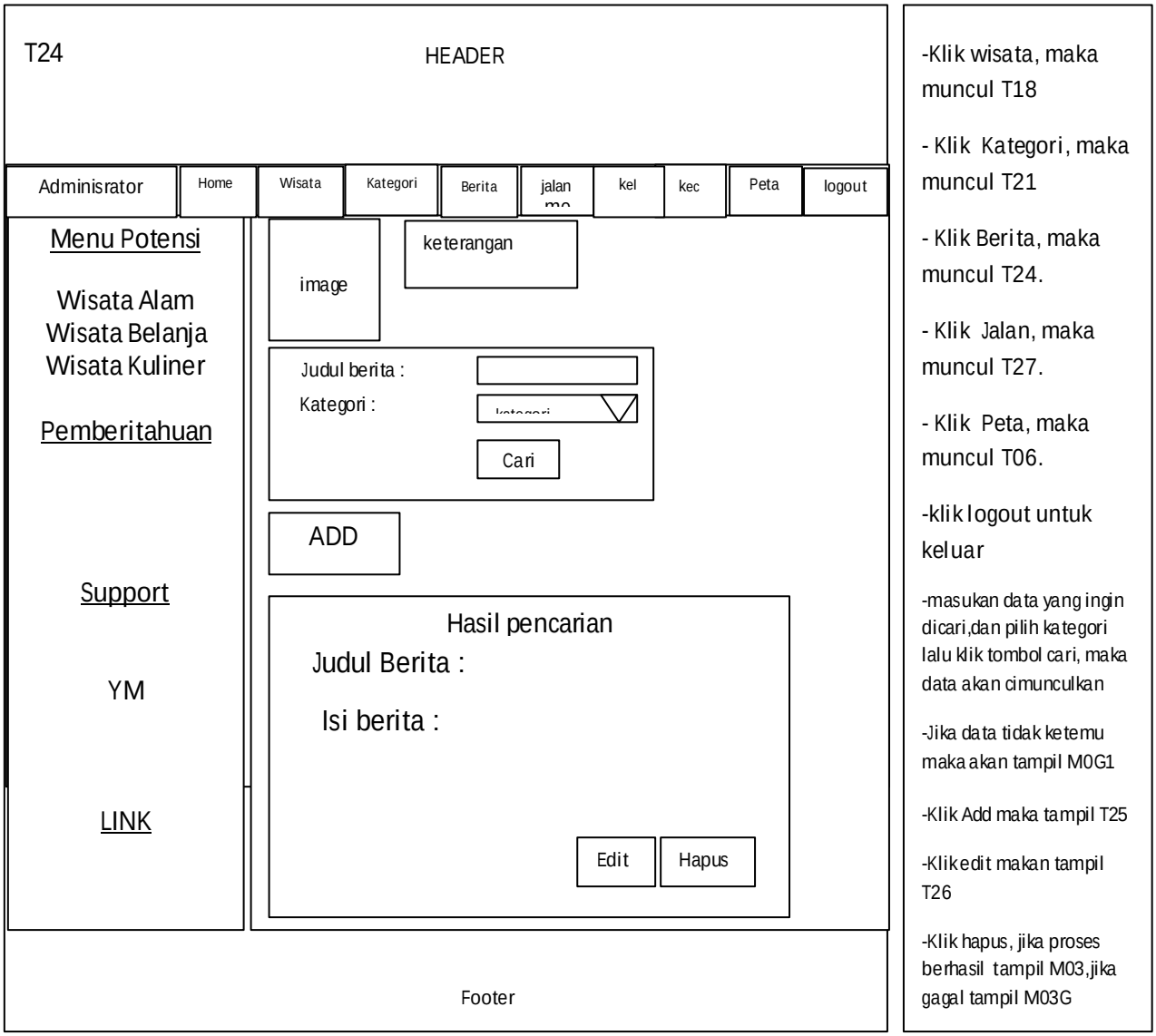
-jika penyimpanan data berhasil maka akan muncul M02

-jika penyimpanan tidak berhasil maka akan tampil M02G

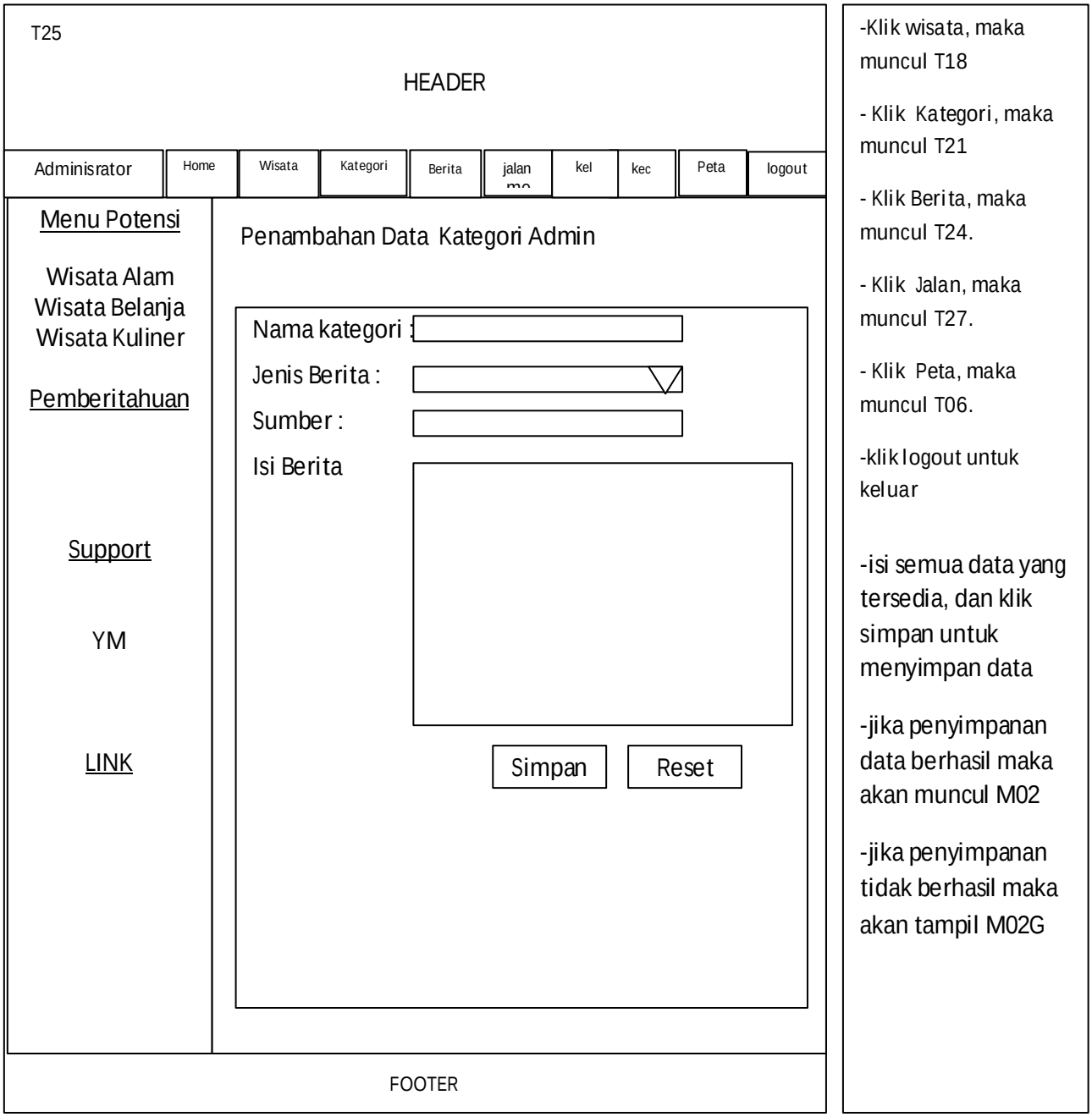
Gambar 3.39 Tampilan Tambah Data Kategori Admin



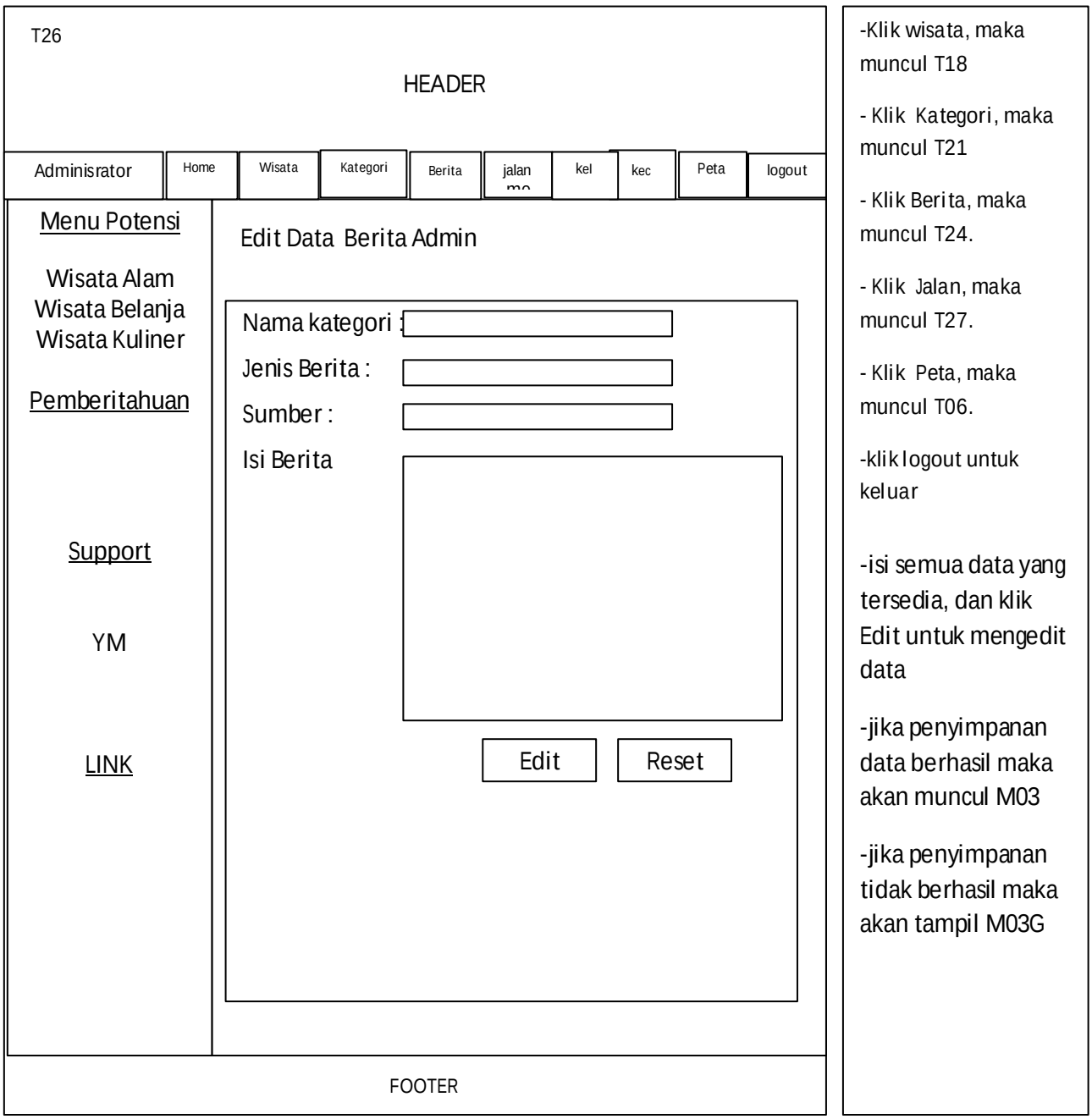
Gambar 3.40 Tampilan Edit Data Kategori Admin



