



**PENGEMBANGAN SISTEM KLASIFIKASI DOKUMEN BERBASIS
ALGORITMA *NAÏVE BAYES CLASSIFIER* (STUDI KASUS: KONTEN
BERITA POLITIK)**

SKRIPSI

Oleh:

Bazliah Budi Arini

NIM 142410101048

PROGRAM STUDI SISTEM INFOMASI

FAKULTAS ILMU KOMPUTER

UNIVERSITAS JEMBER

2018



**PENGEMBANGAN SISTEM KLASIFIKASI DOKUMEN BERBASIS
ALGORITMA *NAÏVE BAYES CLASSIFIER* (STUDI KASUS: KONTEN
BERITA POLITIK)**

SKRIPSI

Diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan Pendidikan Sarjana (S1) Program Studi Sistem Informasi
Universitas Jember dan mendapat gelar Sarjana Komputer

Oleh:

Bazliah Budi Arini

NIM 142410101048

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS ILMU KOMPUTER

UNIVERSITAS JEMBER

2018

PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Allah SWT yang senantiasa memberikan rahmat dan hidayah-Nya untuk mempermudah dan melancarkan dalam mengerjakan skripsi;
2. Ibunda tercinta Wiwin Mindiarti dan Ayahanda Sunarto;
3. Saudara perempuan Alifah Jessica Andharini;
4. Keluarga besar;
5. Sahabatku bersama dukungan dan doanya;
6. Guru-guruku sejak taman kanak-kanan sampai dengan perguruan tinggi;
7. Almamater Fakultas Ilmu Komputer Universitas Jember;

MOTO

“Kesuksesan bukan sesuatu pemberian, itu sesuatu yang kamu dapat perjuangan.

So go out there, and make your dreams come true.”

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Bazliah Budi Arini

NIM : 142410101048

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Pengembangan Sistem Klasifikasi Dokumen Berbasis Algoritma *Naive Bayes Classifier* (Studi Kasus: Konten Berita Politik)” adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, belum pernah diajukan pada institusi mana pun, 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, 16 Juli 2018

Yang menyatakan,

Bazliah Budi Arini

NIM 142410101048

SKRIPSI

PENGEMBANGAN SISTEM KLASIFIKASI DOKUMEN BERBASIS ALGORITMA *NAÏVE BAYES CLASSIFIER* (STUDI KASUS: KONTEN BERITA POLITIK)

Oleh :

Bazliah Budi Arini

NIM 142410101048

Pembimbing :

Dosen Pembimbing Utama : Achmad Maududie, ST, M.Sc

Dosen Pembimbing Pedamoing : Ifrina Nuritha, S.Kom.,M.Kom

PENGESAHAN PEMBIMBING

Skripsi berjudul “Pengembangan Sistem Klasifikasi Dokumen Berbasis Algoritma *Naïve Bayes Classifier* (Studi Kasus: Konten Berita Politik)”, telah diuji dan disahkan pada:

hari, tanggal : Senin, 16 Juli 2018

tempat : Program Studi Sistem Informasi Universitas Jember

Disetujui oleh:

Pembimbing I,

Pembimbing II,

Achmad Maududie, ST, M.Sc.

Ifrina Nuritha, S.Kom.,M.Kom

NIP 197004221995121001

NIP 760016786

PENGESAHAN PENGUJI

Skripsi berjudul “Pengembangan Sistem Klasifikasi Dokumen Berbasis Algoritma *Naïve Bayes Classifier* (Studi Kasus: Konten Berita Politik)”, telah diuji dan disahkan pada:

hari, tanggal : Senin, 16 Juli 2018

tempat : Program Studi Sistem Informasi Universitas Jember

Tim Penguji:

Penguji I,

Penguji II,

Anang Andrianto, ST., MT

Nova El Maidah S.Si., M.Cs.

NIP. 196906151997021002

NIP. 198411012015042001

Mengesahkan

Dekan Fakultas Ilmu Komputer

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

NIP. 19670420 1992011001

RINGKASAN

Pengembangan Sistem Klasifikasi Dokumen Berbasis Algoritma *Naïve Bayes Classifier* (Studi Kasus:Konten Berita Politik); Bazliah Budi Arini, 142410101048; 2018, 145 halaman; Fakultas Ilmu Komputer Universitas Jember.

Saat ini kecendrungan masyarakat untuk mengakses berita khususnya melalui dunia maya sangat tinggi. Salah satu sumber informasi berita adalah melalui web portal berita. Web portal berita merupakan situs yang memberikan dan mengumpulkan berita atau informasi dari berbagai sumber dan kategori yang beragam untuk disediakan kepada pengguna. Untuk mempercepat dan mempermudah pencarian berita, biasanya portal berita telah mengelompokkan dalam sejumlah kategori. Namun demikian kategori yang digunakan masih bersifat umum sebagai contoh berita politik, ekonomi, olahraga, dan entertainment. Apabila pembaca berita ingin mengelompokkan menurut topik yang lebih spesifik lagi, maka pembaca tersebut harus melakukannya secara manual dengan membaca seluruh berita dalam sebuah kategori dan kemudian mengelompokkannya berdasarkan subkategori yang lebih detail. Maka dari itu diperlukan adanya sistem yang dimana sistem tersebut dapat mengklasifikasikan atau mengkategorikan berita secara otomatis sesuai dengan kategori berita yang lebih spesifik.

Penelitian ini menggunakan metode *Naïve Bayes Classifier* untuk melakukan klasifikasi dokumen. Model pengembangan dari sistem ini mengacu pada model *waterfall*. Penelitian ini dilaksanakan dalam 4 tahap yaitu analisis kebutuhan fitur, desain sistem, implementasi sistem dan pengujian sistem. Pada tahap implementasi sistem diterapkan metode *Naïve Bayes Classifier* dalam menentukan klasifikasi dokumen. Uji performansi klasifikasi pada sistem ini menggunakan 3 perhitungan yaitu *precion*, *recall*, dan *fmeasure*. Hasil uji performansi terhadap sistem menunjukkan bahwa sistem dapat melakukan klasifikasi dengan baik yaitu sebesar 98.37%.

PRAKATA

Puji Syukur kehadiran Allah SWT atas segala rahmat dan karunia-nya, sehingga penulis dapat menyelesaikan skripsi dengan judul “Pengembangan Sistem Klasifikasi Dokumen Berbasis Algoritma *Naïve Bayes Classifier* (Studi Kasus: Konten Berita Politik)”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan Pendidikan Strata Satu (S1) pada Program Studi Sistem Informasi Universitas Jember.

Penyusun skripsi ini tidak lepas dari dukungan berbagai pihak. Oleh karena itu penulis menyampaikan terima kasih kepada:

1. Prof. Drs. Slamin, M.Comp.Sc., PhD., selaku Ketua Program Studi Sistem Informasi Universitas Jember.
2. Achmad Maududie, ST, M.Sc., selaku Dosen Pembimbing Utama dan Ifrina Nuritha, S.Kom., M.Kom., selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi.
3. Windi Eka Yulia Retnani, S.Kom.,M.T., selaku dosen pembimbing akademik yang telah mendampingi penulis sebagai mahasiswa.
4. Seluruh Bapak dan Ibu Dosen beserta Staff Karyawan di Program Studi Sistem Informasi Universitas Jember.
5. Ibunda tercinta Wiwin Mindiarti dan Ayahanda Sunarto yang selalu mendukung dan mendoakan.
6. Saudara perempuan Alifah Jessica Andharini yang telah mendukung dan mendoakan.
7. Difari Afreyna Fauziah dan Awanda Prisma, yang telah menjadi sahabat yang setia mendengar keluh kesah, memberi semangat, motivasi dan menemani penulis.
8. Teman seperjuangan Layli Indah Palupi, Amalia Tri Okta, Farus Tri Wahyuning, Khaira Ummah, Anne Ruffa, Selvi Bastian, Nurul Aeni, yang telah memberi semangat dan membantu penulis.
9. Arief Setiawan dan Andry Dermawan yang telah membantu penulis.

10. Nabila Nur Aisyah Al Ayyubi, Afifah Putri Ridziana, Annisa Aghnia Darajah, Elita Ismi Mientarini, dan Hesti Fitria Ilmi atas dukungannya.
11. Keluarga KKN 32 Desa Pacoran 2017 Siska, Wilda, Rizka, Claudia, Intan, Yudha, Adi, Zain, dan Agus atas semua keceriaan, pengalaman, kasih serta semangat.
12. Teman-teman seperjuangan SENSATION angkatan 2014 atas semua pengalaman, perjuangan, kasih dan dukungannya.
13. Teman-teman Program Studi Sistem Informasi di semua angkatan yang telah mendukung dan mendoakan.
14. Semua pihak yang tidak dapat disebutkan satu persatu yang telah membantu penulis.

Dengan harapan bahwa penelitian ini nantinya terus berlanjut dan berkembang kelak, penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan skripsi ini. Penulis berharap skripsi ini dapat bermanfaat bagi semua pihak.

Jember, 16 Juli 2018

Penulis

DAFTAR ISI

PERSEMBAHAN	iii
MOTO	iv
PERNYATAAN.....	v
SKRIPSI.....	vi
PENGESAHAN PEMBIMBING.....	vii
PENGESAHAN PENGUJI.....	viii
RINGKASAN	ix
PRAKATA.....	x
DAFTAR ISI.....	xii
DAFTAR TABEL.....	xvi
DAFTAR GAMBAR	xviii
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah.....	2
1.3 Tujuan.....	2
1.4 Manfaat.....	2
1.5 Batasan Masalah.....	3
1.6 Sistematika Penulisan.....	3
BAB 2. TINJAUAN PUSTAKA	5
2.1 Penelitian Terdahulu.....	5
2.2 <i>Text Mining</i> dan Klasifikasi	6
2.3 <i>Naïve Bayes Classifier</i>	9
2.4 Uji Performansi	11
2.5 Pengembangan Sistem.....	13
2.5.1 Analisis Kebutuhan (<i>Requirement</i>).....	14
2.5.2 Desain Sistem.....	14
2.5.3 Implementasi Sistem dan <i>Unit Testing</i>	19
2.5.4 Pengujian Sistem	20
2.5.5 Pemeliharaan Sistem	20
BAB 3. METODOLOGI PENELITIAN.....	21

3.1	Gambaran Umum Sistem	21
3.2	Jenis Penelitian	21
3.3	Tahapan Penelitian	22
3.3.1	Analisa Kebutuhan	22
3.3.2	Desain Sistem	23
3.3.3	Implementasi Sistem	23
3.3.4	Pengujian Sistem	26
BAB 4. PENGEMBANGAN SISTEM.....		28
4.1	Analisa Data	28
4.1.1	Kebutuhan Fungsional.....	28
4.1.2	Kebutuhan Non Fungsional.....	29
4.2	Desain Sistem	29
4.3.1	<i>Bussiness Process Modelling Notation (BPMN)</i>	29
4.3.2	<i>Usecase Diagram</i>	31
4.3.3	Skenario <i>Usecase</i>	33
4.3.4	<i>Activity Diagram</i>	40
4.3.5	<i>Sequence Diagram</i>	44
4.3.6	<i>Class Diagram</i>	49
4.3.7	<i>Entity Relationship Diagram (ERD)</i>	53
4.3	Implementasi Sistem	56
4.4	Pengujian Sistem	61
4.4.1	Pengujian <i>Blackbox</i>	61
4.4.2	Uji Performasi	61
BAB 5. HASIL DAN PEMBAHASAN.....		62
5.1	Hasil dan Pembahasan Implementasi Pada Sistem Klasifikasi Dokumen Berita Politik.....	62
5.1.1	Fitur Cek Kategori Berita	62
5.1.2	Fitur Data <i>Training</i>	64
5.1.3	Fitur Data <i>Testing</i>	69
5.1.4	Fitur Data <i>Stopword</i>	73
5.2	Hasil dan Pembahasan Uji Performansi	78
BAB 6. PENUTUP		80

6.1	Kesimpulan.....	80
6.2	Saran.....	81
	DAFTAR PUSTAKA	82
	LAMPIRAN.....	85
A.	Skenario <i>Usecase</i>	85
A.1	Skenario Cek Kategori Dokumen	85
A.2	Skenario Hasil Cek Kategori Dokumen	85
A.3	Skenario <i>Login</i>	86
A.4	Skenario Melihat <i>Dashboard</i>	86
A.5	Skenario Melihat <i>Text Preprocessing Data Training</i>	87
A.6	Skenario Melihat Frekuensi Kata Data <i>Training</i>	90
A.7	Skenario Melihat Daftar Kata Data <i>Training</i>	91
A.8	Skenario Melihat <i>Probability</i> Kata Data <i>Training</i>	91
A.9	Skenario Mengelola Data <i>Testing</i>	92
A.10	Skenario Melihat <i>Text Preprocessing Data Testing</i>	95
A.11	Skenario Melihat Hasil Klasifikasi Data <i>Testing</i>	97
A.12	Skenario Mengelola Data <i>Stopword</i>	98
A.13	Skenario <i>Logout</i>	100
A.14	Skenario Melihat Berita Politik.....	100
B.	<i>Activity Diagram</i>	101
B.1	<i>Activity Diagram</i> Hasil Cek Kategori Dokumen.....	101
B.2	<i>Activity Diagram</i> Melihat Berita Politik	101
B.3	<i>Activity Diagram Login</i>	102
B.4	<i>Activity Diagram</i> Melihat <i>Dashboard</i>	102
B.5	<i>Activity Diagram</i> Melihat <i>Text Preprocessing Data Training</i>	103
B.6	<i>Activity Diagram</i> Cek Kategori Dokumen	104
B.7	<i>Activity Diagram</i> Melihat Frekuensi Kata Data <i>Training</i>	104
B.8	<i>Activity Diagram</i> Melihat Daftar Kata Data <i>Training</i>	105
B.9	<i>Activity Diagram</i> Melihat <i>Probability</i> Kata Data <i>Training</i>	105
B.10	<i>Activity Diagram</i> Melihat Hasil Klasifikasi Data <i>Testing</i>	106
B.11	<i>Activity Diagram Logout</i>	106
B.12	<i>Activity Diagram</i> Mengelola Data <i>Testing</i>	107

B.13	<i>Activity Diagram</i> Melihat <i>Text Preprocessing</i> Data <i>Testing</i>	108
B.14	<i>Activity Diagram</i> Mengelola Data <i>Stopword</i>	109
C.	<i>Sequence Diagram</i>	110
C.1	<i>Sequence Diagram</i> Cek Kategori Dokumen	110
C.2	<i>Sequence Diagram</i> Hasil Cek Kategori Dokumen.....	110
C.3	<i>Sequence Diagram</i> Melihat <i>Text Preprocessing</i> Data <i>Training</i>	111
C.4	<i>Sequence Diagram</i> Melihat Daftar Kata Data <i>Training</i>	112
C.5	<i>Sequence Diagram</i> Melihat <i>Dashboard</i>	112
C.6	<i>Sequence Diagram</i> Mengelola Data <i>Testing</i>	113
C.7	<i>Sequence Diagram</i> <i>Login</i>	114
C.8	<i>Sequence Diagram</i> <i>Logout</i>	114
C.9	<i>Sequence Diagram</i> Melihat <i>Text Preprocessing</i> Data <i>Testing</i>	115
C.10	<i>Sequence Diagram</i> Mengelola Data <i>Stopword</i>	116
C.11	<i>Sequence Diagram</i> Melihat Hasil Klasifikasi Data <i>Testing</i>	117
C.12	<i>Sequence Diagram</i> Melihat Frekuensi Kata Data <i>Training</i>	117
C.13	<i>Sequence Diagram</i> Melihat Probability Kata <i>Training</i>	118
C.14	<i>Sequence Diagram</i> Melihat Berita Politik	118
D.	Data Uji Performasi.....	119
E.	Pengujian <i>Blackbox</i>	122

DAFTAR TABEL

Tabel 2. 1 Matrik Kontingensi Evaluasi Kinerja Sistem	11
Tabel 3. 1 Dataset.....	26
Tabel 4. 1 Definisi Aktor	32
Tabel 4. 2 Definisi <i>Usecase</i>	32
Tabel 4. 3 Mengelola Data <i>Training</i>	34
Tabel 4. 4 <i>Method Class Controller ArtikelContrller</i>	50
Tabel 4. 5 <i>Method Class Controller HomeController</i>	51
Tabel 4. 6 <i>Method Class Controller LoginController</i>	51
Tabel 4. 7 <i>Method Class Controller SettingController</i>	52
Tabel 4. 8 <i>Method Class Controller StopwordController</i>	52
Tabel 4. 9 <i>Class Model User</i>	52
Tabel 4. 10 <i>Class Model Stopword</i>	52
Tabel 4. 11 <i>Class Model MasterTraining</i>	52
Tabel 4. 12 <i>Class Model Setting</i>	53
Tabel 4. 13 <i>Class Model Artikel</i>	53
Tabel 4. 14 <i>Class Model Cek Berita</i>	53
Tabel 4. 15 Entitas users	54
Tabel 4. 16 Entitas stopword.....	55
Tabel 4. 17 Entitas tb_kadadasar	55
Tabel 4. 18 Entitas artikel	55
Tabel 4. 19 Entitas master_training	55
Tabel 4. 20 Entitas setting.....	55
Tabel 4. 21 Entitas cek_berita.....	56
Tabel 4. 22 Entitas rekap_artikel	56
Tabel 5. 1 Pengujian <i>Blackbox</i> Cek Kategori Dokumen.....	63
Tabel 5. 2 Pengujian <i>Blackbox</i> Menampilkan Data <i>Training</i>	65
Tabel 5. 3 Kode Program Kelas <i>Controller</i> Menambah Data <i>Training</i>	66
Tabel 5. 4 Pengujian <i>Blackbox</i> Mengubah Data <i>Training</i>	67
Tabel 5. 5 Pengujian <i>Blackbox</i> Menghapus Data <i>Training</i>	69
Tabel 5. 6 Pengujian <i>Blackbox</i> Menampilkan Data <i>Testing</i>	69
Tabel 5. 7 Pengujian <i>Blackbox</i> Menambah Data <i>Testing</i>	71
Tabel 5. 8 Pengujian <i>Blackbox</i> Mengubah Data <i>Testing</i>	72
Tabel 5. 9 Pengujian <i>Blackbox</i> Menghapus Data <i>Testing</i>	73
Tabel 5. 10 Pengujian <i>Blackbox</i> Menampilkan Data <i>Stopword</i>	74

Tabel 5. 11 Pengujian <i>Blackbox</i> Menambah Data <i>Stopwr</i> d.....	75
Tabel 5. 12 Pengujian <i>Blackbox</i> Mengubah Data <i>Stopword</i>	76
Tabel 5. 13 Pengujian <i>Blackbox</i> Menghapus Data <i>Stopword</i>	77
Tabel 5. 14 Kesimpulan Pengujian <i>Blackbox</i> dengan Kode Program	77
Tabel 5. 15 Data Uji	78
Tabel 5. 16 Matrik Kontingensi untuk Data Uji	78
Tabel 5. 17 Uji Performansi Kelas Kategori Undang-Undang Ormas	79
Tabel 5. 18 Uji Performansi Kelas Kategori Reshuffle Kabinet.....	79
Tabel 5. 19 Uji Performansi Kelas Kategori Pilkada.....	79
Tabel 5. 20 Uji Performansi Kelas Kategori Rata-Rata.....	79
Tabel A. 1 Skenario Cek Kategori Dokumen	85
Tabel A. 2 Skenario Hasil Cek Kategori Dokumen.....	85
Tabel A. 3 Skenario <i>Login</i>	86
Tabel A. 4 Skenario Melihat <i>Dashboard</i>	86
Tabel A. 5 Skenario Melihat <i>Text Preprocessing</i> Data <i>Training</i>	87
Tabel A. 6 Skenario Melihat Frekuensi Kata Data <i>Training</i>	90
Tabel A. 7 Skenario Melihat Daftar Kata Data <i>Training</i>	91
Tabel A. 8 Skenario Melihat <i>Probability</i> Kata Data <i>Training</i>	91
Tabel A. 9 Skenario Mengelola Data <i>Testing</i>	92
Tabel A. 10 Skenario Melihat <i>Text Preprocessing</i> Data <i>Testing</i>	95
Tabel A. 11 Skenario Melihat Hasil Klasifikasi Data <i>Testing</i>	97
Tabel A. 12 Skenario Mengelola Data <i>Stopword</i>	98
Tabel A. 13 Skenario <i>Logout</i>	100
Tabel A. 14 Skenario Melihat Bertia Politik.....	100
Tabel D. 1 Hasil Data Uji.....	119
Tabel E. 1 Pengujian <i>Blackbox</i>	122

DAFTAR GAMBAR

Gambar 2. 1 Tahapan <i>Text Preprocessing</i>	6
Gambar 2. 2 Tahap <i>Casefolding</i>	7
Gambar 2. 3 Tahap <i>Tokenizing</i>	7
Gambar 2. 4 Tahap <i>Stopwords</i>	8
Gambar 2. 5 Tahap <i>Stemming</i>	8
Gambar 2. 6 Model <i>Waterfall</i>	13
Gambar 2. 7 Contoh <i>Bussiness Process Modelling Notation</i> Perkuliahan	15
Gambar 2. 8 Contoh <i>Diagram Usecase</i>	16
Gambar 2. 9 Contoh <i>Activity Diagram</i>	17
Gambar 2. 10 Contoh <i>Sequence Diagram</i>	18
Gambar 2. 11 Contoh <i>Class Diagram</i>	18
Gambar 2. 12 Contoh <i>Entity Relationship Diagram</i>	19
Gambar 3. 1 Flowchart Training.....	24
Gambar 3. 2 Flowchart Testing.....	25
Gambar 4. 1 <i>Bussines Process Modelling Notation</i>	30
Gambar 4. 2 <i>Usecase Diagram</i>	31
Gambar 4. 3 <i>Activity Diagram</i> Mengelola Data <i>Training</i>	43
Gambar 4. 4 <i>Sequence Diagram</i> Mengelola Data <i>Training</i>	46
Gambar 4. 5 <i>Class Diagram</i>	49
Gambar 4. 6 <i>Entity Relationship Diagram</i>	54
Gambar 4. 7 Kode Program Proses <i>Casefolding</i>	57
Gambar 4. 8 Kode Program Proses <i>Tokenizing</i>	57
Gambar 4. 9 Kode Program Proses <i>Stopword 1</i>	57
Gambar 4. 10 Kode Program Proses <i>Stopword 2</i>	58
Gambar 4. 11 Kode Program Proses <i>Stemming 1</i>	58
Gambar 4. 12 Kode Program Proses <i>Stemming 2</i>	59
Gambar 4. 13 Kode Program Proses <i>Stemming 3</i>	59
Gambar 4. 14 Kode Program Proses <i>Stemming 4</i>	60
Gambar 5. 1 Kode Program Kelas <i>Controller</i> Fitur Cek Kategori Berita	63
Gambar 5. 2 Tampilan Awal Cek Kategori Dokumen.....	63
Gambar 5. 3 Tampilan Mengisi <i>Textbox</i> Artikel Cek Kategori Dokumen	64
Gambar 5. 4 Tampilan Hasil Cek Kategori Dokumen.....	64
Gambar 5. 5 Kode Program Kelas <i>Model</i> Data <i>Training</i> Gambar.....	65
Gambar 5. 6 Kode Program Kelas <i>Controler</i> Menampilkan Data <i>Training</i>	65
Gambar 5. 7 Tampilan Mengelola Data <i>Training</i>	66

Gambar 5. 8 Kode Program Menambah Data <i>Training</i>	66
Gambar 5. 9 Tampilan Tambah Data <i>Training</i>	67
Gambar 5. 10 Kode Program Kelas <i>Controller</i> Mengubah Data <i>Training</i>	67
Gambar 5. 11 Tampilan <i>Update</i> Data <i>Training</i>	68
Gambar 5. 12 Kode Program Menghapus Data <i>Training</i>	68
Gambar 5. 13 Kode Program Kelas <i>Model</i> Data <i>Testing</i>	69
Gambar 5. 14 Kode Program Kelas <i>Controller</i> Data <i>Testing</i>	69
Gambar 5. 15 Tampilan Mengelola Data <i>Testing</i>	70
Gambar 5. 16 Kode Program Menambah Data <i>Testing</i>	70
Gambar 5. 17 Tampilan Tambah Data <i>Testing</i>	71
Gambar 5. 18 Kode Program Kelas <i>Controller</i> Mengubah Data <i>Training</i>	71
Gambar 5. 19 Tampilan <i>Update</i> Data <i>Testing</i>	72
Gambar 5. 20 Kode Program <i>Controller</i> Menghapus Data <i>Testing</i>	72
Gambar 5. 21 Kode Program Kelas <i>Model</i> Data <i>Stopword</i>	73
Gambar 5. 22 Kode Program Kelas <i>Controller</i> Menampilkan Data <i>Testing</i>	73
Gambar 5. 23 Tampilan Data <i>Stopword</i>	74
Gambar 5. 24 Kode Program Kelas <i>Controller</i> Menambah Data <i>Stopword</i>	74
Gambar 5. 25 Tampilan Tambah Data <i>Stopword</i>	75
Gambar 5. 26 Kode Program Kelas <i>Controller</i> Mengubah Data <i>Stopword</i>	75
Gambar 5. 27 Tampilan <i>Update</i> Data <i>Stopword</i>	76
Gambar 5. 28 Kode Program Kelas <i>Controller</i> Menghapus Data <i>Stopword</i>	76
Gambar B. 1 <i>Activity Diagram</i> Hasil Cek Kategori Dokumen.....	101
Gambar B. 2 <i>Activity Diagram</i> Melihat Berita Politik.....	101
Gambar B. 3 <i>Activity Diagram</i> <i>Login</i>	102
Gambar B. 4 <i>Activity Diagram</i> Melihat <i>Dashboard</i>	102
Gambar B. 5 <i>Activity Diagram</i> Melihat <i>Text Preprocessing</i> Data <i>Training</i>	103
Gambar B. 6 <i>Activity Diagram</i> Cek Kategori Dokumen	104
Gambar B. 7 <i>Activity Diagram</i> Melihat Frekuensi Kata Data <i>Training</i>	104
Gambar B. 8 <i>Activity Diagram</i> Melihat Daftar Kata Data <i>Training</i>	105
Gambar B. 9 <i>Activity Diagram</i> Melihat <i>Probability</i> Kata Data <i>Training</i>	105
Gambar B. 10 <i>Activity Diagram</i> Melihat Hasil Klasifikasi Data <i>Testing</i>	106
Gambar B. 11 <i>Activity Diagram</i> <i>Logout</i>	106
Gambar B. 12 <i>Activity Diagram</i> Mengelola Data <i>Testing</i>	107
Gambar B. 13 <i>Activity Diagram</i> Melihat <i>Text Preprocessing</i> Data <i>Testing</i>	108
Gambar B. 14 <i>Activity Diagram</i> Mengelola Data <i>Stopword</i>	109
Gambar C. 1 <i>Sequence Diagram</i> Cek Kategori Dokumen	110
Gambar C. 2 <i>Sequence Diagram</i> Hasil Cek Kategori Dokumen.....	110
Gambar C. 3 <i>Sequence Diagram</i> Melihat <i>Text Preprocessing</i> Data <i>Training</i>	111
Gambar C. 4 <i>Sequence Diagram</i> Melihat Daftar Kata Data <i>Training</i>	112
Gambar C. 5 <i>Sequence Diagram</i> Melihat <i>Dashboard</i>	112

Gambar C. 6 <i>Sequence Diagram</i> Mengelola Data <i>Testing</i>	113
Gambar C. 7 <i>Sequence Diagram</i> Login	114
Gambar C. 8 <i>Sequence Diagram</i> Logout	114
Gambar C. 9 <i>Sequence Diagram</i> Melihat <i>Text Preprocessing</i> Data <i>Testing</i>	115
Gambar C. 10 <i>Sequence Diagram</i> Mengelola Data <i>Stopword</i>	116
Gambar C. 11 <i>Sequence Diagram</i> Melihat Hasil Klasifikasi Data <i>Testing</i>	117
Gambar C. 12 <i>Sequence Diagram</i> Melihat Frekuensi Kata Data <i>Training</i>	117
Gambar C. 13 <i>Sequence Diagram</i> Melihat <i>Probability</i> Kata <i>Training</i>	118
Gambar C. 14 <i>Sequence Diagram</i> Melihat Berita Politik.....	118

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Berita merupakan bentuk laporan atau informasi mengenai sesuatu yang sedang terjadi baru ini atau keterangan terbaru dari suatu peristiwa dan biasanya disajikan dengan bentuk media cetak, siaran, internet atau dari mulut ke mulut kepada masyarakat (Chandra et al., 2016). Salah satu sumber informasi berita adalah web portal berita berupa situs yang memberikan dan mengumpulkan berita atau informasi dari berbagai sumber bagi pengguna.

Saat ini kecendrungan masyarakat untuk mengakses berita khususnya melalui dunia maya sangat tinggi. Beberapa portal berita *online* yang sering dikunjungi masyarakat untuk mencari berita yaitu kompas, sindo, detik, liputan dan tempo (Alexa Internet, 2018). Untuk mempercepat dan mempermudah pencarian berita, biasanya portal berita telah mengelompokkan dalam sejumlah kategori. Namun demikian kategori yang digunakan masih bersifat umum sebagai contoh berita politik, ekonomi, olahraga, dan entertainment. Apabila pembaca berita ingin mengelompokkan menurut topik yang lebih spesifik lagi, maka pembaca tersebut harus melakukannya secara manual dengan membaca seluruh berita di sebuah kategori dan kemudian mengelompokkannya berdasarkan subkategori yang lebih detail. Hal ini tidak menjadi mudah karena setiap berita memiliki tingkat similaritas yang tinggi dan jumlah berita yang terus meningkat dengan sangat cepat

Klasifikasi adalah proses untuk mengklasifikasikan atau memberi label terhadap dokumen di kelas tertentu agar lebih mudah dikelola. Prinsip kerja proses klasifikasi adalah menentukan kelas objek yang belum diketahui kelasnya dengan cara membandingkannya dengan data training yang telah disiapkan. Proses pengelompokkan dokumen teks, penggunaan klasifikasi telah banyak dilakukan oleh para peneliti dengan menerapkan berbagai metode yang salah satunya adalah *Naïve Bayes*. Berdasarkan penelitian terdahulu, penerapan metode *Naïve Bayes Classifier* mampu melakukan klasifikasi dengan memperoleh tingkat akurasi rata-rata di atas 80% (Hamzah, 2012).

Melihat permasalahan di atas, peneliti mencoba mengembangkan sistem klasifikasi dengan menerapkan metode *Naïve Bayes* yang mampu melakukan klasifikasi terhadap dokumen berita yang belum diketahui kelasnya. Sistem ini diharapkan dapat membantu mengelompokkan dokumen berita pada kategori yang lebih detail. Untuk membatasi pembahasan, kategori dalam dokumen berita yang digunakan adalah berita politik.

1.2 Perumusan Masalah

Berdasarkan uraian yang telah disampaikan di atas, maka rumusan masalah yang harus diselesaikan dalam penelitian ini adalah sebagai berikut:

1. Bagaimana mengembangkan sistem klasifikasi dokumen menggunakan metode *Naïve Bayes Classifier*?
2. Seberapa baik performansi klasifikasi metode *Naïve Bayes Classifier* apabila diterapkan pada dokumen dengan tingkat data similaritas yang tinggi?

1.3 Tujuan

Tujuan yang ingin dicapai dalam penelitian ini adalah:

1. Merancang mengembangkan sistem klasifikasi dokumen menggunakan metode *Naïve Bayes Classifier*.
2. Mengetahui performansi klasifikasi menggunakan metode *Naïve Bayes Classifier* apabila diterapkan pada dokumen dengan tingkat data similaritas yang tinggi.

1.4 Manfaat

Manfaat yang diharapkan dari hasil penelitian ini adalah:

- a. Bagi peneliti yaitu memberikan pengalaman dalam mengsystemkan teori yang telah diperoleh dibangku kuliah dan sebagai wahana untuk memperoleh pengetahuan baru dalam bidang penelitian dan penulisan karya ilmiah. Selain itu, untuk mengetahui bagaimana proses penerapan metode *Naïve Bayes Classifier* terhadap sebuah sistem.

- b. Bagi penelitian lain dapat menambah referensi bagi peneliti dalam melakukan penelitian yang sama.
- c. Bagi akademis yaitu penelitian ini diharapkan dapat memberikan masukan informasi mengenai penerapan algoritma *Naïve Bayes Classifier* kepada pembaca pada umumnya dan kepada Fakultas Ilmu Komputer Universitas Jember.

1.5 Batasan Masalah

Beberapa hal yang menjadi batasan masalah dalam penelitian ini adalah:

1. Dokumen dengan similaritas yang tinggi digunakan dalam penelitian ini adalah dokumen konten berita politik yang didapatkan dari 4 portal berita *online* yaitu detik, kompas, sindo, dan liputan6.
2. Dokumen berita politik diunduh secara manual sejumlah 300 artikel dan kemudian akan diklasifikasikan dalam 3 subkategori berita politik yaitu berita pilkada, berita uu ormas, dan berita reshuffle kabinet.
3. Sistem yang dibangun berbasis *website* yang dapat diakses oleh admin, editor dan pengguna secara umum (*guest*). Admin dapat melakukan pengelolaan data *training*, data *testing*, dan data *stopword*. Editor dapat mengetahui kategori dari dokumen berita politik yang dimilikinya dengan cara mengcopy teks berita tersebut dalam *text box* yang sudah disediakan. Guest dapat melihat data berita politik dan membaca berita.
4. Tahapan pembangunan sistem dalam penelitian ini dibatasi hingga tahapan pengujian sistem dan tidak melakukan *unit testing*.

1.6 Sistematika Penulisan

Sistematika dalam penulisan tugas akhir ini adalah sebagai berikut:

1. Pendahuluan
Bab ini menjelaskan mengenai latar belakang, perumusan masalah, tujuan, batasan masalah, dan sistematika penulisan.
2. Tinjauan Pustaka

Bab ini menjelaskan tentang penelitian terdahulu, kajian-kajian teori terkait dengan penelitian yang dapat mendukung penelitian yang dilakukan.

3. Metode Penelitian

Bab ini memaparkan tentang metode yang digunakan dalam penelitian ini yang meliputi tahap pengumpulan data dan analisis data, model pengembangan sistem, dan gambaran umum sistem yang akan dibangun.

4. Analisis dan Pengembangan Sistem

Bab ini menguraikan tentang analisis dan pengembangan sistem yang dibangun.

5. Hasil dan Pembahasan

Bab ini menjelaskan mengenai hasil penelitian yang telah dilakukan serta pembahasan sistem yang telah dibuat. Pembahasan dilakukan guna menjelaskan dan memaparkan bagaimana penelitian ini menjawab perumusan masalah serta tujuan dan manfaat dari penelitian ini seperti apa yang telah ditentukan pada awal penelitian.

6. Penutup

Bab ini terdiri atas kesimpulan atas penelitian yang telah dilakukan dan saran untuk penelitian selanjutnya.

BAB 2. TINJAUAN PUSTAKA

Bab ini dipaparkan tinjauan yang berkaitan dengan masalah yang dibahas, kajian teori yang berkaitan dengan masalah, dan juga penelitian-penelitian terdahulu.

2.1 Penelitian Terdahulu

Penelitian di tahun 2016 yang dilakukan oleh Agus Setiawan, Indah Fitri Astuti, dan Awang Harsa yang berjudul “Klasifikasi dan Pencarian Buku Referensi Akademik Menggunakan Metode *Naïve Bayes Classifier* (NBC)”. Penelitian dilakukan di Perpustakaan Daerah Provinsi Kalimantan Timur untuk mengklasifikasikan buku-buku perpustakaan sesuai dengan kategorinya agar pengunjung dapat mencari dengan tepat buku yang sesuai dengan kebutuhan mereka. Hasil dari penelitian ini mengklasifikasikan 5 kategori buku dengan menggunakan data latih sebanyak 250 buku yang masing-masing kategorinya terdiri 50 buku dan data uji yang digunakan sebanyak 150 buku. Dalam kesimpulannya, jumlah data latih sangat berpengaruh dalam proses klasifikasi, karena semakin banyak data latih maka tingkat keakuratan dalam penentuan kategori akan semakin tinggi (Setiawan et al., 2016).

