

## **BAB VI**

### **KESIMPULAN DAN SARAN**

#### **VI.1 Kesimpulan**

Berdasarkan pembahasan pada bab-bab sebelumnya, maka dapat ditarik kesimpulan, yaitu: Aplikasi Sistem Pakar Alergi dan Penanganannya berhasil dibangun dengan baik, yang menghasilkan diagnosis jenis alergi berdasarkan gejala-gejala yang dialami serta memberi informasi penanganan alergi tersebut.

#### **VI.2 Saran**

Beberapa saran yang diberikan untuk pengembangan Aplikasi Sistem Pakar Alergi dan Penanganannya lebih lanjut antara lain menambahkan jenis-jenis alergi lain yang dapat didiagnosis, agar sistem pakar ini menjadi lebih sempurna. Sistem pakar ini dapat dikembangkan menjadi aplikasi berbasis web agar semua orang dapat mengakses kapan saja dan dimana saja.

## DAFTAR PUSTAKA

- Atika, Linda, Sri Hartati, 2006, *Aplikasi sitem Pakar Sebagai Alat Bantu Pendiagnosa Penyakit Stoke*, Jurnal, Program Studi Ilmu Komputer, Fakultas Matematika dan Pengetahuan Alam, Unversitas Gadjah Mada, Yogyakarta.
- Dhany, Safia, 2009, *Perancangan Sistem Pakar Untuk Diagnosa Penyakit Anak*, Skripsi, Program Studi Ilmu Komputer, Fakultas Matematika Dan Ilmu Pengetahuan Alam, Unversitas Sumatera Utara, Medan.
- Handayani, Lina, Telo Sutikno, 2008, *Sistem Pakar Untuk Diagnosis Penyakit THT Berbasis Web Dengan "e2glite Expert System Shell"*, Jurnal, Program Studi Teknik Elektro, Universitas Ahmad Dahlan, Yogyakarta.
- Handojo, Andreas, M. Isa Irawan, Fendhy Ongko, 2004, *Perancangan Dan Pembuatan Aplikasi Sistem Pakar Untuk Permasalahan Tindakan Pidana Terhadap Harta Kekayaan*, Jurusan Teknik Informatika, Universitas Kristen Petra, Surabaya.
- Hartati, Sri, 2005, *Media Konsultasi Penyakit Kelamin Pria Dengan Penanganan Ketidakpastian Menggunakan Certainty Factor Bayesian*, Seminar Nasional Aplikasi Teknologi Informasi 2005 (SNATI 2005), Yogyakarta.
- Heckerman, David, Edward H. Shortliffe, 1992, *From Certainty Factors to Belief Network*, Departments of Computer Science and Pathology, University of Southern California, California.

- Jaenudin, 2006, *Belajar Sendiri .NET dengan Visual C# 2005*, Andi Offset, Yogyakarta.
- Kusrini, 2005, *Penggunaan Certainty Factor dalam Sistem Pakar untuk Melakukan Diagnosis dan Memberikan Terapi Penyakit Epilepsi dan Keluarganya*, Seminar Nasional: "Soft Computing, Intelligent Systems and Information Teknologi" (SIIT 2005), Surabaya.
- Kusrini, 2007, *Question Quantification to Obtain User Certainty Factor in Expert System Application for Disease Diagnosis*, International Conference on Electrical Engineering and Informatics, Bandung.
- Mateo, Romeo Mark A., jaewan Lee, Bobby D. Gerardo, 2008, *Healthcare Expert System Based on Group Cooperation Model*, Journal, School of Electronic and Information Engineering, Kusnan National Unoversity, South Korea.
- Nafisah, Syifaun, Nazrul Effendy, 2007, *Implementasi Sistem Pakar Dalam Bidang Farmakologi dan Terapi Sebagai Pendukung Keputusan Berbasis Web*, Teknik Informatika, Universitas Pembangunan Nasional "Veteran", Yogyakarta.
- Noviyanto, Andri, 2008, *Sistem Pakar Untuk Analisa Pada Pasar Mata Uang Asing*, IT Telkom, Bandung.
- Pradana, Chandara Putra, Sri Kusumadewi, 2007, *Aplikasi Diagnosis Penyakit Hepatitis Untuk Mobile Devices Menggunakan J2ME*, Jurnal, Jurusan Teknik Informatika, Fakultas Teknologi Industri, Unversitas Islam Indonesai, Yogyakarta.
- Pranoto, M. Suryo, 2009, *C# Part 1-Pengenalan Logika Basic*, IlmuKomputer.Com

- Rohman, Feri Fahrur, Ami Fauzijah, 2008, *Rancang Bangun Aplikasi Sistem Pakar Untuk Menentukan Jenis Gangguan Perkembangan Pada Anak*, Jurnal, Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia, Yogyakarta,
- Roitt, Ivan M., 2003, *Imunologi*, Edisi 8, Widya Medika, Jakarta.
- Subakti, Ifran, Rahmat Hidayatullah, 2007, *Aplikasi Sistem Pakar Untuk Diagnosis Awal Gangguan Kesehatan Secara Mandiri Menggunakan Variable-Centered Intelligent Rule Sistem*, Jurnal, Jurusan Teknik Informatika, Institut Teknologi Sepuluh November, Surabaya.
- Suddeth, Jeffery, 2006, *Programing C# With Visual Studio .NET 2005*, Lulu Press, California.
- Sulistiyohati, Aprilia, Taufiq Hidayat, 2008, *Aplikasi Sistem Pakar Diagnosa Penyakit Ginjal Dengan Metode Dempster-Shafer*, Seminar Nasional Aplikasi Teknologi Informasi 2008 (SNATI 2008), Yogyakarta.
- Tjay, Tan Hoan, Kirana Rahardja, 2007, *Obat-Obat Penting Kasiat, Penggunaan Dan Efek-Efek Sampingnya*, Edisi Ke Enam, PT Elex Media Komputindo, Jakarta.
- Tjokronegoro, Arjatmo, 1976, *Dasar-Dasar Alergi*, Cermin Dunia Kedokteran No. 6, Jakarta.
- Turban, Efraim, 1992, *Expert Systems and Applied Artificial Intelligence*, Macmillan Publishing Company, New York.
- Wardana, I Nyoman Kusuma, Alfa Antariksa, Nazrul Effendy, 2008, *Perancangan Sistem Pakar Untuk Diagnosa*

Penyakit Mulut dan Gigi Menggunakan Bahasa Pemrograman CLIPS, Seminar Nasional Aplikasi Teknologi Informasi 2008 (SNATI 2008), Yogyakarta.

Wijaya, Rahmadi, 2007, *Penggunaan Sistem Pakar Dalam Pengembangan Portal Informasi Untuk Spesifikasi Jenis Penyakit Infeksi*, Jurnal, Sekolah Tinggi Manajemen Informatika Dan Computer CIC, Cirebon.





# LAMP IRAN

# SKPL

## SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK

### APLIKASI SISTEM PAKAR DIAGNOSA JENIS ALERGI DAN PENANGANANNYA PADA PENDERITA ALERGI (SiPaGi)

Untuk:

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Fani Damayanti

06 07 04912

Program Studi Teknik Informatika - Fakultas Teknologi  
Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika	Nomor Dokumen		Halaman
		SKPL-SiPaGi		1/32
	Fakultas Teknologi Industri	Revisi		

## DAFTAR PERUBAHAN

Revisi	Deskripsi
<b>A</b>	
<b>B</b>	
<b>C</b>	
<b>D</b>	
<b>E</b>	
<b>F</b>	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								

## Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi



## Daftar Isi

1	Pendahuluan .....	6
1.1	Tujuan.....	6
1.2	Lingkup Masalah.....	6
1.3	Definisi, Akronim dan Singkatan.....	7
1.4	Referensi.....	7
1.5	Deskripsi umum (Overview).....	8
2	Deskripsi Kebutuhan .....	9
2.1	Perspektif produk.....	9
2.2	Fungsi Produk.....	10
2.3	Karakteristik Pengguna.....	13
2.4	Batasan-batasan.....	13
2.5	Asumsi dan Ketergantungan.....	14
3	Kebutuhan khusus .....	14
3.1	Kebutuhan antarmuka eksternal.....	14
3.1.1	Antarmuka pemakai.....	14
3.1.2	Antarmuka perangkat keras.....	14
3.1.3	Antarmuka perangkat lunak.....	15
3.1.4	Antarmuka Komunikasi.....	15
3.2	Kebutuhan fungsionalitas Perangkat Lunak.....	16
3.2.1	Use Case Diagram.....	16
4	Spesifikasi Rinci Kebutuhan .....	16
4.1	Spesifikasi Kebutuhan Fungsionalitas.....	16
4.1.1	Spesifikasi Use Case : Login.....	16
4.1.2	Spesifikasi Use Case : Identifikasi Jenis Alergi	18
4.1.3	Spesifikasi Use Case : Pengelolaan Admin.....	19
4.1.4	Spesifikasi Use Case : Pengelolaan Gejala Alergi	21
4.1.5	Spesifikasi Use Case : Pengelolaan Jenis Alergi.	24
4.1.6	Spesifikasi Use Case : Pengelolaan Penanganan Alergi.....	26
4.1.7	Spesifikasi Use Case : Pengelolaan Relasi Antara Gejala dan Jenis Alergi.....	29
5	Entity Relationship Diagram (ERD) .....	32

## Daftar Gambar

Gambar 2.1	Arsitektur SiPaGi.....	9
Gambar 3.1	Use Case Diagram SiPaGi.....	16
Gambar 5.1	Entity Relationship Diagram (ERD) .....	32



## **1 Pendahuluan**

### **1.1 Tujuan**

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak SiPaGi (Sistem Pakar Alergi) untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain perangkat lunak dan perangkat keras, dan pengguna) dan atribut (*feature-feature* tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-SiPaGi ini juga mendefinisikan batasan perancangan perangkat lunak.

### **1.2 Lingkup Masalah**

Perangkat Lunak SiPaGi dikembangkan dengan tujuan untuk:

1. Memilih dan mengidentifikasi gejala-gejala alergi bagi user.
2. Menangani pengelolaan Admin.
3. Menangani pengelolaan gejala alergi.
4. Menangani pengelolaan jenis-jenis alergi.
5. Menangani pengelolaan penanganan dari alergi.
6. Menangani pengelolaan relasi antara gejala dan jenis alergi.

Perangkat lunak SiPaGi berjalan pada lingkungan dengan platform Desktop.

### 1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan:

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-SIPAGI-XXX	Kode yang merepresentasikan kebutuhan pada SIPAGI (Sistem Pakar Alergi) dimana XXX merupakan nomor fungsi produk.
SIPAGI	Perangkat lunak Sistem Pakar Alergi yang akan dibangun.
ERD	Entity Relationship Diagram merupakan teknis grafis/diagram yang menggambarkan objek dan hubungan antar objek.
Admin	Orang yang mengelola user yang menggunakan perangkat lunak, gejala-gejala alergi, jenis alergi, penanganan alergi dan relasi antara gejala alergi dan jenis alergi.
Pasien	Orang yang menggunakan perangkat lunak untuk memilih dan mengidentifikasi alergi.

### 1.4 Referensi

Referensi yang digunakan pada perangkat lunak SiPaGi adalah:

1. GLO1, *Spesifikasi Kebutuhan Perangkat Lunak*, Jurusan Teknik Informatika-UAJY.
2. Heryanto, Andrie, *Spesifikasi Kebutuhan Perangkat Lunak-Pengembangan Sistem Pakar Identifikasi Ikan*

*Hias Laut (SisKaFiKanHiaLa)*, Program Studi Teknik Informatika UAJY, 2008.

3. Triseptian, Anastasius, *Spesifikasi Kebutuhan Perangkat Lunak-Pengembangan Sistem Informasi Tugas Akhir (SITA) Berbasis Web Untuk Sistem Informasi Akademik Universitas Atma Jaya Yogyakarta*, Program Studi Teknik Informatika UAJY, 2010.

### **1.5 Deskripsi umum (Overview)**

Secara umum dokumen SKPL ini terbagi atas 3 bagian utama. Bagian utama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak tersebut, definisi, referensi dan deskripsi umum tentang dokumen SKPL ini.

Bagian kedua berisi penjelasan umum tentang perangkat lunak SiPaGi yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak SiPaGi tersebut.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak SiPaGi yang akan dikembangkan.

Program Studi Teknik Informatika	SKPL-SiPaGi	8/ 32
<b>Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika</b>		

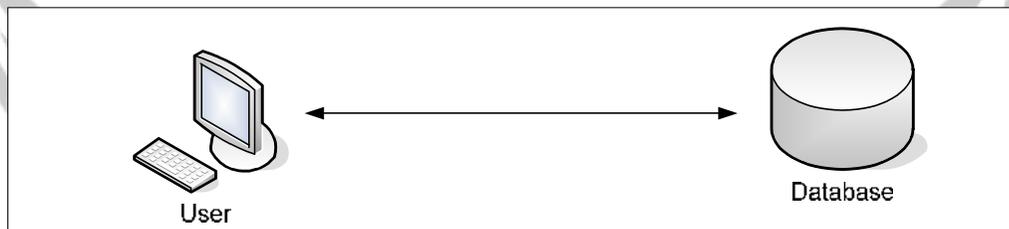
## 2 Deskripsi Kebutuhan

### 2.1 Perspektif produk

SiPaGi merupakan perangkat lunak yang dikembangkan untuk mendiagnosis jenis alergi dari gejala yang dipilih. Sistem ini diharapkan dapat menjadi pedoman bagi para pengguna dalam mengidentifikasi alergi yang dialami.

Pada aplikasi ini terdapat 2 buah role, yaitu admin dan pasien. Pasien akan memilih gejala-gejala alergi dan melihat hasil identifikasi alergi berdasarkan gejala yang dipilih. Admin dapat melakukan pengelolaan admin, pengelolaan gejala alergi, pengelolaan jenis alergi, pengelolaan penanganan alergi dan pengelolaan relasi antara gejala dan jenis alerginya.

Perangkat lunak SiPaGi ini berjalan pada platform Windows XP pada komputer (PC), dan dibuat menggunakan bahasa pemrograman Microsoft Visual C#. Sedangkan untuk lingkungan pemrogramannya menggunakan Microsoft Visual Studio 2005.



**Gambar 2.1 Arsitektur SiPaGi**

## 2.2 Fungsi Produk

Fungsi produk perangkat lunak SiPaGi adalah sebagai berikut:

### A. Pasien

#### 1. Fungsi *Identifikasi Alergi* (**SKPL-SiPaGi-A-01**).

Fungsi ini merupakan fungsi yang digunakan oleh user dengan role Pasien untuk memilih gejala-gejala alergi yang akan diidentifikasi oleh sistem dan menampilkan hasil identifikasi alergi beserta penanganannya.

### B. Admin

#### 1. Fungsi *Login* (**SKPL-SiPaGi-B-01**).

Fungsi Login merupakan fungsi yang digunakan oleh user dengan role Admin untuk masuk ke dalam sistem.

#### 2. Fungsi *Pengelolaan Admin* (**SKPL-SiPaGi-B-02**).

Fungsi ini merupakan fungsi yang digunakan untuk melakukan pengelolaan administrator.

Fungsi Pengelolaan Admin meliputi:

a. Fungsi *Add Admin* (**SKPL-SiPaGi-B-02-01**). Fungsi yang digunakan untuk menambahkan admin baru.

b. Fungsi *Edit Admin* (**SKPL-SiPaGi-B-02-02**). Fungsi yang digunakan untuk mengubah data admin.

c. Fungsi *Delete Admin* (**SKPL-SiPaGi-B-02-03**). Fungsi yang digunakan untuk menghapus data admin.

d. Fungsi *Display Admin* (**SKPL-SiPaGi-B-02-04**). Fungsi yang digunakan untuk menampilkan data admin.

e. Fungsi *Search Admin* (**SKPL-SiPaGi-B-02-05**). Fungsi yang digunakan untuk mencari data admin.

3. Fungsi *Pengelolaan Gejala* (**SKPL-SiPaGi-B-03**).

Fungsi ini merupakan fungsi yang digunakan untuk melakukan pengelolaan gejala.

Fungsi *Pengelolaan Gejala* meliputi:

a. Fungsi *Add Gejala* (**SKPL-SiPaGi-B-03-01**). Fungsi yang digunakan untuk menambahkan gejala baru.

b. Fungsi *Edit Gejala* (**SKPL-SiPaGi-B-03-02**). Fungsi yang digunakan untuk mengubah gejala.

c. Fungsi *Delete Gejala* (**SKPL-SiPaGi-B-03-03**). Fungsi yang digunakan untuk menghapus gejala.

d. Fungsi *Display Gejala* (**SKPL-SiPaGi-B-03-04**). Fungsi yang digunakan untuk menampilkan gejala.

e. Fungsi *Search Gejala* (**SKPL-SiPaGi-B-03-05**). Fungsi yang digunakan untuk mencari gejala tertentu.

4. Fungsi *Pengelolaan Alergi* (**SKPL-SiPaGi-B-04**).

Fungsi ini merupakan fungsi yang digunakan untuk melakukan pengelolaan alergi.

Fungsi *Pengelolaan Alergi* meliputi:

a. Fungsi *Add Alergi* (**SKPL-SiPaGi-B-04-01**). Fungsi yang digunakan untuk menambahkan alergi.

b. Fungsi *Edit Jenis Alergi* (**SKPL-SiPaGi-B-04-02**). Fungsi yang digunakan untuk mengubah data alergi.

c. Fungsi *Delete Alergi* (**SKPL-SiPaGi-B-04-03**). Fungsi yang digunakan untuk menghapus data alergi tertentu.

- d. Fungsi *Display Alergi* **(SKPL-SiPaGi-B-04-04)**.  
Fungsi yang digunakan untuk menampilkan alergi.
- e. Fungsi *Search Alergi* **(SKPL-SiPaGi-B-04-05)**.  
Fungsi yang digunakan untuk mencari jenis tertentu.
5. Fungsi *Pengelolaan Penanganan* **(SKPL-SiPaGi-B-05)**.  
Fungsi ini merupakan fungsi yang digunakan untuk melakukan Pengelolaan Penanganan.  
Fungsi Pengelolaan Penanganan meliputi:
- a. Fungsi *Add Penanganan* **(SKPL-SiPaGi-B-05-01)**.  
Fungsi yang digunakan untuk menambahkan data penanganan baru.
- b. Fungsi *Edit Penanganan* **(SKPL-SiPaGi-B-05-02)**.  
Fungsi yang digunakan untuk mengubah data penanganan tertentu.
- c. Fungsi *Delete Penanganan* **(SKPL-SiPaGi-B-05-03)**.  
Fungsi yang digunakan untuk menghapus penanganan tertentu.
- d. Fungsi *Display Penanganan* **(SKPL-SiPaGi-B-05-04)**.  
Fungsi yang digunakan untuk menampilkan data penanganan.
- e. Fungsi *Search Penanganan* **(SKPL-SiPaGi-B-05)**.  
Fungsi yang digunakan untuk mencari penanganan tertentu.
6. Fungsi *Pengelolaan Relasi* **(SKPL-SiPaGi-B-06)**.  
Fungsi ini merupakan fungsi yang digunakan untuk melakukan pengelolaan relasi antara gejala alergi dan jenis alerginya.

Fungsi Pengelolaan Relasi mencakup:

- a. Fungsi *Add Relasi* (**SKPL-SiPaGi-B-06-01**). Fungsi yang digunakan untuk menambahkan relasi baru antara gejala dan jenis alerginya.
- b. Fungsi *Edit Relasi* (**SKPL-SiPaGi-B-06-02**). Fungsi yang digunakan untuk mengubah relasi tertentu.
- c. Fungsi *Delete Relasi* (**SKPL-SiPaGi-B-06-03**). Fungsi yang digunakan untuk menghapus relasi tertentu.
- d. Fungsi *Display Relasi* (**SKPL-SiPaGi-B-06-04**). Fungsi yang digunakan untuk menampilkan relasi antara gejala dan jenis alerginya.
- e. Fungsi *Search Relasi* (**SKPL-SiPaGi-B-06-05**). Fungsi yang digunakan untuk mencari relasi tertentu.

### **2.3 Karakteristik Pengguna**

Karakteristik dari pengguna perangkat lunak SiPaGi adalah sebagai berikut:

1. Memahami pengoperasian komputer PC.
2. Mengerti dan memahami perangkat lunak yang digunakan.

### **2.4 Batasan-batasan**

Batasan-batasan dalam pengembangan perangkat lunak SiPaGi tersebut adalah:

1. Kebijakan Umum

Kebijakan umum berpedoman pada tujuan dari pengembangan perangkat lunak SiPaGi.

Program Studi Teknik Informatika	SKPL-SiPaGi	13/ 32
<b>Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika</b>		

## 2. Keterbatasan Perangkat Keras

Keterbatasan perangkat keras dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

### 2.5 Asumsi dan Ketergantungan

Sistem ini dapat dijalankan pada komputer PC yang menggunakan sistem operasi Windows.

## 3 Kebutuhan khusus

### 3.1 Kebutuhan antarmuka eksternal

Kebutuhan antar muka eksternal pada perangkat lunak SiPaGi meliputi kebutuhan antarmuka pemakai, antarmuka perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi.

#### 3.1.1 Antarmuka pemakai

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam layar komputer dengan format windows form dengan pilihan fungsi dan form untuk pengisian data dan tampilan informasi pada layar monitor.

#### 3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan dalam perangkat lunak SIPAGI adalah:

1. Komputer (PC)
2. Keyboard dan Mouse
3. Monitor

### 3.1.3 Antarmuka perangkat lunak

Perangkat lunak yang dibutuhkan untuk mengoperasikan perangkat lunak SIPAGI adalah sebagai berikut :

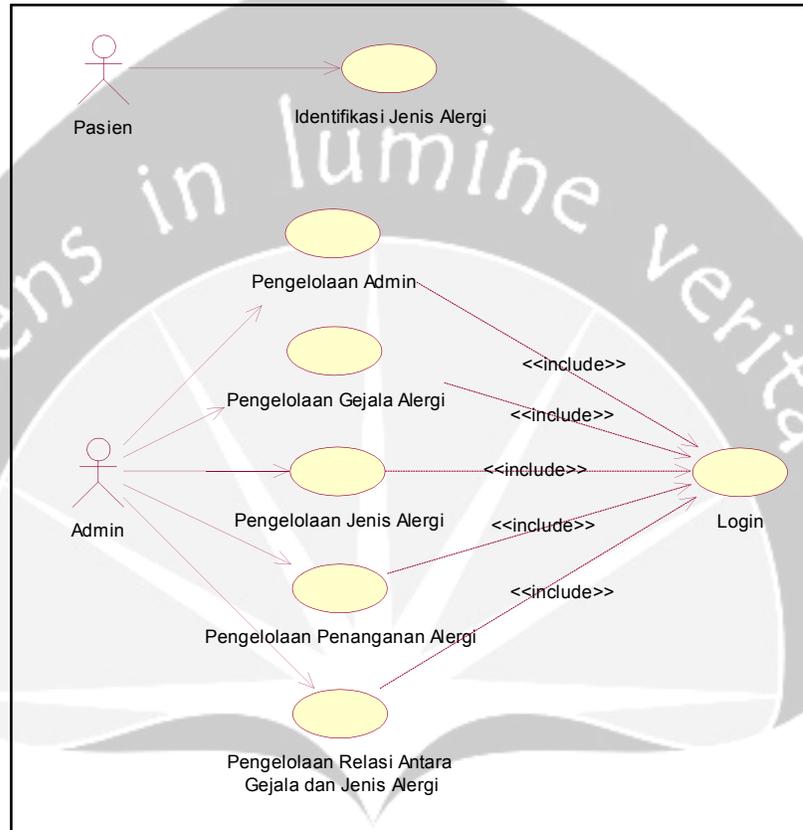
1. Nama : SQL Server 2005  
Sumber : Microsoft  
Sebagai database management system (DBMS) yang digunakan untuk menyimpan data di sisi server.
2. Nama : Windows XP  
Sumber : Microsoft.  
Sebagai sistem operasi untuk komputer.
3. Nama : Microsoft Visual C#.NET  
Sumber : Microsoft.  
Sebagai aplikasi tool development.

### 3.1.4 Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak SIPAGI menggunakan protocol TCP/IP.

### 3.2 Kebutuhan fungsionalitas Perangkat Lunak

#### 3.2.1 Use Case Diagram



Gambar 3.1 Use Case Diagram SiPaGi

## 4 Spesifikasi Rinci Kebutuhan

### 4.1 Spesifikasi Kebutuhan Fungsionalitas

#### 4.1.1 Spesifikasi Use Case : Login

Use Case ID	UC-SIPAGI-01
Use Case Name	Login

Use Case Type	Essential
Priority	High
Primary Actors	Admin
Supporting Actors	-
Description	Use case ini digunakan Aktor supaya dapat masuk ke sistem. Didasarkan pada nama yang unik atau username, password dan rolenya.
Preconditions	-
Basic Path	<ol style="list-style-type: none"> <li>1. Sistem menampilkan antar muka untuk login.</li> <li>2. Aktor memasukkan username dan password.</li> <li>3. Sistem memeriksa username dan password ke dalam basis data.</li> <li>4. Sistem memberikan akses ke Aktor sesuai rolenya.</li> </ol>
Alternative Paths	-
Exception Paths	<p>E1. Nama dan password tidak sesuai (setelah Basic Path 2)</p> <ol style="list-style-type: none"> <li>a Sistem menampilkan peringatan bahwa nama user atau passwordnya tidak sesuai.</li> <li>b Kembali ke Basic Path langkah 2.</li> </ol>
Extends	-
Includes	-

Postconditions	Aktor masuk ke sistem dan dapat menggunakan fungsi-fungsi dalam sistem.
----------------	---

#### 4.1.2 Spesifikasi Use Case : Identifikasi Jenis Alergi

Use Case ID	UC-SIPAGI-02
Use Case Name	Identifikasi Jenis Alergi
Use Case Type	Essential
Priority	High
Primary Actors	Pasien
Supporting Actors	-
Description	Use case ini digunakan Aktor untuk mengidentifikasi jenis alergi.
Preconditions	-
Basic Path	<ol style="list-style-type: none"> <li>1. Sistem menampilkan antar muka untuk memilih gejala alergi.</li> <li>2. Aktor memilih gejala alergi.</li> <li>3. Sistem memasukkan gejala alergi yang akan diidentifikasi.</li> <li>4. Sistem menampilkan antar muka hasil identikasi alergi.</li> <li>5. Sistem akan menampilkan hasil identifikasi alergi berdasarkan gejala yang dipilih aktor serta menampilkan penanganan dari alergi tersebut.</li> </ol>
Alternative	-

Paths	
Exception Paths	-
Extends	-
Includes	-
Postconditions	Gejala alergi telah dipilih.

