BAB IV
IMPLEMENTASI dan PENGUJIAN

Pada tahap ini diterjemahkan kebutuhan pembangunan aplikasi ke dalam representasi perangkat lunak sesuai denganhasil analisis yang telah dilakukan. Setelah tahap implementasi maka dilakukan pengujian sistem yang dibangun sehingga dapat diketahui kekurangan-kekurangan pada aplikasi untuk selanjutnya dievaluasi.

4.1 Implementasi

Tahap implementasi sistem merupakan tahap menerjemahkan hasil perancangan yang telah dianalisis ke dalam bahasa yang dapat dimengerti oleh mesin, serta penerapan perangkat lunak pada keadaan sesungguhnya.

4.1.1 Implementasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan agar sistem berjalan dengan baik adalah sebagai berikut :

1. *Processor* : minimum Intel Atom Z520
2. RAM : minimum 1 G
3. *Harddisk* : minimum 250 Gb
4. Monitor : resolusi 1366 x 768
5. VGA : Intel GMA500
6. Aksesoris : *keyboard* dan *mouse*

4.1.2 Implementasi Perangkat Lunak

Perangkat lunak yang digunakan untuk mengoperasikan aplikasi ini adalah sebagai berikut :

1. Sistem Operasi Windows XP, Windows Vista, Windows 7, Windows 8.
2. *Web browser* seperti Google Chrome, Mozila Firefox, dll.

4.1.3 Implementasi Basis Data

Pembuatan basis data dilakukan dengan menggunakan MySQL adalah sebagai berikut :

Struktur Tabel t_aktivitas

```
CREATE TABLE IF NOT EXISTS `t_aktivitas` (
  `id_aktivitas` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `aktivitas` varchar(50) NOT NULL,
  `nilai_lk` float unsigned NOT NULL,
  `nilai_pr` float unsigned NOT NULL,
  PRIMARY KEY (`id_aktivitas`) )
```

Struktur Tabel t_bahan_makanan

```
CREATE TABLE IF NOT EXISTS `t_bahan_makanan` (
  `id_bahan_makanan` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `nama_bahan` varchar(50) NOT NULL,
  `id_kategori` int(10) unsigned NOT NULL,
  `id_jenis` int(10) unsigned NOT NULL,
  `takaran` varchar(30) NOT NULL,
  `gram` float unsigned NOT NULL,
  `kalori` float unsigned NOT NULL,
  `nilai_protein` float unsigned NOT NULL,
  `nilai_lemak` float unsigned NOT NULL,
  `nilai_karbo` float unsigned NOT NULL,
  PRIMARY KEY (`id_bahan_makanan`),
  KEY `t_bahan_makanan_id_kategori_fkey` (`id_kategori`),
  KEY `t_bahan_makanan_id_jenis_fkey` (`id_jenis`)
)
```

Struktur Tabel t_diet

```
CREATE TABLE IF NOT EXISTS `t_diet` (
  `id_diet` int(11) NOT NULL AUTO_INCREMENT,
  `diet` varchar(70) NOT NULL,
  `tujuan` text NOT NULL,
  `saran` text NOT NULL,
  PRIMARY KEY (`id_diet`)
)
```

Struktur tabel t_diet_larangan

```
CREATE TABLE IF NOT EXISTS `t_diet_larangan` (
  `id_diet` int(10) unsigned NOT NULL,
  `id_jenis` int(10) unsigned NOT NULL,
  KEY `id_diet_t_diet_id_diet_fkey` (`id_diet`),
  KEY `id_jenis_t_jenis_id_jenis_fkey` (`id_jenis`) )
```

Struktur Tabel t_jenis

```
CREATE TABLE IF NOT EXISTS `t_jenis` (
  `id_jenis` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `nama_jenis` varchar(50) NOT NULL,
  PRIMARY KEY (`id_jenis`)
)
```

Struktur Tabel t_kategori

```
CREATE TABLE IF NOT EXISTS `t_kategori` (
  `id_kategori` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `nama_kategori` varchar(50) NOT NULL,
```

```
  PRIMARY KEY (`id_kategori`)
)
```

Struktur Tabel t_pekerjaan

```
CREATE TABLE IF NOT EXISTS `t_pekerjaan` (
  `id_pekerjaan` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `pekerjaan` varchar(100) NOT NULL,
  `id_aktivitas` int(10) unsigned NOT NULL,
  PRIMARY KEY (`id_pekerjaan`),
  KEY `t_pekerjaan_id_aktivitas_fkey` (`id_aktivitas`)
)
```

Struktur Tabel t_penyakit

```
CREATE TABLE IF NOT EXISTS `t_penyakit` (
  `id_penyakit` int(11) NOT NULL AUTO_INCREMENT,
  `penyakit` varchar(50) NOT NULL,
  `id_diet` int(11) NOT NULL,
  PRIMARY KEY (`id_penyakit`),
  KEY `id_diet` (`id_diet`)
)
```

