



**SISTEM PENUNJANG KEPUTUSAN PEMBELIAN PADI TERBAIK
MENGUNAKAN METODE SIMPLE MULTI ATTRIBUTE RATING
TECHNIQUE (SMART).**

(Studi Kasus UD. Kamal Lestari)

SKRIPSI

Oleh

Yofanda Putra Prayogi

NIM 132410101005

**PROGRAM STUDI SISTEM INFORMASI
UNIVERSITAS JEMBER**

2017



**SISTEM PENUNJANG KEPUTUSAN PEMBELIAN PADI TERBAIK
MENGUNAKAN METODE SIMPLE MULTI ATTRIBUTE RATING
TECHNIQUE (SMART).**

(Studi Kasus UD. Kamal Lestari)

SKRIPSI

Diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan pendidikan di Program Studi Sistem Informasi Universitas
Jember dan mendapat gelar Sarjana Sistem Informasi

Oleh

Yofanda Putra Prayogi

NIM 132410101005

**PROGRAM STUDI SISTEM INFORMASI
UNIVERSITAS JEMBER**

2017

PERSEMBAHAN

Skripsi ini saya persembahkan untuk :

1. Allah SWT.
2. Ayahanda Suprayogi dan Ibunda Siti Holifa.
3. Adik Adilah Amalia Putri.
4. UD Kamal Lestari.
5. Sahabat-sahabatku.
6. Guru-guru yang telah mendidikku.
7. Almamater Program Studi Sistem Informasi Universitas Jember.

MOTO

“Berpikirlah positif, sebab dengan berfikir positif lingkungan disekitarmu akan memancarkan energi positif pula kepadamu”

Anonymous



PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Yofanda Putra Prayogi

NIM : 132410101005

Menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Sistem Penunjang Keputusan Pembelian Padi Terbaik Menggunakan Metode Simple Multi Attribute Rating Technique (SMART) (Studi Kasus UD. Kamal Lestari)” adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, belum pernah diajukan pada institusi manapun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika di kemudian hari pernyataan ini tidak benar.

Jember, 26 Mei 2017

Yang menyatakan,

Yofanda Putra Prayogi

NIM 132410101005

SKRIPSI

**SISTEM PENUNJANG KEPUTUSAN PEMBELIAN PADI TERBAIK
MENGUNAKAN METODE SIMPLE MULTI ATTRIBUTE RATING
TECHNIQUE (SMART).**

(Studi Kasus UD. Kamal Lestari)

Oleh :

Yofanda Putra Prayogi

132410101005

Pembimbing :

Dosen Pembimbing Utama : Drs. Antonius C.P., M.App.Sc.,Ph.D.

Dosen Pembimbing Pendamping : Windi Eka Yulia Retani, S.Kom., M.T.

PENGESAHAN PEMBIMBING

Skripsi berjudul “Sistem Penunjang Keputusan Pembelian Padi Terbaik Menggunakan Metode Simple Multi Attribute Rating Technique (SMART) (Studi Kasus UD. Kamal Lestari)”, telah diuji dan disahkan pada:

Hari, Tanggal : Jum’at, 26 Mei 2017

Tempat : Program Studi Sistem Informasi Universitas Jember.

Disetujui oleh:

Pembimbing I,

Pembimbing II,

Drs. Antonius C.P., M.App.Sc.,Ph.D.
NIP. 196909281993021001

Windy Eka Yulia Retani, S.Kom., M.T.
NIP. 198403052010122002

PENGESAHAN PENGUJI

Skripsi berjudul “Sistem Penunjang Keputusan Pembelian Padi Terbaik Menggunakan Metode Simple Multi Attribute Rating Technique (SMART) (Studi Kasus UD. Kamal Lestari)”, telah diuji dan disahkan pada:

Hari, Tanggal : Jum’at, 26 Mei 2017

Tempat : Program Studi Sistem Informasi Universitas Jember

Tim Penguji :

Penguji I,

Penguji II,

Prof. Dr.Saiful Bukhori, ST., M.Kom

Muhamad Arief Hidayat S.Kom,.M.Kom.

NIP. 196811131994121001

NIP. 198101232010121003

Mengesahkan

Ketua Program Studi

Prof. Drs. Slamin, M.Comp.Sc.,Ph.D

NIP. 196704201992011001

RINGKASAN

Sistem Penunjang Keputusan Pembelian Padi Terbaik Menggunakan Metode Simple Multi Attribute Rating Technique (SMART) (Studi Kasus: UD. Kamal Lestari); Yofanda Putra Prayogi, 132410101005 2017, 157 halaman ; Program Studi Sistem Informasi Universitas Jember.

Sistem Penunjang Keputusan Pembelian Padi Terbaik Menggunakan Metode Simple Multi Attribute Rating Technique (SMART) merupakan sistem penunjang keputusan yang ditujukan untuk produsen beras atau pabrik untuk dapat membeli beras berdasarkan mutu terbaik yang diinginkan pabrik. Pembelian beras oleh pabrik seringkali mengalami kesulitan dalam menjaga mutu produksi beras melalui pembelian stok gabah maupun beras, dengan adanya sistem ini diharapkan para pekerja dapat membeli gabah dan beras dengan mudah dan mendapatkan klasifikasi mutu dan harga pasaran saat itu juga. Pengguna dalam sistem adalah admin selaku pemilik dan pekerja atau pegawai. Sistem mengelola proses pembelian bahan baku pembuatan beras mulai dari proses penentuan stok yang diinginkan oleh pemilik.

Hasil perhitungan sistem adalah berupa perangkaan, klasifikasi mutu dan harga pasaran padi. Hasil tersebut dihitung menggunakan metode SMART dengan mengolah kriteria, subkriteria, pembobotan, dan nilai *Utility*. Range penentuan mutu gabah adalah, mutu satu dengan nilai lebih dari sama dengan 61,250. Mutu dua dengan range nilai lebih dari sama dengan 50,000 sampai kurang dari 61,250. Mutu tiga memiliki range kurang dari 50,000 sampai lebih dari sama dengan 38,750, Sedangkan kurang dari 38,750 dianggap sebagai gabah mutu tiga. Penentuan range mutu beras adalah sebagai berikut, mutu satu 73,158, mutu dua 60,175, mutu tiga 48,596, mutu empat 41,462, dan mutu lima 28,655, sedangkan untuk nilai kurang dari 2,655 masuk pada beras mutu lima. Perangkaan ditampilkan berdasarkan nilai perhitungan SMART. Penentuan harga gabah maupun beras mengacu pada nilai mutu yang ditetapkan oleh sistem, perlu diketahui bahwa harga pasaran sewaktu – waktu dapat berubah.

PRAKATA

Puji syukur kehadiran Allah SWT atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul “Sistem Penunjang Keputusan Pembelian Padi Terbaik Menggunakan Metode Simple Multi Attribute Rating Technique (SMART) (Studi Kasus UD. Kamal Lestari)”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Strata Satu (S1) pada Program Studi Sistem Informasi Universitas Jember.

Penyusunan skripsi ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Prof. Drs. Slamini, M.Comp.Sc., Ph.D., selaku Ketua Program Studi Sistem Informasi Universitas Jember dan Pembimbing akademik;
2. Drs. Antonius Cahya Prihandoko M.App.Sc., Ph.D. selaku Dosen Pembimbing Utama dan Winda Eka Yulia Retnani, S.Kom., M.T. selaku Dosen Pembimbing Pendamping yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi;
3. Seluruh Bapak dan Ibu dosen beserta staf karyawan di Program Studi Sistem Informasi Universitas Jember.
4. Keluarga Besar Suprayogi, ibunda Siti Holifa, dan adik Adilah Amalia Putri yang selalu mendukung dan mendoakan.
5. Teman – temanku grup kadal dan bbm yang selalu menemani.
6. Yusuf Eka Sayogana, Andre Hardika, Safitri Febryanti Astuti, yang membantu penulis.
7. Fikriatul Hidayah yang menjadi motivasi untuk dapat sampai di tahap ini.
8. Teman-teman seperjuangan Intention angkatan 2013.
9. Segenap keluarga besar Laboratorium RPL periode 2016/2017
10. Teman-teman Program Studi Sistem Informasi seluruh angkatan.
11. Semua pihak yang tidak dapat disebutkan satu persatu.

Besar harapan bahwa penelitian ini nantinya akan terus berlanjut dan berkembang kelak, penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan skripsi ini. Akhirnya penulis berharap, semoga skripsi ini dapat bermanfaat.

Jember, 26 Mei 2017

Penulis



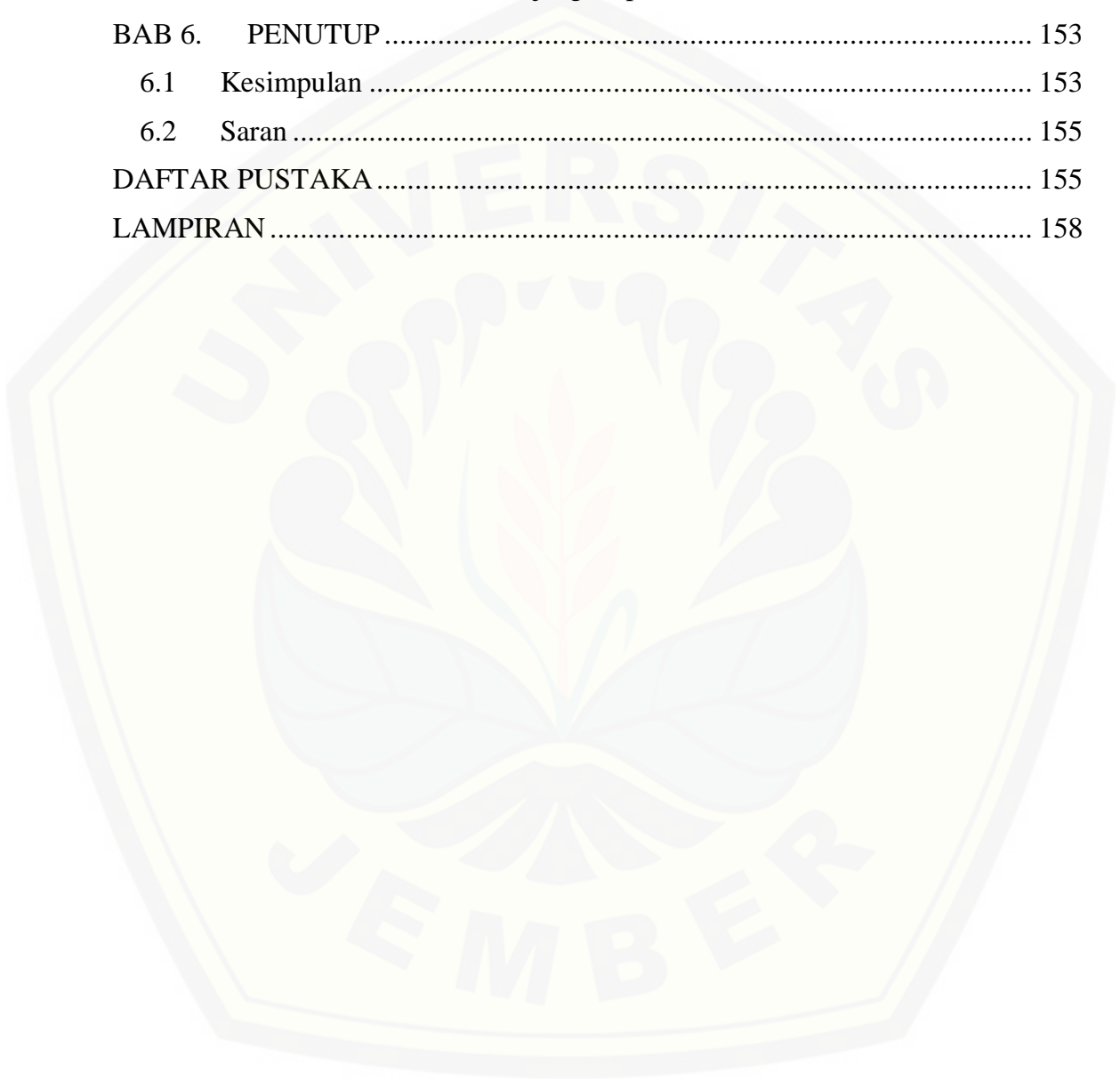
DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSEMBAHAN	ii
HALAMAN MOTO	iii
HALAMAN PERYATAAN	iv
HALAMAN PENGESAHAN PEMBIMBING	vi
HALAMAN PENGESAHAN PENGUJI	vii
RINGKASAN	viii
PRAKATA	ix
DAFTAR ISI	xi
DAFTAR TABEL	xv
DAFTAR GAMBAR	xvii
DAFTAR LAMPIRAN	xxi
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian	3
1.5 Batasan Masalah	4
1.6 Sistematika Penulisan	5
BAB 2. TINJAUAN PUSTAKA	7
2.1 Penelitian Terdahulu	7
2.2 Gabah	9
2.3 Beras	10
2.4 Sistem Pendukung Keputusan	11
2.5 Metode Simple Multi Atribut Rating Technique (SMART)	13
BAB 3. METODOLOGI PENELITIAN	16
3.1 Jenis Penelitian	16

3.2	Tempat dan Waktu Penelitian	16
3.3	Alur Penelitian.....	16
3.4	Metode Pengembangan Sistem	17
3.4.1	Requirement Analysis (Analisis Kebutuhan).....	18
3.4.2	System Design (Desain Sistem)	23
3.4.3	Implementation System (Implementasi Sistem)	24
3.4.4	Testing System (Pengujian Sistem).....	24
3.4.5	Maintenance System (Perawatan sistem)	25
3.5	Gambaran Umum Sistem.....	25
BAB 4.	PENGEMBANGAN SISTEM.....	29
4.1	Analisis Kebutuhan Perangkat Lunak	29
4.1.1	Kebutuhan Fungsional	29
4.1.2	Kebutuhan Non Fungsional	31
4.2	Desain Sistem.....	31
4.2.1	<i>Business Process</i>	31
4.2.2	<i>Use Case Diagram</i>	32
4.2.3	<i>Usecase Scenario</i>	36
4.2.4	<i>Activity Diagram</i>	53
4.2.5	<i>Sequence Diagram</i>	59
4.2.6	<i>Class Diagram</i>	68
4.2.7	<i>Entity Relations Diagram</i>	70
4.3	Implementasi	71
4.4	Pengujian Sistem	80
4.4.1	Pengujian <i>White Box</i>	80
4.4.2	Pengujian <i>Black Box</i>	91
BAB 5.	HASIL DAN PEMBAHASAN	99
5.1	Hasil Penerapan Perhitungan Metode Simple Multi Attribute Rating Technique (SMART).....	99
5.1.1	Data Kriteria.....	99

5.1.2	Data Sub Kriteria.....	101
5.1.3	<i>Range</i> Mutu dan Harga Pembelian Padi	104
5.2	Hasil Implementasi <i>Coding</i> Pada Sistem Penunjang Keputusan Pembelian Padi Terbaik dengan Metode Simple Multi Attribute Rating Technique (SMART) 104	
5.2.1	Halaman Masuk (<i>Login</i>)	105
5.2.2	Halaman <i>Dashboard</i> Admin	106
5.2.3	Halaman Mengelola Pengguna.....	106
5.2.4	Halaman Mengelola Kriteria Gabah.....	108
5.2.5	Halaman Mengelola Kriteria Beras	111
5.2.6	Halaman Mengelola Stok Pembelian Beras.....	114
5.2.7	Halaman Mengelola Stok Pembelian Gabah	115
5.2.8	Halaman Mengelola Penilaian Mutu dan Harga Beras	117
5.2.9	Mengelola Penilaian Mutu dan Harga Gabah	119
5.2.10	Halaman Mengelola Pemasok Beras	120
5.2.11	Halaman Mengelola Pemasok Gabah.....	123
5.2.12	Halaman Membuat Penilaian Pemasok Beras.....	125
5.2.13	Halaman Membuat Penilaian Pemasok Gabah	125
5.2.14	Halaman Melihat Penilaian Gabah.....	126
5.2.15	Halaman Melihat Penilaian Beras	128
5.2.16	Halaman Mengelola Pembelian Gabah	129
5.2.17	Halaman Mengelola Pembelian Beras.....	130
5.2.18	Halaman Melihat Riwayat Pembelian Gabah	131
5.2.19	Halaman Melihat Riwayat Pembelian Beras	132
5.2.20	Halaman Keluar.....	132
5.3	Pengujian Sistem Penunjang Keputusan Pembelian Padi Terbaik dengan Metode <i>Simple Multi Attribute Rating Technique</i> (SMART)	133
5.3.1	Perhitungan Manual Nilai Pembelian Gabah.....	133
5.3.2	Perhitungan Manual Nilai Pembelian Beras	139
5.3.3	Perbandingan Hasil Perhitungan Manual dengan Perhitungan Sistem	145

5.4 Hasil Implementasi Metode SMART pada Sistem Penunjang Keputusan Pembelian Padi Terbaik.....	147
5.5 Pembahasan Sistem Penunjang Keputusan Pembelian Padi Terbaik... ..	151
BAB 6. PENUTUP	153
6.1 Kesimpulan	153
6.2 Saran	155
DAFTAR PUSTAKA.....	155
LAMPIRAN	158



DAFTAR TABEL

Tabel 2.1 Spesifikasi persyaratan mutu Gabah	9
Tabel 2.2 Spesifikasi persyaratan mutu beras	11
Tabel 3.1 Tabel Kriteria dan Subkriteria Gabah	20
Tabel 3.2 Tabel Kriteria dan Subkriteria Beras	20
Tabel 3.3 Tabel Uji Black Box.....	25
Tabel 4.1 Definisi Aktor	34
Tabel 4.2 Definisi Use Case	34
Tabel 4.3 Skenario Mengelola Kriteria, Subkriteria, dan Pembobotan Gabah	37
Tabel 4.4 Skenario Membuat Penilaian Pemasok Gabah	47
Tabel 4.5 Test Case Fungsi insertKgabah.....	82
Tabel 4.6 Test Case Fungsi Totalbobot	84
Tabel 4.7 Test Case Fungsi perhitunganSMARTgabah	86
Tabel 4.8 Test Case Fungsi perhitunganMutugabah	88
Tabel 4.9 Test Case Fungsi get_nilaimutugabah.....	90
Tabel 4.10 Pengujian Blackbox Fitur SPK Pembelian Padi Terbaik	91
Tabel 5.1 Tabel Kriteria Pembelian Gabah.....	100
Tabel 5.2 Tabel Kriteria Pembelian Beras	101
Tabel 5.3 Tabel Subkriteria Penilaian Gabah.....	102
Tabel 5.4 Tabel Subkriteria Penilaian Beras	103
Tabel 5.5 Tabel Range Mutu Gabah.....	104
Tabel 5.6 Tabel Range Mutu Beras	104
Tabel 5.7 Data Pemasok Gabah Pertama	133
Tabel 5.8 Data Penilaian Pemasok Gabah Pertama.....	134
Tabel 5.9 Data Pemasok Gabah Kedua	135
Tabel 5.10 Data Penilaian Pemasok Gabah Kedua	136
Tabel 5.11 Data Pemasok Gabah Ketiga	137

Tabel 5.12 Data Penilaian Pemasok Gabah Ketiga	138
Tabel 5.13 Data Pemasok Beras Pertama	140
Tabel 5.14 Data Penilaian Pemasok Beras Pertama	140
Tabel 5.15 Data Pemasok Beras Kedua	142
Tabel 5.16 Data Penilaian Pemasok Beras Kedua.....	142
Tabel 5.17 Data Pemasok Beras Ketiga.....	143
Tabel 5.18 Data Penilaian Pemasok Beras Ketiga	144
Tabel 5.19 Perbandingan Perhitungan Manual dengan Sistem untuk Pemasok Gabah	146
Tabel 5.20 Perbandingan Perhitungan Manual dengan Sistem untuk Pemasok Beras	146

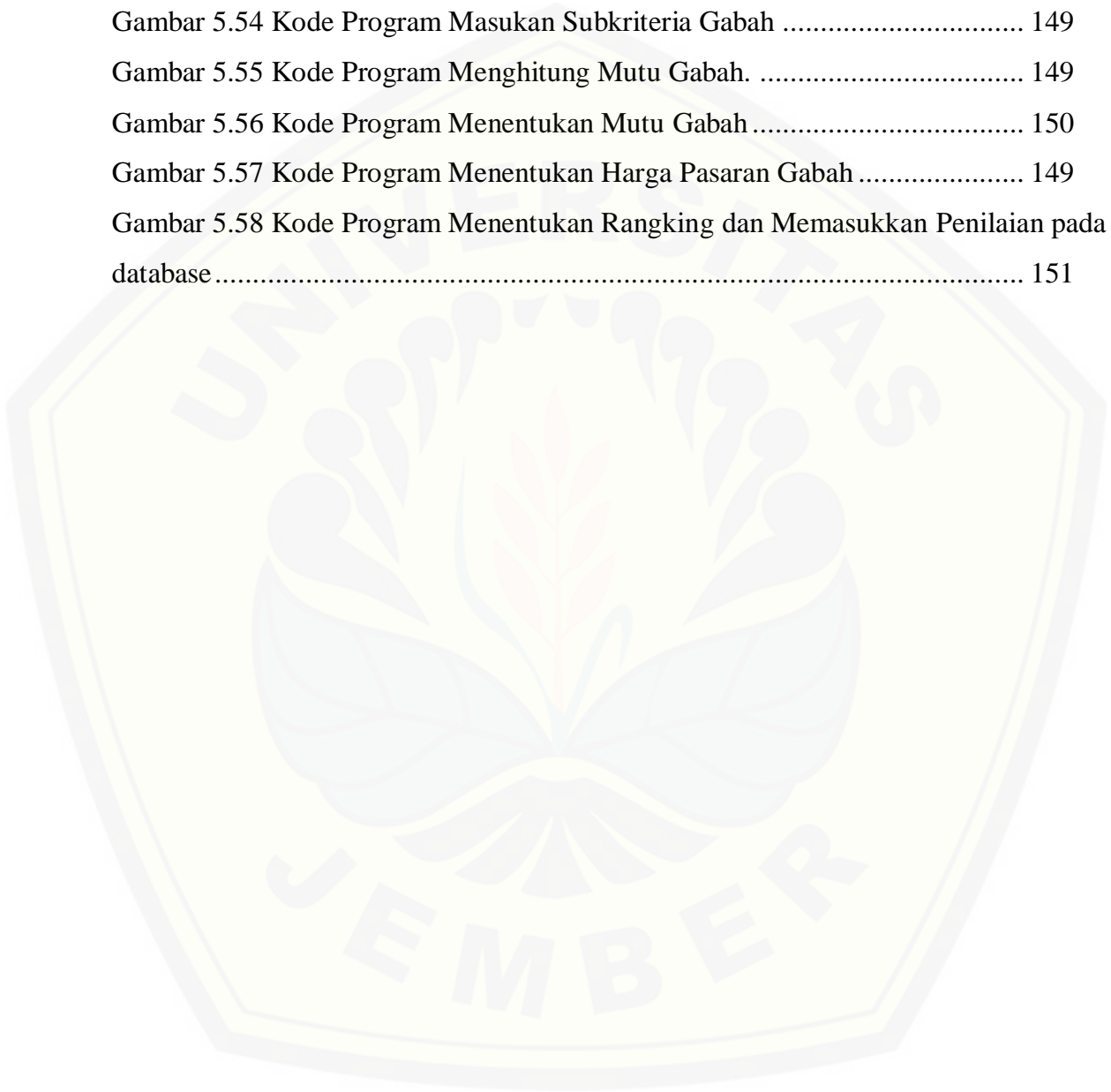
DAFTAR GAMBAR

Gambar 2.1 Karakteristik DSS	12
Gambar 2.2 Diagram alir metode SMART	15
Gambar 3.1 Alur Penelitian.....	17
Gambar 3.2 Model Waterfall	18
Gambar 3.3 Diagram alir penerapan metode SMART pada SPK Pembelian Padi Terbaik	22
Gambar 3.4 Flowchart admin / pakar	27
Gambar 3.5 Flowchart User / Pengguna	28
Gambar 4.1 Business Process.....	32
Gambar 4.2 Use Case Diagram	33
Gambar 4.3 Activity Diagram Mengelola Kriteria, Subkriteria, dan Pembobotan Gabah	55
Gambar 4.4 Activity diagram Membuat Penilaian Pemasok Gabah	57
Gambar 4.5 Sequence Diagram Mengelola Mengelola Kriteria, Subkriteria, dan Pembobotan Gabah.....	62
Gambar 4.6 Sequence diagram Membuat Penilaian Pemasok Gabah	66
Gambar 4.7 Class diagram Sistem Penunjang Keputusan Pembelian Padi Terbaik	69
Gambar 4.8 Entity Relationship Diagram.....	70
Gambar 4.9 Kode Program Kelas C_kriteriaGabah	73
Gambar 4.10 Kode Program Kelas M_kriteriaGabah	75
Gambar 4.11 Kode Program Kelas C_PenilaianGabah	77
Gambar 4.12 Kode Program Kelas M_penilaianGabah	80
Gambar 4.13 Listing Program Fungsi insertKgabah	81
Gambar 4.14 Diagram Alir Fungsi insertKgabah.....	81
Gambar 4.15 Listing Program Fungsi Totalbobot.....	83
Gambar 4.16 Diagram Alir Fungsi TotalBobot.....	83

Gambar 4.17 Listing Program Fungsi perhitunganSMARTgabah.....	85
Gambar 4.18 Diagram Alir Fungsi perhitunganSMARTgabah	85
Gambar 4.19 Listing Program fungsi perhitunganMutugabah	87
Gambar 4.20 Diagram Alir Fungsi perhitunganMutugabah	87
Gambar 4.21 Listing Program Fungsi get_nilaimutugabah	89
Gambar 4.22 Diagram Alir Fungsi get_nilaimutugabah.....	89
Gambar 5.1 Halaman Awal Sistem	105
Gambar 5.2 Halaman Login Sistem	105
Gambar 5.3 Halaman Dashboard Admin	106
Gambar 5.4 Halaman Hak Akses Pengguna	107
Gambar 5.5 Modal Tambah Data Pengguna	107
Gambar 5.6 Modal Ubah Data Pengguna	108
Gambar 5.7 Alert Hapus Data Pengguna	108
Gambar 5.8 Halaman Mengelola Kriteria Gabah.....	109
Gambar 5.9 Modal Tambah Kriteria Gabah	109
Gambar 5.10 Modal Tambah Subkriteria Gabah	110
Gambar 5.11 Modal Detail Subkriteria Gabah.....	110
Gambar 5.12 Pesan Konfirmasi Hapus Subkriteria Gabah.....	111
Gambar 5.13 Pesan Konfirmasi Hapus Kriteria Gabah	111
Gambar 5.14 Halaman Mengelola Kriteria Beras	112
Gambar 5.15 Modal Tambah Kriteria Beras	112
Gambar 5.16 Modal Tambah Subkriteria Beras.....	112
Gambar 5.17 Modal Detail Subkriteria Beras	113
Gambar 5.18 Modal Hapus Subkriteria Beras	113
Gambar 5.19 Pesan Konfirmasi Hapus Kriteria Beras	113
Gambar 5.20 Halaman Utama Mengelola Stok Pembelian Beras.....	114
Gambar 5.21 Modal Tambah Pembelian Stok Beras.....	115
Gambar 5.22 Modal Pengurangan Pembelian Stok Beras	115

Gambar 5.23 Halaman Utama Kelola Stok Pembelian Gabah.....	116
Gambar 5.24 Modal Penambahan Pembelian Gabah	116
Gambar 5.25 Modal Pengurangan Pembelian Stok Gabah.....	117
Gambar 5.26 Halaman Utama Penilaian Mutu dan Harga Beras	117
Gambar 5.27 Modal Penilaian Mutu dan Harga Beras	118
Gambar 5.28 Pesan Konfirmasi Reset Ulang Penilaian Mutu dan Harga Beras.	119
Gambar 5.29 Halaman Utama Penilaian Mutu dan Harga Gabah.....	119
Gambar 5.30 Modal Penilaian Mutu dan Harga Gabah.....	120
Gambar 5.31 Pesan Konfirmasi Reset Ulang Penilaian Mutu dan Harga Gabah	120
Gambar 5.32 Halaman Utama Mengelola Pemasok Beras	121
Gambar 5.33 Modal Tambah Pemasok Beras	121
Gambar 5.34 Modal Ubah Data Pemasok Beras	122
Gambar 5.35 Pesan Konfirmasi Hapus Data Pemasok Beras	122
Gambar 5.36 Halaman Utama Mengelola Pemasok Gabah.....	123
Gambar 5.37 Modal Tambah Pemasok Gabah.....	124
Gambar 5.38 Modal Ubah Data Pemasok Gabah.....	124
Gambar 5.39 Pesan Konfirmasi Hapus Data Pemasok Gabah.....	124
Gambar 5.40 Halaman Penilaian Pemasok Beras	125
Gambar 5.41 Halaman Penilaian Pemasok Gabah	126
Gambar 5.42 Halaman Melihat Hasil Penilaian Gabah	127
Gambar 5.43 Modal Detail Penilaian Pemasok Gabah.....	127
Gambar 5.44 Halaman Melihat Hasil Penilaian Beras	128
Gambar 5.45 Modal Detail Penilaian Pemasok Beras	129
Gambar 5.46 Tampilan Modal Beli Gabah	129
Gambar 5.47 Tampilan Modal Beli Beras	131
Gambar 5.48 Halaman Melihat Riwayat Pembelian Gabah	131
Gambar 5.49 Halaman Melihat Riwayat Pembelian Beras.....	132
Gambar 5.50 Tampilan Fitur Logout.....	132

Gambar 5.52 Kode Program Memasukkan Kriteria Gabah dan Beras.....	147
Gambar 5.53 Kode Program Perhitungan Bobot dan Normalisasi.....	148
Gambar 5.54 Kode Program Masukan Subkriteria Gabah	149
Gambar 5.55 Kode Program Menghitung Mutu Gabah.	149
Gambar 5.56 Kode Program Menentukan Mutu Gabah	150
Gambar 5.57 Kode Program Menentukan Harga Pasaran Gabah	149
Gambar 5.58 Kode Program Menentukan Rangkaing dan Memasukkan Penilaian pada database.....	151



DAFTAR LAMPIRAN

LAMPIRAN A. <i>USE CASE SKENARIO</i>	158
A.1 <i>Use Case Skenario Login</i>	158
A.2 <i>Use Case Skenario Mengelola Data Pengguna</i>	160
A.3 <i>Use Case Skenario Mengelola Kriteria Beras</i>	168
A.4 <i>Use Case Skenario Mengelola Stok Pembelian Gabah</i>	181
A.5 <i>Use Case Skenario Mengelola Stok Pembelian Beras</i>	186
A.6 <i>Use Case Skenario Mengelola Penilaian Mutu dan Harga Gabah</i>	191
A.7 <i>Use Case Skenario Mengelola Penilaian Mutu dan Harga Beras</i>	196
A.8 <i>Use Case Skenario Mengelola Pemasok Gabah</i>	202
A.9 <i>Use Case Skenario Mengelola Pemasok Beras</i>	209
A.10 <i>Use Case Skenario Membuat Penilaian Pemasok Beras</i>	217
A.11 <i>Use Case Skenario Melihat Penilaian Gabah</i>	221
A.12 <i>Use Case Skenario Melihat Penilaian Beras</i>	224
A.13 <i>Use Case Skenario Membuat Pembelian Gabah</i>	226
A.14 <i>Use Case Skenario Membuat Pembelian Beras</i>	232
A.15 <i>Use Case Skenario Melihat Riwayat Pembelian Gabah</i>	238
A.16 <i>Use Case Skenario Melihat Riwayat Pembelian Beras</i>	240
A.17 <i>Use Case Skenario Keluar</i>	241
LAMPIRAN B. <i>ACTIVITY DIAGRAM</i>	244
B.1 <i>Activity Diagram Login (Masuk)</i>	244
B.2 <i>Activity Diagram Mengelola Data Pengguna</i>	246
B.3 <i>Activity Diagram Mengolah Kriteria Beras</i>	247
B.4 <i>Activity Diagram Mengelola Stok Pembelian Gabah</i>	248
B.5 <i>Activity Diagram Mengelola Stok Pembelian Beras</i>	249
B.6 <i>Activity Diagram Mengelola Penilaian Mutu dan Harga Beras</i>	250
B.7 <i>Activity Diagram Mengelola Penilaian Mutu dan Harga Gabah</i>	251

B.8	<i>Activity Diagram</i> Mengelola Pemasok Beras	252
B.9	<i>Activity Diagram</i> Mengelola Pemasok Gabah	253
B.10	<i>Activity Diagram</i> Membuat Penilaian Pemasok Beras	254
B.11	<i>Activity Diagram</i> Melihat Penilaian Gabah.....	255
B.12	<i>Activity Diagram</i> Melihat Penilaian Beras	256
B.13	<i>Activity Diagram</i> Mengelola Pembelian Gabah	257
B.14	<i>Activity Diagram</i> Mengelola Pembelian Beras.....	258
B.15	<i>Activity Diagram</i> Melihat Riwayat Pembelian Gabah	259
B.16	<i>Activity Diagram</i> Melihat Riwayat Pembelian Beras	259
B.17	<i>Activity Diagram Logout</i> (Keluar)	260
LAMPIRAN C. SEQUENCE DIAGRAM		261
C.1	<i>Sequence Diagram</i> Login (Masuk)	261
C.2	<i>Sequence Diagram</i> Mengelola Data Pengguna.....	262
C.3	<i>Sequence Diagram</i> Mengolah Kriteria Beras	263
C.4	<i>Sequence Diagram</i> Mengelola Stok Pembelian Beras	265
C.5	<i>Sequence Diagram</i> Mengelola Stok Pembelian Gabah	266
C.6	<i>Sequence Diagram</i> Mengelola Penilaian Mutu dan Harga Beras	267
C.7	<i>Sequence Diagram</i> Mengelola Penilaian Mutu dan Harga Gabah.....	268
C.8	<i>Sequence Diagram</i> Mengelola Pemasok Beras.....	269
C.9	<i>Sequence Diagram</i> Mengelola Pemasok Gabah	270
C.10	<i>Sequence Diagram</i> Membuat Penilaian Pemasok Beras.....	271
C.11	<i>Sequence Diagram</i> Melihat Penilaian Gabah.....	272
C.12	<i>Sequence Diagram</i> Melihat Penilaian Beras	272
C.13	<i>Sequence Diagram</i> Mengelola Pembelian Gabah	273
C.14	<i>Sequence Diagram</i> Mengelola Pembelian Beras.....	274
C.15	<i>Sequence Diagram</i> Melihat Riwayat Pembelian Gabah	275
C.16	<i>Sequence Diagram</i> Melihat Riwayat Pembelian Beras	275
C.17	<i>Sequence Diagram Logout</i> (Keluar)	276
LAMPIRAN D. TRANSKRIP WAWANCARA.....		277

BAB 1. PENDAHULUAN

Bab ini merupakan langkah awal dalam penulisan tugas akhir. Bab ini berisi latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, dan sistematika penulisan.

1.1 Latar Belakang

Indonesia merupakan negara agraris dengan mayoritas penduduk bermata pencaharian sebagai petani. Dikarenakan hal tersebut, banyak sekali bahan olahan pangan yang dihasilkan di Indonesia. Salah satu yang paling banyak dihasilkan adalah dari segi pertanian padi. Padi telah menjadi komoditi penting bagi masyarakat Indonesia, hal ini dikarenakan masyarakat Indonesia memiliki makanan pokok berupa nasi yang didapatkan melalui hasil olahan padi. Selain hal tersebut kandungan gizi dari beras juga menjadikan beras sebagai hasil olahan padi menjadi perhatian di Indonesia untuk memenuhi kebutuhan beras (Lutfi dkk., 2013).

Padi mengalami beberapa proses pendistribusian sehingga pada akhirnya sampai ke tangan konsumen. Pertama padi ditanam oleh petani, dan hasil panennya dijual pada tengkulak maupun pabrik beras dalam bentuk gabah, selanjutnya gabah diolah oleh pabrik penggilingan beras sehingga menjadi beras. Pabrik penggilingan beras menjual kembali beras menuju pabrik produsen beras untuk dikemas dan dijual kepada konsumen. Namun Selain membeli beras, pabrik produsen beras juga dapat secara langsung membeli gabah dan mengolahnya hingga menjadi beras yang siap dipasarkan.

Salah satu pabrik produsen beras yang mengolah gabah dan juga membeli beras untuk proses produksinya adalah UD Kamal Lestari. Pabrik ini merupakan pabrik penggilingan dan produsen beras kualitas kepala yang berada di Dusun Krajan Kabupaten Jember. Pabrik tersebut memproduksi dan mengolah beberapa merek beras, salah satunya yang menjadi merek unggulannya adalah beras Cap Kobra dan Dua

Kelinci. Beras produksi pabrik didapatkan melalui pembelian gabah dari petani maupun tengkulak. Untuk pembelian gabah, UD Kamal Lestari menargetkan pembelian untuk gabah dengan Jenis *IR 64* untuk menghasilkan beras kepala.

Pembelian beras dan gabah pada pabrik tersebut dilakukan secara manual. Gabah dan beras yang akan dibeli harus diseleksi berdasarkan kriteria yang telah ditetapkan sebelumnya. Hal ini menimbulkan permasalahan bagi pembuat keputusan untuk menentukan pembelian beras maupun gabah dengan mutu terbaik dan sesuai dengan kriteria yang ditetapkan oleh pabrik tersebut. Selama ini pihak pabrik melakukan proses pemilihan pembelian gabah dan beras secara manual, hal ini tentunya akan menimbulkan dampak kurang akuratnya pemilihan gabah dan beras yang akan dibeli, sehingga jika terdapat pembelian bahan yang tidak sesuai kriteria dan mutu dapat mengakibatkan kerugian materi dan menurunnya mutu produksi pada pabrik tersebut. Selain hal tersebut jika terdapat banyak sekali pilihan gabah dan beras yang akan di beli, hal ini dapat menimbulkan kesulitan dalam mengolah data yang banyak, hal tersebut juga berimbas pada waktu yang dibutuhkan untuk proses pembelian yang semakin lama. Selama ini usaha yang dilakukan oleh UD Kamal Lestari untuk menangani permasalahan tersebut adalah memilih secara cermat dan teliti dalam pembelian gabah dan beras secara manual sehingga didapatkan gabah maupun beras yang sesuai dengan kriteria dan mutu pabrik tersebut.

Permasalahan diatas memunculkan sebuah gagasan pemikiran untuk mengembangkan sebuah sistem yang dapat membantu pabrik beras dalam pengolahan data untuk membantu memberikan solusi terbaik dalam mendapatkan gabah dan beras yang sesuai kriteria. Oleh karena itu sistem penunjang keputusan adalah salah satu komponen yang cukup penting dalam membantu memutuskan suatu masalah. Dengan adanya sistem ini diharapkan pabrik produsen beras dapat dengan mudah menyeleksi gabah dan beras siapakah yang akan dibeli untuk proses produksi selanjutnya, sehingga nantinya kualitas mutu dapat terjaga dan memperoleh hasil produksi secara maksimal.

Sistem yang akan dibuat merupakan sistem yang akan memberikan penunjang keputusan terhadap pembelian gabah maupun padi pada UD Kamal Lestari dengan menggunakan metode SMART. Konsep dasar dari metode tersebut adalah mencari penjumlahan terbobot dan perankingan yang ditentukan melalui setiap alternatif pada semua kriteria dan sub kriteria yang telah ditentukan. Hasil akhir dari sistem ini adalah perankingan mutu gabah maupun beras dengan mutu terbaik, sehingga dapat memudahkan pembuat keputusan dalam mengambil keputusan terbaik.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang diatas, adapun rumusan masalah dalam penelitian ini adalah sebagai berikut :

1. Bagaimana mengimplementasikan metode *Simple Multy-Attribute Rating Technique (SMART)* dalam penentuan pembelian gabah dan beras terbaik?
2. Bagaimana membangun sistem penunjang keputusan pembelian gabah dan beras terbaik dengan metode *Simple Multy-Attribute Rating Technique (SMART)*?

1.3 Tujuan Penelitian

Tujuan dilakukannya penelitian ini adalah sebagai berikut :

1. Mengimplementasikan metode *Simple Multy-Attribute Rating Technique (SMART)* dalam penentuan pembelian gabah dan beras terbaik.
2. Membangun sistem penunjang keputusan pembelian gabah dan beras dengan metode *Simple Multy-Attribute Rating Technique (SMART)*.

1.4 Manfaat Penelitian

Manfaat yang dapat diambil dari penelitian yang akan dicapai adalah sebagai berikut :

a. Bagi akademis

Penelitian yang dilakukan diharapkan memberikan hasil yang mampu memberikan masukan informasi yang terkait dengan judul penelitian kepada pembaca pada umumnya dan pada Program Studi Sistem Informasi Universitas Jember pada khususnya.

b. Bagi peneliti

1. Mengetahui data yang digunakan dalam menentukan kelayakan pembelian gabah dan beras terbaik pada UD Kamal Lestari.
2. Mengetahui proses penerapan sistem penunjang keputusan dalam penentuan pembelian gabah dan beras terbaik.
3. Mengetahui cara kerja metode *Simple Multy-Attribute Rating Technique (SMART)* dalam menentukan sebuah sistem penunjang sebuah keputusan.

c. Bagi objek penelitian

Mempermudah proses penentuan pembelian gabah dan beras terbaik terhadap petani yang sebelumnya dilakukan secara manual dengan memberikan rangking kelayakan sebagai alat bantu pihak terkait.

1.5 Batasan Masalah

Beberapa hal yang membatasi penelitian ini adalah:

1. Sistem penunjang keputusan ini hanya sebagai alat bantu bagi pihak pabrik produsen beras dalam menentukan pembelian beras terbaik terhadap petani berdasarkan kriteria yang ditentukan oleh pihak pabrik dan keputusan akhir tetap berada di pihak pabrik.
2. Penentuan Kriteria penilaian beras mengacu pada *SNI 6128:2008* tentang beras, dan beberapa kriteria pendukung lainnya.

3. Penentuan Kriteria penilaian gabah mengacu pada *SNI 01-0224-1987* tentang gabah, dan beberapa kriteria pendukung lainnya.
4. Hasil dari sistem penunjang keputusan ini adalah urutan prioritas gabah dan beras yang layak dan tidak layak untuk dibeli oleh tengkulak ataupun pabrik mulai dari yang tertinggi sampai terendah.
5. Sistem berbasis *website*.

1.6 Sistematika Penulisan

Sistematika penulisan dalam penulisan tugas akhir adalah sebagai berikut :

1. Pendahuluan

Bab pertama ini menjelaskan tentang latar belakang, perumusan masalah, batasan masalah, tujuan, manfaat, dan sistematika penulisan yang terkait dengan penelitian yang dilakukan.

2. Tinjauan Pustaka

Bab ini menjelaskan tentang tinjauan terhadap penelitian terdahulu yang berkaitan dengan penelitian yang dilakukan dan kajian teori yang dapat mendukung penelitian yang dilakukan.

3. Metodologi Penelitian

Bab ini menjelaskan tentang tahapan – tahapan penelitian yang akan dilakukan , mulai dari perancangan sampai pembuatan sistem terkait penelitian.

4. Pengembangan Sistem

Bab ini menjelaskan tentang tahapan-tahapan yang dilakukan dalam merancang dan membangun sistem, mulai dari proses desain, pengkodean, dan pengujian sistem.

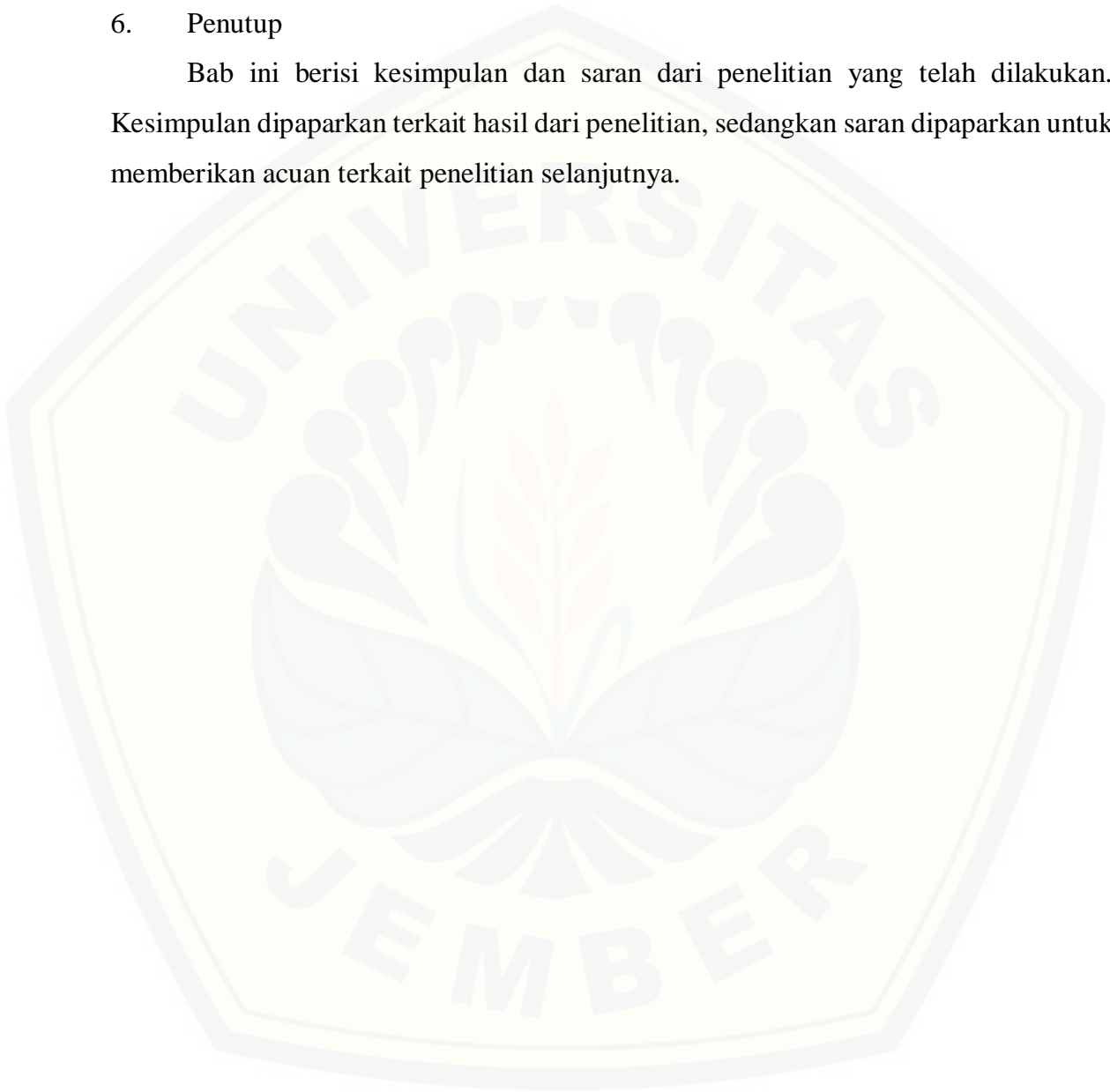
5. Hasil dan Pembahasan

Bab ini menjelaskan tentang hasil dan pembahasan penelitian yang telah dilakukan, dari bab ini peneliti akan memberikan penjelasan apakah metode dan landasan teori yang telah dilakukan untuk memperoleh hasil sesuai dengan apa yang

diinginkan peneliti, hasil dari penelitian dipaparkan melalui tabel dan juga gambar untuk mendukung hipotesa yang diperoleh.

6. Penutup

Bab ini berisi kesimpulan dan saran dari penelitian yang telah dilakukan. Kesimpulan dipaparkan terkait hasil dari penelitian, sedangkan saran dipaparkan untuk memberikan acuan terkait penelitian selanjutnya.



BAB 2. TINJAUAN PUSTAKA

Bab ini menjelaskan tentang penelitian terdahulu yang berkaitan dengan penelitian yang dilakukan dan kajian teori yang dapat mendukung penelitian yang dilakukan.

2.1 Penelitian Terdahulu

Dalam tinjauan pustaka peneliti melihat beberapa referensi, penelitian sebelumnya mengenai Sistem penunjang keputusan penentu kualitas mutu dan harga beli dengan menggunakan metode *Analytical Hierarchy Process* menjelaskan bahwa dengan menggunakan metode AHP didapatkan penentuan kualitas mutu serta harga beli komoditi sawah untuk beras dan jagung berdasarkan standarisasi pembobotan Kaidah Pembobotan Indikator (Jannah, 2013). penelitian tersebut berfokus pada penetapan mutu beras dan jagung yang sesuai standart SNI yang nantinya dapat dibeli oleh Koperasi Unit Desa (KUD). Penetapan mutu sesuai SNI didapat melalui perbandingan pembobotan sepasang kriteria dengan skala perbandingan AHP, selanjutnya hubungan antar kriteria yang diperbandingkan diberikan bobot antara 2 hingga 9, tahap selanjutnya adalah menjumlahkan skala perbandingan setiap kriteria perbaris yang telah didapatkan lalu membaginya dengan jumlah skala perbandingan itu sendiri dalam setiap kriteria, sampai dengan melakukan standarisasi bobot sesuai dengan kaidah KPI. Hasil akhir dari penelitian diatas adalah standarisasi bobot dan penilaian mutu untuk setiap kriteria beras dan juga jagung.

Pada penelitian selanjutnya yang mengimplementasikan Metode SMART dalam pemilihan pembelian mobil keluarga menjelaskan bahwa metode SMART cocok untuk menyelesaikan masalah yang sifatnya multikriteria dan multiobjek (Atiqah, 2013). Pada penilitian ini dipaparkan bahwa sistem yang dibuat mampu menampilkan perangkaan terkait mobil yang akan dibeli. Hasil yang diperoleh didapat dari implemetasi metode SMART ke dalam kriteria dan juga subkriteria yang dibutuhkan.

Metode SMART mendukung keputusan multikriteria sehingga cocok digunakan dalam penunjang keputusan yang mempertimbangkan kriteria dan juga subkriteria yang cukup banyak. Adapun kriteria yang digunakan dalam pemilihan mobil keluarga mulai dari harga, fasilitas, kebutuhan dan kualitasnya.

Berdasarkan pemaparan penelitian – penelitian terdahulu yang telah dijelaskan, terdapat beberapa kekurangan pada penelitian pertama, antara lain dalam menentukan penilaian mutu yang hanya difokuskan dari kriteria SNI (Standart Nasional Indonesia) saja tanpa memperhitungkan kriteria pendukung lainnya sebagai acuan dalam menentukan mutu beras dan jagung, selain hal tersebut alangkah baiknya jika diberikan perangkingan untuk pengelompokkan beras maupun jagung dengan mutu yang sama. Selain itu dalam pemaparan atikel kedua masih terdapat kekurangan dalam memaksimalkan hasil yang diperoleh. Kriteria yang digunakan dalam mengambil keputusan relatif kurang banyak dan bervariasi, sehingga perbandingan antar subkriteria sangat dekat dengan nilai subkriteria lainnya. Pemberian nilai bobot yang terlalu dekat tersebut menyebabkan kurang maksimalnya hasil akhir penilaian keputusan yang didapat. Selain hal tersebut sistem yang dibangun masih belum berbasis website, sehingga lingkup pengguna sistem kurang luas, dan kurang fleksibel.

Metode *Simple Multi Attribute Rating Technique* (SMART) sesuai digunakan pada penelitian yang akan dilakukan, karena mendukung pengambilan keputusan multikriteria. Kriteria pemilihan beras didasarkan pada mutu yang mengacu pada Standart Nasional Indonesia (SNI) tentang beras dan penambahan kriteria pendukung lain yang mampu memberikan penilaian lebih spesifik terhadap pemberi keputusan melalui perantara sistem penunjang keputusan tersebut. Dengan adanya sistem ini diupayakan dapat membantu mencari gabah dan beras terbaik dengan cara pemberian bobot disetiap kriteria yang diberikan, selanjutnya di hitung dengan menggunakan metode SMART, dan di rangking untuk mendapatkan gabah dan beras terbaik berdasarkan bobot terbesar.

2.2 Gabah

Gabah biasa disebut sebagai bulir padi. Sedangkan menurut Badan Standarisasi Nasional Gabah adalah butir padi (*Oryza sativa L*) yang telah terlepas dari malainya atau bisa disebut sebagai tangkai padi (BSN, 1987). Sebelum menjadi beras, padi terlebih dahulu diproses menjadi gabah. Gabah merupakan objek yang penting dalam pengolahan padi, karena sebagian besar komoditas perdagangan padi dimulai dari gabah.

Penentuan kualitas mutu gabah dijelaskan dalam Standar Nasional Indonesia (SNI) 01-0224-1987. Gabah digolongkan dalam 3 jenis mutu, yaitu mutu I, II dan III. Terdapat beberapa persyaratan mutu gabah, diantaranya persyaratan kualitatif dan kuantitatif.

Persyaratan kualitatif meliputi

1. Bebas hama dan penyakit
2. Bebas bau busuk, atau bau-bau lainnya
3. Bebas dari bahan kimia seperti sisa-sisa pupuk, insektisida, fungisida dan bahan kimia lainnya.
4. Gabah tidak boleh panas.

Sedangkan untuk persyaratan kualitatif penentuan mutu gabah dapat dijelaskan pada Tabel 2.1.