#### 4.1.3 Spesifikasi Use Case : Pengelolaan Admin

Use Case ID	UC-SIPAGI-04
Use Case Name	Pengelolaan Admin
Use Case Type	Essential
Priority	High
Primary Actors	Admin
Supporting Actors	-
Description	Use case yang digunakan Aktor supaya dapat melakukan pengelolaan data admin.
Preconditions	Use Case : Login sukses dijalankan, role : Admin
Basic Paths	Add Data Admin <ol style="list-style-type: none"> <li>1. Sistem menampilkan antarmuka untuk penambahan admin.</li> <li>2. Aktor memasukkan detil data admin baru.</li> <li>3. Sistem menyimpan data admin baru ke basis data.</li> <li>4. Sistem menampilkan pesan bahwa</li> </ol>

	proses add data admin berhasil dijalankan.
Alternative Paths	<p>A1. Edit Data Admin</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan antarmuka untuk edit data admin.</li> <li>2. Aktor memilih data admin yang akan diubah.</li> <li>3. Aktor mengubah data admin.</li> <li>4. Sistem menyimpan data admin yang sudah diubah ke dalam basis data.</li> </ol> <p>A2. Delete Data Admin</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan antar muka untuk menghapus data admin.</li> <li>2. Aktor memilih data admin yang akan dihapus.</li> <li>3. Sistem menghapus data admin dari basis data.</li> </ol> <p>A3. Searching dan Display Data Admin</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan antar muka untuk menampilkan data admin.</li> <li>2. Aktor memasukkan data admin yang akan ditampilkan.</li> <li>3. Sistem mencari dan menampilkan data admin.</li> </ol>
Exception Paths	E1. Data admin telah terdaftar dalam basis data (setelah Basic

	<p>Path 2 atau Alternative Paths A1-3).</p> <p>a. Sistem menampilkan peringatan bahwa data admin telah ada.</p> <p>b. Kembali ke Basic Path 2 atau Alternative Paths A1-3.</p> <p>E2. Data admin tidak ada dalam basis data (setelah Alternative Paths A2-2 atau Alternative Paths A3-2).</p> <p>a. Sistem menampilkan peringatan bahwa data admin tidak ada.</p> <p>b. Kembali ke Alternative Paths A2-2 atau Alternative Paths A3-2.</p>
Extends	-
Includes	Use Case : Login
Postconditions	Data admin di dalam basis data ter-update.

#### 4.1.4 Spesifikasi Use Case : Pengelolaan Gejala

Use Case ID	UC-SIPAGI-05
Use Case Name	Pengelolaan Gejala
Use Case Type	Essential
Priority	High
Primary Actors	Admin
Supporting Actors	-

Description	Use case yang digunakan Aktor supaya dapat melakukan pengelolaan data gejala.
Preconditions	Use Case : Login sukses dijalankan, role : Admin
Basic Paths	Add Data Gejala 1. Sistem menampilkan antarmuka untuk penambahan gejala. 2. Aktor memasukkan detail data gejala baru. 3. Sistem menyimpan data gejala baru ke basis data. 4. Sistem menampilkan pesan bahwa proses add data gejala berhasil dijalankan.
Alternative Paths	A1. Edit Data Gejala 1. Sistem menampilkan antarmuka untuk edit data gejala. 2. Aktor memilih data gejala yang akan diubah. 3. Aktor mengubah data gejala. 4. Sistem menyimpan data gejala yang sudah diubah ke dalam basis data. A2. Delete Data Gejala 1. Sistem menampilkan antar muka untuk menghapus data gejala. 2. Aktor memilih data gejala yang

	<p>akan dihapus.</p> <p>3. Sistem menghapus data gejala dari basis data.</p> <p>A3. Searching dan Display Data Gejala</p> <p>1. Sistem menampilkan antar muka untuk menampilkan data gejala.</p> <p>2. Aktor memasukkan data gejala yang akan ditampilkan.</p> <p>3. Sistem mencari dan menampilkan data gejala.</p>
Exception Paths	<p>E1. Data gejala telah terdaftar dalam basis data (setelah Basic Path 2 atau Alternative Paths A1-3).</p> <p>a. Sistem menampilkan peringatan data gejala telah ada.</p> <p>b. Kembali ke Basic Path 2 atau Alternative Paths A1-3.</p> <p>E2. Data gejala tidak ada dalam basis data (setelah Alternative Paths A2-2 atau Alternative Paths A3-2).</p> <p>a. Sistem menampilkan peringatan bahwa data gejala tidak ada.</p> <p>b. Kembali ke Alternative Paths A2-2 atau Alternative Paths A3-2.</p>

Extends	-
Includes	Use Case : Login
Postconditions	Data gejala alergi di dalam basis data ter-update.

#### 4.1.5 Spesifikasi Use Case : Pengelolaan Alergi

Use Case ID	UC-SIPAGI-06
Use Case Name	Pengelolaan Alergi
Use Case Type	Essential
Priority	High
Primary Actors	Admin
Supporting Actors	-
Description	Use case yang digunakan Aktor supaya dapat melakukan pengelolaan data alergi.
Preconditions	Use Case : Login sukses dijalankan, role : Admin
Basic Paths	<p>Add Data Alergi</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan antarmuka untuk penambahan alergi.</li> <li>2. Aktor memasukkan detil data alergi baru.</li> <li>3. Sistem menyimpan data alergi baru ke basis data.</li> <li>4. Sistem menampilkan pesan bahwa proses add data alergi berhasil dijalankan.</li> </ol>

<p>Alternative Paths</p>	<p>A1. Edit Data Alergi</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan antarmuka untuk edit data alergi.</li> <li>2. Aktor memilih data alergi yang akan diubah.</li> <li>3. Aktor mengubah data alergi.</li> <li>4. Sistem menyimpan data alergi yang sudah diubah ke dalam basis data.</li> </ol> <p>A2. Delete Data Alergi</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan antar muka untuk menghapus data alergi.</li> <li>2. Aktor memilih data alergi yang akan dihapus.</li> <li>3. Sistem menghapus data alergi dari basis data.</li> </ol> <p>A3. Searching dan Display Data Alergi</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan antar muka untuk menampilkan data alergi.</li> <li>2. Aktor memasukkan data alergi yang akan ditampilkan.</li> <li>3. Sistem mencari dan menampilkan data alergi.</li> </ol>
<p>Exception Paths</p>	<p>E1. Data alergi telah terdaftar dalam basis data (setelah Basic Path 2 atau Alternative Paths A1-3).</p>

	<p>4. Sistem menampilkan peringatan bahwa data alergi telah ada.</p> <p>a. Kembali ke Basic Path 2 atau Alternative Paths A1-3.</p> <p>E2. Data alergi tidak ada dalam basis data (setelah Alternative Paths A2-2 atau Alternative Paths A3-2).</p> <p>a. Sistem menampilkan peringatan bahwa data alergi tidak ada.</p> <p>b. Kembali ke Alternative Paths A2-2 atau Alternative Paths A3-2.</p>
Extends	-
Includes	Use Case : Login
Postconditions	Data jenis alergi di dalam basis data ter-update.

#### 4.1.6 Spesifikasi Use Case : Pengelolaan Penanganan

Use Case ID	UC-SIPAGI-07
Use Case Name	Pengelolaan Penanganan
Use Case Type	Essential
Priority	High
Primary Actors	Admin
Supporting Actors	-
Description	Use case yang digunakan Aktor supaya dapat melakukan

	pengelolaan data penanganan .
Preconditions	Use Case : Login sukses dijalankan, role : Admin
Basic Paths	Add Data Penanganan Alergi <ol style="list-style-type: none"> <li>1. Sistem menampilkan antarmuka untuk penambahan penanganan .</li> <li>2. Aktor memasukkan detil data penanganan baru.</li> <li>3. Sistem menyimpan data penanganan baru ke basis data.</li> <li>4. Sistem menampilkan pesan bahwa proses add data penanganan berhasil dijalankan.</li> </ol>
Alternative Paths	A1. Edit Data Penanganan <ol style="list-style-type: none"> <li>1. Sistem menampilkan antarmuka untuk edit data penanganan .</li> <li>2. Aktor memilih data penanganan yang akan diubah.</li> <li>3. Aktor mengubah data penanganan.</li> <li>4. Sistem menyimpan data penanganan yang sudah diubah ke dalam basis data.</li> </ol> A2. Delete Data Penanganan <ol style="list-style-type: none"> <li>1. Sistem menampilkan antar muka untuk menghapus data penanganan.</li> <li>2. Aktor memilih data penanganan yang akan dihapus.</li> </ol>

	<p>3. Sistem menghapus data penanganan dari basis data.</p> <p>A3. Searching dan Display Data Penanganan</p> <p>1. Sistem menampilkan antar muka untuk menampilkan data penanganan.</p> <p>2. Aktor memasukkan data penanganan yang akan ditampilkan.</p> <p>3. Sistem mencari dan menampilkan data penanganan.</p>
<p>Exception Paths</p>	<p>E1. Data penanganan telah terdaftar dalam basis data (setelah Basic Path 2 atau Alternative Paths A1-3).</p> <p>a. Sistem menampilkan peringatan bahwa data penanganan telah ada.</p> <p>b. Kembali ke Basic Path 2 atau Alternative Paths A1-3.</p> <p>E2. Data penanganan tidak ada dalam basis data (setelah Alternative Paths A2-2 atau Alternative Paths A3-2).</p> <p>a. Sistem menampilkan peringatan bahwa data penanganan tidak ada.</p>

	b. Kembali ke Alternative Paths A2-2 atau Alternative Paths A3-2.
Extends	-
Includes	Use Case : Login
Postconditions	Data penanganan di dalam basis data ter-update.

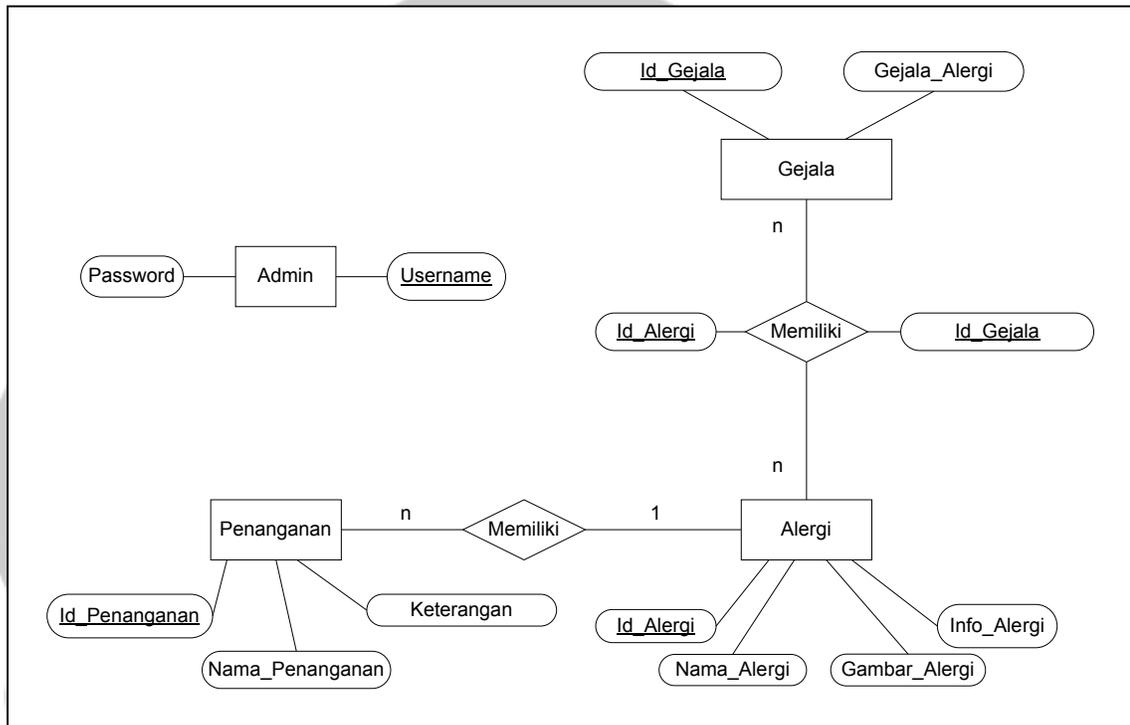
#### 4.1.7 Spesifikasi Use Case : Pengelolaan Relasi

Use Case ID	UC-SIPAGI-08
Use Case Name	Pengelolaan Relasi
Use Case Type	Essential
Priority	High
Primary Actors	Admin
Supporting Actors	-
Description	Use case yang digunakan Aktor supaya dapat melakukan pengelolaan data relasi antara gejala dan jenis alergi.
Preconditions	Use Case : Login sukses dijalankan, role : Admin
Basic Paths	Add Data Relasi 1. Sistem menampilkan antarmuka untuk penambahan relasi antara gejala dan jenis alergi. 2. Aktor memasukkan detail data relasi baru.

	<p>3. Sistem menyimpan data relasi baru ke basis data.</p> <p>4. Sistem menampilkan pesan bahwa proses add data relasi berhasil dijalankan.</p>
Alternative Paths	<p>A1. Edit Data Relasi</p> <p>1. Sistem menampilkan antarmuka untuk edit data relasi.</p> <p>2. Aktor memilih data relasi antara gejala dan jenis alergi yang akan diubah.</p> <p>3. Aktor mengubah data relasi.</p> <p>4. Sistem menyimpan data relasi yang sudah diubah ke dalam basis data.</p> <p>A2. Delete Data Relasi</p> <p>1. Sistem menampilkan antar muka untuk menghapus data relasi.</p> <p>2. Aktor memilih data relasi antara gejala dan jenis alergi yang akan dihapus.</p> <p>3. Sistem menghapus data relasi dari basis data.</p> <p>A3. Searching dan Display Data Relasi</p> <p>1. Sistem menampilkan antar muka untuk menampilkan data relasi.</p> <p>2. Aktor memasukkan data relasi</p>

	<p>antara gejala dan jenis alergi yang akan ditampilkan.</p> <p>3. Sistem mencari dan menampilkan data relasi.</p>
Exception Paths	<p>E1. Data relasi telah terdaftar dalam basis data (setelah Basic Path 2 atau Alternative Paths A1-3).</p> <p>a. Sistem menampilkan peringatan bahwa data relasi antara gejala dan jenis alergi telah ada.</p> <p>b. Kembali ke Basic Path 2 atau Alternative Paths A1-3.</p> <p>E2. Data relasi tidak ada dalam basis data (setelah Alternative Paths A2-2 atau Alternative Paths A3-2).</p> <p>a. Sistem menampilkan peringatan bahwa data relasi antara gejala dan jenis alergi tidak ada.</p> <p>b. Kembali ke Alternative Paths A2-2 atau Alternative Paths A3-2.</p>
Extends	-
Includes	Use Case : Login
Postconditions	Data relasi antara gejala dan jenis alergi di dalam basis data <i>ter-update</i> .

## 5 Entity Relationship Diagram (ERD)



Gambar 5.1 Entity Relationship Diagram (ERD)

**DPPL**

**DESKRIPSI PERANCANGAN PERANGKAT LUNAK**

**APLIKASI SISTEM PAKAR DIAGNOSIS JENIS ALERGI DAN  
PENANGANANNYA PADA PENDERITA ALERGI  
(SiPaGi)**

**Untuk:**

**Universitas Atma Jaya Yogyakarta**

**Dipersiapkan oleh:**

**Fani Damayanti**

**06 07 04912**

**Program Studi Teknik Informatika - Fakultas Teknologi  
Industri  
Universitas Atma Jaya Yogyakarta**

	Program Studi Teknik Informatika  Fakultas Teknologi Industri	<b>Nomor Dokumen</b>		<b>Halaman</b>
		<b><i>DPPL-SiPaGi</i></b>		<i>1/109</i>
		<b>Revisi</b>		

## DAFTAR PERUBAHAN

Revisi	Deskripsi
<b>A</b>	
<b>B</b>	
<b>C</b>	
<b>D</b>	
<b>E</b>	
<b>F</b>	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								

## Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi



## Daftar Isi

1. Pendahuluan .....	8
1.1 Tujuan.....	8
1.2 Ruang Lingkup.....	8
1.3 Definisi dan Akronim.....	9
1.4 Referensi.....	9
2. Perancangan Sistem .....	10
2.1 Rancangan Arsitektur.....	10
2.2 Perancangan Rinci.....	11
2.2.1 Sequence Diagram.....	11
2.2.2 Class Diagram.....	25
2.2.3 Deskripsi Kelas.....	25
3. Perancangan Data .....	50
3.1 Dekomposisi Data.....	50
3.1.1 Deskripsi Entitas Admin.....	50
3.1.2 Deskripsi Entitas Gejala.....	50
3.1.3 Deskripsi Entitas Alergi.....	50
3.1.4 Deskripsi Entitas Penanganan.....	50
3.1.5 Deskripsi Entitas Relasi.....	51
3.2 Physical Data Model.....	52
4. Deskripsi Perancangan Antarmuka .....	52
4.1 Menu Utama.....	52
4.2 Menu Pengelolaan.....	54
4.3 Pengelolaan Admin.....	56
4.3.1 Add Admin.....	57
4.3.2 Edit Admin.....	58
4.3.3 Delete Admin.....	61
4.4 Pengelolaan Gejala.....	63
4.4.1 Add Gejala.....	64
4.4.2 Edit Gejala.....	65
4.4.3 Delete Gejala.....	68
4.5 Pengelolaan Alergi.....	70
4.5.1 Add Alergi.....	71
4.5.2 Edit Alergi.....	73
4.5.3 Delete Alergi.....	76
4.6 Pengelolaan Relasi.....	78
4.6.1 Add Relasi.....	79
4.6.2 Edit Relasi.....	82
4.6.3 Delete Relasi.....	86
4.7 Pengelolaan Penanganan.....	89
4.7.1 Add Penanganan.....	90
4.7.2 Edit Penanganan.....	92
4.7.3 Delete Penanganan.....	96
4.8 Identifikasi Alergi.....	99
4.8.1 Sistem Pakar Alergi.....	101
4.8.2 Pemilihan Gejala.....	102

4.8.3	Konfirmasi Pemilihan Gejala.....	104
4.8.4	Hasil Identifikasi Alergi.....	106
4.8.5	Penanganan Alergi.....	108



## Daftar Gambar

Gambar 2.1	Arsitektur SiPaGi (Admin) .....	10
Gambar 2.2	Arsitektur SiPaGi (Pasien) .....	11
Gambar 2.3	Sequence Diagram - Login .....	11
Gambar 2.4	Sequence Diagram - Add Admin .....	12
Gambar 2.5	Sequence Diagram - Edit Admin .....	12
Gambar 2.6	Sequence Diagram - Delete Admin .....	13
Gambar 2.7	Sequence Diagram - Add Gejala .....	13
Gambar 2.8	Sequence Diagram - Edit Gejala .....	14
Gambar 2.9	Sequence Diagram - Delete Gejala .....	15
Gambar 2.10	Sequence Diagram - Add Alergi .....	16
Gambar 2.11	Sequence Diagram - Edit Alergi .....	16
Gambar 2.12	Sequence Diagram - Delete Alergi .....	17
Gambar 2.13	Sequence Diagram - Add Penanganan .....	18
Gambar 2.14	Sequence Diagram - Edit Penanganan .....	19
Gambar 2.15	Sequence Diagram - Delete Penanganan .....	20
Gambar 2.16	Sequence Diagram - Add Relasi .....	21
Gambar 2.17	Sequence Diagram - Edit Relasi .....	22
Gambar 2.18	Sequence Diagram - Delete Relasi .....	23
Gambar 2.19	Sequence Diagram - Identifikasi Alergi .....	24
Gambar 2.20	Class Diagram SiPaGi .....	25
Gambar 3.1	Physical Data Model .....	52
Gambar 4.1	Rancangan Antarmuka Menu Utama .....	52
Gambar 4.2	Rancangan Antarmuka Menu Pengelolaan .....	54
Gambar 4.3	Rancangan Antarmuka Pengelolaan Admin .....	56
Gambar 4.4	Rancangan Antarmuka Add Admin .....	57
Gambar 4.5	Rancangan Antarmuka Edit Admin .....	58
Gambar 4.6	Rancangan Antarmuka Delete Admin .....	61
Gambar 4.7	Rancangan Antarmuka Pengelolaan Gejala .....	63
Gambar 4.8	Rancangan Antarmuka Add Gejala .....	64
Gambar 4.9	Rancangan Antarmuka Edit Gejala .....	65
Gambar 4.10	Rancangan Antarmuka Delete Gejala .....	68
Gambar 4.11	Rancangan Antarmuka Pengelolaan Alergi .....	70
Gambar 4.12	Rancangan Antarmuka Add Alergi .....	71
Gambar 4.13	Rancangan Antarmuka Edit Alergi .....	73
Gambar 4.14	Rancangan Antarmuka Delete Alergi .....	76
Gambar 4.15	Rancangan Antarmuka Pengelolaan Relasi .....	78
Gambar 4.16	Rancangan Antarmuka Add Relasi .....	79
Gambar 4.17	Rancangan Antarmuka Edit Relasi .....	82
Gambar 4.18	Rancangan Antarmuka Delete Relasi .....	86
Gambar 4.19	Rancangan Antarmuka Pengelolaan Penanganan .....	89
Gambar 4.20	Rancangan Antarmuka Add Penanganan .....	90
Gambar 4.21	Rancangan Antarmuka Edit Penanganan .....	92
Gambar 4.22	Rancangan Antarmuka Delete Penanganan .....	96
Gambar 4.23	Rancangan Antarmuka Identifikasi Alergi .....	99

Gambar 4.24	Rancangan Antarmuka Halaman Awal Identifikasi .	101
Gambar 4.25	Rancangan Antarmuka Pemilihan Gejala .....	102
Gambar 4.26	Rancangan Antarmuka Konfirmasi Pemilihan Gejala	104
Gambar 4.27	Rancangan Antarmuka Hasil Indentifikasi Alergi	106
Gambar 4.28	Rancangan Antarmuka Penanganan Alergi .....	108



## **1. Pendahuluan**

### **1.1 Tujuan**

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) bertujuan untuk mendefinisikan perancangan perangkat lunak Sistem Pakar Alergi (SiPaGi) yang akan dikembangkan. Dokumen DPPL SiPaGi berisi rancangan sistem, rancangan data dan rancangan antarmuka dari perangkat lunak SiPaGi yang akan dibuat. Dokumen DPPL tersebut digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap selanjutnya.

### **1.2 Ruang Lingkup**

Perangkat Lunak SiPaGi dikembangkan dengan tujuan untuk:

1. Memilih dan mengidentifikasi gejala-gejala alergi bagi pasien.
2. Menangani pengelolaan Admin.
3. Menangani pengelolaan gejala alergi.
4. Menangani pengelolaan jenis-jenis alergi.
5. Menangani pengelolaan penanganan dari alergi.
6. Menangani pengelolaan relasi antara gejala dan jenis alergi.

Aplikasi ini berjalan pada lingkungan dengan platform Desktop.

Program Studi Teknik Informatika	DPPL – SiPaGi	8/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

### 1.3 Definisi dan Akronim

Daftar definisi akronim dan singkatan:

Keyword/Phrase	Definisi
DPPL	Deskripsi Perancangan Perangkat Lunak disebut juga Software Design Description (SDD) merupakan deskripsi dari perancangan produk/perangkat lunak yang akan dikembangkan.
SiPaGi	Perangkat lunak Sistem Pakar Alergi yang akan dibangun.
ERD	Entity Relationship Diagram merupakan teknis grafis/diagram yang menggambarkan objek dan hubungan antar objek.
Admin	Orang yang mengelola user yang menggunakan perangkat lunak, gejala-gejala alergi, jenis alergi, penanganan alergi dan relasi antara gejala alergi dan jenis alergi.
Pasien	Orang yang menggunakan perangkat lunak untuk memilih dan mengidentifikasi alergi.

### 1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

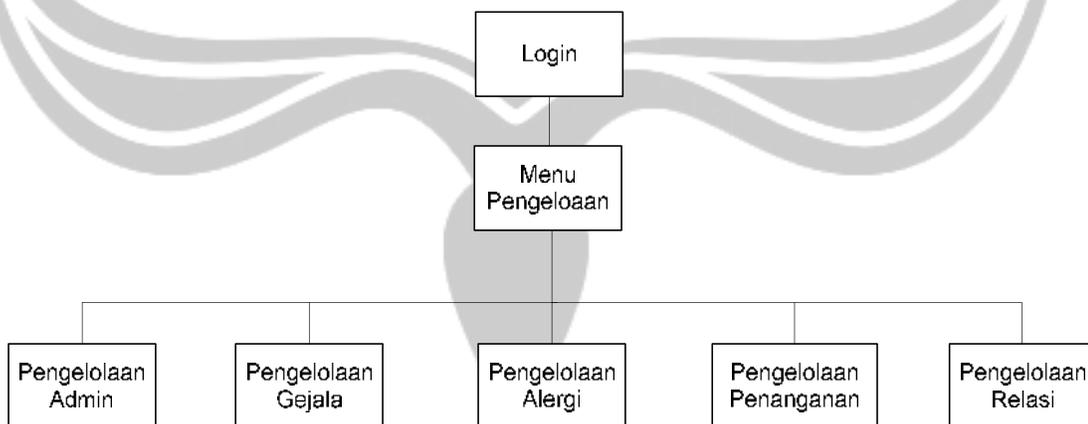
1. GL01, *Deskripsi Perancangan Perangkat Lunak*, Jurusan Teknik Informatika-UAJY.

Program Studi Teknik Informatika	DPPL – SiPaGi	9/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Heryanto, Andrie, *Deskripsi Perancangan Perangkat Lunak-Pengembangan Sistem Pakar Identifikasi Ikan Hias Laut (SisKaFiKanHiaLa)*, Program Studi Teknik Informatika UAJY, 2008.
3. Triseptian, Anastasius, *Deskripsi Perancangan Perangkat Lunak-Pengembangan Sistem Informasi Tugas Akhir (SITA) Berbasis Web Untuk Sistem Informasi Akademik Universitas Atma Jaya Yogyakarta*, Program Studi Teknik Informatika UAJY, 2010.
4. Damayanti, Fani, *Spesifikasi Kebutuhan Perangkat Lunak Pembangunan Aplikasi Sistem Pakar Diagnosis Jenis Alergi dan Penanganannya Pada Penderita Alergi*, Program Studi Teknik Informatika UAJY, 2010.

