

TUGAS AKHIR

**PENYARING KOMENTAR CYBER-BULLYING
PADA KONTEN BLOG**



Oleh:

Danar Dono

215310285

**PROGRAM SARJANA
PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
INSTITUT SAINS DAN TEKNOLOGI TERPADU SURABAYA
SURABAYA
2021**

TUGAS AKHIR

PENYARING KOMENTAR CYBER-BULLYING PADA KONTEN BLOG

**Diajukan Guna Memenuhi Sebagian Persyaratan Untuk Memperoleh Gelar
Sarjana Komputer
Pada
Institut Sains dan Teknologi Terpadu Surabaya**

Disetujui oleh Tim Pengaji Tugas Akhir:

1. Eka Rahayu Setyaningsih, S.Kom., M.Kom. (Pembimbing)
2. C. Pickerling, S.Kom., M.Kom. (Co-Pembimbing)
3. Reddy Alexandro H., S.Kom., M.Kom. (Pengaji I)
4. Dr. Joan Santoso, S.Kom., M.Kom. (Pengaji II)
5. Dr. Esther Irawati Setiawan, S.Kom, M.Kom. (Pengaji III)

**SURABAYA
JANUARI 2021**

SURAT PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama : Danar Dono
Fakultas/ Prodi : Sains dan Teknologi/ Informatika
NRP : 215310285

dengan ini menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir dengan judul:

PENYARING KOMENTAR CYBER-BULLYING PADA KONTEN BLOG

adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 21-01-2021
Yang Membuat Pernyataan,

Danar Dono
215310285

ABSTRAK

Cyberbullying merupakan ancaman nyata dalam interaksi di antara penulis konten blog dan pembaca blog. Tugas akhir ini bertujuan membahas tentang masalah sebagai berikut ini. Pertama, mengembangkan sistem manajemen konten blog menggunakan *framework* CodeIgniter. Kedua, mengembangkan fitur penyaring *cyberbullying* untuk mengklasifikasikan komentar menjadi *cyberbullying* atau *non-cyberbullying*.

Multinomial Naive Bayes digunakan sebagai metode untuk klasifikasi komentar. Dengan menggunakan *4-fold cross validation*, dataset sejumlah 6.537 komentar dibagi menjadi 4 sub himpunan dataset D_1, D_2, D_3, D_4 yang berisi masing-masing 3 *fold* untuk data latih dan 1 *fold* untuk data uji. *Accuracy and error rate* digunakan untuk mengukur kinerja model dalam mengklasifikasikan komentar. Adapun indikator-indikator *usability*, *reliability*, *functionality*, *efficiency* digunakan untuk mengukur kualitas *website* dengan melibatkan 39 responden untuk menilai kualitas *website* menggunakan skala likert 1-5.

Berdasarkan pengujian dapat disimpulkan bahwa: pengembangan sistem manajemen konten menggunakan *framework* CodeIgniter memberikan pengalaman bagi penggunanya dengan tingkat *usability* sebesar 3,49 dari skala 5,00; memiliki tingkat *reliability* sebesar 4,28 dari skala 5,00; memiliki tingkat *functionality* sebesar 4,11 dari skala 5,00; dan memiliki tingkat *efficiency* sebesar 3,78 dari skala 5,00. Pengembangan sistem manajemen konten memiliki beberapa fitur seperti penggunaan *pretty URL*; penggunaan Ion Auth dan Facebook API untuk otentifikasi secara ringkas dan ringan; penggunaan Chartjs untuk menyajikan grafik laporan; menyediakan interaksi antara pembaca dan penulis melalui fitur rating, komentar dan pelaporan penyalahgunaan komentar; menyediakan fitur pengelompokan artikel berdasarkan topik artikel dan kata kunci. Pengembangan fitur penyaring komentar *cyberbullying* menghasilkan: pertama, model dengan fitur *stemming-unigram*, model tanpa fitur *stemming-unigram*, model dengan fitur *stemming-bigram*, model dengan tanpa fitur *stemming-bigram* masing-masing memiliki akurasi sebesar 73%, 70%, 67%, 66%. Kedua, model *stemming-unigram* memiliki ukuran fitur kata sebesar 21% atau setara 4.910 fitur kata dibandingkan ukuran fitur kata dari model dengan tanpa fitur *stemming-bigram* sebesar 23.787 fitur kata. Ketiga, penggunaan fitur *stemming-unigram* mampu memberikan akurasi yang baik sebesar 73% dan tingkat *error* 27% serta efisien secara ukuran fitur kata.

ABSTRACT

Cyberbullying is a real threat to the interactions between blog content writers and blog readers. This final project aims to discuss the following problems. First, develop a blog content management system using the CodeIgniter framework. Second, developing a cyberbullying filter feature to classify comments as cyberbullying or non-cyberbullying.

The Multinomial Naive Bayes was used as a method for classification of comments. By using 4-fold cross-validation, the dataset of 6,537 comments is divided into 4 sub-sets of datasets D1, D2, D3, D4, each containing 3 folds for training data and 1 fold for test data. Accuracy and error rate is used to measure the performance of the model in classifying comments. The indicators of usability, reliability, functionality, efficiency are used to measure the quality of the website by involving 39 respondents to assess the quality of the website using a Likert scale of 1-5.

Based on the test it can be concluded that: the development of a content management system using the CodeIgniter framework provides an experience for its users with a usability level of 3.49 on a scale of 5.00; has a reliability level of 4.28 from a scale of 5.00; it has a level of functionality of 4.11 from a scale of 5.00, and has an efficiency level of 3.78 from a scale of 5.00. The development of the content management system has several features such as using pretty URLs; use of Ion Auth and Facebook API for concise and lightweight authentication; use of Chartjs to present chart reports; provide interaction between readers and writers through rating, comment and comment abuse reporting features; provides a feature grouping articles based on article topics and keywords. The development of the cyberbullying comment filter feature resulted in: first, a model with a stemming-unigram feature, a model without the stemming-unigram feature, a model with the stemming-bigram feature, a model without the stemming-bigram feature, each with an accuracy of 73%, 70%, 67 %, 66%. Second, the stemming-unigram model has a word feature size of 21% or the equivalent of 4,910 word features compared to the word feature size of the model without the stemming-bigram feature of 23,787 word features. Third, the use of the stemming-unigram feature can provide good accuracy of 73% and an error rate of 27% and efficient in terms of word feature size.

KATA PENGANTAR

Puji syukur atas kehadirat Tuhan Yang Maha Esa yang telah memberikan rahmat serta karunia-Nya kepada penulis sehingga penulis berhasil menyelesaikan tugas akhir berjudul “Penyaring Komentar Cyber-Bullying pada Konten Blog”. Buku ini berisikan mengenai pengembangan sistem manajemen konten blog dan pengembangan sistem penyaring komentar *cyberbullying*. Penulis menyadari bahwa penulisan tugas akhir ini sulit terealisasi tanpa bantuan dari berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada pihak-pihak yang telah membantu dalam penyusunan tugas akhir di antaranya:

1. Kedua orang tua, kakak perempuan, kakak laki-laki, dan adik perempuan yang telah memberikan dukungan baik material maupun immaterial.
2. Ibu Eka Rahayu Setyaningsih, S.Kom., M.Kom. selaku Dosen Pembimbing yang sabar membimbing penulis dalam menyelesaikan tugas akhir.
3. Ibu C. Pickerling, S.Kom., M.Kom. selaku Co. Pembimbing yang senantiasa memberikan dukungan kepada penulis untuk menyelesaikan tugas akhir ini.
4. Reddy Alejandro H., S.Kom., M.Kom., Dr. Joan Santoso, S.Kom., M.Kom., Dr. Esther Irawati Setiawan, S.Kom, M.Kom. selaku dosen pengaji tugas akhir yang memberikan masukan berharga untuk perbaikan tugas akhir ini.
5. Rekan-rekan yang telah bersedia membantu dan memberikan motivasi kepada penulis untuk menyelesaikan tugas akhir ini.

Akhir kata, penulis sampaikan terima kasih kepada semua pihak yang telah berperan serta dalam penyusunan buku ini dari awal sampai akhir. Semoga buku ini dapat membawa manfaat dan wawasan bagi para pembaca.

Surabaya, Januari 2021

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	i
HALAMAN PENGESAHAN.....	ii
SURAT PERNYATAAN KEASLIAN	iii
ABSTRAK	iv
ABSTRACT	v
KATA PENGANTAR	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR	xii
DAFTAR TABEL.....	xv
DAFTAR ALGORITMA.....	xviii
DAFTAR SEGMENT PROGRAM	xix
DAFTAR RUMUS	xx
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Tujuan	2
1.3 Ruang Lingkup.....	2
1.3.1 Deskripsi Sistem.....	3
1.3.2 Fitur-Fitur Sistem	4
1.4 Tahapan Penyelesaian	8
1.5 Spesifikasi Sistem	9
1.6 Batasan Sistem	10
1.7 Sistematika Pembahasan	10
BAB II TINJAUAN PUSTAKA.....	13
2.1 Cyberbullying.....	13
2.2 Text Preprocessing	17
2.3 Algoritma Stemming Nazief dan Adriani	20
2.4 CodeIgniter.....	26

2.5	Teorema Bayes.....	28
2.5.1	Naive Bayes Classifier	31
2.5.2	Multinomial Naïve Bayes.....	35
2.6	Ukuran Evaluasi Model.....	38
2.7	Teknik K-Fold Cross Validation.....	40
2.8	Ukuran Evaluasi Website.....	41
2.9	Tool dan Library Pendukung	42
BAB III	DATASET DAN PREPROCESSING	46
3.1	Pembentukan Dataset	46
3.2	Pelabelan dan Vertifikasi Kategori Dataset Komentar	47
3.3	Preprocessing Dataset	52
3.3.1	Case Folding Dataset.....	52
3.3.2	Cleaning Dataset	54
3.3.3	Normalisasi Bahasa Gaul Dataset	55
3.3.4	Stemming Dataset	56
3.3.5	Stopword Dataset	57
3.3.6	Tokenizing Dataset.....	59
3.3.7	N-gram Dataset	60
BAB IV	MULTINOMINAL NAIVE BAYES	63
4.1	Arsitektur Sistem Penyaring Komentar Cyber-Bullying	63
4.2	Arsitektur Sistem Pelatihan Model	65
4.2.1	Preprocessing Dataset Komentar	67
4.2.2	Pembentukan Fitur Kata.....	68
4.2.3	Perhitungan Probabilitas Prior	70
4.2.4	Perhitungan Probabilitas Fitur Kata	71
4.3	Arsitektur Pengujian Data	73
4.4	Tampilan Daftar Komentar	76
BAB V	ANALISIS SISTEM	80
5.1	Analisis Sistem Manajemen Konten Blog	80
5.1.1	Deskripsi Sistem Lama.....	80
5.1.2	Analisis Permasalahan Sistem Lama.....	83

5.1.3	Deskripsi Sistem Baru	84
5.1.4	Perbandingan Fitur Sistem Lama dan Sistem Baru..	86
5.2	Data Flow Diagram.....	87
5.2.1	DFD Level Konteks	88
5.2.2	DFD Level 0.....	89
5.2.3	DFD Level 1.....	91
5.2.4	DFD Level 2.....	95
5.2.5	DFD Level 3.....	101
BAB VI	DESAIN SISTEM	103
6.1	Desain Arsitektur Sistem.....	103
6.2	Desain Database	112
6.2.1	Conceptual Data Model.....	112
6.2.2	Physical Data Model	115
6.2.3	Deskripsi Struktur Tabel	118
6.3	Desain Antar Muka	134
6.3.1	Desain Front End.....	134
6.3.2	Desain Back End	143
6.4	Desain Prosedural.....	155
6.4.1	Desain Prosedural Case Folding	156
6.4.2	Desain Prosedural Cleaning	156
6.4.3	Desain Prosedural Normalisasi Bahasa Gaul	157
6.4.4	Desain Prosedural Stemming	158
6.4.5	Desain Prosedural Stopword	159
6.4.6	Desain Prosedural Tokenizing	160
6.4.7	Desain Prosedural Bigram	161
6.4.8	Desain Prosedural Pembuatan Model	162
6.4.9	Desain Prosedural Kalkulasi Kategori Komentar	163
6.4.10	Desain Prosedural Pengujian Model	165
6.4.11	Desain Prosedural Pembuatan Artikel.....	166
6.4.12	Desain Prosedural Pengeditan Artikel.....	167
6.4.13	Desain Prosedural Penghapusan Artikel	168

6.4.14	Desain Prosedural Penguploadan Galeri	168
6.4.15	Desain Prosedural Follow-Unfollow Account	169
BAB VII	IMPLEMENTASI	172
7.1	Segmen Program Penyaring Komentar Cyberbullying.....	172
7.1.1	Segmen Program Case Folding	172
7.1.2	Segmen Program Cleaning.....	173
7.1.3	Segmen Program Normalisasi Bahasa Gaul.....	174
7.1.4	Segmen Program Stemming.....	175
7.1.5	Segmen Program Stopword.....	176
7.1.6	Segmen Program Tokenizing	177
7.1.7	Segmen Program Bigram	178
7.1.8	Segmen Program Model Penyaring Cyberbullying .	179
7.1.9	Segmen Program Kalkulasi Kategori Komentar.....	181
7.1.10	Segmen Program Pengujian Model.....	184
7.2	Segmen Program Sistem Manajemen Konten.....	186
7.2.1	Segmen Program Pembuatan Artikel	187
7.2.2	Segmen Program Pengeditan Artikel	189
7.2.3	Segmen Program Penghapusan Artikel.....	190
7.2.4	Segmen Program Penguploadan Galeri.....	191
7.2.5	Segmen Program Follow-Unfollow Account.....	192
7.2.6	Segmen Program Pretty URL.....	196
7.2.7	Segmen Program Laporan Artikel.....	197
7.2.8	Segmen Program Registrasi Akun	199
BAB VIII	PENGUJIAN	204
8.1	Pengujian Fungsionalitas	204
8.1.1	Pengujian Fungsionalitas Login dan Registrasi	204
8.1.2	Pengujian Fungsionalitas Konfigurasi Akun.....	206
8.1.3	Pengujian Fungsionalitas Manajemen Galeri.....	207
8.1.4	Pengujian Fungsionalitas Manajemen Artikel	209
8.1.5	Pengujian Fungsionalitas Manajemen Komentar.....	211
8.1.6	Pengujian Fungsionalitas Following-Follower	212

8.1.7 Pengujian Fungsionalitas Notifikasi.....	213
8.1.8 Pengujian Fungsionalitas Konfigurasi Website	215
8.1.9 Pengujian Fungsionalitas Tentang Website	216
8.1.10 Pengujian Fungsionalitas Kebijakan Website	217
8.1.11 Pengujian Fungsionalitas FAQ.....	218
8.1.12 Pengujian Fungsionalitas Manajemen Tagar	219
8.1.13 Pengujian Fungsionalitas Manajemen User	220
8.1.14 Pengujian Fungsionalitas Manajemen Kategori.....	221
8.1.15 Pengujian Fungsionalitas Evaluasi Website.....	222
8.1.16 Pengujian Fungsionalitas Dashboard	223
8.1.17 Pengujian Fungsionalitas Penyaring Cyberbullying	225
8.2 Pengujian Model Penyaring Cyberbullying	230
8.3 Pengujian Sistem Berdasarkan Pengalaman Pengguna.....	240
BAB IX PENUTUP	245
9.1 Kesimpulan	245
9.2 Saran.....	247
DAFTAR PUSTAKA	248
RIWAYAT HIDUP.....	251
LAMPIRAN A CONTOH DATASET KOMENTAR	A-1
LAMPIRAN B KUESIONER EVALUASI WEBSITE	B-1
LAMPIRAN C DAFTAR PROFIL RESPONDEN.....	C-1
LAMPIRAN D DAFTAR JAWABAN	D-1

DAFTAR GAMBAR

Gambar	Halaman
2.1 Cara Kerja CodeIgniter	27
2.2 4-Fold Cross-Validation	40
3.1 Kerangka Teori Cyberbullying.....	48
3.2 Contoh Artikel dan Komentar	50
3.3 Tampilan Halaman Vertifikasi Sentimen Komentar	51
3.4 Tampilan Halaman Preprocessing Dataset Komentar.....	52
4.1 Arsitektur Penyaring Komentar Cyberbullying	63
4.2 Arsitektur Pelatihan Data	65
4.3 Arsitektur Pengujian Data Komentar	74
4.4 Data Uji Sebelum dan Setelah Di-Preprocessing	75
4.5 Ilustrasi Peringatan Komentar Terdeteksi Cyberbullying	77
4.6 Tampilan Daftar Komentar di Sisi Front End	78
4.7 Tampilan Daftar Komentar di Sisi Back End.....	79
5.1 DFD Level Konteks	88
5.2 DFD Level 0.....	90
5.3 DFD Level 1 Manajemen User dan Artikel	92
5.4 DFD Level 1 Manajemen Website.....	93
5.5 DFD Level 1 Penyaring Komentar.....	95
5.6 DFD Level 2 Manajemen User	96
5.7 DFD Level 2 Manajemen Artikel.....	97
5.8 DFD Level 2 Preprocessing Komentar	98
5.9 DFD Level 2 Model Penyaring Cyberbullying	99
5.10 DFD Level 2 Pengujian Model Cyberbullying	100
5.11 DFD Level 3 Statistik Artikel	101
6.1 CDM I Tentang Sistem Manajemen Konten.....	113
6.2 CDM II Tentang Sistem Manajemen Konten.....	115
6.3 PDM I Tentang Sistem Manajemen Konten	116

6.4	PDM II Tentang Sistem Manajemen Konten	117
6.5	Desain Login	135
6.6	Desain Registrasi.....	136
6.7	Desain Beranda	137
6.8	Desain Statistik Detail Artikel.....	138
6.9	Desain Konten Artikel.....	138
6.10	Desain Rating Artikel	139
6.11	Desain Pesan Peringatan Cyberbullying	140
6.12	Desain Tentang Website.....	140
6.13	Desain Kebijakan Website	141
6.14	Desain Profil.....	141
6.15	Desain FAQ.....	142
6.16	Desain Evaluasi Website	142
6.17	Desain Dashboard	143
6.18	Desain Notifikasi.....	144
6.19	Desain Konfigurasi Profil.....	144
6.20	Desain Manajemen User	145
6.21	Desain Manajemen Galeri	145
6.22	Desain Konfigurasi Tentang Website	146
6.23	Desain Konfigurasi Kebijakan Website	146
6.24	Desain Manajemen FAQ	147
6.25	Desain Manajemen Kategori	147
6.26	Desain Manajemen Tagar.....	148
6.27	Desain Manajemen Following	148
6.28	Desain Manajemen Artikel.....	149
6.29	Desain Pembuatan Artikel.....	149
6.30	Desain Konfigurasi Website.....	150
6.31	Desain Manajemen Dataset Artikel.....	151
6.32	Desain Manajemen Dataset Komentar	151
6.33	Desain Manajemen Dataset Stopword	152
6.34	Desain Manajemen Dataset Bahasa Gaul.....	152

6.35	Desain Model Fitur Kata	153
6.36	Desain Pengujian Model	153
6.37	Desain Daftar Komentar Beserta Kategorinya.....	154
6.38	Desain Pengelolaan Komentar	154
6.39	Desain Klasifikasi Kategori Komentar	155
7.1	Pretty URL	196
7.2	Laporan Artikel Terpopuler Memanfaatkan Chartjs	199
7.3	Registrasi Dengan Facebook	200
7.4	Konfirmasi Perizinan Akun Facebook	200
8.1	Demografi Respondn Berdasarkan Pekerjaan	241
8.2	Demografi Respondn Berdasarkan Gender	241
8.3	Grafik Perbandingan Kualitas Website	243

DAFTAR TABEL

Tabel	Halaman
2.1 Tabel Kombinasi Prefix dan Suffix yang Tidak Diperbolehkan	24
2.2 Tabel Penentuan Tipe Awalan Untuk Kata Dengan Awalan “Te”	24
2.3 Tabel Penentuan Awalan dari Jenis Awalan	25
2.4 Tabel Ilustrasi Daftar Kehadiran Siswa Berdasarkan Kondisi Cuaca	30
2.5 Tabel Ilustrasi Prior Kehadiran Siswa Berdasarkan Kondisi Cuaca ..	30
2.6 Tabel Ilustrasi Rekomendasi HP	34
2.7 Tabel Confusion Matrix	38
3.1 Tabel Spesifikasi Dataset	47
3.2 Tabel Case Folding Dataset.....	53
3.3 Tabel Cleaning Dataset	54
3.4 Tabel Normalisasi Bahasa Gaul	56
3.5 Tabel Stemming Dataset	57
3.6 Tabel Stopword Dataset	58
3.7 Tabel Tokenizing Dataset.....	59
3.8 Tabel Bigram Dataset.....	61
4.1 Tabel Hasil Preprocessing Data Komentar	67
4.2 Tabel Pembentukan Fitur Kata.....	69
4.3 Tabel Perhitungan Probabilitas Prior	71
4.4 Tabel Perhitungan Probabilitas Fitur Kata	72
5.1 Tabel Perbandingan Fitur Tugas Akhir, Kompasiana, Indonesiana...	86
6.1 Tabel Arsitektur Sistem Manajemen Konten Blog	103
6.2 Tabel Penilai.....	118
6.3 Tabel Jawaban	119
6.4 Tabel Kuesioner	119
6.5 Tabel Nilai.....	120
6.6 Tabel Kritik_Saran	120
6.7 Tabel Kamus Bahasagaul	121

6.8	Tabel Prior.....	121
6.9	Tabel Kamus Stemmer	122
6.10	Tabel Dataset_Komentar.....	122
6.11	Tabel Dataset_Artikel	123
6.12	Tabel Kelas_Kata	124
6.13	Tabel Pengujian.....	124
6.14	Tabel Detail_Pengujian	125
6.15	Tabel Kontak	126
6.16	Tabel Gallery.....	126
6.17	Tabel Pengikut.....	127
6.18	Tabel Notifikasi.....	127
6.19	Tabel Detail_Pengikut.....	128
6.20	Tabel Mengikuti	128
6.21	Tabel Detail_Mengikuti	129
6.22	Tabel Pengguna	129
6.23	Tabel View	130
6.24	Tabel Komentar.....	130
6.25	Tabel Tagar	131
6.26	Tabel Identitas	131
6.27	Tabel Rating	132
6.28	Tabel Kategori.....	132
6.29	Tabel Posting.....	133
6.30	Tabel Balasan_Komentar	133
8.1	Tabel Pengujian Fungsionalitas Manajemen Artikel	205
8.2	Tabel Pengujian Fungsionalitas Konfigurasi Akun	207
8.3	Tabel Pengujian Fungsionalitas Manajemen Galeri.....	208
8.4	Tabel Pengujian Fungsionalitas Manajemen Artikel	209
8.5	Tabel Pengujian Fungsionalitas Manajemen Komentar.....	211
8.6	Tabel Pengujian Fungsionalitas Manajemen Follower-Following	213
8.7	Tabel Pengujian Fungsionalitas Notifikasi	214
8.8	Tabel Pengujian Fungsionalitas Konfigurasi Website	215

8.9	Tabel Pengujian Fungsionalitas Manajemen Tentang Website	216
8.10	Tabel Pengujian Fungsionalitas Manajemen Kebijakan Website	217
8.11	Tabel Pengujian Fungsionalitas Manajemen FAQ.....	218
8.12	Tabel Pengujian Fungsionalitas Manajemen Tagar	220
8.13	Tabel Pengujian Fungsionalitas Manajemen User	221
8.14	Tabel Pengujian Fungsionalitas Manajemen Kategori.....	222
8.15	Tabel Pengujian Fungsionalitas Manajemen Evaluasi Website.....	223
8.16	Tabel Pengujian Fungsionalitas Dashboard	224
8.17	Tabel Pengujian Fungsionalitas Manajemen Dataset Artikel	225
8.18	Tabel Pengujian Fungsionalitas Manajemen Dataset Komentar.....	226
8.19	Tabel Pengujian Fungsionalitas Manajemen Dataset Bahasa Gaul ...	227
8.20	Tabel Pengujian Fungsionalitas Dataset Stopword	228
8.21	Tabel Pengujian Fungsionalitas Fitur Kata dan Prior	229
8.22	Tabel Pengujian Fungsionalitas Pengujian Model	230
8.23	Tabel Evaluasi Model Dengan Fitur Stemming-Bigram.....	231
8.24	Tabel Contoh Hasil TP dan TN	232
8.25	Tabel Contoh Hasil FP dan FN	233
8.26	Tabel Evaluasi Model Dengan Fitur Tanpa Stemming-Bigram.....	236
8.27	Tabel Evaluasi Model Dengan Fitur Stemming-Unigram	236
8.28	Tabel Evaluasi Model Dengan Fitur Tanpa Stemming-Unigram	237
8.29	Tabel Perbandingan Ukuran Fitur Kata Antar Model	239
8.30	Tabel Skala Predikat Ukuran Kualitas Website	242
8.31	Tabel Hasil Pengujian Kualitas Website	243
A.1	Tabel Contoh Dataset Komentar	A-1
B.1	Tabel Kuesioner Evaluasi Website.....	B-1
C.1	Tabel Daftar Profil Responden.....	C-1
D.1	Tabel Daftar Jawaban	D-1

DAFTAR ALGORITMA

Algoritma	Halaman
6.1 Algoritma Case Folding	156
6.2 Algoritma Cleaning	156
6.3 Algoritma Normalisasi Bahasa Gaul.....	157
6.4 Algoritma Stemming	158
6.5 Algoritma Stopword	160
6.6 Algoritma Tokenizing	160
6.7 Algoritma Bigram	161
6.8 Algoritma Pembuatan Model	162
6.9 Algoritma Kalkulasi Kategori Komentar	163
6.10 Algoritma Pengujian Model	165
6.11 Algoritma Pembuatan Artikel	166
6.12 Algoritma Pengeditan Artikel	167
6.13 Algoritma Penghapusan Artikel	168
6.14 Algoritma Pengulangan Galeri	169
6.15 Algoritma Follow-Unfollow Account.....	170

DAFTAR SEGMENT PROGRAM

Segment Program	Halaman
7.1 Segmen Program Case Folding	173
7.2 Segmen Program Cleaning	173
7.3 Segmen Program Normalisasi Bahasa Gaul.....	174
7.4 Segmen Program Stemming.....	175
7.5 Segmen Program Stopword.....	177
7.6 Segmen Program Tokenizing	178
7.7 Segmen Program Bigram	178
7.8 Segmen Program Pembuatan Model Penyaring Cyberbullying.....	180
7.9 Segmen Program Kalkulasi Kategori Komentar	181
7.10 Segmen Program Pengujian Model.....	184
7.11 Segmen Program Pembuatan Artikel	187
7.12 Segmen Program Pengeditan Artikel	189
7.13 Segmen Program Penghapusan Artikel	190
7.14 Segmen Program Pengulangan Galeri.....	191
7.15 Segmen Program Follow-Unfollow Account Bagian 1	192
7.16 Segmen Program Follow-Unfollow Account Bagian 2	194
7.17 Segmen Program Pembuatan Pretty URL.....	197
7.18 Segmen Program Laporan Artikel Terpopuler.....	197
7.19 Segmen Program Mengambil Data Profil Facebook API	201
7.20 Segmen Program Registrasi Member Menggunakan Ion Auth.....	201

DAFTAR RUMUS

Rumus	Halaman
2.1 Aturan Stemming Nazief dan Adriani.....	21
2.2 Probabilitas Bersyarat $P(A B)$	28
2.3 Probabilitas $P(B \cap A)$	29
2.4 Probabilitas $P(A B)$	29
2.5 Probabilitas (Absen Hujan)	30
2.6 Probabilitas $P(H X)$	31
2.7 Probabilitas $P(C_i X)$	32
2.8 Probabilitas $P(X_k C_i)$	33
2.9 Cmap	36
2.10 Cmap dengan Logaritma	37
2.11 Probabilitas Prior $P(c)$	37
2.12 Probabilitas Posterior $P(t_k c)$	37
2.13 Probabilitas Posterior $P(t_k c)$ dengan Laplace Smoothing	38
4.1 Probabilitas Prior $P(c)$	70
4.2 Probabilitas Posterior $P(w c)$	71
4.3 Nilai Cmap	75
8.1 Rata-Rata Kualitas Web	242

BAB I

PENDAHULUAN

Pada Bab I dijelaskan pendahuluan. Adapun penjelasan sub babnya meliputi latar belakang, tujuan, ruang lingkup, tahapan penyelesaian dan sistematika pembahasan. Latar belakang membahas tentang latar permasalahan dalam penyusunan tugas akhir. Tujuan membahas tentang hasil yang hendak dicapai dari penyusunan tugas akhir. Ruang lingkup membahas tentang batasan masalah terkait penyusunan tugas akhir. Tahapan penyelesaian membahas tentang langkah-langkah yang dilakukan dalam pengerjaan tugas akhir. Sementara sistematika pembahasan membahas tentang kerangka Bab penyusunan tugas akhir. Berikut dijelaskan penjelasan selengkapnya:

1.1 Latar Belakang

Dewasa ini perkembangan teknologi dan informasi begitu pesat. Hal ini memudahkan masyarakat dalam mengakses informasi. Masyarakat juga lebih mudah menyebarkan informasi secara masif dan cepat. Masyarakat biasanya mengakses informasi dari berbagai *website* yang terdapat di internet.

Pembuatan *website* bisa dikatakan cukup rumit bagi orang-orang awam yang tidak paham mengenai bahasa pemograman *web*. Untuk itu, diperlukan sistem aplikasi *website* yang berguna untuk membuat *website* baik secara personal maupun organisasi atau dikenal sebagai *content management system* (CMS). Belakangan ini banyak beredar banyak CMS, seperti joomla, wordpress, dan sebagainya. Banyaknya CMS yang beredar tidak hanya membuat masyarakat mudah dalam mengembangkan *website* tetapi juga rentan terhadap keamanan karena dikembangkan oleh banyak orang di seluruh dunia.

Selain itu, CMS biasanya dilengkapi dengan fitur komentar bagi penggunanya, sehingga memungkinkan interaksi antara pembuat konten dan pembaca artikel. Komentar itu bisa mengandung muatan positif maupun negatif. Komentar positif biasanya berkaitan dengan apresiasi, kritik dan saran oleh

pembaca terhadap artikel *web*. Di sisi lain, terdapat pula komentar negatif yang biasanya berkaitan dengan tindakan *cyberbullying* yang menyerang pembuat artikel *web* maupun kelompok masyarakat tertentu.

Cyberbullying merupakan bentuk serangan yang bersifat dengki untuk memberikan kepuasaan atau kesenangan pelaku dengan cara menyiksa orang lain. *Cyberbullying* berdampak secara psikologis, emosional dan trauma sosial. Dampak *cyberbullying* mempengaruhi kepercayaan diri korban menjadi menurun, munculnya kegelisahan dan depresi pada korban hingga efek terburuk adalah korban yang frustasi bisa melakukan tindakan bunuh diri. Untuk itu, diperlukan diidentifikasi terhadap komentar-komentar yang diberikan pengguna dalam suatu *website* guna mengelompokkannya dalam kategori komentar yang bermuatan *cyberbullying* atau tidak sebagai tindakan antisipasi terhadap bahaya *cyberbullying*. Untuk itu, dibutuhkan fitur penyaring komentar *cyberbullying* untuk menyaring berbagai komentar yang berpotensi mengandung *cyberbullying* dalam suatu *website*. Dalam tugas akhir ini akan dibahas bagaimana pengembangan sistem penyaring komentar *cyberbullying* pada konten blog yang selanjutnya dapat digunakan untuk meminimalisir penyalahgunaan komentar khususnya terkait tindakan perundungan yang dilakukan pengguna melalui fitur komentar pada suatu situs *blog*.

1.2 Tujuan

Berdasarkan permasalahan di atas, maka tugas akhir ini memiliki tujuan sebagai berikut :

- Mengembangkan *website* manajemen konten *blog* yang memudahkan pengguna dalam membuat konten *blog* secara efektif dan efisien.
- Mengembangkan fitur penyaring komentar *cyberbullying* untuk klasifikasi komentar menjadi komentar *cyberbullying* dan komentar *non-cyberbullying*.

1.3 Ruang Lingkup

Pada sub bab ini akan dijelaskan tentang ruang lingkup pengembangan sistem penyaring komentar *cyberbullying* pada konten blog. Berikut ini akan

dijelaskan ruang lingkup dalam pengembangan sistem meliputi deskripsi sistem dan fitur-fitur aplikasi. Deskripsi sistem menjelaskan gambaran umum terkait sistem manajemen konten blog yang akan dikembangkan dalam tugas akhir ini. Fitur-fitur aplikasi menjelaskan batasan-batasan pengembangan fitur-fitur yang dikembangkan dalam sistem manajemen konten blog. Berikut penjelasan selengkapnya:

1.3.1 Deskripsi Sistem

Berikut ini dijelaskan arsitektur sistem manajemen konten *blog* yang akan dikembangkan. Sistem terdiri atas penyaring komentar *cyberbullying* dan sistem manajemen konten *blog*. Sistem penyaring komentar *cyberbullying* merupakan fitur untuk klasifikasi komentar menjadi *cyberbullying* atau *non-cyberbullying*. Sementara sistem manajemen konten *blog* merupakan situs yang dikembangkan dalam rangka mengembangkan manajemen pengelolaan konten-konten *blog*. Kedua arsitektur sistem tersebut menjadi fokus dalam penyusunan tugas akhir ini.

Berikut ini dijelaskan masing-masing deskripsi sistemnya meliputi: Pertama, sistem penyaring komentar *cyberbullying* berfungsi untuk menyaring berbagai komentar yang berpotensi mengandung *cyberbullying* baik berupa ujaran kebencian, komentar SARA, maupun pelecehan verbal. Sistem penyaring komentar *cyberbullying* terdiri atas fitur input-output, pelatihan data, dan pengujian data. Fitur input-output merupakan masukan dan keluaran sistem dimana komentar yang diinput pengguna akan diproses model hingga menghasilkan keluaran berupa hasil klasifikasi *cyberbullying* atau *non-cyberbullying*. Jika komentar terindikasi mengandung *cyberbullying*, maka sistem akan menampilkan pesan peringatan kepada pengguna untuk mempertimbangkan kembali komentarnya sebelum dikirim untuk meminimalisir penyalahgunaan komentar dalam perundungan. Sedangkan pelatihan data merupakan rancangan model pelatihan data terhadap berbagai komentar yang berpotensi mengandung *cyberbullying* menggunakan algoritma *Multinomial Naive Bayes*. Pelatihan data berfungsi untuk melatih model dari dataset komentar yang sebelumnya telah dikumpulkan untuk diproses menjadi model menggunakan *Multinomial Naive Bayes* untuk mengklasifikasikan

komentar menjadi *cyberbullying* atau *non-cyberbullying*. Model hasil pelatihan data akan digunakan sebagai sistem untuk mengidentifikasi dan mengelompokkan berbagai komentar yang berpotensi mengandung *cyberbullying* atau *non-cyberbullying*. Sementara pengujian data berfungsi untuk mengevaluasi model untuk menilai akurasi dan *error* dari model yang dikembangkan. Dari hasil evaluasi tersebut selanjutnya bisa disempurnakan model untuk mencari efektifitas sistem terbaik dalam menyaring komentar *cyberbullying* dengan berdasarkan akurasi terbesar dan *error* terkecil.

Kedua, dikembangkan sistem manajemen konten *blog* yakni sistem yang mengelola konten *blog* dan komentarnya. Sistem ini dapat digunakan pengguna dalam menyusun, mengelola, menyunting konten *blog*, dan berbagi konten *blog* ke media sosial serta memberikan dukungan komentar maupun rating dari pengguna lainnya terhadap konten *blog*. Sistem manajemen konten *blog* terdiri atas 2 bagian meliputi *front end* dan *back end*. Bagian *front end* terdiri atas fitur membaca artikel, komentar artikel, rating artikel, laporan statistik artikel, kategori artikel, *following*, pelaporan komentar *cyberbullying*, dan *share to social media*. Sementara bagian *back end* terdiri atas fitur meliputi konfigurasi pengguna, mengelola registrasi pengguna, notifikasi, mengelola artikel, mengelola *follower*, mengelola rating, mengelola artikel, menyaring komentar *cyberbullying*, mengelola kategori, dan pengaturan *layout website*.

1.3.2 Fitur-Fitur Sistem

Pada sub bab ini akan dijelaskan tentang fitur-fitur sistem penyaring komentar *cyberbullying* pada konten *blog*. Adapun fitur-fitur sistem meliputi manajemen artikel, manajemen komentar, rating artikel, registrasi dan *login*, notifikasi, *following-follower*, menyaring komentar *cyberbullying*, pelaporan komentar *cyberbullying*, kategorisasi, konfigurasi pengguna, *share to social media*, pengaturan *layout website*, laporan statistik artikel, dan *pretty URL*. Berikut ini dijelaskan masing-masing dari fitur-fitur yang dikembangkan dalam tugas akhir ini. Penjelasan selengkapnya sebagai berikut:

- Manajemen artikel

Manajemen artikel merupakan fitur untuk mengelola artikel. Fitur ini hanya bisa diakses admin dan member. Melalui fitur ini, member bisa membuat artikel, mengedit artikel, menjadwalkan penayangan artikel, dan menghapus artikel dari tampilan *front end*. Manajemen artikel dapat diakses di sisi *back end website*.

- Manajemen komentar

Manajemen komentar merupakan fitur untuk mengelola komentar dari pengguna. Fitur ini bisa diakses admin dan member. Melalui fitur ini, member bisa memberikan komentar pada artikel yang ditayangkan di *website*. Selain itu, penulis bisa menayangkan maupun menghapus komentar dari tampilan *front end*. Manajemen komentar dapat diakses di sisi *back end website*.

- Rating artikel

Rating artikel merupakan fitur untuk memberikan rating terhadap artikel yang dibaca pembaca. Fitur ini bisa diakses member dan admin. Melalui fitur ini, pembaca bisa memberikan rating berdasarkan skala penilaian yang ditentukan terhadap artikel yang mereka baca. Rating artikel dapat diakses di sisi *front end website*.

- Registrasi dan login

Registrasi adalah fitur untuk mendaftar pengguna sehingga bisa terdaftar sebagai member *web*. Melalui fitur ini, pengguna bisa mendaftarkan dirinya ke *website*. Sehingga pengguna bisa mendapatkan hak akses sebagai akun member. Untuk registrasi member dapat dilakukan melalui akun Facebook dengan menggunakan Facebook API. Fitur registrasi terdapat di sisi *front end website*.

Sedangkan *login* merupakan fitur untuk mengakses akun pengguna dengan memasukkan *username* dan *password*-nya. Sehingga pengguna bisa mengakses fitur-fitur sistem sebagai admin atau member. Fitur login juga menyediakan peresetan *password* untuk pengguna yang lupa *password*-nya. Fitur *login* terdapat di sisi *back end website*

- Notifikasi

Notifikasi merupakan fitur untuk memberitahukan kepada pengguna bahwa terdapat informasi terbaru terkait artikel yang ditulisnya. Fitur ini bisa diakses member dan admin. Informasi yang diberitahukan ini berkaitan dengan adanya berita, pesan peringatan, komentar, dan rating terbaru. Fitur ini terdapat di sisi *back end website*.

- *Following-follower*

Following-follower merupakan fitur untuk mengikuti akun member yang terdaftar dalam *website*. Sehingga pengikut bisa mendapatkan kabar terbaru terkait artikel yang dibuat oleh akun yang diikutinya. Selain itu, pengguna juga dapat mengelola *follower* yang mengikuti akun pada sistem manajemen konten. Fitur ini terdapat di sisi *back end website*.

- Penyaring komentar *cyberbullying*

Penyaring komentar *cyberbullying* merupakan fitur untuk menyaring berbagai komentar yang berpotensi mengandung *cyberbullying* pada sejumlah komentar yang diberikan pembaca pada suatu artikel. Melalui fitur ini, member bisa menyaring komentar-komentar pada artikel secara otomatis. Fitur penyaring komentar *cyberbullying* akan mengelompokkan komentar menjadi *cyberbullying* atau *non-cyberbullying*. Fitur ini bekerja di sisi *front end* dan *back end website*.

- Pelaporan komentar *cyberbullying*

Pelaporan komentar *cyberbullying* merupakan fitur untuk melaporkan berbagai komentar yang mengandung muatan *cyberbullying* secara manual. Fitur ini berguna mengantisipasi *cyberbullying* yang lolos dari penyaring komentar *cyberbullying*. Melalui fitur ini, member bisa melaporkan komentar *cyberbullying* yang tidak tersaring oleh sistem. Fitur ini terdapat di sisi *front end website*.

- Kategorisasi

Kategorisasi merupakan fitur untuk mengelompokkan artikel yang ke dalam beberapa kategori, seperti sosial-budaya, politik dan olahraga. Fitur ini bisa diakses admin. Melalui fitur ini, pembaca maupun member bisa mencari

dan menampilkan daftar artikel berdasarkan temanya. Fitur ini terdapat di sisi *front end* dan *back end website*.

- Konfigurasi pengguna

Konfigurasi pengguna merupakan fitur untuk mengatur privasi biodata pengguna. Fitur ini bisa diakses member dan admin. Melalui fitur ini, member bisa mengubah biodatanya sesuai dengan keinginannya seperti password, nama pengguna, gender, alamat, dan lain-lain. Fitur ini terdapat di sisi *back end website*.

- *Share to social media*

Share to social media merupakan fitur untuk membagikan artikel pada media sosial. Fitur ini memudahkan distribusi artikel ke media sosial. Melalui fitur ini, member bisa membagikan artikel yang dibuatnya secara luas melalui media sosial yakni Facebook dan Twitter. Fitur ini terdapat di sisi *front end website*.

- Pengaturan *template web*

Pengaturan *template web* merupakan fitur untuk menganti *template blog*. Fitur ini hanya bisa diakses admin. Disediakan dua macam *template blog* yang bisa dipilih oleh admin untuk menganti tampilan *website*. Fitur ini terdapat di sisi *back end website*.

- Laporan statistik artikel

Fitur ini berguna untuk melihat laporan statistik terkait artikel-artikel yang dipublikasikan, seperti jumlah komentar, rata-rata rating, dan jumlah pembaca. Fitur ini tersedia pada setiap artikel. Fitur ini bisa diakses pembaca artikel. Fitur ini terdapat di sisi *front end website*.

- *Pretty URL*

Fitur ini berguna untuk merapikan URL *blog* supaya *link blog* lebih mudah dibaca dan dipahami oleh pembaca. *Pretty URL* bekerja berdasarkan segmen. Sehingga URL dipisahkan menggunakan tanda segmen (/), misalnya, <https://kebaikankita.com/admin>. Adapun *pretty URL* diatur menggunakan .hta access pada *framework CI*. Dengan pemakaian *pretty UR*, URL situs menjadi bersih dan rapi.

1.4 Tahapan Penyelesaian

Perancangan sistem ini menggunakan metodologi *iterative waterfall* yang terdiri atas empat tahapan meliputi: analisis sistem, desain sistem, implementasi, pengujian. Analisis sistem menjelaskan tentang analisis kebutuhan dan spesifikasi sistem. Desain sistem menjelaskan rancangan pengembangan sistem. Implementasi menjelaskan penerapan desain sistem dalam bentuk bahasa pemrograman. Sementara pengujian menjelaskan uji coba yang dilakukan terhadap fitur-fitur sistem untuk mengukur kinerjanya. Berikut ini tahapan-tahapan penyelesaiannya:

1. Analisis Sistem

Pada tahap ini dilakukan analisis terhadap kebutuhan sistem. Untuk itu, diperlukan sejumlah literatur terkait berupa jurnal dan buku-buku yang relevan terkait pengembangan sistem manajemen konten dan pengembangan penyaring komentar *cyberbullying*. Sehingga didapatkan informasi terkait kebutuhan, fitur dan batasan dalam pengembangan sistem. Hasilnya ialah dokumen terkait kebutuhan, spesifikasi dan fitur-fitur sistem.

2. Desain Sistem

Pada tahap ini dilakukan perancangan terhadap sistem yang dikembangkan. Adapun perancangan sistemnya meliputi desain arsitektur, desain *database*, desain antar muka, dan desain prosedural. Desain arsitektur menjelaskan kerangka fitur-fitur sistem. Desain *database* menjelaskan rancangan *database* beserta detail tabel relasinya dan hubungan antar tabelnya. Desain antar muka menjelaskan rancangan tampilan sistem. Sementara desain prosedural menjelaskan rancangan algoritma yang akan diimplementasikan dalam pemrograman. Desain tersebut dikembangkan sesuai dengan kebutuhan, spesifikasi dan fitur-fitur sistem.

3. Implementasi

Pada tahap ini dilakukan implementasi kode program terhadap modul-modul yang dikembangkan. Adapun bahasa pemogramannya menggunakan bahasa PHP. Untuk kerangka kerjanya digunakan kerangka kerja CodeIgniter. Penggunaan Codeigneter dipilih untuk mempermudah dalam mengimplementasikan modul-modul dari sistem.

4. Pengujian

Pada tahap ini dilakukan pengujian terhadap sistem baik berupa pengujian fungsional maupun non fungsional, guna mengetahui bahwa sistem yang dikembangkan dapat berjalan secara baik. Adapun pengujian fungsional sistem menggunakan *black box testing*. Sedangkan pengujian non fungsionalitas meliputi *usability*, *functionality*, *reliability*, dan *efficiency* dilakukan melalui kuesioner dimana sejumlah 39 responden diminta menjawab kuesioner dengan jawaban likert skala 1 – 5 untuk mengetahui respon pengguna terhadap kualitas sistem yang dikembangkan. Sedangkan untuk mengetahui tingkat keakuratan sistem dalam mengklasifikasikan komentar menjadi *cyberbullying* digunakan teknik *4-fold cross validation* dengan $k=4$, maka didapatkan himpunan data D_1 berisi 3 *fold* yakni f_2, f_3, f_4 untuk data latih dan f_1 untuk data uji. Himpunan data D_2 berisi 3 *fold* yakni f_1, f_3, f_4 untuk data latih dan f_2 untuk data uji. Demikian seterusnya untuk himpunan data D_3 dan D_4 . Sehingga setiap *fold* pernah menjadi data uji sebanyak satu kali.. Adapun model akan dievaluasi menggunakan parameter *accuracy and error rate*.

1.5 Spesifikasi Sistem

Berikut ini dijelaskan mengenai spesifikasi sistem manajemen konten yang akan dikembangkan. Selengkapnya sebagai berikut:

- Database menggunakan MySQL.
- Menggunakan kerangka kerja CodeIgniter.
- Kategori komentar hanya terdiri atas dua kategori yakni kategori *non-cyberbullying* dan kategori *cyberbullying* dimana jenis *cyberbullying* difokuskan pada *flaming* (perang kata-kata), *harassment* (pelecehan), *denigration* (pencemaran nama baik), dan *cyberstalking* (intimidasi).
- *Dataset* diambil dari berbagai komentar dari berbagai situs portal berita dan *blog* seperti kaskus.co.id, kompas.com, kompasiana.com, detik.com, id.quora.com, youtube.com dan akun berita instagram Indonesia seperti tempo, liputan6, dan vivanews, lalu komentarnya dilabeli secara manual.

1.6 Batasan Masalah

Untuk membatasi permasalahan yang dapat ditanggani sistem supaya tidak terlalu melebar dan fokus pada masalahnya, maka dilakukan pembatasan masalah. Berikut dijelaskan tentang batasan masalah dalam pembuatan sistem manajemen konten. Selengkapnya sebagai berikut:

- Sistem manajemen konten *blog* menggunakan bahasa Indonesia dan tidak mendukung bahasa lainnya.
- Tidak mendukung sistem otentifikasi khusus.
- Sistem manajemen konten *blog* tidak melayani berbagi sumber daya foto maupun video kepada pengguna lainnya.
- Sistem manajemen konten *blog* tidak melayani afiliasi terhadap layanan periklanan tertentu.
- Sistem penyaring komentar *cyberbullying* tidak menangani komentar bermuatan *emoticon*.
- Sistem penyaring komentar *cyberbullying* hanya bisa menangani komenter berbahasa Indonesia.

1.7 Sistematika Pembahasan

Sistematika pembahasan terdiri atas tujuh bab meliputi: pendahuluan, tinjauan pustaka, dataset dan *preprocessing*, *Multinomial Naive Bayes*, analisis sistem, desain sistem, implementasi, pengujian, dan penutup. Berikut ini dipaparkan sistematika laporan sebagai berikut:

- BAB I : PENDAHULUAN
Pada Bab I dijelaskan latar belakang dalam pengembangan sistem penyaring komentar *cyberbullying* pada konten *blog*. Untuk itu dijelaskan tentang latar belakang, tujuan, ruang lingkup, tahapan penyelesaian, dan sistematika pembahasan.
- BAB II : TINJAUAN PUSTAKA
Pada Bab II dijelaskan tentang teori-teori yang digunakan dalam pengembangan sistem penyaring komentar *cyberbullying*.

Untuk itu dijelaskan tentang *cyberbullying*, *preprocessing*, Algoritma *Stemming* Nazief dan Adriani, Codeigneter, *Multinomial Naive Bayes*, *K-Fold Cross Validation*, ukuran evaluasi model, dan *tool* dan *library* penunjang.

- BAB III : DATASET DAN PREPROCESSING

Pada bab III dijelaskan tentang bagaimana proses pembuatan dataset dan bagaimana melakukan *preprocessing* komentar yang selanjutnya dipakai sebagai *dataset* model.

- BAB IV : MULTINOMONAL NAIVE BAYES

Pada Bab IV dijelaskan tentang bagaimana menggunakan *Multinomial Naive Bayes* untuk pengembangan model penyaring komentar *cyberbullying* pada konten *blog* beserta perhitungannya.

- BAB V : ANALISIS SISTEM

Pada Bab V dijelaskan tentang analisis sistem manajemen konten blog. Pada bab ini akan dijelaskan bagaimana melakukan *fact finding*, menganalisis sistem lama, mengidentifikasi permasalahan yang dihadapi sistem lama, menganalisis kebutuhan sistem, batasan sistem, hingga merancang *data flow diagram* (DFD) untuk sistem baru.

- BAB VI : DESAIN SISTEM

Pada Bab VI dijelaskan tentang bagaimana mendesain sistem penyaring *cyberbullying* pada konten *blog*. Pada bab ini akan dijelaskan tentang desain arsitektur sistem, desain tabel, dan desain antar muka, dan desain prosedural.

- BAB VII : IMPLEMENTASI

Pada Bab VII dijelaskan tentang bagaimana mengimplementasikan sistem dalam kode menggunakan bahasa PHP pada kerangka kerja Codeigneter. Pada bab ini akan dijelaskan tentang beberapa segmen program.

- BAB VIII: PENGUJIAN

Pada BAB VIII dijelaskan bagaimana pengujian yang dilakukan untuk menguji sistem. Pada bab ini dijelaskan ujicoba sistem *cyberbullying* meliputi evaluasi model dan evaluasi sistem.

- BAB IX : PENUTUP

Pada BAB IX dijelaskan tentang kesimpulan dan saran terkait pengembangan sistem manajemen konten blog dan sistem penyaring komentar *cyberbullying*.

BAB II

TINJAUAN PUSTAKA

Pada Bab II dijelaskan tentang berbagai teori penunjang untuk pengembangan sistem. Dalam Bab ini akan dijelaskan berbagai teori, metode, *tool* dan *library* yang dipakai dalam tugas akhir. Adapun teori penunjangnya meliputi *cyberbullying*, *preprocessing*, algoritma, stemming, Codeigneter, Teorema Bayes, ukuran evaluasi model, *K-Fold Cross Validation*, ukuran evaluasi *website*, dan *tool* dan *library* pendukung untuk pengembangan sistem. Berikut penjelasan selengkapnya:

2.1 Cyberbullying

Menurut Williard (2005), *cyberbullying* merupakan perlakuan kejam yang dilakukan dengan sengaja kepada orang lain dengan mengirimkan atau mengedarkan bahan yang berbahaya atau terlibat dalam bentuk-bentuk agresi sosial menggunakan internat atau teknologi digital lainnya.¹ Hinduja dan Patchin (2009) mendefinisikan *cyberbullying* sebagai perbuatan merugikan yang dilakukan dengan sengaja dan berulang-ulang melalui komputer, telepon genggam, dan perangkat elektronik lainnya. Perbuatan ini dilakukan dengan mengirimkan pesan, mengancam atau memermalukan seseorang melalui pesan teks, surel atau email, menulis komentar menghina seseorang di website atau media sosial, mengancam atau mengintimidasi seseorang melalui berbagai bentuk daring atau dalam jaringan.² Menurut endcyberbullying.org, *cyberbullying* adalah tindakan yang menimbulkan kerugian berulang kali secara sengaja. Dalam hal ini yang membedakan antara kecerobohan dengan *cyberbullying* ialah bahwa dalam *cyberbullying* senantiasa melibatkan niat pelaku dalam memainkan peran utama dimana pelaku akan selalu terlibat dalam situasi membuat kesalahan terhadap

¹ Ranny Rastati, *Bentuk Perundungan Siber di Media Sosial dan Pencegahannya Bagi Korban dan Pelaku*, Jurnal Sosioteknologi, vol. 15, No. 2, Agustus 2016, pp. 176-178.

² Sameer Hinduja dan Justin W. Patchin, 2020. *Cyberbullying: Identification, Prevention, and Response*, Cyberbullying.org, diakses tanggal 21 Desember 2020.

seseorang secara sengaja dengan memanfaatkan teknologi komunikasi. Para pelakunya biasanya merupakan remaja yang menggunakan berbagai macam teknologi komunikasi untuk menindas orang lain, dengan telepon seluler melalui SMS, situs media sosial, situs mengunggah video seperti YouTube. Tidak ada batasan tentang apa yang telah dilakukan atau bagaimana.³ Sementara itu, Menurut Kowalski sebagaimana dikutip jurnal El Chirs Natalia mendefinisikan cyberbullying sebagai kegiatan perundungan yang terjadi pada *instant messaging*, internet, dan media sosial.⁴

Dalam hal ini terdapat perbedaan antara *cyberbullying* dengan perundungan tradisional. Pertama, dalam banyak kasus para korban tidak mengetahui siapa pelaku yang melakukan *cyberbullying* terhadap mereka dan tidak pula mengetahui alasan mengapa mereka dirundung. *Cyberbullying* bekerja di balik layar komputer bahkan mungkin menggunakan identitas palsu (anonim) sehingga para pelaku sulit ditemukan hingga memberi mereka rasa kendali lebih lanjut atas situasi dimana mereka dapat merundung para korban secara leluasa. Kedua, tindakan *cyberbullying* biasanya bersifat viral. Ketika korban mengalami *cyberbullying*, orang lain pada media yang sama dapat mengetahuinya secara instan dan orang lain itu akan menyebarkan berita, mempermalukan korban bahkan lebih dari yang telah dilakukan oleh pelaku awal *cyberbullying*. Ketiga, *cyberbullying* seringkali lebih mudah dan lebih kejam menggunakan teknologi informasi daripada perundungan tradisional yang biasanya dilakukan secara tatap muka. Hal ini karena pelaku *cyberbullying* tidak harus melihat reaksi korban. Beberapa remaja tidak menyadari kerugian yang mereka timbulkan dari tindakan *cyberbullying* karena mereka tidak tahu reaksi korbannya, sehingga mereka lebih leluasa untuk menyakiti korbannya tanpa keraguan. Keempat, banyak orang dewasa tidak mempunyai cukup pengetahuan untuk melacak apa yang dilakukan para remaja tatkala mereka menggunakan teknologi informasi. Banyak orang dewasa bahkan setelah

³ <https://www.endcyberbullying.org/definition-what-does-cyberbullying-exactly-mean/>, diakses tanggal 21 Desember 2020.

⁴ El Chirs Natalia, Desember 2016, *Remaja, Media Sosial dan Cyberbullying*, Jurnal Ilmiah Komunikasi, Vol. 5, No. 2, p. 129.

mengetahui pelaku *cyberbullying*, mendapati diri mereka tidak siap untuk memberikan tanggapan lebih lanjut.⁵

Cyberbullying berdampak negatif terhadap kehidupan remaja. Para korban biasanya merasa tertekan, sedih, marah, dan frustasi. *Cyberbullying* juga berdampak negatif terhadap harga diri, masalah keluarga, kesulitan akademik, kekerasan di sekolah, dan berbagai perilaku menyimpang lainnya. Bahkan sejumlah laporan menyebutkan para korban memutuskan untuk bunuh diri.⁶

Menurut endcyberbullying.org, sebagaimana perundungan secara tradisional memiliki berbagai jenis perundungan seperti pelecehan verbal dan kekerasan fisik, terdapat setidaknya 5 jenis *cyberbullying* meliputi:⁷

- *Harassment* yakni perundungan daring dengan mengirimkan pesan yang menyinggung dan berbahaya kepada individu atau kelompok dan sering diulang beberapa kali. *Cyberstalking* adalah salah satu bentuk pelecehan yang melibatkan pesan-pesan kasar dan ancaman terus-menerus, dan dapat menyebabkan pelecehan fisik di dunia nyata.
- *Flaming*, mirip dengan *harassemement*, tetapi merujuk pada pertarungan verbal secara daring yang dipertukarkan melalui email, SMS, atau grup media sosial. Flaming adalah jenis *cyberbullying* yang dilakukan di muka publik yang sering kali mengarahkan bahasa kasar, atau penggunaan gambar (meme) ke orang tertentu.
- *Exclusion* merupakan perundungan daring secara sengaja menyisihkan dan mengeluarkan seseorang dari grup media sosial seperti chatting dan forum. Kemudian grup tersebut meninggalkan komentar jahat dan melecehkan yang mereka pilih.
- *Outing* merupakan tindakan perundungan dimana pelaku perundungan membagikan informasi, gambar, atau video pribadi tentang seseorang secara

⁵ <https://www.endcyberbullying.org/cyberbullying-vs-traditional-bullying/>, diakses tanggal 21 Desember 2020.

⁶ Hinduja dan Patchin, Loc. Cit.

⁷ <https://www.endcyberbullying.org/5-different-types-of-cyberbullying/>, diakses tanggal 21 Desember 2020.

publik. Seseorang menjadi “outed” ketika informasinya telah disebarluaskan melalui internet.

- *Masquerading* merupakan perundungan daring di mana pelaku perundungan membuat identitas palsu untuk melecehkan seseorang secara anonim. Selain membuat identitas palsu, pelaku intimidasi dapat menyamar sebagai orang lain untuk mengirim pesan berbahaya kepada korban.

Sementara itu, menurut Willard sebagaimana dikutip dalam jurnal Ranny Rastati, terdapat 7 jenis *cyberbullying* meliputi:⁸

- *Flaming* (pertengkar daring) yakni perang kata-kata di dunia maya dengan menggunakan bahasa yang mengandung amarah, vulgar, mengancam, dan merendahkan. Pertengkar daring biasanya terjadi di surel, ruang obrol, dan media sosial.
- *Harassment* (pelecehan) yakni perundungan daring yang menggunakan kata-kata kasar, menyerang, dan melecehkan seseorang secara berulang-ulang.
- *Denigration* (pencemaran nama baik) merupakan perundungan daring yang dilakukan dengan cara mengubur keburukan seseorang di internet seperti menuliskan postingan, komentar hinaan yang bohong, gosip yang kejam, dan rumor tentang seseorang untuk merusak reputasinya.
- *Impersonation* (akun palsu) yakni perilaku berpura-pura menjadi orang lain dan mengirimkan pesan-pesan atau status yang tidak baik dengan maksud merusak reputasi seseorang.
- *Outing and trickery* (tipu daya) merupakan perilaku memperdaya seseorang untuk melakukan sesuatu yang memalukan, membuka informasi memalukan tentang dirinya sendiri berupa teks, foto, dan video untuk disebar secara luas di internet. Tidak jarang perundungan daring ini dilanjutkan dengan pemerasan disertai ancaman agar korban memberikan apa yang diinginkan pelaku.
- *Exclusion* (pengucilan) yakni perundungan daring dengan cara mengucilkan atau mengeluarkan seseorang dari grup sosial media secara sengaja. Kasus ini

⁸ Rastati, Loc. Cit.

banyak terjadi di kalangan masyarakat umum berupa peer-group atau kelompok pertemanan.

- *Cyberstalking* (penguntitan daring) merupakan perilaku berulang kali mengirimkan ancaman membahayakan atau pesan-pesan yang mengintimidasi dengan menggunakan komunikasi elektronik dengan tujuan membuat orang itu tidak nyaman dan merasa khawatir atas keselamatannya.

Menurut Ranny Rastati (2016), selain tujuh bentuk *cyberbullying* yang diajukan oleh Willard, dalam konteks Indonesia, *cyberbullying* memiliki karakter sendiri. Dalam hal ini, *cyberbullying* yang terjadi di Indonesia selain ditujukan kepada individu, ditemukan pula tiga objek *cyberbullying* meliputi (1) perundungan daring terhadap lokasi seperti mengejek kota atau daerah tertentu melalui media sosial; (2) perundungan daring terhadap keagamaan seperti mengejek perayaan agama tertentu melalui media sosial; (3) dan perundungan daring terhadap institusi atau profesi seperti mengejek DPR melalui media sosial. Adanya tiga bentuk lain itu disebabkan oleh terbatasnya saluran komunikasi dari masyarakat ke birokrat dan kritik yang ditujukan terhadap pihak tersebut. Kritik masyarakat yang selama ini disampaikan melalui rubrik opini pembaca koran kemudian bermigrasi menuju ranah media sosial. Hal ini membuka ruang komunikasi yang lebih interaktif dan memungkinkan pertukaran informasi yang masif dan cepat. Namun, sayangnya praktiknya tidak diikuti oleh pembentukan kearifan dalam beretika di dunia daring atau netiket, sehingga membuka ruang bagi netizen untuk melakukan perundungan daring terhadap lokasi, keagamaan, dan institusi tertentu.⁹ Dalam tugas akhir ini, jenis *cyberbullying* difokuskan pada *flaming*, *harassment*, *denigration*, *impersonation*, dan *cyberstalking* dengan objek perundungan daring baik secara individu, lokasi, keagamaan maupun institusi tertentu.

2.2 Text Preprocessing

Text preprocessing adalah proses mengubah bentuk data yang belum memiliki struktur menjadi data yang terstruktur ssuai dengan kebutuhan untuk

⁹ Ibid, pp. 178-180.

proses minning yang lebih lanjut. Sebelum melakukan proses klasifikasi, data yang terkumpul perlu dilakukan *preprocessing* terlebih dulu dengan cara menghilangkan *noise* agar data yang diolah sesuai dengan kebutuhan. Menurut Aurangzeb (2010) sebagaimana dikutip tesis Fachrian Anugerah, mereduksi data penting dilakukan dalam klasifikasi teks karena data tidak relevan dan berlebihan seringkali menurunkan kinerja algoritma baik kecepatan maupun akurasi klasifikasi dan kecenderungan mengurangi *overfitting*. Pada proses *text preprocessing*, data komentar mentah diproses melalui tahapan *cleaning*, *case folding*, *tokenizing*, normalisasi bahasa gaul, *stemming*, dan *stopword-list* serta *n-gram*. Berikut penjelasan tiap tahapannya:¹⁰

1. Case Folding

Case folding adalah proses mengubah huruf dalam dokumen menjadi huruf kecil. Dokumen mengandung beragam variasi bentuk huruf sampai tanda baca. Variasi huruf ini harus diseragamkan dan tanda bacanya harus dihilangkan. Hanya huruf ‘a’ sampai dengan ‘z’ yang diterima. Karakter lainnya dihilangkan dan dianggap sebagai delimiter. Pada tahap *case folding*, teks *komentar* diubah dari karakter kapital menjadi karakter kecil, contoh: Mengajar → mengajar, MELEMAH → melemah, MenJadi → menjadi.

2. Cleaning

Cleaning (pembersihan derau) yakni proses menghilangkan dokumen dari mention, hastag, link, emoticon, dan karakter lainnya yang tak berguna. Pada tahap *cleaning*, teks komentar dibersihkan dari *hastag* (#), *mention* (@), *link*, *emoticon*, dan karakter atau tanda baca yang tak bernilai lainnya seperti titik (.), koma (,), titik dua (:), titik koma (;), dan sebagainya. Dengan demikian, data menjadi lebih bersih dari karakter-karakter yang tidak penting. Misalnya, kalimat “@aniesbaswedan adalah gubernur DKI” di-*cleaning* menjadi “aniesbaswedan adalah gubernur DKI” dimana tanda *mention* dihilangkan dari kalimat supaya kalimat menjadi bersih.

¹⁰ Fachrian Anugerah, *Perbaikan Kinerja Praproses Karakter Berulang dalam Mengenali Kata Pada Klasifikasi Sentimen Berbahasa Indonesia*, Surabaya: Institut Teknologi Sepuluh Nopember, 2017, pp.26-31.

3. Normalisasi Bahasa Gaul

Normalisasi bahasa gaul (konversi kata tidak baku) adalah proses mengubah kata-kata tak lazim berupa kata-kata gaul menjadi kata-kata formal berbahasa Indonesia. Umumnya, tidak semua komentar pada suatu artikel menggunakan bahasa formal. Pembaca biasanya juga memakai bahasa gaul, seperti nggak, gue, loe, dll. Kata-kata yang tak lazim ini perlu diseragamkan melalui normalisasi bahasa menjadi bahasa formal berbahasa Indonesia. Kata-kata tak lazim ini biasanya berupa singkatan populer maupun kata-kata gaul. Contoh singkatan populer meliputi : qt → kita, sy → saya, q → aku, jk → jika, btw → ngomong-ngomong. Contoh kata-kata gaul meliputi loe → kamu, gue → aku, bro → saudara.

4. *Stemming*

Stemming adalah proses mengubah kata menjadi kata dasar. Pada umumnya kata dalam dokumen memiliki variasi kombinasi imbuhan kata yang beragam, seperti imbuhan awalan, akhiran, sisipan, dan kombinasi. Penyeraman kata menjadi kata dasarnya juga berguna mengurangi ukuran model sehingga model menjadi lebih ringan. Kata-kata tersebut perlu diseragamkan menjadi kata dasar supaya seragam dan mengurangi kompleksitas kata. Misalnya: mengajar → ajar, melemah → lemah, menjadi → jadi, dan sebagainya.

5. *Stopword*

Stopword-list adalah kata-kata yang sering muncul dan tidak memiliki arti. Daftar kata *stopwords* yakni kumpulan kata yang tidak memiliki arti dan disimpan dalam kamus khusus *stopwords*, seperti ‘yang’, ‘bahkan’, ‘akhirnya’, ‘di’, dll. Dengan dibuangkan *stopwords*, ukuran kosakata menjadi berkurang sehingga hanya kata-kata penting dan memiliki bobot tinggi yang terdapat dalam dokumen. Pada tahap *stopword-list*, kata-kata yang tidak penting dibuang dari daftar kata, misalnya kata “yang”, “dimana”, “mengapa”, “yaitu”, “yakni”, dan sebagainya.

6. Tokenizing

Tokenizing adalah proses pemotongan dokumen menjadi kata-kata setelah menjadi proses *filtering*. Hasil pemotongan kata-kata tersebut dijadikan kumpulan kata dan membentuk daftar kata. Hasil pemotongan tersebut dikenal dengan istilah token. Misalnya, hasil *tokenizing* dari kalimat “korupsi menyengsarakan masyarakat luas” terdiri atas 4 token yakni : korupsi, menyengsarakan, masyarakat, luas.

7. N-gram

N-gram merupakan proses yang secara luas digunakan dalam pengolahan teks dan bahasa. *N-gram* merupakan kumpulan kata yang diberikan dalam sebuah paragraf dan ketika menghitung *n-gram* biasanya dilakukan dengan menggerakkan satu kata maju ke depan. *N-gram* memberikan probabilitas atau kemungkinan kata-kata berikutnya yang mungkin bisa digunakan dalam pengabungan pada keseluruhan kalimat. Adapun tipe *n-gram* yang biasanya digunakan seperti *unigram*, *bigram*, *trigram*. Jika diasumsikan X adalah jumlah kata dalam suatu kalimat K, maka jumlah n-gram dari kalimat K adalah $N\text{gram-K} = X - (N-1)$.¹¹ Misalnya, kalimat K “korupsi menyengsarakan masyarakat luas” memiliki 3 *bigram* yang terdiri atas : (1) korupsi menyengsarakan, (2) menyengsarakan masyarakat, (3) masyarakat luas.

2.3 Algoritma Stemming Nazief dan Adriani

Salah satu algoritma *stemming* bahasa Indonesia yang cukup terkenal ialah algoritma *stemming* yang dibuat oleh Bobby Nazief dan Mirna Adriani. Algoritma ini dikembangkan berdasarkan aturan morfologi Bahasa Indonesia yang mengelompokkan imbuhan menjadi *prefix* (awalan), *infix* (sisipan), dan *suffix* (akhiran) dan *prefix-suffix* (gabungan awalan-akhiran). Algoritma ini menggunakan kamus kata dasar dan mendukung *recording*, yakni penyusunan kembali kata-kata

¹¹ Wahyu Candra Indhiarta, *Penggunaan N-Gram Pada Analisis Sentimen Pemilihan Kepala Daerah Jakarta Menggunakan Algoritma Naive Bayes*, Surakarta: Universitas Muhammadiyah Surakarta, 2017, p.4.

yang mengalami proses *stemming* berlebih. Algoritma ini menggunakan morfologi imbuhan sebagai berikut:¹²

- *Inflection suffixes* merupakan kumpulan akhiran (suffiks) yang tidak mengubah kata dasar. Misalnya kata ‘duduk’ + ‘-lah’ → ‘duduklah’. *Inflection suffixes* terbagi menjadi:
 - *Particles* (P) termasuk ‘-lah’ dan ‘-kah’. Contoh, makanlah, maukah.
 - *Possessive pronouns* (PP) termasuk ‘-ku’, ‘-mu’ dan ‘-nya’. Contoh, diriku, dirimu, dirinya.
- *Derivation suffixes* (DS) merupakan kumpulan akhiran (*suffixes*) yang langsung menempel pada kata dasar. Misalnya, kata dasar ‘lapor’ ditambah *derivation suffix* ‘-kan’ menjadi ‘laporkan’. Setelah itu ditambah *inflection suffix* ‘-lah’ menjadi ‘laporkanlah’.
- *Derivation prefixes* (DP) merupakan himpunan awalan (*prefixes*) yang menempel langsung pada kata dasar maupun terhadap kata yang telah mempunyai sampai dua *derivation prefixes*. Misalnya, *derivation prefixes* ‘mem-’ dan ‘per-’ + ‘-perindahkannya’ menjadi memperindahkannya. Di bahwa ini merupakan urutan pemakaian imbuhan sebagai *inflections* dan *derivations*:

$$\left[DP + [DP + [DP +]] \right] \text{kata dasar} [[+DS][+PP][+P]] \quad \dots (2.1)$$

Dalam aturan *stemming* di atas, kurung siku menandakan bahwa imbuhan bersifat optional. Hal ini berarti imbuhan tersebut bisa jadi terdapat dalam suatu kata atau bisa jadi tidak ada dalam suatu kata.

Algoritma *stemming* Nazief dan Adriani mempunyai 5 langkah kerja. Adapun urutan langkah-langkah algoritma *stemming* Nazief dan Adriani dijelaskan sebagai berikut:¹³

¹² J. Asian & H.E. Williams & Tahaghoghi, *Stemming Indonesian*, Proceedings of the Twenty-eighth Australasian conference on Computer Science, 2004, Australian Computer Society Inc, p.2.

¹³ PM Prihatini, dkk, *Stemming Algoritm for Indonesian Digital News Text Precossing*, International Journal of Engineering and Emerging Technology, Vol.2, No.2, Juli – Desember 2017, pp.3-4.

1. Algoritma mengecek kata yang belum di-*stemming* di kamus kata dasar. Jika kata tersebut berhasil ditemukan, maka algoritma berhenti dan kata itu dianggap sebagai kata dasar. Tetapi jika algoritma tidak menemukan kata tersebut di kamus kata dasar, maka lanjut ke langkah selanjutnya.
2. Algoritma menghapus *inflection suffixes* (“-lah”, “-kah”, “-ku”, “-mu”, “-nya”). Jika penghapusan nya berhasil dan *suffixes* adalah partikel (“-lah” atau “-kah”), algoritma menghapus akhiran *possessive pronoun* (‘-ku’, ‘-mu’ atau ‘-nya’).
3. *Derivation suffixes* (“-i” atau “-an”) dihapus. Jika penghapus berhasil, algoritma berlanjut ke langkah keempat. Jika gagal, maka lakukan:
 - Jika *derivation suffixes* adalah “-an” dan karakter terakhir kata adalah “-k”, maka algoritma menghapus karakter “-k”. Kemudian lanjut ke langkah keempat. Tetapi jika gagal, maka lanjut ke langkah 3b.
 - Algoritma mengembalikan *suffixes* (“-i”, “-an”, “-kan”) yang terhapus, lalu lanjut ke langkah keempat.
4. Algoritma menghapus *derivation prefix*, yang terdiri dari beberapa langkah meliputi:
 - Jika penghapusan *suffix* berhasil di langkah ketiga, algoritma memeriksa *prefix-suffix* yang tidak diizinkan. Jika algoritma menemukannya, algoritma selesai. Hika algoritma tidak menemukannya, maka lanjut ke langkah 4b.
 - Jika *prefix* mirip dengan *prefix* sebelumnya, algoritma selesai.
 - Jika penghapusan *derivation prefix* telah terjadi selama 3 kali, maka algoritma selesai.
 - Jika algoritma mengecek jenis *derivation prefix* dan menghapus *prefix*.
 - Jika kata dasar ditemukan, algoritma selesai. Sebaliknya, langkah keempat diulangi kembali untuk menghapus *prefix* kedua.
 - Algoritma melakukan proses *recording*, bergantung pada jenis *prefix*.
5. Jika semua langkah di atas gagal, algoritma berhenti dan mengembalikan kata awalnya dan proses selesai.