Struktur Tabel t_user

```
CREATE TABLE IF NOT EXISTS `t_user` (
  `username` varchar(20) NOT NULL,
  `password` varchar(20) NOT NULL
)
```

Struktur Tabel t_gejala
CREATE TABLE IF NOT EXISTS `t_gejala` (`id_gejala` int(11) NOT NULL AUTO_INCREMENT, `gejala` varchar(50) NOT NULL, PRIMARY KEY (`id_gejala`))

Struktur Tabel t_cf
CREATE TABLE IF NOT EXISTS `t_cf` (`id_penyakit` int(11) NOT NULL, `id_gejala` int(11) NOT NULL, `cf` double NOT NULL, KEY `id_penyakit` (`id_penyakit`,`id_gejala`), KEY `id_gejala` (`id_gejala`))

4.1.4 Implementasi Form

Implementasi *form* dilakukan untuk mengetahui setiap halaman program dan pengkodeannya. Implementasi form dibagi menjadi dua yaitu implementasi form admin dan pengguna.

4.1.4.1 Implementasi Form Admin

Implementasi *form* yang untuk Admin dapat dilihat pada tabel 4.1 berikut ini :

Tabel 4.1 Implementasi *form* admin

Menu	**Deskripsi**	**Nama File**
Halaman Login	Tampilan untuk masuk ke halaman admin	index.php

Halaman Admin	Halaman utama admin dengan wewenang mengelola semua pengetahuan sistem	home.php
Halaman Kelola Penyakit	Halaman yang digunakan untuk mengelola data penyakit	penyakit.php
Halaman Kelola Bahan Makanan	Halaman yang digunakan untuk mengelola data bahan pangan	bahan.php
Halaman Kelola Pekerjaan	Halaman yang digunakan untuk mengelola data pekerjaan	pekerjaan.php
Halaman Kelola Kategori	Halaman yang digunakan untuk mengelola data kategori	kategori.php
Halaman Kelola Jenis	Halaman yang digunakan untuk mengelola data jenis	jenis.php
Halaman Kelola Diet	Halaman yang digunakan untuk mengelola data diet	diet.php
Halaman Kelola Gejala	Halaman yang digunakan untuk mengelola data gejala	gejala.php
Halaman Kelola NilaiCF	Halaman yang digunakan untuk mengelola data cf	nilaicf.php
Log Out	Untuk keluar dari halaman admin	logout.php

4.1.4.2 Implementasi Form Pengguna

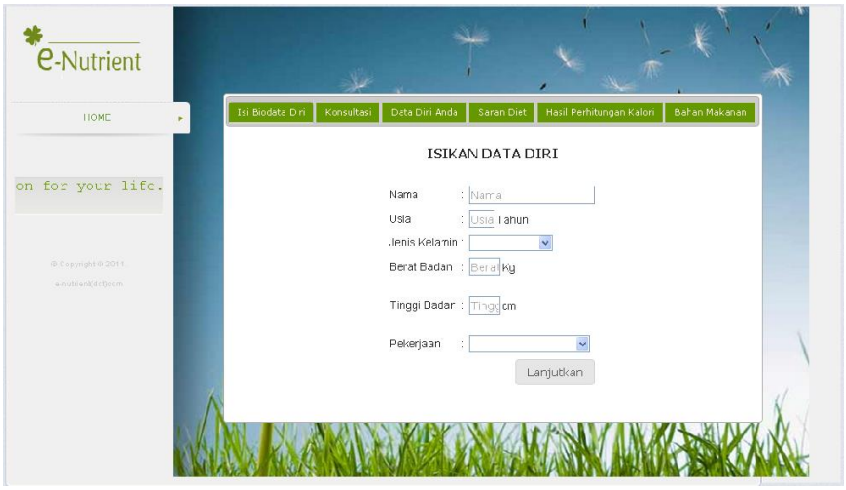
Implementasi form pada perangkat lunak yang ditujukan untuk pengguna dapat dilihat pada tabel 4.2 berikut ini :

Tabel 4.2 Impelmentasi *form* pengguna

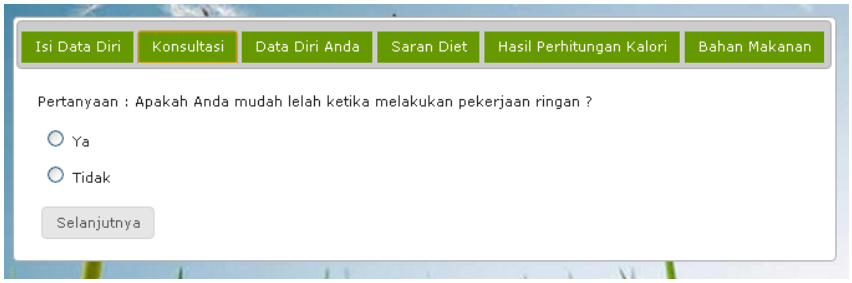
Menu Deskripsi	Deskripsi	Nama file
Halaman Utama	Halaman yang digunakan untuk berkonsultasi	index.php
Halaman Hasil	Halaman untuk melihat hasil dari konsultasi	pakar.php

4.2 Implementasi Antar Muka

Implementasi antar muka dilakukan dengan setiap halaman program yang dibuat dan pengkodeannya dalam bentuk file program. Berikut adalah implementasi antar muka pada aplikasi sistem pakar yang dibangun.



