



IMPLEMENTASI ALGORITMA BOYER MOORE UNTUK PERNYARINGAN EMAIL

Skripsi

diajukan sebagai salah satu persyaratan untuk memperoleh gelar Sarjana Pendidikan Program
Studi Pendidikan Teknik Informatika dan Komputer

Oleh:

Wahyu Prakoso

NIM. 5302415028

**PENDIDIKAN TEKNIK INFORMATIKA DAN KOMPUTER
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS NEGERI SEMARANG
2020**

PERSETUJUAN PEMBIMBING

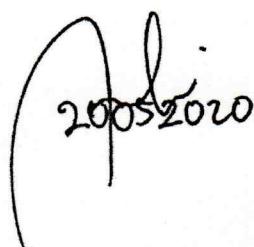
Nama : Wahyu Prakoso
NIM : 5302415028
Program Studi : S-1 Pendidikan Teknik Informatika dan Komputer
Judul Skripsi : Implementasi Algoritma Boyer Moore Untuk Penyaringan

Email Spam

Proposal skripsi ini telah disetujui oleh pembimbing untuk diajukan ke panitia ujian skripsi Program Studi S-1 Pendidikan Teknik Informatika dan Komputer Fakultas Teknik Universitas Negeri Semarang.

Semarang, 20 Mei 2020

Dosen Pembimbing,



Arief Arfandi's handwritten signature in black ink, consisting of stylized letters and the date 20052020 written vertically below it.

Arief Arfandi, S.T., M.Eng.
NIP.198208242014041001

PENGESAHAN

Skripsi dengan judul "Implementasi Algoritma Boyer Moore untuk Pernyaringan Email" telah dipertahankan di hadapan sidang Panitia Ujian Skripsi Jurusan Teknik Elektro Fakultas Teknik UNNES pada tanggal 17 Juni 2020

Oleh

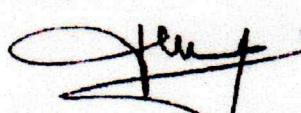
Nama Wahyu Prakoso
NIM 5302415028
Program Studi Pendidikan Teknik Informatika dan Komputer

Panitia:

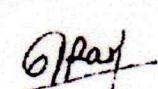
Ketua Panitia

Sekretaris


Ir. Ulfah Mediati Arief, M.T., IPM
NIP. 196605051997022001


Budi Sunarko, S.T., M.T., Ph.D.
NIP. 197101042006041001

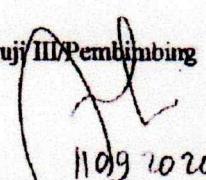
Pengaji I


Dr. Ing. Dhidik P., S.T., M.T.
NIP. 197805312005011002

Pengaji II


Dr. B. U. S.T., M.T.
NIP. 19809092012121002

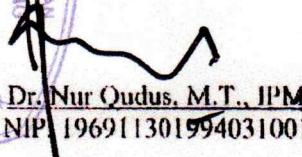
Pengaji III/Pembimbing


Arief A., S.T., M.Eng.
NIP. 198208242014041001

Mengetahui

Dekan Fakultas Teknik UNNES




Dr. Nur Qudus, M.T., IPM
NIP. 196911301994031001

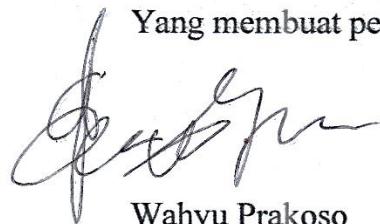
PERNYATAAN KEASLIAN

Dengan ini saya menyatakan bahwa:

1. Skripsi/TA ini, adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (sarjana, magister, dan/atau doktor), baik di Universitas Negeri Semarang (UNNES) maupun di perguruan tinggi lain.
2. Karya tulis ini adalah murni gagasan, rumusan, dan penelitian saya sendiri, tanpa bantuan pihak lain, kecuali arahan Pembimbing dan masukan Tim Pengaji.
3. Dalam karya tulis ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya ini, serta sanksi lainnya sesuai dengan norma yang berlaku di perguruan tinggi ini.

