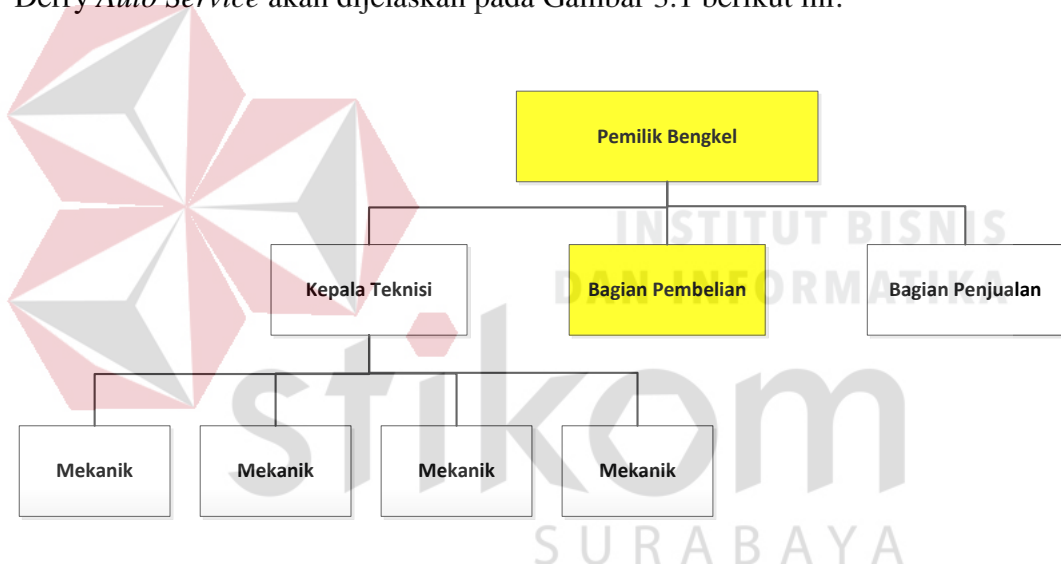


## BAB III

### ANALISIS DAN PERANCANGAN SISTEM

#### 3.1 Identifikasi Masalah, Peluang, dan Tujuan

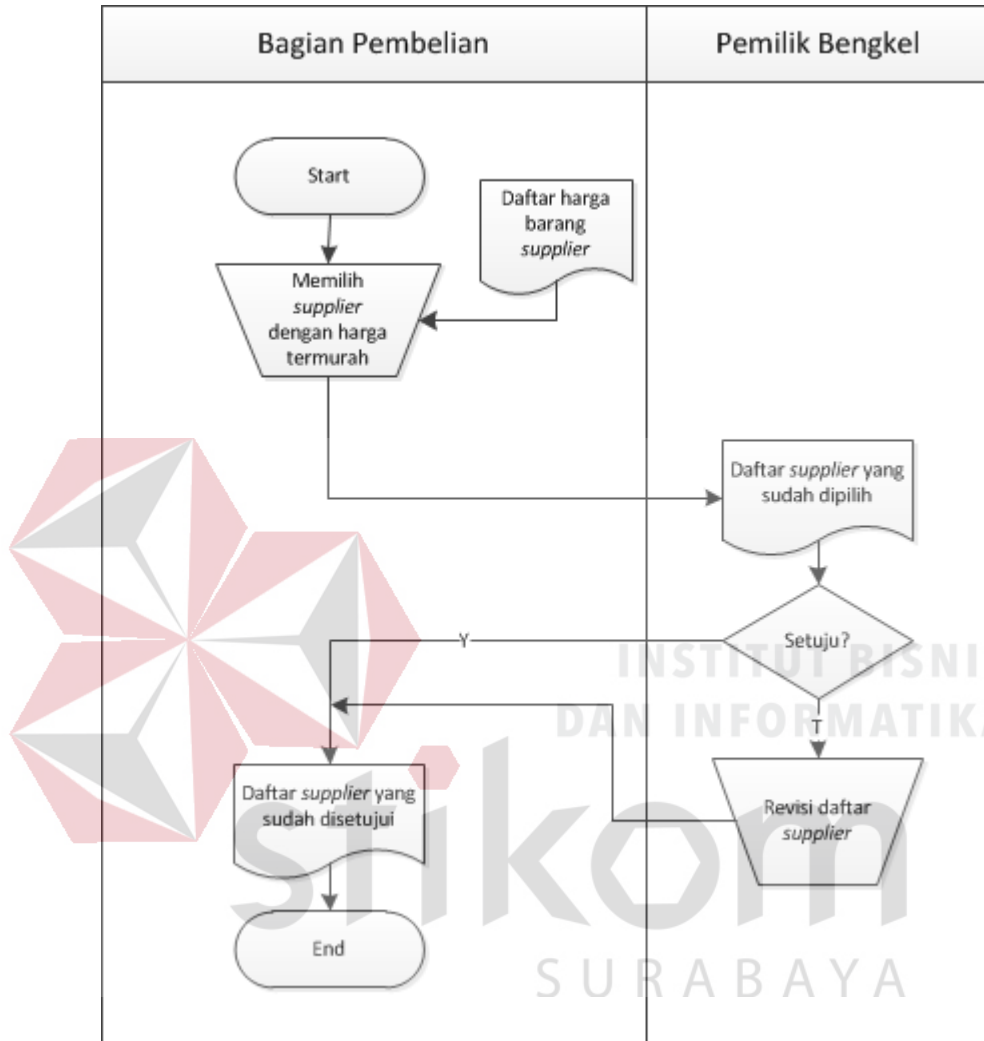
Tahap pertama yaitu melakukan identifikasi masalah dan melakukan pengumpulan data dengan cara wawancara pemilik bengkel *Derry Auto Service* dan observasi. Setiap organisasi atau perusahaan pasti mempunyai struktur organisasi begitu pula dengan *Derry Auto Service*. Adapun struktur organisasi dari *Derry Auto Service* akan dijelaskan pada Gambar 3.1 berikut ini:



Gambar 3.1 Struktur Organisasi *Derry Auto Service*

Dari hasil wawancara diperoleh permasalahan pada bulan Januari sampai dengan Mei 2013 mendapatkan 32 keluhan pelanggan dari 114 transaksi. Hal ini berarti jumlah keluhan pelanggan sebesar 28,07% dari jumlah transaksi. Pelanggan mengeluh karena kendaraan yang telah diperbaiki *Derry Auto Service* mengalami kerusakan yang sama setelah beberapa bulan digunakan. Salah satu penyebab kerusakan karena kualitas *spare part* yang diganti tidak baik. Hal ini terjadi karena pada saat melakukan pemilihan *supplier* untuk pembelian spare

part, Derry *Auto Service* memilih daftar harga barang yang paling murah. Untuk lebih jelasnya lihat Gambar 3.2.



Gambar 3.2 *Document Flow* Memilih *Supplier*

Penjelasan *document flow* memilih *supplier* pada Gambar 3.2 bisa dilihat pada Tabel 3.1.

Tabel 3.1 Penjelasan *Document Flow* Memilih *Supplier*

Nama Proses	Keterangan	Output
Memilih <i>supplier</i> dengan harga	Bagian pembelian memilih <i>supplier</i> dengan harga termurah dari daftar barang tiap-tiap <i>supplier</i> lalu diberikan kepada pemilik bengkel	Daftar <i>supplier</i> yang sudah

<b>Nama Proses</b>	<b>Keterangan</b>	<b>Output</b>
termurah		dipilih
<i>Decision</i> setuju	Jika pemilik bengkel menyetujui daftar <i>supplier</i> yang sudah dipilih, maka daftar <i>supplier</i> tersebut ditandatangani diberikan kepada bagian pembelian. Jika tidak, maka pemilik bengkel akan merevisi daftar <i>supplier</i>	-
Proses revisi daftar <i>supplier</i>	Pemilik bengkel merevisi daftar <i>supplier</i> yang sudah dipilih oleh bagian pembelian. Jika sudah selesai merevisi, daftar <i>supplier</i> yang sudah disetujui diberikan kepada bagian pembelian	Daftar <i>supplier</i> yang sudah disetujui

Setelah menjelaskan *document flow* memilih *supplier*, Tabel 3.2 merupakan tugas dan tanggung jawab dari pemilik bengkel dan bagian pembelian.

Tabel 3.2 Tugas dan Tanggung Jawab Pemilik Bengkel dan Bagian Pembelian

<b>Pengguna</b>	<b>Tugas</b>	<b>Tanggung Jawab</b>
Bagian Pembelian	<ol style="list-style-type: none"> <li>1. Mencatat data barang</li> <li>2. Mencatat data <i>supplier</i></li> <li>3. Mencatat data pembelian</li> <li>4. Melakukan pemilihan <i>supplier</i></li> </ol>	<ol style="list-style-type: none"> <li>1. Bertanggung jawab terhadap proses pencatatan data barang</li> <li>2. Bertanggung jawab terhadap proses pencatatan data <i>supplier</i></li> <li>3. Bertanggung jawab terhadap proses pencatatan data pembelian</li> </ol>
Pemilik Bengkel	<ol style="list-style-type: none"> <li>1. Menyetujui daftar <i>supplier</i></li> <li>2. Revisi daftar <i>supplier</i></li> </ol>	Bertanggung jawab terhadap daftar <i>supplier</i> yang disetujui

Setelah mengetahui tugas pemilik bengkel dan bagian pembelian dalam pemilihan *supplier*, tugas-tugas tersebut akan dianalisis agar bisa mengurangi permasalahan keluhan pelanggan karena kualitas *spare part* yang diganti tidak baik.

### 3.1.1 Analisis tugas bagian pembelian

Bagian pembelian melakukan pemilihan *supplier* hanya berdasarkan satu kriteria yaitu kriteria harga barang yang paling murah karena perusahaan Derry

*Auto Service* tidak memiliki sistem yang mampu melakukan pemilihan *supplier* dengan kriteria yang beragam.

### **3.1.2 Analisis tugas pemilik bengkel**

Pemilik bengkel melihat daftar *supplier* yang sudah dipilih oleh bagian pembelian. Jika menyetujui daftar *supplier* yang sudah dipilih oleh bagian pembelian, maka daftar *supplier* tersebut ditandatangani dan diberikan kepada bagian pembelian. Jika tidak, maka pemilik bengkel akan merevisi daftar *supplier*. Daftar *supplier* yang disetujui pemilik bengkel adalah daftar *supplier* yang dipilih berdasarkan harga barang yang paling murah.

Setelah melakukan analisis tugas masing-masing pengguna, maka ditemukan beberapa kelemahan yang harus diselesaikan. Solusi yang diberikan yaitu membuat sistem pendukung keputusan pemilihan *supplier* dengan metode AHP yang akan memudahkan pengambilan keputusan pemilihan *supplier* dengan kriteria yang beragam.

## **3.2 Menentukan Kebutuhan Informasi**

Sistem Pendukung Keputusan Pemilihan *Supplier* pada *Derry Auto Service* dengan metode AHP membutuhkan data kriteria, data *intensity rating*, data barang, data *supplier*, data pembelian, dan data penilaian barang. Berikut ini penjelasan mengenai data-data yang dibutuhkan oleh pemilik bengkel dan bagian pembelian:

### **3.2.1 Kebutuhan informasi untuk pemilik bengkel**

Berikut ini adalah kebutuhan informasi yang dibutuhkan oleh pemilik bengkel:

a. Data kriteria

Data kriteria merupakan data yang mengungkapkan kriteria apa saja yang digunakan perusahaan dalam menentukan *supplier* untuk pembelian *spare part*.

b. Data *intensity rating*

Data *intensity rating* merupakan skala intensitas yang berbeda untuk tiap kriteria. Setiap skala dapat mempunyai sifat spesifik untuk tiap kriteria.

Dengan sistem yang baru, pemilik bengkel menggunakan fungsi-fungsi sebagai berikut:

1. Menentukan kriteria pemilihan *supplier*
2. Menentukan bobot penilaian kriteria
3. Menilai produk *supplier*

### 3.2.2 Kebutuhan informasi untuk bagian pembelian

Berikut ini adalah kebutuhan informasi yang dibutuhkan oleh bagian pembelian:

a. Data barang

Data barang merupakan data-data yang berisi informasi mengenai *spare part* apa saja yang dibeli perusahaan dalam memenuhi kegiatan operasional.

b. Data *supplier*

Data *supplier* merupakan data-data yang berisi informasi mengenai *supplier* yang digunakan dalam membeli *spare part*.

c. Data pembelian

Data pembelian merupakan data-data yang berisi informasi mengenai pembelian yang dilakukan perusahaan.

Dengan sistem yang baru, bagian pembelian menggunakan fungsi-fungsi sebagai berikut:

1. Mencatat data barang
2. Mencatat data *supplier*
3. Mencatat data pembelian
4. Melakukan pemilihan *supplier*

Sesuai dengan hasil menentukan kebutuhan informasi yang sudah dijabarkan diatas, terdapat beberapa fungsi yang tidak perlu dilakukan dan penambahan fungsi baru pada sistem yang akan dibuat. Perbandingan antara fungsi-fungsi saat ini dengan fungsi-fungsi baru dapat dilihat pada Tabel 3.3.

Tabel 3.3 Perbandingan Fungsi-Fungsi Saat Ini dengan Fungsi-Fungsi Baru

Pengguna	Fungsi		Keterangan
	Saat Ini	Baru	
Pemilik Bengkel	Menyetujui daftar <i>supplier</i>	-	Fungsi ini diganti dengan tiga fungsi, yaitu: 1. Menentukan kriteria pemilihan <i>supplier</i> 2. Menentukan bobot penilaian kriteria 3. Menilai produk <i>supplier</i>
	Revisi daftar <i>supplier</i>	-	Fungsi ini diganti dengan menilai produk <i>supplier</i>
	-	Menentukan kriteria pemilihan <i>supplier</i>	Fungsi ini menggantikan fungsi menyetujui daftar <i>supplier</i>
	-	Menentukan bobot penilaian kriteria	Fungsi ini menggantikan fungsi menyetujui daftar <i>supplier</i>
	-	Menilai produk <i>supplier</i>	Fungsi ini menggantikan fungsi menyetujui daftar <i>supplier</i> dan revisi daftar <i>supplier</i>
Bagian Pembelian	Mencatat data barang	-	Fungsi mencatat data barang dilakukan secara manual. Fungsi ini diganti dengan mencatat data barang pada database

Pengguna	Fungsi		Keterangan
	Saat Ini	Baru	
	Mencatat data <i>supplier</i>	-	Fungsi mencatat data <i>supplier</i> dilakukan secara manual. Fungsi ini diganti dengan mencatat data <i>supplier</i> pada database
	Mencatat data pembelian	-	Fungsi mencatat data pembelian dilakukan secara manual. Fungsi ini diganti dengan mencatat data pembelian pada database
	Melakukan pemilihan <i>supplier</i>	-	Fungsi melakukan pemilihan <i>supplier</i> ini dilakukan dengan melihat harga termurah. Fungsi ini diganti dengan melakukan pemilihan <i>supplier</i> dengan melihat nilai akhir barang tertinggi
	-	Mencatat data barang	Fungsi mencatat data barang pada database digunakan untuk menggantikan fungsi lama yang dilakukan secara manual.
	-	Mencatat data <i>supplier</i>	Fungsi mencatat data <i>supplier</i> pada database digunakan untuk menggantikan fungsi lama yang dilakukan secara manual.
	-	Mencatat data pembelian	Fungsi mencatat data pembelian pada database digunakan untuk menggantikan fungsi lama yang dilakukan secara manual.
	-	Melakukan pemilihan <i>supplier</i>	Fungsi melakukan pemilihan <i>supplier</i> ini dilakukan dengan melihat nilai akhir barang tertinggi yang digunakan untuk menggantikan fungsi lama yang dilakukan dengan melihat harga termurah.

### 3.3 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem yaitu melakukan identifikasi kebutuhan fungsional dan nonfungsional sistem pendukung keputusan pemilihan *supplier* untuk dapat menentukan modul-modul pada sistem yang akan dibuat.

### 3.3.1 Pemilik bengkel

Pemilik bengkel memiliki 3 fungsi yaitu menentukan kriteria pemilihan supplier, menentukan bobot penilaian kriteria, dan menilai produk *supplier*.

Adapun penjelasan setiap fungsi dapat dilihat sebagai berikut:

a Fungsi menentukan kriteria pemilihan *supplier*

Keterangan dari fungsi menentukan kriteria pemilihan *supplier* oleh pemilik bengkel dapat dilihat pada Tabel 3.4.