Algoritma *stemming* Nazief dan Adriani dikembangkan kembali oleh Asian Jelita dijelaskan sebagai berikut:¹⁴

1. Memakai kamus kata dasar yang lebih lengkap.
2. Menambah aturan untuk penanganan kata jamak, misalnya kata “buku-buku”, maka stemming dikembalikan menjadi “buku”. Namun kehati-hatian harus diibeirkan pada beberapa kata dengan tanda hubung “-” seperti “bolak-balik”, “berbalas-balasan”, “seolah-olah”. Sebagai contohnya, kata “berbalas-balasan” memiliki kata dasar “balas”, sehingga *stemming* bisa dilakukan dengan mengembalikan kata awal sebelum tanda hubung atau kata akhir setelah tanda hubung. Sebaliknya, kata “bolak-balik” tidak memiliki kata dasar yang sama, sehingga *stemming* kembalikan sebagai kata yang sama sebagai kata “bolak-balik”.
3. Menambah *prefix* dan *suffix* dan penambahan aturan sebagai berikut:
 - Penambahan partikel (*inflectin suffix*) “pun”, seperti untuk kata “siapapun”.
 - Untuk tipe *prefix* "ter-" dimodifikasi kondisinya, sehingga baris 4 dalam tabel 2.2 diatur tipe ke "ter-", alih-alih "none". Aturan ini mendukung kasus seperti kata "terpercaya" yang memiliki kata dasar "percaya".
 - Untuk *prefix* tipe “pe-” dimodifikasi kondisinya sehingga kata “pekerja” dan “peserta” memiliki *prefix* bertipe “pe-”.
 - Untuk *prefix* bertipe “mem-” dimodifikasi kondisinya sehingga kata dengan permulaan *prefix* “memp-” adalah bertipe “mem-”.
 - Untuk *prefix* bertipe “meng-” dimodifikasi kondisinya sehingga kata dengan permulaan *prefix* “mengk-“ adalah “meng-”.
4. Menyesuaikan prioritas aturan:
 - Jika kata memiliki *prefix* dengan “ber-“ dan *suffix* dengan *inflection suffix* “-lah”, diusahakan untuk dihapus *prefix*-nya sebelum *suffix*, seperti untuk kata “bermasalah” dimana kata dasarnya adalah “masalah”.

¹⁴ J. Asian, H.E. Williams, Tahaghoghl, Loc. Cit., p.5.

- Jika kata memiliki *prefix* dengan “ber-” dan *suffix* dengan *derivation suffix* “-an”, diusahakan untuk dihapus *prefix* sebelum *suffix*, seperti untuk kata “berbadan” dimana kata dasarnya adalah “badan”.
- Jika kata memiliki *prefix* dengan “men-” dan *suffix* dengan *derivation suffix* “-i”, diusahakan untuk dihapus *prefix* sebelum *suffix*, seperti untuk kata “menilai” dimana kata dasarnya adalah “nilai”.
- Jika kata memiliki *prefix* dengan “di-” dan *suffix* dengan *derivation suffix* “-i”, diusahakan untuk dihapus *prefix* sebelum *suffix*, seperti untuk kata “dimulai” dimana kata dasarnya adalah “mulai”.
- Jika kata memiliki *prefix* dengan “pe-” dan *suffix* dengan *derivation suffix* “-i”, diusahakan untuk dihapus *prefix* sebelum *suffix*, seperti untuk kata “petani” dimana kata dasarnya adalah “tani”.
- Jika kata memiliki *prefix* dengan “ter-” dan *suffix* dengan “-i”, diusahakan untuk dihapus *prefix* sebelum *suffix*, seperti untuk kata “terkendali” dimana kata dasarnya adalah “kendali”.

Tabel 2.1
Kombinasi Prefix (Awalan) dan Suffix (Akhiran) yang Tidak Diperbolehkan

<i>Prefix</i>	<i>Suffix Disallowed</i>
be-	-i
di-	-an
ke-	-i, -kan
me-	-an
se-	-i, -kan
te-	-an

Pada tabel 2.1 terdapat satu pengecualian, kata dasar “tahu” diperbolehkan untuk kombinasi awalan “ke-“ dan akhiran “-i”. Sedangkan untuk penentuan tipe awalan untuk kata dengan awalan “te-” mengikuti aturan pada tabel 2.2:

Tabel 2.2
Penentuan Tipe Awalan Untuk Kata Dengan Awalan “Te-”

<i>Following Characters</i>				<i>Prefix Type</i>
Set 1	Set 2	Set 3	Set 4	
“-r”	“-r”	-	-	none

**Tabel 2.2
(Lanjutan)**

<i>Following Characters</i>				<i>Prefix Type</i>
Set 1	Set 2	Set 3	Set 4	
“-r”	vowel	-	-	ter-luluh
“-r”	Not (“-r” or vowel)	“-er-”	vowel	ter
“-r”	Not (“-r” or vowel)	“-er-”	Not vowel	none
“-r”	Not (“-r” or vowel)	Not “-er”	-	ter
Not (“-r” or vowel)	“-er”	vowel	-	none
Not (“-r” or vowel)	“-er”	Not vowel	-	te

Keterangan: Jika awalan “te-” tidak sesuai dengan salah satu dari aturan pada tabel 2.2, maka dikembalikan ke “none”. Aturan yang sama berlaku untuk “be-”, “me-” dan “pe-”.

**Tabel 2.3
Penentuan Awalan dari Jenis Awalan**

<i>Prefix Type</i>	<i>Prefix to be Removed</i>
Di	di-
Ke	ke-
Se	se-
Te	te-
Ter	ter-
ter-luluh	ter-

Adapun kelebihan algoritma Nazief dan Adriani ialah memperhatikan kemungkinan adanya partikel-partikel yang mungkin mengikuti suatu kata berimbahan. Selain itu, algoritma ini menggunakan kamus kata dasa dan mendukung *recording*, yakni penyusunan kembali kata-kata yang mengalami proses *stemming* berlebih.¹⁵ Untuk implementasi algoritma Nazief dan Adriani pada bahasa PHP bisa dilihat selengkapnya pada tautan ini.¹⁶

¹⁵ I Putu Merta Wirayasa, dkk, *Algoritma Bastal: Adaptasi Algoritma Nazief dan Adriani untuk Stemming Teks Bahasa Bali*, Jurnal Nasional Pendidikan Teknik Informatika, Vo.8, No.1, Maret 2019, p.64.

¹⁶ Liyantanto, *Stemming Bahasa Indonesia dengan Algoritma Nazief dan Adriani*, <https://liyantanto.wordpress.com/2011/06/28/stemming-bahasa-indonesia-dengan-algoritma-nazief-dan-andriani/>, diakses tanggal 24 Desember 2020.

2.4 Codeigniter

CodeIgniter (CI) merupakan PHP *framework* yang dikembangkan oleh Rick Ellis dari Ellis Lab. CI dirancang sebagai *framework* yang ringan dan mudah digunakan. Bahkan CI sudah diakui pencipta PHP, Ramus Lerdorf, sebagai *web framework* yang mudah, cepat, dan handal. CI dirilis pertama kali pada tanggal 28 Februari 2006 oleh Ellis Lab. Kemudian kepemilikannya berpindah ke tangan British Columbia Institute of Technology pada bulan Oktober 2014. Kini CI telah merilis Codeigniter 4.0 pada tanggal 24 Februari 2020.¹⁷

CI merupakan *web framework* bagi orang yang ingin membuat aplikasi web menggunakan bahasa PHP. Tujuannya adalah membuat pengembangan proyek menjadi lebih cepat dibandingkan dengan menulis kode dari awal. CI menyediakan sekumpulan *library* untuk tugas-tugas yang sering dilakukan dan sangat mudah mengakses *library* yang tersedia di CI. Dengan CI, pengembangan proyek web menjadi lebih mudah dan meminimalisir jumlah kode yang akan dituliskan.¹⁸

Sebagai *web framework* populer, CI mempunyai berbagai keunggulan sebagaimana dijelaskan pada dokumentasinya meliputi:¹⁹

- *Free*, karena berada di bawah lisensi *open source*, kita dapat melakukan apapun dengan CI.
- *Light weight*, sistem inti CI memerlukan *library* yang sedikit. Berbeda dengan *web framework* lainnya yang membutuhkan banyak sumber daya tambahan. *Library* tambahan akan digunakan ketika *request* secara dinamis, membuat sistem yang akan dibangun menjadi efisien dan cukup cepat.
- *Fast*, menurut dokumentesinya, performa CI terbukti cepat dibandingkan *web framework* lainnya.
- Menggunakan MVC, dengan menggunakan *Model-View-Controller* sistem web menjadi dipisahkan antara *logic* dan presentasinya. Hal ini membuat cocok

¹⁷ Anton Subagia, *Kolaborasi Codeigniter dan Ajax dalam Perancangan CMS*, Jakarta: PT Elex Media Komputindo, 2018, p.1.

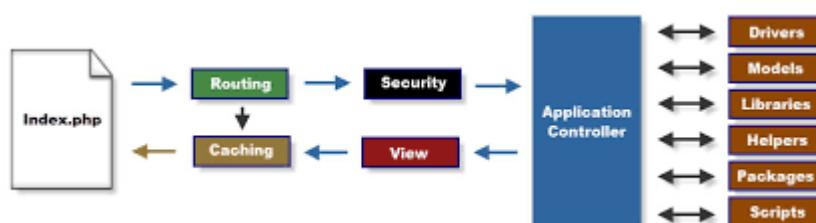
¹⁸ Ibid, p. 2.

¹⁹ Ibid.

dan bagus untuk proyek yang fokus. Desainer fokus pada template web dan programmer fokus pada pembangunan *logic* dari aplikasi web.

- Menghasilkan URL yang bersih, URL yang dihasilkan CI bersih dan ramah terhadap *search engine*, karena CI menggunakan pendekatan *segment-based* dibandingkan *query string*.
- *Pack a punch*, CI hadir dengan berbagai *library* yang akan membantu tugas-tugas dalam pengembangan website yang umum digunakan seperti mengakses database, mengirim email, validasi form, dsb.
- *Extensible*, CI memungkinkan pengembang untuk menambahkan *library* dan *helper*.
- *Thoroughly documented*, hampir semua fitur, *library*, dan *helper* yang ada di CI telah terdokumentasi secara lengkap dan tersusun baik.
- Mempunyai komunitas yang besar dan ramah, dengan bergabung di komunitas CI membantu para pengembang mulai dari pemula hingga mahir untuk berbagi ilmu pengetahuan, sehingga semakin memudahkan pengembangan aplikasi web.

CI menggunakan pendekatan MVC yang bertujuan memisahkan antara logika dan presentasi. Konsep ini membuat pengembangan dapat membagi pekerjaan dimana desain bekerja pada template. Sedangkan programmer bekerja pada logika aplikasi. Berikut ini disajikan skema kerja CI:²⁰



Gambar 2.1
Cara Kerja CodeIgniter

²⁰ Ibid, p.6.

Berikut dijelaskan cara kerja CodeIgniter sebagai berikut:

1. Index.php bertindak sebagai controller terdepan dan menginisiasi resource yang diperlukan untuk menjalankan CI.
 2. Router memeriksa *HTTP request* untuk menentukan apa yang harus dikerjakan.
 3. Jika cache file ada, maka akan ditampilkan langsung, dengan melewati eksekusi normal sistem.
 4. Sebelum memuat controller, *HTTP request* akan memeriksa apa yang di-submit user dan memfilter untuk keamanan.
 5. Controller memuat model, core libraries, plugin, helper dan resource lainnya untuk memproses permintaan tertentu.
 6. View ditampilkan di web browser sesuai proses yang dikerjakan controller. Jika caching dijalankan, view akan di-cache terlebih dahulu agar dapat ditampilkan di-request selanjutnya.

2.5 Teorema Bayes

Teorema bayes menggunakan teorema kuno warisan abad ke-18, yang ditemukan oleh Thomas Bayes (1702 – 1761). Teorema bayes digunakan untuk menghitung probabilitas terjadinya suatu peristiwa berdasarkan pengaruh yang didapatkan dari hasil observasi peristiwa sebelumnya. Teorema bayes menyempurnakan teorema probabilitas bersyarat yang hanya dibatasi oleh 2 kejadian sehingga dapat diperluas untuk n kejadian. Kejadian bersyarat adalah peluang munculnya kejadian A jika diketahui kejadian B, sehingga:²¹

$$P(A | B) = \frac{P(A \cap B)}{P(B)} \quad \dots \quad (2.2)$$

Dari persamaan 2.2 didapatkan persamaan 2.2 sebagai berikut:

²¹ Lian G. Otaya, *Probabilitas Bersyarat, Independensi dan Teorema Bayes Dalam Menentukan Peluang Terjadinya Suatu Peristiwa*, Tadbir: Jurnal Manajemen Pendidikan Islam, Vo.4, No.1, Februari 2016, p.77.

$$P(B \cap A) = P(B | A) \cdot P(A) \quad \dots \quad (2.3)$$

Berdasarkan persamaan di atas maka didapatkan persamaan teorema bayes yang ditunjukan pada gambar 2.2 sebagai berikut:²²

$$P(A | B) = \frac{P(B | A) \cdot P(A)}{P(B)} \quad \dots \quad (2.4)$$

Berikut ini keterangan dari persamaan:

- Peluang *posterior* adalah predksi peluang munculnya suatu kejadian berdasarkan informasi dari kejadian lain. Dalam hal ini, $P(A|B)$ merupakan probabilitas *posterior* A dengan syarat B. Dengan kata lain, $P(A|B)$ adalah probabilitas hipotesis A benar jika diketahui peluang munculnya kejadian B.
- Peluang *prior* adalah peluang munculnya suatu kejadian yang sudah diyakini kebenarannya sebelumnya dan bisa jadi kejadian ini dipengaruhi kejadian yang lainnya. Dalam hal ini, $P(A)$ adalah probabilitas prior munculnya kejadian A.
- Peluang *likelihood* adalah peluang yang menyatakan derajat kemungkinan pengaruh suatu informasi kejadian terhadap kejadian yang lain. Dalam hal ini, $P(B|A)$ adalah peluang *likelihood* dari probabilitas *posterior* B dengan syarat A. Dengan kata lain, $P(B|A)$ adalah probabilitas *posterior* bukti B benar jika diketahui peluang munculnya kejadian A.
- Peluang *evidence* adalah suatu ukuran pembanding konstan berdasarkan peluang suatu informasi kejadian. Dalam hal ini, $P(B)$ adalah probabilitas *prior* munculnya kejadian B.

Misalnya, seorang guru mencatat kejadian kehadiran dan absennya seorang siswa kelas F berdasarkan kondisi cuacanya selama 10 hari berturut-turut. Berdasarkan pengamatann diketahui bahwa guru tersebut memperoleh data kehadiran dan absennya siswa tersebut sebagai berikut:

²² Ibid.

Tabel 2.4
Ilustrasi Daftar Kehadiran Siswa Berdasarkan Kondisi Cuaca

No	Cuaca	Kehadiran
1	Cerah	Hadir
2	Cerah	Hadir
3	Mendung	Hadir
4	Mendung	Absen
5	Hujan	Absen
6	Hujan	Absen
7	Mendung	Hadir
8	Cerah	Hadir
9	Cerah	Hadir
10	Hujan	Hadir

Pada tabel di atas dapat dilihat bagaimana kondisi cuaca mempengaruhi kehadiran siswa. Dalam hal ini, kondisi cuaca cerah, mendung, hujan, akan mempengaruhi hadir atau absennya siswa. Jika total hadir dan absen siswa tersebut dikelompokkan berdasarkan kondisi cuacanya maka didapatkan hasil sebagai berikut:

Tabel 2.5
Tabel Ilustrasi Prior Kehadiran Siwa Berdasarkan Kondisi Cuaca

No	Cuaca	Hadir	Absen
1	Cerah	3	1
2	Mendung	2	1
3	Hujan	1	2
	Total	6	4

Jika ditanya berapa kemungkinan siswa kelas F akan absen jika kondisi cuaca hujan, maka jawabnya:

$$P(Absen|Hujan) = \frac{P(Hujan|Absen) \cdot P(Absen)}{P(Hujan)} \quad \dots \quad (2.5)$$

Berikut dijelaskan cara penyelesaian soal di atas menggunakan persamaan probabilitas bersyarat:

1. Pertama, hitung probabilitas *prior* yakni probabilitas munculnya kejadian absen. Hal ini dinotasikan sebagai $P(\text{Absen})$. Berdasarkan tabel 2.2, diketahui bahwa $P(\text{Absen}) = 4/10$.
2. Kedua, hitung probabilitas *likelihood* yakni peluang munculnya kejadian hujan jika diketahui peluang munculnya absen. Dalam hal ini, probabilitas *likelihood* dinotasikan sebagai $P(\text{Hujan}|\text{Absen})$. Berdasarkan tabel 2.2, diketahui bahwa $P(\text{Hujan}|\text{Absen}) = 2/4$.
3. Ketiga, hitung probabilitas *evidence* yakni peluang munculnya kejadian hujan. Dalam hal ini probabilitas *evidence* dinotasikan sebagai $P(\text{Hujan})$. Berdasarkan tabel 2.1, diketahui bahwa $P(\text{Hujan}) = 3/10$.
4. Keempat, dipahami dulu bahwa yang ditanyakan merupakan probabilitas *posterior* yakni kemungkinan siswa-siswa di kelas F akan absen jika kondisi cuacanya hujan. Dengan demikian, probabilitas *posterior* bisa dinotasikan sebagai $P(\text{Absen}|\text{Hujan})$. Berdasarkan persamaan 2.4 diperoleh perhitungan sebagai berikut: $P(\text{Absen}|\text{Hujan}) = [(2/4) \times (4/10)] / (3/10) = 0,66$

Jadi peluang absennya siswa kelas F jika kondisinya cuacanya hujan sebesar 0,66 atau 66%.

2.5.1 Naive Bayes Classifier

Naive Bayes Classifier (NBC) merupakan metode pembelajaran yang klasik berbasis probabilitas yang sederhana. NBC mampu memberikan performansi yang cukup baik untuk banyak kasus modern dengan data yang besar. Metode NBC menggunakan teorema Bayes sebagaimana telah dijelaskan di atas bahwa suatu probabilitas bersyarat dinyatakan:

$$P(H | X) = \frac{P(X | H) P(H)}{P(X)} \quad \dots \dots \dots \quad (2.6)$$

Dalam *machine learning*, X adalah sebuah tuple atau objek data, H adalah hipotesis atau dugaan bahwa tuple X adalah kelas C. Secara spesifik, dalam masalah klasifikasi dapat dihitung $P(H|X)$ sebagai probabilitas bahwa hipotesis H benar

untuk *tuple* X atau dengan kata lain $P(H|X)$ adalah probabilitas bahwa *tuple* X berada dalam kelas C. Sementara itu, $P(H)$ adalah probabilitas *prior* bahwa hipotesis H benar untuk setiap *tuple* tidak peduli nilai-nilai atributnya sedangkan $P(X)$ adalah probabilitas *prior* dari *tuple* X.²³

Menurut Suyanto (2018) algoritme NBC bekerja sebagai berikut:²⁴

1. Misalnya D adalah himpunan data latih (*training set*) yang berisi sejumlah tuple beserta label kelasnya. Setiap tuple adalah berdimensi n yang dinyatakan sebagai $X = (X_1, (X_2, \dots, (X_n))$ yang didapat dari n atribut A_1, A_2, \dots, A_n .
 2. Misalkan terdapat m kelas, yaitu C_1, C_2, \dots, C_m . Untuk sebuah tuple masukan X , *naive bayes classifier* memprediksi bahwa tuple X termasuk ke dalam kelas C_i jika dan hanya jika $P(C_i|X) > P(C_j|X)$ untuk $1 \leq j \leq m, j \neq i$. Dengan kata lain, *naive bayes classifier* bekerja dengan cara memaksimumkan $P(C_i|X)$. Kelas C_i yang membuat $P(C_i|X)$ bernilai maksimum disebut *maximum posteriori hypothesis*. Dengan teorema bayes, $P(C_i|X)$ diestimasi menggunakan formula sebagai berikut:

$$P(Ci \mid X) = \frac{P(X \mid Ci) P(Ci)}{P(X)} \quad \dots \quad (2.7)$$

3. Mengingat $P(X)$ bernilai sama untuk semua kelas (artinya, tuple X memiliki probabilitas yang sama untuk masuk ke dalam kelas manapun), maka hanya $P(X|C_i) P(C_i)$ yang perlu dimaksimalkan. Jika probabilitas *prior* untuk setiap kelas tidak diketahui, maka probabilitas setiap kelas biasanya diasumsikan sama, $P(C_1) = P(C_2) = \dots = P(C_m)$. Dengan demikian, *Naive Bayes Classifier* hanya memaksimalkan $P(X|C_i)$. Namun, jika probabilitas setiap prior untuk setiap kelas bisa berbeda-beda, maka *Naive Bayes Classifier* harus memaksimalkan $P(X|C_i) P(C_i)$. Misalnya, jika Anda memiliki pengetahuan bahwa jumlah tuple di kelas C_1 adalah 70% dan jumlah tuple di kelas C_2 adalah 30%, maka Anda dapat memberikan $P(C_1) = 0,7$ sedangkan $P(C_2) = 0,3$.

²³ Suyanto, *Machine Learning: Tingkat Dasar dan Lanjut*, Bandung: Informatika Bandung, 2018, p.60.

²⁴ Ibid, pp.60-62.

4. Jika Anda berhadapan dengan himpunan data yang memiliki sangat banyak atribut, Anda dapat mereduksi kompleksitas penghitungan $P(X|C_i)$ dengan asumsi naif tentang independensi bersyarat kelas, yaitu nilai-nilai atribut saling independen. Artinya, antar atribut saling bebas, tidak ada ketergantungan sama sekali. Dengan demikian, *naive bayes classifier* memaksimalkan:

$$\begin{aligned} P(C_i | X) &= P(X_k | C_i) \\ &= P(X_1 | C_i) \times P(X_2 | C_i) \times \dots \times P(X_n | C_i) \end{aligned} \quad \dots \dots \dots \quad (2.8)$$

Berdasarkan tuple-tuple pada himpunan data latih, Anda dapat mengestimasi $P(x_1 | C_i), P(x_2 | C_i), \dots, P(x_n | C_i)$ berdasarkan tipe atributnya. Untuk atribut bernilai kategorial, $P(x_k | C_i)$ didefinisikan sebagai jumlah tuple di kelas C_i dalam himpunan data D yang memiliki nilai X_k pada atribut A_k dibagi dengan total tuple di kelas C_i dalam D yang disimbolkan sebagai $|C_{i,D}|$

5. Untuk memprediksi label kelas dari tuple X , maka perlu menghitung probabilitas $P(X | C_i) P(C_i)$ untuk setiap kelas C_i . Selanjutnya hanya perlu memaksimalkan probabilitas tersebut, yaitu mencari kelas C_i yang menghasilkan probabilitas $P(X | C_i) P(C_i)$ maksimum sebagai kelas himpunan. Secara matematis, tuple X diberi label kelas C_i jika hanya jika: $P(X | C_i) P(C_i) > P(X | C_j) P(C_j)$ untuk $1 \leq j \leq m, j \neq i$.
6. Bagaimana jika probabilitas $P(C_i | X)$ menghasilkan nilai 0 karena atributnya tidak pernah muncul pada data latih? Untuk mengatasi masalah tersebut bisa menambahkan tuple pura-pura dengan kelas yang belum pernah muncul pada himpunan data latih untuk setiap atribut yang ada. Tuple pura-pura ini untuk menghindari probabilitas bernilai 0. Teknik ini dikenal sebagai *Laplacian correction* atau *laplace estimator*.

Berikut diberikan ilustrasi studi kasus penggunaan *Naive Bayes* dalam menentukan rekomendasi pembelian HP berdasarkan fitur-fiturnya. Suatu konter memiliki dataset rekomendasi HP. Terdapat himpunan data HP dengan 3 atribut meliputi baterai, kamera, harga, dan rekomendasi (layak dibeli atau tidak) sebagaimana ditunjukkan pada tabel 2.6. Jika diberikan HP dengan kriteria bateria

sedang, kamera sedang, harga murah, dengan menggunakan *Naive Bayes* tentukanlah apakah HP tersebut layak direkomendasikan atau tidak? Berikut ini merupakan ilustrasi rekomendasi HP berdasarkan kriteria baterai, kamera, dan harga sebagaimana ditunjukkan tabel 2.6:

Tabel 2.6
Ilustrasi Rekomendasi HP

HP	Baterai	Kamera	Harga	Rekomendasi
H1	Kuat	Tinggi	Sangat Murah	Ya
H2	Kuat	Tinggi	Sangat Mahal	Ya
H3	Kuat	Sedang	Mahal	Ya
H4	Kuat	Rendah	Mahal	Tidak
H5	Sedang	Tinggi	Murah	Ya
H6	Sedang	Sedang	Mahal	Ya
H7	Lemah	Rendah	Sangat Mahal	Tidak
H8	Sedang	Sedang	Murah	Tidak
H9	Lemah	Tinggi	Murah	Tidak
H10	Lemah	Tinggi	Sangat Mahal	Tidak

Untuk menyelesaikan permasalahan di atas menggunakan *Naive Bayes* dapat mengikuti langkah-langkah berikut:

1. Misalnya, diasumsikan *tuple* X atau tuple masukan = (bateria sedang, kamera sedang, harga murah).
2. Hitung probabilitas *prior* untuk kelas C_i yakni kelas layak direkomendasikan (C_1) dan kelas tidak layak direkomendasikan (C_2). Karena terdapat 8 *tuple* layak direkomendasikan dari total 10 tuple himpunan data, maka $P(C_1) = P(\text{Rekomendasi}=Ya) = 5/10 = 0,5$ dan $P(C_2) = P(\text{Rekomendasi}=Tidak) = 5/10 = 0,5$.
3. Hitung probabilitas bersyarat untuk setiap kelas $P(X|C_i)$ dan untuk setiap atribut masukan yakni:
 - $P(\text{Baterai}=\text{Sedang} | \text{Rekomendasi}=Ya) = 2/5$
 - $P(\text{Baterai}=\text{Sedang} | \text{Rekomendasi}=Tidak) = 1/5$
 - $P(\text{Kamera}=\text{Sedang} | \text{Rekomendasi}=Ya) = 2/5$
 - $P(\text{Kamera}=\text{Sedang} | \text{Rekomendasi}=Tidak) = 1/5$
 - $P(\text{Harga}=\text{Murah} | \text{Rekomendasi}=Ya) = 1/5$

- $P(\text{Harga}=\text{Murah} | \text{Rekomendasi}=\text{Tidak}) = 2/5$
4. Hitung probabilitas $P(X|C_i)$ untuk setiap kelas C_i yakni:
- $P(X|\text{Rekomendasi}=\text{Ya}) = P(\text{Baterai}=\text{Sedang}|\text{Rekomendasi}=\text{Ya}) \times P(\text{Kamera}=\text{Sedang}|\text{Rekomendasi}=\text{Ya}) \times P(\text{Harga}=\text{Murah}|\text{Rekomendasi}=\text{Ya}) = 2/5 \times 2/5 \times 1/5 = 0,032$
 - $P(X|\text{Rekomendasi}=\text{Tidak}) = P(\text{Baterai}=\text{Sedang}|\text{Rekomendasi}=\text{Tidak}) \times P(\text{Kamera}=\text{Sedang}|\text{Rekomendasi}=\text{Tidak}) \times P(\text{Harga}=\text{Murah}|\text{Rekomendasi}=\text{Tidak}) = 1/5 \times 1/5 \times 2/5 = 0,016$
5. Hitung nilai Cmap untuk tuple X di kelas C_i dengan memaksimalkan $P(X|C_i) P(C_i)$ yakni:
- $\text{Cmap}(\text{Rekomendasi}=\text{Ya}) = 0,032 \times 0,5 = 0,032$
 - $\text{Cmap}(\text{Rekomendasi}=\text{Tidak}) = 0,016 \times 0,5 = 0,008$
6. Terakhir, bandingkan hasil $P(X | \text{Rekomendasi} = \text{Ya})$ dengan $P(X | \text{Rekomendasi} = \text{Tidak})$. Karena $P(X|\text{Rekomendasi}=\text{Ya})$ lebih besar daripada $P(X | \text{Rekomendasi} = \text{Tidak})$ maka dapat disimpulkan bahwa HP dengan kriteria bateria sedang, kamera sedang, harga murah, layak untuk direkomendasikan. Dengan demikian, $X = (\text{bateria sedang}, \text{kamera sedang}, \text{harga murah})$, masuk ke dalam $P(X | \text{Rekomendasi} = \text{Ya})$.

2.5.2 Multinomial Naive Bayes

Multinomial Naive Bayes (MNB) merupakan variasi dari metode pembelajaran *Naive Bayes* yang digunakan untuk mengklasifikasikan sekumpulan dokumen. MNB merupakan metode *supervised learning*. Sehingga setiap data perlu diberikan label sebelum dilakukan pelatihan data. Metode pembelajaran yang membangkitkan fungsi untuk memetakan input ke output yang diinginkan. Dengan kata lain, kualitas hasil pembelajaran bergantung pada kesesuaian antara input dan output yang diberikan user. Dengan demikian, user sangat berperan dalam memvalidasi input dan output tersebut.

Dalam hal ini, yang membedakan antara *Naive Bayes Classifier* dengan *Multinomial Naive Bayes* ialah *Multinomial Naive Bayes* adalah salah satu variasi dari pengklasifikasi *Naive Bayes* yang menggunakan distribusi multinomial untuk

masing-masing fitur. Hal ini bekerja dengan baik pada data yang bisa diubah menjadi jumlah seperti jumlah kata dalam suatu dokumen teks. Sedangkan, umumnya istilah *Naive Bayes* merujuk pada probabilitas bersyarat dari masing-masing fitur dalam suatu model. Sementara *Multinomial Naive Bayes* adalah *Naive Bayes Classifier* yang menggunakan distribusi nominal dari masing-masing fitur dalam suatu model.²⁵

Metode MNB menempuh dua tahap yakni pelatihan data dan pengujian (klasifikasi) data. Pada tahap pelatihan data dilakukan proses analisis terhadap sampel dokumen berupa pemilihan *vocabulary* yakni kata yang mungkin muncul dalam koleksi dokumen sampel yang sedapat mungkin merepresentasikan dokumen. Selanjutnya penentuan probabilitas *prior* bagi tiap sampel dokumen. Pada tahap pengujian ditentukan nilai kategori dari suatu dokumen berdasarkan *term* yang muncul dalam dokumen yang diklasifikasikan.²⁶ Klasifikasi dokumen bertujuan untuk menentukan kelas terbaik untuk suatu dokumen. Kelas terbaik dalam MNB ditentukan dengan mencari maximum a posteriori (C_{map}) kelas C melalui persamaan:²⁷

$$C_{map} = \operatorname{argmax} P(c) \prod_1^n P(t_k|c) \quad \dots \quad (2.9)$$

Dimana $P(C)$ merupakan probabilitas *prior* dari kelas C. Sementara $P(t_k|C)$ merupakan probabilitas fitur-fitur kata dari kelas C. Pada persamaan di atas, terdapat banyak probabilitas *posterior* atau probabilitas bersyarat yang dikalikan. Hal ini dapat menyebabkan *floating point underflow*. Masalah ini timbul pada dokumen yang memiliki kata yang sangat besar. Hasil perkalian probabilitas *posterior* dari seluruh fitur kata membuat nilai C_{map} sangat kecil sekali, sehingga menimbulkan kesalahan saat dilakukan proses perbandingan. Karena itu, proses perhitungan akan lebih baik jika dilakukan penjumlahan pada logaritma dari

²⁵ <https://stats.stackexchange.com/questions/33185/difference-between-naive-bayes-multinomial-naive-bayes/>, diakses tanggal 1 Januari 2021.

²⁶ Aria Wibisono, dkk, *Filtering Spam Email Menggunakan Naive Bayes*, Journal of Telematics and Information Technology, Vo.1, No.1, 2020, p.10.

²⁷ Alif Sabrani, dkk, *Metode Multinomial Naive Bayes Untuk Klasifikasi Artikel Online Tentang Gempa di Indonesia*, Jtika, Vo.2, No.1, Maret 2020, p.91.

probabilitas. Kelas dengan logaritma tertinggi merupakan kelas dengan probabilitas terbaik untuk dokumen $\log(xy) = \log(x) + \log(y)$. Persamaan yang menggunakan logaritma dari probabilitas dapat dinyatakan dalam persamaan:

$$Cmap = \arg \max [logP(c) + \sum_{1 \leq k \leq n} logP(tk|c)] \quad \dots \dots \quad (2.10)$$

$P(t_k|c)$ merupakan probabilitas *posterior* dari kelas C. Sementara $P(c)$ merupakan probabilitas prior dari kelas C. Probabilitas prior $P(c)$ dapat dihitung dengan persamaan:

$$P(c) = \frac{Nc}{N} \quad \dots \quad (2.11)$$

Berikut ini keterangan dari persamaan:

- $P(c)$ merupakan probabilitas prior suatu dokumen yang berada di kelas c.
 - N_c merupakan jumlah dokumen dengan kelas c.
 - N merupakan jumlah seluruh dokumen.

Sementara $P(t_k|c)$ merupakan probabilitas *posterior* pada fitur kata t yang berada pada kelas c. $P(t_k|c)$ bisa didapatkan dengan menghitung melalui persamaan:

Berikut ini keterangan dari persamaan:

- $P(t_k|c)$ merupakan probabilitas *posterior* pada fitur kata t dari kelas c.
 - n_k merupakan jumlah kemunculan fitur kata t pada semua dokumen yang memiliki kelas c.
 - n merupakan jumlah fitur kata pada kelas c.

Perhitungan *maximum likelihood* memiliki kelemahan yakni suatu kata dalam kelas yang tidak terlihat pada data pelatihan akan memiliki pembagian

dengan nilai 0 (*devision by zero*). Untuk mengatasi masalah ini diterapkan teknik *laplace smoothing*, sehingga persamaannya menjadi:

$$P(tk|c) = \frac{nk+1}{n+|kata|} \quad \dots \quad (2.13)$$

Dimana $|kata|$ merupakan jumlah kata yang unik yang muncul pada seluruh dokumen. Dengan kata lain, $|kata|$ merupakan total fitur kata.

2.6 Ukuran Evaluasi Model Klasifikasi

Untuk mengevaluasi suatu model klasifikasi diperlukan ukuran tertentu. Ukuran tersebut meliputi *accuracy*, *error rate*, *recall*, *specificity*, *precision*, *f-score*. Parameter *accuracy and error rate* sebaiknya digunakan untuk dataset dengan proporsi jumlah kelas yang berimbang. Adapun parameter *recall*, *specificity*, *precision*, *f-score* sebaiknya digunakan untuk dataset dengan proporsi jumlah kelas yang tidak berimbang.. Ukuran-ukuran tersebut dapat dihitung dengan menghitung nilai *confusion matrix* yang diilustrasikan pada tabel 2.1 sebagai berikut:²⁸

Tabel 2.7
Confusion Matrix

	Prediksi Salah	Prediksi Benar
Aktual Salah	<i>True negative (TN)</i>	<i>False positive (FP)</i>
Aktual Benar	<i>False negative (FN)</i>	<i>True positive (TP)</i>

Berikut keterangan dari *confusion matrix*:

- TP atau *true positive*, yakni jumlah tuple positif yang dilabeli dengan benar oleh model klasifikasi. Yang dimaksud tuple positif ialah tuple aktual yang berlabel positif, seperti kategori komentar *cyberbullying*.
 - TN = *true negative*, yakni jumlah tuple negatif yang dilabeli dengan benar oleh model klasifikasi. Yang dimaksud tuple negatif ialah tuple aktual yang berlabel negatif, seperti kategori komentar *non-cyberbullying*.

²⁸ Suyanto, Loc. Cit, p.332.

- FP = *false positive*, yakni jumlah tuple negatif yang salah dilabeli oleh model klasifikasi. Misalnya, kategori komentar *non-cyberbullying*, tetapi oleh model klasifikasi dilabeli *cyberbullying*.
- FN = *false negative*, yakni jumlah tuple positif yang salah dilabeli oleh model klasifikasi. Misalnya, kategori komentar *cyberbullying*, tetapi oleh model klasifikasi dilabeli *non-cyberbullying*.

Berikut ini dijelaskan masing-masing ukuran evaluasi klasifikasi meliputi:²⁹

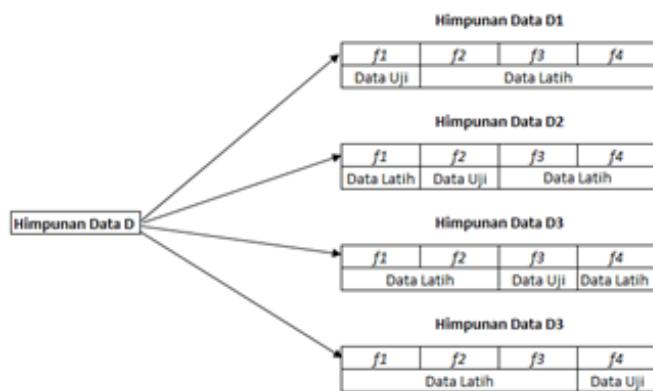
- **Accuracy** atau tingkat pengenalan menyatakan persentase dari jumlah tuple dalam data uji yang diklasifikasikan dengan benar oleh model klasifikasi. Accuracy dihitung dengan rumus: $(TP+TN)/(TP+TN+FP+FN)$.
- **Error rate** menyatakan tingkat persentase kesalahan atau kekeliruan dari jumlah tuple dalam data uji yang diklasifikasikan oleh model klasifikasi. Error rate dihitung dengan rumus: $(FP+FN)/(TP+TN+FP+FN)$.
- **Recall** adalah ukuran kelengkapan yakni berapa persentase tuple positif yang dilabeli sebagai positif. Recall dihitung dengan rumus: $TP/(TP+FN)$.
- **Specificity** atau *true negative rate* dihitung dengan rumus: $TN/(FP+TN)$.
- **Precision** atau ukuran kepastian, yakni berapa persentase tuple yang dilabeli sebagai positif adalah benar pada kenyataannya, yang dirumuskan sebagai: $TP/(TP+FP)$.
- **F-Score** atau rata-rata harmonik dari precision dan recall dihitung dengan rumus: $(2*precision*recall)/(precision+recall)$.

Pada tugas akhir ini, ukuran model lebih ditekankan pada *accuracy and error rate*. Kedua parameter tersebut digunakan untuk mengevaluasi model dalam mengukur performa model dalam mengklasifikasikan komentar *cyberbullying* atau *non-cyberbullying*. Adapun parameter *recall*, *precision*, *f-score* digunakan sebagai ukuran penunjang.

²⁹ Ibid, p.333.

2.7 Teknik K-Fold Cross-Validation

K-fold cross-validation merupakan teknik validasi model yang berguna untuk mengevaluasi kinerja model dimana dataset D dibagi dipisahkan menjadi sub himpunan data (*fold*) sejumlah k sub himpunan. Sehingga masing-masing *fold* berisi 1/k bagian dataset. Selanjutnya dataset D₁, D₂, .., D_n berisi masing-masing (k-1) *fold* untuk data latih dan 1 *fold* untuk data uji. Dengan demikian, setiap data akan dijadikan sebagai data latih dan data uji sebanyak 1 kali.



**Gambar 2.2
4-Fold Cross Validation**

Misalnya, dengan k=4, maka didapatkan himpunan data D₁ berisi 3 *fold* yakni f₂, f₃, f₄ untuk data latih dan f₁ untuk data uji. Himpunan data D₂ berisi 3 *fold* yakni f₁, f₃, f₄ untuk data latih dan f₂ untuk data uji. Demikian seterusnya untuk himpunan data D₃ dan D₄. Sehingga setiap *fold* pernah menjadi data uji sebanyak satu kali.

Dengan metode *K-Fold Cross-Validation* bisa mengukur kualitas semua model klasifikasi yang dibangun. Dengan metode ini dapat juga membandingkan sejumlah model klasifikasi. Selain itu metode ini dapat digunakan untuk menyeleksi dan memilih model yang terbaik di antara semua model yang dibangun.³⁰ Dalam tugas akhir ini, *4-Fold Cross-Validation* digunakan dalam validasi model sehingga model akan dilatih dan diuji sebanyak 4 kali.

³⁰ Ibid, pp.348-351.

2.8 Ukuran Evaluasi Website

Mengevaluasi situs web berguna untuk mengukur kualitas suatu situs web. Salah satu dalam mengevaluasi situs *web* ialah *Web Quality Evaluation Method* (WebQEM). Olsina dan Rossin (2002) mengidentifikasi terdapat empat aspek dalam mengevaluasi suatu *website* meliputi *functionality* (fungsionalitas), *reliability* (kehandalan), *usability* (kegunaan), dan *efficiency* (efisiensi). Berikut dijelaskan aspek-aspek dari WebQEM sebagai berikut:³¹

- Kegunaan

Aspek kegunaan adalah tingkat kualitas dari sistem yang mudah dipelajari, mudah digunakan dan mendorong pengguna untuk menggunakan sistem sebagai alat bantu positif dalam menyelesaikan tugas. Aspek kegunaan dapat diukur dengan mengukur kemudahan pengguna dalam mempelajari tampilan antar muka (*easy to use*). Berikutnya, aspek kegunaan dapat diukur dengan menilai seberapa menarik tampilan antar muka bagi pengguna (*user friendly*). Terakhir, aspek kegunaan dapat diukur dengan menilai seberapa responsif tampilan antar muka baik di layar PC maupun mobile (*responsible*).

- Kehandalan

Aspek kehandalan adalah peluang sebuah komponen, sub-sistem atau sistem melakukan fungsinya dengan baik, seperti yang dipersyaratkan, dalam kurun waktu tertentu dan dalam kondisi operasi tertentu pula. Kehandalan dapat diukur dengan menguji seberapa baik sistem memberikan hasil yang akurat, tanpa kegagalan. Selain itu, kehandalan juga dapat diukur dari penanganan terjadinya kesalahan dan pemulihan dari kegagalan sistem. Kehandalan juga dapat diukur dengan mengukur tingkat ketersediaan konten yang dibutuhkan pengguna (*availability*).

- Fungsionalitas

Aspek fungsionalitas adalah kemampuan sistem dalam menjalankan tugas-tugas atau fungsi utamanya. Suatu fitur dapat dikatakan berfungsi dengan baik

³¹ Fajriyan Nurli, *Analisis Pengaruh Kualitas Website PT. Tiki Jalur Nugraha Ekakurir (JNE) Terhadap Kepuasan Pengguna Dengan Metode WebQEM*, Surabaya: Stikom Surabaya, 2018, pp.14-16.

tatkala masukan dan keluarannya sesuai. Dengan kata lain, fungsionalitas merupakan kesesuaian antara masukan dan keluaran yang direncanakan. Oleh karena itu, aspek fungsionalitas dapat diukur dengan menilai seberapa baik fitur-fitur dalam suatu situs dapat bekerja secara baik.

- Efisiensi

Aspek efisiensi adalah kemampuan menjalankan tugas dengan baik dan tepat tanpa membuang waktu, tenaga maupun biaya. Efisiensi berhubungan dengan respon, waktu pemrosesan, dan pemanfaatan sumber daya yang digunakan sistem. Jika suatu perangkat lunak memerlukan respon, waktu pemrosesan dan sumber daya yang banyak maka dapat dikatakan perangkat lunak itu tidak efisien, berlaku sebaliknya. Adapun aspek efisiensi dapat diukur dengan mengukur seberapa cepat suatu situs dalam memproses konten-kontennya (*time behaviour*). Selain itu, efisiensi juga dapat diukur dengan menilai kemudahan dalam mengakses fitur-fitur atau layanan-layanan dalam suatu situs web (*accessibility*).

2.9 Tool dan Library Pendukung

Untuk mendukung pengembangan sistem, dipakai beberapa *tool* dan *library* tambahan dari pihak ke-3. *Tool* dan *library* dipakai untuk memudahkan dalam implementasi dan pengembangan perangkat lunak. Adapun *tool* dan *library* pendukung yang dipakai dalam tugas akhir ini meliputi MySQL, jQuery, Bootstrap, DataTables, Chartjs, TinyMCE, dan Ion Auth. Berikut penjelasan selengkapnya:

- MySQL

MySQL adalah perangkat lunak manajemen database yang sumber terbuka di bawah persyaratan Lisensi Publik Umum GNU, dan juga tersedia berbagai lisensi. MySQL dimiliki dan disponsori oleh perusahaan Swedia MySQL AB, yang dibeli oleh Sun Microsystems (sekarang Oracle Corporation).³² Pada tugas akhir ini, MySQL dipakai sebagai DBMS atau sistem manajemen basis data untuk mengelola database. MySQL dipilih

³² <https://www.mysql.com/>, diakses tanggal 22 Desember 2020.

sebagai DBMS karena bersumber terbuka, cukup stabil, mendukung transaksi data, fleksibel terhadap bahasa PHP, keamanan cukup baik dan tersedia di berbagai hosting di Indonesia.

- JQuery

JQuery adalah pustaka JavaScript yang dirancang untuk menyederhanakan dan manipulasi HTML DOM tree traversal, serta penanganan *event*, animasi CSS, dan Ajax. Dengan menggunakan jQuery dapat mempermudah dan mempersingkat dalam membuat kode javascript dari kode yang panjang menjadi hanya beberapa kode baris saja. JQuery merupakan perangkat lunak dengan lisensi MIT permisif.³³ Pada tugas akhir ini, jQuery dipakai untuk memudahkan pengembangan tampilan situs CMS supaya situs menjadi lebih interaktif dengan memanfaatkan sejumlah fitur jQuery seperti penanganan *event*, DOM, animasi CSS dan Ajax. Dengan jQuery, implementasi JavaScript menjadi lebih ringkas.

- Bootstrap

Bootstrap adalah kerangka kerja CSS bersumber terbuka yang digunakan dalam pengembangan situs *front-end* yang responsif terhadap berbagai perangkat termasuk layar *mobile*. Bootstrap diciptakan untuk mempermudah proses mendesain web bagi berbagai kalangan dari pemula hingga berpengalaman. Dapat dikatakan bootstrap merupakan template desain web dengan fitur plus (*framework css*).³⁴ Pada tugas akhir ini, bootstrap digunakan untuk pengembangan desain tampilan situs CMS supaya lebih responsif dan menarik.

- DataTables

DataTables adalah *plug-in* jQuery yang bersumber terbuka untuk membuat tabel HTML dinamis. DataTables adalah alat yang sangat fleksibel, dibangun di atas dasar peningkatan progresif, yang menambahkan semua fitur lanjutan ke tabel HTML mana pun. DataTables memiliki lisensi MIT.³⁵ Pada

³³ Subagia, Loc. Cit. p.25.

³⁴ Ibid, p.43.

³⁵ <https://datatables.net/>, diakses tanggal 22 Desember 2020.

tugas akhir ini, DataTables digunakan untuk membantu pembuatan segala tabel yang digunakan di berbagai fitur seperti manajemen artikel dan manajemen komentar. Dengan penggunaan DataTables, tabel menjadi lebih responsif, dan memudahkan dalam pengindeksan data dan pencarian data.

- Chartjs

Chartjs adalah pustaka JavaScript sumber terbuka untuk visualisasi data, yang mendukung 8 jenis bagan: batang, garis, area, pai, gelembung, radar, kutub, dan pencar. Beberapa keuntungan menggunakan Chartjs meliputi bersumber terbuka, memiliki 8 macam tipe grafik, mudah digunakan menggunakan HTML5 canvas, dan responsif terhadap tampilan berbagai layar. Chartjs berdiri di bawah lisensi MIT.³⁶ Pada tugas akhir ini, Chartjs digunakan untuk pembuatan laporan meliputi laporan perkembangan tayangan seluruh artikel, laporan perkembangan tayangan per artikel, laporan artikel terbaik berdasarkan total tayangan, laporan artikel terbaik berdasarkan rata-rata rating, laporan perkembangan rating artikel dan laporan statistik perbandingan komentar cyberbullying dan non-cyberbullying. Dengan chartjs, pembuatan laporan menjadi lebih mudah dan responsif.

- TinyMCE

TinyMCE adalah teks editor yang mewah yang dirilis sebagai perangkat lunak sumber terbuka di bawah lisensi LGPL yang memiliki kemampuan untuk mengubah bidang textarea HTML atau elemen HTML lainnya menjadi teks editor. Adapun keuntungan menggunakan TinyMCE yakni fleksibel dan bisa di-custom. TinyMCE begitu fleksibel hanya membutuhkan beberapa kode untuk membuat *text editor* yang mewah. Selain itu, TinyMCE bisa didesain dengan konfigurasi *custom* menyesuaikan kebutuhan pengguna.³⁷ Pada tugas akhir ini, TinyMCE digunakan sebagai *text editor* dalam fitur pembuatan artikel. Dengan TinyMCE, penulisan artikel dalam situs blog menjadi lebih mudah dan fleksibel.

³⁶ <https://www.chartjs.org/>, diakses tanggal 22 Desember 2020.

³⁷ <https://www.tiny.cloud/>, diakses tanggal 22 Desember 2020.

- Ion Auth

Ion Auth adalah pustaka CodeIgniter yang dikembangkan oleh Ben Edmunds untuk otentikasi sederhana dan ringan. Dengan menggunakan Ion Auth, pembuatan login dan registrasi pada *framework* Codeigneter menjadi lebih mudah. Tersedia berbagai fitur yang mudah digunakan seperti registrasi, login, logout, forgot password, aktivasi akun, dan berbagai fitur lainnya.³⁸ Pada

tugas akhir ini, Ion Auth digunakan untuk pembuatan fitur login, registrasi, reset password dan manajemen user. Dengan Ion Auth, proses pengembangan fitur otentifikasi dan manajemen user menjadi lebih ringkas dan cepat dalam proses pengembangan CMS.

³⁸ <https://github.com/benedmunds/CodeIgniter-Ion-Auth/>, diakses tanggal 22 Desember 2020.

BAB III

DATASET DAN PREPROCESSING

Pada Bab III dijelaskan tentang bagaimana proses menjalankan *preprocessing* hingga menghasilkan dataset yang akan dipakai sebagai model sistem penyaring komentar *cyberbullying* pada konten blog. Adapun penjelasannya terbagi tiga sub bab yakni pembentukan *dataset*, vertifikasi *dataset*, dan *preprocessing dataset*. Pada sub bab pembentukan *dataset* akan dijelaskan bagaimana pembentukan dataset dari pengumpulan data hingga menghasilkan *dataset* dengan spesifikasi yang ditetapkan. Sementara pada sub bab *preprocessing dataset* akan dijelaskan bagaimana proses pengolahan *dataset* sebelum diproses lebih lanjut ke model. Penjelasan selengkapnya sebagai berikut:

3.1 Pembentukan Dataset Komentar

Dataset yang digunakan dalam tugas akhir ini mengambil sampling data secara random dari berbagai situs berita dan media sosial. Adapun situs-situsnya meliputi detik.com, kaskus.com, kompasiana.com, id.quora.com, youtube.com dan akun instagram berita indonesia seperti liputan6, tempo, dan vivanews. Sementara itu, konten situsnya lebih terfokus pada topik sosial budaya, politik dan olahraga, karena ketiga topik tersebut biasanya ramai diperbincangkan bahkan tak jarang menuai pro dan kontra. Sehingga berita-berita tersebut rentan menimbulkan *cyberbullying* pada komentar antar para pembacanya.

Adapun langkah-langkah pembentukan dataset komentarnya sebagai berikut: pertama, komentar-komentar dikumpulkan satu per satu dari berbagai sumber situs berita dan media sosial. Kedua, komentar yang dikumpulkan itu kemudian diberikan label kelas komentar secara manual, apakah mengandung kategori komentar *cyberbullying* atau *non-cyberbullying*. Ketiga, komentar yang telah dilabeli kategorinya kemudian divertifikasi kategorinya. Vertifikasi melibatkan validasi internal dan validasi eksternal. Validasi internal ialah pemeriksaan kembali yang dilakukan penulis terhadap *dataset* komentar terkait kandungan *cyberbullying*

dan jenis-jenis *cyberbullying*. Sementara itu, validasi eksternal merupakan pemeriksaan terhadap kategori komentar *cyberbullying* atau *non-cyberbullying* pada *dataset* komentar yang dilakukan oleh pihak-pihak eksternal. Setelah itu, dataset komentar siap dipakai untuk diproses lebih lanjut pada *preprocessing*.

Dataset terdiri atas 508 artikel dengan 6.537 komentar. *Dataset* komentar terdiri atas 2 kelas yakni *cyberbullying* dan *non-cyberbullying*. Setiap artikel yang dikumpulkan diberi judul, topik, sumber artikel, tanggal artikel, isi artikel dan dikumpulkan berbagai komentarnya dalam dataset. Berikut ini merupakan spesifikasi dataset selengkapnya ditunjukan pada tabel 3.1:

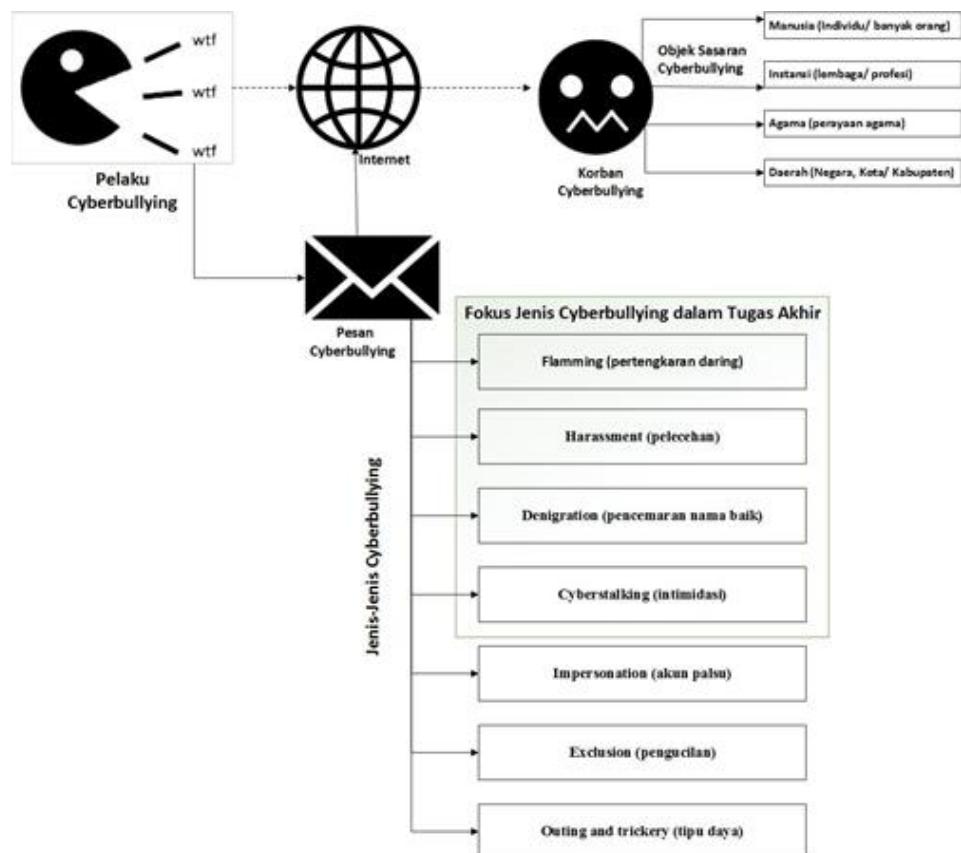
Tabel 3.1
Spesifikasi Dataset

Aspek	Deskripsi
Topik Artikel	Sosial-budaya, politik, dan olahraga
Total Artikel	508 artikel
Total Komentar	6.537 komentar
Total Kategori Komentar	2 kategori: <i>cyberbullying</i> dan <i>non-cyberbullying</i>
Total <i>Cyberbullying</i>	3.942 komentar
Total <i>Non-Cyberbullying</i>	2.595 komentar

3.2 Pelabelan dan Vertifikasi Kategori Dataset Komentar

Pada sub bab ini dijelaskan bagaimana cara melakukan pelabelan dan vertifikasi komentar pada dataset komentar. Adapun untuk melabeli komentar yang mengandung *cyberbullying* atau *non-cyberbullying* dipakai teori *cyberbullying* Williard, *cyberbullying* merupakan perlakuan kejam yang dilakukan dengan sengaja kepada orang lain dengan mengirimkan atau mengedarkan bahan yang berbahaya atau terlibat dalam bentuk-bentuk agresi sosial menggunakan internat atau teknologi digital lainnya. Menurut Williard, terdapat 7 jenis *cyberbullying* meliputi *flamming* (pertengkaran daring), *harasement* (pelecehan), *denigration* (pencemaran nama baik), *impersonation* (akun palsu), *outing and trickery* (tipu data), *exclusion* (pengucilan), dan *cyberstalking* (intimidasi). Dalam tugas akhir ini, jenis *cyberbullying* difokuskan pada *flamming* (pertengkaran daring), *harasement* (pelecehan), *denigration* (pencemaran nama baik), *cyberstalking* (intimidasi).

Adapun sasaran/ objek *cyberbullying* di Indonesia memiliki karakter tersendiri sebagaimana dijelaskan oleh Ranny Rastati (2016) dalam jurnal “Bentuk Perundungan Siber di Media Sosial dan Pencegahannya Bagi Korban dan Pelaku”. Bahwasanya *cyberbullying* yang terjadi di Indonesia selain ditujukan kepada individu, ditemukan pula tiga objek *cyberbullying* meliputi (1) perundungan daring terhadap lokasi; (2) perundungan daring terhadap keagamaan; (3) dan perundungan daring terhadap institusi atau profesi tertentu. Dengan demikian, sasaran objek *cyberbullying* mencakup individu/ banyak orang, lokasi, keagamaan, dan institusi tertentu. Selengkapnya kerangka teori *cyberbullying* bisa dilihat pada gambar 3.1 sebagai berikut:



Gambar 3.1
Kerangka Teori *Cyberbullying*

Berdasarkan kerangka teori di atas, adapun komentar yang mengandung *cyberbullying* beserta jenis *cyberbullying* yang menjadi fokus dalam tugas akhir ini meliputi:

- *Flamming* (pertengkaran daring) yakni perang kata-kata di dunia maya dengan menggunakan bahasa yang mengandung amarah, vulgar, mengancam, dan merendahkan. Pertengkarannya biasanya terjadi di surel, ruang obrol, dan media sosial.
- *Harassment* (pelecehan) yakni perundungan daring yang menggunakan kata-kata kasar, menyerang, dan melecehkan seseorang secara berulang-ulang.
- *Denigration* (pencemaran nama baik) merupakan perundungan daring yang dilakukan dengan cara mengubur keburukan seseorang di internet seperti menuliskan postingan, komentar hinaan yang bohong, gossip yang kejam, dan rumor tentang seseorang untuk merusak reputasinya.
- *Cyberstalking* (intimidasi) merupakan perilaku berulang kali mengirimkan ancaman membahayakan atau pesan-pesan yang mengintimidasi dengan menggunakan komunikasi elektronik dengan tujuan membuat orang itu tidak nyaman dan merasa khawatir atas keselamatannya.

Adapun Cyberbullying difokuskan pada objek sasaran *cyberbullying* meliputi (1) individu/ kelompok tertentu, (2) keagamaan, (3) lokasi, dan (4) institusi tertentu. Komentar-komentar yang terkumpul kemudian diberi label satu per satu sesuai dengan kategorinya apakah *cyberbullying* atau *non-cyberbullying*. Pelabelan ini dilakukan secara manual melibatkan penilaian pakar. Berikut ini dijelaskan contoh komentar *cyberbullying* yang diambil pada salah satu situs berita:

Pada artikel berjudul “Jadi Korban Klitih, Seorang Pelajar di Bantul Tewas” pada gambar 3.2 dapat dilihat terdapat berbagai komentar baik berisi *cyberbullying* maupun *non-cyberbullying*. Untuk mengetahui komentar dari pembaca bermuatan *cyberbullying* atau tidak, maka terlebih dahulu harus diidentifikasi muatan kategori komentarnya baik *cyberbullying* maupun *non-cyberbullying*. Pada komentar 1 terdapat muatan “kampungan” sebagai bentuk pelecehan terhadap orang-orang kampung yang ditujukan kepada pelaku klitih. Sementara komentar 4 mengandung kata makian “bunuh” sebagai bentuk intimidasi/ main hakim sendiri kepada pelaku

klitih. Dengan demikian komentar 1 dan 2 termasuk kategori komentar *cyberbullying*. Di sisi lain, komentar 2 dan 3 bisa diidentifikasi sebagai komentar berkategori *non-cyberbullying* karena komentar tersebut lebih berisi pesan keprihatinan dan nasihat dalam penegakan hukum bagi pelaku klitih.

detikNews > Berita Jawa Tengah

Jadi Korban Klitih, Seorang Pelajar di Bantul Tewas

Pradito Rida Pertana - detikNews

Selasa, 14 Jan 2020 18:20 WIB

12 komentar

SHARE   



Himawan Aribowo
9 bulan yang lalu
Orangng kampungan.. Kalo mau eksis ya gitu.. Caranya kampungan juga.. Hahahahaha

Enjoyaja
9 bulan yang lalu
Ya Allah .. nak nak opo gak eling dosa

iwak teri
9 bulan yang lalu
orang-orang macam ini harus di didik dalam lapas, gak bakal bisa di sekolah formal. Klo bisa langsung seumur hidup.

Enggok2an
9 bulan yang lalu
Bunuh sajalah

Gambar 3.2
Contoh Artikel dan Komentar

Adapun untuk memverifikasi kategori komentar dilakukan validasi dengan melibatkan validasi internal dan validasi eksternal. Validasi internal merupakan pemeriksaan kembali yang dilakukan penulis terhadap komentar-komentar yang telah dilabeli terkait kandungan kategori *cyberbullying*-nya. Sementara validasi eksternal adalah pemeriksaan kategori komentar yang dilakukan oleh pihak-pihak eksternal yang bertugas sebagai pengujii atau validator terhadap komentar-komentar yang telah dilabeli kategorinya. Validasi eksternal melibatkan 3 orang pengujii.

Adapun penguji validasi dari pihak eksternal meliputi: (1) Daris Afif Muafa, S.Sos. (24 tahun), (2) Fiktor Setiawan, S.Sos. (24 tahun), dan (3) Febri Sawaluddin, S.Sos. (24 tahun). Tiga penguji tersebut dipilih sebagai validator karena mereka memiliki pengalaman dalam mendidik dan membina moral para remaja di Yayasan Al Kahfi, sehingga sudah cukup familiar dalam memahami perundungan. Mereka bertiga bertugas dalam memvalidasi dataset komentar terkait kategorinya beserta jenis *cyberbullying*-nya. Dengan demikian, hasil verifikasi kategori komentar diharapkan menjadi lebih akurat atau valid. Berikut disajikan contoh tampilan verifikasi dataset komentar selengkapnya pada gambar 3.2:

Validasi Dataset Komentar				
ID	Judul Artikel	Komentar	Sentimen	Jenis Cyberbullying
1344	Jadi Korban Klitih, Seorang Pelajar di Bantul Tewas	Bunuh sajalah	Bully	cyberstalking
1345	Jadi Korban Klitih, Seorang Pelajar di Bantul Tewas	menurut saya gantian di iseng tembak kepala...nyawa ya bayar nyawa...besok2 drpd bikin mati yg lain...	Bully	cyberstalking
1346	Jadi Korban Klitih, Seorang Pelajar di Bantul Tewas	Klithih itu binatang apa sih? Koq dibiarkan hidup...?	Bully	harassment
1347	Jadi Korban Klitih, Seorang Pelajar di Bantul Tewas	Tembak aja...ntar tuanya bs lebih sadis lg	Bully	cyberstalking
1348	Jadi Korban Klitih, Seorang Pelajar di Bantul Tewas	Iseng tapi gak pake otak. Di sekolah diajari apa, di rumah dididik apa sama orang tuanya. Banyak orang sekarang cuma bisa bikin anak, tdk bisa mendidik yg baik.	Bully	harassment
1349	Jadi Korban Klitih, Seorang Pelajar di Bantul Tewas	Kalau Hukuman mati tidak di berlakukan pagi pelaku kejahatan yg membuat korban terbunuh. Makin banyak nyawa orang tidak bersalah akan hilang sia2.	Nonbully	bukan perundungan

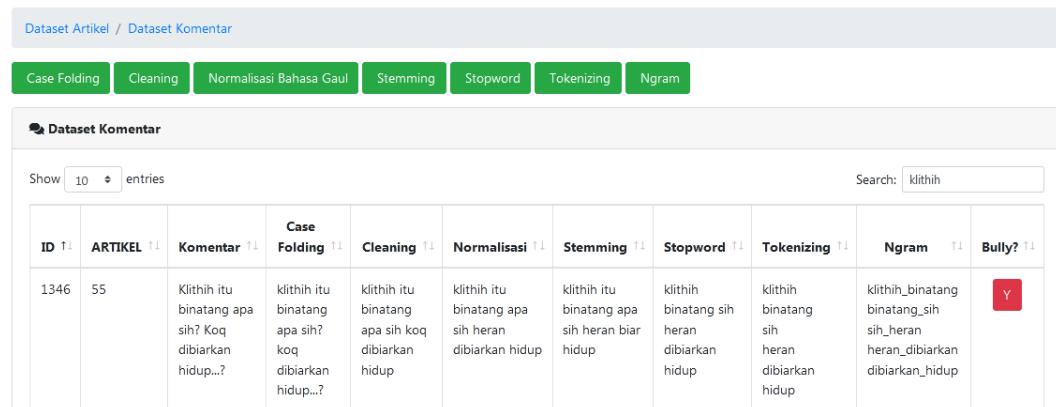
Gambar 3.3
Tampilan Halaman Vertifikasi Kategori Komentar

Pada gambar di atas, verifikasi kategori komentar bisa dilihat bahwa pada komentar dengan ID 1344, 1345, dan 1347 termasuk *cyberbullying* berjenis *cyberstalking* karena komentar-komentar tersebut mengandung ancaman intimidasi untuk membunuh atau menembak pelaku klitih. Sementara pada komentar dengan ID 1346 dan 1348 termasuk *cyberbullying* berjenis *harassment* karena komentar-komentar tersebut mengandung pelecehan terhadap pelaku klitih. Pada komentar dengan ID 1346, pelecehan terjadi dengan menyamakan pelaku

klithih sama dengan binatang. Sementara komentar dengan ID 1348, pelecehan terjadi karena pesannya mengandung kata-kata tidak sopan seperti “gak pake otak”. Sementara itu, komentar dengan ID 1349 bukanlah *cyberbullying* karena komentar tersebut hanya berisi imbauan dan saran saja dari penulisnya.

3.3 Preprocessing Dataset

Preprocessing merupakan tahapan mengolah data lebih lanjut sebelum digunakan untuk membuat suatu model. *Preprocessing* dilakukan sebelum pelatihan dan pengujian data. Adapun proses preprocessing terdiri atas enam tahapan meliputi *case folding*, *cleaning*, normalisasi bahasa gaul, *stemming*, *stopword*, *tokenizing* dan *ngram*. Berikut ini dijelaskan *preprocessing* tiap tahapannya beserta contohnya:



The screenshot shows a web application for dataset preprocessing. At the top, there are tabs for 'Dataset Artikel' and 'Dataset Komentar'. Below the tabs is a horizontal menu with buttons for 'Case Folding', 'Cleaning', 'Normalisasi Bahasa Gaul', 'Stemming', 'Stopword', 'Tokenizing', and 'Ngram'. The main area is titled 'Dataset Komentar' and contains a table with the following columns: ID, ARTIKEL, Komentar, Case Folding, Cleaning, Normalisasi, Stemming, Stopword, Tokenizing, Ngram, and Bully?. The table has one row of data corresponding to ID 1346. The 'Komentar' column contains the original text: 'Klithih itu binatang apa sih? Koq dibiarkan hidup...?'. The 'Case Folding' column shows the text converted to lowercase: 'klithih itu binatang apa sih? koq dibiarkan hidup...?'. The 'Normalisasi' column shows the text with diacritics removed: 'klithih itu binatang apa sih koq dibiarkan hidup'. The 'Stemming' column shows the base form of words: 'klithih itu binatang apa sih heran dibiarkan hidup'. The 'Stopword' column shows the text with common words removed: 'klithih binatang sih heran dibiarkan hidup'. The 'Tokenizing' column shows the text split into tokens: 'klithih', 'binatang', 'sih', 'heran', 'dibiarkan', 'hidup'. The 'Ngram' column shows the text as n-grams: 'klithih_binatang', 'binatang_sih', 'sih_heran', 'heran_dibiarkan', 'dibiarkan_hidup'. The 'Bully?' column contains a red square with a white 'Y', indicating that this comment is identified as a bully comment.

ID	ARTIKEL	Komentar	Case Folding	Cleaning	Normalisasi	Stemming	Stopword	Tokenizing	Ngram	Bully?
1346	55	Klithih itu binatang apa sih? Koq dibiarkan hidup...?	klithih itu binatang apa sih? koq dibiarkan hidup...?	klithih itu binatang apa sih koq dibiarkan hidup	klithih itu binatang apa sih heran dibiarkan hidup	klithih itu binatang apa sih heran biar hidup	klithih binatang sih heran dibiarkan hidup	klithih binatang sih heran dibiarkan hidup	klithih_binatang binatang_siheran_dibiarkan hidup	Y

Gambar 3.4
Tampilan Halaman Preprocessing Dataset Komentar

3.3.1 Case Folding Dataset

Case folding adalah proses mengubah huruf dalam dokumen menjadi huruf kecil. Dokumen mengandung beragam variasi bentuk huruf sampai tanda baca. Variasi huruf ini harus diseragamkan dan tanda bacanya harus dihilangkan. Hanya huruf ‘a’ sampai dengan ‘z’ yang diterima. Karakter lainnya dihilangkan dan dianggap sebagai delimiter. Pada tahap *case folding*, teks *komentar* diubah dari karakter kapital menjadi karakter kecil.

Selama proses *case folding*, dataset komentar akan diseragamkan menjadi varian huruf kecil. Setiap komentar diproses satu per satu melalui *case folding*. Mengingat jumlah datasetnya mencapai ribuan, maka proses *case folding* bisa berlangsung beberapa detik. Lalu hasilnya akan disimpan pada kolom *case folding* pada dataset komentar di database.

Pada tabel 3.2 diberikan ilustrasi contoh bagaimana *case folding* bekerja. Pada tabel tersebut terdapat 4 kolom meliputi no, sebelum *case folding*, setelah *case folding* dan kategori. Pada kolom sebelum *case folding* merupakan teks komentar asli yang diambil dari situs detik.com. Sementara kolom setelah *case folding* merupakan hasil komentar setelah dilakukan *case folding*. Berikut ditunjukkan pada tabel 3.2 tentang bagaimana ilustrasi *case folding dataset* sebagai berikut:

Tabel 3.2
Case Folding Dataset

No	Sebelum Case Folding	Setelah Case Folding	Kategori
1	Orangng kampungan.. Kalo mau eksis ya gitu.. Caranya kampungan juga.. Hahahhahahaha	orangng kampungan.. kalo mau eksis ya gitu.. caranya kampungan juga.. hahahhahahaha	<i>Cyberbullying</i>
2	Ya Allah .. nak nak opo gak eling dosa	ya allah .. nak nak opo gak eling dosa	<i>Non-cyberbullying</i>
3	orang-orang macam ini harus di didik dalam lapas, gak bakal bisa di sekolah formal. Klo bisa langsung seumur hidup.	orang-orang macam ini harus di didik dalam lapas, gak bakal bisa di sekolah formal. klo bisa langsung seumur hidup.	<i>Non-cyberbullying</i>
4	Klithih itu binatang apa sih? Koq dibiarkan hidup...?	klithih itu binatang apa sih? koq dibiarkan hidup...?	<i>Cyberbullying</i>
5	Tembak aja...ntar tuanya bs lebih sadis lg	tembak aja...ntar tuanya bs lebih sadis lg	<i>Cyberbullying</i>
6	Kalau Hukuman mati tidak di berlakukan pagi pelaku kejahanan yg membuat korban terbunuh. Makin banyak nyawa orang tidak bersalah akan hilang sia2.	kalau hukuman mati tidak di berlakukan pagi pelaku kejahanan yg membuat korban terbunuh. makin banyak nyawa orang tidak bersalah akan hilang sia2.	<i>Non-cyberbullying</i>

3.3.2 Cleaning Dataset

Cleaning yakni proses menghilangkan dokumen dari mention, hastag, link, emoticon, dan karakter lainnya yang tak berguna. *Cleaning* berfungsi membersihkan dataset dari karakter-karakter tidak penting. Teks komentar dibersihkan dari hastag (#), mention (@), *emoticon*, dan karakter atau tanda baca yang tak bernilai lainnya seperti titik (.), koma (,), titik dua (:), titik koma dan (;). Sehingga hanya meninggalkan karakter dengan pola a – z , 0–9.

Selama proses *cleaning*, dataset komentar akan dibersihkan dari karakter tidak penting yang tidak punya signifikansi terhadap kategori komentar. Setiap komentar diproses satu per satu melalui *cleaning*. Mengingat jumlah datasetnya mencapai ribuan, maka proses *cleaning* bisa berlangsung beberapa detik. Lalu hasilnya akan disimpan pada kolom *cleaning* tabel dataset komentar di database.