Tabel 2.1 Spesifikasi persyaratan mutu Gabah

(Sumber: BSN, 1987)

No Urut	Komponen Mutu	Kualitas		
		I	II	III
1	Kadar Air (% maksimum)	14,0	14,0	14,0
2	Gabah Hampa (% maksimum)	1,0	2,0	3,0
3	Butir rusak + Butir kuning (% maksimum)	2,0	5,0	7,0
4	Butir mengapur + Gabah muda (% maksimum)	1,0	5,0	10,0
5	Butir merah (% maksimum)	1,0	2,0	4,0
6	Benda asing (% maksimum)	-	0,5	1,0
7	Gabah varietas lain (% maksimum)	2,,0	5,0	10,0

2.3 Beras

Beras memiliki banyak pengertian, dari penelitian sebelumnya banyak para peneliti menafsirkan arti beras dengan berbagai macam arti. Beras adalah gabah yang bagian kulitnya sudah dibuang dengan cara digiling dan disosoh menggunakan alat pengupas dan penggiling (*huller*) serat alat penyosoh (*polisher*) (Koswara, 2009). Beras berasal dari kata weas dalam bahasa Jawa Kuno.

Beras dipilih menjadi bahan pangan pokok karena sumber daya alam dan iklim di Indonesia yang mendukung. Hal tersebut tentunya dapat mendukung dalam proses penyediaan beras yang cukup, karena beras diyakini mudah proses produksinya, cepat pengolahannya, serta aman dari segi kesehatan. Selain hal tersebut sebagaimana bulir serelia lain, bagian terbesar beras didominasi oleh pati sekitar 80 – 85%, protein, vitamin terutama pada bagian aleuron, mineral, air dan karbohidrat (Hidayati, 2014). Tidak heran bahwa hasil olahan beras banyak diminati oleh masyarakat Indonesia.

Kualitas beras berdasarkan mutu diatur di dalam Standar Nasional Indonesia (SNI) 6128:2008 tentang beras, dimana SNI ini merupakan revisi dari SNI 01-6128-1999, Beras giling berdasarkan usulan dari seluruh stakeholder beras dengan memperhatikan kondisi mutu beras Indonesia di pasaran dan standar mutu beras yang digunakan oleh negara-negara produsen beras lainnya.

Standar ini bertujuan untuk mengantisipasi terjadinya manipulasi mutu beras di pasaran, terutama pencampuran/pengoplosan antar kualitas atau antar varietas. Oleh karena itu dilakukan beberapa perubahan/penyempurnaan pada beberapa bagian yaitu pada bagian syarat mutu, cara uji dan penandaan. Standar ini dirumuskan oleh Panitia Teknis 65-03 Pertanian dan telah dibahas dalam rapat-rapat teknis dan terakhir disepakati dalam rapat konsensus di Jakarta pada tanggal 28 November 2006 yang dihadiri oleh anggota Panitia Teknis.

Standar ini telah melalui tahap jajak pendapat pada tanggal 23 Juli 2007 sampai dengan 23 September 2007 dan tahap pemungutan suara pada tanggal 7 April 2008

sampai dengan 7 Juni 2008, namun untuk mencapai kuorum diperpanjang sampai dengan tanggal 7 Juli 2008 dan langsung disetujui menjadi RASNI (Rancangan Akhir Standar Nasional Indonesia).

Beras dapat digolongkan dalam 5 (lima) kelas mutu yaitu I, II, III, IV, dan V. Syarat umum penggolongan beras adalah sebagai berikut :

1. Bebas hama dan penyakit.
2. Bebas bau apek, asam atau bau asing lainnya.
3. Bebas dari campuran dedak dan bekatul.
4. Bebas dari bahan kimia yang membahayakan dan merugikan konsumen.

Spesifikasi persyaratan mutu beras dapat dilihat pada tabel 2.2

Tabel 2.2 Spesifikasi persyaratan mutu beras

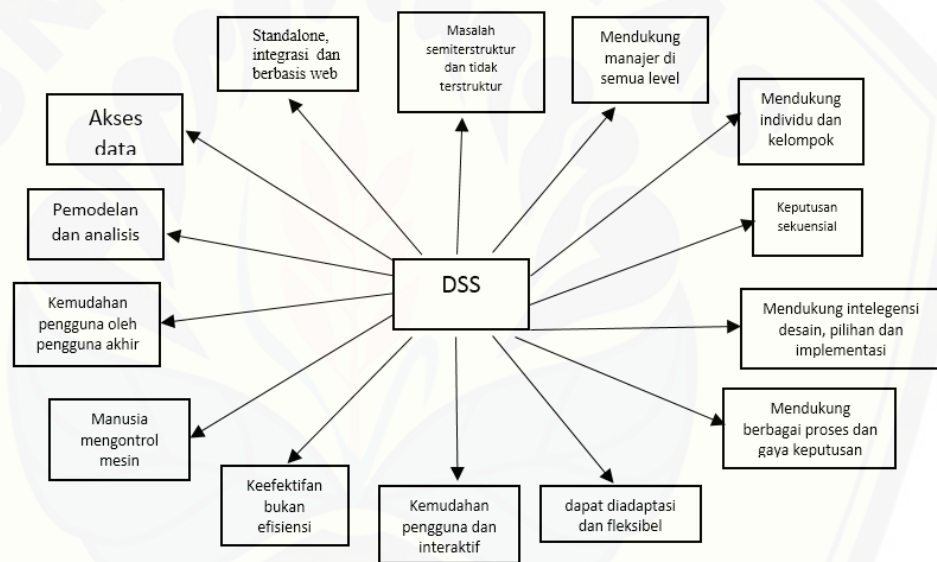
(Sumber: BSN, 2008)

No	Komponen Mutu	Satuan	Mutu				
			I	II	III	IV	V
1	Derajat sosoh (min)	(%)	100	100	95	95	85
2	Kadar air (maks)	(%)	14	14	14	14	15
3	Butir kepala (min)	(%)	95	89	78	73	60
4	Butir patah (maks)	(%)	5	10	20	25	35
5	Butir menir (maks)	(%)	0	1	2	2	5
6	Butir merah (maks)	(%)	0	1	2	3	3
7	Butir kuning/rusak (maks)	(%)	0	1	2	3	5
8	Butir mengapur (maks)	(%)	0	1	2	3	5
9	Benda asing (maks)	(%)	0	0,02	0,02	0,05	0,20
10	Butir gabah (maks)	Butir/100g)	0	1	1	2	3

2.4 Sistem Pendukung Keputusan

Sistem pendukung keputusan adalah bagian dari sistem informasi berbasis komputer (termasuk sistem berbasis pengetahuan) yang dipakai untuk mendukung pengambilan keputusan dalam suatu organisasi atau perusahaan. Dapat juga dikatakan sebagai sistem komputer yang mengolah data menjadi informasi untuk mengambil keputusan dari masalah semi-terstruktur yang spesifik (Ardianto dkk., 2014).

Sedangkan menurut Turban dkk (2005) Sistem Penunjang Keputusan (SPPK) atau disebut juga *Decision Support Sistem (DSS)* merupakan suatu pendekatan dalam pengambilan keputusan dengan menggunakan data, *interface* yang mudah serta menggabungkan pemikiran pengambil keputusan. DSS menggunakan berbagai *model* dan dibangun dengan oleh suatu proses interaktif dan iteratif sehingga mendukung pada fase pengambilan keputusan disertai dengan memasukkan suatu komponen pengetahuan. Dalam Gambar 2.1 dijelaskan bahwa DSS memiliki karakteristik dalam menentukan pengambilan keputusan.



Gambar 2.1 Karakteristik DSS

(Sumber: Turban dkk, 2005)

Menurut definisi tersebut peneliti dapat menyimpulkan bahwa sistem penunjang keputusan tidak ditekankan untuk membuat keputusan secara mutlak, melainkan melengkapi kemampuan untuk mengolah informasi yang diperlukan untuk membuat keputusan. Dengan kata lain, Sistem Penunjang Keputusan membantu manusia dalam proses membuat keputusan, bukan menggantikan perannya dalam mengambil keputusan.

2.5 Metode Simple Multi Attribut Rating Tachnique (SMART)

SMART (Simple Multi Attribut Rating Tachnique) merupakan metode penunjang pengambilan keputusan yang multiatribut yang dikembangkan oleh Edward pada tahun 1977. Teknik pembuatan keputusan multiatribut ini digunakan untuk mendukung pembuat keputusan dalam memilih antara beberapa alternatif. Setiap pembuat keputusan harus memilih sebuah alternatif yang sesuai dengan tujuan yang telah dirumuskan (Kasie, 2013).

Teknik pembuatan keputusan multiatribut ini digunakan untuk membantu pengambil keputusan dalam memilih antara beberapa alternatif. Setiap alternatif terdiri dari sekumpulan atribut dan setiap atribut mempunyai nilai-nilai, nilai ini dirata-rata dengan skala tertentu. Setiap atribut mempunyai bobot yang menggambarkan seberapa penting ia dibandingkan dengan atribut lain (Kustiyahningsih dkk., 2013).

Pembobotan atribut pada *SMART* dilakukan dengan dua langkah yaitu:

1. Mengurutkan kepentingan suatu atribut dari level terburuk ke level terbaik.
2. Membuat perbandingan rasio kepentingan setiap atribut dengan atribut lain dibawahnya.

Beberapa langkah-langkah yang harus dilakukan dalam menggunakan metode *SMART* antara lain :

1. Menentukan kriteria.
2. Menentukan nilai skala 0 - 100 berdasarkan prioritas untuk menilai bobot dan kemudian dilakukan normalisasi.

Normalisasi dilakukan menggunakan rumus dengan persamaan 2.1:

$$nw_j = \frac{w_j}{\sum w_j} \dots\dots\dots(2.1)$$

dimana :

w_j = bobot kriteria

$\sum w_j$ = total bobot semua kriteria

3. Menentukan nilai *utility* untuk setiap sub kriteria masing-masing menggunakan rumus seperti pada persamaan 2.2 atau langsung memberikan nilai *utility* berdasarkan prioritas.

$$u_i(a_i) = 100 \frac{(C_{max} - C_{out\ i})}{(C_{max} - C_{min})} \% \quad \dots\dots\dots (2.2)$$

4. Hitung nilai akhir masing-masing kriteria dan subkriteria dengan menggunakan rumus persamaan 2.3

$$SMART = \sum_{j=1}^k w_j u_i(a_i), \quad i = 1, 2, \dots, k \quad \dots\dots\dots (2.3)$$

Keterangan:

a_i = alternatif ke i

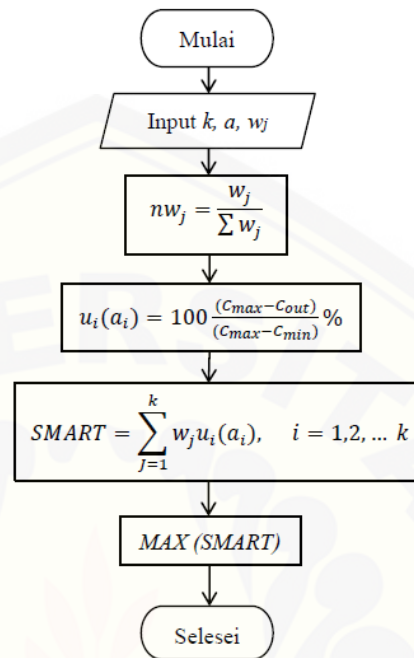
w_j = bobot kriteria ke j

u_i = utility/nilai sub kriteria dari kriteria ke i

$u(a_i)$ = nilai dari alternative ke i

SMART meranking dengan melihat range nilai akhir masing-masing data. Hasil dari perhitungan dengan menggunakan *SMART* adalah perangkaian bobot tertinggi sampai bobot terendah dimana bobot tertinggi merupakan hasil yang dibutuhkan oleh pengambil keputusan (Atiqah, 2013).

Berikut gambar diagram alir untuk memperjelas alur dari perhitungan metode *SMART* yang dapat dilihat pada gambar 2.2



Gambar 2.2 Diagram alir metode SMART

BAB 3. METODOLOGI PENELITIAN

Bab ini memaparkan tentang bagaimana penelitian dilakukan mulai dari tahap pengumpulan data dan juga informasi yang dapat diolah untuk memecahkan permasalahan pada penelitian ini.

3.1 Jenis Penelitian

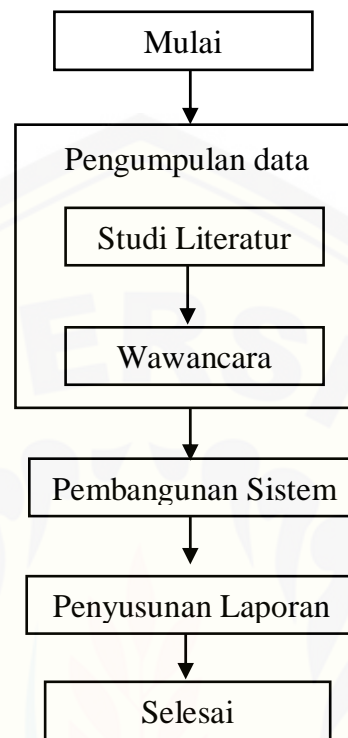
Jenis Penelitian yang digunakan dalam penelitian ini merupakan penelitian pengembangan. Disebut sebagai penelitian pengembangan karena dalam penelitian ini memiliki tujuan untuk mengembangkan proses yang sebelumnya dilakukan secara manual menjadi proses yang terkomputerisasi, seperti yang dijelaskan oleh Munawaroh (2016) Penelitian dan pengembangan bertujuan untuk menghasilkan sebuah produk baru atau menyempurnakan produk yang sudah ada yang dapat dipertanggung jawabkan. Dalam penelitian ini Peneliti berfokus terhadap pengembangan perangkat lunak dengan mengimplementasikan Simple Multi Attribute Technique Rating (SMART) sebagai metode penilaian untuk menghasilkan keluaran yang lebih akurat dan dapat digunakan sebagai penunjang keputusan pihak pabrik beras.

3.2 Tempat dan Waktu Penelitian

Tempat dilakukannya penelitian adalah UD Kamal Lestari yang terletak di Desa Krajan Kalisat. Waktu penelitian dilakukan selama 3 bulan yaitu dimulai dari bulan Oktober 2016 sampai dengan bulan Desember 2016.

3.3 Alur Penelitian

Alur penelitian merupakan urutan langkah penelitian yang dilakukan mulai dari studi literatur, pengumpulan data, pengolahan data, perancangan sistem, dan pengimplementasian rancangan sistem. diagram alur pada penelitian ini dapat dilihat pada gambar 3.1 berikut.



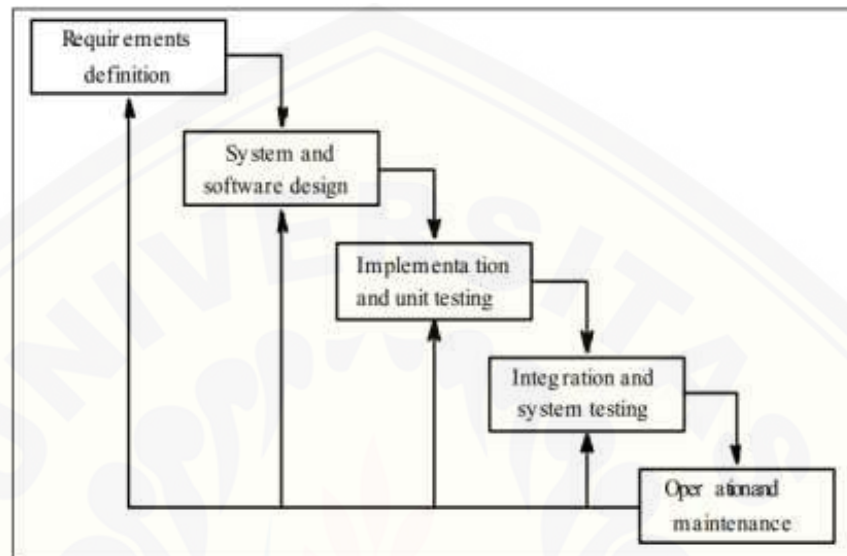
Gambar 3.1 Alur Penelitian

(Sumber: Hasil Analisis, 2016)

3.4 Metode Pengembangan Sistem

Untuk membangun sebuah sistem dibutuhkan sebuah model dalam pengembangannya. Dalam penelitian ini model yang digunakan adalah model *waterfall*. Model ini dipakai karena umum digunakan dalam pengembangan *software* berskala kecil. Model *waterfall* menitik beratkan pengguna sebagai prioritas utama penetapan kebutuhan *software*. Tahap pengembangan *software* dalam model *waterfall* dikerjakan bertahap mulai dari tahap terendah yaitu *Requirement Analysis*, *System Design*, *Implementation & Unit Testing*, *Integration & Testing*, *Operations & Maintenance*. Setiap proses pengerjaan dalam model *waterfall* tidak dapat dikerjakan sebelum tahap sebelumnya selesai, sebagai contoh tahap *System Design* tidak dapat

dikerjakan sebelum *Requirement Analysis* selesai. Berikut penjelasan dari model *waterfall* dijelaskan pada Gambar 3.2 berikut.



Gambar 3.2 Model Waterfall

(Sumber: Sommerville, 2011)

3.4.1 Requirement Analysis (Analisis Kebutuhan)

Tahap *requirement analysis* adalah tahap menganalisa apa yang dibutuhkan oleh sistem. Data kebutuhan dibagi menjadi kebutuhan fungsional dan non-fungsional. Pada sistem ini dibutuhkan juga data-data kriteria yang akan digunakan sebagai penunjang keputusan. Data yang sudah terkumpul akan menentukan bagaimana fitur yang akan dibangun pada sistem.

Proses pencarian data dilakukan melalui 2 tahap, yaitu tahap wawancara dan studi literatur

1. Wawancara

Wawancara adalah percakapan dengan maksud tertentu. Hampir pada semua penelitian, metode wawancara adalah yang paling banyak digunakan di dalam pengumpulan data. Dengan melaksanakan wawancara yang baik maka kesalahan di dalam pengumpulan data dapat ditekan serendah mungkin (Soegijono, 1993).

Wawancara dilakukan secara langsung kepada kepala pabrik produsen beras UD Kamal Lestari untuk mendapatkan data tentang kriteria pembelian gabah dan beras dengan mengacu pada SNI 6128:2008, SNI 01-0224-1987 dan faktor pendukung lainnya, selain hal tersebut juga untuk memperoleh informasi mengenai prioritas setiap kriteria yang dibutuhkan untuk menentukan gabah dan beras dengan mutu terbaik.

2. Studi Literatur

Studi Literatur dilakukan untuk mencari informasi yang berkaitan dengan sistem yang akan dibangun dengan mengacu pada teori – teori terkait penelitian yang bersumber dari jurnal, internet dan dokumen maupun informasi perusahaan. Data hasil studi literatur dapat menunjang dalam menguraikan informasi mengenai gabah dan beras yang menjadi objek penelitian. Pencarian Teori-teori bertujuan untuk Mencari data – data yang berhubungan dengan sistem penunjang keputusan pembelian gabah dan beras terbaik dengan menggunakan metode *Simple Multi Attribute Technique Rating* (SMART).

Data yang telah terkumpul kemudian dianalisis dan digunakan sebagai parameter kriteria dan subkriteria untuk mendapatkan hasil penilaian pembelian gabah dan beras terbaik. Kriteria yang dibutuhkan selanjutnya diproses dengan menggunakan metode *Simple Multi Attribute Technique Rating* (SMART) untuk mendapatkan peringkat teratas sampai terbawah gabah dan beras dengan mutu terbaik sehingga didapatkan gabah maupun beras yang layak untuk dibeli dan sesuai dengan kriteria yang dibutuhkan oleh pabrik.

Kriteria yang digunakan dalam penilaian memiliki sub kriteria yang penting untuk penilaian keakuratan dalam metode SMART. Kriteria dalam penelitian berdasarkan hasil wawancara dari narasumber terkait pembelian beras dan gabah. Penjelasan terkait kriteria dan subkriteria gabah dapat dilihat pada tabel 3.1 sedangkan untuk beras dapat dilihat pada tabel 3.2 berikut.

Tabel 3.1 Tabel Kriteria dan Subkriteria Gabah

No	Kriteria	Subkriteria
1	Gabah varietas lain	0% - maks 2%
		>2% - maks 5%
		>5% - maks 10%
2	Kadar Air	< 14%
		14%
		> 14%
3	Gabah Hampa	0% - maks 1%
		>1 - maks 2%
		>2 - maks 3%
4	Kondisi Gabah	Kering Sawah (gabah kering panen)
		Kering Lahan (gabah kering giling)
5	Hama	Ada
		Tidak ada
6	Bau	Bau Umum Gabah
		Bau Asam
		Bau Busuk
7	Butir mengapur dan Gabah muda	0% - maks 1%
		>1% - maks 5%
		>5% - maks 10%
8	Benda asing	0
		>0 - maks 0,5%
		>0,5 - maks 1%
9	Butir merah	0% - maks 1%
		>1% - maks 2%
		>2% - maks 4%
10	Butir rusak dan butir kuning	0% - maks 2%
		>2% - maks 5%
		>5% - maks 7%

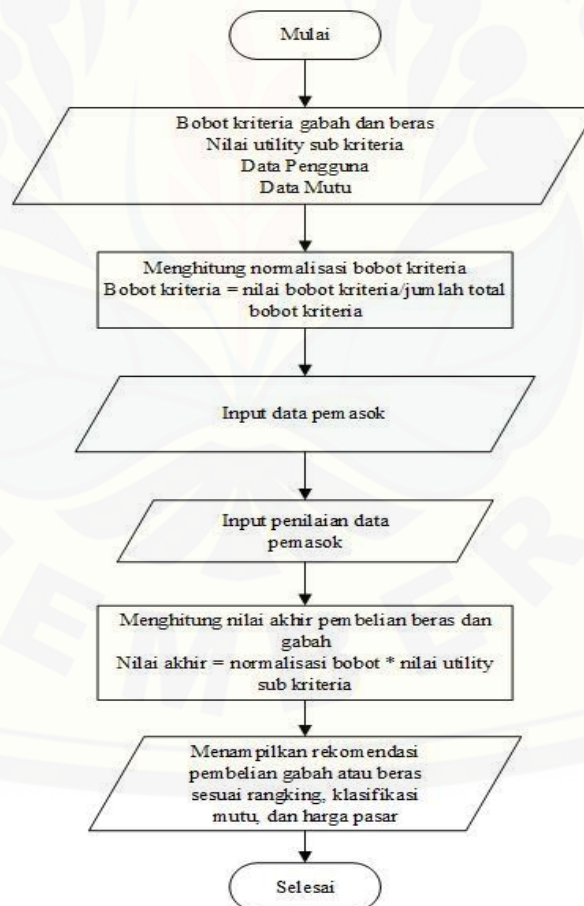
Tabel 3.2 Tabel Kriteria dan Subkriteria Beras

No	Kriteria	Subkriteria
1	Butir Patah	0% - maks 5%
		>5% - maks 10%
		>10% - maks 20%
		>20% - maks 25%
		>25% - maks 35%
Dilanjutkan		

Lanjutan		
No	Kriteria	Subkriteria
2	Kadar Air	< 14%
		14%
		> 14%
3	Butir Kepala	95% - 100%
		89% - 94%
		78% - 88%
		73% - 77%
		60% - 72%
4	Hama	Ada
		Tidak ada
5	Bau Asing	Bau Beras
		Bau Apek
6	Butir Mengapur	0%
		>0% - maks 1%
		>1% - maks 2%
		>2% - maks 3%
		>3 - maks 4%
7	Benda Asing	0%
		>0% - maks 0,02%
		>0,02% - maks 0,05%
		> 0,05%
8	Butir Merah	0%
		>0% - maks 1%
		>1% - maks 2%
		>2% - maks 3%
9	Butir Kuning	0%
		>0% - maks 1%
		>1% - maks 2%
		>2% - maks 3%
		> 3%
10	Butir Menir	0%
		>0% - maks 1%
		>1% - maks 2%
		> 2%
Dilanjutkan		

Lanjutan		
No	Kriteria	Subkriteria
11	Butir Gabah	0 butir
		1 butir
		2 butir
		>= 3 butir
12	Derajat Sosoh	96% - 100%
		86% - 95%
		< 86%

Penerapan metode SMART pada Sistem Penunjang Keputusan Pembelian Padi Terbaik dapat dilihat melalui diagram alir pada gambar 3.3 berikut.



Gambar 3.3 Diagram alir penerapan metode SMART pada SPK Pembelian Padi Terbaik

3.4.2 System Design (Desain Sistem)

Tahap *system design* adalah tahap yang dilakukan setelah data analisis telah terkumpul. Pada tahap desain ini menggunakan desain pemodelan *Unified Modeling Language (UML)*. Desain struktur ini digunakan karena mendukung konsep *Object Oriented Design (OOP)* sesuai dengan kode program yang akan digunakan *programmer*. Beberapa diagram dalam desain struktur *UML* adalah sebagai berikut :

1. *Business Process*

Business process adalah diagram yang memperlihatkan data apa saja yang menjadi inputan yang digunakan sistem, output data yang dihasilkan sistem, *platform* yang digunakan sistem, dan tujuan pembuatan sistem.

2. *Use Case Diagram*

Use case diagram adalah diagram yang menggambarkan fitur yang dapat dilakukan oleh sistem berdasarkan hak akses aktor.

3. *Sequance Diagram*

Sequance diagram adalah diagram yang menggambarkan interaksi antara method satu dengan yang lain di dalam kelas yang berbeda – beda. Sehingga terlihat alur data yang diproses.

4. *Activity Diagram*

Activity diagram adalah diagram yang menggambarkan alir sistem secara detail dari awal diakses, sampai selesai. Dimana alir sistem digambarkan pada dua dimensi yag berbeda, yaitu dari dimensi sistem dan juga pengguna.

5. *Class Diagram*

Class diagram adalah diagram yang menggambarkan kelas-kelas dalam sebuah sistem dan hubungannya antara satu kelas dengan yang lain. Dalam kelas ini juga ditampilkan atribut dan method yang ada pada kelas.

6. *Entity Relation Diagram (ERD)*

Entity relation diagram (ERD) adalah diagram yang menggambarkan atribut dan relasi tabel dalam sebuah basis data.

3.4.3 Implementation System (Implementasi Sistem)

Desain yang sudah dibuat pada tahap sebelumnya digunakan sebagai acuan dalam pembangunan sistem. Sistem dibangun menggunakan bahasa pemrograman *Page Hypertext Pre-Processor (PHP)* dengan *tool* yang digunakan *Sublime text 3*, dan database yang digunakan adalah *MySQL*. Proses pengkodean menggunakan *framework Codeigniter* karena mendukung penggunaan konsep *Object Oriented Programming (OOP)*.

3.4.4 Testing System (Pengujian Sistem)

Tahap pengujian harus dilakukan sebelum sistem diserahkan kepada *user*. Pengujian perangkat lunak adalah serangkaian kegiatan yang dilakukan dengan maksud menemukan kesalahan pada sistem yang dikembangkan (Khan, 2011). Tahap dilakukan agar programmer dapat mengetahui apakah sistem yang dibangun sesuai dengan kebutuhan yang telah dianalisis diawal. Serta agar mengetahui apakah terdapat kesalahan pada sistem yang dibangun. Tahap testing dilakukan guna menyempurnakan sistem sebelum diserahkan kepada *user*. Pada tahap testing ini dilakukan pengujian dengan metode *white-box* dan metode *black-box*.

White-box adalah pengujian dengan menggunakan struktur *internal* seperti pengujian aliran kontrol atau aliran data dari program (Shao dan Perry, 2006). Pada penelitian ini metode yang digunakan dalam pengujian *white box* adalah *independent path* dengan menggunakan *cyclomatic complexity*.

Cyclomatic complexity yaitu metrik perangkat lunak yang menyajikan ukuran kuantitatif dari kekompleksan logikal suatu program. *Cyclomatic complexity* digunakan untuk mencari jumlah *path* dalam suatu *flowgraph*. Rumus yang digunakan dalam menghitung *cyclomatic complexity* dapat dilihat pada persamaan 2.4 sebagai berikut.

$$V(G) = P + 1 \dots\dots\dots (2.4)$$

Keterangan :

P : Jumlah node yang memiliki kondisi

Black box testing adalah menguji perangkat lunak dengan melihat antarmuka luar sistem tanpa menguji desain dan kode program (Shao dan Perry, 2006). Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian *black box* seperti pada tabel 3.3 berikut:

Tabel 3.3 Tabel Uji *Black Box*

Kelas Uji	Skenario Uji	Hal yang Diharapkan	Kesimpulan

3.4.5 Maintenance System (Perawatan sistem)

Tahap *operations & maintenance* adalah tahap akhir dari model waterfall. Tahap ini dilakukan setelah sistem melalui tahap pengujian baik melalui pengujian *white-box* maupun pengujian *black-box*. Tahap *operations & maintenance* dilakukan guna pengembangan sistem atau mengatasi masalah (*bugs*) yang muncul setelah sistem diserahkan kepada *user*.

3.5 Gambaran Umum Sistem

Sistem penunjang keputusan pembelian gabah dan beras terbaik menggunakan metode Simple Multi Attribute Rating Technique (SMART) merupakan sebuah sistem penunjang keputusan berbasis web yang berfungsi untuk mencari gabah dan beras terbaik yang akan dibeli dari petani maupun mitra dengan memperhatikan kriteria pendukung dan SNI sebagai Standart Nasional Indonesia yang mengatur tentang segala macam peraturan mengenai beras dan gabah. Sistem ini nantinya dapat diakses oleh 2 user, dimana admin sebagai pakar yang dapat mengelola bobot, harga pasar, kriteria dan subkriteria dari sistem yang akan dijadikan parameter untuk penilaian pembelian gabah dan beras. User yang kedua adalah sebagai pihak pekerja pabrik yang

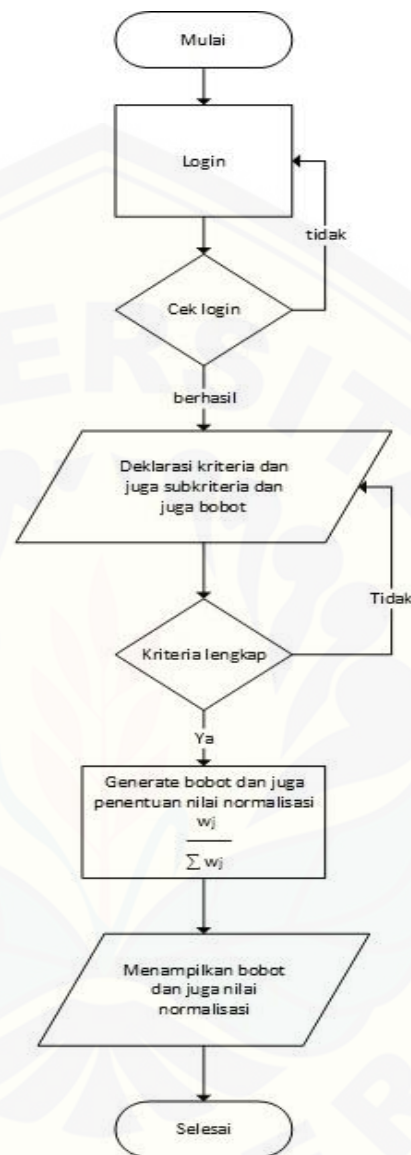
membutuhkan sebuah sistem untuk menentukan gabah dan beras terbaik yang akan mereka beli berdasarkan penjualan dari petani maupun mitra yang telah bekerja sama.

Kriteria yang digunakan dalam sistem ini berdasarkan yang terdapat pada standart mutu beras dan gabah dengan penambahan hasil analisis pribadi melalui wawancara terkait kriteria gabah dan beras dengan mutu baik. Dari data yang telah diinputkan nantinya akan diberi pembobotan tersendiri dan di proses menggunakan metode Simple Multi Attribute Rating Technique (SMART) sehingga menghasilkan output berupa rangking mutu gabah maupun beras dengan peringkat teratas, dan harga pasar saat ini, dimana rangking gabah maupun beras terbesar dapat dijadikan penunjang keputusan untuk pembelian gabah dan beras oleh pabrik beras terkait. Selain hal tersebut status gabah dan beras yang terbeli akan tersimpan dalam sistem, sehingga memudahkan pengguna dalam melihat catatan pembelian beras yang pernah dilakukan sebelumnya. Berikut adalah penjelasan singkat tentang pengguna sistem.

Penerapan metode SMART pada sistem yang akan dikembangkan dapat dilihat pada diagram alir pada gambar 3.4 untuk hak akses admin dan gambar 3.5 untuk hak akses pekerja atau pegawai.

1. Admin atau Pakar

Admin atau pakar memiliki hak akses tersendiri dalam mengelola setiap fitur. Gambar 3.4 merupakan alur kerja sistem yang dapat dilakukan oleh admin. Pertama adalah admin dapat melakukan login. Selanjutnya admin dapat mengelola setiap kriteria dan subkriteria. Nilai bobot dan normalisasi selanjutnya dihitung oleh sistem dan menampilkannya pada tabel yang ada pada tampilan sistem. Penentuan range mutu beras yang akan dijadikan acuan dalam menentukan mutu beras juga dikelola oleh admin

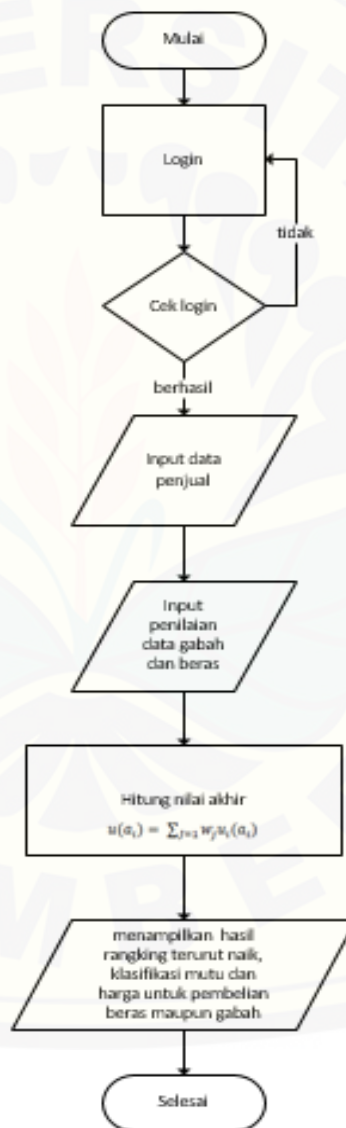


Gambar 3.4 Flowchart admin / pakar

2. Pegawai/Pekerja

Pegawai atau pekerja bertugas membuat penilaian dan mengelola pembelian padi. Gambar 3.5 merupakan alur yang dapat dilakukan oleh pegawai atau pekerja dalam mengelola pembelian padi. Alur pertama adalah pegawai melakukan *login* sistem, selanjutnya memasukkan data pemasok maupun mitra yang menjual hasil penennya. Alur selanjutnya pegawai berhak melakukan penilaian untuk setiap pemasok

gabah maupun beras sehingga didapatkan hasil olahan data yang dapat dijadikan sebagai penunjang keputusan untuk pembelian padi. Klasifikasi mutu dan harga selanjutnya ditampilkan pada halaman sistem sebagai acuan penunjang keputusan pembelian padi.



Gambar 3.5 Flowchart User / Pengguna

BAB 4. PENGEMBANGAN SISTEM

Bab ini akan menguraikan tentang tahapan – tahapan proses pengembangan sistem dimulai dari proses analisa kebutuhan fungsional dan nonfungsional, pembuatan usecase diagram, skenario, activity diagram, sequence diagram, class diagram dan entity relation diagram (ERD yang mengacu pada pengimplementasian metode SMART ke dalam sistem pembelian padi terbaik.

4.1 Analisis Kebutuhan Perangkat Lunak

Analisis kebutuhan perangkat lunak dilakukan berdasarkan *Software Developmet Life Cycle Waterfall* yang digunakan, dimana hal pertama adalah proses pencarian data dan informasi dalam pengembangan sistem yang didapatkan dari objek penelitian yang dituju. Melalui wawancara yang telah dilakukan pada objek penelitian, didapatkan hasil berupa data dan informasi yang selanjutnya diolah untuk menganalisis kebutuhan fungsional dan non fungsional sistem sebagai acuan dalam proses pengembangan sistem. dokumen wawancara terkait pencarian kebutuhan awal hingga terbentuknya kebutuhan fungsional dan non fungsional sistem, dapat dilihat pada lampiran D.

4.1.1 Kebutuhan Fungsional

Kebutuhan fungsional sistem merupakan kebutuhan pokok yang harus dibangun agar sistem memiliki fitur yang dapat bekerja sesuai dengan apa yang diharapkan oleh pengguna. Kebutuhan fungsional sistem adalah sebagai berikut :

1. Sistem memiliki proses *login* (masuk) kedalam halaman utama sesuai dengan hak akses pengguna
2. Sistem mampu mengelola data pengguna meliputi hak akses masuk admin dan pegawai

3. Sistem mampu mengelola kriteria, subkriteria, dan juga pembobotan yang digunakan sebagai bahan penilaian pembelian gabah.
4. Sistem mampu mengelola kriteria, subkriteria, dan juga pembobotan yang digunakan sebagai bahan penilaian pembelian beras.
5. Sistem mampu mengelola stok pembelian gabah terkait berapa banyak gabah yang nanti harus dibeli untuk dijadikan stok persediaan.
6. Sistem mampu mengelola stok pembelian beras terkait berapa banyak beras yang nanti harus dibeli untuk dijadikan stok persediaan.
7. Sistem mampu mengelola penilaian mutu dan harga gabah untuk dijadikan acuan penentuan harga dan mutu gabah yang akan dibeli.
8. Sistem mampu mengelola penilaian mutu dan harga beras untuk dijadikan acuan penentuan harga dan mutu beras yang akan dibeli.
9. Sistem mampu mengelola pemasok gabah.
10. Sistem mampu mengelola pemasok beras.
11. Sistem mampu membuat penilaian untuk menentukan mutu dan harga pada setiap pemasok gabah.
12. Sistem mampu membuat penilaian untuk menentukan mutu dan harga pada setiap pemasok beras.
13. Sistem mampu menampilkan hasil penilaian, detail penilaian, mutu, dan harga setiap pemasok gabah.
14. Sistem mampu menampilkan hasil penilaian, detail penilaian, mutu, dan harga setiap pemasok beras.
15. Sistem mampu mengelola pembelian gabah untuk setiap pemasok gabah.
16. Sistem mampu mengelola pembelian beras untuk setiap pemasok beras.
17. Sistem mampu menampilkan riwayat pembelian gabah sesuai dengan hak akses pengguna.
18. Sistem mampu menampilkan riwayat pembelian beras sesuai dengan hak akses pengguna.

19. Sistem mampu logout (keluar) dari sistem sesuai dengan hak akses pengguna menuju halaman login awal.

4.1.2 Kebutuhan Non Fungsional

Kebutuhan non fungsional merupakan kebutuhan yang menjadi pendukung agar sistem yang dibangun dapat dengan mudah, aman dan nyaman digunakan oleh pengguna. Kebutuhan non fungsional demi mendukung fungsionalitas sistem adalah sebagai berikut :

1. Sistem memiliki tampilan yang memudahkan terhadap pengguna (*user friendly*)
2. Sistem memiliki batasan hak akses untuk setiap pengguna yang masuk.
3. Sistem dapat dijalankan dengan *browser* yang berbeda.
4. Sistem mampu digunakan kapan saja dan dimana saja dengan syarat pengguna menggunakan akses internet, terkecuali sistem digunakan hanya pada server lokal.

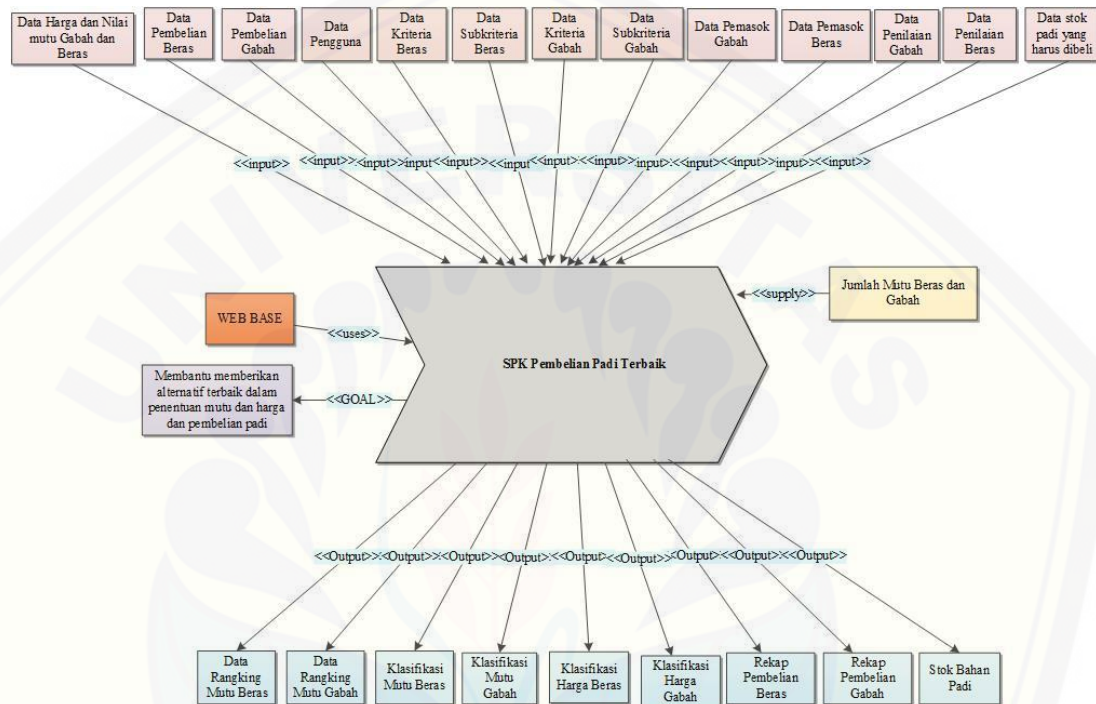
4.2 Desain Sistem

Tahap selanjutnya setelah analisis kebutuhan adalah desain sistem. Desain sistem dibuat untuk memberikan gambaran fungsionalitas sistem dan rancangan untuk proses selanjutnya yaitu pengkodean. Desain sistem yang akan dibuat dalam penelitian ini meliputi *Business Process*, *Use Case Diagram*, *Scenario*, *Activity Diagram*, *Sequence Diagram*, *Class diagram*, dan *Entity Relation Diagram*.

4.2.1 *Business Process*

Business Process merupakan diagram yang menggambarkan proses terjadinya olah data yang terjadi pada sistem meliputi *input*, *output*, *uses*, *supply*, dan *goal*. *Input* menjelaskan tentang apa saja data yang nantinya masuk ke dalam sistem, sehingga selanjutnya diolah dengan menggunakan *uses* dan menghasilkan data *output* yang sesuai dengan *goal* atau tujuan dari sistem itu sendiri, sedangkan *supply* merupakan

data masukan yang bersifat tetap dan tidak berubah. Data *input* didapatkan dari pengguna sedangkan untuk *output* adalah data dari hasil olahan sistem. *Business process* dari sistem yang akan dikembangkan dapat dilihat pada gambar 4.1 berikut.



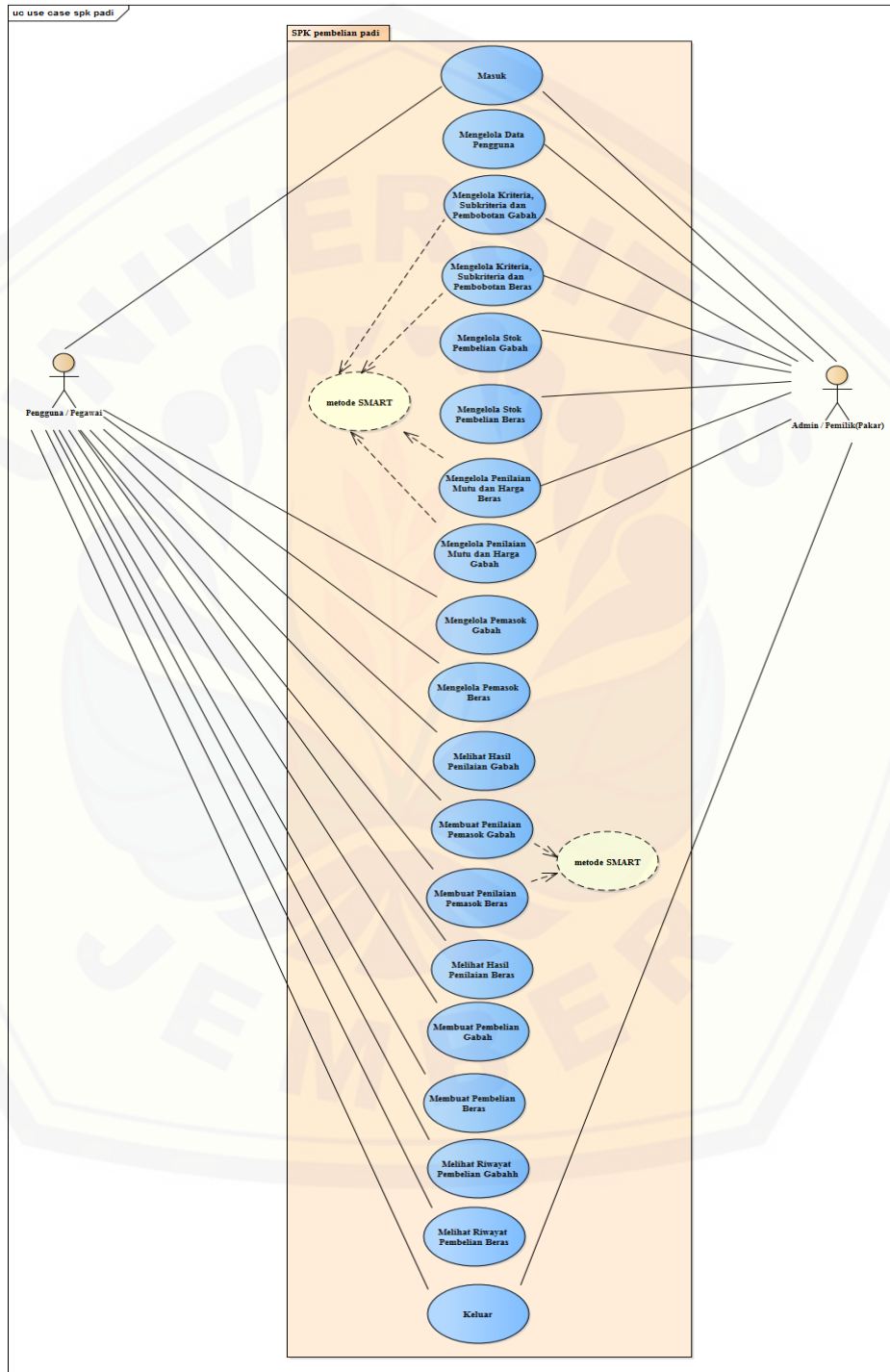
Gambar 4.1 Business Process

Gambar 4.1 merupakan bisnis proses sistem penunjang keputusan pembelian padi terbaik. Bisnis proses tersebut menjelaskan data apa saja yang menjadi *input*, *output*, *uses*, *supply*, dan *goal* dari sistem yang akan dikembangkan.

4.2.2 Use Case Diagram

Use Case Diagram merupakan kumpulan dari serangkaian kegiatan yang dapat dilakukan oleh sebuah sistem (Ibrahim dkk., 2011). *Use Case diagram* berisi tentang fitur yang akan dikembangkan dalam sistem penunjang keputusan pembelian padi terbaik, selain hal diagram ini juga berisi hak akses yang diberikan untuk setiap

pengguna dalam mengakses setiap fitur yang ada. Gambar use case diagram sistem penunjang keputusan pembelian padi terbaik dapat dilihat pada gambar 4.2 berikut



Gambar 4.2 Use Case Diagram

Gambar 4.2 menjelaskan use case diagram dari sistem yang akan dibuat, dimana terdapat dua aktor yang memiliki hak akses tersendiri dalam mengakses fitur yang akan dikembangkan. Fitur yang akan dikembangkan berjumlah sembilan belas fitur dimana terdapat beberapa fitur yang menggunakan metode SMART sebagai *include* agar fitur dapat berjalan sesuai dengan apa yang diharapkan.

Penjelasan definisi tugas aktor terkait dengan *use case* dapat dilihat pada tabel 4.1, dimana pada tabel tersebut dijelaskan tugas dari aktor atau pengguna terkait dengan fitur yang bisa dilakukan dalam sistem. Penjelasan lainnya dapat dilihat pada tabel 4.2, dimana pada tabel tersebut menjelaskan fungsionalitas dari setiap *use case* yang telah dibuat.

Tabel 4.1 Definisi Aktor

No.	Aktor	Definisi Tugas
1.	Admin / Pakar	Mengelola hak akses pengguna, mengelola kriteria dan subkriteria padi (gabah dan beras), mengelola seberapa banyak pembelian padi, melihat riwayat pembelian padi oleh aktor, dan mengelola mutu dan harga padi.
2.	Pegawai / Pekerja	Mengelola pemasok padi, mengelola penilaian padi, melakukan pembelian padi sesuai dengan yang dibutuhkan, melihat riwayat pembelian setiap hak akses pegawai.

Tabel 4.2 Definisi *Use Case*

No.	Usecase	Deskripsi
1.	Masuk	Fitur yang berfungsi untuk memproses pengguna yang ingin masuk ke dalam sistem sesuai dengan hak akses pengguna.
2.	Mengelola Data Pengguna	Fitur yang berfungsi untuk mengelola data pengguna sebagai acuan untuk melakukan <i>login</i> atau masuk ke dalam sistem.
3.	Mengelola Kriteria, Subkriteria dan Pembobotan Gabah	Fitur yang berfungsi untuk mengelola kriteria, subkriteria dan juga pembobotan, meliputi perhitungan hasil normalisasi kriteria dan pemberian nilai utility untuk setiap subkriteria gabah.
4.	Mengelola Kriteria, Subkriteria dan Pembobotan Beras	Fitur yang berfungsi untuk mengelola kriteria, subkriteria dan juga pembobotan, meliputi perhitungan hasil normalisasi kriteria dan pemberian nilai utility untuk setiap subkriteria beras.
Dilanjutkan		

Lanjutan		
No.	Usecase	Deskripsi
5.	Mengelola Stok Pembelian Gabah	Fitur yang berfungsi untuk mengelola seberapa banyak pembelian gabah yang diperlukan untuk stok produksi, dan melihat riwayat pembelian gabah oleh pegawai.
6.	Mengelola Stok Pembelian Beras	Fitur yang berfungsi untuk mengelola seberapa banyak pembelian beras yang diperlukan untuk stok produksi, dan melihat riwayat pembelian beras oleh pegawai.
7.	Mengelola Penilaian Mutu dan Harga Beras	Fitur yang berfungsi untuk mengelola penilaian mutu dan harga beras, meliputi memberikan range nilai mutu beras dengan menggunakan metode SMART dan juga menetapkan standart harga beli beras per kilogram.
8.	Mengelola Penilaian Mutu dan Harga Gabah	Fitur yang berfungsi untuk mengelola penilaian mutu dan harga gabah, meliputi memberikan range nilai mutu gabah dengan menggunakan metode SMART dan juga menetapkan standart harga beli gabah per kilogram.
9.	Mengelola Pemasok Gabah	Fitur yang berfungsi mengelola data pemasok yang menjual gabahnya.
10.	Mengelola Pemasok Beras	Fitur yang berfungsi mengelola data pemasok yang menjual berasnya.
11.	Membuat Penilaian Pemasok Gabah	Fitur yang berfungsi untuk membuat penilaian gabah dengan menggunakan metode SMART sesuai dengan kriteria dan subkriteria yang telah ditentukan untuk setiap pemasok gabah, penilaian digunakan sebagai acuan untuk melakukan klasifikasi mutu, harga, dan peringkatan penilaian gabah terbaik.
12.	Membuat Penilaian Pemasok Beras	Fitur yang berfungsi untuk membuat penilaian beras dengan menggunakan metode SMART sesuai dengan kriteria dan subkriteria yang telah ditentukan untuk setiap pemasok beras, penilaian digunakan sebagai acuan untuk melakukan klasifikasi mutu, harga, dan peringkatan penilaian beras terbaik.
13.	Melihat Hasil Penilaian Gabah	Fitur yang berfungsi untuk melihat penilaian dan klasifikasi mutu dan harga gabah sesuai dengan urutan ranking terurut naik.
14.	Melihat Hasil Penilaian Beras	Fitur yang berfungsi untuk melihat penilaian dan klasifikasi mutu dan harga beras sesuai dengan urutan ranking terurut naik.
15.	Membuat Pembelian Gabah	Fitur yang berfungsi untuk membuat pembelian gabah sesuai dengan peringkat yang telah ditetapkan sistem, pembelian gabah dari pemasok dapat dilakukan sebagian, dan juga semua.
16.	Membuat Pembelian Beras	Fitur yang berfungsi untuk membuat pembelian beras sesuai dengan peringkat yang telah ditetapkan sistem, pembelian beras dari pemasok dapat dilakukan sebagian, dan juga semua.
17.	Melihat Riwayat Pembelian Gabah	Fitur yang berfungsi untuk melihat riwayat pembelian gabah sesuai dengan hak akses pegawai
Dilanjutkan		

Lanjutan		
No.	Usecase	Deskripsi
18.	Melihat Riwayat Pembelian Beras	Fitur yang berfungsi untuk melihat riwayat pembelian beras sesuai dengan hak akses pegawai
19.	Keluar	Fitur yang berfungsi untuk menghapus <i>session</i> setiap pengguna yang masuk ke dalam sistem, dan keluar dari halaman <i>dashboard</i> pengguna menuju halaman login awal.