## 2. Perancangan Sistem

### 2.1 Rancangan Arsitektur



Gambar 2.1 Arsitektur SiPaGi (Admin)

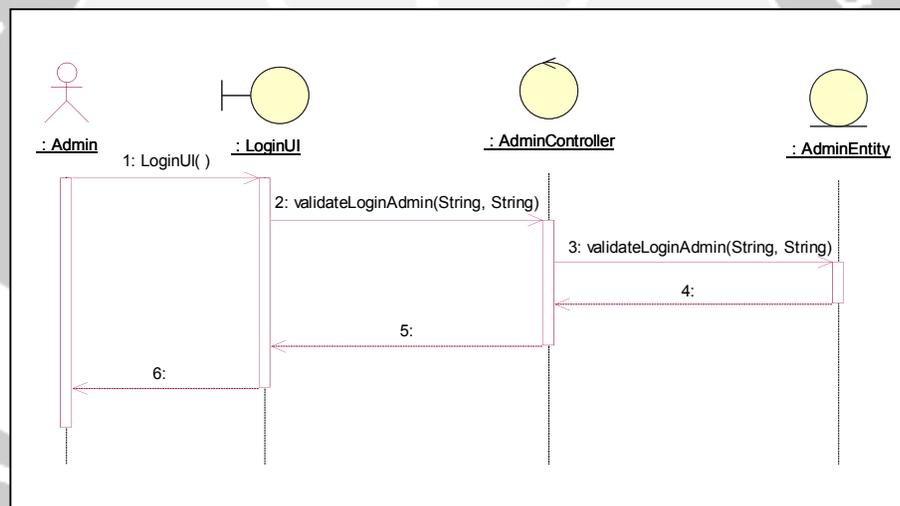


Gambar 2.2 Arsitektur SiPaGi (Pasien)

## 2.2 Perancangan Rinci

### 2.2.1 Sequence Diagram

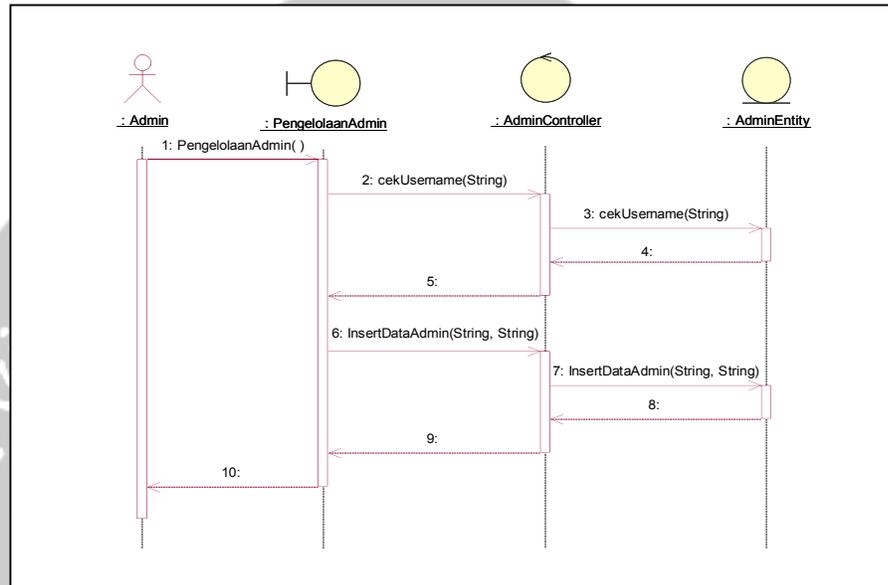
#### 2.2.1.1 Login



Gambar 2.3 Sequence Diagram - Login

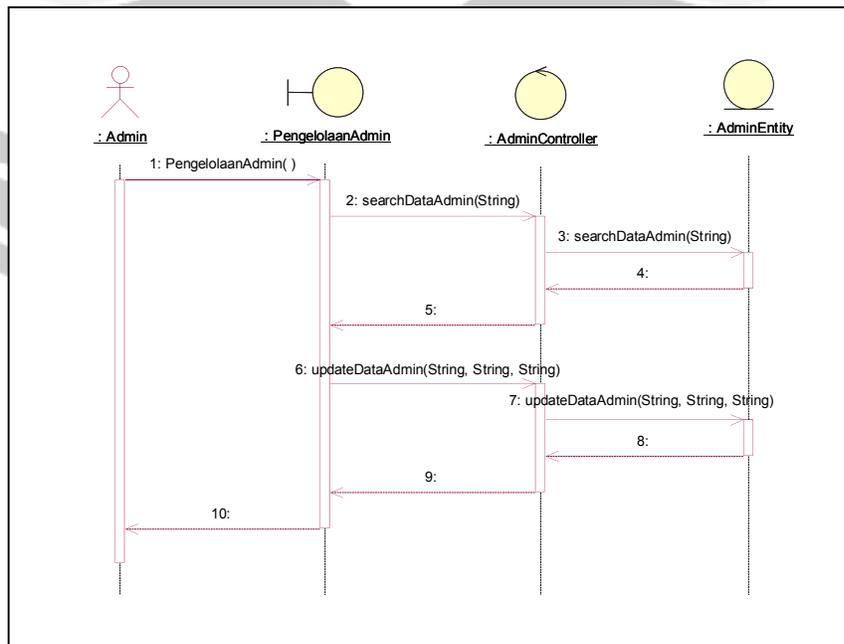
## 2.2.1.2 Pengelolaan Admin

### 2.2.1.2.1 Add Admin



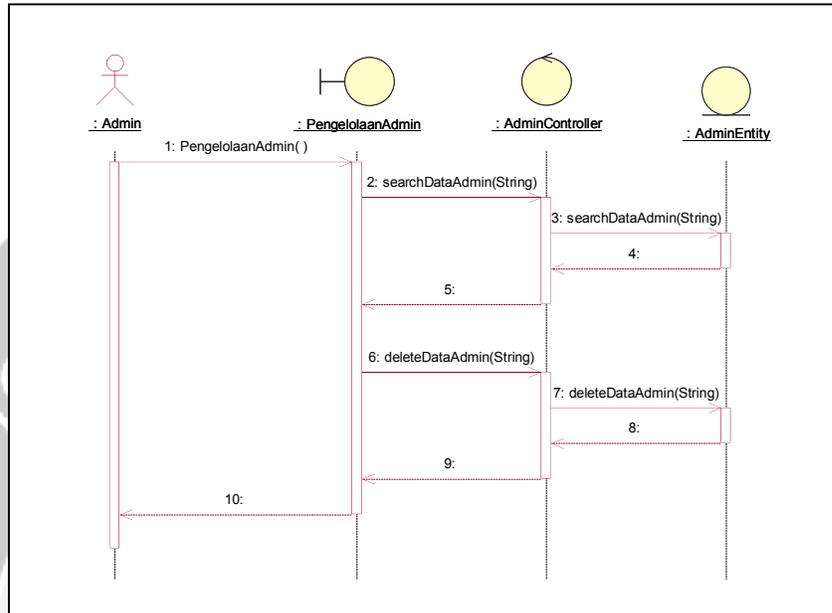
Gambar 2.4 Sequence Diagram - Add Admin

### 2.2.1.2.2 Edit Admin



Gambar 2.5 Sequence Diagram - Edit Admin

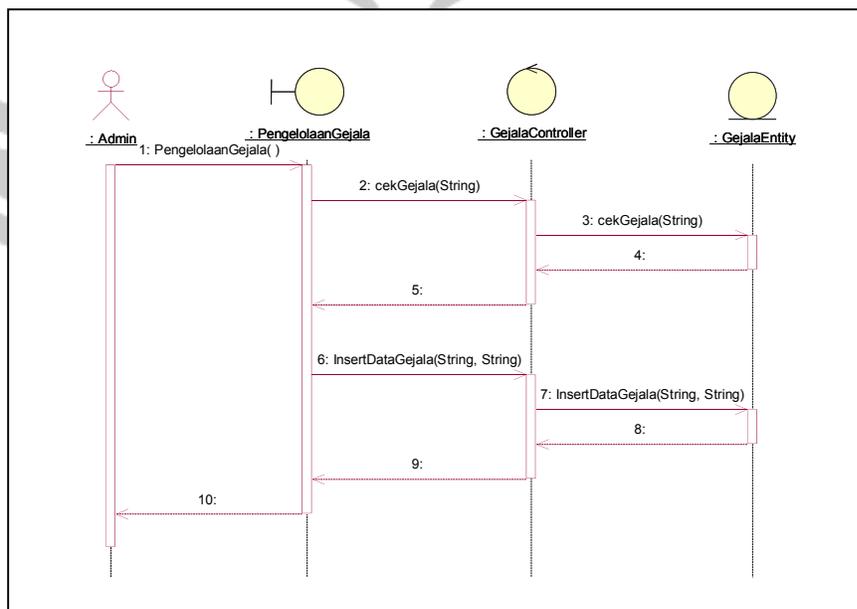
### 2.2.1.2.3 Delete admin



Gambar 2.6 Sequence Diagram - Delete Admin

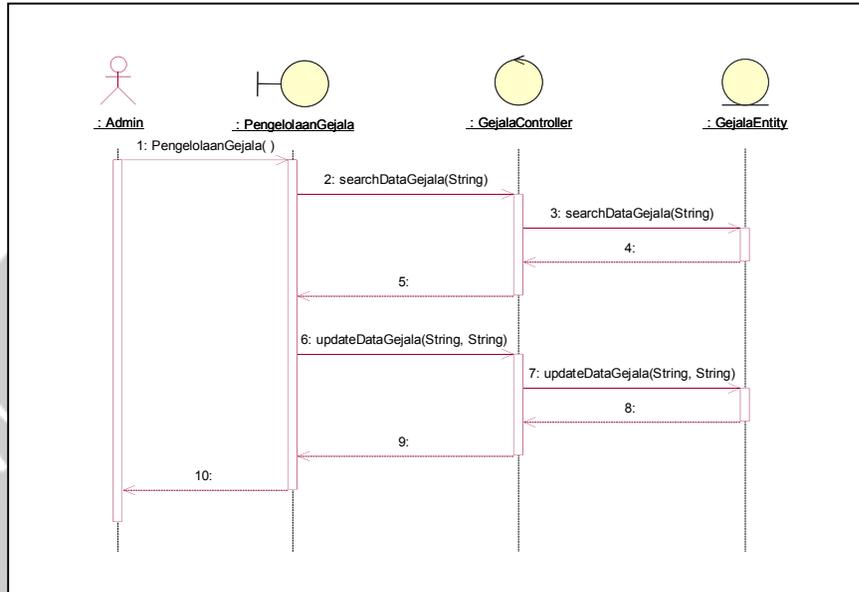
### 2.2.1.3 Pengelolaan Gejala

#### 2.2.1.3.1 Add Gejala



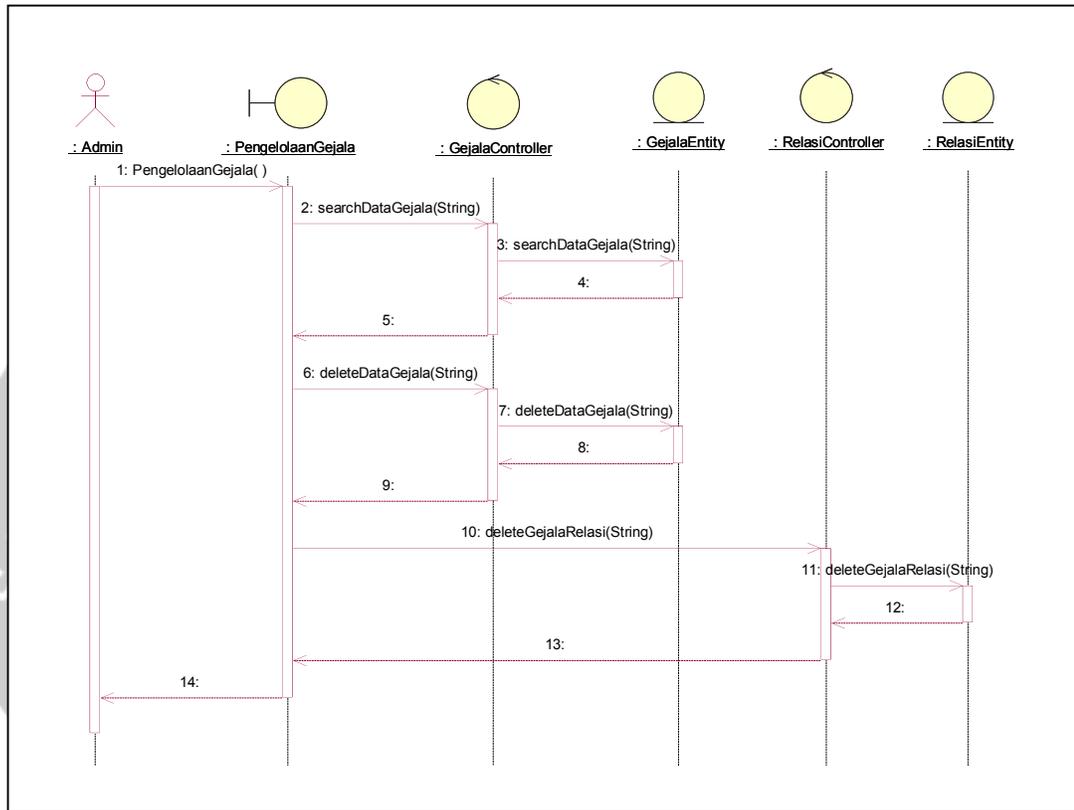
Gambar 2.7 Sequence Diagram - Add Gejala

### 2.2.1.3.2 Edit Gejala



Gambar 2.8 Sequence Diagram - Edit Gejala

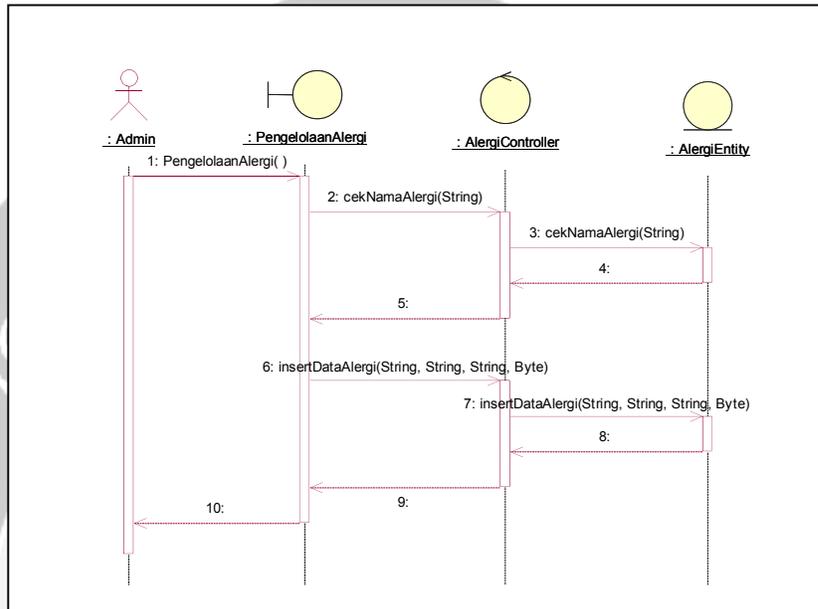
### 2.2.1.3.3 Delete Gejala



Gambar 2.9 Sequence Diagram - Delete Gejala

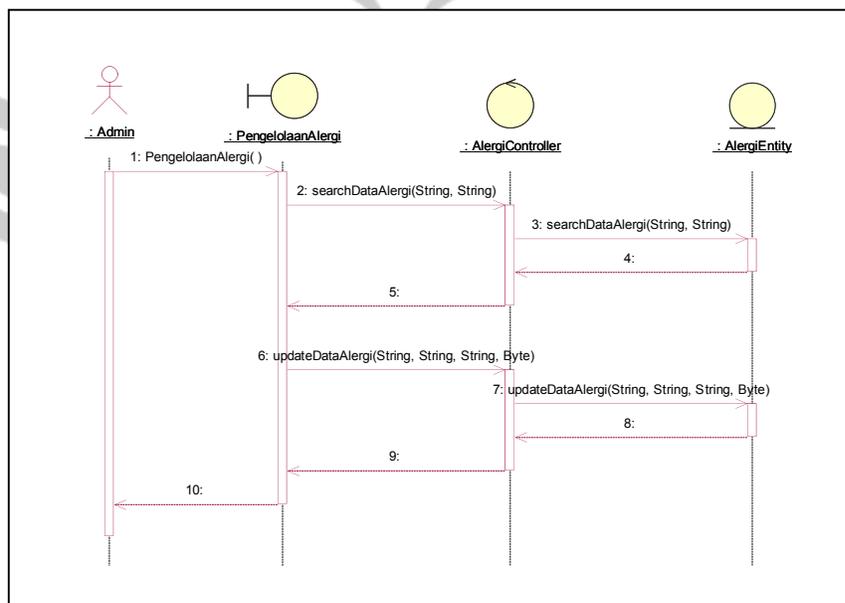
## 2.2.1.4 Pengelolaan Alergi

### 2.2.1.4.1 Add Alergi



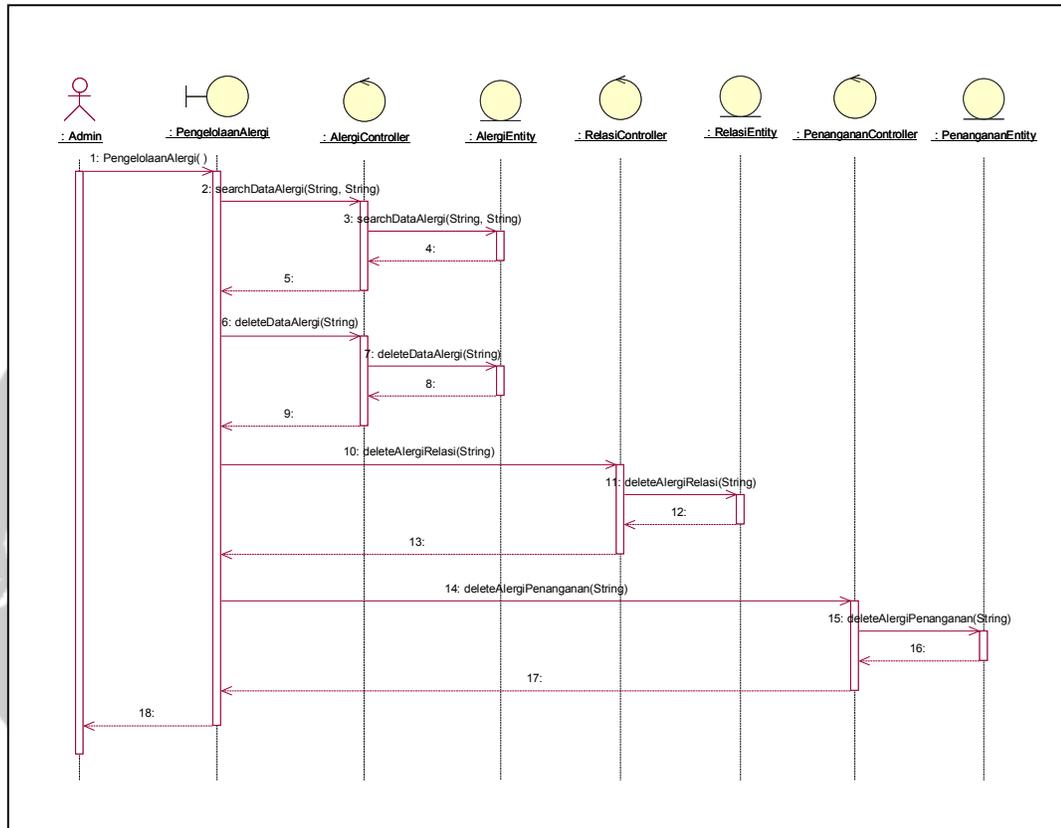
Gambar 2.10 Sequence Diagram - Add Alergi

### 2.2.1.4.2 Edit Alergi



Gambar 2.11 Sequence Diagram - Edit Alergi

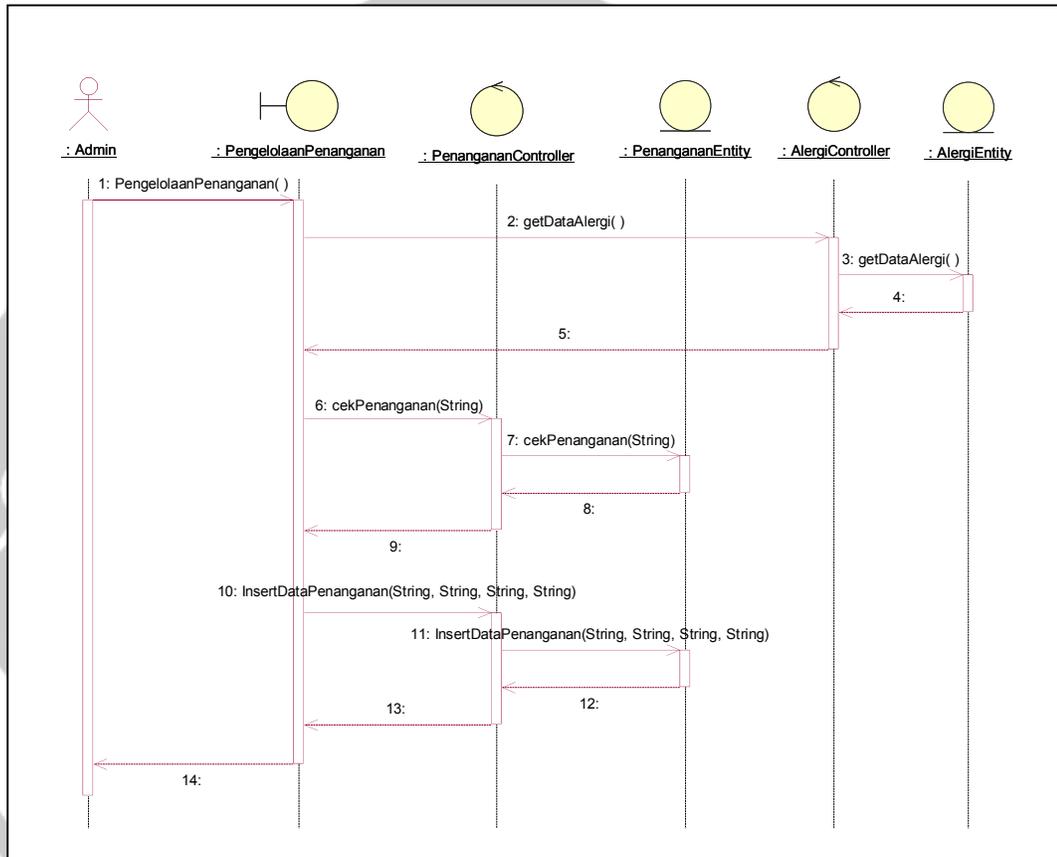
### 2.2.1.4.3 Delete Alergi



Gambar 2.12 Sequence Diagram - Delete Alergi

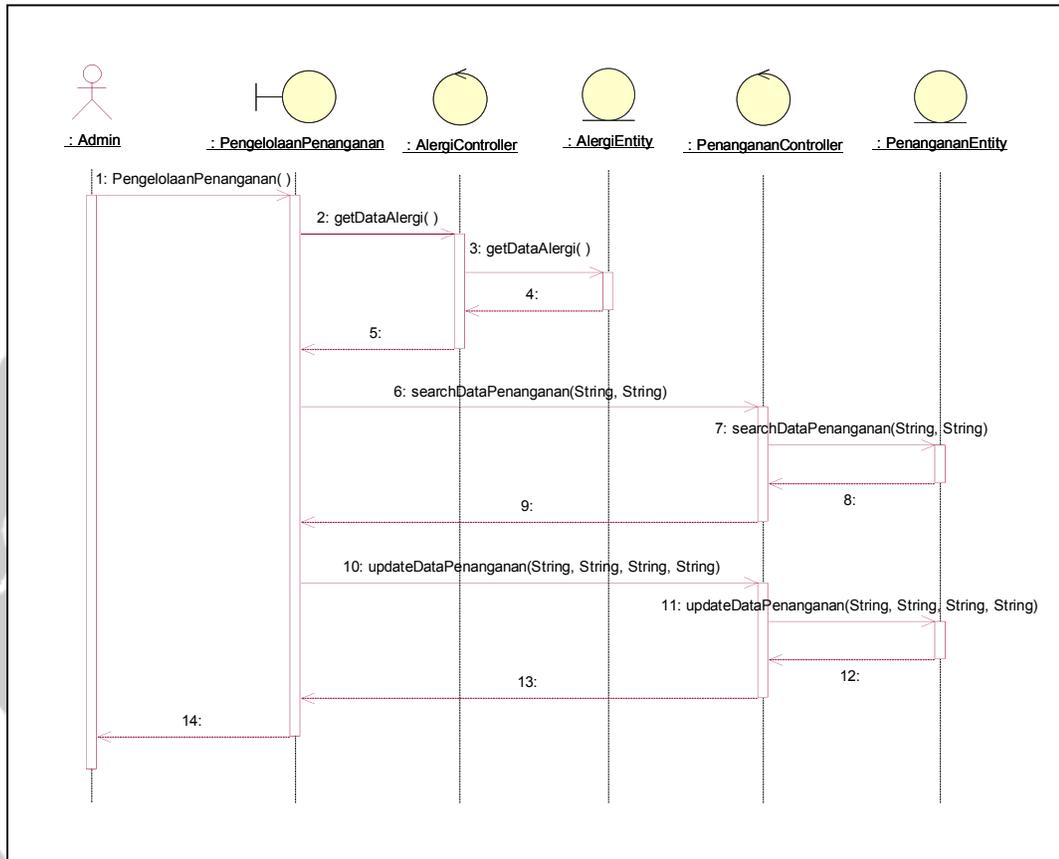
## 2.2.1.5 Pengelolaan Penanganan

### 2.2.1.5.1 Add Penanganan



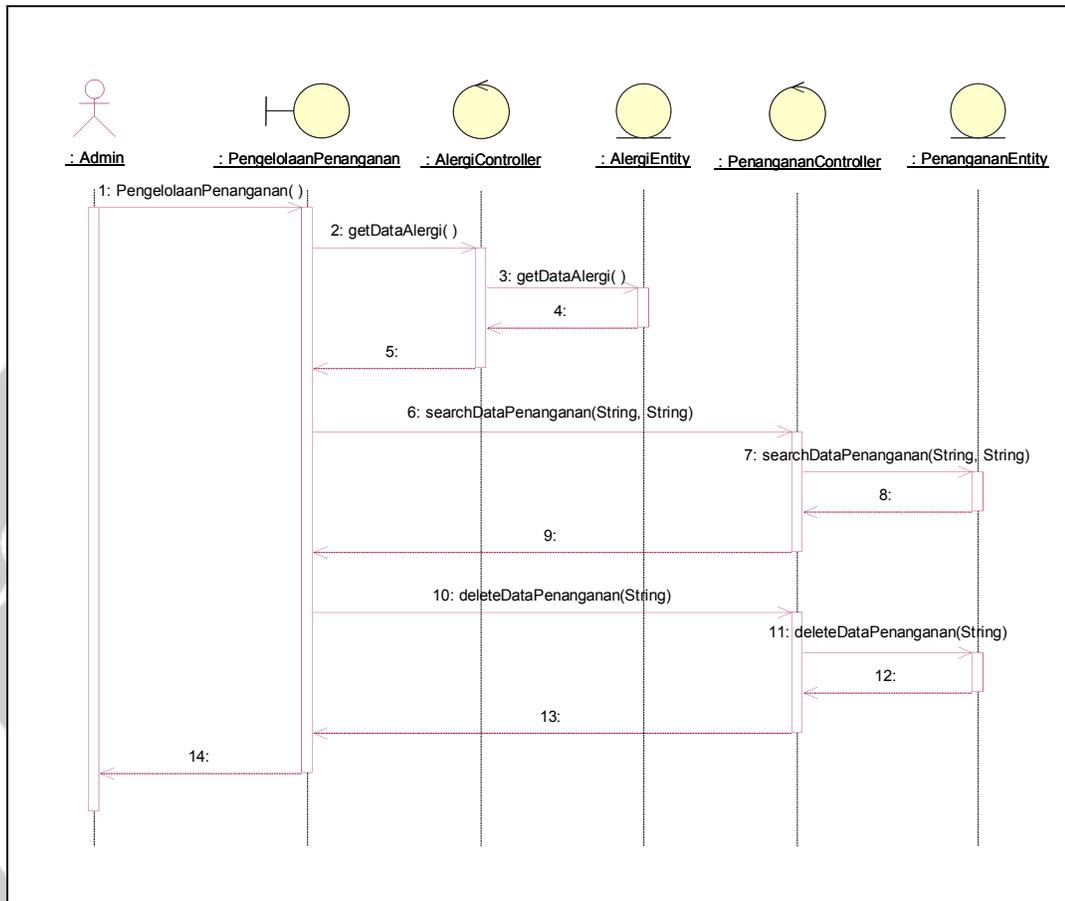
Gambar 2.13 Sequence Diagram - Add Penanganan