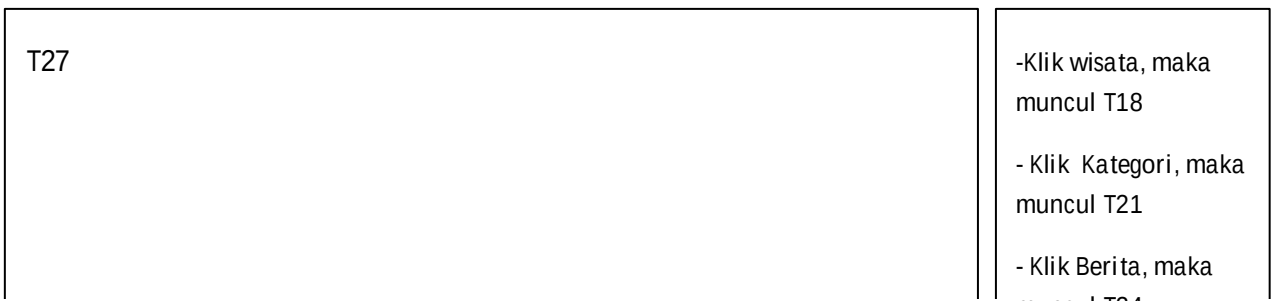
Gambar 3.41 Tampilan Data Berita Admin

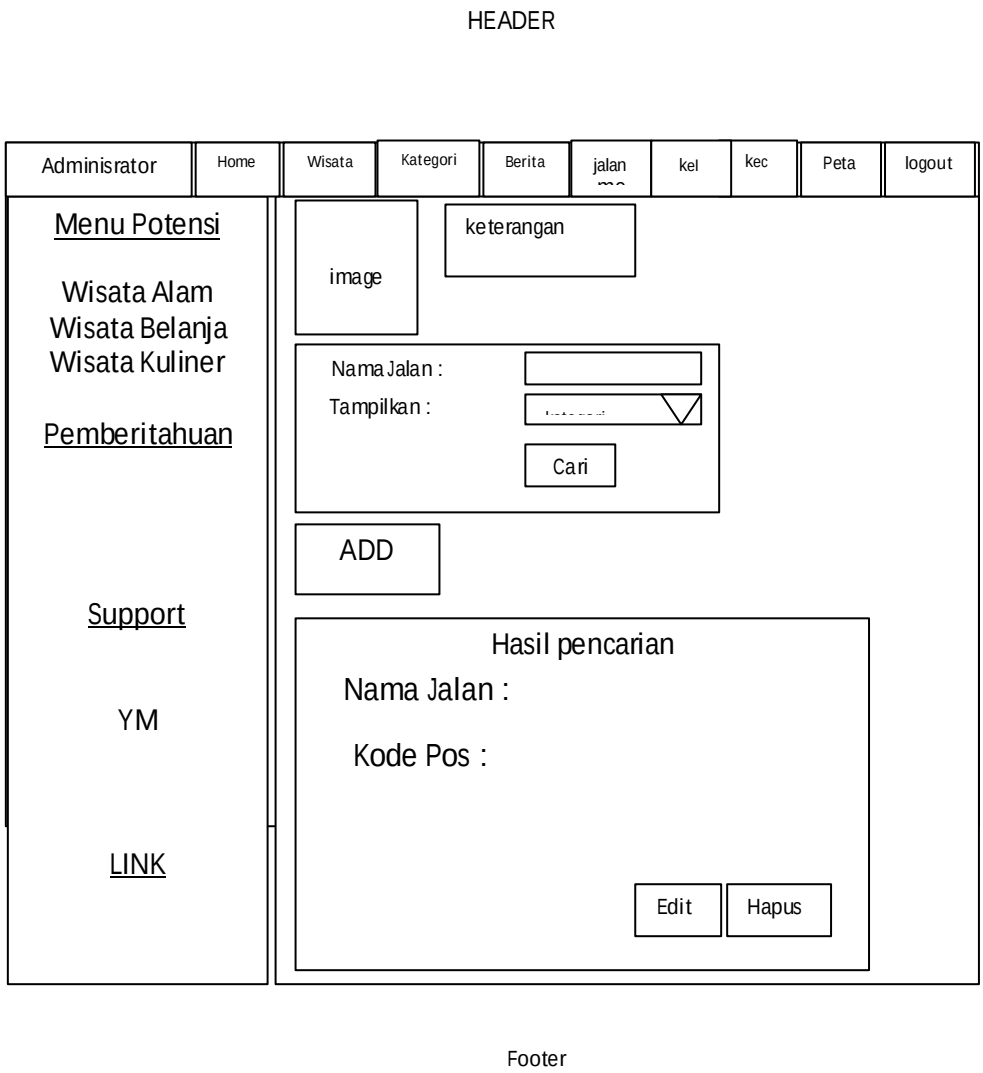


Gambar 3.42 Tampilan Tambah Data Berita Admin

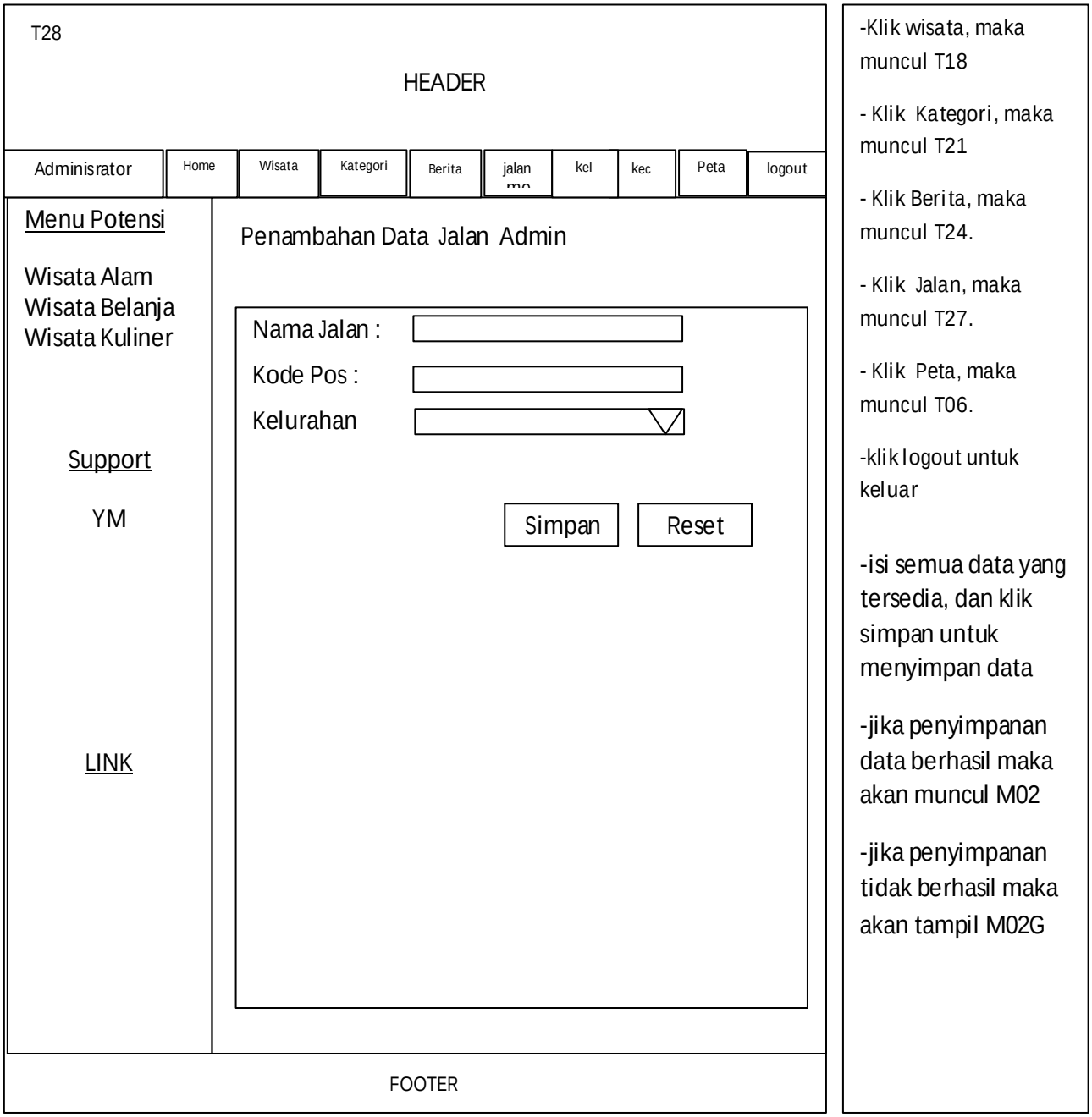


Gambar 3.43 Tampilan Edit Data Berita Admin