Pada tabel 3.3 diberikan ilustrasi contoh bagaimana *cleaning* bekerja. Pada tabel tersebut terdapat 4 kolom meliputi no, sebelum *cleaning*, setelah *cleaning* dan kategori. Pada kolom sebelum *cleaning* merupakan teks komentar hasil dari *case folding* yang sebelumnya telah berhasil dilakukan. Sementara kolom setelah *cleaning* merupakan hasil komentar setelah dilakukan *cleaning*. Berikut disajikan contoh *cleaning* pada dataset komentar, selengkapnya ditunjukkan pada tabel 3.3:

Tabel 3.3
Cleaning Dataset

No	Sebelum Cleaning	Setelah Cleaning	Kategori
1	orangng kampungan.. kalo mau eksis ya gitu.. caranya kampungan juga.. hahahhahahaha	orangng kampungan kalo mau eksis ya gitu caranya kampungan juga hahahhahahaha	<i>Cyber-bullying</i>
2	ya allah .. nak nak opo gak eling dosa	ya allah nak nak opo gak eling dosa	<i>Non-cyberbullying</i>
3	orang-orang macam ini harus di didik dalam lapas, gak bakal bisa di sekolah formal. klo bisa langsung seumur hidup.	orang orang macam ini harus di didik dalam lapas gak bakal bisa di sekolah formal klo bisa langsung seumur hidup	<i>Non-cyberbullying</i>
4	klithih itu binatang apa sih? koq dibiarkan hidup...?	klithih itu binatang apa sih koq dibiarkan hidup	<i>Cyberbullying</i>

**Tabel 3.3
(Lanjutan)**

No	Sebelum Cleaning	Setelah Cleaning	Kategori
5	tembak aja...ntar tuanya bs lebih sadis lg	tembak aja ntar tuanya bs lebih sadis lg	<i>Cyberbullying</i>
6	kalau hukuman mati tidak di berlakukan pagi pelaku kejahatan yg membuat korban terbunuh. makin banyak nyawa orang tidak bersalah akan hilang sia2.	kalau hukuman mati tidak di berlakukan pagi pelaku kejahatan yg membuat korban terbunuh makin banyak nyawa orang tidak bersalah akan hilang sia	<i>Non-cyberbullying</i>

3.3.3 Normalisasi Bahasa Gaul Dataset

Normalisasi bahasa gaul adalah proses mengubah kata-kata tak lazim berupa kata-kata gaul menjadi kata-kata formal berbahasa Indonesia. Umumnya, tidak semua komentar pada suatu artikel menggunakan bahasa formal. Pembaca biasanya juga memakai bahasa gaul, seperti nggak, gue, loe, dll. Kata-kata yang tak lazim ini perlu diseragamkan melalui normalisasi bahasa menjadi bahasa formal berbahasa Indonesia. Kata-kata tak lazim ini biasanya berupa singkatan populer maupun kata-kata gaul.

Selama proses normalisasi bahasa gaul, dataset komentar akan diseragamkan kata-kata gaulnya menjadi kata-kata formal. Setiap komentar diproses satu per satu melalui normalisasi bahasa gaul. Mengingat jumlah datasetnya mencapai ribuan, maka proses normalisasi bahasa gaul bisa berlangsung beberapa menit. Lalu hasilnya akan disimpan pada kolom normalisasi bahasa gaul tabel dataset komentar di database.

Pada tabel 3.4 diberikan ilustrasi contoh bagaimana normalisasi bahasa gaul bekerja. Pada tabel tersebut terdapat 4 kolom meliputi no, sebelum normalisasi bahasa gaul, setelah normalisasi bahasa gaul, dan kategori. Pada kolom sebelum normalisasi merupakan teks komentar hasil dari *cleaning* yang sebelumnya telah berhasil dilakukan. Sementara kolom setelah normalisasi merupakan hasil komentar setelah dilakukan normalisasi bahasa gaul. Berikut ini disajikan proses normalisasi bahasa gaul pada dataset komentar yang ditunjukan pada tabel 3.4:

Tabel 3.4
Normalisasi Bahasa Gaul Dataset

No	Sebelum Normalisasi	Setelah Normalisasi	Kategori
1	orangng kampungan kalo mau eksis ya gitu caranya kampungan juga hahahahaha	orang kampungan kalau mau eksis ya begitu caranya kampungan juga tawa	Cyberbullying
2	ya allah nak nak opo gak eling dosa	ya allah anak anak apa tidak ingat dosa	Non-cyberbullying
3	orang orang macam ini harus di didik dalam lapas gak bakal bisa di sekolah formal klo bisa langsung seumur hidup	orang orang macam ini harus di didik dalam lapas tidak bakal bisa di sekolah formal kalau bisa langsung seumur hidup	Non-cyberbullying
4	klithih itu binatang apa sih koq dibiarkan hidup	klithih itu binatang apa heran dibiarkan hidup	Cyberbullying
5	tembak aja ntar tuanya bs lebih sadis lg	tembak aja entar tuanya bisa lebih sadis lagi	Cyberbullying
6	kalau hukuman mati tidak di berlakukan pagi pelaku kejahatan yg membuat korban terbunuh makin banyak nyawa orang tidak bersalah akan hilang sia	kalau hukuman mati tidak di berlakukan bagi pelaku kejahatan yang membuat korban terbunuh makin banyak nyawa orang tidak bersalah akan hilang sia	Non-cyberbullying

3.3.4 Stemming Dataset

Stemming adalah proses mengubah kata menjadi kata dasar. Pada umumnya kata dalam dokumen memiliki variasi kombinasi imbuhan kata yang beragam, seperti imbuhan awalan, akhiran, sisipan, dan kombinasi. Kata-kata tersebut perlu diseragamkan menjadi kata dasar supaya seragam dan mengurangi kompleksitas kata. *Stemming* ini menggunakan algoritma Nazief dan Adriani untuk stemming kata-kata bahasa Indonesia.

Selama proses *stemming*, dataset komentar akan diseragamkan kata-katanya menjadi kata-kata dasar. Setiap komentar diproses satu per satu melalui *stemming*. Mengingat jumlah datasetnya mencapai ribuan, maka proses *stemming* bisa berlangsung beberapa menit. Lalu hasilnya akan disimpan pada kolom *stemming* tabel dataset komentar di database.

Pada tabel 3.5 diberikan ilustrasi contoh bagaimana *stemming* bekerja. Pada tabel tersebut terdapat 4 kolom meliputi no, sebelum *stemming*, setelah *stemming*,

dan kategorinya. Pada kolom sebelum *stemming* merupakan teks komentar hasil dari normalisasi bahasa gaul yang sebelumnya telah berhasil dilakukan. Sementara kolom setelah *stemming* merupakan hasil komentar setelah dilakukan *stemming*. Berikut disajikan proses *stemming* pada tabel 3.5 sebagai berikut:

Tabel 3.5
Stemming Dataset

No	Sebelum Stemming	Setelah Stemming	Kategori
1	orang kampungan kalau mau eksis ya begitu caranya kampungan juga tawa	orang kampung eks kampung tawa	<i>Cyberbullying</i>
2	ya allah anak anak apa tidak ingat dosa	allah anak anak tidak dosa	<i>Non-cyberbullying</i>
3	orang orang macam ini harus di didik dalam lapas tidak bakal bisa di sekolah formal kalau bisa langsung seumur hidup	orang orang didik lapas tidak sekolah formal langsung umur hidup	<i>Non-cyberbullying</i>
4	klithih itu binatang apa heran dibiarkan hidup	klithih binatang heran biar hidup	<i>Cyberbullying</i>
5	tembak aja entar tuanya bisa lebih sadis lagi	tembak aja entar tua sadis	<i>Cyberbullying</i>
6	kalau hukuman mati tidak di berlakukan bagi pelaku kejahanan yang membuat korban terbunuh makin banyak nyawa orang tidak bersalah akan hilang sia	hukum mati tidak laku bagi laku jahat korban bunuh nyawa orang tidak salah hilang sia	<i>Non-cyberbullying</i>

3.3.5 Stopword Dataset

Stopword-list adalah kata-kata yang sering muncul dan tidak memiliki arti. Daftar kata *stopwords* yakni kumpulan kata yang tidak memiliki arti dan disimpan dalam kamus khusus *stopwords*, seperti ‘yang’, ‘bahkan’, ‘akhirnya’, ‘di’, dan sebagainya. Dengan dibuangkan *stopwords*, ukuran kosakata menjadi berkurang sehingga hanya kata-kata penting dan memiliki bobot tinggi yang terdapat dalam dokumen. Pada tahap *stopword-list*, kata-kata yang tidak penting dibuang dari daftar kata, misalnya kata “yang”, “dimana”, “mengapa”, “yaitu”, “yakni”, dan sebagainya.

Stopword ini menggunakan *stopword* dari Fadillah Z. Tala disusun bersamaan dengan penyusunan tesis master yang bersangkutan. *Stopword* ini disaring dengan meneliti koran kompas.com dari Januari 2001 hingga Desember 2001. Hasilnya daftar *stopword*. Buku tesis beserta *stopword*nya bisa diakses pada tautan berikut ini.³⁹

Selama proses *stopword*, dataset komentar akan dibersihkan dari kata-kata yang tidak bermakna seperti “yang”, “dan”, “di”, dan sebagainya. Setiap komentar diproses satu per satu melalui *stopword*. Mengingat jumlah datasetnya mencapai ribuan, maka proses *stopword* bisa berlangsung beberapa menit. Lalu hasilnya akan disimpan pada kolom *stopword* tabel dataset komentar di database.

Pada tabel 3.6 diberikan ilustrasi contoh bagaimana *stopword* bekerja. Pada tabel tersebut terdapat 4 kolom meliputi no, sebelum *stopword*, setelah *stopword*, dan kategori. Pada kolom sebelum *stopword* merupakan teks komentar hasil dari *stemming* yang sebelumnya telah berhasil dilakukan. Sementara kolom setelah *stopword* merupakan hasil komentar setelah dilakukan *stopword*. Berikut disajikan proses *stopword* pada dataset komentar pada tabel 3.6:

Tabel 3.6
Stopword Dataset

No	Sebelum Stopword	Setelah Stopword	Kategori
1	orang kampung eks kampung tawa	orang kampung eks kampung tawa	<i>Cyberbullying</i>
2	allah anak anak tidak dosa	allah anak anak tidak dosa	<i>Non-cyberbullying</i>
3	orang orang didik lapas tidak sekolah formal langsung umur hidup	orang orang didik lapas tidak sekolah formal langsung umur hidup	<i>Non-cyberbullying</i>
4	klithih binatang heran biar hidup	klithih binatang heran hidup	<i>Cyberbullying</i>
5	tembak aja entar tua sadis	tembak tua sadis	<i>Cyberbullying</i>
6	hukum mati tidak laku bagi laku jahat korban bunuh nyawa orang tidak salah hilang sia	hukum mati tidak laku laku jahat korban bunuh nyawa orang tidak salah hilang sia	<i>Non-cyberbullying</i>

³⁹ <https://eprints.llc.uva.nl/740/1/MoL-2003-02.text.pdf> diakses 05 Januari 2021.

3.3.6 Tokenizing Dataset

Tokenizing adalah proses pemotongan dokumen menjadi kata-kata setelah menjadi proses *filtering*. Pemecahan kalimat dan kata dilakukan berdasarkan pada spasi di dalam kalimat atau paragraf. Hasil pemotongan kata-kata tersebut dijadikan kumpulan kata dan membentuk daftar kata. Hasil pemotongan kata tersebut disebut *token*.

Selama proses *tokenizing*, dataset komentar akan dipisahkan berdasarkan spasi hingga membentuk daftar kata. Setiap komentar diproses satu per satu melalui *tokenizing*. Mengingat jumlah datasetnya mencapai ribuan, maka proses *tokenizing* bisa berlangsung beberapa menit. Lalu hasilnya akan disimpan pada kolom *tokenizing* tabel dataset komentar di database.

Pada tabel 3.7 diberikan ilustrasi contoh bagaimana *tokenizing* bekerja. Pada tabel tersebut terdapat 4 kolom meliputi no, sebelum *tokenizing*, setelah *tokenizing*, dan kategori. Pada kolom sebelum *tokenizing* merupakan teks komentar hasil dari *stopword* yang sebelumnya telah berhasil dilakukan. Sementara kolom setelah *tokenizing* merupakan hasil komentar setelah dilakukan *tokenizing*. Proses *tokenizing* dalam sistem ini dilakukan dengan menggunakan hasil dari dataset setelah dilakukan *stopword*.

Tabel 3.7
Tokenizing Dataset

No	Sebelum Tokenizing	Setelah Tokenizing	Kategori
1	orang kampung eks kampung tawa	✓ orang ✓ kampung ✓ eks ✓ kampung ✓ tawa	<i>Cyberbullying</i>
2	allah anak anak tidak dosa	✓ allah ✓ anak ✓ anak ✓ tidak ✓ dosa	<i>Non-cyberbullying</i>
3	orang orang didik lapas tidak sekolah formal langsung umur hidup	✓ orang ✓ orang ✓ didik ✓ lapas	<i>Non-cyberbullying</i>

Tabel 3.7
(Lanjutan)

No	Sebelum Tokenizing	Setelah Tokenizing	Kategori
		<ul style="list-style-type: none"> ✓ tidak ✓ sekolah ✓ formal ✓ langsung ✓ umur ✓ hidup 	
4	klithih binatang heran biar hidup	<ul style="list-style-type: none"> ✓ klithih ✓ binatang ✓ heran ✓ hidup 	<i>Cyberbullying</i>
5	tembak aja entar tua sadis	<ul style="list-style-type: none"> ✓ tembak ✓ tua ✓ sadis 	<i>Cyberbullying</i>
6	hukum mati tidak laku bagi laku jahat korban bunuh nyawa orang tidak salah hilang sia	<ul style="list-style-type: none"> ✓ hukum ✓ mati ✓ tidak ✓ laku ✓ laku ✓ jahat ✓ korban ✓ bunuh ✓ nyawa ✓ orang ✓ tidak ✓ salah ✓ hilang ✓ sia 	<i>Non-cyberbullying</i>

3.3.7 N-gram Dataset

N-gram merupakan proses yang secara luas digunakan dalam pengolahan teks dan bahasa. *N-gram* merupakan kumpulan kata yang diberikan dalam sebuah paragraf dan ketika menghitung *n-gram* biasanya dilakukan dengan menggerakkan satu kata maju ke depan. *N-gram* memberikan probabilitas atau kemungkinan kata-kata berikutnya yang mungkin bisa digunakan dalam pengabungan pada keseluruhan kalimat. Jika diasumsikan X adalah jumlah kata dalam suatu kalimat k dan n adalah jumlah *n-gram* (*unigram* = 1, *bigram* = 2, *trigram* = 3), maka jumlah *n-gram* dari kalimat K adalah $n\text{-gram}\text{-}k = x - (n-1)$.

N-gram biasanya digunakan dalam berbagai macam proses pengolahan bahasa. Adapun tipe *n-gram* yang biasanya digunakan seperti unigram, bigram, dan trigram. *N-gram* biasanya digunakan dalam proses *supervised machine learning* seperti multinomial naive bayes. Dalam tugas akhir ini, *n-gram* digunakan karena biasanya kata-kata dalam bahasa Indonesia memiliki keterkaitan antar kata seperti kata-kata yang membentuk suatu frasa. Adapun *n-gram* yang digunakan dalam tugas akhir ini meliputi *unigram* dan *bigram*.

Pada tabel 3.8 diberikan ilustrasi contoh bagaimana *ngram* bekerja. Pada tabel tersebut terdapat 4 kolom meliputi no, sebelum *n-gram*, setelah *n-gram*, dan kategori. Pada kolom sebelum *n-gram* merupakan teks komentar hasil dari *tokenizing* yang sebelumnya telah berhasil dilakukan. Hasil *tokenizing* juga sekaligus mewakili hasil *unigram*. Sementara kolom setelah *ngram* merupakan hasil komentar setelah dilakukan *bigram*. Berikut disajikan hasil sebelum dan setelah *ngram* selengkapnya sebagai berikut:

Tabel 3.8
Bigram Dataset

No	Sebelum Bigram	Setelah Bigram	Kategori
1	<ul style="list-style-type: none"> ✓ orang ✓ kampung ✓ eks ✓ kampung ✓ tawa 	<ul style="list-style-type: none"> ✓ orang kampung ✓ kampung eks ✓ eks kampung ✓ kampung tawa 	<i>Cyberbullying</i>
2	<ul style="list-style-type: none"> ✓ allah ✓ anak ✓ anak ✓ tidak ✓ dosa 	<ul style="list-style-type: none"> ✓ allah anak ✓ anak anak ✓ anak tidak ✓ tidak dosa 	<i>Non-cyberbullying</i>
3	<ul style="list-style-type: none"> ✓ orang ✓ orang ✓ didik ✓ lapas ✓ tidak ✓ sekolah ✓ formal ✓ langsung ✓ umur ✓ hidup 	<ul style="list-style-type: none"> ✓ orang orang ✓ orang didik ✓ didik lapas ✓ lapas tidak ✓ tidak sekolah ✓ sekolah formal ✓ formal langsung ✓ langsung umur ✓ umur hidup 	<i>Non-cyberbullying</i>

Tabel 3.8
(Lanjutan)

No	Sebelum Bigram	Setelah Bigram	Kategori
4	✓ klithih ✓ binatang ✓ heran ✓ hidup	✓ klithih binatang ✓ binatang heran ✓ heran hidup	<i>Cyberbullying</i>
5	✓ tembak ✓ tua ✓ sadis	✓ tembak tua ✓ tua sadis	<i>Cyberbullying</i>
6	✓ hukum ✓ mati ✓ tidak ✓ laku ✓ laku ✓ jahat ✓ korban ✓ bunuh ✓ nyawa ✓ orang ✓ tidak ✓ salah ✓ hilang ✓ sia	✓ hukum mati ✓ mati tidak ✓ tidak laku ✓ laku laku ✓ laku jahat ✓ jahat korban ✓ korban bunuh ✓ bunuh nyawa ✓ nyawa orang ✓ orang tidak ✓ tidak salah ✓ salah hilang ✓ hilang sia	<i>Non-cyberbullying</i>

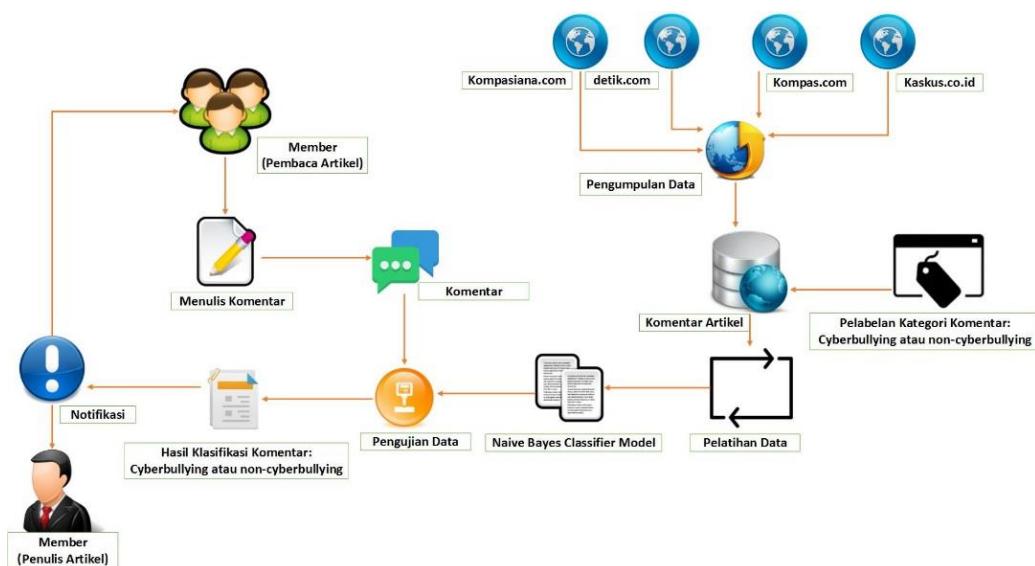
BAB IV

MULTINOMINAL NAIVE BAYES

Pada Bab IV dijelaskan tentang bagaimana penerapan *Multinomial Naive Bayes* dalam mengklasifikasikan komentar menjadi kategori *cyberbullying* dan kategori *non-cyberbullying*. Pada bab ini dijelaskan arsitektur sistem penyaring *cyberbullying*, arsitektur pelatihan model, dan arsitektur pengujian model.. Selain itu, pada bab ini akan dijelaskan ilustrasi contohnya perhitungan *Multinomial Naive Bayes*. Berikut penjelasan selengkapnya:

4.1 Arsitektur Sistem Penyaring Komentar Cyberbullying

Pada sub bab ini dijelaskan arsitektur sistem penyaring komentar *cyberbullying*. Arsitektur sistem penyaring komentar *cyberbullying* merupakan gambaran kerangka sistem dalam mengklasifikasi komentar menjadi *cyberbullying* dan *non-cyberbullying*. Sistem ini terdiri atas empat tahapan meliputi pengumpulan dataset, pelatihan data komentar, pengujian data komentar, dan notifikasi. Selengkapnya diilustrasikan pada gambar 4.1:



Gambar 4.1
Arsitektur Penyaring Komentar Cyberbullying

Adapun proses kerjanya sebagai berikut: pertama, pengumpulan dataset komentar dilakukan secara manual melalui *browsing* data dengan mengumpulkan berbagai komentar dari situs portal berita dan *blog* seperti detik.com, kompasiana.com, kaskus.com, kompas.com, dan youtube.com. Kemudian data komentar yang terkumpul disimpan ke database. Berikutnya data komenter tersebut diberikan label kategori berupa *cyberbullying* untuk komentar yang teridentifikasi mengandung ujaran *cyberbullying* atau *non-cyberbullying* untuk komentar yang tidak teridentifikasi mengandung ujaran *cyberbullying*. Berikutnya *dataset* komentar tersebut akan diproses menjadi *dataset* untuk model pelatihan dan pengujian data.

Kedua, pelatihan data adalah sistem yang berfungsi untuk melatih pola data terhadap berbagai komentar yang berpotensi mengandung *cyberbullying* menggunakan *Multinomial Naive Bayes*. Sehingga dihasilkan sebuah model klasifikasi *cyberbullying* bernama *Multinomial Naive Bayes Model*. Selama proses pelatihan data, *dataset* komentar akan diproses melalui *preprocessing* komentar seperti *case folding*, *cleaning*, normalisasi bahasa gaul, *stemming*, *stopword*, dan *tokenizing*. Kemudian hasil *preprocessing* dataset komentar akan diolah menjadi model fitur kelas kata dan model *prior* dimana kedua model tersebut akan digunakan untuk menghitung kategori komentar.

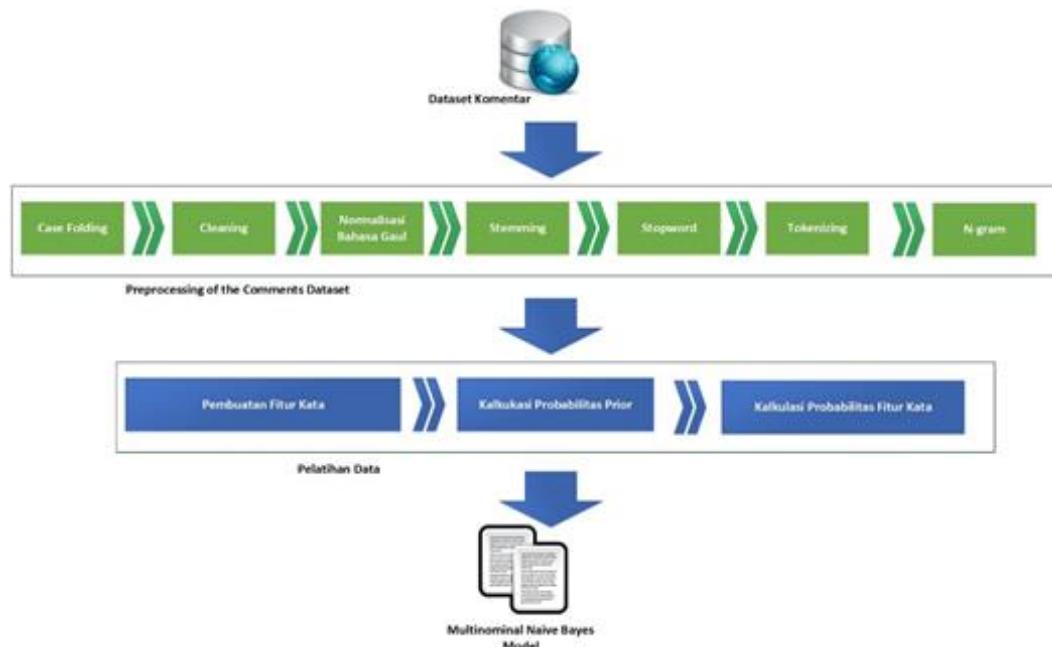
Ketiga, tahapan selanjutnya ialah pengujian data komentar. Pada tahapan ini adalah proses mengidentifikasi berbagai komentar yang diinputkan member untuk dianalisis apakah komenter tersebut berpotensi mengandung *cyberbullying* atau tidak. Pada tahap ini, berbagai data komentar yang diinputkan ke sistem akan dihitung kategorinya menggunakan model data pelatihan. Sehingga menghasilkan kategori komentar.

Keempat, notifikasi yakni proses memberitahukan kepada member bahwa terdapat komentar yang berpotensi mengandung *cyberbullying*. Sehingga diharapkan member bisa memikirkan lebih lanjut apakah akan mengirim komentarnya ke artikel atau tidak. Pada tahap ini, jika teridentifikasi komentar yang diinputkan member memiliki kategori *cyberbullying*, maka sistem akan memberikan pesan notifikasi peringatan kepada member terkait bahaya dari

tindakan *cyberbullying*. Selanjutnya, member yang mendapatkan pesan peringatan itu bisa melakukan tindakan berupa membatalkan pengiriman komentar atau melanjutkannya dengan segala resikonya. Khusus bagi penulis artikel dapat menonaktifkan komentar yang mengandung kategori *cyberbullying* melalui halaman *back end*. Sehingga komentar tersebut tidak akan ditampilkan pada *front end* tampilan artikel website.

4.2 Arsitektur Sistem Pelatihan Model

Pada sub bab ini dijelaskan arsitektur sistem pelatihan model penyaring komentar *cyberbullying*. Arsitektur sistem pelatihan model menggambarkan bagaimana membentuk *Multinomial Naive Bayes Model* dengan mengambil *dataset* komentar. Pada gambar 4.2, proses pelatihan data dimulai dengan mengambil *dataset* komentar dari *database*. Dilanjutkan *preprocessing dataset* komentar. Kemudian dilakukan pelatihan data meliputi, pembentukan fitur kata, perhitungan probabilitas *prior*, dan perhitungan probabilitas fitur kata. Berikut ini merupakan runtutan proses pelatihan data:



Gambar 4.2
Arsitektur Pelatihan Data

1. *Preprocessing dataset* komentar

Pada tahapan ini, *dataset* komentar yang sebelumnya telah dikumpulkan dari berbagai situs diolah melalui *preprocessing*. Proses ini melibatkan *case folding*, *cleaning*, normalisasi bahasa gaul, *stemming*, *stopword*, *tokenizing* dan *n-gram*. Mengingat datasetnya berjumlah ribuan komentar, maka selama proses ini akan memakan waktu cukup lama. Hasilnya menghasilkan sejumlah *token dataset* komentar, yakni kumpulan kata-kata dari daftar dataset baik secara *unigram* (1 kata) maupun *bigram* (2 kata).

2. Pembentukan fitur kata

Pada tahapan ini, hasil dari *preprocessing* dataset komentar menjadi dokumen bersih berupa *token* kata dibentuk menjadi fitur kata. Fitur kata ialah kata kunci dari komentar yang memiliki nilai kategori baik kategori *cyberbullying* maupun kategori *non-cyberbullying*. Fitur kata tersebut dihitung frekuensi kemunculan katanya dari tiap kategori baik kategori *cyberbullying* maupun kategori *non-cyberbullying*. Kemudian hasil pembentukan fitur kata disimpan ke *database*.

3. Perhitungan probabilitas *prior*

Tahapan perhitungan *prior* dilakukan setelah melakukan pembentukan fitur kata. Pada tahapan ini, terjadi perhitungan nilai probabilitas *prior* berdasarkan hasil pembentukan fitur kata. Dari perhitungan probabilitas *prior* tersebut menghasilkan nilai probabilitas tiap kategori baik kategori *cyberbullying* maupun kategori *non-cyberbullying*. Kemudian hasil perhitungan pribabilitas prior disimpan ke *database*.

4. Perhitungan probabilitas fitur kata

Setelah berhasil menghitung *prior*, tahapan selanjutnya ialah perhitungan probabilitas fitur kata. Pada tahapan ini, setiap fitur kata dihitung nilai probabilitasnya baik dari kategori *cyberbullying* maupun kategori *non-cyberbullying*. Selanjutnya hasil perhitungan probilitas fitur kata disimpan pada *database*. Probabilitas fitur kata akan dipakai dalam perhitungan nilai Cmap untuk menentukan kategori komentar.

Keempat tahapan tersebut dilakukan secara hirarkis dalam mengolah *dataset* komentar hingga menghasilkan model. Selama proses pelatihan data ini memerlukan waktu yang cukup lama beberapa menit bahkan jam. Tergantung jumlah *dataset* komentarnya. Semakin banyak *dataset* yang dilatih, waktu pelatihannya akan semakin lama. Berikutnya dijelaskan bagaimana proses pelatihan data pada tiap tahapannya selengkapnya sebagai berikut:

4.2.1 Preprocessing Dataset Komentar

Pada sub bab ini dijelaskan tentang *preprocessing* dataset komentar. Setelah *dataset* komentar berhasil dikumpulkan dari berbagai situs berita dan *blog*, *dataset* tersebut diolah melalui sejumlah tahapan meliputi *case folding*, *cleaning*, normalisasi bahasa gaul, *stemming*, *stopword*, *tokenizing* dan *n-gram*. Dalam ilustrasi ini, *n-gram* menggunakan fitur *unigram*. Sehingga menghasilkan *token* fitur kata yang masing-masing fitur kata terdiri atas 1 kata. *Token* fitur kata tersebut akan digunakan untuk pembuatan model lebih lanjut.

Inti dari *preprocessing* dataset komentar ialah menyederhanakan komentar menjadi daftar kata sebelum digunakan dalam proses pelatihan data. Proses *preprocessing* bisa berlangsung cukup lama karena besarnya dataset komentar. Semakin besar *dataset*, semakin lama waktu *preprocessing*. Hasilnya dari pengolahan ini selanjutnya disimpan pada tabel dataset komentar di database. Berikut ilustrasi contohnya ditunjukkan pada tabel 4.1:

Tabel 4.1
Hasil Preprocessing Data Komentar

Data	Sebelum Preprocessing	Setelah Preprocessing	Kategori
D1	Orangng kampungan.. Kalo mau eksis ya gitu.. Caranya kampungan juga.. Hahahhahahaha	orang, kampung, eks, kampung, tawa	Cyberbullying
D2	Ya Allah .. nak nak opo gak eling dosa	allah, anak, anak, tidak, dosa	Non-cyberbullying
D3	Orang-orang macam ini harus di didik dalam lapas, gak bakal bisa di sekolah formal.	orang, orang, didik, lapas, tidak, sekolah, formal, langsung, umur, hidup	Non-cyberbullying

Tabel 4.1
(Lanjutan)

Data	Sebelum Preprocessing	Setelah Preprocessing	Kategori
	Klo bisa langsung seumur hidup.		
D4	Klithih itu binatang apa sih? Koq dibiarkan hidup...?	klithih, binatang, heran, hidup	Cyberbullying
D5	Tembak aja...ntar tuanya bs lebih sadis lg	tembak, tua, sadis	Cyberbullying
D6	Kalau Hukuman mati tidak di berlakukan pagi pelaku kejahatan yg membuat korban terbunuh. Makin banyak nyawa orang tidak bersalah akan hilang sia2.	hukum, mati, tidak, laku, laku, jahat, korban, bunuh, nyawa, orang, tidak, salah, hilang, sia	Non-cyberbullying

Pada tabel 4.1 diberikan ilustrasi contoh bagaimana *preprocessing* bekerja. Pada tabel tersebut terdapat 4 kolom meliputi no, sebelum *preprocessing*, setelah *preprocessing*, dan kategori komentar. Pada kolom sebelum *preprocessing* merupakan teks komentar hasil asli yang diambil dari salah satu situs sumber data yakni detik.com. Sementara kolom setelah *preprocessing* merupakan hasil komentar setelah dilakukan *preprocessing*. Sementara kolom kategori menyatakan kelas komentar termasuk *cyberbullying* atau *non-cyberbullying*.

4.2.2 Pembentukan Fitur Kata

Pada sub bab ini dijelaskan pembentukan fitur kata dari pelatihan data. Fitur kata merupakan kumpulan kata unik yang berisi frekuensi kemunculannya dari seluruh dataset komentar. Fitur kata dibentuk dengan menggunakan hasil dari *preprocessing dataset*. Adapun tahapan pembentukan fitur kata meliputi:

Pertama, sistem mengambil dataset hasil dari *preprocessing dataset*. Kedua, sistem mengelompokan kata-kata dalam dataset sesuai dengan fitur kata (kata yang unik) untuk kemudian disimpan ke *database*. Ketiga, setiap fitur kata akan dihitung frekuensi kemunculan dari tiap dokumen baik dari kategori *cyberbullying* maupun kategori *non-cyberbullying*. Kemudian hasilnya perhitungan frekuensi katanya akan disimpan kembali ke *database*.

Pada tabel 4.2 merupakan ilustrasi hasil dari pembentukan fitur kata. Kolom fitur kata merupakan kata-kata yang unik setelah mengolah dataset *preprocessing*. Kolom Jumlah *cyberbullying* menyatakan frekuensi kemunculan fitur kata dari kategori *cyberbullying*. Kolom Jumlah *non-cyberbullying* menyatakan frekuensi kemunculan fitur kata dari kategori *non-cyberbullying*. Selengkapnya ditunjukkan pada tabel 4.2 sebagai berikut:

Tabel 4.2
Pembentukan Fitur Kata

Fitur Kata	Jumlah Cyberbullying	Jumlah Non-Cyberbullying
orang	1	3
kampung	2	0
eks	1	0
tawa	1	0
allah	0	1
anak	0	2
tidak	0	4
dosa	0	1
didik	0	1
lapas	0	1
sekolah	0	1
formal	0	1
langsung	0	1
umur	0	1
hidup	1	1
klithih	1	0
binatang	1	0
heran	1	0
tembak	1	0
tua	1	0
sadis	1	0
hukum	0	1
mati	0	1
laku	0	1
Jahat	0	1
korban	0	1
bunuh	0	1
nyawa	0	1
hilang	0	1
sia	0	1
salah	0	1

4.2.3 Perhitungan Probabilitas Prior

Pada sub bab ini dijelaskan perhitungan probabilitas *prior*. Probabilitas *prior* merupakan perbandingan total antara jumlah dokumen komentar yang berkategori *cyberbullying* dan jumlah dokumen komentar yang berkategori *non-cyberbullying*. Proporsi antara dokumen komentar *cyberbullying* dan dokumen komentar *non-cyberbullying* dihitung berdasarkan total dataset komentar yang digunakan dalam proses pelatihan data. Perhitungan probabilitas *prior* dilakukan setelah perhitungan fitur kata.

Probabilitas *prior* dihitung berdasarkan perbandingan jumlah data komentar *cyberbullying* dibandingkan total seluruh data dan jumlah data komentar *non-cyberbullying* dibandingkan total seluruh data. Sehingga menghasilkan 2 kategori meliputi kategori *cyberbullying* dan kategori *non-cyberbullying*. Selanjutnya probabilitas prior akan digunakan dalam menghitung Cmap dalam menentukan kategori suatu komentar. Dalam hal ini, probabilitas prior dihitung menggunakan persamaan:

Berikut ini merupakan penjelasan dari persamaannya sebagai berikut:

- $P(C) = \text{probabilitas prior}$
 - $N = \text{jumlah seluruh dokumen komentar yang digunakan dalam pelatihan.}$
 - $N_c = \text{jumlah dokumen yang memiliki kategori } c$

Pada tabel 4.3 dijelaskan tentang ilustrasi probabilitas *prior*. Tabel tersebut terdiri atas tiga kolom meliputi kategori, jumlah dokumen, dan probabilitas kategori. Kolom kategori menyatakan kategori dari *dataset* komentar dimana dalam hal ini dataset komentar hanya memiliki dua macam kategori yakni *cyberbullying* dan *non-cyberbullying*. Sementara kolom jumlah dokumen menyatakan total dokumen komentar dari tiap kategori dimana dalam hal ini masing-masing kategori *cyberbullying* dan *non-cyberbullying* memiliki tiga dokumen. Sedangkan kolom probabilitas kategori menyatakan hasil dari perhitungan probabilitas kategori

dimana dalam hal ini masing-masing kategori *cyberbullying* dan kategori *non-cyberbullying* memiliki probabilitas sebesar 0,5. Selengkapnya ditunjukan pada tabel 4.3 sebagai berikut:

Tabel 4.3
Hasil Perhitungan Probabilitas *Prior*

Kategori	Jumlah Dokumen	Probabilitas <i>Prior</i>
<i>Cyberbullying</i>	3	$\frac{3}{6} = 0,5$
<i>Non-cyberbullying</i>	3	$\frac{3}{6} = 0,5$

4.2.4 Perhitungan Probabilitas Fitur Kata

Pada sub ini dijelaskan perhitungan probabilitas fitur kata. Tahapan peritungan probabilitas fitur kata dilakukan setelah melakukan perhitungan *prior*. Probabilitas fitur kata merupakan probabilitas kemunculan kategori *cyberbullying* dan kategori *non-cyberbullying* pada tiap fitur kata. Probabilitas fitur kata dihitung berdasarkan total *dataset* komentar yang digunakan dalam proses pelatihan data.

Setelah fitur kata berhasil dibentuk, selanjutnya dihitung masing-masing nilai probabilitas fitur dari tiap kategorinya. Setiap fitur kata dihitung probabilitas fitur katanya lalu hasilnya disimpan ke *database*. Sehingga menghasilkan *Multinomial Naive Bayes Model*. Adapun untuk menghitung probabilitas fitur kata menggunakan persamaan berikut:

$$P(w | c) = \frac{\text{count}(w, c)+1}{\text{count}(c)+|V|} \quad \dots \dots \dots \quad (4.2)$$

Berikut ini merupakan penjelasan dari persamaannya:

- $P(w | c)$ = probabilitas fitur kata dari tiap kategori c
- $\text{count}(w, c)$ = total kemunculan fitur kata pada kategori c
- $|V|$ = total fitur kata yang unik dari seluruh kategori
- $\text{count}(c)$ = jumlah fitur kata dari kategori c

Pada tabel 4.4 merupakan ilustrasi contoh perhitungan probabilitas fitur kata. Tabel tersebut terdiri atas tiga kolom meliputi kolom fitur kata, kolom $P(\text{Kata}|\text{Cyberbullying})$, dan kolom $P(\text{Kata}|\text{Non-cyberbullying})$. Berdasarkan fitur kata dari tabel 4.2 diketahui bahwa. Total kata unik atau $|V|$ berjumlah 30. Total kata kategori *cyberbullying* atau count(*cyberbullying*) berjumlah 11. Total kata kategori *non-cyberbullying* atau count(*non-cyberbullying*) berjumlah 22.

Tabel 4.4
Perhitungan Probabilitas Fitur Kata

Fitur Kata	$P(\text{Kata} \text{Cyberbullying})$	$P(\text{Kata} \text{Non-Cyberbullying})$
orang	$\frac{1+1}{30+11} = 0,0487$	$\frac{1+3}{30+22} = 0,0769$
kampung	$\frac{1+2}{30+11} = 0,0731$	$\frac{1+0}{30+22} = 0,0192$
eks	$\frac{1+1}{30+11} = 0,0487$	$\frac{1+0}{30+22} = 0,0192$
tawa	$\frac{1+1}{30+11} = 0,0487$	$\frac{1+0}{30+22} = 0,0192$
allah	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
anak	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+2}{30+22} = 0,0576$
Tidak	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+4}{30+22} = 0,0961$
Dosa	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
Didik	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
lapas	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
sekolah	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
formal	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
langsung	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
umur	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
Hidup	$\frac{1+1}{30+11} = 0,0487$	$\frac{1+1}{30+22} = 0,0384$
klithih	$\frac{1+1}{30+11} = 0,0487$	$\frac{1+0}{30+22} = 0,0192$

Tabel 4.4
(Lanjutan)

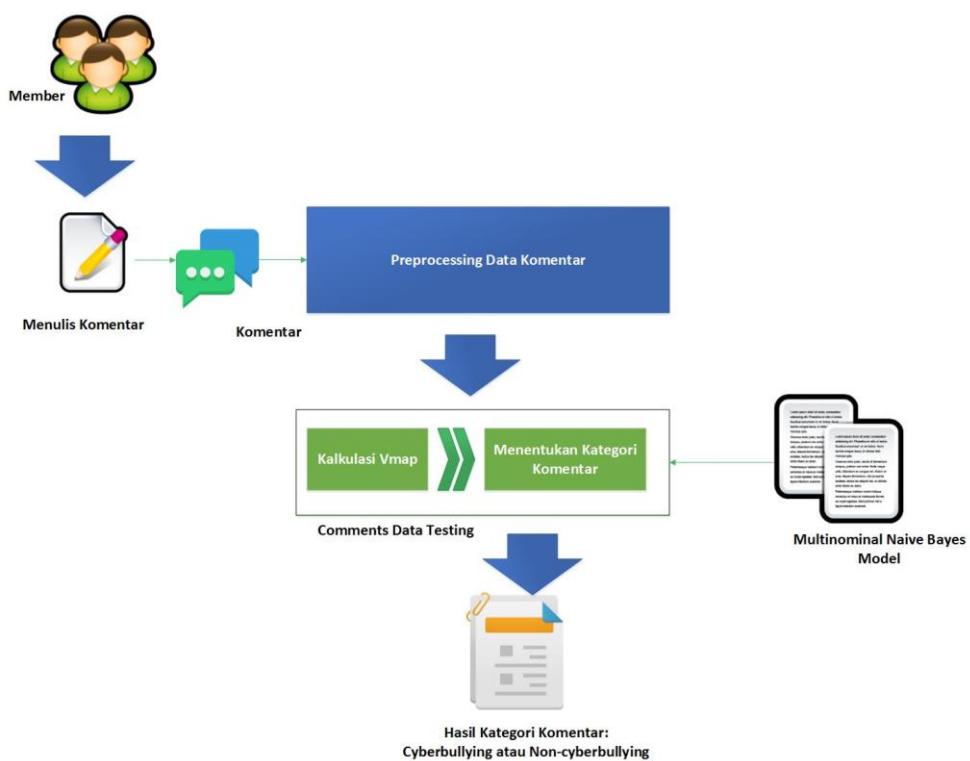
Fitur Kata	P(Kata Cyberbullying)	P(Kata Non-Cyberbullying)
binatang	$\frac{1+1}{30+11} = 0,0487$	$\frac{1+0}{30+22} = 0,0192$
heran	$\frac{1+1}{30+11} = 0,0487$	$\frac{1+0}{30+22} = 0,0192$
tembak	$\frac{1+1}{30+11} = 0,0487$	$\frac{1+0}{30+22} = 0,0192$
tua	$\frac{1+1}{30+11} = 0,0487$	$\frac{1+0}{30+22} = 0,0192$
sadis	$\frac{1+1}{30+11} = 0,0487$	$\frac{1+0}{30+22} = 0,0192$
hukum	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
mati	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
laku	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
jahat	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
korban	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
bunuh	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
nyawa	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
Hilang	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
Sia	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$
Salah	$\frac{1+0}{30+11} = 0,0243$	$\frac{1+1}{30+22} = 0,0384$

4.3 Arsitektur Pengujian Data

Pada sub bab ini dijelaskan perhitungan pengujian model. Pengujian data dilakukan setelah hasil dibuat, *Multinomial Naive Bayes Model* berhasil dibentuk. Selanjutnya dilakukan perhitungan Cmap menggunakan model fitur kata dan model prior untuk menentukan bobot pada masing-masing kategori komentar baik *cyberbullying* maupun *non-cyberbullying*. Setelah nilai Cmap dari masing-masing kategori komentar diketahui hasilnya, maka hasil Cmap dari *cyberbullying* dan hasil

Cmap dari *non-cyberbullying* dibandingkan. Hasil Cmap dengan nilai terbesar selanjutnya menjadi hasil klasifikasi kategori komentar.

Pada gambar 4.3 ditunjukkan arsitektur pengujian data komentar. Sistem ini dimulai ketika member menginputkan komentar pada sistem. Kemudian sistem akan memproses komentar melalui *preprocessing* komentar. Hasilnya berupa *token* komentar akan diproses untuk dihitung Cmap dan ditentukan kategori komentarnya berdasarkan *Multinomial Naive Baye Model*. Sehingga hasilnya adalah keluaran kategori komentar bisa berbentuk kategori *cyberbullying* atau kategori *non-cyberbullying*. Berikut arsitektur pengujian data komentar pada gambar 4.3 sebagai berikut:



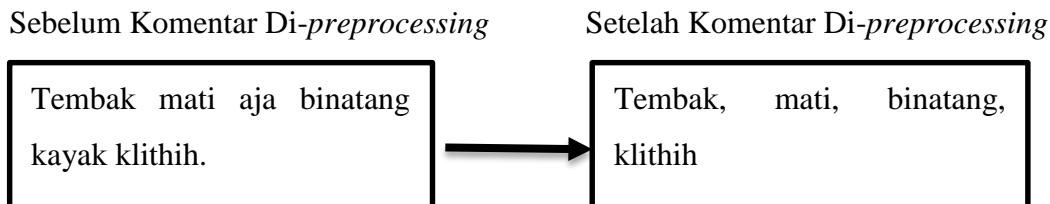
Gambar 4.3
Arsitektur Pengujian Data Komentar

Berikutnya dijelaskan tentang proses pengujian komentar mulai dari *preprocessing* komentar, kalkulasi kategori komentar, hingga menentukan kategori komentar. *Preprocessing* di sini ialah *proprocessing* komentar yang dilakukan terhadap data uji. Sementara kalkulasi kategori komentar merujuk pada perhitungan

nilai Cmap dari masing-masing kategori komentar baik *cyberbullying* maupun *non-cyberbullying*. Selengkapnya dijelaskan sebagai berikut:

1. Lakukan *preprocessing* pada komentar yang diuji

Pada tahap ini, member menginputkan data komentar yang akan diujikan. Data komentar tersebut selanjutnya dilakukan *preprocessing* hingga menghasilkan *token* fitur kata. Dalam hal ini, mengingat pelatihan datanya menggunakan fitur *unigram*, maka pengujian data juga menggunakan fitur *unigram*. Hasilnya berupa *token* kata yang terdiri atas masing-masing 1 kata.



Gambar 4.4
Data Uji Sebelum dan Setelah Di-*preprocessing*

2. Hitung Cmap dari masing-masing kategori

Setelah dilakukan *preprocessing* data komentar, tahapan selanjutnya ialah menghitung nilai Cmap. Pada tahap ini, token hasil *preprocessing* data komentar dikalkukasi kategorinya menggunakan *Multinomial Naive Bayes Model* yang sebelumnya sudah dibuat untuk menentukan nilai Cmap. Sehingga menghasilkan nilai Cmap dari kategori komentar baik *cyberbullying* dan *non-cyberbullying*. Adapun perhitungan Cmap menggunakan persamaan:

$$Cmap = \arg \max P(c) \prod_i P(wi | c) \quad \dots \dots \dots \quad (4.3)$$

Berikut ini merupakan penjelasan dari persamaannya sebagai berikut:

- C_{map} = nilai bobot kategori komentar
 - $P(c)$ = probabilitas *prior*
 - $P(w | c)$ = probabilitas fitur kata dari tiap kategori komentar

Berdasarkan komentar pada gambar 4.4, selanjutnya dilakukan perhitungan Cmap pada masing-masing kategori menggunakan model fitur kata dan model *prior*. Selengkapnya sebagai berikut:

- Cmap untuk kategori *cyberbullying*

Hitunglah Cmap untuk kategori *cyberbullying*. $Cmap(cyberbullying) = P(cyberbullying) \times P(\text{tembak} | cyberbullying) \times P(\text{mati} | cyberbullying) \times P(\text{binatang} | cyberbullying) \times P(\text{klithih hukum} | cyberbullying) = \frac{1}{2} \times \frac{2}{41} \times \frac{1}{41} \times \frac{2}{41} \times \frac{2}{41} = 0,0000014155.$

- Cmap untuk kategori *non-cyberbullying*

Hitunglah Cmap untuk kategori *non-cyberbullying*. $Cmap(non-cyberbullying) = P(non-cyberbullying) \times P(\text{tembak} | non-cyberbullying) \times P(\text{mati} | non-cyberbullying) \times P(\text{binatang} | non-cyberbullying) \times P(\text{klithih hukum} | non-cyberbullying) = \frac{1}{2} \times \frac{1}{52} \times \frac{2}{52} \times \frac{1}{52} \times \frac{1}{52} = 0,0000001368.$

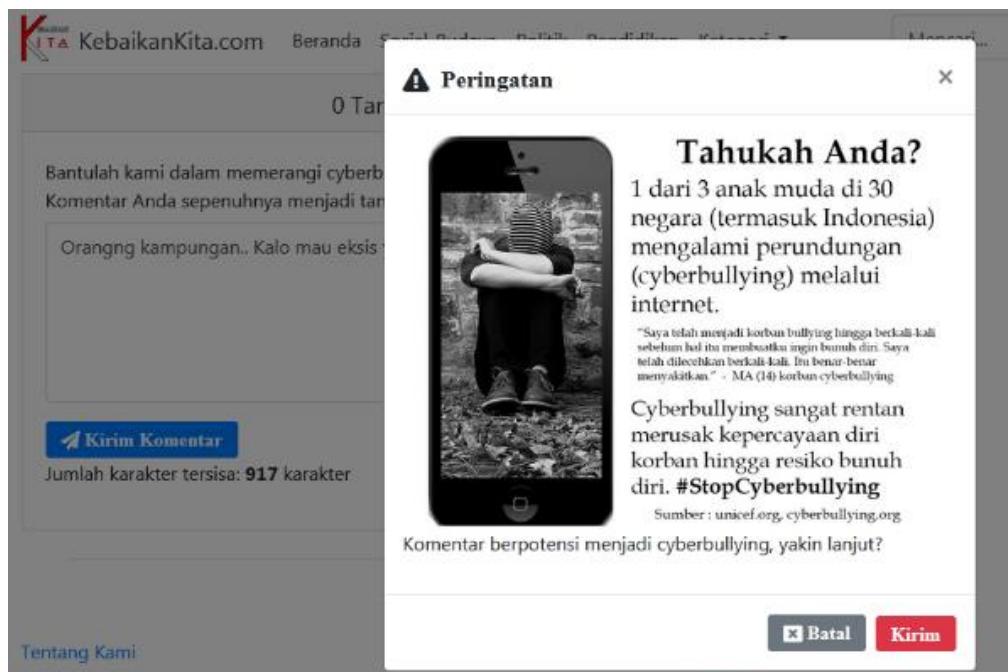
3. Tentukan hasil kalkulasi kategori

Terakhir dilakukan perbandingan antar nilai Cmap pada masing-masing kategori komentar. Pada tahap ini, hasil kalkulasi kategori selanjutnya dilakukan penentuan kategori. Jika hasil perhitungan kategori komentar *cyberbullying* > hasil perhitungan kategori komentar *non-cyberbullying*, berarti komentar tersebut diklasifikasikan menjadi kategori *cyberbullying*. Sebaliknya, jika hasil perhitungan kategori komentar *cyberbullying* < hasil perhitungan kategori komentar *non-cyberbullying*, berarti komentar tersebut diklasifikasikan menjadi kategori *non-cyberbullying*. Berdasarkan perhitungan Cmap di atas, karena kategori komentar *cyberbullying* bernilai 0,0000014155 > dibandingkan kategori komentar *non-cyberbullying* bernilai 0,0000001368, maka dapat disimpulkan bahwa komentar tersebut termasuk kategori *cyberbullying*.

4.4 Tampilan Daftar Komentar

Pada sub bab ini dijelaskan hasil tampilan komentar setelah sistem mengklasifikasikan komentar menjadi *cyberbullying* dan *non-cyberbullying*.

Setelah komentar mengalami proses perhitungan kategori *cyberbullying*, maka sistem akan menampilkan hasilnya pada antar muka dari situs baik di sisi *front end* maupun *back end*. Dalam hal ini, ketika komentar yang diinputkan member terdeteksi sebagai *cyberbullying*, maka sistem akan menampilkan pesan peringatan *cyberbullying* yang memberitahukan kepada member terkait bahaya *cyberbullying*. Dengan demikian diharapkan member dapat mempertimbangkan kembali komentarnya sebelum dikirim. Berikut ditunjukkan ilustrasi pesan peringatan terkait potensi bahaya *cyberbullying* dari komentar yang dikirimkan pengguna:



Gambar 4.5
Ilustrasi Peringatan Komentar Terdeteksi Cyberbullying

Akan tetapi jika member tidak mengindahkan peringatan sistem, maka sistem akan memberikan label *cyberbullying* terhadap komentar member yang mengandung *cyberbullying*. Komentar yang terdeteksi sebagai *cyberbullying* akan diberikan label kategori *cyberbullying* pada sisi *front end*. Member bisa menghapus komentar yang mengandung *cyberbullying* pada sisi *front end* situs. Berikut ditunjukkan ilustrasi pelabelan komentar pada sisi *front end* di situs kebaikankita.com:

5 Tanggapan Pembaca:

Bantulah kami dalam memerangi cyberbullying! Berkomentarlah secara bijaksana dan arif. Komentar Anda sepenuhnya menjadi tanggung jawab Anda seperti diatur dalam UU ITE.

 **Kirim Komentar**



[mikasa ackerman](#) | 2020-12-22 10:06:06

Orangng kampungan.. Kalo mau eksis ya gitu.. Caranya kampungan juga..
Hahahahaha

Balas

Laporkan

Hapus

Cyberbullying



[Danar Dono](#) | 2020-12-22 10:08:06

Ya Allah .. nak nak opo gak eling dosa

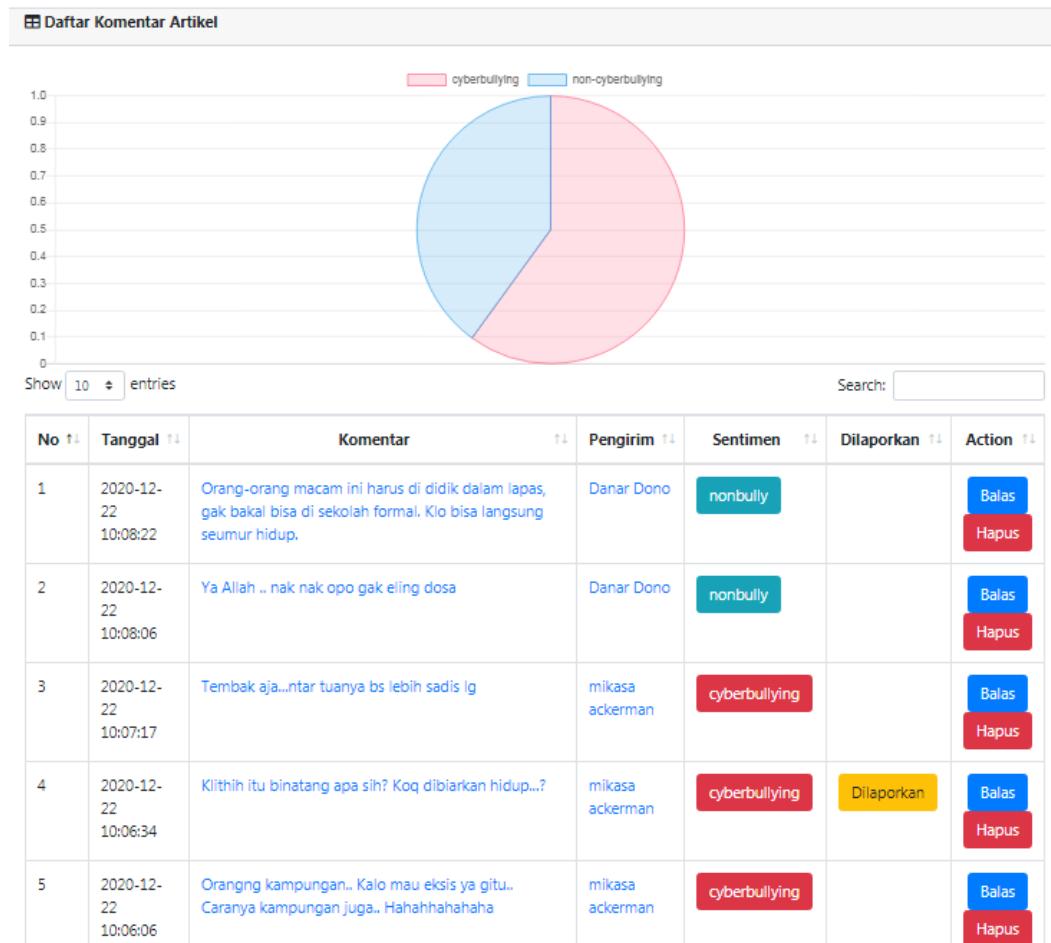
Balas

Laporkan

Non-Cyberbullying

Gambar 4.6 **Tampilan Daftar Komentar di Sisi Front End**

Pada sisi *back end*, penulis artikel dapat mengelola daftar komentar beserta kategorinya. Dapat dilihat pada gambar 4.7, terdapat komentar-komentar yang telah diklasifikasikan kategorinya menjadi *cyberbullying* dan *non-cyberbullying*. Pada daftar komenatar juga terdapat komentar yang telah dilaporkan member kepada penulis. Penulis bisa melakukan penanganan terhadap komentar yang mengandung *cyberbullying* dengan menghapus komentar tersebut. Penulis juga bisa memberikan balasan kepada member yang memberi komentar. Selain itu, daftar komentar artikel juga memuat grafik *pie* perbandingan total komentar yang mengandung *cyberbullying* dan total komentar yang *non-cyberbullying*. Berikut ditunjukkan ilustrasi daftar komentar pada sisi *back end* di situs:



Gambar 4.7
Tampilan Daftar Komentar di Sisi *Backend End*

BAB V

ANALISIS SISTEM

Pada Bab V dijelaskan tentang bagaimana menganalisis sistem. Pada Bab V dijelaskan analisis terhadap kebutuhan sistem. Adapun penjelasannya meliputi menganalisis sistem manajemen konten pada situs *blog*, mengidentifikasi permasalahan pada sistem lama, mengembangkan sistem baru, perbandingan fitur antara sistem lama dan sistem baru hingga mengembangkan *data flow diagram* pada sistem baru. Penjelasan selengkapnya sebagai berikut:

5.1 Analisis Sistem Manajemen Konten Blog

Pada sub bab ini dijelaskan tentang analisis sistem *website CMS* dari sejumlah *website CMS blog* seperti kompasiana.com dan indonesiana.com. Pada sub bab ini dijelaskan analisis terhadap sistem lama hingga analisis sistem baru. Adapun analisis sistem melibatkan deskripsi sistem lama yang berjalan pada *platform website* tersebut, analisis permasalahan pada sistem lama, deskripsi sistem baru yang akan dikembangkan dan perbandingan fitur antara sistem lama dan sistem baru. Berikut penjelasan selengkapnya:

5.1.1 Deskripsi Sistem Lama

Website yang dijadikan pembanding atas tugas akhir ini ialah kompasiana.com dan indonesiana.com. Kedua *platform website* tersebut sama-sama merupakan *website CMS blog*. *Platform website* tersebut menyediakan tempat bagi penggunanya untuk menulis *blog* dan membagikan artikel *blog* baik secara internal di dalam *website* maupun eksternal atau membagikan link artikel ke media sosial lainnya. Berikut dijelaskan deskripsi dari Kompasiana dan Indonesiana:

Kompasiana adalah sebuah platform blog dan publikasi *online* yang dikembangkan oleh Kompas Cyber Media sejak 22 Oktober 2008. Setiap konten (artikel, foto, komentar) dibuat dan ditayangkan langsung oleh pengguna internet

yang telah memiliki akun Kompasiana (disebut Kompasianer). Di tahun pertama kehadirannya, Kompasiana dibangun sebagai blog jejaring internal untuk jurnalis dan karyawan Kompas Gramedia. Memasuki tahun 2009, produk yang didirikan oleh Pepih Nugraha ini berubah menjadi platform blog untuk semua orang. Nama Kompasiana sendiri diambil dari nama kolom yang diisi oleh Pendiri Harian Kompas, PK Ojong. Kompasiana mengusung tag line “*Beyond Blogging*”. Hal ini mempertegas posisi Kompasiana sebagai saluran berita dan opini masyarakat

Kompasiana menyediakan halaman manajemen konten blog untuk penulisnya supaya bisa mengelola artikelnya lebih lanjut seperti menulis artikel, mengedit artikel dan menghapus artikel hingga kata kunci. Di dashboard kompasiana sendiri terdapat statistik artikel seperti total artikel, total dilihat, total komentar, total rating, total artikel masuk headline, dan total artikel masuk rubrik pilihan. Selain itu, kompasiana juga menyediakan fitur following-follower yang memungkinkan antar pengguna mengikuti akun mereka satu sama lain. Kompasiana juga menyediakan fitur notifikasi untuk memberitahukan informasi kepada penggunanya. Kompasiana juga menyediakan fitur komentar terhadap artikel untuk memudahkan interaksi diskusi di antara para pengguna seputar konten artikel. Kompasiana juga menyediakan fitur rating untuk penilaian artikel sehingga pengguna lainnya bisa memberikan penilaian terhadap kualitas artikel. Kompasiana juga menyediakan fitur bagikan artikel ke media sosial lainnya seperti facebook, twitter dan linked. Walaupun demikian, Kompasiana juga memungkinkan antar pengguna untuk melaporkan komentar lebih lanjut yang melanggar ketentuan yang berlaku. Kompasiana menyediakan pengaturan bagi pengguna untuk mengatur halaman profilnya seperti mengubah foto profil, nama lengkap, alamat dan validasi profil pengguna. Selain itu, kompasiana menyediakan fitur percakapan yang memungkinkan antar pengguna untuk saling bertukar pesan. Terakhir, Kompasiana mengelompokkan artikel lebih lanjut berdasarkan kategori, terpopuler, terbaru, pilihan editor, dan topik pilihan. Kompasiana juga menyediakan berbagai fitur menarik lainnya seperti berbagi video, event hingga afiliasi. Keunikan Kompasiana terletak pada sisi pengelolaan konten yang dilakukan secara simultan. Meskipun setiap artikel yang dibuat Kompasianer langsung tayang, Kompasiana memoderasi

isinya setiap saat untuk memastikan tidak ada yang melanggar syarat dan ketentuan. Dan untuk setiap konten terbaik, Kompasiana mengolahnya dan mempromosikannya 7 x 24 jam agar dibaca oleh lebih banyak orang.

Di sisi lain, Indonesiana adalah sebuah blog publik milik TEMPO yang menampung konten unggahan pengguna. Pengguna Indonesiana memiliki kebebasan untuk menyampaikan gagasan, pendapat, ulasan, ataupun menampilkan karya dalam bentuk tulisan, foto, audio, atau video sepanjang memenuhi ketentuan hukum dan norma yang berlaku di Indonesia, *terms of use* Tempo.co, dan ketentuan konten Indonesiana dan tempo.co. Indonesiana menyediakan halaman admin bagi penggunanya. Halaman admin berfungsi untuk melakukan pengelolaan konten blog seperti menulis artikel, mengedit artikel dan menghapus artikel hingga kata kunci. Indonesiana juga menyediakan fitur media yang memungkinkan pengguna untuk menyimpan foto atau gambar original yang bisa dipakai untuk gambar pada konten. Indonesiana juga memberikan fitur pengaturan yang memungkinkan pengguna menyunting profilnya. Indonesiana mengelompokkan artikel lebih lanjut berdasarkan kategorinya seperti gaya hidup, olahraga, opini, peristiwa, travel, seleb, pilihan editor dan berita utama. Menariknya Indonesiana melakukan pengulasan pada artikel pengguna sebelum ditayangkan pada halaman homepage website-nya. Dengan adanya pengulasan ini meminimalisir penyalahgunaan artikel oleh pengguna melanggar ketentuan yang berlaku. Moderator berhak menindaklanjuti artikel-artikel yang melanggar ketentuan yang berlaku sesuai dengan Terms of Use Tempo.co, Ketentuan Konten Indonesiana,tempo.c, dan norma dan hukum yang berlaku di Indonesia. Penindaklanjutan ini seperti menghapus konten, menghapus tautan (link) yang mengarahkan pembaca ke web selain tempo.co, mengirimkan imbauan kepada pengguna mengenai ketentuan yang berlaku untuk pelanggaran kedua, mengirimkan peringatan untuk pelanggaran kedua, dan memblokir akun pelanggar untuk pelanggaran ketiga.

Jika dilihat dari manajemen kontennya, kompasiana dan indonesia memiliki perbedaan peninjauan/ memoderasi artikel sebelum ditayangkan. Pada kompasiana artikel yang ditulis pengguna bisa ditayangkan secara simultan dan bisa saja mengalami memoderasi ketika ditemukan pelanggaran. Sementara itu

indonesiana mengharuskan artikel yang ditulis pengguna melalui proses pengulasan dari moderatornya sebelum ditayangkan. Ditinjau dari interaksi antar pengguna dalam platform websitenya, kompasiana lebih memberikan kebebasan kepada penggunanya untuk berkomentar terhadap artikel yang ditulis pengguna lainnya. Kompasiana juga memberikan kebebasan bagi pengguna untuk memberikan rating terhadap artikel pengguna lainnya. Bahkan antar pengguna bisa bertukar pesan secara privat melalui fitur percakapan. Sementara itu, Indonesiana tidak memiliki fitur interaksi antar pengguna seperti komentar, rating dan pesan privat.

5.1.2 Analisis Permasalahan Sistem Lama

Berdasarkan deskripsi sistem di atas dapat dipahami permasalahan lebih lanjut terkait sistem manajemen konten *blog* dari *platform* situs kompasiana dan indonesiana. Setelah dipaparkan deskripsi sistem dari *platform* situs kompasiana dan indonesiana, maka dianalisis kekurangan dari masing-masing situs tersebut. Berikut ini dijelaskan kekurangan dari *platform* situs kompasiana dan indonesiana sebagai berikut:

Kekurangan yang pertama ialah baik pada platform website Kompasiana dan Indonesiana sama-sama kurang mendukung fitur laporan dashboard. Di Indonesiana statistik artikel sama sekali tidak ada. Di sisi lain, statistik artikel yang ditampilkan pada Kompasiana terlalu ringkas hanya memberikan informasi umum saja seperti total artikel, total dilihat, total komentar. Tidak ada pengolahan informasi lebih lanjut seperti perkembangan penayangan berdasarkan rentang tanggal tertentu, rating artikel terbaik, perkembangan rating artikel berdasarkan rentang tanggal tertentu, dan statistik artikel yang dikomentari.

Kekurangan yang kedua adalah baik pada platform website Kompasiana maupun Indonesiana tidak memiliki fitur untuk menyaring komentar lebih lanjut untuk mencegah penyalahgunaan komentar khususnya berkaitan dengan tindakan *cyberbullying*. Padahal fitur penyaring komentar itu penting demi mendukung interaksi antar pengguna secara sehat. Kompasiana sendiri hanya menyediakan fitur melaporkan komentar. Di sisi lain, Indonesiana tidak memiliki fitur komentar

sehingga tidak memberikan pengalaman pengguna untuk berinteraksi secara lebih interaktif dalam *website*.

5.1.3 Deskripsi Sistem Baru

Berdasarkan analisis permasalahan di atas selanjutkan dikembangkan deskripsi sistem baru untuk mengatasi permasalahan pada sistem lama. Hal ini bertujuan untuk mengembangkan spesifikasi sistem baru demi mengatasi segala kekurangan dari sistem lama. Berikut dijelaskan secara rinci deskripsi sistem baru yang akan dikembangkan. Adapun penjelasan selengkapnya sebagai berikut:

Pertama, sistem baru memiliki kemampuan mengelola konten meliputi menulis, mengedit, dan menghapus artikel, tag hingga kategorisasi artikel. Dalam hal ini, pengelolaan konten akan dilakukan pada sisi backend dimana hanya pengguna yang terdaftar sebagai member yang bisa mengaksesnya. Sementara penayangan konten akan berjalan pada sisi *front end* dimana konten bisa dibaca oleh semua user baik pengunjung website, member dan admin. Fitur tag berfungsi untuk memudahkan penindeksan konten berdasarkan kata kunci. Terakhir, kategorisasi berfungsi untuk mengelompokkan artikel berdasarkan topik tertentu. Kategorisasi dikelola sepenuhnya oleh admin.

Kedua, untuk mendukung interaksi antara pengunjung website dan penulis konten, maka sistem akan dilengkapi dengan beberapa fitur interaktif meliputi rating artikel, penghitung tayangan, berbagi ke sosial media, pemberian komentar, hingga pelaporan komentar. Rating berfungsi untuk memberikan penilaian pembaca dalam hal ini member terhadap konten artikel. Penghitung tayangan berfungsi untuk menghitung jumlah konten telah dikunjungi dan dibaca oleh pengunjung website. Fitur berbagai ke sosial media berfungsi untuk membagikan artikel ke sosial media yakni Twitter dan Facebook. Fitur pemberian komentar berfungsi sebagai sarana diskusi dan mengemukakan pendapat dari member yang membaca artikel terhadap konten. Terakhir fitur pelaporan berfungsi untuk memberikan pelaporan oleh member yang membaca komentar terhadap berbagai komentar yang mengandung muatan *cyberbullying* demi mendukung ekosistem yang lebih kondusif.

Ketiga, sistem baru hendaknya memiliki fitur laporan *dashboard* yang cukup lengkap seperti perkembangan penayangan berdasarkan rentang tanggal tertentu, rating artikel terbaik, perkembangan rating artikel berdasarkan rentang tanggal tertentu, dan statistik artikel yang dikomentari. Hal ini bertujuan menyajikan informasi terkait artikel secara informatif. Di *platform* Kompasiana dan Indonesiana tidak menyediakan cukup laporan yang memadai. Dengan adanya laporan di halaman *dashboard* secara lengkap, diharapkan pengguna bisa mendapatkan informasi lebih lanjut terkait statistik dan perkembangan artikelnya.

Keempat, untuk mengatasi kekurangan sistem lama berupa tiadanya tindakan antisipasi untuk menyeleksi komentar, maka sistem baru hendaknya memiliki fitur untuk menyaring komentar pengguna. Sistem juga harus mampu memberikan pesan peringatan kepada pengguna yang melakukan penyalahgunaan komentar terkait tindakan *cyberbullying*. Fitur penyaring komentar *cyberbullying* ini akan berkerja otomatis sehingga bisa mengurangi beban administrator dalam memantau komentar pada situsnya. Dengan demikian, diharapkan interaksi antar pengguna dalam sistem baru menjadi lebih kondusif.

Kelima, sistem baru mendukung manajemen user meliputi registrasi, login, manajemen akses user, hingga manajemen profil. Registrasi dapat digunakan pengunjung untuk mendaftarkan dirinya sebagai member baru. Login digunakan untuk member dan admin bisa mengakses halaman backend. Sementara manajemen akses user bisa digunakan admin untuk memberikan hak akses kepada member menjadi admin baru. Selain itu, manajemen profil berfungsi untuk mengelola profil pengguna seperti pengantian password, nama, foto hingga biodata seputar profilnya.

Keenam, sistem baru mendukung manajemen website meliputi kontak, kebijakan, FAQ, pengaturan layout, evaluasi *website*, hingga *pretty URL*. Fitur kontak berisi informasi terkait deskripsi umum *website*. Sementara fitur kebijakan berisi deskripsi umum terkait kebijakan, syarat dan ketentuan website. Fitur FAQ berisi pertanyaan dan jawaban yang umumnya ditanyakan terkait *website*. Terakhir fitur pengaturan *website* berguna untuk mengubah tampilan *template website* sesuai dengan pilihan yang disediakan. Fitur ini dikelola sepenuhnya oleh admin.

Sementara member dan pengunjung hanya bisa melihat informasi-informasi tersebut pada halaman *front end website*. Fitur evaluasi *website* berfungsi untuk memberikan penilaian *website* oleh member terhadap kinerja *website*. Terakhir fitur *pretty URL* membuat tampilan URL *website* menjadi lebih bersih dan rapi.

5.1.4 Perbandingan Fitur Sistem Lama dan Sistem Baru

Pada sub bab ini dijelaskan tentang perbandingan fitur sistem lama dan sistem baru. Fitur sistem dari platform situs Kompasiana maupun Indonesiana akan dibandingkan dengan fitur dari sistem baru yang dikembangkan dalam tugas akhir ini. Hal ini bertujuan melihat perbedaan di antara fitur-fitur yang disediakan di antara tugas akhir, kompiana dan Indonesiana. Berikut penjelasan selengkapnya:

Tabel 5.1
Perbandingan Fitur Antara Tugas Akhir (TA), Kompasiana dan Indonesiana

No	Fitur	Tugas Akhir	Kompasiana.com	Indonesiana.id
1	Manajemen artikel	V	V	V
2	Kata kunci	V	V	V
3	Manajemen komentar	V	V	X
4	Rating	V	V	X
5	Penyaring komenatr <i>cyberbullying</i>	V	X	X
6	Notifikasi	V	V	X
7	Laporan statistik artikel	V	V	X
8	<i>Following</i>	V	V	X
9	Kategorisasi	V	V	V
10	Konfigurasi pengguna	V	V	V
11	<i>Share to sosmed</i>	V	V	X
12	Registrasi dan login	V	V	V
13	Event	X	V	X
14	Berbagi video	X	V	X
15	Berbagi foto	V	V	V
16	Afiliasi	X	V	X
17	Pelaporan komentar <i>cyberbullying</i>	V	V	X
18	About us	V	V	V

Tabel 5.1
(Lanjutan)

No	Fitur	Tugas Akhir	Kompasiana.com	Indonesiana.id
19	Kebijakan	V	V	V
20	FAQ	V	V	X
21	Pengaturan <i>template</i>	V	X	X
22	Evaluasi website	V	X	X
23	<i>Pretty URL</i>	V	V	V

Bisa dilihat pada tabel 5.1 bahwa sistem baru pada tugas akhir ini memiliki kelebihan dibandingkan sistem lama pada platform website Kompasiana dan Indonesiana yakni adanya fitur penyaring komentar *cyberbullying*. Selain itu, sistem baru juga memiliki fitur laporan statistik artikel seperti Kompasiana bahkan lebih lengkap seperti adanya laporan seperti perkembangan penayangan berdasarkan rentang tanggal tertentu, rating artikel terbaik, perkembangan rating artikel berdasarkan rentang tanggal tertentu, dan statistik artikel yang dikomentari. Sehingga pengguna bisa mendapatkan informasi lebih lanjut terkait statistik dan perkembangan artikelnya. Sistem baru juga memiliki fitur *following-follower*, pelaporan komentar, manajemen artikel, notifikasi, kategorisasi, berbagi artikel ke media sosial lainnya, konfigurasi pengguna, rating artikel, registrasi, login, *about us*, kebijakan, FAQ, pengaturan layout, penerapan *pretty URL*.