Penelitian lainnya yang berjudul “Klasifikasi Teks Dengan *Naïve Bayes Classifier* (NBC) Untuk Pengelompokan Teks Berita Dan Abstract Akademi”. Tujuan penelitian ini adalah mengklasifikasikan dokumen teks baik dokumen berita maupun dokumen akademik. Penelitian ini menggunakan data sebanyak 1000 dokumen berita dan 450 dokumen abstrak akademik. Hasil penelitian menunjukkan pada dokumen berita akurasi yang didapat maksimal mencapai 91% sedangkan pada dokumen akademik 82% (Hamzah, 2012a).

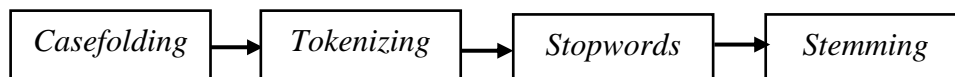
Penelitian lain yang berjudul “Penerapan Metode *Naive Bayes* Untuk Sistem Klasifikasi SMS pada *Smartphone Android* (Ebranda et al., 2013). Penelitian ini mengklasifikasikan suatu sms yang masuk apakah spam dan bukan spam secara

otomatis. Penerapan *Naïve Bayes Classifier* dapat mengklasifikasikan spam dengan baik dari data latih sebanyak 80 SMS sehingga memperoleh tingkat akurasi 85,11%.

Kesimpulan dari ketiga penelitian tersebut dapat memberikan gambaran bahwa penggunaan metode *Naïve Bayes Classifier* dalam proses klasifikasi terhadap sebuah *object* dapat menghasilkan akurasi yang tinggi dan meminimalisir *misclassification*. Diharapkan penggunaan metode ini juga dapat dilakukan dalam proses klasifikasi konten berita politik.

2.2 *Text Mining* dan Klasifikasi

Text Mining adalah suatu proses menggali informasi yang baru dan tidak diketahui sebelumnya yang berasal dari sumber yang diekstrak secara otomatis yang berupa data teks tak terstruktur (Sussolaikah and Alwi, 2016). Informasi yang diperoleh dari *text mining* biasanya melalui pencarian pola dan kecenderungan melalui sarana seperti pembelajaran pola *statistic*. Proses *text mining* biasanya melibatkan proses penataan teks input, menentukan pola dalam data tidak terstruktur dan akhirnya mengevaluasi serta menginterpretasi *output* (Saraswati, 2013). Pengelolaan *text mining* terdapat tahapan pemrosesan awal terhadap suatu teks yang disebut *text preprocessing*. *Text preprocessing* merupakan suatu proses pengubahan bentuk data tekstual yang belum terstruktur menjadi data yang terstruktur. Proses *preprocessing* ini meliputi 4 tahapan yaitu *casefolding*, *tokenizing*, *stopword*, dan *stemming* (Rahman and Doewes, 2017) yang ditunjukkan pada Gambar 2.1 yaitu:

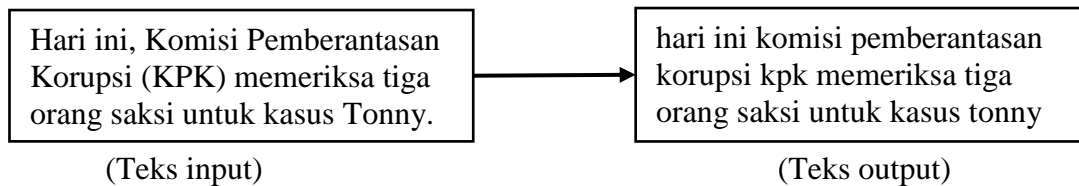


Gambar 2. 1 Tahapan *Text Preprocessing*

a. *Casefolding*

Tidak semua dokumen teks konsisten dalam penggunaan huruf kapital. Oleh karena itu, peran *casefolding* dibutuhkan dalam mengkonversi keseluruhan teks dalam dokumen menjadi suatu bentuk standar. *Casefolding* adalah mengubah semua huruf dalam dokumen menjadi huruf kecil. Hanya huruf ‘a’ sampai dengan ‘z’ yang diterima. Karakter selain huruf dihilangkan dan dianggap *delimiter*.

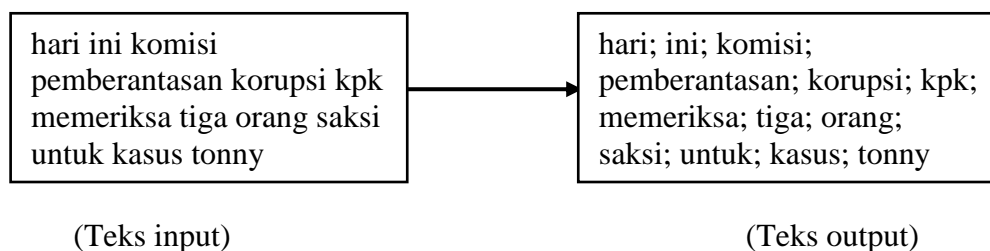
Sebagai contoh, user yang ingin mendapatkan informasi “KOMPUTER” dan mengetik “KOMPUTER”, “KomPUter”, atau “komputer”, tetap diberikan hasil yang sama yakni “komputer”. Implementasi *casefolding* pada kode program sistem menggunakan php *strlower* (Rahman and Doewes, 2017). Contoh konten berita yang melalui tahap *casefolding* ditunjuk pada Gambar 2.2



Gambar 2. 2 Tahap *Casefolding*

b. *Tokenizing*

Tahap *Tokenizing* adalah tahap pemotongan *string input* atau kalimat berdasarkan kata yang menyusunnya. Implementasi *tokenizing* pada kode program sistem menggunakan fungsi *explode* pada php (Rahman and Doewes, 2017). Contoh konten berita yang melalui tahap ini dapat dilihat pada Gambar 2.3.

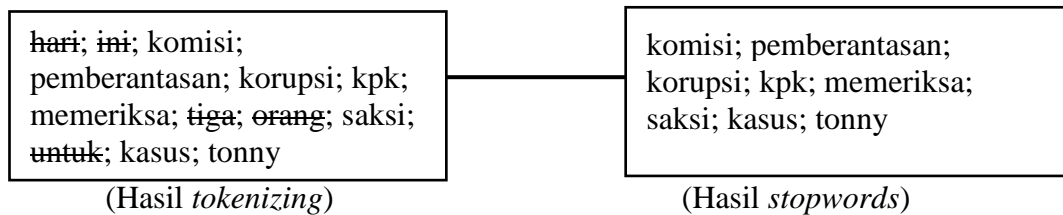


Gambar 2. 3 Tahap *Tokenizing*

c. *Stopwords*

Tahap *Stopwords* adalah tahap mengambil kata-kata penting dari hasil token. *Stopwords* merupakan tahap mengambil kata-kata penting dari hasil *token* atau menghapus kata-kata yang terlalu umum sehingga tidak memiliki makna atau karakter khusus terhadap suatu kategori atau kelas tertentu. Sebelum melakukan *stopword removal* harus ada daftar *stopword (stoplist)*. Jika termasuk dalam *stoplist* maka kata-kata tersebut dihapus sehingga kata-kata yang tersisa dianggap kata-kata yang mencirikan isi dari dokumen atau *keyword* (Manalu, 2014). Contoh *stopword* adalah “yang”, “dan”, “di” dst. Kata-kata seperti “dari”, “yang”, “di” dan “ke”

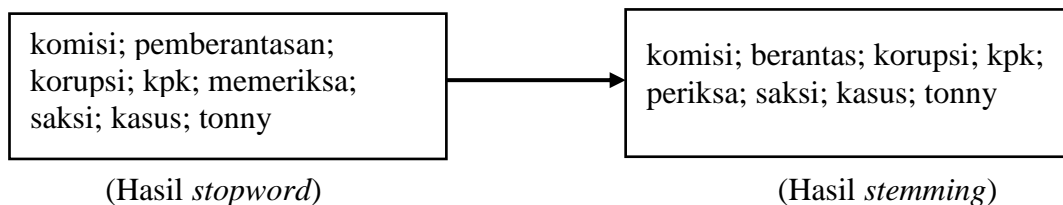
adalah beberapa contoh kata-kata yang berfrekuensi tinggi dan dapat ditemukan hampir dalam setiap dokumen. Penghilangan *stopwords* ini dapat mengurangi ukuran *index* dan waktu pemrosesan. Dalam penelitian, *stopword* yang digunakan berasal dari Sastrawi (Chandra et al., 2016b). Contoh konten berita yang sudah melalui tahap *stopword* dapat dilihat pada Gambar 2.4.



Gambar 2. 4 Tahap *Stopwords*

d. *Stemming*

Teknik *Stemming* diperlukan untuk melakukan pengelompokan kata-kata yang memiliki kata dasar (*root word*) dan arti yang serupa namun memiliki bentuk yang berbeda karena mendapatkan imbuhan yang berbeda. Sebagai contoh kata bersama, kebersamaan, dan menyamai akan dikembalikan ke bentuk awal kata dasarnya yaitu “*sama*”. Proses *stemming* pada teks berbahasa Indonesia semua kata imbuhan baik itu *prefiks* (awalan) dan *sufiks* (akhiran) juga dihilangkan. Pada penelitian ini, proses *stemming* menggunakan algoritma *Confix-Stripping* (CS) yang dikembangkan oleh Jelita Asian yang merupakan penyempurnaan dari algoritma Nazief dan Adriani (Chandra et al., 2016b). Contoh hasil *stemming* dapat dilihat pada Gambar 2.5.



Gambar 2. 5 Tahap *Stemming*

Terkaitannya dengan pengelompokan dokumen berupa teks, klasifikasi adalah salah satu teknik yang sangat sering digunakan untuk kebutuhan tersebut. Klasifikasi adalah proses untuk menemukan model atau fungsi yang dapat

membedakan konsep atau kelas data yang belum diketahui kelasnya untuk menentukan kelas dari suatu obyek (Bustami, 2013). Penentuan kelas dari suatu dokumen dilakukan dengan cara membandingkan nilai probabilitas suatu sampel berada di kelas yang satu dengan nilai probabilitas suatu sampel berada di kelas yang lain. Dengan menggunakan atribut dan data *training*, metode-metode klasifikasi tersebut dapat memprediksi kelas dari data lain yang belum di klasifikasikan (Setyaji, 2016).

2.3 *Naïve Bayes Classifier*

Algoritma *Naive Bayes Classifier* merupakan algoritma yang digunakan untuk mencari nilai probabilitas tertinggi untuk mengklasifikasi data uji pada kategori yang paling tepat (Pakpahan and Widyastuti, 2014). Ada dua tahap pada klasifikasi dokumen. Tahap pertama adalah proses pelatihan, yaitu membentuk model berdasarkan dokumen yang sudah diketahui kategorinya. Sedangkan tahap kedua adalah proses klasifikasi dokumen, yaitu menentukan kelas dari dokumen yang belum diketahui kelasnya berdasarkan model yang dibentuk. Persamaan umum teorema *bayes* ditunjukkan pada Persamaan 2.1 sebagai berikut (Anugroho et al., 2010).

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \dots\dots\dots(2.1)$$

Keterangan :

$P(H|X)$ = probabilitas hipotesis H berdasarkan kondisi X

$P(X|H)$ = probabilitas X berdasarkan kondisi hipotesis H

$P(H)$ = probabilitas hipotesis H

$P(X)$ = probabilitas X

Proses pengimplementasian algoritma *Naïve Bayes Classifier*, kumpulan kelas dokumen direpresentasikan dengan nilai V misalnya kelas Pilkada (V_{pilkada}), kelas UU Ormas ($V_{\text{UU Ormas}}$), dan kelas *Reshuffle* Kabinet ($V_{\text{Reshuffle Kabinet}}$). Setiap dokumen direpresentasikan dengan pasangan atribut “ $X_1, X_2, X_3, \dots, X_n$ ” dimana X_1 adalah kata pertama, X_2 adalah kata kedua dan seterusnya. Pada proses klasifikasi,

biasanya terdapat sejumlah kelas yang telah dibentuk dari tahapan proses pelatihan (Anugroho et al., 2010).

Proses pelatihan, probabilitas kelas $P(V_j)$ dan probabilitas kata terhadap kelas $P(x_i|V_j)$ dihitung menggunakan rumus yang persamaannya dapat ditulis pada Persamaan 2.2 dan Persamaan 2.3 sebagai berikut (Anugroho et al., 2010).

$$P(V_j) = \frac{|docs\ j|}{|contoh|} \dots\dots\dots(2.2)$$

$$P(x_i|V_j) = \frac{n_k+1}{n+|kosakata|} \dots\dots\dots(2.3)$$

Keterangan :

$|docs\ j|$ = Jumlah dokumen setiap kategori j

$|contoh|$ = Jumlah semua kategori

n_k = Jumlah frekuensi kemunculan setiap kata pada dokumen

n = Jumlah frekuensi kemunculan kata dari setiap kategori

$|kosakata|$ = Jumlah semua kosakata dari semua kategori

Setelah selesai proses pelatihan, maka tahapan berikutnya adalah melakukan proses klasifikasi terhadap data uji dengan menggunakan algoritma *Naïve Bayes Classifier*. Algoritma ini digunakan untuk mencari probabilitas tertinggi dari semua kelas berdasarkan data latih yang telah dibentuk. Probabilitas tertinggi dikenal sebagai *maximum a posteriori* (nilai *posterior probability* tertinggi diantara semua kelas) atau yang biasanya disebut dengan V_{MAP} . Persamaan V_{MAP} dapat ditulis sebagai berikut (Persamaan 2.4) (Anugroho et al., 2010).

$$V_{MAP} = \underset{V_j \in V}{\arg \max} \frac{P(x_1, x_2, x_3, \dots, x_n | V_j) P(V_j)}{P(x_1, x_2, x_3, \dots, x_n)} \dots\dots\dots(2.4)$$

Keterangan :

$P(x_1, x_2, x_3, \dots, x_n | V_j)$ = probabilitas kemunculan kata x , di dalam dokumen kelas J

$P(V_j)$ = probabilitas dokumen kelas J di dalam semua dokumen

$P(x_1, x_2, x_3, \dots, x_n)$ = probabilitas pola independen kata X_n

Untuk $P(x_1, x_2, x_3, \dots, x_n)$ nilainya konstan untuk semua kategori (V_j) sehingga Persamaan 2.4 dapat disederhanakan pada Persamaan 2.5. Atau dapat ditulis sebagai berikut (Persamaan 2.6) (Anugroho et al., 2010).

$$V_{MAP} = \arg \max_{V_j \in V} P(x_1, x_2, x_3, \dots, x_n | V_j) P(V_j) \dots \dots \dots (2.5)$$

$$V_{MAP} = \arg \max_{V_j \in V} \prod_{i=1}^n P(x_i | V_j) P(V_j) \dots \dots \dots (2.6)$$

2.4 Uji Performansi

Uji performansi adalah salah satu cara untuk mengevaluasi kinerja dari suatu metode. Evaluasi klasifikasi dilakukan dengan menguji hasil klasifikasi yang benar maupun yang salah. Performa dari sistem tersebut biasanya dievaluasi menggunakan data dalam sebuah *confussion matrix*. *Confussion matrix* adalah sebuah tabel yang menyatakan jumlah data uji hasil klasifikasi yang benar dan yang salah (Ling et al., 2014). Contoh tabel *confussion matrix* adalah sebagai berikut (Tabel 2.1).

Tabel 2. 1 Matrik Kontingensi Evaluasi Kinerja Sistem

		Kelas Prediksi		
		A	B	C
Kelas Aktual	A	TP _{AA}	AB	AC
	B	BA	TP _{BB}	BC
	C	CA	CB	TP _{CC}

Keterangan :

1. TP (*true positive*) adalah jumlah data uji yang berada di kelas aktual dan prediksi yang positif sesuai. Contoh nilai TP adalah AA, BB, dan CC.
2. FP (*false positive*) adalah kelas yang diprediksi positif dan ternyata faktanya negative. Jumlah total FP untuk setiap kelas adalah jumlah nilai dalam kolom yang sesuai kecuali nilai TP. Contoh nilai FP untuk Kelas A adalah jumlah dari kolom BA dan CA.
3. FN (*false negative*) adalah kelas yang diprediksi negative dan ternyata faktanya positif. Jumlah total FN untuk kelas adalah jumlah nilai dalam baris yang sesuai

kecuali nilai TP. Contoh nilai FN untuk Kelas A adalah jumlah dari baris AB dan AC.

4. TN (*true negative*) adalah kelas yang diprediksi negatif dan ternyata faktanya negatif. Jumlah total TN untuk kelas tertentu adalah jumlah semua kolom dan baris yang mengecualikan kolom dan baris kelas tersebut. Contoh nilai TN untuk Kelas A adalah jumlah dari BB, BC, CB, dan CC.

Pengukuran performansi sistem akan dilakukan dengan menghitung *precision*, *recall*, dan *f-measure*. *Precision* adalah rasio jumlah ketepatan prediksi suatu kelas terhadap jumlah total prediksi yang diklasifikasikan dalam kelas tersebut. *Precision* dapat diartikan sebagai ketepatan atau kecocokan antara permintaan informasi dengan jawaban terhadap permintaan tersebut. Sebagai contoh, jika seseorang mencari dokumen tentang “politik” dan sistem tersebut memiliki 100 dokumen, apabila performa sistem tersebut baik, maka sistem akan menemukan 100 dokumen tentang politik. Jika dari 100 dokumen tersebut, hanya 50 dokumen politik yang ditemukan maka nilai *precisionnya* adalah 50% (Ling et al., 2014). Rumus mencari *precision* dapat ditulis seperti Persamaan 2.7 sebagai berikut:

$$Precision = \frac{TP}{TP+FP} \times 100\% \dots\dots\dots(2.7)$$

Recall adalah rasio jumlah ketepatan prediksi suatu kelas terhadap jumlah total fakta yang diklasifikasikan dalam kelas tersebut. *Recall* juga dapat diartikan sebagai tingkat keberhasilan sistem dalam menemukan kembali informasi yang relevan. Semakin rendah nilai *recall* maka efektivitas sistem dinyatakan kurang baik. Sebagai contoh, jika suatu sistem memberikan 100 dokumen dan hanya ada 10 dokumen yang relevan, maka nilai *recallnya* adalah 10% (Ling et al., 2014). Rumus mencari *recall* dapat ditulis pada Persamaan 2.8 sebagai berikut:

$$Recall = \frac{TP}{TP+FN} \times 100\% \dots\dots\dots(2.8)$$

Hasil observasi dari berbagai penelitian yang ditemukan, kenyataan bahwa nilai *recall* dan *precision* ini cenderung berlawanan atau berbanding terbalik. Jika *recall* tinggi, besar kemungkinan *precision* rendah. Sebuah sistem informasi akan dianggap baik jika tingkat *recall* maupun *precisionnya* tinggi atau sama besarnya

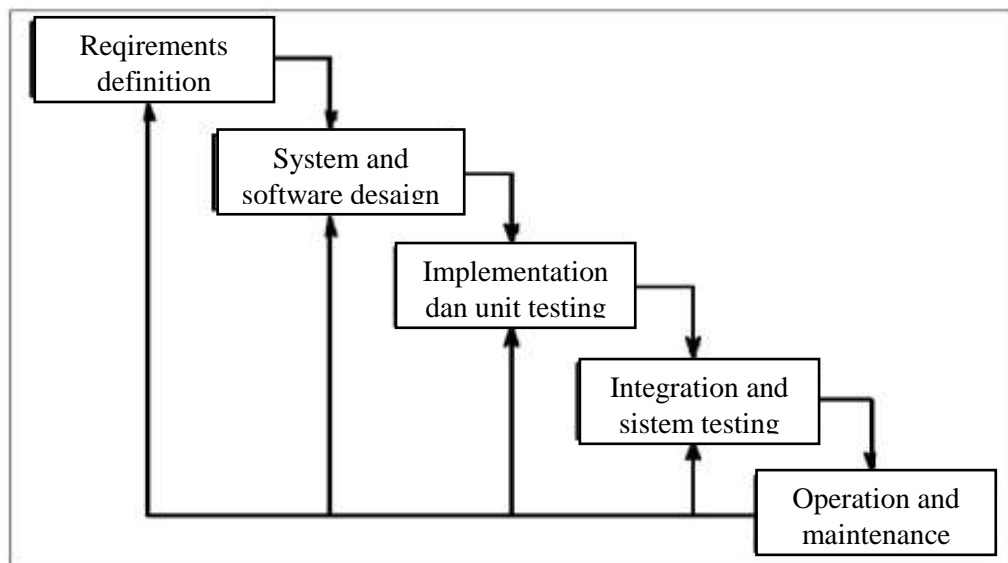
(1:1). Selain itu, suatu sistem dinyatakan efektif apabila hasil penelusuran mampu menunjukkan *precision* yang tinggi sekalipun *recall*nya rendah (Lestari, 2016).

F-Measure adalah kombinasi rata-rata *harmonic* dari *precision* dan *recall* yang berbanding lurus dengan nilai keduanya. Nilai *recall* dan *precision* pada suatu keadaan dapat memiliki bobot yang berbeda. Rumus mencari *F-Measure* dapat ditulis sebagai Persamaan 2.9 berikut ini: (Ling et al., 2014)

$$F_{measure} = \frac{2*precision*recall}{precision+recall} \dots\dots\dots (2.9)$$

2.5 Pengembangan Sistem

Model *Waterfall* adalah model klasik yang bersifat sistematis sekuensial dimana perkembangan terlihat seperti mengalir semakin ke bawah melalui daftar tahapan yang harus dijalankan dalam membangun *software* (Pressman, 2010). Tahapan model *Waterfall* dapat ditunjukkan pada Gambar 2.6.



Gambar 2. 6 Model Waterfall

(Sumber: Pressman, 2010)

Berdasarkan Gambar 2.6 dapat dilihat bahwa dalam penggunaan model *Waterfall* dilakukan secara bertahap dan apabila ada kesalahan maka harus kembali ke tahap sesuai pada Gambar 2.6. Tahap pertama yaitu menganalisis kebutuhan (*requirement*) dalam pembuatan sistem. Tahap kedua yaitu dengan mendesain

sistem yang akan dibuat dan tahap ketiga yaitu mulai dengan mengimplementasikan koding ke dalam sistem. Tahap terakhir yaitu pengujian sistem dan pemeliharaan sistem.

2.5.1 Analisis Kebutuhan (*Requirement*)

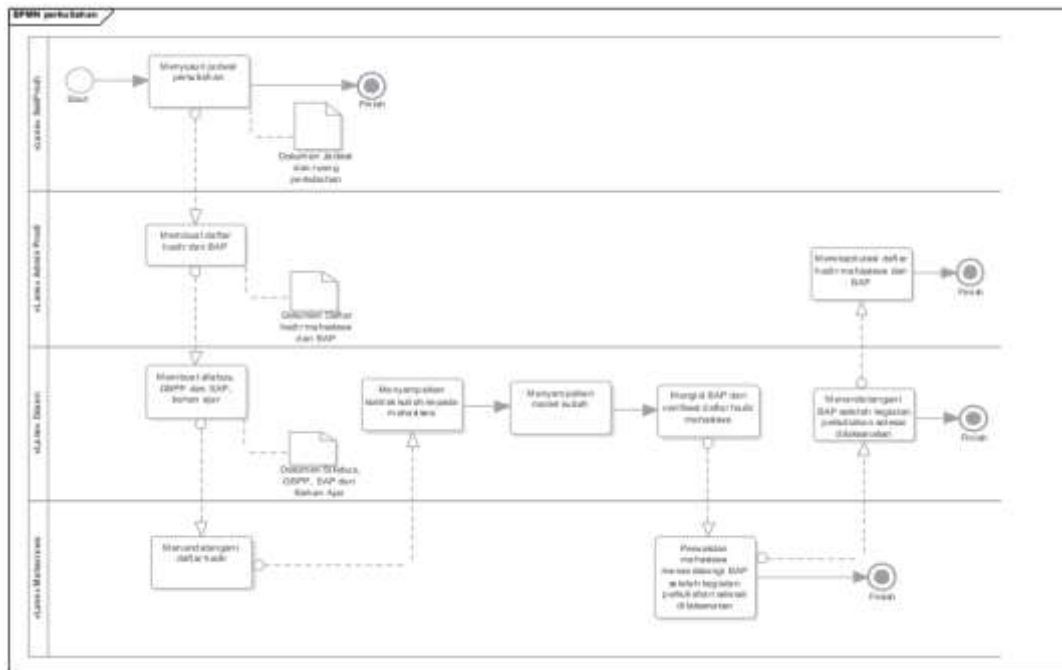
Tahapan awal dalam perancangan dan pengembangan sistem adalah analisis kebutuhan (*requirement*) tentang bagaimana sistem dibangun dari kebutuhan sistem berupa fitur yang diperlukan serta menentukan kebutuhan fungsional dan nonfungsional (Romadhoni et al., 2015). Kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan agar sistem dapat berjalan dengan baik. Sedangkan, kebutuhan non-fungsional menggambarkan kebutuhan luar sistem yang diperlukan untuk menjelaskan sistem yang dibangun (Sasmito, 2017).

2.5.2 Desain Sistem

Tahap desain sistem dibangun menggunakan beberapa model desain salah satunya adalah *Unified Modeling Language* (UML) yang mendukung konsep pemodelan *programming* berbasis objek. Desain sistem dilakukan untuk memberikan gambaran secara umum mengenai sistem yang akan dibangun. Penjelasan pemodelan UML yang digunakan adalah sebagai berikut (Manalu, 2014):

a. *Business Process Modelling Notation*

BPMN adalah singkatan dari *Business Process Modeling Notation* yaitu suatu metodologi baru yang dikembangkan oleh *Business Process Modeling Initiative* sebagai standar baru pada pemodelan proses bisnis dan juga sebagai alat desain pada sistem yang kompleks. Tujuan utama dari BPMN adalah menyediakan notasi yang mudah digunakan dan bisa dimengerti oleh semua orang yang terlibat dalam bisnis analisis yang memodelkan proses bisnis dan pengembang teknik yang membangun sistem. Berikut contoh gambar BPMN dapat dilihat pada Gambar 2.7 (Rosmalal and Falahah, 2007).



Gambar 2. 7 Contoh *Bussiness Process Modelling Notation* Perkuliahan

(Sumber: Yunitarini and Hastarita, 2016)

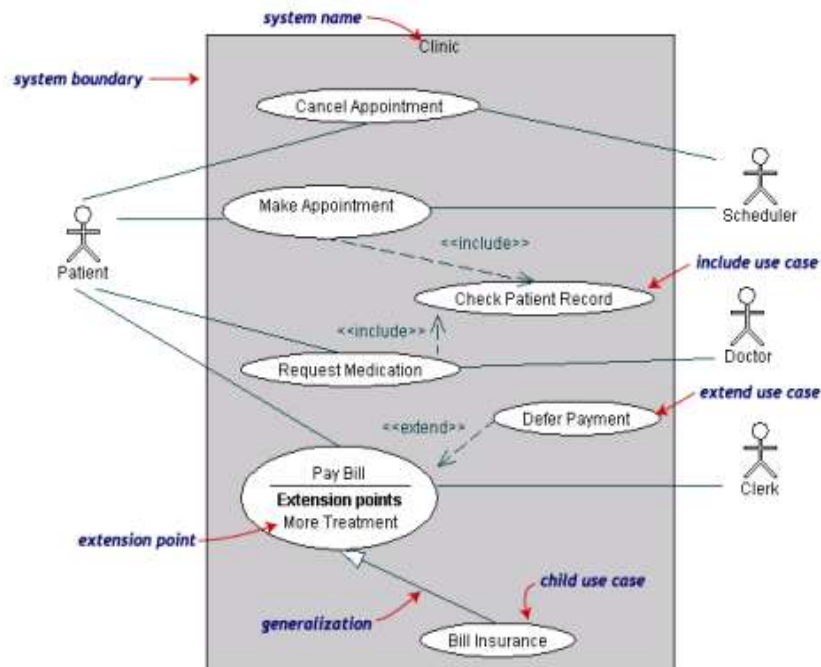
b. *Diagram Usecase*

Diagram usecase merupakan salah satu diagram yang menggambarkan fungsionalitas dari suatu sistem. Diagram *usecase* terdiri dari beberapa elemen antara lain aktor, *case*, dan relasi. Contoh gambar *diagram usecase* dapat dilihat pada Gambar 2.8 (Manalu, 2014).

1. Aktor adalah pengguna yang berinteraksi dengan suatu sistem. Aktor mempresentasikan peran dari sistem. Contoh penamaan aktor seperti admin, pengunjung, pegawai, dan lain sebagainya.

2. *Case* merupakan gambaran fitur sistem.

3. Relasi merupakan keterhubungan yang terjadi antara *case* satu dengan *case* lainnya. Ada beberapa relasi yang terjadi pada *diagram usecase* antara lain: *include*, *extends*, dan *communicate*.



Gambar 2. 8 Contoh *Diagram Usecase*
(Sumber: Dharwiyanti and Wahono, 2003)

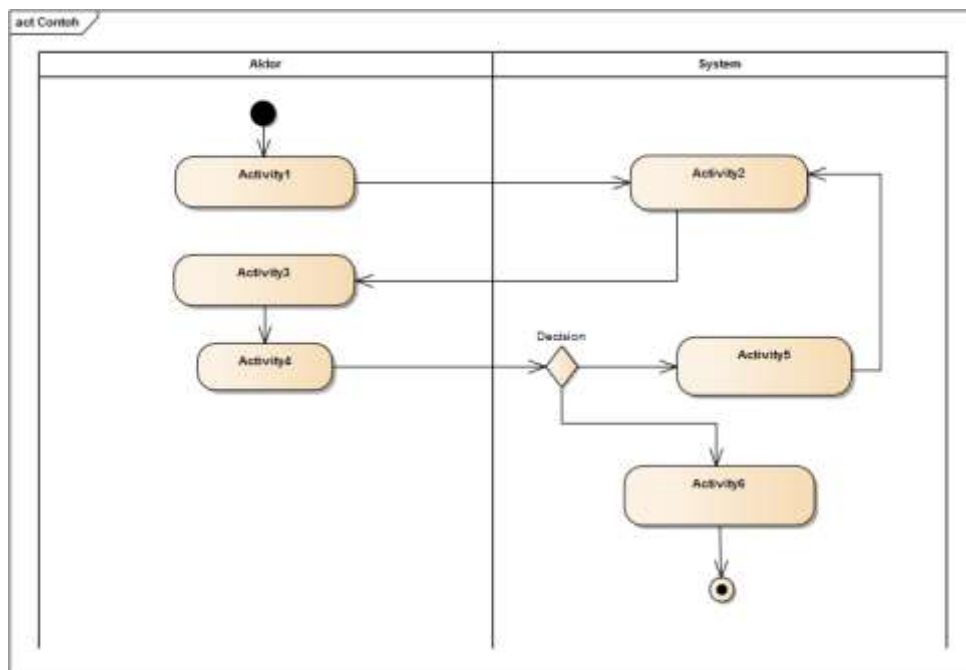
c. Skenario *Usecase*

Skenario *usecase* merupakan penjabaran dari *diagram usecase*. Skenario dilakukan berdasarkan *case* yang ada pada *diagram usecase* dan biasanya terdiri dari: (Manalu, 2014)

1. Tujuan *usecase* yaitu menjelaskan apa tujuan dari *case* yang terjadi.
2. Deskripsi yaitu menjelaskan apa yang terjadi pada *case*.
3. Skenario menjelaskan cara kerja *case* mulai dari awal hingga akhir.
4. Kondisi awal yaitu keadaan apa yang terjadi sebelum *case* berlangsung.
5. Kondisi akhir yaitu keadaan apa atau apa *output* yang dihasilkan setelah *case* berlangsung.

d. *Activity Diagram*

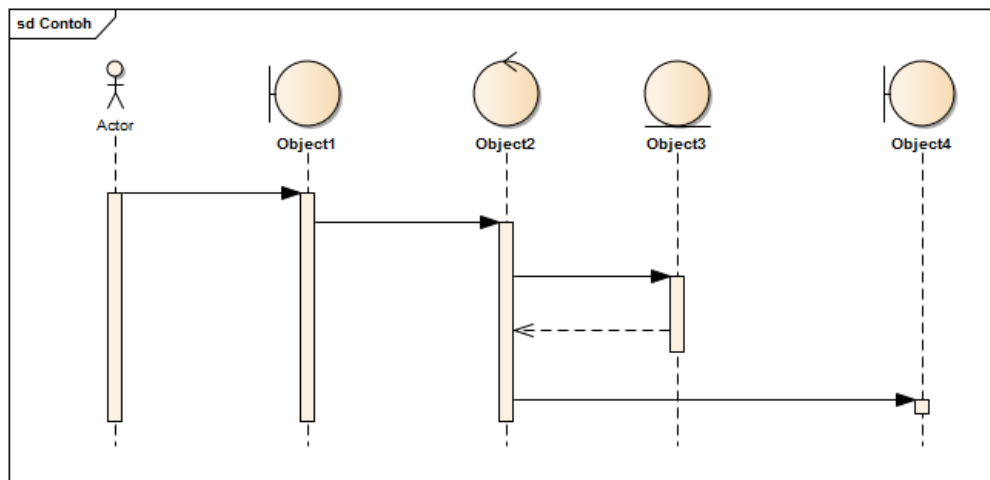
Activity Diagram adalah diagram *flowchart* yang diperluas untuk menunjukkan aliran kendali satu aktivitas ke aktivitas yang lain. Diagram aktifitas digunakan untuk memodelkan aspek dinamis sistem. Diagram aktivitas berupa operasi-operasi dan aktivitas-aktivitas di *usecase*. Contoh gambar *activity diagram* dapat dilihat pada Gambar 2.9 (Manalu, 2014).



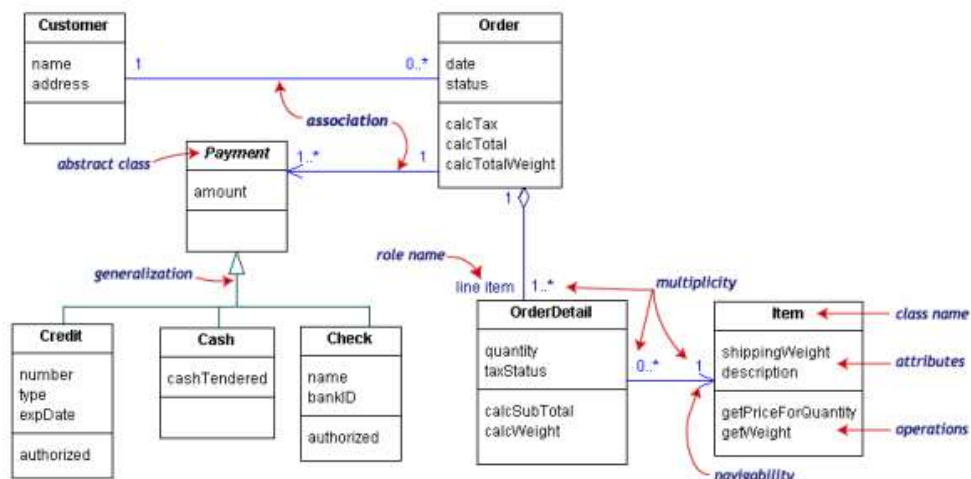
Gambar 2. 9 Contoh *Activity Diagram*

e. *Sequence Diagram*

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk *user*, *view* dan sebagainya) berupa *message* (pesan) yang digambarkan terhadap waktu. *Sequence diagram* digunakan untuk memodelkan skenario penggunaan. Skenario penggunaan adalah barisan kejadian yang terjadi selama satu eksekusi sistem. *Sequence diagram* menunjukkan objek sebagai garis vertical dan tiap kejadian sebagai panah horizontal dari objek pengirim ke objek penerima. Contoh gambar *sequence diagram* dapat dilihat pada Gambar 2.10 (Manalu, 2014).