### 2.2.1.5.2 Edit Penanganan



Gambar 2.14 Sequence Diagram - Edit Penanganan

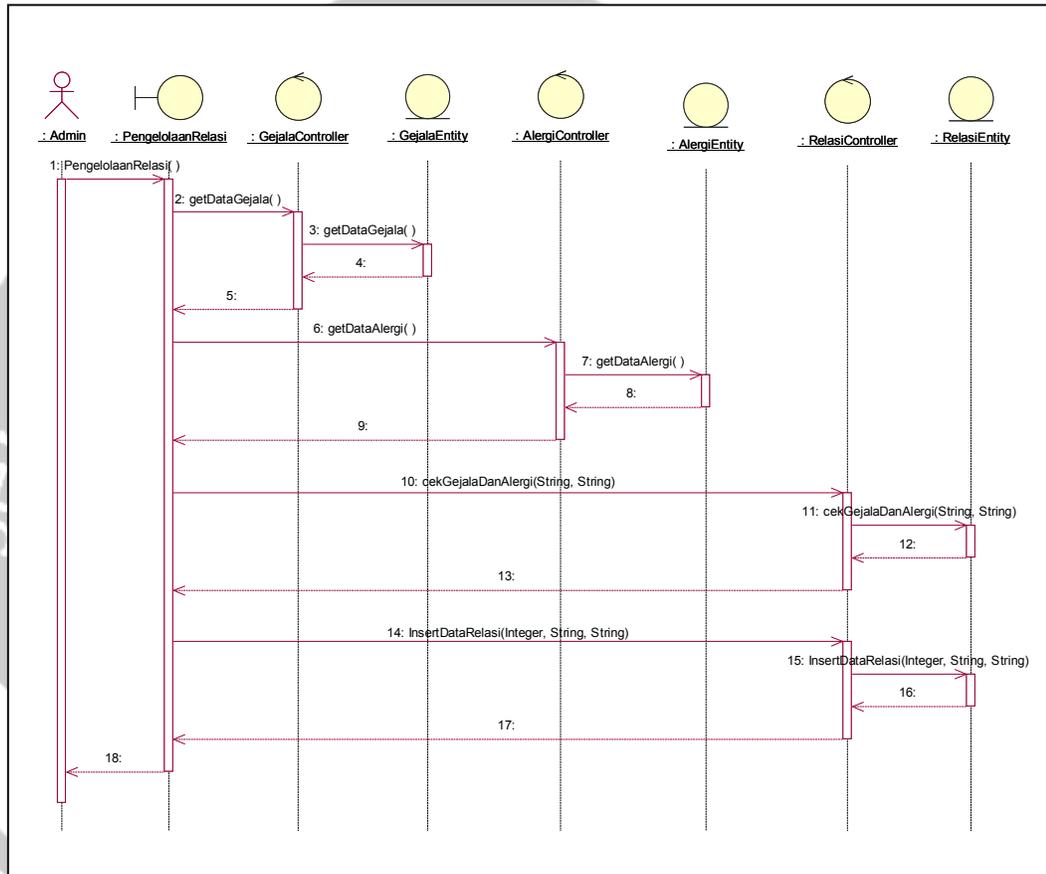
### 2.2.1.5.3 Delete Penanganan



Gambar 2.15 Sequence Diagram - Delete Penanganan

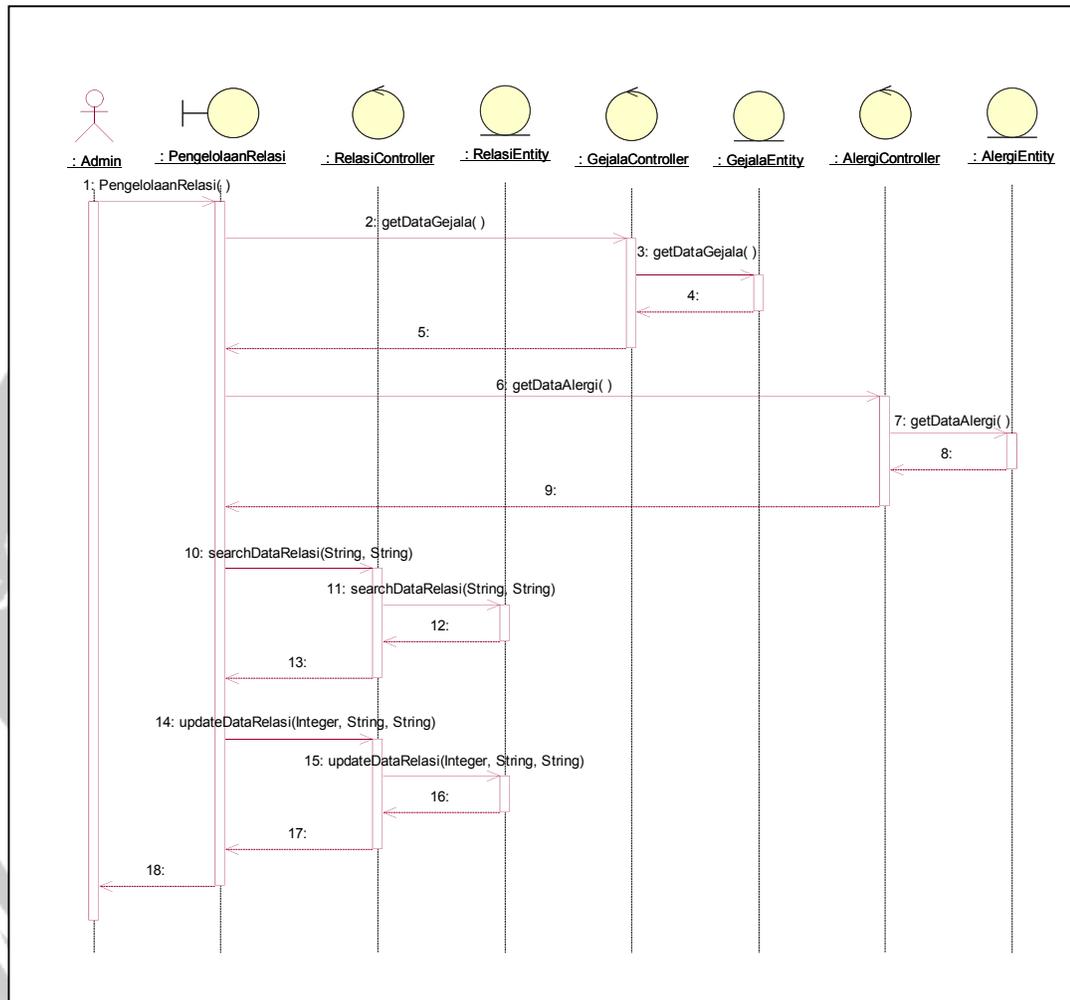
## 2.2.1.6 Pengelolaan Relasi

### 2.2.1.6.1 Add Relasi



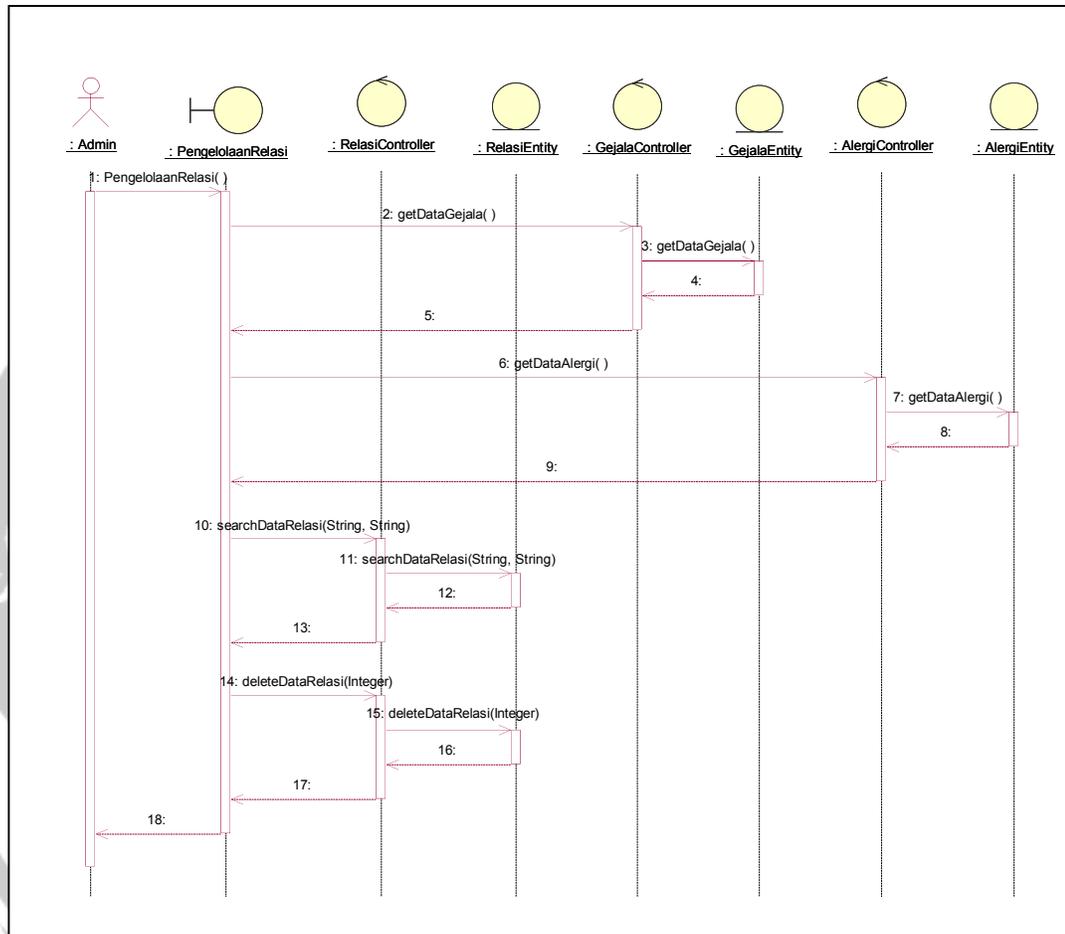
Gambar 2.16 Sequence Diagram - Add Relasi

### 2.2.1.6.2 Edit Relasi



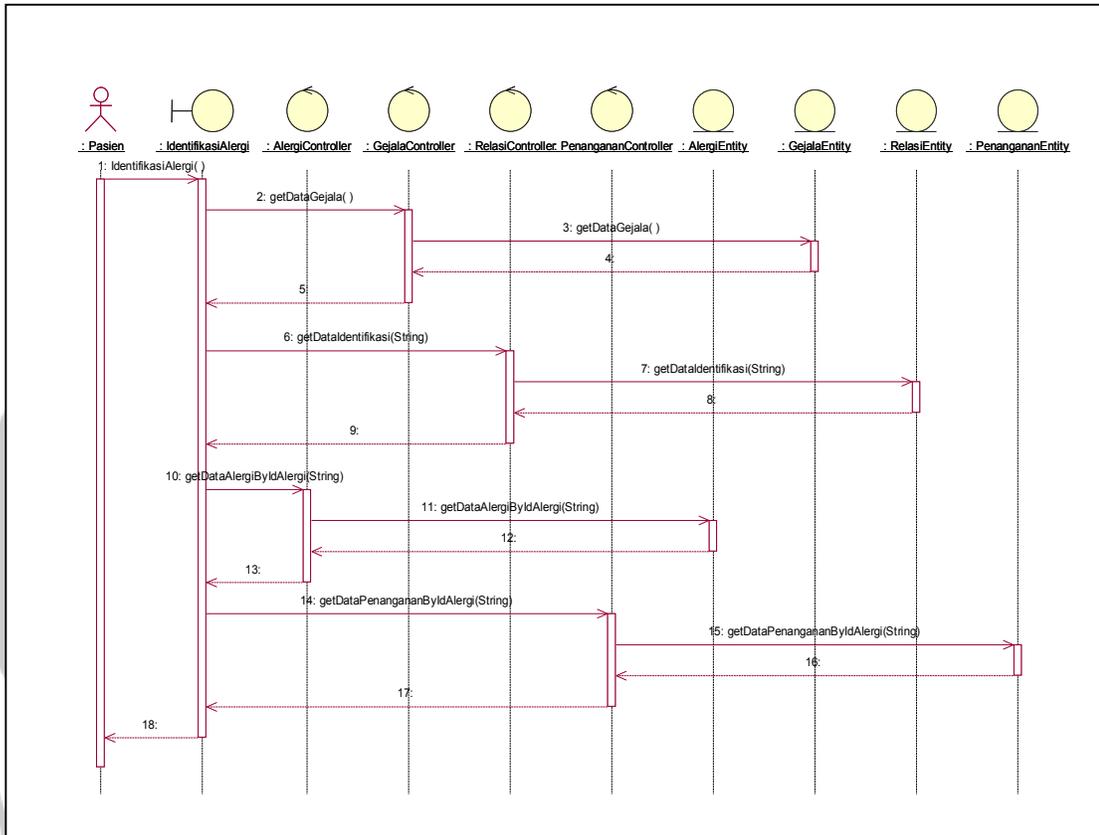
Gambar 2.17 Sequence Diagram - Edit Relasi

### 2.2.1.6.3 Delete Relasi



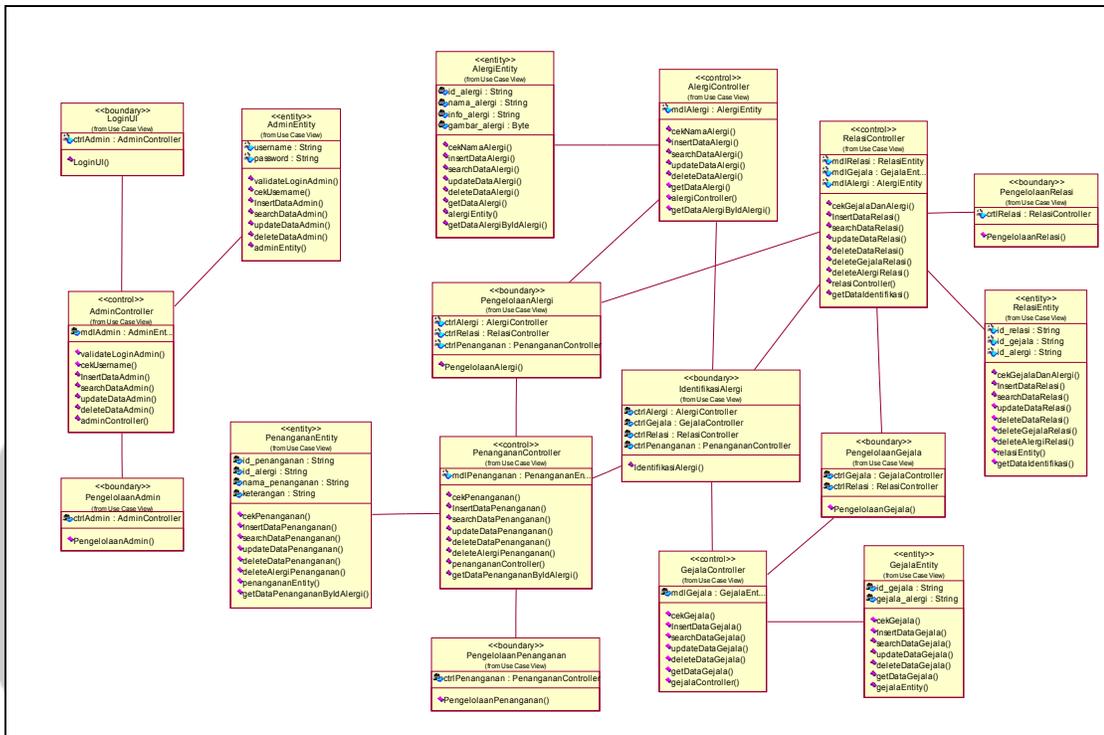
Gambar 2.18 Sequence Diagram - Delete Relasi

### 2.2.1.7 Identifikasi Alergi



Gambar 2.19 Sequence Diagram - Identifikasi Alergi

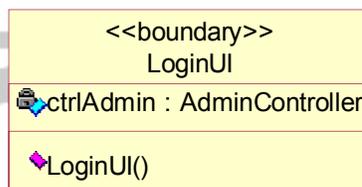
## 2.2.2 Class Diagram



Gambar 2.20 Class Diagram SiPaGi

## 2.2.3 Deskripsi Kelas

### 2.2.3.1 Specific Design Class LoginUI



#### Deskripsi

Class ini merepresentasikan class boundary/GUI untuk Login dalam SiPaGi.

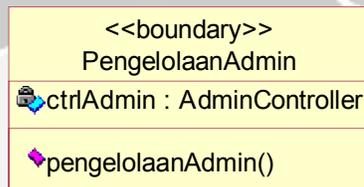
#### Atribut

- ctrlAdmin: AdminController  
Merepresentasikan objek control class dari Admin Controller.

### Method

- + LoginUI()  
Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

#### 2.2.3.2 Specific Design Class PengelolaanAdmin



### Deskripsi

Class ini merepresentasikan class boundary/GUI untuk melakukan pengelolaan admin dalam SiPaGi.

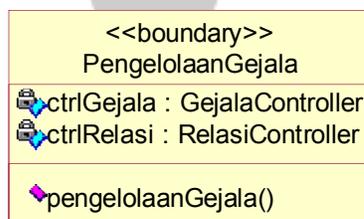
### Atribut

- - ctrlAdmin: AdminController  
Merepresentasikan objek control class dari Admin Controller.

### Method

- + PengelolaanAdmin()  
Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

#### 2.2.3.3 Specific Design Class PengelolaanGejala



### Deskripsi

Class ini merepresentasikan class boundary/GUI untuk melakukan pengelolaan gejala dalam SiPaGi.

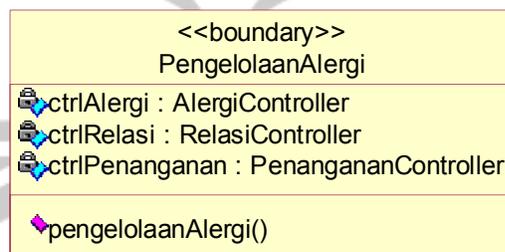
### Atribut

- - ctrlGejala: GejalaController  
Merepresentasikan objek control class dari Gejala Controller.
- - ctrlRelasi: RelasiController  
Merepresentasikan objek control class dari Relasi Controller.

### Method

- + PengelolaanGejala()  
Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

#### 2.2.3.4 Specific Design Class PengelolaanAlergi



### Deskripsi

Class ini merepresentasikan class boundary/GUI untuk melakukan pengelolaan alergi dalam SiPaGi.

### Atribut

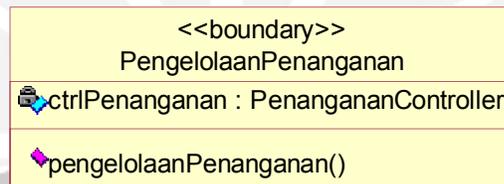
- - ctrlAlergi: AlergiController  
Merepresentasikan objek control class dari Alergi Controller.

- - ctrlRelasi: RelasiController  
Merepresentasikan objek control class dari Relasi Controller.
- - ctrlPenanganan: PenangananController  
Merepresentasikan objek control class dari Penanganan Controller.

**Method**

- + PengelolaanAlergi()  
Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

**2.2.3.5 Specific Design Class PengelolaanPenanganan**



**Deskripsi**

Class ini merepresentasikan class boundary/GUI untuk melakukan pengelolaan penanganan dalam SiPaGi.

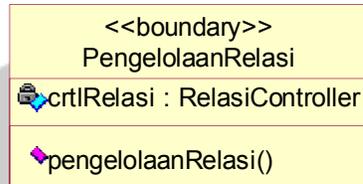
**Atribut**

- - ctrlPenanganan: PenangananController  
Merepresentasikan objek control class dari Penanganan Controller.

**Method**

- + PengelolaanPenanganan()  
Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

### 2.2.3.6 Specific Design Class PengelolaanRelasi



#### Deskripsi

Class ini merepresentasikan class boundary/GUI untuk melakukan pengelolaan relasi dalam SiPaGi.

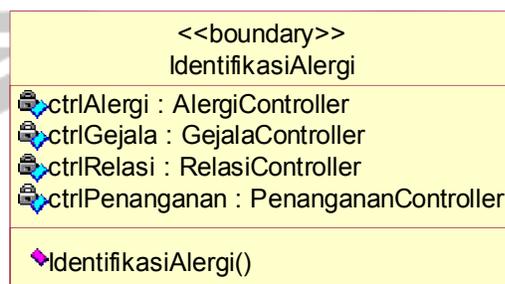
#### Atribut

- - ctrlRelasi: RelasiController  
Merepresentasikan objek control class dari Relasi Controller.

#### Method

- + PengelolaanRelasi()  
Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

### 2.2.3.7 Specific Design Class IdentifikasiAlergi



#### Deskripsi

Class ini merepresentasikan class boundary/GUI untuk melakukan identifikasi alergi dalam SiPaGi.

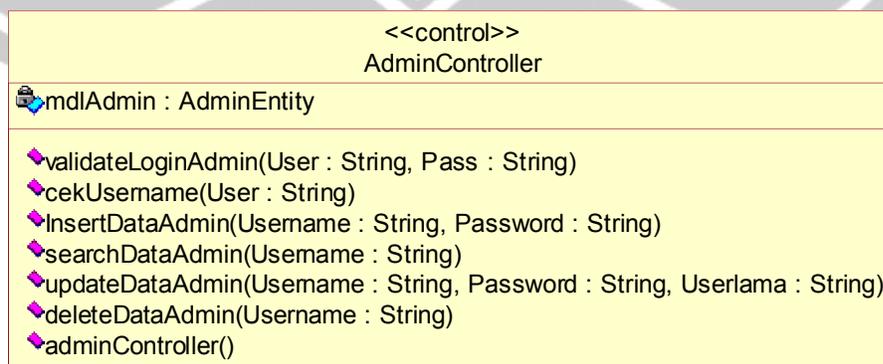
### Atribut

- - ctrlAlergi: AlergiController  
Merepresentasikan objek control class dari Alergi Controller.
- - ctrlGejala: GejalaController  
Merepresentasikan objek control class dari Gejala Controller.
- - ctrlRelasi: RelasiController  
Merepresentasikan objek control class dari Relasi Controller.
- - ctrlPenanganan: PenangananController  
Merepresentasikan objek control class dari Penanganan Controller.

### Method

- + IdentifikasiAlergi()  
Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

### 2.2.3.8 Specific Design Class AdminController



### **Deskripsi**

Class ini merepresentasikan control class untuk data Admin dalam SiPaGi.

### **Atribut**

- - mdlAdmin: AdminEntity  
Merepresentasikan objek entity class dari Admin.

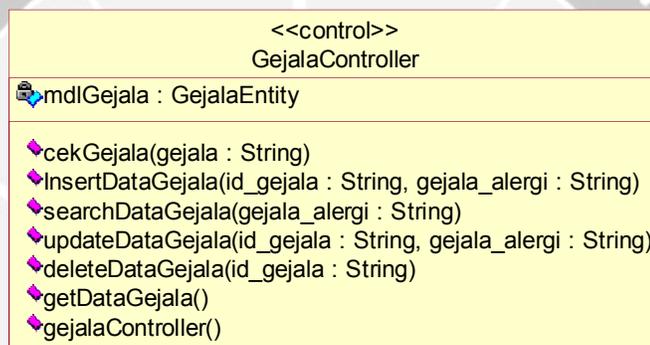
### **Method**

- + validateLoginAdmin(user: String, pass: String)  
Operasi ini digunakan untuk mengecek username dan password yang dimasukkan admin.
- + cekUsername(user: String)  
Operasi ini digunakan untuk mengecek username yang dimasukkan admin sudah ada atau belum dalam basis data.
- + InsertDataAdmin(username: String, password: String)  
Operasi ini digunakan untuk memasukkan data admin ke dalam basis data.
- + searchDataAdmin(username: String)  
Operasi ini digunakan untuk mencari data admin berdasarkan username dalam basis data.
- + updateDataAdmin(useraname:String, password:String, userlama:String)  
Operasi ini digunakan untuk mengubah data admin dari basis data.

Program Studi Teknik Informatika	DPPL – SiPaGi	31/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

- + deleteDataAdmin(username:String)  
Operasi ini digunakan untuk menghapus data admin dari basis data.
- + adminController()  
Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

### 2.2.3.9 Specific Design Class GejalaController



#### Deskripsi

Class ini merepresentasikan control class untuk data Gejala dalam SiPaGi.

#### Atribut

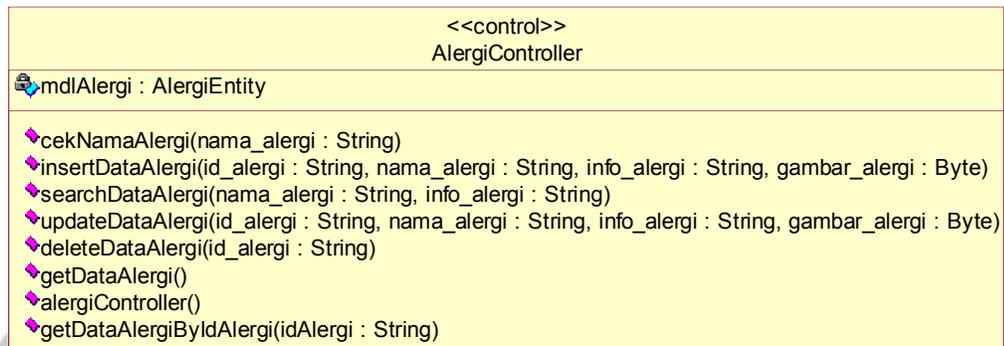
- - mdlGejala: GejalaEntity  
Merepresentasikan objek entity class dari Gejala.

#### Method

- + cekGejala(gejala: String)  
Operasi ini digunakan untuk mengecek gejala yang dimasukkan admin sudah ada atau belum dalam basis data.

- + InsertDataGejala(id\_gejala: String, gejala\_alergi: String)  
Operasi ini digunakan untuk memasukkan data gejala ke dalam basis data.
- + searchDataGejala(gejala\_alergi: String)  
Operasi ini digunakan untuk mencari data gejala berdasarkan nama gejala dalam basis data.
- + updateDataGejala(id\_gejala:String, gejala\_alergi:String)  
Operasi ini digunakan untuk mengubah data gejala dari basis data.
- + deleteDataGejala(id\_gejala:String)  
Operasi ini digunakan untuk menghapus data gejala dari basis data.
- + getDataGejala()  
Operasi ini digunakan untuk mengambil semua data gejala dari basis data.
- + gejalaController()  
Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

### 2.2.3.10 Specific Design Class AlergiController



#### Deskripsi

Class ini merepresentasikan control class untuk data Alergi dalam SiPaGi.

#### Atribut

- - mdlAlergi: AlergiEntity

Merepresentasikan objek entity class dari Alergi.

#### Method

- + cekNamaAlergi(nama\_alergi: String)  
Operasi ini digunakan untuk mengecek nama alergi yang dimasukkan admin sudah ada atau belum dalam basis data.
- + insertDataAlergi(id\_alergi: String, nama\_alergi: String, info\_alergi: String, gambar\_alergi: Byte)  
Operasi ini digunakan untuk memasukkan data alergi ke dalam basis data.
- + searchDataAlergi(nama\_alergi: String, info\_alergi: String)

Operasi ini digunakan untuk mencari data alergi berdasarkan nama alergi dan info alergi dalam basis data.