Walaupun demikian, sistem baru juga masih memiliki beberapa kekurangan dibandingkan sistem lama. Sistem baru tidak memiliki fitur event sebagaimana yang terdapat pada Kompasiana. Sistem baru juga tidak mendukung berbagi video sebagaimana yang terdapat pada Kompasiana. Sistem baru juga tidak mendukung fitur afiliasi sebagaimana yang terdapat pada Kompasiana.

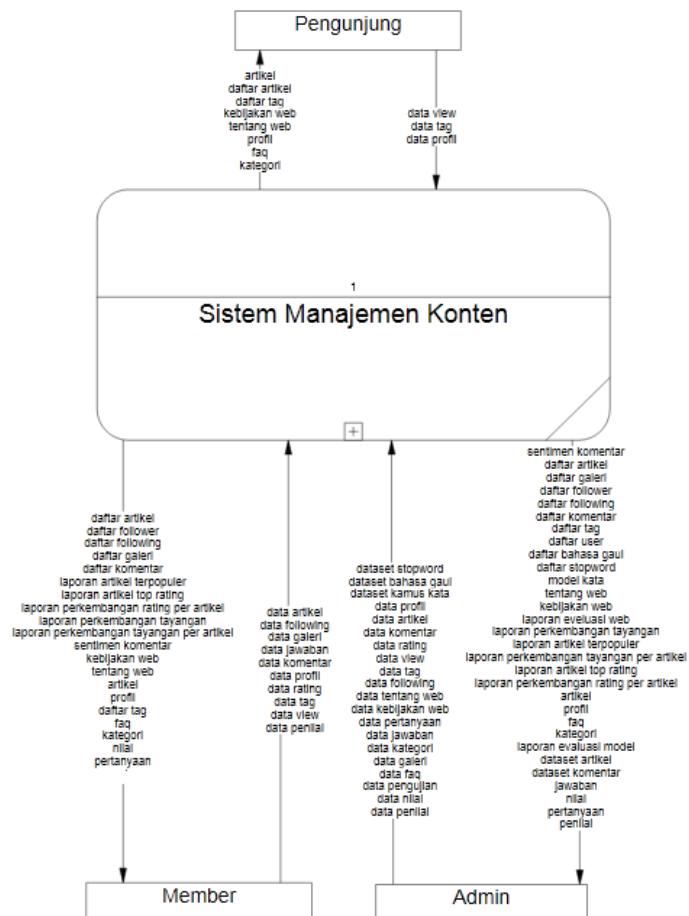
5.2 Data Flow Diagram

DFD adalah suatu diagram yang menggambarkan aliran data dari sebuah proses atau sistem (biasanya sistem informasi). DFD juga menyediakan informasi mengenai luaran dan masukan dari setiap entitas dan proses itu sendiri. Tujuan dari DFD adalah menyediakan jembatan antara pengguna dan pengembang system.

Pada sub bab ini dijelaskan bagaimana data flow diagram pada sistem baru yang akan dikembangkan pada tugas akhir ini. Berikut penjelasan selengkapnya:

5.2.1 DFD Level Konteks

Pada bagian ini dijelaskan tentang bagaimana DFD level konteks yang berjalan pada sistem baru. DFD level konteks mengambarkan hubungan/ interaksi antara sistem dan pengguna secara makro. Pada sistem baru, pengguna terdiri atas 3 macam meliputi pengunjung, admin dan member. Pengunjung adalah orang-orang yang mengakses sistem untuk sekedar melihat konten blog. Sementara admin adalah orang yang mengelola dan memelihara sistem. Sedangkan member adalah orang yang telah terdaftar di sistem blog sebagai penulis.



Gambar 5.1
DFD Level Konteks

Berikut dijelaskan interaksi dalam DFD level konteks. Dapat dilihat pada gambar 5.1, DFD level konteks terdapat pengunjung yang memberikan masukan berupa data view, data tag, dan data profil. Lalu sistem mengembalikan masukan dari pengunjung menjadi keluaran berupa artikel, daftar artikel, daftar tag, kebijakan web, tentang web, profil, faq, dan kategori. Selain itu, terdapat member dimana ia memberikan masukan berupa data artikel, data following, data galeri, data jawaban, data komentar, data profil, data rating, data tag, data view, dan data penilai. Lalu sistem mengembalikan masukan dari member menjadi keluaran berupa daftar artikel, daftar follower, daftar following, daftar galeri, laporan artikel terpopuler, laporan artikel top rating, laporan perkembangan rating per artikel, laporan perkembangan tayangan, laporan perkembangan tayangan per artikel, sentimen komentar, kebijakan web, tentang web, artikel, profil, daftar tag, faq, kategori, nilai, dan pertanyaan.

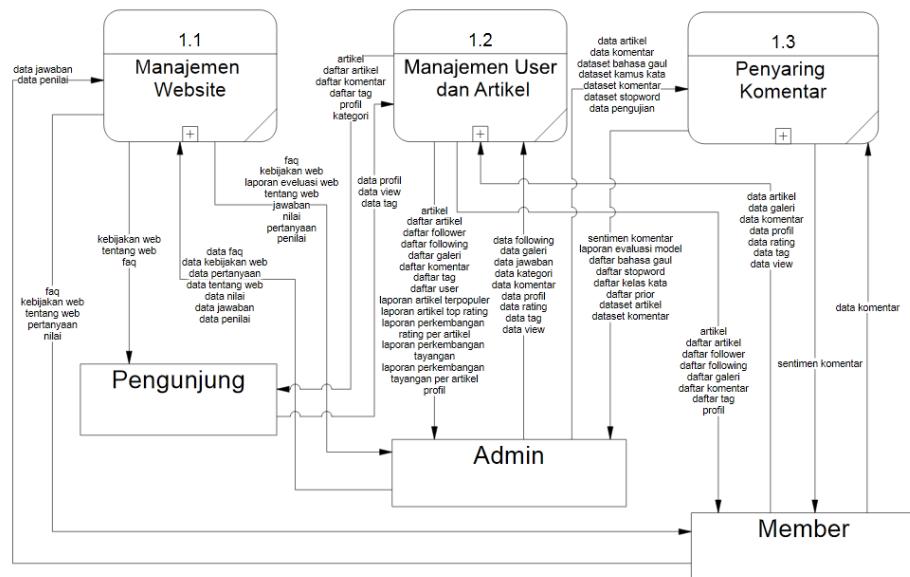
Terakhir, admin memberikan masukan terhadap sistem berupa dataset stopword, dataset bahasa gaul, dataset kamus kata, data profil, data artikel, data komentar, data rating, data view, data tag, data following, data tentang web, data kebijakan web, data pertanyaan, data jawaban, data kategori, data galeri, data faq, data pengujian, data nilai dan data penilai. Lalu sistem mengembalikan masukan dari admin menjadi keluaran berupa kategori komentar, daftar artikel, daftar galeri, daftar follower, daftar following, daftar komentar, daftar tag, daftar user, daftar bahasa gaul, daftar *stopword*, model kata, tentang web, kebijakan web, laporan evaluasi web, laporan perkembangan tayangan, laporan artikel terpopuler, laporan perkembangan tayangan per artikel, laporan artikel top rating, laporan perkembangan rating per artikel, artikel, profil, faq, kategori, laporan evaluasi model, dataset artikel, dataset komentar, jawaban, nilai, pertanyaan, dan penilai. Adanya berbagai laporan tersebut merupakan hasil keluaran dari inputan yang dikirim pengguna. Demikian, bagaimana tiga user berinteraksi dengan CMS.

5.2.2 DFD Level 0

DFD level 0 mengambarkan proses lebih rinci yang terjadi pada sistem di DFD level konteks. DFD level 0 terdiri atas beberapa sub sistem yang saling

berinteraksi dengan pengguna. DFD level 0 pada sistem baru digambarkan pada gambar 5.2. DFD tersebut terdiri atas 3 sub sistem meliputi manajemen website, manajemen user dan artikel, serta penyaring komentar, dimana ketiga sub sistem tersebut berinteraksi dengan pengguna meliputi pengunjung, admin, dan member.

Misalnya, admin mengirimkan inputan berupa data faq, data kebijakan web, data pertanyaan, data tentang web, data nilai, data jawaban, hingga data penilai kepada sub sistem manajemen website. Lalu sub sistem manajemen website mengembalikan keluaran berupa faq, kebijakan web, laporan evaluasi web, tentang web, jawaban nilai, pertanyaan, dan penilai kepada admin. Adapula pengunjung yang berinteraksi terhadap sub sistem manajemen dan artikel dimana pengunjung mengirimkan masukan berupa data profil, data view, data tag kepada sub sistem manajemen user dan artikel. Lalu sub sistem manajemen user dan artikel mengembalikan keluaran berupa artikel, daftar artikel, daftar komentar, daftar tag, profil, dan kategori kepada pengunjung.



Gambar 5.2
DFD Level 0

Selain itu member yang berinteraksi dengan sub sistem penyaring komentar dimana member memberikan masukan berupa data komentar kepada sub sistem penyaring komentar. Kemudian sub sistem penyaring mengembalikan keluaran

berupa kategori komentar. Member juga berinteraksi dengan sub sistem manajemen user dan artikel. Member mengirimkan masukan berupa data artikel, data galeri, data komentar, data profil, data rating, data tag, dan data view. Kemudian sub sistem manajemen user dan artikel mengembalikan keluaran berupa artikel, daftar artikel, daftar follower, daftar following, daftar galeri, daftar komentar, daftar tag dan profil.

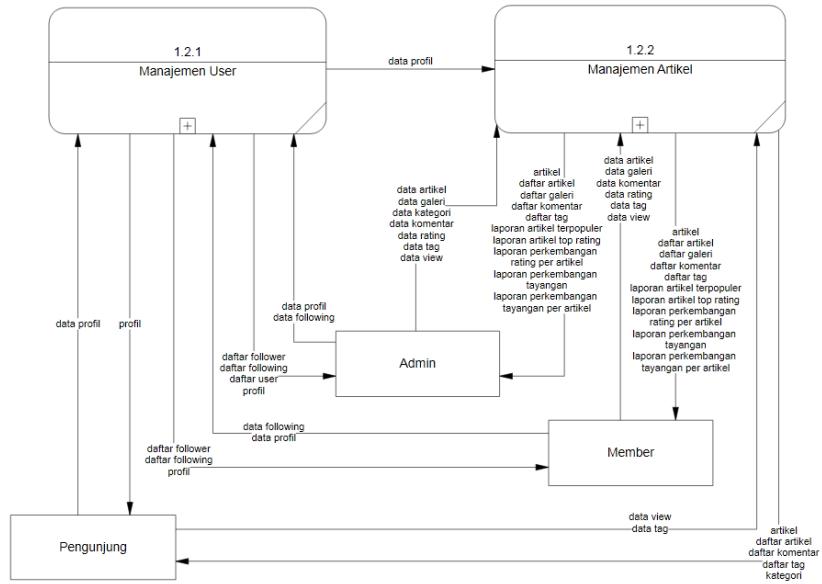
5.2.3 DFD Level 1

DFD level 1 mengembangkan penjelasan rinci dari tiap proses yang terjadi pada DFD level 0. Pada sistem baru, DFD level 1 terdiri atas 3 DFD yang masing-masing mengambarkan proses dari sub sistem meliputi manajemen website, penyaring komentar, manajemen user dan artikel. Ketiga sub proses tersebut berinteraksi dengan 3 pengguna meliputi admin, member dan pengunjung. Berikut ini dijelaskan satu per satu tiap prosesnya:

- DFD Level 1 Manajemen User dan Artikel

Pada gambar 5.3 dijelaskan tentang DFD level 1 manajemen user dan artikel. DFD level 1 terdiri atas 2 sub proses meliputi manajemen user dan manajemen artikel. Manajemen user merupakan sub proses yang menangani pengelolaan user. Sementara manajemen artikel merupakan sub proses yang menangani pengelolaan artikel. Kedua sub proses tersebut berinteraksi dengan tiga pengguna meliputi pengunjung, member dan admin.

Misalnya, pengunjung berinteraksi dengan sub proses manajemen user dimana pengunjung mengirimkan masukan berupa data profil. Kemudian sub sistem manajemen user mengembalikannya menjadi keluaran berupa profil. Pengunjung juga berinteraksi dengan sub sistem manajemen artikel dimana pengunjung mengirimkan masukan berupa data view dan data tag. Kemudian dikembalikan sub sistem manajemen artikel menjadi artikel, daftar artikel, daftar komentar, daftar tag, dan kategori.



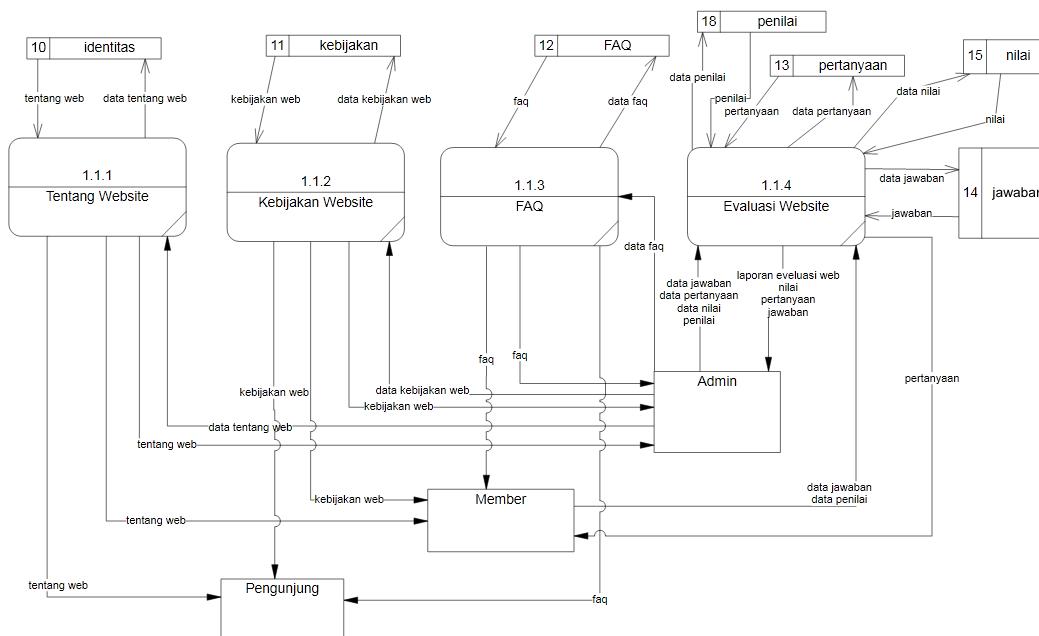
Gambar 5.3 DFD Level 1 Manajemen User dan Artikel

- DFD Level 1 Manajemen *Website*

Pada gambar 5.4 dijelaskan tentang DFD level 1 manajemen web. DFD terdiri atas 4 sub proses meliputi tentang *website*, kebijakan *website*, FAQ dan evaluasi *website*. Sub proses tentang *website* merupakan sub proses yang menangani pengelolaan terkait identitas *website*. Sub proses kebijakan *website* merupakan sub proses yang menangani pengelolaan terkait kebijakan *website*. Sub proses FAQ merupakan sub proses yang menangani pengelolaan terkait segala pertanyaan dan jawaaban terkait *website*. Sub proses evaluasi *website* merupakan sub proses yang menangani pengelolaan terkait mengukur kualitas *website*. Keempat sub proses tersebut berinteraksi dengan 3 pengguna meliputi pengunjung, member, dan admin.

Misalnya, admin mengirimkan masukan berupa data tentang web kepada sub sistem tentang *website*. Lalu sub sistem tersebut menyimpan datanya dalam basis penyimpanan data identitas. Lalu penyimpanan data identitas mengembalikan keluaran berupa tentang web lalu diproses sub sistem manajemen *website* untuk diteruskan kepada admin menjadi keluaran tentang *website*. Selain itu, admin juga berinteraksi dengan sub sistem kebijakan *website* dimana admin mengirimkan masukan berupa data kebijakan web

kemudian data tersebut diproses sub sistem kebijakan *website* untuk disimpan ke dalam basis penyimpanan data kebijakan. Sub sistem memproses data kebijakan web dengan meminta basis data penyimpanan data mengembalikan keluaran kebijakan web, kemudian sub sistem kebijakan *website* meneruskan kepada admin.



Gambar 5.4
DFD Level 1 Manajemen Website

Adapula admin berinteraksi dengan evaluasi *website* dimana admin mengirimkan berbagai masukan berupa data penilai, data nilai, data pertanyaan, data jawaban kepada sub sistem evaluasi *website*. Kemudian sub sistem evaluasi *website* memproses dan menyimpan berbagai masukan tersebut ke dalam sesuai dengan basis penyimpanan datanya seperti data penilai disimpan dalam basis data penilai, data pertanyaan disimpan dalam basis data pertanyaan, data nilai disimpan dalam basis data nilai, dan data jawaban disimpan dalam basis data jawaban. Sub sistem evaluasi *website* memproses data tersebut dengan meminta berbagai basis data meliputi penilai, pertanyaan, nilai, dan jawaban untuk mengirimkan masukan masing-masing berupa penilai, pertanyaan, nilai, dan jawaban. Kemudian sub sistem evaluasi *website*

memproses data tersebut menjadi informasi menjadi laporan evaluasi web, penilai, pertanyaan dan jawaban kepada admin.

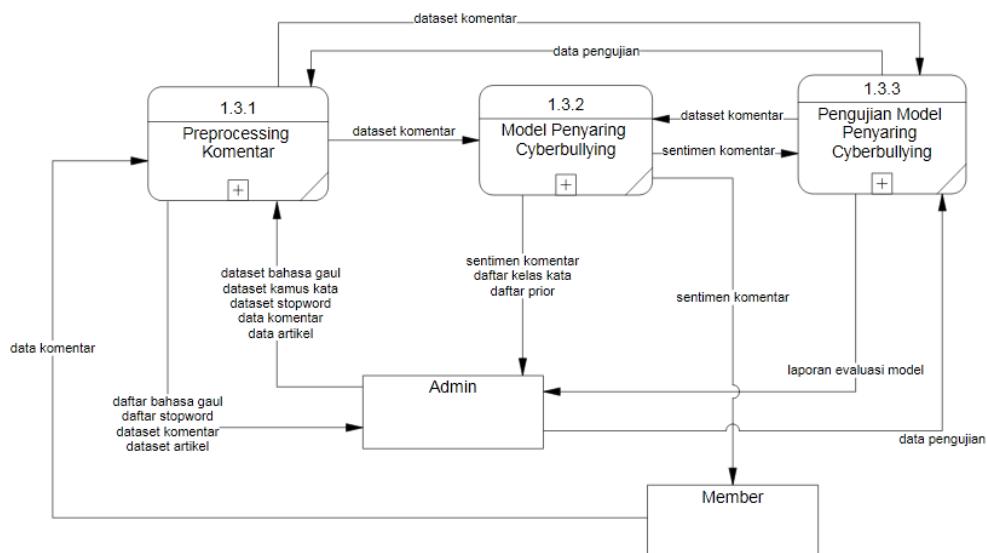
- DFD Level 1 Penyaring Komentar

Pada gambar 5.5 dijelaskan tentang DFD Level 1 Penyaring Komentar.

DFD tersebut terdiri atas 3 sub sistem meliputi *preprocessing*, model penyaring *cyberbullying*, dan pengujian model penyaring *cyberbullying*. Sub proses *preprocessing* merupakan sub proses yang menangani pengelolaan terkait pengolahan dataset sebelum dilatih. Sub proses model penyaring *cyberbullying* merupakan sub proses yang menangani pengelolaan terkait pembuatan model klasifikasi *cyberbullying*. Sub proses pengujian model penyaring *cyberbullying* merupakan sub proses yang menangani pengelolaan terkait uji coba model penyaring *cyberbullying*. Ketiga sub sistem tersebut berinteraksi dengan 2 pengguna meliputi admin dan member.

Misalnya, admin berinteraksi dengan sub sistem *preprocessing* dimana admin mengirimkan masukan berupa dataset bahasa gaul, dataset kamus kata, dataset *stopword*, data komentar, data komentar, data artikel. Kemudian sub sistem *preprocessing* memproses masukan tersebut lalu dikembalikan menjadi keluaran berupa daftar bahasa gaul, daftar *stopword*, dataset komentar, dan dataset artikel kepada admin. Sub sistem *preprocessing* komentar juga mengirimkan masukan dataset komentar kepada sub sistem model penyaring komentar. Lalu sub sistem model penyaring komentar mengembalikannya menjadi keluaran berupa sentimen komentar, daftar kelas kata, daftar *prior* kepada admin. Admin juga mengirimkan data pengujian kepada sub sistem pengujian model penyaring *cyberbullying*. Kemudian sub sistem pengujian model penyaring *cyberbullying* memproses data tersebut untuk diteruskan kepada sub sistem *preprocessing* komentar. Sehingga dikembalikan menjadi dataset komentar oleh sub sistem *preprocessing* komentar kepada sub sistem pengujian model penyaring *cyberbullying*. Lalu sub sistem pengujian model penyaring *cyberbullying* memproses dataset komentar untuk diteruskan kepada sub sistem model penyaring *cyberbullying* sehingga dikembalikannya menjadi kategori komentar kepada sub sistem pengujian model penyaring

cyberbullying. Terakhir sub sistem pengujian model penyaring *cyberbullying* memproses masukan kategori komentar tersebut, lalu dikembalikan menjadi informasi laporan evaluasi model kepada admin. Secara keseluruhan proses ini menggambarkan bagaimana pemrosesan data komentar menjadi kategori komentar dan laporan evaluasi model dalam sistem penyaring komentar.



Gambar 5.5
DFD Level 1 Penyaring Komentar

5.2.4 DFD Level 2

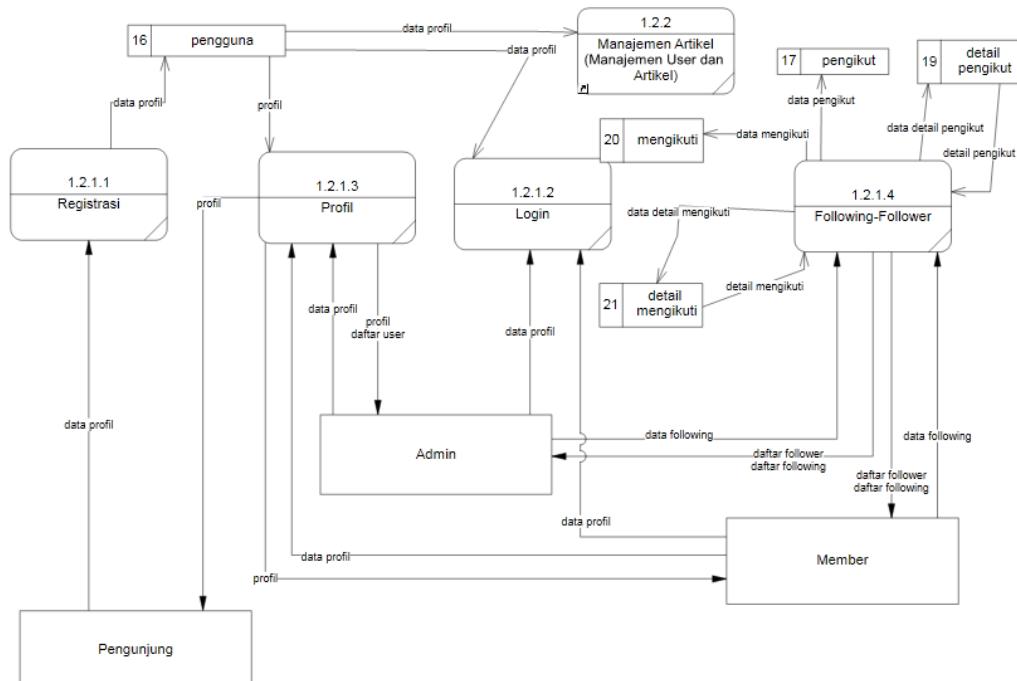
DFD level 2 menggambarkan proses lebih rinci pada tiap proses di DFD level

- Dalam hal ini, DFD level 2 akan menjelaskan interaksi dalam sub proses dari DFD level 1 secara lebih rinci. Pada DFD level 2 dalam sistem ini terdiri atas 5 DFD level 1 meliputi DFD level 2 manajemen user, DFD level 2 manajemen artikel, DFD level 2 *preprocessing* komentar, DFD level 2 model penyaring *cyberbullying*, dan DFD level 2 pengujian model penyaring *cyberbullying*. Berikut penjelasan tiap DFD selengkapnya:

- **DFD Level 2 Manajemen User**

Pada gambar 5.6 dijelaskan DFD level 2 manajemen user. DFD tersebut terdiri atas 4 sub proses meliputi registrasi, profil, *login*, dan *following-follower*. Sub proses registrasi merupakan sub proses yang menangani

pengelolaan terkait pendaftaran akun baru di *website*. Sub proses profil merupakan sub proses yang menangani pengelolaan profil pengguna di *website*. Sub proses *login* merupakan sub proses yang menangani pengelolaan pengaksesan pengguna dalam masuk-keluar ke halaman situs. Sub proses *following-follower* merupakan sub proses yang menangani pengelolaan akun mengikuti dan yang diikuti antar pengguna di *website*. Keempat sub proses tersebut berinteraksi dengan 3 pengguna meliputi pengunjung, member, dan admin.



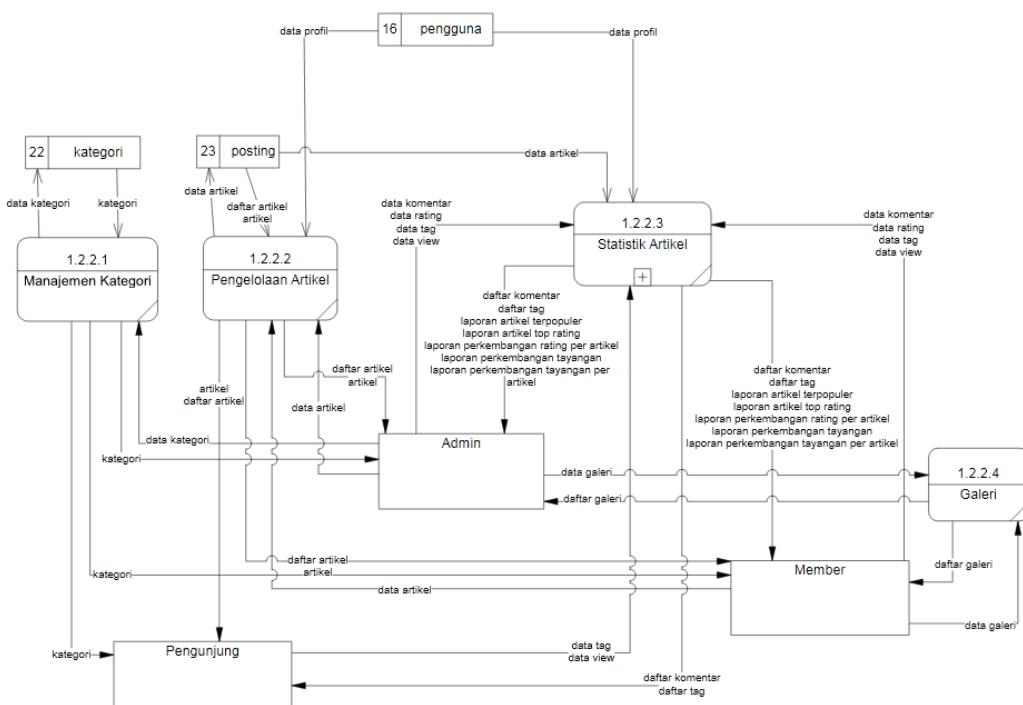
Gambar 5.6
DFD Level 2 Manajemen User

Misalnya, member berinteraksi terhadap sub sistem *following-follower*. Member memberikan masukan kepada sub sistem *following-follower* berupa data following yakni akun profil yang diikuti member. Kemudian sub sistem *following-follower* memproses masukan tersebut mengorganisasinya ke dalam basis penyimpanan data masing-masing data pengikut disimpan dalam basis penyimpanan pengikut, data detail pengikut disimpan dalam basis

penyimpanan data detail pengikut, data mengikuti disimpan dalam basis penyimpanan data mengikuti, dan data detail mengikuti disimpan dalam basis penyimpanan data detail mengikuti. Kemudian sub sistem *following-follower* mengembalikannya menjadi keluaran berupa daftar follower dan daftar following kepada member.

- DFD Level 2 Manajemen Artikel

Gambar 5.7 mengambarkan DFD level 2 manajemen artikel. DFD tersebut terdiri atas 3 sub sistem meliputi manajemen kategori, pengelolaan artikel, dan statistik artikel. Sub proses manajemen kategori merupakan sub proses yang menangani pengelolaan terkait kategori-kategori artikel di *website*. Sub proses tentang *website* merupakan sub proses yang menangani pengelolaan terkait berbagai artikel di *website*. Sub proses artikel merupakan sub proses yang menyajikan statistik terkait artikel yang dipublikasi di *website*. Ketiga sub sistem tersebut berinteraksi dengan 3 pengguna meliputi pengunjung, admin, dan member.

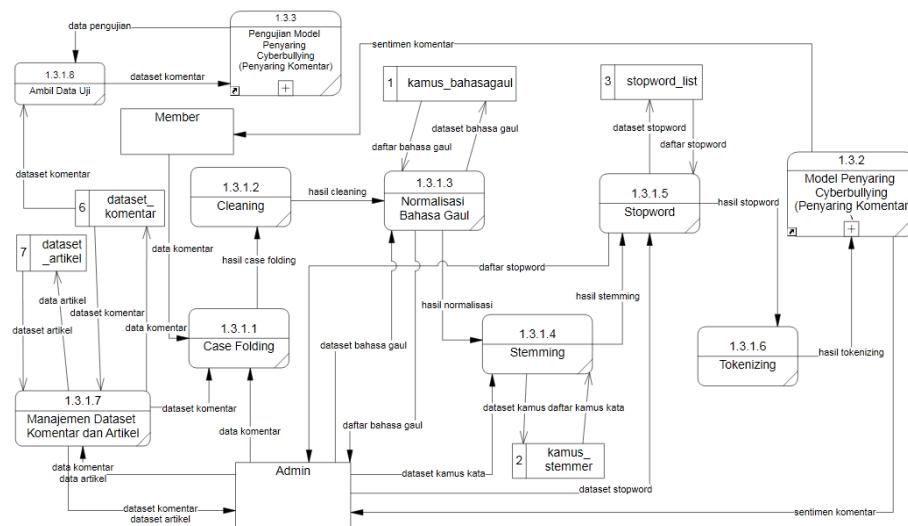


Gambar 5.7
DFD Level 2 Manajemen Artikel

Misalnya, member berinteraksi dengan sub sistem pengelolaan artikel. Member mengirimkan masukan berupa data artikel kepada sub sistem pengelolaan artikel. Kemudian sub sistem pengelolaan artikel menyimpan data tersebut dalam basis penyimpanan data posting. Sub sistem pengelolaan artikel memproses data tersebut dengan meminta basis data posting. Kemudian dikembalikan menjadi daftar artikel dan artikel untuk selanjutkan diteruskan sub sistem pengelolaan artikel menuju member.

- DFD Level 2 *Preprocessing* Komentar

Gambar 5.8 mengambarkan DFD level 2 *preprocessing* komentar. DFD tersebut terdiri 6 sub proses meliputi *case folding*, *cleaning*, normalisasi bahasa gaul, *stemming*, *stopword*, dan *tokenizing*. Keenam sub proses secara berturut-turut merupakan penjabaran dari tahapan-tahapan sub proses *preprocessing* komentar. Sub proses tersebut berinteraksi satu sama lain dan berinteraksi dengan 2 pengguna meliputi admin dan member.



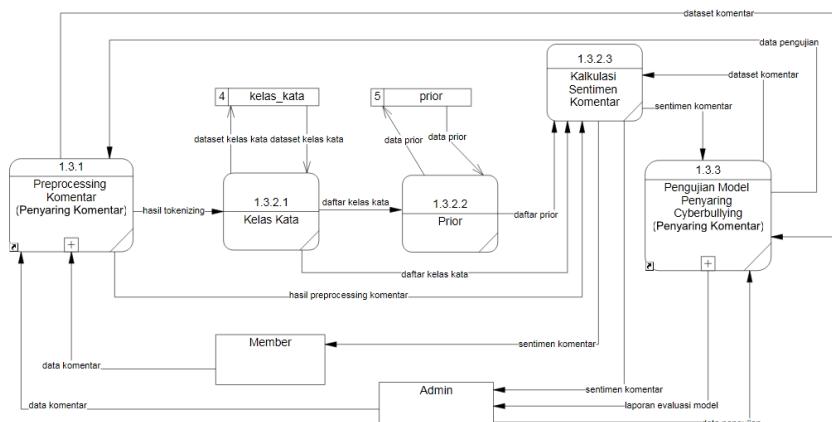
Gambar 5.8
DFD Level 2 *Preprocessing* Komentar

Misalnya, admin mengirimkan masukan data komentar kepada sub sistem *case folding*. Kemudian sub sistem *case folding* memproses data komentar menjadi hasil *case folding* untuk diteruskan menuju sub sistem *cleaning*. Sub sistem *cleaning* memproses data hasil *case folding* menjadi hasil

cleaning untuk diteruskan menuju sub sistem normalisasi bahasa gaul. Sub sistem normalisasi bahasa gaul memproses data hasil *cleaning* menjadi hasil normalisasi untuk diterukan menuju sub sistem *stemming*. Sub sistem *stemming* memproses data hasil normalisasi menjadi hasil *stemming* untuk diteruskan menuju sub sistem *stopword*. Sub sistem *stopword* memproses data hasil *stemming* menjadi hasil *stopword* untuk diteruskan menuju sub sistem *tokenizing*. Lalu sub sistem *tokenizing* memproses hasil *stopword* menjadi hasil *tokenizing* untuk diterukan menuju sub sistem model penyaring *cyberbullying*. Sub sistem model penyaring *cyberbullying* mengembalikan menjadi keluaran berupa kategori komentar kepada admin.

- DFD Level 2 Model Penyaring Cyberbullying

Gambar 5.9 mengambarkan tentang DFD level 2 model penyaring cyberbullying. DFD tersebut terdiri atas 3 sub proses meliputi kelas kata, prior, dan kalkulasi sentimen komentar. Tiga sub proses tersebut merupakan penjabaran dari sub model penyaring *cyberbullying*. Tiga sub proses tersebut berinteraksi dengan 2 pengguna meliputi member dan admin.



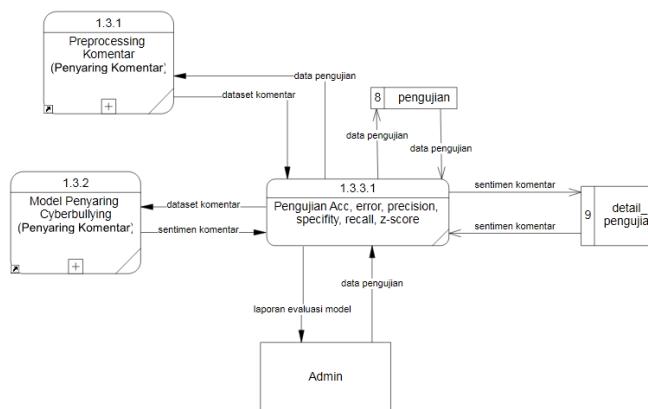
Gambar 5.9
DFD Level 2 Model Penyaring Cyberbullying

Misalnya, admin mengirimkan data komentar menuju sub proses *preprocessing*. Kemudian sub proses *preprocessing* memprosesnya menjadi hasil *tokenizing* untuk diteruskan menuju sub proses kelas kata. Sub proses kelas kata memproses hasil *tokenizing* menjadi dataset kelas kata kemudian

disimpan dalam basis penyimpanan data kelas kata. Kemudian sub kelas kata mengambil dataset kelas kata untuk diproses lebih lanjut sehingga menghasilkan daftar kelas kata untuk diterukan menuju sub proses *prior*. Lalu sub proses *prior* memproses daftar kelas kata menjadi data *prior* untuk disimpan dalam basis penyimpanan data *prior*. Lalu sub proses *prior* mengambil data *prior* untuk diproses menjadi daftar prior lalu diteruskan menuju sub proses kalkulasi sentimen komentar. Sub proses kalkulasi kategori komentar kemudian memproses daftar prior menjadi sentimen komentar untuk selanjutkan diteruskan menuju admin.

- DFD Level 2 Pengujian Model Penyaring Cyberbullying

Gambar 5.10 mengambarkan tentang DFD level 2 pengujian model cyberbullying. DFD tersebut memiliki sub proses pengujian *acc*, *error*, *precision*, *specificity*, *recall*, dan *z-score*. Sub proses tersebut menangani proses perhitungan akurasi, *error*, *precision*, *specificity*, *recall*, dan *z-score*. Sub proses tersebut berinteraksi dengan admin.



Gambar 5.10
DFD Level 2 Pengujian Model Cyberbullying

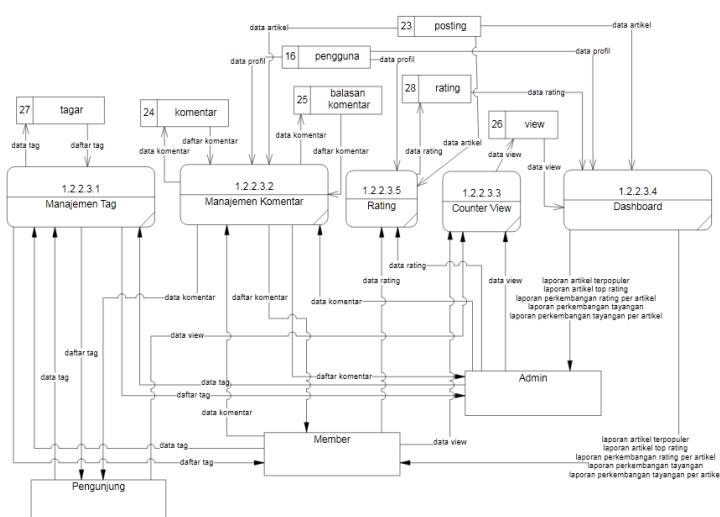
Admin mengirimkan data pengujian kepada sub proses tersebut. Kemudian sub proses tersebut memprosesnya lebih lanjut untuk diteruskan menuju sub proses preprocessing. Lalu sub proses *preprocessing* lalu mengembalikannya menjadi *dataset* komentar. Kemudian sub proses meneruskannya menuju sub proses model penyaring *cyberbullying*. Lalu model

penyaring *cyberbullying* mengembalikannya menjadi kategori komentar. Akhirnya sub proses memproses lebih lanjut menjadi laporan evaluasi model menuju admin.

5.2.5 DFD Level 3

DFD level 3 menjelaskan lebih rinci terkait interaksi yang terjadi di dalam DFD level 2. Mengingat DFD level 2 manajemen artikel memiliki sub proses statistik artikel, maka sub proses statistik artikel dijelaskan lebih lanjut pada DFD level 3. DFD tersebut menggambarkan interaksi antara sub proses yang terjadi dalam statistik artikel meliputi manajemen tag, manajemen komentar, *rating*, *counter view* dan *dashboard* dengan 3 pengguna meliputi pengunjung, member dan admin. DFD level 3 ditunjukkan pada ilustrasi gambar 5.11.

Misalnya, member memberikan data komentar kepada sub proses manajemen komentar. Sub proses manajemen komentar memproses data komentar lebih lanjut untuk disimpan dalam basis penyimpanan data komentar untuk data komentar atau disimpan dalam basis penyimpanan data balasan komentar. Sub proses manajemen komentar mengambil daftar komentar baik dari basis penyimpanan data komentar dan basis penyimpanan data balasan komentar. Selanjutnya hasilnya ditampilkan menjadi daftar komentar kepada member.



Gambar 5.11
DFD Level 3 Statistik Artikel

Selain itu, member bisa berinteraksi dengan sub sistem rating. Member memberikan *rating* kepada sub proses *rating*. Kemudian sub proses *rating* memproses data *rating* disimpan pada basis penyimpanan data *rating*. Member juga berinteraksi dengan sub proses manajemen tag. Member mengirimkan data tag menuju sub proses manajemen tag. Lalu sub proses manajemen tag menyimpan data tag ke basis penyimpanan data tagar. Kemudian, sub proses manajemen tag menampilkan daftar tag kepada member.

BAB VI

DESAIN SISTEM

Pada Bab VI dijelaskan tentang bagaimana mendesain sistem baru. Adapun penjelasannya meliputi desain arsitektur sistem, desain database, desain antar muka, dan desain prosedural. Desain arsitektur sistem menjelaskan gambaran fitur-fitur dari sistem yang dikembangkan. Desain database menggambarkan rancangan tabel relasi beserta hubungan antar tabelnya. Desain antar muka menjelaskan rancangan antar muka baik di sisi *back end* maupun *front end*. Desain prosedural menggambarkan rancangan algoritma yang akan diimplementasikan dalam pengembangan fitur sistem. Penjelasan selengkapnya sebagai berikut:

6.1 Desain Arsitektur Sistem

Pada sub bab ini dijelaskan desain arsitektur sistem manajemen konten blog. Desain arsitektur menggambarkan keseluruhan fitur sistem yang akan dikembangkan. Berdasarkan data flow dari sistem baru, dirancang desain arsitektur yang terdiri atas hak akses meliputi pengunjung, member, dan admin. Fitur-fitur sistemnya terbagi dalam fitur *front end* dan fitur *back end*. Selengkapnya ditunjukkan pada tabel 6.1 sebagai berikut:

Tabel 6.1
Arsitektur Sistem Manajemen Konten Blog

Fitur	Pembaca	Member	Admin
Front End			
1. Registrasi	V	X	X
2. <i>Login</i>	X	V	V
3. Artikel	V	V	V
4. Komentar artikel	X	V	V
5. Rating artikel	X	V	V
6. Statistik artikel	X	V	V
7. Kategori artikel	V	V	V
8. <i>Following</i>	X	V	V
9. Peringatan <i>cyberbullying</i>	X	V	V

Tabel 6.1
(Lanjutan)

Fitur	Pembaca	Member	Admin
10. Pelaporan Komentar <i>cyberbullying</i>	X	V	V
11. Tag Terpopuler	V	V	V
12. Artikel Terpopuler	V	V	V
13. Berbagi ke sosial media	V	V	V
14. Tentang Web	V	V	V
15. Kebijakan Web	V	V	V
16. Evaluasi Web	X	V	V
17. Profil	V	V	V
Back End			
1. Konfigurasi profil	X	V	V
2. Manajemen user	X	X	V
3. Notifikasi	X	V	V
4. Mengelola artikel	X	V	V
5. Mengelola <i>follower-following</i>	X	V	V
6. Mengelola rating	X	V	V
7. Mengelola komentar	X	V	V
8. Penyaringan komentar <i>cyberbullying</i>	X	V	V
9. Mengelola kategori	X	X	V
10. Manajemen galeri	X	V	V
11. Manajemen tag	X	X	V
12. Manajemen tentang web	X	X	V
13. Manajemen kebijakan web	X	X	V
14. Manajemen evaluasi web	X	X	V
15. Dashboard	X	V	V
16. Pengaturan <i>layout template</i>	X	X	V

Dapat dilihat pada tabel 1, arsitektur *frontend* terdiri atas pengguna yakni member dan pengunjung. Pengunjung dapat melakukan registrasi, membaca artikel, mencari dan mengelompokkan artikel berdasarkan kategori tertentu. Sedangkan member dapat membuat artikel, mengikuti akun pengguna lain (*following*), membaca artikel, melihat laporan statistik tiap artikel yang dibacanya, memberi rating pada artikel yang dibacanya, memberi komenter terhadap artikel yang dibacanya, mencari dan mengurutkan artikel berdasarkan kategorinya. Namun sebelum melakukan itu semua, member perlu login ke sistem terlebih dahulu. Berikut ini penjelasan fitur-fiturnya *front end* selengkapnya:

- Registrasi

Fitur registrasi digunakan untuk mendaftarkan pengunjung menjadi member di website. Pengunjung dapat mendaftar melalui akun Facebook maupun langsung ke halaman registrasi *website*. Fitur ini hanya bisa dipakai oleh pengunjung yakni pengguna yang belum terdaftar sebagai member atau admin di *website*. Pada fitur ini, pengguna harus mengisi biodatanya untuk bisa mendaftar ke sistem.

- *Login*

Fitur *login* digunakan untuk pengguna bisa mengakses halaman admin dengan jalan memasukan *username* dan *password*. Fitur ini hanya bisa diakses oleh member dan admin. Jika pengguna salah memasukan *username* dan *password* maka sistem akan mengeluarkan pesan *error*. Selain itu, jika pengguna lupa tentang *password*-nya, maka pengguna bisa memanfaatkan fitur lupa *password* untuk mereset *password*. Caranya, pengguna tinggal memasukan e-mailnya pada form yang disediakan, lalu sistem akan mengirimkan e-mail mereset password dimana pengguna bisa mengklik tautan pada e-mail tersebut untuk diarahkan pada halaman peresetan *password*.

- Artikel

Fitur artikel digunakan untuk menampilkan atau mempublikasikan konten posting pada halaman *front end website*. Sehingga artikel bisa dibaca oleh pengguna *website*. Artikel yang ditulis member setelah disimpan akan segera ditampilkan pada tampilan artikel *front end*. Fitur ini bisa diakses pengunjung, member, dan admin.

- Komentar artikel

Fitur komentar artikel digunakan untuk memberikan komentar terhadap konten posting. Dari fitur komentar artikel, pengguna bisa memberikan komentar. Selain itu, pengguna juga bisa menghapus komentarnya di kolom komentar pada artikel. Fitur ini hanya bisa dipakai member dan admin.

- Tag terpopuler

Fitur tag terpopuler menampilkan 5 tag terpopuler yang sering diakses pengguna. Sehingga pengguna bisa melihat dan mengetahui tag yang sering

diakses. Dari fitur ini, pengguna akan diarahkan menuju artikel terkait yang menggunakan kata kunci tag tersebut. Fitur ini bisa dipakai pengunjung, member, dan admin.

- Artikel terpopuler

Fitur artikel terpopuler menampilkan 5 artikel terpopuler yang sering dikunjungi pengunjung. Sehingga pengguna bisa mengetahui dan melihat artikel yang sering diakses. Pengunjung bisa memanfaatkan fitur ini untuk melihat artikel terpopuler dari segi total tayangan. Fitur ini bisa dipakai pengunjung, member, dan admin

- Berbagi ke sosial media

Fitur berbagai ke media sosial digunakan untuk membagikan tautan artikel ke media sosial lainnya seperti twitter dan facebook. Dengan fitur ini, pengguna dimudahkan untuk mendistribusikan artikel ke media sosial. Dengan demikian, pengguna bisa membagikan artikel secara praktis. Fitur ini bisa dipakai pengunjung, member, dan admin.

- Tentang web

Fitur tentang web menampilkan informasi terkait website seperti kontak dan deskripsi web. Dengan fitur ini, pengguna bisa melihat rincian deskripsi terkait website. sehingga pengguna bisa melihat informasi terkait identitas web. Fitur ini bisa diakses pengunjung, member, dan admin.

- Kebijakan web

Fitur kebijakan web menampilkan berbagai ketentuan terkait penggunaan website. Dengan fitur ini, pengguna bisa melihat ketentuan dan larangan dalam menggunakan situs. Sehingga pengguna bisa mengetahui segala ketentuan menyangkut penggunaan *website*. Fitur ini bisa diakses pengunjung, member dan admin.

- Rating artikel

Fitur rating artikel digunakan untuk memberikan penilaian terhadap artikel dari skala likert 1 – 5. Fitur ini memudahkan pembaca memberikan penilaian terhadap artikel dan memudahkan penulis menilai kualitas artikelnya. Hasil dari rating artikel selanjutnya akan dihitung rata-rata per artikelnya

sehingga bisa diketahui kualitas kontennya. Fitur ini hanya bisa dipakai member dan admin.

- Statistik artikel

Fitur statistik artikel digunakan untuk menampilkan statistik artikel seperti jumlah tayang, rata-rata rating, dan jumlah komentar. Fitur ini memudahkan pengunjung mengetahui statistik umum terkait artikel. Dengan kata lain, statistik artikel disajikan melalui fitur ini. Fitur ini bisa dilihat pengunjung, member, dan admin pada tiap konten artikelnya.

- Kategori artikel

Fitur kategori artikel digunakan untuk menampilkan dan mencari artikel berdasarkan kategori topik konten artikel. Fitur ini memudahkan pengunjung mengklasifikasikan artikel berdasarkan kategori topiknya. Sehingga pengguna bisa mencari artikel berdasarkan kategori topik artikelnya. Fitur ini bisa dipakai pengunjung, member dan admin.

- *Following*

Fitur *following* digunakan untuk mengikuti profil/ akun member lainnya. Dari fitur ini, pengguna bisa mengikuti aktivitas member dalam hal publikasi konten artikelnya yang akan diberitakan melalui notifikasi. Pengguna cukup menekan tombol ikuti pada profil penulis untuk menggunakan fitur ini. Fitur ini bisa dipakai member dan admin.

- Peringatan *cyberbullying*

Fitur peringatan *cyberbullying* digunakan untuk memberikan peringatan kepada pengguna terkait komentar yang dikirimnya itu berpotensi mengandung muatan *cyberbullying*. Sehingga pengguna diharapkan bisa meninjau kembali komentarnya sebelum dikirim. Fitur ini hanya aktif tatkala suatu komentar teridentifikasi sebagai *cyberbullying*. Fitur ini dipakai member dan admin.

- Pelaporan komentar *cyberbullying*

Terkadang terjadi kelolosan komentar sekalipun komentarnya menandung muatan *cyberbullying*. Fitur pelaporan komentar *cyberbullying* digunakan untuk melaporkan secara manual terhadap berbagai komentar yang

melanggar ketentuan *website* seperti tindakan *cyberbullying*, provokasi, pornografi, ujaran kebencian, spam, hingga tulisan tidak baik. Laporan tersebut selanjutnya akan ditindak lanjuti oleh admin maupun member pemilik konten terkait. Fitur ini bisa dipakai pengunjung, member, dan admin.

- Evaluasi web

Fitur evaluasi web digunakan bagi pengguna untuk memberikan penilaian terhadap kinerja *website*. Fitur ini memudahkan pengguna menilai kualitas website secara menyeluruh. Untuk memakai fitur ini, pengguna diminta mengisi biodata dan menjawab pertanyaan-pertanyaan terkait *website*. Fitur ini bisa diakses member dan admin.

- Profil

Fitur profil menampilkan biodata terkait penulis konten artikel. Fitur ini bisa diakses pengunjung, member, dan admin. Fitur ini memudahkan pengunjung melihat biodata terkait penulis artikel serta mengakses daftar kontribusi artikel yang relavan dengan penulis. Dari fitur profil, user bisa melihat informasi berkaitan dengan penulis seperti total pengikuti, total mengikuti, total artikel, beserta rincian daftar kontribusi artikel.

Sementara itu, dapat dilihat pada tabel 6.1, arsitektur *back end* terdiri atas dua pengguna meliputi admin dan member. Member dapat melakukan konfigurasi profil, mengelola artikel, mengelola komentar terhadap artikel, menyaring komentar yang berpotensi mengandung *cyberbullying*, mengelola rating terhadap artikel, dan mengelola pengikut dan notifikasi. Sedangkan admin dapat melakukan konfigurasi pengguna, mengelola registrasi pengguna, mengelola kategori, mengelola artikel, mengelola komentar terhadap artikel, menyaring komentar yang berpotensi mengandung *cyberbullying*, mengelola rating terhadap artikel, dan mengelola pengikut serta notifikasi. Baik member maupun admin perlu melakukan login terlebih dahulu sebelum dapat mengakses fitur-fitur *backend*. Berikut penjelasan fitur-fitur *back end* selengkapnya:

- Konfigurasi profil

Fitur konfigurasi profil digunakan untuk mengelola biodata member atau admin seperti menganti password, nama, foto dan biodata lainnya. Fitur

ini memudahkan pengguna untuk mengelola akunnya. Untuk menggunakan fitur ini, pengguna bisa mengubah biodatanya lalu disimpan ke halaman profil. Fitur ini bisa diakses admin dan member.

- **Manajemen user**

Fitur manajemen user digunakan untuk mengelola member. Melalui fitur ini, admin bisa memberikan hak akses terhadap member lainnya untuk dijadikan sebagai admin baru atau mengubah status admin suatu akun menjadi member biasa. Selain itu, manajemen user juga berguna untuk mengaktifasi dan menonaktifkan akun user. Fitur ini hanya bisa diakses oleh admin saja.

- **Notifikasi**

Fitur notifikasi digunakan untuk memberikan pemberitahuan kepada pengguna terkait pemberitahuan tertentu seperti berita artikel baru, rating baru, komentar baru, hingga adanya laporan *cyberbullying*. Pengguna bisa melihat segala pemberitahuan terkait artikel dan akunnya. Pengguna bisa menandai notifikasi sudah dibaca untuk menghilangkan notifikasi yang aktif. Fitur ini hanya bisa diakses member maupun admin.

- **Mengelola artikel**

Fitur mengelola artikel digunakan oleh penulis untuk mengelola artikel. Pengelolaan artikel meliputi menulis konten, mengedit konten, dan menghapus konten. Dari fitur ini, pengguna juga bisa melihat daftar artikel yang dimilikinya. Fitur ini hanya bisa diakses oleh admin maupun member.

- **Mengelola *following-follower***

Fitur mengelola *following-follower* digunakan untuk menampilkan daftar akun yang pengguna ikuti maupun daftar akun yang pengguna diikuti. Fitur ini mudahkan pengelolaan pengikut dan yang diikuti pengguna. Pengguna bisa memanfaatkan fitur ini untuk melihat akun yang diikutinya dan akun yang mengikutinya. Fitur ini bisa diakses oleh admin maupun member.

- **Mengelola rating**

Fitur mengelola rating digunakan untuk menampilkan daftar pengguna lain yang memberikan rating terhadap konten artikel. Fitur ini mudahkan penulis dalam mengelola rating artikelnya secara lebih rinci. Pengguna bisa

melihat rincian rating dari tiap artikelnya melalui fitur ini. Fitur ini bisa diakses oleh admin maupun member.

- Mengelola komentar

Fitur mengelola komentar digunakan untuk mengelola komentar pada tiap artikel. Dari fitur ini member maupun admin bisa melihat daftar komentar pada tiap artikel. Pengguna juga bisa menonaktifkan komentar dengan memanfaatkan fitur ini. Pengguna juga bisa membalas komentar menggunakan fitur ini. Fitur ini bisa diakses oleh admin maupun member.

- Penyaring komentar *cyberbullying*

Fitur penyaring komenter *cyberbullying* digunakan untuk memeriksa dan menyaring komentar yang berpotensi mengandung *cyberbullying*. Pengguna bisa melihat kategori komentar baik *cyberbullying* maupun *non-cyberbullying* dengan memanfaatkan fitur ini. Pengguna juga bisa menindak lanjuti komentar yang mengandung *cyberbullying* dengan cara menonaktifkan komentar tersebut supaya tidak ditampilkan di publikasi artikel. Fitur ini bisa diakses oleh admin maupun member.

- Mengelola kategori

Fitur mengelola kategori digunakan untuk mengelola kategori artikel. Pengelolaan kategori meliputi membuat kategori, mengedit kategori, dan menghapus kategori. Pengguna juga bisa mengaktifkan kategori artikel yang aktif di sisi *front end website* dengan memanfaatkan fitur ini. Fitur ini memudahkan pengelolaan terkait kategori topik artikel pada tampilan *front end*. Fitur ini hanya bisa dipakai admin.

- Manajemen galeri

Fitur manajemen galeri digunakan untuk mengelola galeri. Pengelolaan galeri meliputi mengupload galeri gambar, menghapus daftar gambar, hingga menyalin URL gambar. URL gambar yang telah disalin selanjutnya dapat dipakai untuk ditampilkan pada konten artikel. Fitur ini bisa dipakai member dan admin.

- Manajemen tag

Fitur manajemen tag digunakan untuk mengelola tag pada konten artikel. Pengelolaan tag meliputi menampilkan daftar tag, membuat tag, mengedit tag, hingga menghapus tag. Fitur ini memudahkan pengelolaan tag artikel secara menyeluruh. Fitur ini hanya bisa dipakai admin.

- Manajemen tentang web

Fitur manajemen tentang web digunakan untuk mengelola data tentang *website*. Pengelolaan *website* meliputi menulis kontak dan deskripsi web, mengeditnya, hingga menghapusnya. Fitur ini memudahkan pengelolaan terkait identitas situs. Fitur ini hanya bisa dipakai admin.

- Manajemen kebijakan web

Fitur manajemen kebijakan web digunakan untuk mengelola kebijakan web. Pengelolaan kebijakan web meliputi menulis kebijakan web, mengeditnya, dan menghapusnya. Fitur ini memudahkan pengelolaan terkait kebijakan situs. Fitur ini hanya bisa dipakai admin.

- Manajemen evaluasi web

Fitur evaluasi web digunakan untuk mengelola evaluasi web seperti membuat kuesioner, memberikan nilai, menampilkan daftar jawaban, hingga mengelolanya menjadi laporan evaluasi web. Selain itu, fitur ini juga menyajikan penilaian terhadap kualitas *website* dalam bentuk grafik. Fitur ini memudahkan pengelolaan evaluasi terkait kualitas website. Fitur ini bisa diakses admin.

- *Dashboard*

Fitur *dashboard* digunakan untuk menampilkan informasi terkait konten artikel seperti laporan perkembangan tayangan, laporan artikel populer, laporan perkembangan tayangan per artikel, laporan artikel top rating, dan laporan perkembangan rating per artikel. Fitur ini memudahkan pengguna untuk menyajikan informasi terkait artikelnya. Fitur ini menyediakan lima jenis laporan meliputi grafik tayangan seluruh artikel, grafik tayangan per artikel, grafik perbandingan rating lima artikel terpopuler, grafik perkembangan rating

per artikel, dan grafik perbandingan tayangan lima artikel terpopuler. Fitur ini bisa diakses member dan admin.

- Pengaturan *template*

Fitur pengaturan template digunakan untuk mengubah tampilan *template front end website*. Fitur pengaturan *template* menyediakan dua *template website* yang bisa dipilih. Selain itu, admin juga bisa mengelola tag meta website seperti nama web, alamat web, meta deskripsi, dan meta keyword. Fitur ini hanya bisa diakses admin.

6.2 Desain Database

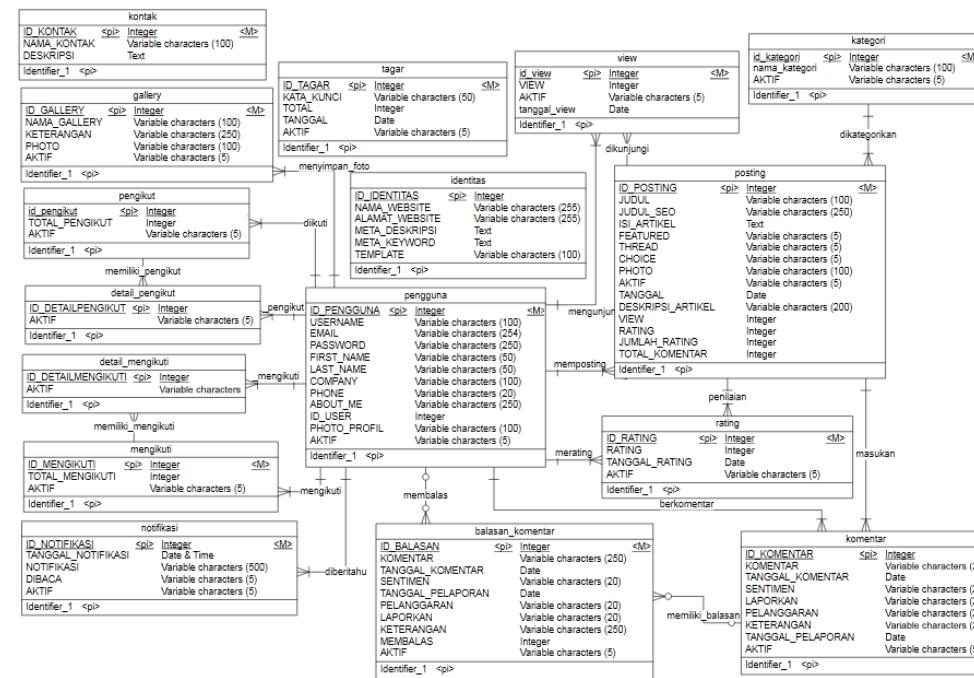
Pada sub bab ini dijelaskan bagaimana mendesain database pada sistem manajemen konten blog dan penyaring komentar *cyberbullying*. Pembahasannya mencakup *conceptual data model* (CDM) untuk menjelaskan desain database secara *logic*. *Physical data model* (PDM) untuk menjelaskan desain database secara fisik pada MySQL 5.0. Dan penjelasan mengenai desain tabel relasinya. Berikut penjelasan selengkapnya:

6.2.1 Conceptual Data Model

CDM dipakai untuk menggambarkan secara detail struktur basis data dalam bentuk logik. Struktur ini independen terhadap semua software maupun struktur data storage tertentu yang digunakan dalam aplikasi ini. CDM terdiri dari objek yang tidak diimplementasikan secara langsung ke dalam basis data yang sesungguhnya. Pada gambar 6.1 dijelaskan tentang bagaimana gambaran CDM dari database sistem manajemen konten yang dikembangkan. Berikut ini dijelaskan tentang relasi antar entitas dalam CDM. Berikut dijelaskan relasi antar tabelnya.

Pengguna mempunyai hubungan *one to many* dengan galeri. Hal ini berarti setiap pengguna bisa memiliki banyak galeri. Tapi setiap galeri hanya bisa memiliki satu pengguna. Pengguna memiliki hubungan *one to many* dengan pengikut. Hal ini berarti setiap pengguna bisa memiliki banyak pengikut. Tapi setiap pengikut hanya dimiliki satu pengguna. Pengguna memiliki hubungan *one to many* dengan

detail_pengikut. Hal ini berarti setiap pengguna bisa memiliki banyak detail_pengikut. Tapi setiap detail_pengikut hanya memiliki satu pengguna.



Gambar 6.1
CDM I Tentang Sistem Manajemen Konten

Pengikut memiliki hubungan *one to many* dengan detail_pengikut. Hal ini berarti setiap pengikut bisa memiliki banyak detail_pengikut. Tapi setiap detail_pengikut hanya dimiliki satu pengikut. Pengguna memiliki hubungan *one to many* dengan mengikuti. Hal ini berarti setiap pengguna bisa memiliki banyak mengikuti. Tapi setiap mengikuti hanya dimiliki satu pengguna. Pengguna memiliki hubungan *one to many* dengan detail_mengikuti. Hal ini berarti setiap pengguna bisa memiliki banyak detail_mengikuti. Tapi setiap detail_mengikuti hanya memiliki satu pengguna.

Mengikuti memiliki hubungan *one to many* dengan detail_mengikuti. Hal ini berarti setiap mengikuti bisa memiliki banyak detail_mengikuti. Tapi setiap detail_mengikuti hanya dimiliki satu mengikuti. Pengguna memiliki hubungan *one to many* dengan notifikasi. Hal ini berarti setiap pengguna bisa memiliki banyak notifikasi. Tapi setiap notifikasi hanya bisa dimiliki satu pengguna. Pengguna

memiliki hubungan *one to many* dengan view. Hal ini berarti setiap pengguna bisa memiliki banyak view. Tapi setiap view hanya bisa memiliki satu pengguna.

Pengguna memiliki hubungan *one to many* dengan rating. Hal ini berarti setiap pengguna bisa memiliki banyak rating. Tapi setiap rating hanya memiliki satu pengguna. Pengguna memiliki hubungan *one to many* dengan posting. Hal ini berarti setiap pengguna bisa memiliki banyak posting. Tapi setiap posting hanya memiliki satu pengguna.

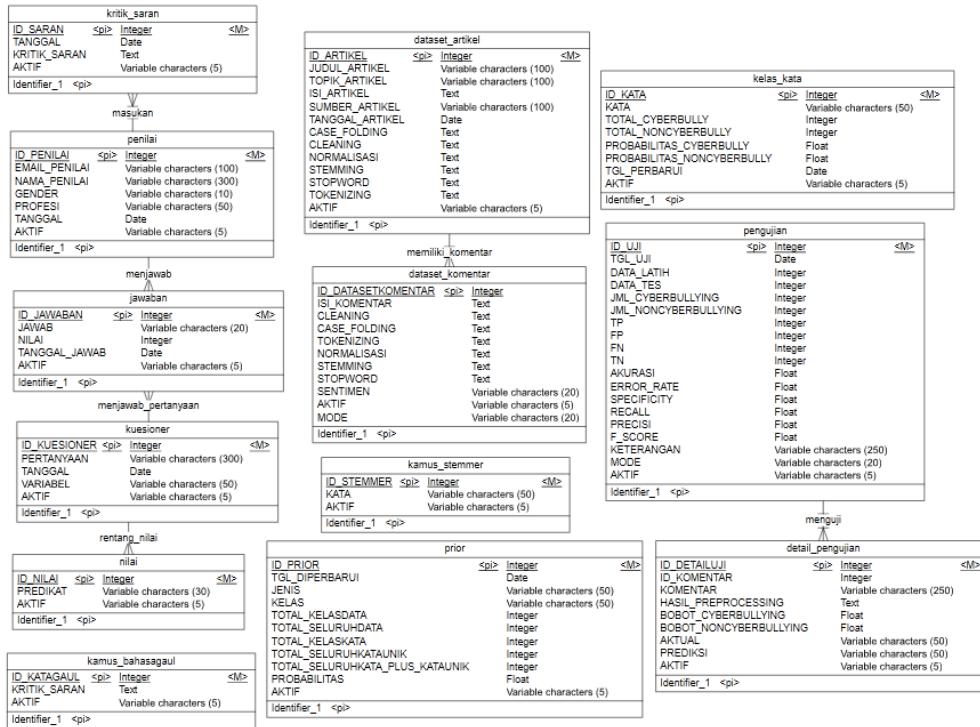
Kategori memiliki hubungan *one to many* dengan posting. Hal ini berarti setiap kategori bisa memiliki banyak posting. Tapi setiap posting hanya memiliki satu kategori. Posting memiliki hubungan *one to many* dengan komentar. Hal ini berarti setiap posting memiliki banyak komentar. Tapi setiap komentar hanya memiliki satu posting.

Pengguna memiliki hubungan *one to many* dengan komentar. Hal ini berarti setiap pengguna bisa memiliki banyak komentar. Tapi setiap komentar hanya bisa memiliki satu pengguna. Pengguna memiliki hubungan *one to many* dengan balasan_komentar. Hal ini berarti pengguna bisa memiliki banyak balasan_komentar. Tapi setiap balasan_komentar hanya memiliki satu pengguna. Komentar memiliki hubungan *one to many* dengan balasan_komentar. Hal ini berarti setiap komentar memiliki banyak balasan_komentar. Tapi setiap balasan_komentar hanya memiliki satu komentar.

Selanjutnya dijelaskan CDM II tentang tabel relasi dataset sebagaimana digambarkan pada gambar 6.2. Berbeda dengan CDM I yang lebih menggambarkan tabel relasi sistem manajemen konten. CDM II lebih menjelaskan pada tabel relasi yang dipakai dalam membangun dataset penyaring *cyberbullying*. Berikut dijelaskan relasi antar tabelnya selengkapnya sebagai berikut:

Penilai memiliki hubungan *one to many* dengan jawaban. Hal ini berarti setiap penilai bisa memiliki banyak jawaban. Tapi setiap jawaban hanya memiliki satu penilai. Kuesioner memiliki hubungan *one to many* dengan jawaban. Hal ini berarti setiap kuesioner memiliki banyak jawaban. Tapi setiap jawaban hanya memiliki satu kuesioner. Kuesioner memiliki hubungan *one to many* dengan nilai.

Hal ini berarti setiap kuesioner memiliki banyak nilai. Tapi setiap nilai hanya memiliki satu kuesioner.



**Gambar 6.2
CDM II Tentang Dataset Sistem Penyaring Cyberbullying**

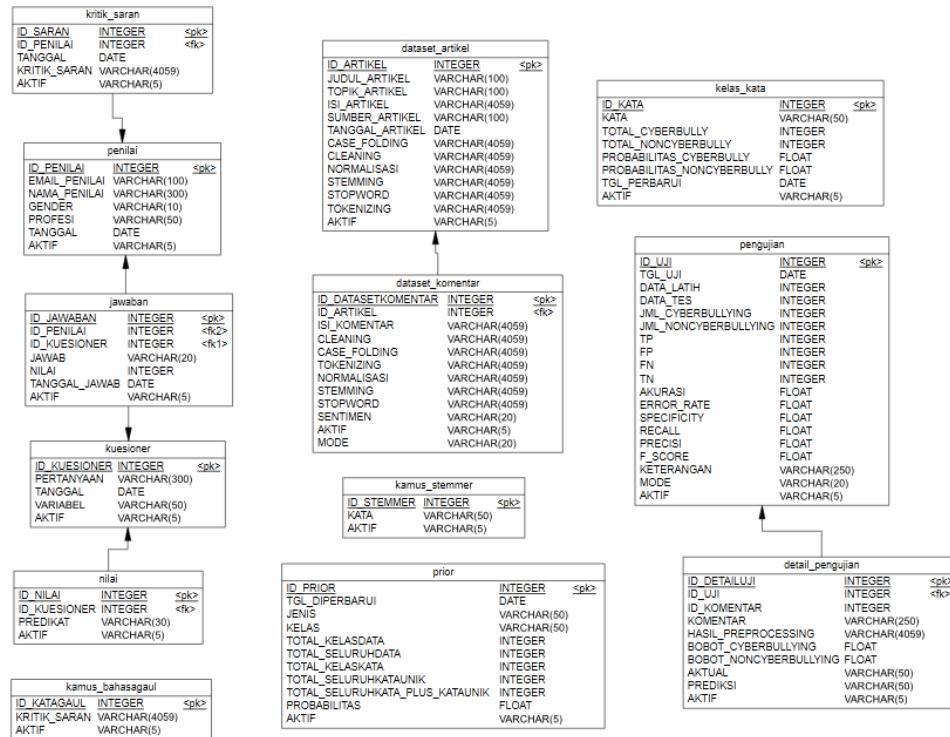
Dataset_artikel memiliki hubungan *one to many* dengan dataset_komentar.

Hal ini berarti setiap dataset_artikel memiliki banyak dataset_komentar. Tapi setiap dataset_komentar hanya memiliki satu dataset_artikel. Pengujian memiliki hubungan *one to many* dengan detail_pengujian. Hal ini berarti pengujian bisa memiliki banyak detail_pengujian. Tapi detail_pengujian hanya memiliki satu pengujian.

6.2.2 Physical Data Model

PDM merupakan gambaran secara detail basis data dalam bentuk fisik. Penggambaran rancangan PDM memperlihatkan struktur penyimpanan data yang benar pada basis data yang digunakan sesungguhnya. Dalam hal ini, PDM diimplementasikan ke dalam database MySQL versi 5.0. Sebagaimana ditunjukkan

pada gambar 6.1 yang menjelaskan bagaimana implementasi CDM I menjadi PDM I tentang sistem manajemen konten.

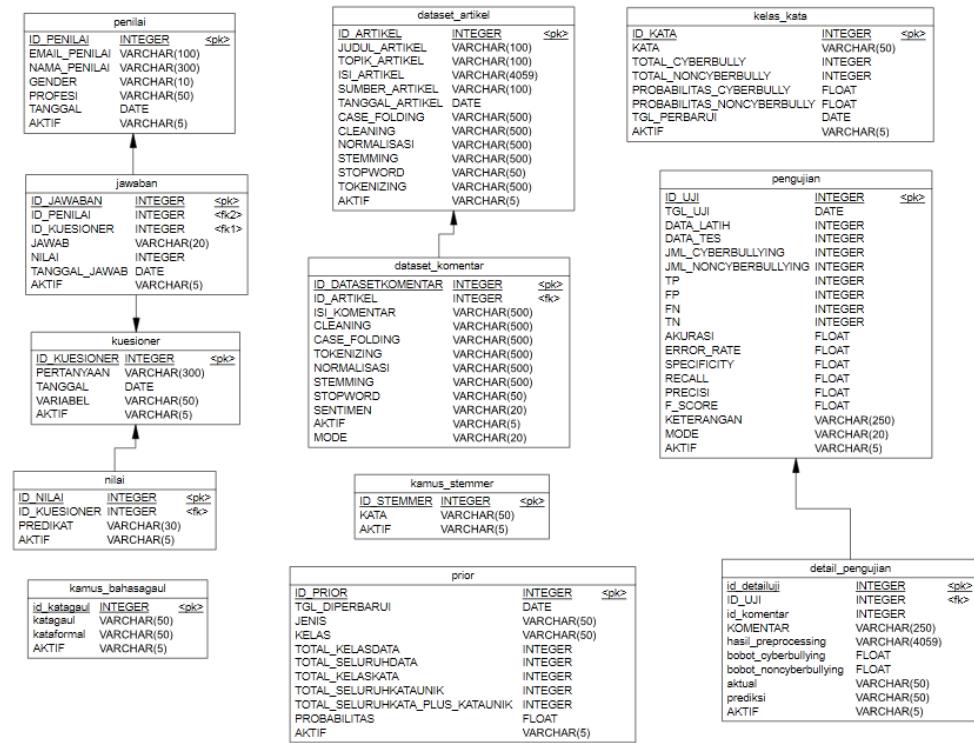


Gambar 6.3
PDM I Tentang Sistem Manajamen Konten

Dalam hal ini yang membedakan antara CDM dan PDM terletak pada implementasi relasi antar tabel menggunakan *foreign key* (FK). Dimana tabel relasi yang tergantung pada tabel relasi lainnya menjadi memiliki FK. Misalnya, tabel relasi galeri memiliki FK ID_PENGUNA dimana ID_PENGUNA merupakan *primary key* (PK) dari tabel relasi pengguna. Begitu juga dengan tabel relasi pengikut, detail_pengikut, mengikuti, detail_mengikuti, komentar, rating, view, notifikasi, balasan_komentar, yang masing-masing tergantung pada tabel pengguna sehingga masing-masing tabel relasi memiliki FK ID_PENGUNA. Adapula tabel relasi detail_pengikut tergantung pada tabel relasi pengikut sehingga tabel relasi detail_pengikut memiliki FK ID_PENGIKUT.