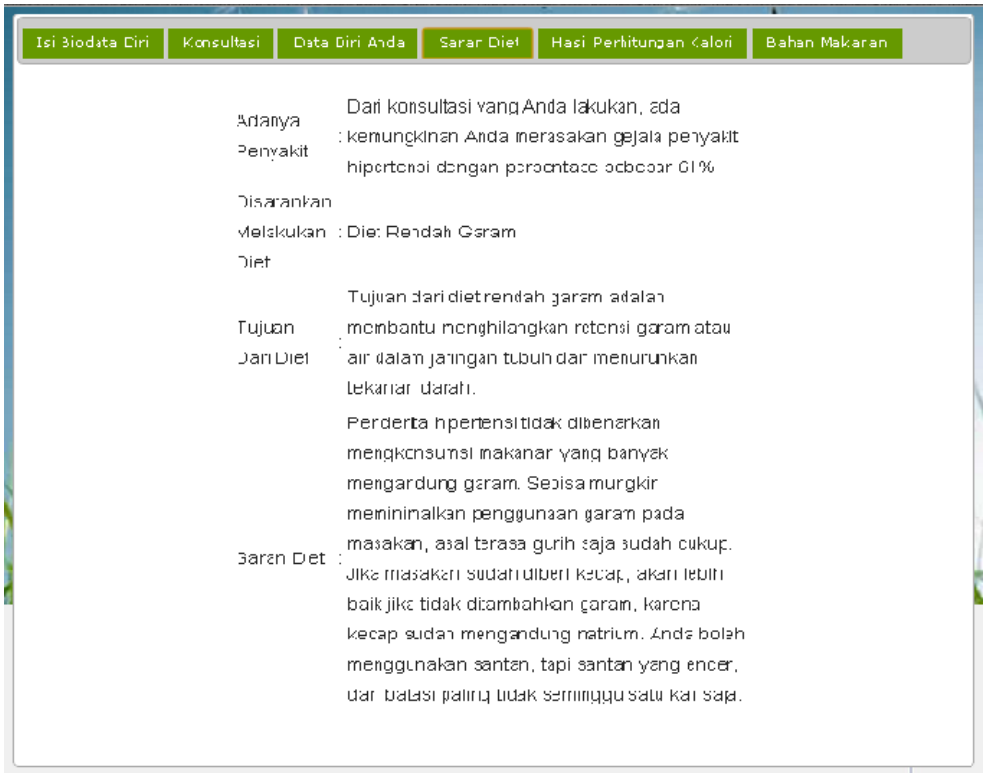
Gambar 4.1 Halaman utama isi data diri



Gambar 4.2 Halaman konsultasi



Gambar 4.3 Halaman hasil data diri



Gambar 4.4 Halaman saran diet



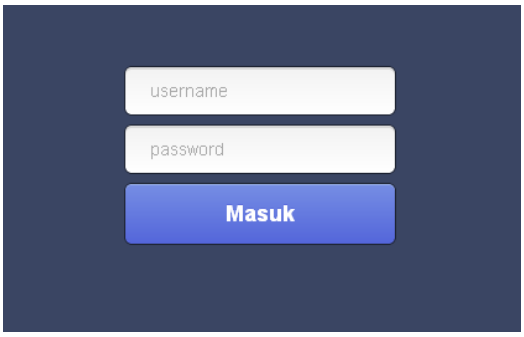
Gambar 4.5 Halaman hasil perhitungan kebutuhan kalori harian