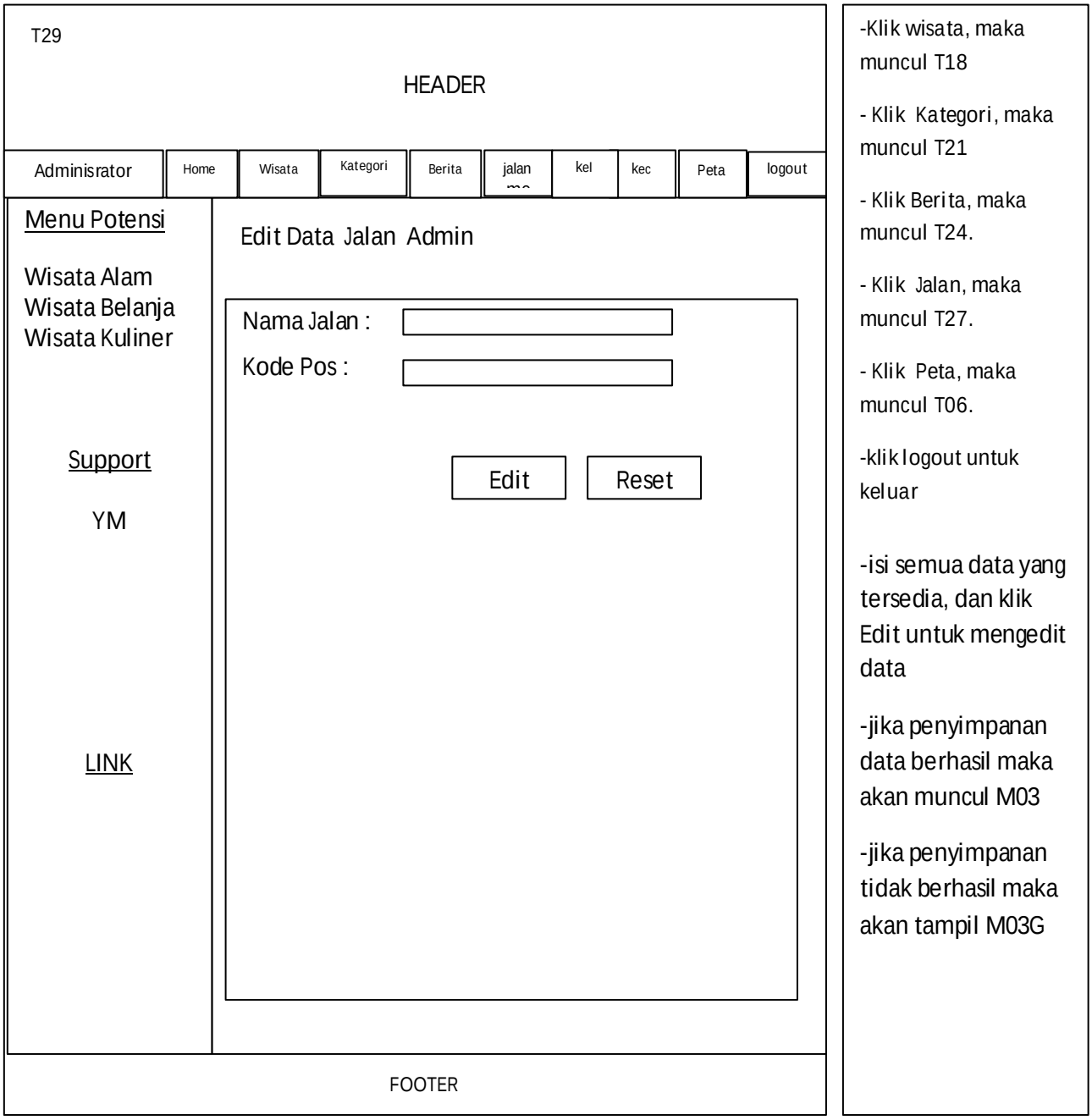




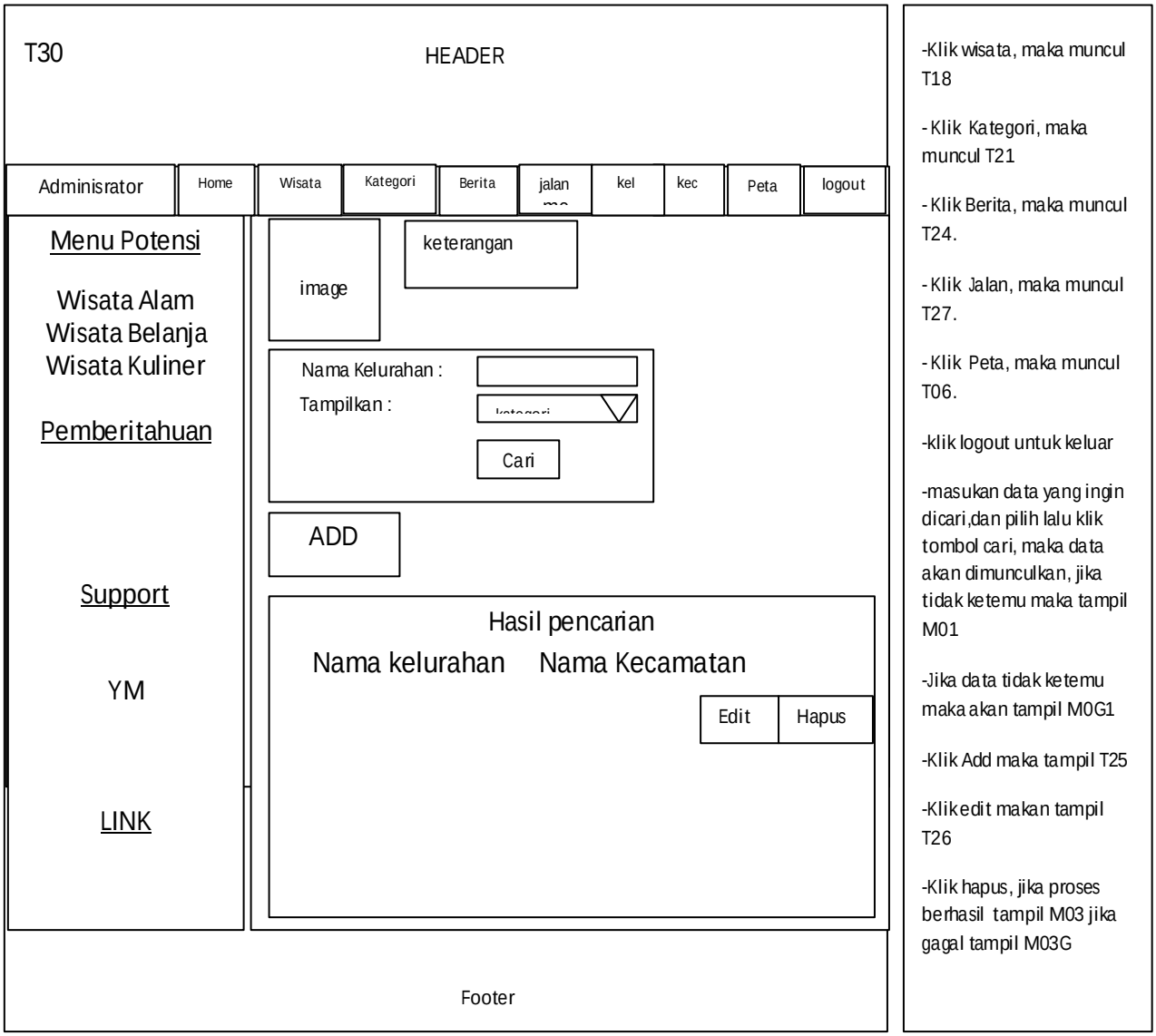
Gambar 3.44 Tampilan Data Berita Admin



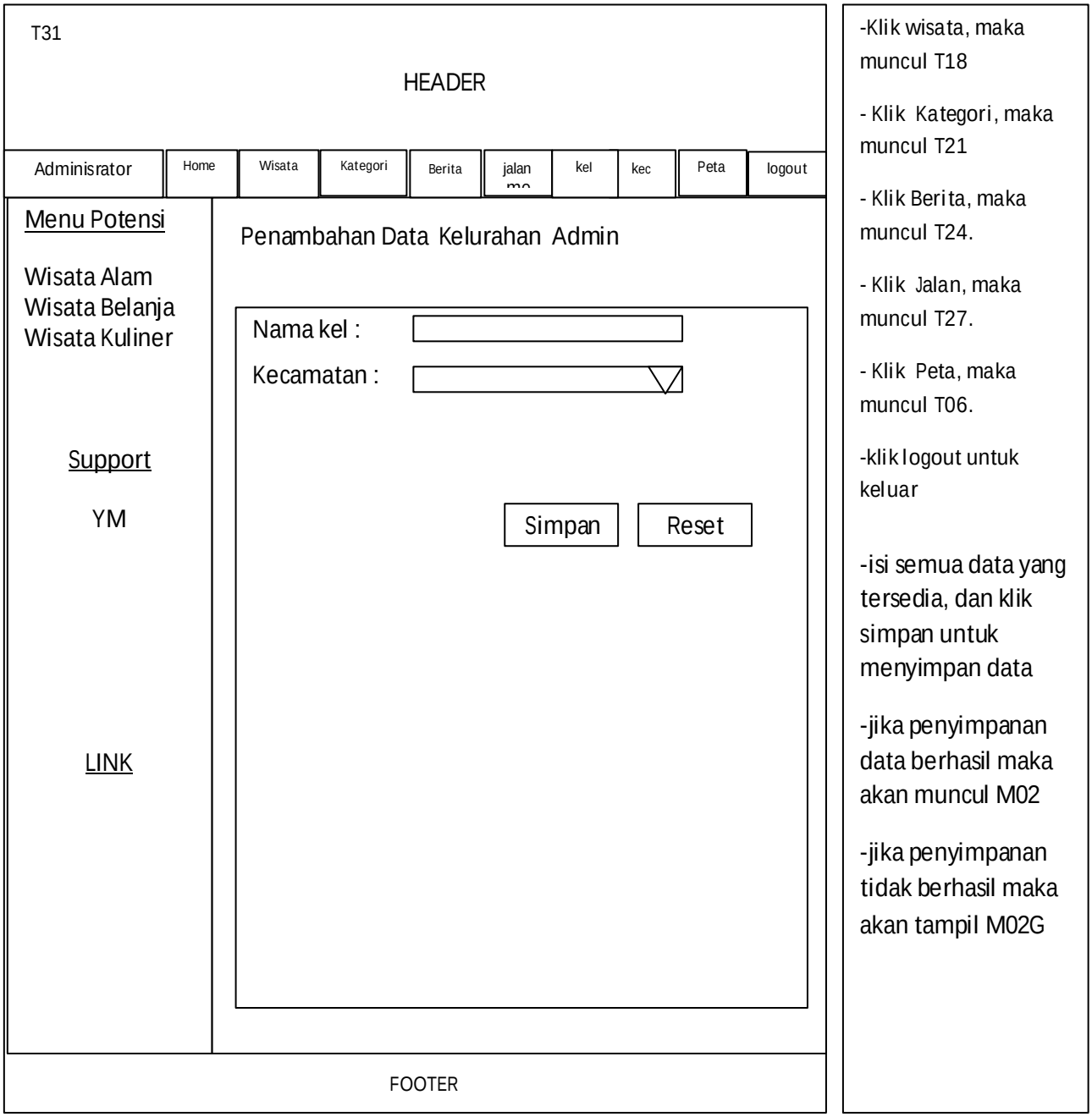
Gambar 3.45 Tampilan Tambah Data Jalan Admin