4.2.3 Usecase Scenario

Skenario *Usecase* merupakan penjelasan alur kerja sistem secara rinci meliputi penjelasan prekondisi dan prakondisi juga aksi yang akan dikerjakan sistem ketika pengguna menggunakan sistem. Diagram ini juga menjelaskan *alternatif flow* atau kondisi alternatif ketika pengguna melakukan kesalahan dalam menggunakan sistem, diagram ini dibuat sesuai dengan fitur yang ada pada *use case diagram*.

1. Skenario Masuk.

Skenario masuk menjelaskan tentang alur reaksi antara aktor atau pengguna dengan sistem dimulai dari pertama kali sistem dijalankan, skenario masuk dimulai dari mengisi form pada tampilan login hingga masuk pada tampilan dashboard sesuai dengan hak akses pengguna. Skenario utama dan alternatif pada skenario masuk ditunjukkan pada lampiran A.

2. Skenario Mengelola Data Pengguna.

Skenario mengelola data pengguna merupakan diagram alur reaksi antara aktor atau pengguna dengan sistem, skenario mengelola data pengguna menjelaskan alur aksi utama ketika pengguna memasukkan, mengganti, dan menghapus data pengguna, sedangkan untuk skenario alternatif berisi alur yang akan dijalankan oleh sistem ketika terdapat suatu data yang tidak diisi dan tidak valid. Skenario utama dan alternatif pada skenario ditunjukkan pada lampiran A.

3. Skenario Mengelola Kriteria, Subkriteria, dan Pembobotan Gabah.

Skenario mengelola kriteria, subkriteria, dan pembobotan gabah berisi skenario utama menambahkan dan menghapus kriteria, subkriteria, dan pembobotan yang

digunakan sebagai acuan dalam pembelian gabah, sedangkan skenario alternatif menjelaskan alur yang akan terjadi apabila data yang dimasukkan melebihi range dari pembobotan dan data yang akan dimasukkan kosong. Skenario utama dan alternatif pada skenario ditunjukkan pada tabel 4.3.

Tabel 4.3 Skenario Mengelola Kriteria, Subkriteria, dan Pembobotan Gabah

Nomor usecase	03
Nama usecase	Mengelola Kriteria dan Pembobotan Gabah
Aktor	Admin atau Pakar
Deskripsi	Merupakan fitur untuk mengelola kriteria, subkriteria dan pembobotan sebagai parameter dalam pemilihan gabah terbaik
Pre-Kondisi	Pengguna berada pada halaman utama admin, dan akan memasukkan, mengubah, menghapus kriteria, sub kriteria, dan pembobotan setiap parameter.
Psca Kondisi	Pengguna telah berada pada halaman utama kelola kriteria gabah, dan telah memasukkan, mengubah, menghapus kriteria, sub kriteria, dan pembobotan setiap parameter.
Event Flow	
ALIRAN NORMAL	
Tambah Kriteria	
1. Klik menu Kriteria Gabah	
	2. Menampilkan submenu kelola bobot
3. Klik sub menu kelola bobot	
	4. Menampilkan halaman kriteria dan bobot: <ul style="list-style-type: none"> a. Tombol + Tambah Kriteria b. Tabel kriteria: <ul style="list-style-type: none"> • ID • Kriteria • Nilai

	<ul style="list-style-type: none"> • Bobot (presentase) • Normalisasi • Aksi <p>c. Tombol + (tambah subkriteria)</p> <p>d. Tombol mata (untuk detail subkriteria berdasarkan kriteria)</p> <p>e. Tombol sampah (untuk menghapus kriteria)</p>
5. Klik tombol + Tambah Kriteria	
	<p>6. Menampilkan modal form kriteria:</p> <ul style="list-style-type: none"> a. Nama kriteria (tipe text) b. Bobot kriteria (tipe angka) c. Tombol tambah
<p>7. Mengisi form kriteria:</p> <ul style="list-style-type: none"> a. Nama kriteria (tipe text) b. Bobot kriteria (tipe angka) 	
8. Klik tombol tambah	
	<p>9. Menyimpan data ke database</p>
	<p>10. Menghitung Nilai Bobot dan Normalisasi dengan rumus</p> <p>Bobot : $(\text{nilai} / \text{total nilai}) * 100$</p> <p>Normalisasi : $((\text{nilai} / \text{total nilai}) * 100) / 100$</p>
	<p>11. Menampilkan flash alert "Data Kriteria Gabah Berhasil di Tambahkan"</p>
12. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW	
Data kosong	
7. Klik tombol tambah	

	8. Menampilkan span “data tidak boleh kosong
ALTERNATIF FLOW	
Data tidak sesuai	
7. Klik tombol tambah	
	8. Menampilkan flash alert “bobot harus dalam range 0 - 100”
9. Klik icon (x) untuk menutup flash alert	
ALIRAN NORMAL	
Hapus Kriteria	
1. Klik menu Kriteria Gabah	
	2. Menampilkan submenu kelola bobot
3. Klik sub menu kelola bobot	
	4. Menampilkan halaman kriteria dan bobot: <ol style="list-style-type: none"> a. Tombol + Tambah Kriteria b. Tabel kriteria: <ul style="list-style-type: none"> • ID • Kriteria • Nilai • Bobot (presentase) • Normalisasi • Aksi c. Tombol + (tambah subkriteria) d. Tombol mata (untuk detail subkriteria berdasarkan kriteria) e. Tombol sampah (untuk menghapus kriteria)
5. Klik tombol sampah (untuk menghapus kriteria)	

	6. menampilkan confirm message “ Hapus kriteria ini ?”
7. Klik OK	
	8. Menghapus kriteria dalam database
	9. Menampilkan flash alert “Data Kriteria Gabah Berhasil di Hapus” pada halaman kriteria gabah.
10. Klik icon (x) untuk menutup flash alert	
ALIRAN NORMAL	
Batal Hapus kriteria	
1. Klik menu Kriteria Gabah	
	2. Menampilkan submenu kelola bobot
3. Klik sub menu kelola bobot	
	4. Menampilkan halaman kriteria dan bobot: <ol style="list-style-type: none"> a. Tombol + Tambah Kriteria b. Tabel kriteria: <ul style="list-style-type: none"> • ID • Kriteria • Nilai • Bobot (presentase) • Normalisasi • Aksi c. Tombol + (tambah subkriteria) d. Tombol mata (untuk detail subkriteria berdasarkan kriteria) <p>Tombol sampah (untuk menghapus kriteria)</p>
11. Klik tombol sampah (untuk menghapus kriteria)	

	12. menampilkan confirm message “ Hapus kriteria ini ?”
13. Klik Cancel	
	14. Menampilkan halaman kriteria gabah
ALIRAN NORMAL	
Tambah sub kriteria	
1. Klik menu Kriteria Gabah	
	2. Menampilkan submenu kelola bobot
3. Klik sub menu kelola bobot	
	4. Menampilkan halaman kriteria dan bobot: <ol style="list-style-type: none"> a. Tombol + Tambah Kriteria b. Tabel kriteria: <ul style="list-style-type: none"> • ID • Kriteria • Nilai • Bobot (presentase) • Normalisasi • Aksi c. Tombol + (tambah subkriteria) d. Tombol mata (untuk detail subkriteria berdasarkan kriteria) e. Tombol sampah (untuk menghapus kriteria)
5. Klik Tombol + (tambah subkriteria)	
	6. Menampilkan modal form subkriteria: <ol style="list-style-type: none"> a. ID Kriteria (tipe text) b. Nama Kriteria (tipe text) c. Nama Subkriteria (Tipe Text) d. Nilai Utility (Tipe Angka) e. Tombol Tambah

<p>7. Mengisi modal form subkriteria:</p> <ul style="list-style-type: none"> • Nama Subkriteria (Tipe Text) • Nilai Utility (Tipe Angka) 	
<p>8. Klik Tombol Tambah</p>	
	<p>9. Menyimpan data ke database</p>
	<p>10. Menampilkan flash alert “Data Subkriteria Berhasil di Tambahkan pada kriteria” pada halaman kriteria gabah</p>
<p>11. Klik icon (x) untuk menutup flash alert</p>	
<p>ALIRAN ALTERNATIF Data kosong</p>	
<p>8. Klik Tombol Tambah</p>	
	<p>9. Menampilkan span “data tidak boleh kosong”</p>
<p>ALIRAN NORMAL Lihat detail subkriteria</p>	
<p>1. Klik menu Kriteria Gabah</p>	
	<p>2. Menampilkan submenu kelola bobot</p>
<p>3. Klik sub menu kelola bobot</p>	
	<p>4. Menampilkan halaman kriteria dan bobot:</p> <ol style="list-style-type: none"> a. Tombol + Tambah Kriteria b. Tabel kriteria: <ul style="list-style-type: none"> • ID • Kriteria • Nilai • Bobot (presentase) • Normalisasi • Aksi

	<ul style="list-style-type: none"> c. Tombol + (tambah subkriteria) d. Tombol mata (untuk detail subkriteria berdasarkan kriteria) e. Tombol sampah (untuk menghapus kriteria)
5. Klik Tombol Mata (untuk detail subkriteria berdasarkan kriteria)	
	<ul style="list-style-type: none"> 6. Menampilkan modal tabel detail subkriteria gabah sesuai kriteria <ul style="list-style-type: none"> a. Tabel detail subkriteria <ul style="list-style-type: none"> • No • Subkriteria • Utility • Aksi b. Tombol sampah (untuk menghapus kriteria)
ALIRAN NORMAL	
Hapus sub kriteria	
1. Klik menu Kriteria Gabah	
	2. Menampilkan submenu kelola bobot
3. Klik sub menu kelola bobot	
	<ul style="list-style-type: none"> 4. Menampilkan halaman kriteria dan bobot: <ul style="list-style-type: none"> a. Tombol + Tambah Kriteria b. Tabel kriteria: <ul style="list-style-type: none"> • ID • Kriteria • Nilai • Bobot (presentase) • Normalisasi • Aksi

	<ul style="list-style-type: none"> c. Tombol + (tambah subkriteria) d. Tombol mata (untuk detail subkriteria berdasarkan kriteria) e. Tombol sampah (untuk menghapus kriteria)
5. Klik Tombol Mata (untuk detail subkriteria berdasarkan kriteria)	
	<ul style="list-style-type: none"> 6. Menampilkan modal tabel detail subkriteria gabah sesuai kriteria <ul style="list-style-type: none"> a. Tabel detail subkriteria <ul style="list-style-type: none"> • No • Subkriteria • Utility • Aksi b. Tombol sampah (untuk menghapus subkriteria)
7. Klik tombol sampah (untuk menghapus subkriteria)	
	8. menampilkan confirm message “ Hapus subkriteria ini ?”
9. Klik OK	
	10. Menghapus subkriteria dalam database
	11. Menampilkan flash alert “Data Subkriteria Gabah Berhasil di Hapus” pada halaman kriteria gabah
12. Klik icon (x) untuk menutup flash alert	
ALIRAN NORMAL	
Batal Hapus sub kriteria	
1. Klik menu Kriteria Gabah	

	2. Menampilkan submenu kelola bobot
3. Klik sub menu kelola bobot	
	<p>4. Menampilkan halaman kriteria dan bobot:</p> <ol style="list-style-type: none"> a. Tombol + Tambah Kriteria b. Tabel kriteria: <ul style="list-style-type: none"> • ID • Kriteria • Nilai • Bobot (presentase) • Normalisasi • Aksi c. Tombol + (tambah subkriteria) d. Tombol mata (untuk detail subkriteria berdasarkan kriteria) e. Tombol sampah (untuk menghapus kriteria)
5. Klik Tombol Mata (untuk detail subkriteria berdasarkan kriteria)	
	<p>6. Menampilkan modal tabel detail subkriteria gabah sesuai kriteria</p> <ol style="list-style-type: none"> a. Tabel detail subkriteria <ul style="list-style-type: none"> • No • Subkriteria • Utility • Aksi b. Tombol sampah (untuk menghapus subkriteria)
7. Klik tombol sampah (untuk menghapus subkriteria)	

	8. menampilkan confirm message “ Hapus subkriteria ini ?”
9. Klik Tombol Cancel	
	10. Menampilkan halaman kriteria gabah

4. Skenario Mengelola Kriteria, Subkriteria, dan Pembobotan Beras.

Skenario mengelola kriteria, subkriteria, dan pembobotan beras berisi skenario utama menambahkan dan menghapus kriteria, subkriteria, dan pembobotan yang digunakan sebagai acuan dalam pembelian beras, sedangkan skenario alternatif menjelaskan alur yang akan terjadi apabila data yang dimasukkan melebihi range dari pembobotan dan data yang akan dimasukkan kosong. Skenario utama dan alternatif pada skenario ditunjukkan pada lampiran A.

5. Skenario Mengelola Stok Pembelian Gabah

Skenario mengelola stok pembelian gabah merupakan diagram yang berisi skenario utama menambah dan mengurangi stok pembelian gabah yang harus dibeli, sedangkan skenario alternatif menangani alur ketika data yang dimasukkan tidak valid. Skenario utama dan alternatif pada skenario ditunjukkan pada lampiran A.

6. Skenario Mengelola Stok Pembelian Beras

Skenario mengelola stok pembelian beras merupakan diagram yang berisi skenario utama menambah dan mengurangi stok pembelian beras yang harus dibeli, sedangkan skenario alternatif menangani alur ketika data yang dimasukkan tidak valid. Skenario utama dan alternatif pada skenario ditunjukkan pada lampiran A.

7. Skenario Mengelola Penilaian Mutu dan Harga Gabah

Skenario mengelola penilaian mutu dan harga gabah merupakan diagram yang berisi skenario utama membuat penilaian untuk standarisasi mutu dan harga gabah, dan juga mengatur ulang mutu dan harga setiap jenis mutu gabah, sedangkan skenario alternatif hanya berisi alur yang akan dijalankan ketika data yang akan di masukkan

kosong. Skenario utama dan alternatif pada skenario mengelola penilaian mutu dan harga gabah ditunjukkan pada lampiran A.

8. Skenario Mengelola Penilaian Mutu dan Harga Beras

Skenario mengelola penilaian mutu dan harga beras merupakan diagram yang berisi skenario utama membuat penilaian untuk standarisasi mutu dan harga beras, dan juga mengatur ulang mutu dan harga setiap jenis mutu beras, sedangkan skenario alternatif hanya berisi alur yang akan dijalankan ketika data yang akan di masukkan kosong. Skenario utama dan alternatif pada skenario ditunjukkan pada lampiran A.

9. Skenario Mengelola Pemasok Gabah

Skenario mengelola pemasok gabah merupakan diagram yang menjelaskan skenario utama menambah, mengubah, dan menghapus data pemasok yang menjual gabahnya. Data pemasok dimasukkan seperti nama, berat yang akan dijual, dan alamat. Skenario utama dan alternatif pada skenario ditunjukkan pada lampiran A.

10. Skenario Mengelola Pemasok Beras

Skenario mengelola pemasok beras merupakan diagram yang menjelaskan skenario utama menambah, mengubah, dan menghapus data pemasok yang menjual berasnya. Skenario utama dan alternatif pada skenario ditunjukkan pada lampiran A.

11. Skenario Membuat Penilaian Pemasok Gabah

Skenario membuat penilaian pemasok gabah adalah diagram yang menjelaskan skenario utama berupa alur penilaian gabah dari setiap pemasok gabah. Sedangkan skenario alternatif berupa penanganan error jika data yang akan dimasukkan kosong. Skenario utama dan alternatif pada skenario ditunjukkan pada tabel 4.4.

Tabel 4.4 Skenario Membuat Penilaian Pemasok Gabah

Nomor usecase	11
Nama usecase	Membuat Penilaian Pemasok Beras
Aktor	Pegawai

Deskripsi	Merupakan fitur untuk membuat penilaian mutu dan juga penilaian metode SMART pada pemasok beras sehingga didapatkan nilai yang dapat dijadikan acuan untuk perbandingan dan penentuan mutu juga harga untuk setiap pemasok beras.
Pre-Kondisi	Pengguna berada pada halaman utama pengguna, dan akan membuat penilaian mutu dan penilaian metode SMART untuk setiap pemasok beras.
Pasca Kondisi	Pengguna telah berada pada halaman utama data pemasok beras, dan membuat penilaian mutu dan penilaian metode SMART untuk setiap pemasok beras.
Event Flow	
ALIRAN NORMAL	
Membuat Penilaian Pemasok Beras	
1. Klik menu Penilaian Padi	
	2. Menampilkan submenu data pemasok beras dan data pemasok beras
3. Klik sub menu data pemasok beras	
	4. Menampilkan halaman data pemasok beras <ol style="list-style-type: none"> a. Tombol “+ Tambah Pemasok” b. Tabel list pengguna: <ul style="list-style-type: none"> • ID • Nama • Alamat • Berat(Kg) • Tanggal Masuk

	<ul style="list-style-type: none">• Aksic. Tombol Note (Penilaian pemasok beras)d. Tombol pensil (mengubah data)e. Tombol sampah (menghapus data)
5. Klik Tombol Note (Penilaian pemasok beras)	
	<p>6. Menampilkan halaman penilaian pemasok beras</p> <ul style="list-style-type: none">a. ID Pemasok (tipe text) readonlyb. Nama Pemasok (tipe text) readonlyc. Butir Patah (tipe combobox)d. Kadar Air (tipe combobox)e. Butir Kepala (tipe combobox)f. Hama (tipe combobox)g. Bau Asing (tipe combobox)h. Butir Mengapur (tipe combobox)i. Benda Asing (tipe combobox)j. Butir Merah (tipe combobox)k. Butir Kuning (tipe combobox)l. Butir Menir (tipe combobox)

	<ul style="list-style-type: none">m. Butir Gabah (tipe combobox)n. Derajat Sosoh (tipe combobox)o. Tombol Hitung
<p>7. Mengisi form tambah pemasok beras</p> <ul style="list-style-type: none">a. Butir Patah (tipe combobox)b. Kadar Air (tipe combobox)c. Butir Kepala (tipe combobox)d. Hama (tipe combobox)e. Bau Asing (tipe combobox)f. Butir Mengapur (tipe combobox)g. Benda Asing (tipe combobox)h. Butir Merah (tipe combobox)i. Butir Kuning (tipe combobox)j. Butir Menir (tipe combobox)k. Butir Gabah (tipe combobox)l. Derajat Sosoh (tipe combobox)	
<p>8. Klik tombol Hitung</p>	

	<p>9. Menghitung Hasil Penilaian Mutu dan Perhitungan metode SMART dengan Rumus sebagai berikut</p> $SMART = \sum_{j=1}^k w_j u_i(a_i), \quad i = 1, 2, \dots, k$
	10. Menyimpan hasil perhitungan kedalam database.
11. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW Data Kosong	
8. klik tombol Hitung	
	9. Menampilkan span “data tidak boleh kosong”

12. Skenario Membuat Penilaian Pemasok Beras

Skenario membuat penilaian pemasok beras adalah diagram yang menjelaskan skenario utama berupa alur penilaian beras dari setiap pemasok beras. Sedangkan skenario alternatif berupa penanganan error jika data yang akan dimasukkan kosong. Skenario utama dan alternatif pada skenario ditunjukkan pada lampiran A.

13. Skenario Melihat Hasil Penilaian Gabah

Skenario melihat hasil penilaian gabah adalah diagram yang menjelaskan alur dari skenario utama berupa aksi yang dikerjakan aktor dan reaksi yang dihasilkan sistem untuk melihat hasil dari penilaian gabah. Skenario utama pada skenario ditunjukkan pada lampiran A

14. Skenario Melihat Hasil Penilaian Beras

Skenario melihat hasil penilaian beras adalah diagram yang menjelaskan alur dari skenario utama berupa aksi yang dikerjakan aktor dan reaksi yang dihasilkan

sistem untuk melihat hasil dari penilaian beras. Skenario utama pada skenario ditunjukkan pada lampiran A.

15. Skenario Membuat Pembelian Gabah

Skenario membuat pembelian gabah merupakan diagram yang menggambarkan alur dari proses membuat pembelian gabah, dimana skenario utama adalah melakukan pembelian gabah baik sebagian maupun semua stok dari pemasok gabah. Skenario alternatif menangani jika terdapat kesalahan dalam memasukkan jumlah stok yang akan dibeli baik terlalu banyak atau melebihi batas yang seharusnya dibeli. Skenario utama pada skenario ditunjukkan pada lampiran A.

16. Skenario Membuat Pembelian Beras

Skenario membuat pembelian beras merupakan diagram yang menggambarkan alur dari proses membuat pembelian beras, dimana skenario utama adalah melakukan pembelian beras baik sebagian maupun semua stok dari pemasok beras. Skenario alternatif menangani jika terdapat kesalahan dalam memasukkan jumlah stok yang akan dibeli baik terlalu banyak atau melebihi batas yang seharusnya dibeli. Skenario utama pada skenario ditunjukkan pada lampiran A.

17. Skenario Melihat Riwayat Pembelian Gabah

Skenario melihat riwayat pembelian gabah merupakan diagram yang menggambarkan alur dari proses melihat riwayat pembelian gabah, dimana skenario utama hanyalah proses menampilkan riwayat pembelian gabah. Skenario utama pada skenario ditunjukkan pada lampiran A.

18. Skenario Melihat Riwayat Pembelian Beras

Skenario melihat riwayat pembelian beras merupakan diagram yang menggambarkan alur dari proses melihat riwayat pembelian beras, dimana skenario utama hanyalah proses menampilkan riwayat pembelian beras. Skenario utama pada skenario ditunjukkan pada lampiran A.

19. Skenario Keluar

Skenario keluar adalah diagram alur yang menggambarkan bagaimana aksi aktor dan reaksi sistem ketika akan keluar dari halaman *dashboard* masing – masing hak akses. Skenario utama pada skenario ditunjukkan pada lampiran A.

4.2.4 Activity Diagram

Activity diagram merupakan diagram presentasi grafis yang menggambarkan proses operasional dan penyebab alur dari sebuah tahap dalam sistem (Kaewchinporn dan Limpiyakorn, 2013). Activity diagram ditampilkan dalam dua pembagian aksi dari sebuah aktor dan sistem, mulai dari proses *start*, *decission*, sampai *finish* dari sebuah alur sistem. Diagram aktifitas dari sistem yang akan dikembangkan adalah sebagai berikut.

1. Activity diagram Masuk.

Diagram aktifitas pada fitur masuk menjelaskan alur ketika pengguna melakukan aksi dan apa reaksi yang dilakukan sistem sesuai dengan tanda panah yang mengarah, setiap kondisi *alternatif flow* pada skenario digambarkan dengan diagram berbentuk belah ketupat dengan setiap alternatif proses yang akan dikerjakan dari setiap kemungkinan yang akan terjadi. Diagram aktifitas dari fitur masuk dapat dilihat pada lampiran B.

2. Activity diagram Mengelola Data Pengguna.

Diagram aktifitas pada fitur Mengelola data pengguna menjelaskan alur dalam membuat, mengubah, dan menghapus data pengguna yang dapat dilihat melalui pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam mengelola data pengguna dapat dilihat pada lampiran B.

3. Activity diagram Mengelola Kriteria, Subkriteria, dan Pembobotan Gabah.

Diagram aktifitas pada fitur mengelola kriteria, subkriteria, dan pembobotan gabah dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram

aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam mengelola kriteria, subkriteria, dan pembobotan gabah dapat dilihat pada gambar 4.3.

4. *Activity diagram* Skenario Mengelola Kriteria, Subkriteria, dan Pembobotan Beras.

Diagram aktifitas pada fitur mengelola kriteria, subkriteria, dan pembobotan beras dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam mengelola kriteria, subkriteria, dan pembobotan beras dapat dilihat pada lampiran B.

5. *Activity diagram* Mengelola Stok Pembelian Gabah

Diagram aktifitas pada fitur mengelola stok pembelian gabah dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam mengelola sto pembelian gabah dapat dilihat pada lampiran B.

6. *Activity diagram* Mengelola Stok Pembelian Beras

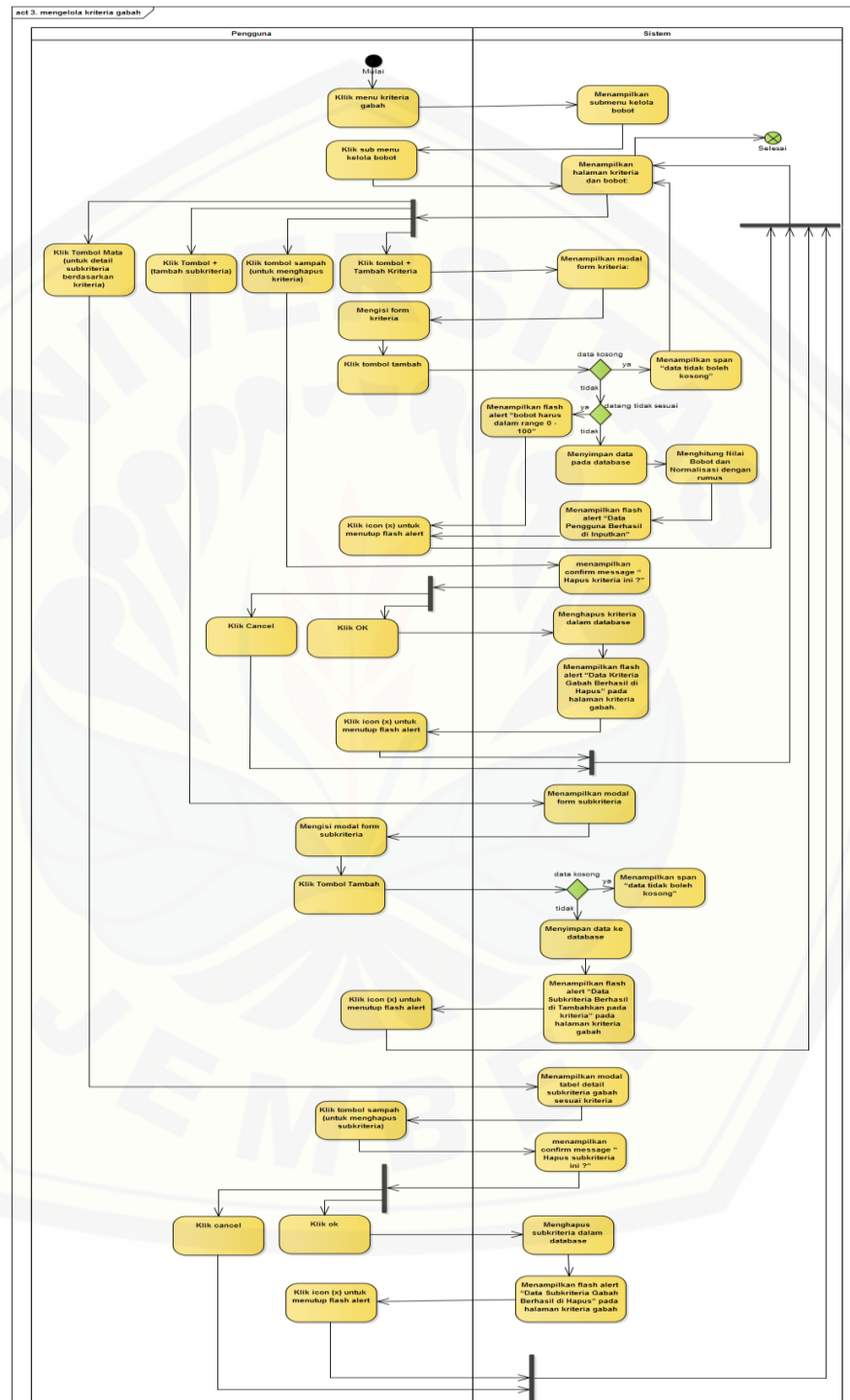
Diagram aktifitas pada fitur mengelola stok pembelian beras dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam mengelola stok pembelian beras dapat dilihat pada lampiran B.

7. *Activity diagram* Mengelola Penilaian Mutu dan Harga Gabah

Diagram aktifitas pada fitur mengelola penilaian mutu dan harga gabah dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam mengelola penilaian mutu dan harga gabah dapat dilihat pada lampiran B.

8. *Activity diagram* Mengelola Penilaian Mutu dan Harga Beras

Diagram aktifitas pada fitur mengelola penilaian mutu dan harga beras dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam mengelola penilaian mutu dan harga beras dapat dilihat pada lampiran B.



Gambar 4.3 Activity Diagram Mengelola Kriteria, Subkriteria, dan Pembobotan Gabah.

9. *Activity diagram* Mengelola Pemasok Gabah

Diagram aktifitas pada fitur mengelola pemasok gabah dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam mengelola pemasok gabah dapat dilihat pada lampiran B.

10. *Activity diagram* Mengelola Pemasok Beras

Diagram aktifitas pada fitur mengelola pemasok beras dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam mengelola pemasok beras dapat dilihat pada lampiran B.

11. *Activity diagram* Membuat Penilaian Pemasok Gabah

Diagram aktifitas pada fitur membuat penilaian pemasok gabah dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam membuat penilaian pemasok gabah dapat dilihat pada gambar 4.4.

12. *Activity diagram* Membuat Penilaian Pemasok Beras

Diagram aktifitas pada fitur membuat penilaian pemasok beras dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam membuat penilaian pemasok beras dapat dilihat pada lampiran B.

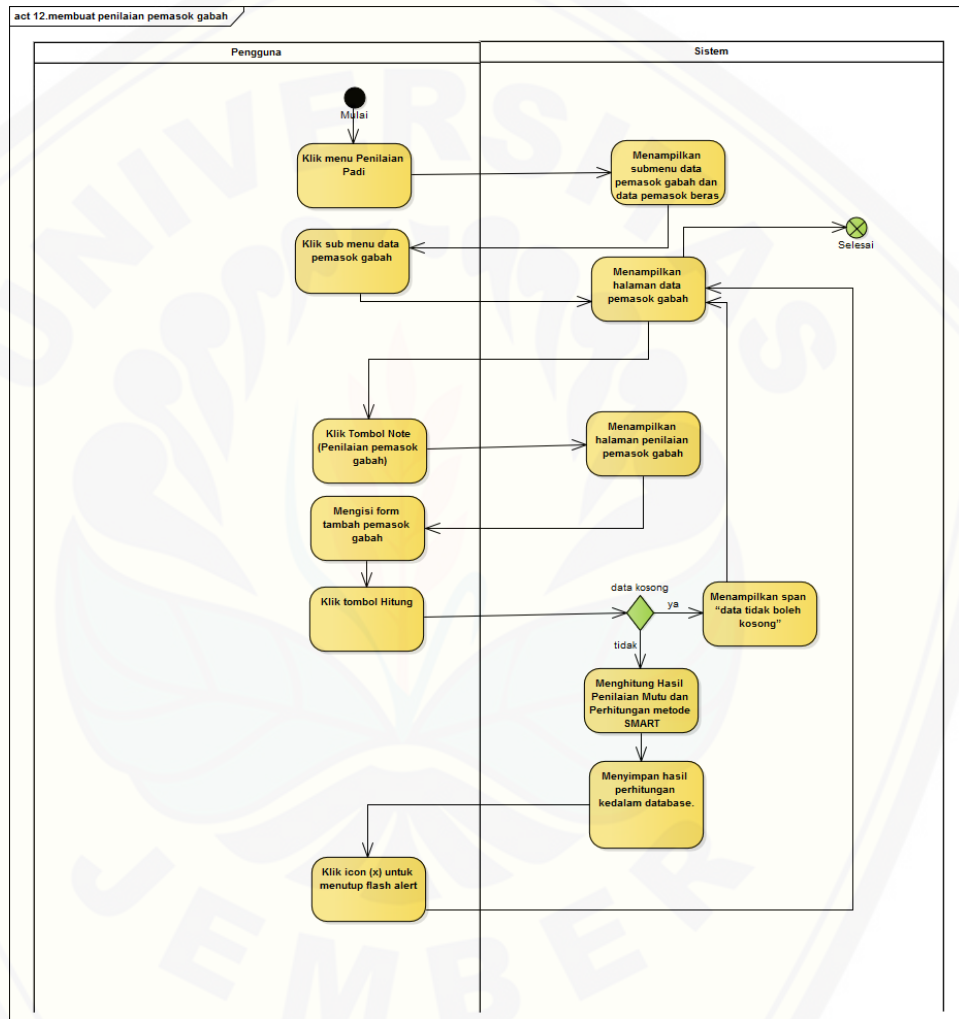
13. *Activity diagram* Melihat Hasil Penilaian Gabah

Diagram aktifitas pada fitur melihat hasil penilaian gabah dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam melihat hasil penilaian gabah dapat dilihat pada lampiran B.

14. *Activity diagram* Melihat Hasil Penilaian Beras

Diagram aktifitas pada fitur melihat hasil penilaian beras dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol

dan reaksi yang akan dikerjakan sistem dalam melihat hasil penilaian beras dapat dilihat pada lampiran B.



Gambar 4.4 Activity diagram Membuat Penilaian Pemasok Gabah

Gambar 4.4 menggambarkan diagram alur kerja sistem yang dilihat dari dua sisi yaitu pengguna dan sistem, dari sisi sistem pengguna hanya memiliki satu tombol yang dapat digunakan untuk proses penilaian pemasok gabah, setelah tombol ditekan maka sistem menjalankan aksi untuk menampilkan halaman penilaian sehingga pengguna berhak melakukan penilaian pemasok gabah berdasarkan gabah yang mereka jual.

Diagram juga menunjukkan terdapat kondisi dimana jika salah satu form ada yang tidak terisi, maka sistem melakukan aksi untuk menampilkan span berupa peringatan agar form harus diisi, dan jika data telah lengkap maka proses penilaian dapat dilakukan dan menyimpan ke dalam *database* sistem lalu mengembalikan pada halaman pemasok gabah, dan dinyatakan selesai dengan petunjuk *finish* yang ada pada diagram aktifitas.

15. *Activity diagram* Membuat Pembelian Gabah

Diagram aktifitas pada fitur membuat pembelian gabah dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam membuat pembelian gabah dapat dilihat pada lampiran B.

16. *Activity diagram* Membuat Pembelian Beras

Diagram aktifitas pada fitur membuat pembelian beras dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam membuat pembelian beras dapat dilihat pada lampiran B.

17. *Activity diagram* Melihat Riwayat Pembelian Gabah

Diagram aktifitas pada fitur melihat riwayat pembelian gabah dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam melihat riwayat pembelian gabah dapat dilihat pada lampiran B.

18. *Activity diagram* Melihat Riwayat Pembelian Beras

Diagram aktifitas pada fitur melihat riwayat pembelian beras dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam melihat riwayat pembelian beras dapat dilihat pada lampiran B.

19. *Activity diagram* Keluar

Diagram aktifitas pada fitur keluar dapat dilihat pada pembagian partisi berdasarkan aktor dan sistem. Diagram aktifitas dari setiap tombol dan reaksi yang akan dikerjakan sistem dalam fitur keluar atau *logout* dapat dilihat pada lampiran B.

4.2.5 Sequence Diagram

Sequence diagram merupakan diagram yang menunjukkan interaksi antara *method*, *prosedur*, *class* dalam sebuah urutan secara tepat (Minhas dkk., 2015). *Sequence diagram* merupakan desain perancangan yang mendekati proses pengkodean, maka dari itu di dalam *sequence diagram* bahasa yang dituliskan merupakan bahasa pengkodean, dimana bahasa pengkodean tersebut menjelaskan jalannya alur dalam fitur secara berurutan. *Sequence diagram* pada aplikasi yang dikembangkan meliputi :

a. *Sequence Masuk.*

Pada diagram *sequence* masuk terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan antara dua pengguna yaitu admin dan pegawai dengan *controller C_ceklogin* dan *model M_login*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

b. *Sequence Mengelola Data Pengguna.*

Pada diagram *sequence* mengelola data pengguna terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan untuk pengguna admin, dengan *controller C_pegawai* dan *model M_pegawai*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

c. *Sequence* Mengelola Kriteria, Subkriteria, dan Pembobotan Gabah.

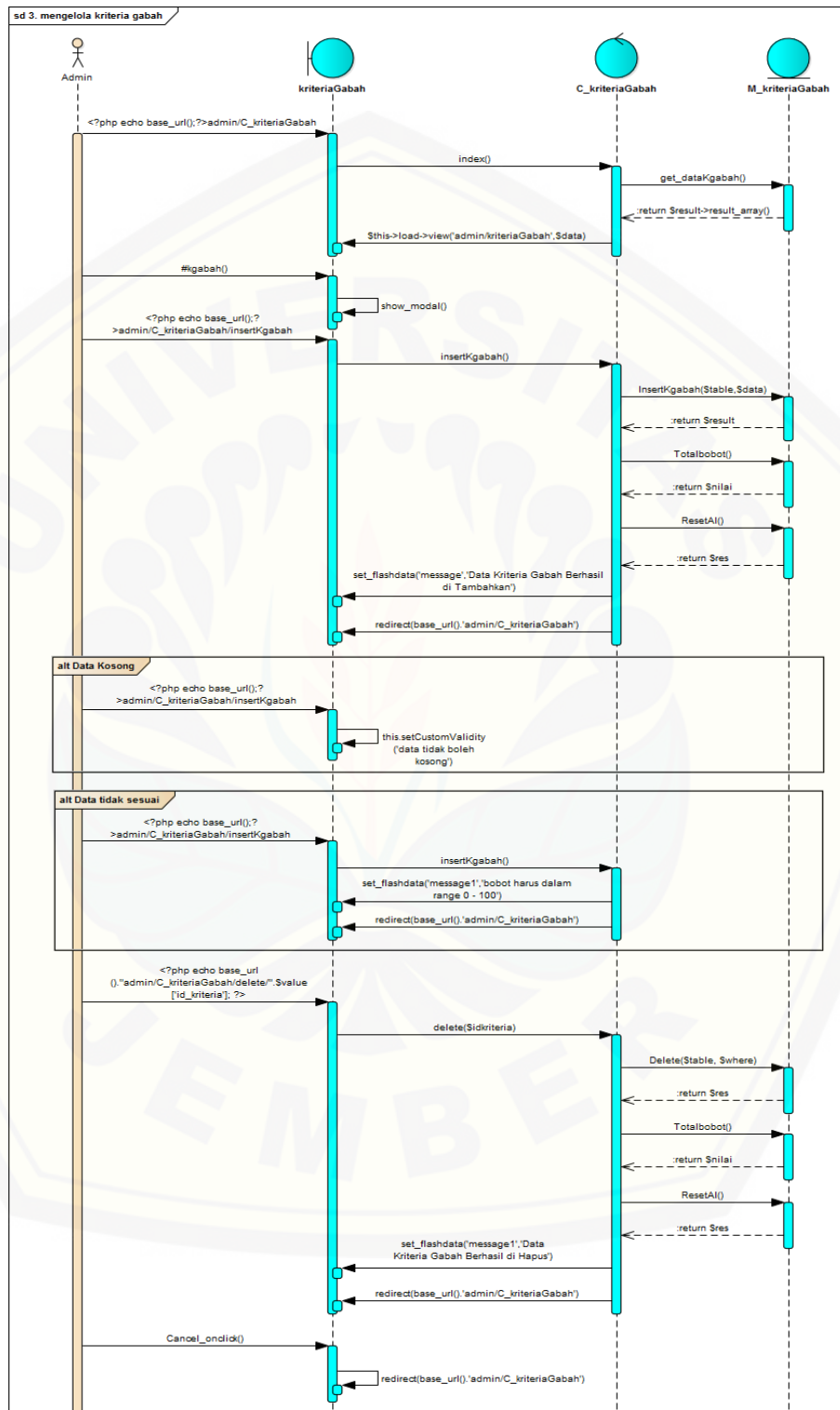
Pada diagram *sequence* mengelola kriteria, subkriteria, dan pembobotan gabah terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *kriteriaGabah* yang dikelola oleh admin dengan *controller C_kriteriaGabah* dan *model M_kriteriaGabah*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada gambar 4.5.

d. *Sequence* Mengelola Kriteria, Subkriteria, dan Pembobotan Beras.

Pada diagram *sequence* mengelola kriteria, subkriteria, dan pembobotan beras terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *kriteriaBeras* yang dikelola oleh admin dengan *controller C_kriteriaBeras* dan *model M_kriteriaBeras*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

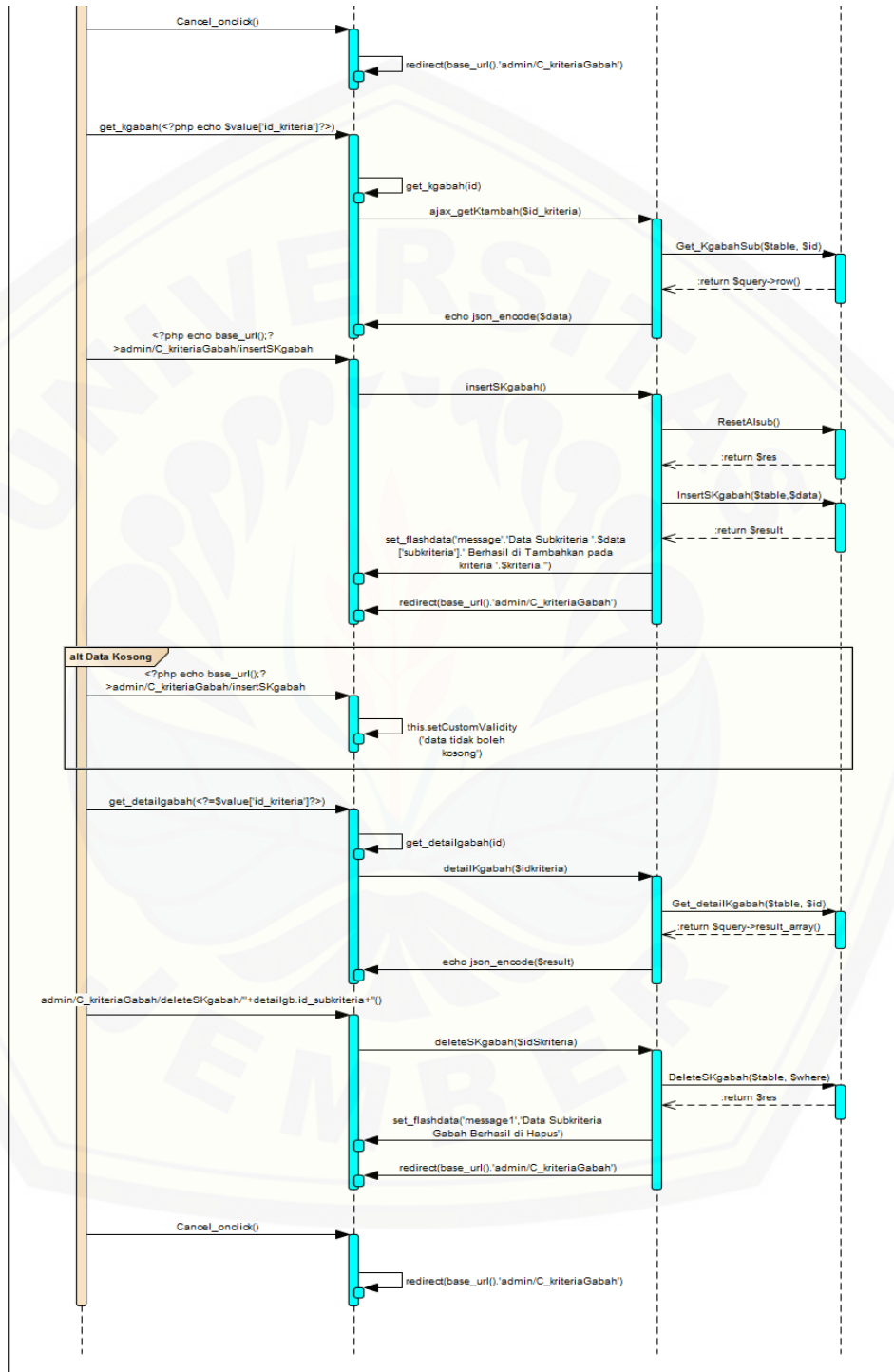
e. *Sequence* Mengelola Stok Pembelian Gabah

Pada diagram *sequence* mengelola stok pembelian gabah terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *kelolaStokGabah* yang dikelola oleh admin dengan *controller C_kelolaStokGbh* dan *model M_stokGabah*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.



Dilanjutkan

Lanjutan



Gambar 4.5 Sequence Diagram Mengelola Mengelola Kriteria, Subkriteria, dan Pembobotan Gabah

f. *Sequence* Mengelola Stok Pembelian Beras

Pada diagram *sequence* mengelola stok pembelian beras terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *kelolaStokBeras* yang dikelola oleh admin dengan *controller C_kelolaStokBrs* dan *model M_stokBeras*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

g. *Sequence* Mengelola Penilaian Mutu dan Harga Gabah

Pada diagram *sequence* mengelola penilaian mutu dan harga gabah terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *PenilaianMHgabah* yang dikelola oleh admin dengan *controller C_penilaianMHgabah* dan *model M_penilaianGabah*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

h. *Sequence* Mengelola Penilaian Mutu dan Harga Beras

Pada diagram *sequence* mengelola penilaian mutu dan harga beras terdapat beberapa interaksi antar kelas yang menghubungkan fungsi yang ada. Kelas yang saling terhubung adalah kelas tampilan *PenilaianMHberas* yang dikelola oleh admin dengan *controller C_penilaianMHberas* dan *model M_penilaianBeras*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

i. *Sequence* Mengelola Pemasok Gabah

Pada diagram *sequence* mengelola pemasok gabah terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *pemasokGabah* yang dikelola oleh pegawai dengan *controller C_pemasokGabah* dan *model M_pemasokGabah*. Penggambaran

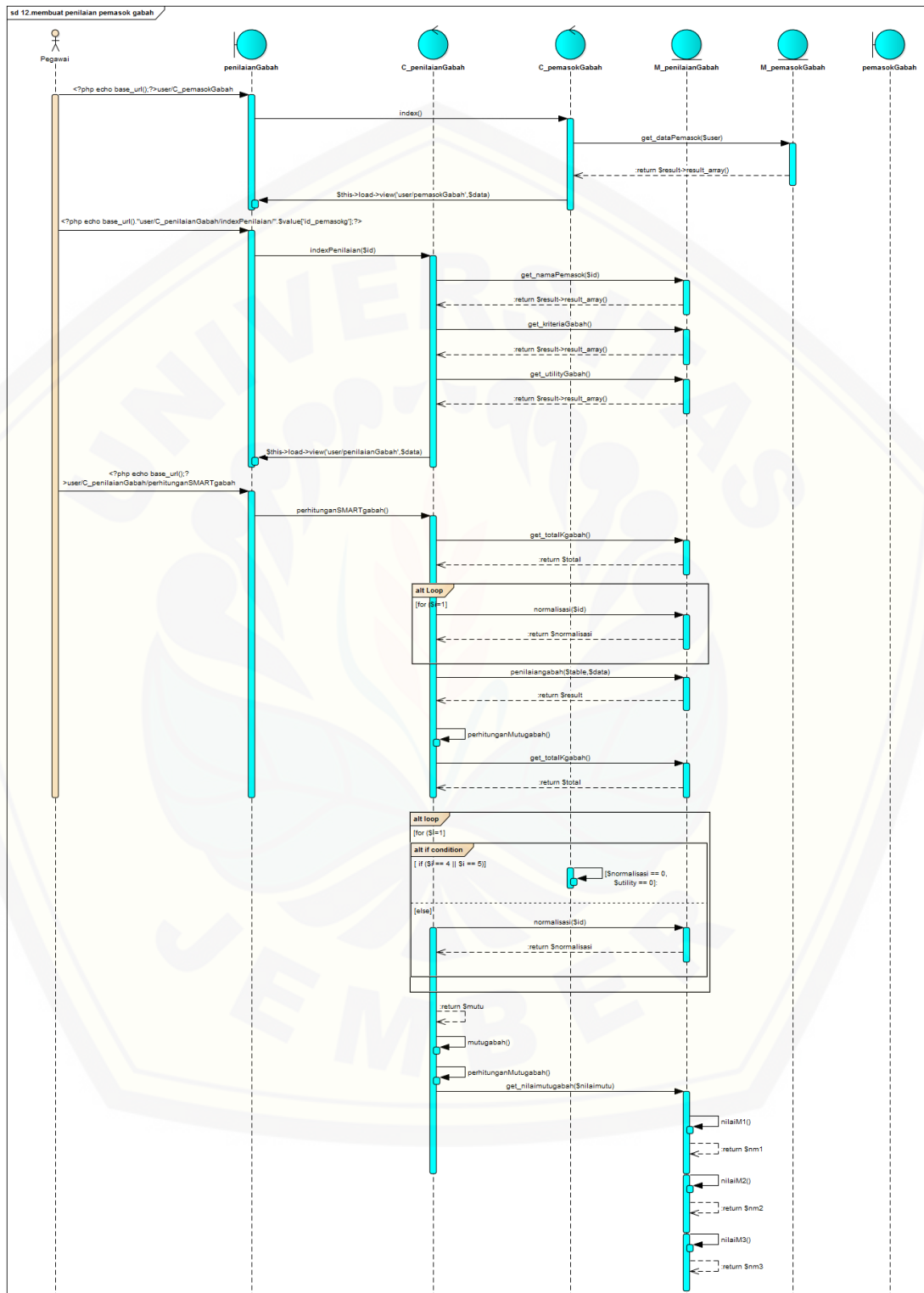
interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

j. *Sequence* Mengelola Pemasok Beras

Pada diagram *sequence* mengelola pemasok beras terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *pemasokBeras* yang dikelola oleh pegawai dengan *controller* *C_pemasokBeras* dan *model* *M_pemasokBeras*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

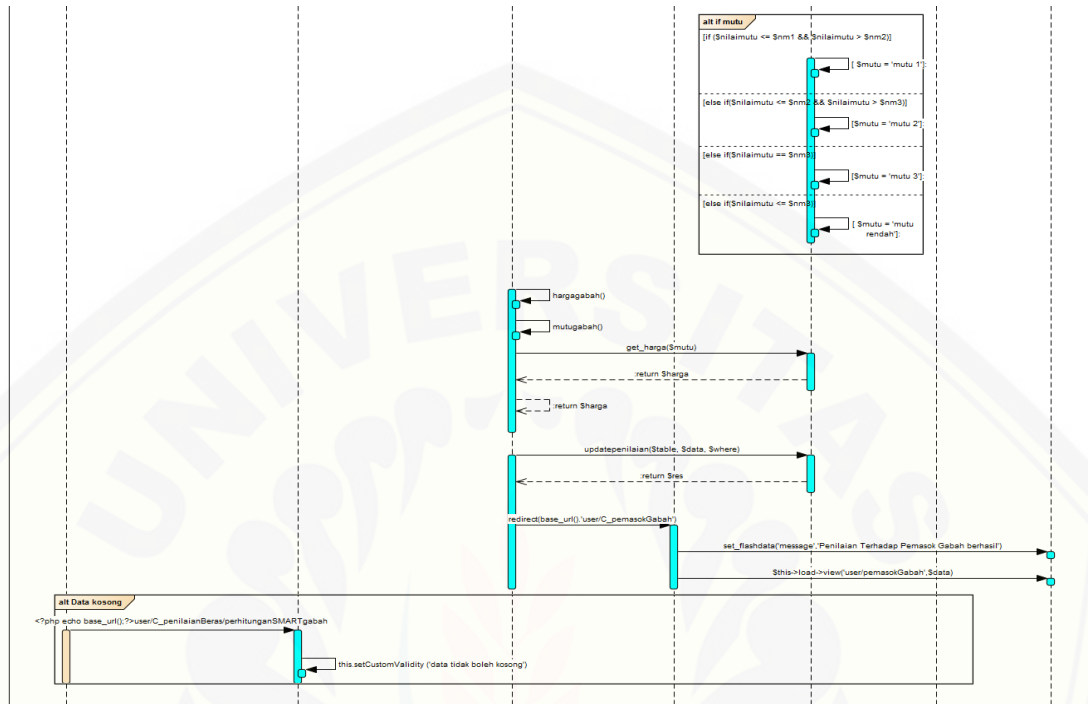
k. *Sequence* Membuat Penilaian Pemasok Gabah

Pada diagram *sequence* membuat penilaian pemasok gabah terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *penilaianGabah* dan *pemasokGabah* yang dikelola oleh pegawai dengan *controller* *C_penilaianGabah* dan *C_pemasokGabah* serta dengan *model* *M_pemasokGabah* dan *M_penilaian gabah*. Pada kelas tampilan pengguna menekan tombol pada tampilan dan mengirim proses kepada controller, method dalam controller dipanggil sehingga fungsi perhitungan yang didapatkan dari model juga terpanggil oleh controller untuk selanjutnya diproses dengan nilai yang telah dikembalikan dari model, selanjutnya controller mengirim kembali proses ke dalam *database* dan tampilan sehingga hasil dari perhitungan penilaian setiap pemasok gabah dapat ditampilkan. Fungsi yang ada pada *controller* dan *model* tidak jarang saling panggil antar fungsi yang lain untuk mengoptimalkan proses pengkodean dan supaya pengkodean tidak terlalu panjang. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada gambar 4.6.