Gambar 2. 10 Contoh *Sequence Diagram*f. *Class Diagram*

Class Diagram menggambarkan struktur dan penjelasan class, *package*, dan objek serta hubungan satu sama lain seperti pewarisan, asosiasi, dan lain-lain. Selain itu *class diagram* juga menjelaskan hubungan antar class dalam sebuah sistem yang sedang dirancang sehingga bagaimana caranya setiap *class* saling berkolaborasi untuk mencapai sebuah tujuan. Contoh *class diagram* dapat dilihat pada Gambar 2.11 (Dharwiyanti and Wahono, 2003).

Gambar 2. 11 Contoh *Class Diagram*

(Sumber: Dharwiyanti and Wahono, 2003)

Unit Testing adalah proses pengujian fungsi pada suatu sistem untuk memastikan bahwa fungsi yang digunakan terbebas dari bug. Penggunaan framework untuk pengujian suatu fungsi akan sangat memudahkan proses pengujian karena pada framework sudah terdapat beberapa fungsi generik yang dapat langsung digunakan. Pengujian ini dilakukan oleh developer (Adi, 2015).

2.5.4 Pengujian Sistem

Pengujian sistem sangat diperlukan untuk memastikan *software* yang sudah atau sedang dibuat dapat berjalan sesuai dengan fungsionalitas yang diharapkan. Pengembang atau penguji *software* harus menyiapkan sesi khusus untuk menguji program yang sudah dibuat agar kesalahan ataupun kekurangan dapat dideteksi sejak awal dan dikoreksi secepatnya. Pengujian sendiri merupakan elemen kritis dari jaminan kualitas perangkat lunak dan merupakan bagian yang tidak terpisahkan dari siklus hidup pengembangan *software* seperti halnya analisis, desain, dan pengkodean. Dalam pengujian sistem ada sejumlah pengujian yang telah digunakan salah satunya adalah *Black Box Testing*. *Black Box Testing* berfokus pada spesifikasi fungsional dari perangkat lunak. *Tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program (Mustaqbal et al., 2015).

2.5.5 Pemeliharaan Sistem

Pemeliharaan diadakan untuk mengatasi masalah pada sistem dilain waktu ketika sistem sudah dapat digunakan oleh *user*. Selama *user* menemui *bug* pada sistem, maka *user* langsung konfirmasi kepada *developer* untuk segera ditangani (Sagita and Sugiarto, 2016).

BAB 3. METODOLOGI PENELITIAN

Bab ini menjelaskan tentang metode yang digunakan dalam merancang dan membangun sistem menggunakan metode pengembangan.

3.1 Gambaran Umum Sistem

Sistem klasifikasi dokumen yang dikembangkan adalah sebuah sistem berbasis web yang dapat mengklasifikasikan kategori berita politik dengan menggunakan metode *Naive Bayes Classifier*. Seperti yang telah disebutkan dalam batasan masalah, input data sistem ini adalah teks tentang berita politik (yang dalam penelitian ini disebut sebagai dokumen berita politik) dan diharapkan dapat mengklasifikasikan dokumen berita politik sesuai dengan kategori yang lebih spesifik. Sistem klasifikasi dokumen ini memiliki 4 fitur utama didalamnya antara lain: data *training*, data *testing*, data *stopword*, dan cek kategori dokumen. Fitur data training, data testing, data *stopword* hanya dapat diakses oleh admin. Admin harus mempunyai email dan password untuk autentifikasi login kedalam sistem. Sedangkan *editor* tidak membutuhkan autentifikasi login karna fitur cek kategori dokumen yang digunakan dapat diakses tanpa mempunyai email dan password. Fitur cek kategori dokumen digunakan untuk mengetahui kategori dokumen yang lebih spesifik. Guest (Tamu) dapat melihat berita politik yang lebih spesifik yang telah diinputkan oleh Editor.

3.2 Jenis Penelitian

Jenis penelitian yang dilakukan merupakan penelitian terapan. Penelitian terapan bertujuan untuk memberikan solusi atas permasalahan tertentu secara praktis. Penelitian ini tidak berfokus pada pengembangan sebuah ide, teori, atau gagasan, tetapi lebih fokus kepada penerapan penelitian dalam kehidupan sehari-hari. Pada penelitian ini, sistem yang dikembangkan menerapkan metode *Naive Bayes Classifier* dengan mengimplementasikan ke dalam sebuah sistem klasifikasi dokumen berita politik.

3.3 Tahapan Penelitian

Tahapan penelitian merupakan urutan langkah penelitian yang dilakukan oleh peneliti berdasarkan model pengembangan yang digunakan. Pada penelitian ini dilakukan menggunakan model SDLC (*System Development Life Cycle*) *waterfall*. Model *waterfall* ini dipilih karena mempunyai tahapan pengembangan sistem yang dilakukan secara terurut, mulai dari analisis kebutuhan, desain sistem, implementasi sistem dan pengujian sistem. Sistem klasifikasi dokumen ini tergolong sistem yang tidak terlalu kompleks dan pengerjaannya dilakukan oleh perseorangan sehingga model *waterfall* sangat sesuai jika diterapkan pada penelitian ini. Tahapan penelitian model *waterfall* di mulai dari analisa kebutuhan, desain sistem, implementasi sistem, dan pengujian sistem.

3.3.1 Analisa Kebutuhan

Tahap pertama pada proses pengembangan sistem adalah analisis kebutuhan. Analisis kebutuhan merupakan langkah menganalisis kebutuhan-kebutuhan sistem informasi yang akan dibangun. Kebutuhan sistem informasi dibagi menjadi 2 yaitu kebutuhan fungsional dan kebutuhan non-fungsional. Kebutuhan yang didapat berasal dari data yang telah dikumpulkan dan telah diolah yang berhubungan dengan rekomendasi menu makanan yang melibatkan kalori. Data-data tersebut kemudian dikelompokkan menjadi kebutuhan fungsional dan kebutuhan non-fungsional. Kebutuhan fungsional dari sistem ini yaitu sistem dapat cek kategori berita, mengelola data *training*, mengelola data *testing* hingga mengelola *stopword* sehingga dapat membantu menentukan hasil klasifikasi. Analisis kebutuhan non-fungsional menggambarkan kebutuhan diluar sistem yang diperlukan untuk menjalankan sistem yang dibangun.

Untuk menentukan kebutuhan fungsional dan non-fungsional, maka dilakukan pengumpulan data:

1. Melakukan studi *literature*, jurnal, media maupun internet mengenai pembangunan sistem klasifikasi dokumen seperti data berita dan metode yang digunakan.

3.3.2 Desain Sistem

Pembuatan desain sistem menggunakan *Unified Modeling Language* (UML) yang keseluruhannya menggunakan *tools Enterprise Modelling*. Pemodelan UML yang digunakan adalah sebagai berikut :

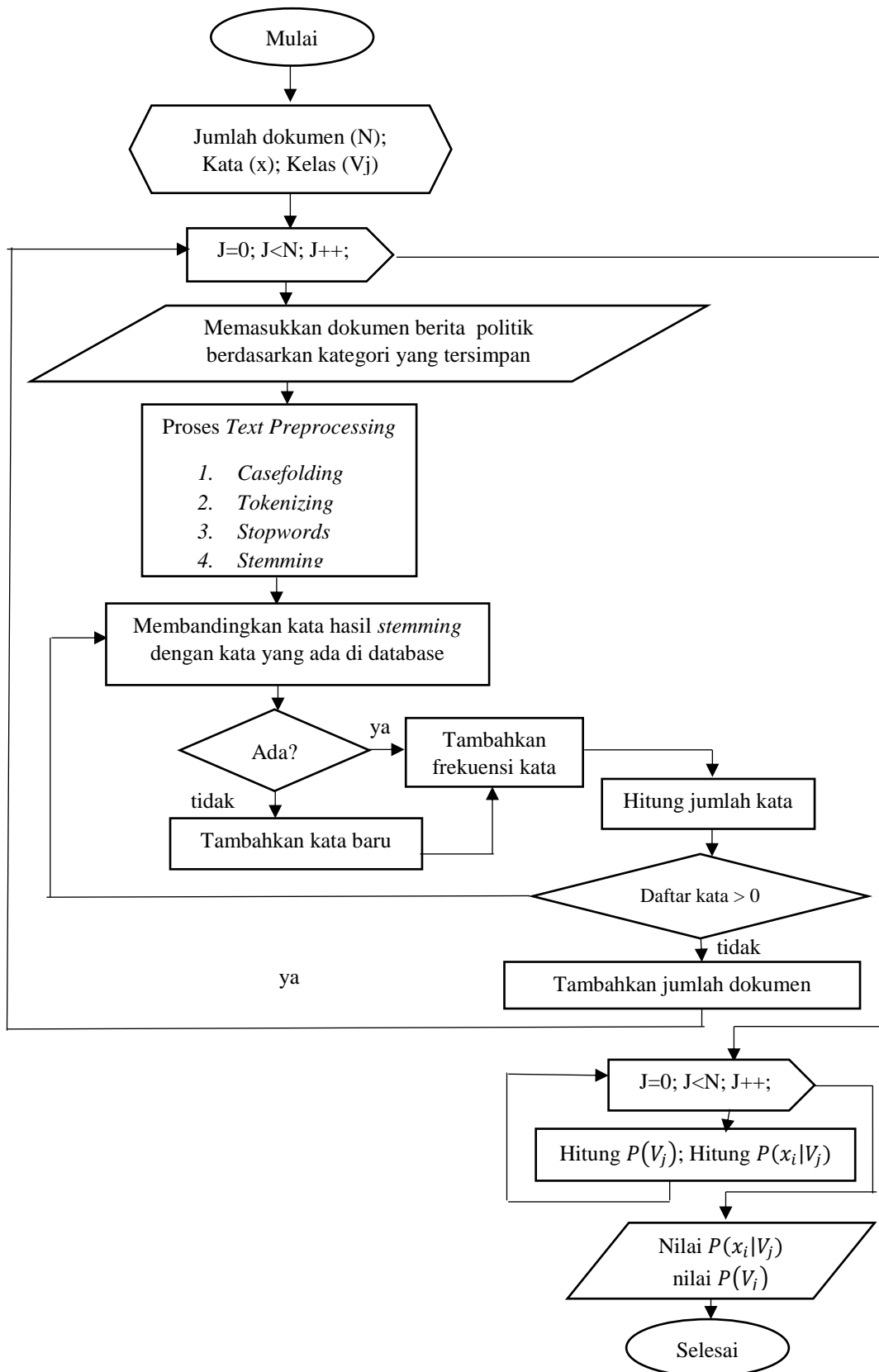
1. *Business Process Modeling Notation*
2. *Usecase Diagram*
3. *Skenario Usecase*
4. *Activity Diagram*
5. *Sequence Diagram*
6. *Class Diagram*
7. *Entity Relationship Diagram* (ERD)

3.3.3 Implementasi Sistem

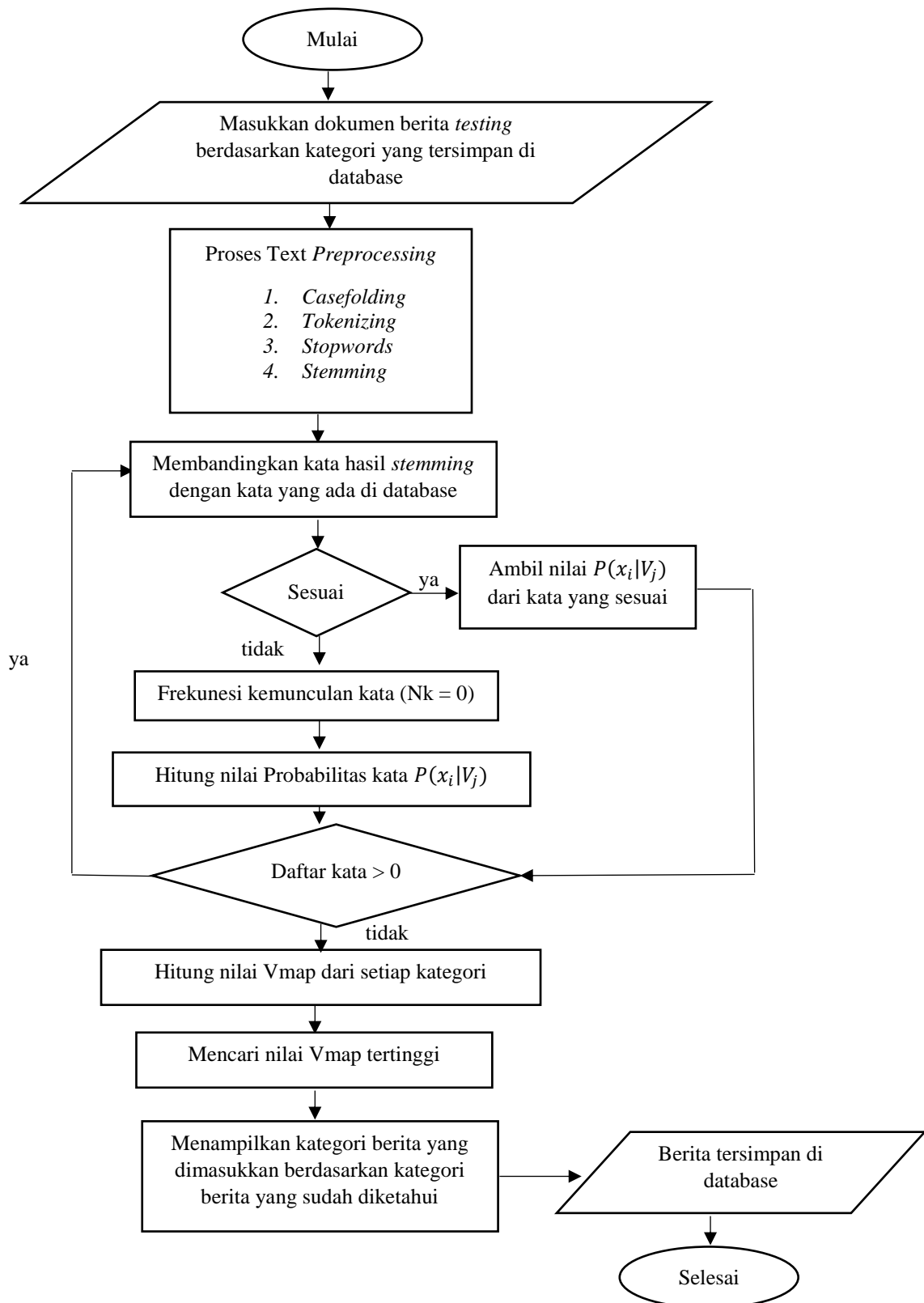
Implementasi sistem dikerjakan dalam bentuk kode program yang dilakukan menggunakan *tool sublime text 3* sebagai editor dengan bahasa pemrograman *PHP version 7.1.7* dan menggunakan *framework laravel*. Sedangkan untuk manajemen basis data yaitu *DBMS MySql version 5.0*. Tahapan awal pada pengimplementasi kode program adalah proses *text preprocessing* dan selanjutnya proses perhitungan klasifikasi menggunakan metode *Naïve Bayes Classifier*. Pada tahapan pengimplementasi kode program pada *text preprocessing* dilakukan dengan mengimplementasikan beberapa fungsi php antara lain:

1. *Implementasi casefolding* pada kode program sistem menggunakan php *strlower*.
2. *Implementasi tokenizing* pada kode program sistem menggunakan fungsi *explode* pada php.
3. Proses *stopword* yang digunakan berasal dari Sastrawi.
4. Proses *stemming* menggunakan algoritma *Confix-Stripping* (CS).

Untuk memperjelas alur menggunakan metode *Naïve Bayes Classifier* berikut *flowchart* untuk *training* dan *testing* yang ditunjukkan pada Gambar 3.1 dan 3.2 dengan menggunakan metode *Naive Bayes Classifier*.



Gambar 3. 1 Flowchart Training



Gambar 3. 2 Flowchart Testing

3.3.4 Pengujian Sistem

a. Pengumpulan Dataset

Untuk melakukan pengujian klasifikasi, diperlukan data sebagai inputan sistem agar dapat diproses. Data yang diperoleh merupakan *dataset* yang diambil dari beberapa portal berita online antara lain: *detik.com*, *kompas.com*, *sindo.com* dan *liputan6.com*. Jumlah data set yang digunakan dalam penelitian ini berjumlah 300 konten berita politik. Dimana 240 data *training* konten berita politik, dan sebanyak 60 konten berita politik sebagai data *testing*. Dari 240 data *training*, masing-masing kategori berita politik memiliki jumlah data yang sama yaitu 80 data. Dari model klasifikasi yang dihasilkan, kemudian di uji dengan menggunakan 60 data *testing*, dimana masing-masing kategori konten berita politik memiliki 20 data *testing*. Jumlah data *training* dan data *testing* yang dipakai merupakan hasil analisis berdasarkan pada penelitian sebelumnya. Pada penelitian sebelumnya menjelaskan, semakin banyak jumlah data *training* yang dipakai maka semakin besar peluang data *test* terklasifikasi dengan benar. Tabel 3.1 menjelaskan secara detail mengenai *dataset* yang digunakan dalam penelitian ini (Nurhadi, 2015).

Tabel 3. 1 *Dataset*

<i>Class Label</i>	<i>Data Latih (Training)</i>	<i>Data Uji (Testing)</i>
Pilkada	80	20
UU Ormas	80	20
Reshuffle Kabinet	80	20
Total	240	60

Sumber: Hasil Analisis (Hidayatullah and Ma'arif, 2016)

a. Pengujian Klasifikasi

Setelah melakukan tahapan klasifikasi konten berita politik dengan menggunakan metode *Naive Bayes Classifier*, peneliti melakukan pengujian klasifikasi berupa menghitung tingkat *precision*, *recall*, dan *fmeasure* agar mengetahui tingkat keakuratan dan ketepatan sistem. *Precision* merupakan keakuratan atau ketepatan hasil klasifikasi dari seluruh dokumen. *Recall* merupakan tingkat keberhasilan sistem. Semakin banyak data latih maka tingkat keakuratan dalam penentuan kategori akan semakin tinggi. Dalam berbagai penelitian juga

menjelaskan bahwa suatu sistem dinyatakan efektif apabila hasil penelusuran mampu menunjukkan *precision* yang tinggi meskipun *recall* yang didapatkan rendah. Maka dari itu *precision* menjadi salah satu ukuran yang digunakan untuk menilai keefektifan suatu sistem. (Lestari, 2016)

b. Pengujian Sistem

Setelah melakukan implementasi sistem ke dalam bentuk kode program, maka dilakukan pengujian sistem. Tujuan dari pengujian ini untuk bisa menghasilkan program yang lebih maksimal dan berjalan baik sesuai dengan harapan. Pengujian pada penelitian ini menggunakan *blackbox testing*. *Black box testing* membantu validasi fungsi keseluruhan sistem. *Black box testing* dilakukan berdasarkan *requirement*, sehingga *requirement* yang diinginkan pengguna dapat berjalan dengan baik. Jika *requirement* tersebut tidak sesuai dengan keinginan pengguna, maka *developer* dapat dengan mudah mengidentifikasinya dan mengatasi kesalahan tersebut (Nidhra, 2012).

BAB 4. PENGEMBANGAN SISTEM

Bab ini akan membahas tentang pengembangan sistem klasifikasi dokumen berita politik dengan mengimplementasikan metode *naïve bayes classifier*. Tahap pengembangan dilaksanakan berdasarkan tahapan penelitian.

4.1 Analisa Data

Tahap ini merupakan tahap dalam menentukan kebutuhan-kebutuhan apa saja yang dapat dilakukan oleh sistem. Kebutuhan sistem dibagi menjadi dua yaitu kebutuhan fungsional dan kebutuhan non-fungsional. Sistem ini digunakan oleh 3 pengguna yaitu admin, editor dan *guest*.

4.1.1 Kebutuhan Fungsional

Kebutuhan fungsional mengacu pada gambaran umum sistem yang telah ditentukan, maka kebutuhan fungsional sistem adalah sebagai berikut.

1. Sistem menggunakan fitur *login* untuk mengautentifikasi hak akses pengguna sistem dan *logout*.
2. Sistem dapat mengelola data *training* meliputi *view*, *edit*, *insert*, dan *delete*.
3. Sistem dapat melihat frekuensi, daftar kata, dan nilai probability dari data *training*.
4. Sistem dapat mengelola data *testing* meliputi *view*, *edit*, *insert*, dan *delete*.
5. Sistem dapat menampilkan hasil klasifikasi kategori berita politik yang terdiri dari kategori pilkada, reshuffle kabinet, dan uu ormas dengan menerapkan metode *Naïve Bayes Classifier*.
6. Sistem dapat melakukan hasil pengecekan kategori dokumen berita dan mengklasifikasikan kategori sesuai dengan artikel yang diinputkan secara otomatis.
7. Sistem dapat mengelola data *stopword* meliputi *view*, *edit*, *insert*, dan *delete*.
8. Sistem dapat melihat data berita politik.

4.1.2 Kebutuhan Non Fungsional

Kebutuhan non-fungsional merupakan hal yang dibutuhkan oleh sistem untuk mendukung aktivitas sistem sesuai dengan kebutuhan fungsional yang telah disusun. Kebutuhan non-fungsional dari sistem ini yaitu:

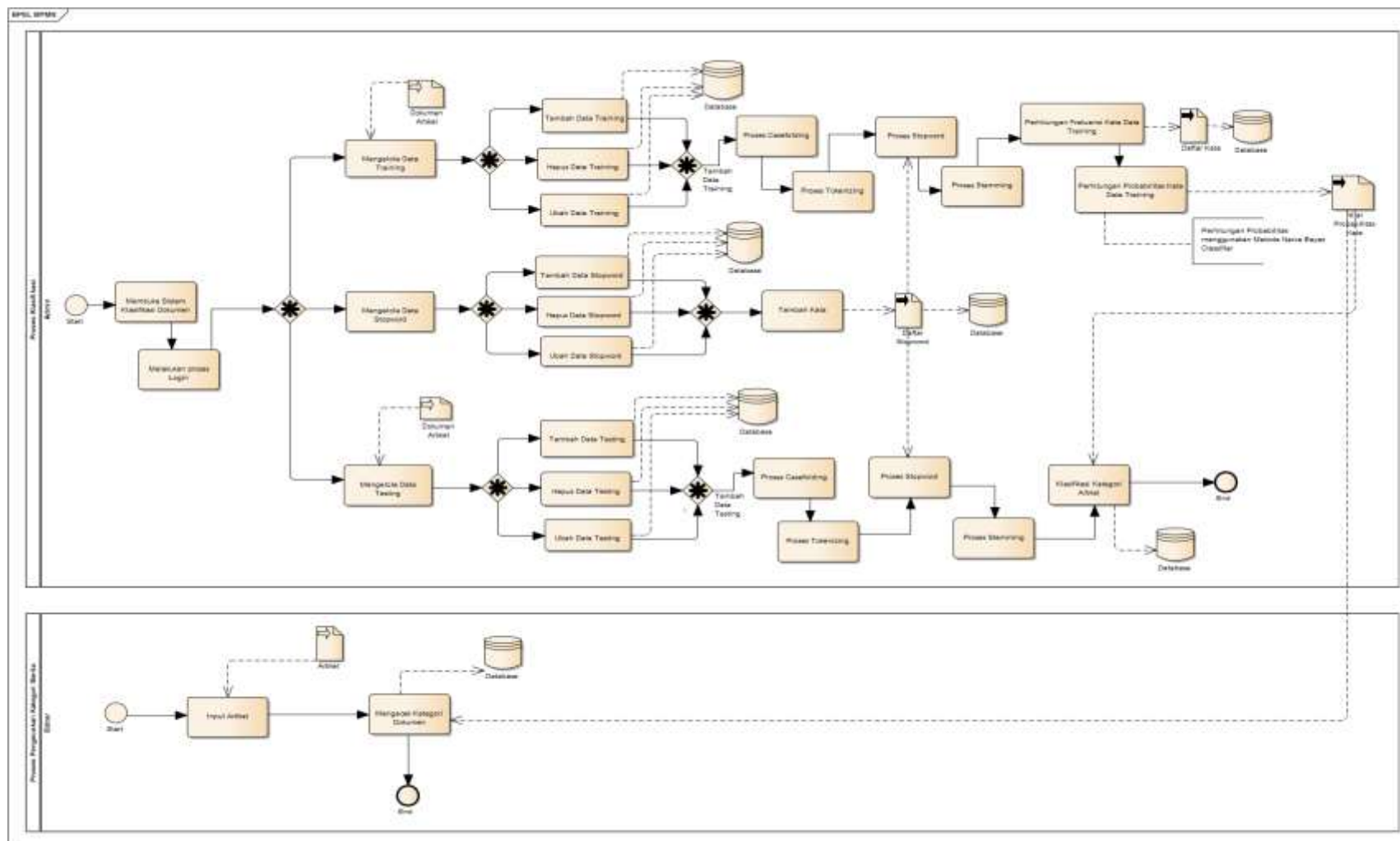
1. Sistem menggunakan *username* dan *password* untuk *autentifikasi* akses terhadap sistem.
2. Sistem dapat melakukan perhitungan menggunakan *Naïve Bayes Classifier*.
3. Sistem dibangun berbasis website agar dapat diakses dimanapun.

4.2 Desain Sistem

Tahapan yang dilakukan setelah melakukan analisis data yaitu tahap perencanaan pembangunan sistem yang dapat digambarkan dengan desain sistem. Desain sistem yang dibuat meliputi *business process modelling notation*, *usecase diagram*, *usecase scenario*, *sequence diagram*, *activity diagram*, *class diagram*, dan *entity relationship diagram* (ERD).

4.3.1 Bussiness Process Modelling Notation (BPMN)

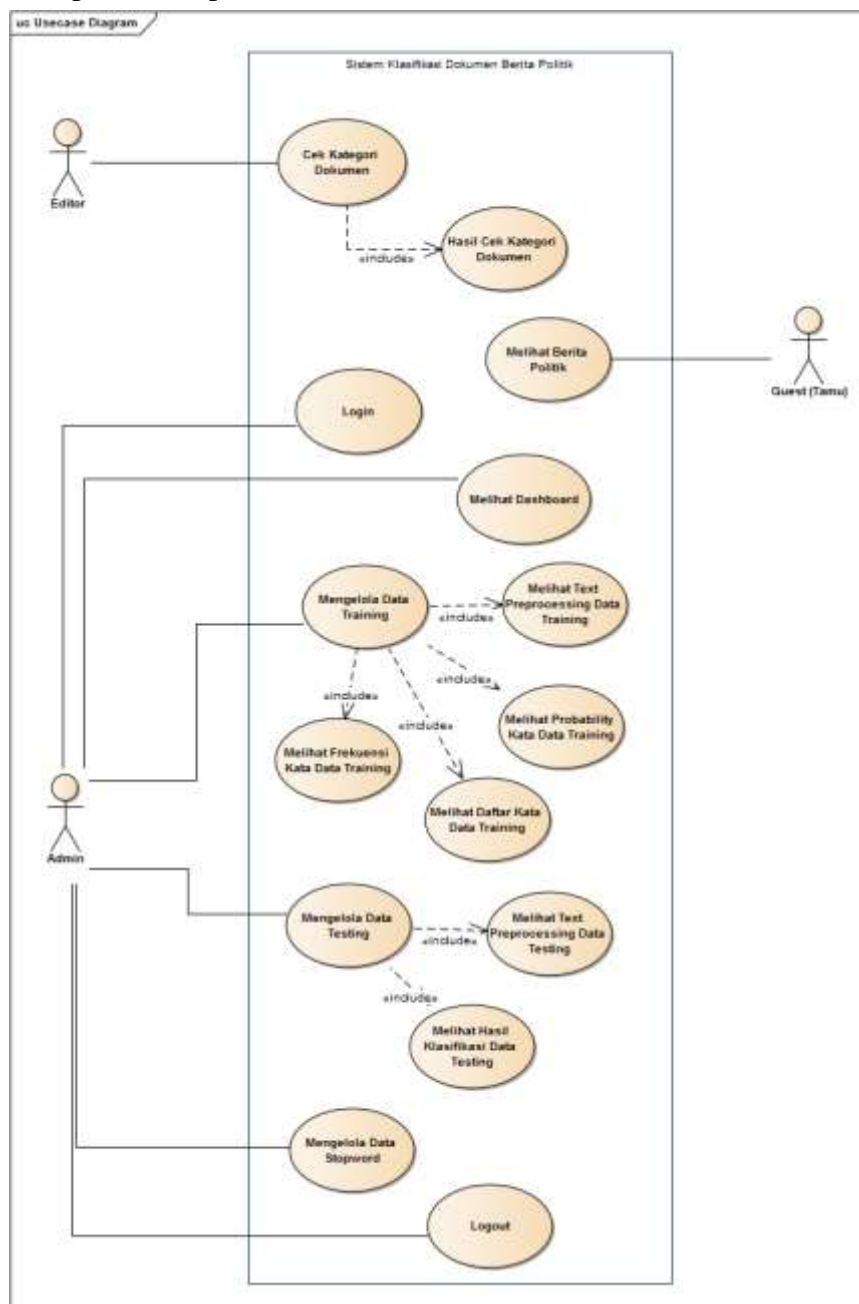
Berdasarkan dari analisa data didapatkan gambaran *bussines process* yang dapat ditunjukkan pada Gambar 4.1. Bisnis proses dimulai dari Admin mempunyai hak untuk mengelola data *training*, data *testing*, dan data *stopword*. Selanjutnya *Guest* (Tamu) dapat melakukan pengecekan kategori dokumen dengan menginputkan artikel berita. Data berita politik, data *training*, dan data *testing* digunakan sebagai inputan sistem untuk dapat menentukan hasil kelas klasifikasi kategori dan hasil pengecekan dokumen berita politik. Selain itu, data *stopword* digunakan untuk sebagai acuan data pada proses *text preprocessing stopwords*. Output yang diharapkan adalah hasil klasifikasi kategori berita. Tujuan dari sistem ini adalah dapat mengklasifikasikan dokumen berita politik sesuai dengan kategori yang lebih spesifik.



Gambar 4. 1 *Bussines Process Modelling Notation*

4.3.2 Usecase Diagram

Usecase Diagram merupakan gambaran fungsional dari sistem yang dapat menunjukkan fitur-fitur yang disediakan oleh sistem serta aktor yang dapat mengakses fitur tersebut. Pembuatan *usecase diagram* merujuk pada kebutuhan fungsional, kebutuhan nonfungsional dan bisnis prosesnya. *Usecase Diagram* pada sistem ini dapat dilihat pada Gambar 4.2.



Gambar 4. 2 *Usecase Diagram*

Penjelasan tentang definisi aktor dan definisi *usecase* dapat dilihat pada Tabel 4.1 dan Tabel 4.2.

1. Definisi Aktor

Aktor merupakan pengguna yang mengakses sistem klasifikasi dokumen berita politik. Terdapat 2 aktor yang dijelaskan pada Tabel 4.1.

Tabel 4. 1 Definisi Aktor

No	Aktor	Deskripsi
1.	Admin	Aktor admin memiliki hak akses secara penuh. Admin dapat melakukan <i>login</i> dan mengelola data sistem secara keseluruhan meliputi mengelola data training, mengelola data testing, dan data <i>stopword</i> .
2.	Editor	Aktor guest pada sistem ini memiliki peran untuk dapat cek kategori dokumen berita politik yang diinputkan.
3.	Tamu (<i>Guest</i>)	Aktor Tamu dapat melihat data berita politik dan membaca berita tersebut.

2. Definisi *Usecase*

Usecase yaitu penjelasan tentang fitur-fitur yang tersedia dalam sistem klasifikasi dokumen berita politik. Terdapat 15 *usecase* yang dijelaskan pada Tabel 4.2.

Tabel 4. 2 Definisi *Usecase*

No	<i>Usecase</i>	Deskripsi
1	Cek Kategori Dokumen	Menggambarkan proses untuk cek kategori dokumen.
2.	Hasil Cek Kategori Dokumen	Menggambarkan proses untuk melihat hasil cek kategori dokumen dengan menginputkan artikel.
3.	<i>Login</i>	Menggambarkan proses <i>autentifikasi</i> admin yang digunakan untuk masuk ke sistem.
4.	Melihat Dashboard	Menggambarkan proses melihat halaman dashboard yang menampilkan chart jumlah data training, data testing, <i>stopword</i> , visitor dan chart jumlah cek kategori dokumen.
5.	Mengelola Data <i>Training</i>	Menggambarkan proses pengelolaan data <i>training</i> yang meliputi menambahkan, melihat, mengedit, menghapus data <i>training</i> .
6.	Melihat <i>Text Preprocessing</i> Data <i>Training</i>	Menggambarkan proses untuk melihat <i>text preprocessing</i> yang terdiri dari proses <i>casefolding</i> ,

No	Usecase	Deskripsi
		<i>tokenizing, stopword, dan stemming</i> pada data <i>training</i> .
7.	Melihat Frekuensi Kata Data <i>Training</i>	Menggambarkan proses untuk melihat frekuensi kata pada setiap data <i>training</i> .
8.	Melihat Daftar Kata Data <i>Training</i>	Menggambarkan proses melihat daftar kata dari semua kategori data <i>training</i> .
9.	Melihat <i>Probability</i> Kata Data <i>Training</i>	Menggambarkan proses melihat nilai <i>probability</i> kata data <i>training</i> dari hasil perhitungan menggunakan metode Naïve Bayes.
10.	Mengelola Data <i>Testing</i>	Menggambarkan proses pengelolaan data <i>testing</i> yang meliputi menambahkan, melihat, mengedit, menghapus data <i>testing</i> .
11.	Melihat <i>Text Preprocessing</i> Data <i>Testing</i>	Menggambarkan proses untuk melihat <i>text preprocessing</i> yang terdiri dari proses <i>casefolding, tokenizing, stopword, dan stemming</i> pada data <i>testing</i> .
12.	Melihat Hasil Klasifikasi Data <i>Testing</i>	Menggambarkan proses untuk melihat hasil klasifikasi dari data <i>testing</i> yang diinputkan.
13.	Mengelola Data <i>Stopword</i>	Menggambarkan proses pengelolaan data <i>stopword</i> meliputi melihat, menambahkan, mengedit, dan menghapus <i>stopword</i> .
14.	<i>Logout</i>	Menggambarkan proses keluar sistem.
15.	Melihat data berita politik	Menggambarkan proses melihat data berita politik.

4.3.3 Skenario *Usecase*

Skenario merupakan penjabaran alur kerja sistem yang terdapat pada *usecase diagram* seperti Gambar 4.2.

1. Skenario Cek Kategori Dokumen

Skenario Cek Kategori Dokumen merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor mengakses menu Cek Kategori Dokumen. Aktor harus membuka sistem untuk dapat mengakses fitur ini. Skenario Cek Kategori Dokumen dapat dilihat pada lampiran A.

2. Skenario Hasil Cek Kategori Dokumen

Skenario Hasil Cek Kategori Dokumen merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor mengakses menu ini. Aktor harus mengisi artikel di *textbox* yang disediakan pada Cek Kategori Dokumen untuk mengetahui hasil kategori dokumennya. Dari pengecekan ini mendapatkan hasil

kategori dokumen berdasarkan artikel yang masuk. Skenario Hasil Cek Kategori Dokumen dapat dilihat pada lampiran A.

3. Skenario *Login*

Skenario *login* merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor masuk ke dalam sistem. Hanya aktor yang memiliki *email* dan *password* yang dapat mengakses sistem. Pada sistem hanya Admin yang memiliki hak akses untuk login ke sistem. Skenario *Login* dapat dilihat pada lampiran A.

4. Skenario Melihat *Dashboard*

Skenario Melihat *Dashboard* merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor masuk ke dalam sistem. Tampilan menu awal setelah melakukan login adalah *Dashboard*. Pada sistem hanya Admin yang dapat melihat *Dashboard*. Pada menu *Dashboard* menampilkan beberapa *chart* antara lain; chart jumlah data *training* yang telah diinputkan berdasarkan kategori, chart jumlah data *testing*, chart jumlah cek kategori dokumen, *chart* jumlah *stopword*, dan *chart* jumlah *visitor* yang sudah mengakses sistem. Skenario Melihat *Dashboard* dapat dilihat pada Lampiran A.