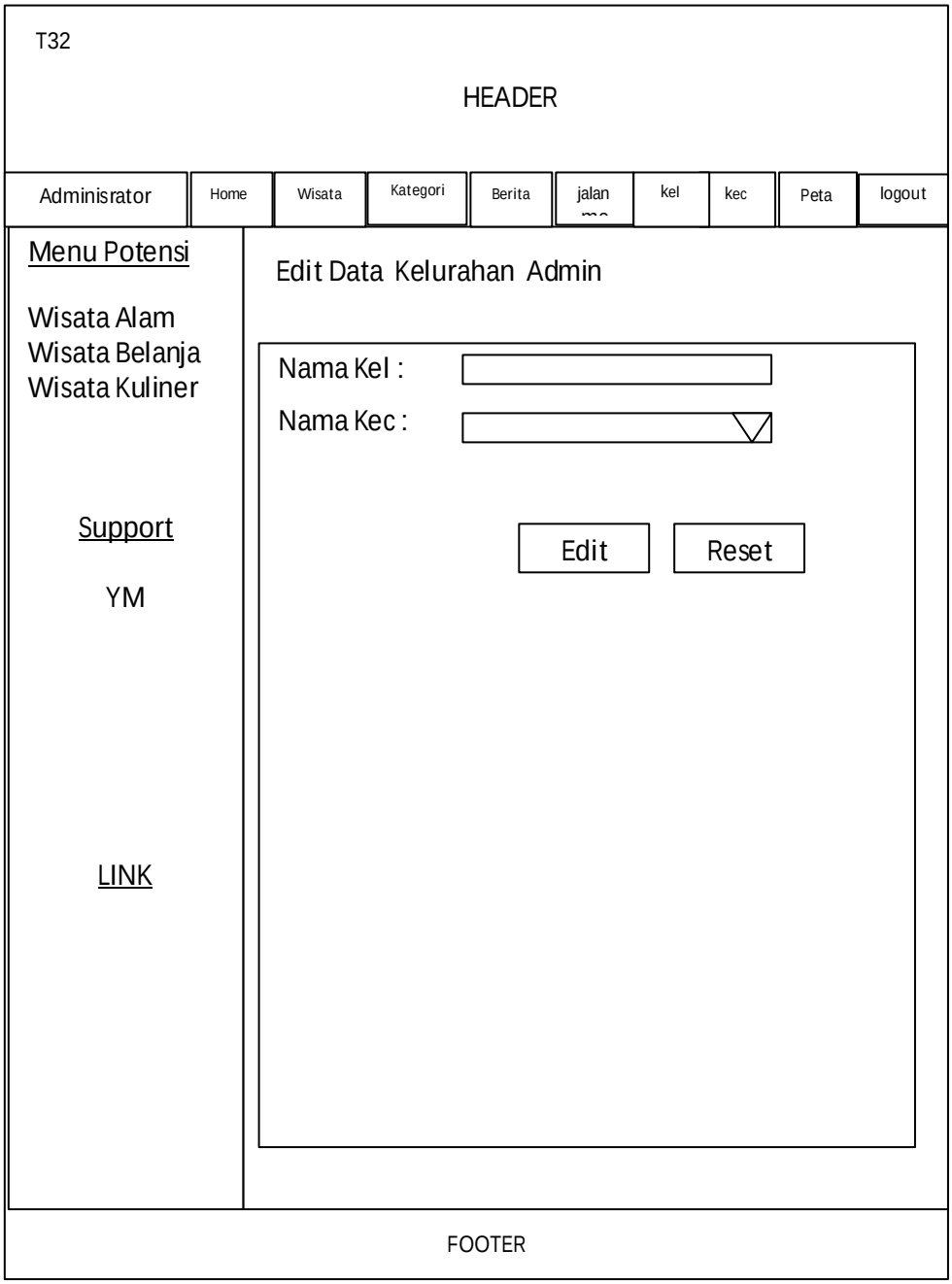
Gambar 3.46 Tampilan Edit Data Jalan Admin



Gambar 3.47 Tampilan Data Kelurahan Admin



Gambar 3.48 Tampilan Tambah Data Kelurahan Admin



-Klik wisata, maka muncul T18

- Klik Kategori, maka muncul T21

- Klik Berita, maka muncul T24.

- Klik Jalan, maka muncul T27.

- Klik Peta, maka muncul T06.

-klik logout untuk keluar

-isi semua data yang tersedia, dan klik Edit untuk mengedit data

-jika penyimpanan data berhasil maka akan muncul M04

-jika penyimpanan tidak berhasil maka akan tampil M04G

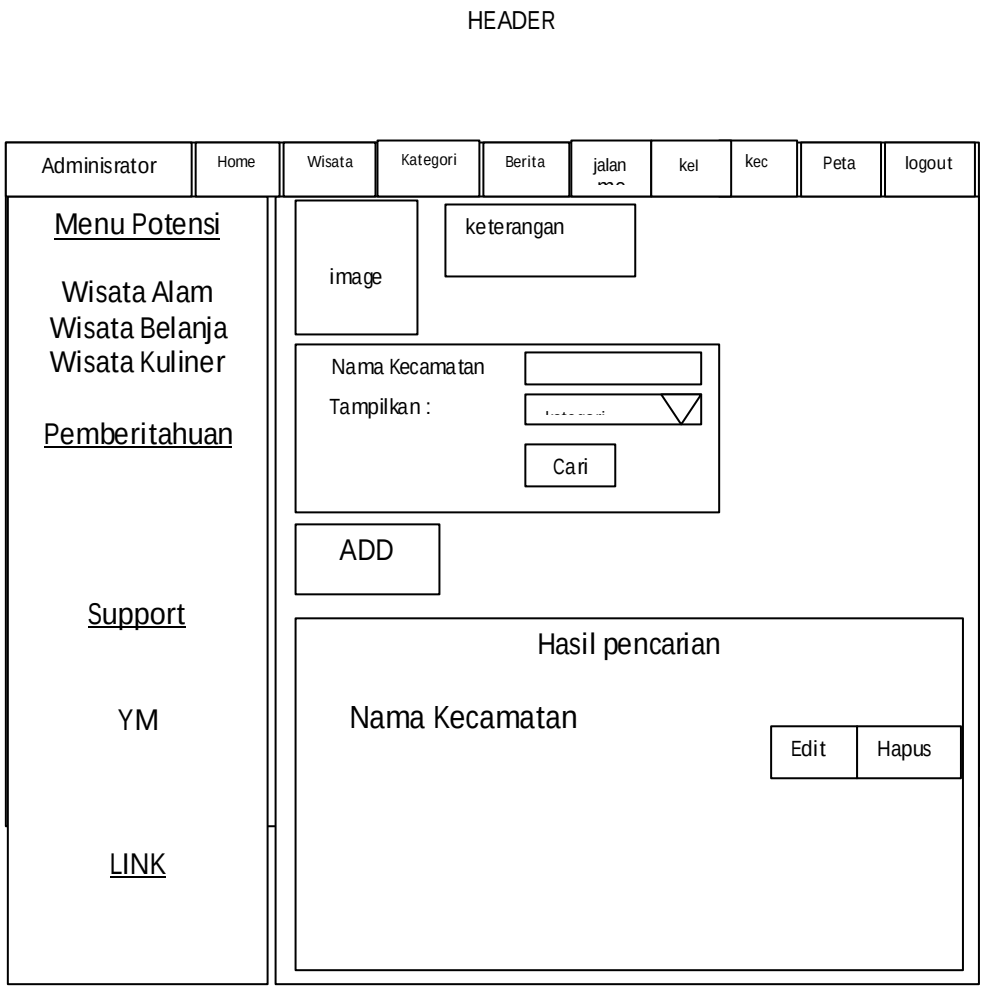
Gambar 3.49 Tampilan Edit Data Kelurahan Admin