Sementara itu, tabel relasi detail_mengikuti tergantung pada tabel relasi mengikuti. Sehingga tabel relasi detail_mengikuti memiliki FK ID_MENGIKUTI.

Begitu pula tabel relasi posting yang tergantung pada tabel relasi kategori. Sehingga tabel relasi posting memiliki FK ID_KATEGORI yang merupakan PK dari tabel relasi kategori. Begitu juga tabel relasi *view*, rating, komentar tergantung pada tabel relasi posting sehingga masing-masing tabel relasi tersebut memiliki FK ID_POSTING. Adapula tabel relasi balasan_komentar tergantung pada tabel relasi komentar sehingga tabel relasi balasan_komentar memiliki FK ID_KOMENTAR.



Gambar 6.4
PDM II Tentang Dataset Penyaring Komentar Cyberbullying

Selanjutnya digambarkan PDM II yang merupakan implementasi dari CDM tentang dataset penyaring komentar *cyberbullying*. Pada gambar 6.4 dapat dilihat bahwa tabel relasi jawaban dan nilai tergantung pada tabel relasi kuesioner. Sehingga masing-masing tabel relasi tersebut memiliki FK ID_KUESIONER. Adapula tabel relasi dataset_komentar tergantung pada tabel relasi dataset_artikel. Sehingga tabel relasi dataset_komentar memiliki FK ID_DATASET_KOMENTAR. Terakhir tabel relasi detail_pengujian tergantung

pada tabel relasi pengujian sehingga tabel relasi detail_pengujian memiliki FK ID_UJI.

6.2.3 Deskripsi Struktur Tabel

Pada sub bab ini dijelaskan tentang deskripsi struktur tabel terkait manajemen sistem konten. Struktur tabel terdiri atas 30 tabel relasi. Adapun tabel relasinya meliputi tabel penilai, tabel jawaban, tabel kuesioner, tabel nilai, tabel kamus_bahasagaul, tabel prior, tabel kamus_stemmer, tabel dataset_komentar, tabel dataset_artikel, tabel kelas_kata, tabel pengujian, tabel detail_pengujian, tabel kontak, gallery, tabel pengikut, tabel detail_pengikut, tabel mengikuti, tabel detail_mengikuti, tabel notifikasi, tabel pengguna, tabel tagar, tabel identitas, tabel balasan_komentar, tabel komentar, tabel rating, tabel view, tabel posting, tabel kategori dan tabel kritik_saran. Berikut penjelasan selengkapnya:

- Tabel penilai

Tabel penilai digunakan untuk menyimpan biodata para responden dalam menilai kualitas website. Tabel ini terdiri atas 7 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel penilai, admin bisa melihat daftar para responden beserta biodatanya. Berikut ini rincian tabel penilai pada tabel 6.2:

Tabel 6.2
Tabel Penilai

Kolom	Tipe Data	Panjang	Keterangan
ID_PENILAI	Integer		Primary key
EMAIL_PENILAI	Varchar	100	
NAMA_PENILAI	Varchar	300	
GENDER	Varchar	10	
PROFESI	Varchar	50	
TANGGAL	Date		
AKTIF	Varchar	5	1=tampil, 0=tidak tampil

- Tabel jawaban

Tabel jawaban digunakan untuk menyimpan jawaban dari kuesioner yang dijawab oleh para responden dalam menilai kualitas *website*. Tabel ini

terdiri atas 7 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel penilai, admin bisa melihat daftar jawaban dari para responden. Dari jawaban ini akan diolah menjadi laporan evaluasi website Berikut ini penjelasan rinci dari tabel jawaban pada tabel 6.3:

**Tabel 6.3
Tabel Jawaban**

Kolom	Tipe Data	Panjang	Keterangan
ID_JAWABAN	<i>Integer</i>		<i>Primary key</i>
ID_PENILAI	<i>Integer</i>		<i>Foreign key</i>
ID_KUESIONER	<i>Integer</i>		<i>Foreign key</i>
JAWAB	<i>Varchar</i>	20	
NILAI	<i>Integer</i>		
TANGGAL_JAWAB	<i>Date</i>		
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel kuesioner

Tabel kuesioner digunakan untuk menyimpan kuesioner atau pertanyaan dalam menilai kualitas website. Tabel ini terdiri atas 5 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Pertanyaan tersebut selanjutnya akan dijawab oleh pengunjung website dalam menilai kualitas website.

**Tabel 6.4
Tabel Kuesioner**

Kolom	Tipe Data	Panjang	Keterangan
ID_KUESIONER	<i>Integer</i>		<i>Primary key</i>
PERTANYAAN	<i>Varchar</i>	300	
TANGGAL	<i>Date</i>		
VARIABEL	<i>Varchar</i>	50	
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel nilai

Tabel nilai digunakan untuk menyimpan nilai atau skala jawaban dari pertanyaan untuk mengukur kualitas website. Tabel ini terdiri atas 4 kolom.

Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel nilai, admin bisa membuat skala jawaban yang berguna dalam menilai kualitas website. Berikut ini penjelasan rinci dari tabel nilai pada tabel 6.5:

Tabel 6.5
Tabel Nilai

Kolom	Tipe Data	Panjang	Keterangan
ID_NILAI	<i>Integer</i>		<i>Primary key</i>
ID_KUESIONER	<i>Integer</i>		<i>Foreign key</i>
PREDIKAT	<i>Varchar</i>	30	
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel kritik_saran

Tabel kritik_saran digunakan untuk menyimpan masukan kritik dan saran dari responden terkait penilaian kualitas website. Tabel ini terdiri atas 5 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin bisa menyimpan dan melihat daftar kritik dan saran responden untuk pengembangan website.

Tabel 6.6
Tabel Kritik_Saran

Kolom	Tipe Data	Panjang	Keterangan
ID_SARAN	<i>Integer</i>		<i>Primary key</i>
ID_PENILAI	<i>Integer</i>		<i>Foreign key</i>
TANGGAL	<i>Date</i>		
KRITIK_SARAN	<i>Text</i>		
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel Kamus_Bahasagaul

Tabel kamus_bahasagaul digunakan untuk menyimpan kata-kata bahasa gaul dalam membangun dataset bahasa gaul. Tabel ini terdiri atas 4 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin bisa

menyimpan dan melihat daftar kata-kata bahasa gaul beserta kata formalnya. Berikut ini rincian tabel kamus_bahasagaul pada tabel 6.7:

**Tabel 6.7
Tabel Kamus_Bahasagaul**

Kolom	Tipe Data	Panjang	Keterangan
ID_KATAGAUL	<i>Integer</i>		<i>Primary key</i>
KATAGAUL	<i>Varchar</i>	50	
KATAFORMAL	<i>Varchar</i>	50	
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel *prior*

Tabel *prior* digunakan untuk menyimpan nilai prior dari model dataset penyaring *cyberbullying*. Tabel ini terdiri atas 11 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin bisa menyimpan dan melihat daftar prior dari model penyaring *cyberbullying* setelah dilakukan pelatihan data.

Berikut ini penjelasan rinci tabel prior pada tabel 6.8:

**Tabel 6.8
Tabel Prior**

Kolom	Tipe Data	Panjang	Keterangan
ID_PRIOR	<i>Integer</i>		<i>Primary key</i>
TGL_DIPERBARUI	<i>Varchar</i>	50	
JENIS	<i>Varchar</i>	50	
TOTAL_KELASDATA	<i>Integer</i>		
TOTAL_SELURUHDATA	<i>Integer</i>		
TOTAL_KELASKATA	<i>Integer</i>		
TOTAL_SELURUHKATAU_NIK	<i>Integer</i>		
TOTAL_SELURUHKATA_PLUS_KATAUNIK	<i>Integer</i>		
PROBABILITAS	<i>Float</i>		
NAMA_MODEL	<i>Varchar</i>	100	
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel kamus_Stemmer

Tabel stemmer digunakan untuk menyimpan kata-kata formal dalam dataset kamus stemmer. Tabel ini terdiri atas 3 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Proses stemming melibatkan penyederhanaan kata-kata menjadi kata dasarnya. Oleh sebab itu, tabel ini berguna untuk mengecek kata-kata yang di-stemming itu terdapat dalam daftar kata dasar baku.

Tabel 6.9
Tabel Kamus_Stemmer

Kolom	Tipe Data	Panjang	Keterangan
ID_STEMMER	Integer		<i>Primary key</i>
KATA	Varchar	50	
AKTIF	Varchar	5	1 = tampil, 0 = tidak tampil

- Tabel dataset_komentar

Tabel dataset_komentar digunakan untuk menyimpan komentar artikel menjadi dataset komentar. Tabel ini terdiri atas 12 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin bisa menyimpan dan melihat daftar komentar beserta seluruh proses *preprocessing*. Dataset komentar selanjutnya akan diproses menjadi model penyaring komentar *cyberbullying*. Berikut ini penjelasan rinci dari tabel dataset komentar pada tabel 6.10:

Tabel 6.10
Tabel Dataset_Komentar

Kolom	Tipe Data	Panjang	Keterangan
ID_DATASETKOMENTAR	Integer		<i>Primary key</i>
ISI_KOMENTAR	Varchar	500	
CLEANING	Varchar	500	
CASE_FOLDING	Varchar	500	
TOKENIZING	Varchar	500	
NORMALISASI	Varchar	500	
STEMMING	Varchar	500	
STOPWORD	Varchar	500	
NGRAM	Varchar	500	

**Tabel 6.10
(Lanjutan)**

Kolom	Tipe Data	Panjang	Keterangan
KATEGORI	Varchar	20	
MODE	Varchar	20	
AKTIF	Varchar	5	1=aktif, 0=non-aktif

- Tabel dataset_artikel

Tabel dataset_artikel digunakan untuk menyimpan berbagai konten artikel. Tabel ini terdiri atas 13 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin bisa menyimpan dan melihat daftar konten artikel yang telah dikumpulkan dari berbagai sumber. Berikut ini penjelasan rinci dari tabel dataset komentar pada tabel 6.11:

**Tabel 6.11
Tabel Dataset_Artikel**

Kolom	Tipe Data	Panjang	Keterangan
ID_ARTIKEL	Integer		Primary key
JUDUL_ARTIKEL	Varchar	100	
TOPIK_ARTIKEL	Varchar	100	
ISI_ARTIKEL	Text		
SUMBER_ARTIKEL	Varchar	100	
TANGGAL_ARTIKEL	Date		
CLEANING	Text		
CASE_FOLDING	Text		
TOKENIZING	Text		
NORMALISASI	Text		
STEMMING	Text		
STOPWORD	Text		
AKTIF	Varchar	5	1 = tampil, 0 = tidak tampil

- Tabel kelas_kata

Tabel kelas_kata digunakan untuk menyimpan kelas kata hasil proses pengolahan dataset_komentar menjadi model penyaring komentar *cyberbullying*. Tabel ini terdiri atas 8 kolom. Pada kolom aktif memiliki nilai

1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin bisa menyimpan dan melihat rincian kategori dari tiap kelas kata. Berikut ini penjelasan rinci dari tabel kelas_kata yang ditunjukkan pada tabel 6.12:

**Tabel 6.12
Tabel Kelas_Kata**

Kolom	Tipe Data	Panjang	Keterangan
ID_KATA	<i>Integer</i>		<i>Primary key</i>
KATA	<i>Varchar</i>	50	
TOTAL_CYBERBULLYING	<i>Integer</i>		
TOTAL_NONCYBERBULLYING	<i>Integer</i>		
PROBABILITAS_CYBERBULLYING	<i>Float</i>		
PROBABILITAS_NONCYBERBULLYING	<i>Float</i>		
TGL_PERBARUI	<i>Date</i>		
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel pengujian

Tabel pengujian digunakan untuk menyimpan informasi terkait hasil pengujian. Tabel ini terdiri atas 19 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin bisa menyimpan dan melihat rincian hasil evaluasi model seperti akurasi, *error*, *specificity*, *recall*, *precision*, dan *f-score*. Berikut ini penjelasan rinci dari tabel pengujian pada tabel 6.13:

**Tabel 6.13
Tabel Pengujian**

Kolom	Tipe Data	Panjang	Keterangan
ID_UJI	<i>Integer</i>		<i>Primary key</i>
TGL_UJI	<i>Date</i>		
DATA_LATIH	<i>Integer</i>		
DATA_TES	<i>Integer</i>		
JML_CYBERBULLYING	<i>Float</i>	100	
JML_NONCYBERBULLYING	<i>Float</i>		

**Tabel 6.13
(Lanjutan)**

Kolom	Tipe Data	Panjang	Keterangan
TP	<i>Integer</i>		
FP	<i>Integer</i>		
FN	<i>Integer</i>		
TN	<i>Integer</i>		
AKURASI	<i>Float</i>		
ERROR_RATE	<i>Float</i>		
SPECIFICITY	<i>Float</i>		
RECALL	<i>Float</i>		
PRECISION	<i>Float</i>		
F_SCORE	<i>Float</i>		
KETERANGAN	<i>Varchar</i>	250	
MODE	<i>Varchar</i>	20	
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel detail_pengujian

Tabel detail_pengujian digunakan untuk menyimpan data terkait rincian hasil pengujian. Tabel ini terdiri atas 10 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin bisa menyimpan dan melihat rincian hasil kategori per komentar yang diuji. Berikut ini rincian tabel detail_pengujian pada tabel 6.14:

**Tabel 6.14
Tabel Detail_Pengujian**

Kolom	Tipe Data	Panjang	Keterangan
ID_DETAILUJI	<i>Integer</i>		<i>Primary key</i>
ID_UJI	<i>Integer</i>		<i>Foreign key</i>
ID_KOMENTAR	<i>Integer</i>		
KOMENTAR	<i>Varchar</i>	250	
HASIL_PREPROCESSING	<i>Text</i>		
BOBOT_CYBERBULLYING	<i>Float</i>		
BOBOT_NONCYBERBULLYING	<i>Float</i>		
AKTUAL	<i>Varchar</i>	50	
PREDIKSI	<i>Varchar</i>	50	
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel kontak

Tabel kontak digunakan untuk menyimpan informasi terkait deskripsi website. Tabel ini terdiri atas 4 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin bisa menyimpan dan melihat informasi seputar deskripsi website. Berikut ini rincian kontak pada tabel 6.15:

**Tabel 6.15
Tabel Kontak**

Kolom	Tipe Data	Panjang	Keterangan
ID_KONTAK	<i>Integer</i>		<i>Primary key</i>
NAMA_KONTAK	<i>Varchar</i>	100	
DESKRIPSI	<i>Text</i>		
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel gallery

Tabel gallery digunakan untuk menyimpan data galeri yang bisa digunakan member maupun admin untuk ditampilkan pada posting. Tabel ini terdiri atas 6 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Atribut PHOTO berfungsi menyimpan URL gambar. Atribut nama gambar menyimpan nama galeri. Atribut keterangan menyimpan keterangan berkaitan dengan galeri. Melalui tabel ini, admin bisa menyimpan dan melihat daftar galeri serta menyalin url galeri untuk selanjutnya digunakan dalam posting artikel.

**Tabel 6.16
Tabel Gallery**

Kolom	Tipe Data	Panjang	Keterangan
ID_GALLERY	<i>Integer</i>		<i>Primary key</i>
ID_PENGGUNA	<i>Integer</i>		<i>Foreign key</i>
NAMA_GALLERY	<i>Varchar</i>	100	
KETERANGAN	<i>Varchar</i>	250	
PHOTO	<i>Varchar</i>	100	
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel pengikut

Tabel pengikut digunakan untuk menyimpan ringkasan pengikut dari member atau admin. Tabel ini terdiri atas 4 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin maupun member bisa menyimpan dan melihat ringkasan pengikutnya seperti total pengikutnya. Berikut ini penjelasan rinci dari pengikut pada tabel 6.17:

**Tabel 6.17
Tabel Pengikut**

Kolom	Tipe Data	Panjang	Keterangan
ID_PENGIKUT	Integer		<i>Primary key</i>
ID_PENGGUNA	Integer		<i>Foreign key</i>
TOTAL_PENGIKUT	Integer		
AKTIF	Varchar	5	1 = tampil, 0 = tidak tampil

- Tabel notifikasi

Tabel notifikasi digunakan untuk menyimpan data notifikasi terkait berbagai aktivitas yang diikuti admin maupun member. Tabel ini terdiri atas 6 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin maupun member bisa daftar pemberitahuan tertentu seperti posting baru, komentar baru, rating baru, dan menandai sudah dibacanya. Berikut ini penjelasan rinci dari notifikasi pada tabel 6.18:

**Tabel 6.18
Tabel Notifikasi**

Kolom	Tipe Data	Panjang	Keterangan
ID_NOTIFIKASI	Integer		<i>Primary key</i>
ID_PENGGUNA	Integer		<i>Foreign key</i>
TANGGAL_NOTIFIKASI	Date & Time		
NOTIFIKASI	Varchar	100	
DIBACA	Varchar	5	
AKTIF	Varchar	5	1 = tampil, 0 = tidak tampil

- Tabel detail_pengikut

Tabel detail_pengikut digunakan untuk menyimpan data pengikut dari member maupun admin. Tabel ini terdiri atas 4 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin maupun member bisa daftar pengikutnya. Berikut ini rincian detail_pengikut pada tabel 6.19:

**Tabel 6.19
Tabel Detail_Pengikut**

Kolom	Tipe Data	Panjang	Keterangan
ID_DETAILPENGIKUT	Integer		<i>Primary key</i>
ID_PENGIKUT	Integer		<i>Foreign key</i>
ID_PENGGUNA	Integer		<i>Foreign key</i>
AKTIF	Varchar	5	1 = tampil, 0 = tidak tampil

- Tabel mengikuti

Tabel mengikuti digunakan untuk menyimpan ringkasan mengikuti dari member atau admin. Tabel ini terdiri atas 4 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin maupun member bisa menyimpan dan melihat ringkasan akun yang diikutinya seperti total mengikuti. Berikut ini penjelasan rinci dari mengikuti pada tabel 6.20:

**Tabel 6.20
Tabel Mengikuti**

Kolom	Tipe Data	Panjang	Keterangan
ID_MENGIKUTI	Integer		<i>Primary key</i>
ID_PENGGUNA	Integer		<i>Foreign key</i>
TOTAL_MENGIKUTI	Integer		
AKTIF	Varchar	5	1 = tampil, 0 = tidak tampil

- Tabel detail_mengikuti

Tabel detail_mengikuti digunakan untuk menyimpan data mengikuti dari member maupun admin. Tabel ini terdiri atas 4 kolom. Pada kolom aktif

memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Admin maupun member bisa daftar akun yang diikutinya.

Tabel 6.21
Tabel Detail_Mengikuti

Kolom	Tipe Data	Panjang	Keterangan
ID_DETAILMENGIKUTI	Integer		<i>Primary key</i>
ID_MENGIKUTI	Integer		<i>Foreign key</i>
ID_PENGGUNA	Integer		<i>Foreign key</i>
AKTIF	Varchar	5	1 = tampil, 0 = tidak tampil

- Tabel pengguna

Tabel pengguna digunakan untuk menyimpan biodata dari member maupun admin. Tabel ini terdiri atas 11 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Atribut username berfungsi sebagai akun login. Sementara atribut password akan menyimpan data dalam bentuk enkripsi MD5. Melalui tabel ini, admin dan member melihat dan mengubah biodata profilnya.

Tabel 6.22
Tabel Pengguna

Kolom	Tipe Data	Panjang	Keterangan
ID_PENGGUNA	Integer		<i>Primary key</i>
USERNAME	Varchar	100	
EMAIL	Varchar	254	
PASSWORD	Varchar	250	
FIRST_NAME	Varchar	50	
LAST_NAME	Varchar	50	
COMPANY	Varchar	100	
PHONE	Varchar	20	
ABOUT_ME	Varchar	250	
PHOTO_PROFIL	Varchar	100	
AKTIF	Varchar	5	1 = tampil, 0 = tidak tampil

- Tabel view

Tabel view digunakan untuk menyimpan data tayangan yang diakses pengujung terhadap artikel. Tabel ini terdiri atas 5 kolom. Pada kolom aktif

memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini nantinya data akan diolah menjadi laporan perkembangan tayangan. Berikut rincian tabel *view* pada tabel 6.23:

**Tabel 6.23
Tabel View**

Kolom	Tipe Data	Panjang	Keterangan
ID_VIEW	<i>Integer</i>		<i>Primary key</i>
ID_POSTING	<i>Integer</i>		<i>Foreign key</i>
VIEW	<i>Integer</i>		
TANGGAL_VIEW	<i>Date</i>		
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel komentar

Tabel komentar digunakan untuk menyimpan data komentar yang diberikan member maupun admin terhadap posting artikel. Tabel ini terdiri atas 11 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin maupun member melihat daftar komentar beserta pelanggaran yang ada dari member maupun admin terhadap konten artikel sehingga memungkinkan interaksi antar pengguna. Berikut ini rincian komentar pada tabel 6.24:

**Tabel 6.24
Tabel Komentar**

Kolom	Tipe Data	Panjang	Keterangan
ID_KOMENTAR	<i>Integer</i>		<i>Primary key</i>
ID_POSTING	<i>Integer</i>		<i>Foreign key</i>
ID_PENGGUNA	<i>Integer</i>		<i>Foreign key</i>
KOMENTAR	<i>Varchar</i>	250	
TANGGAL_KOMENTAR	<i>Date</i>		
KATEGORI	<i>Varchar</i>	20	
LAPORKAN	<i>Varchar</i>	20	
PELANGGARAN	<i>Varchar</i>	20	
KETERANGAN	<i>Varchar</i>	250	
TANGGAL_PELAPORAN	<i>Date</i>		
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel tagar

Tabel tagar digunakan untuk menyimpan data tag atau kata kunci yang diakses pengunjung. Tabel ini terdiri atas 5 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin melihat rincian daftar kata kunci yang sering dikunjungi pengunjung. Berikut ini rincian tabel tagar ditunjukkan tabel 6.25:

**Tabel 6.25
Tabel Tagar**

Kolom	Tipe Data	Panjang	Keterangan
ID_TAGAR	<i>Integer</i>		<i>Primary key</i>
KATA_KUNCI	<i>Varchar</i>	50	
TOTAL	<i>Integer</i>		
TANGGAL	<i>Date</i>		
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel identitas

Tabel identitas digunakan untuk menyimpan data identitas dan pengaturan website. Tabel ini terdiri atas 6 kolom. Melalui tabel ini, admin melihat dan mengubah identitas website seperti nama website, alamat website, meta deskripsi, meta keyword, hingga mengubah template *front end*. Berikut ini penjelasan rinci dari identitas pada tabel 6.26:

**Tabel 6.26
Tabel Identitas**

Kolom	Tipe Data	Panjang	Keterangan
ID_IDENTITAS	<i>Integer</i>		<i>Primary key</i>
NAMA_WEBSITE	<i>Varchar</i>	255	
ALAMAT_WEBSITE	<i>Varchar</i>	255	
META_DESKRIPSI	<i>Text</i>		
META_KEYWORD	<i>Text</i>		
TEMPLATE	<i>Vachar</i>	100	

- Tabel rating

Tabel rating digunakan untuk menyimpan data rating yang diberikan member maupun admin terhadap konten artikel. Tabel ini terdiri atas 6 kolom.

Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin maupun member melihat daftar rating berdasarkan artikelnya. Berikut ini penjelasan rinci dari rating pada tabel 6.27:

**Tabel 6.27
Tabel Rating**

Kolom	Tipe Data	Panjang	Keterangan
ID_RATING	<i>Integer</i>		<i>Primary key</i>
ID_POSTING	<i>Integer</i>		<i>Foreign key</i>
ID_PENGGUNA	<i>Integer</i>		<i>Foreign key</i>
RATING	<i>Integer</i>		
TANGGAL_RATING	<i>Date</i>		
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel kategori

Tabel kategori digunakan untuk menyimpan data kategori artikel. Tabel ini terdiri atas 3 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin menyimpan, melihat dan mengubah daftar kategori artikel. Berikut ini penjelasan rinci dari kategori pada tabel 6.28:

**Tabel 6.28
Tabel Kategori**

Kolom	Tipe Data	Panjang	Keterangan
ID_KATEGORI	<i>Integer</i>		<i>Primary key</i>
NAMA_KATEGORI	<i>Varchar</i>	100	
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel posting

Tabel posting digunakan untuk menyimpan data posting yang ditulis member maupun admin. Tabel ini terdiri atas 17 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin maupun member menyimpan,

melihat dan mengubah konten artikelnya. Berikut ini penjelasan rinci dari posting pada tabel 6.29:

**Tabel 6.29
Tabel Posting**

Kolom	Tipe Data	Panjang	Keterangan
ID_POSTING	<i>Integer</i>		<i>Primary key</i>
ID_PENGGUNA	<i>Integer</i>		<i>Foreign key</i>
ID_KATEGORI	<i>Integer</i>		<i>Foreign key</i>
JUDUL	<i>Integer</i>		
JUDUL_SEO	<i>Date</i>		
ISI_ARTIKEL	<i>Text</i>		
FEATURED	<i>Varchar</i>	5	
THREAD	<i>Varchar</i>	5	
CHOICE	<i>Varchar</i>	5	
PHOTO	<i>Varchar</i>	100	
TANGGAL	<i>Date</i>		
DESKRIPSI_ARTIKEL	<i>Varchar</i>	200	
VIEW	<i>Integer</i>		
RATING	<i>Integer</i>		
JUMLAH_RATING	<i>Integer</i>		
TOTAL_KOMENTAR	<i>Integer</i>		
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

- Tabel balasan_komentar

Tabel balasan_komentar digunakan untuk menyimpan data balasan_komentar yang diberikan member maupun admin terhadap komentar lainnya. Tabel ini terdiri atas 12 kolom. Pada kolom aktif memiliki nilai 1 yang berarti menampilkan data dan nilai 0 berarti tidak menampilkan data. Melalui tabel ini, admin maupun member melihat daftar balasan_komentar beserta pelanggaran yang ada dari member maupun admin terhadap komentar lainnya. Berikut ini penjelasan rinci dari balasan_komentar pada tabel 6.30:

**Tabel 6.30
Tabel Balasan_Komentar**

Kolom	Tipe Data	Panjang	Keterangan
ID_BALASAN	<i>Integer</i>		<i>Primary key</i>
ID_KOMENTAR	<i>Integer</i>		<i>Foreign key</i>

**Tabel 6.30
(Lanjutan)**

Kolom	Tipe Data	Panjang	Keterangan
ID_PENGGUNA	<i>Integer</i>		<i>Foreign key</i>
KOMENTAR	<i>Varchar</i>	250	
TANGGAL_KOMENTAR	<i>Date</i>		
KATEGORI	<i>Varchar</i>	20	
LAPORKAN	<i>Varchar</i>	20	
PELANGGARAN	<i>Varchar</i>	20	
KETERANGAN	<i>Varchar</i>	250	
TANGGAL_PELAPORAN	<i>Date</i>		
MEMBALAS	<i>Integer</i>		
AKTIF	<i>Varchar</i>	5	1 = tampil, 0 = tidak tampil

6.3 Desain Antar Muka

Pada sub bab ini dijelaskan tentang desain antar muka pada sistem manajemen konten. Adapun penjelasannya terbagi menjadi 2 sub bab meliputi *front end* dan *back end*. Bagian *front end* merupakan antar muka depan yang berinteraksi langsung dengan pembaca artikel. Sementara *back end* merupakan antar muka belakang yang berinteraksi dengan admin dan member. Berikut penjelasan selengkapnya:

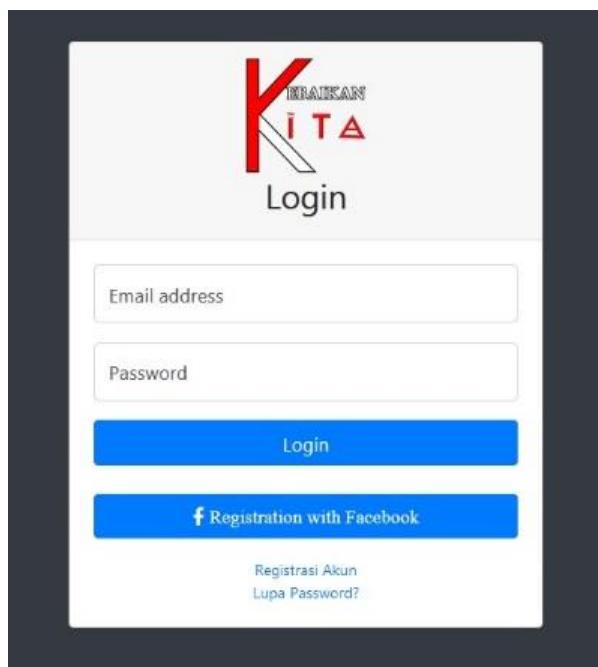
6.3.1 Desain Front End

Desain frontend merupakan desain antar muka yang terdapat pada depan *website*. *Front end website* bisanya berinteraksi langsung dengan pembaca. Bagian *front end* meliputi: desain *login*, desain registrasi, desain beranda, desain artikel, desain peringatan *cyberbullying*, desain tentang *website*, desain kebijakan *website*, desain kebijakan *website*, desain FAQ, dan desain evaluasi *website*. Berikut penjelasan selengkapnya:

- Desain *login*

Desain login ditunjukkan pada gambar 6.5 terdiri atas form *login*, tautan registrasi dan tautan reset *password*. Form *login* berisi textbox username dan textbox password serta tombol login. *Username* harus diisi dengan e-mail pengguna yang terdaftar dalam sistem. Pengguna harus mengimputkan

username dan *password* untuk bisa *login* ke sistem. Jika pengguna lupa *password*, dia bisa mengklik link lupa *password* sehingga otomatis diarahkan ke halaman reset *password*. Nantinya pengguna diminta memasukkan e-mailnya sehingga bisa dikirim halaman untuk reset *password*. Selain itu, pengguna baru bisa mendaftarkan dirinya dengan mengklik tautan registrasi sehingga diarahkan ke halaman registrasi pengguna baru.



Gambar 6.5
Desain Login

- Desain registrasi

Desain registrasi ditunjukkan pada gambar 6.6 terdiri atas form registrasi dan tombol login. Form registrasi terdiri atas textbox nama depan, textbox nama belakang, textbox e-mail, textbox no telp, textbox nama organisasi, textbox password dan konfirmasi password, serta tombol register. Pengguna baru diwajibkan mengisi form tersebut. Biodata form dibuat singkat supaya pengguna baru tidak merasa keberatan dengan berbagai isian yang menyusahkannya. Untuk isian organisasi, pengguna baru bisa menginputkan nama organisasinya bisa berupa nama sekolah, nama perusahaan, maupun nama organisasi tertentu. Adapun untuk isian password dan konfirmasi

password harus sama. Keduanya akan dihidden selama penginputan untuk menjaga kerahasiaan data. Jika terdapat isian yang dikosongkan, sistem secara otomatis akan mengirimkan pesan error. Pengguna baru wajib mengisi semua isian form tersebut hingga berhasil. Setelah berhasil, sistem akan mengirimkan pesan sukses, selanjutnya pengguna baru bisa mengklik tombol login untuk mengakses sistem.

The screenshot shows a registration form titled "Registrasi Akun". At the top center is a logo consisting of a red stylized "K" and "I" followed by the word "ITA" in red, with the text "SEBAIKAN" above it. Below the title, there are several input fields:

- Two input fields side-by-side: "Nama depan" and "Nama belakang".
- A single input field for "E-mail".
- A single input field for "No telp".
- A single input field for "Perusahaan/ Sekolah/ Organisasi".
- Two input fields side-by-side: "Password" and "Konfirmasi password".

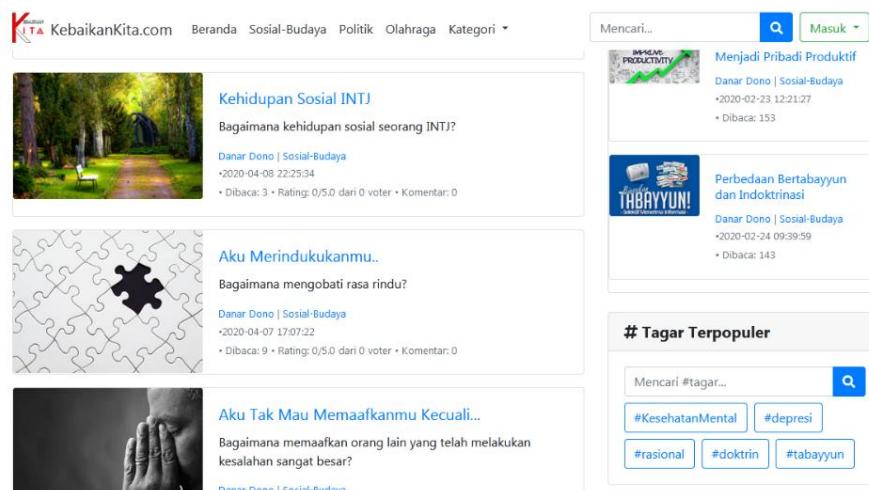
At the bottom of the form is a large blue button labeled "Register". Below the "Register" button is a smaller blue link labeled "Login".

Gambar 6.6
Desain Registrasi

- Desain beranda

Desain beranda menampilkan berbagai konten aktual yang telah dipublikasi oleh para member sebagaimana ditunjukkan pada gambar 6.7. Desain home terdiri atas beberapa bagian meliputi navigasi, artikel, side bar dan footer. Navigasi berisi kategori topik artikel seperti sosial budaya, politik dan olahraga. Pengguna juga dapat mencari artikel melalui kotak pencarian. Pengguna juga bisa mengakses tab login masuk. Di bawahnya terdapat kumpulan artikel terkini yang telah dipublikasikan lengkap dengan judul artikel, deskripsi, penulisnya, kategorinya hingga rincian statistik umum artikelnya. Di kanannya terdapat side bar yang berisi daftar artikel terpopuler

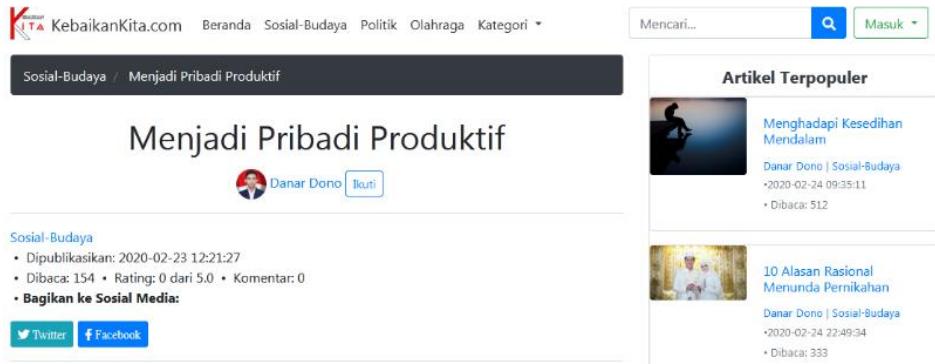
dan tagar terpopuler. Daftar artikel terpopuler menunjukkan 5 kumpulan artikel yang paling sering diakses pengunjung. Sedangkan tagar populer menunjukkan 5 tag terpopuler yang sering diakses pengunjung. Di bawahnya terdapat footer yang berisi informasi umum terkait website seperti kontak, kebijakan, FAQ, kategori, hingga evaluasi website. Pengunjung bisa memanfaatkan footer untuk mencari tahu informasi umum berhubungan dengan website seperti melihat kontaknya, kebijakan umumnya, tanya jawab paling umum terkait website, hingga memberikan masukan evaluasi bagi pengembangan website ke depannya.



Gambar 6.7
Desain Beranda

- Desain detail artikel

Desain artikel menampilkan detail konten artikel. Detail artikel terdiri atas beberapa bagian meliputi statistik artikel, konten artikel, rating artikel dan komentar artikel. Statistik artikel menampilkan rincian statistik umum terkait artikel seperti total dibaca, rata-rata rating, total komentar, dan tangga publikasi. Pengguna juga bisa membagikan artikel dengan memanfaatkan fitur bagikan ke media sosial lainnya baik twitter maupun facebook. Selain itu, member bisa mengikuti akun penulis dengan memanfaatkan fitur following.



**Gambar 6.8
Statistik Detail Artikel**

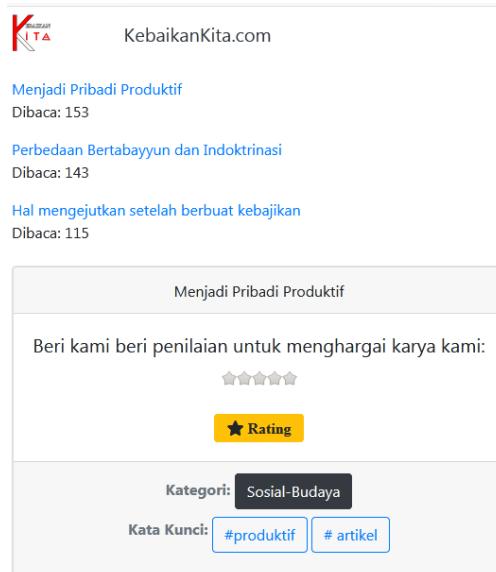
Konten artikel menampilkan rincian konten artikel seperti galeri, deskripsi artikel dan isi artikel sebagaimana ditunjukkan pada gambar 6.9. Galeri menampilkan ilustrasi artikel. Deskripsi artikel merupakan ringkasan artikel secara umum. Sedangkan isi artikel menjelaskan detail informasi artikel.



**Gambar 6.9
Konten Artikel**

Rating artikel menampilkan fitur penilaian artikel, judul artikel lain yang ditulis penulis, dan kata kunci artikel. Member bisa memberikan penilaian artikel memanfaatkan fitur rating dengan skala 1 sampai 5. Skala 1 berarti sangat buruk, 2 berarti buruk, 3 berarti cukup, 4 berarti baik, 5 berarti sangat baik. Setiap member hanya boleh memberikan 1 rating per artikelnya. Jika

member memberikan rating lebih dari 1 maka sistem akan memunculkan pesan peringatan kepada member untuk tidak merating artikel yang telah dinilainya. Penilaian artikel tersebut akan diolah menjadi laporan baik berupa statistik rata-rata rating artikel maupun perbandingan rating antar artikel. Pengguna juga bisa mengakses konten artikel lainnya yang ditulis penulis dari daftar judul. Di bawah rating, terdapat kategori dan kata kunci artikel. Pengguna bisa mengakses kategori dan kata kunci (tag) tersebut untuk mencari konten lainnya yang berhubungan dengan kategori maupun kata kunci. Hasilnya menampilkan daftar artikel yang relevan dengan kategori atau kata kunci artikel.



Gambar 6.10
Rating Artikel

- Desain peringatan *cyberbullying*

Desain peringatan *cyberbullying* menampilkan pesan peringatan terkait komentar yang berpotensi mengandung *cyberbullying*. Fitur ini akan aktif tatkala member mengirimkan komentar yang berpotensi mengandung *cyberbullying*. Sehingga diharapkan member terkait bisa memikirkan komentarnya lebih lanjut untuk dikirim atau membantalkannya. Berikut disajikan desain pesan peringatan *cyberbullying*:



Gambar 6.11 Desain Pesan Peringatan *Cyberbullying*

- Desain tentang *website*

Desain tentang *website* menampilkan informasi umum berhubungan dengan website. Halaman ini menampilkan deskripsi umum seperti deskripsi website hingga panduan umum terkait website. Fitur bisa diakses semua pengguna. Berikut disajikan desain tentang *website*:



Gambar 6.12 *Tentang Website*

- Desain kebijakan *website*

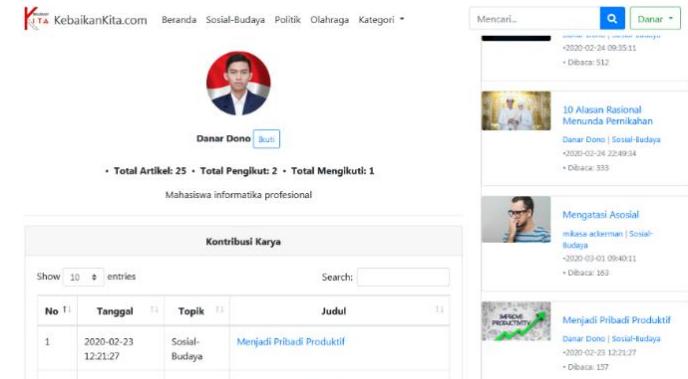
Desain kebijakan *website* menampilkan informasi berhubungan dengan kebijakan *website*. Kebijakan ini berkaitan dengan aturan penggunaan *website* beserta larangan penyalahgunaannya. Dari halaman ini, pengguna bisa melihat segala kebijakan terkait penggunaan *website*. Berikut disajikan desain kebijakan *website*:



Gambar 6.13
Kebijakan Website

- Desain profil

Desain profil menampilkan informasi umum berhubungan dengan penulis konten artikel. Melalui halaman profil, pengunjung bisa melihat rincian profil penulis seperti total artikel, total pengikut, total yang diikuti, dan biodatanya. Pengunjung juga bisa melihat daftar artikel yang dimiliki penulis. Berikut disajikan desain tampilan profil:



Gambar 6.14
Desain Profil

- Desain FAQ

Desain FAQ menampilkan pertanyaan dan jawaban umum yang sering ditanyakan pengguna terkait website. Pengunjung bisa melihat pertanyaan dan jawaban umum yang berhubungan dengan website serta bisa juga memberikan masukan pertanyaan. Segala pertanyaan yang sering ditanyakan berhubungan

dengan *website* bisa dijawab menggunakan FAQ. Berikut disajikan desain tampilan FAQ.

The screenshot shows a FAQ section with two questions:

- Apa itu cyber-bullying?**
- Apa itu anti-cyberbullying?**

The first question has a detailed answer:

Cyberbullying adalah kerugian yang disengaja dan berulang kali ditimbulkan melalui penggunaan komputer ponsel, dan perangkat elektronik lainnya. Para remaja biasanya menggunakan teknologi untuk melecehkan, mengancam, dan mencemarkan nama baik teman-teman mereka. Misalnya, seorang remaja menyebarkan desas-desus melalui ponselnya untuk disebarluaskan pada *platform* media sosial dengan tujuan mengolok-olok temannya. **Cyberbullying** berdampak negatif terhadap kehidupan remaja. Para korban biasanya merasa tertekan, sedih, marah, dan frustasi. **Cyberbullying** juga berdampak negatif terhadap harga diri, masalah keluarga, kesulitan akademik, kekerasan di sekolah, dan berbagai perilaku menyimpang lainnya. Bahkan sejumlah laporan menyebutkan para korban memutuskan untuk bunuh diri.

Gambar 6.15
Desain FAQ

- Desain evaluasi *website*

Desain evaluasi *website* menampilkan halaman evaluasi website yang digunakan pengunjung untuk menilai kualitas website. Pengunjung diharuskan mengisi identitasnya lalu menjawab pertanyaan. Pengunjung juga bisa memberikan kritik, saran dan pertanyaan berhubungan dengan website. berikut disajikan desain tampilan evaluasi *website*:

Identitas Pengguna

Nama Lengkap	<input type="text"/>
E-mail	<input type="text"/>
Gender	<input type="radio"/> Pria <input type="radio"/> Wanita
Profesi	<input type="radio"/> Pengawai Negeri Sipil <input type="radio"/> Pegawai Swasta <input type="radio"/> Wirausaha / Pengusaha <input type="radio"/> Pelajar/ Mahasiswa <input type="radio"/> Lainnya

Pertanyaan

Seberapa cepat loading halaman situs KebaikanKita.com?	<input type="radio"/> Sangat cepat <input type="radio"/> Cepat <input type="radio"/> Cukup
--	--

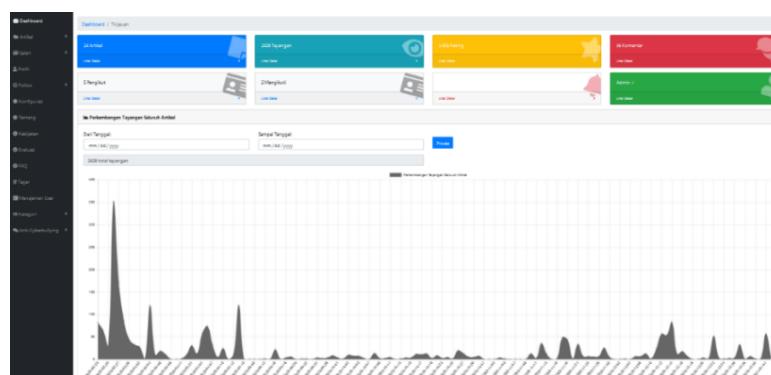
Gambar 6.16
Evaluasi Website

6.3.2 Desain Back End

Desain *back end* merupakan desain interface yang hanya bisa diakses oleh pengguna yang sudah *login* sebagai member maupun admin. *Back end* biasanya berinteraksi di belakang tampilan *website*, tepatnya berhubungan dengan pengelolaan konten *website*. Bagian *back end* meliputi: desain dashboard, desain notifikasi, desain konfigurasi profil, desain manajemen user, desain manajemen galeri, desain manajemen artikel, desain konfigurasi website, desain manajemen tentang website, desain manajemen kebijakan website, desain manajemen FAQ dan desain manajemen kategori. Berikut penjelasan selengkapnya:

- Desain dashboard

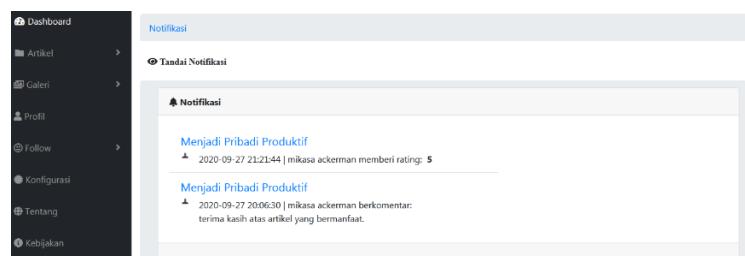
Desain dashboard menampilkan berbagai statistik umum seputar akun pengguna maupun laporan artikel. Member maupun admin bisa melihat informasi terkait akunnya seperti total artikel, total tayang, rata-rata rating artikel, total komentar, total komentar, total pengikut, dan total akun yang diikuti. Selain itu, member dan admin juga bisa melihat berbagai laporan terkait artikel seperti laporan perkembangan tayangan seluruh artikel, laporan perkembangan tayangan per artikel, laporan perkembangan rating artikel, laporan artikel terpopuler dan laporan artikel top rating. Member dan admin juga bisa mencari artikelnnya melalui kotak pencarian, mengakses notifikasi serta mengakses konfigurasi akunnya dengan memanfaatkan header di atas dashboard.



Gambar 6.17
Desain Dashboard

- Desain notifikasi

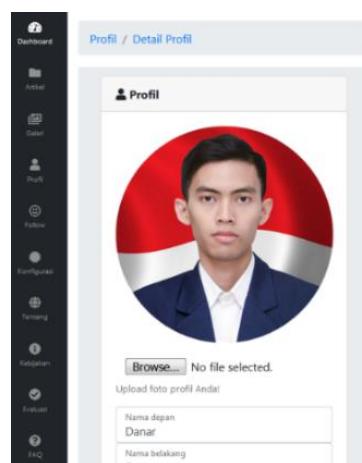
Desain notifikasi menampilkan berbagai pemberitahuan penting terkait akun member maupun admin. Pemberitahuan itu seperti adanya komentar baru, rating baru, dan artikel baru yang dipublikasi akun yang diikutinya. Member maupun admin bisa mengklik tandai notifikasi untuk menonaktifkan seluruh notifikasi telah dibaca. Berikut disajikan desain tampilan notifikasi:



Gambar 6.18
Desain Notifikasi

- Desain konfigurasi profil

Desain konfigurasi profil menampilkan halaman pengaturan akun. Pengguna bisa mengelola biodata akunnya seperti menganti password, foto profil, namanya, hingga mengubah biodata umum terkait profilnya. Tampilan ini hanya bisa diakses member maupun admin. Berikut disajikan desain tampilan konfigurasi profil:



Gambar 6.19
Desain Konfigurasi Profil

- Desain manajemen user

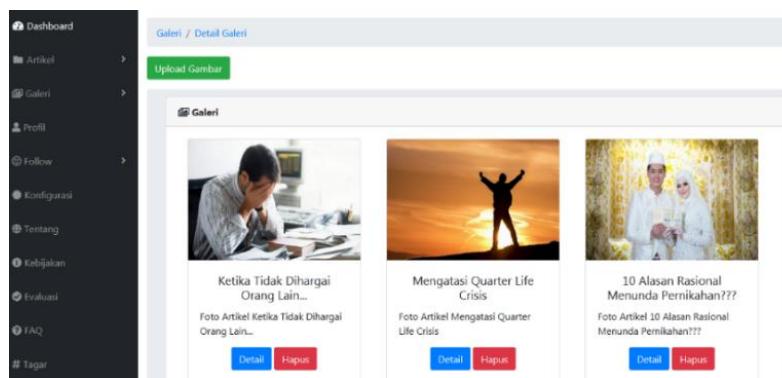
Desain manajemen user menampilkan daftar member dan admin yang terdaftar pada website. Halaman ini menampilkan biodata akun member maupun admin seperti nama, email dan grup aksesnya. Admin bisa memberikan atau mencabut hak administrator pada akun tertentu memanfaatkan manajemen user. Berikut disajikan desain manajemen user:

No	Nama Depan	Nama Belakang	E-mail	Grup	Action
1	Admin	istrator	admin@admin.com	admin members	<button>Edit</button> <button>Aktivasi</button>
2	Danar	Dono	donodanar35@gmail.com	admin members	<button>Edit</button> <button>Aktivasi</button>
3	Danar	Dono	danardono@stts.edu	members	<button>Edit</button> <button>Aktivasi</button>
4	dono	dono	dono@gmail.com	members	<button>Edit</button> <button>Aktivasi</button>
5	Eren	Yeager	eren@yeager.com	members	<button>Edit</button> <button>Aktivasi</button>
6	mikasa	ackerman	mikasa@ackerman	admin members	<button>Edit</button> <button>Aktivasi</button>

Gambar 6.20
Desain Manajemen User

- Desain manajemen galeri

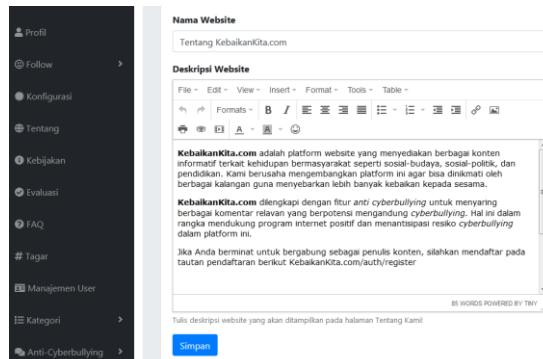
Desain manajemen galeri menampilkan daftar galeri yang dimiliki member. Member bisa mengupload galeri memanfaatkan manajemen galerinya. Selain itu, gambar artikel yang diupload pada manajemen artikel secara otomatis ditampilkan pada manajemen galeri. Member bisa melihat daftar galerinya, menyalin url galeri, dan menghapus galerinya.



Gambar 6.21
Desain Manajemen Galeri

- Desain konfigurasi tentang *website*

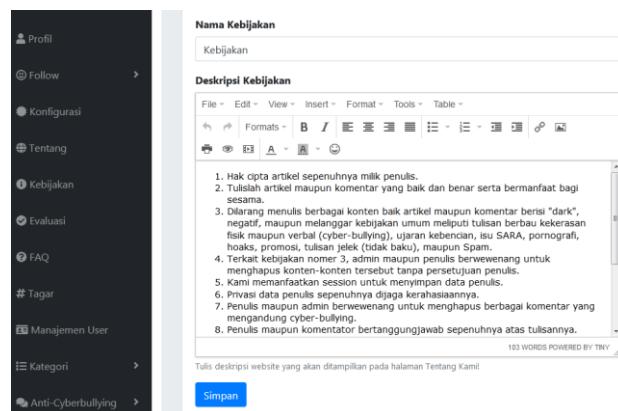
Desain konfigurasi tentang *website* menampilkan konfigurasi berhubungan dengan informasi umum *website*. Admin bisa melihat informasi umum *website* dan mengubahnya nama *website* maupun deskripsinya. Selain itu, admin juga bisa mengelola konten tentang website. Berikut disajikan desain tampilan konfigurasi tentang *website*:



Gambar 6.22
Desain Konfigurasi Tentang Website

- Desain konfigurasi kebijakan *website*

Desain konfigurasi kebijakan *website* menampilkan pengaturan tentang kebijakan website. Admin bisa melihat kebijakan *website*. Admin juga bisa mengubah kebijakan tersebut. Berikut disajikan desain tampilan konfigurasi kebijakan *website*:



Gambar 6.23
Desain Konfigurasi Kebijakan Website

- Desain manajemen FAQ

Desain manajemen FAQ menampilkan halaman pengelolaan pertanyaan dan jawaban yang sering ditanyakan pengunjung. Admin bisa membuat pertanyaan yang sering ditanyakan pengunjung dan menjawabnya secara rinci. Berikut disajikan desain tampilan manajemen FAQ:

The screenshot shows a sidebar menu on the left with options like Dashboard, Artikel, Galeri, Profil, Follow, Konfigurasi, Tentang, Kebijakan, Evaluasi, and FAQ. The main area is titled 'FAQ' and contains a green button 'Tambah Pertanyaan'. Below it is a table titled 'Sunting FAQ' with columns: No, Diperbarui, Pertanyaan, and Action. Two entries are listed:

No	Diperbarui	Pertanyaan	Action
1	2020-04-11	Apa itu cyber-bullying?	Hapus
2	2020-04-11	Apa itu anti-cyberbullying?	Hapus

At the bottom, it says 'Showing 1 to 2 of 2 entries' and has 'Previous' and 'Next' buttons.

Gambar 6.24
Desain Manajemen FAQ

- Desain manajemen kategori

Desain manajemen kategori menampilkan halaman pengelolaan kategori yang merupakan topik dari konten artikel. Halaman ini menampilkan daftar kategori artikel. Admin bisa membuat kategori baru, mengedit, dan menghapus kategori. Berikut disajikan desain tampilan manajemen kategori:

The sidebar menu is identical to the one in the FAQ section. The main area is titled 'Kategori / Detail Kategori' and contains a green button 'Tambah Data'. Below it is a table titled 'Daftar Kategori' with columns: No, Nama Kategori, Featured, and Action. Four entries are listed:

No	Nama Kategori	Featured	Action
1	Sosial-Budaya	Y	Edit Hapus
2	Politik	Y	Edit Hapus
3	Olahraga	Y	Edit Hapus
4	Pendidikan	N	Edit Hapus

At the bottom, it says 'Showing 1 to 4 of 4 entries' and has 'Previous' and 'Next' buttons.

Gambar 6.25
Desain Manajemen Kategori

- Desain manajemen tagar

Desain manajemen tagar menampilkan halaman tag yang sering diakses pengunjung. Admin bisa melihat berbagai tagar atau kata kunci yang sering diakses pengunjung. Admin juga bisa membuat, mengedit dan menghapus tagar yang ada. Berikut disajikan desain tampilan manajemen tagar:

No	Tagar	Diakses	Diperbarui	Action
1	#KesehatanMental	109	2020-04-07	<button>Edit</button> <button>Hapus</button>
2	#depresi	23	2020-04-10	<button>Edit</button> <button>Hapus</button>
3	#rasional	21	2020-09-19	<button>Edit</button> <button>Hapus</button>

Gambar 6.26
Desain Manajemen Tagar

- Desain manajemen *following-following*

Desain *following-following* menampilkan halaman manajemen *following-follower* yang berisi daftar akun yang mengikuti member maupun daftar akun yang diikuti member. Member bisa melihat daftar akun yang mengikutinya dan daftar akun yang diikuti. Member juga bisa berhenti untuk mengikuti akun yang diikutinya. Berikut disajikan desain tampilan manajemen *following-follower*:

No	Nama	Total Artikel	Action
1	mikasa ackerman	2	<button>Berhenti</button>

Gambar 6.27
Desain Manajemen Following

- Desain manajemen artikel

Desain manajemen artikel menampilkan halaman pengelolaan artikel. Member bisa melihat daftar artikelnya, membuat artikel baru, mengedit artikelnya maupun menghapus artikelnya. Manajemen artikel menampung dataset artikel dari berbagai sumber. Dalam hal ini, dalam setiap artikel menampung komentar-komentar. Berikut disajikan desain manajemen artikel:

No	Judul	Tanggal	Kategori	Dibaca	Rating	Komentar	Action
1	Jangan Tertipu, Tidak Mungkin yang Meninggal Bisa Dihubungi Kembali...	04:49:19 08-04-2020	Sosial-Budaya	23	0	0	Hapus
2	Yang baik tidak selalu dihargai, tetapi bukan berarti berhenti berbuat baik...	04:43:31 08-04-2020	Sosial-Budaya	10	0	0	Hapus
3	Mengapa ada orang yang jahat di dunia ini?	04:38:05 08-04-2020	Sosial-Budaya	2	0	0	Hapus
4	Kehidupan Sosial INTJ	04:25:34 08-04-2020	Sosial-Budaya	3	0	0	Hapus

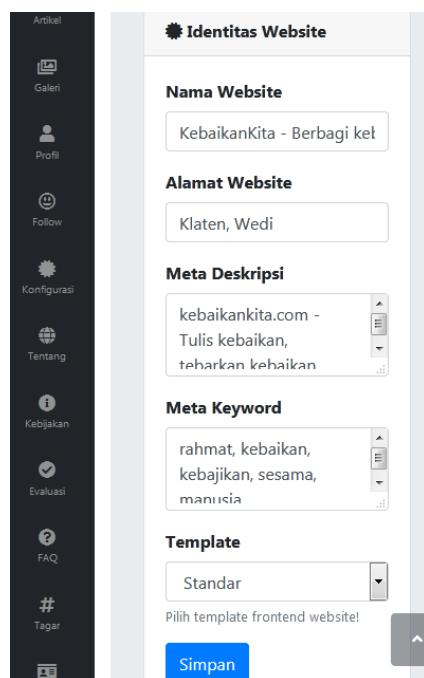
Gambar 6.28
Manajemen Artikel

Ketika member membuat atau mengedit artikel, admin akan diarahkan ke halaman pembuatan artikel. Member bisa mengisi form artikel seperti judul, kategori, kata kunci, foto ilustrasi artikel, deskripsi artikel, dan isi artikel. Kemudian menekan tombol simpan. Dengan demikian, member bisa membuat atau menyunting artikel. Berikut disajikan desain halaman pembuatan artikel:

Gambar 6.29
Desain Halaman Pembuatan Artikel

- Desain konfigurasi *website*

Desain konfigurasi *website* menampilkan halaman pengaturan umum *website*. Admin bisa memanfaatkan halaman ini untuk mengatur mengubah identitas website seperti nama *website*, alamat *website*, meta deskripsi, dan meta *keyword*. Selain itu, admin juga bisa mengubah tampilan *template front end*, setidaknya disediakan 2 tampilan template *front end* yakni standar dan meranda. Berikut disajikan desain tampilan konfigurasi *website*:



Gambar 6.30
Desain Konfigurasi Website

- Desain manajemen penyaring komentar *cyberbullying*

Desain manajemen penyaring komentar *cyberbullying* menampilkan halaman pengelolaan model penyaring komentar *cyberbullying*. Manajemen komentar *cyberbullying* terdiri atas beberapa bagian meliputi dataset artikel, dataset komentar, stopword list, kamus bahasa gaul, kelas kata, dan pengujian. Pada halaman manajemen dataset artikel, admin bisa menginput, mengedit dan menghapus artikel. Admin juga bisa melakukan *preprocessing* terhadap artikel. Berikut disajikan desain tampilan manajemen *dataset* artikel:

ID	Judul	Tanggal	Topik	Case Folding	Cleaning	Normalisasi	Stemming	Stop
1	Bocah SD di Bandung Tewas Dipatuk Ular Weling	23-01-2020	sosbud	tragedi ular kembali memakan korban. kali ini seorang bocah	tragedi ular kembali memakan korban kali ini seorang bocah	tragedi ular kembali memakan korban kali ini seorang bocah	tragedi ular kembali makan korban kali ini orang bocah laki laki	tragedi makar bocah patuk weling

Gambar 6.31
Desain Manajemen Dataset Artikel

Melalui desain manajemen *dataset* komentar, admin bisa menginput, mengedit, dan menghapus komentar pada artikel dataset. Admin juga bisa melakukan *preprocessing* terhadap komentar secara menyeluruh. *Preprocessing* ini meliputi *case folding*, *cleaning*, normalisasi bahasa gaul, *stemming*, *stopword* dan *tokenizing*. Berikut disajikan desain manajemen *dataset* komentar:

Komentar	Case Folding	Cleaning	Normalisasi	Stemming	Stopword	Tokenizing	Ngram	Bulky
Serem iih.	serem iih.	serem iih.	serem ya	serem ya	serem	serem	serem	N
hmmn	hmmn	hmmn	heran	heran	heran	heran	heran	N

Dari artikel yg gv bacanya kalo abis digigit ular weleg bikin ngantuk terus kalo korban tidur langsung meninggal. Para insomnia silahkan piara ni uler.

dari artikel yg gv bacanya kalo abis digigit ular weleg bikin ngantuk terus kalo korban tidur langsung meninggal. para insomnia silahkan piara ni uler.

dari artikel yg gv bacanya kalo abis digigit ular weleg bikin ngantuk terus kalo korban tidur langsung meninggal. para insomnia silahkan piara ni uler.

dari artikel yg gv bacanya kalo abis digigit ular weleg bikin ngantuk terus kalo korban tidur langsung meninggal. para insomnia silahkan piara ni uler.

dari artikel yg gv bacanya kalo abis digigit ular weleg bikin ngantuk terus kalo korban tidur langsung meninggal. para insomnia silahkan piara ni uler.

artikel bacaa habis gigit ular weleg bikin ngantuk terus korban tidur langsung tinggal para insomnia silahkan pelihara ular terus kornban hidar langsung tinggal insomnia silahkan pelihara ular.

artikel bacaa habis gigit ular weleg bikin ngantuk terus korban tidur langsung tinggal bikin antuk terus korban hidar langsung tinggal insomnia silahkan pelihara ular.

artikel bacaa habis gigit ular weleg bikin ngantuk terus korban tidur langsung tinggal bikin antuk terus korban hidar langsung tinggal insomnia silahkan pelihara ular.

Gambar 6.32
Manajemen Dataset Komentar

Desain *stopword* menampilkan halaman manajemen *stopword*. *Stopword* adalah daftar kata-kata yang akan dibuang selama proses *stopword* dalam *preprocessing*. Admin bisa melihat daftar *stopword*. Admin juga bisa melakukan penginputan, pengeditan dan penghapus terhadap kata *stopword*. Berikut disajikan desain tampilan manajemen *dataset stopword*:

ID	Stopword	Aktif	Action
1	ada	Y	<button>Ubah</button> <button>Hapus</button>
2	adalah	Y	<button>Ubah</button> <button>Hapus</button>

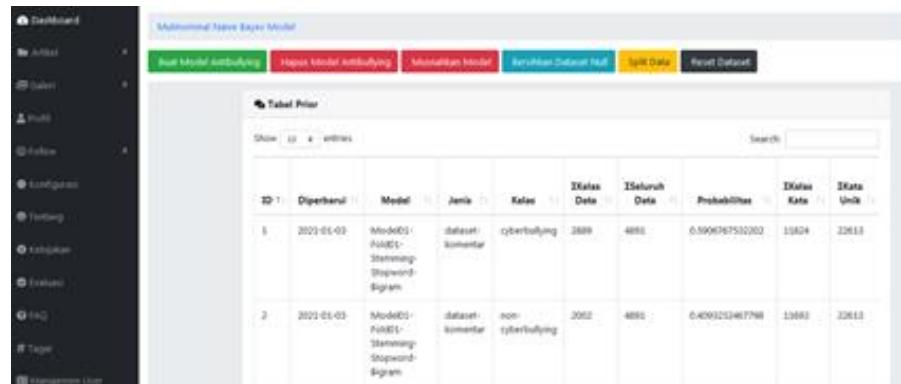
Gambar 6.33
Manajemen Dataset Stopword

Desain bahasa gaul menampilkan halaman manajemen bahasa gaul. Bahasa gaul ini merupakan kamus yang berisi daftar kata gaul dan terjemahan kata formalnya. Admin bisa menginput, mengedit, dan menghapus kata-kata gaul melalui halaman ini. Berikut disajikan desain tampilan manajemen dataset bahasa gaul:

ID	Kata Gaul	Kata Formal	Aktif	Action
1	nggak	tidak	Y	<button>Ubah</button> <button>Hapus</button>
2	engga	tidak	Y	<button>Ubah</button> <button>Hapus</button>

Gambar 6.34
Manajemen Dataset Bahasa Gaul

Desain kelas kata menampilkan halaman manajemen fitur kata sebagai model penyaring *cyberbullying*. Admin bisa membuat model *cyberbullying* baru melalui halaman ini. Admin juga bisa menghapus model tertentu. Admin juga bisa memusnahkan seluruh model. Admin juga bisa melakukan pembersihan terhadap *dataset* dan membagi *dataset* menjadi *dataset* latih dan *dataset* uji. Admin juga bisa melakukan pembuatan model menggunakan *k-fold cross validation*. Berikut disajikan desain tampilan model fitur kata:



Gambar 6.35
Desain Model Fitur Kata

Desain pengujian menampilkan halaman pengujian model *cyberbullying*. Admin bisa menguji model melalui halaman ini seperti mengetes akurasi, *error*, *specificity*, *recall*, *precision* dan *f-score* untuk menemukan model terbaik. Hasil pengujian tersebut selanjutnya ditampilkan dalam tabel. Berikut disajikan desain tampilan pengujian model:

Daftar Pengujian Antibully Model																	
Mode	Segment Awal	Segment Akhir	IBully	INon	IDataset	ITes	TP	FP	FN	TN	Acc%	Error%	Precision%	Recall%	Specificity%	F_Score%	Action
																	Search
Model03-Fold04-NoStemming-Stopword-Unigram	4951	6600	912	712	1624	1624	836	378	76	334	72.04	27.96	68.86	91.67	46.91	78.65	Hapus
Model03-Fold03-NoStemming-Stopword-Unigram	3301	4950	975	652	1627	1627	871	331	104	321	73.26	26.74	72.46	89.33	49.23	80.02	Hapus
Model01-Fold01-Stemming-Stopword-Bigram	1	1650	1053	593	1646	1646	936	417	117	176	67.25	32.44	69.18	88.89	29.68	77.81	Hapus

Gambar 6.36
Desain Pengujian Model

Pada gambar 6.37 ditunjukkan daftar komentar beserta kategorinya baik *cyberbullying* maupun *non-cyberbullying*. Daftar komentar ini terdapat dalam halaman artikel. Setiap komentar yang dikirim ke sistem ditampilkan pada halaman artikelnya tepat di bawah kolom komentarnya. Sistem juga menampilkan kategori komentarnya, sehingga pembaca bisa mengetahui komentar-komentar termasuk *cyberbullying* atau bukan. Berikut disajikan halaman daftar komentar beserta kategorinya pada gambar 6.37:

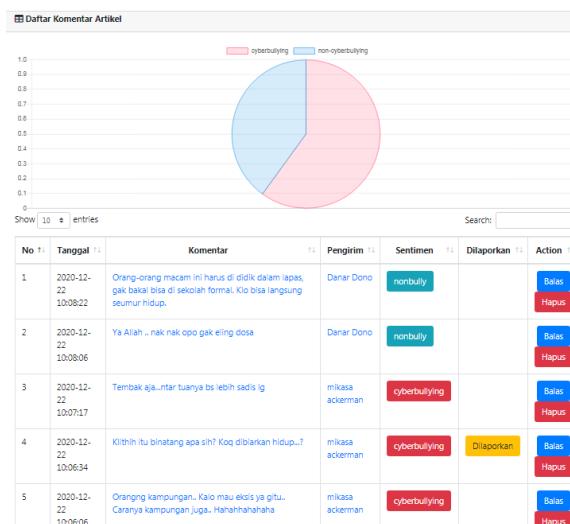
5 Tanggapan Pembaca:

mikasa ackerman | 2020-12-22 10:06:06
Orangng kampungan.. Kalo mau eksis ya gitu.. Caranya kampungan juga..
Hahahahahaha

Danar Dono | 2020-12-22 10:08:06
Ya Allah .. nak nak opo gak eling dosa

Gambar 6.37
Desain Daftar Komentar Beserta Kategorinya

Selain itu, terdapat pula halaman pengelolaan komentar pada sisi *back end website*. Sebagai ditunjukkan pada gambar 6.38, pengguna bisa melihat komentar-komentar yang dikirim pengguna lainnya terkait artikel yang ditulisnya. Pada halaman ini, penulis dapat mengelola komenatarnya lebih lanjut seperti membalas komentar, menghapus komentar, dan mengecek kembali kategori komentarnya. Daftar komentar sudah dilabeli kategorinya apakah *cyberbullying* atau *non-cyberbullying*. Berikut disajikan halaman pengelolaan komentar pada gambar 6.38:



Gambar 6.38
Desain Pengelolaan Komentar

Desain klasifikasi kategori komentar menampilkan halaman untuk menguji suatu komentar termasuk kategori *cyberbullying* atau kategori *non-cyberbullying*. Pada halaman tersebut terdapat kolom komentar dimana pengguna dapat memasukkan komentar untuk diuji kategorinya, lalu menekan proses. Sehingga sistem akan menampilkan hasil analisis klasifikasi kategori komentar beserta hasil perhitungannya. Pada halaman tersebut ditampilkan hasil *preprocessing*, bobot kategori *cyberbullying*, bobot kategori *non-cyberbullying*, dan hasil klasifikasi kategori komentarnya. Berikut disajikan halaman klasifikasi kategori komentar pada gambar 6.39:

Aspek	Keterangan
Komentar	dasar cebong busuk
Hasil preprocessing	dasar cebong busuk
Bobot Cyberbullying	1.1698069321823e-8
Bobot Non-cyberbullying	1.1734359610984e-11
Hasil klasifikasi kategori komentar	cyberbullying

Gambar 6.39
Desain Klasifikasi Kategori Komentar

6.4 Desain Prosedural

Pada sub bab ini dijelaskan tentang desain prosedural dari sistem baru. . Desain prosedural ini berguna untuk memudahkan dalam memahami alur kerja sistem untuk selanjutnya diterapkan dalam implementasi coding. Adapun desain proseduralnya meliputi proses *preprocessing* komentar, pembuatan model, pengujian model, hingga pengelolaan artikel. Selengkapnya sebagai berikut:

6.4.1 Desain Prosedural Case Folding

Desain prosedural *case folding* digunakan untuk melakukan proses mengubah huruf dalam data komentar menjadi huruf kecil. Komentar biasanya karakternya tidak sama, karena itu, perlu diseragamkan menjadi huruf keci. Sehingga datanya bisa seragam huruf kecil semua. Berikut dijelaskan algoritma proses case folding:

Algoritma 6.1 Case Folding

- 01: Masukkan data komentar
- 02: Lakukan penyeragaman data komentar menjadi huruf kecil
- 03: Kembalikan hasil penyeragaman data komentar

Proses *case folding* diawali dengan memasukan data komentar pada fungsi case folding. Kemudian data komentar diseragamkan menjadi huruf kecil semua. Terakhir kembalikan hasil *case folding*. Sehingga proses *case folding* selesai.

Desain prosedural ini akan digunakan pada *preprocessing* baik pelatihan data maupun pengujian data. Penyeragaman karakter penting untuk menyamakan data. Dengan desain prosedural ini teks komentar akan diseragamkan menjadi huruf kecil semua. Sehingga teks komentar seragam sebelum diproses ke tahapan *cleaning*.

6.4.2 Desain Prosedural Cleaning

Desain prosedural *cleaning* digunakan untuk melakukan proses pembersihan data komentar. Pembersihan data ini melibatkan penghapusan terhadap karakter-karakter yang tidak penting seperti *mention*, *hastag*, *link*, *emoticon*, titik (.), koma (,), titik dua (:), titik koma (;) dan karakter lainnya yang tak berguna. Dengan penghapusan karakter tidak penting tersebut, data menjadi lebih ringan dan mengurangi kompleksitas data. Berikut dijelaskan algoritma proses *cleaning*:

Algoritma 6.2 Cleaning

- 01: Masukkan data komentar
- 02: Lakukan penyeleksian karakter a - z dan angka 0 - 9 dari data komentar

Algoritma 6.2 (Lanjutan)

- 03: Ubah karakter tak penting di luar karakter a – z dan angka 0 – 9 menjadi karakter spasi
- 04: Kembalikan hasil cleaning

Proses *cleaning* diawali dengan memasukan data komentar pada fungsi *cleaning*. Kemudian data komentar diseleksi dengan hanya diambil karakter a – z dan angka 0 – 9. Lalu karakter tak penting di luar karakter a – z dan angka 0 – 9 diubah menjadi spasi. Terakhir kembalikan hasil *cleaning* sehingga proses *cleaning* selesai.

Desain prosedural ini digunakan pada *preprocessing* baik pelatihan data maupun pengujian data. Pembersihan data dari karakter tak penting membuat data komentar menjadi lebih kecil. Dengan desain prosedural ini teks komentar akan dibersihkan dari karakter tidak penting. Sehingga teks komentar menjadi bersih dari karakter tidak penting sebelum diproses ke tahapan normalisasi bahasa gaul.