Dilanjutkan

Lanjutan



Gambar 4.6 Sequence diagram Membuat Penilaian Pemasok Gabah

1. Sequence Membuat Penilaian Pemasok Beras

Pada diagram *sequence* membuat penilaian pemasok beras terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *penilaianBeras* dan *pemasokBeras* yang dikelola oleh pegawai dengan *controller C_penilaianBeras* dan *C_pemasokBeras* serta dengan *model M_pemasokBeras* dan *M_penilaianberas*.. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

m. Sequence Melihat Hasil Penilaian Gabah

Pada diagram *sequence* membuat melihat hasil penilaian gabah terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *hasilPengabah* yang dikelola oleh pegawai dengan *controller C_penilaianGabah* dan *C_pembelianGabah* serta

dengan *model M_pembelianGabah dan M_penilaianGabah*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

n. *Sequence* Melihat Hasil Penilaian Beras

Pada diagram *sequence* membuat melihat hasil penilaian beras terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *hasilPenberas* yang dikelola oleh pegawai dengan *controller C_penilaianBeras dan C_pembelianBeras* serta dengan *model M_pembelianBeras dan M_penilaianBeras*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

o. *Sequence* Membuat Pembelian Gabah

Pada diagram *sequence* membuat pembelian gabah terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *hasilPengabah* yang dikelola oleh pegawai dengan *controller C_pembelianGabah* serta dengan *model M_pembelianGabah*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

p. *Sequence* Membuat Pembelian Beras

Pada diagram *sequence* membuat pembelian beras terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *hasilPenberas* yang dikelola oleh pegawai dengan *controller C_pembelianBeras* serta dengan *model M_pembelianBeras*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

q. *Sequence* Melihat Riwayat Pembelian Gabah

Pada diagram *sequence* membuat pembelian gabah terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang

saling terhubung adalah kelas tampilan *gabahbeli* yang dikelola oleh pegawai dengan *controller C_gabahbelli* serta dengan *model M_stokGabah*. Diagram menggambarkan juga proses terjadinya kondisi atau *if* yang mengatur proses pembelian gabah secara sebagian atau seluruh stok yang dimiliki pemasok. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

r. *Sequence* Melihat Riwayat Pembelian Beras

Pada diagram *sequence* membuat pembelian beras terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas tampilan *berasbeli* yang dikelola oleh pegawai dengan *controller C_berasbelli* serta dengan *model M_stokBeras*. Diagram menggambarkan juga proses terjadinya kondisi atau *if* yang mengatur proses pembelian beras secara sebagian atau seluruh stok yang dimiliki pemasok. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

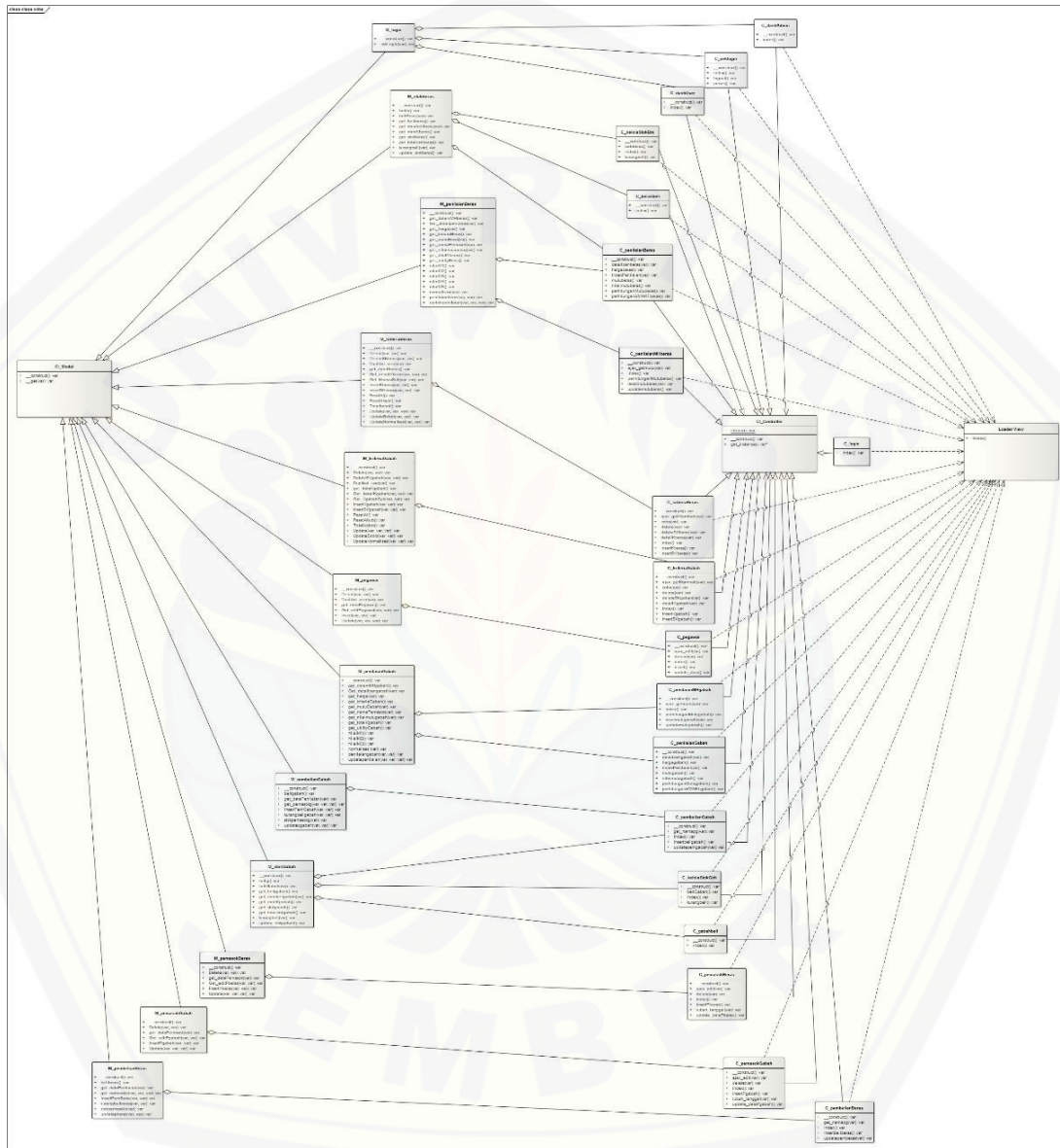
s. *Sequence* Keluar

Pada diagram *sequence* keluar terdapat beberapa interaksi antar kelas yang menghubungkan fungsi – fungsi dalam kelas yang ada. Kelas yang saling terhubung adalah kelas pada setiap tampilan *dashboard* yang dikelola oleh admin dan pegawai dengan *controller C_ceklogin* serta dengan *model session* yang disediakan oleh *framework Codeigniter*. Penggambaran interaksi antara kelas dan method yang akan dikembangkan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.

4.2.6 *Class Diagram*

Class diagram merupakan struktur yang menggambarkan desain perangkat lunak sehingga programmer dapat menggunakannya untuk mengetahui apa yang akan dilakukan selanjutnya (Boustedt, 2010). *Class diagram* berisi relasi antar sebuah kelas dimana didalam setiap kelas terdapat fungsi, parameter, dan atribut. Relasi dalam kelas

dapat digunakan untuk memudahkan dalam pengkodean oleh *programmer*. Class diagram pada penelitian ini ditunjukkan pada gambar 4.7



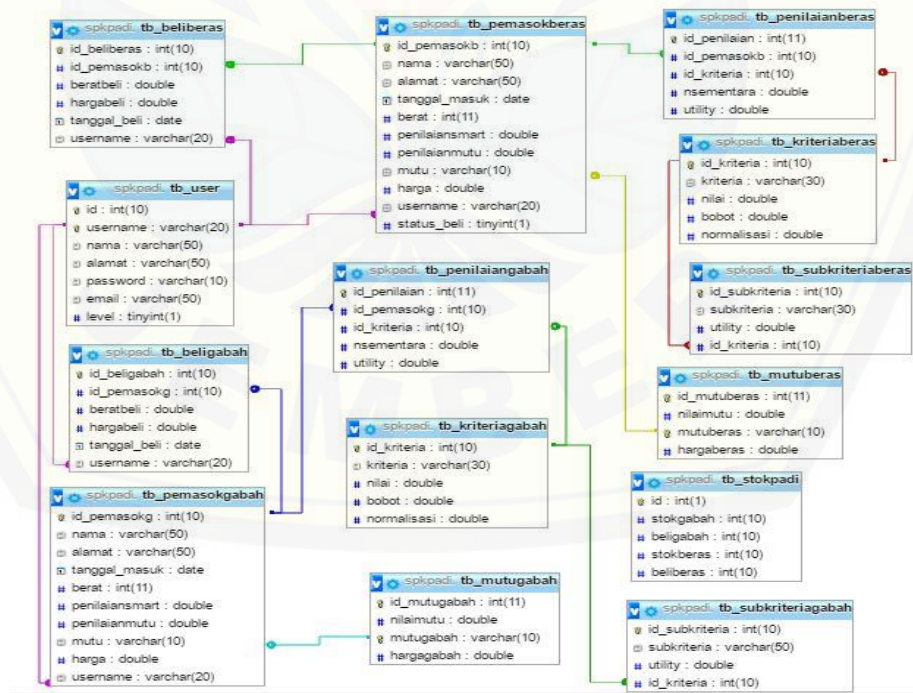
Gambar 4.7 *Class diagram* Sistem Penunjang Keputusan Pembelian Padi Terbaik

Gambar 4.7 menggambarkan kelas yang akan digunakan untuk proses pengkodean. Dari setiap kelas *controller* berelasi generalisasi dengan kelas *CI_Controller*, bergitu juga dengan setiap kelas *model* yang memiliki relasi yang sama.

Generalisasi berarti class tersebut memiliki hubungan antara kelas umum ke kelas khusus yang memang disediakan oleh *CI_Controller*. Sedangkan untuk kelas *Controller* dengan model berelasi agregasi yang berarti memiliki hubungan secara keseluruhan antar fungsinya. Kelas *Controller* dan *View_loader* memiliki relasi *dependency* atau kebergantungan, dikarenakan kelas view tidak dapat menampilkan data jika kelas controller tidak ada. Maka dari semua bagian kelas tersebut, terlihat interaksi antara kelas model, view, dan controller.

4.2.7 Entity Relations Diagram

Entity Relations Diagram merupakan diagram yang digolongkan pada proses desain *database*. Desain *database* adalah proses pengembangan *database* yang dilakukan oleh analis dengan mempertimbangkan beberapa definisi masalah yang didapatkan untuk membangun struktur logis data (Letkowski, 2014). Desain *database* dari sistem yang akan dikembangkan dapat dilihat pada gambar 4.8



Gambar 4.8 Entity Relationship Diagram

Gambar 4.8 menggambarkan rancangan data yang saling berelasi dalam database, relasi ini dihubungkan melalui *primary key* atau kunci utama, nilai unik, yang direlasikan ke *foreign key* atau kunci turunan. Dapat dicontohkan dari gambar 4.8 bahwa tabel *tb_kriteriaberas* berelasi dengan *tb_subkriteriaberas*, relasi ini menunjukkan relasi *one to many*, dimana dapat diartikan satu subkriteria bisa memiliki banyak subkriteria.

4.3 Implementasi

Tahap selanjutnya setelah proses desain sistem merupakan tahap implementasi, tahap implementasi merupakan tahap pembuatan program dengan cara pengkodean atau koding dengan mengacu kepada desain sistem yang telah dibuat. Pengkodean dikerjakan dengan menggunakan bahasa *Page Hyper Text Pre-Processor (PHP)* pada framework *Codeigniter*. Manajemen database dikerjakan dengan bantuan *Database Managemet System MySQL*. Proses pengkodean dilakukan guna mendapatkan sistem yang dapat dioperasikan oleh pengguna.

Proses implementasi metode SMART kedalam metode koding terletak pada fitur mengelola kriteria, subkriteria, dan pembobotan gabah juga beras. Setelah itu perhitungan SMART ada dalam proses perhitungan penilaian setiap pemasok gabah dan beras. Proses pembobotan kriteria, subkriteria gabah dan perhitungan normalisasi terdapat pada kelas *C_kriteriaGabah* dan *M_kriteriaGabah*, sedangkan untuk proses perhitungan metode SMART terdapat pada kelas *C_penilaianGabah* dan *M_penilaianGabah*.

1. Kelas *C_kriteriaGabah*

Kelas *C_kriteriaGabah* berfungsi mengatur jalannya data antara model dengan tampilan, pada kelas tersebut terdapat proses perhitungan bobot dan juga normalisasi untuk setiap data kriteria dan juga subkriteria gabah yang telah dimasukkan. Kelas tersebut berada pada folder *default controller* yang telah disediakan oleh *framework Codeigniter*. Penulisan kode program pada kelas *C_kriteriaGabah* dapat dilihat pada gambar 4.9 berikut.

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class C_kriteriaGabah extends CI_Controller {
    public function __construct(){
        parent::__construct();
        $this->load->library('session');
        $this->load->helper('url');

        if($this->session->userdata('logged_in') == '1' && $this->session->userdata('level') == '1')
        {
            $this->load->model('M_kriteriaGabah');
        }else{
            redirect(base_url()).'C_login';
        }
    }
    public function index(){
        $data['kgabah'] = $this->M_kriteriaGabah->get_dataKgabah();
        $this->load->view('admin/kriteriaGabah',$data);
    }
    public function insertKgabah(){
        $cekbobot = $this->input->post('nilai');
        if ($cekbobot >= 0 && $cekbobot <= 100) {
            $data = array(
                'kriteria' => $this->input->post('kriteria'),
                'nilai' => $this->input->post('nilai')
            );
            $this->M_kriteriaGabah->InsertKgabah('tb_kriteriagabah', $data);
            $this->M_kriteriaGabah->Totalbobot();
            $this->M_kriteriaGabah->ResetAI();
            $this->session->set_flashdata('message','Data Kriteria Gabah Berhasil di Tambahkan');
            redirect(base_url()).'admin/C_kriteriaGabah';
        }else{
            $this->session->set_flashdata('message1','bobot harus dalam range 0 - 100');
            redirect(base_url()).'admin/C_kriteriaGabah';
        }
    }
    public function delete($idkriteria){
        $idkriteria = array('id_kriteria' => $idkriteria);
        $this->M_kriteriaGabah->Delete('tb_kriteriagabah', $idkriteria);
        $this->M_kriteriaGabah->Totalbobot();
        $this->M_kriteriaGabah->ResetAI();
        $this->session->set_flashdata('message1','Data Kriteria Gabah Berhasil di Hapus');
        redirect(base_url()).'admin/C_kriteriaGabah';
    }
    public function ajax_getKtambah($id_kriteria){
        $data = $this->M_kriteriaGabah->Get_KgabahSub('tb_kriteriagabah',$id_kriteria);
        echo json_encode($data);
    }
    public function insertSKgabah(){
        $data = array(
            'id_kriteria' => $this->input->post('idk'),
            'subkriteria' => $this->input->post('subkriteria'),
            'utility' => $this->input->post('utility')
        );
        $skriteria = $this->input->post('gkriteria');
        $this->M_kriteriaGabah->ResetAISub();
        $this->M_kriteriaGabah->InsertSKgabah('tb_subkriteriagabah', $data);
        $this->session->set_flashdata('message','Data Subkriteria '.$data['subkriteria'].' Berhasil di Tambahkan pada kriteria '.$skriteria.'');
        redirect(base_url()).'admin/C_kriteriaGabah';
    }
}

```

```
public function deleteSKgabah($idSkriteria){
    $id_skgabah = array('id_subkriteria' => $idSkriteria);
    $proses = $this->M_kriteriaGabah->DeleteSKgabah('tb_subkriteriagabah', $id_skgabah);
    if ($proses) {
        $this->session->set_flashdata('message1','Data Subkriteria Gabah Berhasil di Hapus');
        redirect(base_url()).'admin/C_kriteriaGabah';
    }else{
        $this->session->set_flashdata('message2','Data Subkriteria Gagal di Hapus');
        redirect(base_url()).'admin/C_kriteriaGabah';
    }
}
}
public function detailKgabah($idkriteria){
    $result = @$this->M_kriteriaGabah->Get_detailKgabah('tb_subkriteriagabah',$idkriteria);
    echo json_encode($result);
}
}
```

Gambar 4.9 Kode Program Kelas *C_kriteriaGabah*

Pada gambar 4.9 merupakan kelas *C_kriteriaGabah*, dalam kelas tersebut terdapat beberapa fungsi yang bekerja untuk mengitung normalisasi pembobotan, fungsi tersebut adalah fungsi *insertKgabah()*, fungsi *insertKgabah()* mempunyai alur proses pertama kali yaitu mengecek apakah data yang diinputkan lebih dari 100 dan kurang dari 0, jika tidak maka yang dilakukan selanjutnya adalah memasukkan data ke dalam *array* dan memanggil beberapa fungsi yang ada pada kelas model yaitu *insertKgabah()*, *Totalbobot()*, dan *ResetAI()* jika olah data telah selesai selanjutnya menampilkan pesan berhasil dan menuju halaman yang ditentukan dengan sintak *redirect*. Fungsi lain seperti *delete()* berfungsi untuk menghapus data kriteria, fungsi *ajax_getKtambah(\$id_kriteria)* berfungsi untuk menambahkan kriteria, fungsi *insertSKgabah()* berfungsi menambahkan subkriteria gabah dan juga nilai *utility*, fungsi *deleteSKgabah(\$idSkriteria)* berfungsi untuk menghapus subkriteria gabah sesuai dengan id subkriteria, dan yang terakhir fungsi *detailKgabah(\$idkriteria)* berfungsi untuk menampilkan detail subkriteria gabah yang telah dimasukkan.

2. Kelas *M_kriteriaGabah*

Kelas *M_kriteriaGabah* merupakan kelas yang berfungsi sebagai logika perhitungan dan perantara interaksi antara *database* dan sistem. Kelas model merupakan anak kelas dari *CI_Model* sehingga beberapa fungsi dari kelas *M_kriteriaGabah* mewarisi kelas dari *CI_Model*. Kelas tersebut berada pada folder

model yang merupakan folder *default* dari *framework Codeigniter*. Penulisan kode program pada kelas *M_kriteriaGabah* dapat dilihat pada gambar 4.10.

```

<?php if (!defined('BASEPATH')) exit('No direct script access allowed');

Class M_kriteriaGabah extends CI_Model {

    public function __construct()
    {
        parent::__construct();
    }
    public function get_dataKgabah()
    {
        $result = $this->db->query("SELECT * from tb_kriteriagabah ");
        return $result->result_array();
    }
    public function InsertKgabah($table,$data){
        $result = $this->db->insert($table, $data); // Kode ini digunakan untuk memasukan record baru kedalam sebuah tabel
        return $result; // Kode ini digunakan untuk mengembalikan hasil $res
    }

    public function Totalbobot(){
        // Mengambil nilai total bobot
        $query = $this->db->query("SELECT sum(nilai) as total FROM tb_kriteriagabah");
        foreach($query->result_array() AS $row) {
            $total = $row['total'];
        }
        // Generate Normalisasi dan bobot
        $selectkriteria = $this->db->query("SELECT * FROM tb_kriteriagabah order by id_kriteria");
        foreach($selectkriteria->result_array() as $row) {
            $nilai = $row['nilai'];
            $hitungbobot = ($nilai / $total)*100;
            $hitung = number_format($hitungbobot,16);
            $normalisasibobot = $hitung / 100;
            $normalisasi = number_format($normalisasibobot,16);
            $this->UpdateBobot($hitung,$nilai);
            $this->UpdateNormalisasi($normalisasi,$nilai);
        }
        return $nilai;
    }

    public function UpdateBobot($data, $where){
        $result = $this->db->query("UPDATE tb_kriteriagabah SET bobot = $data where nilai = $where");
        return $result;
    }

    public function UpdateNormalisasi($data, $where){
        $result = $this->db->query("UPDATE tb_kriteriagabah SET normalisasi = $data where nilai = $where");
        return $result;
    }

    public function Update($table, $data, $where){
        $res = $this->db->update($table, $data, $where); // Kode ini digunakan untuk merubah record yang sudah ada dalam sebuah tabel
        return $res;
    }
    public function Delete($table, $where){
        $res = $this->db->delete($table, $where); // Kode ini digunakan untuk menghapus record yang sudah ada
        return $res;
    }
}

```

```

// SUBKRITERIA

public function Get_detailKgabah($table, $id){
    $this->db->from($table);
    $this->db->where('id_kriteria',$id);
    $query = $this->db->get();
    return $query->result_array();
}

public function Get_KgabahSub($table, $id){
    $this->db->from($table);
    $this->db->where('id_kriteria',$id);
    $query = $this->db->get();
    return $query->row();
}

public function InsertSKgabah($table,$data){
    $result = $this->db->insert($table, $data);
    return $result;
}

public function DeleteSKgabah($table, $where){
    $res = $this->db->delete($table, $where); // Kode ini digunakan untuk menghapus record yang sudah ada
    if ($res) {
        return $res;
    }else{
        return false;
    }
}

public function ResetAI(){
    $res = $this->db->query("ALTER TABLE tb_kriteriagabah AUTO_INCREMENT = 1");
    return $res;
}

public function ResetAISub(){
    $res = $this->db->query("ALTER TABLE tb_subkriteriagabah AUTO_INCREMENT = 1");
    return $res;
}
}

```

Gambar 4.10 Kode Program Kelas *M_kriteriaGabah*

Kode program pada gambar 4.10 merupakan kode program yang digunakan untuk perhitungan normalisasi dan interaksi antara *database*. Perhitungan normalisasi dan pembobotan berada pada fungsi *Totalbobot()*. Pada Fungsi tersebut proses pertama adalah mengambil jumlah dari nilai semua kriteria dan memasukkan pada array, proses selanjutnya adalah proses perulangan untuk menghitung nilai pembobotan dan normalisasi setiap kriteria sampai kriteria habis. Proses selanjutnya adalah mengubah nilai bobot dan normalisasi *default* yang ada pada *database* dengan nilai yang telah dihitung sebelumnya, dengan memanggil fungsi *UpdateBobot()*, dan *UpdateNormalisasi()*. Fungsi lainnya pada kelas *M_kriteriaGabah* digunakan untuk menjalankan program terkait mengelola normalisasi, pembobotan kriteria dan subkriteria gabah.

3. Kelas *C_penilaianGabah*

Kelas *C_penilaianGabah* merupakan kelas yang berfungsi sebagai penghubung antara kelas model dan tampilan. Dalam kelas tersebut terdapat fungsi yang mengimplementasikan perhitungan metode SMART. Setiap proses penilaian penentuan rangking, mutu dan harga pemasok gabah, kelas *C_penilaianGabah* akan dipanggil untuk mengatur jalannya data antara database, tampilan, dan logika perhitungan. Kelas tersebut merupakan melakukan banyak berinteraksi dengan fungsi pada kelasnya sendiri, maupun dengan kelas lainnya. Hal ini yang menyebabkan kelas *C_penilaianGabah* cukup kompleks, begitu juga untuk kelas *controller* penilaian beras yang relatif sama dengan kelas berikut. Kelas ini juga berfungsi menampilkan tampilan perhitungan secara dinamis jika suatu saat kriteria dan subkriteria yang dinilai bertambah. Penulisan kode program pada kelas *C_penilaianGabah* dapat dilihat pada gambar 4.11 berikut.

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class C_penilaianGabah extends CI_Controller {
    public function __construct(){
        parent::__construct();
        $this->load->library('session');
        $this->load->helper('url');
        if($this->session->userdata('logged_in') == '1' && $this->session->userdata('level') == '0')
        {
            $this->load->model('M_penilaianGabah');
        }else{
            redirect(base_url()).'C_login';
        }
    }
    public function indexPenilaian($id){
        $data['namapemasok'] = $this->M_penilaianGabah->get_namaPemasok($id);
        $data['kgabah'] = $this->M_penilaianGabah->get_kriteriaGabah();
        $data['ugabah'] = $this->M_penilaianGabah->get_utilityGabah();
        $this->load->view('user/penilaianGabah',$data);
    }
    public function perhitunganSMARTgabah(){
        $totalkriteria = $this->M_penilaianGabah->get_totalKgabah();
        $tot = 0;
        for ($i=1; $i <= $totalkriteria; $i++) {
            $normalisasi = $this->M_penilaianGabah->normalisasi($i);
            $utility = $this->input->post($i);
            $nsementara = number_format($normalisasi,16) * $utility;
            $tot += number_format($nsementara,16);
            $stotsmart = number_format($tot,3);
            $insert = array(
                'id_pemasok' => $this->input->post('idPgabah'),
                'id_kriteria' => $i,
                'nsementara' => $nsementara,
                'utility' => $utility
            );
            $this->M_penilaianGabah->penilaiangabah('tb_penilaiangabah',$insert);
        }
    }
}
```

```

$permutu = $this->perhitunganMutugabah();
$smutu = $this->mutugabah();
$harga = $this->hargagabah();;
$update = array(
    'penilaianSMART' => $stosmart,
    'penilaianmutu' => $permutu,
    'mutu' => $smutu,
    'harga' => $harga
);
$where = array(
    'id_pemasokg' => $this->input->post('idPgabah')
);
$this->M_penilaianGabah->updatepenilaian('tb_pemasokgabah',$update,$where);
$this->session->set_flashdata('message','Penilaian Terhadap Pemasok Gabah berhasil!');
redirect(base_url().'user/C_pemasokGabah');
}

public function perhitunganMutugabah(){
    $totalkriteria = $this->M_penilaianGabah->get_totalKgabah();
    $stot = 0;
    for ($i=1; $i <= $totalkriteria; $i++) {
        if ($i == 4 || $i == 5 || $i == 6) {
            $normalisasi == 0;
            $utility == 0;
        }else{
            $normalisasi = $this->M_penilaianGabah->normalisasi($i);
            $utility = $this->input->post($i);
            $nsementara = number_format($normalisasi,16) * $utility;
            $stot += number_format($nsementara,16);
            $smutu = number_format($stot,3);
        }
    }
    return $smutu;
}

public function nilaimutugabah(){
    $mutu = @$this->perhitunganMutugabah();
    echo json_encode($mutu);
}

public function detailpengabah($id){
    $result = @$this->M_penilaianGabah->Get_detailpengabah($id);
    echo json_encode($result);
}

public function mutugabah(){
    $nilaimutu = $this->perhitunganMutugabah();
    $mutu = $this->M_penilaianGabah->get_nilaimutugabah($nilaimutu);
    return $mutu;
}

public function hargagabah(){
    $mutug = $this->mutugabah();
    $harga = $this->M_penilaianGabah->get_harga($mutug);
    return $harga;
}
}

```

Gambar 4.11 Kode Program Kelas *C_PenilaianGabah*

Gambar 4.11 menunjukkan kode program kelas Controller dimana pada fungsi *perhitunganSMARTgabah()* berisi proses perhitungan metode SMART, penentuan mutu dan harga gabah dengan. Perhitungan SMART dituliskan dalam kode program menggunakan perulangan untuk mentotal jumlah perkalian antara normalisasi dan

utility. Penentuan mutu dan harga gabah di proses dengan menggunakan fungsi perhitunganMutugabah() sebagai fungsi yang melakukan perhitungan mutu gabah, untuk penentuan mutu gabah di proses menggunakan kelas *mutugabah()*, dan harga gabah diproses dengan fungsi *hargagabah()*. Setelah proses perhitungan dan klasifikasi mutu maka controller memanggil fungsi pada model untuk memasukkan data yang telah di proses. Fungsi lain digunakan untuk mendukung proses penilaian setiap pemasok gabah.

4. Kelas *M_penilaianGabah*

Kelas *M_penilaianGabah* merupakan kelas yang digunakan untuk interaksi data dengan *database* terkait proses penilaian gabah yang telah dilakukan. Kode program kelas *M_penilaianGabah* dapat dilihat pada gambar 4.12

```
<?php if (!defined('BASEPATH')) exit('No direct script access allowed');

Class M_penilaianGabah extends CI_Model {

    public function __construct()
    {
        parent::__construct();
    }
    public function get_namaPemasok($id_pg)
    {
        $result = $this->db->query("SELECT id_pemasokg,nama from tb_pemasokgabah where id_pemasokg = '$id_pg'");
        return $result->result_array();
    }
    public function get_totalKgabah()
    {
        $query = $this->db->query("SELECT COUNT(kriteria) as totalKgabah from tb_kriteriagabah");
        foreach($query->result_array() AS $rowt) {
            $total = $rowt['totalKgabah'];
            return $total;
        }
    }
    public function get_kriteriaGabah($id)
    {
        $result = $this->db->query(" SELECT * from tb_kriteriagabah where id_kriteria = $id order by id_kriteria");
        return $result->result_array();
    }
    public function get_utilityGabah($id)
    {
        $result = $this->db->query(" SELECT * from tb_subkriteriagabah where id_kriteria = $id order by subkriteria");
        return $result->result_array();
    }
    public function normalisasi($id){
        $query = $this->db->query("SELECT normalisasi as nmr from tb_kriteriagabah where id_kriteria = $id");
        foreach($query->result_array() AS $rowt) {
            $normalisasi = $rowt['nmr'];
        }
        return $normalisasi;
    }
}
```

```

public function updatepenilaian($table, $data, $where){
    $res = $this->db->update($table, $data, $where);
    return $res;
}

public function Get_detailpengabah($id){
    $result = $this->db->query(" SELECT kriteria, normalisasi, utility, nsementara from tb_penilaiangabah tp join tb_kriteriagabah tk on
tp.id_kriteria = tk.id_kriteria where tp.id_pemasokg = $id");
    return $result->result_array();
}

public function get_datamMHgabah()
{
    $result = $this->db->query("SELECT * from tb_mutugabah ");
    return $result->result_array();
}

public function get_mutuGabah($id){
    $result = $this->db->query(" SELECT mutugabah from tb_mutugabah where id_mutugabah = $id");
    return $result->row();
}

public function nilaiM1(){
    $query = $this->db->query("SELECT nilaimutu as nm1 from tb_mutugabah where id_mutugabah = 1");
    foreach($query->result_array() AS $rowt) {
        $nm1 = $rowt['nm1'];
    }
    return $nm1;
}

public function nilaiM2(){
    $query = $this->db->query("SELECT nilaimutu as nm2 from tb_mutugabah where id_mutugabah = 2");
    foreach($query->result_array() AS $rowt) {
        $nm2 = $rowt['nm2'];
    }
    return $nm2;
}

public function nilaiM3(){
    $query = $this->db->query("SELECT nilaimutu as nm3 from tb_mutugabah where id_mutugabah = 3");
    foreach($query->result_array() AS $rowt) {
        $nm3 = $rowt['nm3'];
    }
    return $nm3;
}

public function get_nilaimutugabah($nilaimutu){
    $mutu = "";
    $nm1 = $this->nilaiM1();
    $nm2 = $this->nilaiM2();
    $nm3 = $this->nilaiM3();
    if ($nilaimutu <= $nm1 && $nilaimutu > $nm2) {
        $mutu = 'mutu 1';
    }else if($nilaimutu <= $nm2 && $nilaimutu > $nm3){
        $mutu = 'mutu 2';
    }else if($nilaimutu == $nm3){
        $mutu = 'mutu 3';
    }else if($nilaimutu <= $nm3){
        $mutu = 'mutu rendah';
    }
    return $mutu;
}

public function get_harga($mutu){
    $query = $this->db->query("SELECT hargagabah as hrg from tb_mutugabah where mutugabah = '$mutu'");
    foreach($query->result_array() AS $rowt) {
        $harga = $rowt['hrg'];
    }
    return $harga;
}
}
}

```

Gambar 4.12 Kode Program Kelas M_penilaianGabah

Gambar 4.12 menjelaskan kelas dengan beberapa fungsi yang berfungsi menginteraksikan data dengan *database*. Proses penentuan mutu gabah dijelaskan pada fungsi *nilaimutugabah(\$nilaimutu)*. Pada fungsi tersebut parameter *\$nilaimutu* menjadi pembanding dengan range nilai yang telah ditetapkan sebelumnya pada *database*. Proses selanjutnya adalah penentuan mutu gabah dengan melewati beberapa kondisi hingga mutu ditentukan dan nilai mutu dikembalikan, penentuan harga ditentukan pada fungsi *get_harga(\$mutu)* dengan parameter mutu yang telah ditetapkan sebelumnya. Fungsi lain digunakan untuk proses memasukkan, mengubah, dan mengambil data dari sistem ke *database*, begitupun sebaliknya.

4.4 Pengujian Sistem

Pengujian sistem merupakan alur tahapan metode *waterfall* setelah melakukan kode program. Pengujian atau *testing* adalah proses menganalisa perangkat lunak untuk mendeteksi perbedaan antara kondisi yang ada dengan kondisi yang diinginkan sehingga ditemukan *error* atau *bug* yang dapat dijadikan sebagai acuan evaluasi fitur – fitur pada perangkat lunak (Pandara dan , Stanley Karouw, 2014). Proses pengujian dilakukan dengan pengujian *blackbox* dan *whitebox*.

4.4.1 Pengujian White Box

Pengujian *white box* pada sistem ini digambarkan dengan diagram alir *Cyclomatic Complexity* (CC) dan pembuatan *test case*. Pembuatan diagram tersebut bertujuan untuk mengevaluasi alur kerja dari setiap fitur yang dikembangkan. Pengujian *listing program* dilakukan untuk setiap fungsi yang mengimplementasikan metode SMART. Pengujian langkah pertama dalam metode smart adalah penentuan bobot kriteria dan nilai *utility*. Fungsi yang berfungsi sebagai perhitungan kriteria gabah terdapat pada kelas *C_kriteriaGabah* dan *M_kriteriaGabah*, sedangkan untuk

fungsi perhitungan SMART terdapat pada kelas *C_penilaianGabah* dan *M_penilaianGabah*.

a. Fungsi *insertKgabah* pada kelas *C_kriteriaGabah*

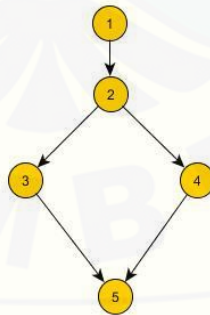
Fungsi yang berfungsi sebagai perhitungan pembobotan kriteria gabah adalah fungsi *insertKgabah()*. Proses listing program pada fungsi tersebut dapat dilihat pada gambar 4. 15, diagram alir pada gambar 4.16, dan *test case* pada tabel 4.5

```

public function insertKgabah(){
    $cekbobot = $this->input->post('nilai'); } 1
    if ($cekbobot >= 0 && $cekbobot <= 100) { } 2
        $data = array(
            'kriteria' => $this->input->post('kriteria'),
            'nilai' => $this->input->post('nilai')
        );
        $this->M_kriteriaGabah->InsertKgabah('tb_kriteriaGabah', $data);
        $this->M_kriteriaGabah->Totalbobot();
        $this->M_kriteriaGabah->ResetAI();
        $this->session->set_flashdata('message','Data Kriteria Gabah Berhasil di Tambahkan');
        redirect(base_url().'admin/C_kriteriaGabah');
    }else{
        $this->session->set_flashdata('message1','bobot harus dalam range 0 - 100');
        redirect(base_url().'admin/C_kriteriaGabah');
    }
}

```

Gambar 4.13 Listing Program Fungsi *insertKgabah*



Gambar 4.14 Diagram Alir Fungsi *insertKgabah*

Pehitungan jalur pada diagram alir gambar 4.14 adalah sebagai berikut

$$CC = 1 + 1$$

$$CC = 2$$

Jalur 1 = 1, 2, 3, 5

Jalur 2 = 1, 2, 4, 5

Pengujian *test case* fungsi *insertKgabah* dapat dilihat pada tabel 4.5

Tabel 4.5 Test Case Fungsi *insertKgabah*

Jalur 1	
<i>Test Case</i>	Menghitung nilai kriteria setelah dinormalisasi dengan memasukan nama kriteria dan bobot setiap kriteria diantara 0 – 100, jika masukan benar maka sistem akan menghitung nilai normalisasi dan bobot setiap kriteria yang dimasukan,
Target yang diharapkan	Sistem menghitung normalisasi dan pembobotan setiap kriteria, memasukkan data hasil perhitungan pada database dan menampilkan hasil perhitungan pada tabel sistem.
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 3 – 5
Jalur 2	
<i>Test Case</i>	Menghitung nilai kriteria setelah dinormalisasi dengan memasukan nama kriteria dan bobot setiap kriteria diantara 0 – 100, jika inputan melebihi batasan yang ditetapkan maka sistem akan menampilkan pesan peringatan.
Target yang diharapkan	Menampilkan pesan peringatan jika data yang dimasukkan harus dalam <i>range</i> yang disediakan.
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 4 – 5

b. Kelas *M_kriteriaGabah*

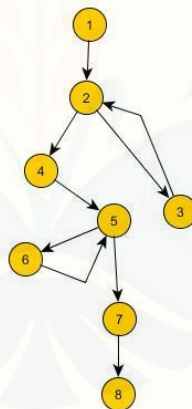
Logika perhitungan pembobotan dan normalisasi terdapat pada kelas *M_kriteriaGabah* pada fungsi *Totalbobot()*. Listing program dapat dilihat pada gambar 4.15, diagram alir pada gambar 4.16, dan test case pada tabel 4.6

```

public function Totalbobot()
{
    // Mengambil nilai total bobot
    $query = $this->db->query("SELECT sum(nilai) as total FROM tb_kriteriagabah"); } 1
    foreach($query->result_array() AS $row) { } 2
    {
        $total = $row['total']; } 3
    }

    // Generate Normalisasi dan bobot
    $selectkriteria = $this->db->query("SELECT * FROM tb_kriteriagabah order by id_kriteria"); } 4
    foreach($selectkriteria->result_array() as $row) { } 5
    {
        $nilai = $row['nilai'];
        $hitungbobot = ($nilai / $total)*100;
        $hitung = number_format($hitungbobot,16);
        $normalisasibobot = $hitung / 100;
        $normalisasi = number_format($normalisasibobot,16); } 6
        $this->UpdateBobot($hitung,$nilai);
        $this->UpdateNormalisasi($normalisasi,$nilai);
    }
    return $nilai; } 7
}

```

Gambar 4.15 Listing Program Fungsi *Totalbobot*Gambar 4.16 Diagram Alir Fungsi *TotalBobot*

Perhitungan Jalur pada diagram alir 4.16 adalah sebagai berikut

$$CC = 2 + 1$$

$$CC = 3$$

$$\text{Jalur 1} = 1, 2, 3, 2, 4, 5, 6, 5, 7, 8$$

$$\text{Jalur 2} = 1, 2, 4, 5, 6, 5, 7, 8$$

$$\text{Jalur 3} = 1, 2, 4, 5, 7, 8$$

Tabel 4.6 *Test Case* Fungsi *Totalbobot*

Jalur 1	
<i>Test Case</i>	Mengambil jumlah total data nilai yang ada pada <i>database</i> dengan perintah perulangan <i>foreach</i> sampai data ditemukan, dan memasukkan data berupa array pada sebuah variabel, lalu mengambil kembali data nilai pada <i>database</i> dengan perintah perulangan <i>foreach</i> , dan memasukkan data berupa array pada sebuah variabel dan menghitung normalisasi dan juga pembobotan sesuai dengan menggunakan perulangan <i>foreach</i> yang menghitung secara berulang hingga kriteria yang ada pada <i>database</i> habis.
Target yang diharapkan	Sistem mengambil data total nilai dan menghitung normalisasi serta pembobotan setiap kriteria secara berulang, dan mengembalikan variabel nilai
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 3 – 2 – 4 – 5 – 6 – 5 – 7 – 8
Jalur 2	
<i>Test Case</i>	Mengambil data nilai pada <i>database</i> dengan perintah perulangan <i>foreach</i> , dan memasukkan data berupa array pada sebuah variabel, juga menghitung normalisasi dan pembobotan secara berulang.
Target yang diharapkan	Data yang telah diambil disimpan pada variabel nilai, dan menghitung normalisasi dan pembobotan secara berulang, lalu mengembalikan variabel nilai
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 4 – 5 – 6 – 5 – 7 – 8
Jalur 3	
<i>Test Case</i>	Perulangan pengambilan total nilai dan perhitungan normalisasi dan pembobotan selesai, sehingga variabel nilai dikembalikan
Target yang diharapkan	Mengembalikan variabel nilai.
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 4 – 5 – 7 – 8

c. Kelas *C_penilaianGabah*

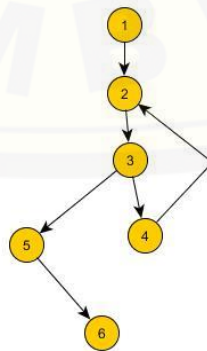
Pada kelas *C_penilaianGabah* logika perhitungan metode SMART berada pada fungsi *perhitunganSMARTgabah()*. Listing program fungsi tersebut ditunjukkan pada gambar 4.17, diagram alir pada gambar 4.18, dan test case pada tabel 4.7.

```

public function perhitunganSMARTgabah()
{
    $totalkriteria = $this->M_penilaianGabah->get_totalKgabah(); }-1
    $stot = 0;
    for ($si=1; $si <= $totalkriteria; $si++) { }-2 }-4
    {
        $normalisasi = $this->M_penilaianGabah->normalisasi($si);
        $utility = $this->input->post($si);
        $nsementara = number_format($normalisasi,16) * $utility;
        $stot += number_format($nsementara,16);
        $stotsmart = number_format($stot,13);
        $insert = array(
            'id_pemasokg' => $this->input->post('idPgabah'),
            'id_kriteria' => $si,
            'nsementara' => $nsementara,
            'utility' => $utility
        );
        $this->M_penilaianGabah->penilaiangabah('tb_penilaiangabah',$insert); }-3
    }
    $spermutu = $this->perhitunganMutugabah();
    $smutu = $this->mutugabah();
    $sharga = $this->hargagabah();
    $update = array(
        'penilaianSMART' => $stotsmart,
        'penilaianmutu' => $spermutu,
        'mutu' => $smutu,
        'harga' => $sharga
    );
    $where = array(
        'id_pemasokg' => $this->input->post('idPgabah')
    );
    $this->M_penilaianGabah->updatepenilaian('tb_pemasokgabah',$update,$where);
    $this->session->set_flashdata('message','Penilaian Terhadap Pemasok Gabah berhasil');
    redirect(base_url(). 'user/C_pemasokGabah');
} }-5
}

```

Gambar 4.17 Listing Program Fungsi *perhitunganSMARTgabah*



Gambar 4.18 Diagram Alir Fungsi *perhitunganSMARTgabah*

Perhitungan Jalur pada diagram alir 4.18 adalah sebagai berikut

$$CC = 1 + 1$$

$$CC = 2$$

$$\text{Jalur 1} = 1, 2, 3, 4, 2, 3, 5, 6$$

$$\text{Jalur 2} = 1, 2, 3, 5, 6$$

Tabel 4.7 *Test Case* Fungsi *perhitunganSMARTgabah*

Jalur 1	
<i>Test Case</i>	Menghitung nilai metode SMART dengan menggunakan perulangan sebanyak total kriteria dan menetapkan mutu gabah serta harga gabah.
Target yang diharapkan	Menghitung dengan perhitungan SMART dan menetapkan mutu serta harga gabah sesuai dengan banyak kriteria.
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 3 – 4 – 2 – 3 – 5 – 6
Jalur 2	
<i>Test Case</i>	Memberikan penetapan mutu dan harga gabah sesuai dengan perhitungan SMART terkait mutu gabah.
Target yang diharapkan	Merubah mutu dan harga pada database dengan parameter nilai mutu yang telah dihitung.
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 3 – 5 – 6

d. Kelas *C_penilaianGabah*

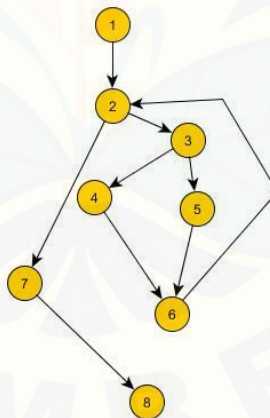
Perhitungan mutu pada kelas *C_penilaianGabah* terletak pada fungsi *perhitunganMutugabah()*. Fungsi tersebut berfungsi menghitung nilai SMART dengan pengecualian beberapa kriteria yang tidak ditetapkan dalam SNI gabah Indonesia. Listing program fungsi tersebut ditunjukkan pada gambar 4.17, diagram alir pada gambar 4.18, dan test case pada tabel 4.8.

```

public function perhitunganMutugabah(){
    $totalKriteria = $this->M_penilaianGabah->get_totalKgabah(); } 1
    $tot = 0;
    for ($i=1; $i <= $totalKriteria; $i++){ } 2 } 6
        if ($i == 4 || $i == 5 || $i == 6) { } 3
            $normalisasi = 0; } 4
            $utility = 0;
        }else{
            $normalisasi = $this->M_penilaianGabah->normalisasi($i);
            $utility = $this->input->post($i);
            $sementara = number_format($normalisasi,16) * $utility; } 5
            $tot += number_format($sementara,16);
            $mutu = number_format($tot,13);
        }
    }
    return $mutu; } 7
}

```

Gambar 4.19 Listing Program fungsi *perhitunganMutugabah*



Gambar 4.20 Diagram Alir Fungsi *perhitunganMutugabah*

Perhitungan Jalur pada diagram alir 4.20 adalah sebagai berikut

$$CC = 2 + 1$$

$$CC = 2$$

Jalur 1 = 1, 2, 3, 5, 6, 2, 7, 8

Jalur 2 = 1, 2, 3, 4, 6, 2, 7, 8

Tabel 4.8 *Test Case* Fungsi perhitungan Mutu gabah

Jalur 1	
<i>Test Case</i>	Menghitung nilai mutu gabah dengan metode SMART dengan menggunakan perulangan sesuai dengan banyaknya kriteria yang telah diambil pada database dan mengembalikan nilai pada variabel mutu
Target yang diharapkan	Menghitung dengan perulangan nilai mutu menggunakan metode SMART dan mengembalikan nilai variabel mutu
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 3 – 5 – 6 – 2 – 7 – 8
Jalur 2	
<i>Test Case</i>	Menghitung nilai mutu gabah dengan metode SMART melalui perulangan sesuai banyaknya kriteria yang ditetapkan namun terdapat kondisi dimana kriteria tertentu tidak dihitung dan menjadikan nilai normalisasi dan <i>utility</i> menjadi nol karena mengikuti ketetapan SNI gabah.
Target yang diharapkan	Merubah nilai normalisasi dan <i>utility</i> menjadi 0 pada kriteria yang tidak masuk standarisasi mutu
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 3 – 4 – 6 – 2 – 7 – 8

e. Kelas *M_penilaianGabah*

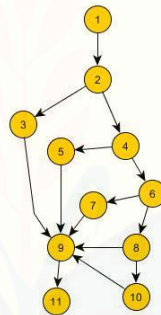
Kelas *M_penilaianGabah* memiliki fungsi *get_nilaimutugabah* dengan parameter nilai mutu yang telah didapatkan sebelumnya melalui perhitungan metode SMART. Fungsi tersebut berfungsi untuk menentukan mutu pada setiap gabah yang telah dinilai, terdapat empat kondisi pada fungsi tersebut untuk penentuan mutu gabah. Proses listing program dapat dilihat pada gambar 4.21, diagram alir fungsi pada gambar 4.22 dan test case pada tabel 4.9 sebagai acuan testing program apakah fitur yang telah dikembangkan sesuai dengan apa yang diinginkan.

```

public function get_nilaimutugabah($nilaimutu){
    $mutu = '';
    $nm1 = $this->nilaiM1();
    $nm2 = $this->nilaiM2();
    $nm3 = $this->nilaiM3();
    if ($nilaimutu >= $nm1) {
        $mutu = 'mutu 1';
    }
    }else if($nilaimutu >= $nm2 && $nilaimutu < $nm1){
        $mutu = 'mutu 2';
    }
    }else if($nilaimutu >= $nm3 && < $nm2){
        $mutu = 'mutu 3';
    }
    }else if($nilaimutu < $nm3){
        $mutu = 'mutu rendah';
    }
    return $mutu;
}

```

Gambar 4.21 Listing Program Fungsi *get_nilaimutugabah*



Gambar 4.22 Diagram Alir Fungsi *get_nilaimutugabah*

Perhitungan Jalur pada diagram alir 4.20 adalah sebagai berikut

$$CC = 4 + 1$$

$$CC = 5$$

$$\text{Jalur 1} = 1, 2, 3, 9, 11$$

$$\text{Jalur 2} = 1, 2, 4, 5, 9, 11$$

$$\text{Jalur 3} = 1, 2, 4, 6, 7, 9, 11$$

$$\text{Jalur 4} = 1, 2, 4, 6, 8, 9, 11$$

$$\text{Jalur 5} = 1, 2, 4, 6, 8, 10, 9, 11$$

Tabel 4.9 Test Case Fungsi get_nilaimutugabah

Jalur 1	
<i>Test Case</i>	Mengambil nilai setiap mutu pada database dan mengecek apakah nilai mutu lebih dari sama dengan range nilai mutu 1 , jika iya maka mengisi variabel mutu dengan nilai mutu 1. Dan mengembalikan variabel mutu.
Target yang diharapkan	Jika kondisi terpenuhi maka mengisi variabel mutu menjadi mutu 1, dan jika tidak terpenuhi menuju kondisi selanjutnya
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 3 – 9 – 11
Jalur 2	
<i>Test Case</i>	Mengambil nilai setiap mutu pada database dan mengecek apakah nilai mutu lebih dari sama dengan range nilai mutu 2 dan kurang dari range nilai mutu 1, jika iya maka mengisi variabel mutu dengan nilai mutu 2. Dan mengembalikan variabel mutu.
Target yang diharapkan	Jika kondisi terpenuhi maka mengisi variabel mutu menjadi mutu 2, dan jika tidak terpenuhi menuju kondisi selanjutnya
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 4 – 5 – 9 – 11
Jalur 3	
<i>Test Case</i>	Mengambil nilai setiap mutu pada database dan mengecek apakah nilai mutu lebih dari sama dengan range nilai mutu 3 dan kurang dari nilai mutu 2, jika iya maka mengisi variabel mutu dengan nilai mutu 3. Dan mengembalikan variabel mutu.
Target yang diharapkan	Jika kondisi terpenuhi maka mengisi variabel mutu menjadi mutu 3, dan jika tidak terpenuhi menuju kondisi selanjutnya
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 4 – 6 – 9 – 7 – 11
Jalur 4	
<i>Test Case</i>	Mengambil nilai setiap mutu pada database dan mengecek apakah nilai mutu kurang dari range nilai mutu 3, jika tidak maka mengembalikan variabel mutu..
Target yang diharapkan	Mengembalikan nilai mutu jika nilai mutu yang diambil tidak memenuhi kondisi kurang dari nilai mutu 3.
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 4 – 6 – 8 – 9 – 11
Jalur 5	
<i>Test Case</i>	Mengambil nilai setiap mutu pada database dan mengecek apakah nilai mutu kurang dari range nilai mutu 3, jika iya maka mengisi variabel mutu dengan nilai mutu rendah dan mengembalikan variabel mutu..

Target yang diharapkan	Jika kondisi terpenuhi maka mengisi variabel mutu menjadi mutu rendah, dan jika tidak terpenuhi menuju maka mengembalikan nilai variabel mutu.
Hasil Pengujian	Benar
Path/Jalur	1 – 2 – 4 – 6 – 8 – 10 – 9 – 11

4.4.2 Pengujian Black Box

Pengujian Black Box dilakukan untuk menguji apakah program sudah menjalankan fitur dengan benar atau tidak. Pengujian tersebut mencakup fungsionalitas program seperti memasukan data, mengubah, dan keluaran yang diharapkan dari sistem jika menjalankan fitur yang ada. Tabel pengujian *black box* dapat dilihat pada tabel 4.9 berikut.