Semarang, 20 April 2020

Yang membuat pernyataan,



Wahyu Prakoso

NIM. 5302415028

MOTTO DAN PERSEMPAHAN

Motto:

“Bahagia”

“Apapun yang terjadi adalah yang terbaik”

“*Never stop learning because life never stop teaching*”

“Dalam hidup ini harus mempunyai tujuan hidup agar bisa terus melangkah dan tidak kehilangan arah”

“Hadapi! Jangan melarikan diri! Hidup itu melangkah ke depan bukan melangkah ke belakang. Manfaatkanlah waktu dengan baik! Waktu dapat digunakan untuk menghasilkan uang tapi uang tidak bisa membeli waktu dan tidak bisa membeli kesempatan kedua”

“Jangan mau terpedaya oleh rayuan menunda-nunda dan berangan-angan panjang sebab setiap detik umur yang terlewatkan dari umur tidak akan tergantikan”

(Hadratussyaikh KH. M. Hasyim Asy’ari)

Persembahan:

1. Bapak dan ibu saya tercinta yang senantiasa mendoakan dan memberikan dukungan kepada saya. Mohon maaf untuk selama ini, saya harap masih mempunyai sisa waktu untuk membahagiakan dan berbakti kepada kalian.
2. Teman-teman seperjuangan Pendidikan Teknik Informatika dan Komputer angkatan 2015.
3. Orang-orang yang hadir di hidup saya yang telah memberikan pelajaran tentang kehidupan yang sangat bermakna.

ABSTRAK

Wahyu Prakoso. 2020. Implementasi Algoritma Boyer Moore untuk Penyaringan *Email*. Skripsi. Pendidikan Teknik Informatika dan Komputer Jurusan Teknik Elektro. Fakultas Teknik Universitas Negeri Semarang.

Pembimbing: Arief Arfandi, S.T., M.Eng.

Email merupakan media komunikasi di internet, sudah banyak *provider email* yang ada, setiap *provider* memiliki sistem penyaringan *email* tersendiri, namun sistem penyaringan *email* yang dimiliki *provider* masih terdapat kesalahan dalam mengkategorikan *email* sebagai *spam*, serta tidak terdapat pemberitahuan kepada pengguna sehingga dapat dimungkinkan *email* penting terlewatt untuk dibaca. Tujuan dari penelitian ini adalah merancang dan membangun sistem penyaringan *email* dan mengimplementasikan algoritma Boyer Moore dengan pengembangan menggunakan *framework* Laravel.

Dalam penelitian ini menggunakan metode *waterfall* yang memiliki lima tahapan yaitu *communication*, *planning*, *modeling*, *construction*, dan *deployment*. Pengujian yang dilakukan diantaranya pengujian performa algoritma, *white box*, dan *black box*. Untuk pengujian *black box* aspek yang diuji adalah *functionality*, *efficiency*, *portability*, dan *usability*. Pengujian menggunakan data *email* hasil dari sinkronisasi melalui *Internet Message Access Protocol* (IMAP) ke *server Email Service Provider* (ESP) yang digunakan pengguna.

Hasil penelitian performa algoritma Boyer Moore pada sistem penyaringan *email* dapat dikategorikan cepat dikarenakan proses pencocokan teks dan *pattern* dalam waktu kurang dari satu detik yaitu rata-rata 20,85 μ s. Performa algoritma Boyer Moore dapat maksimal atau lebih cepat dalam proses pencocokan jika teks dan *pattern* memiliki jumlah karakter unik yang banyak pada teks atau *pattern* yang panjang. Hasil pengujian *white box* diperoleh nilai *cyclometric complexity* yaitu 5 yang berarti algoritma Boyer Moore merupakan algoritma yang sederhana dan memiliki resiko rendah. Pengujian *black box* memperhatikan beberapa aspek yaitu *functionality*, *portability*, *efficiency*, dan *usability*. Pada aspek *functionality* memperoleh hasil 100% yang berarti semua fungsi berjalan dan berfungsi dengan baik. Pada aspek *portability* memperoleh hasil 100% yang berarti sistem dapat diakses dari berbagai macam *browser* dan *device*. Pada aspek *efficiency* sistem sudah baik karena memperoleh rata-rata *grade* PageSpeed A (93%) dan Yslow B (84%), pada rata-rata *page size* 362 KB dengan *load time* rata-rata 3,2 detik. Pada aspek *usability* memperoleh hasil skor SUS yaitu 77,84. Jika berdasarkan cara penilaian SUS maka pada aspek *usability* sistem termasuk pada kategori *acceptable*, memperoleh *grade C*, dan *adjective rating* pada kategori *excellent*.