6.4.3 Desain Prosedural Normalisasi Bahasa Gaul

Desain prosedural normalisasi bahasa gaul digunakan untuk melakukan penyeragaman kata-kata dari gaul menjadi kata-kata formal. Sehingga data komentar menjadi seragam berisi kata-kata formal. Normalisasi bahasa gaul diperlukan karena komentar biasa memiliki bahasa yang tidak baku sehingga perlu diseragamkan menjadi kata-kata baku. Berikut dijelaskan algoritma proses normalisasi bahasa gaul:

Algoritma 6.3 Normalisasi Bahasa Gaul

- 01: Masukkan data komentar
- 02: Jadikan data komentar menjadi array berdasarkan pemisah spasi
- 03: Periksalah setiap elemen array data komentar
- 04: Jika kata dalam elemen terdapat dalam kamus bahasa gaul
- 05: Ambil kata formal dari kamus bahasa gaul
- 06: Simpan kata formal ke elemen array data komentar
- 07: Gabungkan seluruh elemen array data komentar menjadi teks data komentar
- 08: Kembalikan hasil normalisasi bahasa gaul

Proses normalisasi bahasa gaul diawali dengan memasukan data komentar pada fungsi normalisasi bahasa gaul. Kemudian data komentar dijadikan array

dengan pemisah berdasarkan karakter spasi. Lalu periksa setiap elemen array data komentar. Jika kata dalam elemen tersebut terdapat dalam kamus bahasa gaul, maka ambil kata formal terkait dari kamus bahasa gaul dan simpan kata formal tersebut dalam elemen array terkait. Kemudian satukan seluruh elemen array menjadi teks data komentar dengan pemisah spasi. Terakhir kembalikan hasil normalisasi bahasa gaul sehingga proses normalisasi bahasa gaul selesai.

Desain prosedural ini diimplementasikan pada *preprocessing* baik pelatihan data maupun pengujian data. Segala kata yang tak baku akan diubah menjadi kata-kata yang formal. Dengan kata lain, dengan desain prosedural ini teks komentar akan diseragamkan menjadi kata formal. Sehingga teks komentar menjadi seragam berisi kata formal sebelum diproses ke tahapan *stemming*.

6.4.4 Desain Prosedural Stemming

Desain prosedural *stemming* digunakan untuk melakukan penyeragaman kata-kata dalam data komentar menjadi kata dasar. Sehingga data komentar menjadi seragam berisi kata-kata dasar. Hal ini juga bertujuan untuk mengurangi kompleksitas ukuran fitur kata. Berikut dijelaskan algoritma proses *stemming*:

Algoritma 6.4 Stemming

- 01: Masukkan data komentar
- 02: Jadikan data komentar menjadi array berdasarkan pemisah spasi
- 03: Periksalah setiap elemen array data komentar
- 04: Jika kata dalam elemen terdapat dalam kamus kata dasar
- 05: Proses pengecekan berakhir
- 06: Jika kata dasar tersebut tidak ada dalam kamus kata dasar
- 07: Lakukanlah penghapusan inflection suffixes
- 08: Jika kata terdapat dalam kamus kata dasar
- 09: Masukan kata dasar dalam elemen array terkait dan proses pengecekan berakhir
- 10: Lakukan penghapusan derivation suffixes
- 11: Jika kata terdapat dalam kamus kata dasar
- 12: Masukan kata dasar dalam elemen array terkait dan proses pengecekan berakhir
- 13: Lakukanlah penghapusan derivation prefix
- 14: Jika kata terdapat dalam kamus kata dasar
- 15: Masukan kata dasar dalam elemen array terkait dan proses pengecekan berakhir
- 16: Gabungkan seluruh elemen array data komentar menjadi teks data komentar
- 17: Kembalikan stemming dan proses stemming selesai

Proses stemming dimulai dengan memasukan data komentar ke fungsi stemming. Kemudian data komentar dijadikan array dengan pemisah berdasarkan karakter spasi. Lalu periksa setiap elemen array data komentar. Lakukan pengecekan pada kata dalam elemen array jika kata terdapat dalam kamus kata dasar maka proses pengecekan kata berakhir.

Namun jika kata tersebut tidak ada dalam kamus komentar, maka lakukanlah penghapusan *inflection suffixes* pada kata tersebut. Lakukan pengecekan apakah hasil kata dari penghapusan *inflection suffixes* terdapat dalam kamus kata dasar. Jika kata tersebut ada dalam kamus kata dasar maka masukan kata dasar tersebut ke elemen array terkait. Selanjutnya lakukanlah penghapusan *derivation suffixes* pada kata dalam elemen array. Lakukan pengecekan apakah hasil kata dari penghapusan *derivation suffixes* terdapat dalam kamus kata dasar. Jika kata tersebut ada dalam kamus kata dasar maka masukan kata dasar tersebut ke elemen array terkait.

Berikutnya lakukanlah penghapusan *derivation prefix* pada kata dalam elemen array. Lakukan pengecekan apakah hasil kata dari penghapusan *derivation prefix* terdapat dalam kamus kata dasar. Jika kata tersebut ada dalam kamus kata dasar maka masukan kata dasar tersebut ke elemen array terkait. Lalu gabungkan seluruh elemen array yang telah diperiksa dan di-stemming menjadi dokumen teks komentar dengan pemisah spasi. Terakhir kembalikan hasil *stemming*, sehingga proses *stemming* berakhir.

Desain prosedural ini diimplementasikan pada *preprocessing* baik pelatihan data maupun pengujian data. Dengan desain prosedural ini teks komentar akan diseragamkan menjadi kata dasarnya. Penggunaan *stemming* umumnya akan menyederhanakan ukuran fitur kata dalam model. Sehingga teks komentar menjadi seragam berisi kata dasar sebelum diproses ke tahapan *stopword*.

6.4.5 Desain Prosedural Stopword

Desain prosedural *stopword* digunakan untuk melakukan pembuangan kata-kata *stopword-list* yakni kata-kata yang sering muncul dan tidak memiliki arti. Dengan dibuangnya kata-kata tersebut, dokumen menjadi lebih disederhanakan.

Hal ini sekaligus mengurangi ukuran fitur kata dalam model. Berikut dijelaskan algoritma proses *stopword*:

Algoritma 6.5 Stopword

- 01: Masukkan data komentar
- 02: Jadikan data komentar menjadi array berdasarkan pemisah spasi
- 03: Periksa setiap elemen array data komentar
- 04: Jika kata dalam elemen terdapat dalam kamus *stopword-list*
- 05: Hapus elemen array
- 06: Gabungkan seluruh elemen array data komentar menjadi teks data komentar
- 07: Kembalikan hasil *stopword*

Proses *stopword* diawali dengan memasukan data komentar pada fungsi *stopword*. Kemudian data komentar dijadikan array dengan pemisah berdasarkan karakter spasi. Lalu periksa setiap elemen array data komentar. Jika kata dalam elemen tersebut terdapat dalam kamus *stopword-list*, maka hapus elemen array terkait. Kemudian gabungkan seluruh elemen array menjadi teks data komentar dengan pemisah spasi. Terakhir kembalikan hasil *stopword* sehingga proses *stopword* selesai.

Desain prosedural ini diimplementasikan pada *preprocessing* baik pelatihan data maupun pengujian data. Dengan desain prosedural ini teks komentar akan dibersihkan dari kata-kata yang tak bermakna seperti “yang”, “dan”, dan sebagainya. Dengan demikian diharapkan ukuran fitur kata menjadi lebih kecil. Sehingga teks komentar menjadi ringkas sebelum diproses ke tahapan *tokenizing*.

6.4.6 Desain Prosedural Tokenizing

Desain prosedural *tokenizing* digunakan untuk melakukan pemisahan kata-kata dalam data komentar menjadi daftar kata. Daftar kata tersebut selanjutnya akan diproses dalam pelatihan model. Dalam hal ini, proses *tokenizing* sekaligus menghasilkan *unigram*. Berikut dijelaskan algoritma proses *tokenizing*:

Algoritma 6.6 Tokenizing

- 01: Masukkan data komentar
- 02: Lakukan pemisahan data komentar berdasarkan karakter spasi menjadi array *data_komentar*
- 03: Gabungkan seluruh elemen array menggunakan pemisah

Algoritma 6.6 (Lanjutan)

04: Kembalikan hasil tokenizing

Proses *tokening* diawali dengan memasukan data komentar pada fungsi *tokening*. Kemudian data komentar dijadikan array dengan pemisah berdasarkan karakter spasi. Lalu seluruh elemen array disatukan kembali menggunakan pemisah
. Terakhir kembalikan hasil *tokenizing* sehingga proses *tokenizing* selesai.

Desain prosedural ini diimplementasikan pada *preprocessing* baik pelatihan data maupun pengujian data. Dengan desain prosedural ini teks komentar akan diubah menjadi daftar kata. Selanjutnya hasil *tokenizing* bisa langsung diproses oleh model jika memakai fitur *unigram*. Namun jika memakai fitur *bigram*, maka hasil *tokenizing* akan diproses pada tahapan *ngram*.

6.4.7 Desain Prosedural Bigram

Setelah melakukan *tokenizing*, proses selanjutnya ialah membuat *bigram*. Desain prosedural bigram digunakan untuk memproses komentar menjadi sekumpulan kata yang masing-masing data terdiri atas dua kata. *Bigram* berarti gabungan dua kata yang diambil dari teks komentar dengan mengerakkan satu kata maju ke depan hingga membentuk daftar kata *bigram*. Berikut dijelaskan algoritma proses bigram sebagai berikut:

Algoritma 6.7 Bigram

01: Masukkan data komentar
 02: Lakukan pemisahan data komentar berdasarkan karakter
 menjadi array kata_komentar
 03: Lakukan iterasi hingga sejumlah total kata_komentar
 04: Gabungkan kata dari array kata_komentar dengan komposisi kata_komentar berindex i digabung kata_komentar berindex i + 1 dengan pemisah tanda "_" lalu tampung pada array bigram
 05: Satukan data array bigram menjadi satu data dengan pemisah
 lalu kembalikan hasilnya.

Proses bigram diawali dengan masukan data komentar. Data komentar ini diambil dari hasil *tokenizing*. Lalu data komentar dijadikan array _kata komentar berdasarkan pemisah karakter
. Kemudian dilakukan iterasi hingga mencapai total kata_komentar dimana setiap iterasi kata_komentar dengan index i digabung

dengan kata_komentar berindex $i+1$ dengan disatukan berdasarkan pemisah tanda”_”, lalu hasilnya ditampung pada array bigram. Terakhir satukan data array bigram, lalu hasilnya dikembalikan sehingga proses pembuatan bigram berakhir.

Proses *bigram* dilakukan setelah data komentar mengalami *tokenizing*. Dalam hal ini, jika unigram dipilih, maka hasil N-gram sama dengan hasil *tokenizing*. Sementara jika menggunakan bigram, maka setiap token atau daftar kata terdiri atas masing-masing gabungan 2 kata. Selanjutnya hasil *bigram* bisa dipakai dalam pembuatan model.

6.4.8 Desain Prosedural Pembuatan Model

Desain prosedural pembuatan model digunakan untuk membuat pelatihan data hingga menghasilkan model *cyberbullying*. Model *cyberbullying* akan digunakan dalam mengklasifikasi komentar menjadi *cyberbullying* atau *non-cyberbullying*. Desain prosedural ini ditunjukkan untuk melakukan pembuatan model hingga membentuk fitur kata dan *prior* beserta probabilitasnya. Berikut dijelaskan algoritma proses pembuatan model:

Algoritma 6.8 Pembuatan Model

```

01: Hapus model fitur kata terdahulu
02: Hapus model prior terdahulu
03: Reset dataset komentar
04: Jika dataset komenter telah direset
05:     Buat model fitur kata
06:     Simpan model fitur kata ke database
07:     Jika model fitur kata berhasil dibuat
08:         Buat model prior
09:         Simpan model prior ke database
10:         Jika model prior berhasil dibuat
11:             Hitung probabilitas setiap fitur kata
12:             Simpan probabilitas setiap fitur kata
13:             Jika probabilitas fitur kata berhasil dibuat
14:                 Kembalikan pesan “model berhasil dibuat”
15:             Jika model prior gagal dibuat
16:                 Kembalikan pesan “model prior gagal dibuat”
17:             Jika model fitur kata gagal dibuat
18:                 Kembalikan pesan “fitur kata gagal dibuat”
19:             Jika dataset komentar gagal direset
20:                 Kembalikan pesan “dataset komentar gagal direset”
```

Proses pembuatan model diawali dengan penghapusan model fitur kata sebelumnya. Kemudian dilakukan penghapusan pada model prior terdahulu. Dilanjutkan peresetan dataset komentar. Jika dataset komentar berhasil direset maka buatlah model fitur kata baru kemudian simpan model fitur kata ke database. Jika dataset komentar gagal dibuat maka kembalikan pesan error berupa “dataset komentar gagal direset”.

Setelah berhasil dibuat model fitur kata, selanjutnya buatlah model prior, lalu simpan model prior ke database. Jika model prior berhasil dibuat, maka hitunglah probabilitas setiap fitur kata, kemudian simpan hasil probabilitas setiap fitur kata ke database. Jika model prior gagal dibuat, maka kembalikan pesan error berupa “model prior gagal dibuat”. Jika probabilitas fitur kata berhasil dibuat maka kembalikan pesan sukses berupa “model berhasil dibuat”. Dengan demikian proses pembuatan model *cyberbullying* selesai.

6.4.9 Desain Prosedural Kalkulasi Kategori Komentar

Desain prosedural kalkulasi kategori digunakan untuk menghitung dan menentukan kategori komentar menjadi bernilai *cyberbullying* atau *non-cyberbullying*. Dari desain prosedural ini akan dihasilkan penentuan kategori komentar. Kalkulasi komentar dapat terjadi dengan membandingkan nilai Vmap antar kategori komentar. Hanya kategori komentar dengan nilai Vmap tertinggi menjadi hasil dari klasifikasi komentarn. Berikut dijelaskan algoritma proses kalkulasi kategori komentar:

Algoritma 6.9 Kalkulasi Kategori Komentar

- 01: Masukan data komentar
- 02: Lakukan case folding pada data komentar
- 03: Lakukan cleaning pada hasil case folding
- 04: Lakukan normalisasi bahasa gaul pada hasil cleaning
- 05: Lakukan stemming pada hasil normalisasi bahasa gaul
- 06: Lakukan stopword pada hasil stemming
- 07: Lakukan tokenizing pada hasil stopword
- 08: Lakukan pengecekan pada setiap kata hasil tokenizing
- 09: Cek apakah kata terdapat dalam model
- 10: Jika kata terdapat dalam model
- 11: Hitung nilai probabilitas cyberbullying
- 12: Hitung nilai probabilitas noncyberbullying
- 13: Jika kata tidak ada dalam model

Algoritma 6.9 (Lanjutan)

- 14: Hitung nilai probabilitas noncyberbullying dengan mengalikan bobot noncyberbullying dan $1 / (\text{total fitur kata unik pada kelas noncyberbullying} + \text{total kata unik dari seluruh kelas})$
- 15: Hitung nilai probabilitas cyberbullying dengan mengalikan bobot cyberbullying dan $1 / (\text{total fitur kata unik pada kelas cyberbullying} + \text{total kata unik dari seluruh kelas})$
- 16: Ambil nilai prior cyberbullying dari model
- 17: Ambil nilai prior noncyberbullying dari model
- 18: Hitung total bobot kategori cyberbullying dengan mengalikan nilai prior cyberbullying dan nilai probabilitas cyberbullying
- 19: Hitung total bobot kategori noncyberbullying dengan mengalikan nilai prior noncyberbullying dan nilai probabilitas noncyberbullying
- 20: Jika nilai total bobot cyberbullying > nilai total bobot noncyberbullying
- 21: Kembalikan pesan kategori “cyberbullying”
- 22: Jika nilai total bobot cyberbullying < nilai total bobot noncyberbullying
- 23: Kembalikan pesan kategori “non-cyberbullying”

Proses dimulai dengan memasukan data komentar. Kemudian data komentar di-*preprocessing* secara berurutan meliputi *case folding*, *cleaning*, normalisasi bahasa gaul, *stemming*, *stopword*, dan *tokenizing*. Hasil *preprocessing* berupa token kata kemudian dicek satu per satu. Cek apakah kata ada dalam model. Jika kata ada dalam model maka hitung nilai probabilitas *cyberbullying* dan hitung nilai probabilitas *non-cyberbullying*. Jika kata tidak ada dalam model maka hitung nilai probabilitas *non-cyberbullying* dengan mengalikan bobot *non-cyberbullying* dan $1 / (\text{total fitur kata unik pada kelas non-cyberbullying} + \text{total kata unik dari seluruh kelas})$. Hitung juga nilai probabilitas *cyberbullying* dengan mengalikan bobot *cyberbullying* dan $1 / (\text{total fitur kata unik pada kelas cyberbullying} + \text{total kata unik dari seluruh kelas})$.

Selanjutnya ambil nilai *prior cyberbullying* dan nilai *prior non-cyberbullying*. Lalu hitung total bobot *cyberbullying* dengan mengalikan nilai probabilitas *cyberbullying* dan nilai *prior cyberbullying*. Hitung juga total bobot *non-cyberbullying* dengan mengalikan nilai probabilitas *non-cyberbullying* dan nilai *prior non-cyberbullying*. Jika nilai total bobot *cyberbullying* lebih besar daripada total bobot *non-cyberbullying* maka kembalikan pesan “cyberbullying” sehingga proses kalkulasi kategori selesai. Atau jika nilai total bobot *cyberbullying*

lebih kecil daripada total bobot *non-cyberbullying* maka kembalikan pesan “*non-cyberbullying*” sehingga proses kalkulasi kategori selesai.

6.4.10 Desain Prosedural Pengujian Model

Desain prosedural pengujian model digunakan untuk menghitung nilai *accuracy*, *error*, *precision*, *recall*, *specificity*, dan *f-score*. Dalam hal ini ukuran *accuracy* dan *error* menjadi ukuran utama dalam menentukan model terbaik. Dari hasil pengujian tersebut selanjutnya dapat digunakan untuk menentukan model terbaik. Berikut dijelaskan algoritma proses pengujian model:

Algoritma 6.10 Pengujian Model

```

01: Masukan dataset uji
02: Lakukan kalkulasi kategori pada setiap data uji
03:     Hitung nilai fp
04:     Hitung nilai tp
05:     Hitung nilai fn
06:     Hitung nilai tn
07:     Hitung nilai accuracy dengan rumus  $(tp+tn) / (tp+fn+fp+tn)$ 
08:     Hitung nilai error dengan rumus  $(fp+fn) / (tp+fn+fp+tn)$ 
09:     Hitung nilai precision dengan rumus  $tp / (tp+fp)$ 
10:    Hitung nilai recall dengan rumus  $tp / (tp+fn)$ 
11:    Hitung nilai specificity dengan rumus  $tn / (fp+tn)$ 
12:    Hitung nilai fscore dengan rumus
         $(2 * precision * recall) / (precision + recall)$ 
13:    Kembalikan nilai accuracy, error, precision, recall,
        specificity, fscore

```

Proses dimulai dengan memasukan dataset uji pada fungsi pengujian model. Kemudian dataset uji dikalkukasi kategorinya satu per satu. Lalu dihitung nilai fp, tp, fn, dan tn. Berikutnya hitung nilai *accuracy* dengan rumus $(tp+tn) / (tp+fn+fp+tn)$.

Selanjutnya hitung nilai *error* dengan rumus $(fp+fn) / (tp+fn+fp+tn)$. Hitung nilai *precision* dengan rumus $tp / (tp+fp)$. Hitung nilai *recall* dengan rumus $tp / (tp+fn)$. Hitung nilai *specificity* dengan rumus $tn / (fp+tn)$. Hitung nilai *f-score* dengan rumus $(2 * precision * recall) / (precision + recall)$. Terakhir kembalikan nilai *accuracy*, *error*, *precision*, *recall*, *specificity*, dan *f-score*, sehingga proses pengujian model selesai.

6.4.11 Desain Prosedural Pembuatan Artikel

Desain prosedural pembuatan artikel berfungsi sebagai prosedur dalam penulisan artikel hingga dipublikasikan di halaman website. Desain prosedural ini akan aktif tatkala member menekan tombol simpan saat pembuatan artikel. Artikel akan disimpan di *database*. Berikut dijelaskan algoritma proses pembuatan artikel:

Algoritma 6.11 Pembuatan Artikel

- 01: Input data artikel ke form artikel
- 02: Upload gambar artikel ke form upload gambar
- 03: Tekan tombol simpan
- 04: Validasi form artikel
- 05: Jika validasi form artikel gagal
 - 06: Kembalikan pesan error
- 07: Jika validasi form artikel sukses
 - 08: Simpan data artikel ke database
 - 09: Periksa status upload gambar
 - 10: Jika gambar artikel terupload
 - 11: Simpan gambar ke directory website
 - 12: Gunakan gambar upload sebagai headline artikel
 - 13: Jika gambar artikel tidak terupload
 - 14: Gunakan gambar bawaan website sebagai headline artikel
 - 15: Terbitkan artikel ke halaman home website
 - 16: Periksa apakah penulis mempunyai follower
 - 17: Jika penulis mempunyai follower
 - 18: Kirim notifikasi artikel baru kepada follower
 - 19: Kembalikan pesan sukses artikel telah diterbitkan

Proses dimulai dengan menginput data artikel ke form artikel. Lalu pengupload gambar artikel ke form upload gambar. Kemudian user menekan tombol simpan pada halaman pembuatan artikel. Sistem akan memvalidasi data artikel pada form artikel. Jika validasi form artikel gagal, maka sistem akan mengembalikan pesan error. Jika validasi form artikel berhasil, maka sistem akan menyimpan data artikel ke database.

Kemudian sistem memeriksa gambar diupload atau tidak. Jika gambar artikel berhasil diupload, maka sistem akan menyimpan gambar artikel ke *directory* website. Lalu sistem akan menggunakan gambar sebagai headline artikel. Jika gambar tidak terupload, maka sistem akan menggunakan gambar bawaan sebagai headline artikel. Berikutnya sistem akan menerbitkan artikel baru ke halaman home website. Lalu sistem akan memeriksa apakah penulis mempunyai *follower* atau tidak. Jika penulis mempunyai *follower* maka sistem akan mengirimkan notifikasi

artikel baru kepada *follower*. Terakhir, sistem akan mengembalikan pesan sukses bahwa artikel berhasil diterbitkan, sehingga proses pembuatan artikel selesai.

6.4.12 Desain Prosedural Pengeditan Artikel

Desain prosedural pengeditan artikel berfungsi sebagai prosedur dalam melaksanakan pengeditan suatu artikel hingga artikel diperbarui. Desain prosedural ini akan aktif tatkala member menekan tombol simpan saat mengedit artikel. Desain prosedural ini membantu dalam melaksanakan pengeditan artikel. Berikut dijelaskan algoritma proses pengeditan artikel:

Algoritma 6.12 Pengeditan Artikel

- 01: Pilih artikel yang akan diedit
- 02: Tampilkan halaman detail pengeditan artikel
- 03: Ubah data artikel ke form pengeditan artikel
- 04: Upload gambar artikel ke form upload gambar
- 05: Tekan tombol simpan
- 06: Validasi form artikel
- 07: Jika validasi form pengeditan artikel gagal
- 08: Kembalikan pesan error
- 09: Jika validasi form pengeditan artikel sukses
- 10: Simpan perubahan data artikel ke database
- 11: Periksa status upload gambar
- 12: Jika gambar artikel terupload
- 13: Simpan gambar ke directory website
- 14: Gunakan gambar upload sebagai headline artikel
- 15: Jika gambar artikel tidak terupload
- 16: Gunakan gambar bawaan website sebagai headline artikel
- 17: Kembalikan pesan sukses artikel telah diedit

Proses dimulai dengan memilih artikel yang akan diedit. Kemudian sistem menampilkan halaman detail pengeditan artikel. Lalu user menginput perubahan data artikel ke form pengeditan artikel. Lalu pengupload gambar artikel ke form upload gambar. Kemudian user menekan tombol simpan pada halaman pengeditan artikel. Sistem akan memvalidasi data artikel pada form pengeditan artikel. Jika validasi form pengeditan artikel gagal, maka sistem akan mengembalikan pesan error. Jika validasi form pengeditan artikel berhasil, maka sistem akan menyimpan perubahan data artikel ke database.

Kemudian sistem memeriksa gambar diupload atau tidak. Jika gambar artikel berhasil diupload, maka sistem akan menyimpan gambar artikel ke *directory*

website. Lalu sistem akan menggunakan gambar sebagai headline artikel. Jika gambar tidak terupload, maka sistem akan menggunakan gambar bawaan sebagai headline artikel. Terakhir, sistem akan mengembalikan pesan sukses bahwa artikel berhasil diedit, sehingga proses pengeditan artikel selesai.

6.4.13 Desain Prosedural Penghapusan Artikel

Desain prosedural penghapusan artikel digunakan sebagai prosedur dalam menghapus suatu artikel. Adakalanya penulis tidak menghendaki artikelnya dipublikasi lebih lama. Karena itu, artikel terkadang perlu dihapus dari tampilan website sehingga tak bisa diakses member maupun pembaca. Berikut dijelaskan algoritma proses penghapusan artikel:

Algoritma 6.13 Penghapusan Artikel

- 01: Pilih artikel yang akan dihapus
- 02: Tampilkan halaman konfirmasi penghapusan artikel
- 03: Jika ditekan tombol batal
- 04: Batalkan penghapusan artikel
- 05: Jika ditekan tombol hapus
- 06: Nonaktifkan artikel yang dipilih dari halaman website
- 07: Kembalikan pesan sukses artikel telah dihapus

Proses dimulai dengan memilih artikel yang akan dihapus. Kemudian sistem menampilkan halaman konfirmasi penghapusan artikel. Jika user menekan tombol batal, maka penghapusan batal dilaksanakan. Sebaliknya jika ditekan tombol hapus, maka proses akan dilanjutkan pada nonaktifkan data artikel.

Jika user menekan tombol hapus, maka sistem akan menonaktifkan artikel yang dipilih dari halaman website. Artikel tidak dapat diakses pembaca maupun penulisnya. Lalu sistem mengembalikan pesan sukses telah menghapus artikel. Sehingga proses penghapusan artikel selesai.

6.4.14 Desain Prosedural Penguploadan Galeri

Desain prosedural penguploadan galeri digunakan sebagai prosedur dalam mengupload galeri ke penyimpanan galeri member. Desain prosedural ini akan aktif tatkala user mengupload galeri ke *website*. Galeri-galeri tersebut selanjutnya dapat

digunakan member sebagai ilustrasi dalam artikelnya. Berikut dijelaskan algoritma proses penguploadan galeri:

Algoritma 6.14 Penguploadan Galeri

- 01: Pilih gambar yang diupload
- 02: Tekan tombol simpan
- 03: Mengupload gambar ke website
- 04: Jika gambar gagal diupload
 - 05: Kembalikan pesan error gambar gagal terupload
- 06: Jika gambar berhasil diupload
 - 07: Periksa apakah nama gambar telah ada di directory website
 - 08: Jika nama gambar telah ada dalam directory website
 - 09: Kembalikan pesan error gambar telah ada
 - 10: Jika nama gambar belum ada dalam directory website
 - 11: Periksa ukuran gambar
 - 12: Jika ukuran gambar melebihi 2MB
 - 13: Kembalikan pesan error gambar melebihi 2MB
 - 14: Jika ukuran gambar tidak lebih dari 2MB
 - 15: Simpan gambar ke directory website
 - 16: Kembalikan pesan sukses galeri berhasil diupload

Proses dimulai dengan memilih gambar yang akan diupload. Kemudian user menekan tombol simpan. Lalu sistem mengupload gambar ke website. Jika proses upload gagal, maka sistem akan mengembalikan pesan error bahwa gambar gagal diupload. Jika gambar berhasil diupload, maka sistem akan memeriksa apakah nama gambar telah tersedia di directory website atau tidak.

Jika nama gambar telah tersedia, maka sistem mengembalikan pesan error bahwa gambar sudah ada. Jika nama gambar belum tersedia, maka sistem akan memeriksa ukuran gambar. Jika ukuran gambar melebih 2MB, maka sistem akan mengembalikan pesan error bahwa gambar melebihi 2MB. Jika gambar tidak lebih dari 2MB maka sistem akan menyimpan gambar ke directory website. Terakhir sistem mengembalikan pesan sukses bahwa galeri berhasil diupload sehingga proses upload galeri berhasil.

6.4.15 Desain Prosedural Follow-Unfollow Account

Desain prosedural mengikuti digunakan untuk mengikuti atau berhenti mengikuti suatu akun. Desain prosedural ini akan aktif tatkala user menekan tombol *following* pada halaman profil penulis. User bisa mengikuti akun untuk

mendapatkan kabar terbaru. User juga berhenti dari mengikuti akun. Berikut dijelaskan algoritma proses *follow-unfollow account*:

Algoritma 6.15 Follow-Unfollow Account

```

01: Menekan tombol follow account
02: Periksa apakah user telah login atau belum
03: Jika user belum login
04:     Kembalikan pesan error belum login
05: Jika user telah login
06:     Periksa apakah account telah tersedia di database member
07:     Jika account belum tersedia
08:         Tambahkan data account yang diikuti
09:         Simpan data account ke database member
10:         Kembalikan pesan sukses berhasil mengikuti
11:     Jika account telah tersedia
12:         Periksa status account
13:         Jika status account diikuti
14:             Ubah status account menjadi berhenti ikuti
15:             Simpan perubahan data account ke database member
16:             Kembalikan pesan sukses berhasil mengikuti
17:         Jika status account berhenti
18:             Ubah status account diikuti
19:             Simpan perubahan data account ke database member
20:             Kembalikan pesan sukses berhasil diikuti kembali

```

Proses dimulai ketika user menekan tombol *follow account* pada halaman profil member. Kemudian sistem memeriksa apakah user telah login atau belum. Jika user belum login, maka sistem akan mengembalikan pesan error bahwa user belum login. Jika user telah login, maka sistem akan memeriksa apakah *account* telah tersedia di database member. Jika *account* belum tersedia di database member, maka tambahkan data *account* yang diikuti ke database lalu simpan data *account* tersebut ke database member. Kemudian kembalikan pesan sukses bahwa berhasil mengikuti.

Jika *account* telah tersedia, maka sistem memeriksa status *account*. Jika status *account* diikuti, maka sistem mengubah status *account* menjadi berhenti diikuti. Lalu sistem menyimpan perubahan data ke database member. Kemudian sistem mengembalikan pesan sukses bahwa berhenti mengikuti. Jika status *account* berhenti, maka sistem mengubah status *account* diikuti. Lalu menyimpan perubahan data ke database member. Berikutnya mengembalikan pesan sukses

bahwa berhasil diikuti kembali. Dengan demikian proses *follow-unfollow account* selesai.

BAB VII

IMPLEMENTASI

Pada Bab VII dijelaskan tentang implementasi desain prosedural dalam coding. Dalam implementasi ini menggunakan bahasa PHP dengan memanfaatkan *framework* Codeigniter. Adapun penjelasannya meliputi segmen program penyaring komentar *cyberbullying* dan segmen program sistem manajemen konten. Segmen program penyaring komentar menyajikan beberapa segmen program dari fitur-fitur pengembangan sistem penyaring komentar. Sedangkan segmen program sistem manajemen konten menyajikan beberapa contoh segmen program dari fitur-fitur manajemen konten *blog*. Penjelasan selengkapnya sebagai berikut:

7.1 Segmen Program Penyaring Komentar Cyberbullying

Pada sub bab ini dijelaskan tentang beberapa segmen program penyaring komentar *cyberbullying*. Segmen program penyaring komentar menyajikan beberapa segmen program dari fitur-fitur pengembangan sistem penyaring komentar. Adapun penjelasannya meliputi segmen case *folding*, *cleaning*, normalisasi bahasa *gaul*, *stemming*, *stopword*, *tokenizing*, *ngram*, *preprocessing* komentar, modal penyaring *cyberbullying*, hingga kalkulasi kategori komentar. Berikut penjelasan selengkapnya:

7.1.1 Segmen Program Case Folding

Segmen program *case folding* berguna untuk menyeragamkan teks komentar. Teks komentar seringkali mengandung beragam variasi teks entah huruf kapital maupun huruf kecil. Teks komentar yang diproses dalam *preprocessing* terlebih dahulu melalui *case folding* untuk menyeragamkan teks komentar menjadi huruf kecil (*lowercase*). Berikut ini dijelaskan tentang segmen program *case folding*:

Segmen Program 7.1 Case Folding

```
01: function case_folding($komen) {
02:     return strtolower($komen);
03: }
```

Pada segmen program di atas, komentar dimasukan dalam parameter \$komen. Lalu fungsi akan mengembalikannya menjadi *lowercase*. Sehingga hasil kembalinya membuat komentar menjadi seragam huruf kecil. Selanjutnya, hasil case folding akan diproses ke tahap *cleaning*.

Segmen program ini diimplementasikan pada *preprocessing* baik pelatihan data maupun pengujian data. Dengan segmen ini teks komentar akan diseragamkan menjadi huruf kecil semua. Sehingga teks komentar menjadi seragam. Setelah itu, baru komentar akan diproses ke tahapan *cleaning*.

7.1.2 Segmen Program Cleaning

Segmen program *cleaning* berguna untuk menghilangkan karakter tak penting seperti mention (@), hastag (#), titik (.), koma (,), titik dua (:), titik koma (;), kurung kurawal pembuka (<), kurung kurawal penutup (>), dan karakter tak penting lainnya. Dengan segmen program ini, komentar akan dibersihkan dari karakter-karakter yang tidak penting. Berikut ini dijelaskan tentang potongan segmen program *cleaning*:

Segmen Program 7.2 Cleaning

```
01: public function cleansing($komen) {
02:     $komen = preg_replace("/[^a-zA-Z0-9]/", " ", $komen);
03:     $komen = str_replace("/", " ", $komen);
04:     return $komen;
05: }
```

Pada baris 1, teks komentar masuk ke parameter \$komen. Pada baris 2 , parameter \$komen diproses menggunakan *regular expression preg_replace()* untuk mencari kata/ karakter yang sesuai dengan pola *regex* [^a-zA-Z0-9]. Pola *regex* tersebut hanya mengizinkan karakter a – z, A – Z, dan 0 – 9 untuk diproses lebih lanjut. Lalu hasilnya kembali menggunakan *regular expression str_replace()*

untuk menganti karakter (/) menjadi spasi. Sehingga pada baris 4, hasilnya dikembalikan untuk selanjutnya diproses menuju normalisasi bahasa gaul.

Segmen program ini diimplementasikan pada *preprocessing* baik pelatihan data maupun pengujian data. Segmen program ini ditempatkan pada file model dalam *framework* Codeigneter. Dengan segmen ini teks komentar akan dibersihkan dari karakter tidak penting. Sehingga teks komentar menjadi bersih dari karakter tidak penting sebelum diproses ke tahapan normalisasi bahasa gaul.

7.1.3 Segmen Program Normalisasi Bahasa Gaul

Segmen program normalisasi bahasa gaul berguna untuk menyeragamkan kata-kata dari kata gaul (tak baku) menjadi kata-kata formal. Dalam komentar biasanya terdapat kata-kata gaul. Untuk menyeragamkannya dilakukan normalisasi dengan menganti kata-kata gaul menjadi kata-kata formal sesuai kamus bahasa Indonesia. Berikut ini dijelaskan tentang segmen program normalisasi:

Segmen Program 7.3 Normalisasi Bahasa Gaul

```

01: public function normalisasi_bahasagaul($komen) {
02:     $gabungkan_kata = '';
03:     $cek = preg_split('/ /', $komen);
04:     $count = count($cek);
05:     for ($i=0; $i<$count; $i++) {
06:         $cek[$i] = $this->ganti_katagaul($cek[$i]);
07:     }
08:     $gabungkan_kata = implode(' ', $cek);
09:     return $gabungkan_kata;
10: }
```

Pada baris 3 teks komentar yang sebelumnya dimasukan ke paramter \$komen diproses menjadi array menggunakan *preg_split()* lalu hasilnya dimasukan ke variabel \$cek. Para baris 4 hitung total data arrat \$cek lalu masukan ke variabel \$count. Pada baris 5 – 7, lakukan iterasi untuk mengecek setiap data dari arrat \$cek menggunakan fungsi ganti_katagaul. Jika data tersebut mengandung kata gaul maka ganti menjadi kata formal. Lakukan proses iterasi hingga mencapai jumlah \$count. Pada baris 8 – 9 , gabungkan data dalam array \$cek, lalu masukan ke variabel \$gabungkan_kata. Kemudian hasilnya dikembalikan untuk selanjutnya diproses ke tahap stemming.

Segmen program ini diimplementasikan pada *preprocessing* baik pelatihan data maupun pengujian data. Segmen program ini ditempatkan pada file model dalam *framework* Codeigneter. Dengan segmen ini teks komentar akan diseragamkan menjadi kata formal. Sehingga teks komentar menjadi seragam berisi kata formal sebelum diproses ke tahapan *stemming*.

7.1.4 Segmen Program Stemming

Segmen program stemming berguna untuk mengubah kata-kata komentar menjadi kata-kata dasar. Stemming bertujuan menyeragamkan kata-kata dan mengurangi kompleksitas model. Dalam hal ini, segmen program *stemming* mengimplementasikan algoritma Nazief dan Adriani. Berikut ini dijelaskan tentang segmen program stemming:

Segmen Program 7.4 Stemming

```

01: function stemming($kata){
02:     $cekKata = $this->cekKamus($kata);
03:     if ($cekKata == true) {
04:         return $kata;
05:     } else {
06:         $kata = $this->Del_Inflection_Suffixes($kata);
07:         if ($this->cekKamus($kata)) {
08:             return $kata;
09:         }
10:         $kata = $this->Del_Derivation_Suffixes($kata);
11:         if ($this->cekKamus($kata)) {
12:             return $kata;
13:         }
14:         $kata = $this->Del_Derivation_Prefix($kata);
15:         if ($this->cekKamus($kata)) {
16:             return $kata;
17:         }
18:     }
19:     return $kata;
20: }
```

Pada baris 1 – 6, segmen program dimulai dengan mengambil data dari parameter \$kata. Data kata yang berada dalam parameter \$kata diproses menggunakan fungsi cekKamus() untuk memeriksa apakah data kata terdapat dalam kamus atau tidak. Jika kata tersebut ada dalam kamus, maka fungsi akan

mengembalikan kata stemming. Jika tidak tersedia di kamus, maka kata tersebut akan diproses ke baris selanjutnya.

Pada baris 7 – 10, kata diperiksa dan dihapus suffix-nya menggunakan fungsi `Del_Inflection_Suffixes()`. Proses ini melibatkan penghapusan akhiran seperti -ku, -mu, -kah, dsb. Hasilnya kemudian ditampung ke variabel `$kata`. Lalu kata diperiksa kembali fungsi `cekKamus()` untuk memeriksa apakah kata tersebut tersedia di kamus atau tidak. Jika kata tersebut tersedia di kamus, maka fungsi akan mengembalikan hasil stemming kata. Jika tidak tersedia di kamus, maka kata tersebut akan diproses ke baris selanjutnya.

Pada baris 11 – 14, kata diperiksa kembali derivasi suffiksnya menggunakan melalui fungsi `Del_Derivation_Suffixes()`. Proses ini melibatkan penghapusan akhiran "-i", "-an" atau "-kan". Kemudian hasilnya ditampung ke variabel `$kata`. Kemudian kata diperiksa kembali fungsi `cekKamus()` untuk memeriksa apakah kata tersebut tersedia di kamus atau tidak. Jika kata tersebut tersedia di kamus, maka fungsi akan mengembalikan hasil stemming kata. Jika tidak tersedia di kamus, maka kata tersebut akan diproses ke baris selanjutnya.

Pada baris 15 – 20, kata diperiksa derivasi prefiksnya menggunakan fungsi `Del_Derivation_Prefix()`. Proses ini melibatkan penghapusan awalan seperti "di-", "ke-", "se-", "te-", "be-", "me-", atau "pe-". Lalu hasilnya ditampung ke variabel `$kata`. Kemudian kata diperiksa kembali fungsi `cekKamus()` untuk memeriksa apakah kata tersebut tersedia di kamus atau tidak. Jika kata tersebut tersedia di kamus, maka fungsi akan mengembalikan hasil stemming kata. Jika tidak tersedia di kamus, maka kata tersebut selanjutnya dikembalikan sebagai hasil stemming kata.

7.1.5 Segmen Program Stopword

Segmen program *stopword* berguna untuk menghapus kata-kata yang sering muncul dan tidak memiliki arti dalam suatu teks komentar. Pada tahap *stopword*, kata-kata yang tidak penting dibuang dari daftar kata, misalnya kata “yang”, “dimana”, “mengapa”, “yaitu”, “yakni”, dan sebagainya. Proses *stopword* juga

berguna untuk mengurangi kompleksitas teks dokumen. Berikut ini dijelaskan potongan program *stopword*:

Segmen Program 7.5 Stopword

```

01: public function stopword_kata($komen) {
02:     $gabungkan_kata = '';
03:     $cek = preg_split('/ /', $komen);
04:     $count = count($cek);
05:     for($i=0;$i<$count;$i++) {
06:         if($this->stopword_cek($cek[$i]) == true) {
07:             unset($cek[$i]);
08:         }else{
09:             $cek[$i] = $cek[$i];
10:         }
11:     }
12:     $gabungkan_kata = implode(' ', $cek);
13:     return $gabungkan_kata;
14: }
```

Berikut dijelaskan tentang alur kerja segmen program *stopword*. Pada baris 1 – 4, proses *stopword* dimulai dengan mengambil data dari parameter \$komen. Lalu variabel \$gabungkan_kata diinisialisasi. Lalu data komentar dalam parameter \$komen di-split menjadi array \$cek menggunakan fungsi preg_split(). Kemudian variabel \$count diinisialisasi dengan menghitung total data pada array \$cek dari fungsi count().

Kemudian pada baris 5 – 11, dilakukan proses iterasi terhadap array \$cek hingga mencapai jumlah data \$count. Kemudian fungsi stopword_cek() memeriksa setiap kata dari \$cek. Jika terdapat kata tersebut terdapat dalam kamus stopword, maka data tersebut dihapus dari array \$cek. Jika kata tersebut tidak terdapat dalam kamus stopword, maka kata dikembalikan ke indeks array \$cek. Lalu pada baris 12 – 14, array \$cek digabung menjadi satu data menggunakan fungsi implode, hasilnya dimasukan ke variabel \$gabungkan_kata. Terakhir hasil variabel \$gabungkan_kata dikembalikan fungsi menjadi hasil stopword kata.

7.1.6 Segmen Program Tokenizing

Segmen program *tokenizing* berguna untuk memotong teks komentar menjadi daftar kata-kata. *Tokenizing* adalah proses pemotongan dokumen menjadi

kata-kata setelah menjadi proses *filtering*. Hasil pemotongan kata-kata tersebut dijadikan kumpulan kata dan membentuk daftar kata. Dalam hal ini, hasil *token* sekaligus mewakili *unigram*. Berikut ini dijelaskan segmen program tokenizing:

Segmen Program 7.6 Tokenizing

```
01: function tokenizer($komen) {
02:     $komen = explode(" ", $komen);
03:     $komen = implode("<br/>", $komen);
04:     return $komen;
05: }
```

Pada baris 1 – 5, proses *tokenizing* dimulai dengan mengambil data dari parameter \$komen. Lalu data komentar dalam parameter \$komen dipisahkan berdasarkan spasi menggunakan fungsi *explode()*. Hasilnya disimpan ke array \$komen. Kemudian data array \$komen digabungkan dengan menggunakan pemisah “
” menggunakan fungsi *implode()*. Lalu hasilnya disimpan pada variabel \$komen. Terakhir fungsi mengembalikan nilai dari variabel \$komen.

Segmen program ini diimplementasikan pada *preprocessing* baik pelatihan data maupun pengujian data. Segmen program ini ditempatkan pada file model dalam *framework* Codeigneter. Dengan segmen ini teks komentar akan dipotong menjadi daftar kata. Hasilnya berupa token yang berisi daftar kata untuk selanjutnya diproses oleh model.

7.1.7 Segmen Program Bigram

Segmen program *bigram* berguna untuk memproses komentar menjadi sekumpulan kata yang masing-masing data terdiri atas dua kata. Dalam hal ini, *bigram* berarti gabungan 2 kata yang diambil dari teks komentar dengan mengerakkan satu kata maju ke depan. Proses *bigram* terjadi setelah dilakukan *tokenizing* khususnya jika *ngram* memakai *bigram*. Berikut ini dijelaskan segmen program *bigram*:

Segmen Program 7.7 Bigram

```
01: public function bigram($komentar) {
02:     $komen = explode("<br/>", $komentar);
03:     $totalkata = count($komen);
```

Segmen Program 7.7 (Lanjutan)

```

04:     $fitur = "";
05:     $kata = array();
06:     if($totalkata<2) {
07:         $komen = implode("_", $komen);
08:         return $komen;
09:     }else{
10:         for($i=0;$i<$totalkata - 1;$i++) {
11:             $ii = $i + 1;
12:             $kata[$i] = $komen[$i] . '_' . $komen[$ii];
13:         }
14:         $fitur = implode("<br/>", $kata);
15:         return $fitur;
16:     }
17: }

```

Pada baris 1 – 5, data komentar dalam parameter \$komentar dipisahkan berdasarkan
 menggunakan fungsi explode(). Hasilnya disimpan ke array \$komen. Kemudian data array \$komen dihitung total datanya menggunakan fungsi count() lalu hasilnya disimpan ke variabel \$totalkata. Kemudian array \$kata dinisialisasi dimana array \$kata akan digunakan sebagai penampung hasil bigram.

Pada baris 6 – 17, lakukan pengecekan \$totalkata, jika \$totalkata lebih kecil daripada 2, maka gabungkan data pada array \$komen menggunakan fungsi implode() dengan pemisah “_”. Lalu hasilnya dikembalikan. Jika \$totalkata tidak sama dengan 2, maka lakukan iterasi hingga mencapai \$totalkata – 1. Inisialisasi variabel \$ii dari \$i + 1 setiap kali iterasi. Lalu gabungkan array \$komen[\$i] dengan \$komen[\$ii] dengan pemisah tanda “_”, lalu hasilnya disimpan pada array \$kata[\$i]. Kemudian setelah iterasi berakhir, gabungkan data pada array \$kata menggunakan fungsi implode() dengan pemisah
, lalu hasilnya disimpan pada variabel \$fitur. Terakhir kembalikan nilai \$fitur, sehingga proses segmen program bigram berakhir.

7.1.8 Segmen Program Model Penyaring Cyberbullying

Segmen program model penyaring *cyberbullying* berguna untuk membuat model penyaring *cyberbullying* melalui pemrosesan dataset komentar yang sebelumnya telah melalui *preprocessing*. Model ini akan menyimpan model *prior* dan model probabilitas kata. Selanjutnya model tersebut digunakan sebagai alat

klasifikasi komentar menjadi *cyberbullying* atau *non-cyberbullying* berdasarkan perhitungan menggunakan *Multinomial Naive Bayes*. Berikut dijelaskan segmen program pembuatan model penyaring *cyberbullying*:

Segmen Program 7.8 Pembuatan Model Penyaring Cyberbullying

```

01: public function get_antibullying_model(){
02:     $this->MyModel->hapus_datakelaskata();
03:     $this->MyModel->hapus_dataprior();
04:     $is_reset = $this->MyModel->reset_kelaskatamodel();
05:     if($is_reset==1){
06:         $is_kelaskata = $this->MyModel->get_kelaskatamodel();
07:         if($is_kelaskata==1){
08:             $is_prior = $this->MyModel->get_priormodel();
09:             if($is_prior==1){
10:                 $is_probabilitas = $this->MyModel->
11:                     probabilitas_kata();
12:                 if($is_probabilitas==1){
13:                     $callback = array('pesan'=>'Model antibullying
14:                         berhasil dibuat!');
15:                     echo json_encode($callback);
16:                 }else{
17:                     $callback = array('pesan'=>'Model probabilitas
18:                         kata gagal dibuat!');
19:                     echo json_encode($callback);
20:                 }
21:             }else{
22:                 $callback = array('pesan'=>'Model kelas kata
23:                     gagal dibuat!');
24:                 echo json_encode($callback);
25:             }
26:         }else{
27:             $callback = array('pesan'=>'Model kelas kata
28:                     gagal direset!');
29:             echo json_encode($callback);
30:         }
31:     }
32: }
```

Pada baris 1- 4, dilakukan penghapusan dan peresetan terhadap data kelas kat dan data prior dari model sebelumnya. Hasilnya ditampung ke variabel \$is_reset. Pada baris 5 – 6, lakukan pengecekan pada variabel \$is_reset. Jika variabel \$is_reset bernilai 1 maka buatlah model kelas kata menggunakan fungsi get_kelaskatamodel(). Lalu hasilnya ditampung ke variabel \$is_kelaskata. Namun

jika \$is_reset tidak bernilai 1, fungsi akan mengembalikan pesan “model kelas kata gagal direset”.

Pada baris 7 – 8, lakukan pengecekan pada variabel \$is_kelaskata. Jika variabel \$is_kelaskata bernilai 1, maka buatlah model prior menggunakan fungsi get_priormodel(). Lalu hasilnya ditampung ke variabel \$is_prior. Jika variabel \$is_kelaskata tidak bernilai 1, maka fungsi akan mengembalikan pesan “model kelas kata gagal dibuat”.

Pada baris 9 – 10, lakukan pengecekan pada variabel \$is_prior. Jika variabel \$is_prior bernilai 1, maka buatlah model probabilitas kata menggunakan fungsi probabilitas_kata(). Lalu hasilnya ditampung ke variabel \$is_probabilitas. Namun jika \$is_prior bernilai 0, maka fungsi akan mengembalikan pesan “model prior gagal dibuat”. Pada baris 11 – 13, lakukan pengecekan pada variabel \$is_probabilitas. Jika variabel \$is_probabilitas bernilai 1, fungsi akan mengembalikan pesan “model antibullying berhasil dibuat”. Tetapi jika variabel \$is_probabilitas tidak bernilai 1, maka fungsi akan mengembalikan pesan “model probabilitas kata gagal dibuat”.

7.1.9 Segmen Program Kalkulasi Kategori Komentar

Segmen program kalkulasi kategori komentar berguna untuk menghitung nilai Cmap. Hasil perhitungan selanjutnya digunakan untuk menentukan komentar menjadi kategori *cyberbullying* atau kategori *non-cyberbullying*. Segmen program ini bertugas langsung pada segala komentar yang disaring hingga menghasilkan kategori cyberbullying atau tidak. Berikut dijelaskan segmen program kalkulasi kategori:

Segmen Program 7.9 Kalkulasi Kategori

```

01: public function kalkuasi_kategori($tes="") {
02:     $callback = array();
03:     $cyberbullying = 1;
04:     $noncyberbullying = 1;
05:     $cek = $this->case_folding($tes);
06:     $cek = $this->cleansing($cek);
07:     $cek = $this->gabungkan_kata($cek);
08:     $cek = $this->normalisasi_bahasagaul($cek);
09:     $cek = $this->stemmer($cek);

```

Segmen Program 7.9 (Lanjutan)

```

10:     $cek = $this->stopword($cek);
11:     $tokenizing = $this->tokenizer($cek);
12:     $cek = explode("<br/>", $tokenizing);
13:     $count = count($cek);
14:     for($i=0;$i<$count;$i++) {
15:         $teks = $this->cek_kata($cek[$i]);
16:         if(!is_int($teks)) {
17:             $cyberbullying = $cyberbullying *
18:                 $teks->PROBABILITAS_CYBERBULLY;
19:             $noncyberbullying = $noncyberbullying *
20:                 $teks->PROBABILITAS_NONCYBERBULLY;
21:         } else {
22:             $bully = $this->is_prior(
23:                 'dataset-komentar','cyberbullying');
24:             $nonbully = $this->is_prior(
25:                 'dataset-komentar','non-cyberbullying');
26:             $cyberbullying = $cyberbullying * (1 /
27:                 ($bully->TOTAL_KELASKATA_PLUS_KATAUNIK));
28:             $noncyberbullying = $noncyberbullying * (1 /
29:                 ($nonbully->TOTAL_KELASKATA_PLUS_KATAUNIK));
30:         }
31:         $prior_bully = $this->is_prior(
32:             'dataset-komentar','cyberbullying');
33:         $prior_nonbully = $this->is_prior(
34:             'dataset-komentar','non-cyberbullying');
35:         $cyberbullying = $cyberbullying *
36:             $prior_bully->PROBABILITAS;
37:         $noncyberbullying = $noncyberbullying *
38:             $prior_nonbully->PROBABILITAS;
39:         if($cyberbullying<=$noncyberbullying) {
40:             $callback = array('status' => 'sukses',
41:                 'komentar' =>$tes,
42:                 'pesan'=>'non-cyberbullying',
43:                 'preprocessing'=>$tokenizing,
44:                 'bobot_noncyberbullying' => $noncyberbullying,
45:                 'bobot_cyberbullying' => $cyberbullying);
46:         } else{
47:             $callback = array('status' => 'sukses',
48:                 'komentar'=>$tes,
49:                 'pesan'=>'cyberbullying',
50:                 'preprocessing'=>$tokenizing,
51:                 'bobot_noncyberbullying' => $noncyberbullying,
52:                 'bobot_cyberbullying' => $cyberbullying);
53:         }
54:     return $callback;
55: }

```

Pada baris 1 – 4, dilakukan inisialisasi pada masing-masing variabel \$callback berupa array, \$cyberbullying bernilai 1, dan \$noncyberbullying bernilai 1. Pada baris 5 – 11, dilakukan proses preprocessing dari data parameter \$tes untuk selanjutnya diproses ke tahap masing-masing *case_folding*, *cleaning*, normalisasi

bahasa gaul, *stemming*, *stopword*, dan *tokenizing*. Pada baris 12 – 13, hasil tokenizing selanjutnya di-explode menggunakan fungsi explode(). Lalu hasilnya ditampung ke array \$cek. Hitung total array \$cek menggunakan fungsi count(), lalu tampung hasilnya pada variabel \$count.

Pada baris 14 – 25, dilakukan proses iterasi pada array \$cek. Lakukanlah pengecekan setiap kata pada array \$cek menggunakan fungsi cek_kata(). Kemudian hasilnya ditampung pada variabel \$teks. Jika variabel \$teks bukan merupakan integer, maka hitung cyberbullying dengan mengalikan variabel \$cyberbullying dan probabilitas cyberbullying lalu hasilnya ditampung pada variabel \$cyberbullying, dan hitung juga noncyberbullying dengan mengalikan variabel \$noncyberbullying dan probabilitas noncyberbullying lalu hasilnya ditampung pada variabel \$noncyberbullying. Jika variabel \$teks merupakan integer, maka hitung cyberbullying dengan mengalikan variabel \$cyberbullying dan seper hasil total_kelaskata_plus_kataunik dari kelas bully lalu hasilnya ditampung pada variabel \$cyberbullying, dan hitung juga noncyberbullying dengan mengalikan variabel \$noncyberbullying dan seper hasil total_kelaskata_plus_kataunik dari kelas nonbully lalu hasilnya ditampung pada variabel \$noncyberbullying. Lakukanlah iterasi hingga tercapai jumlah \$count.

Pada baris 26 – 36, panggil fungsi is_prior() untuk mendapatkan nilai prior pada kelas bully, lalu hasilnya ditampung ke variabel \$prior_bully. Panggil juga fungsi is_prior() untuk mendapatkan nilai prior pada kelas nonbully, lalu hasilnya ditampung ke variabel \$prior_nonbully. Kemudian hitung total nilai cyberbullying dengan mengalikan variabel \$cyberbullying dan prior bully, lalu hasilnya ditampung ke variabel \$cyberbullying. Hitung juga total nilai noncyberbullying dengan mengalikan variabel \$noncyberbullying dan prior nonbully, lalu hasilnya ditampung ke variabel \$noncyberbullying. Kemudian lakukan pengecekan dimana jika nilai variabel \$cyberbullying lebih kecil daripada sama dengan nilai variabel \$noncyberbullying, maka simpan hasil kategori ke array \$callback meliputi status bernilai sukses, teks komentarnya, pesan bernilai non-cyberbullying, preprocessing bernilai hasil tokenizing, nilai bobot cyberbullying dan nilai bobot noncyberbullying. Namun jika nilai variabel \$cyberbullying tidak lebih kecil

daripada sama dengan nilai variabel \$noncyberbullying, maka simpan hasil kategori ke array \$callback meliputi status bernilai sukses, teks komentarnya, pesan bernilai cyberbullying, preprocessing bernilai hasil tokenizing, nilai bobot cyberbullying dan nilai bobot noncyberbullying. Terakhir fungsi mengembalikan nilai callback.

7.1.10 Segmen Program Pengujian Model

Segmen program pengujian model berguna untuk menguji tingkat kesuksesan model seperti akurasi, *error*, *precision*, *recall*, dan *f-score*. Segmen program ini bertujuan mengevaluasi model. Sehingga bisa ditemukan model terbaik untuk menilai kategori cyberbullying pada dataset komentar. Berikut dijelaskan segmen program pengujian model:

Segmen Program 7.10 Pengujian Model

```

01: public function pengujian($id){
02:     $this->db->trans_begin();
03:     $pengujian = $this->is_uji($id);
04:     $segmentasi = $this->segmentasi(
05:         $id,$pengujian->SEGMENT_AWAL,$pengujian->SEGMENT_AHIR);
06:     $suji = array(); $total_data = $pengujian->TOTAL_DATA;
07:     $totalcyberbullying = 0; $totalnoncyberbullying = 0;
08:     $totaldata = 0;
09:     $segmen = $this->segmentasix($segmentasi);
10:     $tp = $segmen['tp']; $fn = $segmen['fn'];
11:     $fp = $segmen['fp']; $tn = $segmen['tn'];
12:     $totaldata = $totalnoncyberbullying + $totalcyberbullying;
13:     $akurasi = 0; $error = 0; $recall = 0; $specificity = 0;
14:     $precisi = 0; $fscore = 0;
15:     if(($tp+$fn+$fp+$tn)==0){
16:         $akurasi = 0; $error = 0;
17:     }else{
18:         $akurasi = ($tp+$tn) / ($tp+$fn+$fp+$tn);
19:         $error = ($fp+$fn) / ($tp+$fn+$fp+$tn);
20:     }
21:     if(($tp+$fn)==0){
22:         $recall = 0;
23:     }else{
24:         $recall = ($tp) / ($tp+$fn);
25:     }
26:     if(($fp+$tn)==0){
27:         $specificity = 0;
28:     }else{
29:         $specificity = ($tn) / ($fp+$tn);
30:     }
31:     if(($tp+$fp)==0) {
32:         $precision = 0;
33:     }else{
34:         $precision = ($tp) / ($tp+$fp);
35:     }
36:     if(($fn+$tn)==0) {
37:         $f1score = 0;
38:     }else{
39:         $f1score = ((2*$precision*$recall) / ( $precision + $recall));
40:     }
41:     $data = array(
42:         'status' => 'success',
43:         'comment' => $this->getComment($id),
44:         'text' => $this->getPreprocessing($id),
45:         'token' => $this->getTokens($id),
46:         'cyberbullying' => $totalcyberbullying,
47:         'noncyberbullying' => $totalnoncyberbullying,
48:         'akurasi' => $akurasi,
49:         'error' => $error,
50:         'recall' => $recall,
51:         'specificity' => $specificity,
52:         'precision' => $precision,
53:         'f1score' => $f1score
54:     );
55: }
```

Segmen Program 7.10 (Lanjutan)

```

31:         $precisi = 0;
32:     }else{
33:         $precisi = ($tp) / ($tp+$fp);
34:     }
35:     if(($precisi+$recall)==0){
36:         $fscore = 0;
37:     }else{
38:         $fscore = (2*$precisi*$recall)/($precisi+$recall);
39:     }
40:     $evaluasi = array("tp" => $tp, "fp" => $fp, "fn" => $fn,
41:                       "tn" => $tn, "akurasi" => $akurasi, "error" => $error,
42:                       "recall" => $recall, "specificity" => $specificity,
43:                       "precisi" => $precisi, "fscore" => $fscore,
44:                       "data_tes" => $n_tes,
45:                       "jml_cyberbullying" => $totalcyberbullying,
46:                       "jml_noncyberbullying" => $totalnoncyberbullying);
47:     $this->pengujian_ubah($evaluasi,$pengujian->ID_UJI);
48:     $this->db->trans_complete();
49:     if ($this->db->trans_status() === FALSE) {
50:         $this->db->trans_rollback();
51:         return 0;
52:     }else{
53:         $this->db->trans_commit();
54:         return 1;
55:     }
56: }

```

Pada baris 1 - 7, dilakukan inisialisasi transaksi. Data uji bernilai \$id dipanggil menggunakan fungsi `is_uji()`, lalu hasilnya ditampung pada variabel `$pengujian`. Ambil segmen dataset komentar menggunakan fungsi `segmentasi()`, lalu hasilnya ditampung pada variabel `$segmentasi`. Inisialisasi array `$suji`. Ambil total data dari variabel `$pengujian`, lalu hasilnya ditampung pada variabel `$total_data`. Kemudian lakukan inisialisasi pada masing-masing variabel `$cyberbullying`, `$noncyberbullying`, dan `$totaldata` bernilai 0.

Pada baris 8 – 13, lakukan inisialisasi array `$segmen` menggunakan fungsi `segmentasix` dengan memasukan nilai `$segmentasi` pada argumennya. Kemudian lakukan inisialisasi pada masing-masing variabel `$tp`, `$fp`, `$fn`, `$tn`, dari masing-masing nilai `$segmen['tp']`, `$segmen['fp']`, `$segmen['fn']`, dan `$segmen['tn']`. Lalu hitung `$totaldata` dengan menjumlahkan `$cyberbullying` dan `$noncyberbullying`. Kemudian lakukan inisialisasi masing-masing variabel `$akurasi`, `$error`, `$precisi`, `$specificity` `$recall`, `$fsore`, masing-masing bernilai 0.

Pada baris 14 – 39, lakukan pengecekan dimana jika total \$fn, \$tn, \$fp, \$fn sama dengan 0, maka berikanlah nilai \$akurasi dan \$error masing-masing 0. Jika total \$fn, \$tn, \$fp, \$fn tidak sama dengan 0, maka hitung \$akurasi dengan menjumlahkan \$tp dan \$tn dibagi total \$tp, \$fn, \$fp,\$tn. Dan hitung error dengan menjumlahkan \$fp dan \$fn dibagi total \$tp, \$fn, \$fp, \$tn. Lakukan pengecekan dimana jika jumlah \$tp dan \$fn sama dengan 0, maka berilah nilai recall sama dengan 0. Jika jumlah \$tp dan \$fn tidak sama dengan 0, maka hitung \$recall dengan menghitung \$tp dibagi total \$tp dan \$fn. Kemudian lakukan pengecekan dimana jika \$fp ditambah \$tn sama dengan 0, maka berilah \$specificity bernilai 0. Jika \$fp ditambah \$tn tidak sama dengan 0, maka hitung \$specificity dengan menghitung \$tn dibagi total \$fp dan \$tn. Lalu lakukan pengecekan dimana jika \$tp ditambah \$fp sama dengan 0, maka berilah nilai \$precisi sebesar 0. Jika \$tp ditambah \$fp tidak sama dengan 0, maka hitung \$precisi dengan menghitung \$tp dibagi total \$tp dan \$fp. Kemudian lakukan pengecekan dimana jika \$precisi ditambah \$recall sama dengan nol, maka berilah nilai \$fscore sebesar 0. Jika \$precisi ditambah \$recall tidak sama dengan nol, maka hitung \$fscore dengan menghitung 2 kali \$precisi dikali \$recall dibagi total \$precisi dan \$recall.

Pada baris 40 – 50, simpan hasil pengujian pada array \$evaluasi meliputi tp bernilai \$tp, fp bernilai \$fp, fn bernilai \$fn, tn bernilai \$tn, akurasi bernilai \$akurasi, error bernilai \$error, recall bernilai \$recall, specificity bernilai \$specificity, precision bernilai \$precisi, fscore bernilai \$fscore, jumlah tes bernilai \$n_tes, total cyberbullying bernilai \$totalcyberbullying, total totalnoncyberbullying bernilai \$cyberbullying. Kemudian simpan hasil evalasi ke database pengujian menggunakan fungsi pengujian_ubah(). Terakhir akhiri transaksi, jika status transaksi bernilai false, maka dilakukan rollback dan fungsi akan mengembalikan nilai 0. Jika status transaksi tidak false, maka commit dan fungsi akan mengembalikan nilai 1.

7.2 Segmen Program Sistem Manajemen Konten

Pada sub bab ini dijelaskan tentang beberapa segmen program sistem manajemen konten. Segmen program sistem manajemen konten berisi beberapa potongan program yang digunakan dalam pembuatan sistem manajemen konten

menggunakan *framework* Codeigneter. Adapun segmen program tersebut seperti segmen program pembuatan artikel, segmen program pengeditan artikel, segmen program penghapusan artikel, segmen program penghapusan artikel, segmen program penguploadan galeri, segmen program mengikuti akun, dan segmen program berhenti mengikuti akun, segmen program *pretty URL*, dan segmen program registrasi akun. Berikut penjelasan selengkapnya:

7.2.1 Segmen Program Pembuatan Artikel

Segmen program pembuatan artikel digunakan untuk membuat artikel dan menyimpan artikel ke database. Artikel tersebut selanjutnya akan dipublikasi pada situs *blog*. Pada sistem manajemen konten biasanya data artikel akan disimpan di database sebelum dipublikasikan lebih lanjut. Berikut dijelaskan segmen program pembuatan artikel:

Segmen Program 7.11 Pembuatan Artikel

```

01: public function artikel_buat(){
02:     $this->form_validation->set_rules('judul', 'Judul',
03:                                         'required|min_length[5]|max_length[160]');
04:     $this->form_validation->set_rules('deskripsiartikel',
05:                                         'Deskripsi Artikel', 'required|min_length[5]|
06:                                         max_length[500]');
07:     $this->form_validation->set_rules('katakunci', 'Kata
08:                                         Kunci', 'required|min_length[5]|max_length[300]');
09:     $this->form_validation->set_rules('isiartikel',
10:                                         'Isi Artikel', 'required|min_length[160]');
11:     if($this->form_validation->run() == false){
12:         $callback = array('status'=>'gagal',
13:                           'pesan'=>validation_errors());
14:         echo json_encode($callback);
15:     }else{
16:         $return = $this->MyModel->artikel_buat($_POST);
17:         $callback = array('status'=>'sukses',
18:                           'pesan'=>'berhasil disimpan',
19:                           'judul_url' => $return['JUDUL_URL']);
20:         $timezone = new DateTimeZone('Asia/Jakarta');
21:         $tgl = new DateTime();
22:         $tgl->setTimeZone($timezone);
23:         $pengguna = $this->MyModel->
24:             getPenggunaById($_SESSION['user_id']);
25:         $akunpengikut = $this->MyModel->
26:             getpengikut($pengguna->ID);
27:         $pengikutku = $this->MyModel->
28:             get_listpengikut($akunpengikut->ID_PENGIKUT);
29:         $url_photo = "";
30:     }
31: }
```

Segmen Program 7.11 (Lanjutan)

```

21:     if($pengguna->PHOTO_PROFIL<>"") {
22:         $url_photo = base_url($pengguna->PHOTO_PROFIL);
23:     }else{
24:         $url_photo = base_url('asset/logo/user.png');
25:     }
26:     if(!empty($pengikutku)){
27:         foreach($pengikutku->result() as $follower){
28:             $notifikasi2 = array("ID" => $follower->ID,
29:             "TANGGAL" => $tgl->format('Y-m-d H:i:s'),
30:             "NOTIFIKASI" => '<h5><a href="'.
31:             base_url('artikel/detail/').$return['JUDUL_URL']
32:             .'">'. $return['JUDUL'] .'!</a></h5>
33:             <div class="media">'. $tgl->format('Y-m-d
35:             H:i:s') .' | '. $pengguna->FIRST_NAME . ' ' .
36:             $pengguna->LAST_NAME .' telah mempublikasikan
37:             artikel berjudul: '. $return['JUDUL'] .'.</div>
38:             ','AKTIF" => 'Y',"DIBACA" => "N");
39:             $this->MyModel->notifikasi_buat($notifikasi2);
40:         }
41:     }
42:     echo json_encode($callback);
43: }
44: }
```

Pada baris 1 – 13, lakukanlah validasi form pada data post meliputi judul, deskripsi artikel, kata kunci, dan isi artikel. Jika validasi form bernilai false, maka fungsi akan mengembalikan nilai file json dengan isi status gagal dan pesan error. Jika validasi form tidak bernilai false, maka simpan data artikel post ke database menggunakan fungsi artikel_buat(). Lalu lakukan inisialisasi \$callback berisi status sukses, pesan berhasi disimpan, dan judul urlnya.

Pada 14 – 25, lakukanlah inisialisasi tanggal beserta zona waktunya, lalu hasilnya ditampung ke variabel \$tgl. Kemudian lakukanlah inisialisasi pada variabel \$pengguna menggunakan fungsi getPenggunaById() dengan memasukan argumen \$_SESSION['user_id']. Lalu lakukanlah inisialisasi pada variabel \$akunpengikut menggunakan fungsi getpengikut() dengan memasukan argumen \$pengguna->ID. Kemudian lakukanlah inisialisasi pada variabel \$pengikutku menggunakan fungsi get_listpengikut() dengan memasukan argumen \$akunpengikut->ID_PENGIKUT. Lalu lakukan inisialisasi variabel \$url_photo. Lalu lakukan pengecekan jika \$pengguna->PHOTO_PROFIL tidak sama dengan "", maka isi variabel \$url_photo dengan base_url('\$pengguna-

>PHOTO_PROFIL'). Jika \$pengguna->PHOTO_PROFIL sama dengan "", maka isi variabel \$url_photo dengan base_url('asset/logo/user.png'). Pada baris 26 – 34, lakukanlah pengecekan pada variabel \$pengikutku, jika \$pengikutku tidak bernilai kosong, maka lakukanlah iterasi dimana setiap iterasi akan mengirimkan notifikasi berupa pesan adanya artikel baru dari penulis kepada pengikut akunnya. Lalu fungsi mengembalikan nilai \$callback menjadi json.

7.2.2 Segmen Program Pengeditan Artikel

Segmen program pengeditan artikel digunakan untuk melakukan perubahan dan penyimpanan pada artikel yang disunting penulisnya. Segmen program ini akan aktif tatkala penulis menekan tombol simpan untuk perubahan artikel yang disuntingnya. Segela perubahan artikel selanjutnya akan disimpan pada database. Berikut penjelasan segmen program pengeditan artikel:

Segmen Program 7.12 Pengeditan Artikel

```

01: public function artikel_ubah(){
02:     $this->form_validation->set_rules('judul',
03:         'Judul', 'required');
04:     $this->form_validation->set_rules('deskripsiartikel',
05:         'Deskripsi Artikel', 'required|max_length[500]');
06:     $this->form_validation->set_rules('katakunci',
07:         'Kata Kunci', 'required|max_length[300]');
08:     $this->form_validation->set_rules('isiartikel',
09:         'Isi Artikel', 'required|min_length[160]');
10:     if($this->form_validation->run() == false){
11:         $callback = array('status' => 'gagal',
12:             'pesan' => validation_errors());
13:         echo json_encode($callback);
14:     } else{
15:         $callback = array('status' => 'sukses',
16:             'pesan' => 'berhasil disimpan');
17:         $this->MyModel->posting_ubah($_POST);
18:         echo json_encode($callback);
19:     }

```

Pada baris 1 – 6, lakukanlah validasi form masing-masing judul, deskripsi artikel, kata kunci, dan isi artikel. Khusus form deskripsi artikel dibatasi maksimal 500 karakter. Sementara form kata kunci dibatasi maksimal 300 karakter. Lalu form isi artikel dibatasi minimal 160 karakter.

Pada baris 7 – 15, lakukan pengecekan pada form validasi. Jika form validasi bernilai false, maka lakukanlah inisialisasi array \$callback dengan diisi status gagal dan pesan error. Jika form validasi tidak bernilai false, maka lakukan inisialisasi array \$callback dengan diisi status sukses dan pesan sukses. Kemudian fungsi mengembalikan \$callback menjadi json. Lalu simpan perubahan artikel dari data post yang dikirimkan untuk disimpan menggunakan fungsi posting_ubah(). Terakhir fungsi mengembalikan \$callback menjadi json.

7.2.3 Segmen Program Penghapusan Artikel

Segmen program penghapusan artikel berguna untuk menghapus artikel yang tidak diinginkan penulis. Segmen program ini akan aktif tatkala penulis menekan tombol hapus pada artikel yang dipilihnya. Artikel yang berhasil dihapus tidak akan ditampilkan pada halaman manajemen artikel. Berikut dijelaskan segmen program penghapusan artikel:

Segmen Program 7.13 Penghapusan Artikel

```

01: public function artikel_hapus(){
02:     if($this->MyModel->posting_hapus($_POST)>0) {
03:         header('Location: ' . base_url() .
04:                 'admin/artikel_kelola');
05:     }

```

Pada baris 1 – 7, fungsi melakukan pengecekan pada variabel \$_POST menggunakan fungsi posting_hapus. Jika hasilnya bernilai 0, maka lakukan refresh halaman web ke alamat base_url('admin/artikel_kelola'). Sehingga jika artikel berhasil terhapus, sistem otomatis akan mereset halaman baru. Namun jika \$_POST bernilai bukan 0, maka segmen program tidak akan aktif.

Segmen program penghapus akan diimplementasikan untuk menghapus artikel pada sistem manajemen konten. Pada segmen ini, artikel yang terhapus disembunyikan dari tampilan antar muka. Sehingga artikel tidak bisa diakses oleh pengguna. Artikel yang terhapus sebenarnya dapat ditampilkan kembali jika nilai aktifnya dikembalikan menjadi 1.