- + `updateDataAlergi(id_alergi: String, nama_alergi: String, info_alergi: String, gambar_alergi: Byte)`

Operasi ini digunakan untuk mengubah data alergi dari basis data.

- + `deleteDataAlergi(id_alergi:String)`

Operasi ini digunakan untuk menghapus data alergi dari basis data.

- + `getDataAlergi()`

Operasi ini digunakan untuk mengambil semua data alergi dari basis data.

- + `alergiController()`

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

- + `getDataAlergiByIdAlergi(IdAlergi: String)`

Operasi ini digunakan untuk mengambil data alergi berdasarkan id alergi dari basis data.

### 2.2.3.11 Specific Design Class PenangananController

<<control>> PenangananController
mdlPenanganan : PenangananEntity
cekPenanganan(nama_penanganan : String) insertDataPenanganan(id_penanganan : String, id_alergi : String, nama_penanganan : String, keterangan : String) searchDataPenanganan(nama_penanganan : String, keterangan : String) updateDataPenanganan(id_penanganan : String, id_alergi : String, nama_penanganan : String, keterangan : String) deleteDataPenanganan(id_penanganan : String) deleteAlergiPenanganan(id_alergi : String) penangananController() getDataPenangananByIdAlergi(idAlergi : String)

#### Deskripsi

Class ini merepresentasikan control class untuk data Penanganan dalam SiPaGi.

#### Atribut

- mdlPenanganan: PenangananEntity  
Merepresentasikan objek entity class dari Penanganan.

#### Method

- + cekPenanganan(nama\_penanganan: String)  
Operasi ini digunakan untuk mengecek nama penanganan yang dimasukkan admin sudah ada atau belum dalam basis data.
- + insertDataPenanganan(id\_penanganan: String, id\_alergi: String, nama\_penanganan: String, keterangan: String)  
Operasi ini digunakan untuk memasukkan data penanganan ke dalam basis data.
- + searchDataPenanganan(nama\_penanganan: String, keterangan: String)

Program Studi Teknik Informatika	DPPL – SiPaGi	36/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Operasi ini digunakan untuk mencari data penanganan berdasarkan nama penanganan dan keterangan penanganan dalam basis data.

- + `updateDataPenanganan(id_penanganan: String, id_alergi: String, nama_penanganan: String, keterangan: String)`

Operasi ini digunakan untuk mengubah data penanganan dari basis data.

- + `deleteDataPenanganan(id_penanganan: String)`

Operasi ini digunakan untuk menghapus data penanganan dari basis data.

- + `deleteAlergiPenanganan(id_alergi: String)`

Operasi ini digunakan untuk menghapus data penanganan berdasarkan id alergi dari basis data.

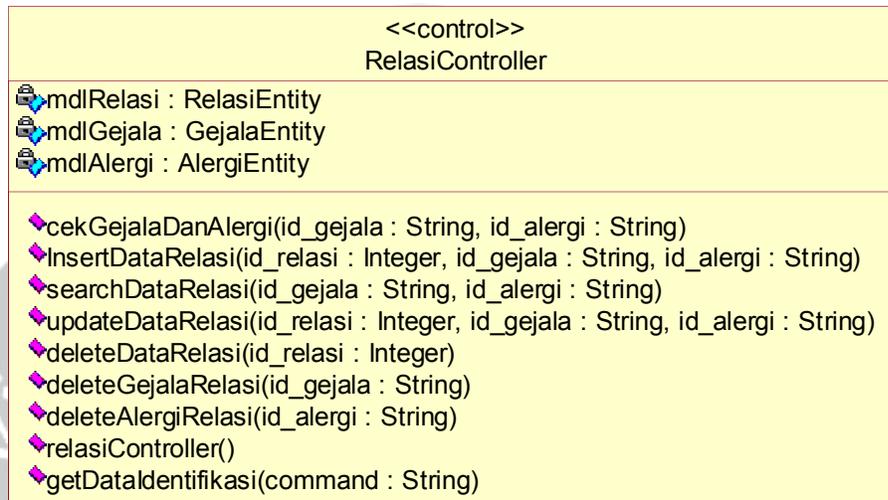
- + `penangananController()`

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

- + `getDataPenangananByIdAlergi(idAlergi: String)`

Operasi ini digunakan untuk mengambil data penanganan berdasarkan id alergi dari basis data.

### 2.2.3.12 Specific Design Class RelasiController



#### Deskripsi

Class ini merepresentasikan control class untuk data Relasi dalam SiPaGi.

#### Atribut

- - mdlRelasi: RelasiEntity  
Merepresentasikan objek entity class dari Relasi.
- - mdlGejala: GejalaEntity  
Merepresentasikan objek entity class dari Gejala.
- - mdlAlergi: AlergiEntity  
Merepresentasikan objek entity class dari Alergi.

#### Method

- + cekGejalaDanAlergi(id\_gejala: String, id\_alergi: String)

Operasi ini digunakan untuk mengecek id gejala dan id alergi yang dimasukkan admin sudah ada atau belum dalam basis data.

- + insertDataRelasi(id\_relasi: Integer, id\_gejala: String, id\_alergi: String)  
Operasi ini digunakan untuk memasukkan data relasi ke dalam basis data.
- + searchDataRelasi(id\_gejala: String, id\_alergi: String)  
Operasi ini digunakan untuk mencari data relasi berdasarkan id gejala dan id alergi dalam basis data.
- + updateDataRelasi(id\_relasi: Integer, id\_gejala: String, id\_alergi: String)  
Operasi ini digunakan untuk mengubah data relasi dari basis data.
- + deleteDataRelasi(id\_relasi: Integer)  
Operasi ini digunakan untuk menghapus data relasi dari basis data.
- + deleteGejalaRelasi(id\_gejala: String)  
Operasi ini digunakan untuk menghapus data relasi berdasarkan id gejala dari basis data.
- + deleteAlergiRelasi(id\_alergi: String)  
Operasi ini digunakan untuk menghapus data relasi berdasarkan id alergi dari basis data.
- + relasiController()

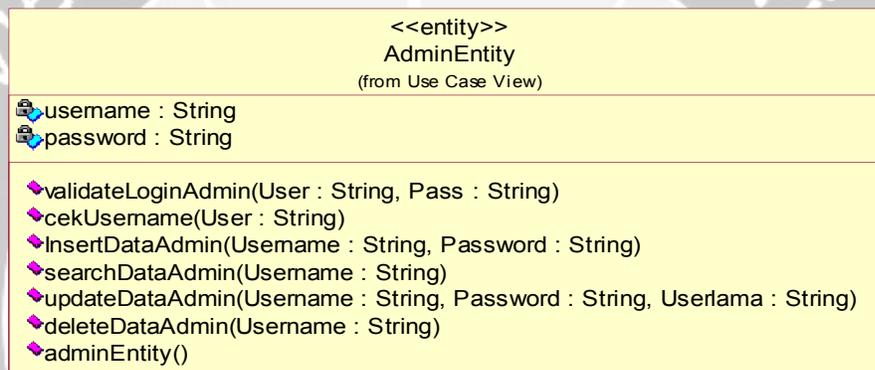
Program Studi Teknik Informatika	DPPL – SiPaGi	39/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

- + getDataIdentifikasi(command: String)

Operasi ini digunakan untuk mendapatkan data relasi antara gejala dan alergi dari basis data.

### 2.2.3.13 Specific Design Class AdminEntity



#### Deskripsi

Class ini merepresentasikan entity class untuk data Admin dalam SiPaGi.

#### Atribut

- - username: String  
Merepresentasikan username dari admin.
- - password: String  
Merepresentasikan password dari admin.

#### Method

- + validateLoginAdmin (user: String, pass: String)

Operasi ini digunakan untuk mengecek username dan password yang dimasukkan admin.

- + cekUsername(user: String)

Operasi ini digunakan untuk mengecek username yang dimasukkan admin sudah ada atau belum dalam basis data.

- + InsertDataAdmin(username: String, password: String)

Operasi ini digunakan untuk memasukkan data admin ke dalam basis data.

- + searchDataAdmin(username: String)

Operasi ini digunakan untuk mencari data admin berdasarkan username dalam basis data.

- + updateDataAdmin(useraname:String, password:String, userlama:String)

Operasi ini digunakan untuk mengubah data admin dari basis data.

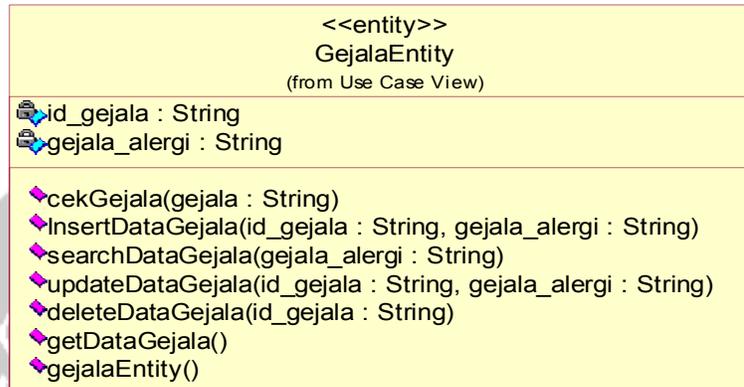
- + deleteDataAdmin(useraname:String)

Operasi ini digunakan untuk menghapus data admin dari basis data.

- + adminEntity()

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

#### 2.2.3.14 Specific Design Class GejalaEntity



##### Deskripsi

Class ini merepresentasikan entity class untuk data Gejala dalam SiPaGi.

##### Atribut

- - `id_gejala: String`  
Merepresentasikan id gejala.
- - `gejala_alergi: String`  
Merepresentasikan gejala alergi.

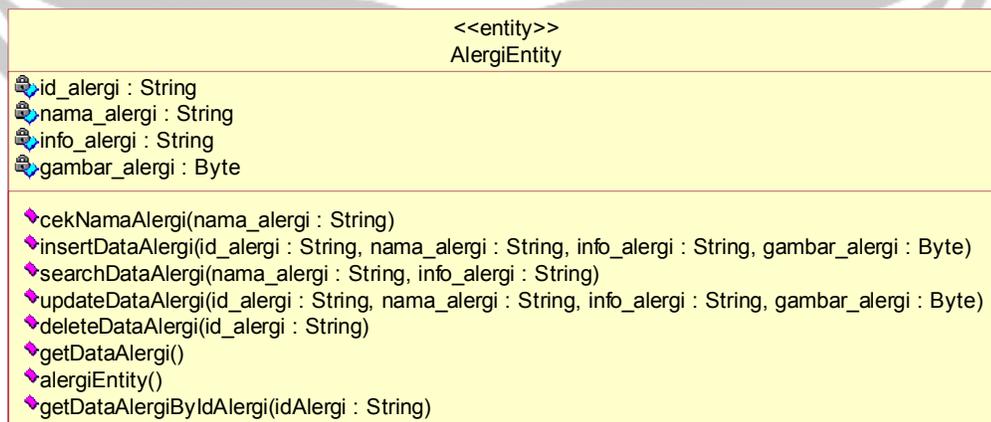
##### Method

- + `cekGejala(gejala: String)`  
Operasi ini digunakan untuk mengecek gejala yang dimasukkan admin sudah ada atau belum dalam basis data.
- + `insertDataGejala(id_gejala: String, gejala_alergi: String)`  
Operasi ini digunakan untuk memasukkan data gejala ke dalam basis data.
- + `searchDataGejala(gejala_alergi: String)`

Operasi ini digunakan untuk mencari data gejala berdasarkan nama gejala dalam basis data.

- + updateDataGejala(id\_gejala:String, gejala\_alergi:String)  
Operasi ini digunakan untuk mengubah data gejala dari basis data.
- + deleteDataGejala(id\_gejala:String)  
Operasi ini digunakan untuk menghapus data gejala dari basis data.
- + getDataGejala()  
Operasi ini digunakan untuk mengambil semua data gejala dari basis data.
- + gejalaEntity()  
Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

### 2.2.3.15 Specific Design Class AlergiEntity



## Deskripsi

Class ini merepresentasikan entity class untuk data Alergi dalam SiPaGi.

## Atribut

- - id\_alergi: String  
Merepresentasikan id alergi.
- - nama\_alergi: String  
Merepresentasikan nama alergi.
- - info\_alergi: String  
Merepresentasikan info alergi.
- - gambar\_alergi: Byte  
Merepresentasikan gambar alergi.

## Method

- + cekNamaAlergi(nama\_alergi: String)  
Operasi ini digunakan untuk mengecek nama alergi yang dimasukkan admin sudah ada atau belum dalam basis data.
- + insertDataAlergi(id\_alergi: String, nama\_alergi: String, info\_alergi: String, gambar\_alergi: Byte)  
Operasi ini digunakan untuk memasukkan data alergi ke dalam basis data.
- + searchDataAlergi(nama\_alergi: String, info\_alergi: String)  
Operasi ini digunakan untuk mencari data alergi berdasarkan nama alergi dan info alergi dalam basis data.

Program Studi Teknik Informatika	DPPL – SiPaGi	44/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

- + updateDataAlergi(id\_alergi: String, nama\_alergi: String, info\_alergi: String, gambar\_alergi: Byte)  
Operasi ini digunakan untuk mengubah data alergi dari basis data.
- + deleteDataAlergi(id\_alergi:String)  
Operasi ini digunakan untuk menghapus data alergi dari basis data.
- + getDataAlergi()  
Operasi ini digunakan untuk mengambil semua data alergi dari basis data.
- + alergiEntity()  
Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.
- + getDataAlergiByIdAlergi(idAlergi: String)  
Operasi ini digunakan untuk menampilkan data alergi berdasarkan id alergi dari basis data.

### 2.2.3.16 Specific Design Class PenangananEntity



## Deskripsi

Class ini merepresentasikan entity class untuk data Penanganan dalam SiPaGi.

## Atribut

- - id\_penanganan: String  
Merepresentasikan id dari penanganan.
- - id\_alergi: String  
Merepresentasikan id dari alergi.
- - nama\_penanganan: String  
Merepresentasikan nama dari penanganan.
- - keterangan: String  
Merepresentasikan keterangan dari penanganan.

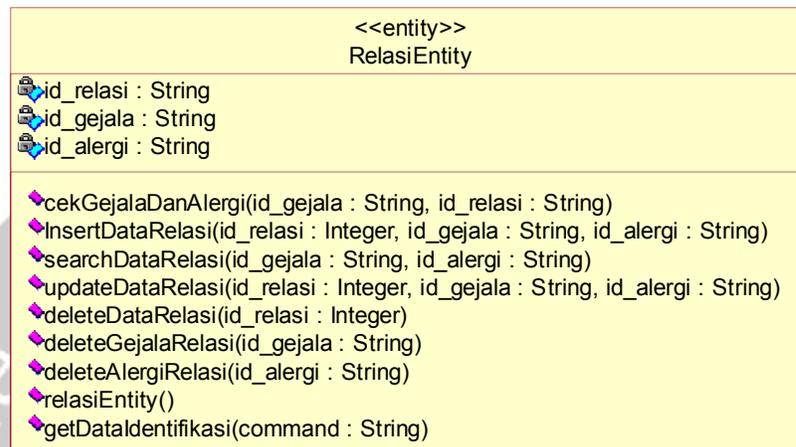
## Method

- + cekPenanganan(nama\_penanganan: String)  
Operasi ini digunakan untuk mengecek nama penanganan yang dimasukkan admin sudah ada atau belum dalam basis data.
- + insertDataPenanganan(id\_penanganan: String, id\_alergi: String, nama\_penanganan: String, keterangan: String)  
Operasi ini digunakan untuk memasukkan data penanganan ke dalam basis data.
- + searchDataPenanganan(nama\_penanganan: String, keterangan: String)  
Operasi ini digunakan untuk mencari data penanganan berdasarkan nama penanganan dan keterangan penanganan dalam basis data.

Program Studi Teknik Informatika	DPPL – SiPaGi	46/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

- + updateDataPenanganan(id\_penanganan: String, id\_alergi: String, nama\_penanganan: String, keterangan: String)  
Operasi ini digunakan untuk mengubah data penanganan dari basis data.
- + deleteDataPenanganan(id\_penanganan: String)  
Operasi ini digunakan untuk menghapus data penanganan dari basis data.
- + deleteAlergiPenanganan(id\_alergi: String)  
Operasi ini digunakan untuk menghapus data penanganan berdasarkan id alergi dari basis data.
- + penangananEntity()  
Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.
- + getDataPenangananByIdAlergi(idAlergi: String)  
Operasi ini digunakan untuk menampilkan data penanganan berdasarkan id alergi dari basis data.

### 2.2.3.17 Specific Design Class



#### Deskripsi

Class ini merepresentasikan entity class untuk data Relasi dalam SiPaGi.

#### Atribut

- - id\_relasi: Integer  
Merepresentasikan id dari Relasi.
- - id\_gejala: String  
Merepresentasikan di dari Gejala.
- - id\_alergi: String  
Merepresentasikan id dari Alergi.

#### Method

- + cekGejalaDanAlergi(id\_gejala: String, id\_alergi: String)  
Operasi ini digunakan untuk mengecek id gejala dan id alergi yang dimasukkan admin sudah ada atau belum dalam basis data.
- + insertDataRelasi(id\_relasi: Integer, id\_gejala: String, id\_alergi: String)

Operasi ini digunakan untuk memasukkan data relasi ke dalam basis data.

- + searchDataRelasi(id\_gejala: String, id\_alergi: String)

Operasi ini digunakan untuk mencari data relasi berdasarkan id gejala dan id alergi dalam basis data.

- + updateDataRelasi(id\_relasi: Integer, id\_gejala: String, id\_alergi: String)

Operasi ini digunakan untuk mengubah data relasi dari basis data.

- + deleteDataRelasi(id\_relasi: Integer)

Operasi ini digunakan untuk menghapus data relasi dari basis data.

- + deleteGejalaRelasi(id\_gejala: String)

Operasi ini digunakan untuk menghapus data relasi berdasarkan id gejala dari basis data.

- + deleteAlergiRelasi(id\_alergi: String)

Operasi ini digunakan untuk menghapus data relasi berdasarkan id alergi dari basis data.

- + relasiEntity()

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

- + getDataIdentifikasi(command: String)

Operasi ini digunakan untuk mendapatkan data relasi antara gejala dan jenis alerginya.

Program Studi Teknik Informatika	DPPL – SiPaGi	49/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

### 3. Perancangan Data

#### 3.1 Dekomposisi Data

##### 3.1.1 Deskripsi Entitas Admin

Nama	Tipe	Panjang	Keterangan
username	Variabel Character	10	Username dari admin, Primary Key
password	Variabel Character	50	Password dari admin

##### 3.1.2 Deskripsi Entitas Gejala

Nama	Tipe	Panjang	Keterangan
id_gejala	Variabel Character	10	id gejala, Primary Key
gejala_alergi	Text	-	Gejala dari alergi

##### 3.1.3 Deskripsi Entitas Alergi

Nama	Tipe	Panjang	Keterangan
id_alergi	Variabel Character	10	Id dari alergi, Primary Key
nama_alergi	Variabel Character	50	Nama dari alergi
info_alergi	Text	-	keterangan dari alergi
gambar_alergi	Byte	-	Gambar dari alergi

##### 3.1.4 Deskripsi Entitas Penanganan

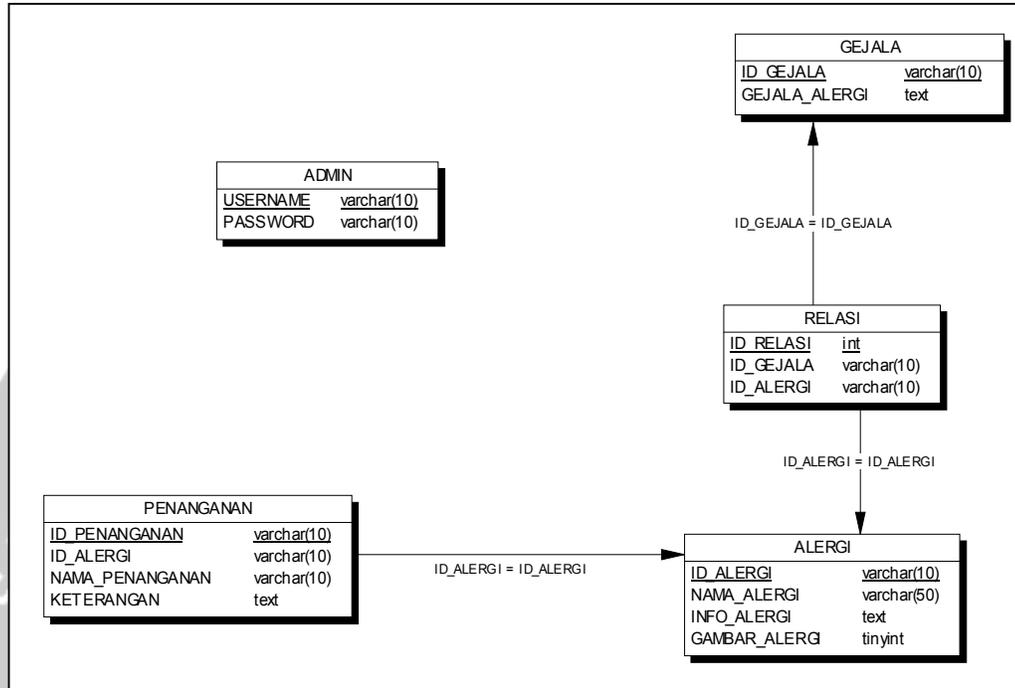
Nama	Tipe	Panjang	Keterangan
id_penanganan	Variabel Character	10	Id dari penanganan, Primary Key

<b>id_alergi</b>	Variabel Character	10	id dari alergi, Foreign Key ke tabel alergi
nama_penanganan	Variabel Character	10	nama dari penanganan
keterangan	Text	-	keterangan dari penanganan

### 3.1.5 Deskripsi Entitas Relasi

Nama	Tipe	Panjang	Keterangan
<b>id_relasi</b>	Integer	-	Id dari relasi, Primary Key
<b>id_gejala</b>	Variabel Character	10	id dari gejala, Foreign Key ke tabel gejala
<b>id_alergi</b>	Variabel Character	10	id dari alergi, Foreign Key ke tabel alergi

### 3.2 Physical Data Model



Gambar 3.1 Physical Data Model

## 4. Deskripsi Perancangan Antarmuka

### 4.1 Menu Utama



Gambar 4.1 Rancangan Antarmuka Menu Utama

## Deskripsi

- Halaman ini digunakan untuk melakukan login bagi Admin untuk masuk ke dalam sistem dan melakukan identifikasi alergi.
- Terdapat link label identifikasi alergi dan link label admin. Link label identifikasi alergi digunakan untuk menampilkan form identifikasi alergi. Link label Admin digunakan untuk menampilkan Grup box login yang berisi tombol Login yang digunakan untuk masuk ke dalam sistem dan tombol Cancel yang digunakan untuk membersihkan textboxUsername dan textboxPassword.

## Event

```
OnClick_linkIdentifikasiAlergi()
```

```
{
```

```
    Memunculkan form Identifikasi alergi.
```

```
}
```

```
OnClick_linkAdmin()
```

```
{
```

```
    Memunculkan group box login.
```

```
}
```

- Login

```
OnClick_buttonLogin()
```

```
{
```

```
    Sistem memeriksa di basis data, apakah data Admin yang diinputkan ada di dalam basis data dengan SQL Statement sebagai berikut:
```

Program Studi Teknik Informatika	DPPL – SiPaGi	53/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

SELECT USERNAME FROM ADMIN WHERE USERNAME=
' [TEXTBOXUSERNAME]' AND PASSWORD=
' [TEXTBOXPASSWORD]';

```

Sistem membandingkan username dan password yang diperoleh dengan password input user. Jika password sesuai, user masuk ke sistem, jika tidak maka sistem menampilkan pesan kesalahan.