Tabel 3.4 Keterangan Fungsi Menentukan Kriteria Pemilihan *Supplier*

Nama Fungsi	Menentukan Kriteria Pemilihan <i>Supplier</i>	
Stakeholder	Pemilik bengkel	
Deskripsi	Menentukan kriteria pemilihan barang dari <i>supplier</i> beserta <i>intensity rating</i>	
Kondisi Awal	-	
Alur Normal	Aksi Pengguna	Respon Sistem
	Mengisi data kriteria	
	1. <i>User</i> memilih menu kriteria.	1. Sistem menampilkan <i>form</i> kriteria.
	2. <i>User</i> melakukan pengisian data kriteria di setiap field yang ada, lalu menekan tombol <i>insert</i> .	2. Sistem melakukan pengecekan terhadap inputan masing-masing field apakah terjadi pengisian field yang tidak tepat, seperti: field nama kriteria diisi dengan angka, dst. 2.1. Jika ada kesalahan input, maka sistem akan memberitahukan kesalahan yang terjadi. 2.2. Sistem akan menyimpan data kriteria tersebut ke dalam database. Setelah data berhasil disimpan akan muncul pesan bahwa proses simpan berhasil dilakukan. Kemudian sistem menampilkan data kriteria yang telah



Nama Fungsi	Menentukan Kriteria Pemilihan <i>Supplier</i>	
		dicatat.
	<b>Mengisi data <i>intensity rating</i></b>	
	1. <i>User</i> memilih menu <i>intensity rating</i> .	1. Sistem menampilkan <i>form intensity rating</i> .
	2. <i>User</i> melakukan pengisian data subkriteria di setiap field yang ada, lalu menekan tombol <i>insert</i> .	2. Sistem melakukan pengecekan terhadap inputan masing-masing field apakah terjadi pengisian field yang tidak tepat, seperti: field nama <i>intensity rating</i> diisi dengan angka, dst. 2.1. Jika ada kesalahan input, maka sistem akan memberitahukan kesalahan yang terjadi. 2.2. Sistem akan menyimpan data <i>intensity rating</i> tersebut ke dalam database. Setelah data berhasil disimpan akan muncul pesan bahwa proses simpan berhasil dilakukan.
<b>Alur Eksepsi</b>	<b>Aksi Pengguna</b>	<b>Respon Sistem</b>
	<i>User</i> salah memasukkan <i>username/password</i> .	Sistem menampilkan pesan " <i>Username/Password Salah</i> " dan sistem tetap berada di menu <i>login</i> .
	<i>User</i> salah mengisi data kriteria.	Sistem menampilkan pesan peringatan yang memberitahukan kesalahan yang sedang terjadi.
	<i>User</i> salah mengisi data <i>intensity rating</i> .	Sistem menampilkan pesan peringatan yang memberitahukan kesalahan yang sedang terjadi.
	<i>User</i> belum mengisi semua field dan menekan tombol <i>insert</i> .	Sistem menampilkan pesan " <i>Data masih ada yang kosong</i> ".
<b>Kondisi Akhir</b>	Data kriteria dan data <i>intensity rating</i> tersimpan.	
<b>Kebutuhan Non-Fungsional</b>	<b>Performance</b>	
	Respon time	Sistem menampilkan hasil request oleh <i>user</i> dalam waktu maksimal 2 detik.

Nama Fungsi	Menentukan Kriteria Pemilihan <i>Supplier</i>	
	<i>Security</i>	
	Otentikasi <i>user</i>	Sistem melakukan otentikasi terhadap <i>user</i> dengan cara pengecekan <i>username</i> dan <i>password</i> yang dimasukkan oleh <i>user</i> .
	<i>Correctness</i>	
	Pengecekan input di setiap field oleh sistem	Sistem melakukan pengecekan untuk setiap input pada masing-masing field. Pengecekan dilakukan untuk mendeteksi kesalahan pengisian data (misal: data yang seharusnya berupa data numerik tetapi diisi huruf) dan atau data yang belum diisi oleh <i>user</i> (field yang seharusnya diisi tetapi dibiarkan kosong).

- b Fungsi menentukan bobot penilaian kriteria
- Keterangan dari fungsi menentukan bobot penilaian kriteria oleh pemilik bengkel dapat dilihat pada Tabel 3.5.

Tabel 3.5 Keterangan Fungsi Menentukan Bobot Penilaian Kriteria

Nama Fungsi	Menentukan Bobot Penilaian Kriteria	
<b>Stakeholder</b>	Pemilik bengkel	
<b>Deskripsi</b>	Melakukan penilaian kriteria beserta <i>intensity rating</i>	
<b>Kondisi Awal</b>	Sudah menentukan kriteria pemilihan <i>supplier</i>	
<b>Alur Normal</b>	<b>Aksi Pengguna</b>	<b>Respon Sistem</b>
	<b>Menghitung nilai kriteria</b>	
	1. <i>User</i> memilih menu kriteria.	1. Sistem menampilkan menu kriteria.
	2. <i>User</i> memasukkan data penilaian kriteria kedalam <i>cell</i> , lalu menekan <i>enter</i> .	2. Sistem melakukan pengecekan terhadap inputan masing-masing <i>cell</i> . <i>Cell</i> hanya bisa diisi angka 1 s/d 9. 2.1. Jika ada kesalahan input, maka sistem akan memberitahukan

Nama Fungsi	Menentukan Bobot Penilaian Kriteria	
		<p>kesalahan yang terjadi.</p> <p>2.2. Sistem akan menyimpan data penilaian kriteria tersebut ke dalam database. Setelah data berhasil disimpan akan muncul pesan bahwa proses simpan berhasil dilakukan.</p> <p>2.2.1. Sistem menghitung normalisasi kriteria lalu mengupdate database.</p> <p>2.2.2. Sistem menghitung matriks penjumlahan kriteria lalu mengupdate database.</p> <p>2.2.3. Sistem menghitung hasil kriteria lalu mengupdate database.</p> <p>2.2.4. Sistem menghitung rasio konsistensi kriteria. Sistem menampilkan data penilaian kriteria yang telah dicatat.</p>
	<b>Menghitung nilai <i>intensity rating</i></b>	
	1. <i>User</i> memilih menu <i>intensity rating</i> .	1. Sistem menampilkan menu <i>intensity rating</i> .
	2. <i>User</i> memasukkan data penilaian <i>intensity rating</i> kedalam <i>cell</i> , lalu menekan <i>enter</i> .	<p>2. Sistem melakukan pengecekan terhadap inputan masing-masing <i>cell</i>. <i>Cell</i> hanya bisa diisi angka 1 s/d 9.</p> <p>2.1. Jika ada kesalahan input, maka sistem akan memberitahukan kesalahan yang terjadi.</p> <p>2.2. Sistem akan menyimpan data penilaian <i>intensity rating</i> tersebut ke dalam database. Setelah data berhasil disimpan akan muncul pesan</p>

Nama Fungsi	Menentukan Bobot Penilaian Kriteria	
		<p>bahwa proses simpan berhasil dilakukan.</p> <p>2.2.1. Sistem menghitung normalisasi <i>intensity rating</i> lalu mengupdate database.</p> <p>2.2.2. Sistem menghitung matriks penjumlahan <i>intensity rating</i> lalu mengupdate database.</p> <p>2.2.3. Sistem menghitung hasil <i>intensity rating</i> lalu mengupdate database.</p> <p>2.2.4. Sistem menghitung rasio konsistensi <i>intensity rating</i>. Sistem menampilkan data penilaian <i>intensity rating</i> yang telah dicatat.</p>
<b>Alur Eksepsi</b>	<b>Aksi Pengguna</b>	<b>Respon Sistem</b>
	<i>User</i> salah memasukkan <i>username/password</i> .	Sistem menampilkan pesan " <i>Username/Password Salah</i> " dan sistem tetap berada di menu <i>login</i> .
	<i>User</i> salah mengisi data penilaian kriteria.	Sistem menampilkan pesan peringatan yang memberitahukan kesalahan yang sedang terjadi.
	<i>User</i> salah mengisi data penilaian <i>intensity rating</i> .	Sistem menampilkan pesan peringatan yang memberitahukan kesalahan yang sedang terjadi.
<b>Kondisi Akhir</b>	Data penilaian kriteria dan data penilaian <i>intensity rating</i> tersimpan	
<b>Kebutuhan Non-Fungsional</b>	<i>Performance</i>	
	Respon time	Sistem menampilkan hasil request oleh <i>user</i> dalam waktu maksimal 2 detik.
	<i>Security</i>	
Otentikasi <i>user</i>	Sistem melakukan otentikasi terhadap <i>user</i> dengan cara	

Nama Fungsi	Menentukan Bobot Penilaian Kriteria	
		pengecekan <i>username</i> dan <i>password</i> yang dimasukkan oleh <i>user</i> .
	<i>Correctness</i>	
	Pengecekan input di setiap field oleh sistem	Sistem melakukan pengecekan untuk setiap input pada masing-masing field. Pengecekan dilakukan untuk mendeteksi kesalahan pengisian data (misal: data yang seharusnya berupa data numerik tetapi diisi huruf) dan atau data yang belum diisi oleh <i>user</i> (field yang seharusnya diisi tetapi dibiarkan kosong).

c Fungsi menilai produk *supplier*

Keterangan dari fungsi menilai produk *supplier* oleh pemilik bengkel dapat dilihat pada Tabel 3.6.

Tabel 3.6 Keterangan Fungsi Menilai Produk *Supplier*

Nama Fungsi	Menilai Produk <i>Supplier</i>	
<b>Stakeholder</b>	Pemilik bengkel	
<b>Deskripsi</b>	Melakukan penilaian produk tiap-tiap <i>supplier</i> sesuai dengan kriteria yang sudah ditetapkan	
<b>Kondisi Awal</b>	<ol style="list-style-type: none"> <li>1. Sudah menentukan kriteria pemilihan <i>supplier</i></li> <li>2. Sudah menentukan bobot penilaian kriteria</li> <li>3. Sudah mencatat data barang</li> <li>4. Sudah mencatat data <i>supplier</i></li> <li>5. Sudah mencatat data pembelian</li> </ol>	
<b>Alur Normal</b>	<b>Aksi Pengguna</b>	<b>Respon Sistem</b>
	<b>Menghitung dan memasukkan nilai</b>	
	1. <i>User</i> memilih menu penilaian barang.	1. Sistem menampilkan menu penilaian barang.
	2. <i>User</i> memasukkan data penilaian barang, lalu menekan tombol <i>insert</i> .	2. Sistem melakukan pengecekan terhadap inputan masing-masing field. 2.1. Jika ada kesalahan input, maka sistem akan

Nama Fungsi	Menilai Produk <i>Supplier</i>	
		<p>memberitahukan kesalahan yang terjadi.</p> <p>2.2. Sistem akan menyimpan data penilaian barang tersebut ke dalam database. Setelah data berhasil disimpan akan muncul pesan bahwa proses simpan berhasil dilakukan.</p> <p>2.3. Sistem menampilkan data penilaian barang <i>supplier</i> yang telah dicatat.</p>
	<b>Update nilai akhir barang</b>	
	-	Sistem melakukan update nilai akhir barang.
<b>Alur Eksepsi</b>	<b>Aksi Pengguna</b>	<b>Respon Sistem</b>
	<i>User</i> salah memasukkan <i>username/password</i> .	Sistem menampilkan pesan " <i>Username/Password Salah</i> " dan sistem tetap berada di menu <i>login</i> .
	<i>User</i> salah mengisi data penilaian barang.	Sistem menampilkan pesan peringatan yang memberitahukan kesalahan yang sedang terjadi.
	<i>User</i> belum mengisi semua field dan menekan tombol <i>insert</i> .	Sistem menampilkan pesan " <i>Data masih ada yang kosong</i> ".
<b>Kondisi Akhir</b>	Data penilaian barang tersimpan	
<b>Kebutuhan Non-Fungsional</b>	<b>Performance</b>	
	Respon time	Sistem menampilkan hasil request oleh <i>user</i> dalam waktu maksimal 2 detik.
	<b>Security</b>	
	Otentikasi <i>user</i>	Sistem melakukan otentikasi terhadap <i>user</i> dengan cara pengecekan <i>username</i> dan <i>password</i> yang dimasukkan oleh <i>user</i> .
	<b>Correctness</b>	
	Pengecekan input di setiap field oleh sistem	Sistem melakukan pengecekan untuk setiap input pada masing-masing field. Pengecekan dilakukan untuk mendeteksi kesalahan

Nama Fungsi	Menilai Produk <i>Supplier</i>	
		pengisian data (misal: data yang seharusnya berupa data numerik tetapi diisi huruf) dan atau data yang belum diisi oleh <i>user</i> (field yang seharusnya diisi tetapi dibiarkan kosong).

### 3.3.2 Bagian pembelian

Bagian pembelian memiliki 4 fungsi yaitu mencatat data barang, mencatat data *supplier*, mencatat data pembelian, dan melakukan pemilihan *supplier*.

Adapun penjelasan setiap fungsi dapat dilihat sebagai berikut:

a Fungsi mencatat data barang

Keterangan dari fungsi mencatat data barang oleh bagian pembelian dapat dilihat pada Tabel 3.7.

Tabel 3.7 Keterangan Fungsi Mencatat Data Barang

Nama Fungsi	Mencatat Data Barang	
Stakeholder	Bagian pembelian	
Deskripsi	Mencatat data barang yang akan digunakan untuk penilaian barang	
Kondisi Awal	-	
Alur Normal	Aksi Pengguna	Respon Sistem
	Mengisi data barang	
	1. <i>User</i> memilih menu barang.	1. Sistem menampilkan menu barang.
	2. <i>User</i> memasukkan data barang, lalu menekan tombol <i>insert</i> .	2. Sistem melakukan pengecekan terhadap inputan masing-masing field. 2.1. Jika ada kesalahan input, maka sistem akan memberitahukan kesalahan yang terjadi. 2.2. Sistem akan menyimpan data barang tersebut ke dalam database. Setelah

Nama Fungsi	Mencatat Data Barang	
		<p>data berhasil disimpan akan muncul pesan bahwa proses simpan berhasil dilakukan.</p> <p>2.3. Sistem menampilkan data barang yang telah dicatat.</p>
<b>Alur Eksepsi</b>	<b>Aksi Pengguna</b>	<b>Respon Sistem</b>
	<i>User</i> salah memasukkan <i>username/password</i> .	Sistem menampilkan pesan " <i>Username/Password Salah</i> " dan sistem tetap berada di menu <i>login</i> .
	<i>User</i> salah mengisi data barang.	Sistem menampilkan pesan peringatan yang memberitahukan kesalahan yang sedang terjadi.
<b>Kondisi Akhir</b>	Data barang tersimpan	
<b>Kebutuhan Non-Fungsional</b>	<i>Performance</i>	
	Respon time	Sistem menampilkan hasil request oleh <i>user</i> dalam waktu maksimal 2 detik.
	<i>Security</i>	
	Otentikasi <i>user</i>	Sistem melakukan otentikasi terhadap <i>user</i> dengan cara pengecekan <i>username</i> dan <i>password</i> yang dimasukkan oleh <i>user</i> .
<i>Correctness</i>		
Pengecekan input di setiap field oleh sistem	Sistem melakukan pengecekan untuk setiap input pada masing-masing field. Pengecekan dilakukan untuk mendeteksi kesalahan pengisian data (misal: data yang seharusnya berupa data numerik tetapi diisi huruf) dan atau data yang belum diisi oleh <i>user</i> (field yang seharusnya diisi tetapi dibiarkan kosong).	



b Fungsi mencatat data *supplier*

Keterangan dari fungsi mencatat data *supplier* oleh bagian pembelian dapat dilihat pada Tabel 3.8.