Kata Kunci: Penyaringan *Email*, Algoritma Boyer Moore, Pencocokan *Pattern*, *Framework* Laravel.

DAFTAR ISI

LEMBAR PERSETUJUAN PEMBIMBING	ii
LEMBAR PENGESAHAN	iii
LEMBAR PERNYATAAN KEASLIAN	iv
MOTTO DAN PERSEMBERAHAN	v
ABSTRAK	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Identifikasi Masalah	6
1.3 Batasan Masalah	6
1.4 Perumusan Masalah	7
1.5 Tujuan Penelitian.....	7
1.6 Manfaat Penelitian.....	7
BAB II KAJIAN PUSTAKA DAN LANDASAN TEORI	8
2.1 Kajian Pustaka	8
2.2 Dasar Teori	11
2.2.1 Algoritma.....	11
2.2.2 Algoritma Boyer Moore	14

2.2.3 <i>Email</i>	18
2.2.4 <i>Internet Message Access Protocol (IMAP)</i>	19
2.2.5 <i>Simple Mail Transfer Protocol (SMTP)</i>	20
2.2.6 <i>HTML</i>	20
2.2.7 <i>Hypertext Preprocessor (PHP)</i>	21
2.2.8 <i>Model-View-Controller (MVC)</i>	22
2.2.9 <i>Framework Laravel</i>	22
2.2.10 <i>MySQL</i>	23
2.2.11 <i>Flowchart</i>	23
2.2.12 <i>Use Case</i>	25
2.2.13 <i>ISO 9126</i>	26
BAB III METODE PENELITIAN.....	28
3.1 Waktu dan Tempat Pelaksanaan.....	28
3.2 Desain Penelitian	28
3.2.1 <i>Communication</i>	29
3.2.2 <i>Planning</i>	29
3.2.3 <i>Modelling (Analysis & Design)</i>	30
3.2.3.1 Desain <i>Database</i>	30
3.2.3.2 <i>Use Case Diagram</i>	32
3.2.3.3 Desain <i>Interface (Mockup)</i>	34
3.2.4 <i>Construction</i>	56
3.2.5 <i>Deployment</i>	60
3.3 Alat dan Bahan Penelitian	61

3.3.1 <i>Hardware</i>	61
3.3.2 <i>Software</i>	62
3.4 Teknik Pengumpulan Data	63
3.4.1 Instrumen Performa Algoritma	63
3.4.2 Instrumen <i>White Box</i>	63
3.4.3 Instrumen <i>Black Box</i>	64
3.4.3.1 Instrumen <i>Functionality</i>	64
3.4.3.2 Instrumen <i>Portability</i>	64
3.4.3.3 Instrumen <i>Efficiency</i>	64
3.4.3.4 Instrumen <i>Usability</i>	64
3.5 Teknik Analisis Data	65
3.5.1 Analisis <i>Basic Path Testing</i>	65
3.5.2 Analisis <i>Cyclomatic Complexity</i>	66
3.5.3 Analisis <i>Independent Path</i>	67
3.5.4 Analisis <i>Test Case</i>	67
3.5.5 Analisis <i>Functionality</i>	68
3.5.6 Analisis <i>Portability</i>	68
3.5.7 Analisis <i>Efficiency</i>	69
3.5.8 Analisis <i>Usability</i>	69
BAB IV HASIL DAN PEMBAHASAN	73
4.1 Hasil Penelitian.....	73
4.1.1 Hasil Desain Sistem	73
4.1.1.1 Hasil Desain <i>Database</i>	73

4.1.1.1 Hasil Desain Tampilan.....	78
4.1.2 Hasil Pengujian	95
4.1.2.1 Hasil Pengujian <i>White Box</i>	95
4.1.2.2 Hasil Pengujian Performa Algoritma.....	107
4.1.2.3 Hasil Pengujian <i>Black Box</i>	116
4.2 Pembahasan	130
BAB V KESIMPULAN DAN SARAN.....	134
5.1 Kesimpulan	134
5.2 Saran	135
DAFTAR PUSTAKA	136
LAMPIRAN.....	139