```

}
OnClick_buttonCancel()
{
    Membersihkan      TextboxUsername      dan
    TextboxPassword
}

```

## 4.2 Menu Pengelolaan



Gambar 4.2 Rancangan Antarmuka Menu Pengelolaan

### Deskripsi

Program Studi Teknik Informatika	DPPL – SiPaGi	54/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

- Halaman ini digunakan untuk memilih pengelolaan yang akan dilakukan oleh admin.
- Terdapat link label Pengelolaan Admin, Pengelolaan Gejala, Pengelolaan Alergi, Pengelolaan Relasi dan Pengelolaan Penanganan.

#### **Event**

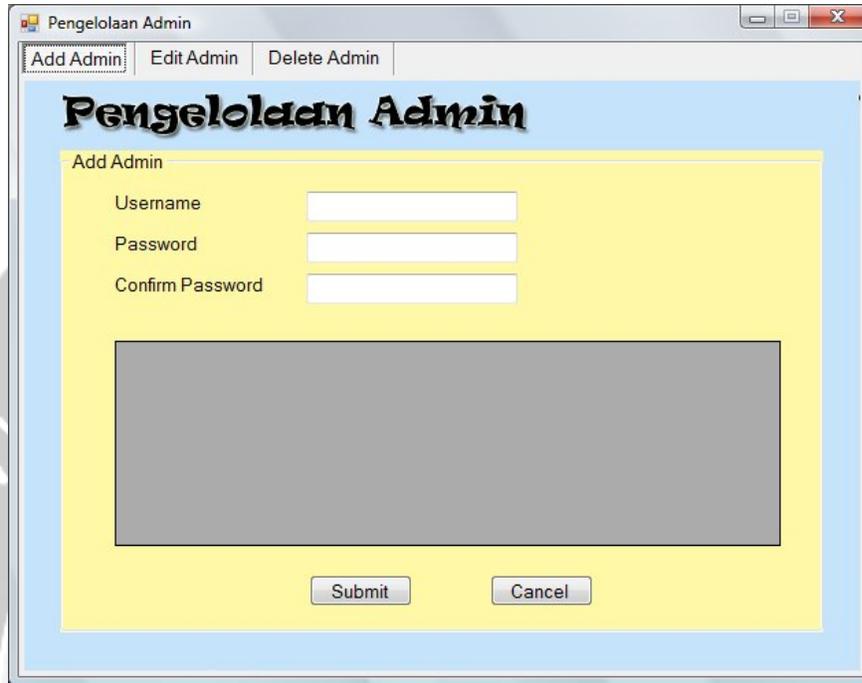
```

OnClick_linkPengelolaanAdmin()
{
    Memunculkan form Pengelolaan Admin untuk
    melakukan pengelolaan data admin.
}
OnClick_linkPengelolaanGejala()
{
    Memunculkan form Pengelolaan Gejala untuk
    melakukan pengelolaan data gejala alergi.
}
OnClick_linkPengelolaanAlergi()
{
    Memunculkan form Pengelolaan Alergi untuk
    melakukan pengelolaan data alergi.
}
OnClick_linkPengelolaanRelasi()
{
    Memunculkan form Pengelolaan Relasi untuk
    melakukan pengelolaan data relasi antara
    gejala dan jenis alergi.
}
OnClick_linkPengelolaanPenanganan()
{
    Memunculkan form Pengelolaan Penanganan
    untuk melakukan pengelolaan data penanganan
    jenis alergi.
}

```

Program Studi Teknik Informatika	DPPL – SiPaGi	55/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

### 4.3 Pengelolaan Admin

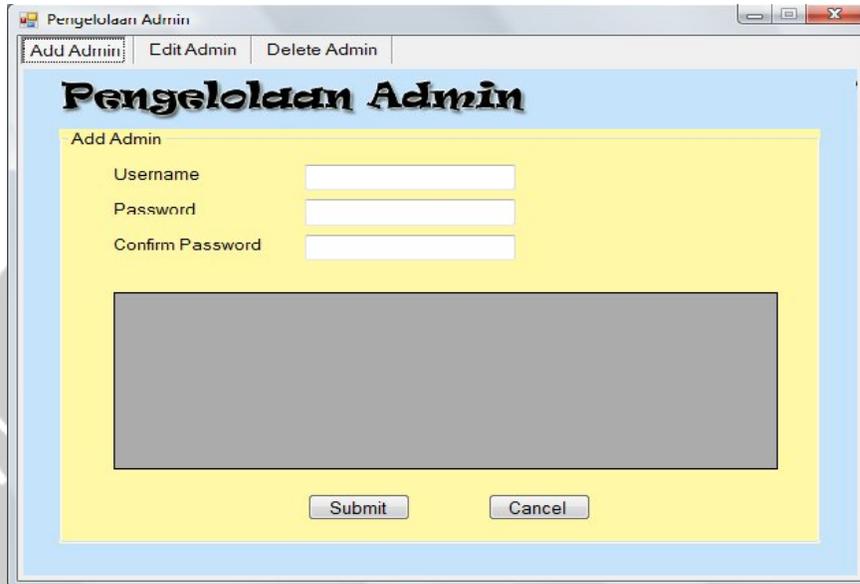


**Gambar 4.3 Rancangan Antarmuka Pengelolaan Admin**

#### **Deskripsi**

- Halaman ini digunakan oleh admin untuk melakukan pengelolaan data admin.
- Terdapat tiga tab control yang dapat dilakukan untuk melakukan pengelolan admin. Add Admin digunakan untuk menambahkan data admin baru, Edit Admin digunakan untuk mengubah data admin, Delete Admin digunakan untuk menghapus data admin.

### 4.3.1 Add Admin



Gambar 4.4 Rancangan Antarmuka Add Admin

#### Deskripsi

- Halaman ini digunakan oleh admin untuk menambahkan data admin baru ke dalam basis data.
- Terdapat dua buah tombol, yaitu Submit yang digunakan untuk memasukkan data ke dalam basis data dan tombol Cancel untuk membatalkan masukkan data.

#### Event

```
Page_Load()  
{  
    Mengisi dataGridViewAddAdmin dengan username  
    dan password dari semua data admin dengan  
    SQL Statement sebagai berikut:  
    SELECT * FROM ADMIN;  
}
```

Program Studi Teknik Informatika	DPPL – SiPaGi	57/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

OnClick_ buttonSubmitAddAdmin()
{
    Memasukkan data admin di
    dataGridViewAddAdmin dengan SQL Statement
    sebagai berikut:

```

```

INSERT INTO ADMIN VALUES
( '[TEXTBOXUSERNAMEADDADMIN] ',
' [TEXTBOXPASSADDADMIN] ' );

```

```

OnClick_ buttonCancelAddAdmin()
{
    Membersihkan textBoxPassAddAdmin,
    textBoxUsernameAddAdmin,
    textBoxConfPassAddAdmin.
}

```

#### 4.3.2 Edit Admin

**Gambar 4.5 Rancangan Antarmuka Edit Admin**

## Deskripsi

- Halaman ini digunakan oleh admin untuk mengubah data admin yang ada dalam basis data.
- Terdapat dua buah tombol, yaitu Submit yang digunakan untuk memasukkan data admin yang telah diubah ke dalam basis data dan tombol Cancel untuk membatalkan memasukkan data. Terdapat dua buah check box, yaitu check box search by yang digunakan untuk memilih kategori pencarian dan check box username yang digunakan untuk melakukan pencarian berdasarkan username admin.

## Event

```
Page_Load()  
{  
    Mengisi dataGridViewEditAdmin dengan  
    username dan password dari semua data admin  
    dengan SQL Statement sebagai berikut:  
    SELECT * FROM ADMIN;  
}  
OnClick_buttonSubmitEditAdmin()  
{  
    Jika checkBoxCategoryEditAdmin status cek  
    sama dengan true maka:  
    SELECT * FROM ADMIN WHERE USERNAME LIKE  
    '%[TEXTBOXUSERNAMEEDITADMIN]%' ;  
    Namun jika tidak maka akan memasukkan data  
    admin yang telah diubah di
```

Program Studi Teknik Informatika	DPPL – SiPaGi	59/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

dataGridViewEditAdmin dengan SQL Statement sebagai berikut:

```
UPDATE ADMIN SET USERNAME=  
' [TEXTBOXUSERNAMEEDITADMIN] ', PASSWORD=  
' [TEXTBOXPASSEEDITADMIN] ' ;
```

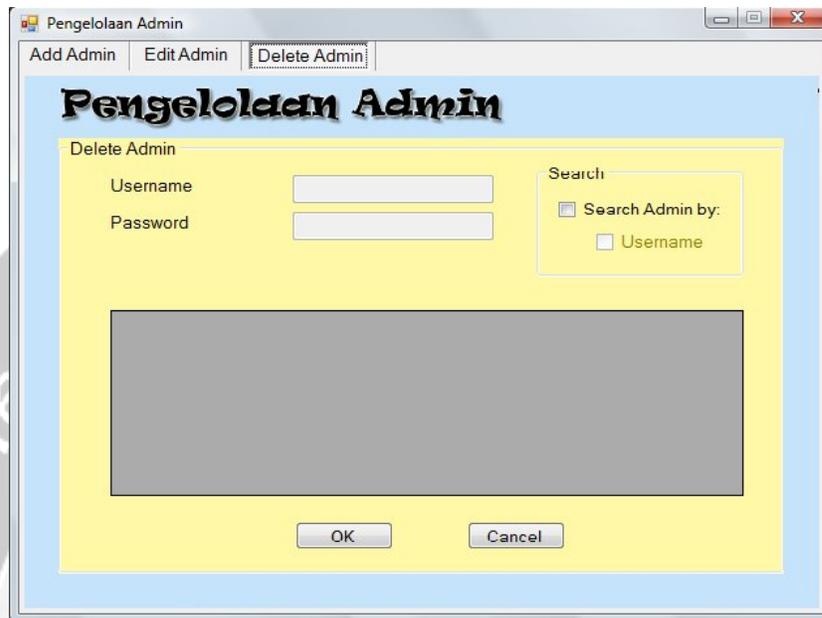
```
}  
OnClick_buttonCancelEditAdmin()  
{  
    Membersihkan textBoxPassEditAdmin,  
    textBoxUsernameEditAdmin,  
    textBoxConfPassEditAdmin.  
}
```

- Search

```
OnClick_checkBoxCategoryEditAdmin()  
{  
    Mengaktifkan  
    checkBoxSearchUsernameEditAdmin.  
}
```

```
OnClick_checkBoxSearchUsernameEditAdmin()  
{  
    Mengaktifkan textBoxUsernameEditAdmin.  
}
```

### 4.3.3 Delete Admin



The screenshot shows a web browser window titled 'Pengelolaan Admin'. At the top, there are three tabs: 'Add Admin', 'Edit Admin', and 'Delete Admin'. The 'Delete Admin' tab is active. The main content area has a yellow background and is titled 'Pengelolaan Admin' in a large, bold, black font. Below the title, there is a section titled 'Delete Admin'. This section contains two input fields for 'Username' and 'Password'. To the right of these fields is a 'Search' section with a checkbox labeled 'Search Admin by:' and a sub-section with a checkbox labeled 'Username'. Below the input fields and search section is a large, empty rectangular area. At the bottom of the form, there are two buttons: 'OK' and 'Cancel'.

Gambar 4.6 Rancangan Antarmuka Delete Admin

#### Deskripsi

- Halaman ini digunakan oleh admin untuk menghapus data admin yang ada dalam basis data.
- Terdapat dua buah tombol, yaitu Submit yang digunakan untuk memasukkan data admin yang akan dihapus dari dalam basis data dan tombol Cancel untuk membatalkan penghapusan data. Terdapat dua buah check box, yaitu check box search by yang digunakan untuk memilih kategori pencarian dan check box username yang digunakan untuk melakukan pencarian berdasarkan username admin.

#### Event

```
Page_Load()  
{
```

Program Studi Teknik Informatika	DPPL – SiPaGi	61/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Mengisi dataGridViewDeleteAdmin dengan username dan password dari semua data admin dengan SQL Statement sebagai berikut:

```
SELECT * FROM ADMIN;
```

```
}  
OnClick_buttonSubmitDeleteAdmin()  
{  
    Jika checkBoxCategoryDeleteAdmin status cek  
    sama dengan true maka:
```

```
SELECT * FROM ADMIN WHERE USERNAME LIKE  
'[TEXTBOXUSERNAMEDELETETADMIN]';
```

Namun jika tidak maka akan memasukkan data admin yang akan dihapus dari dataGridViewDeleteAdmin dengan SQL Statement sebagai berikut:

```
DELETE ADMIN WHERE USERNAME=  
'[TEXTBOXUSERNAMEDELETEADMIN]';
```

```
}  
OnClick_buttonCancelDeleteAdmin()  
{  
    Membersihkan textBoxPassEditAdmin,  
    textBoxUsernameEditAdmin,  
    textBoxConfPassEditAdmin.  
}
```

- Search

```
OnClick_checkBoxCategoryDeleteAdmin()  
{  
    Mengaktifkan  
    checkBoxSearchUsernameDeleteAdmin.  
}
```

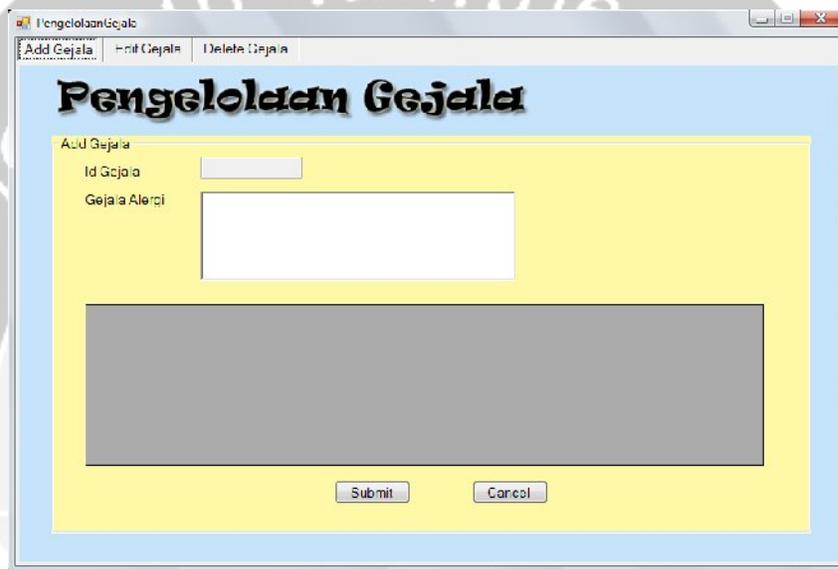
Program Studi Teknik Informatika	DPPL – SiPaGi	62/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

OnClick_checkBoxSearchUsernameDeleteAdmin()
{
    Mengaktifkan textBoxUsernameEditAdmin.
}

```

#### 4.4 Pengelolaan Gejala

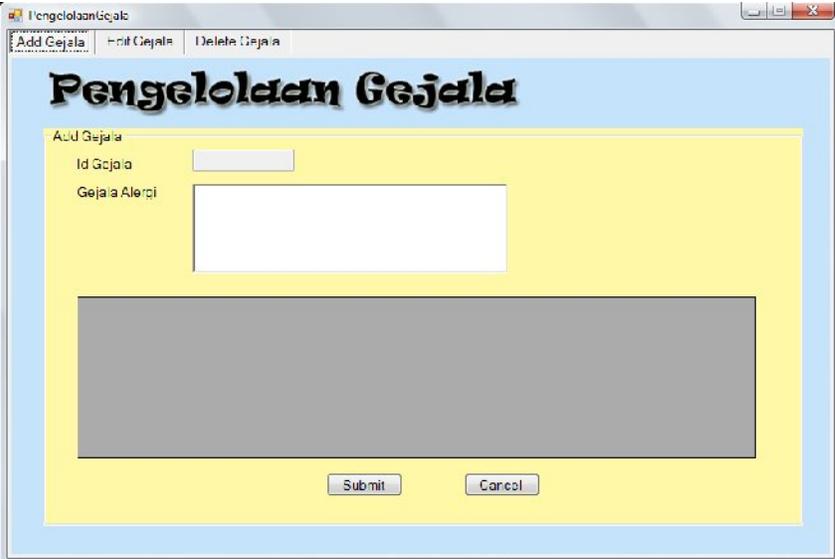


Gambar 4.7 Rancangan Antarmuka Pengelolaan Gejala

##### Deskripsi

- Halaman ini digunakan oleh admin untuk melakukan pengelolaan data gejala alergi.
- Terdapat tiga tab control yang dapat dilakukan untuk melakukan pengelolaan gejala. Add Gejala digunakan untuk menambahkan data gejala baru, Edit Gejala digunakan untuk mengubah data gejala, Delete Gejala digunakan untuk menghapus data gejala.

#### 4.4.1 Add Gejala



Gambar 4.8 Rancangan Antarmuka Add Gejala

##### Deskripsi

- Halaman ini digunakan oleh admin untuk menambahkan data gejala baru ke dalam basis data.
- Terdapat dua buah tombol, yaitu Submit yang digunakan untuk memasukkan data ke dalam basis data dan tombol Cancel untuk membatalkan masukkan data.

##### Event

```
Page_Load()  
{  
    Mengisi dataGridViewAddGejala dengan semua  
    data gejala dengan SQL Statement sebagai  
    berikut:  
    SELECT * FROM GEJALA;  
}
```

```

OnClick_ buttonSubmitAddGejala()
{
    Memasukkan data gejala di
    dataGridViewAddGejala dengan SQL Statement
    sebagai berikut:

INSERT INTO GEJALA VALUES
( '[TEXTBOXIDGEJALAADDGEJALA] ',
' [RICHTEXTBOXADDGEJALA] ' );
}

OnClick_ buttonCancelAddGejala()
{
    Membersihkan richTextBoxAddGejala.
}

```

#### 4.4.2 Edit Gejala



Gambar 4.9 Rancangan Antarmuka Edit Gejala

#### Deskripsi

- Halaman ini digunakan oleh admin untuk mengubah data gejala yang ada dalam basis data.

- Terdapat dua buah tombol, yaitu Submit yang digunakan untuk memasukkan data admin yang akan dihapus dari dalam basis data dan tombol Cancel untuk membatalkan perubahan data. Terdapat dua buah check box, yaitu check box search by yang digunakan untuk memilih kategori pencarian dan check box gejala yang digunakan untuk melakukan pencarian berdasarkan gejala dari alergi tertentu.

#### Event

```
Page_Load()
```

```
{
```

```
    Mengisi dataGridViewEditGejala dengan id
    gejala dan gejala alergi dari semua data
    gejala dengan SQL Statement sebagai berikut:
```

```
SELECT * FROM GEJALA;
```

```
}
```

```
OnClick_buttonSubmitEditGejala()
```

```
{
```

```
    Jika checkBoxCategoryEditGejala status cek
    sama dengan true maka:
```

```
SELECT * FROM GEJALA WHERE GEJALA_ALERGI
LIKE '%[RICHTEXTBOXEDITGEJALA]%' ;
```

```
    Namun jika tidak maka akan memasukkan data
    gejala yang akan diubah ke
    dataGridViewEditGejala dengan SQL Statement
    sebagai berikut:
```

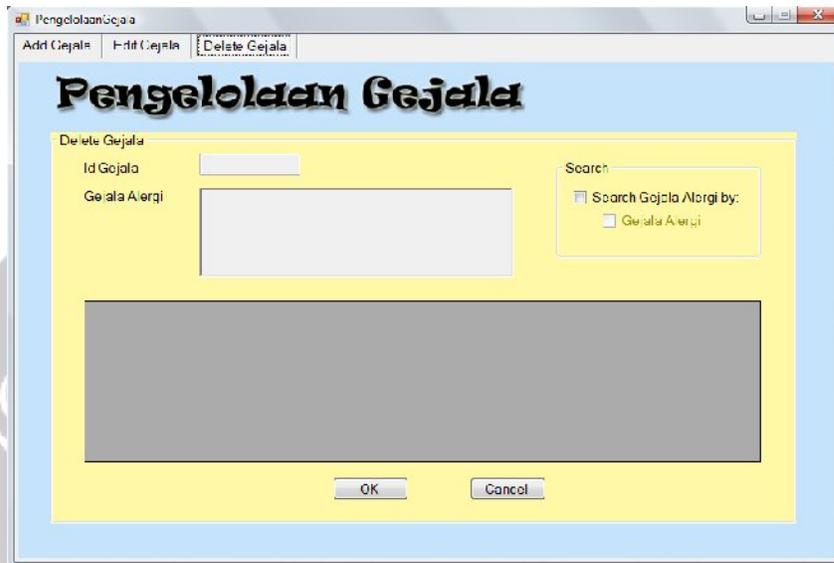
Program Studi Teknik Informatika	DPPL – SiPaGi	66/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

UPDATE      GEJALA      SET      GEJALA_ALERGI=
' [RICHTEXTBOXEDITGEJALA]', WHERE ID_GEJALA=
' [TEXTBOXIDGEJALAEDITGEJALA]'
}
OnClick_ buttonCancelEditGejala()
{
    Membersihkan      richTextBoxEditGejala,
    textBoxIdGejalaEditGejala.
}
• Search
OnClick_ checkBoxCategoryEditGejala()
{
    Mengaktifkan
    checkBoxSearchGejalaAlergiEditGejala.
}
OnClick_ checkBoxSearchGejalaAlergiEditGejala()
{
    Mengaktifkan richTextBoxEditGejala.
}

```

### 4.4.3 Delete Gejala



Gambar 4.10 Rancangan Antarmuka Delete Gejala

#### Deskripsi

- Halaman ini digunakan oleh admin untuk menghapus data gejala yang ada dalam basis data.
- Terdapat dua buah tombol, yaitu Ok yang digunakan untuk menghapus data gejala dari dalam basis data dan tombol Cancel untuk membatalkan penghapusan data. Terdapat dua buah check box, yaitu check box search by yang digunakan untuk memilih kategori pencarian dan check box gejala alergi yang digunakan untuk melakukan pencarian berdasarkan gejala.

#### Event

```
Page_Load()  
{
```

Mengisi dataGridViewDeleteGejala dengan id gejala dan gejala alergi dari semua data gejala dengan SQL Statement sebagai berikut:

```
SELECT * FROM GEJALA;
```

```
}  
OnClick_buttonOkDeleteGejala()  
{  
    Jika checkBoxCategoryEditGejala status cek  
    sama dengan true maka:
```

```
SELECT * FROM GEJALA WHERE GEJALA_ALERGI  
LIKE '%[RICHTEXTBOXDELETEGEJALA]%' ;
```

Namun jika tidak maka akan memasukkan data gejala yang akan dihapus dari dataGridViewDeleteGejala dengan SQL Statement sebagai berikut:

```
DELETE GEJALA WHERE ID_GEJALA=  
' [TEXTBOXIDGEJALADELETEGEJALA] ' ;
```

```
}  
OnClick_buttonCancelDeleteGejala()  
{  
    Membersihkan richTextBoxDeleteGejala,  
    textBoxIdGejalaDeleteGejala.  
}  
}
```

- Search

```
OnClick_checkBoxCategoryDeleteGejala()  
{  
    Mengaktifkan  
    checkBoxSearchGejalaAlergiEditGejala.  
}  
}
```

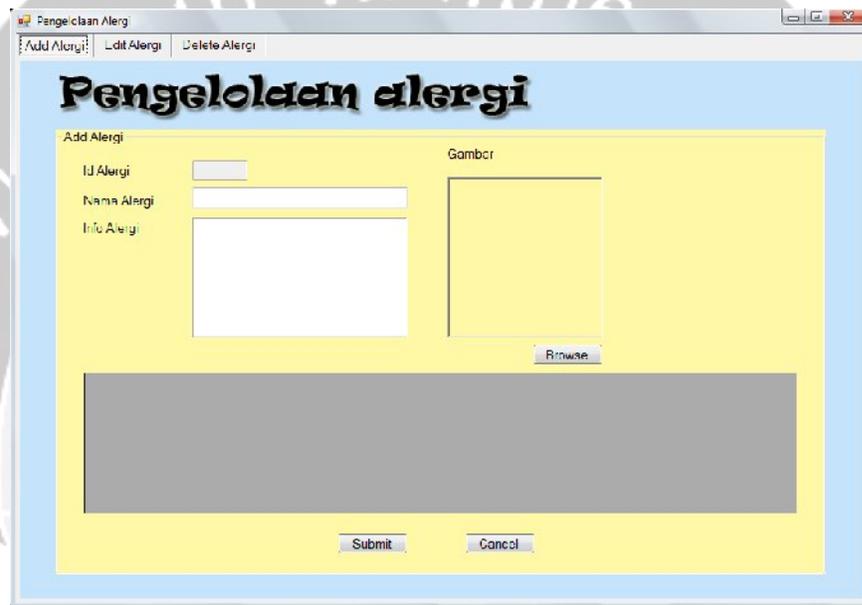
Program Studi Teknik Informatika	DPPL – SiPaGi	69/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

OnClick_checkBoxSearchGejalaAlergiEditGejala()
{
    Mengaktifkan richTextBoxDeleteGejala.
}

```

#### 4.5 Pengelolaan Alergi



**Gambar 4.11 Rancangan Antarmuka Pengelolaan Alergi**

##### **Deskripsi**

- Halaman ini digunakan oleh admin untuk melakukan pengelolaan data alergi.
- Terdapat tiga tab control yang dapat dilakukan untuk melakukan pengelolaan alergi. Add Alergi digunakan untuk menambahkan data alergi baru, Edit Alergi digunakan untuk mengubah data alergi, Delete Alergi digunakan untuk menghapus data alergi.

Program Studi Teknik Informatika	DPPL – SiPaGi	70/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

#### 4.5.1 Add Alergi

The screenshot shows a web application window titled 'Pengelolaan Alergi'. At the top, there are three menu items: 'Add Alergi!', 'Lihat Alergi', and 'Delete Alergi'. The main content area has a yellow background and is titled 'Pengelolaan alergi' in a large, bold, black font. Below the title, there is a section labeled 'Add Alergi'. This section contains three input fields: 'ID Alergi' (a small text box), 'Nama Alergi' (a larger text box), and 'Info Alergi' (a large text area). To the right of these fields is a 'Gambar' field, which is a large empty box with a 'Browse' button below it. At the bottom of the form, there are two buttons: 'Submit' and 'Cancel'. The entire form is set against a light blue background.

Gambar 4.12 Rancangan Antarmuka Add Alergi

#### Deskripsi

- Halaman ini digunakan oleh admin untuk menambahkan data gejala baru ke dalam basis data.
- Terdapat tiga buah tombol, yaitu Submit yang digunakan untuk memasukkan data ke dalam basis data, tombol Cancel untuk membatalkan memasukkan data dan tombol Browse digunakan untuk mencari gambar dari alergi yang akan dimasukkan ke dalam basis data.

#### Event

```
Page_Load()
```

```
{
```

```
    Mengisi dataGridViewAddAlergi dengan semua data alergi dengan SQL Statement sebagai berikut:
```

Program Studi Teknik Informatika	DPPL – SiPaGi	71/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

SELECT ID_ALERGI, NAMA_ALERGI, INFO_ALERGI
FROM ALERGI;
}
OnClick_buttonSubmitAddAlergi()
{
    Memasukkan data alergi di
    dataGridViewAddAlergi dengan SQL Statement
    sebagai berikut:
    INSERT INTO ALERGI VALUES
    (' [TEXTBOXIDALERGIADDAALERGI] ',
    ' [TEXTBOXNAMAALERGIADDAALERGI] ',
    ' [RICHTEXTBOXINFOALERGIADDAALERGI] ',
    ' [PICTUREBOXADDAALERGI] ');
}
OnClick_buttonCancelAddAlergi()
{
    Membersihkan textBoxNamaAlergiAddAlergi,
    richTextBoxInfoAlergiAddAlergi,
    pictureBoxAddAlergi.
}

```

#### 4.5.2 Edit Alergi

The screenshot shows a web browser window titled 'Pengelolaan Alergi'. The interface has a light blue header with the title 'Pengelolaan alergi' in a stylized font. Below the header, there are three navigation buttons: 'Add Alergi', 'Edit Alergi', and 'Delete Alergi'. The main content area is yellow and contains the following elements:

- Edit Alergi** section with three input fields: 'ID Alergi', 'Nama Alergi', and 'Info Alergi'.
- Gambar** section with a large empty box and a 'Browse' button below it.
- Search** section with a search input field and three checkboxes: 'Search Alergi by:', 'Nama Alergi', and 'Info Alergi'.
- At the bottom of the form, there are two buttons: 'Submit' and 'Cancel'.

Gambar 4.13 Rancangan Antarmuka Edit Alergi

#### Deskripsi

- Halaman ini digunakan oleh admin untuk mengubah data alergi yang ada dalam basis data.
- Terdapat tiga buah tombol, yaitu Submit yang digunakan untuk memasukkan data admin yang akan dihapus dari dalam basis data, tombol Cancel untuk membatalkan perubahan data dan tombol Browse digunakan untuk mencari gambar alergi. Terdapat tiga buah check box, yaitu check box search by yang digunakan untuk memilih kategori pencarian, check box nama alergi yang digunakan untuk melakukan pencarian berdasarkan nama alergi dan check box info alergi yang digunakan untuk melakukan pencarian berdasarkan info dari alergi tertentu.