Tabel 3.8 Keterangan Fungsi Mencatat Data *Supplier*

Nama Fungsi	Mencatat Data <i>Supplier</i>	
<b>Stakeholder</b>	Bagian pembelian	
<b>Deskripsi</b>	Mencatat data <i>supplier</i> yang akan digunakan untuk penilaian barang	
<b>Kondisi Awal</b>	Sudah mencatat data barang	
<b>Alur Normal</b>	<b>Aksi Pengguna</b>	<b>Respon Sistem</b>
	<b>Mengisi data <i>supplier</i></b>	
	1. <i>User</i> memilih menu <i>supplier</i> .	1. Sistem menampilkan menu <i>supplier</i> .
2. <i>User</i> memasukkan data <i>supplier</i> , lalu menekan tombol <i>insert</i> .	2. Sistem melakukan pengecekan terhadap inputan masing-masing field. 2.1. Jika ada kesalahan input, maka sistem akan memberitahukan kesalahan yang terjadi. 2.2. Sistem akan menyimpan data <i>supplier</i> tersebut ke dalam database. Setelah data berhasil disimpan akan muncul pesan bahwa proses simpan berhasil dilakukan. 2.3. Sistem menampilkan data <i>supplier</i> yang telah dicatat.	
<b>Alur Eksepsi</b>	<b>Aksi Pengguna</b>	<b>Respon Sistem</b>
	<i>User</i> salah memasukkan <i>username/password</i> .	Sistem menampilkan pesan " <i>Username/Password Salah</i> " dan sistem tetap berada di menu <i>login</i> .
	<i>User</i> salah mengisi data <i>supplier</i> .	Sistem menampilkan pesan peringatan yang memberitahukan kesalahan yang sedang terjadi.
	<i>User</i> belum mengisi semua field dan menekan tombol	Sistem menampilkan pesan " <i>Data masih ada yang</i>

Nama Fungsi	Mencatat Data <i>Supplier</i>	
	<i>insert.</i>	kosong".
<b>Kondisi Akhir</b>	Data <i>supplier</i> tersimpan	
<b>Kebutuhan Non-Fungsional</b>	<i>Performance</i>	
	Respon time	Sistem menampilkan hasil request oleh <i>user</i> dalam waktu maksimal 2 detik.
	<i>Security</i>	
	Otentikasi <i>user</i>	Sistem melakukan otentikasi terhadap <i>user</i> dengan cara pengecekan <i>username</i> dan <i>password</i> yang dimasukkan oleh <i>user</i> .
<i>Correctness</i>		
Pengecekan input di setiap field oleh sistem	Sistem melakukan pengecekan untuk setiap input pada masing-masing field. Pengecekan dilakukan untuk mendeteksi kesalahan pengisian data (misal: data yang seharusnya berupa data numerik tetapi diisi huruf) dan atau data yang belum diisi oleh <i>user</i> (field yang seharusnya diisi tetapi dibiarkan kosong).	

c Fungsi mencatat data pembelian

Keterangan dari fungsi mencatat data pembelian oleh bagian pembelian dapat dilihat pada Tabel 3.9.

Tabel 3.9 Keterangan Fungsi Mencatat Data Pembelian

Nama Fungsi	Mencatat Data Pembelian	
<b>Stakeholder</b>	Bagian pembelian	
<b>Deskripsi</b>	Mencatat data pembelian untuk menghitung kecepatan pengiriman <i>supplier</i>	
<b>Kondisi Awal</b>	<ol style="list-style-type: none"> <li>Sudah mencatat data barang</li> <li>Sudah mencatat data <i>supplier</i></li> </ol>	
<b>Alur Normal</b>	<b>Aksi Pengguna</b>	<b>Respon Sistem</b>
	<b>Mengisi data pembelian</b>	
	1. <i>User</i> memilih menu transaksi pembelian.	1. Sistem menampilkan menu transaksi

Nama Fungsi	Mencatat Data Pembelian	
	<p>2. <i>User</i> memasukkan data pembelian, lalu menekan tombol <i>insert</i>.</p>	<p>pembelian.</p> <p>2. Sistem melakukan pengecekan terhadap inputan masing-masing field.</p> <p>2.1. Jika ada kesalahan input, maka sistem akan memberitahukan kesalahan yang terjadi.</p> <p>2.2. Sistem akan menyimpan data pembelian tersebut ke dalam database. Setelah data berhasil disimpan akan muncul pesan bahwa proses simpan berhasil dilakukan.</p> <p>2.3. Sistem menampilkan data pembelian yang telah dicatat.</p>
Alur Eksepsi	Aksi Pengguna	Respon Sistem
	<i>User</i> salah memasukkan <i>username/password</i> .	Sistem menampilkan pesan " <i>Username/Password Salah</i> " dan sistem tetap berada di menu <i>login</i> .
	<i>User</i> salah mengisi data pembelian.	Sistem menampilkan pesan peringatan yang memberitahukan kesalahan yang sedang terjadi.
	<i>User</i> belum mengisi semua field dan menekan tombol <i>insert</i> .	Sistem menampilkan pesan " <i>Data masih ada yang kosong</i> ".
Kondisi Akhir	Data pembelian tersimpan	
Kebutuhan Non-Fungsional	<i>Performance</i>	
	Respon time	Sistem menampilkan hasil request oleh <i>user</i> dalam waktu maksimal 2 detik.
	<i>Security</i>	
	Otentikasi <i>user</i>	Sistem melakukan otentikasi terhadap <i>user</i> dengan cara pengecekan <i>username</i> dan <i>password</i> yang dimasukkan oleh <i>user</i> .
	<i>Correctness</i>	
	Pengecekan input di setiap field oleh sistem	Sistem melakukan pengecekan untuk setiap

Nama Fungsi	Mencatat Data Pembelian	
		input pada masing-masing field. Pengecekan dilakukan untuk mendeteksi kesalahan pengisian data (misal: data yang seharusnya berupa data numerik tetapi diisi huruf) dan atau data yang belum diisi oleh <i>user</i> (field yang seharusnya diisi tetapi dibiarkan kosong).

d Fungsi melakukan pemilihan *supplier*

Keterangan dari fungsi melakukan pemilihan *supplier* oleh bagian pembelian dapat dilihat pada Tabel 3.10.

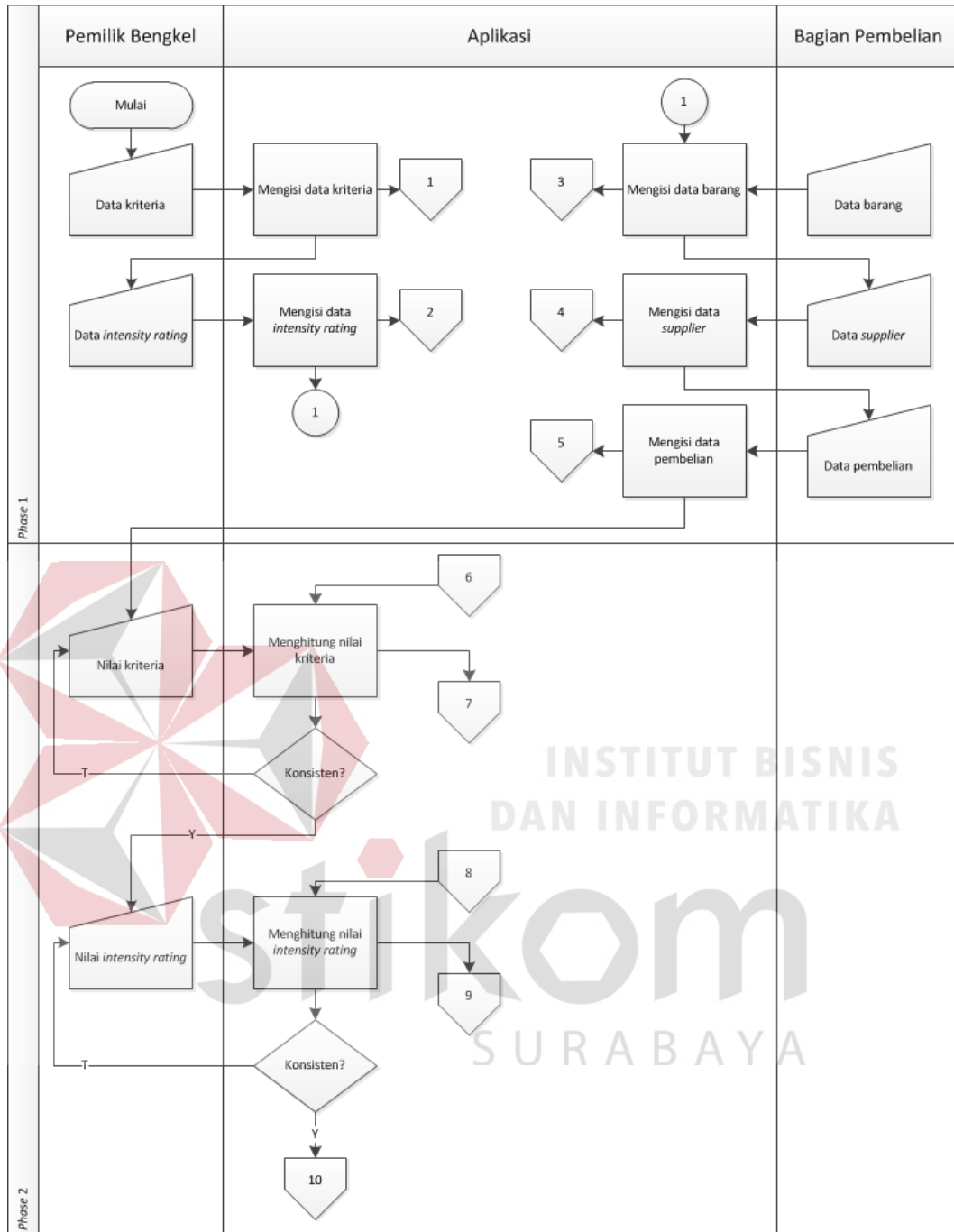
Tabel 3.10 Keterangan Fungsi Melakukan Pemilihan *Supplier*

Nama Fungsi	Melakukan Pemilihan <i>Supplier</i>													
<b>Stakeholder</b>	Bagian pembelian													
<b>Deskripsi</b>	Melakukan pemilihan <i>supplier</i> sesuai barang yang diinginkan													
<b>Kondisi Awal</b>	<ol style="list-style-type: none"> <li>1. Sudah menentukan kriteria pemilihan <i>supplier</i></li> <li>2. Sudah menentukan bobot penilaian kriteria</li> <li>3. Sudah menilai produk <i>supplier</i></li> <li>4. Sudah mencatat data barang</li> <li>5. Sudah mencatat data <i>supplier</i></li> <li>6. Sudah mencatat data pembelian</li> </ol>													
<b>Alur Normal</b>	<table border="1"> <thead> <tr> <th data-bbox="531 1377 951 1424">Aksi Pengguna</th> <th data-bbox="951 1377 1370 1424">Respon Sistem</th> </tr> </thead> <tbody> <tr> <td colspan="2" data-bbox="531 1424 1370 1471" style="text-align: center;"><b>Menampilkan rekomendasi <i>supplier</i></b></td> </tr> <tr> <td data-bbox="531 1471 951 1576">1. <i>User</i> memilih menu rekomendasi <i>supplier</i>.</td> <td data-bbox="951 1471 1370 1576">1. Sistem menampilkan menu rekomendasi <i>supplier</i>.</td> </tr> <tr> <td data-bbox="531 1576 951 1653">2. <i>User</i> memasukkan nama barang yang diinginkan.</td> <td data-bbox="951 1576 1370 1653">2. Sistem menampilkan rekomendasi <i>supplier</i>.</td> </tr> <tr> <td colspan="2" data-bbox="531 1653 1370 1700" style="text-align: center;"><b>Mencetak rekomendasi <i>supplier</i></b></td> </tr> <tr> <td data-bbox="531 1700 951 1771"><i>User</i> menekan tombol print.</td> <td data-bbox="951 1700 1370 1771">Sistem mencetak daftar rekomendasi <i>supplier</i>.</td> </tr> </tbody> </table>	Aksi Pengguna	Respon Sistem	<b>Menampilkan rekomendasi <i>supplier</i></b>		1. <i>User</i> memilih menu rekomendasi <i>supplier</i> .	1. Sistem menampilkan menu rekomendasi <i>supplier</i> .	2. <i>User</i> memasukkan nama barang yang diinginkan.	2. Sistem menampilkan rekomendasi <i>supplier</i> .	<b>Mencetak rekomendasi <i>supplier</i></b>		<i>User</i> menekan tombol print.	Sistem mencetak daftar rekomendasi <i>supplier</i> .	
Aksi Pengguna	Respon Sistem													
<b>Menampilkan rekomendasi <i>supplier</i></b>														
1. <i>User</i> memilih menu rekomendasi <i>supplier</i> .	1. Sistem menampilkan menu rekomendasi <i>supplier</i> .													
2. <i>User</i> memasukkan nama barang yang diinginkan.	2. Sistem menampilkan rekomendasi <i>supplier</i> .													
<b>Mencetak rekomendasi <i>supplier</i></b>														
<i>User</i> menekan tombol print.	Sistem mencetak daftar rekomendasi <i>supplier</i> .													
<b>Alur Eksepsi</b>	<table border="1"> <thead> <tr> <th data-bbox="531 1771 951 1818">Aksi Pengguna</th> <th data-bbox="951 1771 1370 1818">Respon Sistem</th> </tr> </thead> <tbody> <tr> <td data-bbox="531 1818 951 1966"><i>User</i> salah memasukkan <i>username/password</i>.</td> <td data-bbox="951 1818 1370 1966">Sistem menampilkan pesan "<i>Username/Password Salah</i>" dan sistem tetap berada di menu <i>login</i>.</td> </tr> <tr> <td data-bbox="531 1966 951 2000"><i>User</i> salah mengisi data</td> <td data-bbox="951 1966 1370 2000">Sistem menampilkan pesan</td> </tr> </tbody> </table>	Aksi Pengguna	Respon Sistem	<i>User</i> salah memasukkan <i>username/password</i> .	Sistem menampilkan pesan " <i>Username/Password Salah</i> " dan sistem tetap berada di menu <i>login</i> .	<i>User</i> salah mengisi data	Sistem menampilkan pesan							
Aksi Pengguna	Respon Sistem													
<i>User</i> salah memasukkan <i>username/password</i> .	Sistem menampilkan pesan " <i>Username/Password Salah</i> " dan sistem tetap berada di menu <i>login</i> .													
<i>User</i> salah mengisi data	Sistem menampilkan pesan													

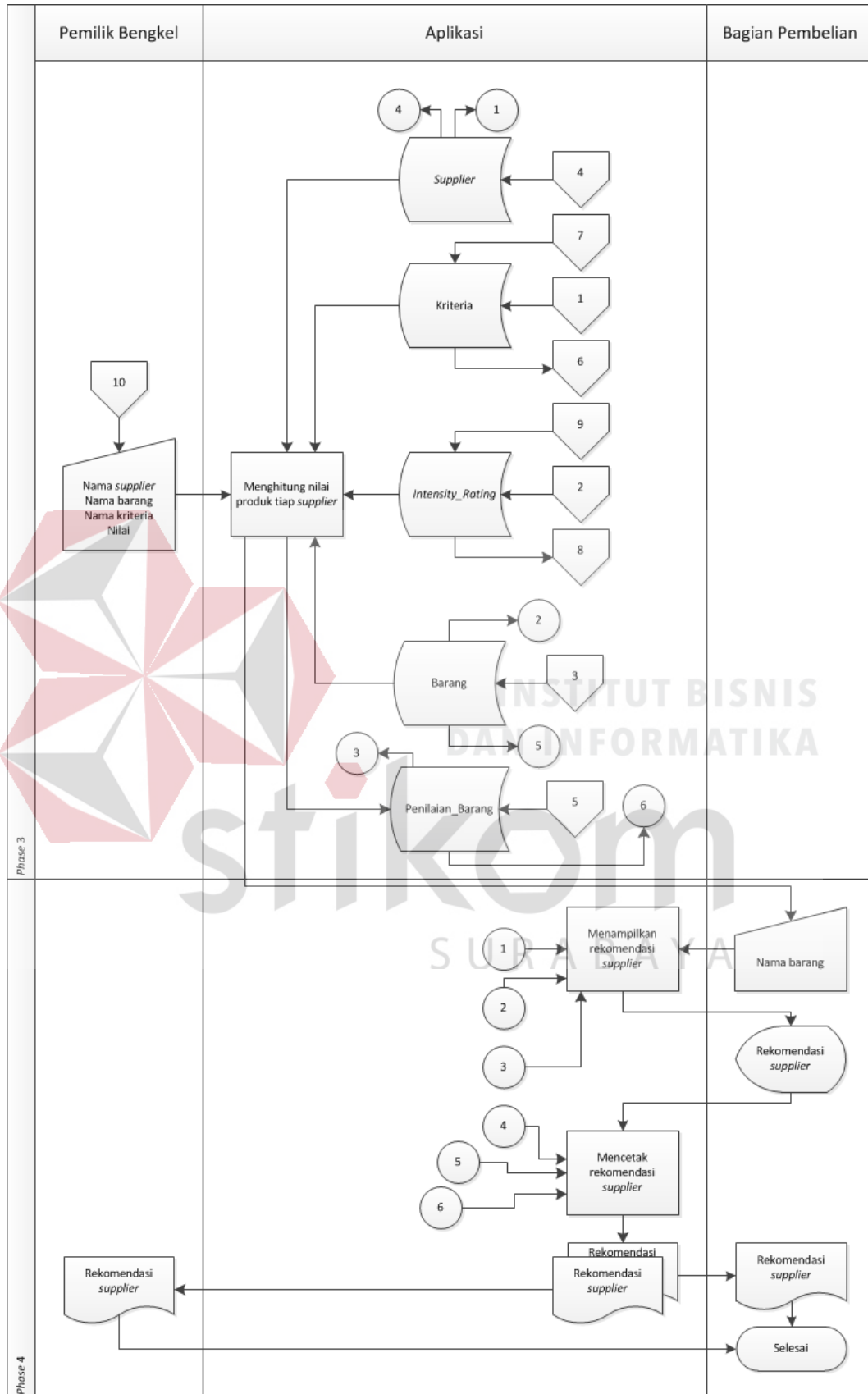
Nama Fungsi		Melakukan Pemilihan <i>Supplier</i>	
	pembelian.	peringatan yang memberitahukan kesalahan yang sedang terjadi.	
	<i>User</i> belum mengisi semua field dan menekan tombol <i>insert</i> .	Sistem menampilkan pesan "Data masih ada yang kosong".	
Kondisi Akhir		Data pembelian tersimpan	
Kebutuhan Non-Fungsional	<i>Performance</i>		
	Respon time	Sistem menampilkan hasil request oleh <i>user</i> dalam waktu maksimal 2 detik.	
	<i>Security</i>		
	Otentikasi <i>user</i>	Sistem melakukan otentikasi terhadap <i>user</i> dengan cara pengecekan <i>username</i> dan <i>password</i> yang dimasukkan oleh <i>user</i> .	
<i>Correctness</i>			
	Pengecekan input di setiap field oleh sistem	Sistem melakukan pengecekan untuk setiap input pada masing-masing field. Pengecekan dilakukan untuk mendeteksi kesalahan pengisian data (misal: data yang seharusnya berupa data numerik tetapi diisi huruf) dan atau data yang belum diisi oleh <i>user</i> (field yang seharusnya diisi tetapi dibiarkan kosong).	