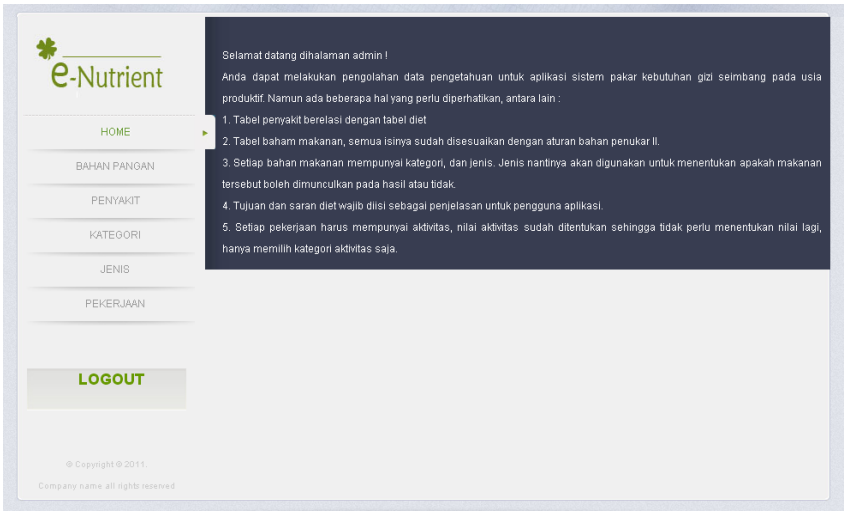
Bahan Makanan Untuk Sarapan		
Bahan	**Gram**	**Takaran**
tepung beras	50	8 sdm
daging sapi	35	1 ptg sdg
keju kacang tanah	15	1 sdm
brokoli	100	1 gls
peach	115	1 bh kcl
kacang almond	10	7 bj
Total Jumlah Kalori		**400**

Bahan Makanan Untuk Selingan Pertama		
Bahan	**Gram**	**Takaran**
bihun	50	1/2 gelas
daging sapi	35	1 ptg sdg
susu skim bubuk	20	4 sdm
kacang almond	10	7 bj
Total Jumlah Kalori		**275**

Gambar 4.6 Halaman hasil acak bahan makanan



Gambar 4.7 Tampilan menu login admin



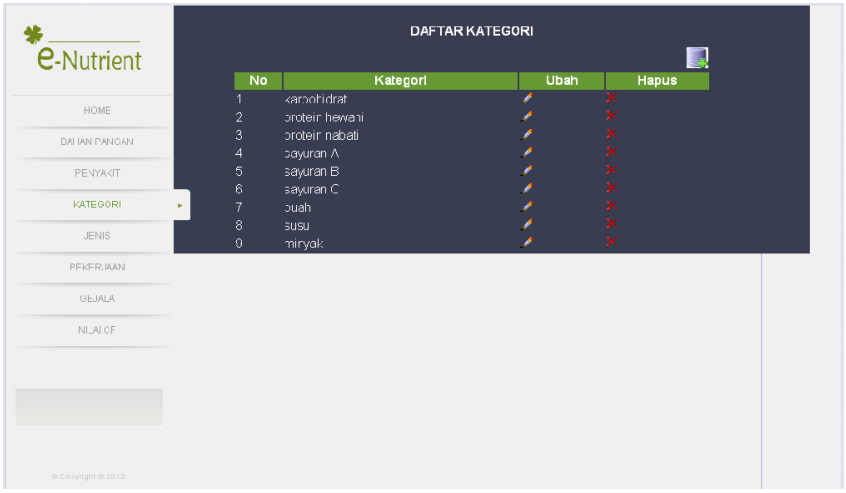
Gambar 4.8 Halaman admin



Gambar 4.9 Halaman kelola bahan makanan



Gambar 4.10 Halaman kelola penyakit



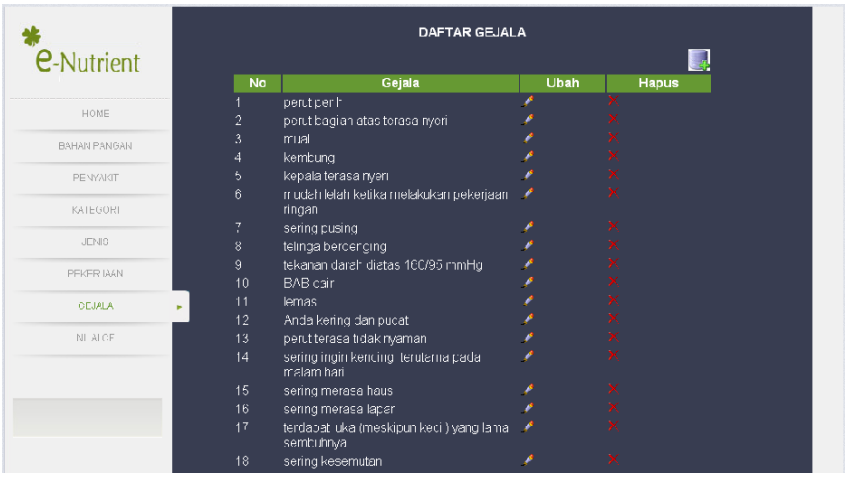
Gambar 4.11 Halaman kelola kategori



Gambar 4.12 Halaman kelola jenis



Gambar 4.13 Halaman kelola pekerjaan



Gambar 4.14 Halaman kelola gejala



Gambar 4.15 Halaman kelola nilai cf

4.3 Pengujian Perangkat Lunak

Pengujian merupakan hal yang penting agar diketahui kekurangan dan kesalahan pada perangkat lunak yang dibangun. Pengujian dilakukan untuk mengetahui apakah sistem yang dibanguns sudah memenuhi kriteria dari tujuan dibangunnya sistem.