## Event

Page\_Load()

{

Mengisi dataGridViewEditAlergi dengan id alergi, nama alergi dan info alergi dari semua data alergi dengan SQL Statement sebagai berikut:

```
SELECT ID_ALERGI, NAMA_ALERGI, INFO_ALERGI  
FROM ALERGI;
```

}

OnClick\_buttonSubmitEditAlergi()

{

Jika checkBoxCategoryEditAlergi status cek sama dengan true maka:

```
SELECT ID_ALERGI, NAMA_ALERGI, INFO_ALERGI  
FROM ALERGI WHERE NAMA_ALERGI LIKE  
'[TEXTBOXNAMAALERGIEDITALERGI]%' AND  
INFO_ALERGI LIKE  
'[RICHTEXTBOXINFOALERGIEDITALERGI]%;'
```

Namun jika tidak maka akan memasukkan data alergi yang akan diubah ke dataGridViewEditAlergi dengan SQL Statement sebagai berikut:

```
UPDATE ALERGI SET NAMA_ALERGI=  
'[TEXTBOXNAMAALERGIEDITALERGI] ',  
INFO_ALERGI=  
'[RICHTEXTBOXINFOALERGIEDITALERGI] ',  
GAMBAR_ALERGI= '[PICTUREBOXEDITALERGI]'
```

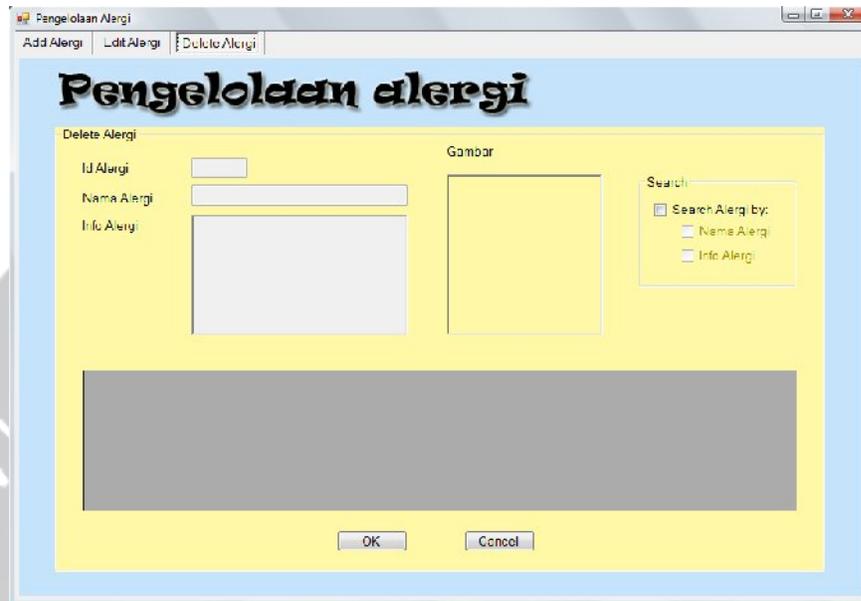
Program Studi Teknik Informatika	DPPL – SiPaGi	74/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

WHERE ID_ALERGI=
      '[TEXTBOXIDALERGIEDITALERGI]'
}
OnClick_buttonCancelEditAlergi()
{
    Membersihkan textBoxIdAlergiEditAlergi,
    textBoxNamaAlergiEditAlergi,
    richTextBoxInfoAlergiEditAlergi,
    pictureBoxEditAlergi.
}
• Search
OnClick_checkBoxCategoryEditAlergi()
{
    Mengaktifkan checkBoxNamaAlergiEditAlergi,
    checkBoxInfoAlergiEditAlergi.
}
OnClick_checkBoxNamaAlergiEditAlergi()
{
    Mengaktifkan textBoxNamaAlergiEditAlergi.
}
OnClick_checkBoxInfoAlergiEditAlergi()
{
    Mengaktifkan
    richTextBoxInfoAlergiEditAlergi.
}

```

### 4.5.3 Delete Alergi



Gambar 4.14 Rancangan Antarmuka Delete Alergi

#### Deskripsi

- Halaman ini digunakan oleh admin untuk menghapus data gejala yang ada dalam basis data.
- Terdapat dua buah tombol, yaitu Ok yang digunakan untuk menghapus data gejala dari dalam basis data dan tombol Cancel untuk membatalkan penghapusan data. Terdapat dua buah check box, yaitu check box search by yang digunakan untuk memilih kategori pencarian dan check box gejala alergi yang digunakan untuk melakukan pencarian berdasarkan gejala.

#### Event

```
Page_Load()  
{
```

Mengisi dataGridViewDeleteGejala dengan id gejala dan gejala alergi dari semua data gejala dengan SQL Statement sebagai berikut:

```
SELECT ID_ALERGI, NAMA_ALERGI, INFO_ALERGI  
FROM ALERGI;
```

```
}  
OnClick_ buttonOkDeleteAlergi()  
{  
    Jika checkBoxCategoryDeleteAlergi status  
    ceknya sama dengan true maka:
```

```
SELECT ID_ALERGI, NAMA_ALERGI, INFO_ALERGI  
FROM ALERGI WHERE NAMA_ALERGI LIKE  
'[TEXTBOXNAMAALERGIDELETEALERGI]%' AND  
INFO_ALERGI LIKE  
'[RICHTEXTBOXINFOALERGIDELETEALERGI]%;'
```

Namun jika tidak maka akan memasukkan data alergi yang akan dihapus dari dataGridViewDeleteAlergi dengan SQL Statement sebagai berikut:

```
DELETE ALERGI WHERE ID_ALERGI=  
'[TEXTBOXIDALERGIDELETEALERGI]';
```

```
}  
OnClick_ buttonCancelDeleteAlergi()  
{  
    Membersihkan textBoxIdAlergiDeleteAlergi,  
    textBoxNamaAlergiDeleteAlergi,  
    richTextBoxInfoAlergiDeleteAlergi,  
    pictureBoxDeleteAlergi.  
}
```

Program Studi Teknik Informatika	DPPL – SiPaGi	77/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

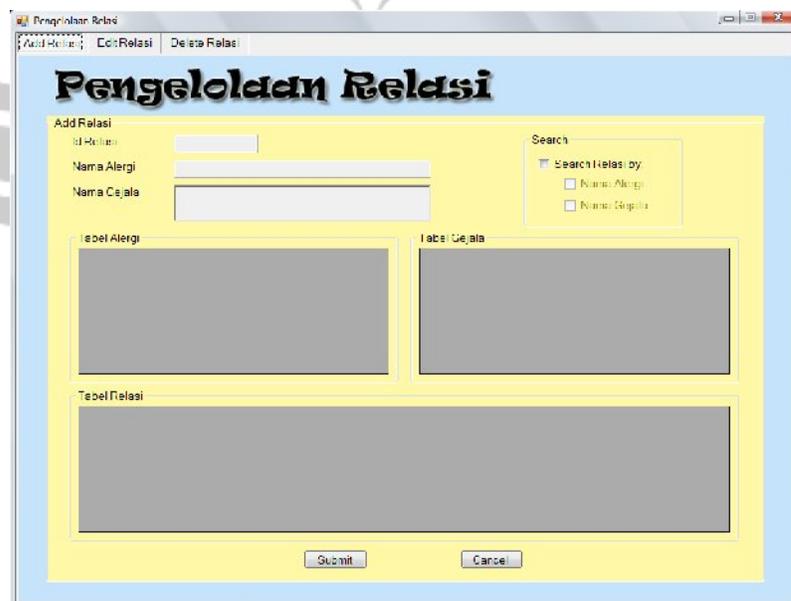
- Search

```

OnClick_checkBoxCategoryDeleteAlergi ()
{
    Mengaktifkan checkBoxNamaAlergiDeleteAlergi,
    checkBoxInfoAlergiDeleteAlergi.
}
OnClick_checkBoxNamaAlergiDeleteAlergi ()
{
    Mengaktifkan textBoxNamaAlergiDeleteAlergi.
}
OnClick_checkBoxInfoAlergiDeleteAlergi ()
{
    Mengaktifkan
    richTextBoxInfoAlergiDeleteAlergi.
}

```

#### 4.6 Pengelolaan Relasi



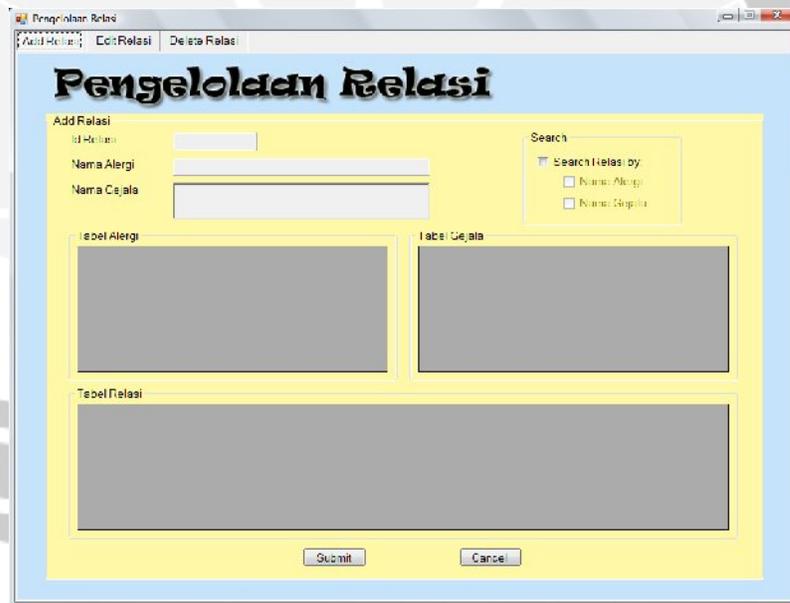
**Gambar 4.15 Rancangan Antarmuka Pengelolaan Relasi**

Program Studi Teknik Informatika	DPPL – SiPaGi	78/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## Deskripsi

- Halaman ini digunakan oleh admin untuk melakukan pengelolaan data relasi.
- Terdapat tiga tab control yang dapat dilakukan untuk melakukan pengelolan relasi. Add Relasi digunakan untuk menambahkan data relasi baru, Edit Relasi digunakan untuk mengubah data relasi dan Delete Relasi digunakan untuk menghapus data relasi.

### 4.6.1 Add Relasi



The screenshot shows a web browser window titled 'Pengelolaan Relasi'. The page has a blue header with the title 'Pengelolaan Relasi' in a stylized font. Below the header, there are three tabs: 'Add Relasi', 'Edit Relasi', and 'Delete Relasi'. The 'Add Relasi' tab is active. The form contains several input fields: 'Id Relasi', 'Nama Alergi', and 'Nama Gejala'. There is also a 'Search' section with a checkbox for 'Search Relasi by:' and two radio buttons for 'Nama Alergi' and 'Nama Gejala'. Below the input fields, there are three table placeholders: 'Tabel Alergi', 'Tabel Gejala', and 'Tabel Relasi'. At the bottom of the form, there are 'Submit' and 'Cancel' buttons.

Gambar 4.16 Rancangan Antarmuka Add Relasi

## Deskripsi

- Halaman ini digunakan oleh admin untuk menambahkan data relasi baru ke dalam basis data.

- Terdapat dua buah tombol, yaitu Submit yang digunakan untuk memasukkan data ke dalam basis data dan tombol Cancel untuk membatalkan masukkan data. Terdapat tiga buah check box, yaitu check box search by yang digunakan untuk memilih kategori pencarian, check box gejala yang digunakan untuk melakukan pencarian berdasarkan gejala dan check box alergi yang digunakan untuk melakukan pencarian berdasarkan alergi tertentu.

#### Event

```

Page_Load()
{
    Mengisi dataGridViewRelasiAddRelasi dengan
    semua data relasi dengan SQL Statement
    sebagai berikut:
SELECT ID_RELASI, A.IDALERGI, A.ID_GEJALA,
NAMAALERGI, GEJALALERGI FROM RELASI A
JOIN ALERGI B ON A.IDALERGI=B.IDALERGI
JOIN GEJALA C ON A.ID_GEJALA=C.ID_GEJALA;
    Mengisi dataGridViewGejalaAddRelasi dengan
    semua data gejala dengan SQL Statement
    sebagai berikut:
SELECT * FROM GEJALA;
    Mengisi dataGridViewAlergiAddRelasi dengan
    semua data alergi dengan SQL Statement
    sebagai berikut:
SELECT * FROM ALERGI;
}

```

Program Studi Teknik Informatika	DPPL – SiPaGi	80/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

OnClick_ buttonSubmitAddRelasi()
{
    Jika checkBoxCategoryAddRelasi status cek
    sama dengan true maka:
SELECT * FROM ALERGI WHERE NAMA_ALERGI LIKE
'%[TEXTBOXNAMAALERGIADDRELASI]%' ;
SELECT * FROM GEJALA WHERE GEJALA_ALERGI
LIKE '%[RICHTEXTBOXNAMAGEJALAADDRELASI]%' ;
SELECT ID_RELASI, A.ID_ALERGI, A.ID_GEJALA,
NAMA_ALERGI, GEJALA_ALERGI FROM RELASI A
JOIN ALERGI B ON A.ID_ALERGI=B.ID_ALERGI
JOIN GEJALA C ON A.ID_GEJALA=C.ID_GEJALA
WHERE NAMA_ALERGI LIKE
'%[TEXTBOXNAMAALERGIADDRELASI]%' AND
GEJALA_ALERGI LIKE
'%[RICHTEXTBOXNAMAGEJALAADDRELASI]%' ;
    Namun jika tidak maka akan memasukkan data
    relasi di dataGridViewAddRelasi dengan SQL
    Statement sebagai berikut:
INSERT INTO RELASI (ID_ALERGI, ID_GEJALA)
VALUES (' [TEXTBOXNAMAALERGIADDRELASI] ',
' [RICHTEXTBOXNAMAGEJALAADDRELASI] ' ) ;
}

OnClick_ buttonCancelAddRelasi()
{
    Membersihkan textBoxNamaAlergiAddRelasi,
    richTextBoxNamaGejalaAddRelasi.
}

```

Program Studi Teknik Informatika	DPPL – SiPaGi	81/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

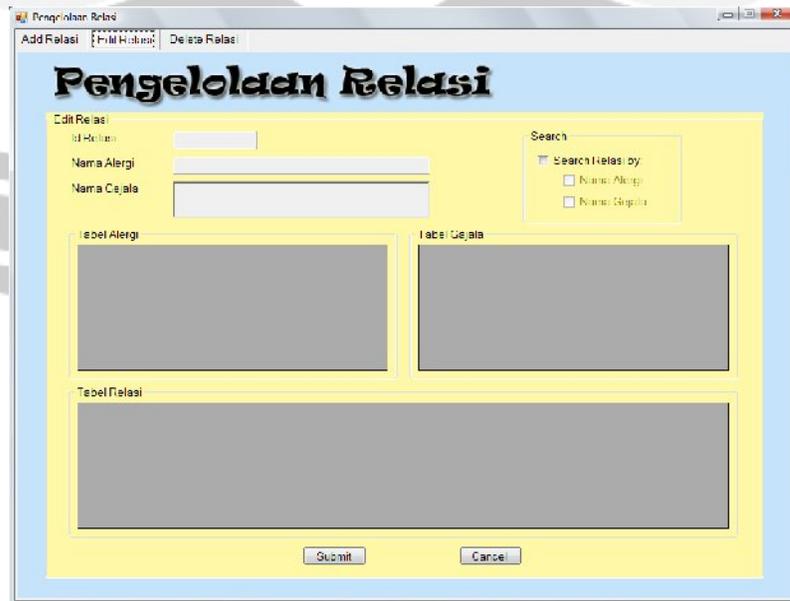
- Search

```

OnClick_checkBoxCategoryAddRelasi()
{
    Mengaktifkan checkBoxNamaAlergiAddRelasi,
    checkBoxNamaGejalaAddRelasi.
}
OnClick_checkBoxNamaAlergiAddRelasi()
{
    Mengaktifkan textBoxNamaAlergiAddRelasi.
}
OnClick_checkBoxNamaGejalaAddRelasi()
{
    Mengaktifkan richTextBoxNamaGejalaAddRelasi.
}

```

#### 4.6.2 Edit Relasi



Gambar 4.17 Rancangan Antarmuka Edit Relasi

#### Deskripsi

Program Studi Teknik Informatika	DPPL – SiPaGi	82/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

- Halaman ini digunakan oleh admin untuk mengubah data relasi yang ada dalam basis data.
- Terdapat dua buah tombol, yaitu Submit yang digunakan untuk memasukkan data relasi yang akan diubah dan tombol Cancel untuk membatalkan perubahan data. Terdapat tiga buah check box, yaitu check box search by yang digunakan untuk memilih kategori pencarian, check box gejala yang digunakan untuk melakukan pencarian berdasarkan gejala dan check box alergi yang digunakan untuk melakukan pencarian berdasarkan alergi tertentu.

#### Event

```
Page_Load()
{
```

Mengisi dataGridViewRelasiEditRelasi dengan semua data relasi dengan SQL Statement sebagai berikut:

```
SELECT ID_RELASI, A.IDALERGI, A.ID_GEJALA,
NAMAALERGI, GEJALALERGI FROM RELASI A
JOIN ALERGI B ON A.IDALERGI=B.IDALERGI
JOIN GEJALA C ON A.ID_GEJALA=C.ID_GEJALA;
```

Mengisi dataGridViewGejalaEditRelasi dengan semua data gejala dengan SQL Statement sebagai berikut:

```
SELECT * FROM GEJALA;
```

Mengisi dataGridViewAlergiEditRelasi dengan semua data alergi dengan SQL Statement sebagai berikut:

Program Studi Teknik Informatika	DPPL – SiPaGi	83/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

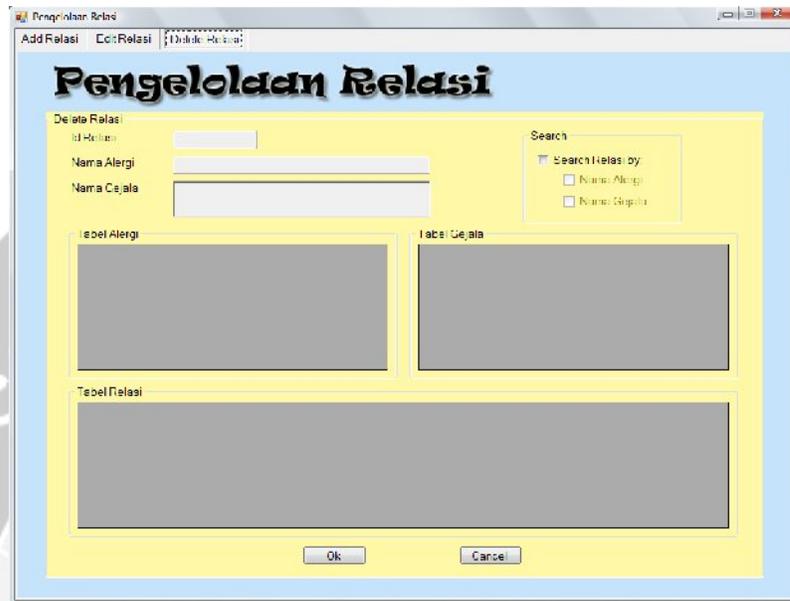
SELECT * FROM ALERGI;
}
OnClick_ buttonSubmitEditRelasi()
{
    Jika checkBoxCategoryEditRelasi status cek
    sama dengan true maka:
    SELECT * FROM ALERGI WHERE NAMA_ALERGI LIKE
    '[TEXTBOXNAMAALERGIEDITRELASI]';
    SELECT * FROM GEJALA WHERE GEJALA_ALERGI
    LIKE '[RICHTEXTBOXNAMAGEJALAEDITRELASI]';
    SELECT ID_RELASI, A.ID_ALERGI, A.ID_GEJALA,
    NAMA_ALERGI, GEJALA_ALERGI FROM RELASI A
    JOIN ALERGI B ON A.ID_ALERGI=B.ID_ALERGI
    JOIN GEJALA C ON A.ID_GEJALA=C.ID_GEJALA
    WHERE NAMA_ALERGI LIKE
    '[TEXTBOXNAMAALERGIEDITRELASI]' AND
    GEJALA_ALERGI LIKE
    '[RICHTEXTBOXNAMAGEJALAEDITRELASI]';
    Namun jika tidak maka akan memasukkan data
    relasi yang akan diubah ke
    dataGridViewEditRelasi dengan SQL Statement
    sebagai berikut:
    UPDATE RELASI SET ID_ALERGI=
    '[TEXTBOXNAMAALERGIEDITRELASI]', ID_GEJALA=
    '[RICHTEXTBOXNAMAGEJALAEDITRELASI]' WHERE
    ID_RELASI= '[TEXTBOXIDRELASIEDITRELASI]'
}
OnClick_ buttonCancelDeleteAdmin()
{

```

Program Studi Teknik Informatika	DPPL – SiPaGi	84/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```
Membersihkan      textBoxIdRelasiEditRelasi,
textBoxNamaAlergiEditRelasi,
richTextBoxNamaGejalaEditRelasi.
}
• Search
OnClick_checkBoxCategoryEditRelasi()
{
    Mengaktifkan      checkBoxNamaAlergiEditRelasi,
checkBoxNamaGejalaEditRelasi.
}
OnClick_checkBoxNamaAlergiEditRelasi()
{
    Mengaktifkan textBoxNamaAlergiEditRelasi.
}
OnClick_checkBoxNamaGejalaEditRelasi()
{
    Mengaktifkan
richTextBoxNamaGejalaEditRelasi.
}
```

### 4.6.3 Delete Relasi



Gambar 4.18 Rancangan Antarmuka Delete Relasi

#### Deskripsi

- Halaman ini digunakan oleh admin untuk menghapus data relasi yang ada dalam basis data.
- Terdapat dua buah tombol, yaitu Ok yang digunakan untuk menghapus data relasi dari dalam basis data dan tombol Cancel untuk membatalkan penghapusan data. Terdapat tiga buah check box, yaitu check box search by yang digunakan untuk memilih kategori pencarian, check box alergi yang digunakan untuk melakukan pencarian berdasarkan alergi dan check box gejala yang digunakan untuk melakukan pencarian berdasarkan gejala tertentu.

#### Event

Program Studi Teknik Informatika	DPPL – SiPaGi	86/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

Page_Load()
{
    Mengisi      dataGridViewRelasiDeleteRelasi
dengan      semua data relasi dengan SQL
Statement sebagai berikut:
SELECT ID_RELASI, A.ID_ALERGI, A.ID_GEJALA,
NAMA_ALERGI, GEJALA_ALERGI FROM RELASI A
JOIN ALERGI B ON A.ID_ALERGI=B.ID_ALERGI
JOIN GEJALA C ON A.ID_GEJALA=C.ID_GEJALA;
    Mengisi      dataGridViewGejalaDeleteRelasi
dengan      semua data gejala dengan SQL
Statement sebagai berikut:
SELECT * FROM GEJALA;
    Mengisi      dataGridViewAlergiDeleteRelasi
dengan      semua data alergi dengan SQL
Statement sebagai berikut:
SELECT * FROM ALERGI;
}
OnClick_buttonOkDeleteRelasi()
{
    Jika checkBoxCategoryEditRelasi status cek
sama dengan true maka:
SELECT * FROM ALERGI WHERE NAMA_ALERGI LIKE
'[%[TEXTBOXNAMAALERGIDELETERELASI]%' ;
SELECT * FROM GEJALA WHERE GEJALA_ALERGI
LIKE
'[%[RICHTEXTBOXNAMAGEJALADELETERELASI]%' ;
SELECT ID_RELASI, A.ID_ALERGI, A.ID_GEJALA,
NAMA_ALERGI, GEJALA_ALERGI FROM RELASI A

```

Program Studi Teknik Informatika	DPPL – SiPaGi	87/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

JOIN ALERGI B ON A.ID_ALERGI=B.ID_ALERGI
JOIN GEJALA C ON A.ID_GEJALA=C.ID_GEJALA
WHERE NAMA_ALERGI LIKE
'%[TEXTBOXNAMAALERGIDELATERELASI]%' AND
GEJALA_ALERGI LIKE
'%[RICHTEXTBOXNAMAGEJALADELATERELASI]%' ;

```

Memasukkan data relasi yang akan dihapus dari dataGridViewDeleteRelasi dengan SQL Statement sebagai berikut:

```

DELETE RELASI WHERE ID_RELASI=
'[TEXTBOXIDRELASIDELATERELASI]';

```

```

}
OnClick_buttonCancelDeleteRelasi()
{
    Membersihkan textBoxIdRelasiDeleteRelasi,
    textBoxNamaAlergiDeleteRelasi,
    richTextBoxNamaGejalaDeleteRelasi.
}

```

- Search

```

OnClick_checkBoxCategoryDeleteRelasi()
{
    Mengaktifkan checkBoxNamaAlergiDeleteRelasi,
    checkBoxNamaGejalaDeleteRelasi.
}
OnClick_checkBoxNamaAlergiDeleteRelasi()
{
    Mengaktifkan textBoxNamaAlergiDeleteRelasi.
}

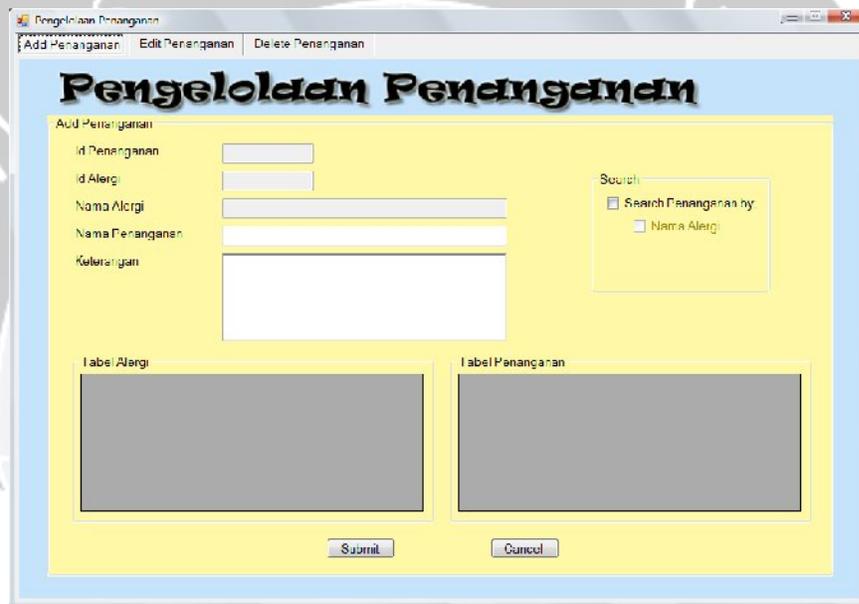
```

```

OnClick_checkBoxNamaGejalaDeleteRelasi()
{
    Mengaktifkan
    richTextBoxNamaGejalaDeleteRelasi.
}

```

#### 4.7 Pengelolaan Penanganan



**Gambar 4.19 Rancangan Antarmuka Pengelolaan Penanganan**

#### **Deskripsi**

- Halaman ini digunakan oleh admin untuk melakukan pengelolaan data penanganan.
- Terdapat tiga tab control yang dapat dilakukan untuk melakukan pengelolaan penanganan. Add Penanganan digunakan untuk menambahkan data penanganan baru, Edit Penanganan digunakan untuk mengubah data penanganan, Delete Penanganan digunakan untuk menghapus data penanganan.

Program Studi Teknik Informatika	DPPL – SiPaGi	89/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

#### 4.7.1 Add Penanganan

The screenshot shows a web browser window titled 'Pengelolaan Penanganan'. The main content area has a yellow background and is titled 'Pengelolaan Penanganan'. Below the title, there is a section for 'Add Penanganan' with several input fields: 'Id Penanganan', 'Id Alergi', 'Nama Alergi', 'Nama Penanganan', and 'Keterangan'. To the right of these fields is a search section with a 'Search' input, a 'Search Penanganan by' checkbox, and a 'Nama Alergi' checkbox. Below the input fields are two large grey boxes labeled 'Label Alergi' and 'Label Penanganan'. At the bottom of the form are 'Submit' and 'Cancel' buttons.

Gambar 4.20 Rancangan Antarmuka Add Penanganan

#### Deskripsi

- Halaman ini digunakan oleh admin untuk menambahkan data penanganan baru ke dalam basis data.
- Terdapat dua buah tombol, yaitu Submit yang digunakan untuk memasukkan data ke dalam basis data dan tombol Cancel untuk membatalkan masukkan data. Terdapat dua buah check box, yaitu check box search by yang digunakan untuk memilih kategori pencarian, check box nama alergi yang digunakan untuk melakukan pencarian berdasarkan nama alergi.