DAFTAR TABEL

Tabel 2.1 Nilai MH dan OH.....	15
Tabel 2.2 Simbol-Simbol <i>Flowchart</i>	24
Tabel 2.3 Simbol-Simbol <i>Use Case</i>	25
Tabel 3.1 Jadwal Estimasi Waktu	29
Tabel 3.2 Definisi <i>Use Case</i> Sistem Penyaringan <i>Email</i>	33
Tabel 3.3 Spesifikasi Laptop Dell N4050	61
Tabel 3.4 Spesifikasi VPS	62
Tabel 3.5 <i>Software</i> yang digunakan pada penelitian.....	62
Tabel 3.6 Instrumen Pengujian <i>System Usability Scale</i> (SUS)	64
Tabel 3.7 Hubungan <i>Cyclomatic Complexity</i> dengan Resiko.....	67
Tabel 4.1 Tanda <i>Email Spam</i>	88
Tabel 4.2 Skenario <i>Test Case</i> Jalur Independen 1	98
Tabel 4.3 Hasil Skenario <i>Test Case</i> Jalur Independen 1	98
Tabel 4.4 Skenario <i>Test Case</i> Jalur Independen 2	99
Tabel 4.5 Hasil Skenario <i>Test Case</i> Jalur Independen 2	100
Tabel 4.6 Skenario <i>Test Case</i> Jalur Independen 3	102
Tabel 4.7 Hasil Skenario <i>Test Case</i> Jalur Independen 3	102
Tabel 4.8 Skenario <i>Test Case</i> Jalur Independen 4	104
Tabel 4.9 Hasil Skenario <i>Test Case</i> Jalur Independen 4	104
Tabel 4.10 Skenario <i>Test Case</i> Jalur Independen 5	106
Tabel 4.11 Hasil Skenario <i>Test Case</i> Jalur Independen 5	107

Tabel 4.12 Pengujian Performa Algoritma Penyaringan <i>Email</i> Berdasarkan Subyek.....	108
Tabel 4.13 Pengujian Performa Algoritma Penyaringan <i>Email</i> Berdasarkan <i>Email</i>	114
Tabel 4.14 Pengujian Performa Algoritma Penyaringan <i>Email</i> Berdasarkan <i>Domain</i>	115
Tabel 4.15 Pengujian Performa Algoritma Penyaringan <i>Email</i> Berdasarkan TLD	116
Tabel 4.16 Pengujian Aspek <i>Functionality</i>	117
Tabel 4.17 Pengujian Aspek <i>Portability</i>	123
Tabel 4.18 Pengujian Aspek <i>Efficiency</i>	128
Tabel 4.19 Hasil Pengujian Aspek <i>Usability</i>	128

DAFTAR GAMBAR

Gambar 3.1. <i>Model Classic Life Cycle</i>	28
Gambar 3.2. Desain <i>Database</i> Sistem Penyaringan <i>Email</i>	31
Gambar 3.3. <i>Use Case Diagram</i> Sistem Penyaringan <i>Email</i>	32
Gambar 3.4. Halaman Masuk	34
Gambar 3.5. Halaman Lupa Kata Sandi.....	35
Gambar 3.6. Halaman Daftar	35
Gambar 3.7. Halaman Atur Ulang Kata Sandi.....	36
Gambar 3.8. Halaman Beranda	37
Gambar 3.9. Halaman Informasi Pribadi	38
Gambar 3.10. Halaman Ubah Kata Sandi	39
Gambar 3.11. Halaman IMAP	40
Gambar 3.12. Halaman SMTP	41
Gambar 3.13. Halaman Daftar Anti <i>Spam</i>	42
Gambar 3.14. Halaman Tulis	43
Gambar 3.15. Halaman Kotak Masuk.....	44
Gambar 3.16. Halaman Berbintang.....	45
Gambar 3.17. Halaman Terkirim	46
Gambar 3.18. Halaman Draf	47
Gambar 3.19. Halaman <i>Spam</i>	48
Gambar 3.20. Halaman Sampah.....	49