5. Skenario Mengelola Data *Training*

Skenario Mengelola Data *Training* merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor mengelola data *training*. Aktor dapat menambah, melihat, mengedit, dan menghapus data *training*. Pada sistem hanya Admin yang dapat mengakses menu Data *Training*. Penjelasan dan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternative Mengelola Data *Training* dapat dilihat pada Tabel 4.3.

Tabel 4. 3 Mengelola Data *Training*

No. Usecase	UCS 05
Nama Usecase	Mengelola Data <i>Training</i>
Aktor	Admin
PreKondisi	Admin memilih menu Data <i>Training</i>

PraKondisi	Admin berhasil menambahkan, melihat, mengupdate, dan menghapus data <i>training</i>
Flow Events	
Skenario Normal : Tambah data <i>training</i>	
Aksi Aktor	Aksi Sistem
1. Klik menu Data <i>Training</i>	2. Menampilkan halaman menu Data <i>Training</i> yang berisi : <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action button Frekuensi</i> • <i>Action Button Daftar Kata</i> • <i>Action button Probability</i> • <i>Action button Tambah Dokumen Data Training</i>
3. Klik Tambah Dokumen Data Training	4. Menampilkan form Tambah Data <i>Training</i> yang berisi <i>text field</i> : <ul style="list-style-type: none"> • Judul • Artikel • Kategori Berita • <i>Action button Save</i> • <i>Action button Close</i>
5. Mengisi form Tambah Data <i>Training</i>	
6. Klik <i>button Save</i>	7. Menampilkan halaman Data <i>Training</i>
Skenario Alternative : Data tidak lengkap	
Aksi Aktor	Aksi Sistem
6. Klik <i>button Save</i>	7. Menampilkan <i>alert "Please fill out this field"</i>
	7. Menampilkan form Tambah Data <i>Training</i>
Skenario Alternatif : Batal tambah data	
Aksi Aktor	Aksi Sistem
5. Klik <i>button Close</i>	6. Menampilkan halaman data <i>training</i>
Skenario Normal : Lihat data <i>training</i>	
Aksi Aktor	Aksi Sistem

1. Klik menu <i>Data Training</i>	<p>2. Menampilkan halaman menu <i>Data Training</i> yang berisi tabel:</p> <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action button Frekuensi</i> • <i>Action Button Daftar Kata</i> • <i>Action button Probability</i> • <i>Action button Tambah Dokumen Data Training</i>
-----------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Skenario Normal : *Update data training*

Aksi Aktor	Aksi Sistem
1. Klik menu <i>Data Training</i>	<p>2. Menampilkan halaman menu <i>Data Training</i> yang berisi tabel:</p> <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action button Frekuensi</i> • <i>Action Button Daftar Kata</i> • <i>Action button Probability</i> • <i>Action button Tambah Dokumen Data Training</i>
2. Klik <i>button Update</i>	<p>3. Menampilkan form <i>Update Data Training</i> yang dipilih:</p> <ul style="list-style-type: none"> • Judul • Artikel • Kategori Berita • <i>Action button Save</i> • <i>Action button Close</i>

4. Mengubah data <i>training</i>	
5. Klik <i>button Save</i>	6. Menampilkan halaman data <i>training</i>
Skenario Alternatif : Data tidak lengkap	
Aksi Aktor	Aksi Sistem
5. Klik <i>button Save</i>	6. Menampilkan <i>alert</i> “ <i>Please fill out this field</i> ”
	7. Menampilkan form <i>update</i> data <i>training</i>
Skenario Alternatif : Batal update data	
Aksi Aktor	Aksi Sistem
5. Klik <i>button Close</i>	6. Menampilkan halaman data <i>training</i>
Skenario Normal : Hapus data <i>training</i>	
Aksi Aktor	Aksi Sistem
1. Klik menu <i>Data Training</i>	2. Menampilkan halaman menu <i>Data Training</i> yang berisi : <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action button Frekuensi</i> • <i>Action Button Daftar Kata</i> • <i>Action button Probability</i> • <i>Action button Tambah Dokumen Data Training</i>
3. Klik <i>button Delete</i> pada data yang diinginkan	4. Menampilkan halaman data <i>training</i>

6. Skenario Melihat *Text Preprocessing* Data *Training*

Skenario Melihat *Text Preprocessing* Data *Training* merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor melihat *text preprocessing* data *training*. Aktor dapat melihat proses *text preprocessing* dari data *training* yang dipilih. Proses *text preprocessing* meliputi *casefolding*, *tokenizing*, *stopword*, dan *stemming* yang setiap prosesnya dapat dilihat. Hanya Admin yang dapat mengakses

fitur ini. Penjelasan dan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternative Melihat *Text Preprocessing Data Training* dapat dilihat pada Lampiran A.

7. Skenario Melihat Frekuensi Kata Data *Training*

Skenario Melihat Frekuensi Kata Data *Training* merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor mengakses fitur ini. Aktor dapat melihat frekuensi kata dari data *training* yang telah diinputkan. Hanya Admin yang dapat mengakses fitur ini. Penjelasan Skenario Melihat Frekuensi Kata Data *Training* dapat dilihat pada Lampiran A.

8. Skenario Melihat Daftar Kata Data *Training*

Skenario Melihat Daftar Kata Data *Training* merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor mengakses fitur ini. Hanya Admin yang dapat mengaksesnya. Admin dapat melihat daftar kata dari semua kategori pada data *training* yang telah diinputkan. Penjelasan Skenario Melihat Daftar Kata Data *Training* dapat dilihat pada Lampiran A.

9. Skenario Melihat *Probability* Kata Data *Training*

Skenario Melihat *Probability* Kata Data *Training* merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor mengakses fitur ini. Aktor dapat melihat nilai *probability* dari setiap kata yang diinputkan pada data *training*. Hanya Admin yang dapat mengakses fitur ini. Penjelasan Skenario Melihat *Probability* Kata Data *Training* dapat dilihat pada Lampiran A.

10. Skenario Mengelola Data *Testing*

Skenario Mengelola Data *Testing* merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor mengelola data *testing*. Aktor dapat menambah, melihat, mengedit, dan menghapus data *testing*. Hanya Admin yang dapat mengakses fitur ini. Penjelasan dan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternative Mengelola data *testing* dapat dilihat pada Lampiran A.

11. Skenario Melihat *Text Preprocessing* Data *Testing*

Skenario Melihat *Text Preprocessing* Data *Testing* merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor melihat *text preprocessing* data *testing*. Aktor dapat melihat proses *text preprocessing* yang dipilih. Proses *text preprocessing* terdiri dari *casefolding*, *tokenizing*, *stopword* dan *stemming*. Hanya Admin yang dapat mengakses fitur ini. Penjelasan dan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternative Melihat *Text Preprocessing* Data *Testing* dapat dilihat pada Lampiran A.

12. Skenario Melihat Hasil Klasifikasi Data *Testing*

Skenario Melihat Hasil Klasifikasi Data *Testing* merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor mengakses fitur ini. Aktor dapat melihat hasil klasifikasi dari data *testing* yang diinputkan oleh Admin. Penjelasan Skenario Melihat Hasil Klasifikasi Data *Testing* dapat dilihat pada Lampiran A.

13. Skenario Mengelola Data *Stopword*

Skenario Mengelola Data *Stopword* merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor mengelola data *Stopword* yang meliputi *edit*, *view*, *delete*, dan *insert*. Aktor dapat melihat mengelola Data *Stopword* yang dipilih. Hanya Admin yang dapat mengakses fitur ini. Penjelasan dan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternative Mengelola Data *Stopword* dapat dilihat pada Lampiran A.

14. Skenario *Logout*

Skenario *logout* merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor melakukan *logout* atau keluar dari sistem. Aktor dapat keluar dari sistem dengan memilih tombol keluar. Skenario *logout* dapat dilihat pada Lampiran A.

15. Skenario Melihat Berita Politik

Skenario Melihat Berita Politik merupakan alur yang menjelaskan aksi aktor dan reaksi sistem pada saat aktor mengakses fitur ini. Aktor dapat melihat berita politik. Penjelasan Skenario Melihat Berita Politik dapat dilihat pada Lampiran A.

4.3.4 *Activity Diagram*

Setelah membuat Usecase Skenario tahap selanjutnya membuat *Activity Diagram*. *Activity Diagram* merupakan diagram yang menggambarkan alur aktivitas proses-proses yang terjadi pada sistem. *Activity Diagram* dalam sistem ini antara lain sebagai berikut:

1. *Activity Diagram* Cek Kategori Dokumen

Activity Diagram Cek Kategori Dokumen menggambarkan tentang proses yang terjadi saat aktor mengakses fitur ini. Aktor dapat mengecek kategori dokumen dengan cara membuka sistem yang kemudian langsung menampilkan halaman awal sistem yaitu Cek Kategori Dokumen. *Activity Diagram* Cek Kategori Dokumen dapat dilihat pada Lampiran B.

2. *Activity Diagram* Hasil Cek Kategori Dokumen

Activity Diagram Hasil Cek Kategori Dokumen menggambarkan tentang proses yang terjadi saat aktor mengakses fitur ini. Aktor dapat melihat hasil cek kategori dokumen dengan cara membuka sistem yang kemudian langsung menampilkan halaman awal sistem yaitu Cek Kategori Dokumen. Aktor kemudian mengisi artikel pada textbox Cek Kategori Dokumen yang disediakan dan kemudian sistem akan memproses untuk menampilkan hasil cek kategori dokumen. *Activity Diagram* Hasil Cek Kategori Dokumen dapat dilihat pada Lampiran B.

3. *Activity Diagram* Login

Activity Diagram login menggambarkan tentang proses yang terjadi saat aktor masuk ke dalam sistem. Aktor memasukkan *email* dan *password* dan mengklik tombol “*login*”. Kemudian sistem akan melakukan pengecekan apakah

data yang dimasukkan kosong atau salah, jika salah atau kosong maka sistem akan menampilkan halaman form *login*. Namun jika benar, maka sistem akan menampilkan halaman dashboard. *Activity Diagram login* dapat dilihat pada Lampiran B.

4. *Activity Diagram Melihat Dashboard*

Activity Diagram Melihat Dashboard menggambarkan tentang proses yang terjadi saat aktor masuk ke dalam sistem. Aktor yang dapat melihat dashboard hanya Admin. Admin harus masuk kedalam sistem dengan login kemudian akan masuk ke halaman awal yaitu halaman *dashboard*. Pada halaman *dashboard* Admin dapat melihat chart jumlah data training yang telah diinputkan setiap kategori, chart jumlah data testing, chart jumlah cek kategori dokumen, chart jumlah data *stopword*, dan chart jumlah *visitor* yang telah mengakses sistem. *Activity Diagram Melihat Dashboard* dapat dilihat pada Lampiran B.

5. *Activity Diagram Mengelola Data Training*

Activity diagram Mengelola Data Training menggambarkan tentang proses yang terjadi saat aktor mengelola data *training*. Aktor yang dapat mengelola data *training* adalah Admin. Admin dapat menambah, mengubah, dan menghapus data *training*. *Activity diagram* mengelola data training dapat dilihat pada Gambar 4.3.

6. *Activity Diagram Melihat Text Preprocessing Data Training*

Activity Diagram Melihat Text Preprocessing Data Training menggambarkan tentang proses yang terjadi saat aktor melihat *text preprocessing* data *training* Aktor yang mengakses fitur ini hanya Admin. Admin dapat melihat proses text preprocessing yang terdiri dari *casefolding*, *tokenizing*, *stopword*, dan *stemming*. *Activity Diagram* melihat *text preprocessing* data *training* dapat dilihat pada Lampiran B.

7. *Activity Diagram Melihat Frekuensi Kata Data Training*

Activity diagram Melihat Frekuensi Kata Data Training menggambarkan tentang proses yang terjadi saat aktor mengakses fitur ini. Aktor yang dapat melihat

frekuensi kata data *training* adalah Admin. Admin memilih menu Data Training dan kemudian memilih *button* menu Frekuensi. *Activity diagram* Melihat Frekuensi Kata Data *Training* dapat dilihat pada Lampiran B.

8. *Acivity Diagram* Melihat Daftar Kata Data *Training*

Activity diagram Melihat Daftar Kata Data *Training* menggambarkan tentang proses yang terjadi saat aktor mengakses fitur ini. Aktor yang dapat melihat daftar kata data *training* adalah Admin. Admin memilih menu Data Training dan kemudian memilih *button* menu Daftar Kata. *Activity diagram* Melihat Daftar Kata Data *Training* dapat dilihat pada Lampiran B.

9. *Acivity Diagram* Melihat *Probability* Kata Data *Training*

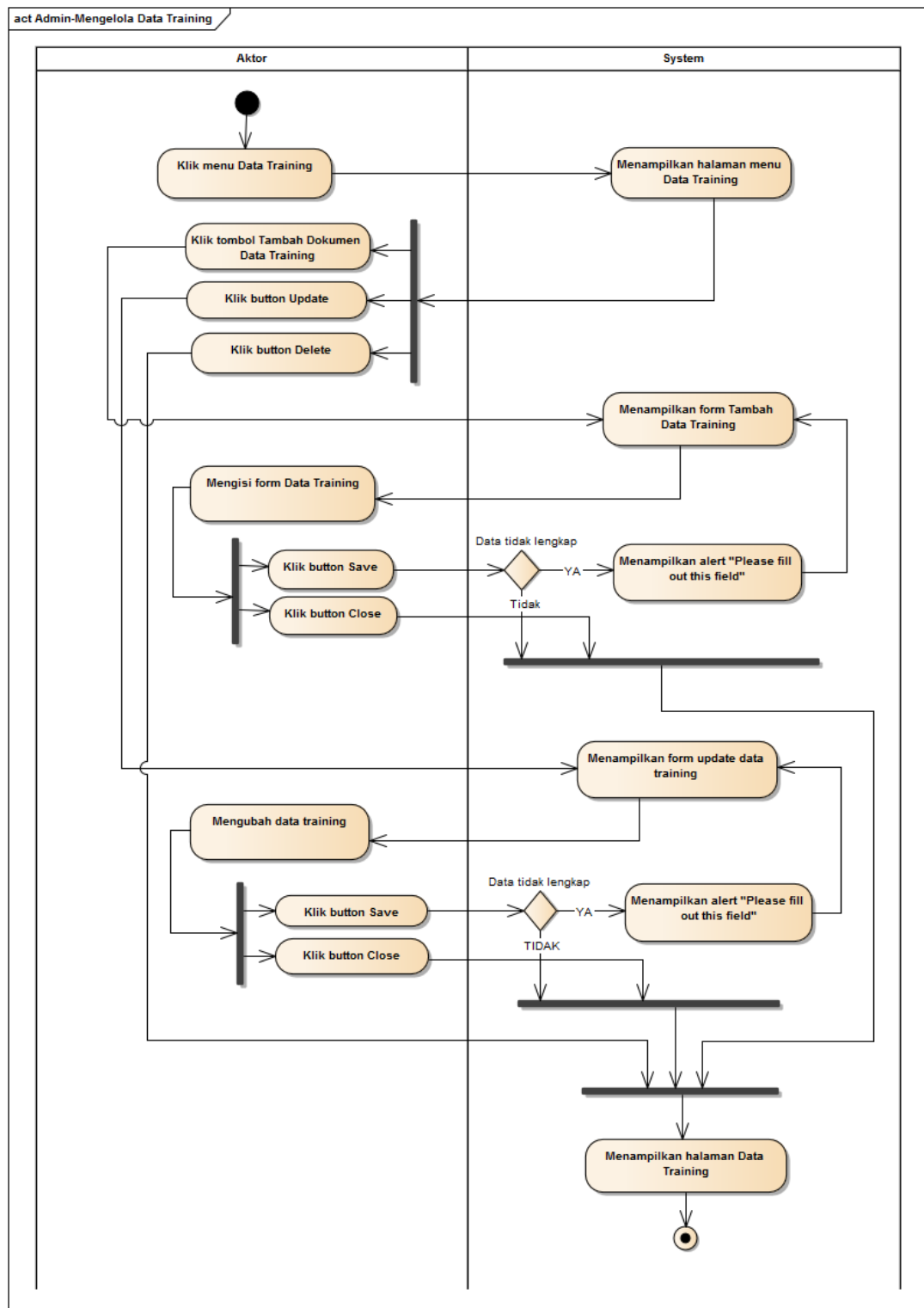
Activity diagram Melihat *Probability* Kata Data *Training* menggambarkan tentang proses yang terjadi saat aktor mengakses fitur ini. Aktor yang dapat melihat *probability* kata data *training* adalah Admin. Admin memilih menu Data Training dan kemudian memilih *button* menu *Probability*. *Activity diagram* Melihat *Probability* Kata Data *Training* dapat dilihat pada Lampiran B.

10. *Acivity Diagram* Mengelola Data *Testing*

Activity diagram Mengelola Data *Testing* menggambarkan tentang proses yang terjadi saat aktor mengelola data *testing*. Aktor yang dapat mengelola data *testing* adalah Admin. Admin dapat menambah, mengubah, dan menghapus data *testing*. *Activity Diagram* mengelola data *testing* dapat dilihat pada Lampiran B.

11. *Acivity Diagram* Melihat *Text Preprocessing* Data *Testing*

Activity Diagram Melihat *Text Preprocessing* Data *Testing* menggambarkan tentang proses yang terjadi saat aktor melihat *text preprocessing* data *testing* Aktor yang mengakses fitur ini hanya Admin. Admin dapat melihat proses *text preprocessing* yang terdiri dari *casefolding*, *tokenizing*, *stopword*, dan *stemming*. *Activity Diagram* melihat *text preprocessing* data *testing* dapat dilihat pada Lampiran B.



Gambar 4. 3 Activity Diagram Mengelola Data Training

12. *Activity Diagram* Melihat Hasil Klasifikasi Data *Testing*

Activity diagram Melihat Hasil Klasifikasi Data *Testing* menggambarkan tentang proses yang terjadi saat aktor mengakses fitur ini. Aktor yang dapat melihat hasil klasifikasi data *testing* adalah Admin. Admin memilih menu Data Testing dan kemudian memilih *button* menu Hasil. *Activity diagram* Melihat Hasil Klasifikasi Data *Testing* dapat dilihat pada Lampiran B.

13. *Activity Diagram* Mengelola Data *Stopword*

Activity diagram Mengelola Data *Stopword* menggambarkan tentang proses yang terjadi saat aktor mengelola data *stopword*. Aktor yang dapat mengelola data *stopword* adalah Admin. Admin dapat menambah, mengubah, dan menghapus data *stopword*. *Activity Diagram* mengelola data *stopword* dapat dilihat pada Lampiran B.

14. *Activity Diagram* Logout

Activity Diagram *logout* menggambarkan tentang proses yang terjadi saat aktor keluar ke dalam sistem. *Activity Diagram* *logout* dapat dilihat pada Lampiran B.

15. *Activity Diagram* Melihat Berita Politik

Activity diagram Melihat Berita Politik menggambarkan tentang proses yang terjadi saat aktor mengakses fitur ini. Aktor yang dapat melihat berita politik. *Activity diagram* Melihat Berita Politik dapat dilihat pada Lampiran B.

4.3.5 *Sequence Diagram*

Tahap selanjutnya pada penelitian ini ialah membuat *Sequence Diagram*. *Sequence Diagram* merupakan diagram yang menggambarkan interaksi antar kelas yang dilakukan sesuai dengan alurnya masing-masing. *Sequence Diagram* dalam sistem ini antara lain sebagai berikut:

1. *Sequence Diagram* Cek Kategori Dokumen

Sequence Diagram Cek Kategori Dokumen terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah kelas tampilan *index*, *controller HomeController*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran C.

2. *Sequence Diagram* Hasil Cek Kategori Dokumen

Sequence Diagram Hasil Cek Kategori Dokumen terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah *view hasil* dengan *controller HomeController dan ArtikelController* dan *model Artikel*. Interaksi antar kelas dan yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran C.

3. *Sequence Diagram* Login

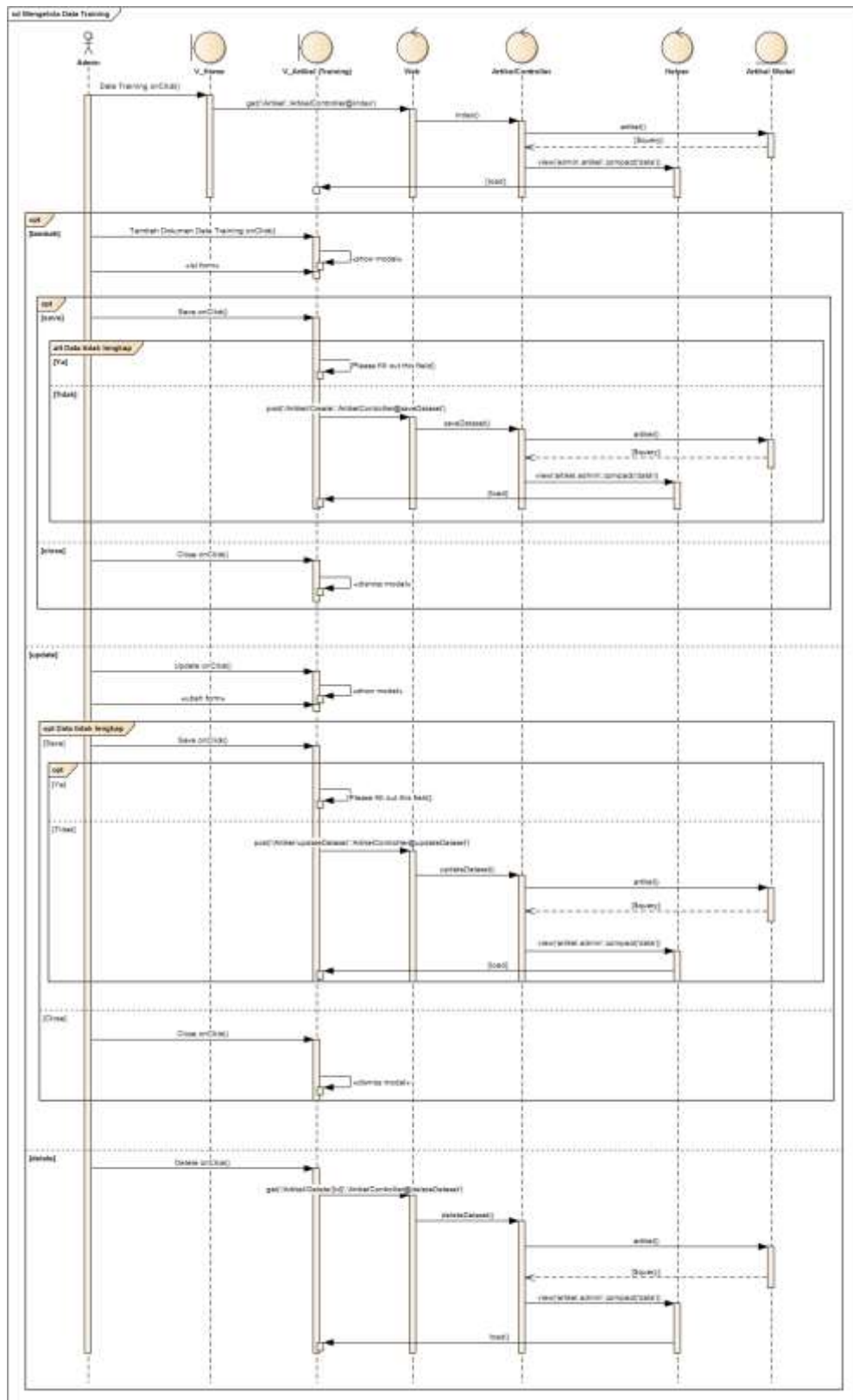
Sequence Diagram login terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah kelas tampilan *login* dengan *controller LoginController* dan *model User*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran E.

4. *Sequence Diagram* Melihat Dashboard

Sequence Diagram Melihat Dashboard terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah kelas tampilan *home* dengan *controller HomeController*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran C.

5. *Sequence Diagram* Mengelola Data Training

Sequence Diagram Mengelola Data Training terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah kelas tampilan *artikel* dengan *controller ArtikelController* dan *model Artikel*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Gambar 4.4.



Gambar 4. 4 Sequence Diagram Mengelola Data Training

6. *Sequence Diagram Melihat Text Preprocessing Data Training*

Sequence Diagram Melihat Text Preprocessing Data Training terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah kelas tampilan *artikel* dengan *casefolding*, *tokenizing*, *stopword*, *stemming*, dengan *controller ArtikelController* dan *model Artikel*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran C.

7. *Sequence Diagram Melihat Frekuensi Kata Data Training*

Sequence Diagram Melihat Frekuensi Kata Data Training terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah kelas tampilan *hasilFrekuensi* dengan *controller ArtikelController* dan *model Artikel*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran C.

8. *Sequence Diagram Melihat Daftar Kata Data Training*

Sequence Diagram Melihat Daftar Kata Data Training terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah kelas tampilan *kata* dengan *controller ArtikelController* dan *model MasterTraining*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran C.

9. *Sequence Diagram Melihat Probability Kata Data Training*

Sequence Diagram Melihat Probability Kata Data Training terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah kelas tampilan *hasilProbability* dengan *controller ArtikelController* dan *model Artikel*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran C.

10. *Sequence Diagram Mengelola Data Testing*

Sequence Diagram Mengelola Data Testing terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah kelas tampilan

testing dengan *controller ArtikelController* dan *model Artikel*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran C.

11. *Sequence Diagram Melihat Text Preprocessing Data Testing*

Sequence Diagram Melihat Text Preprocessing Data Testing terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah kelas tampilan *artikel*, *casefolding*, *tokenizing*, *stopword*, *stemming*, dengan *controller ArtikelController* dan *model Artikel*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran C.

12. *Sequence Diagram Melihat Hasil Klasifikasi Data Testing*

Sequence Diagram Melihat Hasil Klasifikasi Data Testing terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah kelas tampilan *hasilProbability2* dengan *controller ArtikelController* dan *model Artikel*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran C.

13. *Sequence Diagram Mengelola Data Stopword*

Sequence Diagram Mengelola Data Stopword terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah kelas tampilan *stopword*, *masterstopword* dengan *controller StopwordController* dan *model Stopword*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran C.

14. *Sequence Diagram Logout*

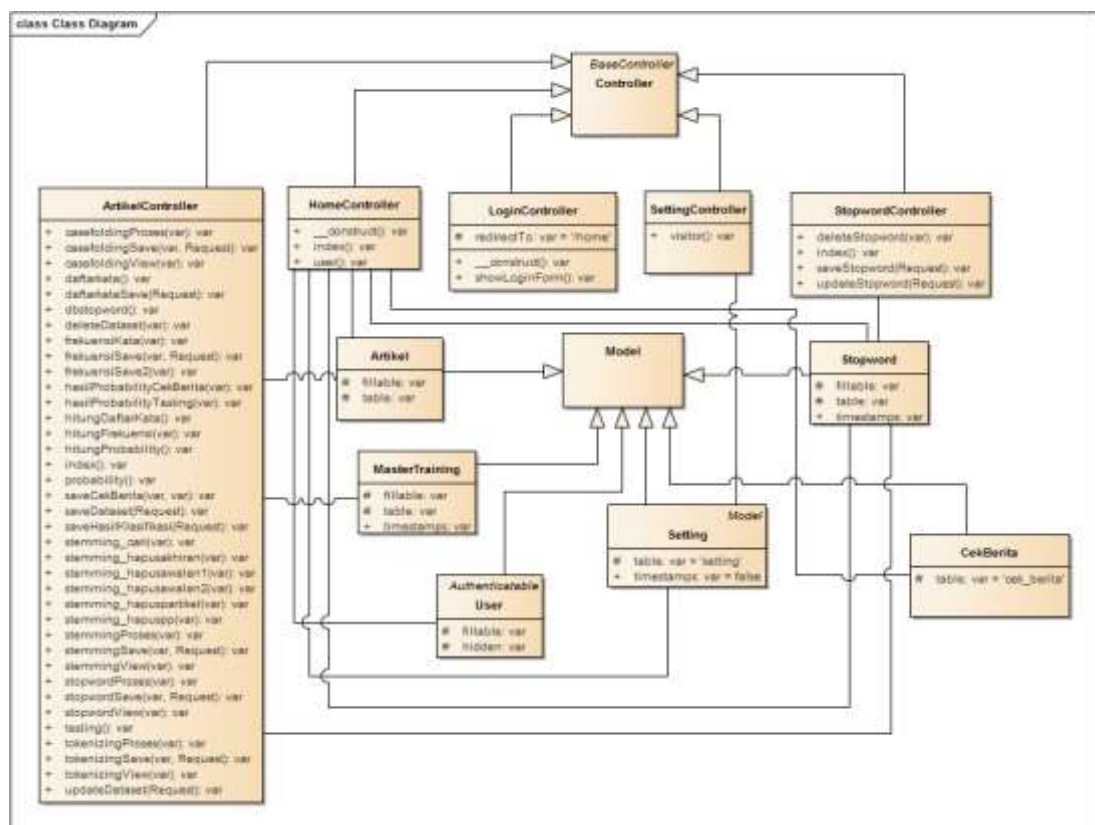
Sequence Diagram Logout terdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah view home dan *controller LoginController*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran C.

15. Sequence Diagram Melihat Berita Politik

Sequence Diagram Melihat Berita Politik erdapat beberapa interaksi antar kelas yang digunakan. Kelas yang saling terhubung adalah kelas tampilan *list*, *detail* dengan *controller HomeController* dan *model CekBerita*. Interaksi antar kelas yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada Lampiran C.

4.3.6 Class Diagram

Setelah membuat *sequence diagram*, tahap selanjutnya ialah membuat *Class Diagram*. *Class Diagram* menggambarkan hubungan antarkelas yang digunakan untuk membangun sistem. *Class Diagram* sistem klasifikasi dokumen berita politik dapat dilihat pada Gambar 4.5.



Gambar 4. 5 Class Diagram

Dalam *Class Diagram* ini terdapat 2 objek utama yaitu *model* dan *controller*. *Class model* terdiri dari *mastertrainingModel*, *artikelModel*,

stopwordModel, *UserModel*, *CekBerita Model* dan *settingModel* yang merupakan *subclass* dari *class model*. *Class model* tersebut menjadi *superclass* dari setiap *subclassnya*. Objek kedua yaitu *class controller* yang merupakan *superclass* dari *artikelController*, *settingController*, *homeController*, *loginController*, dan *stopwordController*. Adapun relasi yang digunakan dalam *class diagram* ini adalah asosiasi dan *generalization*.

Relasi *generalization* adalah relasi antar kelas dengan makna umum ke khusus atau menyatakan suatu hubungan *inheritance* (pewarisan). Contoh dari relasi *generalization* adalah kelas *Controller* yang merupakan *parent* dengan kelas *artikelController*, *homeController*, *loginController*, *settingController*, dan *stopwordController* yang merupakan *child*. Relasi asosiasi adalah relasi antar kelas dengan makna umum. Contoh dari relasi asosiasi adalah kelas *artikelController* dengan *artikelModel* dan *settingController* dan *settingModel*. Berikut fungsi dari setiap kelas yang ada pada *controller* dan *model* yang ditunjukkan pada Tabel 4.4-
Tabel 4.14.

Tabel 4. 4 Method Class Controller ArtikelContrller

No	Nama Method	Kegunaan Method
1	index	Menampilkan halaman data training
2	saveDataset	Menyimpan dataset meliputi data training dan data testing
3	saveHasilKlasifikasi	Menyimpan hasil klasifikasi sistem
4	saveCekBerita	Menyimpan input cek berita
5	updateDataset	Mengubah dataset meliputi data training dan data testing
6	deleteDataset	Menghapus data training dan data testing
7	casefoldingProses	Memproses tahap casefolding, yaitu mengubah semua huruf menjadi <i>lowercase</i> dan menghapus tanda baca maupun angka
8	casefoldingView	Menampilkan kata hasil proses casefolding
9	casefoldingSave	Menyimpan kata hasil casefolding pada database
10	tokenizingProses	Memproses tahap tokenizing, yaitu memotong kalimat artikel menjadi potongan kata.
11	tokenizingView	Menampilkan kata hasil proses tokenizing
12	tokenizingSave	Menyimpan hasil proses tokenizing pada database
13	dbstopword	Menampilkan data stopword dari database library stopword

No	Nama Method	Kegunaan Method
14	stopwordProses	Memproses tahap stopwords, yaitu menghapus kata-kata yang terlalu umum berdasarkan data stopwords yang ada
15	stopwordView	Menampilkan kata hasil stopwords
16	stopwordSave	Menyimpan hasil stopwords pada database
17	stemming_cari	Mencari kata dasar yang ada pada database pada proses stemming
18	stemming_hapuspartikel	Penghapusan partikel seperti 'lah', 'kah', 'pun' pada proses stemming
19	stemming_hapuspp	Penghapusan possessive pronoun seperti 'ku', 'mu', 'nya' pada proses stemming
20	stemming_hapusawalan1	Penghapusan kata awalan pada proses stemming
21	stemming_hapusawalan2	Penghapusan kata awalan pada proses stemming
22	stemming_hapusakhiran	Penghapusan kata akhiran pada proses stemming
23	stemmingProses	Memproses tahap stemming, yaitu mengembalikan kata ke bentuk kata dasar
24	stemmingView	Menampilkan kata hasil stemming
25	stemmingSave	Menyimpan kata hasil stemming pada database
26	hitungFrekuensi	Perhitungan frekuensi kata
27	frekuensiKata	Menampilkan frekuensi kata
28	frekuensiSave	Menyimpan frekuensi kata pada database
29	hitungDaftarKata	Menghitung jumlah kata
30	daftarkata	Menampilkan daftar kata dari semua kategori
31	daftarkataSave	Menyimpan/Mengupdate hasil daftar kata
32	hitungProbability	Menghitung nilai probability
33	probability	Menampilkan nilai probability
34	testing	Menampilkan data testing
35	hasilProbabilityTesting	Menampilkan hasil probability data testing
36	hasilProbabilityCekBerita	Menampilkan hasil Cek Berita

Tabel 4. 5 Method Class Controller HomeController

No	Nama Method	Kegunaan Method
1	<i>__construct</i>	Method yang pertama kali dijalankan
2	<i>index</i>	Menampilkan halaman tampilan data
3	<i>user</i>	Menampilkan hak akses user

Tabel 4. 6 Method Class Controller LoginController

No	Nama Method	Kegunaan Method
1	<i>construct</i>	Method yang pertama kali dijalankan
2	<i>showLoginForm</i>	Menampilkan view login

Tabel 4. 7 *Method Class Controller SettingController*

No	Nama Method	Kegunaan Method
1	<i>visitor</i>	Mengambil data pada tabel setting

Tabel 4. 8 *Method Class Controller StopwordController*

No	Nama Method	Kegunaan Method
1	<i>index</i>	Menampilkan halaman data stopword
2	<i>saveStopword</i>	Menyimpan data stopword
3	<i>updateStopword</i>	Mengubah data stopword
4	<i>deleteStopword</i>	Menghapus data stopword

Tabel 4. 9 *Class Model User*

No	Nama Method	Kegunaan Method
1	<i>fillable</i>	Untuk mendaftarkan atribut (nama kolom) yang bisa kita isi ketika melakukan insert atau update ke database. Berisi atribut nama-nama column yang ada di dalam tabel tersebut dan bersifat publik/akan diisi. Tabel tersebut berisi atribut kata.
2	<i>hidden</i>	Untuk menyembunyikan atribut model (kolom tabel) ketika kita memanggil fungsi

Tabel 4. 10 *Class Model Stopword*

No	Nama Method	Kegunaan Method
1	<i>table</i>	Pada Model stopword, properti \$table perlu didefinisikan karna penamaan tabel yang digunakan tidak sesuai dengan aturan baku di Eloquent. Pada Model Stopword property \$table didefinisikan dengan syntax yaitu stopword.
2	<i>fillable</i>	Untuk mendaftarkan atribut (nama kolom) yang bisa kita isi ketika melakukan insert atau update ke database. Berisi atribut nama-nama column yang ada di dalam tabel tersebut dan bersifat publik/akan diisi. Tabel tersebut berisi atribut kata.
3	<i>timestamps</i>	Mencatat kapan suatu record ditambahkan/diupdate ke database, namun karna tidak digunakan dalam sistem maka bernilai false

Tabel 4. 11 *Class Model MasterTraining*

No	Nama Method	Kegunaan Method
1	<i>table</i>	Pada Model MasterTraining, properti \$table perlu didefinisikan karna penamaan tabel yang digunakan tidak sesuai dengan aturan baku di Eloquent. Pada Model MasterTraining property \$table didefinisikan dengan syntax yaitu master_training.
2	<i>fillable</i>	Untuk mendaftarkan atribut (nama kolom) yang bisa kita isi ketika melakukan insert atau update ke database. Berisi atribut

No	Nama Method	Kegunaan Method
		nama-nama column yang ada di dalam tabel tersebut dan bersifat publik/akan diisi. Tabel tersebut berisi atribut id, kata, dan frekuensi..
3	<i>timestamps</i>	Mencatat kapan suatu record ditambahkan/diupdate ke database, namun karna tidak digunakan dalam sistem maka bernilai false

Tabel 4. 12 *Class Model Setting*

No	Nama Method	Kegunaan Method
1	<i>table</i>	Pada Model Setting, properti \$table perlu didefinisikan karna penamaan tabel yang digunakan tidak sesuai dengan aturan baku di Eloquent. Pada Model Setting property \$table didefinisikan dengan syntax yaitu setting
2	<i>timestamps</i>	Mencatat kapan suatu record ditambahkan/diupdate ke database, namun karna tidak digunakan dalam sistem maka bernilai false

Tabel 4. 13 *Class Model Artikel*

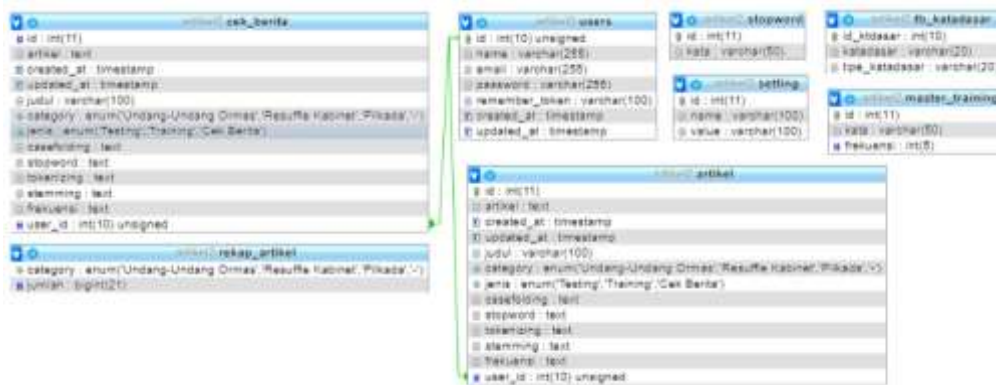
No	Nama Method	Kegunaan Method
1	<i>table</i>	Pada Model <i>Artikel</i> , properti \$table perlu didefinisikan karna penamaan tabel yang digunakan tidak sesuai dengan aturan baku di Eloquent. Pada Model <i>Artikel</i> property \$table didefinisikan dengan syntax yaitu artikel.
2	<i>fillable</i>	Untuk mendaftarkan atribut (nama kolom) yang bisa kita isi ketika melakukan insert atau update ke database. Berisi atribut nama-nama column yang ada di dalam tabel tersebut dan bersifat publik/akan diisi. Tabel tersebut berisi atribut judul, artikel, jenis, casefolding, stopword, tokenizing, stemming, frekuensi, user_id, category

Tabel 4. 14 *Class Model Cek Berita*

No	Nama Method	Kegunaan Method
1	<i>table</i>	Pada Model <i>CekBerita</i> , properti \$table perlu didefinisikan karna penamaan tabel yang digunakan tidak sesuai dengan aturan baku di Eloquent. Pada Model <i>CekBerita</i> property \$table didefinisikan dengan syntax yaitu <i>CekBerita</i> .