Tabel 4.10 Pengujian *Blackbox* Fitur SPK Pembelian Padi Terbaik

No	Fitur	Aksi	Hasil	Ket.
1	Masuk sistem (<i>Login</i>)	Mengisi form username dan password lalu klik tombol login	Login berhasil dan masuk ke dalam sistem sesuai hak akses pengguna	Berhasil
		Klik tombol login namun form username atau password kosong	Menampilkan span “data tidak boleh kosong”	Berhasil
		Klik tombol login namun username atau password salah	menampilkan flash alert “Gagal Login, username/password salah”	Berhasil
2.	Mengelola data pengguna	Klik menu pengguna, lalu klik submenu kelola hak akses	Menampilkan halaman kelola hak akses dan tabel pengguna, tombol tambah ubah, dan hapus	Berhasil
		Klik tombol tambah data pengguna, lalu klik daftar	Menampilkan modal tambah pengguna, form isian, dan tombol daftar. menampilkan flash alert data pengguna berhasil diinputkan	Berhasil
		Klik tombol daftar namun terdapat username pengguna yang sama.	Menampilkan flash alert “Username Telah Digunakan”	Berhasil
Dilanjutkan				

Lanjutan				
No	Fitur	Aksi	Hasil	Ket.
		Klik tombol daftar namun form isian kosong	Menampilkan span “data tidak boleh kosong” pada seetiap form yang kosong	Berhasil
		Klik tombol mengubah data	Menampilkan modal form ubah data beserta data yang akan diubah	Berhasil
		Klik tombol simpan pada modal ubah data	Menyimpan pada database dan menampilkan flash alert “Data Pengguna Berhasil Diubah”	Berhasil
		Klik tombol hapus data	menampilkan <i>confirm message</i> “Hapus Data Pengguna ?”	Berhasil
		Klik tombol ok pada <i>confirm message</i> hapus data pengguna	Meghapus data pada database dan menampilkan flash alert “Data Pengguna Berhasil di Hapus”	Berhasil
3.	Mengelola kriteria gabah	Klik menu kriteria gabah, lalu submenu kelola bobot	Menampilkan tabel dengan data kriteria dan subkriteria beserta bobot dan normalisasi	Berhasil
		Klik tombol tambah kriteria	Menampilkan modal tambah kriteria dan tombol tambah	Berhasil
		Klik tombol tambah namun form ada yang kosong	Menampilkan span “data tidak boleh kosong	Berhasil
		Klik tombol tambah namun data pada form tidak sesuai	Menampilkan <i>flash alert</i> “bobot harus dalam range 0 - 100”	Berhasil
		Klik tombol tambah dan data telah benar	Menyimpan pada <i>database</i> dan Menampilkan flash alert “Data Kriteria Gabah Berhasil di Tambahkan”	Berhasil
		Klik tombol hapus kriteria	Menampilkan <i>confirm message</i> “Hapus kriteria ini ?”	Berhasil
		Klik tombol ok pada <i>confirm message</i> hapus kriteria	Menghapus data pada database, dan menampilkan flash alert “Data Kriteria Gabah Berhasil di Hapus”	Berhasil
		Klik tombol tambah subkriteria	Mnampilkan modal, form, dan tombol tambah	Berhasil
		Klik tombol tambah pada modal tambah subkriteria dengan form terisi dengan benar	Menyimpan pada database dan menampilkan flash alert “Data Subkriteria Berhasil di Tambahkan pada kriteria”	Berhasil
		Klik tombol tambah namun form ada yang kosong pada modal subkriteria	Menampilkan span “data tidak boleh kosong	Berhasil
		Klik detail subkriteria	Menampilkan modal dengan data detail subkriteria	Berhasil
Dilanjutkan				

Lanjutan				
No	Fitur	Aksi	Hasil	Ket.
		Klik tombol ok pada <i>confirm message</i> hapus subkriteria	Menghapus data pada database, dan menampilkan flash alert “Data Subkriteria Gabah Berhasil di Hapus”	Berhasil
4.	Mengelola kriteria beras	Klik menu kriteria beras, lalu submenu kelola bobot	Menampilkan tabel dengan data kriteria dan subkriteria beserta bobot dan normalisasi	Berhasil
		Klik tombol tambah kriteria	Menampilkan modal tambah kriteria dan tombol tambah	Berhasil
		Klik tombol tambah namun form ada yang kosong	Menampilkan span “data tidak boleh kosong	Berhasil
		Klik tombol tambah namun data pada form tidak sesuai	Menampilkan <i>flash alert</i> “bobot harus dalam range 0 - 100”	Berhasil
		Klik tombol tambah dan data telah benar	Menyimpan pada <i>database</i> dan Menampilkan flash alert “Data Kriteria Beras Berhasil di Tambahkan”	Berhasil
		Klik tombol hapus kriteria	Menampilkan <i>confirm message</i> “Hapus kriteria ini ?”	Berhasil
		Klik tombol ok pada <i>confirm message</i> hapus kriteria	Menghapus data pada database, dan menampilkan flash alert “Data Kriteria Beras Berhasil di Hapus”	Berhasil
		Klik tombol tambah subkriteria	Mnampilkan modal, form, dan tombol tambah	Berhasil
		Klik tombol tambah pada modal tambah subkriteria dengan form terisi dengan benar	Menyimpan pada database dan menampilkan flash alert “Data Subkriteria Berhasil di Tambahkan pada kriteria”	Berhasil
		Klik tombol tambah namun form ada yang kosong pada modal subkriteria	Menampilkan span “data tidak boleh kosong	Berhasil
		Klik detail subkriteria	Menampilkan modal dengan data detail subkriteria	Berhasil
		Klik tombol hapus subkriteria	Menampilkan <i>confirm message</i> “Hapus subkriteria ini ?”	Berhasil
		Klik tombol ok pada <i>confirm message</i> hapus subkriteria	Menghapus data pada database, dan menampilkan flash alert “Data Subkriteria Beras Berhasil di Hapus”	Berhasil
5.	Mengelola stok pembelian gabah	Klik menu stok padi, lalu submenu kelola stok gabah	Menampilkan halaman kelola stok padi beserta tabel, dan tombol tambah dan pengurangan pembelian stok	Berhasil
		Klik tombol tambah stok pembelian	Menampilakan form pada modal, serta tombol tambah	Berhasil
Dilanjutkan				

Lanjutan				
No	Fitur	Aksi	Hasil	Ket.
		Klik tombol tambah pada modal tambah stok pembelian	Menghitung dan menyimpan pada database	Berhasil
		Klik tombol tambah namun form kosong pada modal tambah stok gabah	Menampilkan span “data tidak boleh kosong	Berhasil
		Klik tombol pengurangan stok gabah	Menampilkan form pada modal, serta tombol tambah	Berhasil
		Klik tombol tambah pada modal pengurangan stok gabah	Menghitung, menyimpan pada database dan menampilkan flash alert “Pengurangan Pembelian Stok Gabah Berhasil”	Berhasil
		Klik tombol tambah pada modal pengurangan gabah namun data tidak sesuai	Menampilkan span “Pengurangan Pembelian Stok Gabah Tidak Boleh Melebihi Pembelian Stok”	Berhasil
6.	Mengelola stok pembelian beras	Klik menu stok padi, lalu submenu kelola stok beras	Menampilkan halaman kelola stok padi beserta tabel, dan tombol tambah dan pengurangan pembelian stok	Berhasil
		Klik tombol tambah stok pembelian	Menampilkan form pada modal, serta tombol tambah	Berhasil
		Klik tombol tambah pada modal tambah stok pembelian	Menghitung dan menyimpan pada database	Berhasil
		Klik tombol tambah namun form kosong pada modal tambah stok beras	Menampilkan span “data tidak boleh kosong	Berhasil
		Klik tombol pengurangan stok beras	Menampilkan form pada modal, serta tombol tambah	Berhasil
		Klik tombol tambah pada modal pengurangan stok beras	Menghitung, menyimpan pada database dan menampilkan flash alert “Pengurangan Pembelian Stok Beras Berhasil”	Berhasil
		Klik tombol tambah pada modal pengurangan beras namun data tidak sesuai	Menampilkan span “Pengurangan Pembelian Stok Beras Tidak Boleh Melebihi Pembelian Stok”	Berhasil
7.	Mengelola Penilaian mutu dan harga beras	Klik menu mutu dan harga lalu submenu penilaian beras	Menampilkan halaman penilaian beras dengan data mutu beras dan tombol penilaian juga reset.	Berhasil
		Klik tombol penilaian	Menampilkan modal mutu dan harga beras juga tombol hitung	Berhasil
		Klik tombol hitung dengan form terisi semua dengan benar	Menghitung, menyimpan data pada <i>database</i> , dan menghilangkan tombol hitung	Berhasil
Dilanjutkan				

Lanjutan				
No	Fitur	Aksi	Hasil	Ket.
		Klik tombol hitung namun data kosong	Menampilkan span “data tidak boleh kosong”	Berhasil
		Klik tombol reset data mutu dan harga	Mengatur ulang nilai mutu dan harga menjadi nol	Berhasil
8.	Mengelola Penilaian mutu dan harga gabah	Klik menu mutu dan harga lalu submenu penilaian gabah	Menampilkan halaman penilaian gabah dengan data mutu gabah dan tombol penilaian juga reset.	Berhasil
		Klik tombol penilaian	Menampilkan modal mutu dan harga gabah juga tombol hitung	Berhasil
		Klik tombol hitung dengan form terisi semua dengan benar	Menghitung, menyimpan data pada <i>database</i> , dan menghilangkan tombol hitung	Berhasil
		Klik tombol hitung namun data kosong	Menampilkan span “data tidak boleh kosong”	Berhasil
		Klik tombol reset data mutu dan harga	Mengatur ulang nilai mutu dan harga menjadi nol	Berhasil
9.	Mengelola data pemasok beras	Klik menu penilaian padi, lalu klik submenu data pemasok beras	Menampilkan halaman data pemasok beras dan tabel pemasok, tombol tambah ubah, dan hapus	Berhasil
		Klik tombol tambah data pemasok beras, lalu klik submit	Menampilkan modal tambah data pemasok, form isian, dan tombol daftar. Serta menyimpan pada <i>database</i> , dan menampilkan flash alert data pemasok beras berhasil diinputkan	Berhasil
		Klik tombol submit namun form isian kosong	Menampilkan span “data tidak boleh kosong” pada setiap form yang kosong	Berhasil
		Klik tombol mengubah data	Menampilkan modal form ubah data beserta data yang akan diubah	Berhasil
		Klik tombol simpan pada modal ubah data	Menyimpan pada <i>database</i> dan menampilkan flash alert “Data Pemasok Beras Berhasil Diubah”	Berhasil
		Klik tombol hapus data	menampilkan <i>confirm message</i> “Hapus Data Pemasok ?”	Berhasil
		Klik tombol ok pada <i>confirm message</i> hapus data pemasok	Meghapus data pada <i>database</i> dan menampilkan flash alert “Data Pemasok Berhasil di Hapus”	Berhasil
Dilanjutkan				

Lanjutan				
No	Fitur	Aksi	Hasil	Ket.
10.	Mengelola data pemasok gabah	Klik menu penilaian padi, lalu klik submenu data pemasok gabah	Menampilkan halaman data pemasok gabah dan tabel pemasok, tombol penilaian, tambah ubah, dan hapus	Berhasil
		Klik tombol tambah data pemasok gabah, lalu klik submit	Menampilkan modal tambah data pemasok, form isian, dan tombol daftar. Serta menyimpan pada database, dan menampilkan flash alert data pemasok gabah berhasil diinputkan	Berhasil
		Klik tombol submit namun form isian kosong	Menampilkan span “data tidak boleh kosong” pada setiap form yang kosong	Berhasil
		Klik tombol mengubah data	Menampilkan modal form ubah data beserta data yang akan diubah	Berhasil
		Klik tombol simpan pada modal ubah data	Menyimpan pada database dan menampilkan flash alert “Data Berhasil Diubah”	Berhasil
		Klik tombol hapus data	menampilkan <i>confirm message</i> “Hapus Data Pemasok ?”	Berhasil
		Klik tombol ok pada <i>confirm message</i> hapus data pemasok	Meghapus data pada database dan menampilkan flash alert “Data Pemasok Berhasil di Hapus”	Berhasil
11.	Membuat penilaian pemasok beras	Klik menu penilaian padi, lalu klik submenu data pemasok beras	Menampilkan halaman data pemasok gabah dan tabel pemasok, tombol penilaian, tambah ubah, dan hapus	Berhasil
		Klik tombol penilaian	Menampilkan halaman baru serta data pada setiap form <i>dropdown list</i> dan tombol hitung	Berhasil
		Klik tombol hitung namun form isian kosong	Menampilkan span “data tidak boleh kosong” pada seetiap form yang kosong	Berhasil
		Klik tombol hitung namun data yang akan dimasukkan sudah benar	Menghitung perhitungan SMART dan menyimpan pada <i>database</i>	Berhasil
12.	Membuat penilaian pemasok gabah	Klik menu penilaian padi, lalu klik submenu data pemasok gabah	Menampilkan halaman data pemasok gabah dan tabel pemasok, tombol penilaian, tambah ubah, dan hapus	Berhasil
		Klik tombol penilaian	Menampilkan halaman baru serta data pada setiap form <i>dropdown list</i> dan tombol hitung	Berhasil
Dilanjutkan				

Lanjutan				
No	Fitur	Aksi	Hasil	Ket.
		Klik tombol hitung namun form isian kosong	Menampilkan span “data tidak boleh kosong” pada seetiap form yang kosong	Berhasil
13.	Melihat hasil penilaian gabah	Klik menu hasil penilaian, lalu klik submenu penilaian gabah	Menampilkan halaman hasil penilaian pemasok gabah dan tabel penilaian, tombol beli dan detail penilaian	Berhasil
		Klik tombol detail penilaian	Menampilkan modal detail penilaian pemasok gabah	Berhasil
14.	Melihat hasil penilaian beras	Klik menu hasil penilaian, lalu klik submenu penilaian beras	Menampilkan halaman hasil penilaian pemasok beras dan tabel penilaian, tombol beli dan detail penilaian	Berhasil
		Klik tombol detail penilaian	Menampilkan modal detail penilaian pemasok beras	Berhasil
15.	Membuat pembelian gabah	Klik menu hasil penilaian, lalu klik submenu penilaian gabah	Menampilkan halaman hasil penilaian pemasok gabah dan tabel penilaian, tombol beli dan detail penilaian	Berhasil
		Klik tombol pembelian	Menampilkan modal form pembelian dan tombol beli	Berhasil
		Klik tombol pembelian dan mengisi form dengan benar	Menghitung proses pembelian gabah, menyimpan pada <i>database</i> dan Menampilkan flash alert “Pembelian Gabah Berhasil”	Berhasil
		Klik tombol pembelian namun terdapat form yang kosong	Menampilkan span “data tidak boleh kosong”	Berhasil
		Klik tombol pembelian namun pembelian melebihi stok pemasok gabah	Menampilkan flash alert “Pembelian melebihi stok pemasok gabah”	Berhasil
		Klik tombol pembelian namun pembelian melebihi stok yang harus dibeli	Menampilkan flash alert “Pembelian melebihi stok yang harus dibeli”	Berhasil
		Memilih form <i>dropdown</i> pembelian semua	Form banyak pembelian <i>disable</i>	Berhasil
		Memilih form <i>dropdown</i> pembelian sebagian	Form banyak pembelian <i>enable</i>	Berhasil
16.	Membuat pembelian beras	Klik menu hasil penilaian, lalu klik submenu penilaian beras	Menampilkan halaman hasil penilaian pemasok beras dan tabel penilaian, tombol beli dan detail penilaian	Berhasil
Dilanjutkan				

Lanjutan				
No	Fitur	Aksi	Hasil	Ket.
		Klik tombol pembelian	Menampilkan modal form pembelian dan tombol beli	Berhasil
		Klik tombol pembelian dan mengisi form dengan benar	Menghitung proses pembelian beras, menyimpan pada <i>database</i> dan Menampilkan flash alert “Pembelian Beras Berhasil”	Berhasil
		Klik tombol pembelian namun terdapat form yang kosong	Menampilkan span “data tidak boleh kosong”	Berhasil
		Klik tombol pembelian namun pembelian melebihi stok pemasok beras	Menampilkan flash alert “Pembelian melebihi stok pemasok beras”	Berhasil
		Klik tombol pembelian namun pembelian melebihi stok yang harus dibeli	Menampilkan flash alert “Pembelian melebihi stok yang harus dibeli”	Berhasil
		Memilih form <i>dropdown</i> pembelian semua	Form banyak pembelian <i>disable</i>	Berhasil
		Memilih form <i>dropdown</i> pembelian sebagian	Form banyak pembelian <i>enable</i>	Berhasil
17.	Melihat riwayat pembelian gabah	Klik menu riwayat pembelian, lalu klik submenu data gabah dibeli	Menampilkan halaman riwayat pembelian gabah dan tabel riwayat pembelian gabah,	Berhasil
18.	Melihat riwayat pembelian beras	Klik menu riwayat pembelian, lalu klik submenu data beras dibeli	Menampilkan halaman riwayat pembelian beras dan tabel riwayat pembelian beras,	Berhasil
19.	Keluar Sistem (<i>Logout</i>)	Klik menu Nama dan hak akses Pengguna di pojok kanan atas halaman pengguna, lalu klik menu <i>logout</i>	Menghancurkan <i>session</i> yang ada, dan menampilkan halaman <i>login</i>	Berhasil

BAB 6. PENUTUP

Bab ini memaparkan kesimpulan dan saran dari penelitian yang telah dilakukan, kesimpulan dipaparkan agar mempermudah pembaca dalam memahami hasil dari penelitian, sedangkan saran digunakan sebagai acuan setiap pembaca untuk mengembangkan penelitian selanjutnya.

6.1 Kesimpulan

Kesimpulan dalam penelitian yang telah dilakukan untuk menjawab permasalahan dalam rumusan masalah adalah sebagai berikut :

1. Implementasi Metode SMART dalam pembelian padi terbaik dimulai dari penentuan sepuluh jenis kriteria dalam pembelian gabah, dan dua belas kriteria penentuan pembelian beras. Penentuan nilai dan urutan perankingan dilakukan dengan mempertimbangkan seluruh kriteria, terkecuali untuk penentuan mutu dimana kriteria penentu penilaian mengacu pada Standart Nasional Indonesia tentang mutu gabah dan beras. Perhitungan dengan metode SMART dilakukan dengan mengacu pada nilai *utility* setiap subkriteria yang besar nilainya diurutkan berdasarkan prioritas kepentingan dalam pembelian gabah yang diinginkan. Penilaian pembobotan kriteria dilakukan dengan memberikan nilai skala 1 – 100 berdasarkan prioritas kepentingan dalam pembelian gabah, bobot tersebut selanjutnya dihitung untuk mendapatkan nilai normalisasi dengan total penjumlahan nilai normalisasi 100 persen, perhitungan metode SMART dalam penunjang keputusan pembelian beras juga dilakukan dengan cara yang sama dengan pembelian gabah.

Range penentuan mutu gabah adalah, mutu satu dengan nilai lebih dari sama dengan 61,250. Mutu dua dengan range nilai lebih dari sama dengan 50,000 sampai kurang dari 61,250. Mutu tiga memiliki range kurang dari 50,000 sampai lebih dari sama dengan 38,750, Sedangkan kurang dari 38,750 dianggap

sebagai gabah mutu tiga. Penentuan range mutu beras adalah sebagai berikut, mutu satu 73,158, mutu dua 60,175, mutu tiga 48,596, mutu empat 41,462, dan mutu lima 28,655, sedangkan untuk nilai kurang dari 2,655 masuk pada beras mutu lima. Perangkingan ditampilkan berdasarkan nilai perhitungan SMART seluruh kriteria setiap pemasok dari nilai yang terbesar ke yang terkecil. Penentuan harga gabah maupun beras mengacu pada nilai mutu yang ditetapkan oleh sistem, perlu diketahui bahwa harga pasaran sewaktu – waktu dapat berubah.

2. Sistem Penunjang Keputusan Pembelian Padi Terbaik dibangun dengan Software Development Life Cycle model Waterfall. Model Waterfall cocok dalam pengembangan sistem penunjang keputusan pembelian padi terbaik, dikarenakan sistem yang dikembangkan masuk dalam lingkup yang sederhana, apabila terdapat upgrade sistem atau penambahan fitur, maka yang harus dilakukan adalah analisis kebutuhan dari awal, desain sistem, pengkodean, sampai dengan implementasi dan pemeliharaan sistem. Desain sistem dibangun dengan model UML sehingga memudahkan dalam pembangunan sistem yang menerapkan design pattern MVC atau OOP, sedangkan pengkodean sistem dibangun dengan framework php Codeigniter. Pengujian sistem digunakan dengan metode black box, white box, dan pengujian perhitungan manual dengan mencari keakuratan perhitungan sistem. Keakuratan sistem yang dikembangkan dinilai sangat akurat dalam menentukan penilaian perhitungan SMART dikarenakan hasil dari perhitungan manual sama dengan perhitungan sistem.

6.2 Saran

1. Sistem yang dikembangkan selanjutnya diperlukan status bahwa penilaian tidak valid jika penentuan kriteria dan pembobotan telah berubah dari sebelumnya, sehingga hasil perhitungan gabah dan beras setiap pemasok dapat bersifat dinamis.
2. Penentuan pembelian stok beras dan gabah, serta harga pasaran dapat dilakukan dengan menggunakan metode peramalan dengan melihat trend data sebelumnya untuk pembelian setiap bulan atau perhari, sedangkan harga dapat dilihat trend dari harga pembelian beras setiap bulan atau hari.

DAFTAR PUSTAKA

- Ardianto, R. D., W. Anggraeni, dan R. P. Kusumawardani. 2014. Penerapan fuzzy analytical hierarchy process pada sistem penilaian pegawai di. (September):1–7.
- Atiqah. 2013. Implementasi metode smart pada sistem pendukung keputusan pemilihan pembelian mobil keluarga. (November):75–80.
- Boustedt, J. 2010. Ways to understand class diagrams
- BSN. 1987. Gabah, standar mutu
- BSN. 2008. Sni 6128: 2008 beras
- Hidayati, R. 2014. Peningkatan kualitas olahan beras sebagai makanan pokok melalui penambahan daun kelor (moringa oleifera). 3:205–211.
- Ibrahim, N., R. Ibrahim, dan M. Z. Saringat. 2011. Consistency rules between uml use case and activity diagrams using logical approach *. 5(3):498–508.
- Ir. Sutrisno Koswara, Ms. 2009. Teknologi pengolahan beras (teori dan praktek). 1–14.
- Isniatun Munawaroh. 2016. Urgensi penelitian dan pengembangan. 1–5.
- Jannah, D. M. 2013. Sistem penunjang keputusan penentuan mutu beras dengan menggunakan metode analytical hierarchy process
- Kaewchinporn, C. dan Y. Limpiyakorn. 2013. Enhancement of action description language for uml activity diagram review. 7(2):255–272.
- Kasie, F. M. 2013. Combining simple multiple attribute rating technique and analytical hierarchy process for designing multi-criteria performance measurement framework. *The Global Journal of Researches in Engineering: Industrial Engineering*. 13(1):15–30.
- Khan, M. E. 2011. Different approaches to white box testing technique for finding errors. 5(3):1–14.
- Kustiyahningsih, Y., D. R. Anamisa, J. T. Informatika, F. Teknik, dan U. Trunojoyo. 2013. Siswa sma menggunakan metode knn dan smart

- Letkowski, J. 2014. Doing database design with mysql. 6
- Lutfi, M., W. Agung, N. Gunomo, P. Pisang, dan E. Gondok. 2013. (in press) analisis kinerja pita tanam organik sebagai media perkecambahan benih padi (*oryza sativa* l .) sistem tabel dengan desain tertutup dan terbuka. 1(2):59–68.
- Minhas, N. M., A. M. Qazi, S. Shahzadi, dan S. Ghafoor. 2015. An integration of uml sequence diagram with formal specification methods — a formal solution based on z. (August):372–383.
- Pandara, E. dan M. N. , Stanley Karouw. 2014. Implementasi proses uji sistem informasi admisi. 4(2):1–6.
- Shao, D. dan D. E. Perry. 2006. A case for white-box testing using declarative specifications poster abstract. 19(7):2006.
- Soegijono, O. D. K. R. 1993. Wawancara sebagai salah satu metode pengumpulan data. III(1):17–21.
- Turban, E., J. E. Aronson, dan T. Liang. 2005. Decision support systems and intelligent systems

LAMPIRAN

LAMPIRAN A. USE CASE SKENARIO

A.1 Use Case Skenario Login

Nomor usecase	01
Nama usecase	Masuk
Aktor	Admin atau pakar
Deskripsi	Merupakan fitur untuk masuk ke dalam sistem sesuai dengan hak akses pengguna
Pre-Kondisi	Pengguna berada pada halaman login untuk melakukan login dan belum menginputkan data untuk login
Pasca Kondisi	Pengguna telah melakukan login dan masuk dalam halaman utama admin
Event Flow	
ALIRAN NORMAL	
AKTOR	SISTEM
1. klik <i>Login Now</i>	
	2. Menampilkan modal login a. Username (tipe text) b. Password (tipe password)
3. Memasukkan data login: a. Username (tipe text) b. Password (tipe password)	
4. Klik tombol login	

	5. Menampilkan halaman home admin
ALTERNATIF FLOW: Username atau pasword salah	
4. Klik tombol login	
	5. Menampilkan flash alert “Gagal Login, username/password salah” pada halaman login
ALTERNATIF FLOW: Data kosong	
4. klik tombol login	
	5. Menampilkan span “data tidak boleh kosong”

Nomor usecase	01
Nama usecase	Masuk
Aktor	Pegawai
Deskripsi	Merupakan fitur untuk masuk ke dalam sistem sesuai dengan hak akses pengguna
Pre-Kondisi	Pengguna berada pada halaman login untuk melakukan login dan belum menginputkan data untuk login
Pasca Kondisi	Pengguna telah melakukan login dan masuk dalam halaman utama pegawai
Event Flow	
ALIRAN NORMAL	
AKTOR	SISTEM

1. klik Login Now	
	2. Menampilkan modal login a. Username (tipe text) b. Password (tipe password)
3. Memasukkan data login: a. Username (tipe text) b. Password (tipe password)	
4. klik tombol login	
	5. menampilkan halaman home pegawai
ALTERNATIF FLOW: Username atau pasword salah	
4. klik tombol login	
	5. menampilkan flash alert “Gagal Login, username/password salah” pada halaman login
ALTERNATIF FLOW: Data kosong	
4. klik tombol login	
	5. Menampilkan span “data tidak boleh kosong”

A.2 Use Case Skenario Mengelola Data Pengguna

Nomor usecase	02
Nama usecase	Mengelola data pengguna
Aktor	Admin atau pakar

Deskripsi	Fitur untuk admin(pakar) untuk mengelola data pengguna(user) berupa memasukkan, edit, dan hapus data pengguna sistem.
Pre-Kondisi	Pengguna berada pada halaman utama admin, dan akan memasukkan, mengubah, dan menghapus data pengguna
Pasca Kondisi	Pengguna telah beerada pada halaman hak akses pengguna, dan melakukan aksi memasukkan, mengubah, dan menghapus data pengguna
Event Flow	
ALIRAN NORMAL	
Tambah Data Pengguna	
AKTOR	SISTEM
1. Klik Menu Pengguna	
	2. Menampilkan submenu kelola hak akses
3. Klik tombol submenu kelola hak akses	
	4. Menampilkan halaman kelola hak akses <ul style="list-style-type: none"> a. Tombol "+ Tambah Hak Akses" b. Tabel list pengguna: <ul style="list-style-type: none"> • ID

	<ul style="list-style-type: none">• Username• Nama• Email• Password• Hak Akses• Aksi <p>c. Tombol pensil (mengubah data)</p> <p>d. Tombol sampah (menghapus data)</p>
5. Klik Tombol “+ Tambah Hak Akses”	
	<p>6. Menampilkan modal form tambah data pengguna</p> <ul style="list-style-type: none">a. Nama (tipe text)b. Username (tipe text)c. Password (tipe password)d. Alamat email (tipe email)e. Alamat rumah (tipe text)f. Level (tipe dropdown list)g. Tombol daftar
7. Mengisi form tambah data pengguna <ul style="list-style-type: none">a. Nama (tipe text)b. Username (tipe text)c. Password (tipe password)	

d. Alamat email (tipe email)	
e. Alamat rumah (tipe text)	
f. Level (tipe dropdown list)	
8. Klik tombol daftar	
	9. Menyimpan data
	10. Menampilkan flash alert “Data Pengguna Berhasil di Inputkan”
11. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW: Username Ada yang Sama	
8. klik tombol daftar	
	9. Menampilkan flash alert “Username Telah Digunakan”
10. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW Data Kosong	
8. klik tombol daftar	
	9. Menampilkan span “data tidak boleh kosong”
ALIRAN NORMAL Mengubah Data Pengguna	

AKTOR	SISTEM
1. Klik Menu Pengguna	
	2. menampilkan Submenu Kelola Hak Akses
3. Klik tombol submenu Kelola Hak Akses	
	<p>4. Menampilkan Halaman Kelola Hak Akses</p> <ol style="list-style-type: none"> a. Tombol “+ Tambah Hak Akses” b. Tabel list pengguna: <ul style="list-style-type: none"> • ID • Username • Nama • Email • Password • Hak Akses • Aksi c. Tombol pensil (mengubah data) d. Tombol sampah (menghapus data)
5. Klik Tombol pensil (mengubah data)	
	<p>6. Menampilkan modal form edit data pengguna</p> <ol style="list-style-type: none"> a. Nama (tipe text)

	<ul style="list-style-type: none"> b. Username (tipe text) c. Password (tipe password) d. Alamat email (tipe email) e. Alamat rumah (tipe text) f. Level (dropdown list) g. Tombol daftar
<p>7. Mengisi form ubah data pengguna</p> <ul style="list-style-type: none"> a. Nama (tipe text) b. Username (tipe text) c. Password (tipe password) d. Alamat email (tipe email) e. Alamat rumah (tipe text) f. Level (dropdown list) 	
8. Klik tombol simpan	
	9. Menyimpan data yang telah diubah
	10. Menampilkan flash alert “Data Pengguna Berhasil Diubah”
11. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW	
Data Kosong	
8. klik tombol simpan	

	10. Menampilkan span “data tidak boleh kosong”
ALIRAN NORMAL	
Menghapus Data Pengguna	
AKTOR	SISTEM
1. Klik Menu Pengguna	
	2. menampilkan Submenu Kelola Hak Akses
3. Klik tombol submenu Kelola Hak Akses	
	<p>4. Menampilkan Halaman Kelola Hak Akses</p> <p>a. Tombol “+ Tambah Hak Akses”</p> <p>b. Tabel list pengguna:</p> <ul style="list-style-type: none"> • ID • Username • Nama • Email • Password • Hak Akses • Aksi <p>c. Tombol pensil (mengubah data)</p> <p>d. Tombol sampah (menghapus data)</p>

5. Klik Tombol sampah (menghapus data)	
	6. menampilkan confirm message “Hapus Data Pengguna ?”
7. Klik OK	
	8. Menghapus data pengguna dalam database
	9. Menampilkan flash alert “Data Pengguna Berhasil di Hapus”
ALIRAN NORMAL	
Batal Menghapus Data Pengguna	
AKTOR	SISTEM
1. Klik Menu Pengguna	
	2. menampilkan Submenu Kelola Hak Akses
3. Klik tombol submenu Kelola Hak Akses	
	4. Menampilkan Halaman Kelola Hak Akses <ul style="list-style-type: none"> a. Tombol “+ Tambah Hak Akses” b. Tabel list pengguna: <ul style="list-style-type: none"> • ID • Username • Nama • Email • Password

	<ul style="list-style-type: none"> • Hak Akses • Aksi <p>c. Tombol pensil (mengubah data)</p> <p>d. Tombol sampah (menghapus data)</p>
5. Klik Tombol sampah (menghapus data)	
	6. menampilkan confirm message “Hapus Data Pengguna ?”
7. Klik Cancel	
	8. Menampilkan halaman kelola hak akses

A.3 Use Case Skenario Mengelola Kriteria Beras

Nomor usecase	03
Nama usecase	Mengelola Kriteria dan Pembobotan Beras
Aktor	Admin atau Pakar
Deskripsi	Merupakan fitur untuk mengelola kriteria, subkriteria dan pembobotan sebagai parameter dalam pemilihan beras terbaik
Pre-Kondisi	Pengguna berada pada halaman utama admin, dan akan memasukkan, mengubah, menghapus kriteria, sub

	kriteria, dan pembobotan setiap parameter.
Pasca Kondisi	Pengguna telah berada pada halaman utama kelola kriteria beras, dan telah memasukkan, mengubah, menghapus kriteria, sub kriteria, dan pembobotan setiap parameter.
Event Flow	
ALIRAN NORMAL	
Tambah Kriteria	
1. Klik menu Kriteria Beras	
	2. Menampilkan submenu kelola bobot
3. Klik sub menu kelola bobot	
	4. Menampilkan halaman kriteria dan bobot: <ul style="list-style-type: none"> a. Tombol + Tambah Kriteria b. Tabel kriteria: <ul style="list-style-type: none"> • ID • Kriteria • Nilai • Bobot (presentase) • Normalisasi • Aksi c. Tombol + (tambah subkriteria)

	<ul style="list-style-type: none"> d. Tombol mata (untuk detail subkriteria berdasarkan kriteria) e. Tombol sampah (untuk menghapus kriteria)
5. Klik tombol + Tambah Kriteria	
	<ul style="list-style-type: none"> 6. Menampilkan modal form kriteria: <ul style="list-style-type: none"> a. Nama kriteria (tipe text) b. Bobot kriteria (tipe angka) c. Tombol tambah
<ul style="list-style-type: none"> 7. Mengisi form kriteria: <ul style="list-style-type: none"> a. Nama kriteria (tipe text) b. Bobot kriteria (tipe angka) 	
8. Klik tombol tambah	
	9. Menyimpan data ke database
	10. Menghitung Nilai Bobot dan Normalisasi dengan rumus Bobot : $(\text{nilai} / \text{total nilai}) * 100$ Normalisasi : $((\text{nilai} / \text{total nilai}) * 100) / 100$

	11. Menampilkan flash alert “Data Kriteria Beras Berhasil di Tambahkan”
12. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW Data kosong	
8. Klik tombol tambah	
	9. Menampilkan span “data tidak boleh kosong
ALTERNATIF FLOW Data tidak sesuai	
8. Klik tombol tambah	
	9. Menampilkan flash alert “bobot harus dalam range 0 - 100”
10. Klik icon (x) untuk menutup flash alert	
ALIRAN NORMAL Hapus Kriteria	
1. Klik menu Kriteria Beras	
	2. Menampilkan submenu kelola bobot
3. Klik sub menu kelola bobot	
	4. Menampilkan halaman kriteria dan bobot:

	<ol style="list-style-type: none"> a. Tombol + Tambah Kriteria b. Tabel kriteria: <ul style="list-style-type: none"> • ID • Kriteria • Nilai • Bobot (presentase) • Normalisasi • Aksi c. Tombol + (tambah subkriteria) d. Tombol mata (untuk detail subkriteria berdasarkan kriteria) e. Tombol sampah (untuk menghapus kriteria)
5. Klik tombol sampah (untuk menghapus kriteria)	
	6. menampilkan confirm message “Hapus kriteria ini ?”
7. Klik OK	
	8. Menghapus kriteria dalam database
	9. Menampilkan flash alert “Data Kriteria Beras Berhasil di Hapus” pada halaman kriteria beras.

10. Klik icon (x) untuk menutup flash alert	
ALIRAN NORMAL Batal Hapus kriteria	
1. Klik menu Kriteria Beras	
	2. Menampilkan submenu kelola bobot
3. Klik sub menu kelola bobot	
	<p>4. Menampilkan halaman kriteria dan bobot:</p> <ol style="list-style-type: none"> a. Tombol + Tambah Kriteria b. Tabel kriteria: <ul style="list-style-type: none"> • ID • Kriteria • Nilai • Bobot (presentase) • Normalisasi • Aksi c. Tombol + (tambah subkriteria) d. Tombol mata (untuk detail subkriteria berdasarkan kriteria) <p>Tombol sampah (untuk menghapus kriteria)</p>

5. Klik tombol sampah (untuk menghapus kriteria)	
	6. menampilkan confirm message “ Hapus kriteria ini ?”
7. Klik Cancel	
	8. Menampilkan halaman kriteria beras
ALIRAN NORMAL	
Tambah sub kriteria	
1. Klik menu Kriteria Beras	
	2. Menampilkan submenu kelola bobot
3. Klik sub menu kelola bobot	
	4. Menampilkan halaman kriteria dan bobot: <ul style="list-style-type: none"> a. Tombol + Tambah Kriteria b. Tabel kriteria: <ul style="list-style-type: none"> • ID • Kriteria • Nilai • Bobot (presentase) • Normalisasi • Aksi c. Tombol + (tambah subkriteria)

	<p>d. Tombol mata (untuk detail subkriteria berdasarkan kriteria)</p> <p>e. Tombol sampah (untuk menghapus kriteria)</p>
5. Klik Tombol + (tambah subkriteria)	
	<p>6. Menampilkan modal form subkriteria:</p> <p>a. ID Kriteria (tipe text)</p> <p>b. Nama Kriteria (tipe text)</p> <p>c. Nama Subkriteria (Tipe Text)</p> <p>d. Nilai Utility (Tipe Angka)</p> <p>e. Tombol Tambah</p>
<p>7. Mengisi modal form subkriteria:</p> <ul style="list-style-type: none"> • Nama Subkriteria (Tipe Text) • Nilai Utility (Tipe Angka) 	
8. Klik Tombol Tambah	
	9. Menyimpan data ke database
	10. Menampilkan flash alert "Data Subkriteria Berhasil di

	Tambahkan pada kriteria” pada halaman kriteria beras
11. Klik icon (x) untuk menutup flash alert	
ALIRAN ALTERNATIF	
Data kosong	
8. Klik Tombol Tambah	
	9. Menampilkan span “data tidak boleh kosong”
ALIRAN NORMAL	
Lihat detail subkriteria	
1. Klik menu Kriteria Beras	
	2. Menampilkan submenu kelola bobot
3. Klik sub menu kelola bobot	
	4. Menampilkan halaman kriteria dan bobot: <ul style="list-style-type: none"> a. Tombol + Tambah Kriteria b. Tabel kriteria: <ul style="list-style-type: none"> • ID • Kriteria • Nilai • Bobot (presentase) • Normalisasi • Aksi

	<ul style="list-style-type: none"> c. Tombol + (tambah subkriteria) d. Tombol mata (untuk detail subkriteria berdasarkan kriteria) e. Tombol sampah (untuk menghapus kriteria)
5. Klik Tombol Mata (untuk detail subkriteria berdasarkan kriteria)	
	<ul style="list-style-type: none"> 6. Menampilkan modal tabel detail subkriteria beras sesuai kriteria <ul style="list-style-type: none"> a. Tabel detail subkriteria <ul style="list-style-type: none"> • No • Subkriteria • Utility • Aksi b. Tombol sampah (untuk menghapus kriteria)
ALIRAN NORMAL	
Hapus sub kriteria	
1. Klik menu Kriteria Beras	
	2. Menampilkan submenu kelola bobot
3. Klik sub menu kelola bobot	

	<p>4. Menampilkan halaman kriteria dan bobot:</p> <ol style="list-style-type: none">a. Tombol + Tambah Kriteriab. Tabel kriteria:<ul style="list-style-type: none">• ID• Kriteria• Nilai• Bobot (presentase)• Normalisasi• Aksic. Tombol + (tambah subkriteria)d. Tombol mata (untuk detail subkriteria berdasarkan kriteria)e. Tombol sampah (untuk menghapus kriteria)
<p>5. Klik Tombol Mata (untuk detail subkriteria berdasarkan kriteria)</p>	
	<p>6. Menampilkan modal tabel detail subkriteria beras sesuai kriteria</p> <ol style="list-style-type: none">a. Tabel detail subkriteria<ul style="list-style-type: none">• No• Subkriteria• Utility

	<ul style="list-style-type: none"> • Aksi <ul style="list-style-type: none"> b. Tombol sampah (untuk menghapus subkriteria)
7. Klik tombol sampah (untuk menghapus subkriteria)	
	8. menampilkan confirm message “ Hapus subkriteria ini ?”
9. Klik OK	
	10. Menghapus subkriteria dalam database
	11. Menampilkan flash alert “Data Subkriteria Beras Berhasil di Hapus” pada halaman kriteria beras
12. Klik icon (x) untuk menutup flash alert	
ALIRAN NORMAL	
Batal Hapus sub kriteria	
1. Klik menu Kriteria Beras	
	2. Menampilkan submenu kelola bobot
3. Klik sub menu kelola bobot	
	4. Menampilkan halaman kriteria dan bobot: <ul style="list-style-type: none"> a. Tombol + Tambah Kriteria

	<p>b. Tabel kriteria:</p> <ul style="list-style-type: none">• ID• Kriteria• Nilai• Bobot (presentase)• Normalisasi• Aksi <p>c. Tombol + (tambah subkriteria)</p> <p>d. Tombol mata (untuk detail subkriteria berdasarkan kriteria)</p> <p>e. Tombol sampah (untuk menghapus kriteria)</p>
<p>5. Klik Tombol Mata (untuk detail subkriteria berdasarkan kriteria)</p>	
	<p>6. Menampilkan modal tabel detail subkriteria beras sesuai kriteria</p> <p>a. Tabel detail subkriteria</p> <ul style="list-style-type: none">• No• Subkriteria• Utility• Aksi <p>b. Tombol sampah (untuk menghapus subkriteria)</p>

7. Klik tombol sampah (untuk menghapus subkriteria)	
	8. menampilkan confirm message “ Hapus subkriteria ini ?”
9. Klik Tombol Cancel	
	10. Menampilkan halaman kriteria beras

A.4 Use Case Skenario Mengelola Stok Pembelian Gabah

Nomor usecase	05
Nama usecase	Mengelola Stok Pembelian Gabah
Aktor	Admin atau Pakar
Deskripsi	Merupakan fitur untuk mengetahui stok gabah yang tersedia, mengelola seberapa banyak pembelian gabah yang dibutuhkan untuk proses produksi dan juga melihat siapa saja pegawai yang telah membeli beras dari pemasok gabah.
Pre-Kondisi	Pengguna berada pada halaman utama admin, dan akan mengelola pembelian beras yang dibutuhkan.
Pasca Kondisi	Pengguna telah berada pada halaman utama kelola stok gabah, dan telah mengelola pembelian stok gabah.
Event Flow	

ALIRAN NORMAL Tambah Beli Gabah	
1. Klik menu Stok Padi	
	2. Menampilkan submenu kelola stok gabah dan kelola stok beras
3. Klik sub menu kelola stok gabah	
	<p>4. Menampilkan halaman kelola stok gabah:</p> <ol style="list-style-type: none"> a. Tombol + (menambah beli gabah) b. Tombol – (mengurangi beli gabah) c. Form Stok Gabah Tersedia (Readonly) d. Form Beli Gabah Sebanyak (Readonly) e. Tabel Pembelian Gabah: <ul style="list-style-type: none"> • No • Nama • Alamat • Berat (Kilogram) • Tanggal Masuk • Tanggal Beli • Pembeli
5. Klik tombol + (menambah beli gabah)	

	<p>6. Menampilkan modal form beli gabah:</p> <ul style="list-style-type: none"> a. Banyak gabah yang akan dibeli (angka) b. Tombol tambah
<p>7. Mengisi form beli gabah:</p> <ul style="list-style-type: none"> a. Banyak gabah yang akan dibeli (tipe angka) 	
8. Klik tombol tambah	
	9. Menyimpan data ke database
	10. Menghitung total penjumlahan pembelian gabah sebelumnya dengan nilai penambahan yang dimasukkan.
	11. Menampilkan flash alert “Penambahan Pembelian Stok Gabah Berhasil”
12. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW	
Data kosong	
8. Klik tombol tambah	
	9. Menampilkan span “data tidak boleh kosong

ALIRAN NORMAL	
Pengurangan Beli Gabah	
1. Klik menu Stok Padi	
	2. Menampilkan submenu kelola stok gabah dan kelola stok beras
3. Klik sub menu kelola stok gabah	
	<p>4. Menampilkan halaman kelola stok gabah:</p> <ol style="list-style-type: none"> a. Tombol + (menambah beli gabah) b. Tombol – (mengurangi beli gabah) c. Form Stok Gabah Tersedia (Readonly) d. Form Beli Gabah Sebanyak (Readonly) e. Tabel Pembelian Gabah: <ul style="list-style-type: none"> • No • Nama • Alamat • Berat (Kilogram) • Tanggal Masuk • Tanggal Beli • Pembeli
5. Klik tombol - (mengurangi beli gabah)	

	<p>6. Menampilkan modal form beli gabah:</p> <ol style="list-style-type: none"> a. Kurangi banyak gabah yang akan dibeli (angka) b. Tombol tambah
<p>7. Mengisi form beli gabah:</p> <ol style="list-style-type: none"> a. Kurangi banyak gabah yang akan dibeli (tipe angka) 	
<p>8. Klik tombol tambah</p>	
	<p>9. Menyimpan data ke database</p>
	<p>10. Menghitung pengurangan stok pembelian gabah sebelumnya dengan data pengurangan yang dimasukkan.</p>
	<p>11. Menampilkan flash alert “Pengurangan Pembelian Stok Gabah Berhasil”</p>
<p>12. Klik icon (x) untuk menutup flash alert</p>	
<p>ALTERNATIF FLOW</p> <p>Data kosong</p>	
<p>8. Klik tombol tambah</p>	
	<p>9. Menampilkan span “data tidak boleh kosong”</p>
<p>ALTERNATIF FLOW</p>	

Data tidak sesuai	
8. Klik tombol tambah	
	9. Menampilkan span “Pengurangan Pembelian Stok Gabah Tidak Boleh Melebihi Pembelian Stok”
a. Klik icon (x) untuk menutup flash alert	

A.5 Use Case Skenario Mengelola Stok Pembelian Beras

Nomor usecase	06
Nama usecase	Mengelola Stok Pembelian Beras
Aktor	Admin atau Pakar
Deskripsi	Merupakan fitur untuk mengetahui stok beras yang tersedia, mengelola seberapa banyak pembelian beras yang dibutuhkan untuk proses produksi dan juga melihat siapa saja pegawai yang telah membeli beras dari pemasok beras.
Pre-Kondisi	Pengguna berada pada halaman utama admin, dan akan mengelola pembelian beras yang dibutuhkan.

Pasca Kondisi	Pengguna telah berada pada halaman utama kelola stok beras, dan telah mengelola pembelian stok beras.
Event Flow	
ALIRAN NORMAL Tambah Beli Beras	
1. Klik menu Stok Padi	
	2. Menampilkan submenu kelola stok beras dan kelola stok beras
3. Klik sub menu kelola stok beras	
	4. Menampilkan halaman kelola stok beras: <ul style="list-style-type: none"> a. Tombol + (menambah beli beras) b. Tombol – (mengurangi beli beras) c. Form Stok Beras Tersedia (Readonly) d. Form Beli Beras Sebanyak (Readonly) e. Tabel Pembelian Beras: <ul style="list-style-type: none"> • No • Nama • Alamat • Berat (Kilogram) • Tanggal Masuk

	<ul style="list-style-type: none"> • Tanggal Beli • Pembeli
5. Klik tombol + (menambah beli beras)	
	<p>6. Menampilkan modal form beli beras:</p> <p>a. Banyak beras yang akan dibeli (angka)</p>
7. Mengisi form beli beras:	
a. Banyak beras yang akan dibeli (tipe angka)	
8. Klik tombol tambah	
	9. Menyimpan data ke database
	10. Menghitung total penjumlahan pembelian beras sebelumnya dengan nilai penambahan yang dimasukkan.
	11. Menampilkan flash alert “Penambahan Pembelian Stok Beras Berhasil”
12. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW	
Data kosong	
8. Klik tombol tambah	

	9. Menampilkan span “data tidak boleh kosong
ALIRAN NORMAL	
Pengurangan Beli Beras	
1. Klik menu Stok Padi	
	2. Menampilkan submenu kelola stok beras dan kelola stok beras
3. Klik sub menu kelola stok beras	
	4. Menampilkan halaman kelola stok beras: <ul style="list-style-type: none"> a. Tombol + (menambah beli beras) b. Tombol – (mengurangi beli beras) c. Form Stok Beras Tersedia (Readonly) d. Form Beli Beras Sebanyak (Readonly) e. Tabel Pembelian Beras: <ul style="list-style-type: none"> • No • Nama • Alamat • Berat (Kilogram) • Tanggal Masuk

	<ul style="list-style-type: none"> • Tanggal Beli • Pembeli
5. Klik tombol - (mengurangi beli beras)	
	<p>6. Menampilkan modal form beli beras:</p> <p>a. Kurangi banyak beras yang akan dibeli (angka)</p>
7. Mengisi form beli beras:	
a. Kurangi banyak beras yang akan dibeli (tipe angka)	
8. Klik tombol tambah	
	9. Menyimpan data ke database
	10. Menghitung pengurangan stok pembelian beras sebelumnya dengan data pengurangan yang dimasukkan.
	11. Menampilkan flash alert “Pengurangan Pembelian Stok Beras Berhasil”
12. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW	
Data kosong	
8. Klik tombol tambah	

	9. Menampilkan span “data tidak boleh kosong”
ALTERNATIF FLOW	
Data tidak sesuai	
8. Klik tombol tambah	
	9. Menampilkan span “Pengurangan Pembelian Stok Beras Tidak Boleh Melebihi Pembelian Stok”
10. Klik icon (x) untuk menutup flash alert	

A.6 Use Case Skenario Mengelola Penilaian Mutu dan Harga Gabah

Nomor usecase	08
Nama usecase	Mengelola Penilaian Mutu dan Harga Gabah
Aktor	Admin
Deskripsi	Merupakan fitur untuk mengelola penilaian mutu dan juga penetapan harga pasaran gabah sesuai dengan standarisasi penggolongan mutu gabah.
Pre-Kondisi	Pengguna berada pada halaman utama admin, dan akan mengelola penilaian mutu dan harga gabah.
Pasca Kondisi	Pengguna telah berada pada halaman utama penilaian mutu dan harga gabah,

	dan telah mengelola penilaian mutu dan harga gabah.
Event Flow	
ALIRAN NORMAL	
Tambah Penilaian dan Harga Gabah	
1. Klik menu Mutu dan Harga	
	2. Menampilkan submenu penilaian gabah dan penilaian gabah
3. Klik sub menu penilaian gabah	
	<p>4. Menampilkan halaman penilaian mutu dan harga gabah</p> <p>a. Tabel penilaian mutu dan harga gabah:</p> <ul style="list-style-type: none"> • ID • Nilai Mutu • Mutu Gabah • Harga Gabah • Aksi <p>b. Tombol Note (Penilaian Mutu dan Harga)</p> <p>c. Tombol panah melingkar (mereset ulang penilaian dan harga)</p>

5. Klik Tombol “Gambar Note (Catatan)”	
	<p>6. Menampilkan modal form mutu dan harga gabah</p> <ul style="list-style-type: none">a. Mutu Gabah (tipe text) readonlyb. Harga Gabah (tipe angka)c. Gabah Varietas Lain (tipe combobox)d. Kadar Air (tipe combobox)e. Gabah Hampa (tipe combobox)f. Butir Mengapur + Gabah Muda (tipe combobox)g. Benda Asing (tipe combobox)h. Butir Merah (tipe combobox)i. Butir Rusak + Butir Kuning (tipe combobox)j. Tombol Hitung
7. Mengisi form form mutu dan harga gabah <ul style="list-style-type: none">a. Mutu Gabah (tipe text) readonly	

<ul style="list-style-type: none"> b. Harga Gabah (tipe angka) c. Gabah Varietas Lain (tipe combobox) d. Kadar Air (tipe combobox) e. Gabah Hampa (tipe combobox) f. Butir Mengapur + Gabah Muda (tipe combobox) g. Benda Asing (tipe combobox) h. Butir Merah (tipe combobox) i. Butir Rusak + Butir Kuning (tipe combobox) 	
8. Klik tombol Hitung	
	<p>9. Menghitung Hasil Penilaian Mutu dengan Rumus Metode SMART = $\sum_{j=1}^k w_j u_i(a_i)$, $i = 1, 2, \dots, k$ SMART</p>
	<p>10. Menyimpan hasil perhitungan dan inputan harga kedalam database.</p>

	11. Menampilkan flash alert “Perhitungan Mutu dan Harga Gabah Berhasil”
12. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW	
Data Kosong	
8. klik tombol Hitung	
	9. Menampilkan span “data tidak boleh kosong”
ALIRAN NORMAL	
Mengatur Ulang Nilai Penilaian dan Harga Gabah	
AKTOR	SISTEM
1. Klik menu Mutu dan Harga	
	2. Menampilkan submenu penilaian gabah dan penilaian gabah
3. Klik sub menu penilaian gabah	
	4. Menampilkan halaman penilaian mutu dan harga gabah <ol style="list-style-type: none"> a. Tabel penilaian mutu dan harga gabah: <ul style="list-style-type: none"> • ID • Nilai Mutu • Mutu Gabah

	<ul style="list-style-type: none"> • Harga Gabah • Aksi <p>b. Tombol Note (Penilaian Mutu dan Harga)</p> <p>c. Tombol panah melingkar (mereset ulang penilaian dan</p> <p>d. harga)</p>
5. Klik Tombol “Panah Melingkar”	
	6. Merubah nilai mutu dan harga gabah menjadi nol pada database
	7. Menampilkan flash alert “Nilai Mutu dan Harga Berhasil di atur ulang”.
8. Klik icon (x) untuk menutup flash alert	

A.7 Use Case Skenario Mengelola Penilaian Mutu dan Harga Beras

Nomor usecase	07
Nama usecase	Mengelola Penilaian Mutu dan Harga Beras
Aktor	Admin
Deskripsi	Merupakan fitur untuk mengelola penilaian mutu dan juga penetapan

	harga pasaran beras sesuai dengan standarisasi penggolongan mutu beras.
Pre-Kondisi	Pengguna berada pada halaman utama admin, dan akan mengelola penilaian mutu dan harga beras.
Pasca Kondisi	Pengguna telah berada pada halaman utama penilaian mutu dan harga beras, dan telah mengelola penilaian mutu dan harga beras.
Event Flow	
ALIRAN NORMAL	
Tambah Penilaian dan Harga Beras	
1. Klik menu Mutu dan Harga	
	2. Menampilkan submenu penilaian gabah dan penilaian beras
3. Klik sub menu penilaian beras	
	4. Menampilkan halaman penilaian mutu dan harga beras <ol style="list-style-type: none"> a. Tabel penilaian mutu dan harga beras: <ul style="list-style-type: none"> • ID • Nilai Mutu • Mutu Beras • Harga Beras • Aksi

	<ul style="list-style-type: none">b. Tombol Note (Penilaian Mutu dan Harga)c. Tombol panah melingkar (mereset ulang penilaian dan harga)
5. Klik Tombol “Gambar Note (Catatan)”	
	<ul style="list-style-type: none">6. Menampilkan modal form mutu dan harga beras<ul style="list-style-type: none">a. Mutu Beras (tipe text readonly)b. Harga Beras (tipe angka)c. Butir Patah (tipe combobox)d. Kadar Air (tipe combobox)e. Butir Kepala (tipe combobox)f. Butir Mengapur (tipe combobox)g. Benda Asing (tipe combobox)h. Butir Merah (tipe combobox)i. Butir Kuning (tipe combobox)