### 3.4 Merancang Sistem yang Direkomendasikan

Setelah mengetahui beberapa kekurangan sistem yang ada saat ini, maka dibuatkan perancangan sistem yang baru. Dengan perancangan sistem yang baru, bagian pembelian mendapatkan rekomendasi hasil pemilihan *supplier* berdasarkan kondisi atau kriteria yang sudah ditentukan oleh pemilik bengkel. Untuk lebih jelasnya lihat Gambar 3.3 dan Gambar 3.4.



Gambar 3.3 System Flow Baru Derry Auto Service



Gambar 3.4 Lanjutan System Flow Baru Derry Auto Service

Penjelasan *system flow* baru Derry Auto Service pada Gambar 3.3 dan Gambar 3.4 bisa dilihat pada Tabel 3.11

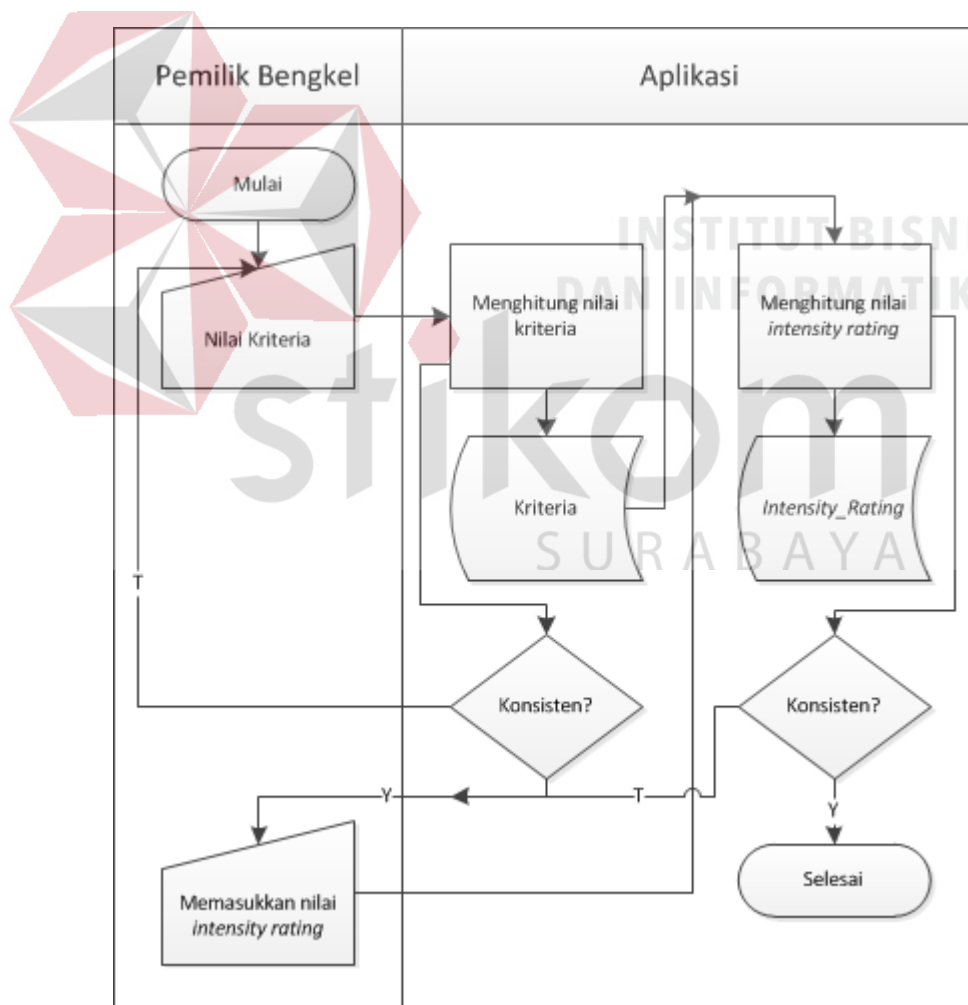
Tabel 3.11 Keterangan *System Flow* Baru Derry Auto Service

Nama Proses	Keterangan
<i>Phase 1</i>	
Mengisi data kriteria	Sistem memasukkan data kriteria dari data yang dimasukkan oleh pemilik bengkel kemudian menyimpan data tersebut ke tabel kriteria.
Mengisi data <i>intensity rating</i>	Sistem memasukkan data <i>intensity rating</i> dari data yang dimasukkan oleh pemilik bengkel kemudian menyimpan data tersebut ke tabel <i>intensity rating</i> .
Mengisi data barang	Sistem memasukkan data barang dari data yang dimasukkan oleh bagian pembelian kemudian menyimpan data tersebut ke tabel barang.
Mengisi data <i>supplier</i>	Sistem memasukkan data <i>supplier</i> dari data yang dimasukkan oleh bagian pembelian kemudian menyimpan data tersebut ke tabel <i>supplier</i> .
Mengisi data pembelian	Sistem memasukkan data pembelian dari data yang dimasukkan oleh bagian pembelian kemudian menyimpan data tersebut ke tabel pembelian.
<i>Phase 2</i>	
Menghitung nilai kriteria	Sistem menghitung nilai kriteria dari data yang dimasukkan oleh pemilik bengkel kemudian menyimpan hasil perhitungan ke tabel kriteria.
<i>Decision</i> konsisten	Jika nilai rasio konsistensi konsisten, maka akan dilanjutkan ke proses menghitung nilai <i>intensity rating</i> . Jika tidak konsisten, maka pemilik bengkel harus mengubah data nilai kriteria.
Menghitung nilai <i>intensity rating</i>	Sistem menghitung nilai <i>intensity rating</i> dari data yang dimasukkan oleh pemilik bengkel kemudian menyimpan hasil perhitungan ke tabel <i>intensity rating</i> .
<i>Decision</i> konsisten	Jika nilai rasio konsistensi konsisten, maka selesai sudah <i>system flow</i> menilai kriteria <i>intensity rating</i> . Jika tidak konsisten, maka pemilik bengkel harus mengubah data nilai <i>intensity rating</i> .
<i>Phase 3</i>	
Menghitung nilai produk tiap <i>supplier</i>	Sistem menghitung nilai produk tiap <i>supplier</i> dari data penilaian yang dimasukkan oleh pemilik bengkel lalu sistem menyimpan hasil perhitungannya nilai produk tiap <i>supplier</i> kedalam tabel penilaian barang.



Nama Proses	Keterangan
<i>Phase 4</i>	
Menampilkan rekomendasi <i>supplier</i>	Sistem menampilkan data rekomendasi <i>supplier</i> sesuai dengan nama barang yang diinginkan bagian pembelian
Mencetak rekomendasi <i>supplier</i>	Sistem mencetak hasil rekomendasi <i>supplier</i> sesuai dengan nama barang yang diinginkan bagian pembelian. Hasil rekomendasi <i>supplier</i> diberikan ke pemilik bengkel dan bagian pembelian.

Berikut ini adalah *system flow* menilai kriteria *intensity rating* yang dilakukan oleh pemilik bengkel.



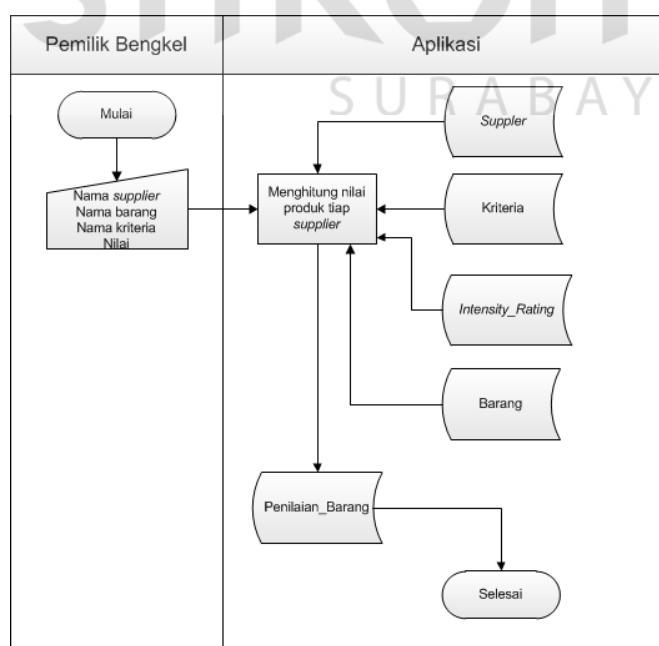
Gambar 3.5 *System Flow* Menilai Kriteria *Intensity rating*

Penjelasan *system flow* menilai kriteria *intensity rating* pada Gambar 3.5 bisa dilihat pada Tabel 3.12

Tabel 3.12 Keterangan *System Flow* Menilai Kriteria *Intensity rating*

Nama Proses	Keterangan
Menghitung nilai kriteria	Sistem menghitung nilai kriteria dari data yang dimasukkan oleh pemilik bengkel kemudian menyimpan hasil perhitungan ke tabel kriteria.
<i>Decision</i> konsisten	Jika nilai rasio konsistensi konsisten, maka akan dilanjutkan ke proses menghitung nilai <i>intensity rating</i> . Jika tidak konsisten, maka pemilik bengkel harus mengubah data nilai kriteria.
Menghitung nilai <i>intensity rating</i>	Sistem menghitung nilai <i>intensity rating</i> dari data yang dimasukkan oleh pemilik bengkel kemudian menyimpan hasil perhitungan ke tabel <i>intensity rating</i> .
<i>Decision</i> konsisten	Jika nilai rasio konsistensi konsisten, maka selesai sudah <i>system flow</i> menilai kriteria <i>intensity rating</i> . Jika tidak konsisten, maka pemilik bengkel harus mengubah data nilai <i>intensity rating</i> .

Berikut ini adalah *system flow* menghitung nilai produk tiap *supplier* yang dilakukan oleh pemilik bengkel.



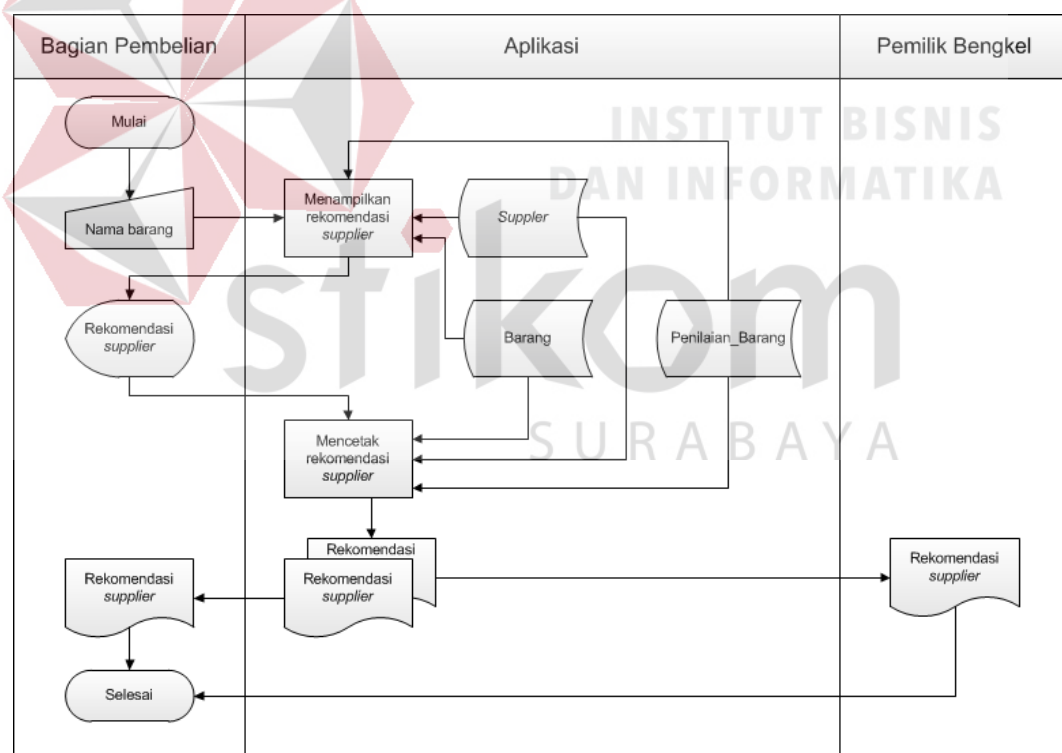
Gambar 3.6 *System Flow* Menghitung Nilai Produk Tiap *Supplier*

Penjelasan *system flow* menghitung nilai produk tiap *supplier* pada Gambar 3.6 bisa dilihat pada Tabel 3.13

Tabel 3.13 Keterangan *System Flow* Menghitung Nilai Produk Tiap *Supplier*

Nama Proses	Keterangan
Menghitung nilai produk tiap <i>supplier</i>	Sistem menghitung nilai produk tiap <i>supplier</i> dari data penilaian yang dimasukkan oleh pemilik bengkel lalu sistem menyimpan hasil perhitungan nilai produk tiap <i>supplier</i> kedalam tabel penilaian barang.

Berikut ini adalah *system flow* memilih *supplier* yang dilakukan oleh bagian pembelian.



Gambar 3.7 *System Flow* Memilih *Supplier*

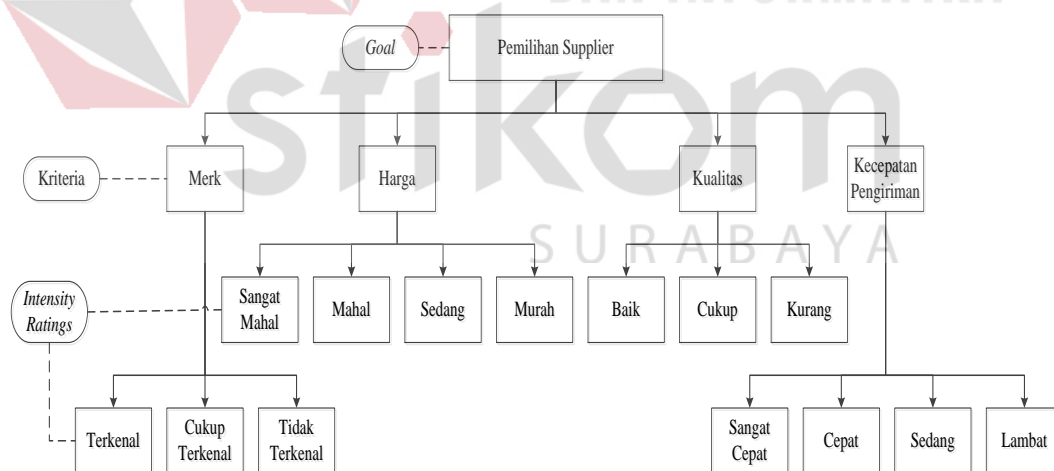
Penjelasan *system flow* memilih *supplier* pada Gambar 3.7 bisa dilihat pada Tabel 3.14

Tabel 3.14 Keterangan *System Flow* Memilih *Supplier*

Nama Proses	Keterangan
Menampilkan rekomendasi <i>supplier</i>	Sistem menampilkan data rekomendasi <i>supplier</i> sesuai dengan nama barang yang diinginkan bagian pembelian.
Mencetak rekomendasi <i>supplier</i>	Sistem mencetak hasil rekomendasi <i>supplier</i> sesuai dengan nama barang yang diinginkan bagian pembelian. Hasil rekomendasi <i>supplier</i> diberikan ke pemilik bengkel dan bagian pembelian.