4.3.7 Entity Relationship Diagram (ERD)

Tahap terakhir ialah membuat desain ERD. *Entity Relationship Diagram* (ERD) merupakan gambaran komponen dan struktur database yang digunakan dalam pembungan sistem. ERD sistem klasifikasi dokumen berita politik dapat dilihat pada Gambar 4.6.



Gambar 4. 6 Entity Relationship Diagram

Dalam ERD sistem ini ada beberapa tabel yang tidak teralasi dengan tabel lain dikarenakan merupakan tabel pendukung dalam sistem ini. Tabel yang tidak teralasi adalah tabel stopwords, tb_katadasar, master training, dan setting. Tabel stopwords digunakan untuk menampung kata stopwords. Tb_katadasar adalah tabel yang berisi kata dasar untuk digunakan pada proses *stemming*. Tabel master_training sendiri merupakan tabel yang menampung daftar kata dari hasil *text-preprocessing* tahap akhir yaitu *stemming*, yang kemudian kata tersebut ditampung kedalam tabel master_training. Pada ERD ini, relasi yang digunakan adalah *one to many*, contohnya tabel users dan tabel artikel. Relasi ini menggambarkan bahwa setiap entitas users dapat berhubungan dengan banyak pada entitas artikel. Artinya bahwa satu users dapat mengelola banyak artikel. Berikut penjelasan pada setiap entitas yang ada pada ERD yang dapat ditunjukkan pada Tabel 4.15-Tabel 4.22.

Tabel 4. 15 Entitas users

No	Nama Atribut	Penjelasan
1	id	Id setiap user
2	name	Nama user
3	email	Atribut untuk autentifikasi login berupa email user
4	password	Atribut untuk autentifikasi login berupa password user
5	remember_token	Autentifikasi keamanan password
6	created_at	Tanggal user dibuat
7	update_at	Tanggal user diupdate

Tabel 4. 16 Entitas *stopword*

No	Nama Atribut	Penjelasan
1	id	Id stopwords bertipe data int
2	kata	Kata stopwords bertipe data varchar

Tabel 4. 17 Entitas *tb_kadadasar*

No	Nama Atribut	Penjelasan
1	id_ktdasar	Id kata dasar untuk proses stemming karena merupakan library
2	katadasar	Nama kata dasar
3	tipe_katadasar	Tipe kata dasar apakah termasuk nomina (kata benda), verba (kata kerja), atau adjektifa (kata sifat)

Tabel 4. 18 Entitas *artikel*

No	Nama Atribut	Penjelasan
1	id	Id setiap artikel yang diinputkan
2	artikel	Isi artikel/berita
3	created_at	Tanggal artikel dibuat
4	updated_at	Tanggal artikel diupdate
5	judul	Judul artikel
6	category	Category artikel yang diinputkan apakah termasuk training, testing, atau cek berita
7	jenis	Jenis kategori artikel apakah kategori uu ormas, kategori reshuffle kabinet, dan kategori uu ormas
8	casefolding	Kata hasil casefolding
9	stopword	Kata hasil stopwords
10	tokenizing	Kata hasil tokenizing
11	stemming	Kata hasil stemming
12	frekuensi	Frekuensi kata dari artikel yang diinputkan
13	user_id	Id user yang menginputkan

Tabel 4. 19 Entitas *master_training*

No	Nama Atribut	Penjelasan
1	id	Id dari daftar kata yang masuk
2	kata	Kata pada daftar kata
3	frekuensi	Jumlah frekuensi setiap kata

Tabel 4. 20 Entitas *setting*

No	Nama Atribut	Penjelasan
1	id	Id visitor
2	name	Nama visitor
3	value	Jumlah yang telah akses sistem

Tabel 4. 21 Entitas cek_berita

No	Nama Atribut	Penjelasan
1	id	Id setiap artikel yang diinputkan
2	artikel	Isi artikel/berita
3	created_at	Tanggal artikel dibuat
4	updated_at	Tanggal artikel diupdate
5	judul	Judul artikel
6	category	Category artikel yang diinputkan apakah termasuk training, testing, atau cek berita
7	jenis	Jenis kategori artikel apakah kategori uu ormas, kategori reshuffle kabinet, dan kategori uu ormas
8	casefolding	Kata hasil casefolding
9	stopword	Kata hasil stopword
10	tokenizing	Kata hasil tokenizing
11	stemming	Kata hasil stemming
12	frekuensi	Frekuensi kata dari artikel yang diinputkan
13	user_id	Id user yang menginputkan

Tabel 4. 22 Entitas rekap_artikel

No	Nama Atribut	Penjelasan
1	category	Category artikel yang diinputkan apakah termasuk training, testing, atau cek berita
2	jumlah	Jumlah

4.3 Implementasi Sistem

Setelah tahapan desain sistem selesai dilakukan, tahapan selanjutnya adalah implementasi sistem ke dalam bahasa pemrograman. Tahap implementasi adalah tahap pembuatan program dengan cara pengkodean atau koding sesuai dengan desain sistem yang telah dibuat. Pengkodean dibuat dengan menggunakan bahasa *Page Hyper Text Pre-Processor* (PHP) pada *framework Laravel 5.4*. Manajemen database dibuat dengan bantuan *Database Management System MySQL*. Pengkodean dibuat guna mendapatkan sistem yang dapat dioperasikan oleh pengguna. Tahap implementasi dilakukan berdasarkan hasil desain sistem yang sudah dilakukan. Pada proses pengkodean mengimplementasikan metode *Naïve Bayes Classifier*. Berikut penulisan kode program proses *Text Preprocessing* yang meliputi *casefolding*, *tokenizing*, *stopword*, dan *stemming* pada *ArtikelController* yang dapat dilihat pada Gambar 4.7-Gambar 4.13.

```

72
73     public function casefoldingProses ($id){ //prosecasefolding
74         $dat = Artikel::find($id);
75         $artikel = strtolower($dat->artikel);
76         $artikel2 = preg_replace('/^[^a-z]/', ' ', $artikel);
77         $artikel3 = preg_replace('/^[^w]/', ' ', $artikel2);
78         return $artikel3;
79     }

```

Gambar 4. 7 Kode Program Proses *Casefolding*

Pada Gambar 4.7 merupakan kode program dari kelas ArtikelController. Pada kode program baris 73 – 79 merupakan fungsi proses untuk *casefolding* untuk mengubah karakter huruf pada artikel menjadi huruf kecil semua dan selain karakter huruf dihilangkan. Proses *casefolding* menggunakan fungsi *strtolower* pada fungsi php.

```

96     public function tokenizingProses ($id){ //proses tokenizing
97         $dat = $this->casefoldingProses($id);
98         $dat2 = preg_replace('!\s+', ' ', $dat);
99         // dd($dat2);
100        $arr = explode(' ', $dat2);
101        $data = $arr;
102        // dd($data);
103        return $data;
104    }

```

Gambar 4. 8 Kode Program Proses *Tokenizing*

Pada Tabel 4.8 merupakan kode program dari kelas ArtikelController. Pada kode program baris 96 – 104 merupakan fungsi proses untuk *tokenizing* untuk memotong kalimat menjadi kata tunggal. Proses *tokenizing* menggunakan fungsi *explode* pada fungsi php.

```

129     public function dbstopword(){ //nampilin dari db stopword
130         $data = Stopword::all();
131         $a = array(); //mebuat array huruf (kosongan) deklarasi
132         foreach ($data as $key => $value) {
133             $a[] = $value->kata; //data yg didatabase diambil
134         }
135         return $a;
136     }

```

Gambar 4. 9 Kode Program Proses *Stopword 1*

Pada Tabel 4.9 merupakan kode program dari kelas ArtikelController. Pada kode program baris 129 – 136 merupakan fungsi proses untuk *stopword* untuk menampilkan daftar stoplist yang ada pada library.

```

138     public function stopwordsProses($id){           //proses stopwords
139         $arr = $this->dbstopword();
140         // dd($data);
141         $data = $this->tokenizingProses($id);
142         $a = array();
143         for ($j=0; $j < count($arr); $j++) {
144             foreach($data as $key => $value) {
145                 if($value == $arr[$j]){
146                     unset($data[$key]);
147                 }
148             }
149             // echo count($data)."<br>";
150             if($j == count($arr)-1){
151                 foreach ($data as $key => $value) {
152                     $a[] = $value;
153                 }
154             }
155         }
156         return $a;
157     }

```

Gambar 4. 10 Kode Program Proses *Stopword 2*

Pada Tabel 4.10 merupakan kode program dari kelas ArtikelController. Pada kode program baris 138 – 157 merupakan fungsi proses untuk *stopword*.

```

200     function stemming_cari($kata){ //mencari di database
201         // include "koneksi.php";
202         $hasil = DB::select("SELECT * FROM tb_katadasar WHERE katadasar='$kata'");
203         // echo count($hasil)."<br>";
204         return count($hasil);
205     }
206
207     //langkah 1 - hapus partikel
208     function stemming_hapuspartikel($kata){
209         if($this->stemming_cari($kata)!=1){
210             if((substr($kata, -3) == 'kah' )|| ( substr($kata, -3) == 'lah' )|| ( substr($kata, -3) == 'pun' )){
211                 $kata = substr($kata, 0, -3);
212             }
213         }
214         return $kata;
215     }
216
217     //langkah 2 - hapus possessive pronoun
218     function stemming_hapuspp($kata){
219         if($this->stemming_cari($kata)!=1){
220             if(strlen($kata) > 4){
221                 if((substr($kata, -2)== 'ku' )|| (substr($kata, -2)== 'mu' )){
222                     $kata = substr($kata, 0, -2);
223                 }else if((substr($kata, -3)== 'nya' )){
224                     $kata = substr($kata,0,-3);
225                 }
226             }
227         }
228         return $kata;
229     }

```

Gambar 4. 11 Kode Program Proses *Stemming 1*

Pada Tabel 4.11 merupakan kode program dari kelas ArtikelController. Pada kode program baris 200-229 merupakan fungsi proses untuk *stemming*. Terdapat beberapa function untuk proses *stemming* yang terdiri dari menghapus partikel dan menghapus *possessive pronoun* pada sebuah kata.

```

200 function stemming_hapusawalan($kata){
201     if($this->stemming_south($kata)==1){
202
203         if(substr($kata,0,4)=="meng"){
204             if(substr($kata,4,1)=="e"||substr($kata,4,1)=="o"){
205                 $kata = "n".substr($kata,4);
206             }else{
207                 $kata = substr($kata,4);
208             }
209         }else if(substr($kata,0,4)=="meny"){
210             $kata = "n".substr($kata,4);
211         }else if(substr($kata,0,3)=="men"){
212             $kata = substr($kata,3);
213         }else if(substr($kata,0,3)=="mem"){
214             if(substr($kata,3,1)=="i" || substr($kata,3,1)=="e" || substr($kata,3,1)=="o" || substr($kata,3,1)=="u"){
215                 $kata = "n".substr($kata,3);
216             }else{
217                 $kata = substr($kata,3);
218             }
219         }else if(substr($kata,0,2)=="me"){
220             $kata = substr($kata,2);
221         }else if(substr($kata,0,4)=="meng"){
222             if(substr($kata,4,1)=="e" || substr($kata,4,1)=="o"){
223                 $kata = "n".substr($kata,4);
224             }else{
225                 $kata = substr($kata,4);
226             }
227         }else if(substr($kata,0,4)=="meny"){
228             if(substr($kata,4,1)=="e" || substr($kata,4,1)=="o" || substr($kata,4,1)=="u" || substr($kata,4,1)=="i"){
229                 $kata = "n".substr($kata,4);
230             }else{
231                 $kata = substr($kata,4);
232             }
233         }else if(substr($kata,0,3)=="mem"){
234             if(substr($kata,3,1)=="i" || substr($kata,3,1)=="e" || substr($kata,3,1)=="o" || substr($kata,3,1)=="u"){
235                 $kata = "n".substr($kata,3);
236             }else{
237                 $kata = substr($kata,3);
238             }
239         }else if(substr($kata,0,3)=="men"){
240             if(substr($kata,3,1)=="i" || substr($kata,3,1)=="e" || substr($kata,3,1)=="o" || substr($kata,3,1)=="u"){
241                 $kata = "n".substr($kata,3);
242             }else{
243                 $kata = substr($kata,3);
244             }
245         }
246     }
247 }

```

Gambar 4. 12 Kode Program Proses *Stemming 2*

Pada Tabel 4.12 merupakan kode program dari kelas ArtikelController. Pada kode program baris 232-272 merupakan fungsi proses untuk *stemming*. Terdapat beberapa function untuk proses *stemming* yaitu menghapus awalan dari sebuah kata.

```

229 public function stemmingProses($id) //proses stemming
230 {
231     $arr = $this->stopwordProses($id);
232     // dd($arr);
233     // $data = implode("", $arr);
234     $data = array();
235     $i = 1;
236     foreach ($arr as $key => $value) {
237         $i = $this->stemming_hapusawalan($this->stemming_hapusawalan($this->stemming_hapuspp($this->stemming_hapuspartikel($value))));
238         $data[] = $i;
239         // echo $i++." : ".$value." | ".$i."chr";
240     }
241     return implode("", $data);
242     // return view('admin.stemming',compact('data'));
243 }

```

Gambar 4. 13 Kode Program Proses *Stemming 3*

Pada Tabel 4.13 merupakan kode program dari kelas ArtikelController. Pada kode program baris 325-338 merupakan fungsi proses untuk *stemming*.

```

273     }else if(substr($kata,0,2)=="di"){
274         $kata = substr($kata,2);
275     }else if(substr($kata,0,3)=="ter"){
276         $kata = substr($kata,3);
277     }else if(substr($kata,0,2)=="ke"){
278         $kata = substr($kata,2);
279     }
280 }
281 return $kata;
282 }
283
284 //langkah 4 hapus second order prefiks (awalan kedua)
285 function stemming_hapusawalan2($kata){
286     if($this->stemming_cari($kata)!=1){
287
288         if(substr($kata,0,3)=="ber"){
289             $kata = substr($kata,3);
290         }else if(substr($kata,0,3)=="bel"){
291             $kata = substr($kata,3);
292         }else if(substr($kata,0,2)=="be"){
293             $kata = substr($kata,2);
294         }else if(substr($kata,0,3)=="per" && strlen($kata) > 5){
295             $kata = substr($kata,3);
296         }else if(substr($kata,0,2)=="pe" && strlen($kata) > 5){
297             $kata = substr($kata,2);
298         }else if(substr($kata,0,3)=="pel" && strlen($kata) > 5){
299             $kata = substr($kata,3);
300         }else if(substr($kata,0,2)=="se" && strlen($kata) > 5){
301             $kata = substr($kata,2);
302         }
303     }
304     return $kata;
305 }
306
307 //langkah 5 hapus suffiks
308 function stemming_hapusakhiran($kata){
309     if($this->stemming_cari($kata)!=1){
310         if (substr($kata, -3)=="kan" ){
311             $kata = substr($kata, 0, -3);
312         }
313         else if(substr($kata, -1)=="i" ){
314             $kata = substr($kata, 0, -1);
315         }else if(substr($kata, -2)=="an"){
316             $kata = substr($kata, 0, -2);
317         }
318     }
319 }

```

Gambar 4. 14 Kode Program Proses *Stemming* 4

Pada Tabel 4.14 merupakan kode program dari kelas ArtikelController. Pada kode program baris 273-319 merupakan fungsi proses untuk *stemming*. Terdapat beberapa function untuk proses *stemming* yaitu menghapus akhiran dan awalan dari sebuah kata.

4.4 Pengujian Sistem

4.4.1 Pengujian Blackbox

Pengujian *black box* merupakan pengujian yang dilakukan oleh *user* dengan menjalankan program secara langsung dan menganalisis *input* dan *output* yang dihasilkan oleh aplikasi. Pengujian *black box* secara lengkap dapat dilihat di lampiran E.

4.4.2 Uji Performasi

Untuk melakukan uji akurasi, peneliti menggunakan metode perbandingan *precision* dan *recall*, serta digabungkan dengan menggunakan perhitungan *f-measure*. Pada tahap ini peneliti melakukan uji pergormasi berdasarkan data uji berita yaitu sebanyak 60 dokumen berita.

Langkah pengujian dilakukan dengan menginputkan data uji pada sistem klasifikasi. Peneliti melakukan pengecekan terhadap berapa jumlah data yang berhasil di klasifikasi dan berapa jumlah data yang sesuai dengan data aktual uji pada dokumen berita politik. Tahap akhir uji performasi adalah menghitung nilai *precision* dan *recall* dari data dokumen berita yang diujikan. Setelah nilai *precision* dan *recall* diketahui, maka dilakukan perhitungan *f-measure*.

BAB 6. PENUTUP

Bab ini berisi mengenai kesimpulan dan saran dari peneliti tentang penelitian yang telah dilakukan. Kesimpulan dan saran tersebut diharapkan dapat digunakan sebagai acuan pada penelitian selanjutnya.

6.1 Kesimpulan

Berdasarkan analisis dan pengujian yang telah dilakukan maka kesimpulan yang dapat diambil adalah sebagai berikut:

1. Sistem Klasifikasi Dokumen yang dikembangkan ini terdiri 4 fitur utama adalah yaitu fitur: pengelolaan data *training*, pengelolaan data *testing*, pengelolaan data *stopword* dan cek kategori berita. Mengacu pada batasan masalah, desain sistem membutuhkan 5 kelas untuk 4 fitur utamanya yang terdiri dari 2 kelas untuk Controller dan 3 kelas untuk Model. Kelas Controller yaitu ArtikelController digunakan untuk menyediakan berbagai fungsi yang akan ditampilkan di view dan pengerjaan proses logika untuk 3 fitur yaitu data training, data testing, dan cek kategori dokumen. Kelas StopwordController untuk memanggil fungsi untuk pengelolaan data stopwords. Kelas Model yaitu Artikel, Stopword, dan MasterTraining. Kelas Model digunakan melakukan pengolahan yang berhubungan dengan database. Selain itu, sistem ini membutuhkan 6 tabel yang disimpan dalam basis data yaitu: artikel, stopwords, master_training, tb_katadasar, setting, dan users.
2. Pengujian klasifikasi diukur dengan uji performansi *precision*, *recall*, dan *fmeasure*. Hasil dari 60 data uji menghasilkan performansi kelas kategori rata-rata dengan nilai *precision* sebesar 0.984126984 atau 98.41%, nilai *recall* sebesar 0.983333333 atau 98.33%, dan nilai *fmeasure* sebesar 0.983729999 atau 98.37%. Hasil dari uji performansi tersebut apabila dibandingkan dengan penelitian terdahulu maka uji performansi dapat dikatakan sangat baik dan sistem mampu melakukan klasifikasi kategori

pada dokumen berita politik. Hal ini membuktikan bahwa algoritma *Naïve Bayes Classifier* dapat diterapkan untuk proses klasifikasi pada dokumen yang mempunyai similaritas yang tinggi.

6.2 Saran

Adapun saran yang ditujukan untuk memberikan masukan yang lebih baik yaitu sebagai berikut:

1. Pada penelitian ini dataset masih didapatkan secara manual. Maka dari itu, diharapkan pada penelitian selanjutnya data yang diperoleh dapat dilakukan secara otomatis atau menggunakan teknik *crawling* data.
2. Pengujian data tidak hanya dapat menggunakan dokumen berita politik, namun dapat menggunakan objek dokumen lainnya.
3. Jumlah dataset yang digunakan pada penelitian selanjutnya diharapkan bisa lebih dari 300 dokumen agar mendapatkan nilai performansi yang lebih baik lagi.

DAFTAR PUSTAKA

Adi, T.N. (2015). GENERATOR KODE UNIT TESTING UNTUK JAVASCRIPT BERBASIS FRAMEWORK QUNIT. 2, 9.

Alexa Internet (2018). Top Sites in Indonesia.

Anugroho, P., Winarno, I., and Mubtada'i, N. (2010). Klasifikasi Spam Email dengan Metode Naïve Bayes Classifier. EEPIS Final Proj.

Bustami, B. (2013). Penerapan Algoritma Naïve Bayes Untuk Mengklasifikasi Data Nasabah Asuransi. TECHSI-J. Tek. Inform. 5.

Chandra, D.N., Indrawan, G., and Sukajaya, I.N. (2016a). Klasifikasi Berita Lokal Radar Malang Menggunakan Metode Naïve Bayes Dengan Fitur N-Gram. 9.

Chandra, D.N., Indrawan, G., and Sukajaya, I.N. (2016b). Klasifikasi Berita Lokal Radar Malang Menggunakan Metode Naïve Bayes Dengan Fitur N-Gram. J. Ilm. Teknol. Dan Informasia ASIA JITIKA 10, 9.

Dharwiyanti, S., and Wahono, R.S. (2003). Pengantar Unified Modeling Language (UML).

Ebranda, Mardiani, and Tinaliah (2013). Penerapan Metode Naive Bayes Untuk Sistem Klasifikasi SMS Pada Smartphone Android.

Hamzah, A. (2012). Klasifikasi teks dengan naïve bayes classifier (nbc) untuk pengelompokan teks berita dan abstract akademis. In Prosiding Seminar Nasional Aplikasi Sains & Teknologi (SNAST) Periode III, pp. 269–277.

Hidayatullah, A.F., and Ma'arif, M.R. (2016). Penerapan Text Mining dalam Klasifikasi Judul Skripsi. In Seminar Nasional Aplikasi Teknologi Informasi (SNATI), p.

Lestari, N.P. (2016). UJI RECALL AND PRECISION SISTEM TEMU KEMBALI INFORMASI OPAC PERPUSTAKAAN ITS SURABAYA. 18.

Ling, J., Kencana, I.P.E.N., and Oka, T.B. (2014). Analisis Sentimen Menggunakan Metode Naïve Bayes Classifier Dengan Seleksi Fitur Chi Square. E-J. Mat. 3, 92–99.

Manalu, B.U. (2014). ANALISIS SENTIMEN PADA TWITTER MENGGUNAKAN TEXT MINING.

Mustaqbal, M.S., Firdaus, R.F., and Rahmadi, H. (2015). PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN). 6.

Nidhra, S. (2012). Black Box and White Box Testing Techniques - A Literature Review. *Int. J. Embed. Syst. Appl.* 2, 29–50.

Nurhadi, A. (2015). Klasifikasi Konten Berita Digital Bahasa Indonesia Menggunakan Support Vector Machines (SVM) Berbasis Particle Swarm Optimization (PSO). 3, 9.

Pakpahan, D., and Widyastuti, H. (2014). Aplikasi Opinion Mining dengan Algoritma Naïve Bayes untuk Menilai Berita Online. *J. Integrasi* 6, 1–10.

Pressman, R.S. (2010). *Software engineering: a practitioner's approach* (New York: McGraw-Hill Higher Education).

Rahman, A., and Doewes, A. (2017). Online News Classification Using Multinomial Naive Bayes. 6, 7.

Romadhoni, E.N.A., Widiyaningtyas, T., and Pujiyanto, U. (2015). IMPLEMENTASI MODEL WATERFALL PADA PENGEMBANGAN SISTEM INFORMASI ALUMNI SMKN 1 JENANGAN PONOROGO. 8.

Rosmalal, D., and Falahah (2007). PEMODELAN PROSES BISNIS B2B DENGAN BPMN (STUDI KASUS PENGADAAN BARANG PADA DIVISI LOGISTIK). 5.

Sagita, R.A., and Sugiarto, H. (2016). Penerapan Metode Waterfall Pada Sistem Informasi Penjualan Furniture Berbasis Web. 5, 7.

Saraswati, N.W.S. (2013). NAÏVE BAYES CLASSIFIER DAN SUPPORT VECTOR MACHINES UNTUK SENTIMENT ANALYSIS. 7.

Sasmito, G.W. (2017). Penerapan Metode Waterfall Pada Desain Sistem Informasi Geografis Industri Kabupaten Tegal. *J. Inform.* 2, 7.

Setiawan, A., Astuti, I.F., and Kridalaksana, A.H. (2016). Klasifikasi Dan Pencarian Buku Referensi Akademik Menggunakan Metode Naïve Bayes Classifier (Nbc)(Studi Kasus: Perpustakaan Daerah Provinsi Kalimantan Timur). *Inform. Mulawarman* 10, 1–10.

Setyaji, G.F.W. (2016). SISTEM PENDUKUNG KEPUTUSAN PENENTUAN JURUSAN PADA PENDAFTARAN SISWA BARU JALUR PRESTASI MENGGUNAKAN METODE NAÏVE BAYES (Studi Kasus di SMKN 1 Boyolangu Tulungagung). 11.

Sussolaikah, K., and Alwi, A. (2016). SENTIMENT ANALYSIS TERHADAP ACARA TELEVISI MATA NAJWA BERDASARKAN OPINI MASYARAKAT PADA MICROBLOGGING TWITTER. 10.

Yuliawan, Y., Sunarto, M.J.D., Soebijono, T., and Baruk, J.R.K. (2013). PENGEMBANGAN SISTEM INFORMASI PENDATAAN JEMAAT. 2, 10.

Yunitarini, R., and Hastarita, F. (2016). PEMODELAN PROSES BISNIS AKADEMIK TEKNIK INFORMATIKA UNIVERSITAS TRUNOJOYO DENGAN BUSINESS PROCESS MODELLING NOTATION (BPMN). 5, 8.

LAMPIRAN

A. Skenario *Usecase*

A.1 Skenario Cek Kategori Dokumen

Tabel A. 1 Skenario Cek Kategori Dokumen

No. Usecase	UCS 01
Nama Usecase	Cek Kategori Dokumen
Aktor	Editor
PreKondisi	Editor membuka halaman sistem utama
PraKondisi	Editor memasukkan artikel berita yang ingin dicek
Flow Events	
Skenario Normal : Cek Kategori Dokumen	
Aksi Aktor	Aksi Sistem
1. Membuka halaman awal sistem	2. Menampilkan halaman awal sistem Cek Kategori Dokumen: <ul style="list-style-type: none"> • <i>Textbox</i> (Artikel) • <i>Button</i> (Cek Berita) • <i>Home</i> • Tentang • Admin • Data Berita Politik

A.2 Skenario Hasil Cek Kategori Dokumen

Tabel A. 2 Skenario Hasil Cek Kategori Dokumen

No. Usecase	UCS 02
Nama Usecase	Hasil Cek Kategori Dokumen
Aktor	Editor
PreKondisi	Editor membuka halaman sistem utama
PraKondisi	Editor memasukkan artikel berita yang ingin dicek
Flow Events	
Skenario Normal : Cek Kategori Dokumen	
Aksi Aktor	Aksi Sistem
1. Membuka halaman awal	2. Menampilkan halaman awal sistem Cek Kategori Dokumen: <ul style="list-style-type: none"> • <i>Textbox</i> (Artikel) • <i>Button</i> (Cek Berita) • Home • Tentang • Admin • Data Berita Politik
4. Mengisi <i>textbox</i> isi artikel	

5. Klik <i>button</i> Cek Berita	6. Menampilkan hasil kategori
Skenario Alternatif : Data kosong	
Aksi Aktor	Aksi Sistem
5. Klik <i>button</i> Cek Berita	6. Menampilkan <i>alert</i> “Please fill out this field”
	7. Menampilkan halaman awal sistem

A.3 Skenario *Login*

Tabel A. 3 Skenario *Login*

No. Usecase	UCS 03
Nama Usecase	<i>Login</i>
Aktor	Admin
PreKondisi	Admin telah memiliki <i>email</i> dan <i>password</i> untuk <i>login</i> sebagai Admin
PraKondisi	Admin berhasil <i>login</i>
Flow Events	
Skenario Normal : <i>Login</i>	
Aksi Aktor	Aksi Sistem
1. Masukkan alamat <i>email</i> dan <i>password</i>	
2. Klik tombol <i>Login</i>	3. Menampilkan halaman Admin
Skenario Alternatif : Data tidak lengkap	
Aksi Aktor	Aksi Sistem
2. Klik tombol <i>Login</i>	3. Menampilkan <i>alert</i> “Please fill out this field”
	4. Menampilkan halaman <i>login</i>
Skenario Alternatif : Data tidak valid	
Aksi Aktor	Aksi Sistem
2. Klik tombol <i>Login</i>	3. Menampilkan <i>alert</i> “These credentials do not match our record.”
	4. Menampilkan halaman <i>login</i>

A.4 Skenario Melihat *Dashboard*

Tabel A. 4 Skenario Melihat *Dashboard*

No. Usecase	UCS 04
Nama Usecase	Melihat <i>Dashboard</i>
Aktor	Admin
PreKondisi	Admin memilih menu <i>Dashboard</i>
PraKondisi	Admin berhasil melihat chart pada dashboard
Flow Events	

Skenario Normal : Login	
Aksi Aktor	Aksi Sistem
1. Klik menu Dashboard	2. Menampilkan halaman Dashboard: <ul style="list-style-type: none"> • <i>Chart</i> Jumlah Dokumen Training Tiap Kategori • <i>Chart</i> Jumlah Dokumen Testing Tiap Kategori • <i>Chart</i> Jumlah Artikel Cek Berita Tiap Kategori • <i>Chart</i> Jumlah <i>Stopword</i> • <i>Chart</i> Jumlah <i>Visitor</i>

A.5 Skenario Melihat *Text Preprocessing* Data Training

Tabel A. 5 Skenario Melihat Text Preprocessing Data Training

No. Usecase	UCS 05
Nama Usecase	Mengelola Data <i>Training</i>
Aktor	Admin
PreKondisi	Admin memilih menu Data <i>Training</i>
PraKondisi	Admin berhasil menambahkan, melihat, mengupdate, dan menghapus data <i>training</i>

Flow Events

Skenario Normal : Tambah data <i>training</i>	
Aksi Aktor	Aksi Sistem
1. Klik menu Data <i>Training</i>	2. Menampilkan halaman menu Data <i>Training</i> yang berisi : <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action Button</i> Frekuensi • <i>Action Button</i> Daftar Kata • <i>Action Button</i> Probability • <i>Action Button</i> Tambah Dokumen Data Training
3. Klik Tambah Dokumen Data Training	4. Menampilkan form Tambah Data <i>Training</i> yang berisi <i>text field</i> :

	<ul style="list-style-type: none"> • Judul • Artikel • Kategori Berita • <i>Action button Save</i> • <i>Action button Close</i>
5. Mengisi form Tambah Data <i>Training</i>	
6. Klik <i>button Save</i>	
	7. Menampilkan halaman Data <i>Training</i>
Skenario Alternative : Data tidak lengkap	
Aksi Aktor	Aksi Sistem
6. Klik <i>button Save</i>	
	7. Menampilkan <i>alert "Please fill out this field"</i>
	7. Menampilkan form Tambah Data <i>Training</i>
Skenario Alternatif : Batal tambah data	
Aksi Aktor	Aksi Sistem
5. Klik <i>button Close</i>	
	6. Menampilkan halaman data <i>training</i>
Skenario Normal : Lihat data <i>training</i>	
Aksi Aktor	Aksi Sistem
1. Klik menu Data <i>Training</i>	
	2. Menampilkan halaman menu Data <i>Training</i> yang berisi tabel: <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action Button Frekuensi</i> • <i>Action Button Daftar Kata</i> • <i>Action Button Probability</i> • <i>Action Button Tambah Dokumen Data Training</i>
Skenario Normal : Update data <i>training</i>	
Aksi Aktor	Aksi Sistem
1. Klik menu Data <i>Training</i>	
	2. Menampilkan halaman menu Data <i>Training</i> yang berisi tabel: <ul style="list-style-type: none"> • No • Judul • Kategori

	<ul style="list-style-type: none"> • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action Button Frekuensi</i> • <i>Action Button Daftar Kata</i> • <i>Action Button Probability</i> • <i>Action Button Tambah Dokumen Data Training</i>
2. Klik <i>button Update</i>	3. Menampilkan form <i>Update Data Training</i> yang dipilih: <ul style="list-style-type: none"> • Judul • Artikel • Kategori Berita • <i>Action button Save</i> • <i>Action button Close</i>
4. Mengubah data <i>training</i>	
5. Klik <i>button Save</i>	6. Menampilkan halaman data <i>training</i>
Skenario Alternatif : Data tidak lengkap	
Aksi Aktor	Aksi Sistem
5. Klik <i>button Save</i>	6. Menampilkan <i>alert "Please fill out this field"</i>
	7. Menampilkan form <i>update data training</i>
Skenario Alternatif : Batal <i>update data</i>	
Aksi Aktor	Aksi Sistem
5. Klik <i>button Close</i>	6. Menampilkan halaman data <i>training</i>
Skenario Normal : Hapus data <i>training</i>	
Aksi Aktor	Aksi Sistem
1. Klik menu <i>Data Training</i>	2. Menampilkan halaman menu <i>Data Training</i> yang berisi : <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i>