Pengujian perangkat lunak ini menggunakan metode *black box*. Metode *black box* berfokus pada fungsional perangkat lunak. Adapun pengujian *black box* ini dibagi menjadi 2 pengujian, yaitu :

1. Pengujian Aplha
2. Pengujian Beta

Rencana pengujian alpha dan berta dapat dilihat pada tabel 4.1 berikut ini :

Tabel 4.3 Tabel rencana pengujian

Kelas Uji	Butir Uji	Jenis Pengujian
Login Admin	Verivikasi Username	*Blackbox*
	Verivikasi Password	
Pengolahan Data Penyakit	Tambah Data Penyakit	*Blackbox*
	Ubah Data Penyakit	

	Hapus Data Penyakit	
Pengolahan Data Diet	Tambah Data Diet	*Blackbox*
	Ubah Data Diet	
	Hapus Data Diet	
Pengolahan Data Kategori	Tambah Data Kategori	*Blackbox*
	Ubah Data Kategori	
	Hapus Data Kategori	
Pengolahan Data Jenis	Tambah Data Jenis	*Blackbox*
	Ubah Data Jenis	
	Hapus Data Jenis	
Pengolahan Data Pekerjaan	Tambah Data Pekerjaan	*Blackbox*
	Ubah Data Pekerjaan	
	Hapus Data Pekerjaan	
Pengolahan Data Gejala	Tambah Data Gejala	*Blackbox*
	Ubah Data Gejala	
	Hapus Data Gejala	
Pengolahan Data CF	Tambah Data CF	*Blackbox*
	Ubah Data CF	
	Hapus Data CF	

4.3.1 Kasus dan Hasil Pengujian Alpha

Berdasarkan pada rencana pengujian, maka dapat dilakukan pengujian Alpha pada aplikasi sistem pakar yang dibangun. Pengujian yang dilakukan dapat dilihat pada tabel-tabel berikut.

4.3.1.1 Pengujian *Login* Admin

Berikut ini adalah pengujian yang dilakukan pada *login* admin.

Tabel 4.4 Pengujian login admin

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Username dan *Password* benar	Data *username* dan *password* diterima dan berhasil masuk ke menu admin	Tombol *Login* berfungsi dengan baik dan muncul tampilan menu pakar sesuai yang diharapkan	[√] Berhasil [ ] Gagal
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
username atau *password* salah atau kosong	Tidak dapat masuk ke menu admin, kembali ke halaman login.	Tidak dapat masuk ke menu admin, kembali ke halaman login.	[√] Berhasil [ ] Gagal

4.3.1.2 PengujianPengolahan Data Penyakit

Tabel 4.5 Pengujian tambah data penyakit

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data penyakit. Contoh : Penyakit : Diare Diet : Rendah Serat	Menambahkan data penyakit..	Data penyakit bertambah.	[√] Berhasil [ ] Gagal

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data penyakit.	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Berhasil [ ] Gagal

Tabel 4.6 Pengujian ubah data penyakit

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengubah data penyakit. Contoh : Penyakit lama : Diare Penyakit baru : Kolesterol	Data penyakit berubah.	Data penyakit berubah.	[√] Berhasil [ ] Gagal
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data penyakit.	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Berhasil [ ] Gagal

4.3.1.3 PengujianPengolahan Data Diet

Tabel 4.7 Pengujian tambah data diet

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data diet. Contoh : Penyakit : Diare Diet : Rendah Serat	Menambahkan data diet..	Data diet bertambah.	[√] Berhasil [ ] Gagal
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data diet.	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Berhasil [ ] Gagal

Tabel 4.8 Pengujian ubah data diet

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengubah data diet. Contoh : diet lama : Rendah serat diet baru : Tinggi Serat	Data diet berubah.	Data diet berubah.	[√] Berhasil [ ] Gagal

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data diet.	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Berhasil [ ] Gagal

4.3.1.4 PengujianPengolahan Data Kategori

Tabel 4.9 Pengujian tambah data kategori

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data kategori. Contoh : kategori : karbohidrat	Menambahkan data kategori..	Data kategori bertambah.	[√] Berhasil [ ] Gagal
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data kategori.	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Berhasil [ ] Gagal

Tabel 4.10 Pengujian ubah data kategori

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengubah data kategori.	Data kategori berubah.	Data kategori berubah.	[√] Berhasil [ ] Gagal

Contoh : kategori lama : berlemak kategori baru : berserat			
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data kategori.	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Berhasil [ ] Gagal