7.2.4 Segmen Program Penguploadan Galeri

Segmen program pengupload galeri berguna untuk mengupload gambar yang akan dijadikan sebagai gambar ilustrasi artikel. Segmen program ini akan menyimpan gambar dan URL-nya ke database sistem. Gambar yang berhasil ter-*upload* akan disimpan pada penyimpanan hosting. Sedangkan URL gambar disimpan pada database sehingga jika gambar dipanggil tinggal memanggill URL gambarnya. Berikut dijelaskan segmen program penguploadan galeri:

Segmen Program 7.14 Penguploadan Galeri

```

01: function upload_file() {
02:     $config['upload_path'] = './images/galeri/';
03:     $config['allowed_types'] = 'png|jpg|jpeg|gif';
04:     $config['max_size'] = '2048';
05:     if (isset($_FILES['file']['name'])) {
06:         if (0 < $_FILES['file']['error']) {
07:             $callback = array("status" => "gagal",
08:                               "pesan" => 'Error selama file diupload'.
09:                                         $_FILES['file']['error']);
10:         }else{
11:             if(file_exists('images/galeri/' .
12:                           $_FILES['file']['name'])) {
13:                 $callback = array("status" => "gagal",
14:                                   "pesan" => 'File ' . $_FILES['file']['name'] .
15:                                         'sudah ada!');
16:             echo json_encode($callback);
17:         }else{
18:             if($_FILES['file']['size']>2048000){
19:                 $callback = array("status" => "gagal",
20:                               "pesan" => 'Ukuran gambar tidak boleh >2MB');
21:             echo json_encode($callback);
22:         }else{
23:             $this->load->library('upload', $config);
24:             if(!$this->upload->do_upload('file')) {
25:                 $callback = array("status" => "gagal",
26:                                   "pesan" => $this->upload->display_errors());
27:             echo json_encode($callback);
28:         }
29:     }
30: }
31: }
32: }else{
33:     echo 'Tolong pilih gambar';
34: }

```

Pada baris 1 – 4, lakukanlah konfigurasi pada array \$config. Elemen ‘upload_path’ dari \$config diisi dengan url ‘./images/galeri/’. Elemen ‘allowed_types’ dari \$config diisi dengan jenis/ format gambar yang diperbolehkan meliputi png, jpg, jpeg, gif. Elemen max_size dari \$config diisi dengan 2048 yang berarti maksimal 2MB besar file gambar yang diizinkan untuk diupload.

Pada baris 5 dan 32 – 34, lakukan pengecekan eksistensi file gambar, jika file belum dipilih, maka munculkan pesan *error* bahwa diwajibkan memilih gambar. Pada baris 6 – 9, lakukan pengecekan pengupload, jika terjadi kesalahan selama proses penguploadan gambar, maka munculkan pesan error bahwa gagal diupload. Pada baris 10 – 13, jika tidak error selama pengupload gambar, maka lakukan pengecekan pada nama fila gambar yang diupload. Jika nama gambar sudah ada di sistem, maka munculkan pesan error bahwa gambar sudah ada. Pada baris 14 – 17, lakukan pengecekan ukuran gambar. Jika ukuran gambar lebih dari 2 MB, maka munculkan pesan error bahwa gambar melebihi ukuran maksimal yang dibolehkan upload. Pada baris 18 – 31, lakukan konfigurasi pada library upload. Lalu lakukan pengecekan file gambar berhasil diupload atau tidak. Jika file gambar gagal diupload, maka munculkan pesan error bahwa gambar gagal diupload. Jika file gambar berhasil diupload, fungsi akan mengembalikan pesan sukses.

7.2.5 Segmen Program Follow-Unfollow Account

Segmen program mengikuti akun berguna bagi member untuk mengikuti akun dari member lainnya. Segmen program ini akan aktif tatkala member menekan tombol ikuti pada profil member lainnya. Jika user menekan tombol ikuti setelah mengikuti akun penulis, maka sistem akan melakukan batal mengikuti. Berikut dijelaskan tentang segmen program mengikuti akun:

Segmen Program 7.15 Follow-Unfollow Account Bagian 1

```

01: public function mengikuti(){
02:     $callback = array('status'=>'gagal',
03:                         'pesan'=>'Anda belum login!');
04:     if($_POST['id_user_mengikuti']==0){
05:         $callback = array('status'=>'gagal',
06:                         'pesan'=>'Anda belum login!');
07:     }
08: }
```

Segmen Program 7.15 (Lanjutan)

```

        'pesan'=>'Anda belum login!');

05:    }else{
06:        $pengguna = $this->MyModel->
07:            getPenggunaById($_POST['id_user_mengikuti']);
08:        if($_POST['id_bio']==$pengguna->ID){
09:            $callback = array('status'=>'gagal',
10:                'pesan'=>'Anda adalah penulisnya!');
11:        }else{
12:            if($pengguna->ID>0){
13:                $mengikuti = $this->MyModel->
14:                    getmengikuti($pengguna->ID);
15:                if(empty($mengikuti)){
16:                    $this->MyModel->
17:                        buat_mengikuti($pengguna->ID);
18:                    $akun_mengikuti = $this->MyModel->
19:                        getmengikuti($pengguna->ID);
20:                    $this->MyModel->
21:                        tambah_detailmengikuti($_POST['id_bio'],
22:                            $akun_mengikuti->ID_MENGIKUTI);
23:                    $callback = array('status'=>'sukses',
24:                        'pesan'=>'Berhasil mengikuti!');
25:                }
26:            }
27:        }
28:    }
29: }
30: 
```

Pada baris 1 – 4, dilakukan inisialisasi array \$callback. Kemudian lakukan pengecekan pada elemen array \$_POST['id_user_mengikuti'], jika nilainya sama dengan 0, maka isi array \$callback dengan status gagal dan pesan belum login. Pada baris 5 – 8, lakukanlah inisialisasi \$pengguna menggunakan fungsi getPenggunaById dengan memasukan argumen \$_POST['id_user_mengikuti']. Kemudian lakukanlah pengecekan \$_POST['id_bio'], jika nilai \$_POST['id_bio'] sama dengan \$pengguna->ID, maka isi \$callback dengan status gagal dan pesan bahwa dia adalah penulisnya. Pada baris 9 – 11, lakukanlah pengecekan pada \$pengguna-ID. Jika \$pengguna->ID lebih besar daripada 0, maka lakukanlah inisialisasi pada variabel \$mengikuti menggunakan fungsi getMengikuti() dengan memasukan argumen \$pengguna->ID. Pada baris 12 – 16, dilakukan pengecekan pada variabel \$mengikuti. Jika \$mengikuti bernilai kosong, maka jalankan beberapa intruksi secara berurutan. Buat akun mengikuti menggunakan fungsi buat_mengikuti() dengan memasukan argumen \$pengguna->ID. Kemudian inisialisasi \$akun mengikuti menggunakan fungsi getmengikuti dengan memasukan argumen \$pengguna->ID. Buatlah akun detail mengikuti menggunakan fungsi tambah_detailmengikuti() dengan memasukan argumen

`$_POST['id_bio']` dan `$akun_mengikuti->ID_MENGIKUTI`. Isilah `$callback` dengan status sukses dan pesan berhasil mengikuti.

Segmen Program 7.15 Follow-Unfollow Account Bagian 2

```

01:     }else{
02:         $is_detailmengikuti = $this->MyModel->
03:             getdetailmengikuti($_POST['id_bio'], $mengikuti->
04:                 ID_MENGIKUTI);
05:         if(!empty($is_detailmengikuti)){
06:             if($is_detailmengikuti->AKTIF=="Y"){
07:                 if($this->MyModel->perbarui_detailmengikuti(
08:                     $is_detailmengikuti->
09:                         ID_DETAILMENGIKUTI, "Y")>0){
10:                 $total = $mengikuti->TOTAL_MENGIKUTI - 1;
11:                 $this->MyModel->perbarui_total_mengikuti(
12:                     $total, $mengikuti->ID_MENGIKUTI);
13:                 $callback = array('status'=>'sukses',
14:                     'pesan'=>'Batal mengikuti!');
15:             }
16:         }
17:     }else{
18:         $this->MyModel->tambah_detailmengikuti(
19:             $_POST['id_bio'], $mengikuti->ID_MENGIKUTI);
20:         $total = $mengikuti->TOTAL_MENGIKUTI + 1;
21:         $this->MyModel->perbarui_total_mengikuti($total,
22:             $mengikuti->ID_MENGIKUTI);
23:         $callback = array('status'=>'sukses',
24:             'pesan'=>'Berhasil mengikutinya!');
25:     }
26: }
27: }
28: return $callback;
29: }
```

Pada baris 1 – 9, jika `$pengguna->ID` lebih besar daripada 0, maka lakukanlah inisialisasi `$is_detailmengikuti` menggunakan fungsi `getdetailmengikuti()` dengan memasukan argumen `$_POST['id_bio']` dan

\$mengikuti->ID_MENGIKUTI. Kemudian lakukanlah pengecekan pada \$is_detailmengikuti. Jika \$is_detailmengikuti tidak bernilai kosong, maka lakukan pengecekan apakah \$is_detailmengikuti->AKTIF sama dengan “Y”. Jika benar, maka periksalah apakah fungsi perbarui_detailmengikuti() dengan argumen \$is_detailmengikuti->ID_MENGIKUTI dan argumen “Y” mengembalikan nilai lebih besar daripada 0. Jika iya, maka lakukan beberapa instruksi. Inisialisasi variabel \$total dengan menghitung \$mengikuti->TOTAL_MENGIKUTI dikurangi 1. Perbarui total mengikuti menggunakan fungsi perbarui_total_mengikuti() dengan memasukan argumen \$total dan argumen \$mengikuti->ID_MENGIKUTI. Isilah \$callback dengan status sukses dan pesan batal mengikuti.

Pada baris 10 – 16, jika \$is_detailmengikuti->AKTIF tidak sama dengan “Y”, maka jalankan instruksi pengecekan. Lalu lakukan pengecekan apakah fungsi perbarui_detailmengikuti() dengan memasukan argumen \$is_detailmengikuti->ID_DETAILMENGIKUTI dan argumen “N” menghasilkan kembalian lebih besar daripada 0. Jika iya, inisialisasi \$total dengan menghitung \$mengikuti->TOTAL_MENGIKUTI ditambah 1. Perbarui akun total mengikuti menggunakan fungsi perbarui_total_mengikuti() dengan memasukan argumen \$total dan \$mengikuti->ID_MENGIKUTI. Isilah \$callback dengan status sukses dan pesan mengikuti kembali.

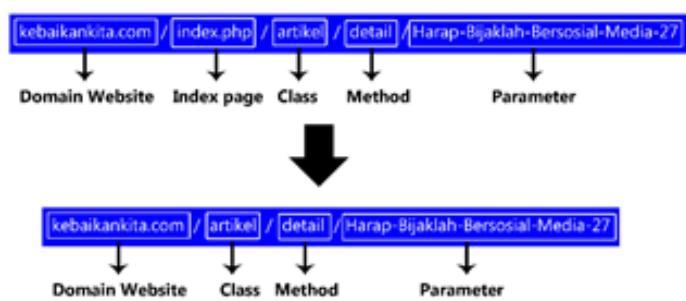
Pada baris 17 – 22, jika \$is_detailmengikuti tidak sama dengan kosong, maka tambalah detail mengikuti menggunakan fungsi tambah_detailmengikuti dengan memasukan argumen \$_POST['id_bio'] dan argumen \$mengikuti->ID_MENGIKUTI. Inisialisasi \$total dengan menghitung \$mengikuti->TOTAL_MENGIKUTI ditambah 1. Perbaruilah total mengikuti menggunakan fungsi perbarui_total_mengikuti() dengan memasukan argumen \$total dan \$mengikuti->ID_MENGIKUTI. Isilah \$callback dengan status sukses dan pesan berhasil mengikuti. Pada baris 23 – 29, jika \$pengguna->ID tidak lebih besar daripada 0, maka isilah \$callback dengan status gagal dan pesan terjadi kesalahan. Terakhir fungsi mengikuti() mengembalikan \$callback sehingga proses selesai.

7.2.6 Segmen Program Pretty URL

Pada sub bab ini dijelaskan bagaimana mengatur *pretty URL* dalam pengembangan sistem manajemen konten. Dengan penggunaan *pretty URL*, alamat atau URL situs menjadi lebih bersih. Berikut dijelaskan tahapan-tahapan pembuatan *pretty URL* pada kerangka kerja Codeigneter: pertama, Codeigneter bekerja menggunakan pendekatan *segment-based* dibandingkan *query string*. Struktur URL dalam Codeigneter dipisahkan berdasarkan segmen URI. Setiap segmen merepresentasikan struktur Codeigneter yakni:

domain.com/index.php/class/method/parameter-1/parameter-2/parameter-n.

Selengkapnya ditunjukan pada gambar 7.1 di bawah ini:



**Gambar 7.1
Pretty URL**

CI memberikan keluasaan bagi pengembang untuk mengatur URL. Dalam hal ini, `base_URL` dalam URL CI bisa diatur supaya mengikuti nama domainnya. Caranya, masuk folder application, lalu masuk folder config. Kemudian edit file `config.php`. Lalu edit `base_url`-nya pada baris `$config['base_url']` untuk kemudian diubah nilainya menjadi `$config['base_url'] = 'https://kebaikankita.com'`. Base URL hendaknya disesuaikan dengan nama domain websitenya, dalam hal ini, nama website pada tugas akhir ini adalah `https://kebaikankita.com`.

Kedua, CI juga memberikan kemudahan bagi pengembangan untuk menghilangkan `index.php` pada URL default-nya. Caranya, bisa diatur pada file `config.php`. Kemudian carilah baris `$config['index_page']='index.php'`. Lalu ubah nilainya menjadi `$config['index_page']=''`. Dengan demikan, member tidak perlu menambahkan `index.php` dalam mengakses halaman *website*.

Ketiga, buatlah nilai parameter unik untuk URL judul artikel. URL Judul dibuat berdasarkan judul artikel yang dipublikasi. Lalu satukan judul artikel per kata menggunakan tanda “-“ dan diakhiri dengan nomer index artikel. Selengkapnya ditunjukkan pada segmen program 7.16 sebagai berikut:

Segmen Program 7.16 Pembuatan Pretty URL

```
01: public function artikel_buat($data){
02:     $jumlah_data = $this->totalArtikelIndex();
03:     $artikelke = $jumlah_data + 1;
04:     $judulku = preg_replace("/[^a-zA-Z0-9]/", " ", 
05:                           $data['judul']);
06:     $judulku = str_replace("/", "-", $judulku);
07:     $judulUrl = $judulku . '-' . $artikelke;
```

Pada baris 1 – 3, sistem membuat nomer unik dari URL judul artikel dengan cara mengambil total artikel dari function totalArtikelIndex(), kemudian ditambah satu untuk selanjutnya hasilnya ditampung pada variabel \$artikelke. Selanjutnya pada baris 4 – 6, judul artikel dipisah berdasarkan spasi menjadi array menggunakan function preg_replace() untuk selanjutnya ditampung pada \$judulku. Lalu data dari \$judulku disatukan dengan penyambung tanda “-“ menggunakan function str_replace. Lalu \$judulku disatukan dengan nomer unik URL pada \$artikelke, sehingga menghasilkan judul URL yang ditampung pada variabel \$judulUrl. Kemudian nilai \$judulUrl disimpan pada kolom JUDUL_URL di tabel relasi posting.

7.2.7 Segmen Program Laporan Artikel

Segmen program laporan berguna untuk menyajikan laporan terkait perkembangan artikel. Segmen program ini dibuat dengan memanfaatkan *library* jQuery sebagai *framework* javascript dan Chartjs sebagai *tool* pembuat grafik laporan. Dengan penggunaan Chartjs, pembuatan grafik menjadi lebih ringkas dan mudah. Berikut dijelaskan tentang segmen program laporan:

Segmen Program 7.17 Laporan Artikel Terpopuler

```
01: $('#prosesartikelpopuler').on('click', function(){
02:     const idx = "<?php echo $_SESSION['user_id'];?>";
```

Segmen Program 7.17 (Lanjutan)

```

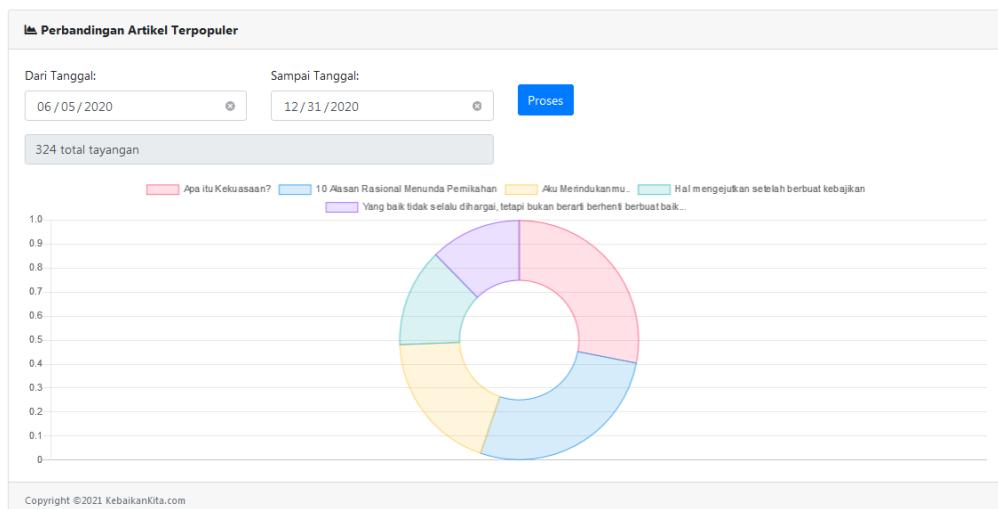
03:     let tglawal = $('#tanggalawalartikelpopuler').val();
04:     let tglakhir = $('#tanggalakhirartikelpopuler').val();
05:     $.ajax({
06:         url: BASE_URL + 'Artikel/laporan_top_artikel',
07:         data: {idx : idx, tglawal : tglawal, tglakhir : tglakhir},
08:         method: 'post',
09:         dataType: 'json',
10:         success: function(response){
11:             $('#statistik-top-artikel').remove();
12:             $('#statistik-by-topartikel').append(
13:                 '<canvas id="statistik-top-artikel" width="100%"'
14:                 'height="30"></canvas>');
15:             let $d = $('#statistik-top-artikel');
16:             let cty = $d.get(0).getContext('2d');
17:             laporan_top(cty, response.artikel, response.view,
18:             response.judul);
19:             $('#totaltayanganpopuler').val(response.total
20:                 + " total tayangan");
21:         },
22:         error: function (xhr, ajaxOptions, thrownError) {
23:             alert(xhr.responseText)
24:         } });
25:     });

```

Pada baris 1 – 4, pembuatan laporan dimulai dengan inisialisasi variabel idx untuk id_user yang diambil dari session \$_SESSION['user_id']. Lalu menginisialisasi variabel tglawal untuk rentang tanggal awal dari tayangan artikel dan menginisialisasi variabel tglakhir untuk rentang tanggal akhir dari tayangan artikel. Pada 5 – 9 baris, dengan memanfaatkan *library* jQuery menggunakan fitur ajax, data id_user yang berada pada variabel idx, tanggal awal yang berada pada variabel tglawal, tanggal akhir yang berada pada variabel tglakhir dikirim ke URL BASE_URL + Artikel/laporan_top_artikel dengan metode POST dan tipe data json. Base_url di sini merupakan alamat situs yakni <https://kebaikankita.com>.

Pada baris 10 – 17, jika pengiriman berhasil, sistem akan merespon dengan mengirimkan callback berupa data laporan artikel terpopuler. Lalu elemen dengan id #statistik-top-artikel di-remove dengan memanfaatkan manipulasi DOM. Peresetan elemen #statistik-top-artikel berfungsi untuk mereset laporan yang akan diperbarui setiap kali user melakukan pembuatan laporan. Kemudian tambahkan elemen <canvas id="statistik-top-artikel" width="100%" height="30"></canvas> pada elemen dengan id #statistik-by-topartikel. Elemen canvas berguna sebagai tempat menampilkan laporan artikel. Lalu hasilkan laporan artikel terpopuler

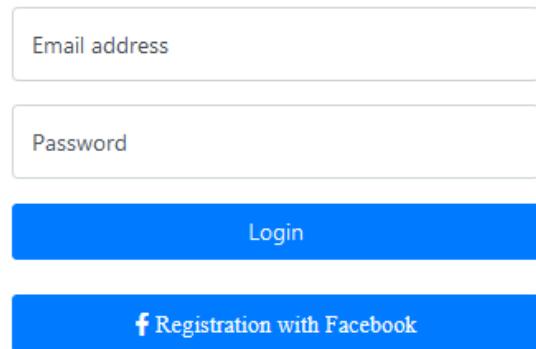
menggunakan fungsi laporan_top(cty, response.artikel, response.view, response.judul). menggunakan fungsi tersebut, laporan akan dihasilkan memanfaatkan *library* Chartjs. Parameter cty diambil dari nilai konteks elemen #statistik-top-artikel dengan menfaatkan *method* .get(0).getContext('2d') pada variabel \$d. Lalu lakukan pembaruan terhadap total tayangan artikel dengan memanipulasi DOM pada elemen \$('#totaltayanganpopuler').val(response.total + " total tayangan"). Pada baris 18 – 21, jika sistem mengalami kesalahan dalam pengiriman data, maka tampilkan pesan errornya. Jika segmen program ini berjalan dengan baik, maka sistem akan menampilkan laporan artikel terpopuler sebagai berikut:



Gambar 7.2
Laporan Artikel Terpopuler Memanfaatkan Chartjs

7.2.8 Segmen Program Registrasi Akun

Segmen program registrasi berguna untuk mendaftarkan pengguna ke sistem sebagai member dari sistem. Segmen program ini memanfaatkan *library* Ion Auth dan Facebook API. Penggunaan *library* Ion Auth dan Facebook API memudahkan dalam proses pendaftaran akun secara ringkas. Untuk registrasi akun dimulai dengan, pertama, pengguna harus menekan tombol *registration with facebook* sebagaimana ditunjukkan pada gambar 7.3 sebagai berikut:



**Gambar 7.3
Registrasi Dengan Facebook**

Kemudian sistem akan mengarahkan user ke halaman konfirmasi facebook. Member harus memasukan password-nya untuk mengkonfirmasi izin pengambilan data profilnya sebagai prasyarat pendaftaran member. Jika disetujui, facebook API akan mengambil data profil user. Selanjutnya data profil tersebut digunakan sebagai data registrasi akun member.



**Gambar 7.4
Konfirmasi Perizinan Akun Facebook**

Setelah konfirmasi berhasil, maka data profil akan dikirim ke sisi *front end* pada web browser user. Tepatnya pada halaman Javascript pada sisi web browser. Selanjutnya data profil itu akan diolah pada sisi halaman *front end* untuk selanjutnya dikirim ke *back end* pada situs kebaikankita.com. Berikut dijelaskan tentang pengambilan data profil facebook memanfaatkan Facebook API. sebagai berikut:

Segmen Program 7.18 Mengambil Data Profil Menggunakan Facebook API

```

01: function getFbUserData() {
02:     FB.api('/me', {locale: 'en_US', fields: 'id,
03:         first_name,last_name,email,link,gender,locale,picture'},
04:         function (response) {
05:             var xhr = new XMLHttpRequest();
06:             var url = "https://kebaikankita.com/Auth/loginfb";
07:             xhr.open("POST", url, true);
08:             xhr.setRequestHeader("Content-Type",
09:                 "application/json");
10:             xhr.onreadystatechange = function () {
11:                 let respon = JSON.parse(xhr.responseText);
12:                 if (xhr.readyState === 4 && xhr.status === 200) {
13:                     if(respon.status == "sukses"){
14:                         alert("Login berhasil!");
15:                         window.location = "https://kebaikankita.com/admin";
16:                     }
17:                 }
18:             var data = JSON.stringify({ "email": response.email,
19:                 "first_name": response.first_name,
19:                 "last_name" : response.last_name});
19:             xhr.send(data);
19:         } );

```

Pada baris 1 – 2, segmen program registrasi dengan akun facebook dimulai ketika user menekan tombol registration with facebook, lalu segmen program akan memanggil fungsi facebook API pada fungsi FB.api('/me', {locale: 'en_US', fields: 'id,first_name,last_name,email,link,gender,locale,picture'}). Dalam fungsi tersebut, data profil akun facebook diambil meliputi id, nama depan, nama belakang, email, link profil, gender, lokasi, dan gambar. Dalam registrasi akun, data nama depan, nama belakang dan e-mail sudah cukup untuk melakukan pendaftaran ke sistem kebaikankita.com. Pada baris 3 - 18, dengan menfaatkan ajax, data profil meliputi nama depan, nama belakang, e-mail dikirim ke situs pada alamat https://kebaikankita.com/Auth/loginfb. Alamat tersebut merupakan URL registrasi member menggunakan akun facebook pada situs kebaikankita.com. Jika berhasil, maka sistem akan menampilkan pesan sukses dan mengarahkan user ke halaman admin. Berikutnya dijelaskan segmen program registrasi akun pada menggunakan Ion Auth sebagai berikut:

Segmen Program 7.19 Registrasi Member Menggunakan Ion Auth

```
01: public function loginfb(){
```

Segmen Program 7.19 (Lanjutan)

```

02: header("Content-Type: application/json");
03: $v = json_decode(stripslashes(
04:     file_get_contents("php://input")));
05: $firstname = $v->{'first_name'};
06: $lastname = $v->{'last_name'};
07: $email = $v->{'email'};
08: $password = "Facebook-2020";
09: $group = array('2'); // Sets user to 'user'.
10: if (!$this->ion_auth->username_check($email)) {
11:     $group_name = 'users';
12:     $additional_data = array(
13:         'first_name' => $firstname,
14:         'last_name' => $lastname,
15:         'company' => "organisasi",
16:         'phone' => "08xxxxxxxxxx ");
17:     $this->ion_auth->register($email, $password, $email,
18:         $additional_data, $group);
19:     $id_users = $this->MyModel->getidprofil($email);
20:     $iduser = $id_users->id;
21:     $data_pengguna = array(
22:         'firstname' => $firstname,
23:         'lastname' => $lastname,
24:         'company' => "organisasi",
25:         'phone' => "+628XXXXXXXXX",
26:         'email' => $email,
27:         'password' => $password,
28:         'id_users' => $iduser);
29:     $this->MyModel->akun_buat($data_pengguna);
30:     $data = array('active' => "1");
31:     $this->ion_auth->update($iduser, $data);
32:     $remember = TRUE;
33:     $this->ion_auth->login($email, $password, $remember);
34:     $callback = array(
35:         'status'=>'sukses',
36:         'pesan'=>"Berhasil registrasi!");
37:     echo json_encode($callback);
38: } else{
39:     $remember = TRUE;
40:     $this->ion_auth->login($email, $password, $remember);
41:     $callback = array(
42:         'status'=>'sukses',
43:         'pesan'=>"Berhasil login!");
44:     echo json_encode($callback);
45: }
46: }

```

Pada baris 1 – 8, dilakukan inisialisasi pada masing-masing variabel \$firstname, \$lastname, \$email, \$password, \$group dengan mengambil hasil data dari \$v yang merupakan data profil facebook pada segmen program sebelumnya. \$group berbentuk array dengan nilai 2 yang berarti akun member di-setting sebagai member. Kemudian pada baris 9, dilakukan pengecekan terhadap ketersediaan

username menggunakan fungsi `username_check()` dari *library* Ion Auth. Jika username telah terdaftar pada sistem maka pada baris 20 – 27 lakukan login menggunakan fungsi `login()` dari *library* ion_auth kemudian sistem akan mengembalikan pesan sukses, tanda login berhasil dilakukan. Namun jika username belum terdaftar di sistem, maka pada baris 10 – 19 lakukan pendaftaran pengguna sebagai member dengan mengisi profilnya masing-masing `first_name` diambil dari nilai `$firstname`, `last_name` diambil dari `$lastname`, `company` diisi “organisasi” dan `phone` diisi “08xxxxxxxxx”, serta `$group` diisi array bernilai 2 untuk mendaftar sebagai member. Kemudian lakukan registrasi menggunakan fungsi `register()` memanfaatkan *library* Ion Auth. Kemudian buat kembali akun cadangan menggunakan fungsi `akun_buat()` dengan memanggil kelas MyModel. Lalu aktivisi akun member dengan mengisi `$data` bernilai array “active” = 1, kemudian lakukan pembaruan data profil member menggunakan fungsi `update()` memanfaatkan *library* Ion Auth. Terakhir sistem akan mengembalikan `$callback` pesan sukses, sehingga proses registrasi selesai.

BAB VIII

PENGUJIAN

Pada Bab VIII dijelaskan tentang pengujian sistem manajemen konten dan penyaring komentar *cyberbullying*. Adapun pengujiannya melibatkan pengujian fungsionalitas sistem, pengujian model, dan evaluasi *website*. Pengujian fungsionalitas sistem menggunakan *black box* untuk mengetahui fitur-fitur sistem bisa berjalan baik atau tidak. Sementara pengujian model komentar *cyberbullying* akan diuji seberapa besar tingkat akurasi, *error*, *precision*, *recall*, *specificity* dan *f-score* dalam mengklasifikasikan komentar menjadi sentimen *cyberbullying* atau *non-cyberbullying*. Berikut penjelasan selengkapnya:

8.1 Pengujian Fungsionalitas

Pada sub bab ini dijelaskan tentang beberapa pengujian fungsionalitas menggunakan *black box*. Pengujian ini dilakukan dengan kesesuaian antara reaksi sistem terhadap inputan. Adapun fitur-fitur yang diuji di antaranya manajemen artikel, komentar, rating, pelaporan komentar, peringatan sentimen, manajemen galeri, konfigurasi pengguna. Berikut penjelasan selengkapnya:

8.1.1 Pengujian Fungsionalitas Login dan Registrasi

Pada sub bab ini dijelaskan pengujian pada fitur login dan registrasi untuk mengetahui fungsionalitas dari fitur login dan registrasi. Pengujian dilakukan dengan menginput data ke form login dan form registrasi, lalu melihat reaksi sistemnya apakah sesuai dengan skenario yang diharapkan atau tidak. Jika hasil yang dikeluarkan sesuai dengan skenario, maka pengujian fungsionalitas berhasil. Jika hasil tidak sesuai dengan output yang diharapkan maka selanjutnya dilakukan tindakan perbaikan terhadap fungsionalitas dari fitur sistem yang bermasalah hingga fitur tersebut bisa berjalan secara normal.

Pada tabel 8.1 dijelaskan skenario pengujian fungsionalitas pada manajemen artikel. Pada tabel tersebut terdapat 8 skenario pengujian fungsionalitas pada login dan registrasi. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya sebagai berikut:

Tabel 8.1
Pengujian Fungsionalitas Manajemen Artikel

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Menginput username dan password secara benar ke sistem login.	Sistem mengarahkan member/ admin ke halaman backend.	Sistem mengarahkan member/ admin ke halaman backend.	Sesuai
2	Menginput username dan password secara salah ke sistem login.	Sistem memberikan pesan error bahwa login bermasalah.	Sistem memberikan pesan error bahwa login bermasalah.	Sesuai
3	Menekan tombol logout.	Sistem keluar dan tidak dapat mengakses halaman backend.	Sistem keluar dan tidak dapat mengakses halaman backend.	Sesuai
4	Mengosongi beberapa form registrasi lalu menekan tombol registrasi.	Sistem mengeluarkan pesan error bahwa ada field dari form registrasi yang kosong.	Sistem mengeluarkan pesan error bahwa ada field dari form registrasi yang kosong.	Sesuai
5	Mengisi form registrasi dengan memasukan password dan konfirmasi password secara berbeda.	Sistem mengeluarkan pesan error bahwa password dan konfirmasi tidak sama.	Sistem mengeluarkan pesan error bahwa password dan konfirmasi tidak sama.	Sesuai

**Tabel 8.1
(Lanjutan)**

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
6	Mengisi form registrasi dengan data pengguna yang sudah terdaftar di sistem.	Sistem mengeluarkan pesan error bahwa akun terkait sudah terdaftar di sistem.	Sistem mengeluarkan pesan error bahwa akun terkait sudah terdaftar di sistem.	Sesuai
7	Mengisi form registrasi dengan data pengguna secara lengkap dan benar.	Sistem mengeluarkan pesan sukses bahwa akun berhasil didaftarkan di sistem.	Sistem mengeluarkan pesan sukses bahwa akun berhasil didaftarkan	Sesuai
8	Menekan tombol reset password.	Sistem mengarahkan ke halaman reset password.	Sistem tidak mengarahkan ke halaman reset password.	Tidak sesuai

Berdasarkan delapan skenario di atas, 7 skenario berhasil. Sedangkan satu skenario gagal yakni studi kasus nomer 8 mengenai menekan tombol reset password. Berikutnya dilakukan perbaikan pada fitur peresetan password hingga error berhasil diperbaiki. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.2 Pengujian Fungsionalitas Konfigurasi Akun

Pada sub bab ini dijelaskan pengujian pada fitur konfigurasi akun untuk mengetahui fungsionalitas dari fitur konfigurasi. Pengujian dilakukan dengan menginput biodata ke form konfigurasi akun, lalu melihat reaksi sistemnya apakah sesuai dengan skenario yang diharapkan atau tidak. Jika sistem menghasilkan output yang sesuai dengan skenario maka pengujian fungsionalitas berhasil. Sebaliknya jika sistem tidak menghasilkan output yang diharapkan, maka selanjutnya dilakukan perbaikan fungsionalitas hingga fitur berjalan normal.

Pada tabel 8.2 dijelaskan skenario pengujian fungsionalitas pada konfigurasi akun. Pada tabel tersebut terdapat 3 skenario pengujian fungsionalitas konfigurasi akun. Masing-masing skenario memiliki hasil yang diharapkan dan

hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya ditunjukan pada tabel 8.2 sebagai berikut:

Tabel 8.2
Pengujian Fungsionalitas Konfigurasi Akun

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Mengubah biodata pengguna, lalu menekan tombol simpan.	Sistem menyimpan biodata dan mengeluarkan pesan sukses biodata telah tersimpan.	Sistem menyimpan biodata dan mengeluarkan pesan sukses biodata telah tersimpan.	Sesuai
2	Mengupload foto akun, lalu menekan tombol simpan.	Sistem menyimpan foto akun dan mengeluarkan pesan foto berhasil diganti.	Sistem menyimpan foto akun dan mengeluarkan pesan foto berhasil diganti.	Sesuai
3	Mengosongi salah satu field dari form akun, lalu menekan tombol simpan.	Sistem mengeluarkan pesan error bahwa salah satu field ada yang kosong.	Sistem mengeluarkan pesan error bahwa salah satu field ada yang kosong.	Sesuai

Berdasarkan skenario pengujian tabel 8.2, diketahui bahwa ketiga skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan perbaikan. Kecuali jika terjadi error di luar kasus ketiga skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.3 Pengujian Fungsionalitas Manajemen Galeri

Pada sub bab ini dijelaskan pengujian pada fitur manajemen galeri. Pengujian dilakukan dengan mengupload gambar sesuai dengan skenario tertentu

untuk melihat reaksi sistem. Jika sistem mengeluarkan output yang diharapkan, maka pengujian fungsionalitas manajemen galeri berhasil. Jika sistem tidak mengeluarkan output yang tidak sesuai skenario, kemudian dilakukan perbaikan pada fitur manajemen galeri hingga bisa berjalan normal.

Pada tabel 8.3 dijelaskan skenario pengujian fungsionalitas pada manajemen galeri. Pada tabel tersebut terdapat 6 skenario pengujian fungsionalitas pada manajemen galeri. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya ditunjukkan pada tabel 8.3 sebagai berikut:

Tabel 8.3
Pengujian Fungsionalitas Manajemen Galeri

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Mengupload gambar >1 MB.	Sistem mengeluarkan pesan error bahwa gambar melebihi 1 MB.	Sistem mengeluarkan pesan error bahwa gambar melebihi 1 MB.	Sesuai
2	Mengosongi field upload, lalu menekan tombol simpan.	Sistem mengeluarkan pesan error bahwa terdapat field yang kosong.	Sistem mengeluarkan pesan error bahwa terdapat field yang kosong.	Sesuai
3	Mengupload gambar < 1MB dan mengisi field-field galeri secara benar, lalu menekan tombol simpan.	Sistem menyimpan data galeri dan menampilkan pesan sukses berhasil diupload.	Sistem menyimpan data galeri dan menampilkan pesan sukses berhasil diupload.	Sesuai
4	Mengakses halaman galeri.	Sistem menampilkan daftar galeri.	Sistem menampilkan daftar galeri.	Sesuai
5	Mengakses detail salah satu galeri.	Sistem menampilkan gambar, nama dan keterangan galeri.	Sistem menampilkan gambar, nama dan keterangan galeri.	Sesuai

Berdasarkan skenario pengujian tabel 8.3, diketahui bahwa keenam skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan perbaikan. Kecuali jika terjadi error di luar kasus keenam skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.4 Pengujian Fungsionalitas Manajemen Artikel

Pada sub bab ini dijelaskan pengujian pada fitur manajemen artikel. Pengujian dilakukan dengan menginput data artikel ke form artikel. Lalu dilihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas fitur hingga fitur bisa berjalan normal sesuai dengan skenario.

Pada tabel 8.4 dijelaskan skenario pengujian fungsionalitas pada manajemen artikel. Pada tabel tersebut terdapat 11 skenario pengujian fungsionalitas pada manajemen artikel. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya ditunjukkan pada tabel 8.4 sebagai berikut:

Tabel 8.4
Pengujian Fungsionalitas Manajemen Artikel

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Mengisi form artikel secara benar, lalu menekan tombol simpan.	Sistem menyimpan artikel dan menampilkan pesan sukses.	Sistem menyimpan artikel dan menampilkan pesan sukses.	Sesuai
2	Mengosongi field dari form artikel.	Sistem menampilkan pesan error.	Sistem menampilkan pesan error.	Sesuai
3	Mengisi field form tidak sesuai ketentuan.	Sistem menampilkan pesan error.	Sistem menampilkan pesan error.	Sesuai

**Tabel 8.4
(Lanjutan)**

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
4	Menghapus artikel.	Sistem tidak menampilkan data artikel yang terhapus.	Sistem tidak menampilkan data artikel yang terhapus.	Sesuai
5	Mengakses detail artikel.	Sistem menampilkan detail artikel pada halaman detail artikel di frontend.	Sistem menampilkan detail artikel pada halaman detail artikel di frontend.	Sesuai
6	Mengedit artikel, lalu menekan tombol simpan.	Sistem menyimpan perubahan edit dan menampilkan pesan sukses.	Sistem menyimpan perubahan edit dan menampilkan pesan sukses.	Sesuai
7	Merefresh halaman detail artikel.	Sistem memperbarui tampilan artikel dan menambah jumlah tayangan.	Sistem memperbarui tampilan artikel dan menambah jumlah tayangan.	Sesuai
8	Mengedit artikel, lalu menekan tombol simpan.	Sistem menyimpan perubahan edit dan menampilkan pesan sukses.	Sistem menyimpan perubahan edit dan menampilkan pesan sukses.	Sesuai
9	Memberikan rating artikel menggunakan akun penulis.	Sistem menampilkan pesan error tidak boleh menggunakan akun penulis untuk rating.	Sistem menampilkan pesan error tidak boleh menggunakan akun penulis untuk rating.	Sesuai
10	Memberikan rating artikel menggunakan akun selain penulis.	Sistem menyimpan rating dan menampilkan pesan sukses.	Sistem menyimpan rating dan menampilkan pesan sukses.	Sesuai

Berdasarkan skenario pengujian tabel 8.4, diketahui bahwa kesebelas skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan perbaikan. Kecuali jika terjadi error di luar kasus kesebelas skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.5 Pengujian Fungsionalitas Manajemen Komentar

Pada sub bab ini dijelaskan pengujian pada fitur manajemen komentar. Pengujian dilakukan dengan menginput data komentar pada form komentar yang ada di artikel. Lalu melihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas fitur manajemen komentar hingga fitur bisa berjalan normal sesuai dengan skenario.

Pada tabel 8.5 dijelaskan skenario pengujian fungsionalitas pada manajemen komentar. Pada tabel tersebut terdapat 6 skenario pengujian fungsionalitas pada manajemen komentar. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya ditunjukkan pada tabel 8.5 sebagai berikut:

Tabel 8.5
Pengujian Fungsionalitas Manajemen Komentar

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Memberikan komentar pada salah satu artikel.	Sistem menyimpan komentar, menampilkan pesan sukses dan menampilkan komentar pada artikel,	Sistem menyimpan komentar, menampilkan pesan sukses.	Sesuai
2	Membalas komentar pada salah satu komentar di artikel.	Sistem menyimpan balasan komentar, menampilkan pesan sukses dan menampilkan balasan komentar di artikel.	Sistem menyimpan balasan komentar, menampilkan pesan sukses dan menampilkan balasan komentar di artikel.	Sesuai
3	Menghapus komentar dan balasan komentar.	Sistem tidak menampilkan komentar dan balasan komentar di artikel.	Sistem tidak menampilkan komentar dan balasan komentar di artikel.	Sesuai

**Tabel 8.5
(Lanjutan)**

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
4	Memberikan komentar yang berisi <i>cyberbullying</i> .	Sistem menampilkan pesan peringatan bahwa pesan tersebut mengandung <i>cyberbullying</i> .	Sistem menampilkan pesan peringatan bahwa pesan tersebut mengandung <i>cyberbullying</i> .	Sesuai
5	Melaporkan komentar yang mengandung <i>cyberbullying</i> secara manual.	Sistem menyimpan laporan dan menampilkan pesan sukses telah mengirimkan laporan.	Sistem menyimpan laporan dan menampilkan pesan sukses telah mengirimkan laporan.	Sesuai
6	Menghapus laporan yang dikirim member.	Sistem menampilkan pesan sukses telah dihapus dan tidak menampilkan komentar pada artikel.	Sistem menampilkan pesan sukses telah dihapus dan tidak menampilkan komentar pada artikel.	Sesuai

Berdasarkan skenario pengujian tabel 8.5, diketahui bahwa keenam skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan perbaikan. Kecuali jika terjadi error di luar kasus keenam skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.6 Pengujian Fungsionalitas Manajemen Following-Follower

Pada sub bab ini dijelaskan pengujian pada fitur manajemen *following-follower*. Pengujian dilakukan dengan mencoba mengikuti akun member lainnya atau berhenti mengikuti akun member lainnya. Lalu dilihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas fitur following-follower hingga fitur bisa berjalan normal sesuai dengan skenario.

Pada tabel 8.6 dijelaskan skenario pengujian fungsionalitas pada manajemen *follower-following*. Pada tabel tersebut terdapat 3 skenario pengujian

fungsionalitas pada manajemen *follower-following*. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya ditunjukkan pada tabel 8.6 sebagai berikut:

Tabel 8.6
Pengujian Fungsionalitas Manajemen Follower-Following

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Mengikuti akun penulis menggunakan akun member lainnya.	Sistem menampilkan pesan sukses berhasil mengikuti dan daftar akun yang diikuti bertambah.	Sistem menampilkan pesan sukses berhasil mengikuti dan daftar akun yang diikuti bertambah.	Sesuai
2	Berhenti mengikuti akun penulis menggunakan akun member lainnya.	Sistem menampilkan pesan sukses berhenti mengikuti dan daftar akun yang diikuti berkurang.	Sistem menampilkan pesan sukses berhenti mengikuti dan daftar akun yang diikuti berkurang.	Sesuai
3	Mengakses daftar akun yang diikuti dan daftar akun yang pengikut.	Sistem menampilkan daftar akun yang diikuti dan pengikut.	Sistem menampilkan daftar akun yang diikuti dan pengikut.	Sesuai

Berdasarkan skenario pengujian tabel 8.6, diketahui bahwa ketiga skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan perbaikan. Kecuali jika terjadi error di luar kasus ketiga skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.7 Pengujian Fungsionalitas Notifikasi

Pada sub bab ini dijelaskan pengujian pada fitur notifikasi. Pengujian dilakukan dengan mengirimkan stimulus kepada sistem seperti membuat artikel, berkomentar, dan memberikan rating. Lalu melihat reaksi sistem apakah sesuai

dengan output skenario yang diharapkan atau tidak. Jika reaksi sistem sesuai dengan yang diharapkan dari hasil inputan, maka pengujian berhasil.

Pada tabel 8.7 dijelaskan skenario pengujian fungsionalitas pada notifikasi. Pada tabel tersebut terdapat 3 skenario pengujian fungsionalitas pada notifikasi. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya ditunjukkan tabel 8.7 sebagai berikut:

Tabel 8.7
Pengujian Fungsionalitas Notifikasi

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Membuat artikel menggunakan akun penulis.	Sistem menampilkan notifikasi artikel baru pada akun pengikut.	Sistem menampilkan notifikasi artikel baru pada akun pengikut.	Sesuai
2	Berkomentar pada artikel penulis menggunakan akun pengikut.	Sistem menampilkan notifikasi komentar baru pada akun penulis.	Sistem menampilkan notifikasi komentar baru pada akun penulis.	Sesuai
3	Mengirimkan laporan <i>cyberbullying</i> menggunakan akun pengikut.	Sistem menampilkan notifikasi laporan <i>cyberbullying</i> baru pada akun penulis.	Sistem menampilkan notifikasi laporan <i>cyberbullying</i> baru pada akun penulis.	Sesuai
4	Memberikan rating pada artikel penulis menggunakan akun pengikut.	Sistem menampilkan notifikasi rating baru pada akun penulis.	Sistem menampilkan notifikasi rating baru pada akun penulis.	Sesuai
5	Menandai notifikasi.	Sistem menghilangkan adanya notifikasi baru.	Sistem menghilangkan adanya notifikasi baru.	Sesuai

Berdasarkan skenario pengujian tabel 8.7, diketahui bahwa kelima skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan perbaikan. Kecuali jika terjadi error di luar kasus kelima skenario di atas maka akan

dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.8 Pengujian Fungsionalitas Konfigurasi Website

Pada sub bab ini dijelaskan pengujian pada fitur konfigurasi website. Pengujian dilakukan dengan mengubah konfigurasi pada website seperti mengubah nama website, meta deskripsi, meta keyword, hingga template *front end*. Lalu dilihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. . Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas fitur konfigurasi website hingga fitur bisa berjalan normal.

Pada tabel 8.8 dijelaskan skenario pengujian fungsionalitas pada konfigurasi website. Pada tabel tersebut terdapat 2 skenario pengujian fungsionalitas pada konfigurasi website. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya ditunjukkan tabel 8.8 sebagai berikut:

Tabel 8.8
Pengujian Fungsionalitas Konfigurasi Website

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Mengubah template frontend pada pilihan yang tersedia.	Sistem mengubah tampilan frontend sesuai dengan pilihan admin.	Sistem mengubah tampilan frontend sesuai dengan pilihan admin.	Sesuai
2	Mengubah identitas website seperti nama website, meta deskripsi, dan meta keyword.	Sistem menyimpan perubahan identitas website lalu menampilkan pesan sukses identitas telah disimpan.	Sistem menyimpan perubahan identitas website lalu menampilkan pesan sukses identitas telah disimpan.	Sesuai

Berdasarkan skenario pengujian tabel 8.8, diketahui bahwa kedua skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan

perbaikan. Kecuali jika terjadi error di luar kasus kedua skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.9 Pengujian Fungsionalitas Manajemen Tentang Website

Pada sub bab ini dijelaskan pengujian pada fitur manajemen tentang website. Pengujian dilakukan dengan menginput data biodata website. Lalu dilihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas fitur manajemen tentang website hingga fitur bisa berjalan normal.

Pada tabel 8.9 dijelaskan skenario pengujian fungsionalitas pada manajemen tentang website. Pada tabel tersebut terdapat 2 skenario pengujian fungsionalitas pada manajemen tentang website. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya ditunjukkan tabel 8.9 sebagai berikut:

Tabel 8.9
Pengujian Fungsionalitas Manajemen Tentang Website

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Mengosongi form tentang website, lalu menekan tombol simpan.	Sistem menampilkan pesan error bahwa terdapat field yang kosong.	Sistem menampilkan pesan error bahwa terdapat field yang kosong.	Sesuai
2	Mengubah deskripsi biodata website, lalu menekan tombol simpan.	Sistem menyimpan perubahan biodata website dan menampilkan pesan sukses telah disimpan.	Sistem menyimpan perubahan biodata website dan menampilkan pesan sukses telah disimpan.	Sesuai

Berdasarkan skenario pengujian tabel 8.8, diketahui bahwa kedua skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan

perbaikan. Kecuali jika terjadi error di luar kasus kedua skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.10 Pengujian Fungsionalitas Manajemen Kebijakan Website

Pada sub bab ini dijelaskan pengujian pada fitur manajemen kebijakan website. Pengujian dilakukan dengan menginput data kebijakan website. Lalu dilihat reaksi sistem apakah sesuai dengan *output* skenario yang diharapkan atau tidak. Jika sesuai dengan skenario, maka pengujian berhasil. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas fitur manajemen kebijakan website hingga fitur bisa berjalan normal.

Pada tabel 8.10 dijelaskan skenario pengujian fungsionalitas pada manajemen kebijakan website. Pada tabel tersebut terdapat 2 skenario pengujian fungsionalitas pada manajemen kebijakan website. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya ditunjukkan tabel 8.10 sebagai berikut:

Tabel 8.10
Pengujian Fungsionalitas Manajemen Kebijakan Website

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Mengosongi form kebijakan website, lalu menekan tombol simpan.	Sistem menampilkan pesan error bahwa terdapat field yang kosong.	Sistem menampilkan pesan error bahwa terdapat field yang kosong.	Sesuai
2	Mengubah kebijakan website, lalu menekan tombol simpan.	Sistem menyimpan perubahan dan menampilkan pesan sukses telah disimpan.	Sistem menyimpan dan menampilkan pesan sukses telah disimpan.	Sesuai

Berdasarkan skenario pengujian tabel 8.10, diketahui bahwa kedua skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan

perbaikan. Kecuali jika terjadi error di luar kasus kedua skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.11 Pengujian Fungsionalitas Manajemen FAQ

Pada sub bab ini dijelaskan pengujian pada fitur manajemen FAQ. Pengujian dilakukan dengan menginput, mengubah dan menghapus data pertanyaan dan jawaban (QnA). Lalu dilihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. Jika reaksi sistem sesuai dengan skenario, maka pengujian berhasil. Jika *output* tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas fitur manajemen FAQ hingga fitur bisa berjalan normal.

Pada tabel 8.11 dijelaskan skenario pengujian fungsionalitas pada manajemen FAQ. Pada tabel tersebut terdapat 4 skenario pengujian fungsionalitas pada manajemen FAQ. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya ditunjukkan tabel 8.11 sebagai berikut:

Tabel 8.11
Pengujian Fungsionalitas Manajemen FAQ

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Mengosongi form FAQ, lalu menekan tombol simpan.	Sistem menampilkan pesan error bahwa terdapat field yang kosong.	Sistem menampilkan pesan error bahwa terdapat field yang kosong.	Sesuai
2	Menambah data FAQ dengan menginput form FAQ, lalu simpan.	Sistem menyimpan data FAQ dan menampilkan pesan sukses.	Sistem menyimpan data FAQ dan menampilkan pesan sukses.	Sesuai
3	Mengakses detail FAQ.	Sistem menampilkan detail	Sistem menampilkan detail	Sesuai

**Tabel 8.11
(Lanjutan)**

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
		pertanyaan dan jawabannya.	pertanyaan dan jawabannya.	
4	Menghapus data FAQ.	Sistem tidak menampilkan data FAQ yang terhapus dan menampilkan pesan sukses telah dihapus.	Sistem tidak menampilkan data FAQ yang terhapus dan menampilkan pesan sukses telah dihapus.	Sesuai

Berdasarkan skenario pengujian tabel 8.11, diketahui bahwa keempat skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan perbaikan. Kecuali jika terjadi error di luar kasus keempat skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.12 Pengujian Fungsionalitas Manajemen Tagar

Pada sub bab ini dijelaskan pengujian pada fitur manajemen tagar. Pengujian dilakukan dengan menginput, mengakses, mengedit, dan menghapus tagar. Lalu melihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. Jika reaksi sistem sesuai dengan skenario, maka pengujian berhasil. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas fitur manajemen tagar hingga fitur bisa berjalan normal.

Pada tabel 8.12 dijelaskan skenario pengujian fungsionalitas pada manajemen tagar. Pada tabel tersebut terdapat 4 skenario pengujian fungsionalitas pada manajemen tagar. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya ditunjukkan tabel 8.12 sebagai berikut:

Tabel 8.12
Pengujian Fungsionalitas Manajemen Tagar

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Menambah data tagar.	Sistem menampilkan data tagar baru di daftar tagar.	Sistem menampilkan data tagar baru di daftar tagar.	Sesuai
2	Mengedit data tagar.	Sistem mengubah data tagar sesuai perubahan editnya.	Sistem mengubah data tagar sesuai perubahan editnya.	Sesuai
3	Menghapus data tagar.	Sistem tidak menampilkan data tagar di daftar tagar.	Sistem tidak menampilkan data tagar di daftar tagar.	Sesuai
4	Mengakses data tagar.	Sistem menampilkan artikel yang relevan dengan tagar.	Sistem menampilkan artikel yang relevan dengan tagar.	Sesuai

Berdasarkan skenario pengujian tabel 8.12, diketahui bahwa keempat skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan perbaikan. Kecuali jika terjadi error di luar kasus keempat skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.13 Pengujian Fungsionalitas Manajemen User

Pada sub bab ini dijelaskan pengujian pada fitur manajemen user. Pengujian dilakukan dengan mengakses dan mengubah hak akses user. Lalu dilihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas fitur manajemen user hingga fitur bisa berjalan normal.

Pada tabel 8.13 dijelaskan skenario pengujian fungsionalitas pada manajemen user. Pada tabel tersebut terdapat 2 skenario pengujian fungsionalitas pada manajemen user. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya ditunjukan tabel 8.13 sebagai berikut:

Tabel 8.13
Pengujian Fungsionalitas Manajemen User

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Mengakses daftar user.	Sistem menampilkan daftar user.	Sistem menampilkan daftar user.	Sesuai
2	Memberikan hak akses admin kepada member.	Sistem memberikan hak akses admin kepada member, sehingga member bisa mengakses fitur admin.	Sistem memberikan hak akses admin kepada member, sehingga member bisa mengakses fitur admin.	Sesuai

Berdasarkan skenario pengujian tabel 8.13, diketahui bahwa kedua skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan perbaikan. Kecuali jika terjadi error di luar kasus kedua skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.14 Pengujian Fungsionalitas Manajemen Kategori

Pada sub bab ini dijelaskan pengujian pada fitur manajemen kategori. Pengujian dilakukan dengan menginput, mengakses, mengedit, dan menghapus data kategori. Lalu dilihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas fitur manajemen kategori hingga fitur bisa berjalan normal.

Pada tabel 8.14 dijelaskan skenario pengujian fungsionalitas pada manajemen kategori. Pada tabel tersebut terdapat 4 skenario pengujian fungsionalitas pada manajemen kategori. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Sebaliknya jika pengujian hasil yang diharapkan tidak sesuai dengan keluaran tesnya maka pengujian gagal. Selengkapnya ditunjukkan tabel 8.14 sebagai berikut:

Tabel 8.14
Pengujian Manajemen Kategori

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Menambah data kategori.	Sistem menyimpan data kategori dan menampilkan pesan sukses data ditambah.	Sistem menyimpan data kategori dan menampilkan pesan sukses data ditambah.	Sesuai
2	Mengedit data kategori.	Sistem menyimpan perubahan data kategori dan menampilkan pesan sukses telah diedit.	Sistem menyimpan perubahan data kategori dan menampilkan pesan sukses telah diedit.	
3	Menghapus data kategori.	Sistem tidak menampilkan data kategori di daftar kategori.	Sistem tidak menampilkan data kategori di daftar kategori.	Sesuai
4	Mengosongi form kategori, lalu menyimpan data.	Menampilkan pesan error bahwa ada field yang kosong.	Menampilkan pesan error bahwa ada field yang kosong.	Sesuai

Berdasarkan skenario pengujian tabel 8.14, diketahui bahwa keempat skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan perbaikan. Kecuali jika terjadi error di luar kasus keempat skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.15 Pengujian Fungsionalitas Manajemen Evaluasi Website

Pada sub bab ini dijelaskan pengujian pada fitur manajemen evaluasi. Pengujian dilakukan dengan menginput, mengakses, mengedit, dan menghapus data pertanyaan. Lalu dilihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas fitur manajemen evaluasi website hingga fitur bisa berjalan normal.

Pada tabel 8.15 dijelaskan skenario pengujian fungsionalitas pada manajemen evaluasi *website*. Pada tabel tersebut terdapat 4 skenario pengujian fungsionalitas pada manajemen evaluasi *website*. Masing-masing skenario

memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Selengkapnya ditunjukkan tabel 8.15 sebagai berikut:

Tabel 8.15
Pengujian Fungsionalitas Manajemen Evaluasi Website

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Menambah data pertanyaan.	Sistem menyimpan data pertanyaan dan menampilkan pesan sukses data pertanyaan telah ditambah.	Sistem menyimpan data pertanyaan dan menampilkan pesan sukses data pertanyaan telah ditambah.	Sesuai
2	Mengakses daftar pertanyaan.	Sistem menampilkan daftar pertanyaan.	Sistem menampilkan daftar pertanyaan.	Sesuai
3	Mengedit data pertanyaan.	Sistem menyimpan perubahan pertanyaan dan menampilkan pesan data telah diubah.	Sistem menyimpan perubahan pertanyaan.	Sesuai
4	Menghapus data pertanyaan	Sistem tidak menghilangkan data pertanyaan dari daftar pertanyaan.	Sistem tidak menghilangkan data pertanyaan dari daftar pertanyaan.	Sesuai

Berdasarkan skenario pengujian tabel 8.15, diketahui bahwa keempat skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan perbaikan. Kecuali jika terjadi error di luar kasus keempat skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.16 Pengujian Fungsionalitas Dashboard

Pada sub bab ini dijelaskan pengujian pada fitur dashboard website. Pengujian dilakukan dengan mengakses beberapa laporan grafik artikel yang disediakan. Lalu dilihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. Jika output tidak sesuai dengan skenario, maka dilakukan

perbaikan pada fungsionalitas fitur dashboard website hingga fitur bisa berjalan normal.

Pada tabel 8.16 dijelaskan skenario pengujian fungsionalitas pada dashboard. Pada tabel tersebut terdapat 5 skenario pengujian fungsionalitas pada dashboard. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Selengkapnya ditunjukkan tabel 8.16 sebagai berikut:

Tabel 8.16
Pengujian Fungsionalitas Dashboard

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Mengakses laporan perkembangan tayangan seluruh artikel.	Sistem menampilkan grafik laporan perkembangan tayangan seluruh artikel.	Sistem menampilkan grafik laporan perkembangan tayangan seluruh artikel.	Sesuai
2	Mengakses laporan perbandingan antar 5 artikel populer.	Sistem menampilkan grafik perbandingan antar 5 artikel terpopuler.	Sistem menampilkan grafik perbandingan antar 5 artikel terpopuler.	Sesuai
3	Mengakses laporan perkembangan tayangan per artikel.	Sistem menampilkan grafik perkembangan tayangan per artikel.	Sistem menampilkan grafik perkembangan tayangan per artikel.	Sesuai
4	Mengakses laporan perbandingan 5 top rating artikel.	Sistem menampilkan grafik perbandingan 5 top rating artikel.	Sistem menampilkan grafik perbandingan 5 top rating artikel.	Sesuai
5	Mengakses laporan perkembangan rating per artikel.	Sistem menampilkan grafik perkembangan rating artikel.	Sistem menampilkan grafik perkembangan rating artikel.	Sesuai

Berdasarkan skenario pengujian tabel 8.16, diketahui bahwa kelima skenario pengujian berhasil dilaksanakan. Dengan demikian tidak dilakukan tindakan perbaikan. Kecuali jika terjadi error di luar kasus kelima skenario di atas maka akan dilakukan perbaikan terhadap fungsionalitas sistem. Dengan demikian, diharapkan sistem menjadi berjalan dengan baik.

8.1.17 Pengujian Fungsionalitas Penyaring Cyberbullying

Pada sub bab ini dijelaskan pengujian penyaring komentar *cyberbullying*. Pengujian dilakukan dengan mengakses, menginput, mengedit, menghapus data yang dibutuhkan sistem penyaring komentar *cyberbullying* seperti dataset artikel, dataset komentar, dataset kamus bahasa gaul, dataset stopword, hingga pengujian model. Lalu dilihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas sistem penyaring komentar *cyberbullying* hingga sistem bisa berjalan normal.

Pada tabel 8.17 dijelaskan tentang pengujian manajemen dataset artikel. Pengujian dilakukan dengan mengakses, menginput, mengedit, menghapus dataset artikel. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas sistem penyaring komentar *cyberbullying* hingga sistem bisa berjalan normal. Pada tabel tersebut terdapat 4 skenario pengujian fungsionalitas pada manajemen dataset artikel. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Selengkapnya ditunjukkan tabel 8.17:

**Tabel 8.17
Pengujian Fungsionalitas Manajemen Dataset Artikel**

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Menambah dataset artikel.	Sistem menyimpan dataset artikel dan menampilkannya di daftar artikel.	Sistem menyimpan dataset artikel dan menampilkannya di daftar artikel.	Sesuai

**Tabel 8.17
(Lanjutan)**

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
2	Mengedit dataset artikel.	Sistem menyimpan perubahan artikel dan menampilkan perubahannya di daftar artikel.	Sistem menyimpan perubahan artikel dan menampilkan perubahannya di daftar artikel.	Sesuai
3	Menghapus dataset artikel.	Sistem menghilangkan dataset artikel dari daftar artikel.	Sistem menghilangkan dataset artikel dari daftar artikel.	Sesuai
4	Mengakses detail dataset artikel.	Sistem menampilkan detail dataset artikel yang dipilih.	Sistem menampilkan detail dataset artikel yang dipilih.	Sesuai

Selanjutnya dijelaskan tentang pengujian manajemen dataset komentar. Pengujian dilakukan dengan mengakses, menginput, mengedit, menghapus dataset komentar. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas sistem penyaring komentar *cyberbullying* hingga sistem bisa berjalan normal. Selengkapnya pengujian dataset komentar ditunjukan pada tabel 8.18:

**Tabel 8.18
Pengujian Fungsionalitas Manajemen Dataset Komentar**

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Menambah dataset komentar.	Sistem menyimpan dataset komentar dan menampilkannya di daftar komentar.	Sistem menyimpan dataset komentar dan menampilkannya di daftar komentar.	Sesuai
2	Mengedit dataset komentar.	Sistem menyimpan perubahan artikel dan menampilkan perubahannya di daftar komentar.	Sistem menyimpan perubahan artikel dan menampilkan perubahannya di daftar komentar.	Sesuai
3	Menghapus dataset komentar.	Sistem menghilangkan dataset komentar dari daftar komentar.	Sistem menghilangkan dataset komentar dari daftar komentar.	Sesuai

**Tabel 8.18
(Lanjutan)**

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
4	Mengakses detail dataset komentar.	Sistem menampilkan detail dataset komentar yang dipilih.	Sistem menampilkan detail dataset komentar yang dipilih.	Sesuai
5	Melakukan <i>preprocessing</i> meliputi <i>case folding</i> , <i>cleaning</i> , normalisasi bahasa gaul, <i>stemming</i> , <i>stopword</i> dan <i>tokenizing</i> pada komentar.	Sistem menghasilkan dokumen komentar yang berhasil di- <i>case folding</i> , <i>cleaning</i> , normalisasi bahasa gaul, <i>stemming</i> , <i>stopword</i> , dan <i>tokenizing</i> .	Sistem menghasilkan dokumen komentar yang berhasil di- <i>case folding</i> , <i>cleaning</i> , normalisasi bahasa gaul, <i>stemming</i> , <i>stopword</i> , dan <i>tokenizing</i> .	Sesuai

Berikutnya dijelaskan tentang pengujian manajemen dataset bahasa gaul. Pengujian dilakukan dengan menginput, mengedit, menghapus dataset bahasa gaul. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas sistem manajemen dataset bahasa gaul hingga sistem bisa berjalan normal. Pada tabel 8.19 dijelaskan skenario pengujian fungsionalitas pada manajemen dataset bahasa gaul. Pada tabel tersebut terdapat 2 skenario pengujian fungsionalitas pada manajemen dataset bahasa gaul. Masing-masing skenario memiliki hasil yang diharapkan dan hasil keluaran tes. Jika hasil yang diharapkan sesuai dengan keluaran tesnya maka pengujian berhasil. Selengkapnya ditunjukkan tabel 8.19 sebagai berikut:

**Tabel 8.19
Pengujian Fungsionalitas Manajemen Dataset Bahasa Gaul**

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Menambah dataset bahasa gaul.	Sistem menyimpan dataset bahasa gaul dan menampilkannya di daftar bahasa gaul.	Sistem menyimpan dataset bahasa gaul dan menampilkannya di daftar bahasa gaul.	Sesuai

**Tabel 8.19
(Lanjutan)**

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
2	Mengedit dataset bahasa gaul.	Sistem menyimpan perubahan data bahasa gaul.	Sistem menyimpan perubahan bahasa gaul.	Sesuai
3	Menghapus dataset bahasa gaul.	Sistem menghilangkan dataset bahasa gaul dari daftar bahasa gaul.	Sistem menghilangkan dataset bahasa gaul dari daftar bahasa gaul.	Sesuai

Berikutnya dijelaskan tentang pengujian manajemen dataset stopword. Pengujian dilakukan dengan menginput, mengedit, menghapus dataset stopword. Lalu dilihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas sistem manajemen dataset stopword hingga sistem bisa berjalan normal. Berikut pengujian manajemen dataset stopword ditunjukkan pada tabel 8.20:

**Tabel 8.20
Pengujian Fungsionalitas Manajemen Dataset Stopword**

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Menambah dataset stopword.	Sistem menyimpan dataset stopword dan menampilkannya di daftar stopword.	Sistem menyimpan dataset stopword dan menampilkannya di daftar stopword.	Sesuai
2	Mengedit dataset stopword.	Sistem menyimpan perubahan data stopword dan menampilkan perubahannya di daftar stopword.	Sistem menyimpan perubahan stopword dan menampilkan perubahannya di daftar stopword.	Sesuai
3	Menghapus dataset stopword.	Sistem menghilangkan dataset stopword dari daftar stopword.	Sistem menghilangkan dataset stopword dari daftar stopword.	Sesuai

Selanjutnya dijelaskan tentang pengujian manajemen kelas kata dan prior. Pengujian dilakukan dengan mengetes fungsionalitas sistem apakah manajemen

kelas kata dan prior dapat membuat model kelas kata dan prior atau tidak. Lalu dilihat reaksi sistem apakah sesuai dengan output skenario yang diharapkan atau tidak. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas sistem manajemen kelas kata dan *prior* hingga sistem bisa berjalan normal. Berikut pengujian manajemen fitur kata dan *prior* ditunjukan tabel 8.21:

Tabel 8.21
Pengujian Fungsionalitas Manajemen Fitur Kata dan *Prior*

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Membuat model kelas kata dan prior.	Sistem memproses hasil <i>preprocessing</i> dari dataset komentar hingga menghasilkan kelas kata dan prior.	Sistem memproses hasil <i>preprocessing</i> dari dataset komentar hingga menghasilkan kelas kata dan prior.	Sesuai
2	Menghapus model kelas kata dan prior.	Sistem menghapus model kelas kata dan prior.	Sistem menghapus model kelas kata dan prior.	Sesuai
3	Membersihkan dataset komentar <i>null</i> .	Sistem menghapus seluruh hasil <i>preprocessing</i> pada dataset komentar yang bernilai <i>null</i> .	Sistem menghapus seluruh hasil <i>preprocessing</i> pada dataset komentar yang bernilai <i>null</i> .	Sesuai
4	Membagi dataset menjadi dataset latih dan testing.	Sistem membagi dataset menjadi dataset latih dan dataset testing dengan porsi data sesuai keinginan admin.	Sistem membagi dataset menjadi dataset latih dan dataset testing dengan porsi data sesuai keinginan admin.	Sesuai

Selanjutnya dijelaskan tentang pengujian manajemen pengujian model. Pengujian dilakukan dengan mengetes fungsionalitas dari sistem pengujian model, apakah pengujian model berjalan atau tidak. Jika output tidak sesuai dengan skenario, maka dilakukan perbaikan pada fungsionalitas sistem pengujian model hingga sistem bisa berjalan normal. Pada tabel 8.22 terdapat 3 skenario pengujian pada manajemen pengujian model. Sebagaimana ditunjukan pada tabel 8.22 sebagai berikut:

Tabel 8.22
Pengujian Fungsionalitas Manajemen Pengujian Model

No	Studi Kasus Pengujian	Hasil yang Diharapkan	Keluaran Tes	Hasil Tes
1	Menghapus data pengujian.	Sistem menghilangkan data pengujian dari daftar pengujian.	Sistem menghilangkan data pengujian dari daftar pengujian.	Sesuai
2	Membuat pengujian.	Sistem melakukan pengujian model untuk menghitung tingkat <i>accuracy</i> , <i>error</i> , <i>precision</i> , <i>recall</i> , <i>specificity</i> , dan f-score sesuai dengan inputan dataset komentar dari admin.	Sistem melakukan pengujian model untuk menghitung tingkat <i>accuracy</i> , <i>error</i> , <i>precision</i> , <i>recall</i> , <i>specificity</i> , dan f-score sesuai dengan inputan dataset komentar dari admin.	Sesuai
3	Mengkalkulasi sentimen komentar pada komentar spesifik.	Sistem menampilkan hasil analisis sentimen beserta nilainya.	Sistem menampilkan hasil analisis sentimen beserta nilainya.	Sesuai

8.2 Pengujian Model Penyaring Komentar Cyberbullying

Pada sub bab ini dijelaskan tentang pengujian model pada sistem penyaring *cyberbullying*. Pengujian model bertujuan mencari akurasi model terbaik dalam mengklasifikasikan data komentar menjadi sentimen *cyberbullying* dan *non-cyberbullying*. Pengujian model menggunakan parameter utama *accuracy and error rate* untuk mengukur akurasi model dalam mengklasifikasikan sentimen komentar menjadi *cyberbullying* atau *non-cyberbullying*. Adapun parameter *precision*, *recall*, dan *f-score* digunakan sebagai parameter penunjang saja.

Adapun teknik evaluasi model menggunakan *k-fold cross validation*. *K-fold cross-validation* merupakan teknik validasi model yang berguna untuk mengevaluasi kinerja model dimana dataset D dibagi dipisahkan menjadi sub himpunan data (*fold*) sejumlah k sub himpunan. Masing-masing *fold* berisi 1/k bagian dataset. Dalam pengujian model ini, setiap model memiliki 4 *fold* yang berasal dari 6527 dataset komentar. Sehingga dataset D₁, D₂, D₃, D₄ berisi masing-masing 3 *fold* untuk data latih dan 1 *fold* untuk data uji.