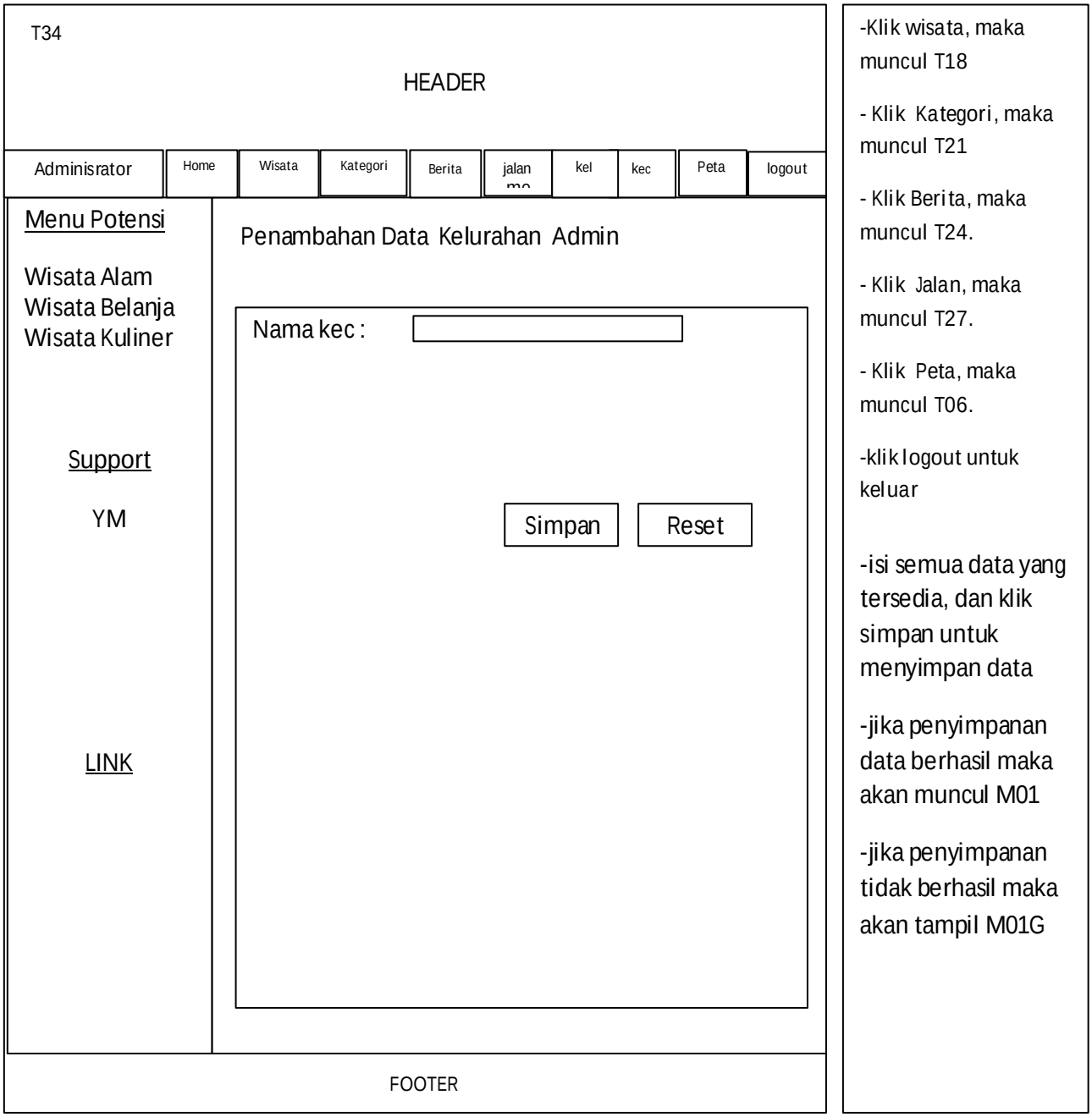
-Klik wisata, maka muncul T18

- Klik Kategori, maka muncul T21

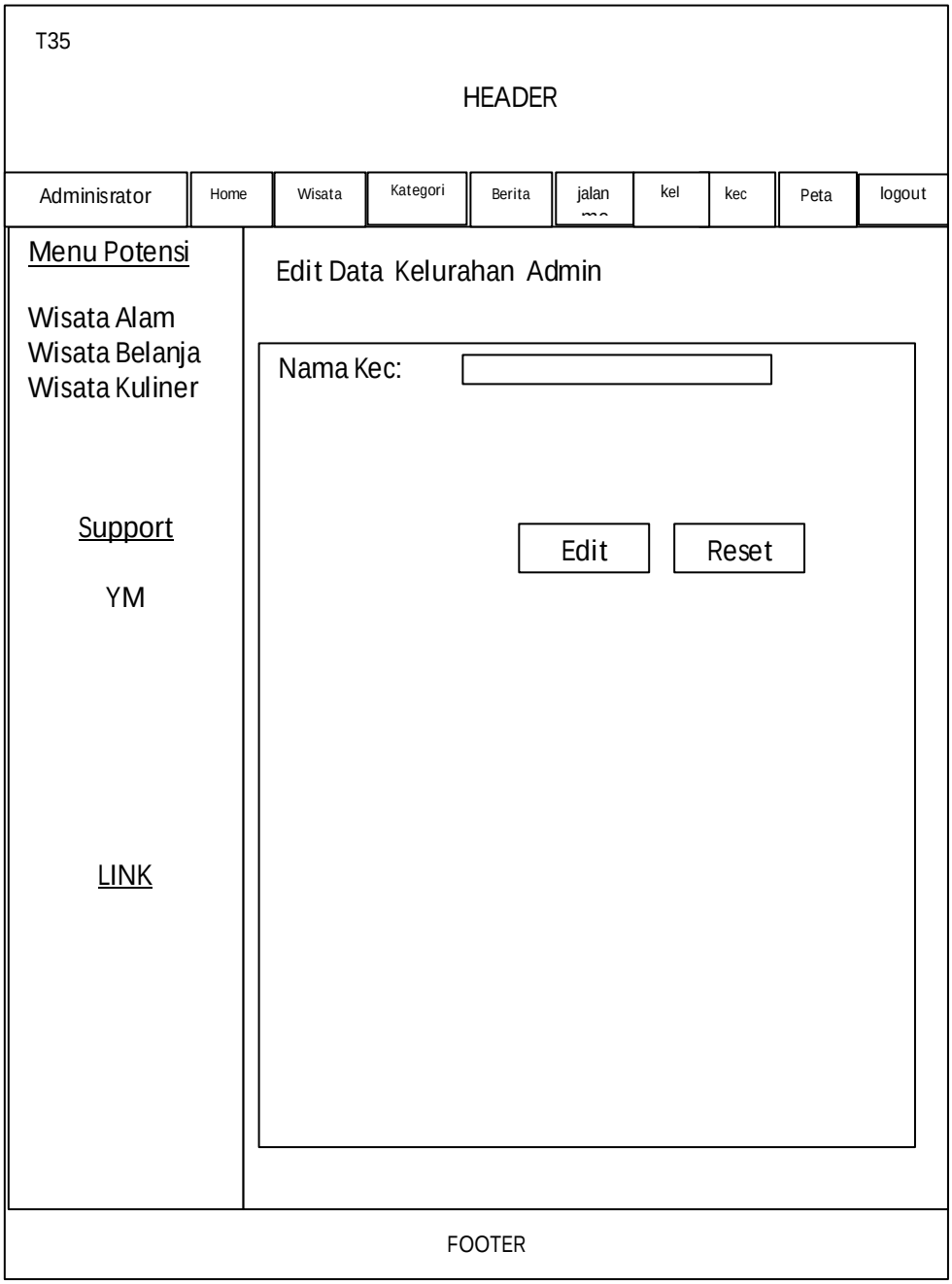
- Klik Berita, maka



Gambar 3.50 Tampilan Data Kecamatan Admin



Gambar 3.51 Tampilan Tambah Data Kecamatan Admin



-Klik wisata, maka muncul T18

- Klik Kategori, maka muncul T21

- Klik Berita, maka muncul T24.

- Klik Jalan, maka muncul T27.

- Klik Peta, maka muncul T06.