- *Action button Stopword*
- *Action button Stemming*
- *Action Button Frekuensi*
- *Action Button Daftar Kata*
- *Action Button Probability*
- *Action Button Tambah Dokumen Data Training*

3. Klik *button Delete* pada data yang diinginkan

4. Menampilkan halaman data *training*

A.6 Skenario Melihat Frekuensi Kata Data *Training*

Tabel A. 6 Skenario Melihat Frekuensi Kata Data *Training*

No. Usecase	UCS 07
Nama Usecase	Melihat Frekuensi Kata Data <i>Training</i>
Aktor	Admin
PreKondisi	Admin memilih menu Data <i>Training</i>
PraKondisi	Admin berhasil melihat frekuensi kata data <i>training</i>

Flow Events

Skenario Normal : Melihat frekuensi kata data *training*

Aksi Aktor

Aksi Sistem

1. Klik menu Data *Training*

2. Menampilkan halaman menu Data *Training* yang berisi :

- No
- Judul
- Kategori
- Artikel
- Tanggal dibuat
- *Action button Update*
- *Action button Delete*
- *Action button Casefolding*
- *Action button Tokenizing*
- *Action button Stopword*
- *Action button Stemming*
- *Action Button Frekuensi*
- *Action Button Daftar Kata*
- *Action Button Probability*
- *Action Button Tambah Dokumen Data Training*

3. Klik *button* Frekuensi

4. Menampilkan frekuensi kata setiap data dokumen data *training*

A.7 Skenario Melihat Daftar Kata Data *Training*Tabel A. 7 Skenario Melihat Daftar Kata Data *Training*

No. Usecase	UCS 08
Nama Usecase	Melihat Daftar Kata Data <i>Training</i>
Aktor	Admin
PreKondisi	Admin memilih menu Data <i>Training</i>
PraKondisi	Admin berhasil melihat daftar kata data <i>training</i>
Flow Events	
Skenario Normal : Melihat daftar kata data <i>training</i>	
Aksi Aktor	Aksi Sistem
1. Klik menu Data <i>Training</i>	2. Menampilkan halaman menu Data <i>Training</i> yang berisi : <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action Button Frekuensi</i> • <i>Action Button Daftar Kata</i> • <i>Action Button Probability</i> • <i>Action Button Tambah Dokumen Data Training</i>
3. Klik <i>button</i> Daftar Kata	4. Menampilkan daftar kata dari semua kategori pada data <i>training</i>

A.8 Skenario Melihat *Probability* Kata Data *Training*Tabel A. 8 Skenario Melihat *Probability* Kata Data *Training*

No. Usecase	UCS 09
Nama Usecase	Melihat <i>Probability</i> Kata Data <i>Training</i>
Aktor	Admin
PreKondisi	Admin memilih menu Data <i>Training</i>
PraKondisi	Admin berhasil melihat <i>probability</i> kata data <i>training</i>
Flow Events	
Skenario Normal : Melihat <i>probability</i> kata data <i>training</i>	
Aksi Aktor	Aksi Sistem

1. Klik menu <i>Data Training</i>	2. Menampilkan halaman menu <i>Data Training</i> yang berisi : <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action Button Frekuensi</i> • <i>Action Button Daftar Kata</i> • <i>Action Button Probability</i> • <i>Action Button Tambah Dokumen Data Training</i>
3. Klik <i>button Probability</i>	4. Menampilkan nprobabilitas setiap kata dari semua kategori data <i>training</i>

A.9 Skenario Mengelola Data *Testing*

Tabel A. 9 Skenario Mengelola Data *Testing*

No. Usecase	UCS 10
Nama Usecase	Mengelola Data <i>Testing</i>
Aktor	Admin
PreKondisi	Admin memilih menu Data <i>Testing</i>
PraKondisi	Admin berhasil menambahkan, melihat, mengupdate dan menghapus data <i>testing</i>

Flow Events

Skenario Normal : Tambah data *testing*

Aksi Aktor	Aksi Sistem
1. Klik menu Data <i>Testing</i>	2. Menampilkan halaman menu Data <i>Testing</i> yang berisi tabel: <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i>

	<ul style="list-style-type: none"> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action Button Hasil</i> • <i>Action Button Tambah Dokumen Data Testing</i>
3. Klik Tambah Dokumen Data Testing	
	4. Menampilkan form Tambah Data <i>Testing</i> yang meliputi <i>text field</i> : <ul style="list-style-type: none"> • Judul • Artikel • <i>Action button Save</i> • <i>Action button Close</i>
4. Mengisi form Tambah Data <i>Testing</i>	
5. Klik <i>button Save</i>	
	6. Menampilkan halaman Data <i>Testing</i>
Skenario Alternative : Data tidak lengkap	
Aksi Aktor	Aksi Sistem
5. Klik <i>button Save</i>	
	6. Menampilkan <i>alert</i> “Please fill out this field”
	7. Menampilkan form Tambah Data <i>Testing</i>
Skenario Alternatif : Batal tambah data	
Aksi Aktor	Aksi Sistem
5. Klik <i>button Close</i>	
	6. Menampilkan halaman data <i>testing</i>
Skenario Normal: Lihat data <i>testing</i>	
Aksi Aktor	Aksi Sistem
1. Klik menu Data <i>Testing</i>	
	2. Menampilkan halaman menu Data <i>Testing</i> yang berisi tabel: <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action Button Hasil</i> • <i>Action Button Tambah Dokumen Data Testing</i>
Skenario Normal : Update data <i>testing</i>	
Aksi Aktor	Aksi Sistem
1. Klik menu Data <i>Testing</i>	

	<p>2. Menampilkan halaman menu Data <i>Testing</i> yang berisi :</p> <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action Button Hasil</i> • <i>Action Button Tambah Dokumen Data Testing</i>
<p>3. Klik <i>button Update</i></p>	<p>4. Menampilkan form <i>Update Data Testing</i> yang dipilih</p> <ul style="list-style-type: none"> • Judul • Artikel • <i>Action button Save</i> • <i>Action button Close</i>
<p>5. Mengubah data <i>testing</i></p>	
<p>6. Klik <i>button Save</i></p>	
<p>7. Menampilkan halaman data <i>testing</i></p>	
<p>Skenario Alternatif : Data tidak lengkap</p>	
<p>Aksi Aktor</p>	<p>Aksi Sistem</p>
<p>6. Klik <i>button Save</i></p>	
	<p>7. Menampilkan <i>alert "Please fill out this field"</i></p>
<p>8. Menampilkan form tambah data <i>testing</i></p>	
<p>Skenario Alternatif : Batal <i>update data</i></p>	
<p>Aksi Aktor</p>	<p>Aksi Sistem</p>
<p>5. Klik <i>button Close</i></p>	
<p>6. Menampilkan halaman data <i>testing</i></p>	
<p>Skenario Normal : Hapus data <i>testing</i></p>	
<p>Aksi Aktor</p>	<p>Aksi Sistem</p>
<p>1. Klik menu Data <i>Testing</i></p>	
	<p>2. Menampilkan halaman menu Data <i>Testing</i> yang berisi tabel:</p>
	<ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i>

- *Action button Casefolding*
- *Action button Tokenizing*
- *Action button Stopword*
- *Action button Stemming*
- *Action Button Hasil*
- *Action Button Tambah Dokumen Data Testing*

3. Klik *button Delete* pada data yang diinginkan

4. Menampilkan halaman data *testing*

A.10 Skenario Melihat *Text Preprocessing* Data *Testing*

Tabel A. 10 Skenario Melihat *Text Preprocessing* Data *Testing*

No. Usecase	UCS 11
Nama Usecase	Melihat <i>Text Preprocessing</i> Data <i>Testing</i>
Aktor	Admin
PreKondisi	Admin memilih menu Data <i>Testing</i>
PraKondisi	Admin dapat melihat <i>Text Preprocessing</i> dari salah satu berita yang dipilih

Flow Events

Skenario Normal : Melihat *Text Preprocessing* Data *Training (Casefolding)*

Aksi Aktor

Aksi Sistem

1. Klik menu Data *Testing*

2. Menampilkan halaman menu Data *Testing* yang berisi :

- No
- Judul
- Kategori
- Artikel
- Tanggal dibuat
- *Action button Update*
- *Action button Delete*
- *Action button Casefolding*
- *Action button Tokenizing*
- *Action button Stopword*
- *Action button Stemming*
- *Action Button Hasil*
- *Action Button Tambah Dokumen Data Testing*

3. Klik *button Casefolding*

4. Menampilkan halaman *Text Preprocessing Casefolding*

Skenario Normal : Melihat *Text Preprocessing* Data *Testing (Tokenizing)*

Aksi Aktor

Aksi Sistem

1. Klik menu Data *Testing*

	2. Menampilkan halaman menu <i>Data Testing</i> yang berisi : <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action Button Hasil</i> • <i>Action Button Tambah Dokumen Data Testing</i>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

3. Klik *button Tokenizing*

4. Menampilkan halaman *Text Preprocessing Tokenizing*

Skenario Normal : Melihat *Text Preprocessing Data Testing (Stopword)*

Aksi Aktor

Aksi Sistem

1. Klik menu *Data Testing*

2. Menampilkan halaman menu *Data Testing* yang berisi :

- No
- Judul
- Kategori
- Artikel
- Tanggal dibuat
- *Action button Update*
- *Action button Delete*
- *Action button Casefolding*
- *Action button Tokenizing*
- *Action button Stopword*
- *Action button Stemming*
- *Action Button Hasil*
- *Action Button Tambah Dokumen Data Testing*

3. Klik *button Stopword*

4. Menampilkan halaman *Text Preprocessing Stopword*

Skenario Normal : Melihat *Text Preprocessing Data Testing (Stemming)*

Aksi Aktor

Aksi Sistem

1. Klik menu *Data Testing*

2. Menampilkan halaman menu *Data Testing* yang berisi :

- No
- Judul

	<ul style="list-style-type: none"> • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action Button Hasil</i> • <i>Action Button Tambah Dokumen Data Testing</i>
2. Klik <i>button Stemming</i>	
	3. Menampilkan halaman <i>Text Preprocessing Stemming</i>

A.11 Skenario Melihat Hasil Klasifikasi Data *Testing*

Tabel A. 11 Skenario Melihat Hasil Klasifikasi Data *Testing*

No. Usecase	UCS 12
Nama Usecase	Melihat Hasil Klasifikasi Data <i>Testing</i>
Aktor	Admin
PreKondisi	Admin memilih menu Data <i>Testing</i>
PraKondisi	Admin berhasil melihat hasil klasifikasi data <i>testing</i>

Flow Events

Skenario : Melihat hasil klasifikasi data *testing*

Aksi Aktor	Aksi Sistem
1. Klik menu Data <i>Testing</i>	2. Menampilkan halaman menu Data <i>Testing</i> yang berisi tabel: <ul style="list-style-type: none"> • No • Judul • Kategori • Artikel • Tanggal dibuat • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action button Casefolding</i> • <i>Action button Tokenizing</i> • <i>Action button Stopword</i> • <i>Action button Stemming</i> • <i>Action Button Hasil</i> • <i>Action Button Tambah Dokumen Data Testing</i>
3. Klik <i>button Hasil</i>	

4. Menampilkan halaman hasil *probability* kata dan hasil klasifikasi kategori data testing

A.12 Skenario Mengelola Data *Stopword*

Tabel A. 12 Skenario Mengelola Data *Stopword*

No. Usecase	UCS 13
Nama Usecase	Mengelola Data <i>Stopword</i>
Aktor	Admin
PreKondisi	Admin memilih menu Data <i>Stopword</i>
PraKondisi	Admin berhasil menambahkan, melihat, mengedit, dan menghapus data <i>stopword</i>
Flow Events	
Skenario Normal : Tambah data <i>stopword</i>	
Aksi Aktor	Aksi Sistem
1. Klik menu Data <i>Stopword</i>	2. Menampilkan halaman Data <i>Stopword</i> yang berisi: <ul style="list-style-type: none"> • No • Kata • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action Button Tambah Stopword</i>
3. Klik Tambah <i>Stopword</i>	4. Menampilkan form tambah data <i>stopword</i> yang berisi <i>text field</i> : <ul style="list-style-type: none"> • <i>Stopword</i> • <i>Action button Save</i> • <i>Action button Close</i>
5. Mengisi form	
5. Klik <i>button Save</i>	6. Menampilkan halaman data <i>stopword</i>
Skenario Alternatif : Data tidak lengkap	
Aksi Aktor	Aksi Sistem
5. Klik <i>button Save</i>	6. Menampilkan <i>alert</i> “ <i>Please fill out this field</i> ”
	7. Menampilkan halaman data <i>stopword</i>
Skenario Alternatif : Batal tambah data	
Aksi Aktor	Aksi Sistem
5. Klik <i>button Close</i>	6. Menampilkan halaman data <i>stopword</i>
Skenario Normal : Lihat data <i>stopword</i>	
Aksi Aktor	Aksi Sistem
1. Klik menu Data <i>Stopword</i>	

	2. Menampilkan halaman menu Data <i>Stopword</i> yang berisi tabel: <ul style="list-style-type: none"> • No • Kata • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action Button Tambah Stopword</i>
Skenario Normal : Update data <i>stopword</i>	
Aksi Aktor	Aksi Sistem
1. Klik menu Data <i>Stopword</i>	2. Menampilkan halaman menu Data <i>Stopword</i> yang berisi tabel: <ul style="list-style-type: none"> • No • Kata • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action Button Tambah Stopword</i>
2. Klik <i>button Update</i>	3. Menampilkan form Update Data <i>Stopword</i> yang dipilih: <ul style="list-style-type: none"> • <i>Stopword</i> • <i>Action button Save</i> • <i>Action button Close</i>
4. Mengubah data <i>stopword</i>	
5. Klik <i>button Save</i>	6. Menampilkan halaman data <i>stopword</i>
Skenario Alternatif : Data tidak lengkap	
Aksi Aktor	Aksi Sistem
5. Klik <i>button Save</i>	6. Menampilkan <i>alert</i> "Please fill out this field"
	7. Menampilkan form tambah data <i>stopword</i>
Skenario Alternatif : Batal update data	
Aksi Aktor	Aksi Sistem
5. Klik <i>button Close</i>	6. Menampilkan halaman data <i>stopword</i>
Skenario Normal : Hapus data <i>stopword</i>	
Aksi Aktor	Aksi Sistem
1. Klik menu Data <i>stopword</i>	2. Menampilkan halaman menu Data <i>Stopword</i> yang berisi : <ul style="list-style-type: none"> • No • Kata • <i>Action button Update</i> • <i>Action button Delete</i> • <i>Action Button Tambah Stopword</i>

3. Klik <i>button Delete</i> pada data yang diinginkan	4. Menampilkan halaman data <i>stopword</i>
--------------------------------------------------------	---------------------------------------------

A.13 Skenario *Logout*

Tabel A. 13 Skenario *Logout*

No. Usecase	UCS 14
Nama Usecase	<i>Logout</i>
Aktor	Admin
PreKondisi	Admin memilih tombol <i>Logout</i> (Keluar)
PraKondisi	Admin berhasil Keluar
Flow Events	
Skenario Normal : <i>Logout</i>	
Aksi Aktor	Aksi Sistem
1. Klik tombol <i>Logout</i>	2. Menampilkan halaman awal sistem

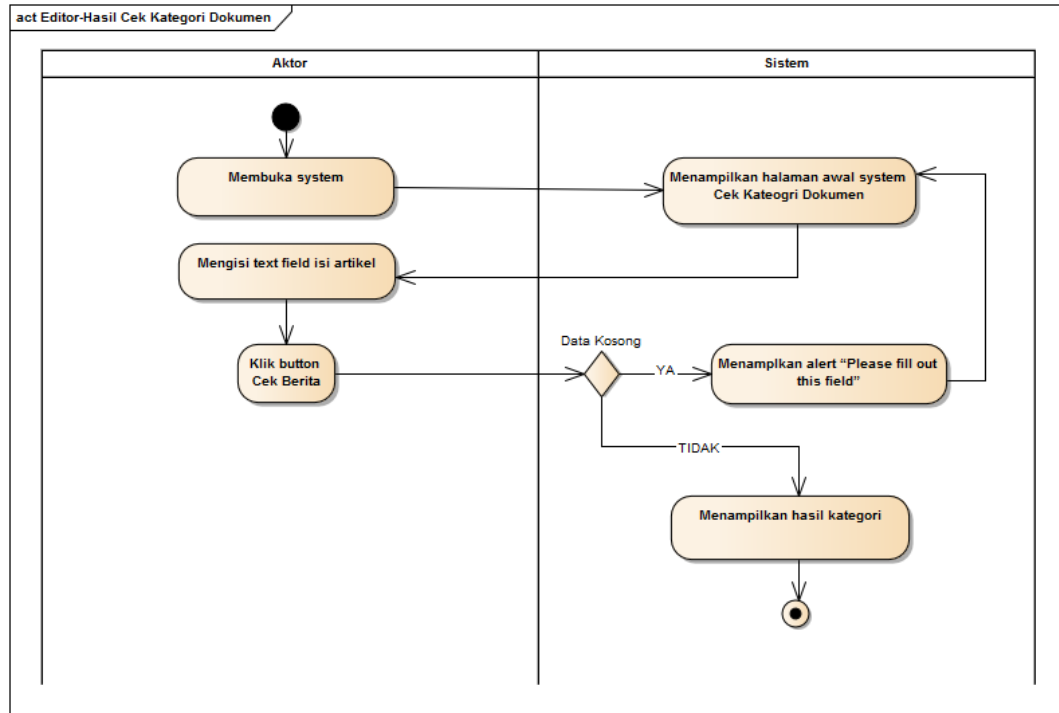
A.14 Skenario Melihat Berita Politik

Tabel A. 14 Skenario Melihat Bertia Politik

No. Usecase	UCS 15
Nama Usecase	Melihat Berita Politik
Aktor	<i>Guest</i> (Tamuh)
PreKondisi	<i>Guest</i> (Tamuh) memilih Data Berita Politik
PraKondisi	<i>Guest</i> (Tamuh) berhasil melihat berita politik
Flow Events	
Skenario : Melihat hasil klasifikasi data <i>testing</i>	
Aksi Aktor	Aksi Sistem
1. Membuka halaman awal sistem	2. Menampilkan halaman awal sistem Cek Kategori Dokumen: <ul style="list-style-type: none"> • <i>Textbox</i> (Artikel) • <i>Button</i> (Cek Berita) • <i>Home</i> • Tentang • Admin • Data Berita Politik
3. Klik <i>button</i> Data Berita Politik	4. Menampilkan list berita politik
5. Klik <i>button</i> Baca	6. Menamapilkan isi artikel

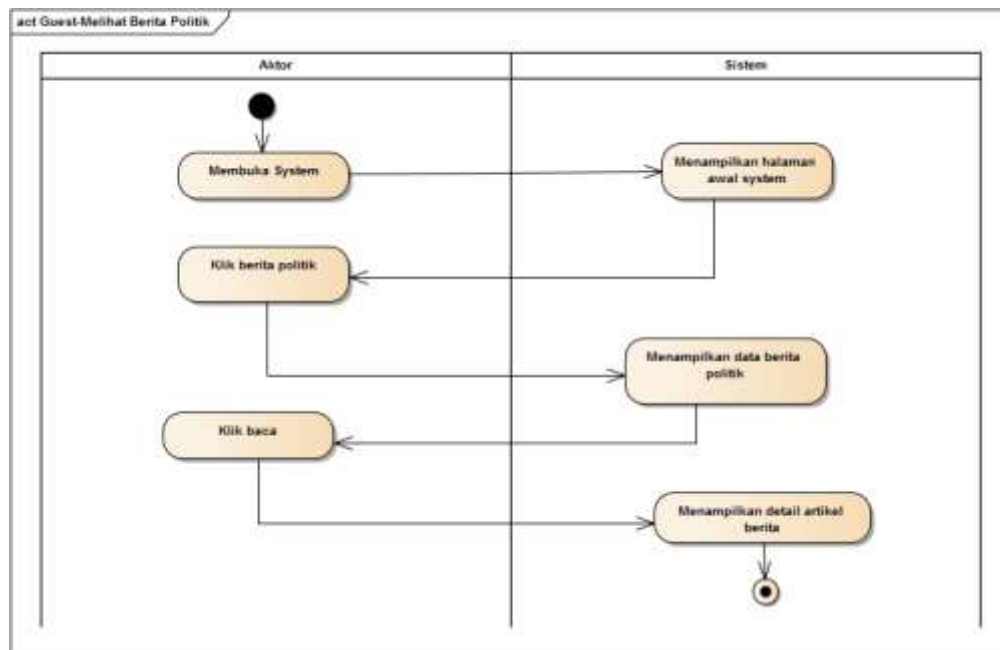
B. Activity Diagram

B.1 Activity Diagram Hasil Cek Kategori Dokumen



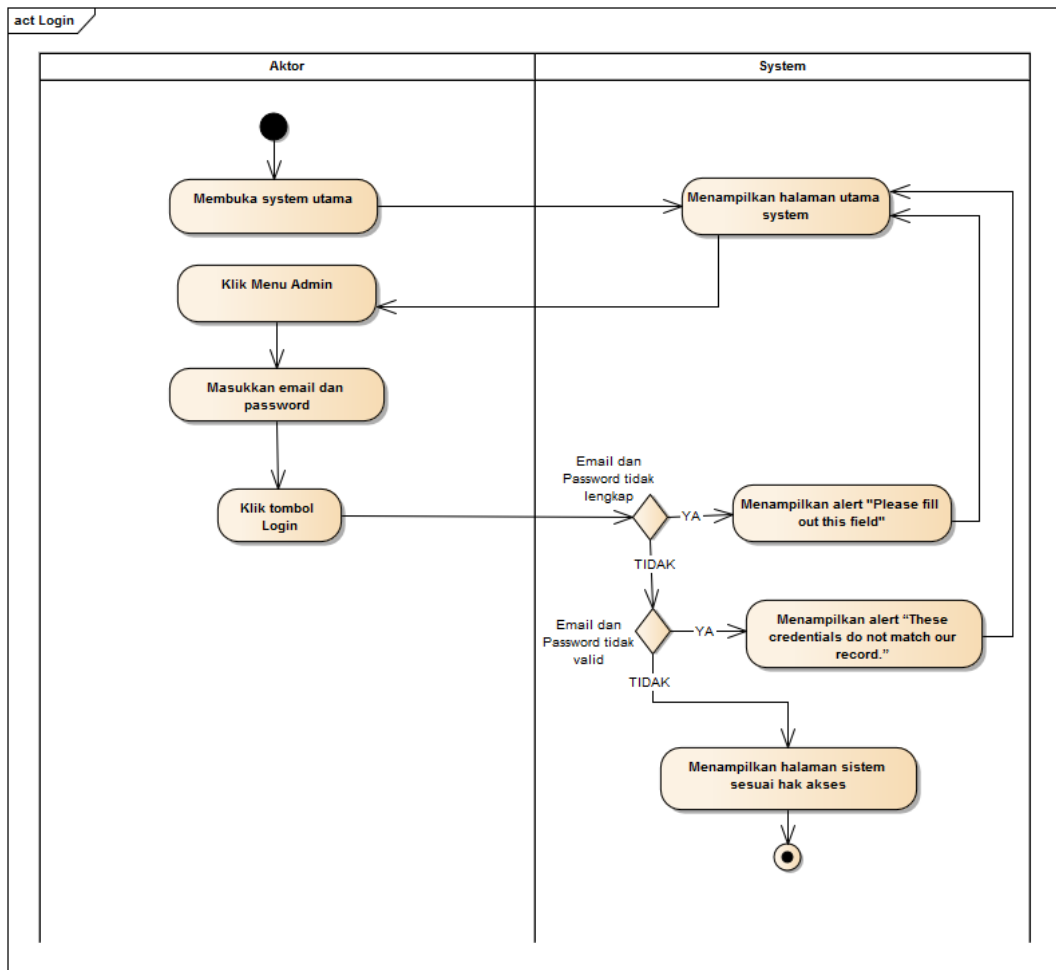
Gambar B. 1 Activity Diagram Hasil Cek Kategori Dokumen

B.2 Activity Diagram Melihat Berita Politik



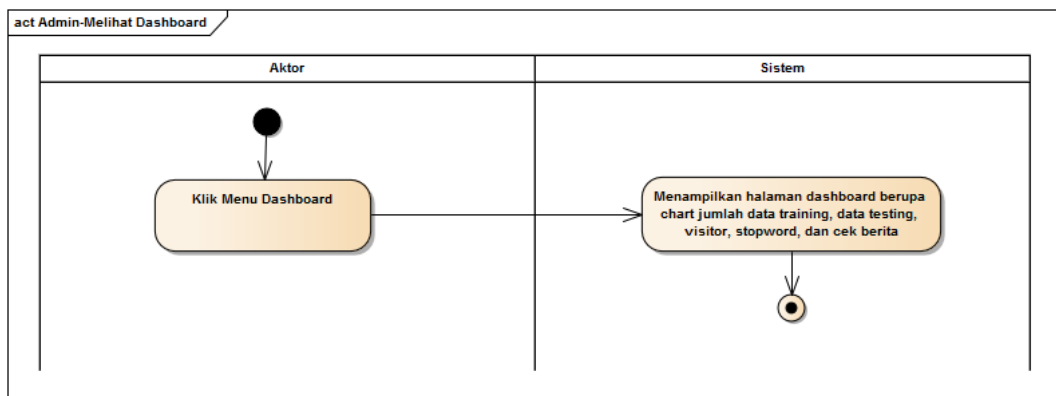
Gambar B. 2 Activity Diagram Melihat Berita Politik

B.3 Activity Diagram Login



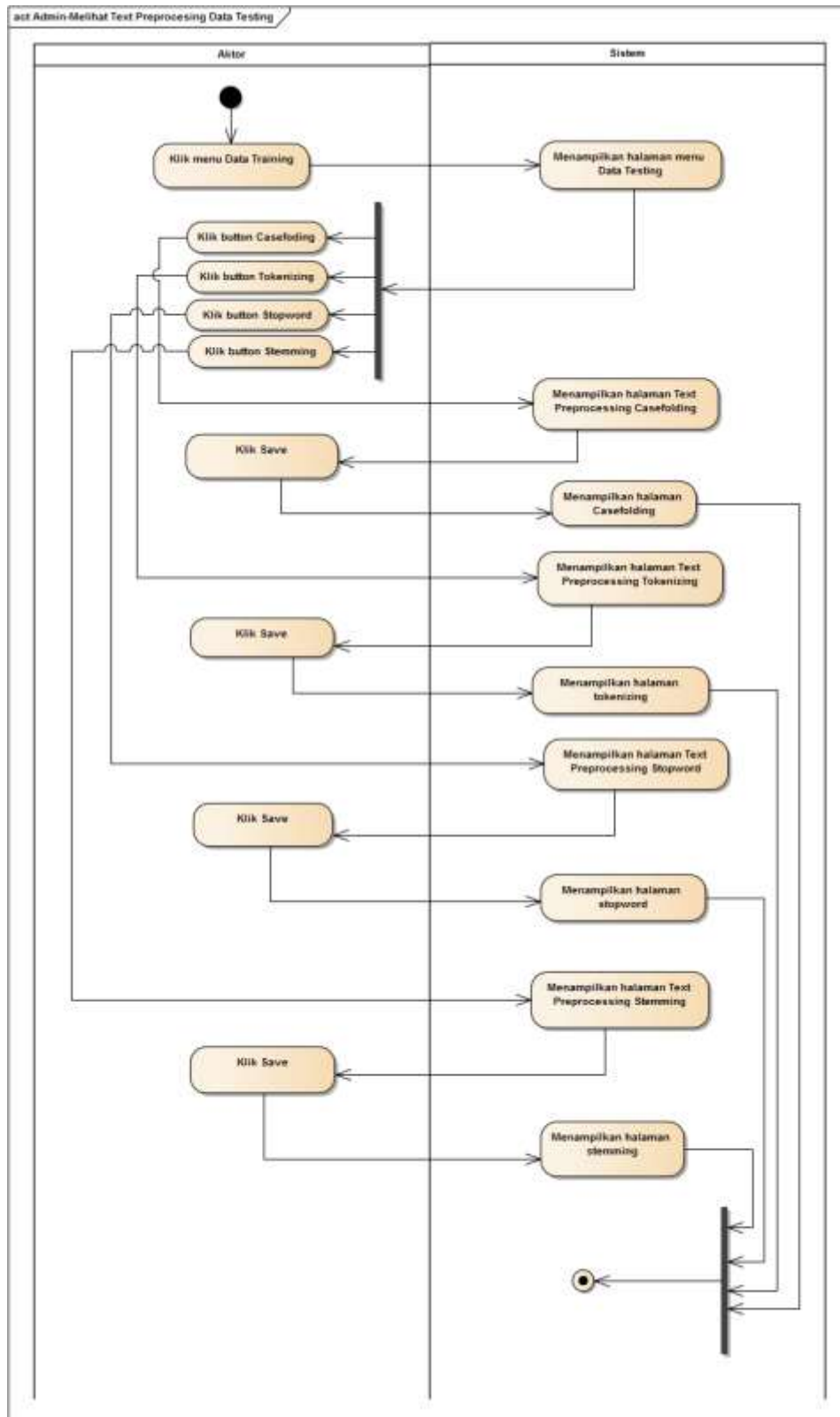
Gambar B. 3 Activity Diagram Login

B.4 Activity Diagram Melihat Dashboard



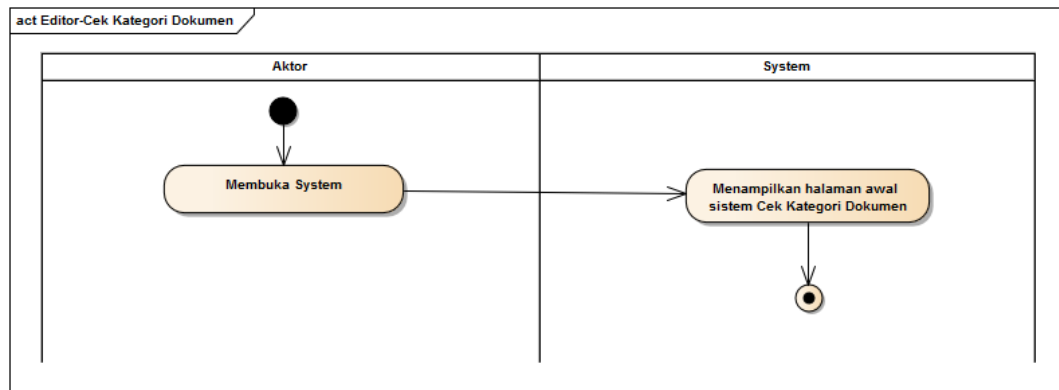
Gambar B. 4 Activity Diagram Melihat Dashboard

B.5 Activity Diagram Melihat Text Preprocessing Data Training



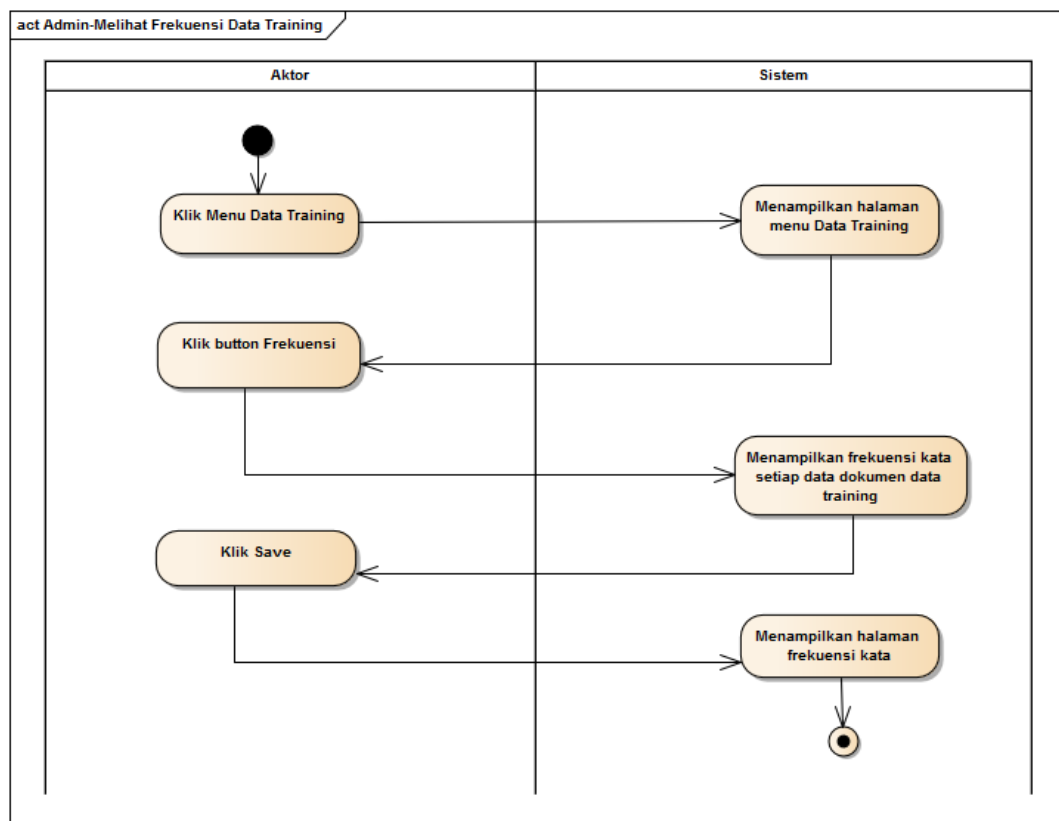
Gambar B. 5 Activity Diagram Melihat Text Preprocessing Data Training

B.6 Activity Diagram Cek Kategori Dokumen



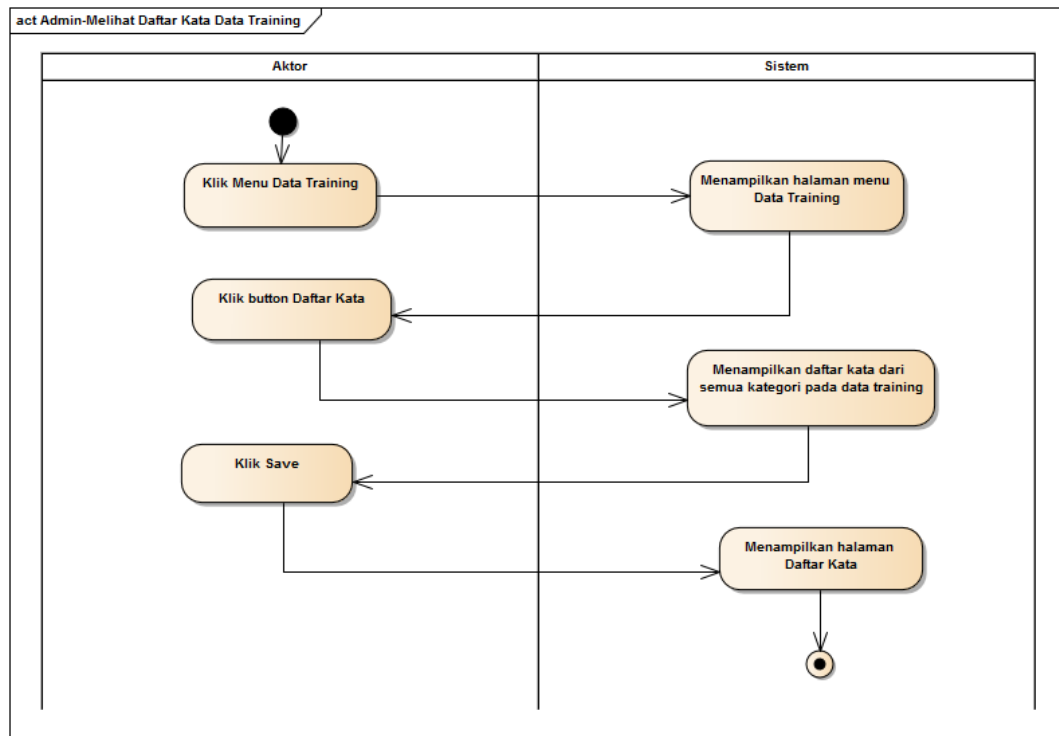
Gambar B. 6 Activity Diagram Cek Kategori Dokumen