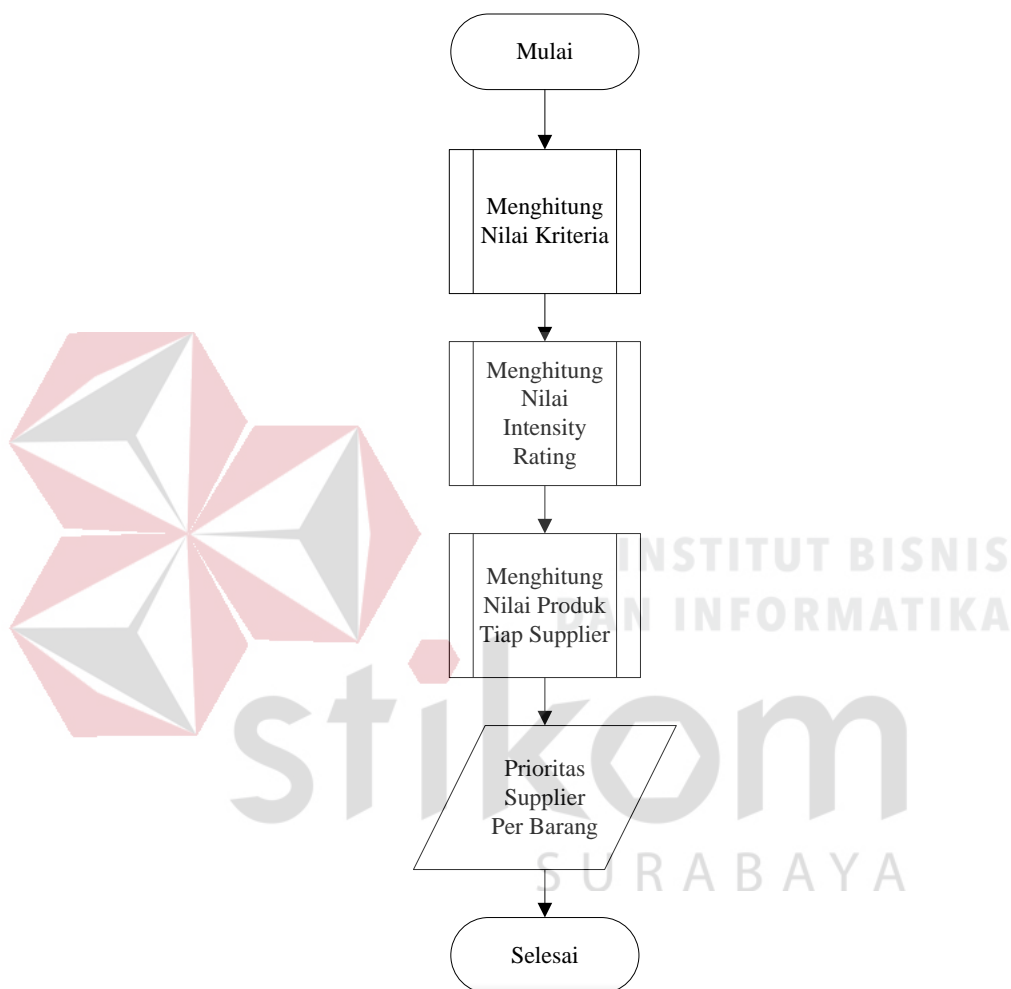
### 3.4.1 Hierarki AHP

Berikut ini pada Gambar 3.8 adalah hierarki AHP pada Derry *Auto Service* yang menggunakan hierarki 4 level yaitu *goal*, kriteria, *intensity rating*, dan alternatif. Alternatif adalah barang yang akan dinilai. Alternatif akan terhubung dengan semua *intensity rating* (tidak digambarkan dalam Gambar 3.8 karena garisnya terpotong dengan *intensity rating* lainnya).

Gambar 3.8 Hierarki *Analytical Hierarchy Process* (AHP)

### 3.4.2 Flowchart AHP

Berikut ini merupakan gambar dari *flowchart* perhitungan menggunakan metode *Analytical Hierarchy Process* (AHP) yang dimulai dari menghitung kriteria, menghitung *intensity rating*, dan menghitung nilai produk tiap *supplier*.



Gambar 3.9 *Flowchart Analytical Hierarchy Process* (AHP)

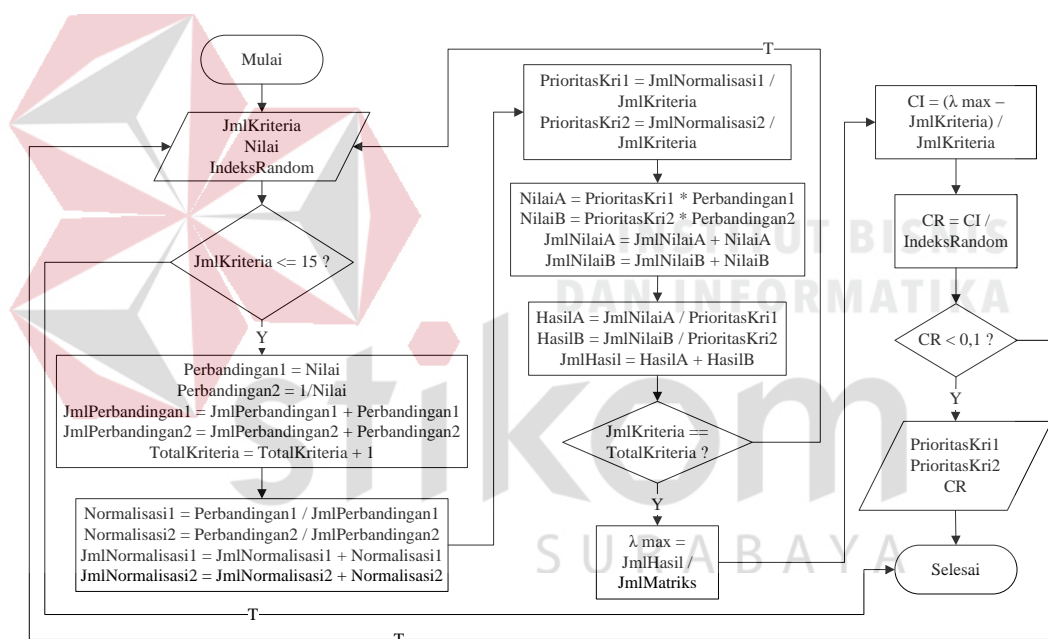
Penjelasan *flowchart Analytical Hierarchy Process* (AHP) pada Gambar 3.9 bisa dilihat pada Tabel 3.15.

Tabel 3.15 Keterangan *Flowchart Analytical Hierarchy Process* (AHP)

Nama Sub Proses	Keterangan
Menghitung nilai	Sistem menghitung nilai kriteria dari data yang dimasukkan

Nama Sub Proses	Keterangan
kriteria	oleh pemilik bengkel kemudian menyimpan hasil perhitungan ke tabel kriteria.
Menghitung nilai <i>intensity rating</i>	Sistem menghitung nilai <i>intensity rating</i> dari data yang dimasukkan oleh pemilik bengkel kemudian menyimpan hasil perhitungan ke tabel <i>intensity rating</i> .
Menghitung nilai produk tiap <i>supplier</i>	Sistem menghitung nilai produk tiap <i>supplier</i> dari data penilaian yang dimasukkan oleh pemilik bengkel lalu sistem menyimpan hasil perhitungannya nilai produk tiap <i>supplier</i> kedalam tabel penilaian barang.

Berikut ini merupakan gambar dari *flowchart* menghitung kriteria. Ada delapan proses pada *flowchart* menghitung kriteria.



Gambar 3.10 Menghitung Kriteria

Penjelasan *flowchart* menghitung kriteria pada Gambar 3.10 bisa dilihat pada Tabel 3.16.

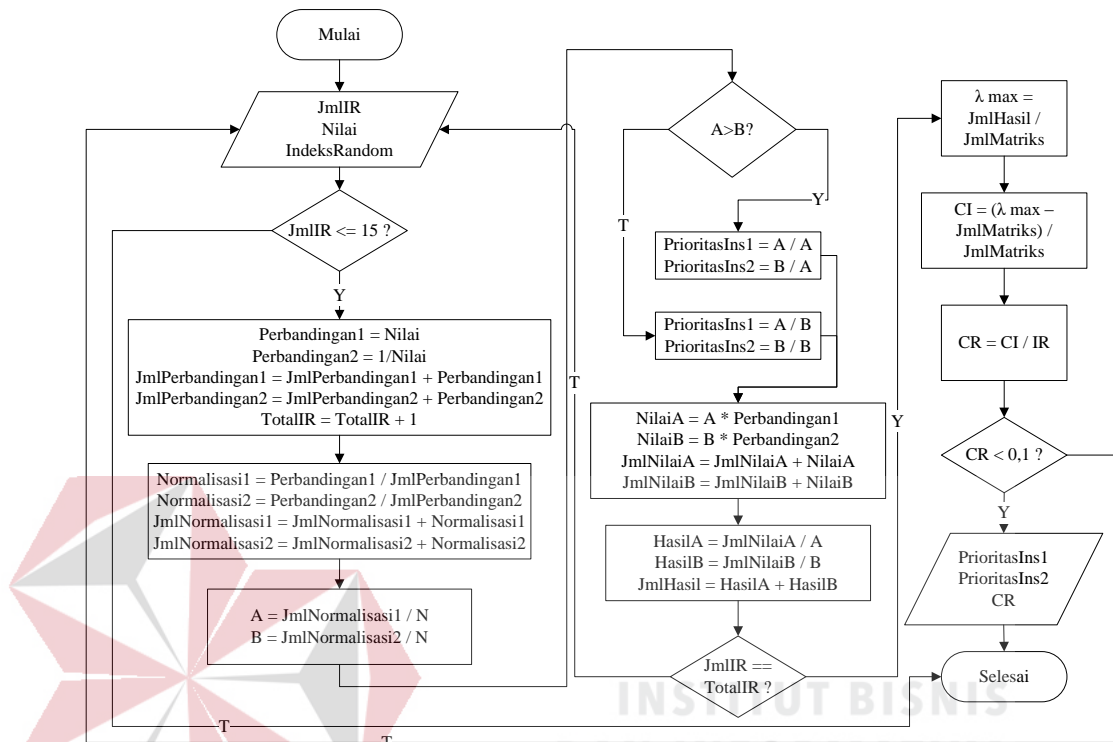
Tabel 3.16 Keterangan *Flowchart* Menghitung Kriteria

Nama Proses	Keterangan
<i>Decision</i>	Jika JmlKriteria lebih kecil atau sama dengan 15 maka lanjut ke

Nama Proses	Keterangan
	proses 1, jika tidak maka selesai.
Proses 1	Nilai yang dimasukkan akan disimpan menjadi Perbandingan1. Perbandingan2 didapatkan dari angka 1 dibagi dengan Perbandingan1. JmlPerbandingan1 didapatkan dari JmlPerbandingan1 ditambah dengan nilai yang ada pada Perbandingan1. JmlPerbandingan2 didapatkan dari JmlPerbandingan2 ditambah dengan nilai yang ada pada Perbandingan2. TotalKriteria didapatkan dari TotalKriteria ditambah dengan 1.
Proses 2	Normalisasi1 didapatkan dari Perbandingan1 dibagi dengan JmlPerbandingan1. Normalisasi2 didapatkan dari Perbandingan2 dibagi dengan JmlPerbandingan2. Perbandingan1, JmlPerbandingan1, Perbandingan2 dan JmlPerbandingan2 sudah didapatkan dari proses 1. JmlNormalisasi1 didapatkan dari JmlNormalisasi1 ditambah dengan Normalisasi1. JmlNormalisasi2 didapatkan dari JmlNormalisasi2 ditambah dengan Normalisasi2.
Proses 3	PrioritasKri1 didapatkan dari JmlNormalisasi1 dibagi dengan N. PrioritasKri2 didapatkan dari JmlNormalisasi2 dibagi dengan N. JmlNormalisasi1 dan JmlNormalisasi2 sudah didapatkan dari proses 2. N adalah jumlah total kriteria.
Proses 4	NilaiA didapatkan dari PrioritasKri1 dikali dengan Perbandingan1. NilaiB didapatkan dari PrioritasKri2 dikali dengan Perbandingan2. PrioritasKri1 dan PrioritasKri2 sudah didapatkan dari proses 3, sedangkan Perbandingan1 dan Perbandingan2 sudah didapatkan dari proses 1. JmlNilaiA didapatkan dari JmlNilaiA ditambah dengan NilaiA. JmlNilaiB didapatkan dari JmlNilaiB ditambah dengan NilaiB.
Proses 5	HasilA didapatkan dari JmlNilaiA dibagi dengan PrioritasKri1. HasilB didapatkan dari JmlNilaiB dibagi dengan PrioritasKri2. JmlNilaiA dan JmlNilaiB sudah didapatkan dari proses 4. JmlHasil didapatkan dari HasilA ditambah dengan HasilB.
<i>Decision</i>	Jika JmlKriteria sama dengan TotalKriteria maka lanjut ke proses 6, jika tidak maka kembali memasukkan data.
Proses 6	$\lambda$ max didapatkan dari JmlHasil yang sudah didapatkan dari proses 5 dibagi dengan N. N adalah jumlah total kriteria.
Proses 7	CI didapatkan dari $\lambda$ max - N lalu hasilnya dibagi dengan N. N adalah jumlah total kriteria.
Proses 8	CR didapatkan dari CI dibagi dengan IR. CI sudah didapatkan dari proses 7, sedangkan IR adalah <i>indeks random</i> dari jumlah total kriteria.
<i>Decision</i>	Jika CR lebih kecil dari 0,1 maka didapatkan hasil PrioritasKri1, PrioritasKri2 dan CR. Nilai-nilai ini akan digunakan untuk proses menghitung nilai produk tiap <i>supplier</i> . Jika CR lebih besar dari 0,1 maka harus dilakukan penilaian ulang.

Berikut ini merupakan gambar dari *flowchart* menghitung *intensity rating*.

Ada sepuluh proses pada *flowchart* menghitung *intensity rating*.



Gambar 3.11 Menghitung *Intensity rating*

Penjelasan *flowchart* menghitung *intensity rating* pada Gambar 3.11 bisa dilihat pada Tabel 3.17.