4.3.1.5 PengujianPengolahan Data Jenis

Tabel 4.11 Pengujian tambah data jenis

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data jenis. Contoh : jenis : beserat	Menambahkan data jenis..	Data jenis bertambah.	[√] Berhasil [ ] Gagal
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data jenis.	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Berhasil [ ] Gagal

Tabel 4.12 Pengujian ubah data jenis

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengubah data jenis. Contoh : jenis lama : beserat jenis baru : berlemak	Data jenis berubah.	Data jenis berubah.	[√] Berhasil [ ] Gagal
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data jenis.	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Diterima [ ] Ditolak

4.3.1.6 PengujianPengolahan Data Gejala

Tabel 4.13 Pengujian tambah data gejala

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data gejala. Contoh : gejala : mual	Menambahkan data gejala..	Data gejala bertambah.	[√] Berhasil [ ] Gagal
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**

Mengosongkan form pengisian data gejala.	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Berhasil [ ] Gagal

Tabel 4.14 Pengujian ubah data gejala

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengubah data gejala. Contoh : gejala lama : mual gejala baru : pusing	Data gejala berubah.	Data gejala berubah.	[√] Berhasil [ ] Gagal
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data gejala.	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Berhasil [ ] Gagal

4.3.1.7 PengujianPengolahan Data CF

Tabel 4.15 Pengujian tambah data cf

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data cf. Contoh : Penyakit : Maag	Menambahkan data cf.	Data cf bertambah.	[√] Berhasil [ ] Gagal

Gejala : perut perih CF : 0.5			
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data cf.	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Berhasil [ ] Gagal
Mengisi nilai cf dengan bilangan bulat. Contoh: CF: 8	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Berhasil [ ] Gagal

Tabel 4.16 Pengujian ubah data cf

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data cf. Contoh : Penyakit lama : Maag Gejala lama : perut perih CF lama: 0.5 Penyakit baru: Hipertensi Gejala baru :	Menambahkan data cf.	Data cf bertambah.	[√] Berhasil [ ] Gagal

Pusing CF baru : 0.3			
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data cf.	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Berhasil [ ] Gagal
Mengisi nilai cf dengan bilangan bulat. Contoh: CF: 8	Muncul pesan penambahan data gagal !	Muncul pesan penambahan data gagal !	[√] Berhasil [ ] Gagal

4.3.1.8 Pengujian Halaman Pengunjung

Pada pengujian pada halaman pengunjung dilakukan untuk mengetahui apakah form sudah berfungsi dengan baik sehingga data yang diterima untuk diolah sudah benar.

Tabel 4.17 Pengujian isi form data diri

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : rikha Usia : 21 Jenis Kelamin : perempuan Berat : 40 Tinggi : 152 Pekerjaan :	Data diterima.	Data diterima.	[√] Berhasil [ ] Gagal

Mahasiswa			
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data.	Muncul pesan “data tidak boleh kosong !”	Muncul pesan ”data tidak boleh kosong !”	[√] Berhasil [ ] Gagal
Mengisi Berat atau Tinggi dengan menggunakan huruf.	Muncul pesan “hanya boleh memasukkan angka!”	Muncul pesan “hanya boleh memasukkan angka!”	[√] Berhasil [ ] Gagal
Mengosongkan Jenis Kelamin	Muncul pesan “jenis kelamin tidak boleh kosong!”	Muncul pesan “jenis kelamin tidak boleh kosong!”	[√] Berhasil [ ] Gagal
Mengosongkan Pekerjaan	Muncul pesan “pekerjaan tidak boleh kosong!”	Muncul pesan “pekerjaan tidak boleh kosong!”	[√] Berhasil [ ] Gagal

Tabel 4.18 Pengujian halaman konsultasi

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukkan gejala. Contoh : Kepala pusing dan mudah lelah.	Data diterima dan sistem mengajukan pertanyaan gejala.	Data diterima dan sistem mengajukan pertanyaan gejala.	[√] Berhasil [ ] Gagal
Kasus dan Hasil Uji (Data Salah)			

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukkan gejala yang tidak ada pada database.	Sistem merespon dengan “data tidak ada pada database, sebutkan gejala lainnya.”	Sistem merespon dengan “data tidak ada pada database, sebutkan gejala lainnya.”	[√] Berhasil [ ] Gagal

4.3.1.9 Kesimpulan Pengujian Alpha

Berdasarkan hasil pengujian sistem yang telah dilakukan secara keseluruhan kesimpulan bahwa pada proses web ini telah melalui tahap perbaikan dan sudah di maksimalkan terhadap proses-proses tersebut dan secara fungsional sistem sudah dapat digunakan dan menghasilkan output yang diharapkan.