Gambar 3.21. Halaman Lihat <i>Email</i>	50
Gambar 3.22. Halaman Petunjuk	51
Gambar 3.23. Halaman Pengguna Aktif	52
Gambar 3.24. Halaman Pengguna Baru.....	53
Gambar 3.25. Halaman Ubah Data Pengguna	54
Gambar 3.26. Email Atur Ulang Kata Sandi.....	55
Gambar 3.27. Email Aktivasi Akun.....	55
Gambar 3.28. <i>Function</i> MHTable	57
Gambar 3.29. <i>Function</i> BoyerMoore	59
Gambar 3.30. Notasi <i>Flow Graph</i>	65
Gambar 3.31. Penilaian <i>System Usability Scale</i> (SUS).....	71
Gambar 4.1. Hasil Tabel <i>Activations</i>	73
Gambar 4.2. Hasil Tabel <i>Emails</i>	74
Gambar 4.3. Hasil Tabel Filterdomain.....	74
Gambar 4.4. Hasil Tabel Filteremail.....	75
Gambar 4.5. Hasil Tabel Filtersubject	75
Gambar 4.6. Hasil Tabel Filtertld	76
Gambar 4.7. Hasil Tabel <i>Migrations</i>	76
Gambar 4.8. Hasil Tabel <i>Password_resets</i>	77
Gambar 4.9. Hasil Tabel <i>Roles</i>	77
Gambar 4.10. Hasil Tabel <i>Role_user</i>	77
Gambar 4.11. Hasil Tabel <i>Users</i>	78
Gambar 4.12. Hasil Tampilan Halaman Masuk.....	79

Gambar 4.13. Hasil Tampilan Halaman Daftar	80
Gambar 4.14. Hasil Tampilan Halaman Lupa Kata Sandi.....	80
Gambar 4.15. Hasil Tampilan Halaman Atur Ulang Kata Sandi	81
Gambar 4.16. Hasil Tampilan Halaman Beranda	82
Gambar 4.17. Hasil Tampilan Halaman Informasi Pribadi.....	82
Gambar 4.18. Hasil Tampilan Halaman Ubah Kata Sandi	83
Gambar 4.19. Hasil Tampilan Halaman IMAP.....	83
Gambar 4.20. Hasil Tampilan Halaman SMTP	84
Gambar 4.21. Hasil Tampilan Halaman Daftar Anti <i>Spam</i>	84
Gambar 4.22. Hasil Tampilan Halaman Tulis	85
Gambar 4.23. Hasil Tampilan Halaman Kotak Masuk	86
Gambar 4.24. Hasil Tampilan Halaman Berbintang	86
Gambar 4.25. Hasil Tampilan Halaman Terkirim	87
Gambar 4.26. Hasil Tampilan Halaman Draf	87
Gambar 4.27. Hasil Tampilan Halaman <i>Spam</i>	88
Gambar 4.28. Hasil Tampilan Halaman Sampah.....	89
Gambar 4.29. Hasil Tampilan Halaman Lihat <i>Email</i>	90
Gambar 4.30. Hasil Tampilan Halaman Petunjuk	91
Gambar 4.31. Hasil Tampilan Halaman Pengguna Aktif	92
Gambar 4.32. Hasil Tampilan Halaman Pengguna Baru	92
Gambar 4.33. Hasil Tampilan Halaman Ubah Data Pengguna.....	93
Gambar 4.34. Hasil Tampilan <i>Email</i> Atur Ulang Kata Sandi.....	94
Gambar 4.35. Hasil Tampilan Email Aktivasi Akun	94

Gambar 4.36. <i>Flow Graph</i> Algoritma Boyer Moore	95
Gambar 4.37. Pengujian Aspek <i>Efficiency</i> pada Halaman Masuk.....	126
Gambar 4.38. Pengujian Aspek <i>Efficiency</i> pada Halaman Daftar	127
Gambar 4.39. Pengujian Aspek <i>Efficiency</i> pada Halaman Lupa Kata Sandi.....	127

LAMPIRAN

Lampiran 1 Surat Penetapan Dosen Pembimbing Skripsi/Tugas Akhir	139
Lampiran 2 Formulir Usulan Topik Skripsi.....	140
Lampiran 3 Surat Tugas Seminar Proposal Skripsi	141
Lampiran 4 SUS Responden 1	142
Lampiran 5 SUS Responden 2	143
Lampiran 6 SUS Responden 3	144
Lampiran 7 SUS Responden 4	145
Lampiran 8 SUS Responden 5	146
Lampiran 9 SUS Responden 6	147
Lampiran 10 SUS Responden 7	148
Lampiran 11 SUS Responden 8	149
Lampiran 12 SUS Responden 9	150
Lampiran 13 SUS Responden 10	151
Lampiran 14 SUS Responden 11	152
Lampiran 15 SUS Responden 12	153
Lampiran 16 SUS Responden 13	154
Lampiran 17 SUS Responden 14	155
Lampiran 18 SUS Responden 15	156
Lampiran 19 SUS Responden 16	157
Lampiran 20 SUS Responden 17	158
Lampiran 21 SUS Responden 18	159