#### Event

```
Page_Load()  
{
```

Program Studi Teknik Informatika	DPPL – SiPaGi	90/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Mengisi dataGridViewAlergiAddPenanganan dengan semua data alergi dengan SQL Statement sebagai berikut:

```
SELECT * FROM ALERGI;
```

Mengisi dataGridViewPenangananAddPenanganan dengan semua data penanganan dengan SQL Statement sebagai berikut:

```
SELECT * FROM PENANGANAN;
```

```
}
```

```
OnClick_ buttonSubmitAddPenanganan()
```

```
{
```

Jika checkBoxCategoryAddPenanganan status cek sama dengan true maka:

```
SELECT ID_ALERGI, NAMA_ALERGI, INFO_ALERGI  
FROM ALERGI WHERE NAMA_ALERGI LIKE  
'[TEXTBOXNAMAALERGIADDPENANGANAN]';
```

Namun jika tidak maka akan memasukkan data penanganan di dataGridViewPenangananAddPenanganan dengan SQL Statement sebagai berikut:

```
INSERT INTO PENANGANAN VALUES  
( '[TEXTBOXIDPENANGANANADDPENANGANAN] ',  
'[TEXTBOXIDALERGIADDPENANGANAN] ',  
'[TEXTBOXNAMAPENANGANANADDPENANGANAN] ',  
'[RICHTEXTBOXKETERANGANADDPENANGANAN] ')";
```

```
}
```

```
OnClick_ buttonCancelAddPenanganan()
```

```
{
```

Membersihkan textBoxIdAlergiAddPenanganan,  
textBoxNamaAlergiAddPenanganan

Program Studi Teknik Informatika	DPPL – SiPaGi	91/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

        textBoxNamaPenangananAddPenanganan,
        richTextBoxKeteranganAddPenanganan.
    }
}

```

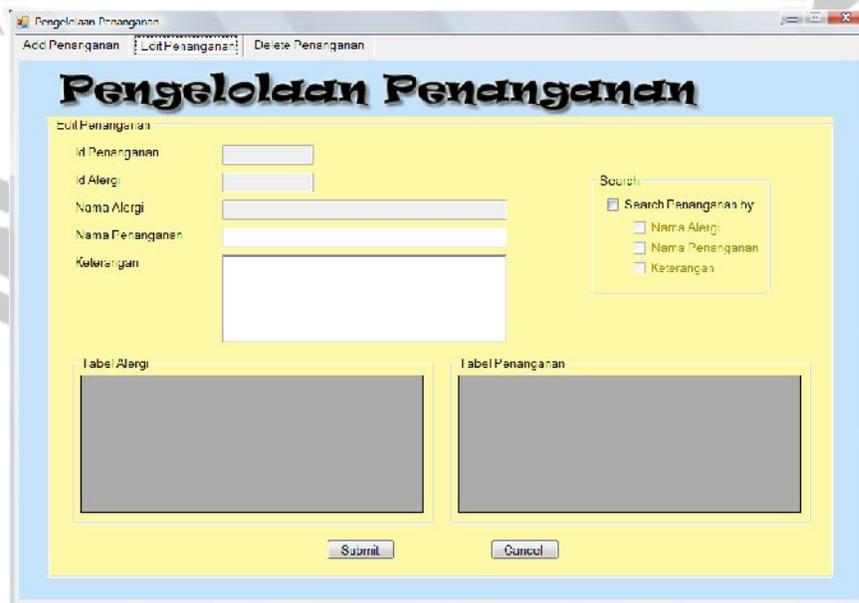
- Search

```

OnClick_checkBoxCategoryAddPenanganan ()
{
    Mengaktifkan
    checkBoxNamaAlergiAddPenanganan.
}
OnClick_checkBoxNamaAlergiAddPenanganan ()
{
    Mengaktifkan textBoxNamaAlergiAddPenanganan.
}

```

#### 4.7.2 Edit Penanganan



Gambar 4.21 Rancangan Antarmuka Edit Penanganan

## Deskripsi

- Halaman ini digunakan oleh admin untuk mengubah data penanganan yang ada dalam basis data.
- Terdapat dua buah tombol, yaitu Submit yang digunakan untuk memasukkan data penanganan yang akan diubah dari dalam basis data dan tombol Cancel untuk membatalkan perubahan data. Terdapat empat buah check box, yaitu check box search by yang digunakan untuk memilih kategori pencarian dan check box nama alergi yang digunakan untuk melakukan pencarian berdasarkan alergi, check box nama penanganan yang digunakan untuk melakukan pencarian berdasarkan penanganan dan check box keterangan digunakan untuk melakukan pencarian berdasarkan keterangan dari penanganan.

## Event

```
Page_Load()  
{  
    Mengisi dataGridViewAlergiEditPenanganan  
    dengan semua data alergi dengan SQL  
    Statement sebagai berikut:  
    SELECT * FROM ALERGI;  
    Mengisi dataGridViewPenangananEditPenanganan  
    dengan semua data penanganan dengan SQL  
    Statement sebagai berikut:  
    SELECT * FROM PENANGANAN;  
}
```

Program Studi Teknik Informatika	DPPL – SiPaGi	93/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

OnClick_ buttonSubmitEditGejala()
{
    Jika checkBoxCategoryEditPenanganan status
    ceknya sama dengan true maka:
SELECT ID_ALERGI, NAMA_ALERGI, INFO_ALERGI
FROM ALERGI WHERE NAMA_ALERGI LIKE
'%[TEXTBOXNAMAALERGIEDITPENANGANAN]%' ;
SELECT * FROM PENANGANAN WHERE
NAMA_PENANGANAN
'%[TEXTBOXNAMAPENANGANANEDITPENANGANAN]%'
AND KETERANGAN
'%[RICHTEXTBOXKETERANGANEDITPENANGANAN]%' ;
    Namun jika tidak maka akan memasukkan data
    penanganan yang diubah ke
    dataGridViewPenangananEditPenanganan dengan
    SQL Statement sebagai berikut:
UPDATE PENANGANAN SET ID_ALERGI=
' [TEXTBOXIDALERGIEDITPENANGANAN] ',
NAMA_PENANGANAN
= ' [TEXTBOXNAMAPENANGANANEDITPENANGANAN] ',
KETERANGAN=
' [RICHTEXTBOXKETERANGANEDITPENANGANAN] '
WHERE ID_PENANGANAN
= ' [TEXTBOXIDPENANGANANEDITPENANGANAN] ' ;
}

OnClick_ buttonCancelEditPenanganan()
{
    Membersihkan
    textBoxIdPenangananEditPenanganan,

```

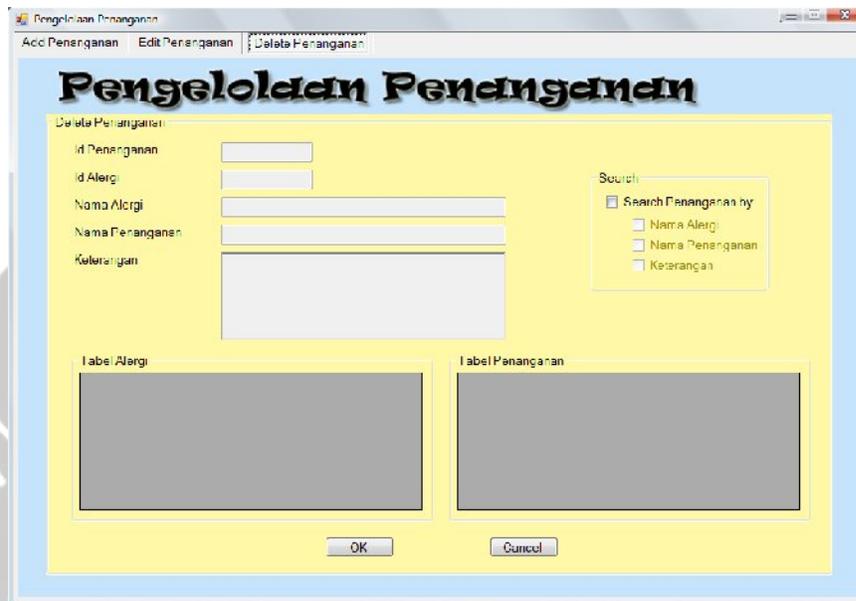
Program Studi Teknik Informatika	DPPL – SiPaGi	94/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

        textBoxIdAlergiEditPenanganan,
        textBoxNamaAlergiEditPenanganan,
        textBoxNamaPenangananEditPenanganan,
        richTextBoxKeteranganEditPenanganan.
    }
    • Search
    OnClick_checkBoxCategoryEditPenanganan()
    {
        Mengaktifkan
        checkBoxNamaAlergiEditPenanganan,
        checkBoxNamaPenangananEditPenanganan,
        checkBoxKeteranganEditPenanganan.
    }
    OnClick_checkBoxNamaAlergiEditPenanganan()
    {
        Mengaktifkan
        textBoxNamaAlergiEditPenanganan.
    }
    OnClick_checkBoxNamaPenangananEditPenanganan()
    {
        Mengaktifkan
        textBoxNamaPenangananEditPenanganan.
    }
    OnClick_checkBoxKeteranganEditPenanganan()
    {
        Mengaktifkan
        richTextBoxKeteranganEditPenanganan.
    }
}

```

### 4.7.3 Delete Penanganan



Gambar 4.22 Rancangan Antarmuka Delete Penanganan

#### Deskripsi

- Halaman ini digunakan oleh admin untuk menghapus data penanganan dari dalam basis data.
- Terdapat dua buah tombol, yaitu Ok yang digunakan untuk menghapus data penanganan dari dalam basis data dan tombol Cancel untuk membatalkan penghapusan data. Terdapat empat buah check box, yaitu check box search by yang digunakan untuk memilih kategori pencarian dan check box nama alergi yang digunakan untuk melakukan pencarian berdasarkan alergi, check box nama penanganan yang digunakan untuk melakukan pencarian berdasarkan penanganan dan check box keterangan digunakan untuk melakukan

pencarian berdasarkan keterangan dari penanganan.

### Event

```
Page_Load()  
{  
    Mengisi dataGridViewAlergiDeletePenanganan  
    dengan semua data alergi dengan SQL  
    Statement sebagai berikut:  
    SELECT * FROM ALERGI;  
    Mengisi  
    dataGridViewPenangananDeletePenanganan  
    dengan semua data penanganan dengan SQL  
    Statement sebagai berikut:  
    SELECT * FROM PENANGANAN;  
}  
OnClick_buttonOkDeletePenanganan()  
{  
    Jika checkBoxCategoryDeletePenanganan status  
    ceknya sama dengan true maka:  
    SELECT ID_ALERGI, NAMA_ALERGI, INFO_ALERGI  
    FROM ALERGI WHERE NAMA_ALERGI LIKE  
    '[TEXTBOXNAMAALERGIDELETEPENANGANAN] %';  
    SELECT * FROM PENANGANAN WHERE  
    NAMA_PENANGANAN LIKE  
    '[TEXTBOXNAMAPENANGANANDELETEPENANGANAN] %'  
    AND KETERANGAN LIKE  
    '[RICHTEXTBOXKETERANGANDELETEPENANGANAN] %';  
    Jika tidak maka akan memasukkan data  
    penanganan yang akan dihapus dari
```

Program Studi Teknik Informatika	DPPL – SiPaGi	97/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

dataGridViewPenangananDeletePenanganan  
dengan SQL Statement sebagai berikut:

```
DELETE PENANGANAN WHERE ID_PENANGANAN=  
' [TEXTBOXIDPENANGANANDELETEPENANGANAN] ' ;
```

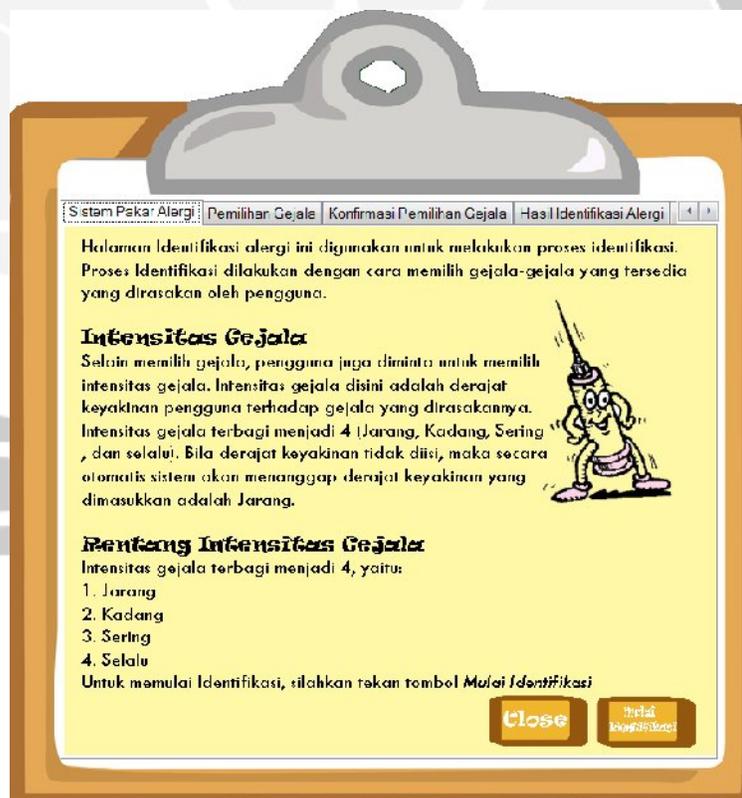
```
}  
OnClick_buttonCancelDeletePenanganan()  
{  
    Membersihkan  
    textBoxIdPenangananDeletePenanganan,  
    textBoxIdAlergiDeletePenanganan,  
    textBoxNamaAlergiDeletePenanganan,  
    textBoxNamaPenangananDeletePenanganan,  
    richTextBoxKeteranganDeletePenanganan.  
}  
• Search  
OnClick_checkBoxCategoryDeletePenanganan()  
{  
    Mengaktifkan  
    checkBoxNamaAlergiDeletePenanganan,  
    checkBoxNamaPenangananDeletePenanganan,  
    checkBoxKeteranganDeletePenanganan.  
}  
OnClick_checkBoxNamaAlergiDeletePenanganan()  
{  
    Mengaktifkan  
    textBoxNamaAlergiDeletePenanganan.  
}  
OnClick_  
checkBoxNamaPenangananDeletePenanganan()
```

```

{
    Mengaktifkan
    textBoxNamaPenangananDeletePenanganan.
}
OnClick_checkBoxKeteranganDeletePenanganan ()
{
    Mengaktifkan
    richTextBoxKeteranganDeletePenanganan.
}

```

#### 4.8 Identifikasi Alergi



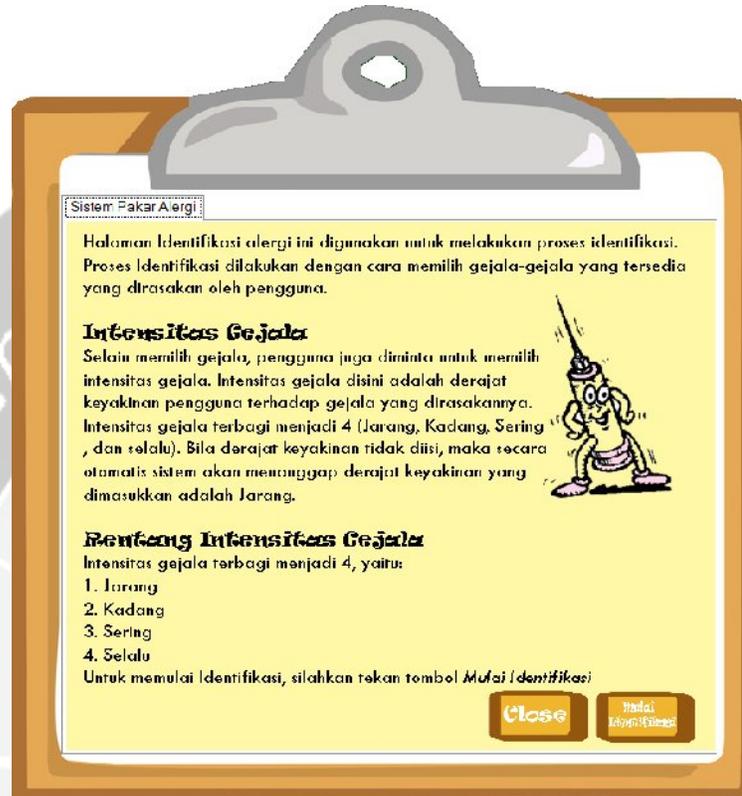
Gambar 4.23 Rancangan Antarmuka Identifikasi Alergi

### **Deskripsi**

- Halaman ini digunakan oleh pasien untuk melakukan identifikasi alergi.
- Terdapat lima buah tab control, yaitu tab control sistem pakar alergi merupakan halaman untuk memulai identifikasi alergi, tab control pemilihan gejala digunakan untuk memilih gejala, tab control konfirmasi pemilihan gejala digunakan untuk memastikan gejala yang dipilih serta menentukan intensitas gejala, tab control hasil identifikasi alergi digunakan untuk menampilkan prosentase dari alergi, tab control penanganan alergi digunakan untuk menampilkan penanganan dari alergi tertentu.

Program Studi Teknik Informatika	DPPL – SiPaGi	100/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

#### 4.8.1 Sistem Pakar Alergi



Gambar 4.24 Rancangan Antarmuka Halaman Awal Identifikasi

#### Deskripsi

- Halaman ini digunakan oleh pasien untuk memulai identifikasi alergi.
- Terdapat dua buah tombol, yaitu tombol Close yang digunakan keluar dari sistem dan tombol Mulai Identifikasi untuk memulai identifikasi alergi.

#### Event

```
OnClick_buttonCloseAwal()  
{  
    Keluar dari sistem identifikasi dan kembali  
    ke Menu Utama.  
}
```

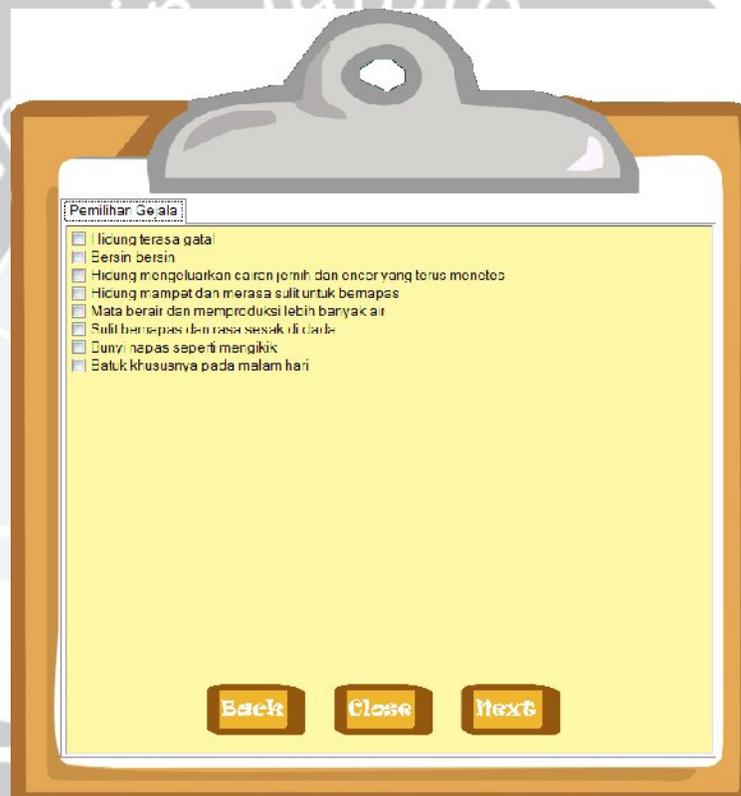
Program Studi Teknik Informatika	DPPL – SiPaGi	101/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

OnClick_ buttonMulaiIdentifikasi()
{
    Memunculkan tab control Pemilihan Gejala.
}

```

#### 4.8.2 Pemilihan Gejala



Gambar 4.25 Rancangan Antarmuka Pemilihan Gejala

#### Deskripsi

- Halaman ini digunakan oleh pasien untuk memilih gejala yang dialami.
- Terdapat tiga buah tombol, yaitu tombol Back yang digunakan untuk kembali ke tab control sebelumnya, tombol Close yang digunakan keluar dari sistem dan tombol Next digunakan untuk melanjutkan ke tab control berikutnya.

Program Studi Teknik Informatika	DPPL – SiPaGi	102/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## Event

```
Page_Load()  
{  
    Mengisi checkedListBoxPemilihanGejala dengan  
    semua data gejala dengan SQL Statement  
    sebagai berikut:  
    SELECT * FROM GEJALA;  
}  
OnClick_buttonBackPemilihanGejala()  
{  
    Kembali ke tab control Sistem Pakar Alergi.  
}  
OnClick_buttonClosePemilihanGejala()  
{  
    Keluar dari sistem dan kembali ke Menu  
    Utama.  
}  
OnClick_buttonNextPemilihanGejala()  
{  
    Memunculkan tab control Pemilihan Gejala.  
}
```

### 4.8.3 Konfirmasi Pemilihan Gejala



Gambar 4.26 Rancangan Antarmuka Konfirmasi Pemilihan Gejala

#### Deskripsi

- Halaman ini digunakan oleh pasien untuk melihat jenis gejala yang dipilih serta menentukan intensitas terjadinya gejala tersebut.
- Terdapat tiga buah tombol, yaitu tombol Back yang digunakan untuk kembali ke tab control sebelumnya, tombol Close yang digunakan keluar dari sistem dan tombol Next digunakan untuk melanjutkan ke tab control berikutnya.

#### Event

```
Page_Load()  
{
```

Mengisi

dataGridViewKonfirmasiPemilihanGejala dengan data gejala dari checkedListBoxPemilihanGejala yang diberi tanda cek dan memberikan atribut intensitas gejala.

```
}
OnClick_ buttonBackKonfirmasiPemilihanGejala()
{
    Kembali ke tab control sebelumnya.
}
OnClick_ buttonCloseKonfirmasiPemilihanGejala()
{
    Keluar dari sistem dan kembali ke Menu
    Utama.
}
OnClick_ buttonNextKonfirmasiPemilihanGejala()
{
    Menghitung jumlah kemiripan gejala di
    masing-masing alergi dengan SQL statement
    sebagai berikut:

SELECT R.ID_ALERGI, A.NAMA_ALERGI 'NAMA
ALERGI', COUNT(R.ID_GEJALA) 'PROSENTASE
KEMIRIPAN (%)' FROM RELASI R JOIN ALERGI A
ON R.ID_ALERGI= A.ID_ALERGI WHERE
R.ID_GEJALA=
' [DATAGRIDVIEWKONFIRMASIPEMILIHANGEJALA] '
GROUP BY R.ID_ALERGI,A.NAMA_ALERGI ORDER BY
R.ID_ALERGI ASC;

    Menghitung gejala masing-masing alergi
    dengan SQL statement sebagai berikut:
```

Program Studi Teknik Informatika	DPPL – SiPaGi	105/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

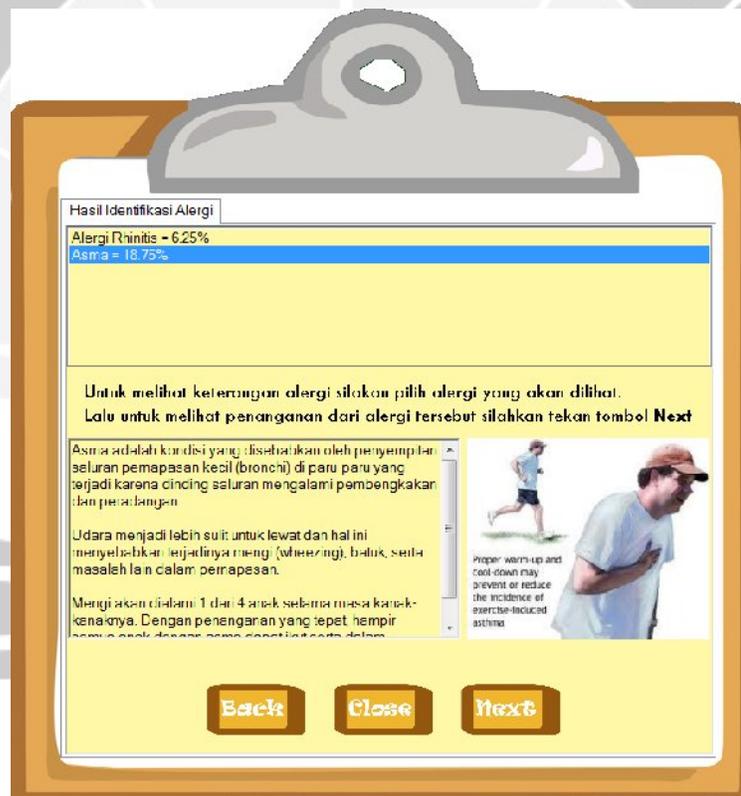
```

SELECT      ID_ALERGI,      COUNT (ID_GEJALA) FROM
RELASI      WHERE          ID_ALERGI=
' [TABELJUMLAHKEMIRIPANGEJALA] ' GROUP BY
ID_ALERGI ORDER BY ID_ALERGI ASC;

```

Memunculkan tab control Hasil Identifikasi Alergi.

#### 4.8.4 Hasil Identifikasi Alergi



Gambar 4.27 Rancangan Antarmuka Hasil Identifikasi Alergi  
Deskripsi

- Halaman ini digunakan oleh pasien untuk melihat hasil identifikasi alergi serta melihat

keterangan dari jenis alergi yang teridentifikasi.

- Terdapat tiga buah tombol, yaitu tombol Back yang digunakan untuk kembali ke tab control sebelumnya, tombol Close yang digunakan keluar dari sistem dan tombol Next digunakan untuk melanjutkan ke tab control berikutnya.

#### Event

```
Page_Load()
{
    Menampilkan hasil perhitungan identifikasi
    berdasarkan inputan user sebelumnya.
}
OnClick_listBoxHasilIdentifikasiAlergi()
{
    Menampilkan info alergi berdasarkan jenis
    alergi yang dipilih user dengan SQL
    Statement sebagai berikut:
SELECT INFO_ALERGI FROM ALERGI WHERE
ID_ALERGI=
' [LISTBOXHASILIDENTIFIKASIALERGI] ';
}
OnClick_buttonBackHasilIdentifikasiAlergi()
{
    Kembali ke tab control sebelumnya.
}
OnClick_buttonCloseHasilIdentifikasiAlergi()
{
    Keluar dari sistem identifikasi dan kembali
    ke Menu Utama.
}
```

Program Studi Teknik Informatika	DPPL – SiPaGi	107/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

OnClick_ buttonNextHasilIdentifikasiAlergi()
{
    Memunculkan tab control Pemilihan Gejala.
}

```

#### 4.8.5 Penanganan Alergi



Gambar 4.28 Rancangan Antarmuka Penanganan Alergi

#### Deskripsi

- Halaman ini digunakan oleh pasien untuk melihat penanganan dari alergi tertentu.
- Terdapat dua buah tombol, yaitu tombol Back yang digunakan untuk kembali ke tab control sebelumnya dan tombol Close yang digunakan keluar dari sistem.

Program Studi Teknik Informatika	DPPL – SiPaGi	108/ 109
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## Event

```
Page_Load()  
{  
    Menampilkan data penanganan dari alergi yang  
    dipilih user sebelumnya dengan SQL Statement  
    sebagai berikut:  
    SELECT KETERANGAN FROM PENANGANAN WHERE  
    ID_PENANGANAN=  
    ' [LISTBOXHASILIDENTIFIKASIALERGI] ';  
}  
OnClick_buttonBackPenangananAlergi()  
{  
    Kembali ke tab control sebelumnya.  
}  
OnClick_buttonClosePenangananAlergi()  
{  
    Keluar dari sistem identifikasi dan kembali  
    ke Menu Utama.  
}
```