4.3.2 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan dengan tujuan untuk mengetahui sejauh mana kualitas system. Untuk itu dalam pengujian beta dilakukan penelitian dengan cara memberikan kuisioner pada calon pengguna system. Pengujian dilakukan pada lingkungan yang berbeda, yaitu :

1. Pakar gizi Nursat Fauziah Begum . Amd

   Penelitian dilakukan untuk mengetahui sejauh mana system yang dibangun dapat menjadi alternatif penyelesaian masalah yang telah dikemukakan diawal.

2. Lingkungan masyarakat umum

   Penelitian di lingkungan masyarakat umum khususnya yang sudah terbiasa dengan teknologi internet, dilakukan untuk mengtahui sejauh mana system yang dibangun dapat memberikan kemudahan bagi masyarakat untuk mendapatkan informasi kebutuhan gizinya.

4.3.2.1 Pengujian Beta untuk pakar (Kuesioner)

Pengujian untuk pakar dilakukan demi mengetahui apakah system sudah bekerja dengan baik dan benar, sehingga hasil yang dikeluarkan sudah benar adanya. Pada pengujian ini pakar melakukan pengujian terhadap system dan mengisi kuesioner untuk pakar (hasil dari pengujian ini dapat dilihat pada lampiran). Kuesioner terdiri dari empat pertanyaan, yaitu :

1. Apakah aplikasi sistem pakar ini bekerja selayaknya seorang ahli gizi ?
2. Apakah perhitungan yang digunakan sudah benar sebagaimana mestinya ?
3. Apakah saran yang diberikan sudah sesuai ?
4. Apakah susunan bahan makanan yang diberikan benar ?

4.3.2.2 Pengujian Beta untuk masyarakat umum (Kuesioner)

Kuisioner terdiri dari lima pertanyaan (contoh kuesioner dapat dilihat pada lampiran) dengan menggunakan skala likert dengan skala satu sampai lima, dengan jumlah responden sebanyak 30 responden, dengan ketentuan pertanyaan skala sebagai berikut :

1. Apakah aplikasi sistem pakar ini mudah digunakan ?

Tabel 4.19 Jawaban pertanyaan no 1

Pilihan Jawaban	Keterangan
A	Sangat Setuju
B	Setuju
C	Cukup Setuju
D	Kurang Setuju
E	Tidak Setuju

2. Apakah pertanyaan yang diajukan mudah dipahami ?

Tabel 4.19 Jawaban pertanyaan no 2

Pilihan Jawaban	Keterangan
A	Sangat Setuju
B	Setuju
C	Cukup Setuju
D	Kurang Setuju
E	Tidak Setuju

3. Apakah aplikasi sistem pakar ini membantu Anda dalam menentukan jenis diet yang cocok ?

Tabel 4.20 Jawaban pertanyaan no 3

Pilihan Jawaban	Keterangan
A	Sangat Setuju
B	Setuju
C	Cukup Setuju
D	Kurang Setuju
E	Tidak Setuju

4. Apakah informasi bahan makanan beserta takaran yang diberikan membantu Anda dalam memilih makanan ?

Tabel 4.21 Jawaban pertanyaan 4

Pilihan Jawaban	Keterangan
A	Sangat Setuju
B	Setuju
C	Cukup Setuju
D	Kurang Setuju
E	Tidak Setuju

5. Apakah informasi yang diberikan mudah diterima ?

Tabel 4.22 Jawaban pertanyaan 5

Pilihan Jawaban	Keterangan
A	Sangat Setuju
B	Setuju
C	Cukup Setuju
D	Kurang Setuju
E	Tidak Setuju

Berdasarkan hasil kuesioner tersebut, dapat dicari masing-masing jawaban dengan menggunakan rumus dibawah ini :

$$Y = \frac{X}{skorideal} \times 100\%$$

Keterangan :

X	=	Jumlah frekuensi dikalikan dengan nilai kategori jawaban ($\sum$ = N x R)
N	=	Nilai dari setiap jawaban
R	=	Frekuensi
Skor Ideal	=	Nilai tertinggi dikalikan dengan jumlah sampel, (5x30=150)
Y	=	Persentase nilai yang dicari

Setelah diperoleh nilai Y maka interpretasi nilai dapat dilihat sebagai berikut :

Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju	
0	20%	40%	60%	80%	100%

Berikut adalah hasil prosentase dari masing-masing jawaban dari lima pertanyaan yang sudah dihitung nilainya menggunakan rumus diatas.

1. Apakah aplikasi sistem pakar ini mudah digunakan ?

   Data kuesioner pertanyaan nomor 1 dapat dilihat pada tabel 4.23 berikut ini.