Lampiran 22 SUS Responden 19	160
Lampiran 23 SUS Responden 20	161
Lampiran 24 SUS Responden 21	162
Lampiran 25 SUS Responden 22	163
Lampiran 26 SUS Responden 23	164
Lampiran 27 SUS Responden 24	165
Lampiran 28 SUS Responden 25	166
Lampiran 29 SUS Responden 26	167
Lampiran 30 SUS Responden 27	168
Lampiran 31 SUS Responden 28	169
Lampiran 32 SUS Responden 29	170
Lampiran 33 SUS Responden 30	171
Lampiran 34 SUS Responden 31	172
Lampiran 35 SUS Responden 32	173
Lampiran 36 SUS Responden 33	174
Lampiran 37 SUS Responden 34	175
Lampiran 38 SUS Responden 35	176
Lampiran 39 SUS Responden 36	177

BAB I

PENDAHULUAN

1.1 Latar Belakang

Menurut perusahaan manajemen hubungan pelanggan EveryAction, rata-rata 7 persen dari *email* nirlaba dikirim ke kotak *spam*, bukan ke *inbox*. Ada banyak hal yang berbeda dalam menentukan *spam* dan sulit untuk melacak penyebabnya sehingga sebagian besar organisasi nirlaba tidak melihat bagaimana hal itu memengaruhi mereka atau apa yang dapat mereka lakukan (Nonprofit Business Advisor, 2016: 1). *Email* dikirimkan berdasarkan reputasi IP dari mana *email* dikirim. Jika *Internet Service Provider* (ISP) mengetahui bahwa *email* yang dikirim ditandai sebagai *spam* maka sistem merutekan *email* ke *folder spam*, atau lebih buruk lagi sepenuhnya memblokir pengirim (Nonprofit Business Advisor, 2016: 2).

Banyak *provider email* yang tersedia, setiap *provider email* mempunyai sistem penyaringan *email* yang berbeda-beda. Pada sistem panyaringan *email* yang *provider* terapkan dimungkinkan masih terdapat kesalahan dalam mengkategorikan *email spam*, hal tersebut pernah terjadi pada *provider email* Zoho terhadap *provider email* Outlook pada tahun 2018. *Email* yang terkirim dari Zoho dikategorikan sebagai *spam* oleh sistem penyaringan *email* Outlook meskipun pengguna Outlook tidak memasukan *provider* Zoho ke dalam *blacklist* di pengaturan penyaringan *email*. Pihak Zoho mengidentifikasi permasalahan tersebut apakah permasalahan tersebut berasal dari sistem Zoho atau dari Outlook. Permasalahan yang terjadi disebabkan oleh *Exchange Online Protection* (EOP) *server* Outlook yang salah

mengidentifikasi *email* yang dikirim dari Zoho sebagai *spam* karena EOP Outlook telah memblokir rentang IP milik Zoho 135.84.80.192/26 dan 136.143.188.0/23 (Ramya, 2019).

Alamat *email* terdiri dari dua bagian yang dipisahkan tanda *axon* (@). Bagian kiri dari tanda @ disebut nama pengguna yang menunjukkan identitas pemilik *email*. Bagian sebelah kanan dari tanda @ disebut nama *domain* yang menunjukkan identitas *mail server*. Nama *domain* dapat dirubah sesuai kebutuhan, biasanya menyesuaikan nama perusahaan atau organisasi. Sebuah nama *domain* memiliki bagian yang disebut *Top Level Domain* (TLD). Terdapat dua jenis TLD yaitu *Generic Top Level Domain* (gTLD) dan *Country Code Top Level Domain* (ccTLD), gTLD digunakan untuk berbagai keperluan umum sedangkan ccTLD digunakan khusus untuk negara tertentu (Catherine R. Easton, 2012: 275).