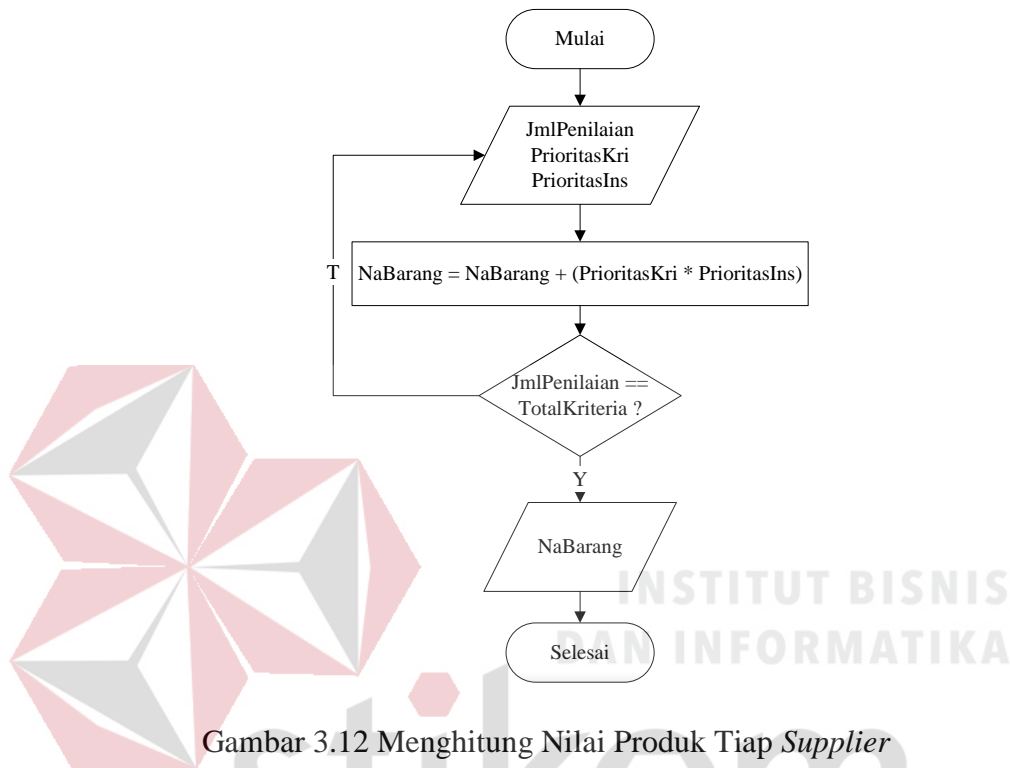
Tabel 3.17 Keterangan *Flowchart* Menghitung *Intensity rating*

Nama Proses	Keterangan
<i>Decision</i>	Jika JmlIR lebih kecil atau sama dengan 15 maka lanjut ke proses 1, jika tidak maka selesai.
Proses 1	Nilai yang dimasukkan akan disimpan menjadi Perbandingan1. Perbandingan2 didapatkan dari angka 1 dibagi dengan Perbandingan1. JmlPerbandingan1 didapatkan dari JmlPerbandingan1 ditambah dengan nilai yang ada pada Perbandingan1. JmlPerbandingan2 didapatkan dari JmlPerbandingan2 ditambah dengan nilai yang ada pada Perbandingan2. TotalIR didapatkan dari TotalIR ditambah dengan 1.
Proses 2	Normalisasi1 didapatkan dari Perbandingan1 dibagi dengan JmlPerbandingan1. Normalisasi2 didapatkan dari Perbandingan2



Nama Proses	Keterangan
	dibagi dengan $JmlPerbandingan2$ . $Perbandingan1$ , $JmlPerbandingan1$ , $Perbandingan2$ dan $JmlPerbandingan2$ sudah didapatkan dari proses 1. $JmlNormalisasi1$ didapatkan dari $JmlNormalisasi1$ ditambah dengan $Normalisasi1$ . $JmlNormalisasi2$ didapatkan dari $JmlNormalisasi2$ ditambah dengan $Normalisasi2$ .
Proses 3	A didapatkan dari $JmlNormalisasi1$ dibagi dengan N. B didapatkan dari $JmlNormalisasi2$ dibagi dengan N. $JmlNormalisasi1$ dan $JmlNormalisasi2$ sudah didapatkan dari proses 2. N adalah jumlah total <i>intensity rating</i> .
Decision	Jika A lebih besar dari B maka lanjut ke proses 4, jika A lebih kecil atau sama dengan B maka lanjut ke proses 5.
Proses 4	$PrioritasIns1$ didapatkan dari A dibagi dengan A. $PrioritasIns2$ didapatkan dari B dibagi dengan A. A dan B sudah didapatkan dari proses 3.
Proses 5	$PrioritasIns1$ didapatkan dari A dibagi dengan B. $PrioritasIns2$ didapatkan dari B dibagi dengan B. A dan B sudah didapatkan dari proses 3.
Proses 6	$NilaiA$ didapatkan dari A dikali dengan $Perbandingan1$ . $NilaiB$ didapatkan dari B dikali dengan $Perbandingan2$ . A dan B sudah didapatkan dari proses 3, sedangkan $Perbandingan1$ dan $Perbandingan2$ sudah didapatkan dari proses 1. $JmlNilaiA$ didapatkan dari $JmlNilaiA$ ditambah dengan $NilaiA$ . $JmlNilaiB$ didapatkan dari $JmlNilaiB$ ditambah dengan $NilaiB$ .
Proses 7	$HasilA$ didapatkan dari $JmlNilaiA$ dibagi dengan A. $HasilB$ didapatkan dari $JmlNilaiB$ dibagi dengan B. $JmlNilaiA$ dan $JmlNilaiB$ sudah didapatkan dari proses 6. $JmlHasil$ didapatkan dari $HasilA$ ditambah dengan $HasilB$ .
Decision	Jika $JmlIR$ sama dengan $TotalIR$ maka lanjut ke proses 8, jika tidak maka kembali memasukkan data.
Proses 8	$\lambda$ max didapatkan dari $JmlHasil$ yang sudah didapatkan dari proses 7 dibagi dengan N. N adalah jumlah total <i>intensity rating</i> .
Proses 9	CI didapatkan dari $\lambda$ max - N lalu hasilnya dibagi dengan N. N adalah jumlah total <i>intensity rating</i> .
Proses 10	CR didapatkan dari CI dibagi dengan IR. CI sudah didapatkan dari proses 9, sedangkan IR adalah <i>indeks random</i> dari jumlah total <i>intensity rating</i> .
Decision	Jika CR lebih kecil dari 0,1 maka didapatkan hasil $PrioritasIns1$ , $PrioritasIns2$ dan CR. Nilai-nilai ini akan digunakan untuk proses menghitung nilai produk tiap <i>supplier</i> . Jika CR lebih besar dari 0,1 maka harus dilakukan penilaian ulang.

Berikut ini merupakan gambar dari *flowchart* menghitung nilai produk tiap *supplier*. Hanya ada satu proses pada *flowchart* menghitung nilai produk tiap *supplier*.



Gambar 3.12 Menghitung Nilai Produk Tiap *Supplier*

Penjelasan *flowchart* menghitung nilai produk tiap *supplier* pada Gambar 3.12 bisa dilihat pada Tabel 3.18.

Tabel 3.18 Keterangan *Flowchart* Menghitung Nilai Produk Tiap *Supplier*

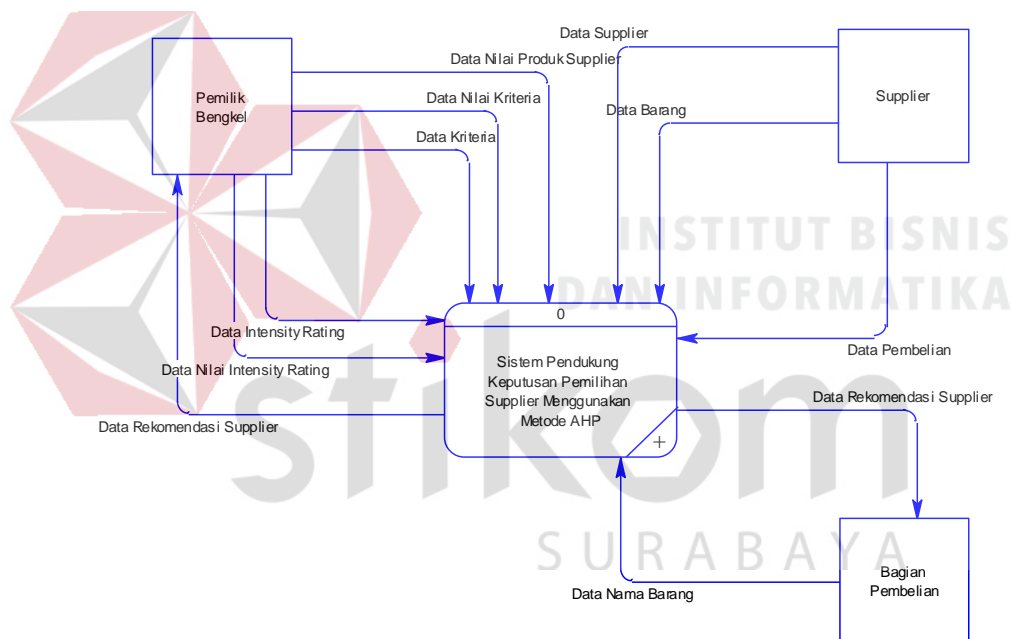
Nama Proses	Keterangan
Proses 1	NaBarang didapatkan dari nilai PrioritasKri dikali dengan PrioritasIns. PrioritasKri adalah prioritas kriteria yang dipilih oleh <i>user</i> . PrioritasIns adalah prioritas <i>intensity rating</i> yang dipilih oleh <i>user</i> . NaBarang adalah nilai akhir barang.
<i>Decision</i>	Jika JmlPenilaian sama dengan TotalKriteria maka mendapatkan hasil NaBarang, jika tidak maka kembali memasukkan data.

### 3.4.3 Data flow diagram

Data flow diagram (DFD) adalah gambaran aliran informasi yang terlibat dalam suatu prosedur yang terdapat dalam suatu sistem. Diagram ini menjelaskan alur data yang terjadi pada setiap proses. Adapun penjelasan dari DFD dapat dilihat sebagai berikut:

#### a Context diagram

DFD level *context* pada Gambar 3.13 ini akan menjelaskan bagaimana sistem pendukung keputusan ini akan berjalan pada Derry Auto Service.

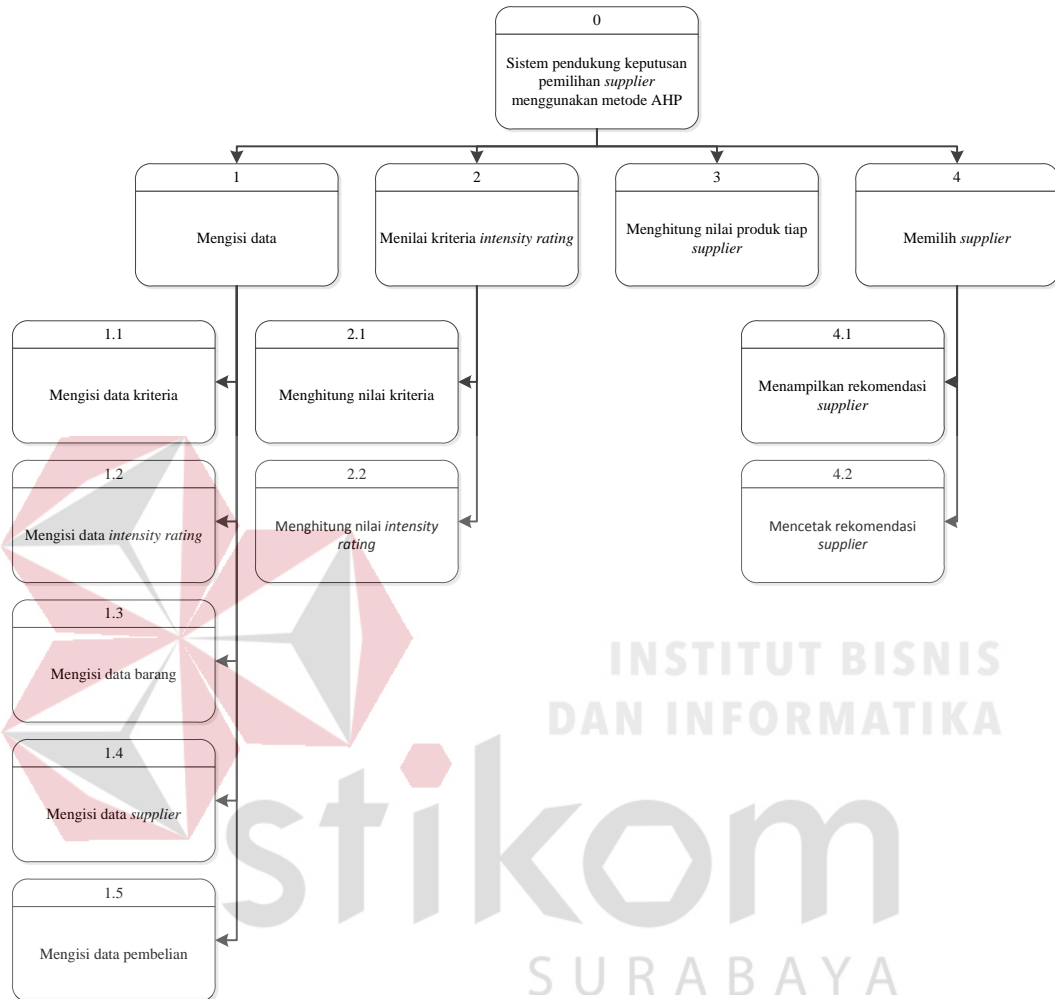


Gambar 3.13 *Context Diagram* Sistem Pendukung Keputusan untuk Pemilihan *Supplier* pada Derry Auto Service dengan Metode AHP

#### b Bagan berjenjang

Bagan berjenjang digunakan untuk mempersiapkan penggambaran DFD ke level-level lebih bawah lagi. Bagan berjenjang dapat digambar dengan menggunakan notasi proses yang digunakan di DFD. Untuk sistem pendukung keputusan ini, semua proses-proses yang ada bila digambarkan

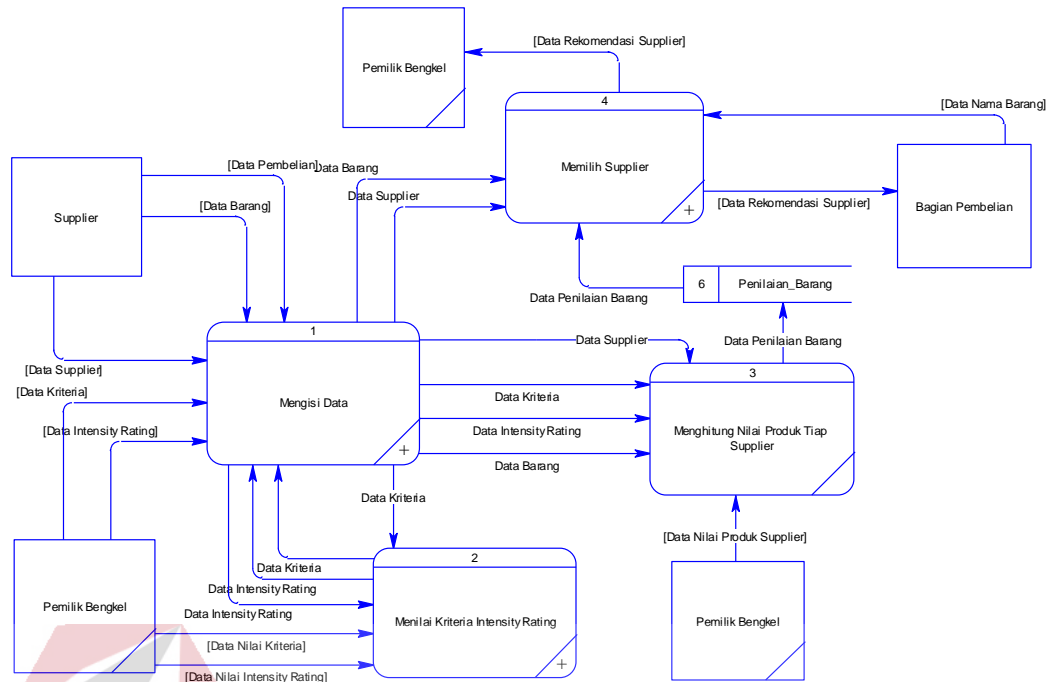
sesuai dengan jenjangnya dalam bagan berjenjang akan tampak seperti Gambar 3.14 berikut ini.