B.7 Activity Diagram Melihat Frekuensi Kata Data Training



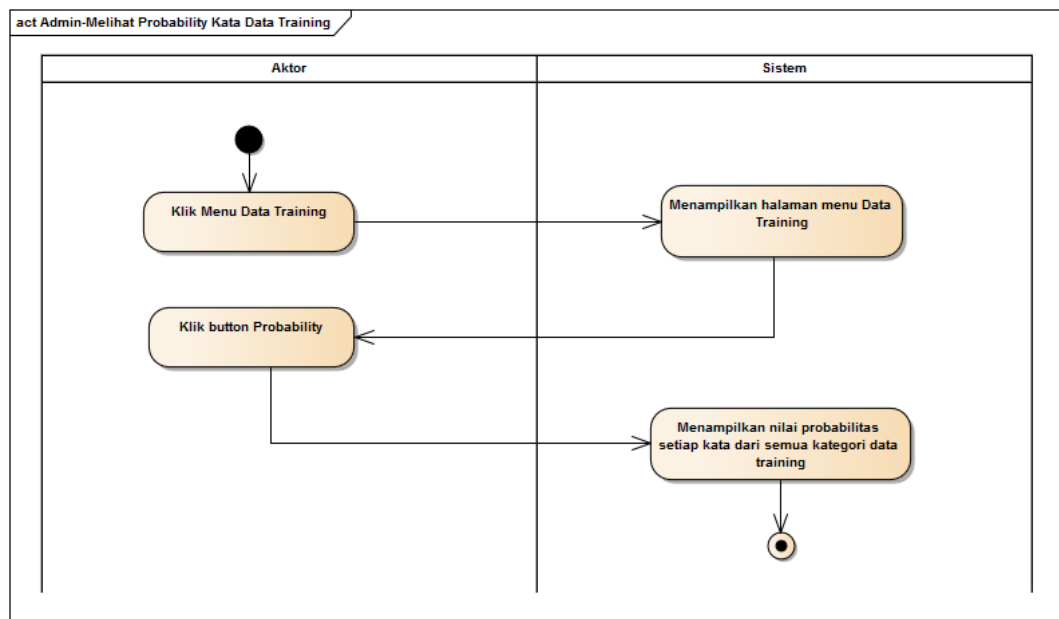
Gambar B. 7 Activity Diagram Melihat Frekuensi Kata Data Training

B.8 Activity Diagram Melihat Daftar Kata Data Training



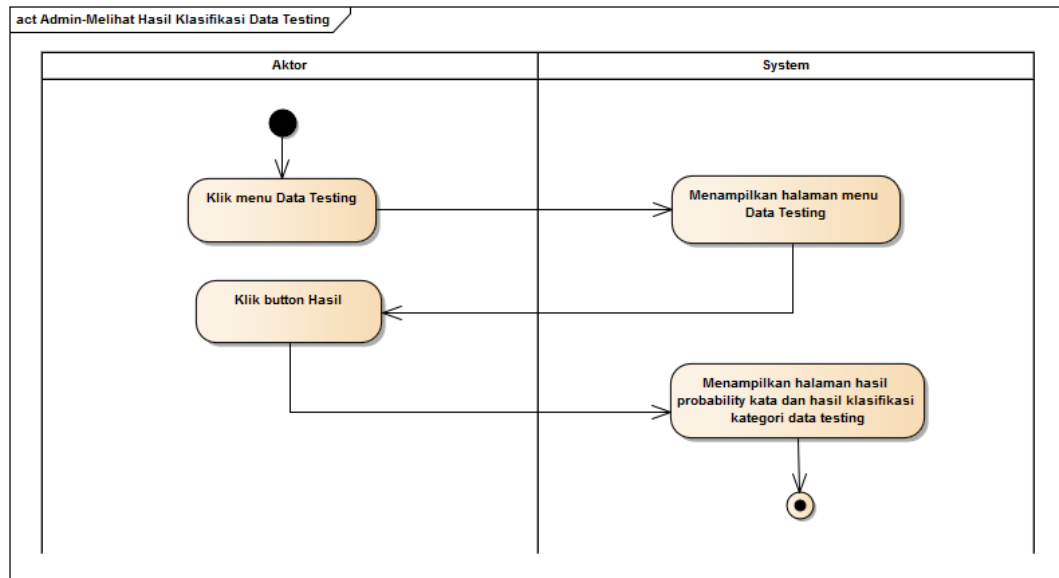
Gambar B. 8 Activity Diagram Melihat Daftar Kata Data Training

B.9 Activity Diagram Melihat Probability Kata Data Training



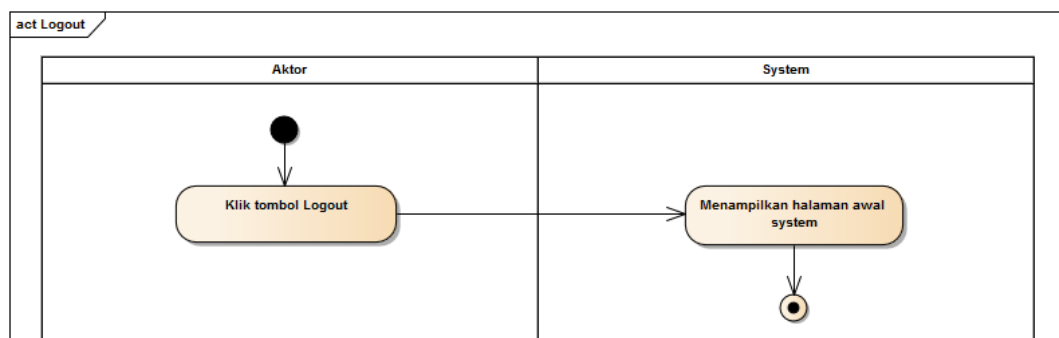
Gambar B. 9 Activity Diagram Melihat Probability Kata Data Training

B.10 Activity Diagram Melihat Hasil Klasifikasi Data Testing



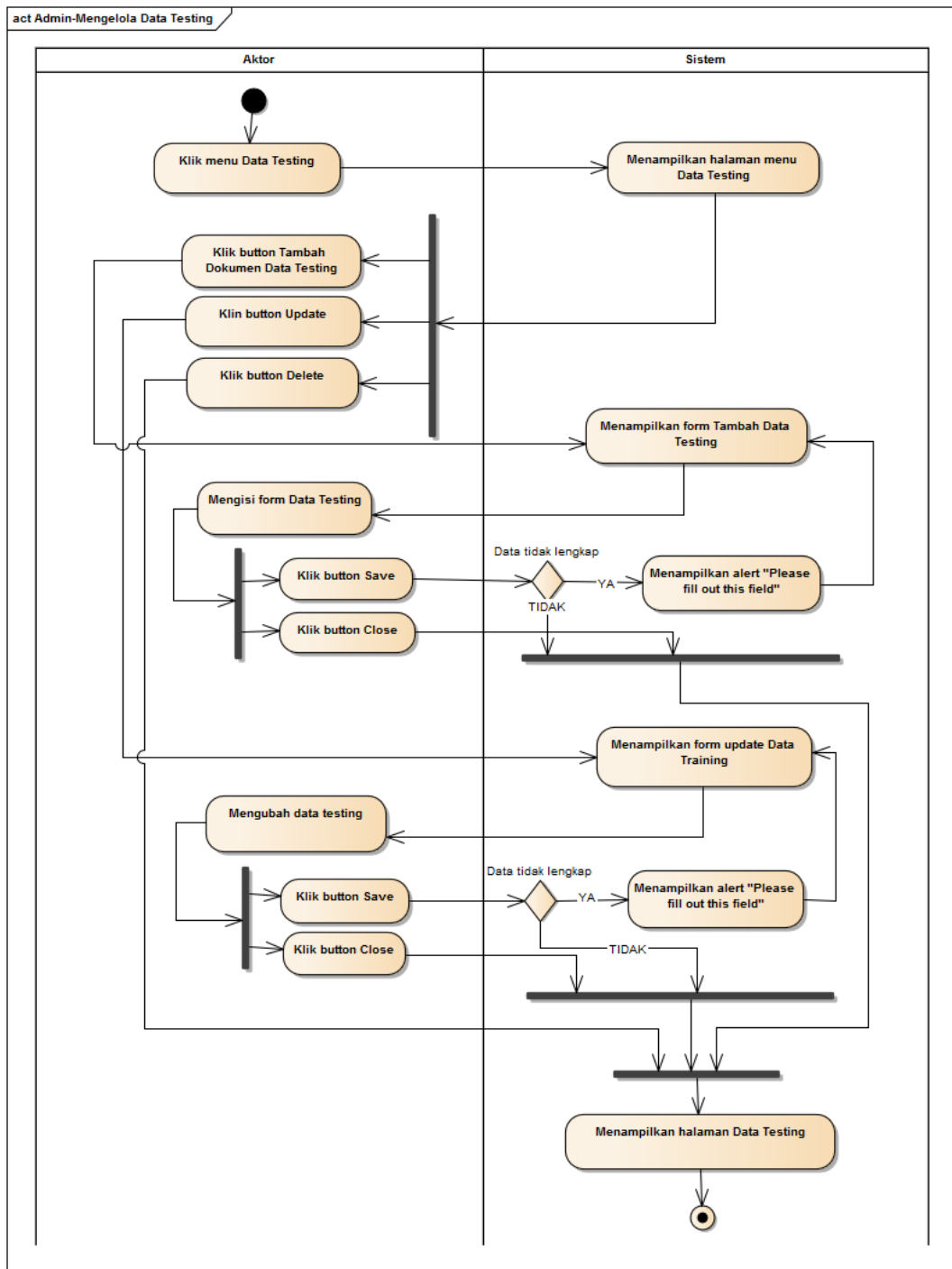
Gambar B. 10 Activity Diagram Melihat Hasil Klasifikasi Data Testing

B.11 Activity Diagram Logout



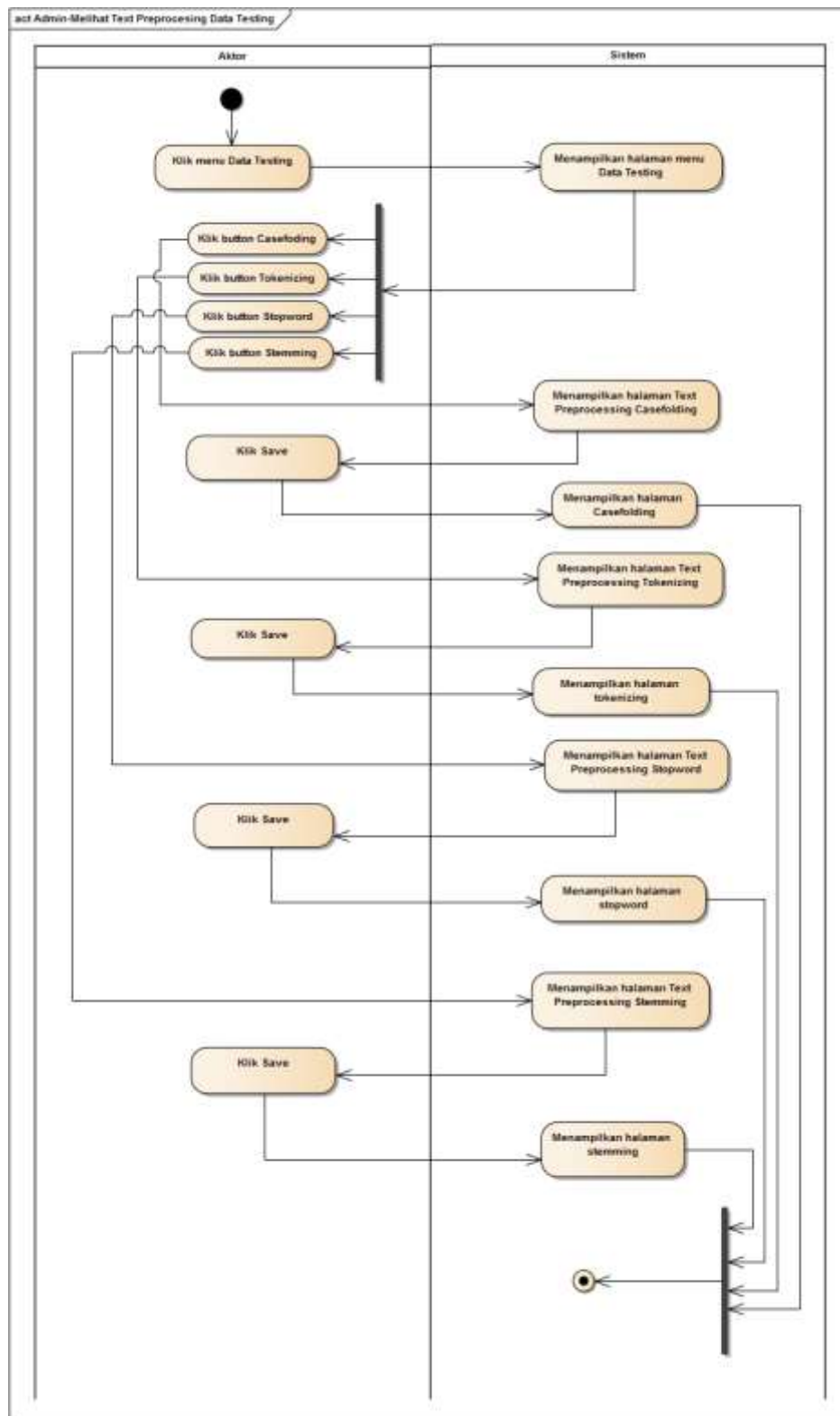
Gambar B. 11 Activity Diagram Logout

B.12 Activity Diagram Mengelola Data Testing



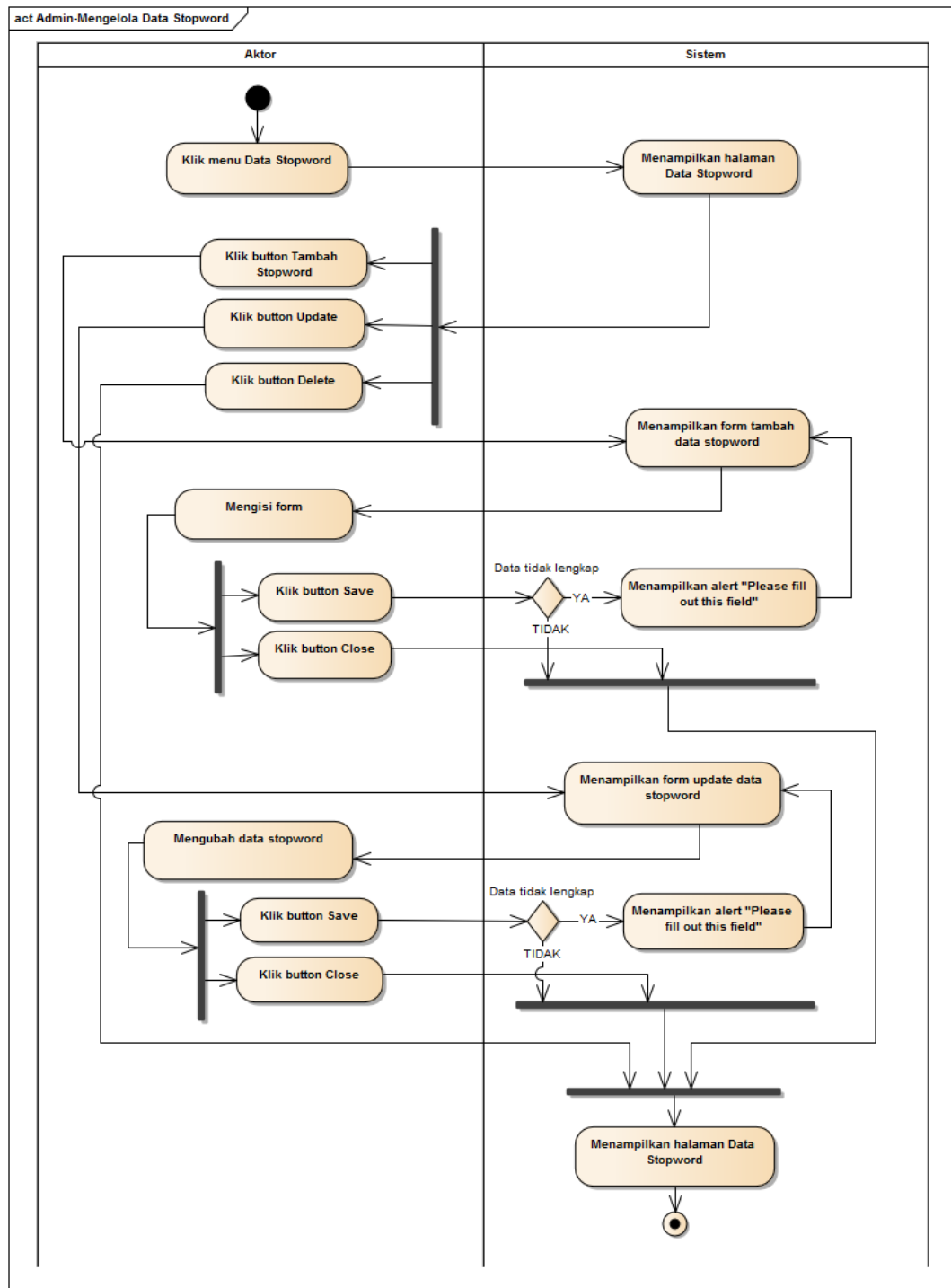
Gambar B. 12 Activity Diagram Mengelola Data Testing

B.13 Activity Diagram Melihat Text Preprocessing Data Testing



Gambar B. 13 Activity Diagram Melihat Text Preprocessing Data Testing

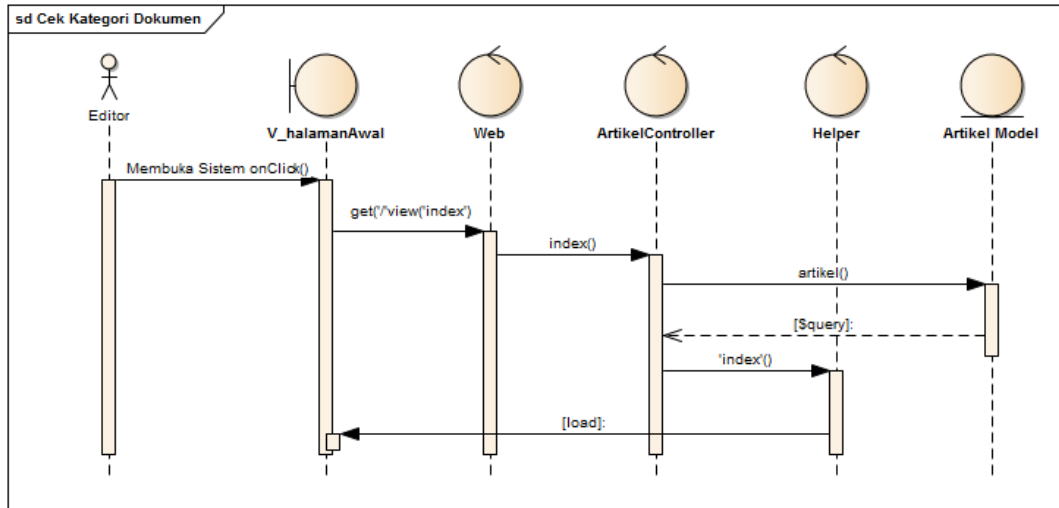
B.14 Activity Diagram Mengelola Data Stopword



Gambar B. 14 Activity Diagram Mengelola Data Stopword

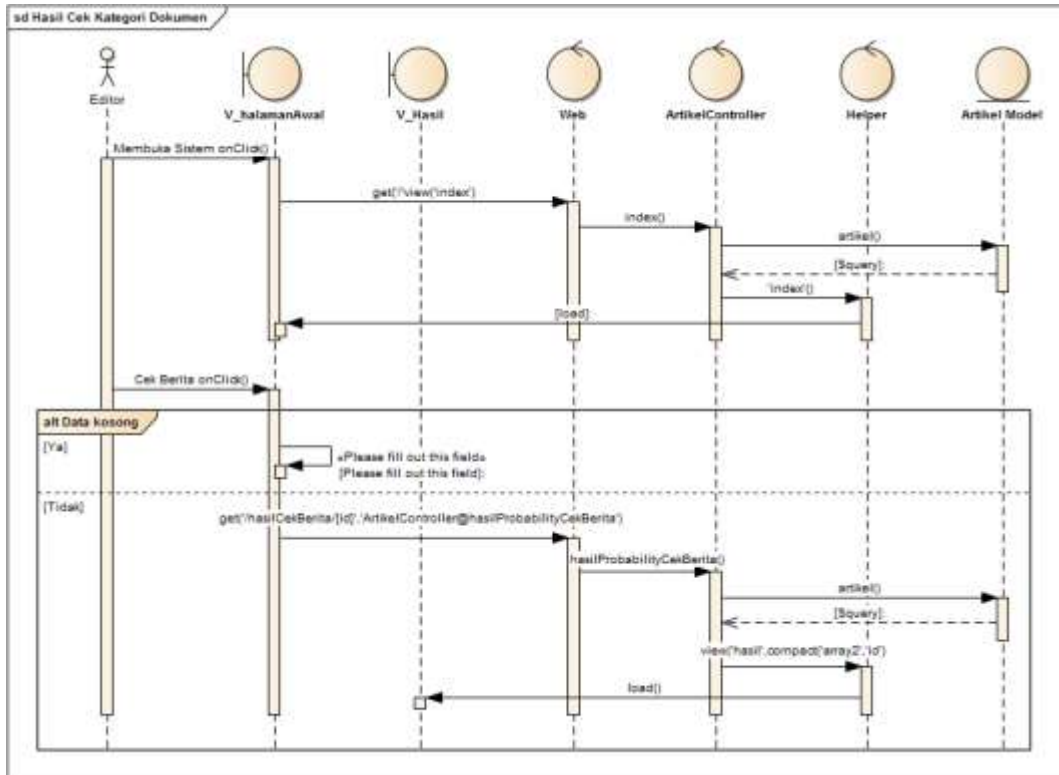
C. Sequence Diagram

C.1 Sequence Diagram Cek Kategori Dokumen



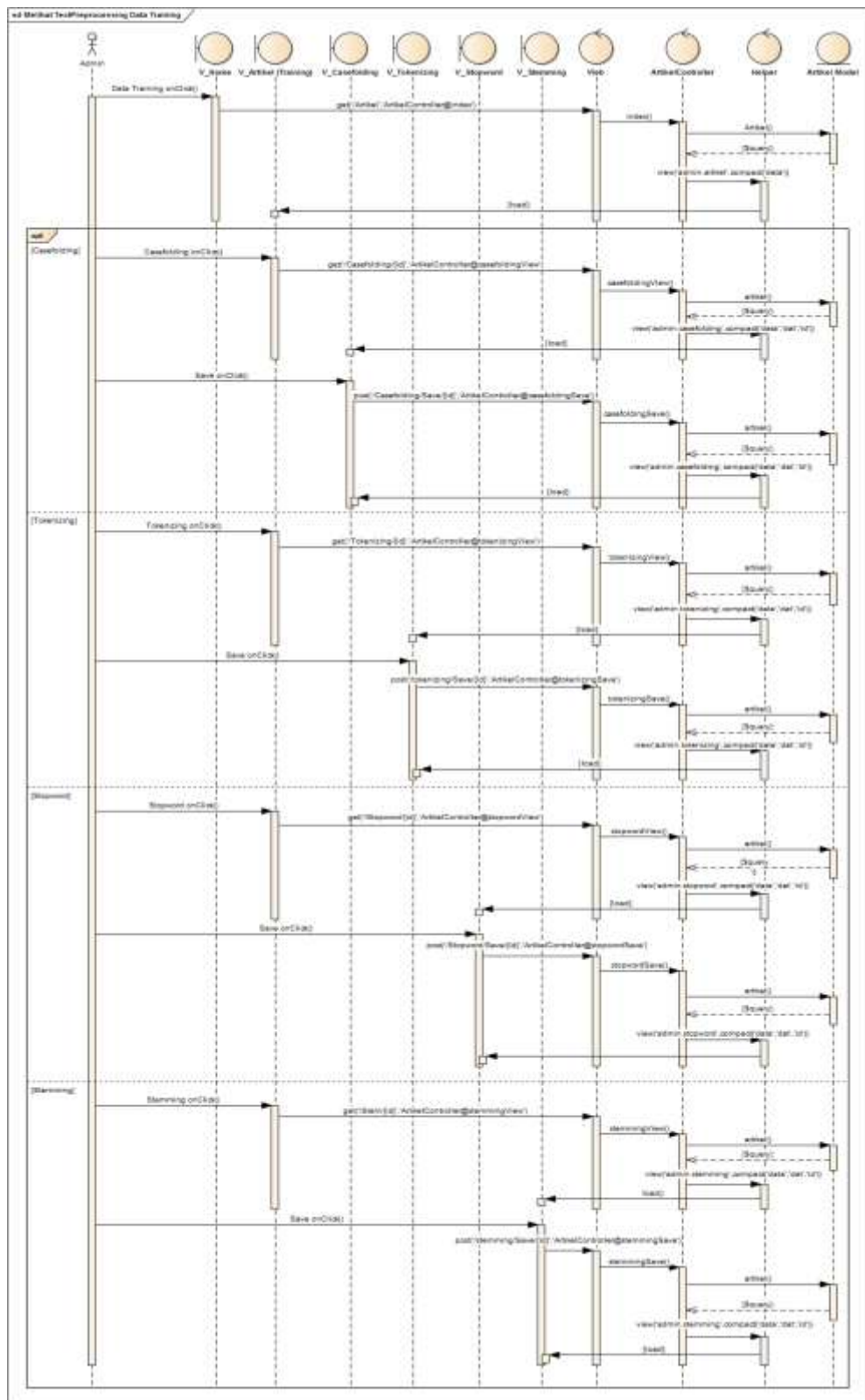
Gambar C. 1 Sequence Diagram Cek Kategori Dokumen

C.2 Sequence Diagram Hasil Cek Kategori Dokumen



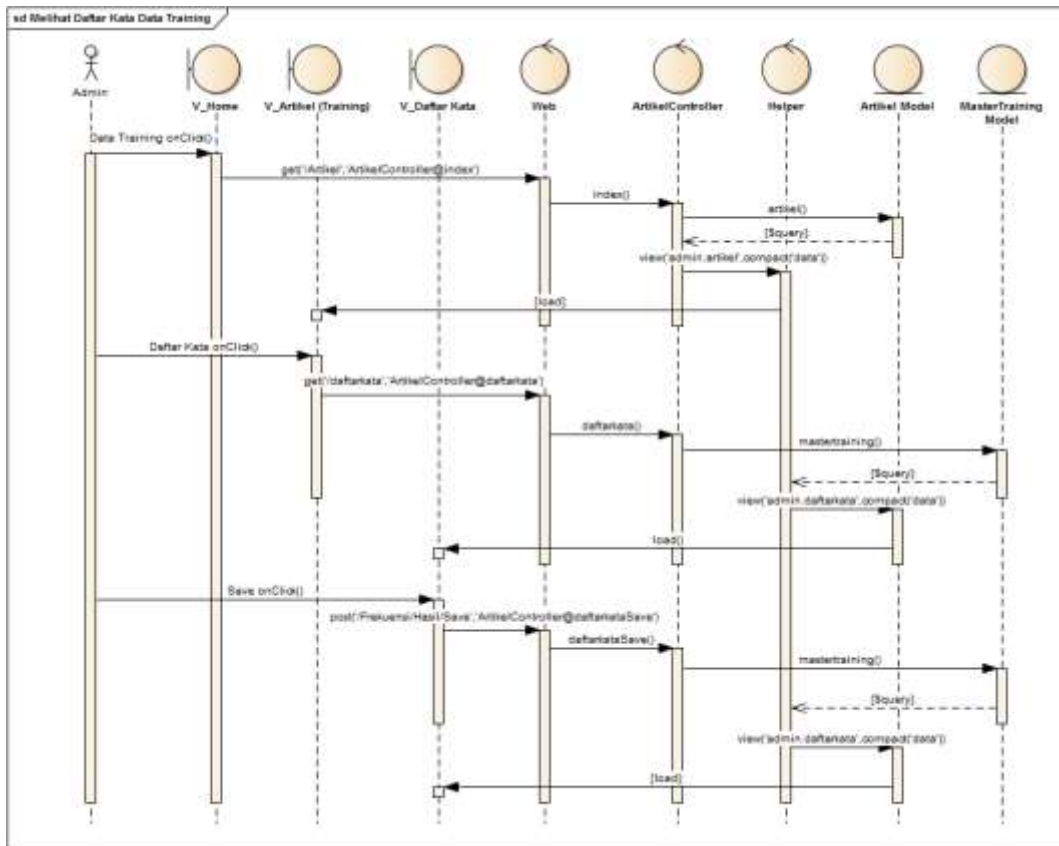
Gambar C. 2 Sequence Diagram Hasil Cek Kategori Dokumen

C.3 Sequence Diagram Melihat Text Preprocessing Data Training



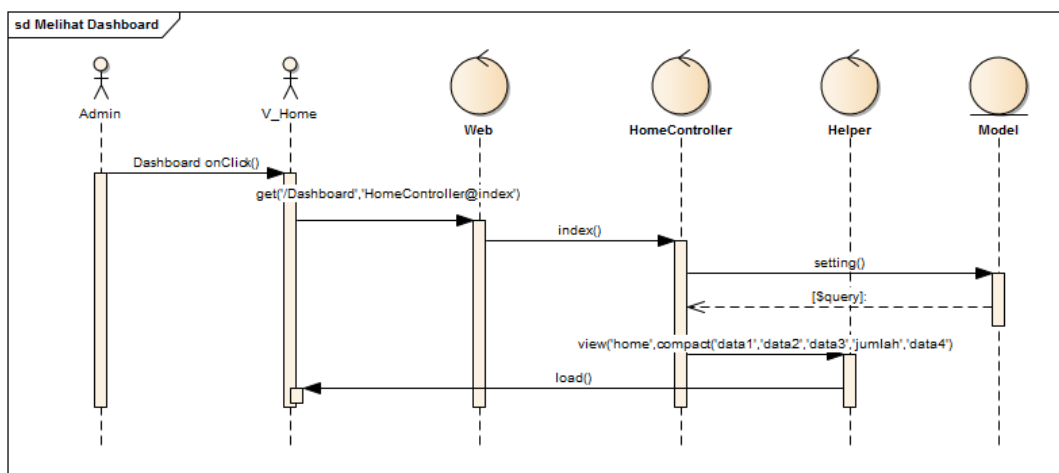
Gambar C. 3 Sequence Diagram Melihat Text Preprocessing Data Training

C.4 Sequence Diagram Melihat Daftar Kata Data Training



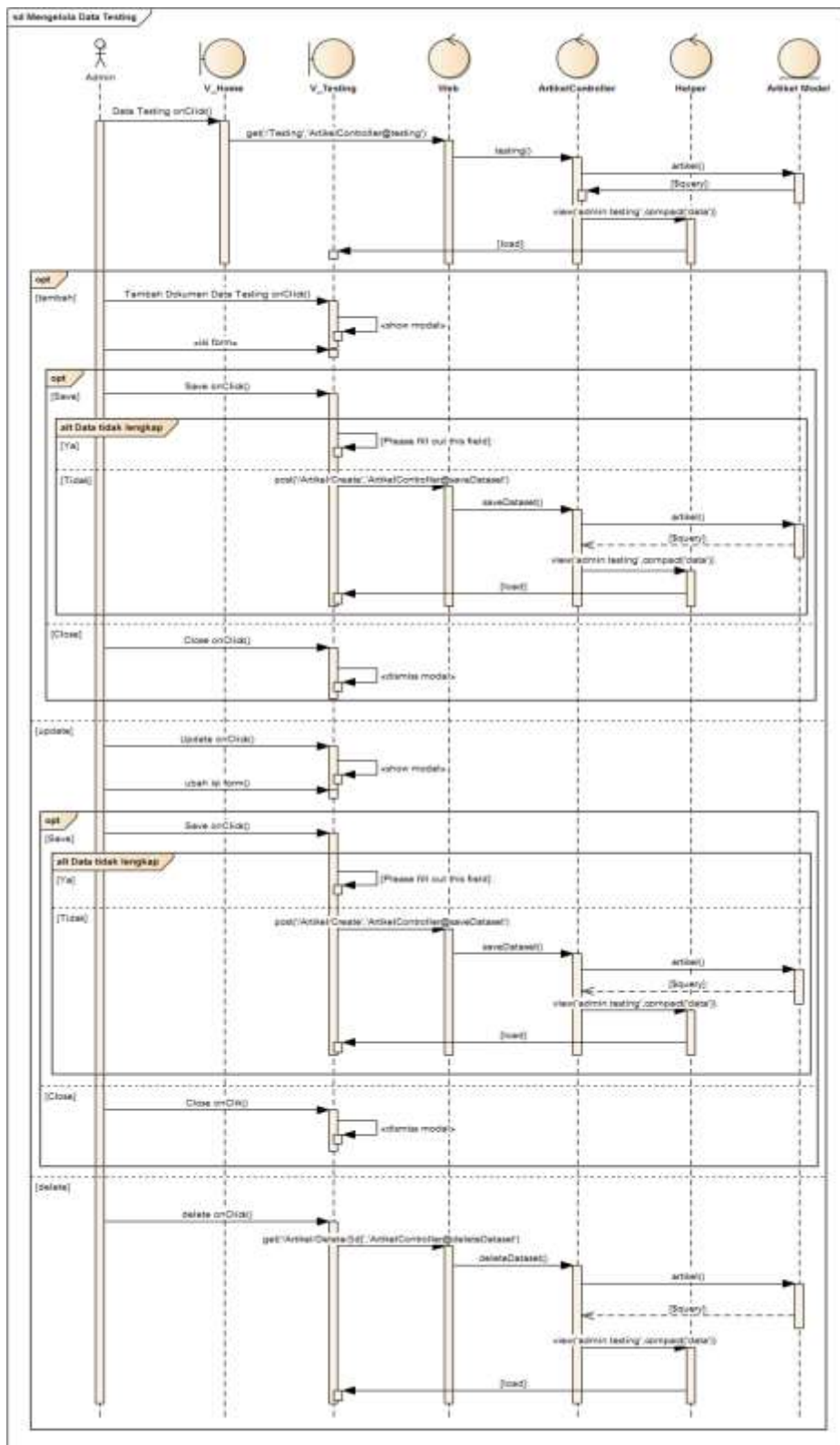
Gambar C. 4 Sequence Diagram Melihat Daftar Kata Data Training