Berdasarkan program gTLD baru pada tahun 2012 *Internet Corporation for Assigned Names and Numbers* (ICANN) meluncurkan putaran pertama secara publik sehingga pemohon dapat mengajukan permohonan untuk nama gTLD baru di Net, program tersebut bertujuan untuk meningkatkan persaingan di pasar nama *domain* dan memperluas pilihan konsumen (Simone Vezzani, 2014: 307). Namun penggunaan gTLD yang baru diluncurkan belum mempunyai reputasi atau gTLD bereputasi buruk pada alamat *email* sehingga memiliki permasalahan yaitu *email* yang dikirim akan dianggap sebagai *spam* oleh sistem penyaringan *email* yang dimiliki *provider email*, sehingga *email* yang yang dianggap *spam* akan masuk ke *folder spam*.

Email yang masuk ke *folder spam* tidak akan memunculkan pemberitahuan kepada pengguna sehingga *email* akan terlewatkan untuk dibaca. Di *folder spam* berisi kumpulan *email* yang dikategorikan *spam* yang merupakan hasil dari sistem penyaringan *email provider* dan pengaturan penyaringan *email* oleh pengguna. *Email* di *folder spam* tidak terdapat sebuah tanda yang membedakan *email spam* hasil dari sistem penyaringan *email* yang dimiliki *provider* atau dari pengaturan pengguna. Hal tersebut dapat menyulitkan pengguna dalam memilah *email* yang ada di *folder spam*.

Shukor Bin Abd Razak dan Ahmad Fahrulrazie Bin Mohamad (2013: 347) menyatakan bahwa identifikasi atau penyaringan *email spam* dapat dilakukan berdasarkan informasi yang diperoleh dari *header email*. Hal tersebut menjadi latar belakang penulis ingin melakukan penelitian penyaringan *email spam* berdasarkan informasi dari *header email*.

Informasi atau data yang diperoleh pada *header email* berupa *string* maka penyaringan *email spam* akan menerapkan teknik *matching string*. Ada berbagai macam algoritma yang dapat diterapkan dalam teknik *matching string*, seperti algoritma Rabin Karp, Boyer Moore, Knuth-Morris-Pratt (KMP) dan Aho Corasick.

Sudah banyak penelitian yang meneliti kinerja berbagai algoritma *matching string* yang ada. Parenrengi, *et al.*, (2017) menganalisis perbandingan antara algoritma Boyer Moore dan algortima Knuth Morris Pratt, untuk menentukan algoritma yang paling baik digunakan dalam aplikasi Tripelka *Foodshop* Kendari. Parameter yang digunakan untuk membandingkan kedua algoritma tersebut adalah

waktu pencarian dan tingkat keakurasaan data yang ditampilkan. Hasil dari penelitian tersebut menunjukkan bahwa algoritma Boyer Moore dan algoritma Knuth Morris Pratt memiliki tingkat keakurasaan yang sama tetapi algoritma Boyer Moore adalah algoritma yang waktu pencarinya lebih cepat dibandingkan algoritma Knuth Morris Pratt.

Pranit Chettri dan Chinmoy Kar (2016: 26) meneliti berbagai algoritma pencarian teks yaitu algoritma Knuth-Morris Pratt, algoritma Pencarian Naïve dan algoritma Boyer Moore dengan menyediakan *input* teks berbagai ukuran dan menganalisis perilaku mereka pada *input* variabel. Setelah menganalisis dan melakukan studi pada algoritma, hasilnya menyatakan bahwa algoritma Boyer Moore bekerja dengan sangat baik dan efisien daripada yang lainnya ketika berhadapan dengan set data yang lebih besar.

Srinivas, *et al.*, (2018), Algoritma Knuth Morris Pratt bergerak maju hanya dalam urutan *input* dan membutuhkan ruang tambahan. Algoritma Rabin Karp digunakan untuk mengidentifikasi *plagiarisme* dan membutuhkan ruang tambahan untuk pencocokan. Algoritma Boyer Moore sangat cepat dalam urutan yang luas, Boyer Moore menghindari banyak perbandingan yang tidak berguna dengan pola pencarian yang secara mendasar berkaitan dengan konten. Kompleksitas *running case* terbaiknya adalah sub linear. Hasil menunjukkan bahwa Boyer-Moore adalah algoritma terbaik untuk mengatasi masalah pencocokan *string* dalam kasus-kasus biasa, dan Rabin-Karp adalah pilihan yang layak untuk beberapa kasus tertentu, misalnya ketika pola dan teks sangat kecil.