Gambar 3.14 Bagan Berjenjang Sistem Pendukung Keputusan untuk Pemilihan *Supplier* pada *Derry Auto Service* dengan Metode AHP

c Data *flow* diagram level 0

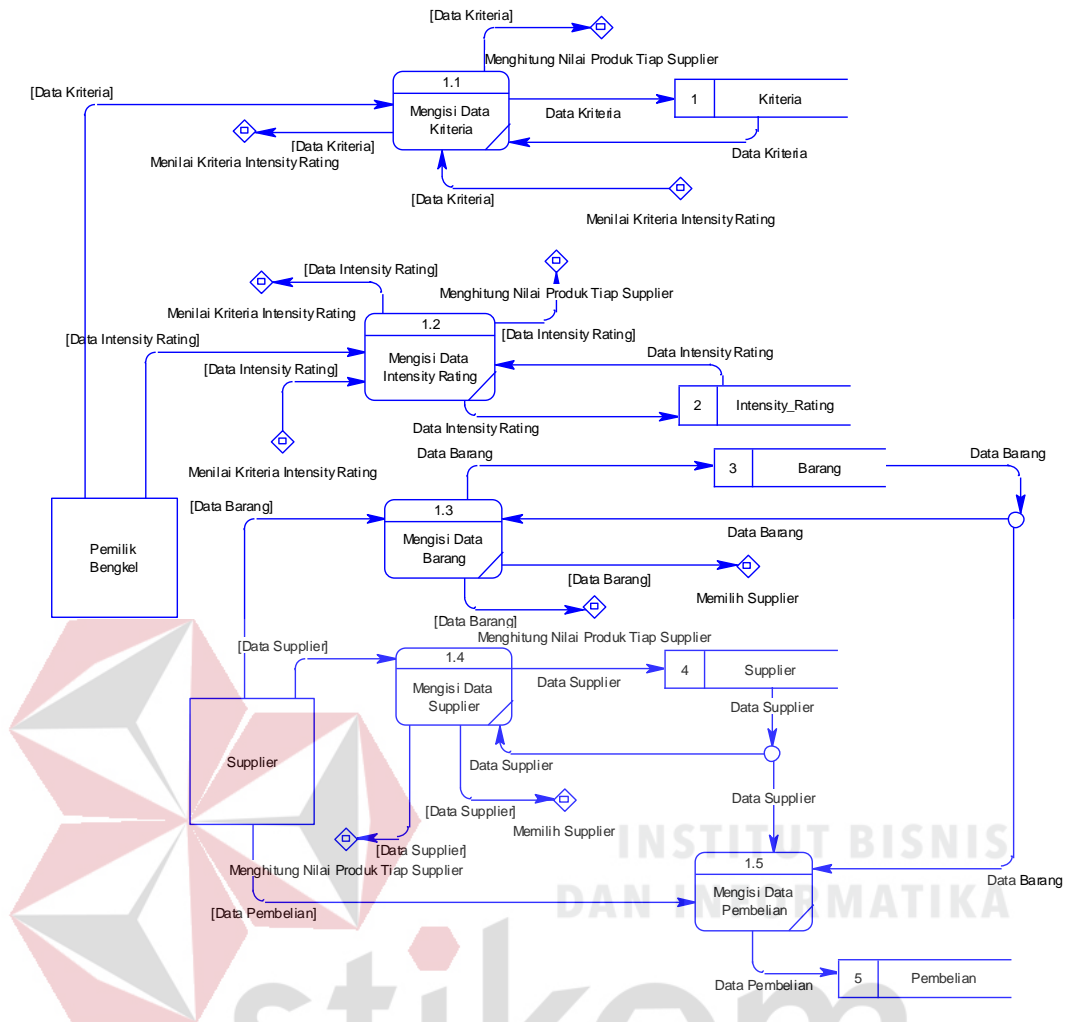
DFD level 0 pada Gambar 3.15 menjelaskan bagaimana sistem pendukung keputusan untuk pemilihan *supplier* yang dibangun dapat dijelaskan secara lebih detail dengan adanya sistem basis data. Sistem pendukung keputusan pada level 0 ini dapat dipecah menjadi 4 yaitu mengisi data, menilai kriteria *intensity rating*, menilai produk *supplier*, dan memilih *supplier*.



Gambar 3.15 DFD Level 0 Sistem Pendukung Keputusan untuk Pemilihan *Supplier* pada Derry Auto Service dengan Metode AHP

d Data flow diagram level 1 mengisi data

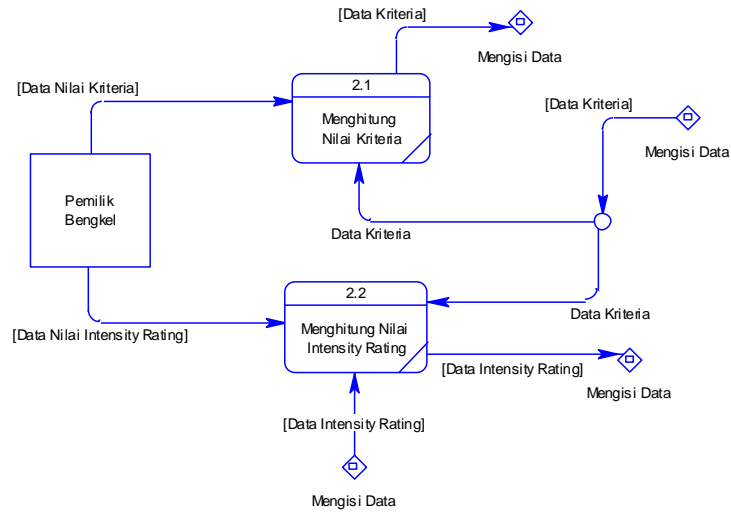
Gambar 3.16 ini merupakan penjabaran lebih detail lagi tentang DFD level 0 mengisi data. DFD level 1 mengisi data ini dijabarkan menjadi 5 proses yaitu mengisi data kriteria, mengisi data *intensity rating*, data barang, mengisi data *supplier*, dan mengisi data pembelian.



Gambar 3.16 DFD Level 1 Mengisi Data Sistem Pendukung Keputusan untuk Pemilihan *Supplier* pada Derry Auto Service dengan Metode AHP

e Data flow diagram level 1 menilai kriteria *intensity rating*

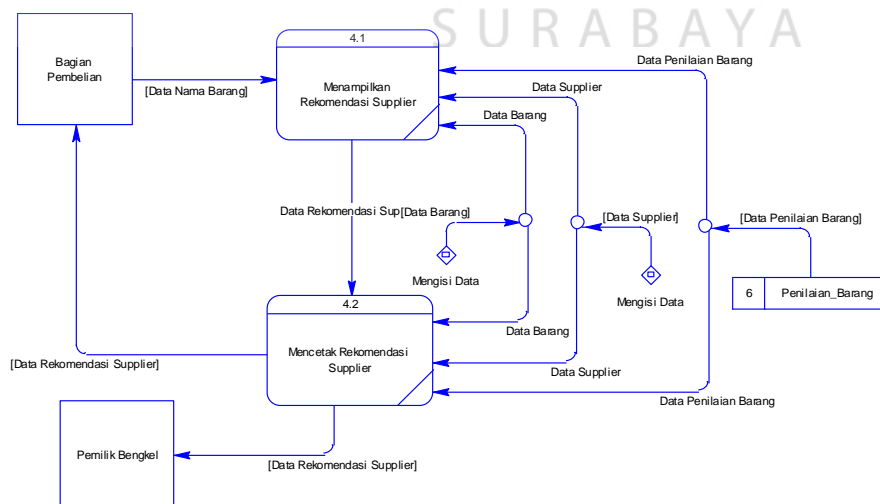
Gambar 3.17 ini merupakan penjabaran lebih detail lagi tentang DFD level 0 menghitung AHP. DFD level 1 menilai kriteria *intensity rating* ini dijabarkan menjadi 2 proses yaitu menghitung nilai kriteria dan menghitung nilai *intensity rating*.



Gambar 3.17 DFD Level 1 Menilai Kriteria *Intensity rating* Sistem Pendukung Keputusan untuk Pemilihan *Supplier* pada Derry Auto Service dengan Metode AHP

f Data flow diagram level 1 memilih *supplier*

Gambar 3.18 ini merupakan penjabaran lebih detail lagi tentang DFD level 0 memilih *supplier*. DFD level 1 memilih *supplier* ini dijabarkan menjadi 2 proses yaitu menampilkan rekomendasi *supplier* dan mencetak rekomendasi *supplier*.



Gambar 3.18 DFD Level 1 Memilih *Supplier* Sistem Pendukung Keputusan untuk Pemilihan *Supplier* pada Derry Auto Service dengan Metode AHP

### 3.4.4 *Entity relationship diagram*

*Entity relationship* diagram (ERD) adalah suatu desain sistem yang digunakan untuk merepresentasikan, menentukan, dan mendokumentasikan kebutuhan-kebutuhan untuk sistem pemrosesan database. ERD juga menyediakan bentuk untuk menunjukkan struktur keseluruhan dari data pemakai. Dalam perencanaan sistem ini telah terbentuk ERD yang merupakan lanjutan dari pembuatan desain dengan menggunakan DFD. Dalam ERD, data-data tersebut digambarkan dengan menggunakan simbol *entity*. Dalam perancangan sistem ini terdapat beberapa *entity* yang saling terkait untuk menyediakan data-data yang dibutuhkan oleh sistem, yaitu:

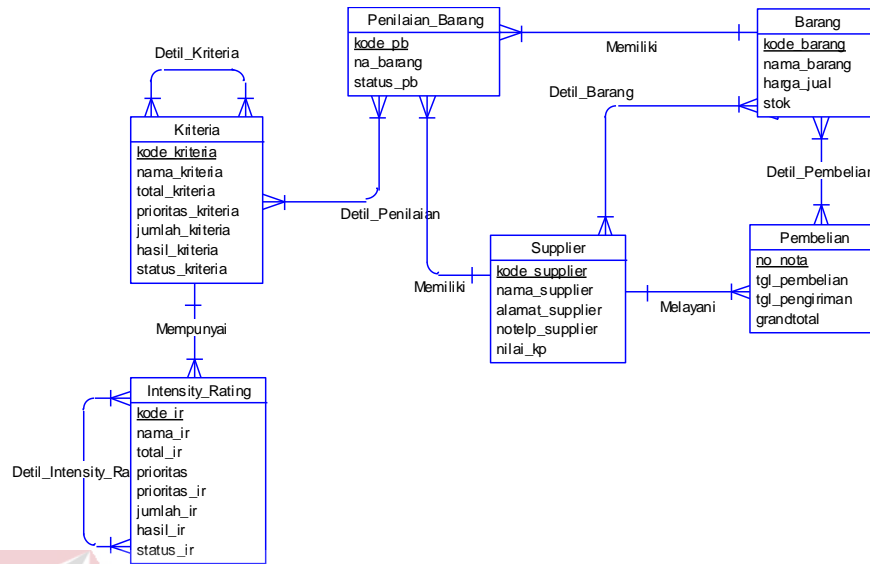
1. *Entity* kriteria
2. *Entity* detail kriteria
3. *Entity intensity rating*
4. *Entity* detail *intensity rating*
5. *Entity* barang
6. *Entity* detail barang
7. *Entity supplier*
8. *Entity* pembelian
9. *Entity* detail pembelian
10. *Entity* penilaian barang
11. *Entity* detail penilaian

INSTITUT BISNIS  
DAN INFORMATIKA

stikom  
SURABAYA

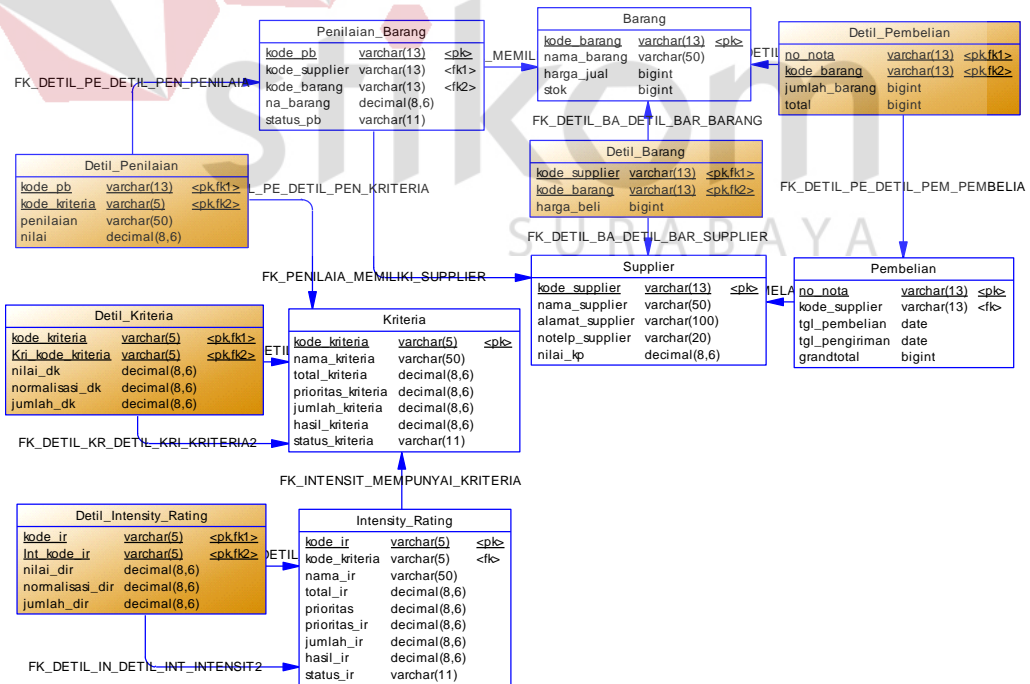


### A Conceptual data model



Gambar 3.19 Conceptual Data Model Sistem Pendukung Keputusan untuk Pemilihan Supplier pada Derry Auto Service dengan Metode AHP

### B Physical data model



Gambar 3.20 Physical Data Model Sistem Pendukung Keputusan untuk Pemilihan Supplier pada Derry Auto Service dengan Metode AHP

### 3.4.5 Struktur tabel

Dari PDM yang sudah terbentuk, dapat disusun struktur basis data yang nantinya akan digunakan untuk menyimpan data yang diperlukan yaitu:

#### 1. Tabel Kriteria

*Primary Key* : Kode\_kriteria

*Foreign Key* : -

Fungsi : Untuk mencatat semua data kriteria yang digunakan untuk pemilihan *supplier* sesuai dengan keinginan pemilik bengkel

Tabel 3.19 Struktur Kriteria

No	Field	Type	Length	Constraint
1.	KODE_KRITERIA	Varchar	5	PK
2.	NAMA_KRITERIA	Varchar	50	
3.	TOTAL_KRITERIA	Decimal	8,6	
4.	PRIORITAS_KRITERIA	Decimal	8,6	
5.	JUMLAH_KRITERIA	Decimal	8,6	
6.	HASIL_KRITERIA	Decimal	8,6	
7.	STATUS_KRITERIA	Varchar	11	

#### 2. Tabel Detil Kriteria

*Primary Key* : Kode\_kriteria, Kri\_kode\_kriteria

*Foreign Key* : Kode\_kriteria *references* dari tabel kriteria

Kri\_kode\_kriteria *references* dari tabel kriteria

Fungsi : Untuk mencatat semua nilai matriks dari tiap - tiap kriteria

Tabel 3.20 Struktur Detil Kriteria

No	Field	Type	Length	Constraint
1.	KODE_KRITERIA	Varchar	5	PK,FK
2.	KRI_KODE_KRITERIA	Varchar	5	PK,FK
3.	NILAI_DK	Decimal	8,6	
4.	NORMALISASI_DK	Decimal	8,6	
5.	JUMLAH_DK	Decimal	8,6	

3. Tabel *Intensity Rating*

*Primary Key* : Kode\_ir

*Foreign Key* : Kode\_kriteria *references* dari tabel kriteria

Fungsi : Untuk mencatat semua data *intensity rating* yang digunakan untuk pemilihan *supplier* sesuai dengan keinginan pemilik bengkel

Tabel 3.21 Struktur *Intensity Rating*

No	Field	Type	Length	Constraint
1.	KODE_IR	Varchar	5	PK
2.	KODE_KRITERIA	Varchar	5	FK
3.	NAMA_IR	Varchar	50	
4.	TOTAL_IR	Decimal	8,6	
5.	PRIORITAS	Decimal	8,6	
6.	PRIORITAS_IR	Decimal	8,6	
7.	JUMLAH_IR	Decimal	8,6	
8.	HASIL_IR	Decimal	8,6	
9.	STATUS_IR	Varchar	11	

4. Tabel Detil *Intensity Rating*

*Primary Key* : Kode\_ir, Int\_kode\_ir

*Foreign Key* : Kode\_ir *references* dari tabel *intensity rating*

Int\_kode\_ir *references* dari tabel *intensity rating*

Fungsi : Untuk mencatat semua nilai matriks dari tiap-tiap *intensity rating*

Tabel 3.22 Struktur Detil *Intensity rating*

No	Field	Type	Length	Constraint
1.	KODE_IR	Varchar	5	PK,FK
2.	INT_KODE_IR	Varchar	5	PK,FK
3.	NILAI_DIR	Decimal	8,6	
4.	NORMALISASI_DIR	Decimal	8,6	
5.	JUMLAH_DIR	Decimal	8,6	

## 5. Tabel Barang

*Primary Key* : Kode\_barang

*Foreign Key* : -

Fungsi : Untuk mencatat data barang yang ada pada Derry Auto Service.

Tabel 3.23 Struktur Barang

No	Field	Type	Length	Constraint
1.	KODE_BARANG	Varchar	13	PK
2.	NAMA_BARANG	Varchar	50	
3.	HARGA_JUAL	Bigint		
4.	STOK	Bigint		

## 6. Tabel Detil Barang

*Primary Key* : Kode\_supplier, Kode\_barang

*Foreign Key* : Kode\_supplier references dari tabel supplier

Kode\_barang references dari tabel barang

Fungsi : Untuk mencatat data barang dari tiap-tiap supplier.