	<ul style="list-style-type: none">j. Butir Menir (tipe combobox)k. Butir Gabah (tipe combobox)l. Derajat Sosoh (tipe combobox)m. Tombol Hitung
<p>7. Mengisi form mutu dan harga beras</p> <ul style="list-style-type: none">a. Mutu Beras (tipe text) readonlyb. Harga Beras (tipe angka)c. Butir Patah (tipe combobox)d. Kadar Air (tipe combobox)e. Butir Kepala (tipe combobox)f. Butir Mengapur (tipe combobox)g. Benda Asing (tipe combobox)h. Butir Merah (tipe combobox)i. Butir Kuning (tipe combobox)	

<p>j. Butir Menir (tipe combobox)</p> <p>k. Butir Gabah (tipe combobox)</p> <p>l. Derajat Sosoh (tipe combobox)</p>	
8. Klik tombol Hitung	
	<p>9. Menghitung Hasil Penilaian Mutu dengan Rumus Metode SMART $SMART = \sum_{j=1}^k w_j u_i(a_i)$, $i = 1, 2, \dots k$</p> <p>SMART</p>
	<p>10. Menyimpan hasil perhitungan dan inputan harga kedalam database.</p>
	<p>11. Menampilkan flash alert “Perhitungan Mutu dan Harga Beras Berhasil”</p>
12. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW	
Data Kosong	
8. klik tombol Hitung	
	<p>9. Menampilkan span “data tidak boleh kosong”</p>
ALIRAN NORMAL	
Mengatur Ulang Nilai Penilaian dan Harga Beras	

AKTOR	SISTEM
1. Klik menu Mutu dan Harga	
	2. Menampilkan submenu penilaian gabah dan penilaian beras
3. Klik sub menu penilaian beras	
	4. Menampilkan halaman penilaian mutu dan harga beras <ol style="list-style-type: none">Tabel penilaian mutu dan harga beras:<ul style="list-style-type: none">IDNilai MutuMutu BerasHarga BerasAksiTombol Note (Penilaian Mutu dan Harga)Tombol panah melingkar (mereset ulang penilaian danharga)
5. Klik Tombol “Panah Melingkar”	
	6. Merubah nilai mutu dan harga beras menjadi nol pada database

	7. Menampilkan flash alert “Nilai Mutu dan Harga Berhasil di atur ulang”.
8. Klik icon (x) untuk menutup flash alert	

A.8 Use Case Skenario Mengelola Pemasok Gabah

Nomor usecase	07
Nama usecase	Mengelola Pemasok Gabah
Aktor	Pegawai
Deskripsi	Merupakan fitur untuk mengelola data pemasok gabah yang menjual gabahnya untuk selanjutnya dilakukan penilaian untuk setiap pemasok.
Pre-Kondisi	Pengguna berada pada halaman utama admin, dan akan mengelola data pemasok gabah.
Pasca Kondisi	Pengguna telah berada pada halaman utama kelola pemasok gabah, dan telah mengelola data pemasok gabah.
Event Flow	
ALIRAN NORMAL	
Tambah Data Pemasok Gabah	
1. Klik menu Penilaian Padi	

	2. Menampilkan submenu data pemasok gabah dan data pemasok beras
3. Klik sub menu data pemasok gabah	
	4. Menampilkan halaman data pemasok gabah <ol style="list-style-type: none">Tombol “+ Tambah Pemasok”Tabel list pengguna:<ul style="list-style-type: none">IDNamaAlamatBerat(Kg)Tanggal MasukAksiTombol Note (Penilaian pemasok gabah)Tombol pensil (mengubah data)Tombol sampah (menghapus data)
5. Klik Tombol “+ Tambah Pemasok”	
	6. Menampilkan modal form tambah pemasok gabah <ol style="list-style-type: none">Nama/Pabrik (tipe text)

	<ul style="list-style-type: none"> b. Alamat (tipe text) c. Berat(Kg) (tipe angka) d. Tanggal Masuk (tipe tanggal) e. Tombol Submit
<p>7. Mengisi form tambah pemasok gabah</p> <ul style="list-style-type: none"> a. Nama/Pabrik (tipe text) b. Alamat (tipe text) c. Berat(Kg) (tipe angka) d. Tanggal Masuk (tipe tanggal) 	
8. Klik tombol Submit	
	9. Menyimpan data pada database
	10. Menampilkan flash alert “Data Pemasok Gabah Berhasil di Tambahkan”
11. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW	
Data Kosong	
8. klik tombol Submit	
	9. Menampilkan span “data tidak boleh kosong”
ALIRAN NORMAL	

Mengubah Data Pemasok Gabah	
AKTOR	SISTEM
1. Klik menu Penilaian Padi	
	2. Menampilkan submenu data pemasok gabah dan data pemasok beras
3. Klik sub menu data pemasok gabah	
	<p>4. Menampilkan Halaman Kelola Hak Akses</p> <p>a. Tombol “+ Tambah Pemasok”</p> <p>b. Tabel list pengguna:</p> <ul style="list-style-type: none"> • ID • Nama • Alamat • Berat(Kg) • Tanggal Masuk • Aksi <p>c. Tombol Note (Penilaian pemasok gabah)</p> <p>d. Tombol pensil (mengubah data)</p> <p>e. Tombol sampah (menghapus data)</p>
5. Klik Tombol pensil (mengubah data)	

	<p>6. Menampilkan modal form tambah pemasok gabah</p> <ol style="list-style-type: none"> a. Nama/Pabrik (tipe text) b. Alamat (tipe text) c. Berat(Kg) (tipe angka) d. Tanggal Masuk (tipe tanggal) e. Tombol Submit
<p>7. Mengisi form ubah data pemasok gabah</p> <ol style="list-style-type: none"> a. Nama/Pabrik (tipe text) b. Alamat (tipe text) c. Berat(Kg) (tipe angka) d. Tanggal Masuk (tipe tanggal) 	
<p>8. Klik tombol Simpan</p>	
	<p>9. Menyimpan data perubahan pada database</p>
	<p>10. Menampilkan flash alert “Data Pemasok Gabah Berhasil Diubah”</p>
<p>11. Klik icon (x) untuk menutup flash alert</p>	
<p>ALTERNATIF FLOW</p> <p>Data Kosong</p>	

8. Klik tombol simpan	
	9. Menampilkan span “data tidak boleh kosong”
ALIRAN NORMAL	
Menghapus Data Pemasok Gabah	
AKTOR	SISTEM
1. Klik menu Penilaian Padi	
	2. Menampilkan submenu data pemasok gabah dan data pemasok beras
3. Klik sub menu data pemasok gabah	
	<p>4. Menampilkan Halaman Kelola Hak Akses</p> <ol style="list-style-type: none"> a. Tombol “+ Tambah Pemasok” b. Tabel list pengguna: <ul style="list-style-type: none"> • ID • Nama • Alamat • Berat(Kg) • Tanggal Masuk • Aksi c. Tombol Note (Penilaian pemasok gabah) d. Tombol pensil (mengubah data)

	e. Tombol sampah (menghapus data)
5. Klik Tombol sampah (menghapus data)	
	6. menampilkan confirm message “ Hapus Data Pemasok?”
7. Klik OK	
	8. Menghapus data pegawai dalam database
	9. Menampilkan flash alert “Data Pegawai Berhasil di Hapus”
ALIRAN NORMAL	
Batal Menghapus Data Pemasok Gabah	
AKTOR	SISTEM
1. Klik menu Penilaian Padi	
	2. Menampilkan submenu data pemasok gabah dan data pemasok beras
3. Klik sub menu data pemasok gabah	
	4. Menampilkan Halaman Kelola Hak Akses <ul style="list-style-type: none"> a. Tombol “+ Tambah Pemasok” b. Tabel list pengguna: <ul style="list-style-type: none"> • ID

	<ul style="list-style-type: none"> • Nama • Alamat • Berat(Kg) • Tanggal Masuk • Aksi <p>c. Tombol Note (Penilaian pemasok gabah)</p> <p>d. Tombol pensil (mengubah data)</p> <p>e. Tombol sampah (menghapus data)</p>
5. Klik Tombol sampah (menghapus data)	
	6. menampilkan confirm message “ Hapus Data Pemasok?”
7. Klik Cancel	
	8. Menampilkan Halaman Pemasok Gabah

A.9 Use Case Skenario Mengelola Pemasok Beras

Nomor usecase	10
Nama usecase	Mengelola Pemasok Beras
Aktor	Pegawai
Deskripsi	Merupakan fitur untuk mengelola data pemasok beras yang menjual berasnya untuk selanjutnya dilakukan penilaian untuk setiap pemasok.

Pre-Kondisi	Pengguna berada pada halaman utama admin, dan akan mengelola data pemasok beras.
Pasca Kondisi	Pengguna telah berada pada halaman utama kelola pemasok beras, dan telah mengelola data pemasok beras.
Event Flow	
ALIRAN NORMAL	
Tambah Data Pemasok Beras	
1. Klik menu Penilaian Padi	
	2. Menampilkan submenu data pemasok beras dan data pemasok beras
3. Klik sub menu data pemasok beras	
	4. Menampilkan halaman data pemasok beras <ol style="list-style-type: none"> a. Tombol “+ Tambah Pemasok” b. Tabel list pengguna: <ul style="list-style-type: none"> • ID • Nama • Alamat • Berat(Kg) • Tanggal Masuk • Aksi

	<ul style="list-style-type: none"> c. Tombol Note (Penilaian pemasok beras) d. Tombol pensil (mengubah data) e. Tombol sampah (menghapus data)
5. Klik Tombol “+ Tambah Pemasok”	
	<p>6. Menampilkan modal form tambah pemasok beras</p> <ul style="list-style-type: none"> a. Nama/Pabrik (tipe text) b. Alamat (tipe text) c. Berat(Kg) (tipe angka) d. Tanggal Masuk (tipe tanggal) e. Tombol Submit
<p>7. Mengisi form tambah pemasok beras</p> <ul style="list-style-type: none"> a. Nama/Pabrik (tipe text) b. Alamat (tipe text) c. Berat(Kg) (tipe angka) d. Tanggal Masuk (tipe tanggal) 	
8. Klik tombol Submit	
	9. Menyimpan data pada database

	10. Menampilkan flash alert “Data Pemasok Beras Berhasil di Tambahkan”
11. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW	
Data Kosong	
8. klik tombol Submit	
	9. Menampilkan span “data tidak boleh kosong”
ALIRAN NORMAL	
Mengubah Data Pemasok Beras	
AKTOR	SISTEM
1. Klik menu Penilaian Padi	
	2. Menampilkan submenu data pemasok beras dan data pemasok beras
3. Klik sub menu data pemasok beras	
	4. Menampilkan Halaman Kelola Hak Akses <ul style="list-style-type: none"> a. Tombol “+ Tambah Pemasok” b. Tabel list pengguna: <ul style="list-style-type: none"> • ID • Nama

	<ul style="list-style-type: none">• Alamat• Berat(Kg)• Tanggal Masuk• Aksi <ol style="list-style-type: none">c. Tombol Note (Penilaian pemasok beras)d. Tombol pensil (mengubah data)e. Tombol sampah (menghapus data)
5. Klik Tombol pensil (mengubah data)	
	<ol style="list-style-type: none">6. Menampilkan modal form ubah pemasok beras<ol style="list-style-type: none">a. Nama/Pabrik (tipe text)b. Alamat (tipe text)c. Berat(Kg) (tipe angka)d. Tanggal Masuk (tipe tanggal)e. Tombol Submit
7. Mengisi form ubah data pemasok beras <ol style="list-style-type: none">a. Nama/Pabrik (tipe text)b. Alamat (tipe text)c. Berat(Kg) (tipe angka)	

d. Tanggal Masuk (tipe tanggal)	
8. Klik tombol Simpan	
	9. Menyimpan data perubahan pada database
	10. Menampilkan flash alert “Data Pemasok Beras Berhasil Diubah”
11. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW	
Data Kosong	
8. Klik tombol Simpan	
	9. Menampilkan span “data tidak boleh kosong”
ALIRAN NORMAL	
Menghapus Data Pemasok Beras	
AKTOR	SISTEM
1. Klik menu Penilaian Padi	
	2. Menampilkan submenu data pemasok beras dan data pemasok beras
3. Klik sub menu data pemasok beras	
	4. Menampilkan Halaman Kelola Hak Akses

	<ol style="list-style-type: none"> a. Tombol “+ Tambah Pemasok” b. Tabel list pengguna: <ul style="list-style-type: none"> • ID • Nama • Alamat • Berat(Kg) • Tanggal Masuk • Aksi c. Tombol Note (Penilaian pemasok beras) d. Tombol pensil (mengubah data) e. Tombol sampah (menghapus data)
5. Klik Tombol sampah (menghapus data)	
	6. menampilkan confirm message “ Hapus Data Pemasok?”
7. Klik OK	
	8. Menghapus data pemasok dalam database
	9. Menampilkan flash alert “Data Pemasok Berhasil di Hapus”
ALIRAN NORMAL	
Batal Menghapus Data Pemasok Beras	
AKTOR	SISTEM


1. Klik menu Penilaian Padi	
	2. Menampilkan submenu data pemasok beras dan data pemasok beras
3. Klik sub menu data pemasok beras	
	4. Menampilkan Halaman Kelola Hak Akses <ol style="list-style-type: none">Tombol “+ Tambah Pemasok”Tabel list pengguna:<ul style="list-style-type: none">IDNamaAlamatBerat(Kg)Tanggal MasukAksiTombol Note (Penilaian pemasok beras)Tombol pensil (mengubah data)Tombol sampah (menghapus data)
5. Klik Tombol sampah (menghapus data)	

	6. menampilkan confirm message “ Hapus Data Pemasok?”
7. Klik Cancel	
	8. Menampilkan Halaman Pemasok Beras

A.10 Use Case Skenario Membuat Penilaian Pemasok Beras

Nomor usecase	12
Nama usecase	Membuat Penilaian Pemasok Beras
Aktor	Pegawai
Deskripsi	Merupakan fitur untuk membuat penilaian mutu dan juga penilaian metode SMART pada pemasok beras sehingga didapatkan nilai yang dapat dijadikan acuan untuk perangkingan dan penentuan mutu juga harga untuk setiap pemasok beras.
Pre-Kondisi	Pengguna berada pada halaman utama pengguna, dan akan membuat penilaian mutu dan penilaian metode SMART untuk setiap pemasok beras.
Pasca Kondisi	Pengguna telah berada pada halaman utama data pemasok beras, dan membuat penilaian mutu dan penilaian metode SMART untuk setiap pemasok beras.
Event Flow	

ALIRAN NORMAL	
Membuat Penilaian Pemasok Beras	
1. Klik menu Penilaian Padi	
	2. Menampilkan submenu data pemasok beras dan data pemasok beras
3. Klik sub menu data pemasok beras	
	4. Menampilkan halaman data pemasok beras <ol style="list-style-type: none">Tombol “+ Tambah Pemasok”Tabel list pengguna:<ul style="list-style-type: none">IDNamaAlamatBerat(Kg)Tanggal MasukAksiTombol Note (Penilaian pemasok beras)Tombol pensil (mengubah data)Tombol sampah (menghapus data)
5. Klik Tombol Note (Penilaian pemasok beras)	

- 
6. Menampilkan halaman penilaian pemasok beras
- a. ID Pemasok (tipe text) readonly
 - b. Nama Pemasok (tipe text) readonly
 - c. Butir Patah (tipe combobox)
 - d. Kadar Air (tipe combobox)
 - e. Butir Kepala (tipe combobox)
 - f. Hama (tipe combobox)
 - g. Bau Asing (tipe combobox)
 - h. Butir Mengapur (tipe combobox)
 - i. Benda Asing (tipe combobox)
 - j. Butir Merah (tipe combobox)
 - k. Butir Kuning (tipe combobox)
 - l. Butir Menir (tipe combobox)
 - m. Butir Gabah (tipe combobox)

	<p>n. Derajat Sosoh (tipe combobox)</p> <p>o. Tombol Hitung</p>
<p>7. Mengisi form tambah pemasok beras</p> <p>a. Butir Patah (tipe combobox)</p> <p>b. Kadar Air (tipe combobox)</p> <p>c. Butir Kepala (tipe combobox)</p> <p>d. Hama (tipe combobox)</p> <p>e. Bau Asing (tipe combobox)</p> <p>f. Butir Mengapur (tipe combobox)</p> <p>g. Benda Asing (tipe combobox)</p> <p>h. Butir Merah (tipe combobox)</p> <p>i. Butir Kuning (tipe combobox)</p> <p>j. Butir Menir (tipe combobox)</p> <p>k. Butir Gabah (tipe combobox)</p>	

1. Derajat Sosoh (tipe combobox)	
8. Klik tombol Hitung	
	9. Menghitung Hasil Penilaian Mutu dan Perhitungan metode SMART dengan Rumus sebagai $SMART = \sum_{j=1}^k w_j u_i(a_i)$, $i = 1, 2, \dots k$ berikut
	10. Menyimpan hasil perhitungan kedalam database.
11. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW	
Data Kosong	
8. klik tombol Hitung	
	9. Menampilkan span “data tidak boleh kosong”

A.11 Use Case Skenario Melihat Penilaian Gabah

Nomor usecase	13
Nama usecase	Melihat hasil penilaian gabah
Aktor	Pegawai
Deskripsi	Merupakan fitur untuk melihat hasil penilaian mutu dan juga penilaian metode SMART pada setiap pemasok

	gabah sesuai dengan ranking peringkat teratas.
Pre-Kondisi	Pengguna berada pada halaman utama pengguna, dan akan melihat penilaian mutu dan penilaian metode SMART untuk setiap pemasok gabah.
Pasca Kondisi	Pengguna telah berada pada halaman utama hasil penilaian pemasok gabah, dan melihat detail penilaian untuk setiap pemasok gabah.
Event Flow	
ALIRAN NORMAL	
Melihat Penilaian Pemasok Gabah	
1. Klik menu Hasil Penilaian	
	2. Menampilkan submenu penilaian gabah dan penilaian beras
3. Klik sub menu penilaian gabah	
	4. Menampilkan halaman hasil penilaian pemasok gabah <ol style="list-style-type: none"> a. Form Beli Gabah Sebanyak (readonly) b. Tabel hasil penilaian pemasok gabah: <ul style="list-style-type: none"> • Peringkat • Nama

	<ul style="list-style-type: none"> • Berat • Penilaian SMART • Penilaian Mutu • Mutu • Harga Pasaran • Aksi <p>c. Tombol keranjang (membeli gabah)</p> <p>d. Tombol segitiga terbalik (melihat detail penilaian)</p>
5. Klik Tombol Segitiga Terbalik (melihat detail penilaian)	
	<p>6. Menampilkan modal tabel detail penilaian gabah</p> <ul style="list-style-type: none"> a. Kriteria b. Utility c. Normalisasi d. Nilai Sementara
7. Klik icon (x) untuk menutup modal	
	<p>8. Menampilkan halaman hasil penilaian pemasok gabah.</p>

A.12 Use Case Skenario Melihat Penilaian Beras

Nomor usecase	14
Nama usecase	Melihat hasil penilaian beras
Aktor	Pegawai
Deskripsi	Merupakan fitur untuk melihat hasil penilaian mutu dan juga penilaian metode SMART pada setiap pemasok beras sesuai dengan ranking peringkat teratas.
Pre-Kondisi	Pengguna berada pada halaman utama pengguna, dan akan melihat penilaian mutu dan penilaian metode SMART untuk setiap pemasok beras.
Pasca Kondisi	Pengguna telah berada pada halaman utama hasil penilaian pemasok beras, dan melihat detail penilaian untuk setiap pemasok beras.
Event Flow	
ALIRAN NORMAL	
Melihat Penilaian Pemasok Beras	
1. Klik menu Hasil Penilaian	
	2. Menampilkan submenu penilaian beras dan penilaian beras
3. Klik sub menu penilaian beras	

	<p>4. Menampilkan halaman hasil penilaian pemasok beras</p> <ol style="list-style-type: none">a. Form Beli Beras Sebanyak (readonly)b. Tabel hasil penilaian pemasok beras:<ul style="list-style-type: none">• Peringkat• Nama• Berat• Penilaian SMART• Penilaian Mutu• Mutu• Harga Pasaran• Aksic. Tombol keranjang (membeli beras)d. Tombol segitiga terbalik (melihat detail penilaian)
<p>5. Klik Tombol Segitiga Terbalik (melihat detail penilaian)</p>	
	<p>6. Menampilkan modal tabel detail penilaian beras</p> <ol style="list-style-type: none">a. Kriteriab. Utility

	c. Normalisasi d. Nilai Sementara
7. Klik icon (x) untuk menutup modal	
	8. Menampilkan halaman hasil penilaian pemasok beras.

A.13 Use Case Skenario Membuat Pembelian Gabah

Nomor usecase	15
Nama usecase	Membuat Pembelian Gabah
Aktor	Pegawai
Deskripsi	Merupakan fitur untuk membuat pembelian Gabah oleh pegawai, dimana pembelian dilakukan sesuai dengan stok yang diinginkan.
Pre-Kondisi	Pengguna berada pada halaman utama pengguna, dan akan melakukan pembelian Gabah sesuai dengan kriteria yang diinginkan.
Pasca Kondisi	Pengguna telah berada pada halaman utama hasil penilaian pemasok Gabah, dan telah melakukan pembelian Gabah untuk setiap hak akses pengguna.
Event Flow	
ALIRAN NORMAL	
Membuat Pembelian Gabah Sebagian	

1. Klik menu Hasil Penilaian	
	2. Menampilkan submenu penilaian Gabah dan penilaian Gabah
3. Klik sub menu penilaian Gabah	
	4. Menampilkan halaman hasil penilaian pemasok Gabah <ol style="list-style-type: none">Form Beli Gabah Sebanyak (readonly)Tabel hasil penilaian pemasok Gabah:<ul style="list-style-type: none">• Peringkat• Nama• Berat• Penilaian SMART• Penilaian Mutu• Mutu• Harga Pasaran• AksiTombol keranjang (membeli Gabah)Tombol segitiga terbalik (melihat detail penilaian)

<p>5. Klik Tombol keranjang (membeli Gabah)</p>	
	<p>6. Menampilkan modal tabel beli Gabah</p> <ol style="list-style-type: none"> a. ID Pemasok (tipe text Readonly) b. Nama Pemasok (tipe text Readonly) c. Jenis Pembelian (tipe Combobox) d. Banyak Pembelian (tipe angka) e. Harga Beli (tipe angka)
<p>7. Pilih Jenis Pembelian “Beli Sebagian”</p>	
	<p>8. Form Banyak Pembelian Aktif</p>
<p>9. Isi Form Pembelian Gabah</p> <ul style="list-style-type: none"> • Banyak Pembelian (tipe angka) • Harga Beli (tipe angka) 	
<p>10. Klik Tombol Beli</p>	
	<p>11. Mengurangi Stok Gabah yang harus dibeli dan mengurangi stok Gabah pemasok Gabah</p>
	<p>12. Menyimpan dan Merubah data pada database</p>

	13. Menampilkan flash alert “Pembelian Gabah Berhasil”
14. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW Data Kosong	
10. Klik tombol Beli	
	11. Menampilkan span “data tidak boleh kosong”
ALTERNATIF FLOW Pembelian melebihi stok pemasok gabah	
10. Klik tombol Beli	
	11. Menampilkan flash alert “Pembelian melebihi stok pemasok gabah”
ALTERNATIF FLOW Pembelian melebihi stok yang harus dibeli	
10. klik tombol Beli	
	11. Menampilkan flash alert “Pembelian melebihi stok yang harus dibeli”
ALIRAN NORMAL Membuat Pembelian Gabah Semua	
1. Klik menu Hasil Penilaian	

	2. Menampilkan submenu penilaian Gabah dan penilaian Gabah
3. Klik sub menu penilaian Gabah	
	4. Menampilkan halaman hasil penilaian pemasok Gabah <ol style="list-style-type: none">Form Beli Gabah Sebanyak (readonly)Tabel hasil penilaian pemasok Gabah:<ul style="list-style-type: none">PeringkatNamaBeratPenilaian SMARTPenilaian MutuMutuHarga PasaranAksiTombol keranjang (membeli Gabah)Tombol segitiga terbalik (melihat detail penilaian)
5. Klik Tombol keranjang (membeli Gabah)	

	<p>6. Menampilkan modal tabel beli Gabah</p> <ol style="list-style-type: none"> ID Pemasok (tipe text Readonly) Nama Pemasok (tipe text Readonly) Jenis Pembelian (tipe Combobox) Banyak Pembelian (tipe angka) Harga Beli (tipe angka)
7. Pilih Jenis Pembelian “Beli Semua”	
	8. Form Banyak Pembelian Tidak Aktif
<p>9. Isi Form</p> <ul style="list-style-type: none"> Harga Beli (tipe angka) 	
10. Klik Tombol Beli	
	11. Mengurangi Stok Gabah yang harus dibeli dan mengurangi stok Gabah pemasok Gabah
	12. Menyimpan dan Merubah data pada database
	13. Menampilkan flash alert “Pembelian Gabah Berhasil”

14. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW	
Data Kosong	
10. Klik tombol Beli	
	11. Menampilkan span “data tidak boleh kosong”
ALTERNATIF FLOW	
Pembelian melebihi stok yang harus dibeli	
10. Klik tombol Beli	
	11. Menampilkan flash alert “Pembelian melebihi stok yang harus dibeli”

A.14 Use Case Skenario Membuat Pembelian Beras

Nomor usecase	16
Nama usecase	Membuat Pembelian Beras
Aktor	Pegawai
Deskripsi	Merupakan fitur untuk membuat pembelian Beras oleh pegawai, dimana pembelian dilakukan sesuai dengan stok yang diinginkan.
Pre-Kondisi	Pengguna berada pada halaman utama pengguna, dan akan melakukan

	pembelian Beras sesuai dengan kriteria yang diinginkan.
Pasca Kondisi	Pengguna telah berada pada halaman utama hasil penilaian pemasok Beras, dan telah melakukan pembelian Beras untuk setiap hak akses pengguna.
Event Flow	
ALIRAN NORMAL	
Membuat Pembelian Beras Sebagian	
1. Klik menu Hasil Penilaian	
	2. Menampilkan submenu penilaian Beras dan penilaian Beras
3. Klik sub menu penilaian Beras	
	4. Menampilkan halaman hasil penilaian pemasok Beras <ol style="list-style-type: none"> a. Form Beli Beras Sebanyak (readonly) b. Tabel hasil penilaian pemasok Beras: <ul style="list-style-type: none"> • Peringkat • Nama • Berat • Penilaian SMART • Penilaian Mutu

	<ul style="list-style-type: none"> • Mutu • Harga Pasaran • Aksi <p>c. Tombol keranjang (membeli Beras)</p> <p>d. Tombol segitiga terbalik (melihat detail penilaian)</p>
5. Klik Tombol keranjang (membeli Beras)	
	<p>6. Menampilkan modal tabel beli Beras</p> <ul style="list-style-type: none"> a. ID Pemasok (tipe text Readonly) b. Nama Pemasok (tipe text Readonly) c. Jenis Pembelian (tipe Combobox) d. Banyak Pembelian(tipe angka) e. Harga Beli(tipe angka)
7. Pilih Jenis Pembelian “Beli Sebagian”	
	8. Form Banyak Pembelian Aktif
9. Isi Form Pembelian Beras <ul style="list-style-type: none"> • Banyak Pembelian (tipe angka) 	

• Harga Beli (tipe angka)	
10. Klik Tombol Beli	
	11. Mengurangi Stok Beras yang harus dibeli dan mengurangi stok Beras pemasok Beras
	12. Menyimpan dan Merubah data pada database
	13. Menampilkan flash alert “Pembelian Beras Berhasil”
14. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW	
Data Kosong	
10. Klik tombol Beli	
	11. Menampilkan span “data tidak boleh kosong”
ALTERNATIF FLOW	
Pembelian melebihi stok pemasok beras	
10. Klik tombol Beli	
	11. Menampilkan flash alert “Pembelian melebihi stok pemasok beras”
ALTERNATIF FLOW	
Pembelian melebihi stok yang harus dibeli	
10. Klik tombol Beli	

	11. Menampilkan flash alert “Pembelian melebihi stok yang harus dibeli”
ALIRAN NORMAL	
Membuat Pembelian Beras Semua	
1. Klik menu Hasil Penilaian	
	2. Menampilkan submenu penilaian Beras dan penilaian Beras
3. Klik sub menu penilaian Beras	
	4. Menampilkan halaman hasil penilaian pemasok Beras <ol style="list-style-type: none"> a. Form Beli Beras Sebanyak (readonly) b. Tabel hasil penilaian pemasok Beras: <ul style="list-style-type: none"> • Peringkat • Nama • Berat • Penilaian SMART • Penilaian Mutu • Mutu • Harga Pasaran • Aksi

	<ul style="list-style-type: none"> c. Tombol keranjang (membeli Beras) d. Tombol segitiga terbalik (melihat detail penilaian)
5. Klik Tombol keranjang (membeli Beras)	
	<ul style="list-style-type: none"> 6. Menampilkan modal tabel beli Beras <ul style="list-style-type: none"> a. ID Pemasok (tipe text Readonly) b. Nama Pemasok (tipe text Readonly) c. Jenis Pembelian (tipe Combobox) d. Banyak Pembelian(tipe angka) e. Harga Beli(tipe angka)
7. Pilih Jenis Pembelian “Beli Semua”	
	8. Form Banyak Pembelian Tidak Aktif
9. Isi Form <ul style="list-style-type: none"> • Harga Beli (tipe angka) 	
10. Klik Tombol Beli	

	11. Mengurangi Stok Beras yang harus dibeli dan mengurangi stok Beras pemasok Beras
	12. Menyimpan dan Merubah data pada database
	13. Menampilkan flash alert “Pembelian Beras Berhasil”
14. Klik icon (x) untuk menutup flash alert	
ALTERNATIF FLOW Data Kosong	
10. Klik tombol Beli	
	11. Menampilkan span “data tidak boleh kosong”
ALTERNATIF FLOW Pembelian melebihi stok yang harus dibeli	
10. klik tombol Beli	
	11. Menampilkan flash alert “Pembelian melebihi stok yang harus dibeli”

A.15 Use Case Skenario Melihat Riwayat Pembelian Gabah

Nomor usecase	17
Nama usecase	Melihat Riwayat Pembelian Gabah
Aktor	Pegawai

Deskripsi	Merupakan fitur untuk melihat riwayat pembelian Gabah untuk setiap hak akses pengguna.
Pre-Kondisi	Pengguna berada pada halaman utama pengguna, dan akan melihat riwayat pembelian Gabah.
Pasca Kondisi	Pengguna telah berada pada halaman utama riwayat pembelian gabah, dan telah melihat riwayat pembelian setiap hak akses pengguna.
ALIRAN NORMAL	
Melihat Riwayat Pembelian Gabah	
1. Klik menu Riwayat Pembelian	
	2. Menampilkan submenu data Gabah dibeli dan data Beras dibeli
3. Klik sub menu data Gabah dibeli	
	4. Menampilkan halaman riwayat pembelian Gabah <ul style="list-style-type: none"> a. Form Beli Gabah Sebanyak (readonly) b. Tabel riwayat pembelian Gabah: <ul style="list-style-type: none"> • ID • Nama

	<ul style="list-style-type: none"> • Alamat • Berat (Kg) • Harga Beli • Tanggal Masuk • Tanggal Beli • Pembeli
--	--

A.16 *Use Case* Skenario Melihat Riwayat Pembelian Beras

Nomor usecase	18
Nama usecase	Melihat Riwayat Pembelian Beras
Aktor	Pegawai
Deskripsi	Merupakan fitur untuk melihat riwayat pembelian beras untuk setiap hak akses pengguna.
Pre-Kondisi	Pengguna berada pada halaman utama pengguna, dan akan melihat riwayat pembelian beras.
Pasca Kondisi	Pengguna telah berada pada halaman utama riwayat pembelian beras, dan telah melihat riwayat pembelian setiap hak akses pengguna.
ALIRAN NORMAL	
Melihat Riwayat Pembelian Beras	
1. Klik menu Riwayat Pembelian	

	2. Menampilkan submenu data gabah dibeli dan data beras dibeli
3. Klik sub menu data beras dibeli	
	<p>4. Menampilkan halaman riwayat pembelian beras</p> <p>a. Form Beli Beras Sebanyak (readonly)</p> <p>b. Tabel riwayat pembelian beras:</p> <ul style="list-style-type: none"> • ID • Nama • Alamat • Berat (Kg) • Harga Beli • Tanggal Masuk • Tanggal Beli • Pembeli

A.17 Use Case Skenario Keluar

Nomor usecase	17
Nama usecase	Keluar
Aktor	Pegawai

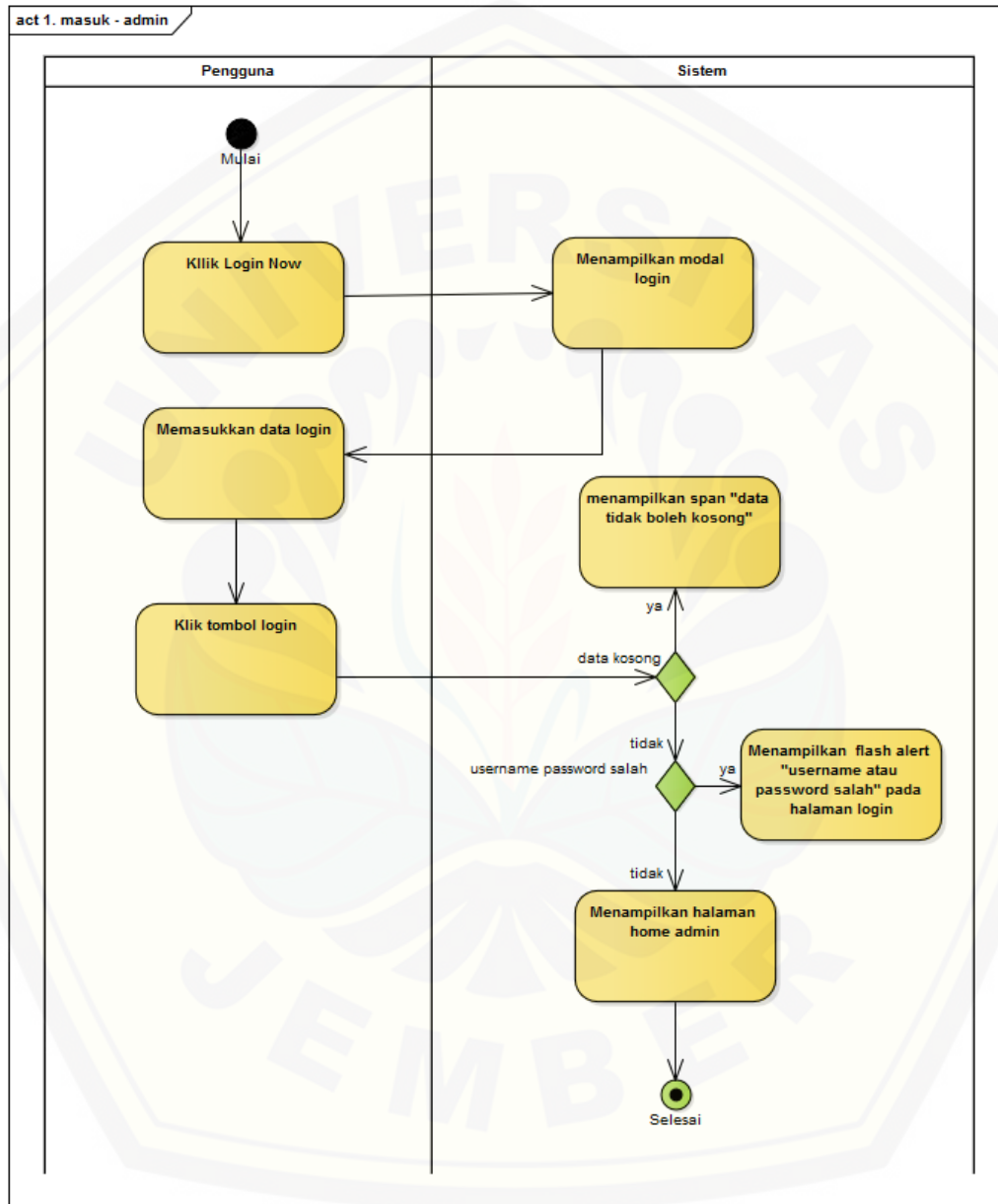
Deskripsi	Merupakan fitur untuk keluar dari sistem sesuai dengan hak akses pengguna
Pre-Kondisi	Pengguna berada pada halaman sistem sesuai dengan hak akses masing – masing pengguna.
Pasca Kondisi	Pengguna berada pada halaman login sistem, dan telah melakukan logout (keluar) dari sistem.
Event Flow	
ALIRAN NORMAL	
AKTOR	SISTEM
1. Klik Menu Nama dan hak akses Pengguna di pojok kanan atas halaman pengguna	
	2. Menampilkan Menu dropdown <ul style="list-style-type: none"> a. Selamat Datang Nama Pengguna b. Logout
3. Klik Menu Logout	
	4. Menghancurkan <i>session</i> yang ada
	5. Menampilkan Halaman Login

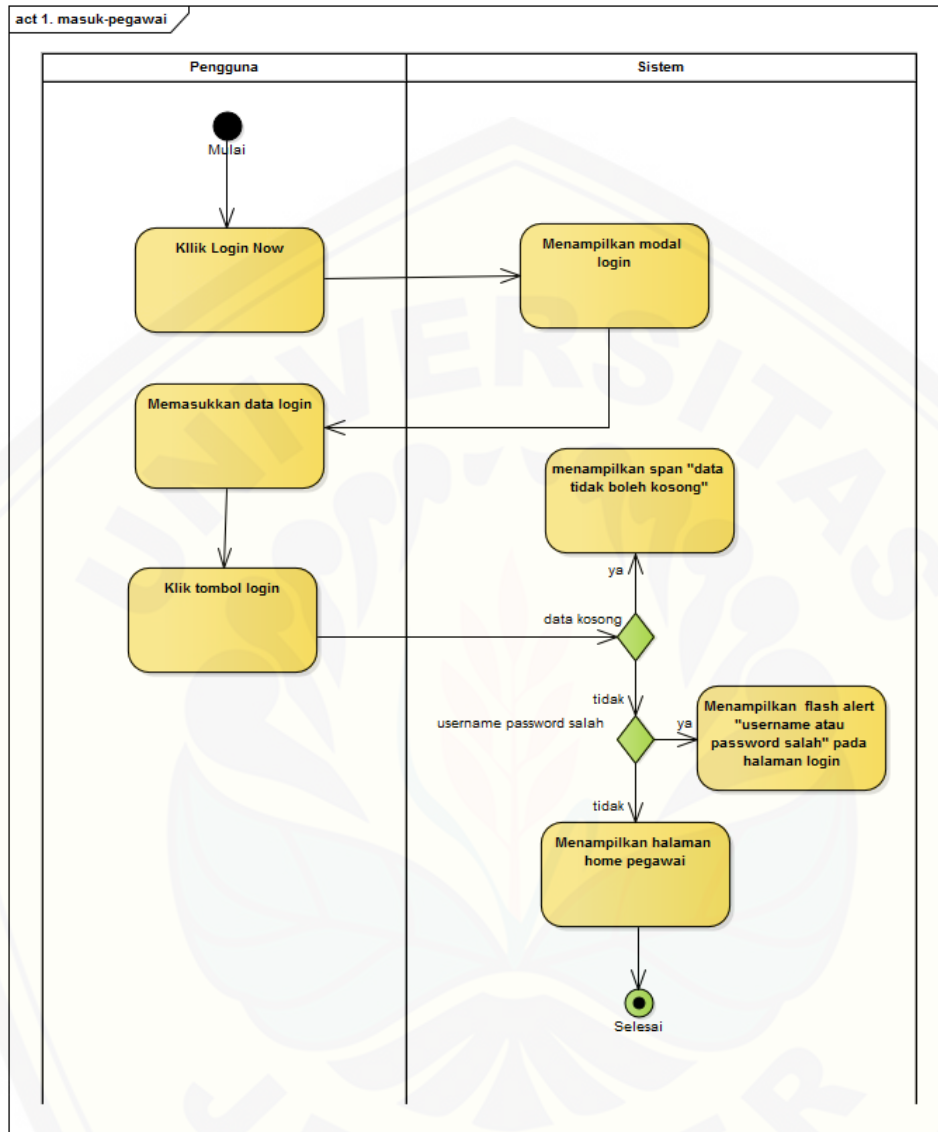
Nomor usecase	17
Nama usecase	Keluar

Aktor	Admin atau pakar
Deskripsi	Merupakan fitur untuk keluar dari sistem sesuai dengan hak akses admin
Pre-Kondisi	Admin berada pada halaman sistem sesuai dengan hak akses masing – masing pengguna.
Pasca Kondisi	Admin berada pada halaman login sistem, dan telah melakukan logout (keluar) dari sistem.
Event Flow	
ALIRAN NORMAL	
AKTOR	SISTEM
1. Klik Menu Nama dan hak akses Pengguna di pojok kanan atas halaman admin	
	2. Menampilkan Menu dropdown <ul style="list-style-type: none"> a. Selamat Datang Nama Pengguna b. Logout
3. Klik Menu Logout	
	4. Menghancurkan session yang ada
	5. Menampilkan Halaman Login

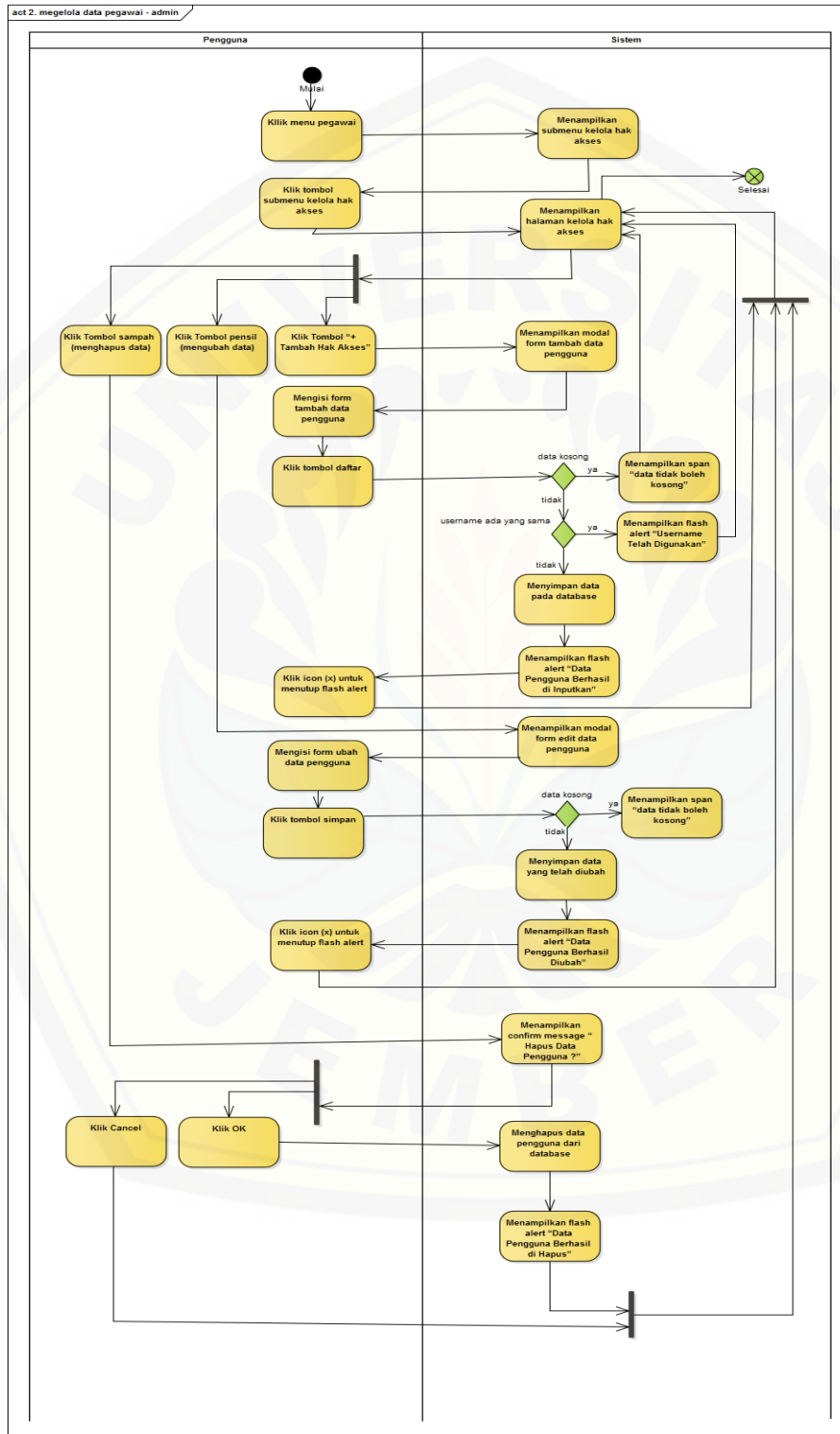
LAMPIRAN B. ACTIVITY DIAGRAM

B.1 Activity Diagram Login (Masuk)

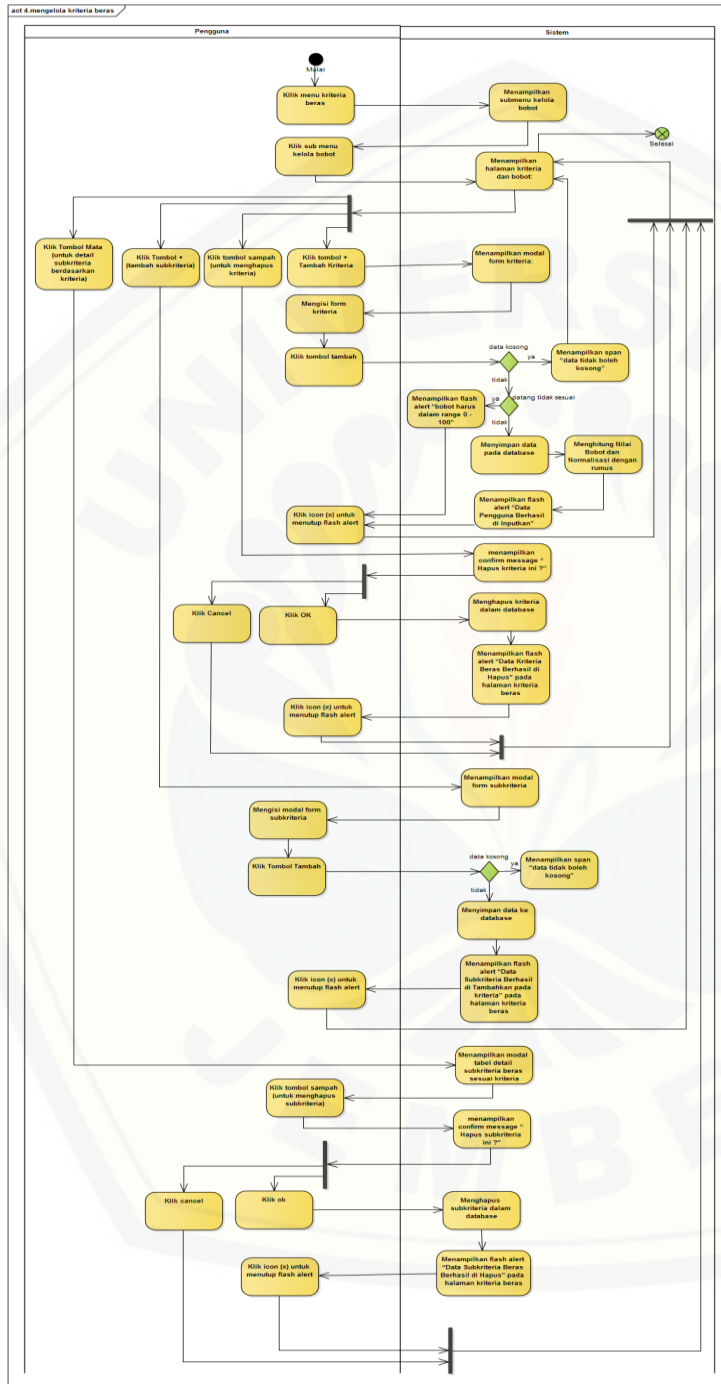




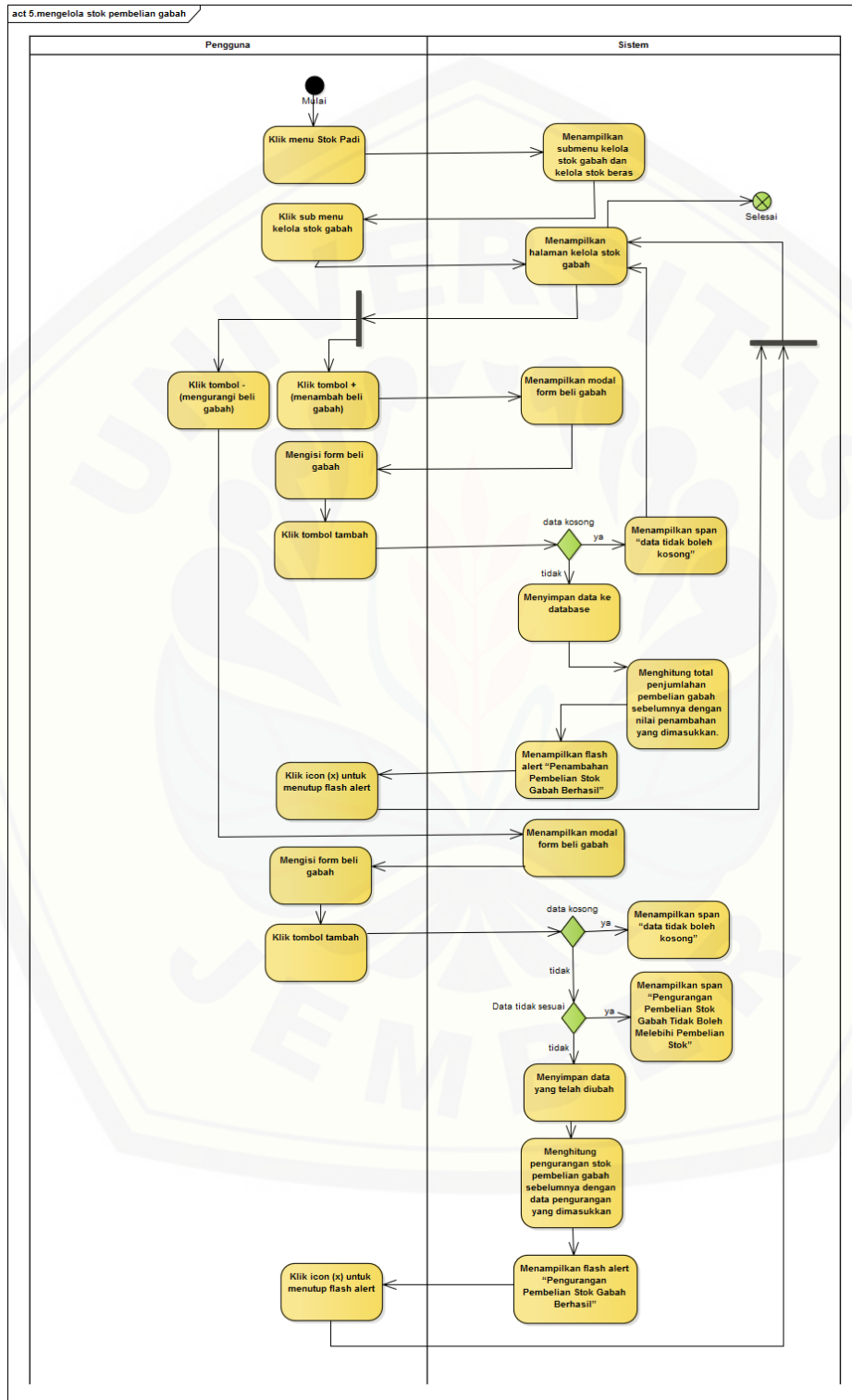
B.2 Activity Diagram Mengelola Data Pengguna