Tabel 4.23 Hasil kuesioner pertanyaan no 1

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
1	Sangat Setuju	5	5	25
	Setuju	4	14	56
	Cukup Setuju	3	8	24
	Kurang Setuju	2	3	6

	Tidak Setuju	1	0	0
Jumlah			30	111

$$Y = \frac{111}{30 \times 5} \times 100\ \% = 74\ \%$$

Berdasarkan hasil prosentase diatas, maka dapat disimpulkan bahwa penilaian sistem pakar ini mudah digunakan oleh pengguna adalah 74 % atau bisa dikategorikan sebagai setuju. Berikut adalah kriteria intepretasi skor :

Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju	
0	20%	40%	60% 74 %	80%	100%

2. Apakah pertanyaan yang diajukan mudah dipahami ?

   Data kuesioner pertanyaan nomor 2 dapat dilihat pada tabel 4.24 berikut ini.

Tabel 4.24 Hasil kuesioner pertanyaan no 2

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
2	Sangat Setuju	5	3	15
	Setuju	4	19	76
	Cukup Setuju	3	8	24
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah			30	115

$$Y = \frac{115}{30 \times 5} \times 100\ \% = 77\ \%$$

Berdasarkan hasil prosentase diatas, maka dapat disimpulkan bahwa penilaian pertanyaan yang diajukan sistem mudah dipahami oleh pengguna adalah 77 % atau bisa dikategorikan sebagai setuju. Berikut adalah kriteria intepretasi skor :

Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju	
0	20%	40%	60% 77 %	80%	100%

3. Apakah aplikasi sistem pakar ini membantu Anda dalam menentukan jenis diet yang cocok ?

   Data kuesioner pertanyaan nomor 3 dapat dilihat pada tabel 4.25 berikut ini.

Tabel 4.25 Hasil kuesioner pertanyaan no 3

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
3	Sangat Setuju	5	10	50
	Setuju	4	12	48
	Cukup Setuju	3	6	18
	Kurang Setuju	2	2	4
	Tidak Setuju	1	0	0
Jumlah			30	120

$$Y = \frac{120}{30 \times 5} \times 100\ \% = 80\ \%$$

Berdasarkan hasil prosentase diatas, maka dapat disimpulkan bahwa penilaian sistem pakar ini membantu pengguna dalam menentukan jenis diet adalah adalah 80 % atau bisa dikategorikan sebagai sangat setuju. Berikut adalah kriteria intepretasi skor :

Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju	
0	20%	40%	60%	80%	100%

4. Apakah informasi bahan makanan beserta takaran yang diberikan membantu Anda dalam memilih makanan ?

   Data kuesioner pertanyaan nomor 4 dapat dilihat pada tabel 4.26berikut ini.

Tabel 4.26 Hasil kuesioner pertanyaan no 4

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
4	Sangat Setuju	5	4	20
	Setuju	4	16	64
	Cukup Setuju	3	19	57
	Kurang Setuju	2	1	2
	Tidak Setuju	1	0	0
Jumlah			30	143

$$Y = \frac{143}{30 \times 5} \times 100\ \% = 95\ \%$$

Berdasarkan hasil prosentase diatas, maka dapat disimpulkan bahwa penilaian sistem pakar ini membantu pengguna dalam memilih bahan makanan adalah adalah

95 % atau bisa dikategorikan sebagai sangat setuju. Berikut adalah kriteria intepretasi skor:

Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju	
0	20%	40%	60%	80% 95 %	100%

5. Apakah informasi yang diberikan mudah diterima ?

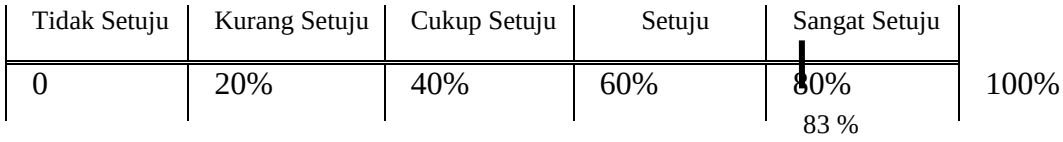
   Data kuesioner pertanyaan nomor 5 dapat dilihat pada tabel 4.27 berikut ini.

Tabel 4.27 Hasil kuesioner pertanyaan no 5

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
5	Sangat Setuju	5	8	40
	Setuju	4	18	72
	Cukup Setuju	3	4	12
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah			30	124

$$Y = \frac{124}{30 \times 5} \times 100\ \% = 83\ \%$$

Berdasarkan hasil prosentase diatas, maka dapat disimpulkan bahwa penilaian sistem pakar ini membantu pengguna dalam menentukan jenis diet adalah adalah 83 % atau bisa dikategorikan sebagai sangat setuju. Berikut adalah kriteria intepretasi skor :



4.3.2.3 Kesimpulan Pengujian *Beta*

Berdasarkan hasil prosentase pengujian beta yang dilakukan kepada narasumber dan lingkungan masyarakat umum, maka didapatkan bahwa sistem pakar ini telah telah sesuai dengan tujuan yaitu memberikan informasi kebutuhan gizi yang sesuai dengan kondisi pengguna, saran diet dan contoh menu bahan makanan beserta takaran.