Tabel 3.24 Struktur Detil Barang

No	Field	Type	Length	Constraint
1.	KODE_SUPPLIER	Varchar	13	PK,FK
2.	KODE_BARANG	Varchar	13	PK,FK
3.	HARGA_BELI	Bigint		

## 7. Tabel Supplier

*Primary Key* : Kode\_supplier

*Foreign Key* : -

Fungsi : Untuk mencatat semua data supplier

Tabel 3.25 Struktur *Supplier*

No	Field	Type	Length	Constraint
1.	KODE_SUPPLIER	Varchar	5	PK
2.	NAMA_SUPPLIER	Varchar	50	
3.	ALAMAT_SUPPLIER	Varchar	100	
4.	NOTELP_SUPPLIER	Varchar	20	
5.	NILAI_KP	Decimal	8,6	

## 8. Tabel Pembelian

*Primary Key* : No\_nota

*Foreign Key* : Kode\_supplier references dari tabel supplier

*Fungsi* : Untuk mencatat data pembelian dari tiap-tiap supplier

Tabel 3.26 Struktur Pembelian

No	Field	Type	Length	Constraint
1.	NO_NOTA	Varchar	13	PK
2.	KODE_SUPPLIER	Varchar	13	FK
3.	TGL_PEMBELIAN	Date		
4.	TGL_PENGIRIMAN	Date		
5.	GRANDTOTAL	Bigint		

## 9. Tabel Detil Pembelian

*Primary Key* : No\_nota, Kode\_barang

*Foreign Key* : No\_nota references dari tabel pembelian

Kode\_barang references dari tabel barang

*Fungsi* : Untuk mencatat barang yang dibeli oleh Derry Auto Service

Tabel 3.27 Struktur Detil Pembelian

No	Field	Type	Length	Constraint
1.	NO_NOTA	Varchar	13	PK,FK
2.	KODE_BARANG	Varchar	13	PK,FK

No	Field	Type	Length	Constraint
3.	JUMLAH_BARANG	Bigint		
4.	TOTAL	Bigint		

#### 10. Tabel Penilaian Barang

*Primary Key* : Kode\_pb

*Foreign Key* : Kode\_supplier references dari tabel supplier

Kode\_barang references dari tabel barang

*Fungsi* : Untuk mencatat data penilaian barang dari tiap-tiap supplier

Tabel 3.28 Struktur Penilaian Barang

No	Field	Type	Length	Constraint
1.	KODE_PB	Varchar	13	PK
2.	KODE_SUPPLIER	Varchar	13	FK
3.	KODE_BARANG	Varchar	13	FK
4.	NA_BARANG	Decimal	8,6	
5.	STATUS_PB	Varchar	11	

#### 11. Tabel Detil Penilaian

*Primary Key* : Kode\_pb, Kode\_kriteria

*Foreign Key* : Kode\_pb references dari tabel penilaian barang

Kode\_kriteria references dari tabel kriteria

*Fungsi* : Untuk mencatat nilai barang dari tiap-tiap supplier

Tabel 3.29 Struktur Detil Penilaian

No	Field	Type	Length	Constraint
1.	KODE_PB	Varchar	13	PKFK
2.	KODE_KRITERIA	Varchar	5	PKFK
3.	PENILAIAN	Varchar	50	
4.	NILAI	Decimal	8,6	

### 3.4.6 Desain antar muka

Desain antar muka merupakan rancangan *form-form* dari masing-masing tabel yang ada dalam aplikasi sistem pendukung keputusan. Desain antar muka ini dibuat sebagai acuan dalam membangun aplikasi Sistem Pendukung Keputusan Pemilihan Supplier dengan Metode *Analytical Hierarchy Process* (AHP). Adapun penjelasan desain antar muka dapat dilihat sebagai berikut:

#### a Desain *Form Login*

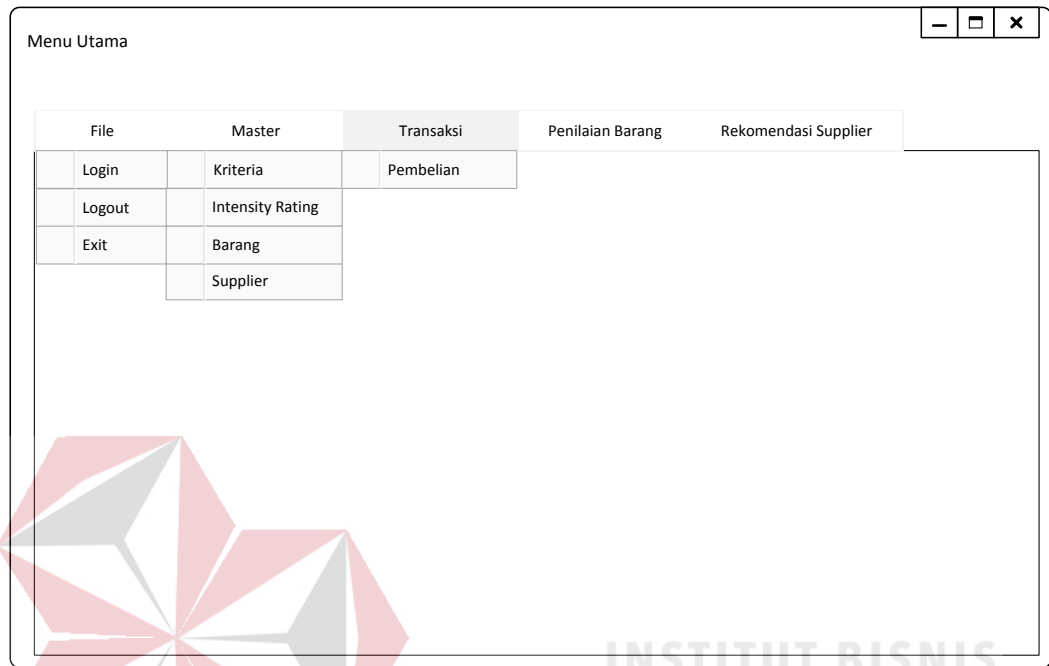
*Form* ini merupakan menu *login* yang berfungsi untuk keamanan sistem. Untuk dapat melakukan aktifitas dalam aplikasi ini, *user* diharuskan memasukkan *username* dan *password* yang benar. Desain *login* bisa dilihat pada Gambar 3.21.

Gambar 3.21 Desain *Login*

#### b Desain *Form Menu Utama*

*Form* ini merupakan menu utama yang tampil pada saat *user* memasukkan *username* dan *password* dengan benar. Pada menu master ada sub menu kriteria, *intensity rating*, *supplier*, dan barang. Pada menu transaksi ada sub menu pembelian. Pada menu penilaian barang tidak ada sub menu. Pada

menu rekomendasi *supplier* tidak ada sub menu. Desain menu utama bisa dilihat pada Gambar 3.22.



Gambar 3.22 Desain Menu Utama

c Desain *Form* Kriteria

*Form* ini merupakan *form* kriteria yang terdiri dari 2 bagian. Bagian sebelah kiri berguna untuk mencatat kriteria yang disimpan kedalam tabel kriteria. Bagian sebelah kanan atas berguna untuk menilai perbandingan antara satu kriteria dengan kriteria yang lain yang disimpan kedalam tabel detil kriteria. Bagian sebelah kanan tengah berguna untuk melihat hasil matriks nilai kriteria. Bagian sebelah kanan bawah berguna untuk melihat hasil matriks penjumlahan setiap baris. Desain Kriteria bisa dilihat pada Gambar 3.23.



Gambar 3.23 Desain Kriteria

#### d. Desain *Form Intensity rating*

*Form* ini merupakan *form intensity rating* yang terdiri dari 2 bagian. Bagian sebelah kiri berguna untuk mencatat *intensity rating* yang disimpan kedalam tabel *intensity rating*. Bagian sebelah kanan atas berguna untuk menilai perbandingan antara satu *intensity rating* dengan *intensity rating* yang lain yang disimpan kedalam tabel detail *intensity rating*. Bagian sebelah kanan tengah berguna untuk melihat hasil matriks nilai *intensity rating*. Bagian sebelah kanan bawah berguna untuk melihat hasil matriks penjumlahan setiap baris. Desain *intensity rating* bisa dilihat pada Gambar 3.24.

Gambar 3.24 Desain *Intensity rating*

### e Desain *Form* Barang

*Form* ini merupakan *form* barang yang berguna untuk mencatat data barang yang disimpan kedalam tabel barang. *User* harus memasukkan nama barang, harga jual, dan stok barang. Desain barang bisa dilihat pada Gambar 3.25.

Gambar 3.25 Desain Barang

f Desain *Form Supplier*

*Form* ini merupakan *form supplier* yang berguna untuk mencatat data-data *supplier* yang disimpan kedalam tabel *supplier*. Pada bagian sebelah kiri, *user* memasukkan nama, alamat, nomor telepon, dan barang-barang apa saja yang dimiliki *supplier*. Bagian sebelah kanan adalah data *supplier* yang sudah tersimpan kedalam tabel *supplier*. Desain *supplier* bisa dilihat pada Gambar 3.26.

Kode Supplier:	<input type="text"/>
Nama:	<input type="text"/>
Alamat:	<input type="text"/>
No. Telp:	<input type="text"/>
Nama Barang	Harga Jual
<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>

Nama Supplier	Alamat Supplier	Notelp Supplier	Nilai KP

Gambar 3.26 Desain *Form Supplier*

g Desain *Form Pembelian*

*Form* ini merupakan *form pembelian* yang berguna untuk mencatat data pembelian yang dilakukan oleh *Derry Auto Service* yang disimpan kedalam tabel pembelian. Pada bagian sebelah kiri, *user* memasukkan nama *supplier* yang sudah dicatat sebelumnya pada *form supplier*, tanggal pembelian, tanggal pengiriman, dan barang apa saja yang dibeli. Bagian sebelah kanan

adalah data pembelian yang sudah tersimpan kedalam tabel pembelian. Desain pembelian bisa dilihat pada Gambar 3.27.

The screenshot shows a software window titled "FormPembelian". On the left side, there is a form with the following fields: "No Nota:" (text input), "Nama Supplier:" (dropdown menu), "Tgl Pembelian:" (dropdown menu), and "Tgl Pengiriman:" (dropdown menu). Below these fields is a table with two columns: "Nama Barang" and "Jumlah Barang". At the bottom left of the form area are two buttons: "INSERT" and "UPDATE". To the right of the form is a large table with four columns: "Kode Supplier", "Tgl Pembelian", "Tgl Pengiriman", and "Grandtotal". The table is currently empty.

Gambar 3.27 Desain Pembelian

#### h Desain *Form* Penilaian Barang

*Form* ini merupakan *form* penilaian barang yang berguna untuk mencatat penilaian barang dari tiap *supplier*. Pada bagian sebelah kiri dan tengah, *user* memasukkan nama *supplier* yang sudah dicatat pada *form supplier*, nama barang, nama kriteria yang sudah dicatat pada *form kriteria* dan penilaian yang sudah dicatat pada *form intensity rating*. Bagian sebelah kanan adalah data harga beli dan kecepatan pengiriman tiap *supplier* yang berguna untuk membantu pemilik memberi penilaian terhadap *supplier*. Bagian sebelah bawah adalah data penilaian barang yang sudah tersimpan kedalam tabel penilaian barang. Desain penilaian barang bisa dilihat pada Gambar 3.28.

Gambar 3.28 Desain Penilaian Barang

i Desain *Form* Rekomendasi *Supplier*

*Form* ini merupakan *form* rekomendasi *supplier* yang berguna untuk melihat *supplier* terbaik dari tiap barang. *User* memasukkan nama barang yang diinginkan lalu sistem menampilkan laporan *supplier* mana saja yang memiliki penilaian terbaik. Desain rekomendasi *supplier* bisa dilihat pada Gambar 3.29.

Laporan Rekomendasi Supplier : Nama Barang				
Nama Supplier	Alamat Supplier	No Telp Supplier	Nilai Akhir Barang	Peringkat

Gambar 3.29 Desain Rekomendasi *Supplier*

### 3.5 Mengembangkan dan Mendokumentasikan Perangkat Lunak

Setelah membuat perancangan model, maka akan dilakukan pembuatan sistem sesuai dengan rancangan yang telah dibuat. Sistem Pendukung Keputusan Pemilihan *Supplier* pada *Derry Auto Service* dibuat dengan menggunakan bahasa pemrograman *visual basic.net* dan RDBMS (*Relational Database Management Sistem*) *SQL Server* 2008.

### 3.6 Menguji dan Mempertahankan Sistem

Desain uji coba menggunakan metode *blackbox equivalence partitioning*.

Desain uji coba berguna untuk memastikan bahwa aplikasi yang akan dibuat nanti melakukan perhitungan AHP dengan benar. Contoh kriteria yang digunakan untuk *test case* yaitu:

1. Merk : Terkenal, Cukup Terkenal, Tidak Terkenal
2. Harga : Sangat Mahal, Mahal, Sedang, Murah
3. Kualitas : Baik, Cukup, Kurang
4. Kecepatan Pengiriman : Sangat Cepat, Cepat, Sedang, Lambat

Tabel 3.30 Desain Uji Coba

No	Tujuan yang ingin dicapai	Input	Output yang diharapkan
1	Mampu melakukan pencatatan data kriteria	Nama kriteria	Semua data kriteria tersimpan
2	Mampu melakukan pencatatan data penilaian kriteria	Harga⇒merk=3, kualitas⇒ merk=7, kecepatan pengiriman⇒merk=5, kualitas⇒harga=5, kecepatan pengiriman⇒harga=3, kualitas⇒kecepatan pengiriman=3	Nilai konsistensi rasio = 0,044
3	Mampu melakukan pencatatan data <i>intensity rating</i>	Nama kriteria dan nama <i>intensity rating</i>	Semua data <i>intensity rating</i> tersimpan
4	Mampu melakukan	Kriteria merk, terkenal⇒cukup	Nilai

No	Tujuan yang ingin dicapai	Input	Output yang diharapkan
	pencatatan data penilaian <i>intensity rating</i> kriteria merk	terkenal=5, terkenal⇒tidak terkenal=9, cukup terkenal⇒tidak terkenal=3	konsistensi rasio = 0,025
5	Mampu melakukan pencatatan data penilaian <i>intensity rating</i> kriteria harga	Kriteria harga, mahal⇒sangat mahal=3, sedang⇒sangat mahal=5, murah⇒sangat mahal=7, sedang⇒mahal=3, murah⇒mahal=5, murah⇒sedang=3	Nilai konsistensi rasio = 0,044
6	Mampu melakukan pencatatan data penilaian <i>intensity rating</i> kriteria kualitas	Kriteria kualitas, baik⇒cukup=5, baik⇒kurang=9, cukup⇒kurang=3	Nilai konsistensi rasio = 0,025
7	Mampu melakukan pencatatan data penilaian <i>intensity rating</i> kriteria kecepatan pengiriman	Kriteria kecepatan pengiriman, sangat cepat⇒cepat=3, sangat cepat⇒sedang=5, sangat cepat⇒lambat=7, cepat⇒sedang=3, cepat⇒lambat=5, sedang⇒lambat=3	Nilai konsistensi rasio = 0,044
8	Mampu melakukan pencatatan data barang	Nama barang, harga jual, dan jumlah stok barang	Semua data barang tersimpan
9	Mampu melakukan pencatatan data <i>supplier</i>	Nama <i>supplier</i> , alamat <i>supplier</i> , nomor telepon <i>supplier</i> , nama barang <i>supplier</i> , dan harga beli	Semua data <i>supplier</i> tersimpan
10	Mampu melakukan pencatatan data pembelian	Nama <i>supplier</i> , tanggal pembelian, tanggal pengiriman, nama barang, dan jumlah barang	Semua data pembelian tersimpan
11	Mampu melakukan penilaian barang	Nama <i>supplier</i> , nama barang, merk= terkenal, harga=mahal, kualitas=baik, kecepatan pengiriman= sedang	Nilai produk <i>supplier</i> = 0,699
12	Memberikan rekomendasi <i>supplier</i>	Nama barang	Laporan rekomendasi <i>supplier</i>

Setelah melakukan desain uji coba, hasil aplikasi harus sama dengan desain uji coba. Jika hasil tidak sama, aplikasi harus dicek ulang apakah perhitungan AHP sudah benar. Jika hasil aplikasi dan desain uji coba sama, maka

akan terlihat *supplier* mana yang produknya menjadi pilihan terbaik. Hasil aplikasi tersebut akan diberikan kepada bagian pembelian dan akan menjadi laporan rekomendasi *supplier*.