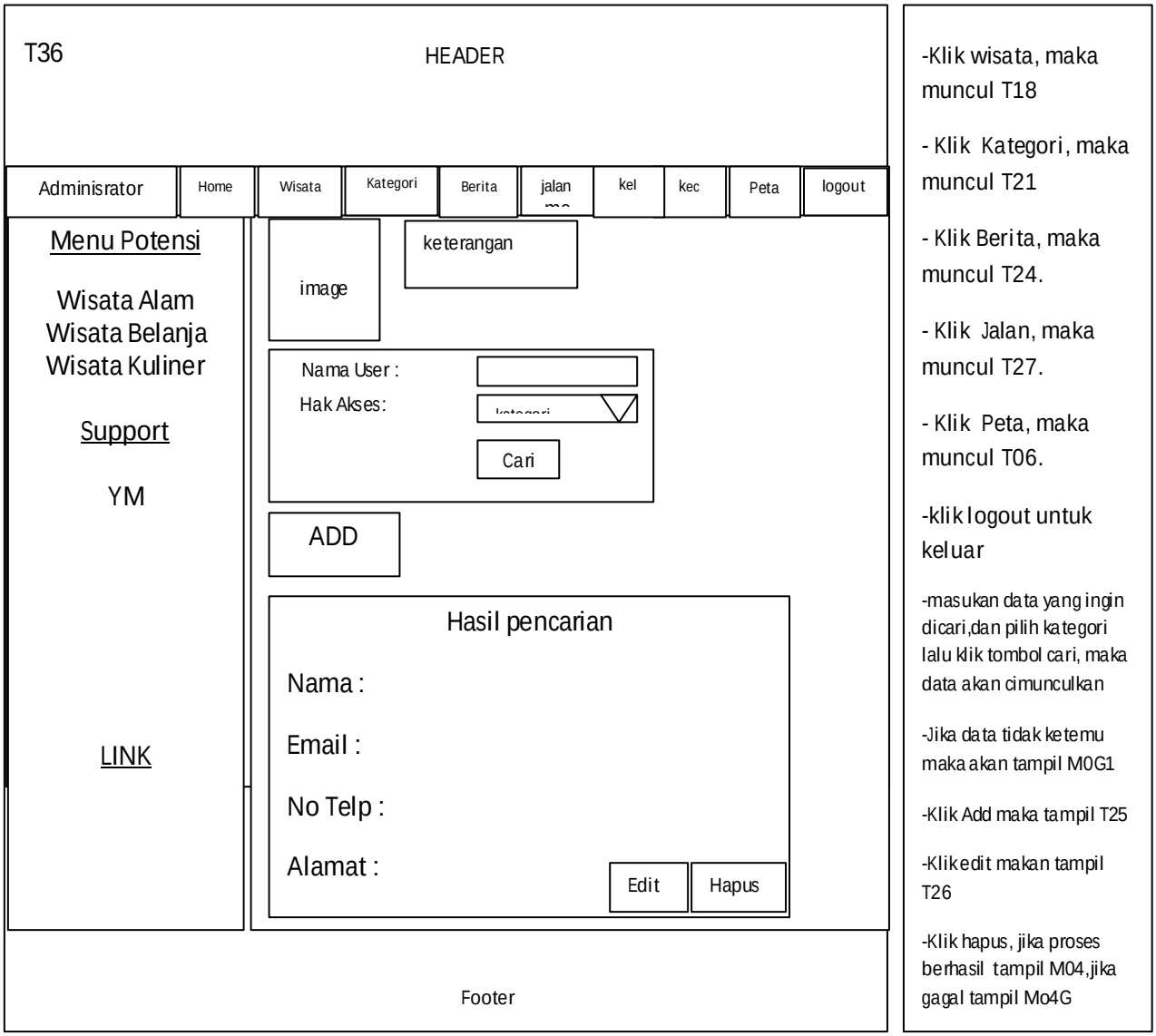
-klik logout untuk keluar

-isi semua data yang tersedia, dan klik Edit untuk mengedit data

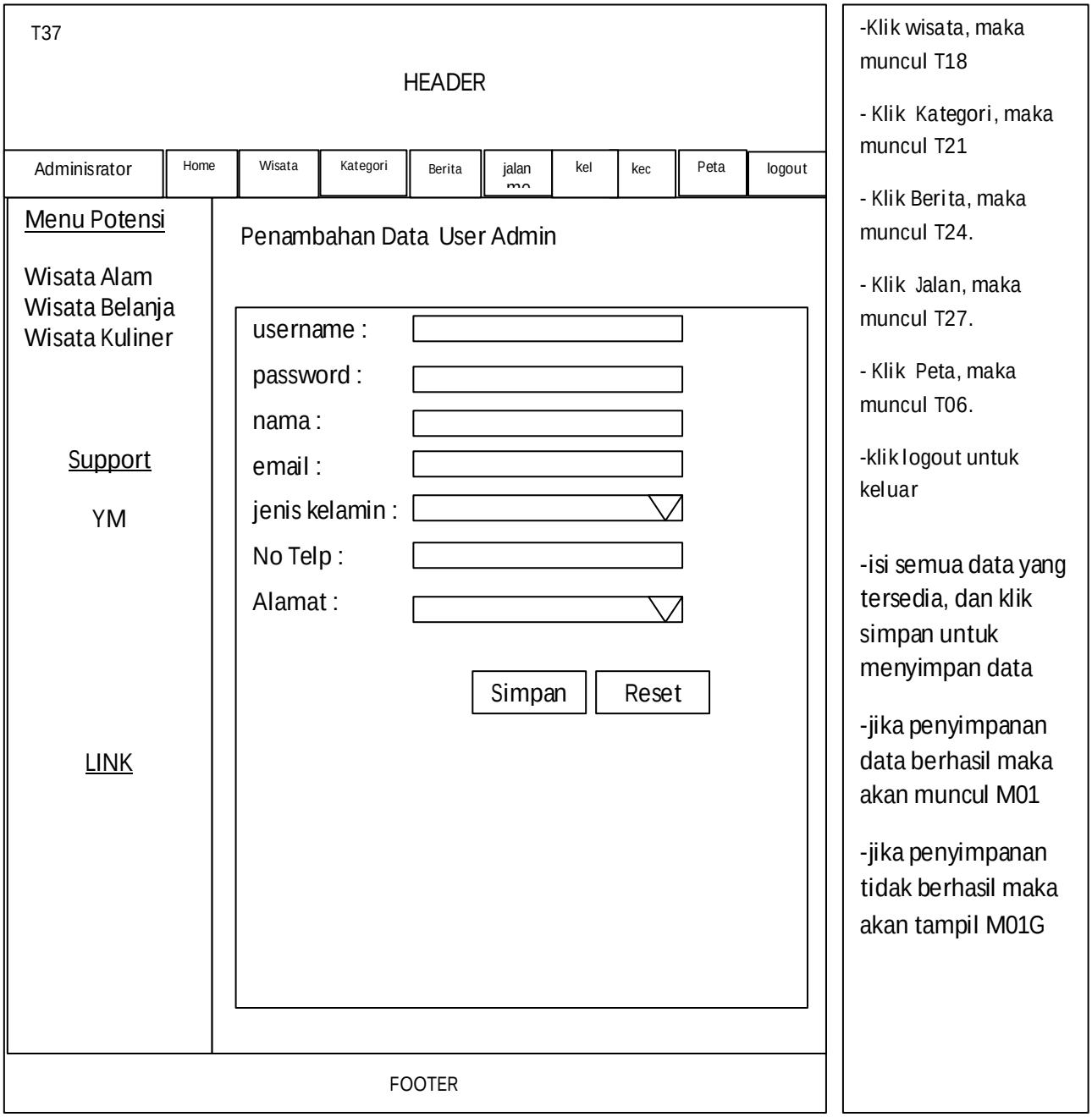
-jika penyimpanan data berhasil maka akan muncul M04

-jika penyimpanan tidak berhasil maka akan tampil M04G

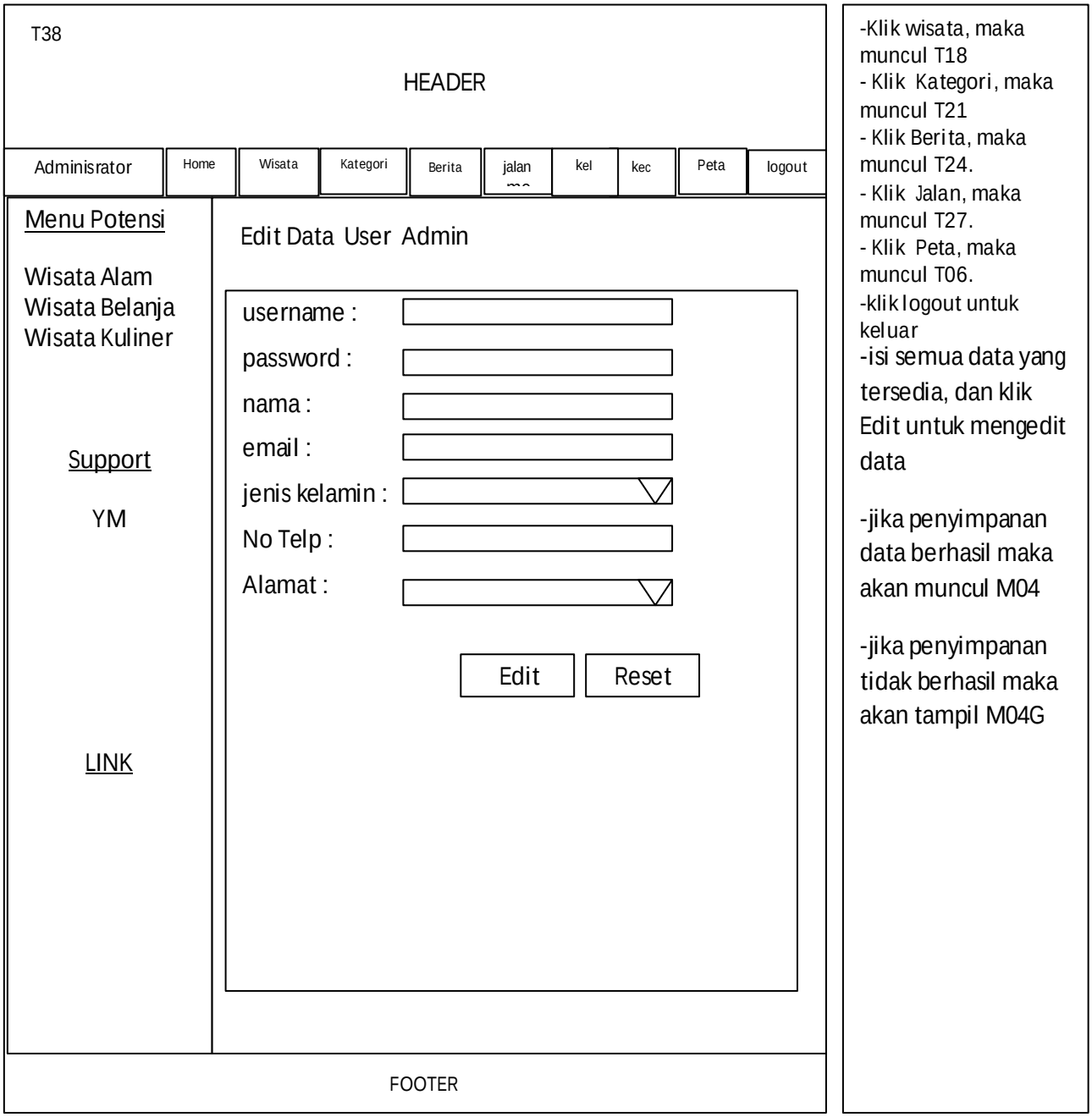
Gambar 3.52 Tampilan Edit Data Kecamatan Admin



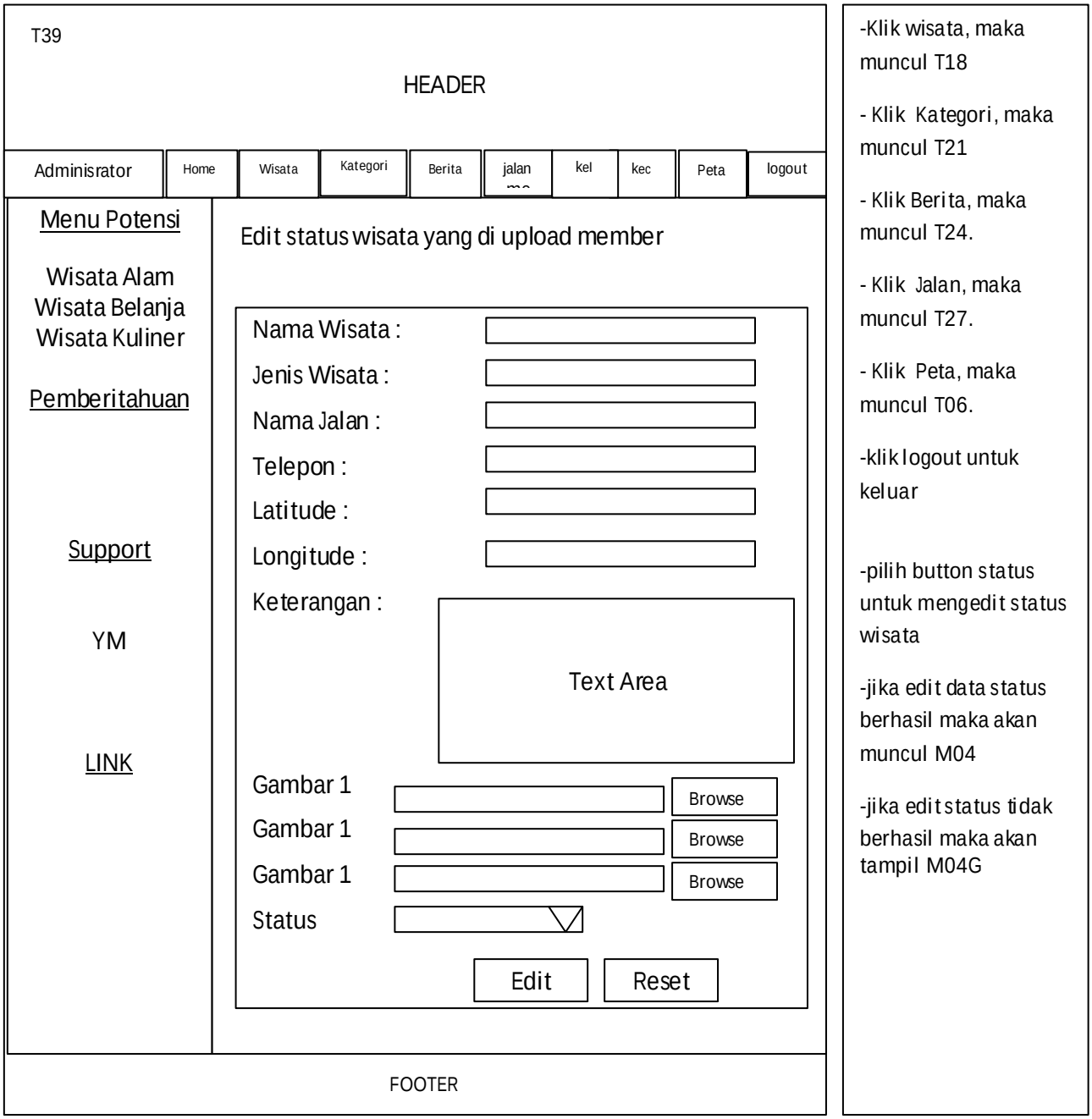
Gambar 3.53 Tampilan Data User Admin



Gambar 3.54 Tampilan Tambah Data User Admin



Gambar 3.55 Tampilan Edit Data User Admin



Gambar 3.56 Tampilan Edit Data Pending Admin

3.4.4.2 Perancangan Pesan

Dibawah ini adalah perancangan pesan yang muncul sebagai informasi kepada pengguna dalam mengoperasikan sistem.