Dalam pengujian model ini, dataset akan diberikan 4 perlakuan yang berbeda. Pertama, model dilatih dengan dataset komentar yang telah mengalami *preprocessing stemming* dan menggunakan fitur *bigram*, kemudian diuji tingkat akurasi dan *error*-nya. Kedua, model dilatih dengan dataset komentar yang telah mengalami *preprocessing* tanpa melalui *stemming* dan menggunakan fitur *bigram*, kemudian diuji tingkat akurasi dan *error*-nya. Ketiga, model dilatih dengan dataset komentar yang mengalami *preprocessing stemming* dan menggunakan fitur *unigram*, kemudian diuji tingkat akurasi dan *error*-nya. Keempat, model dilatih dengan dataset komentar yang mengalami *preprocessing* tanpa melalui *stemming* dan menggunakan fitur *unigram*, kemudian diuji tingkat akurasi dan *error*-nya. Berikut ini merupakan hasil pelatihan model dengan 4 perlakuan berbeda:

1. Hasil Evaluasi Model Menggunakan Fitur *Stemming* dan Fitur *Bigram*

Pada tabel 8.23 merupakan hasil evaluasi model menggunakan fitur *stemming* dan fitur *bigram*. Berdasarkan tabel tersebut, diketahui bahwa penggunaan fitur *stemming* dan fitur bigram mampu memberikan tingkat akurasi sebesar 67% dan tingkat *error* sebesar 33%. Adapun tingkat *recall*, *precision*, dan *f-score* masing-masing sebesar 87%, 67%, 76%. Selengkapnya ditunjukkan pada tabel berikut:

Tabel 8.23
Evaluasi Model Dengan Fitur Stemming dan Bigram

Aspek	Hasil
Total dataset	6537
TP	3436
TN	916
FP	1679
FN	506
Akurasi	67%
Error	33%
Recall	87%
Precision	67%
F-score	76%

Hasil akurasi dan error tersebut diperoleh dari persamaan *confusion matrix*. *Accuracy* (akurasi) merupakan rasio dari jumlah data yang berhasil

diklasifikasikan secara benar oleh model. Akurasi menjawab pertanyaan, “Berapa persen komentar yang benar diprediksi *cyberbullying* dan *non-cyberbullying* dari keseluruhan dataset komentar?” Berikut disajikan perhitungan akurasi:

$$\frac{TP + TN}{TP + TN + FP + FN} = \frac{3436 + 916}{3436 + 916 + 1679 + 506} = 0,67$$

Berdasarkan hasil di atas, model dengan penggunaan fitur *stemming* dan fitur *bigram* memiliki tingkat akurasi sebesar 67%. Hasil akurasi ini berasal dari TP (*true positive*) dan TN (*true negative*) yang dibagi dengan TP, TN, FP, dan FN. Berikut ini disajikan beberapa contoh hasil pengujian dataset yang menghasilkan nilai TP dan TN:

Tabel 8.24
Contoh Hasil TP dan TN

No	Komentar	Aktual	Prediksi	Hasil
1	kPAI omdo,,, bisa nya cuman omdo,,bubarkan KPI,,	<i>cyberbullying</i>	<i>cyberbullying</i>	TP
2	anak alay berebut knalpot bocor	<i>cyberbullying</i>	<i>cyberbullying</i>	TP
3	Livarpooll memang doyan bayar wasit	<i>cyberbullying</i>	<i>cyberbullying</i>	TP
4	Lah cebong kan orang pdip semuanya	<i>cyberbullying</i>	<i>cyberbullying</i>	TP
5	Cemen ah, mentri kelautannya sama aja mlempem	<i>cyberbullying</i>	<i>cyberbullying</i>	TP
6	Sekolah tidak salah. Kok malah dibubarin. Yang salah adalah pemimpin sekolahnya. Kita harus bisa membedakan mana yang bener dan salah . jangan main hajar semua seperti orang tak beradab	<i>Non-cyberbullying</i>	<i>Non-cyberbullying</i>	TN
7	Nice Strategie..... emang hebat Presiden kita	<i>Non-cyberbullying</i>	<i>Non-cyberbullying</i>	TN

**Tabel 8.24
(Lanjutan)**

No	Komentar	Aktual	Prediksi	Hasil
8	sudah surut kok semoga cepat kembali normal	<i>Non-cyberbullying</i>	<i>Non-cyberbullying</i>	TN
9	idolanya membawa nama harum indonesia	<i>Non-cyberbullying</i>	<i>Non-cyberbullying</i>	TN
10	Semoga berhasil timnas Garuda Indonesia	<i>Non-cyberbullying</i>	<i>Non-cyberbullying</i>	TN

Adapun *error* merupakan rasio tingkat kesalahan model dalam mengklasifikasikan data. Error menjawab pertanyaan, “Berapa persen komentar yang salah diprediksi *cyberbullying* dan *non-cyberbullying* dari keseluruhan dataset komentar?” Berikut disajikan perhitungan *error*:

$$\frac{FP + FN}{TP + TN + FP + FN} = \frac{1679 + 506}{3436 + 916 + 1679 + 506} = 0,33$$

Berdasarkan hasil di atas, model dengan penggunaan fitur *stemming* dan fitur *bigram* memiliki tingkat *error* sebesar 33%. Hasil *error* ini berasal dari FP (*false positive*) dan FN (*false negative*) yang dibagi dengan TP, TN, FP, dan FN. Berikut ini disajikan beberapa contoh hasil pengujian dataset yang menghasilkan nilai FP dan FN:

**Tabel 8.25
Contoh Hasil FP dan FN**

No	Komentar	Aktual	Prediksi	Hasil
1	Terbukti kalo LGBT itu adalah penyakit yg harus di sembuhkan	<i>Non-cyberbullying</i>	<i>cyberbullying</i>	FP
2	dia bilang , dia generasi yang gagal untuk timnas . Tapi untuk kita semua The Jak Mania dia adalah legenda dan icon club'	<i>Non-cyberbullying</i>	<i>cyberbullying</i>	FP
3	Apa benar ya si korban (almarhum) senangnya selingkuh ???	<i>Non-cyberbullying</i>	<i>cyberbullying</i>	FP

**Tabel 8.25
(Lanjutan)**

No	Komentar	Aktual	Prediksi	Hasil
3	Bagaimana dengan warga terdampak banjir? Apakah mereka juga harus membuat himpunan sebagai warga biasa korban banjir dan menuntut ke pemprov dengan perhitungan kerugian materiil maupun non materiil? Aneh!!.	<i>Non-cyberbullying</i>	<i>cyberbullying</i>	FP
5	udahlah kasian VA bebaskan dia spy bisa menata hidupnya yg baru, smg bertobat dari kesalahan2nya... dia juga korban broken home...	<i>Non-cyberbullying</i>	<i>cyberbullying</i>	FP
6	Si luhut sok"an kaga tau, berapa sih yg udah lo terima dari cina...	<i>cyberbullying</i>	<i>Non-cyberbullying</i>	FN
7	Orang tenggelam nggak bisa tereak-tereak bikin gaduh, kakinya saja tendang kanan tendang kiri, bisa kena kepala si anis, langsung stroke.	<i>cyberbullying</i>	<i>Non-cyberbullying</i>	FN
8	jongos lulung dan bosnya sikadal gurun aja yg pindah keyaman karena tidak layak tinggal diidonesia	<i>cyberbullying</i>	<i>Non-cyberbullying</i>	FN
9	Bersama anies ngak gaduh tapi klelep.	<i>cyberbullying</i>	<i>Non-cyberbullying</i>	FN
10	brarti b4ncilona tim pujaan loe itu yg di giles 4-0 di anfield tim ketoprak juga dong	<i>cyberbullying</i>	<i>Non-cyberbullying</i>	FN

Adapun *recall* (sensitifitas) merupakan rasio prediksi benar *cyberbullying* dibandingkan dengan keseluruhan data yang benar *cyberbullying*. Recall menjawab pertanyaan, "berapa persen komentar yang

diprediksi *cyberbullying* dibandingkan keseluruhan dataset komentar yang sebenarnya *cyberbullying*?” Berikut dijelaskan perhitungan *recall*:

$$\frac{TP}{TP + FN} = \frac{3.436}{3.436 + 506} = 0,87$$

Berdasarkan hasil perhitungan *recall* di atas diketahui bahwa *recall* memiliki nilai sebesar 87%. Hasil *recall* diperoleh dari hasil TP yang dibagi dengan TP dan FN.

Adapun *precision* merupakan rasio prediksi benar *cyberbullying* dibandingkan dengan keseluruhan hasil yang diprediksi *cyberbullying*. Precision menjawab pertanyaan, “berapa persen dataset komentar yang benar *cyberbullying* dari keseluruhan dataset komentar yang diprediksi *cyberbullying*?” Berikut dijelaskan perhitungan *precision*:

$$\frac{TP}{TP + FP} = \frac{3.436}{3.436 + 1.679} = 0,67$$

Berdasarkan hasil perhitungan *precision* di atas diketahui bahwa *precision* memiliki nilai sebesar 67%. Hasil *precision* diperoleh dari hasil TP yang dibagi dengan TP dan FP.

Adapun *f1 score (f-score)* merupakan perbandingan rata-rata *precision* dan *recall* yang dibobotkan. F-score menjawab pertanyaan, “berapa persen rata-rata harmonik dari *precision* dan *recall* dari model?” Berikut dijelaskan perhitungan *f-score*:

$$\frac{2 \times \textit{precision} \times \textit{recall}}{\textit{precision} + \textit{recall}} = \frac{2 \times 0,67 \times 0,87}{0,67 + 0,87} = 0,76$$

Berdasarkan hasil perhitungan *f-score* di atas diketahui bahwa *f-score* memiliki nilai sebesar 76%. Hasil *f-score* diperoleh dari hasil pengkalian 2 kali dikalikan *precision* dikali *recall*, lalu dibagi jumlah *precision* dan *recall*.

2. Hasil Evaluasi Model Tanpa Fitur *Stemming* dan Penggunaan Fitur *Bigram*

Pada tabel 8.26 merupakan hasil evaluasi model tanpa fitur *stemming* dan menggunakan fitur bigram. Berdasarkan tabel tersebut, diketahui bahwa penggunaan fitur bigram tanpa melalui *stemming* hanya mampu memberikan tingkat akurasi sebesar 66% dan tingkat *error* sebesar 34%. Adapun tingkat *recall*, *precision*, dan *f-score* masing-masing sebesar 89%, 66%, 76%. Selengkapnya ditunjukkan pada tabel berikut:

Tabel 8.26
Evaluasi Model Dengan Fitur Bigram Tanpa Melalui Stemming

Aspek	Hasil
Total dataset	6537
TP	3489
TN	837
FP	1758
FN	453
Akurasi	66%
Error	34%
Recall	89%
Precision	66%
F-score	76%

3. Hasil Evaluasi Model Menggunakan Fitur *Stemming* dan Fitur *Unigram*

Pada tabel 8.27 merupakan hasil evaluasi model menggunakan fitur *stemming* dan menggunakan fitur *unigram*. Berdasarkan tabel tersebut, diketahui bahwa penggunaan fitur *stemming* dan *unigram* mampu memberikan tingkat akurasi sebesar 73% dan tingkat *error* sebesar 27%. Adapun tingkat *recall*, *precision*, dan *f-score* masing-masing sebesar 88%, 73%, 80%. Selengkapnya ditunjukkan pada tabel berikut:

Tabel 8.27
Hasil Evaluasi Model Dengan Fitur Stemming dan Unigram

Aspek	Hasil
Total dataset	6537
TP	3464
TN	1324
FP	1271

**Tabel 8.27
(Lanjutan)**

Aspek	Hasil
FN	478
Akurasi	73%
Error	27%
Recall	88%
Precision	73%
F-score	80%

4. Hasil Evaluasi Model Tanpa Fitur *Stemming* dan Penggunaan Fitur *Unigram*

Pada tabel 8.28 merupakan hasil evaluasi model menggunakan fitur *unigram* tanpa melalui fitur *stemming*. Berdasarkan tabel tersebut, diketahui bahwa tanpa penggunaan fitur *stemming* dan hanya menggunakan fitur *unigram*, model hanya mampu memberikan tingkat akurasi sebesar 70% dan tingkat *error* sebesar 30%. Adapun tingkat *recall*, *precision*, dan *f-score* masing-masing sebesar 87%, 69%, 77%. Selengkapnya ditunjukkan pada tabel berikut:

**Tabel 8.28
Evaluasi Model Dengan Tanpa Fitur Stemming dan Unigram**

Aspek	Hasil
Total dataset	6537
TP	2991
TN	1245
FP	1350
FN	449
Akurasi	70%
Error	30%
Recall	87%
Precision	69%
F-score	77%

Berdasarkan hasil evaluasi di atas, selanjutnya tiap hasil evaluasi model akan dibandingkan. Jika akurasi dari model yang menggunakan fitur *stemming-unigram* dibandingkan dengan model yang menggunakan fitur tanpa *stemming* dan menggunakan *unigram*, maka diketahui bahwa model dengan fitur *stemming-unigram* memiliki akurasi 3% lebih baik dibandingkan model dengan fitur

stemming-unigram. Model dengan fitur *stemming-unigram* memiliki akurasi sebesar 73%. Sementara model dengan fitur tanpa *stemming* dan menggunakan fitur *unigram* memiliki akurasi sebesar 70%. Sementara itu, dari sisi *f-score*, model dengan fitur *stemming-unigram* memiliki nilai *f-score* sebesar 80% atau 3% lebih baik dibandingkan model dengan tanpa fitur *stemming* dan menggunakan fitur *unigram* yang memiliki *f-score* sebesar 77%.

Di sisi lain, jika akurasi dari model yang menggunakan fitur *stemming-bigram* dibandingkan model yang tanpa menggunakan fitur *stemming-bigram*, maka diketahui bahwa model yang menggunakan fitur *stemming-bigram* memiliki akurasi 1% lebih baik dibandingkan model yang tanpa menggunakan fitur *stemming* dan menggunakan *bigram*. Model dengan fitur *stemming-bigram* memiliki akurasi 67%. Sedangkan model yang tanpa fitur *stemming* dan menggunakan fitur *bigram* memiliki akurasi sebesar 66%. Sementara itu, dari sisi *f-score*, jika model dengan fitur *stemming-bigram* dibandingkan dengan model tanpa fitur *stemming* menggunakan fitur *bigram*, maka keduanya sama-sama memiliki nilai *f-score* sebesar 76%.

Sementara itu, jika akurasi dari model yang menggunakan fitur *stemming-unigram* dibandingkan model yang menggunakan fitur *stemming-bigram*, maka diketahui bahwa model dengan fitur *stemming-unigram* memiliki akurasi 6% lebih baik dibandingkan model dengan fitur *stemming-bigram*. Model dengan fitur *stemming-unigram* memiliki akurasi sebesar 73%. Sementara itu, model dengan fitur *stemming-bigram* hanya memiliki akurasi sebesar 67%. Sementara itu, dari sisi *f-score*, model dengan fitur *stemming-unigram* memiliki nilai *f-score* sebesar 80% atau 4% lebih baik dibandingkan model dengan *stemming-bigram* yang memiliki *f-score* sebesar 76%.

Sedangkan jika akurasi dari model yang menggunakan fitur tanpa *stemming* dan menggunakan fitur *unigram* dibandingkan dengan model yang menggunakan fitur tanpa *stemming* dan menggunakan fitur *bigram*, maka diketahui bahwa model yang menggunakan fitur tanpa *stemming* dan menggunakan *unigram* memiliki akurasi 4% lebih baik dibandingkan model yang tanpa fitur *stemming* dan menggunakan fitur *bigram*. Model dengan fitur tanpa *stemming* dan menggunakan

unigram memiliki akurasi sebesar 70%. Sedangkan model dengan fitur tanpa *stemming* dan menggunakan fitur *bigram* hanya memiliki akurasi sebesar 66%. Sementara itu, dari sisi *f-score*, model dengan tanpa fitur *stemming* dan menggunakan *unigram* memiliki nilai *f-score* sebesar 77% atau 1% lebih baik dibandingkan model dengan tanpa fitur *stemming* dan menggunakan fitur *bigram* yang memiliki *f-score* sebesar 76%.

Berdasarkan perbandingan di atas, model dengan fitur *stemming-unigram* memiliki tingkat akurasi dan nilai *f-score* tertinggi. Sementara model tanpa fitur *stemming* dan menggunakan fitur *bigram* memiliki tingkat akurasi dan nilai *f-score* terkecil. Dengan kata lain, penggunaan fitur *stemming-unigram* dapat meningkatkan akurasi dan nilai *f-score* cukup signifikan. Dengan demikian, berdasarkan perbandingan akurasi antar model di atas dapat disimpulkan bahwa model yang menggunakan fitur *stemming-unigram* memiliki akurasi terbaik sebesar 73%. Selanjutnya dijelaskan perbandingan antar model berkaitan dengan ukuran fitur kata dari masing-masing model yang dikembangkan untuk mencari tahu model dengan ukuran fitur kata terkecil atau efisien secara ukurannya. Berikut disajikan tabel 8.27 yang berisi perbandingan ukuran fitur kata dari masing-masing model:

Tabel 8.29
Perbandingan Ukuran Fitur Kata Antar Model

Model	Ukuran Fitur Kata
<i>Model01-Stemming-Unigram</i>	4.910
<i>Model02-TanpaStemming-Unigram</i>	7.295
<i>Model03-Stemming-Bigram</i>	22.613
<i>Model04-TanpaStemming-Bigram</i>	23.787

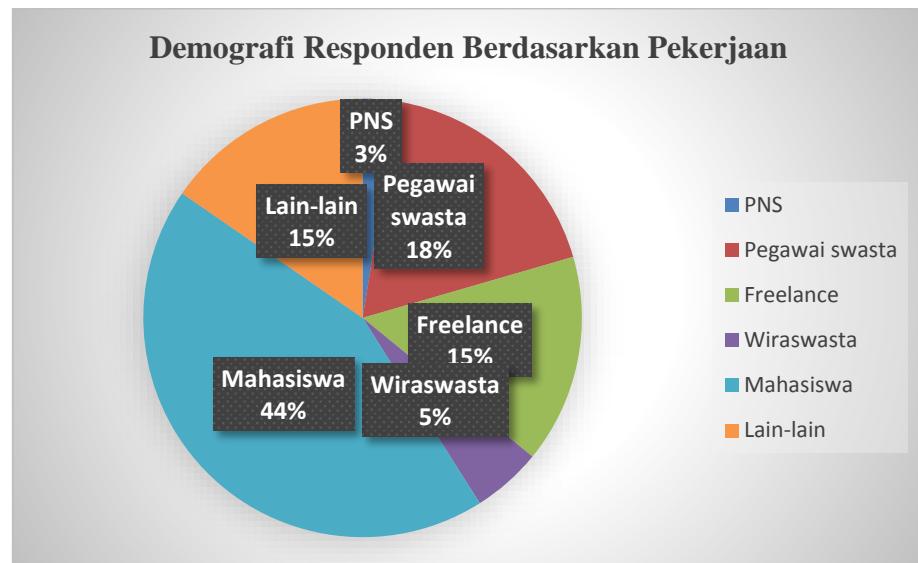
Berdasarkan tabel di atas dapat diketahui bahwa penggunaan fitur *stemming* dan fitur *unigram* dalam pembuatan model klasifikasi dapat memperkecil ukuran fitur kata. Sebaliknya penggunaan fitur *bigram* tanpa menggunakan fitur *stemming* dapat memperbesar ukuran fitur kata. Model dengan fitur *stemming* dan penggunaan *unigram* dalam pembuatan model hanya memberikan ukuran fitur kata sebesar 4.910 kata. Sedangkan penggunaan fitur *bigram* tanpa menggunakan fitur *stemming* dalam pembuatan model memberikan ukuran fitur kata sebesar 23.787

kata. Hal ini berarti *model01-stemming-unigram* hanya sebesar 21% dari ukuran fitur kata *model04-TanpaStemming-Bigram*. Dengan kata lain, model paling efisien secara ukuran fitur kata ialah model dengan penggunaan fitur *stemming-unigram*.

8.3 Pengujian Sistem Berdasarkan Pengalaman Pengguna

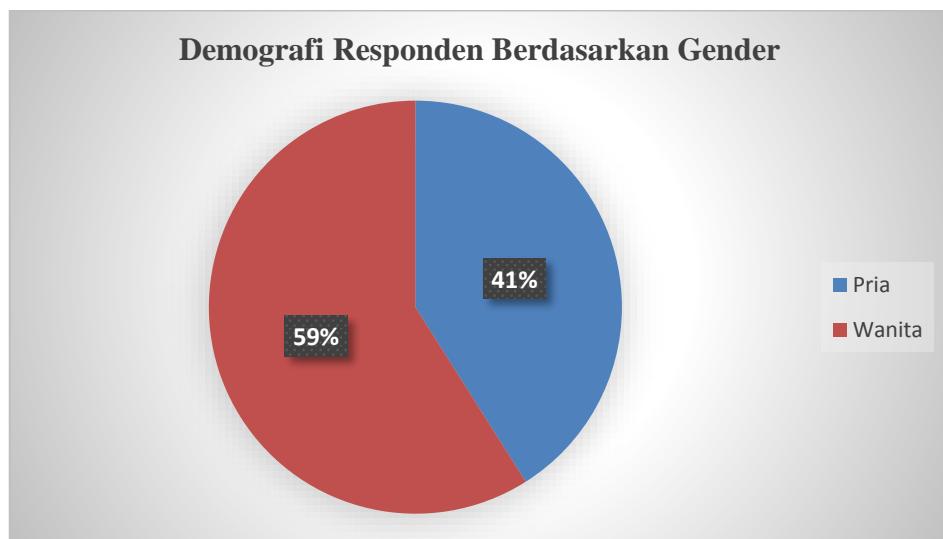
Pada sub bab ini dijelaskan tentang bagaimana pengalaman pengguna dalam menggunakan sistem manajemen konten untuk mengukur kualitas website. Pertama, *usability* adalah tingkat kualitas dari sistem yang mudah dipelajari, mudah digunakan dan mendorong pengguna untuk menggunakan sistem sebagai alat bantu positif dalam menyelesaikan tugas. Adapun beberapa indikator dari *usability* meliputi memiliki tampilan yang menarik (*user friendly*), mudah dioperasikan dan dipahami (*easy to use*), memiliki tampilan responsif (*responsible*). Kedua, *reliability* merupakan peluang fitur sistem menjalankan fungsinya dengan baik, sesuai dengan yang dipersyaratkan, dalam kurun waktu tertentu dan dalam kondisi operasi tertentu pula. Ketersediaan konten yang dibutuhkan pengguna (*availability*) menentukan derajat *reliability*. Ketiga, *functionality* merupakan seberapa baik sebuah situs web bekerja dari aspek teknologinya. Fitur yang berjalan dengan baik (*navigation*) dan mendukung konten yang relevan akan menentukan *functionality* suatu website. Keempat, *efficiency* merupakan kemampuan menjalankan tugas dengan baik dan tepat tanpa membuang waktu, tenaga maupun biaya. Beberapa indikator *efficiency* meliputi kecepatan mengakses konten website (*time behaviour*) dan kemudahan mengakses website (*accessibility*).

Dalam tugas akhir ini, diujikan sebanyak 39 responden yang menguji kualitas website. responden tersebut merupakan pihak-pihak yang mencoba dan menggunakan sistem manajemen konten. Selanjutnya, dijelaskan demografi responden berdasarkan pekerjaannya sebagaimana ditunjukkan pada gambar 8.4. Berdasarkan grafik pada gambar 8.4, responden terdiri atas 44% mahasiswa (17 orang), 18% pegawai swasta (7 orang), 15% freelance (6 orang), 3% PNS (1 orang), 5% wiraswasta (2 orang), dan 16% lain-lain (6 orang) dari total 39 responden. Adapun responden terbesarnya adalah mahasiswa dengan jumlah 17 orang dari total 39 responden.



Gambar 8.1
Demografi Responden Berdasarkan Pekerjaan

Selanjutnya dijelaskan demografi responden berdasarkan gendernya. Berdasarkan grafik pada gambar 8.5, responden terdiri atas 59% wanita (23 orang) dan 41% pria (16 orang) dari total 39 responden. Adapun responden terbesarnya adalah wanita. Selengkapnya ditunjukkan pada gambar 8.5 sebagai berikut:



Gambar 8.2
Demografi Responden Berdasarkan Gender

Untuk pembobotan dan pemberian predikat terkait ukuran kualitas website digunakan skala likert dengan skala 1 – 5 untuk menilai kualitas website. Dalam hal ini, responden akan diberikan kuesioner sejumlah 25 pertanyaan yang masing-masing mewakili kualitas-kualitas website yang diukur. Untuk mengukur kualitas situs terhadap masing-masing aspek meliputi *usability*, *functionality*, *reliability*, *efficiency*, maka hasil jawaban dari responden dijumlahkan seluruhnya berdasarkan aspek-aspeknya. Kemudian, predikat kualitas situs dari masing-masing aspek dihitung dengan persamaan:

$$Rata-rata kualitas web = \frac{\text{Total Nilai Jawaban}}{\text{Jumlah Jawaban}} \quad \dots \dots \dots (8.1)$$

Misalnya, diketahui aspek *usability* memiliki total nilai jawaban sebesar 545 dari seluruh pertanyaan terkait *usability* yang telah dijawab oleh 39 responden, dimana jumlah jawabannya sebesar 156 jawaban. Sehingga rata-rata nilai aspek *usability* dari website sebesar: $545 / 156 = 3,49$. Selanjutnya hasil rata-rata nilai tersebut digolongkan sesuai dengan skala predikat ukuran kualitas website, sehingga dapat diketahui bahwa *usability*-nya berpredikat cukup. Berikut ini merupakan skala predikat ukuran kualitas website:

Tabel 8.30
Skala Predikat Ukuran Kualitas Website

Predikat	Skala Likert (1 – 5)
Sangat baik	5
Baik	4 – 4,99
Cukup	3 – 3,99
Buruk	2 – 2,99
Sangat buruk	<2

Berdasarkan hasil survei kepada 39 responden didapat hasil bahwa: pertama, *website* memiliki kualitas tingkat *usability* berpredikat cukup baik dengan rata-rata skor 3,49 dari skala 5. Kedua, *website* memiliki tingkat *reliability* berpredikat baik dengan rata-rata skor 4,28 dari skala 5. Ketiga, *website* memiliki tingkat *functionality* berpredikat baik dengan rata-rata skor 4,11 dari skala 5.

Keempat, *website* memiliki tingkat *efficiency* berpredikat cukup dengan rata-rata skor 3,78 dari skala 5. Berdasarkan hasil ini dapat disimpulkan bahwa *website* memiliki kekurangan pada aspek *usability* dan *efficiency*. Sementara aspek lainnya meliputi *functionality* dan *reliability* sudah baik. Berikut statistik kualitas *website* berdasarkan pengalaman penggunanya ditunjukkan pada tabel 8.29 sebagai berikut:

Tabel 8.31
Hasil Pengujian Kualitas Website

Aspek	Total Nilai	Jumlah Item	Rata-Rata	Predikat
<i>Usability</i>	545	156	3,49	Cukup
<i>Reliability</i>	501	117	4,28	Baik
<i>Functionality</i>	1.603	390	4,11	Baik
<i>Efficiency</i>	1.180	312	3,78	Cukup

Berdasarkan tabel 8.31, diketahui bahwa aspek *reliability* memiliki nilai rata-rata tertinggi dibandingkan aspek-aspek lainnya. Berikutnya aspek *functionality* juga memiliki nilai rata-rata yang baik. Hal ini berarti dari segi kehandalan dan fungsionalitas, sistem manajemen konten *blog* kebaikankita.com memiliki nilai kehandalan dan fungsionalitas yang berjalan baik. Berikut disajikan statistik perbandingan kualitas website dari segi *usability*, *reliability*, *functionality*, *efficiency* sebagaimana ditunjukkan pada gambar 8.6:



Gambar 8.3
Grafik Perbandingan Kualitas Website

Sedangkan aspek *usability* memiliki nilai rata-rata terendah. Lalu aspek *efficiency* menempati peringkat ketiga terbawah. Hal ini berarti dari segi kegunaan khususnya berkaitan dengan tampilan antar muka, sistem manajemen konten blog perlu ditingkatkan supaya lebih ramah terhadap pengguna. Berikutnya, aspek *efficiency* juga perlu ditingkatkan supaya sistem bisa memberikan kemudahan dan kecepatan bagi pengguna dalam mengelola konten *blog*-nya di situs kebaikankita.com.

BAB IX

PENUTUP

Pada Bab IX dijelaskan tentang kesimpulan dan saran terkait pengembangan sistem manajemen konten dan sistem penyaring komentar *cyberbullying* pada konten blog. Berdasarkan pemaparan dari Bab I sampai Bab VII dapat ditarik beberapa kesimpulan dan saran. Adapun kesimpulan dan sarannya sebagai berikut:

9.1 Kesimpulan

Berdasarkan pemaparan tentang pengembangan sistem manajemen konten dan sistem penyaring komentar *cyberbullying* pada konten *blog* dari BAB I sampai BAB VIII dapat disimpulkan bahwa :

1. Dari hasil pengembangan sistem manajemen konten untuk mengelola suatu situs blog dapat disimpulkan bahwa:
 - Pengembangan sistem manajemen konten menggunakan *framework* CodeIgniter dapat memberikan pengalaman pengguna dengan masing-masing tingkat *usability* yang cukup baik (nilai 3,49 dari skala 5,00), tingkat *reliability* yang baik (nilai 4,28 dari skala 5,00), tingkat *functionality* yang baik (nilai 4,11 dari skala 5,00), dan tingkat *efficiency* yang cukup baik (nilai 3,78 dari skala 5,00) dari hasil penilaian 39 responden.
 - Penggunaan *pretty URL* dari *framework* CodeIgniter dalam pengembangan sistem manajemen konten membuat URL situs menjadi rapi dan bersih.
 - Penggunaan *library* Ion Auth dan Facebook API dalam pengembangan sistem manajemen konten blog berguna untuk pembuatan otentifikasi dan registrasi secara ringkas dan ringan.
 - Penggunaan *library* Chartjs dalam pengembangan sistem manajemen konten blog dapat menyajikan laporan dengan grafik yang menarik.

- Sistem manajemen konten dikembangkan dengan menyediakan interaksi antara pembaca dan penulis artikel seperti fitur rating yang berguna bagi pembaca untuk penilaian artikel, fitur komentar yang memungkinkan diskusi antar pembaca artikel, dan fitur pelaporan *cyberbullying* yang berguna bagi pembaca untuk melaporkan penyalahgunaan komentar kepada penulis, serta pesan peringatan *cyberbullying* yang berguna bagi pembaca untuk memberikan pesan peringatan terkait bahaya *cyberbullying* terkait komentar yang dikirim pembaca kepada penulis artikel.
 - Sistem manajemen konten dikembangkan dengan menyediakan fitur pengelompokan artikel berdasarkan topik artikel dan kata kunci.
2. Berdasarkan uji coba dari 6.537 dataset komentar dengan *4-fold cross validation* dalam pembuatan model klasifikasi komentar *cyberbullying* menggunakan *Multinomial Naive Bayes* dapat disimpulkan bahwa:
 - Model dengan fitur *stemming-unigram* memiliki akurasi 6% (73% - 67%) lebih baik dibandingkan model dengan fitur *stemming-bigram* (67%).
 - Model dengan fitur tanpa *stemming* dan menggunakan fitur *unigram* memiliki akurasi 4% (70% - 66%) lebih baik dibandingkan model dengan fitur tanpa *stemming* dan menggunakan fitur *bigram*.
 - Model dengan fitur *stemming-unigram* memiliki akurasi 3% (73% - 70%) lebih baik dibandingkan model dengan fitur tanpa *stemming* dan menggunakan fitur *unigram*.
 - Model dengan fitur *stemming-bigram* memiliki akurasi 1% (67% - 66%) lebih baik dibandingkan model dengan fitur tanpa *stemming* dan menggunakan fitur *bigram*.
 - Dalam tugas akhir ini, model yang terbaik ialah model dengan fitur *stemming-unigram* yang memiliki sebesar 73% dan tingkat *error* sebesar 27% di antara semua model yang dikembangkan. Sedangkan model yang terburuk ialah model dengan fitur tanpa *stemming* dan menggunakan fitur *bigram* yang akurasinya hanya 66%.
 - Model *stemming-unigram* memiliki ukuran fitur kata hanya sebesar 21% atau 4.910 fitur kata dari ukuran fitur kata dari model dengan tanpa fitur

stemming dan menggunakan fitur *bigram* yang memiliki ukuran fitur kata sebesar 23.787 fitur kata. Model *stemming-unigram* memiliki ukuran fitur kata terkecil. Sedangkan model dengan tanpa fitur *stemming* dan menggunakan fitur *bigram* memiliki ukuran fitur yang terbesar.

- Penggunaan fitur *stemming-unigram* dalam pembuatan model klasifikasi *cyberbullying* mampu memberikan akurasi yang terbaik dan efisien secara ukuran fitur kata. Sebaliknya penggunaan tanpa fitur *stemming* dan menggunakan fitur *bigram* dalam pembuatan model klasifikasi *cyberbullying* memberikan akurasi terburuk dan ukuran fitur kata yang besar.

9.2 Saran

Berikut ini saran-saran untuk mengembangkan sistem informasi ke depannya ini di antaranya:

- Selanjutnya dapat dilakukan tindakan pemeliharaan pada sistem konten manajemen konten.
- Sebaiknya tingkat akurasi model bisa ditingkatkan lebih baik lagi dengan cara menggunakan metode *supervised* lainnya.

DAFTAR PUSTAKA

- Anonymous. 2020. *Chartjs*. [Online]
Available at: <https://www.chartjs.org>
[Accessed 22 December 2020].
- Anonymous. 2020. *Cyberbullying Vs Traditional Bullying*. [Online]
Available at: <https://www.endcyberbullying.org/cyberbullying-vs-traditional-bullying>
[Accessed 21 December 2020].
- Anonymous. 2020. *DataTables*. [Online]
Available at: <https://datatables.net>
[Accessed 22 December 2020].
- Anonymous. 2020. *Definition What Does Cyberbullying Exactly Mean*. [Online]
Available at: <https://www.endcyberbullying.org/definition-what-does-cyberbullying-exactly-mean>
[Accessed 21 December 2020].
- Anonymous. 2020. *Difference Between Naive Bayes and Multinomial Naive Bayes*. [Online]
Available at: <https://stats.stackexchange.com/questions/33185/difference-between-naive-bayes-multinomial-naive-bayes>
[Accessed 1 January 2021].
- Anonymous. 2020. *TinyMCE*. [Online]
Available at: <https://www.tiny.cloud>
[Accessed 22 December 2020].
- Anonymous. 2020. *MySQL*. [Online]
Available at: <https://www.mysql.com>
[Accessed 22 December 2020].
- Anonymous. 2020. *5 Different Types of Cyberbullying*. [Online]
Available at: <https://www.endcyberbullying.org/5-different-types-of-cyberbullying>
[Accessed 21 December 2020].
- Anugerah, Fachrian. 2017. *Perbaikan Kinerja Praproses Karakter Berulang dalam Mengenali Kata Pada Klasifikasi Sentimen Berbahasa Indonesia*.
Surabaya: Institut Teknologi Sepuluh Nopember.

Asian, J. & Williams, H.E. & Tahaghoghl, S.M.M. 2005. *Stemming Indonesian*. Presented at the 28th Australasian Computer Science Conference (ACSC 2005). Melbourne: Australian Computer Society Inc.

Candra Indhiarta, Wahyu. 2017. *Penggunaan N-Gram Pada Analisis Sentimen Pemilihan Kepala Daerah Jakarta Menggunakan Algoritma Naive Bayes*. Surakarta: Universitas Muhammadiyah Surakarta.

Edmunds, Ben. 2020. *Ion Auth*. [Online]

Available at: <https://github.com/benedmunds/CodeIgniter-Ion-Auth>
[Accessed 22 December 2020].

Hinduja, S. & Patchin, J. 2010. *Bullying, Cyberbullying, adn Suicide*. Archives of Suicides Research. Vol.14.

Hinduja, S. & Patchin, J. 2020. *Cyberbullying: Identification, Prevention, and Response*. [Online] Available at: <https://cyberbullying.org> [Accessed 21 December 2020].

Liyantanto. 2011. *Stemming Bahasa Indonesia dengan Algoritma Nazief dan Adriani*. [Online]

Available at: <https://liyantanto.wordpress.com/2011/06/28/stemming-bahasa-indonesia-dengan-algoritma-nazief-dan-andriani>
[Accessed 24 December 2020].

Natalia, El Chris. 2016. *Remaja, Media Sosial dan Cyberbullying*. Jurnal Ilmiah Komunikasi. Vol.5. No.2. Desember 2016.

Nurli, Fajriyan. 2018. *Analisis Pengaruh Kualitas Website PT. Tiki Jalur Nugraha Ekakurir (JNE) Terhadap Kepuasan Pengguna Dengan Metode WebQEM*. Surabaya: Stikom Surabaya.

Otaya, Lian G. 2016. *Probabilitas Bersyarat, Independensi dan Teorema Bayes Dalam Menentukan Peluang Terjadinya Suatu Peristiwa*. Tadbir: Jurnal Manajemen Pendidikan Islam. Vol.4. No.1. Februari 2016.

Prihatini, PM. & D. Putra, IKG & Giriantari, IAD & Sudarma, M. 2017. *Stemming Algoritm for Indonesian Digital News Text Precossing*. International Journal of Engineering and Emerging Technology. Vol.2. No.2. Juli – Desember 2017.

Rastati, Ranny. 2016. *Bentuk Perundungan Siber di Media Sosial dan Pencegahannya Bagi Korban dan Pelaku*. Jurnal Sosioteknologi. Vol.15. No.2.

Subagia, Anton. 2018. *Kolaborasi Codeigneter dan Ajax dalam Perancangan CMS*. Jakarta: PT Elex Media Komputindo.

- Suyanto. 2018. *Machine Learning: Tingkat Dasar dan Lanjut*. Bandung: Informatika Bandung.
- Wibisono, Aria & D. Rizkiono, Sampurna & Wantoro, Agus. 2020. *Filtering Spam Email Menggunakan Naive Bayes*. Journal of Telematics and Information Technology. Vol.1. No.1. 2020.
- Wirayasa, I Putu Merta & Wirawan, I.M.A. & Pradnyana I.M.A, 2019. *Algoritma Bastal: Adaptasi Algoritma Nazief dan Adriani untuk Stemming Teks Bahasa Bali*. Jurnal Nasional Pendidikan Teknik Informatika. Vol.8. No.1. Maret 2019.
- Z. Tala, Fadillah. 2003. *A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia*. [Online]
Available at: <https://eprints.illc.uva.nl/740/1/MoL-2003-02.text.pdf>
[Accessed 05 January 2021].

RIWAYAT HIDUP



Nama : Danar Dono
Alamat Asal : Buntalan RT 44 RW 19,
Kalitengah, Wedi, Klaten
Tempat/ Tanggal Lahir : Klaten, 29-02-1996

Jenjang Pendidikan:

- 2001 – 2008 SD Negeri 1 Kalitengah, Wedi, Klaten
- 2008 – 2011 SMP Negeri 1 Wedi, Klaten
- 2011 – 2014 SMK Negeri 1 Klaten, Klaten
- 2015 – 2021 Institut Sains dan Teknologi Terpadu Surabaya, Surabaya
(Program Studi S1 Informatika)

LAMPIRAN A

CONTOH DATASET KOMENTAR

Tabel A.1
Contoh Dataset Komentar

No	Judul Artikel	Komentar	Kategori
1	Siswa SMP di Solo Dikeluarkan Karena Kirim Ucapan Ultah, KPAI: Berlebihan	TUTUP SEKOLAH MABOK AGAMA...!!!!	<i>cyberbullying</i>
2	Siswa SMP di Solo Dikeluarkan Karena Kirim Ucapan Ultah, KPAI: Berlebihan	Masih ada yang tanya apa kriteria RADIKAL ?	<i>cyberbullying</i>
3	Siswa SMP di Solo Dikeluarkan Karena Kirim Ucapan Ultah, KPAI: Berlebihan	Halooo, siswanya sudah kelas VIII? Bagaimana kalian mendidik siswa tersebut sampai seperti itu? Kelas VII sudah mulai dipanggil, harusnya kalianlah yg introspeksi cara pembinaannya yg mungkin sulit ditangkap siswi itu.. walau seribu alasan ada. Tetap tanggung jawab kepada anak2 siswinya.?	<i>non-cyberbullying</i>
4	Siswa SMP di Solo Dikeluarkan Karena Kirim Ucapan Ultah, KPAI: Berlebihan	Namanya sudah peraturan, ya harus diikuti, gak suka ya cari sekolah lain.? Mengucapkan kan gak harus disekolah....?	<i>non-cyberbullying</i>
5	Siswa SMP di Solo Dikeluarkan Karena Kirim Ucapan Ultah, KPAI: Berlebihan	kalau ngeliat tampang kepseknya, rasanya pengin melempar sendal lili ke mukanya	<i>cyberbullying</i>
6	Siswa SMP di Solo Dikeluarkan Karena Kirim Ucapan Ultah, KPAI: Berlebihan	Sekolah taliban	<i>cyberbullying</i>
7	Siswa SMP di Solo Dikeluarkan Karena Kirim Ucapan Ultah, KPAI: Berlebihan	Tutup izin sekolahnya.	<i>cyberbullying</i>
8	Siswa SMP di Solo Dikeluarkan Karena Kirim Ucapan Ultah, KPAI: Berlebihan	kPAI omdo,,, bisa nya cuman omdo,,bubarkan KPI,,	<i>cyberbullying</i>

Tabel A.1
(Lanjutan)

No	Judul Artikel	Komentar	Kategori
9	Siswa SMP di Solo Dikeluarkan Karena Kirim Ucapan Ultah, KPAI: Berlebihan	Lebay banget itu para guru dan pengurus sekolah. Mang pesan ultah nya apaan, berbau porno. Kalo itu berarti para guru salah didik	<i>cyberbullying</i>
10	Siswa SMP di Solo Dikeluarkan Karena Kirim Ucapan Ultah, KPAI: Berlebihan	akibat ucapan UAS kebykan micin bikin fatwa hahaha.	<i>cyberbullying</i>
11	Bekap Tottenham, Liverpool Ciptakan Rekor Poin Eropa	Semakin banyak juru kunci makam yang jadi fans Liverpool. Terbukti!!!	<i>cyberbullying</i>
12	Bekap Tottenham, Liverpool Ciptakan Rekor Poin Eropa	Keren juga ya	<i>non-cyberbullying</i>
13	Bekap Tottenham, Liverpool Ciptakan Rekor Poin Eropa	lawannya ketoprak semua wkwkwk	<i>cyberbullying</i>
14	Bekap Tottenham, Liverpool Ciptakan Rekor Poin Eropa	Musim dpn perangkat var dukung wolves yg juara biar makin lucu liga ketoprak kompetitif nya wkwk	<i>cyberbullying</i>
15	Bekap Tottenham, Liverpool Ciptakan Rekor Poin Eropa	Pas Firmino ngegolin, jelas jelas GOL.. ee di VAR, Makin Jelas GOL nya.. aya aya wae.... ynwa	<i>non-cyberbullying</i>
16	Bekap Tottenham, Liverpool Ciptakan Rekor Poin Eropa	Setelah 30 tahun baru pada keluar dari dalam goa...jadi wajar girang banget	<i>cyberbullying</i>
17	Bekap Tottenham, Liverpool Ciptakan Rekor Poin Eropa	Luar biasa memang Liverpool... Rekor hanyalah bonus, trophy yg paling penting. Danke Jurgen! #YNWA	<i>non-cyberbullying</i>
18	Bekap Tottenham, Liverpool Ciptakan Rekor Poin Eropa	rekor bantuan VARsit opini owe owlang sebagai penikmat sepak bola netral anti jalan gelap.	<i>cyberbullying</i>
19	Bekap Tottenham, Liverpool Ciptakan Rekor Poin Eropa	semua tidak akan tercapai kalo tidak ada video pembantu livarpool	<i>cyberbullying</i>

Tabel A.1
(Lanjutan)

No	Judul Artikel	Komentar	Kategori
20	Bekap Tottenham, Liverpool Ciptakan Rekor Poin Eropa	untuk mencapai juara Liga primer Inggris..Liverpool sampai menciptakan byk rekor...smoga sampai setrusnya..amiin...saluut utk Mr.Klopp	<i>non-cyberbullying</i>
21	Dianggap Terlalu Lunak Hadapi China di Natuna, Ini Kata Luhut	makanya ente yg ga kenyang jd menteri donktunjukin kalo u mampu....jd menteri....jgn omong doank....omong doank mah anak tk aja bisa	<i>cyberbullying</i>
22	Dianggap Terlalu Lunak Hadapi China di Natuna, Ini Kata Luhut	Boro2 jadi menteri, nyalonin rt aja kagak ada yg milih.	<i>cyberbullying</i>
23	Dianggap Terlalu Lunak Hadapi China di Natuna, Ini Kata Luhut	wihhh galaknya pak menkosaurus ini	<i>cyberbullying</i>
24	Dianggap Terlalu Lunak Hadapi China di Natuna, Ini Kata Luhut	beraninya galak cuman sama niticen,,wk,,wk,,wk,,	<i>cyberbullying</i>
25	Dianggap Terlalu Lunak Hadapi China di Natuna, Ini Kata Luhut	Menteri mah..pembantu presiden...bkn rakyat yg milih tp presiden..yg gaji br rakyat	<i>non-cyberbullying</i>
26	Dianggap Terlalu Lunak Hadapi China di Natuna, Ini Kata Luhut	Kembalikan ibu Susi pada POS nya.	<i>non-cyberbullying</i>
27	Dianggap Terlalu Lunak Hadapi China di Natuna, Ini Kata Luhut	Kalo galak ya fee ga cair lah	<i>cyberbullying</i>
28	Dianggap Terlalu Lunak Hadapi China di Natuna, Ini Kata Luhut	Norak...norak...norak..? Mulut doang galak	<i>cyberbullying</i>
29	Dianggap Terlalu Lunak Hadapi China di Natuna, Ini Kata Luhut	Ah Ngoceh doang kaya emak2, eh yang emak2 malah bisa Tegas	<i>cyberbullying</i>
30	Dianggap Terlalu Lunak Hadapi China di Natuna, Ini Kata Luhut	Coba perusahaannya keganggu pasti langsung galak.hahahaha	<i>cyberbullying</i>
31	Massa Pro-Anies Pajang Spanduk Bersama Anis Baswedan Jakarta Damai	dasar cebong	<i>cyberbullying</i>

**Tabel A.1
(Lanjutan)**

No	Judul Artikel	Komentar	Kategori
32	Massa Pro-Anies Pajang Spanduk Bersama Anis Baswedan Jakarta Damai	dasar kampret	<i>cyberbullying</i>
33	Massa Pro-Anies Pajang Spanduk Bersama Anis Baswedan Jakarta Damai	dari dulu juga sudah banjir artinya dimaklumi saja ga perlu ada perubahan yg penting damai begitu pula macet dari dulu juga macet	<i>cyberbullying</i>
34	Massa Pro-Anies Pajang Spanduk Bersama Anis Baswedan Jakarta Damai	BANJIR DAN MACET DI JKT INSYALLAH LEBIH MUDAH DI ATASI JIKA ANIES JADI PRESIDEN!	<i>non-cyberbullying</i>
35	Massa Pro-Anies Pajang Spanduk Bersama Anis Baswedan Jakarta Damai	Orang tenggelam nggak bisa tereak-tereak bikin gaduh, kakinya saja tendang kanan tendang kiri, bisa kena kepala si anis, langsung stroke.	<i>cyberbullying</i>
36	Massa Pro-Anies Pajang Spanduk Bersama Anis Baswedan Jakarta Damai	Damai tp banjir,? kumuh,? hancur...	<i>cyberbullying</i>
37	Massa Pro-Anies Pajang Spanduk Bersama Anis Baswedan Jakarta Damai	bersama gabener aibon toa jakarta menata kata	<i>cyberbullying</i>
38	Massa Pro-Anies Pajang Spanduk Bersama Anis Baswedan Jakarta Damai	Jakarta sebagai barometer negara butuh gubernur yg revolusioner punya ide2 cemerlang dan bisa kerja bukan sekedar damai.	<i>non-cyberbullying</i>
39	Massa Pro-Anies Pajang Spanduk Bersama Anis Baswedan Jakarta Damai	saya sih netral aja, mengingat bukan warga DKI.. Tapi, ya jelas dan pasti damai.. kan yg nggak suka nggak akan buat aksi.	<i>non-cyberbullying</i>
40	Massa Pro-Anies Pajang Spanduk Bersama Anis Baswedan Jakarta Damai	Lha dikira ini provinsi punya si Wan Aibon..?? Damai darimananya coba..?? Urusan anggaran lem aja bikin gaduh, damai darimananya..?? Gubernur kok gak paham pekerjaan.. Memalukan	<i>non-cyberbullying</i>
41	Positif Narkoba Medina Zein Akan Jalani Rehabilitasi	Dia siapa dih, berita gak penting di up	<i>cyberbullying</i>

**Tabel A.1
(Lanjutan)**

No	Judul Artikel	Komentar	Kategori
42	Positif Narkoba Medina Zein Akan Jalani Rehabilitasi	Aku mau tanya..pemakai narkoba kok di rehabilitasi...kok ada yang di penjara...minta penjelasanya .thx	<i>non-cyberbullying</i>
43	Positif Narkoba Medina Zein Akan Jalani Rehabilitasi	Minta tolong donk plis kurangi artikel yg kurang bermutu kek gini, siapa sih dia, gak penting amat	<i>cyberbullying</i>
44	Positif Narkoba Medina Zein Akan Jalani Rehabilitasi	ASTAGA UKHTI	<i>non-cyberbullying</i>
45	Positif Narkoba Medina Zein Akan Jalani Rehabilitasi	Fulus MULUS	<i>cyberbullying</i>
46	Positif Narkoba Medina Zein Akan Jalani Rehabilitasi	Jilbab ah pakai narkoba	<i>cyberbullying</i>
47	Positif Narkoba Medina Zein Akan Jalani Rehabilitasi	Jangan suudzon dulu	<i>non-cyberbullying</i>
48	Positif Narkoba Medina Zein Akan Jalani Rehabilitasi	Amphetamine ditemukan jd dalam obat2an stimulan otak utk penyintas bipolar, adhd, autism & narcolepsy dll. Masa iya sih itu penyakit ga dipertimbangkan pak polisi?	<i>non-cyberbullying</i>
49	Positif Narkoba Medina Zein Akan Jalani Rehabilitasi	kalau untk pengobatan,polisi pasti bkalan nanya ke dr yg ngresepin.polisi itu uda pinter ,nggak mungkin asal asal nangkap.	<i>non-cyberbullying</i>
50	Positif Narkoba Medina Zein Akan Jalani Rehabilitasi	Fair...?? Bullshitt...!!!	<i>cyberbullying</i>
51	Mantan Ketua Umum PPP Romahurmuzy Divonis 2 Tahun Penjara	Hadeh suka miris	<i>non-cyberbullying</i>
52	Mantan Ketua Umum PPP Romahurmuzy Divonis 2 Tahun Penjara	Hukum di indonesia kok miris	<i>non-cyberbullying</i>

**Tabel A.1
(Lanjutan)**

No	Judul Artikel	Komentar	Kategori
53	Mantan Ketua Umum PPP Romahurmuzy Divonis 2 Tahun Penjara	Ya audahlahhh.. gimana MALING MALING berdasari mau berkurang apalagi hilang.. hukuman dua tahun.. dipotong remisi.. di potong siku.. dipotong tangan.. tinggal dua bulan.. RIP Hukum Indonesia	<i>cyberbullying</i>
54	Mantan Ketua Umum PPP Romahurmuzy Divonis 2 Tahun Penjara	Miris...dungu....Negara sdh kacau...sdh g sabar nunggu thn 2024 ganti rezim.....Kembalikan KPK sblm rezim jokowi....tolak revisi UU KPK....	<i>cyberbullying</i>
55	Mantan Ketua Umum PPP Romahurmuzy Divonis 2 Tahun Penjara	Semua orang pasti mau korupsi	<i>cyberbullying</i>
56	LBH Jakarta ke Jokowi: Ini Ajak Diskusi atau Intimidasi?	instastory ku bisa bikin kalian kaum kerempeng jadi punya badan berisi yang ideal lhooo	<i>cyberbullying</i>
57	LBH Jakarta ke Jokowi: Ini Ajak Diskusi atau Intimidasi?	w tanya omnibus law apa? Tolong jawab yg benar	<i>non-cyberbullying</i>
58	LBH Jakarta ke Jokowi: Ini Ajak Diskusi atau Intimidasi?	Sblom melangkah udah me-judge,	<i>non-cyberbullying</i>
59	Dzawin ke Coki: Menjadikan Musibah Sebagai Objek Komei Kesalahan Besar	Cebong mah bebas berkata apa...	<i>cyberbullying</i>
60	Jitu Lobby Prabowo, Gerindra Buka Pintu Usung Menantu Jokowi	Tinggal PKS	<i>non-cyberbullying</i>
61	Jitu Lobby Prabowo, Gerindra Buka Pintu Usung Menantu Jokowi	Insya Allah PKS selalu di hati masyarakat	<i>non-cyberbullying</i>
62	Kemenpora Umumkan 6 Stadion Penyelenggara Piala Dunia U20 2021	Mandala krida, ? Bagusan jalak harupat lah. Dulu kan pernah di pake asian games	<i>non-cyberbullying</i>
63	Kemenpora Umumkan 6 Stadion Penyelenggara Piala Dunia U20 2021	Di jawa semua	<i>non-cyberbullying</i>

Tabel A.1
(Lanjutan)

No	Judul Artikel	Komentar	Kategori
64	Kemenpora Umumkan 6 Stadion Penyelenggara Piala Dunia U20 2021	Padahal bagusana stadion patriot bekasi dibandingkan stadion mandala krida atau manahan solo. Kapasitas penontonnya juga lebih banyak stadion patriot	<i>non-cyberbullying</i>
65	Kemenpora Umumkan 6 Stadion Penyelenggara Piala Dunia U20 2021	Stadion Patriot,Bekasi??? @kemenpora @fifaworldcup	<i>non-cyberbullying</i>
66	Kemenpora Umumkan 6 Stadion Penyelenggara Piala Dunia U20 2021	Jalak harupat mana, GBLA mana hadeeeuuh	<i>non-cyberbullying</i>
67	Suami yang Istrinya Tepergok Selingkuh Meminta Maafkannya	Smoga gak ada rencana lain	<i>non-cyberbullying</i>
68	Suami yang Istrinya Tepergok Selingkuh Meminta Maafkannya	BUCIN NAJIS	<i>cyberbullying</i>
69	Suami yang Istrinya Tepergok Selingkuh Meminta Maafkannya	Tunggu seminggu atau sebulan lagi istri bakal DEAD	<i>cyberbullying</i>
70	Suami yang Istrinya Tepergok Selingkuh Meminta Maafkannya	Mungkin ada pertimbangan lain misalnya suaminya menyadari kekurangannya atau mungkin ada anak-anak yang masih butuh perhatian kasih sayang kedua orang tuanya dan di dunia ini tidak ada manusia yang sempurna dan tanpa dosa	<i>non-cyberbullying</i>
71	Apa Tolak Ukur Seseorang Berpemikiran Terbuka dan Berpemikiran Tertutup?	Tidak ada otoritas menetapkan ukuran kebenaran. Saya suka yang ini. Pak Ahmad Makki betul. Terima kasih sudah mengingatkan.	<i>non-cyberbullying</i>
72	Apa Tolak Ukur Seseorang Berpemikiran Terbuka dan Berpemikiran Tertutup?	Thanks. Andai kita bisa ngopi bareng.. :)	<i>non-cyberbullying</i>

Tabel A.1
(Lanjutan)

No	Judul Artikel	Komentar	Kategori
73	Apa Tolak Ukur Seseorang Berpemikiran Terbuka dan Berpemikiran Tertutup?	Sama-sama Pak. Saya juga lupa menuliskan dengan jelas waktu saya sebut Voltaire di atas. Pak Syam menjelaskan dengan mudah dan padat. Saya kagum.	<i>non-cyberbullying</i>
74	Apa Tolak Ukur Seseorang Berpemikiran Terbuka dan Berpemikiran Tertutup?	Jika tidak peduli jatuhnya apa, ya? Misal: Orang di kelas saya sering memaksakan pendapat mereka, dan bilang itu betul... saya salah, tapi saya tidak peduli, tidak menerima, pun tidak menolak. Apalagi adu argumen untuk mempertahankan pendapat saya. Mohon pendapatnya, hehe	<i>non-cyberbullying</i>
75	Apa Tolak Ukur Seseorang Berpemikiran Terbuka dan Berpemikiran Tertutup?	Terima kasih jawabannya	<i>non-cyberbullying</i>
76	Apa Tolak Ukur Seseorang Berpemikiran Terbuka dan Berpemikiran Tertutup?	Terima kasih atas apresiasinya,	<i>non-cyberbullying</i>
77	Bagaimana Psikiater atau Psikologi Melihat Fenomena Keserupan?	Itu namanya Sindrom Aksen Asing atau Foreign accent syndrome. Biasanya muncul setelah kena stroke (paling sering), tapi bisa juga dari benturan di kepala atau migrain atau bahkan dari masalah perkembangan	<i>non-cyberbullying</i>
78	Bagaimana Psikiater atau Psikologi Melihat Fenomena Keserupan?	sepertinya masih kurang dapat menjelaskan fenomena keserupan berbahasa asing	<i>non-cyberbullying</i>
79	Bagaimana Psikiater atau Psikologi Melihat Fenomena Keserupan?	dalam arti, sebetulnya otak kita punya pengetahuan bahasa yang tersembunyi. dan terbangkit karena stroke\benturan\migrain ?	<i>non-cyberbullying</i>

**Tabel A.1
(Lanjutan)**

No	Judul Artikel	Komentar	Kategori
80	Bagaimana Psikiater atau Psikologi Melihat Fenomena Keserupan?	Iya juga sih. Beberapa sumber bahkan mengatakan kalau ada yang tadinya bisa bahasa inggris lalu setelah stroke malah lancar jaya berbahasa mandarin.	<i>non-cyberbullying</i>
81	Pelaku Teror Novel Ditangkap, Dewi Tanjung Biasa Saja	Wanita dajjal	<i>cyberbullying</i>
82	Pelaku Teror Novel Ditangkap, Dewi Tanjung Biasa Saja	Mana dewi tanjung...aku hampir percaya sm dia..skg.berani gk counter attack bareskrim lok ada tsk nya...jawab dewi	<i>non-cyberbullying</i>
83	Pelaku Teror Novel Ditangkap, Dewi Tanjung Biasa Saja	Kebal Hukum Taekk lah !!!	<i>cyberbullying</i>
84	Pelaku Teror Novel Ditangkap, Dewi Tanjung Biasa Saja	autobego dia	<i>cyberbullying</i>
85	Kapolri Penuhi Janji Tangkap Penyiram Air Keras Ke Novel, Ini Faktanya	Sekalian dalangnya diungkap ya	<i>non-cyberbullying</i>
86	Kapolri Penuhi Janji Tangkap Penyiram Air Keras Ke Novel, Ini Faktanya	Sya ga yakin klo itu tersangka	<i>non-cyberbullying</i>
87	Kapolri Penuhi Janji Tangkap Penyiram Air Keras Ke Novel, Ini Faktanya	Alhamdulillah Kapolrinya yg kali ini the best	<i>non-cyberbullying</i>
88	Kapolri Penuhi Janji Tangkap Penyiram Air Keras Ke Novel, Ini Faktanya	Manipulasi tersangka.. kurang greget pak.. biar greget bikin senetron aja	<i>cyberbullying</i>
89	Kapolri Penuhi Janji Tangkap Penyiram Air Keras Ke Novel, Ini Faktanya	Melayani, melindungi, menyirami..	<i>cyberbullying</i>
90	2 Pelaku Penyiraman Air Keras Ke Novel Baswedan Ditangkap	Masih percaya dgn rezim DOBOL saat jnj	<i>cyberbullying</i>

Tabel A.1
(Lanjutan)

No	Judul Artikel	Komentar	Kategori
91	Kapolri Penuhi Janji Tangkap Penyiram Air Keras Ke Novel, Ini Faktanya	Semoga.... Demikian adanya	<i>non-cyberbullying</i>
92	2 Pelaku Penyiraman Air Keras Ke Novel Baswedan Ditangkap	Negara dagelan, cuma utk menutupi kasus jiwasraya	<i>cyberbullying</i>
93	2 Pelaku Penyiraman Air Keras Ke Novel Baswedan Ditangkap	Alhamdulillah	<i>non-cyberbullying</i>
94	2 Pelaku Penyiraman Air Keras Ke Novel Baswedan Ditangkap	Setelah ganti kapolri baru terungkap	<i>non-cyberbullying</i>
95	2 Pelaku Penyiraman Air Keras Ke Novel Baswedan Ditangkap	Menyerang anggota KPK aja bertahun2 berlarut2 baru diamankan dan baru diduga. Gimana rakyat biasa? ???? tolong lah pak..	<i>non-cyberbullying</i>
96	2 Pelaku Penyiraman Air Keras Ke Novel Baswedan Ditangkap	Jangan hanya ditangkap tolong diselidiki lebih dalam siapa dalang nya	<i>non-cyberbullying</i>
97	2 Pelaku Penyiraman Air Keras Ke Novel Baswedan Ditangkap	Alhamdulillah udh ketemu pelakunya,jd yg kemarin teriak teriak di tv,si cwe itu tu pasti malu.Udh rakyat indonesia berpikir positif saja.Kita liat kelanjutannya,di doakan untuk negara indonesia yg terbaik.Jgn bicara negatif dluan.Nanti jd doa lo	<i>non-cyberbullying</i>
98	Respon SBY Soal Kasus Jiwasraya: Salahkan Saja Masa Lalu	Mr baper	<i>cyberbullying</i>
99	Respon SBY Soal Kasus Jiwasraya: Salahkan Saja Masa Lalu	Baper lagi	<i>cyberbullying</i>
99	Respon SBY Soal Kasus Jiwasraya: Salahkan Saja Masa Lalu	Baper lagi	<i>cyberbullying</i>
100	Respon SBY Soal Kasus Jiwasraya: Salahkan Saja Masa Lalu	Itu namanya era ini GAGAL	<i>cyberbullying</i>

LAMPIRAN B

KUESIONER EVALUASI WEBSITE

Tabel B.1
Kuesioner Evaluasi Website

Kode	Variabel	Pertanyaan	Skala Likert (1-5)
P1	Usability	Seberapa menarik tampilan website kebaikankita.com?	1. Tidak menarik 2. Kurang menarik 3. Cukup menarik 4. Menarik 5. Sangat menarik
P2	Usability	Apakah Anda mengerti bagaimana menggunakan/mengoperasikan sistem manajemen konten pada website kebaikankita.com?	1. Tidak mengerti 2. Kurang mengerti 3. Cukup mengerti 4. Mengerti 5. Sangat mengerti
P3	Usability	Seberapa responsif tampilan website kebaikankita.com? Catatan: yang dimaksud responsif ialah tampilan website bisa menyesuaikan tampilan layar PC maupun mobile secara otomatis.	1. Tidak responsif 2. Kurang responsif 3. Cukup responsif 4. Responsif 5. Sangat responsif
P4	Usability	Apakah fitur FAQ, kebijakan website dan tentang wbsite (about us) membuat Anda lebih memahami bagaimana mengoperasikan website kebaikankita.com?	1. Sangat membantu 2. Membantu 3. Cukup membantu 4. Kurang membantu 5. Tidak membantu
P5	Functionality	Apakah fitur registrasi dapat berjalan dengan baik untuk membuat akun baru?	1. Tidak berfungsi 2. Agak terkendala 3. Cukup berfungsi 4. Berfungsi 5. Berfungsi lancar
P6	Functionality	Apakah fitur login dapat berjalan dengan baik untuk bisa mengakses halaman admin?	1. Tidak berfungsi 2. Agak terkendala 3. Cukup berfungsi 4. Berfungsi 5. Berfungsi lancar
P7	Functionality	Apakah sistem manajemen konten dapat berjalan dengan baik untuk mengelola konten (menulis, mengedit dan menghapus) artikel?	1. Tidak berfungsi 2. Agak terkendala 3. Cukup berfungsi 4. Berfungsi 5. Berfungsi lancar
P8	Functionality	Apakah fitur komentar dapat berjalan dengan baik untuk memberikan komentar pada artikel?	1. Tidak berfungsi 2. Agak terkendala 3. Cukup berfungsi 4. Berfungsi 5. Berfungsi lancar

Tabel B.1
(Lanjutan)

Kode	Variabel	Pertanyaan	Skala Likert (1-5)
P9	Functionality	Apakah fitur rating dapat berjalan dengan baik untuk menilai kualitas artikel?	1. Tidak berfungsi 2. Agak terkendala 3. Cukup berfungsi 4. Berfungsi 5. Berfungsi lancar
P10	Functionality	Apakah fitur penyaring komentar cyberbullying dapat berjalan dengan baik untuk memberikan pesan peringatan terkait bahaya cyberbullying bagi pembaca yang berkomentar terkait artikel?	1. Tidak berfungsi 2. Agak terkendala 3. Cukup berfungsi 4. Berfungsi 5. Berfungsi lancar
P11	Functionality	Apakah fitur mengikuti (follow) dapat berjalan dengan baik untuk mengikuti akun penulis?	1. Tidak berfungsi 2. Agak terkendala 3. Cukup berfungsi 4. Berfungsi 5. Berfungsi lancar
P12	Functionality	Apakah fitur bagikan artikel ke media sosial dapat berjalan dengan baik?	1. Tidak berfungsi 2. Agak terkendala 3. Cukup berfungsi 4. Berfungsi 5. Berfungsi lancar
P13	Functionality	Apakah fitur pencarian dapat berjalan dengan baik untuk mencari artikel yang relevan dengan kata kuncinya?	1. Tidak berfungsi 2. Agak terkendala 3. Cukup berfungsi 4. Berfungsi 5. Berfungsi lancar
P14	Functionality	Apakah fitur konfigurasi pengguna pada halaman admin dapat berjalan dengan baik?	1. Tidak berfungsi 2. Agak terkendala 3. Cukup berfungsi 4. Berfungsi 5. Berfungsi lancar
P15	Reliability	Seberapa sering Anda menjumpai tautan/ link error dalam website kebaikankita.com?	1. Sangat sering 2. Sering 3. Jarang 4. Pernah sekali 5. Tidak pernah
P16	Reliability	Seberapa sering Anda menjumpai konten yang tak relevan berdasarkan pencarian atau topik artikelnya?	1. Sangat sering 2. Sering 3. Jarang 4. Pernah sekali 5. Tidak pernah

Tabel B.1
(Lanjutan)

Kode	Variabel	Pertanyaan	Skala Likert (1-5)
P17	Efficiency	Seberapa sering Anda menjumpai tautan/ link yang tidak bisa diakses (rusak)?	<ol style="list-style-type: none"> 1. Sangat sering 2. Sering 3. Jarang 4. Pernah sekali 5. Tidak pernah
P18	Efficiency	Seberapa cepat kecepatan akses website kebaikankita.com baik diakses melalui web browser di PC (laptop/ komputer) maupun mobile?	<ol style="list-style-type: none"> 1. Sangat lambat 2. lambat 3. Cukup cepat 4. Cepat 5. Sangat cepat
P19	Efficiency	Seberapa cepat website kebaikankita.com dalam menerbitkan artikel ke halaman utama (home) setelah Anda menulis artikel menggunakan sistem manajemen konten?	<ol style="list-style-type: none"> 1. Sangat lambat 2. Lambat 3. Cukup cepat 4. Cepat 5. Sangat cepat
P20	Efficiency	Seberapa mudah Anda menggunakan sistem manajemen konten untuk mengelola (menulis, mengedit, menghapus) artikel di website kebaikankita.com?	<ol style="list-style-type: none"> 1. Sangat sulit 2. Sulit 3. Cukup mudah 4. Mudah 5. Sangat mudah
P21	Efficiency	Apakah fitur penyaring cyberbullying memudahkan Anda dalam mengelola cyberbullying (perundungan secara online) terhadap konten blog yang Anda kelola khususnya berkaitan dengan komentar pengguna yang dikirimkannya melalui fitur komentar artikel?	<ol style="list-style-type: none"> 1. Sangat sulit 2. Sulit 3. Cukup mudah 4. Mudah 5. Sangat mudah
P22	Efficiency	Apakah website kebaikankita.com memudahkan Anda dalam membagikan/ mendistribusikan artikel ke media sosial?	<ol style="list-style-type: none"> 1. Sangat sulit 2. Sulit 3. Cukup mudah 4. Mudah 5. Sangat mudah
P23	Efficiency	Apakah fitur kata kunci (tagar) memudahkan Anda dalam mencari artikel yang	<ol style="list-style-type: none"> 1. Sangat sulit 2. Sulit 3. Cukup mudah

Tabel B.1
(Lanjutan)

Kode	Variabel	Pertanyaan	Skala Likert (1-5)
		relavan sesuai dengan kata kuncinya?	4. Mudah 5. Sangat mudah
P24	Efficiency	Apakah fitur kategori memudahkan Anda dalam mencari artikel yang relavan sesuai dengan kategori topiknya?	1. Sangat sulit 2. Sulit 3. Cukup mudah 4. Mudah 5. Sangat mudah
P25	Efficiency	Apakah fitur laporan pada dashboard admin memudahkan Anda dalam menyajikan informasi terkait artikel yang Anda kelola?	1. Sangat sulit 2. Sulit 3. Cukup mudah 4. Mudah 5. Sangat mudah

LAMPIRAN C

DAFTAR PROFIL RESPONDEN

Tabel C.1
Daftar Profil Responden

Kode	Nama	Gender	Pekerjaan	E-mail
R1	Nur Hasbiyah	Wanita	Mahasiswa	hasbiyahapril@gmail.com
R2	Daris Muafa	Pria	Freelance	kazekageghon@gmail.com
R3	Nurhidayati Surtiasih	Wanita	Lain-lain	nurhidayati.dayns@gmail.com
R4	Syifa Azkia Purwanti	Wanita	Mahasiswa	SyifaAzkiaP@gmail.com
R5	Namira	Wanita	Pegawai swasta	nira44325@gmail.com
R6	Sukma Paramastuti	Wanita	Lain-lain	paramasukma@gmail.com
R7	Shahnaz Fathirrizky	Wanita	Mahasiswa	shahnazfathirrizky19@gmail.com
R8	Niko Haris	Pria	Lain-lain	mdnharis@gmail.com
R9	Muhammad Wahyu Hidayat	Pria	Mahasiswa	wahyu.cah.kiran@gmail.com
R10	Ita Nurjannah	Wanita	Lain-lain	itanurjanah768@gmail.com
R11	Sanju Amanulloh	Pria	Mahasiswa	samanulloh@gmail.com
R12	Rahayu	Wanita	Mahasiswa	rahayuhayu23@gmail.com
R13	Fardan Zahidu Zaki Al Fahd	Pria	Mahasiswa	fardanzahidu@gmail.com
R14	Nurohmah Kurniasari Dewi, S.Sos.	Wanita	Lain-lain	nurohmahkd@gmail.com
R15	Niko Putra Sadewa	Pria	Pegawai swasta	npsadewa21@gmail.com
R16	Chusnul Habibah	Wanita	Lain-lain	habeebchusnul1109@gmail.com
R17	Laily Nuraini	Wanita	PNS	lailynuraini1995@gmail.com
R18	Esa Yudha Abdillah	Pria	Mahasiswa	Yudhaesa550@gmail.com
R19	Shinta Nisaus Syuroya	Wanita	Freelance	atnis.suasin@gmail.com
R20	Risky Eko Saputro	Pria	Freelance	ekopoetro@gmail.com
R21	Dwi Puji Kusumastuti	Wanita	Pegawai swasta	dwipujikusumastuti@yahoo.co.id
R22	Putri Hermawati	Wanita	Mahasiswa	Putriremember1@gmail.com

Tabel C.1
(Lanjutan)

Kode	Nama	Gender	Pekerjaan	E-mail
R23	Hendra Setiana	Pria	Wiraswasta	hendraqeeta24@gmail.com
R24	Irvan Anggara	Pria	Freelance	irvananggaraputra02@gmail.com
R25	Zendy Santoso	Pria	Mahasiswa	zendysantoso@gmail.com
R26	Elvira Viviani	Wanita	Mahasiswa	elviravivianik25@gmail.com
R27	Surya Habiba	Wanita	Mahasiswa	suryahabibaf@gmail.com
R28	Fajar Shodiq	Pria	Mahasiswa	shodiqf971@gmail.com
R29	Siti Fatimah	Wanita	Freelance	unfortunatelyfat@gmail.com
R30	Utari Eka	Wanita	Wiraswasta	utari.eka31@gmail.com
R31	Bayu Kinanti Rachmadani	Wanita	Freelance	kinantiby@gmail.com
R32	Yeni Nuraini	Wanita	Pegawai swasta	yениnura96@gmail.com
R33	Ricky Raharjo	Pria	Mahasiswa	rickyraharjo2@gmail.com
R34	Santiara Dewi Farizki	Wanita	Mahasiswa	santiaradewif@gmail.com
R35	Muhammad Nuridin	Pria	Mahasiswa	nuricfc27@gmail.com
R36	Dewi Syafa Dinabilla M.	Wanita	Mahasiswa	nabillasyafa09@gmail.com
R37	Febri Sawaludin	Pria	Pegawai Swasta	febrisawaludin@gmail.com
R38	Serlinda Ririk Atika Devi	Wanita	Pegawai Swasta	serlinda.ririkad@gmail.com
R39	Fiktor Setiawan	Pria	Pegawai Swasta	Fiktor.setiawan2507@gmail.com

LAMPIRAN D

DAFTAR JAWABAN

Tabel D.1
Daftar Jawaban

No	Variabel	Per tan ya an	Jawaban (Skala Likert 1 – 5)										Total Nilai	
			Sangat Baik (5)		Baik (4)		Cukup (3)		Buruk (2)		Sangat Buruk (1)			
			Jumlah	Total	Jumlah	Total	Jumlah	Total	Jumlah	Total	Jumlah	Total		
1	<i>Usability</i>	P1	4	20	11	44	17	51	7	14	0	0	129	
2	<i>Usability</i>	P2	2	10	8	32	11	33	14	28	4	4	107	
3	<i>Usability</i>	P3	11	55	16	64	11	33	0	0	1	1	153	
4	<i>Usability</i>	P4	11	55	17	68	11	33	0	0	0	0	156	
5	<i>Functionality</i>	P5	14	70	15	60	10	30	0	0	0	0	160	
6	<i>Functionality</i>	P6	13	65	23	92	3	9	0	0	0	0	166	
7	<i>Functionality</i>	P7	6	30	29	116	4	12	0	0	0	0	158	
8	<i>Functionality</i>	P8	12	60	23	92	3	9	1	2	0	0	163	
9	<i>Functionality</i>	P9	15	75	22	88	2	6	0	0	0	0	169	
10	<i>Functionality</i>	P10	7	35	20	80	10	30	2	4	0	0	149	
11	<i>Functionality</i>	P11	11	55	22	88	5	15	0	0	1	1	159	
12	<i>Functionality</i>	P12	11	55	23	92	4	12	1	2	0	0	161	
13	<i>Functionality</i>	P13	15	75	18	72	4	12	2	4	0	0	163	
14	<i>Functionality</i>	P14	7	35	25	100	6	18	1	2	0	0	155	
15	<i>Reliability</i>	P15	21	105	6	24	12	36	0	0	0	0	165	
16	<i>Reliability</i>	P16	25	125	2	8	11	33	1	2	0	0	168	
17	<i>Reliability</i>	P17	24	120	3	12	12	36	0	0	0	0	168	
18	<i>Efficiency</i>	P18	4	20	14	56	20	60	1	2	0	0	138	
19	<i>Efficiency</i>	P19	4	20	11	44	23	69	1	2	0	0	135	
20	<i>Efficiency</i>	P20	2	10	16	64	20	60	1	2	0	0	136	

**Tabel D.1
(Lanjutan)**

No	Variabel	Per tan ya an	Jawaban (Skala Likert 1 – 5)										Total Nilai	
			Sangat Baik (5)		Baik (4)		Cukup (3)		Buruk (2)		Sangat Buruk (1)			
			Jumlah	Total	Jumlah	Total	Jumlah	Total	Jumlah	Total	Jumlah	Total		
21	<i>Efficiency</i>	P21	9	45	15	60	14	42	1	2	0	0	149	
22	<i>Efficiency</i>	P22	12	60	13	52	13	39	1	2	0	0	153	
23	<i>Efficiency</i>	P23	20	100	10	40	8	24	1	2	0	0	166	
24	<i>Efficiency</i>	P24	12	60	18	72	9	27	0	0	0	0	159	
25	<i>Efficiency</i>	P25	6	30	18	72	12	36	3	6	0	0	144	