C.5 Sequence Diagram Melihat Dashboard



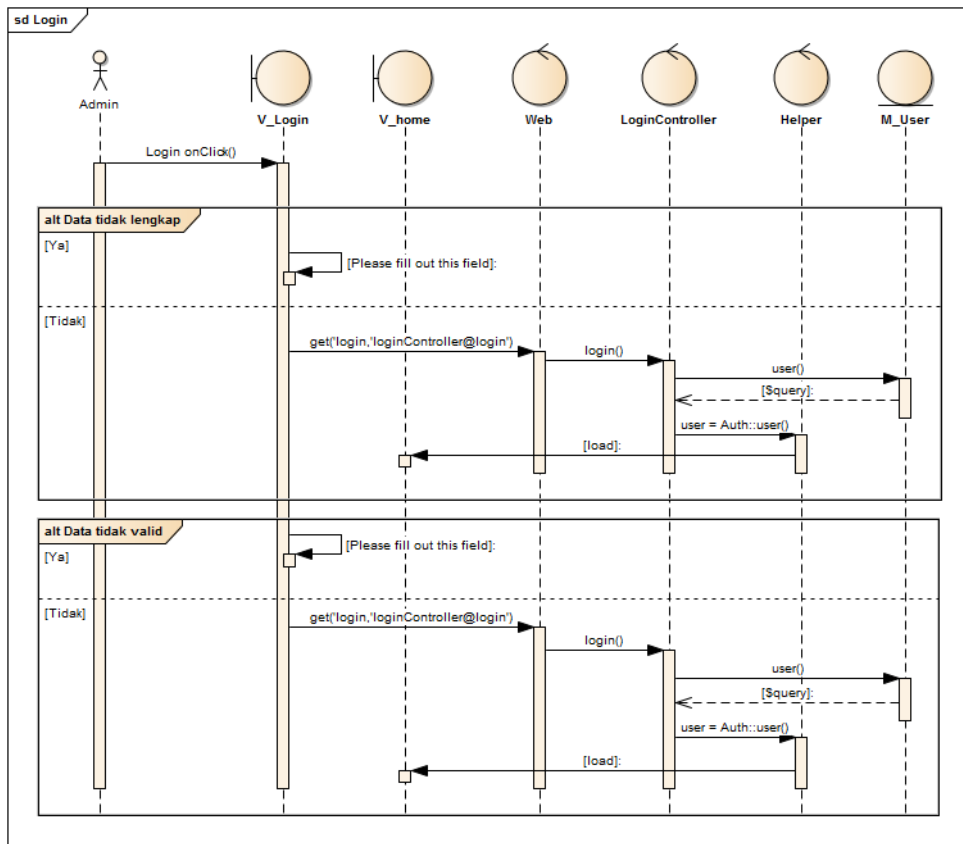
Gambar C. 5 Sequence Diagram Melihat Dashboard

C.6 Sequence Diagram Mengelola Data Testing



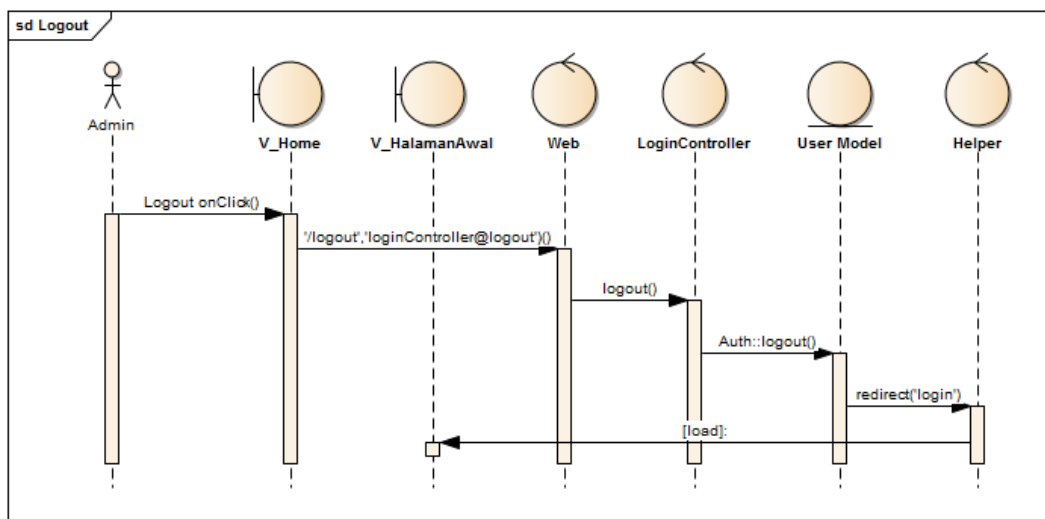
Gambar C. 6 Sequence Diagram Mengelola Data Testing

C.7 Sequence Diagram Login



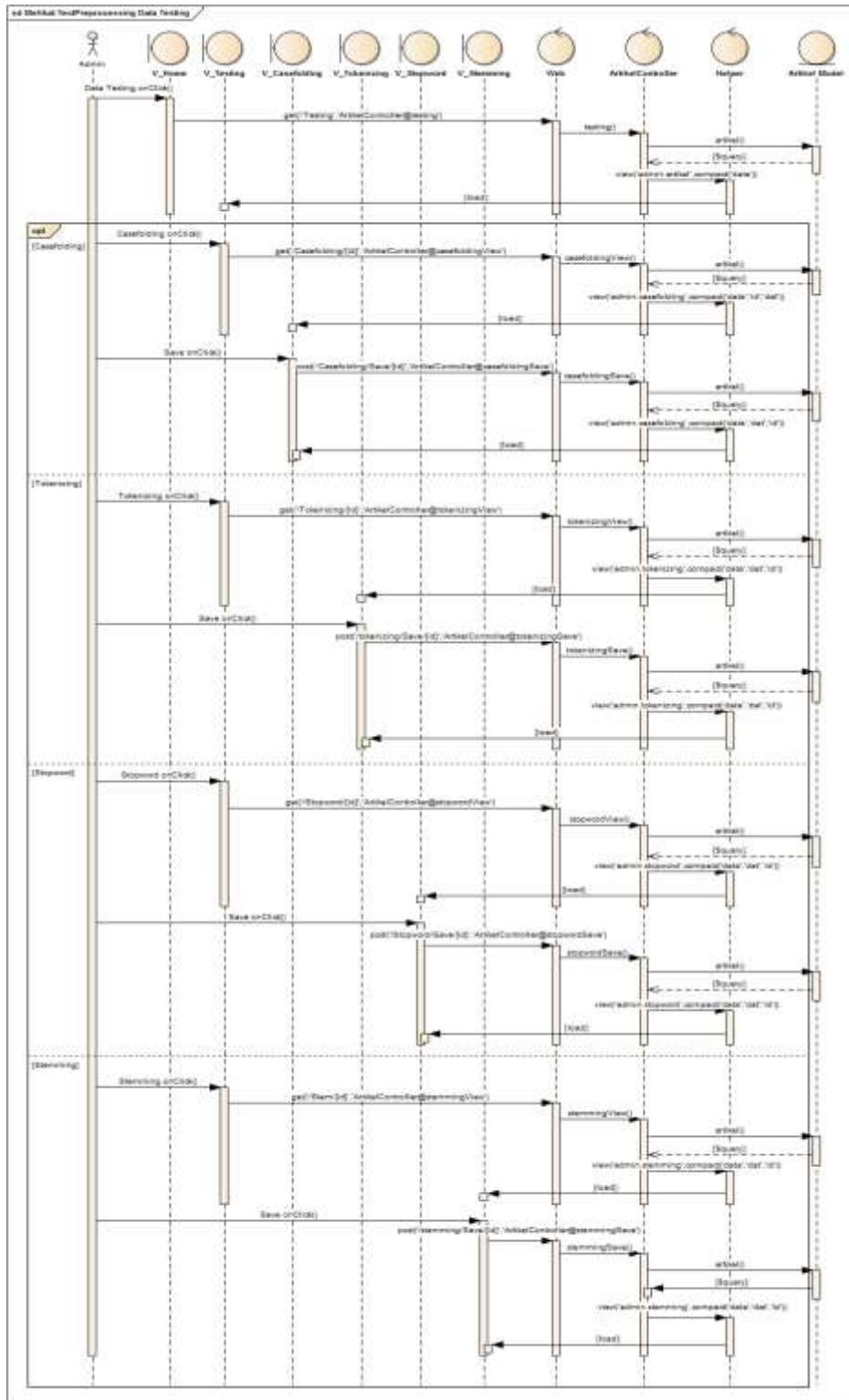
Gambar C. 7 Sequence Diagram Login

C.8 Sequence Diagram Logout



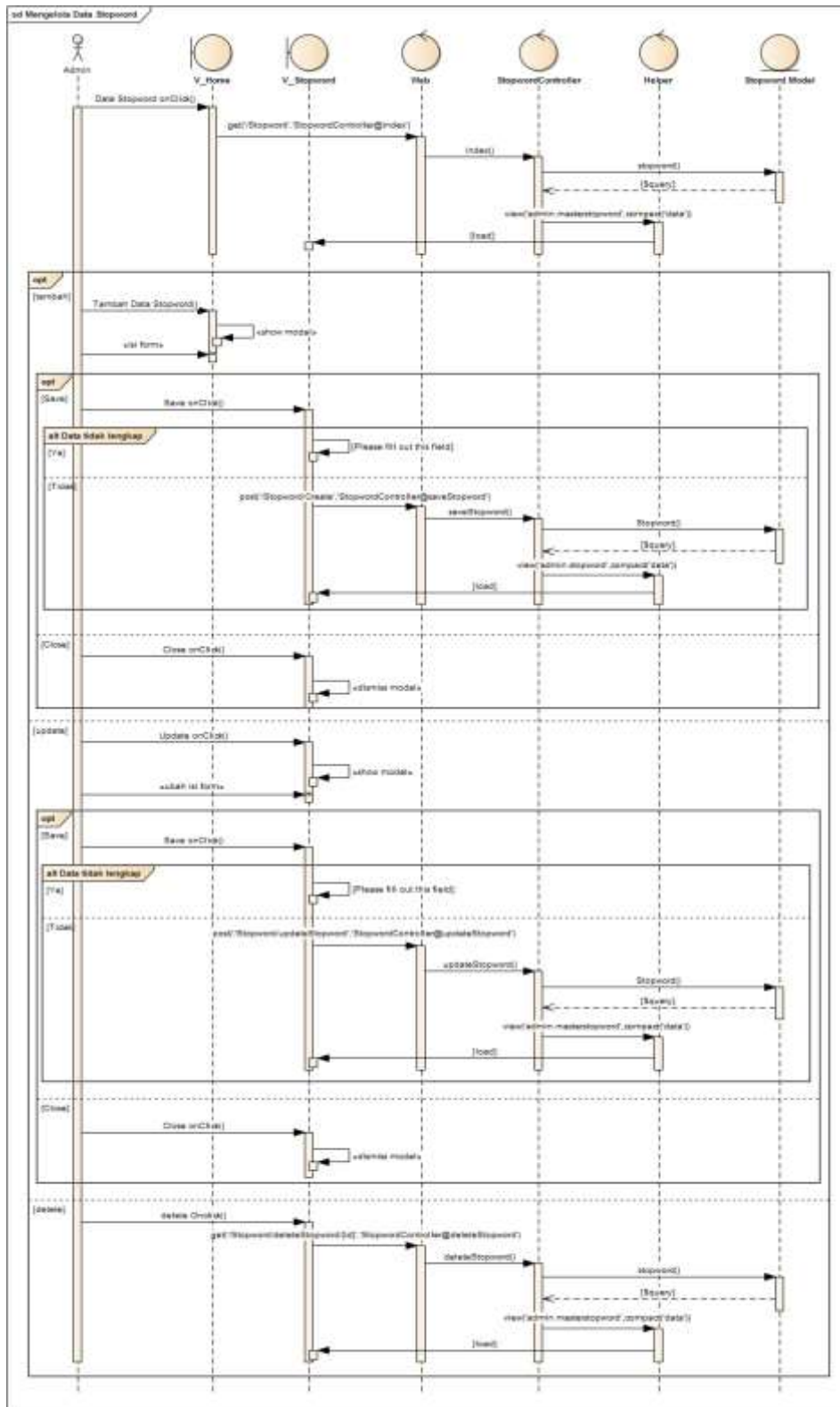
Gambar C. 8 Sequence Diagram Logout

C.9 Sequence Diagram Melihat Text Preprocessing Data Testing



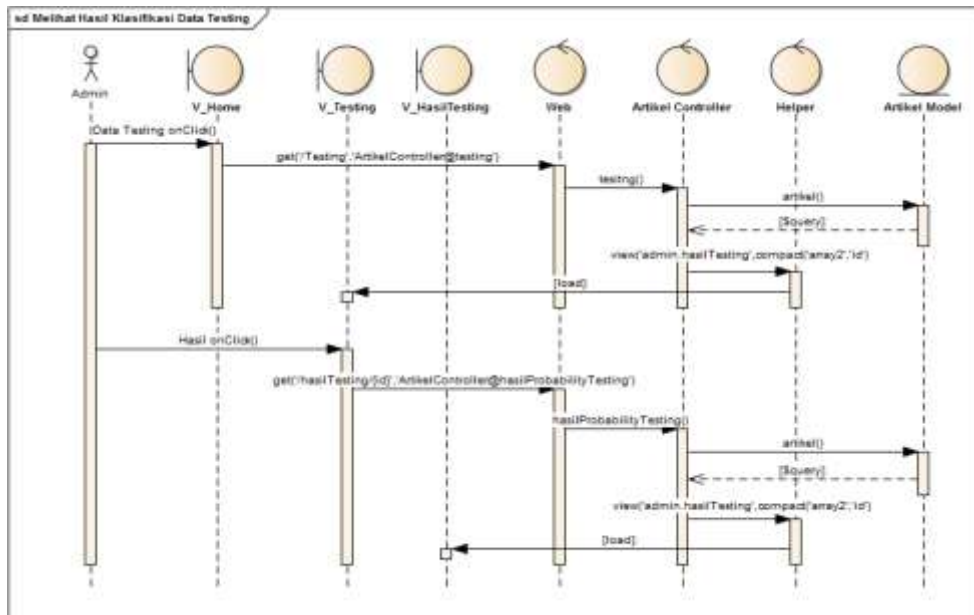
Gambar C. 9 Sequence Diagram Melihat Text Preprocessing Data Testing

C.10 Sequence Diagram Mengelola Data Stopword



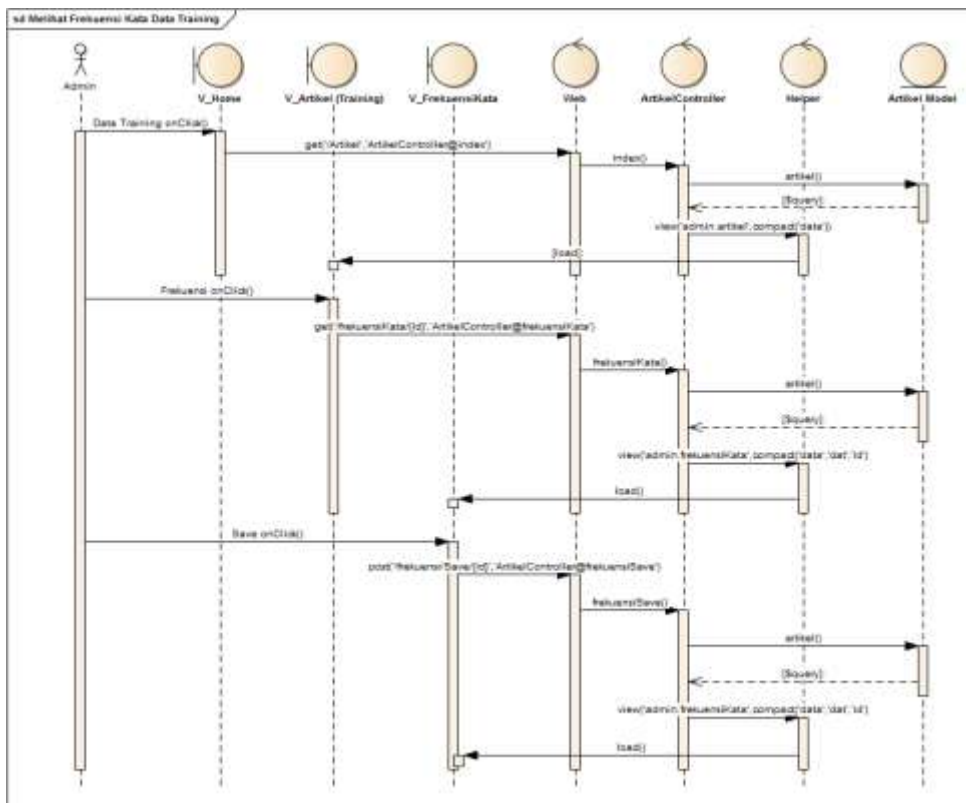
Gambar C. 10 Sequence Diagram Mengelola Data Stopword

C.11 Sequence Diagram Melihat Hasil Klasifikasi Data Testing



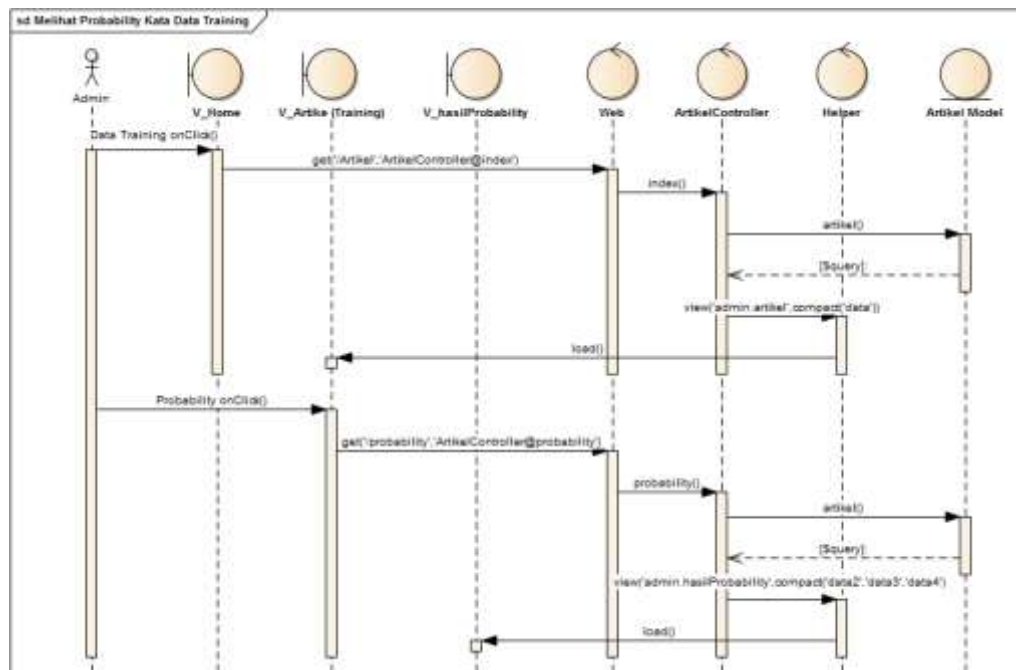
Gambar C. 11 Sequence Diagram Melihat Hasil Klasifikasi Data Testing

C.12 Sequence Diagram Melihat Frekuensi Kata Data Training



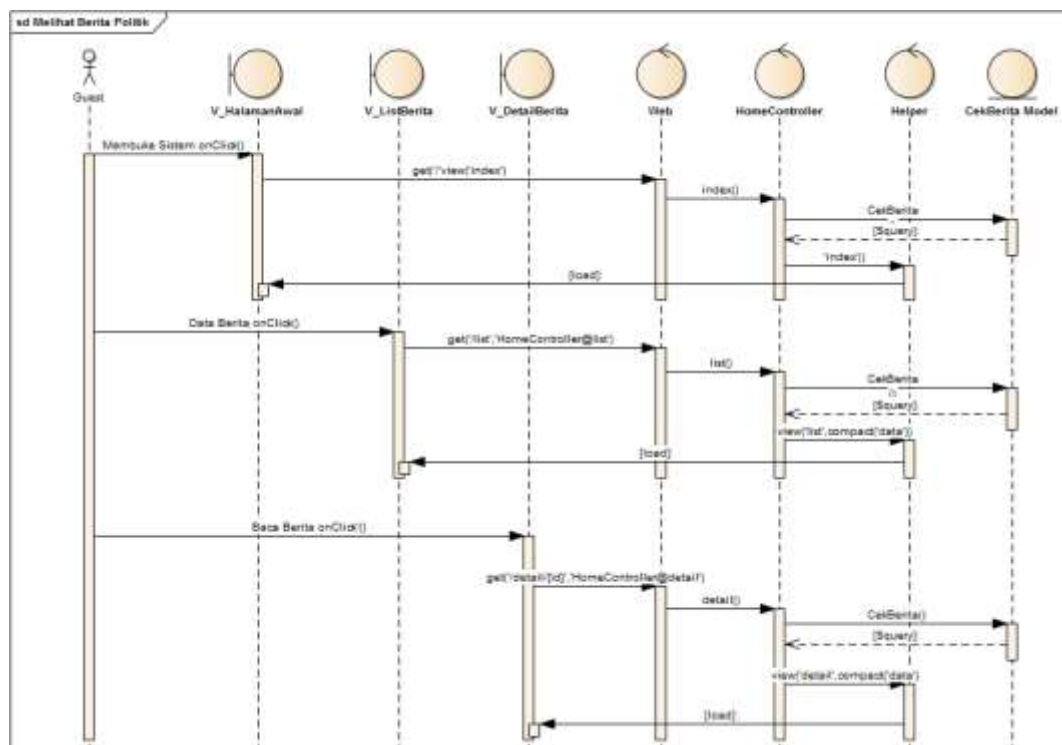
Gambar C. 12 Sequence Diagram Melihat Frekuensi Kata Data Training

C.13 Sequence Diagram Melihat Probability Kata Training



Gambar C. 13 Sequence Diagram Melihat Probability Kata Training

C.14 Sequence Diagram Melihat Berita Politik



Gambar C. 14 Sequence Diagram Melihat Berita Politik

D. Data Uji Perfarmasi

Tabel D. 1 Hasil Data Uji

No	Artikel	Kategori Aktual	Kategori Prediksi Sistem
1	Bawaslu Susun Indeks Kerawanan Pilkada 2018	Pilkada	Pilkada
2	Ridwan Kamil Berharap Pilkada Jabar 2018 Jangan Seperti DKI dan Pilpres 2014	Pilkada	Pilkada
3	Besok, Megawati Pimpin Rapat Konsolidasi Pemenangan Pilkada di Banten	Pilkada	Pilkada
4	Ini Nama-nama Calon yang Diusung Partai Demokrat di Banten	Pilkada	Pilkada
5	KPU Ungkap Tiga Faktor yang Bikin Pilkada 2018 Rawan Konflik	Pilkada	Pilkada
6	Ini Langkah Bawaslu Cegah Isu SARA di Pilkada 2018	Pilkada	Pilkada
7	Ulama Purwakarta Gelar Konvensi, Jaring Kandidat Cawabup Independen	Pilkada	Pilkada
8	I Wayan Koster dan Cok Oka Resmi Diusung PDIP di Pilgub Bali 2018	Pilkada	Pilkada
9	Jelang Pilkada Serentak 2018, Polda Jabar Waspada Peredaran Upa	Pilkada	Pilkada
10	Jelang Pilkada Serentak 2018, Peran Pemuda di Politik Makin Menguat	Pilkada	Pilkada
11	Maju di Pilkada, Cak Eko Impikan Kota Bekasi Bebas Macet dan Banjir	Pilkada	Pilkada
12	Tangkal Kampanye Hitam, Polda Jabar Tingkatkan Cyber Patrol	Pilkada	Pilkada
13	MNC Group Dukung Penuh Pilkada 2018 dan Pemilu 2019	Pilkada	Pilkada
14	Sarjan Dinilai Berpeluang di Pilkada Sumsel	Pilkada	Pilkada
15	11 November, PDI-P Umumkan Sejumlah Calon untuk Pilkada 2018	Pilkada	Pilkada
16	Pilkada Serentak 2018, Jenis Konflik Terbanyak Ada di Jawa Timur dan Papua	Pilkada	Pilkada
17	Demokrat: JR Saragih-Mumtaz Rais Disiapkan Berpasangan di Pilkada Sumut	Pilkada	Pilkada
18	Belum Terlambat Bagi Susi Pudjiastuti untuk Maju Pilkada Jabar	Pilkada	Pilkada
19	Peneliti Senior LIPI: Pilkada DKI Jakarta, Pilkada yang Tidak Sehat	Pilkada	Pilkada
20	Pilkada Jatim, Ketum PAN Bakal Temui Soekarwo	Pilkada	Pilkada
21	Teten Tepis Reshuffle Jilid III untuk Bentuk Timses Jokowi di 2019	Reshuffle Kabinet	Reshuffle Kabinet
22	Reshuffle Kabinet: Khofifah Diganti, Airlangga Tetap Menperin?	Reshuffle Kabinet	Reshuffle Kabinet
23	Soal Isu Reshuffle Jilid III, PAN Ikut Arahan 'Master Chef' Jokowi	Reshuffle Kabinet	Reshuffle Kabinet

No	Artikel	Kategori Aktual	Kategori Prediksi Sistem
24	Pengamat Sebut Reshuffle Kabinet Akan Evaluasi Bidang Hukum	Reshuffle Kabinet	Reshuffle Kabinet
25	Menteri Asal PKB Mengaku Terima jika Kena Reshuffle Kabinet	Reshuffle Kabinet	Reshuffle Kabinet
26	Jokowi Tegaskan Tak Ada Reshuffle Kabinet dalam Waktu Dekat	Reshuffle Kabinet	Reshuffle Kabinet
27	Eks Jubir KPK Bicara Soal Isu Reshuffle Kabinet	Reshuffle Kabinet	Reshuffle Kabinet
28	PDIP Tak Yakin Pertemuan Jokowi-SBY Bahas Reshuffle Kabinet	Reshuffle Kabinet	Undang-Undang Ormas
29	Jokowi Gelar Pertemuan Internal, Persiapkan Reshuffle Kabinet?	Reshuffle Kabinet	Reshuffle Kabinet
30	Diisukan Masuk Kabinet, Ini Kata Idrus Marham	Reshuffle Kabinet	Reshuffle Kabinet
31	Reshuffle Kabinet Tunggu 2 Menteri Maju Pilgub?	Reshuffle Kabinet	Reshuffle Kabinet
32	Pukul 11.00 WIB, Jokowi Umumkan Reshuffle Kabinet	Reshuffle Kabinet	Reshuffle Kabinet
33	Peristiwa 27 Juli dan di Balik Momen "Reshuffle" Kabinet	Reshuffle Kabinet	Reshuffle Kabinet
34	Pengamat: Isu "Reshuffle" Kabinet Hilang-Timbul, Jokowi Dinilai Terkena "SBY Syndrom"	Reshuffle Kabinet	Reshuffle Kabinet
35	Mendagri: Pasti akan Ada Reshuffle di Awal Tahun 2018	Reshuffle Kabinet	Reshuffle Kabinet
36	Reshuffle Kabinet, Jokowi Lantik 2 Menteri dan Watimpres	Reshuffle Kabinet	Reshuffle Kabinet
37	Wapres JK Soal Isu Reshuffle: Belum Bocor, Tanya Presiden Jokowi	Reshuffle Kabinet	Reshuffle Kabinet
38	Teguran Presiden di Tengah Isu Reshuffle	Reshuffle Kabinet	Reshuffle Kabinet
39	Sinyal Kepastian Reshuffle Jilid III dari JK	Reshuffle Kabinet	Reshuffle Kabinet
40	Politikus Golkar Dorong Jokowi Reshuffle Menteri Sektor Polhukam	Reshuffle Kabinet	Reshuffle Kabinet
41	1 RUU Rampung, Komisi IX DPR Usulkan 2 RUU Jadi Prioritas 2018	Undang-Undang Ormas	Undang-Undang Ormas
42	Teten Tepis Reshuffle Jilid III untuk Bentuk Timses Jokowi di 2019	Undang-Undang Ormas	Undang-Undang Ormas
43	Reshuffle Kabinet: Khofifah Diganti, Airlangga Tetap Menperin?	Undang-Undang Ormas	Undang-Undang Ormas

No	Artikel	Kategori Aktual	Kategori Prediksi Sistem
44	Mayoritas Fraksi di DPR Dukung Perppu Ormas Jadi UU	Undang-Undang Ormas	Undang-Undang Ormas
45	Rapat Paripurna DPR Sahkan Perppu Ormas Jadi Undang-undang	Undang-Undang Ormas	Undang-Undang Ormas
46	Bawaslu Susun Indeks Kerawanan Pilkada 2018	Undang-Undang Ormas	Undang-Undang Ormas
47	Ridwan Kamil Berharap Pilkada Jabar 2018 Jangan Seperti DKI dan Pilpres 2014	Undang-Undang Ormas	Undang-Undang Ormas
48	Besok, Megawati Pimpin Rapat Konsolidasi Pemenangan Pilkada di Banten	Undang-Undang Ormas	Undang-Undang Ormas
49	Ini Nama-nama Calon yang Diusung Partai Demokrat di Banten	Undang-Undang Ormas	Undang-Undang Ormas
50	KPU Ungkap Tiga Faktor yang Bikin Pilkada 2018 Rawan Konflik	Undang-Undang Ormas	Undang-Undang Ormas
51	Ini Langkah Bawaslu Cegah Isu SARA di Pilkada 2018	Undang-Undang Ormas	Undang-Undang Ormas
52	Ulama Purwakarta Gelar Konvensi, Jaring Kandidat Cawabup Independen	Undang-Undang Ormas	Undang-Undang Ormas
53	I Wayan Koster dan Cok Oka Resmi Diusung PDIP di Pilgub Bali 2018	Undang-Undang Ormas	Undang-Undang Ormas
54	Jelang Pilkada Serentak 2018, Polda Jabar Waspada Peredaran Upa	Undang-Undang Ormas	Undang-Undang Ormas
55	Jelang Pilkada Serentak 2018, Peran Pemuda di Politik Makin Menguat	Undang-Undang Ormas	Undang-Undang Ormas
56	Maju di Pilkada, Cak Eko Impikan Kota Bekasi Bebas Macet dan Banjir	Undang-Undang Ormas	Undang-Undang Ormas
57	Tangkal Kampanye Hitam, Polda Jabar Tingkatkan Cyber Patrol	Undang-Undang Ormas	Undang-Undang Ormas
58	MNC Group Dukung Penuh Pilkada 2018 dan Pemilu 2019	Undang-Undang Ormas	Undang-Undang Ormas

No	Artikel	Kategori Aktual	Kategori Prediksi Sistem
59	Sarjan Dinilai Berpeluang di Pilkada Sumsel	Undang-Undang Ormas	Undang-Undang Ormas
60	11 November, PDI-P Umumkan Sejumlah Calon untuk Pilkada 2018	Undang-Undang Ormas	Undang-Undang Ormas

E. Pengujian Blackbox

Tabel E. 1 Pengujian *Blackbox*

No.	Fitur	Aksi	Hasil	Keterangan
1.	Melihat Dashboard	Klik menu Dashboard	Menampilkan halaman dashboard berupa chart data training, chart data testing, chart cek kategori dokumen, chart data stopword, dan chart visitor.	Berhasil
2.	Melihat <i>Text Preprocessing Data Training</i>	Klik menu Data Training	Menampilkan halaman menu Data Training, action button update dan delete, serta action button text preprocessing yang meliputi casefolding, tokenizing, stopword, stemming, frekuensi, daftar kata dan probability.	Berhasil
		Klik tombol Casefolding pada halaman menu data training	Menampilkan halaman text preprocessing casefolding	Berhasil
		Klik tombol Tokenizing pada halaman menu data training	Menampilkan halaman text preprocessing tokenizing	Berhasil
		Klik tombol Stopword pada halaman menu data training	Menampilkan halaman text preprocessing stopword	Berhasil
		Klik tombol Stemming	Menampilkan halaman text preprocessing stemming	Berhasil

			pada halaman menu data <i>training</i>		
3.	Melihat Frekuensi Data Training		Klik menu Data <i>Training</i>	Menampilkan halaman menu Data <i>Training</i> , <i>action button update dan delete</i> , serta <i>action button text preprocessing</i> yang meliputi <i>casefolding</i> , <i>tokenizing</i> , <i>stopword</i> , <i>stemming</i> , frekuensi, daftar kata dan <i>probability</i> .	Berhasil
			Klik button Frekuensi	Menampilkan halaman frekuensi kata data training	Berhasil
4.	Melihat Daftar Kata Data Training		Klik menu Data <i>Training</i>	Menampilkan halaman menu Data <i>Training</i> , <i>action button update dan delete</i> , serta <i>action button text preprocessing</i> yang meliputi <i>casefolding</i> , <i>tokenizing</i> , <i>stopword</i> , <i>stemming</i> , frekuensi, daftar kata dan <i>probability</i> .	Berhasil
			Klik button Daftar Kata	Menampilkan halaman daftar kata data training	Berhasil
5.	Melihat <i>Probability</i> Kata Data Training		Klik menu Data <i>Training</i>	Menampilkan halaman menu Data <i>Training</i> , <i>action button update dan delete</i> , serta <i>action button text preprocessing</i> yang meliputi <i>casefolding</i> , <i>tokenizing</i> , <i>stopword</i> , <i>stemming</i> , frekuensi, daftar kata dan <i>probability</i> .	Berhasil
			Klik button <i>Probability</i>	Menampilkan halaman nilai <i>probability</i> kata data training	Berhasil
			Klik tombol <i>Save</i> pada tambah data <i>testing</i> namun data tidak lengkap	Menampilkan alert " <i>Please fill out this field</i> "	Berhasil
			Klik tombol Ubah pada halaman data <i>testing</i> yang sesuai dipilih lalu klik <i>save</i>	Mengubah data ke database dan menampilkan halaman data <i>testing</i>	Berhasil
			Klik tombol Hapus pada halaman data <i>testing</i>	Menghapus data <i>testing</i> dan menampilkan halaman data <i>testing</i> .	Berhasil

6.	Melihat Text Preprocessing Data Testing	Klik menu Data <i>Testing</i>	Menampilkan halaman menu Data <i>Testing</i> , <i>action button update dan delete</i> , serta <i>action button text preprocessing</i> yang meliputi <i>casefolding</i> , <i>tokenizing</i> , <i>stopword</i> , <i>stemming</i> , dan hasil.	Berhasil
		Klik tombol <i>Casefolding</i> pada halaman menu data <i>testing</i>	Menampilkan halaman <i>text preprocessing casefolding</i>	Berhasil
		Klik tombol <i>Tokenizing</i> pada halaman menu data <i>testing</i>	Menampilkan halaman <i>text preprocessing tokenizing</i>	Berhasil
		Klik tombol <i>Stopword</i> pada halaman menu data <i>testing</i>	Menampilkan halaman <i>text preprocessing stopwords</i>	Berhasil
		Klik tombol <i>Stemming</i> pada halaman menu data <i>testing</i>	Menampilkan halaman <i>text preprocessing stemming</i>	Berhasil
7.	Melihat Hasil Klasifikasi Data Testing	Klik menu Data <i>Testing</i>	Menampilkan halaman menu Data <i>Testing</i> , <i>action button update dan delete</i> , serta <i>action button text preprocessing</i> yang meliputi <i>casefolding</i> , <i>tokenizing</i> , <i>stopword</i> , <i>stemming</i> , dan hasil.	Berhasil
		Klik tombol Hasil pada halaman menu data <i>testing</i>	Menampilkan halaman hasil klasifikasi kategori yang telah diinputkan	Berhasil
8.	Mengelola Data Stopword	Klik menu Data <i>Stopword</i>	Menampilkan halaman menu Data <i>Stopword</i> , <i>action button update dan delete</i> , serta <i>action tambah stopwords</i> .	Berhasil
		Klik tombol Tambah Dokumen	Menampilkan modal berupa form tambah data <i>stopword</i> dan tombol <i>save</i>	Berhasil

		Data <i>Stopword</i>		
		Klik tombol <i>Save</i> pada tambah data <i>stopword</i> namun data tidak lengkap	Menampilkan alert " <i>Please fill out this field</i> "	Berhasil
		Klik tombol Ubah pada halaman data <i>stopword</i> yang sesuai dipilih lalu klik <i>save</i>	Mengubah data ke database dan menampilkan halaman data <i>stopword</i>	Berhasil
		Klik tombol Hapus pada halaman data <i>stopword</i>	Menghapus data <i>stopword</i> dan menampilkan halaman data <i>stopword</i> .	Berhasil
9.	<i>Logout</i>	Klik tombol <i>Logout</i> pada dashboard pengguna	Keluar dari halaman pengguna kembali ke halaman awal sistem	Berhasil
10.	Melihat Berita Politik	Membuka system	Menampilkan halaman awal system Cek Kategori Dokumen	Berhasil
		Klik Data Berita Politik	Menampilkan data berita politik yang telah diinputkan oleh Editor	Berhasil
		Klik Baca	Menampilkan isi artikel	Berhasil