M01 Data tidak ditemukan!	M02 Data berhasil disimpan!
M03 Data telah dihapus!	M02G Data gagal disimpan
M04 Data Berhasil di edit	M03G Data gagal dihapus !!
M05 Pendaftaran Berhasil	M04G Data gagal diedit!!
M06 Login Berhasil	M05G Pendaftaran gagal!
	M06G Login Gagal pesan

Gambar 3.57 Tampilan Pesan

3.4.4.3 Jaringan Semantik

Jaringan semantik menggambarkan keterhubungan dari navigasi menu dari satu *form* ke *form* yang lain. Keterhubungan dari navigasi pada setiap menu dari sistem informasi geografis ini adalah sebagai berikut.

1. Jaringan Semantik Pengunjung

T01 : Halaman Index

T02 : Halaman Profile

T03 : Halaman Wisata

T04 : Halaman Berita

T05 : Halaman Daftar

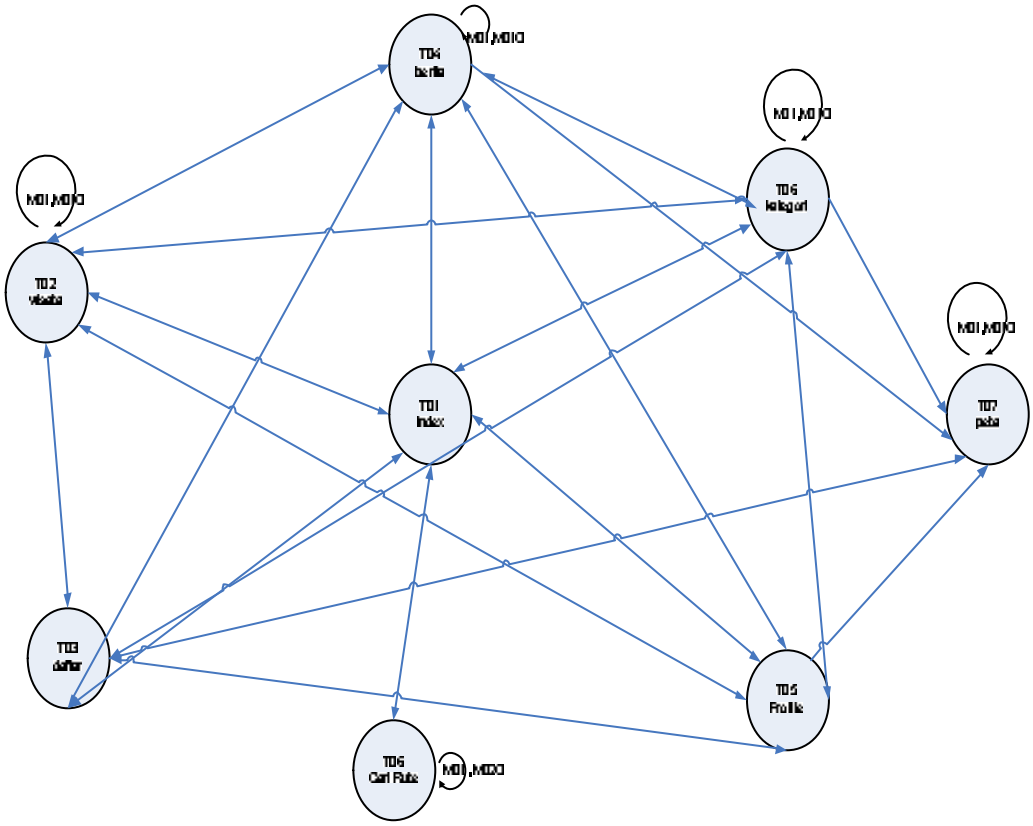
T06 : Halaman Detail Peta

T07 : Halaman Peta

M01 : Data Tidak Ditemukan !

M05 : Pendaftaran Berhasil!

M05G : Pendaftaran Gagal!