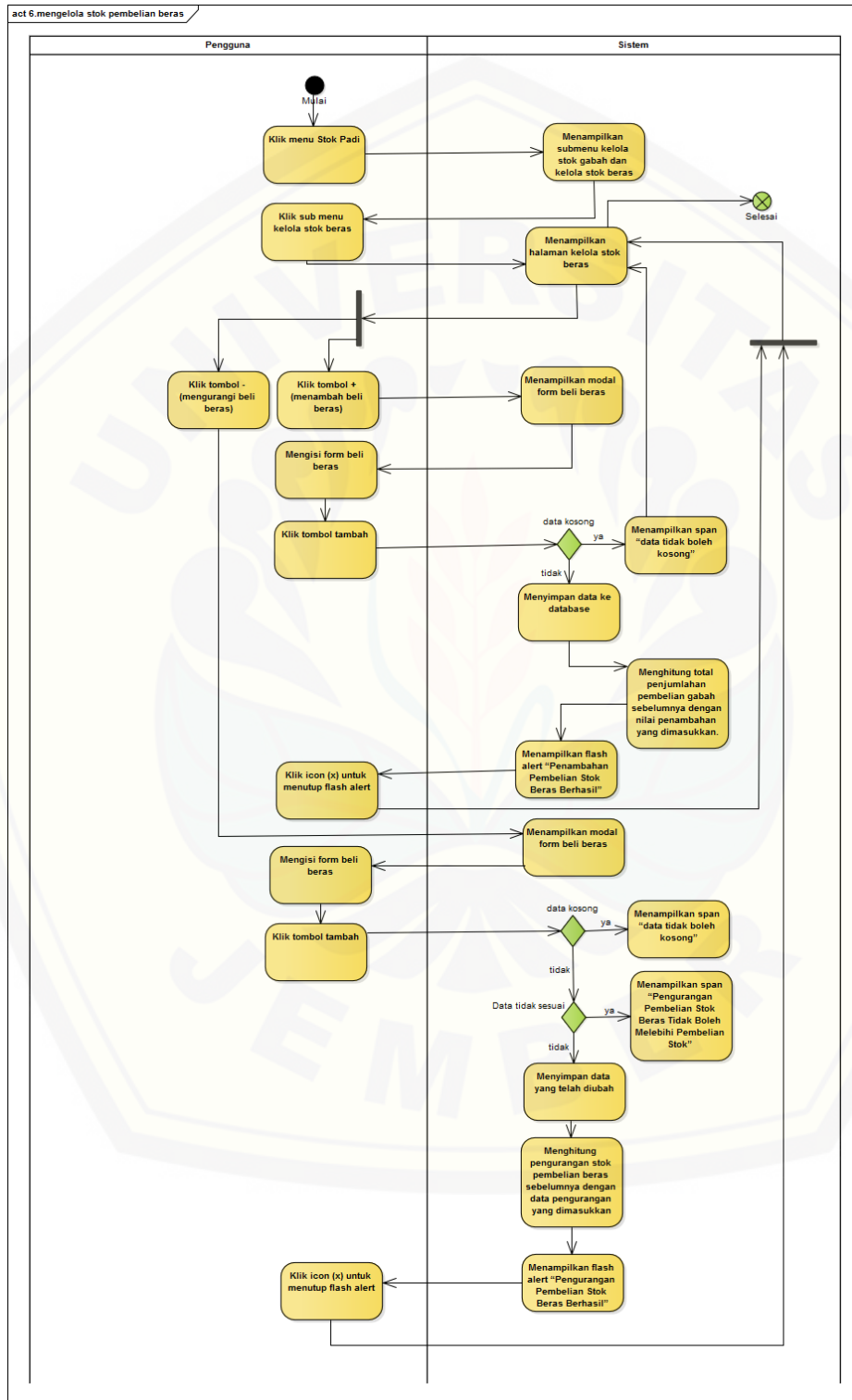
B.3 Activity Diagram Mengolah Kriteria Beras



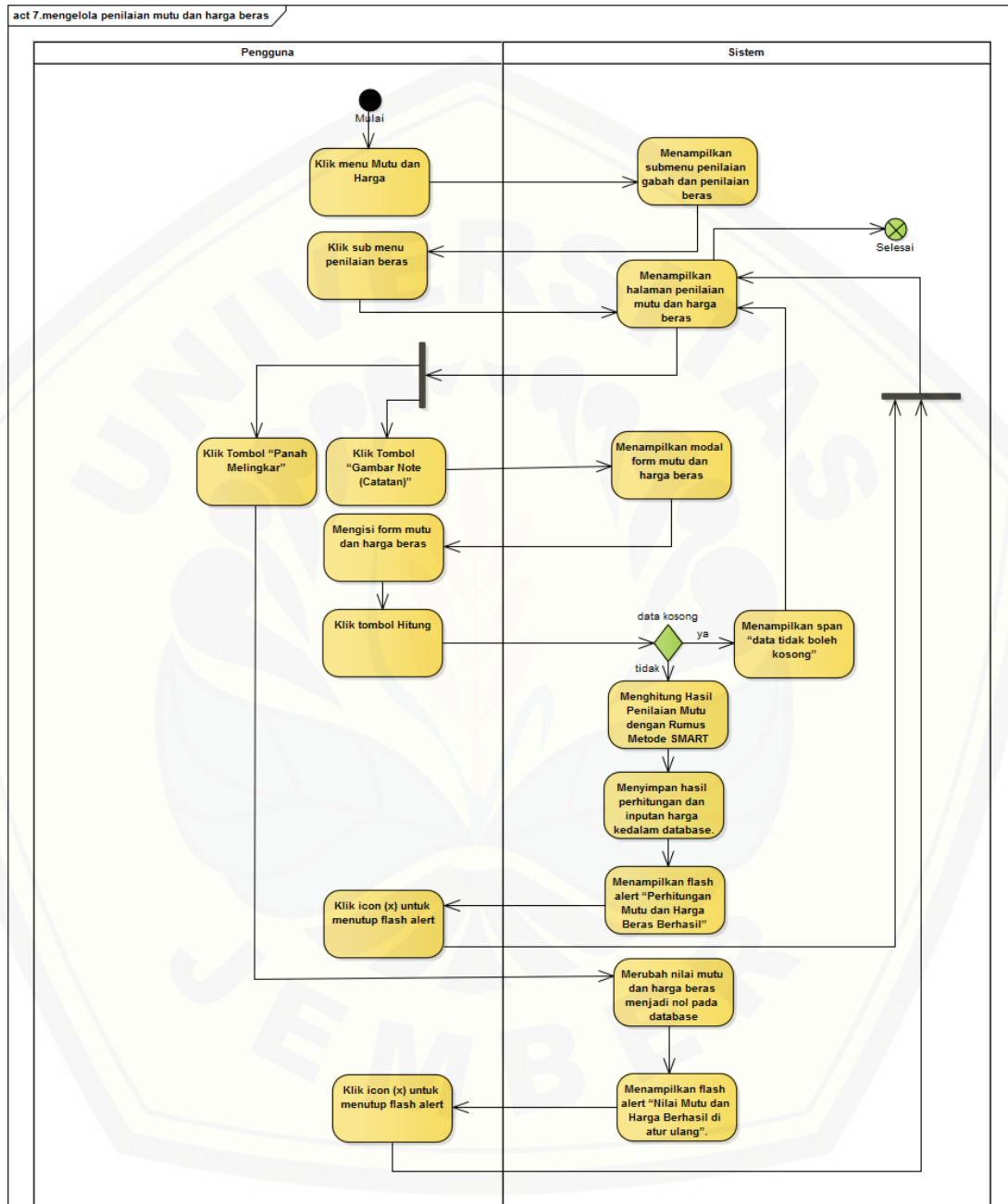
B.4 Activity Diagram Mengelola Stok Pembelian Gabah



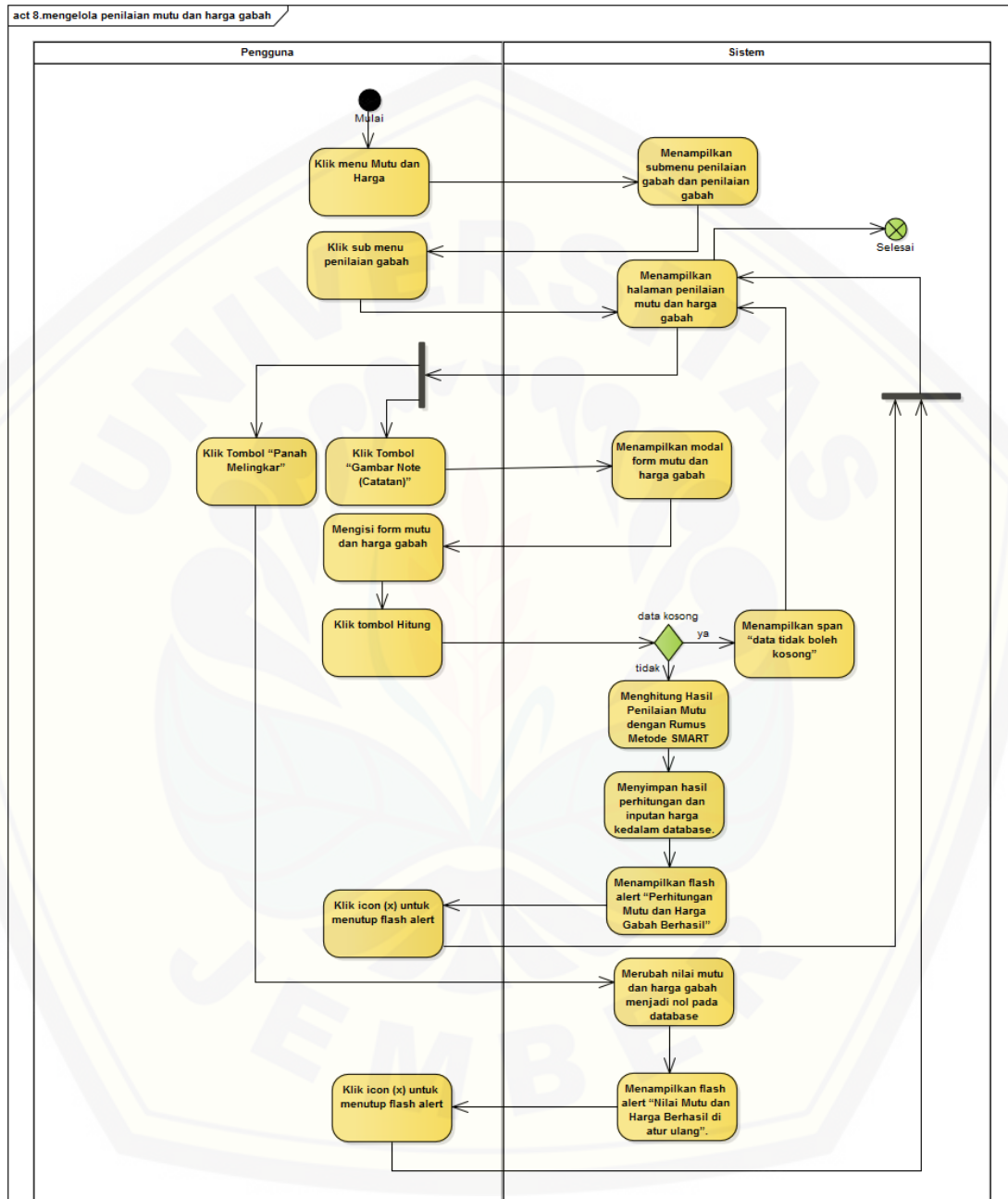
B.5 Activity Diagram Mengelola Stok Pembelian Beras



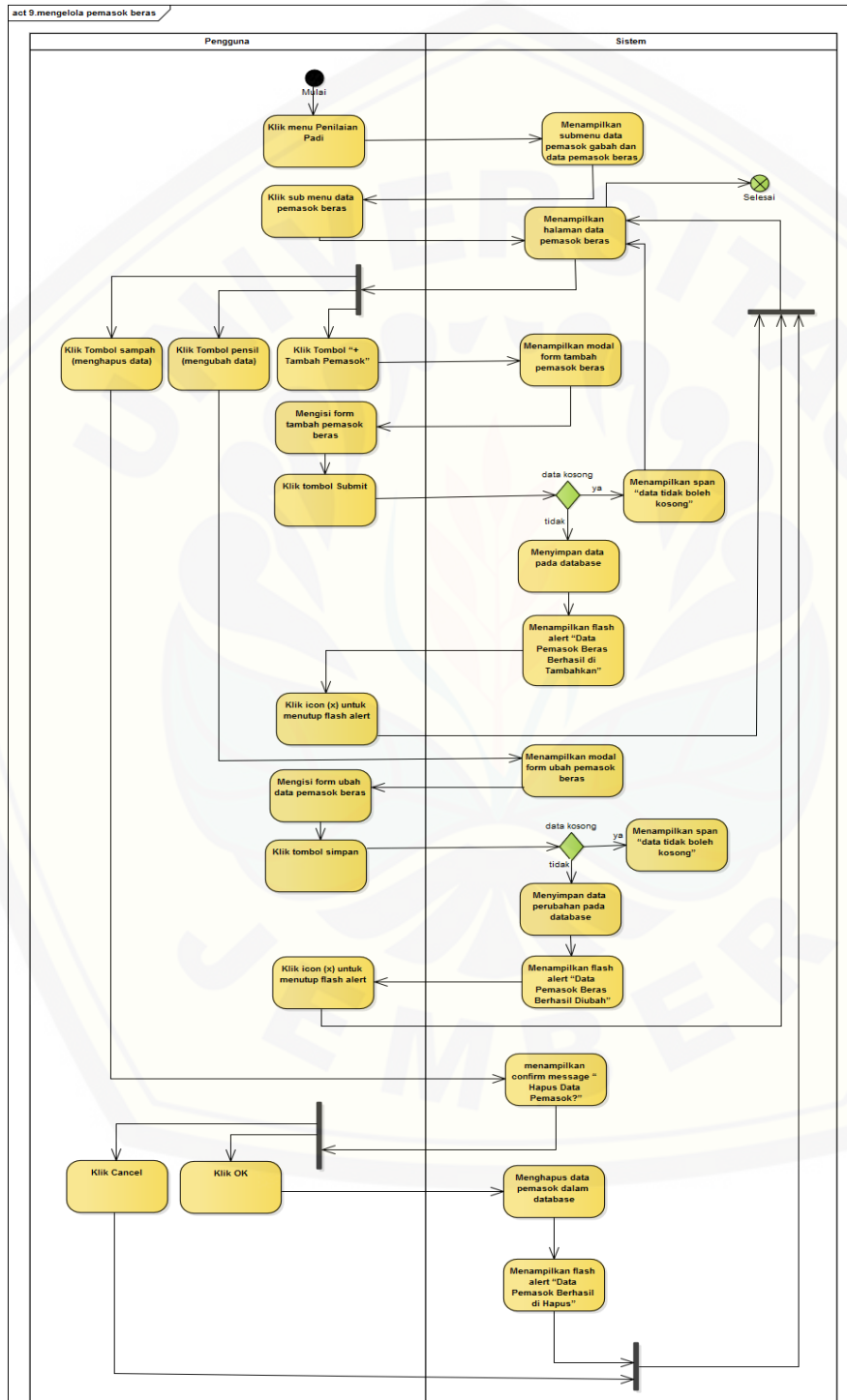
B.6 Activity Diagram Mengelola Penilaian Mutu dan Harga Beras



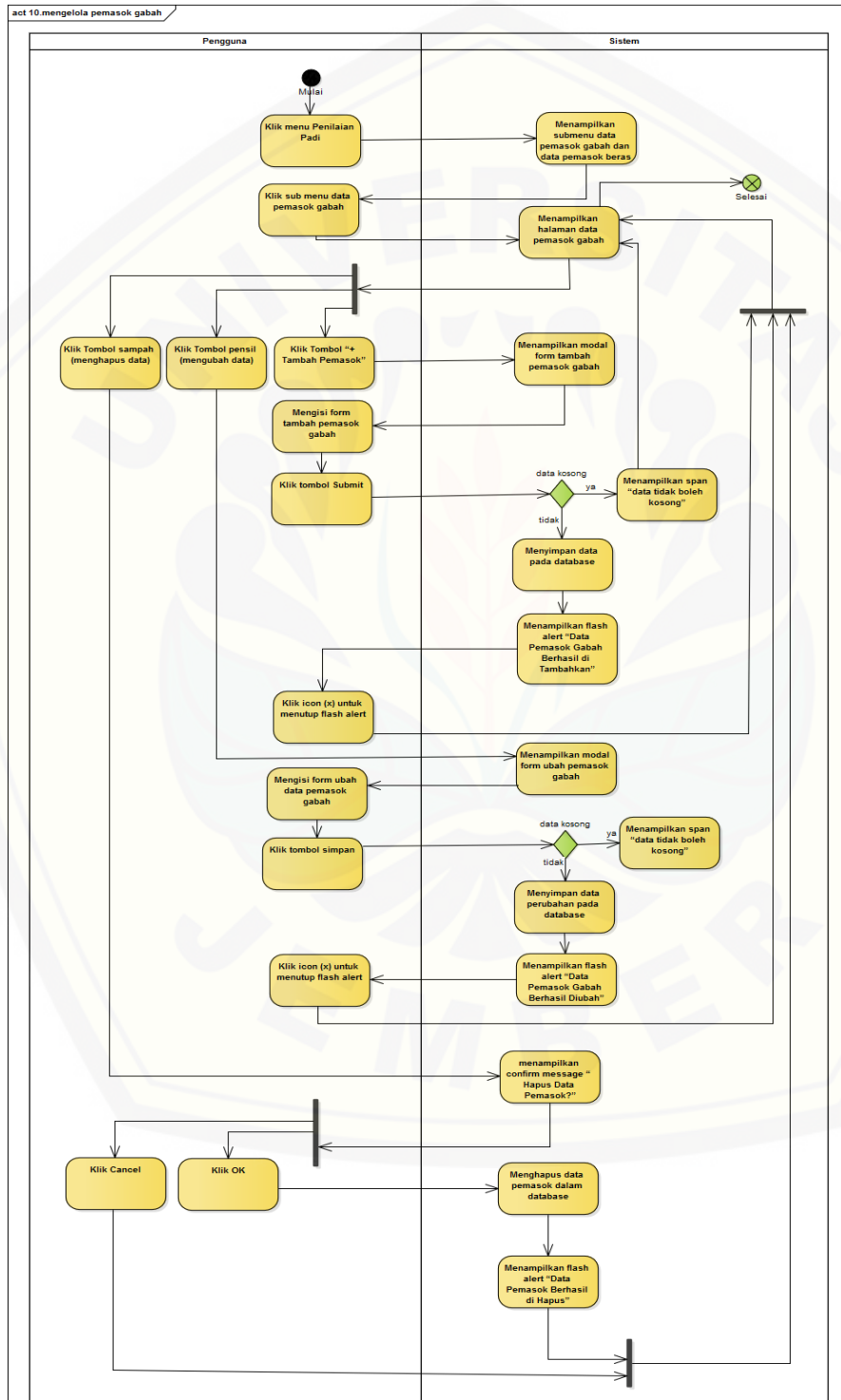
B.7 Activity Diagram Mengelola Penilaian Mutu dan Harga Gabah



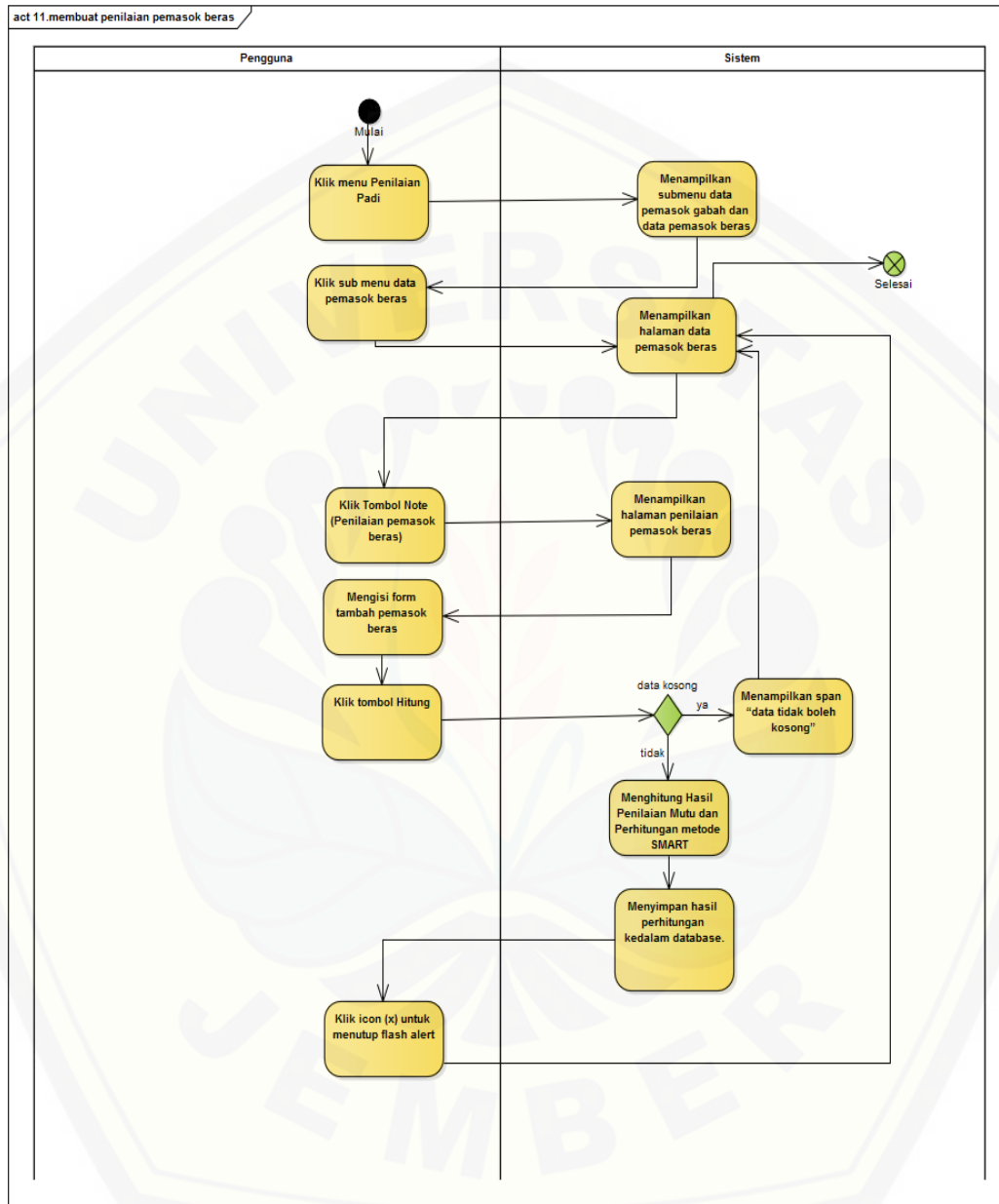
B.8 Activity Diagram Mengelola Pemasok Beras



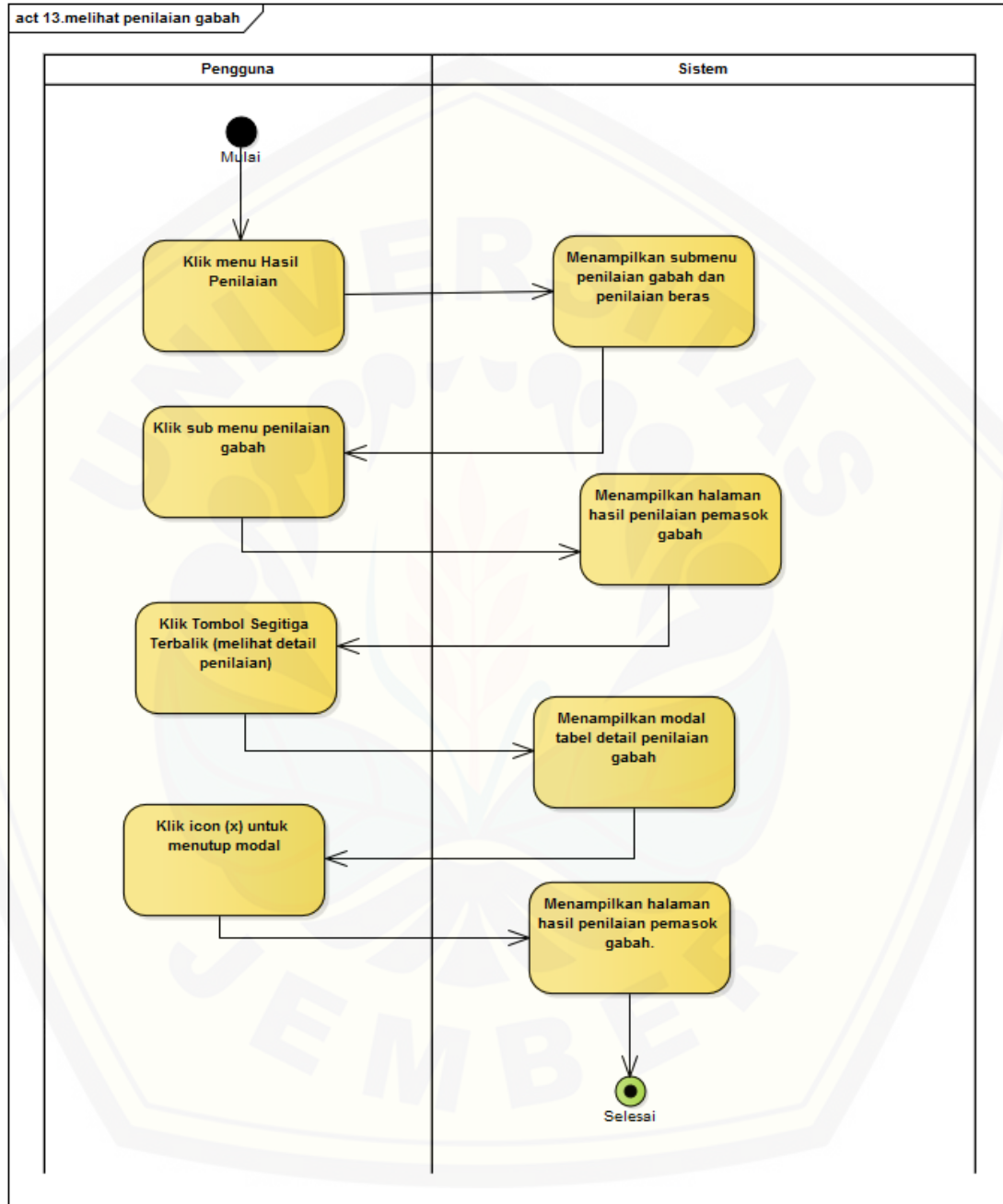
B.9 Activity Diagram Mengelola Pemasok Gabah



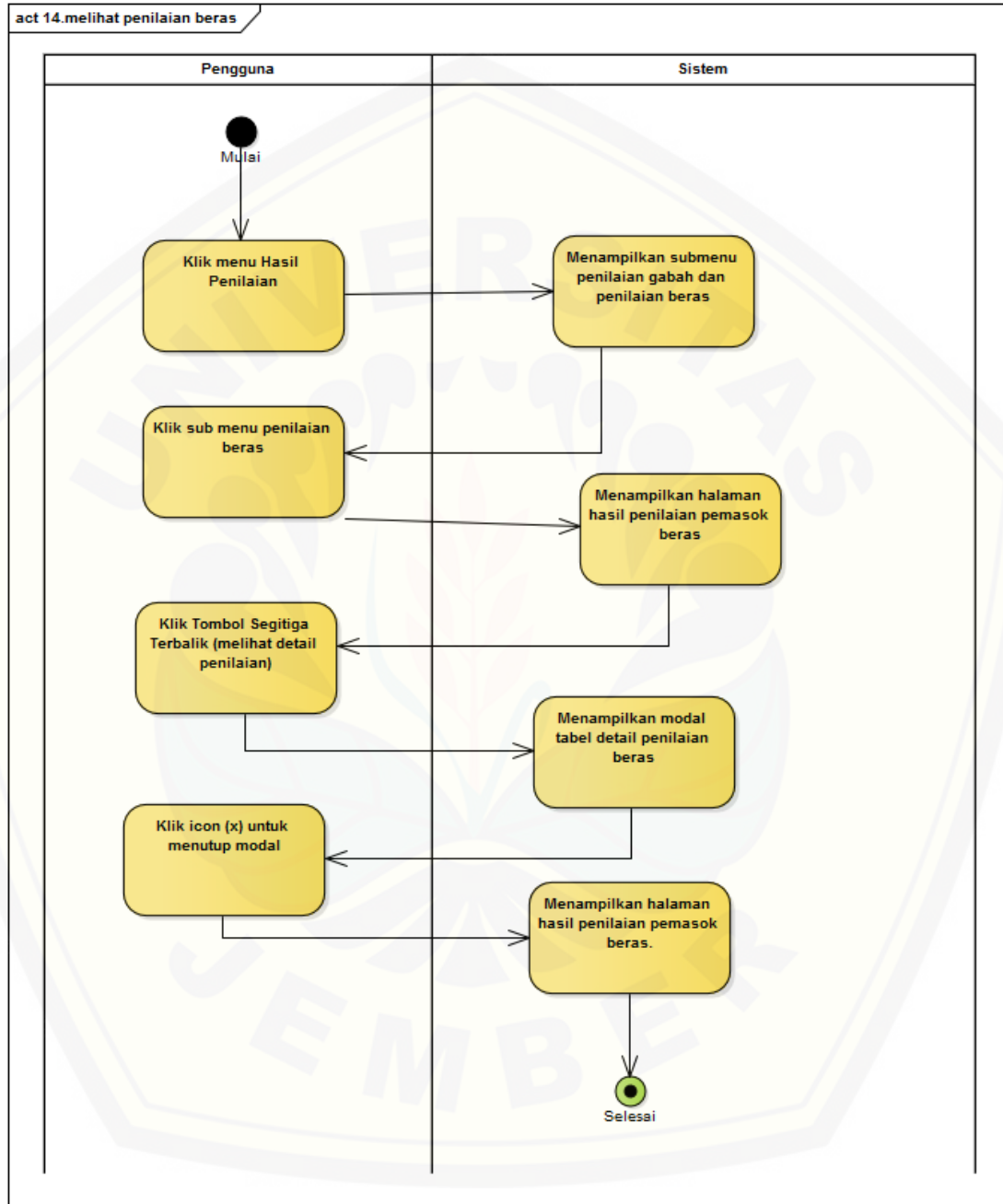
B.10 Activity Diagram Membuat Penilaian Pemasok Beras



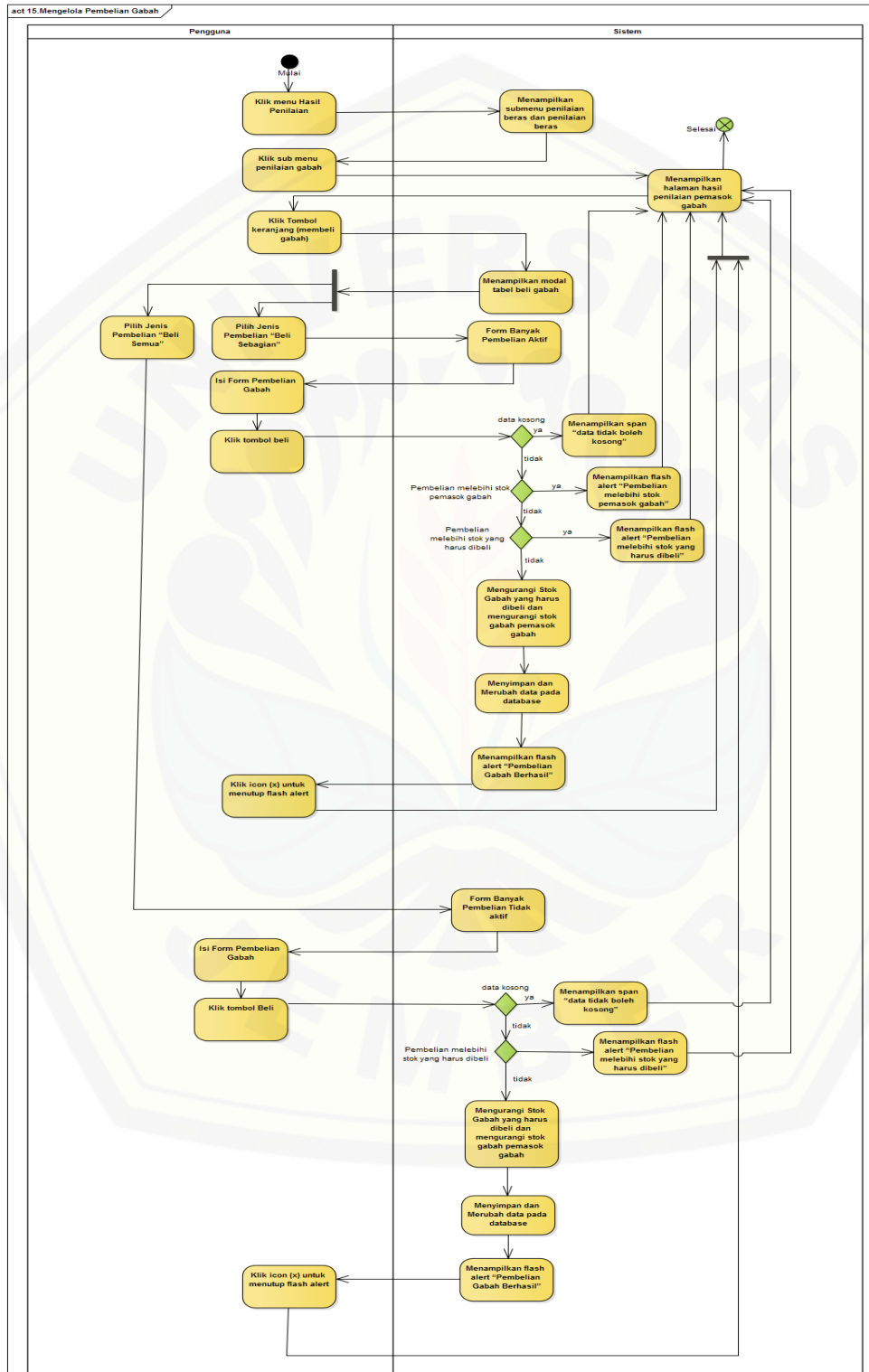
B.11 Activity Diagram Melihat Penilaian Gabah



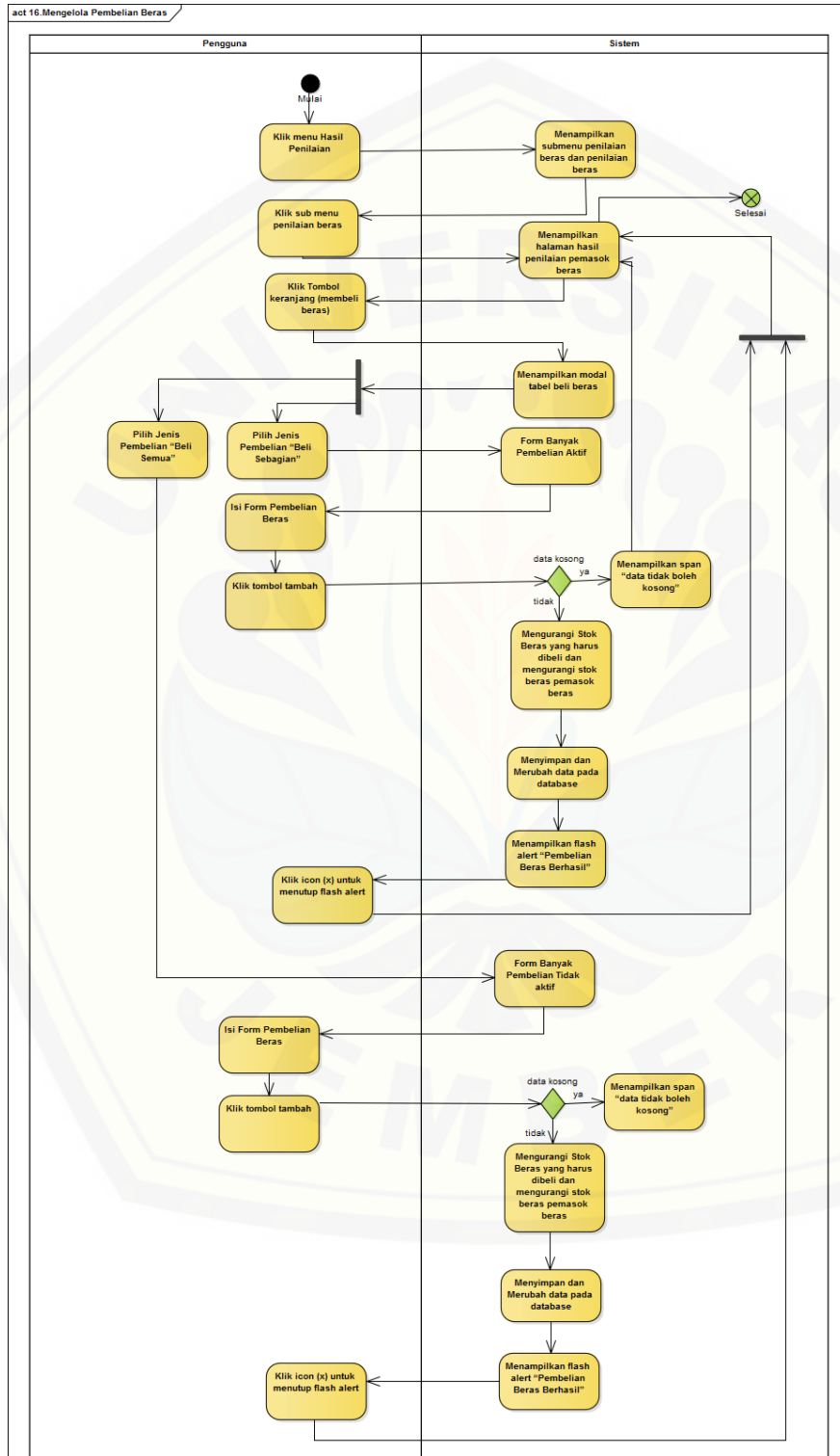
B.12 Activity Diagram Melihat Penilaian Beras



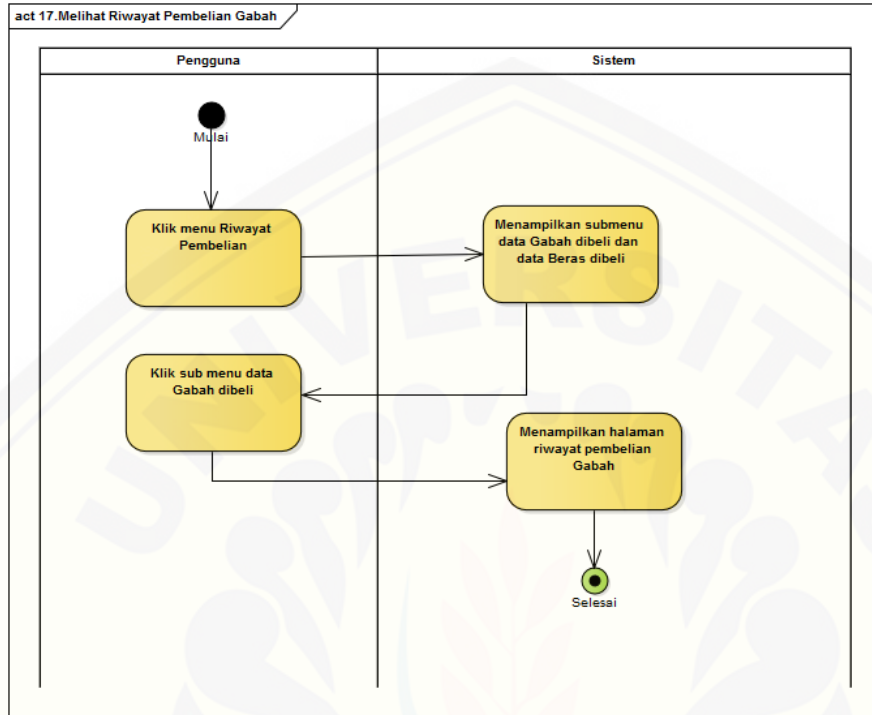
B.13 Activity Diagram Mengelola Pembelian Gabah



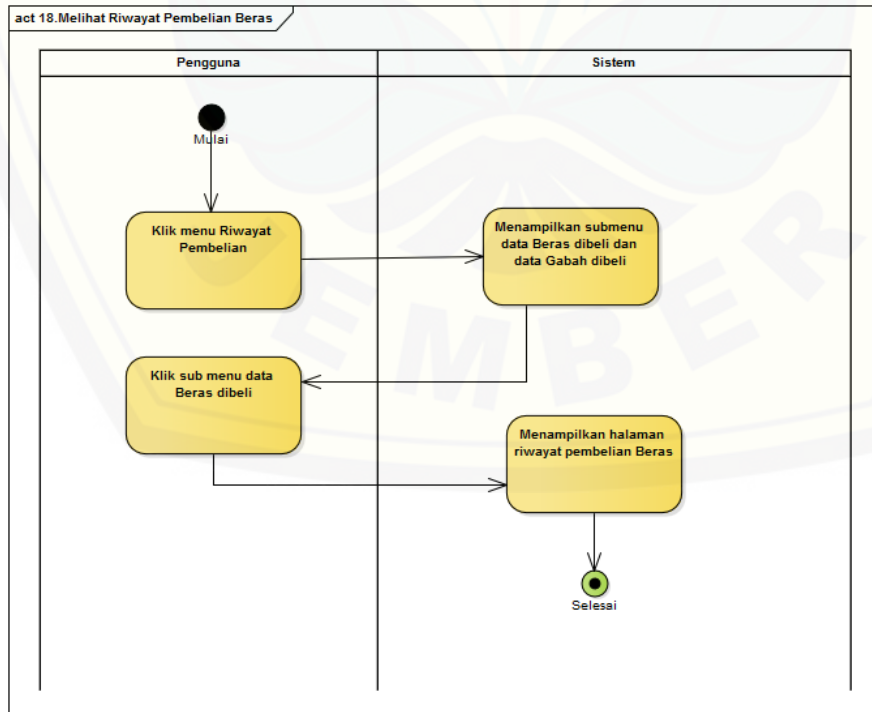
B.14 Activity Diagram Mengelola Pembelian Beras



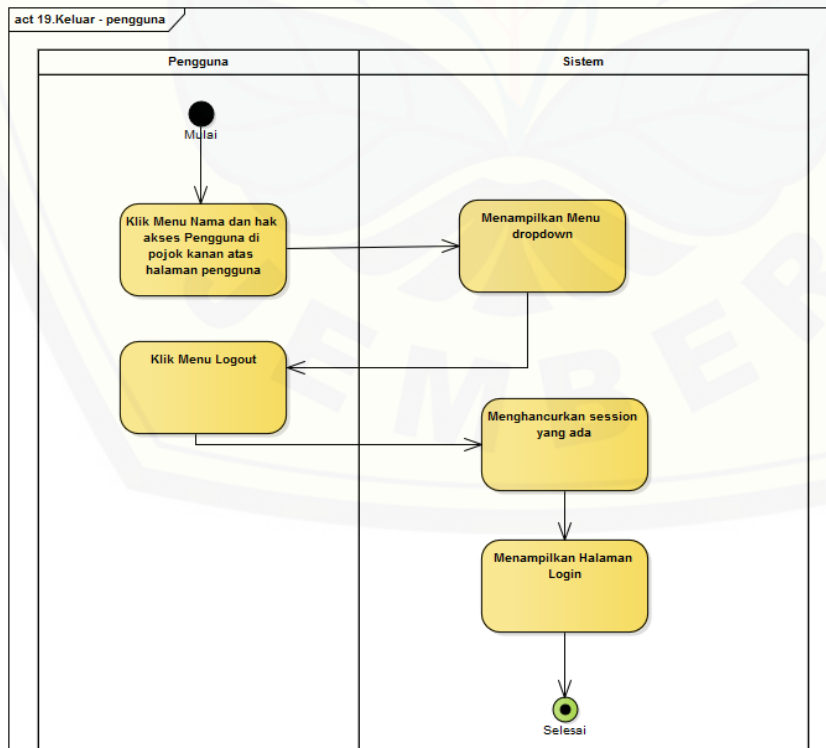
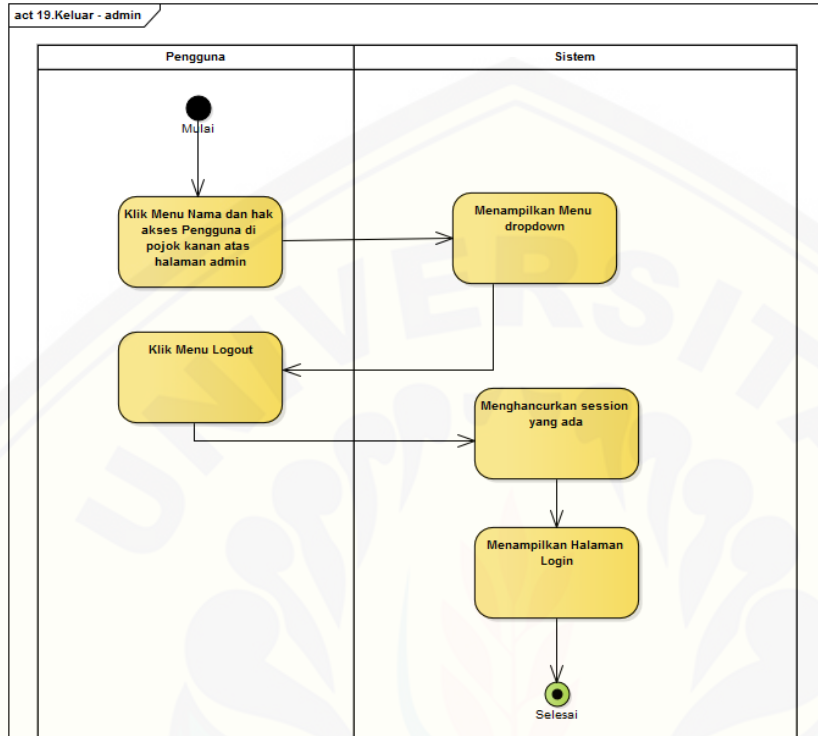
B.15 Activity Diagram Melihat Riwayat Pembelian Gabah



B.16 Activity Diagram Melihat Riwayat Pembelian Beras

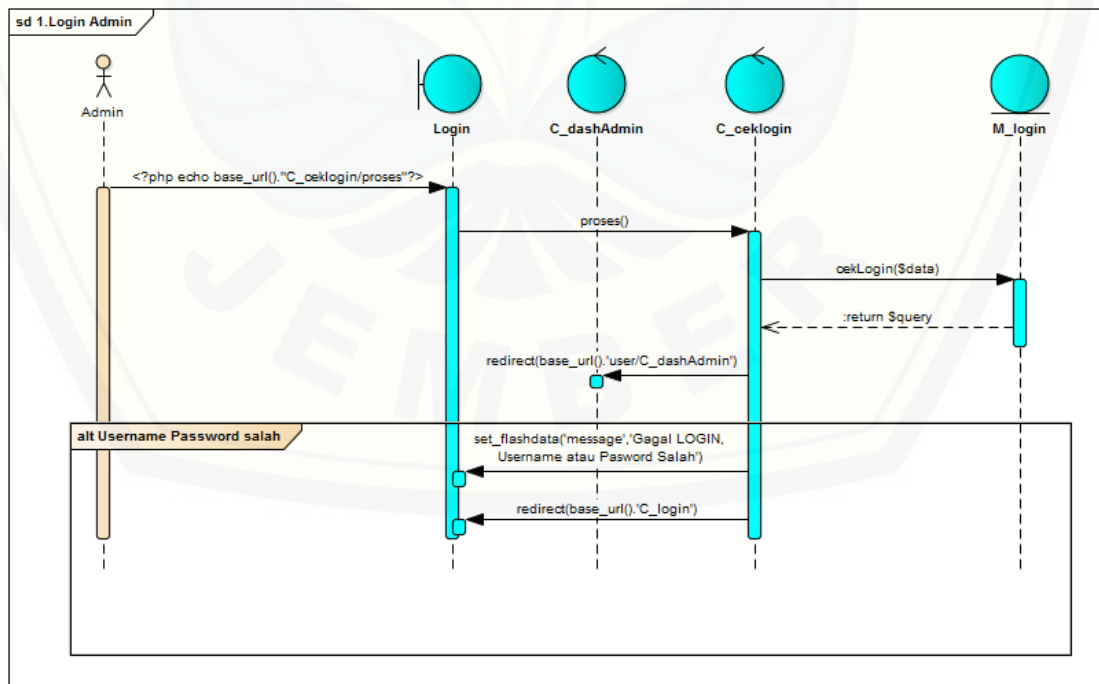
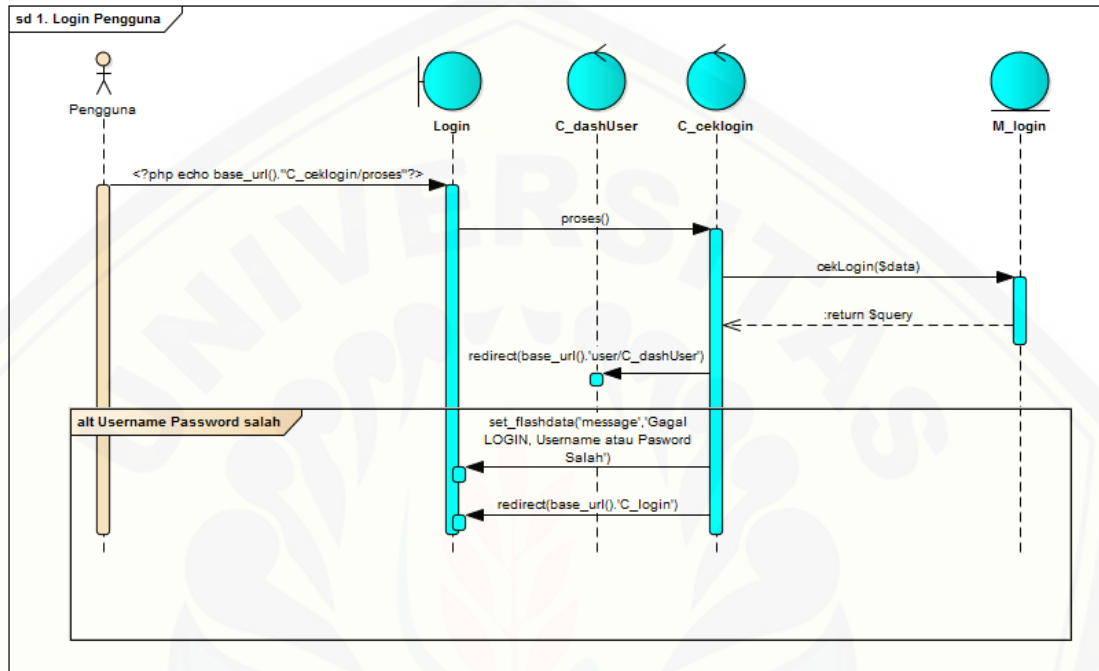


B.17 Activity Diagram Logout (Keluar)