Gambar 3.58 Jaringan Semantik Pengunjung

2. Jaringan Semantik Member

T08 : Tampil login

T09 : Halaman halaman index member

T010 : Halaman wisata

T011 : Halaman tambah data wisata

T012 : Halaman kategori member

T13 : Halaman berita member

T14 : Halaman jalan member

T15 : Halaman my wisata member

T16 : Halaman edit wisata member

M01 : Data Tidak Ditemukan

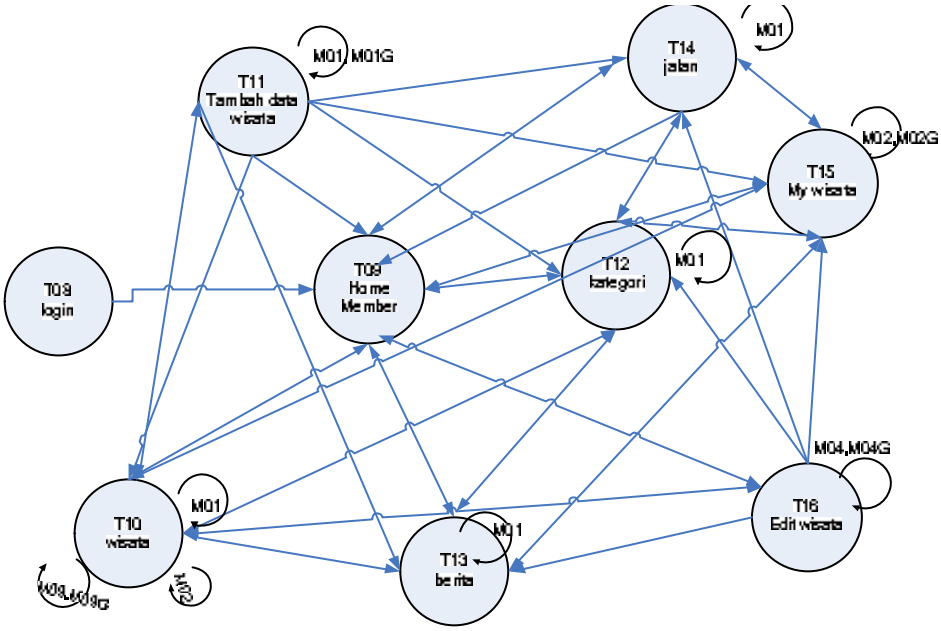
M02 : Data Berhasil Disimpan

M03 : Data Telah Dihapus

M04 : Data Berhasil diedit

M06 : Login Berhasil

M07 : Login Gagal



Gambar 3.59 Jaringan Semantik Admin

3. Jaringan Semantik Admin

T17 : Tampil login Admin

T18 : Halaman halaman home admin

T019 : Halaman wisata admin

T020 : Halaman tambah data wisata admin

T021 : Halaman edit data wisata admin

T22 : Halaman kategori admin

T23 : Halaman tambah data kategori admin

T24 : Halaman edit data kategori admin

T25 : Halaman berita admin

T26 : Halaman tambah berita admin

T27 : Halaman edit berita admin

T28 : Halaman jalan admin

T29 : Halaman tambah jalan admin

T30 : Halaman edit data jalan admin

T31 : Halaman kelurahan admin

T32 : Halaman tambah kelurahan admin

T33 : Halaman edit data kelurahan admin

T34 : Halaman kecamatan admin

T35 : Halaman tambah kecamatan admin

T36 : Halaman edit data kecamatan admin

T37 : Halaman data user admin

T38 : Halaman tambah data user admin

T39 : Halaman edit data user admin

T40 : Halaman data pending admin

M01 : Data Tidak Ditemukan

M02 : Data Berhasil Disimpan

M03 : Data Telah Dihapus

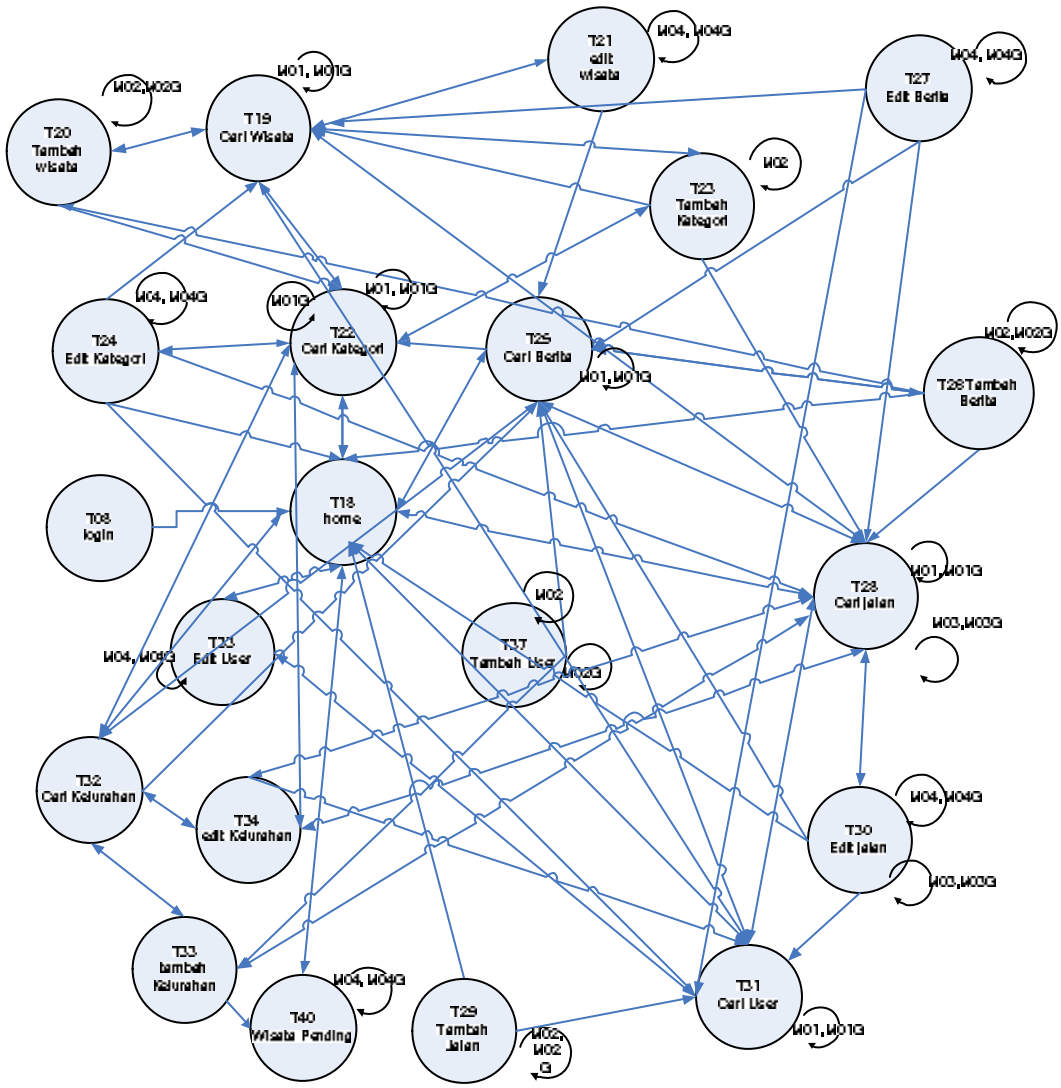
M03G : Data Gagal Dihapus

M04 : Data Berhasil diedit

M04 : Data Berhasil diedit

M06 : Login Berhasil

M06G : Login Gagal

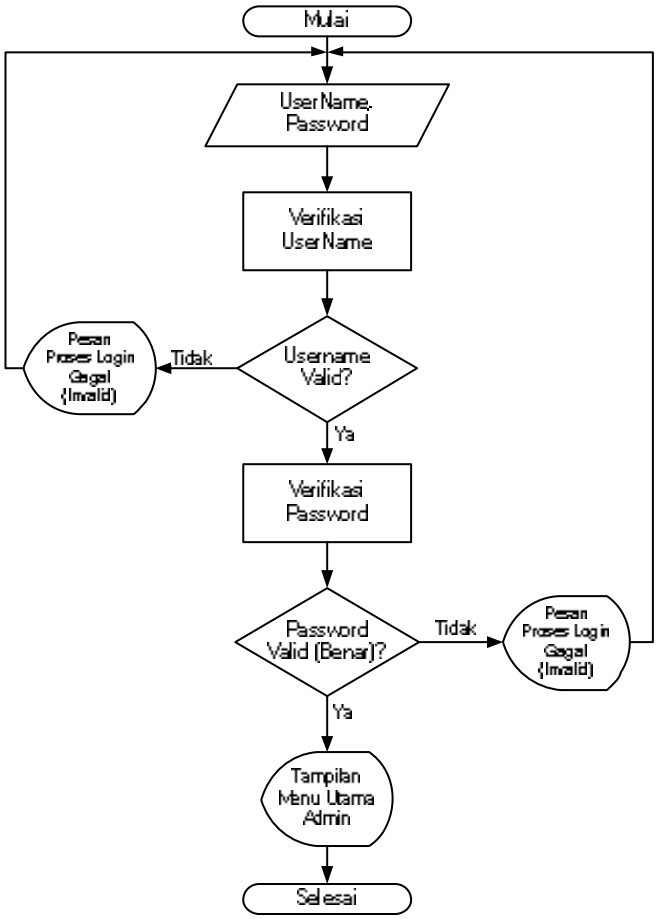


Gambar 3.60 Jaringan Semantik Admin

3.4.4.4 Perancangan Prosedural

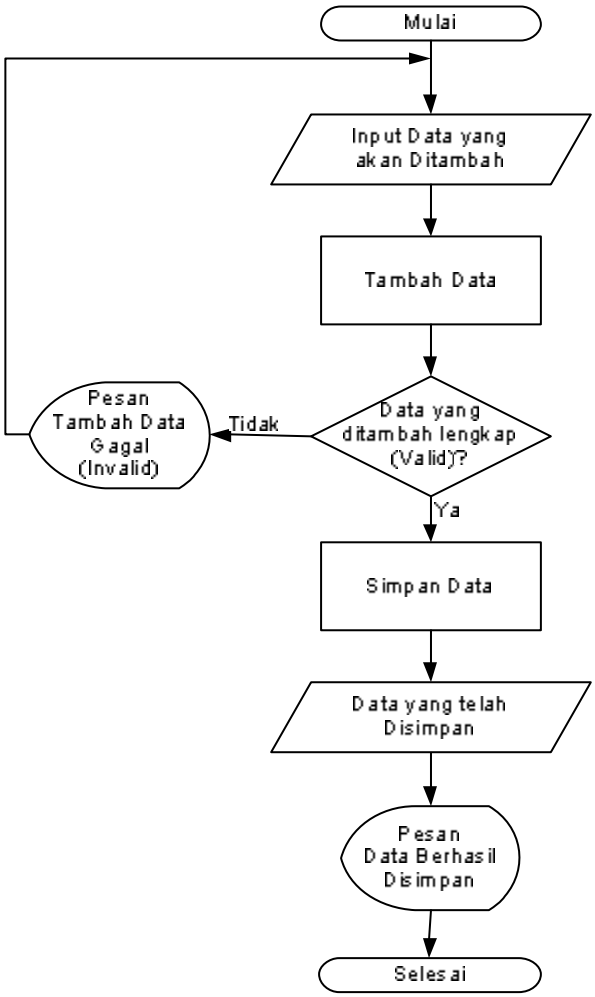
Kinerja dari SIG Potensi Wisata di Kota Tasikamalaya dijelaskan melalui diagram alir *(flowchart)* dari keseluruhan sistem yang bekerja didalamnya. Perancangan procedural tersebut dapat dilihat pada gambar berikut

3.4.4.4.1 Prosedural Login



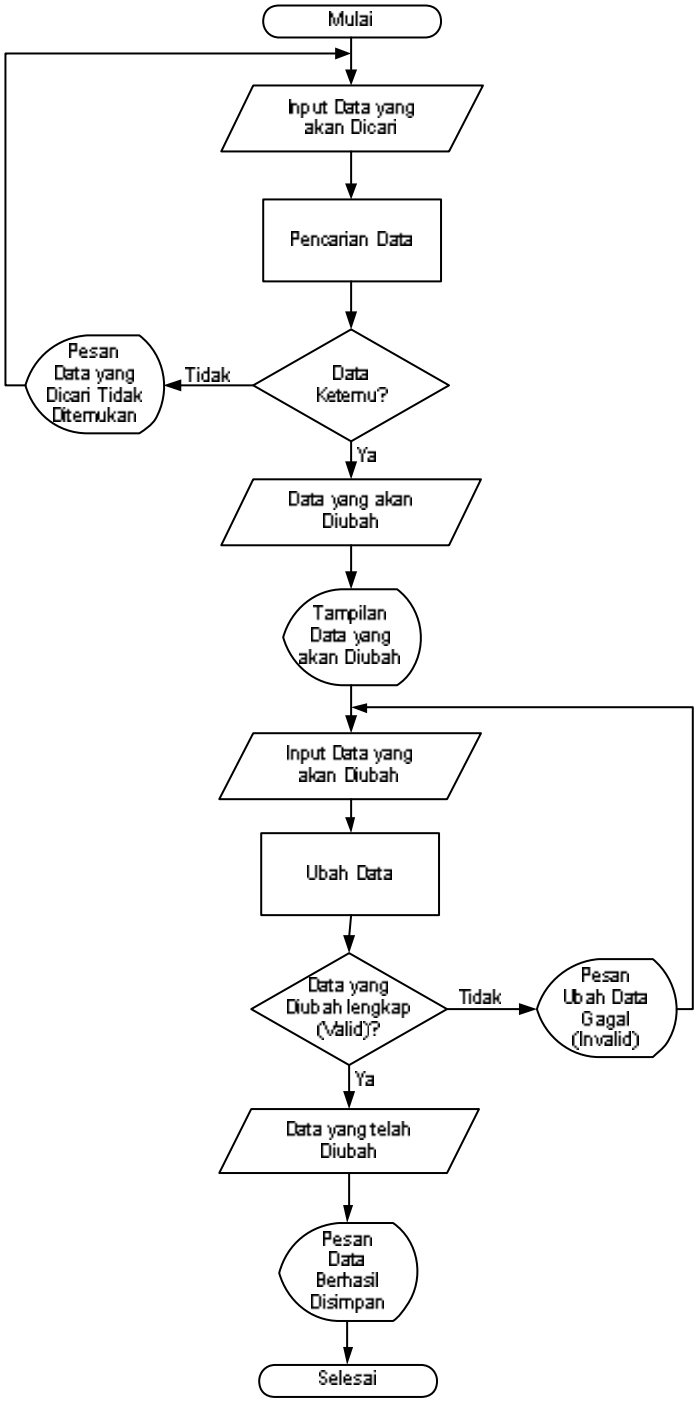
Gambar 3.61 Prosedural Login

3.4.4.4.2 Prosedural Tambah Data



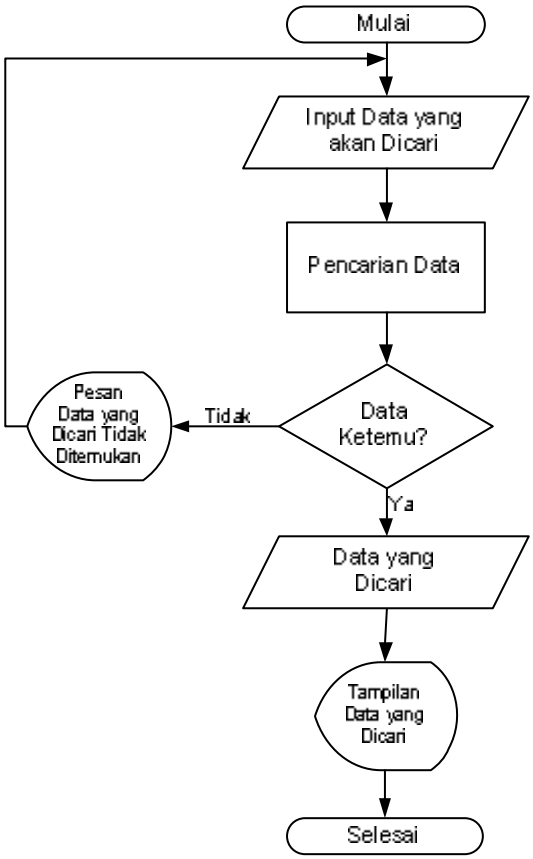
Gambar 3.62 Prosedural Tambah Data

3.4.4.4.3 Prosedural Ubah Data



Gambar 3.63 Prosedural Ubah Data

3.4.4.4.4 Prosedural Cari Data



Gambar 3.64 Prosedural Cari Data