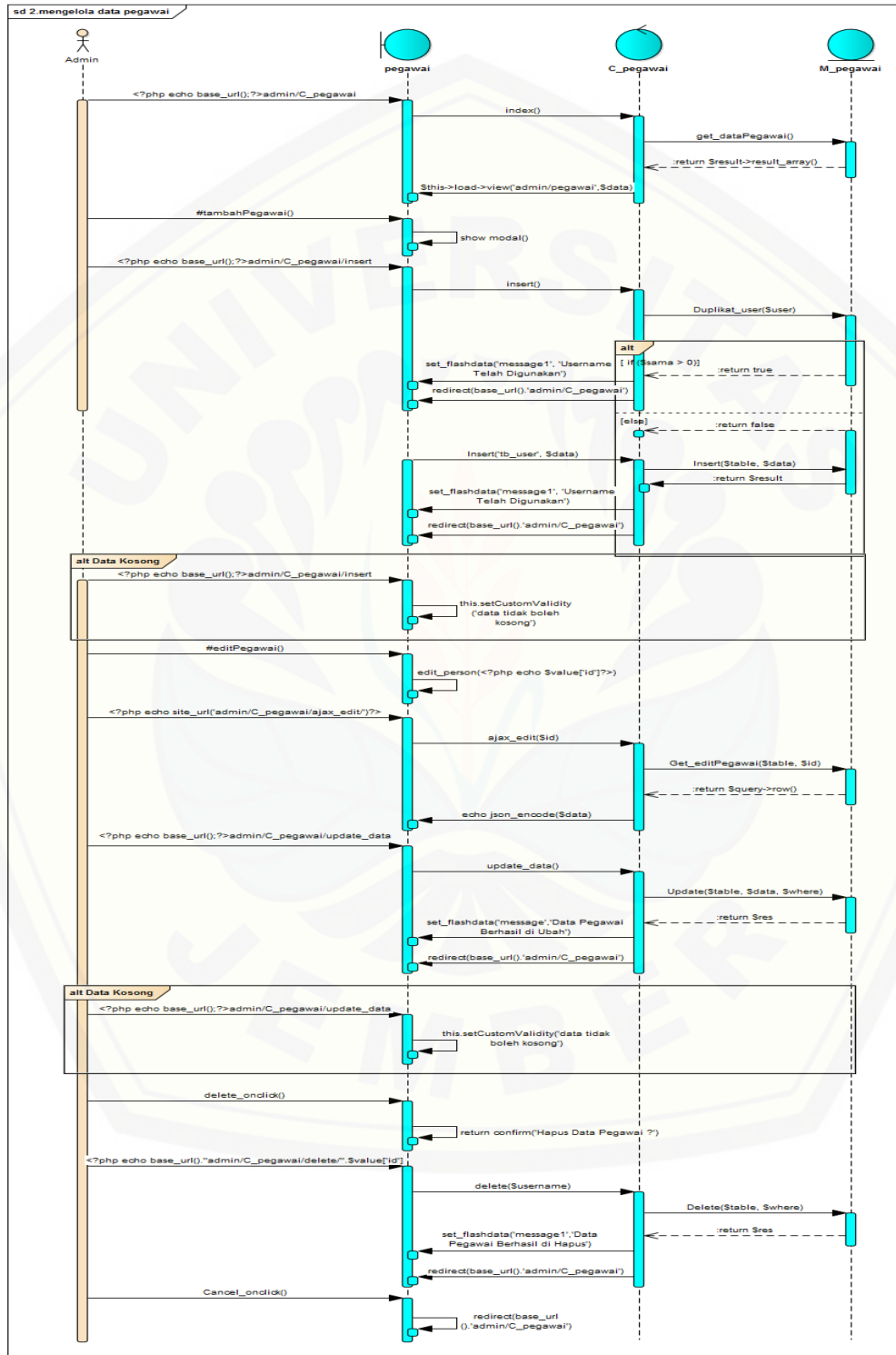


LAMPIRAN C. SEQUENCE DIAGRAM

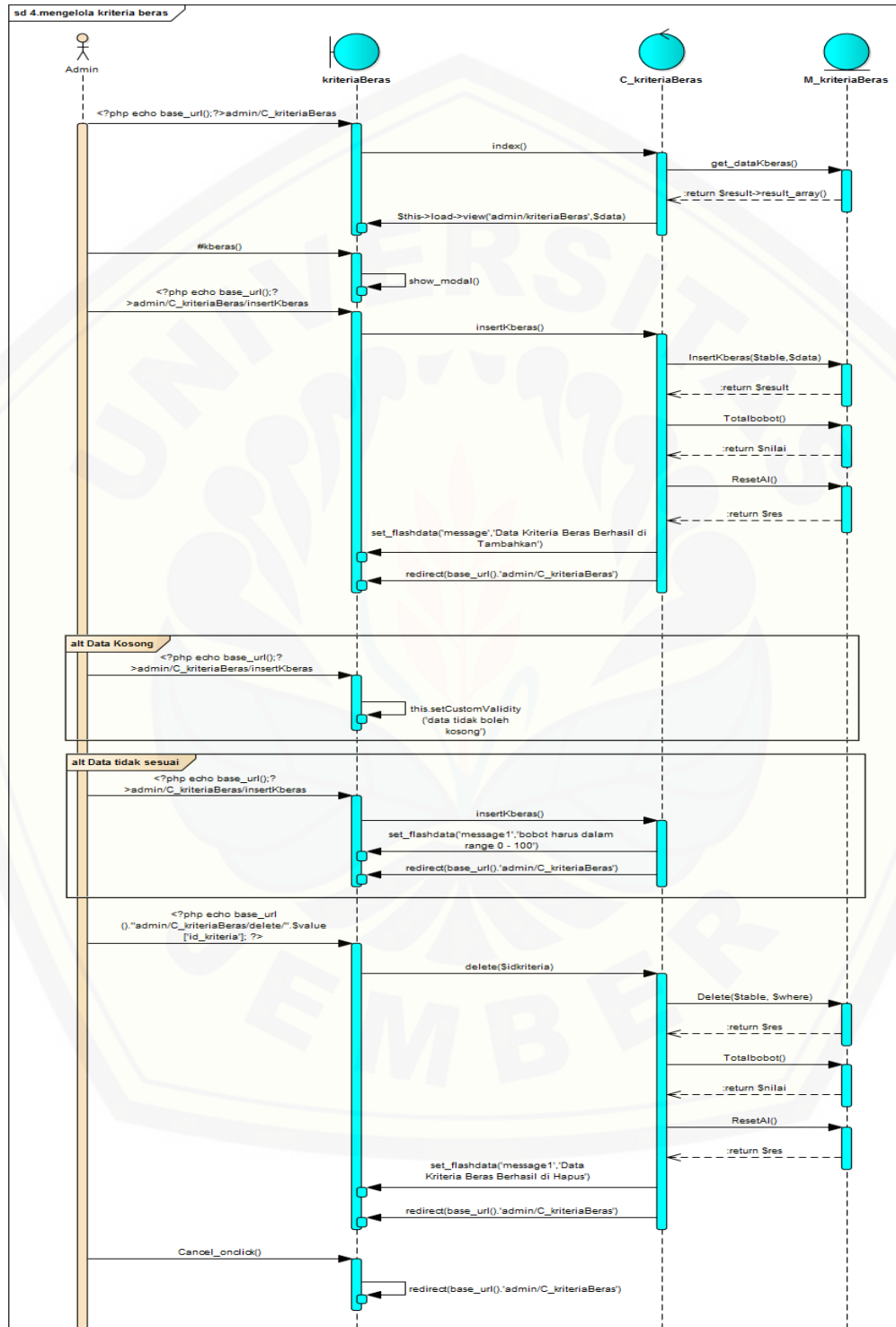
C.1 Sequence Diagram Login (Masuk)



C.2 Sequence Diagram Mengelola Data Pengguna

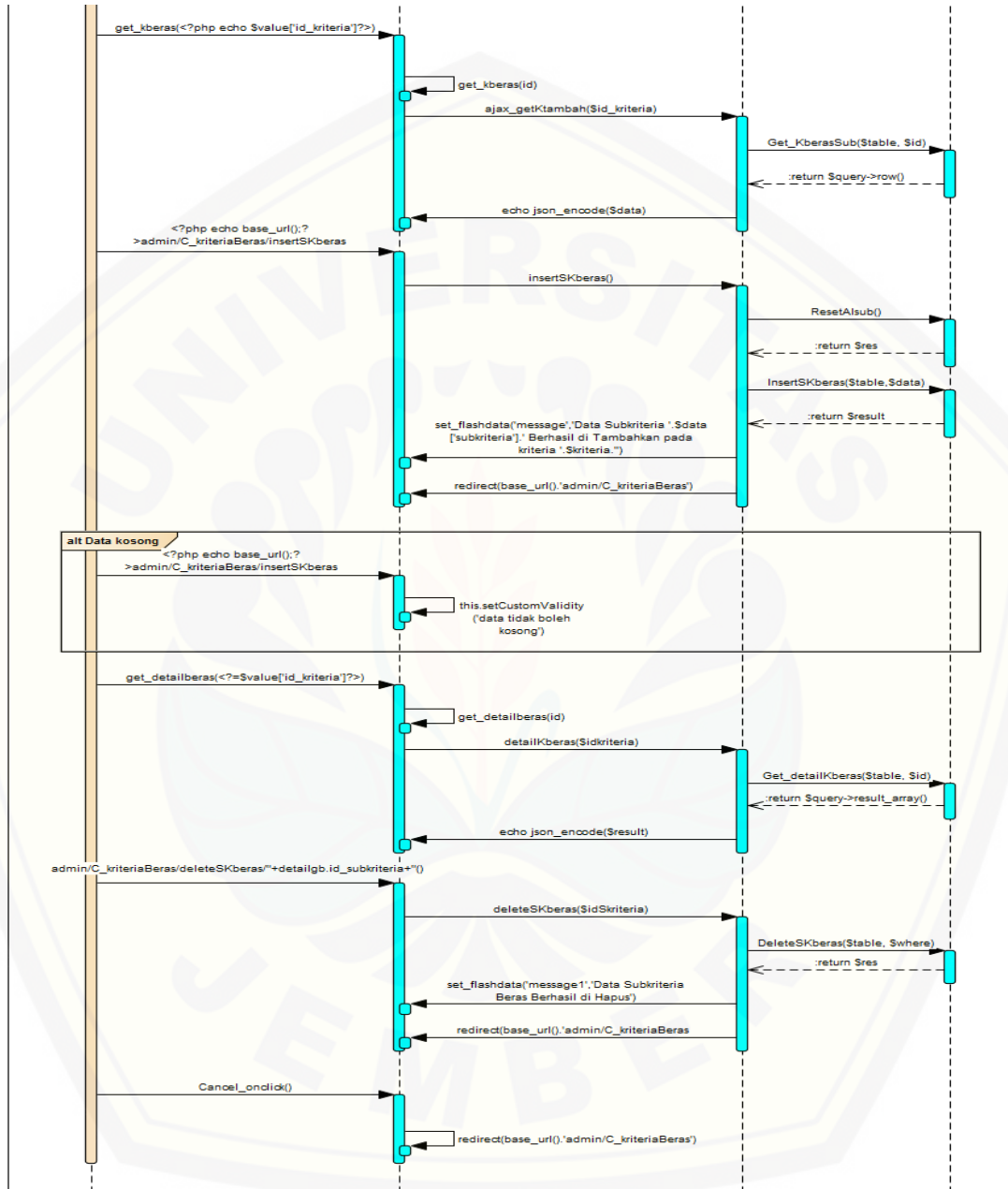


C.3 Sequence Diagram Mengolah Kriteria Beras

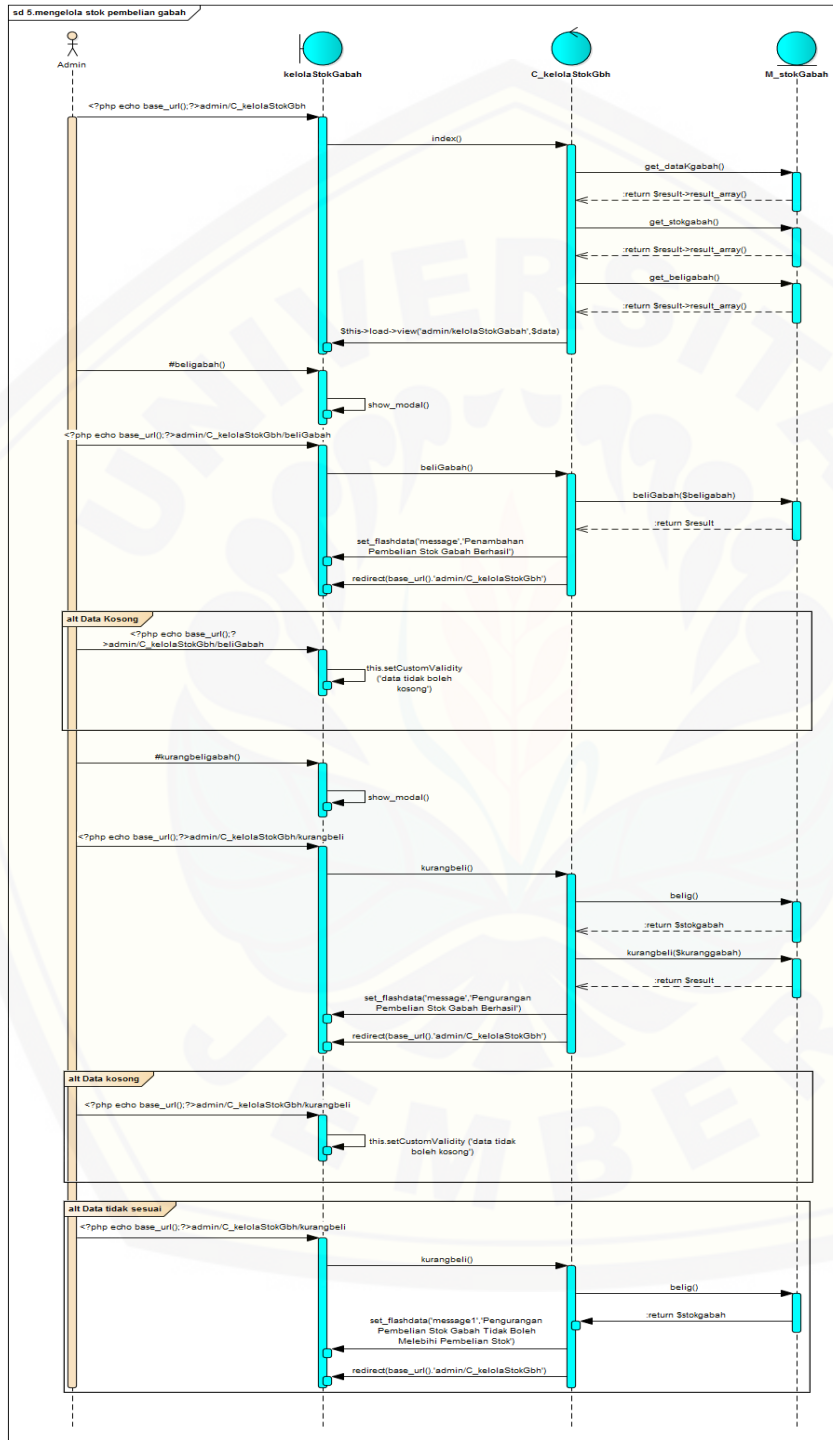


Dilanjutkan

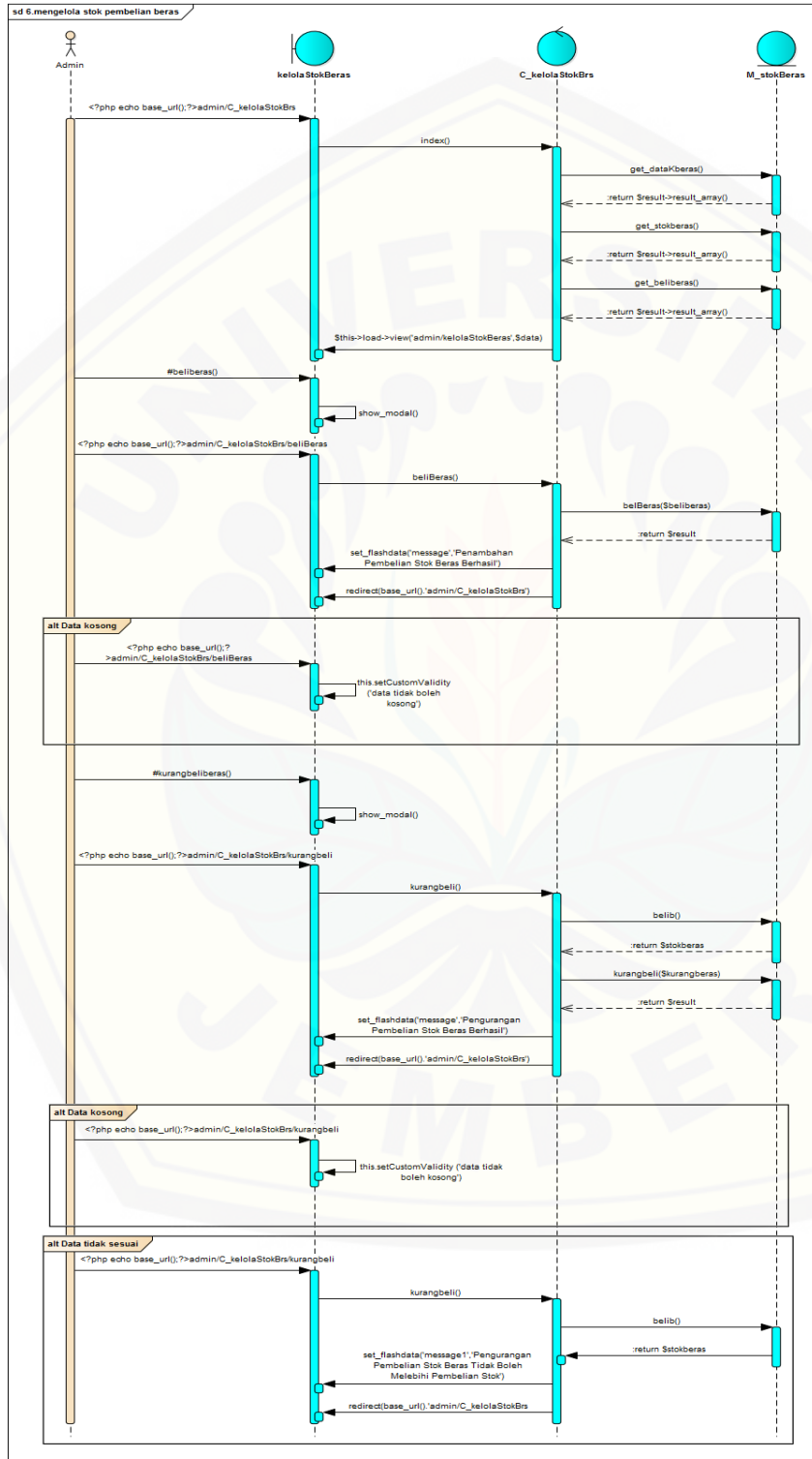
Lanjutan



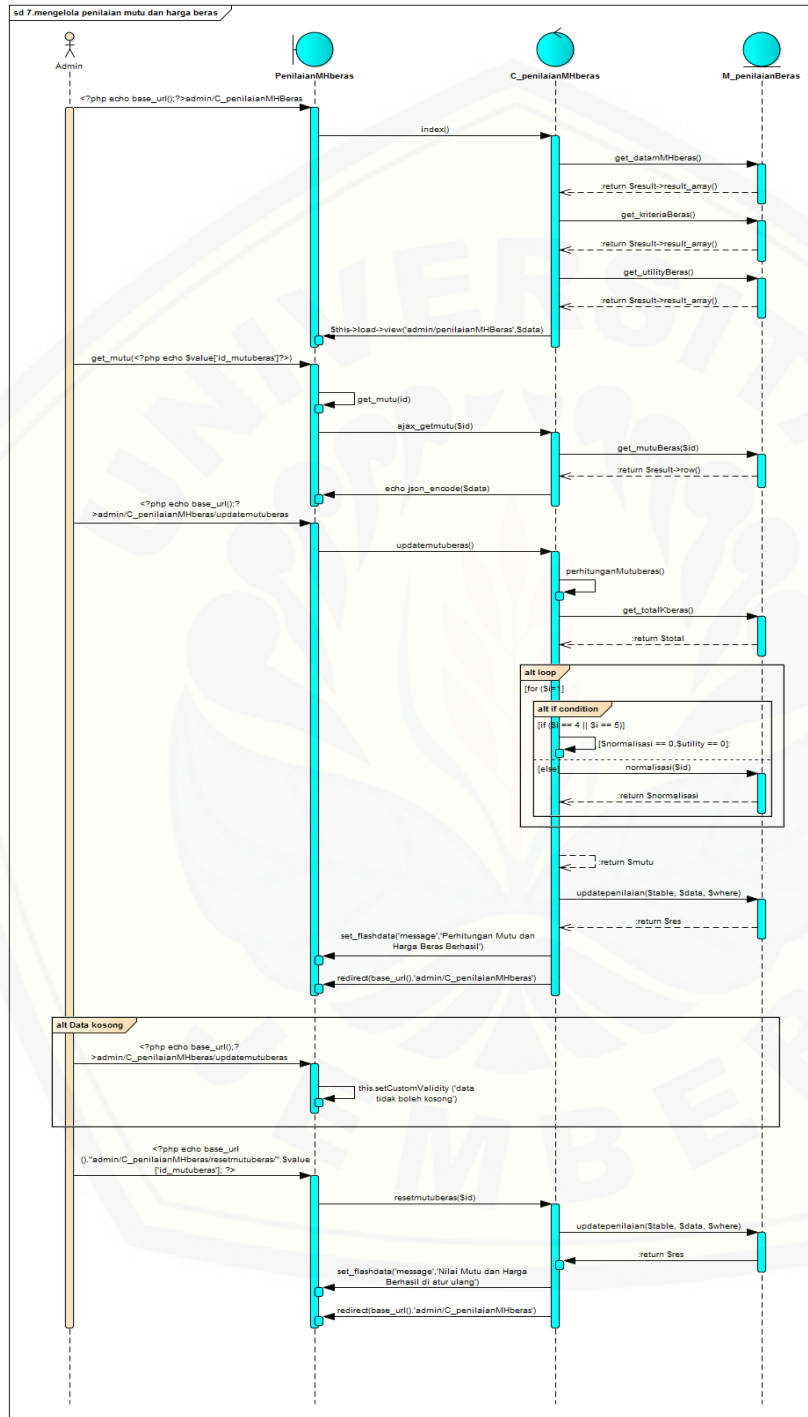
C.4 Sequence Diagram Mengelola Stok Pembelian Beras



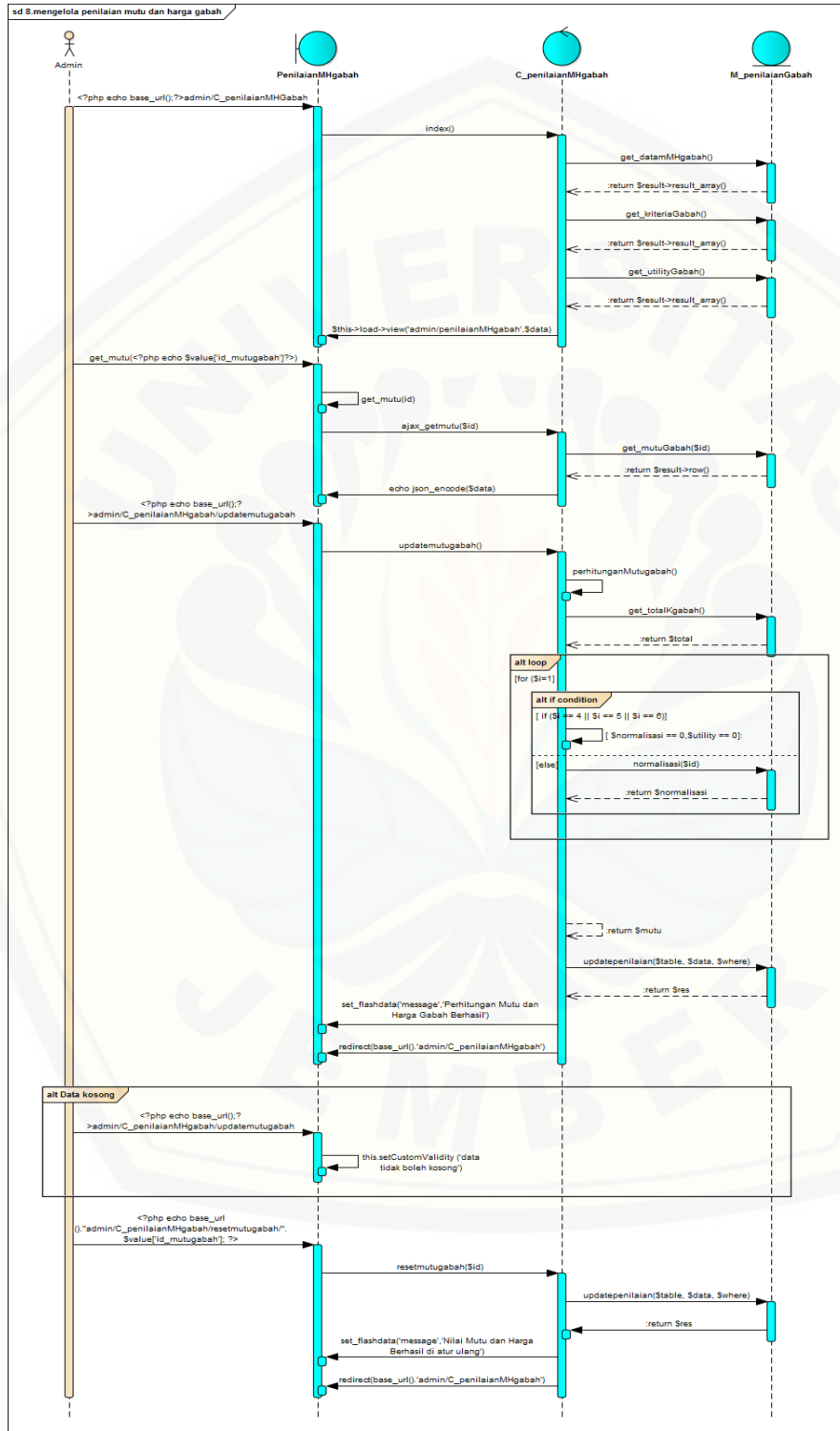
C.5 Sequence Diagram Mengelola Stok Pembelian Gabah



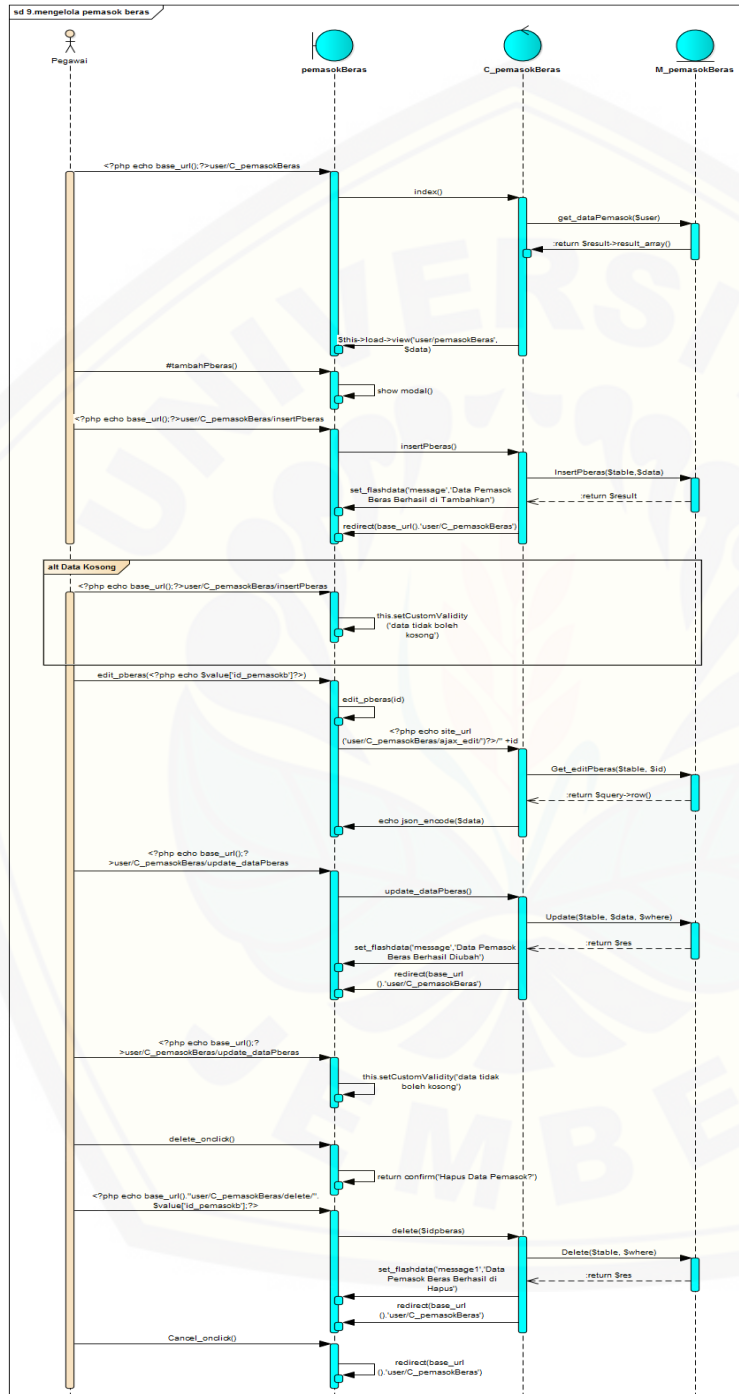
C.6 Sequence Diagram Mengelola Penilaian Mutu dan Harga Beras



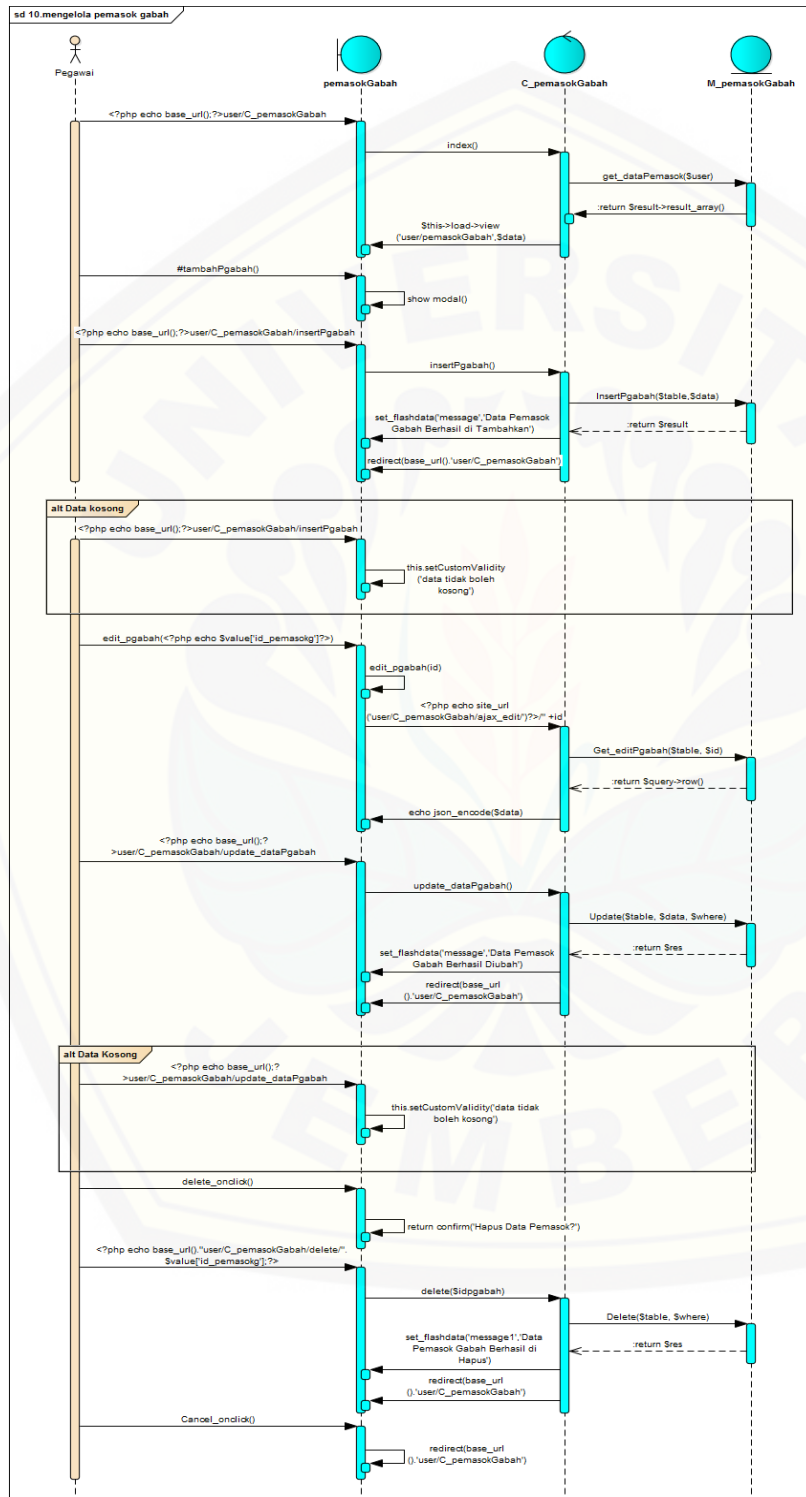
C.7 Sequence Diagram Mengelola Penilaian Mutu dan Harga Gabah



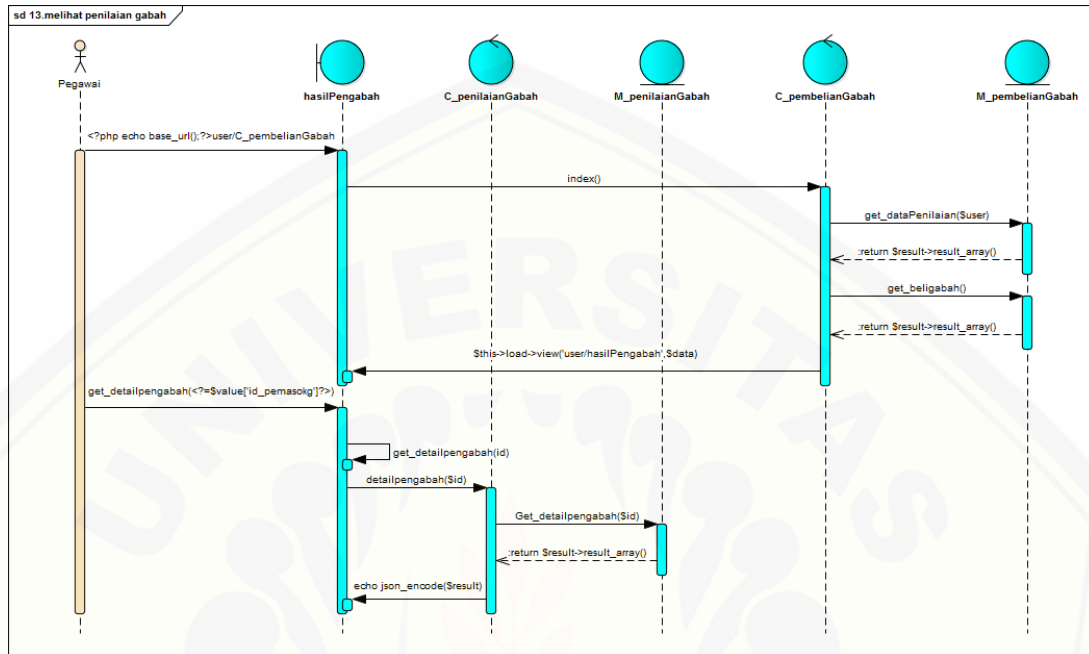
C.8 Sequence Diagram Mengelola Pemasok Beras



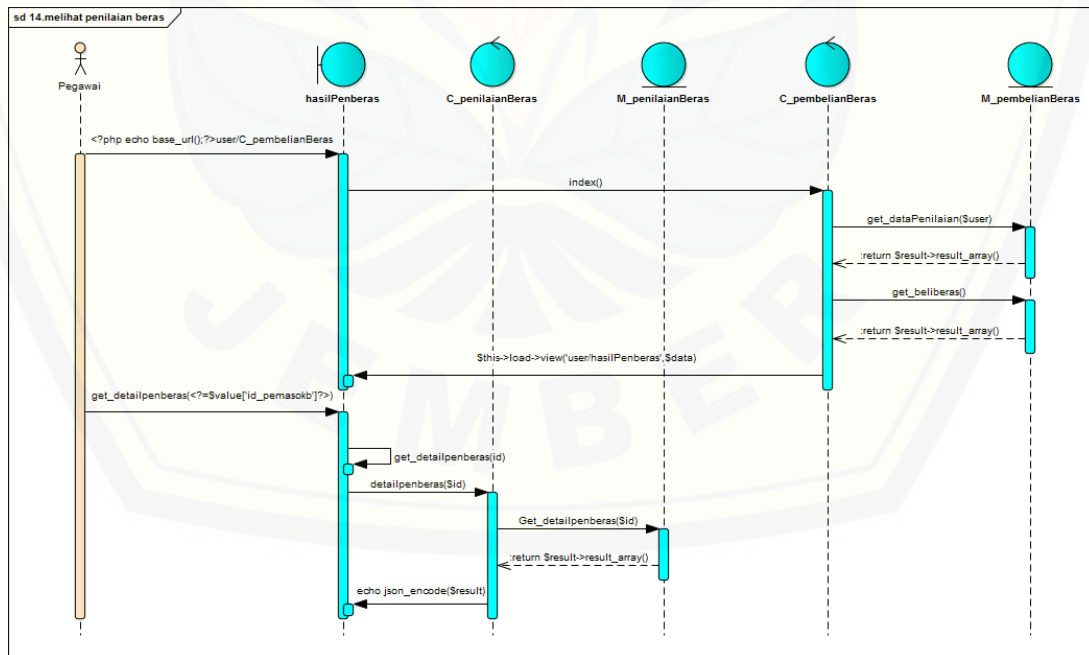
C.9 Sequence Diagram Mengelola Pemasok Gabah



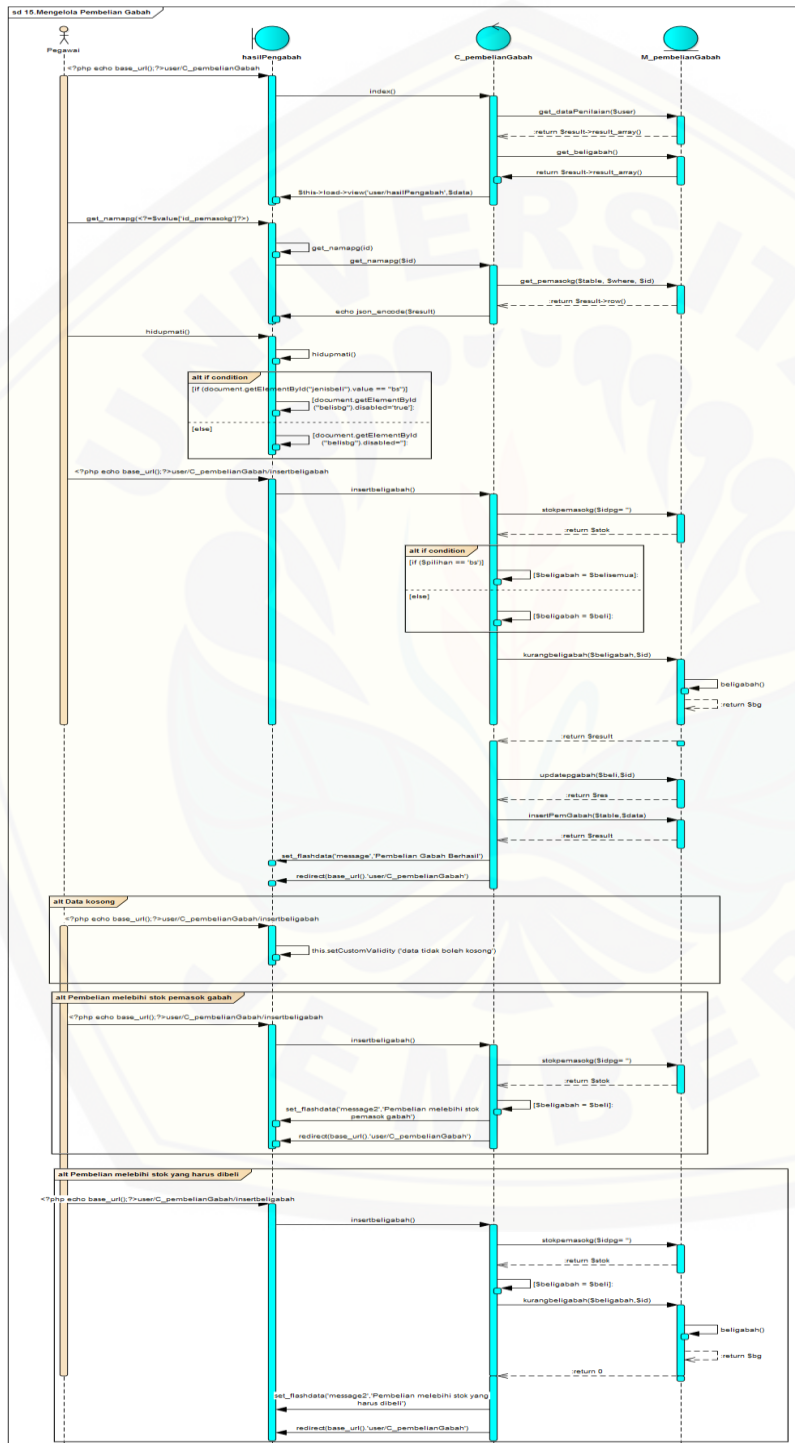
C.11 Sequence Diagram Melihat Penilaian Gabah



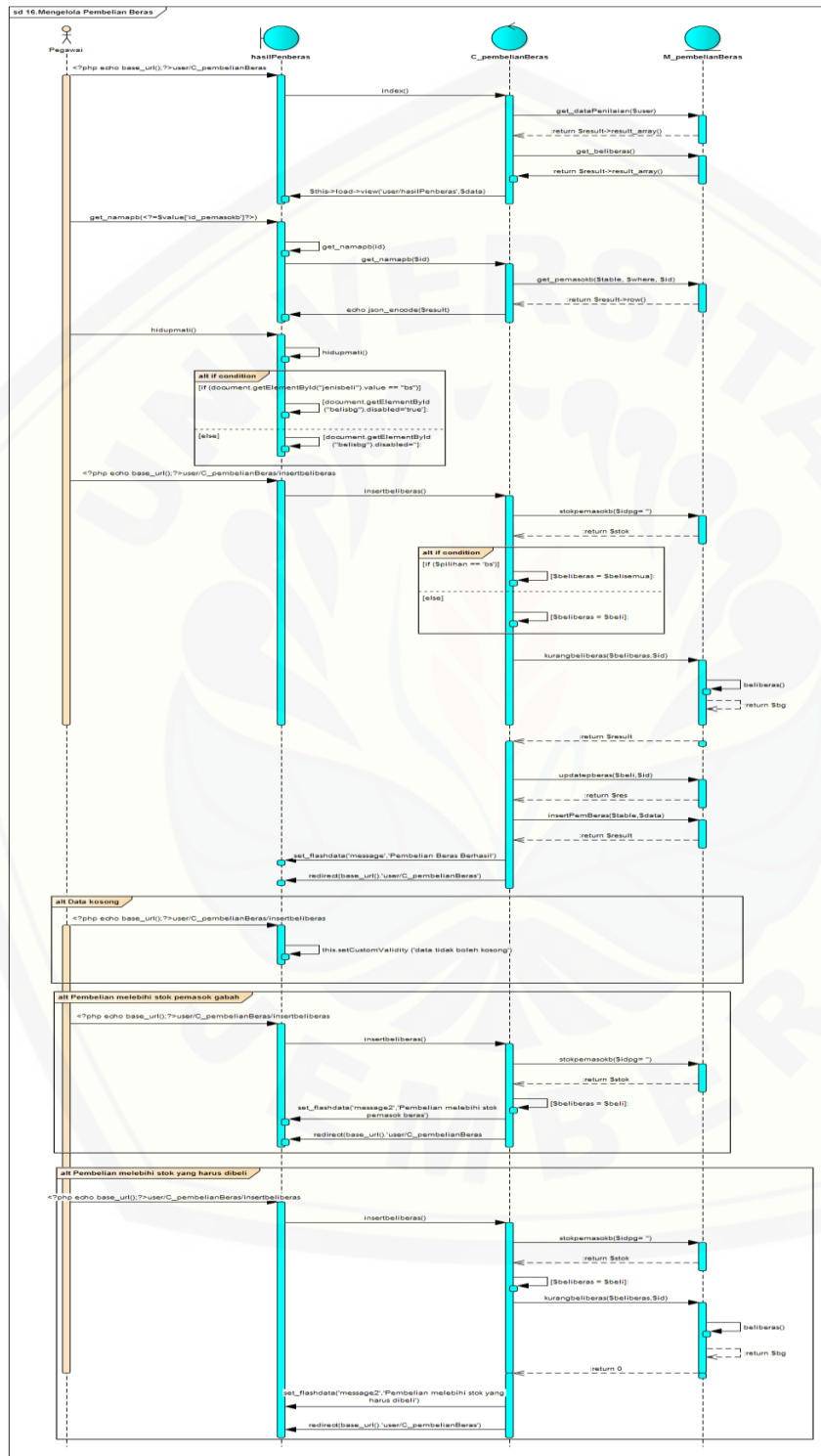
C.12 Sequence Diagram Melihat Penilaian Beras



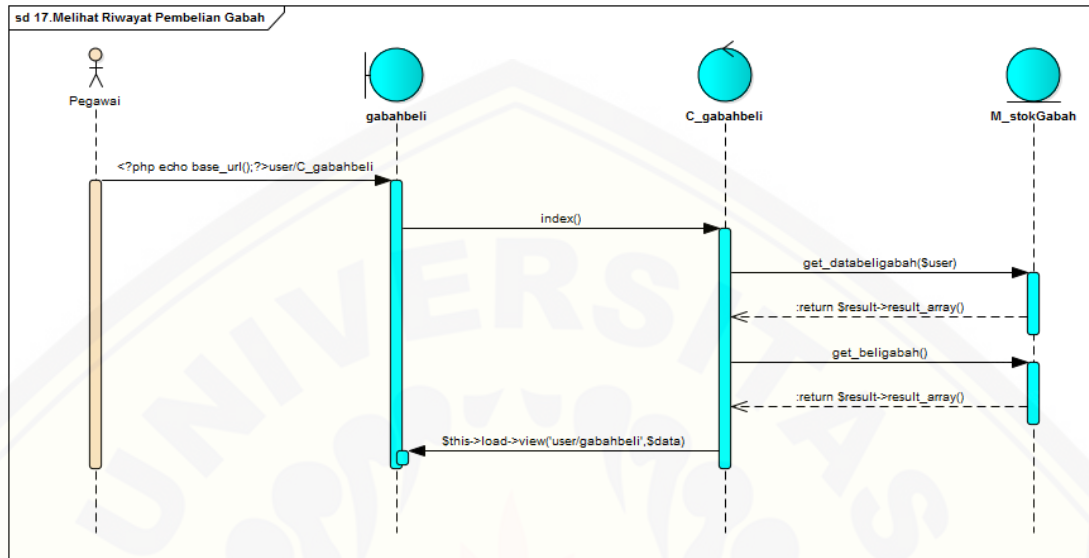
C.13 Sequence Diagram Mengelola Pembelian Gabah



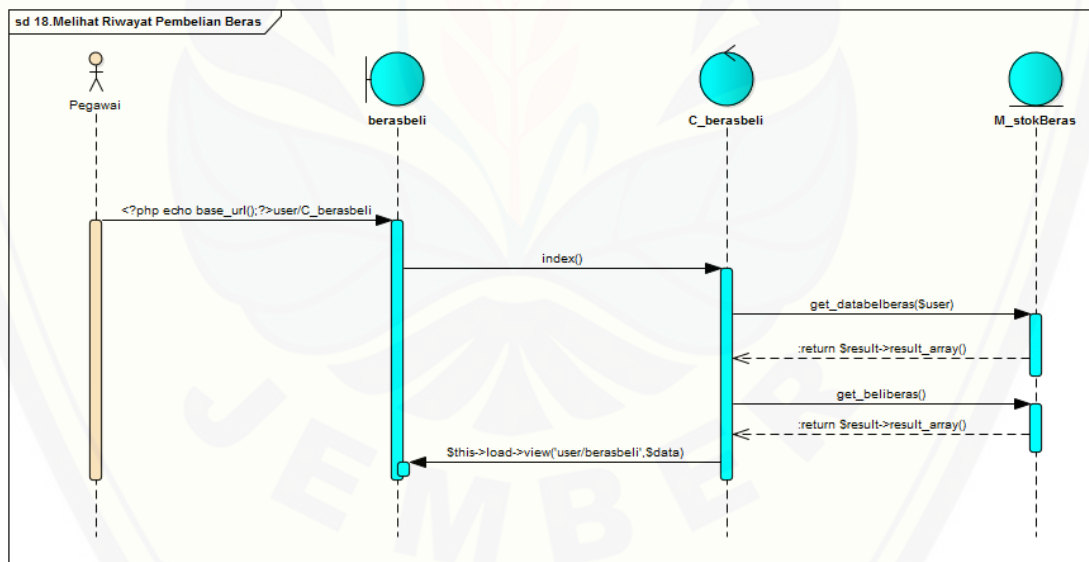
C.14 Sequence Diagram Mengelola Pembelian Beras



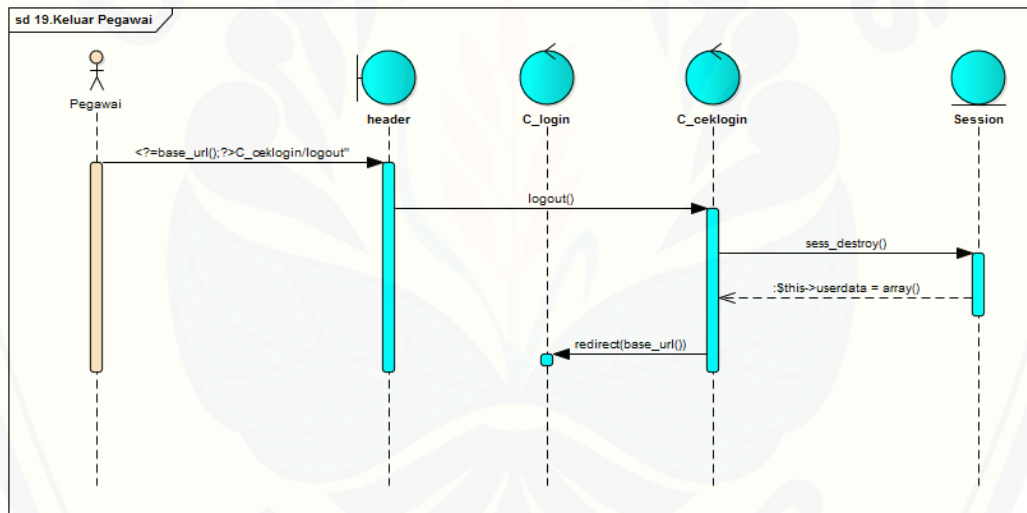
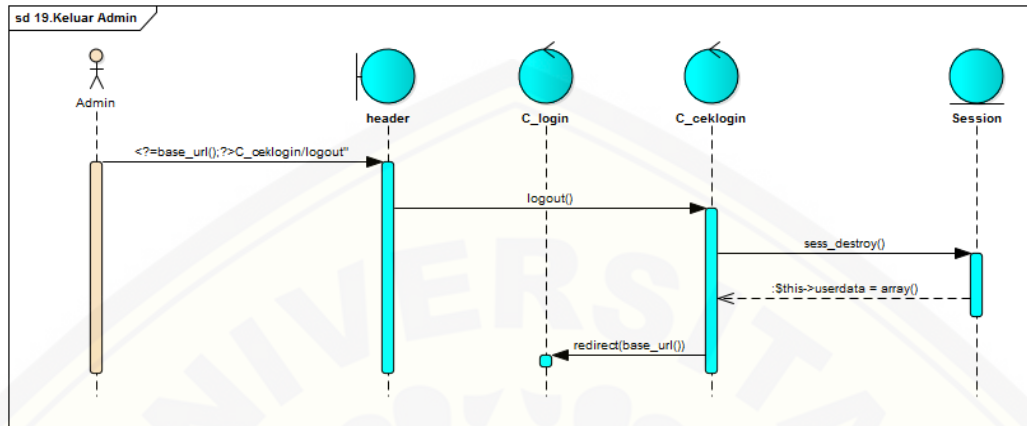
C.15 Sequence Diagram Melihat Riwayat Pembelian Gabah



C.16 Sequence Diagram Melihat Riwayat Pembelian Beras



C.17 Sequence Diagram Logout (Keluar)



LAMPIRAN D. TRANSKRIP WAWANCARA

Wawancara Objek Penelitian

Narasumber : Pak Danis

Jabatan : Pemilik UD. Kamal Lestari

1. UD. Kamal Lestari merupakan pabrik yang bergerak dalam bidang apa ?

Jawaban :

UD. Kamal Lestari merupakan pabrik yang bergerak sebagai produsen beras merek kepala kobra dan dua kelinci.

2. Dalam pembelian gabah, apakah pabrik bapak mementingkan kualitas mutu gabah yang dibeli ?

Jawaban

Benar, pabrik kami memilih gabah dengan kualitas yang baik untuk selanjutnya diolah menjadi beras, gabah dengan mutu yang baik akan menghasilkan beras dengan mutu baik, biasanya pabrik memproduksi beras dengan tipe kepala.

3. Selain pembelian gabah, apakah pabrik bapak juga membeli beras ?

Jawaban

Pabrik saya juga membeli beras, namun tidak dalam skala besar, pembelian beras dilakukan jika bahan baku gabah dalam memproduksi beras kurang, maka dari itu pabrik kami membeli beras secara langsung kepada mitra, atau pabrik yang lain untuk mengimbangi permintaan beras.

4. Dalam proses penentuan pembelian gabah, kriteria apa saja yang dilihat pak, apakah mengacu dalam SNI penentuan gabah ?

Jawaban

Iya, dalam penentuan pembelian gabah terdapat beberapa kriteria yang digunakan untuk menentukan apakah beras tersebut termasuk dalam mutu yang kami cari atau tidak.

5. Jika boleh mengetahui, dari beberapa faktor kriteria dalam SNI penentuan mutu gabah ini, mana urutan kriteria dari yang paling penting sampai yang kurang penting pak ?

Jawaban

Dari yang paling penting yaitu Gabah varietas lain, kadar air, gabah hampa. kondisi gabah, hama, bau, butir mengapur dan gabah muda, benda asing, butir merah, butir rusak dan butir kuning. Gabah varietas lain sangat penting karena nanti akan mempengaruhi beras yang dihasil. Kadar air juga penting, karena jika kadar air kurang dari 14% itu gabah jelek mudah busuk, masih mending yang lebih dari 14% karena masih bisa dikeringkan dengan mesin pengering gabah.

6. Kriteria dalam pembelian beras bagaimana pak urutannya ?

Jawaban

Untuk pembelian beras yang pertama dilihat adalah butir patah, kadar air, butir kepala, hama, bau asing, butir mengapur, benda asing, butir merah, butir kuning, butir menir, butir gabah, dan derajat sosoh, hampir sama sebenarnya dengan urutan kriteria gabah.

7. Untuk pembelian stok produksi beras, setiap berapa hari sekali pak dilakukan proses pembelian ?

Jawaban

Pembelian stok bahan baku dilakukan kadang setiap hari, kadang seminggu sekali, mengikuti melihat stok yang masih tersedia.

8. Pemasok gabah maupun beras, datang darimana saja pak ?

Jawaban

Gabah didapatkan dari petani yang menjual hasil panennya, bisa saja gabah dan beras didapatkan dari pabrik lain.

9. Untuk Pembelian gabah sesuai dengan mutu, kira – kira berapa harga yang ditetapkan ?

Jawaban

Harga Pasaran Pembelian Beras

Mutu	Harga
Mutu 1	Rp. 9100/kg
Mutu 2	Rp. 8300/kg
Mutu 3	Rp 7600/kg
Mutu 4	Rp 6200/kg
Mutu 5	Rp 5500/kg

Harga Pasaran Pembelian Gabah

Mutu	Harga
Mutu 1	Rp. 4700/kg
Mutu 2	Rp. 4500/kg
Mutu 3	Rp 4200/kg

Harga sewaktu – waktu dapat berubah mas, tidak tentu harga tetap segitu, terkadang naik, terkadang juga turun.