Rasool, *et al.*, (2012: 3397) meneliti algoritma pencocokan *string* Boyer Moore, Aho-Corasick dan KMP efisien. Algoritma Boyer Moore cepat dalam hal jumlah *string* yang lebih besar. Tinus Wuruwu dan Rila Mandala (2016: 42) menyatakan bahwa algoritma Boyer Moore lebih cepat daripada algoritma Knuth Morris Pratt dalam waktu akses pencocokan *string*.

Pada penelitian ini akan dibangun sistem penyaringan *email* berbasis *web* yang dapat diakses melalui *browser*. Agar memudahkan dalam pembuatan sistem penyaringan *email* menggunakan *framework*. Ada banyak jenis *framework* antara lain Codeigniter, Symfony, dan Laravel. Laaziri, *et al.*, (2019: 870) melakukan pengujian terhadap *framework* Codeigniter, Symfony, dan Laravel menggunakan metode *Qualification and Selection of Opensource Software* (QSOS) yang secara resmi diakui sebagai standar untuk membandingkan *framework*. Hasil penelitian dapat disimpulkan bahwa *framework* Laravel dan Symfony saat ini memenuhi standar dan persyaratan modern. Laravel dan Symfony memiliki fitur yang banyak dibandingkan dengan Codeigniter yang tidak memenuhi persyaratan saat ini. Untuk evaluasi kinerja tiga *framework*: Laravel, Symfony, dan Codeigniter sesuai dengan kriteria evaluasi seperti permintaan per detik, penggunaan memori, waktu respon dan jumlah *file* yang diperlukan, Hasil yang diperoleh menunjukkan bahwa Laravel melampaui *framework* MVC lainnya. Laravel digunakan untuk pengembangan aplikasi skala besar yang cepat.

Berdasarkan penelitian relevan tersebut penulis membandingkan beberapa algoritma dan menetapkan algoritma Boyer Moore untuk diimplementasikan dalam

penelitian ini, karena data *string* yang diperoleh dari *header* berjumlah besar. Untuk pembuatan sistem menggunakan *framework* Laravel.

1.2 Identifikasi Masalah

Berdasarkan latar belakang yang telah diuraikan, dapat diidentifikasi beberapa permasalahan sebagai berikut:

1. Sistem penyaringan *email* di *provider* dapat terjadi kesalahan dalam mengkategorikan *email* sebagai *spam*.
2. Penggunaan gTLD yang baru diluncurkan belum mempunyai reputasi atau gTLD bereputasi buruk pada alamat *email* memiliki permasalahan yaitu *email* yang dikirim dianggap sebagai *spam* oleh sistem penyaringan *email* yang dimiliki *provider email*.
3. *Email-email* yang berada pada *folder spam* tidak ada tanda yang melatarbelakangi *email* tersebut bisa dianggap sebagai *spam* sehingga pengguna akan sulit memilah.

1.3 Batasan Masalah

Agar tidak melebar dari latar belakang dan permasalahan yang telah dijelaskan sebelumnya maka dalam menyelesaikan penelitian ini penulis membuat batasan masalah sebagai berikut:

1. Penyaringan *email* sebagai *spam* berdasarkan data pada *header email* dengan parameter: *subject*, *email*, *domain*, dan *top level domain*.
2. Sistem penyaringan *email* dirancang menggunakan *framework* Laravel.
3. Menggunakan algoritma Boyer Moore untuk menyaring *email*.

1.4 Perumusan Masalah

Berdasarkan uraian di atas, penelitian ini difokuskan untuk mengimplementasikan algoritma Boyer Moore pada sistem penyaringan *email*, yang permasalahannya diperoleh sebagai berikut:

1. Bagaimana merancang dan membuat sistem penyaringan *email*?
2. Bagaimana menerapkan algoritma Boyer Moore pada sistem penyaringan *email*?
3. Bagaimana hasil pengujian menggunakan pengujian performa algoritma, *white box* dan *black box* pada sistem penyaringan *email*?

1.5 Tujuan Penelitian

Tujuan dari penelitian ini adalah :

1. Merancang dan membuat sistem penyaringan *email*.
2. Menerapkan algoritma Boyer Moore pada sistem penyaringan *email*.
3. Mengetahui hasil pengujian performa algoritma, *white box* dan *black box* pada sistem penyaringan *email* yang telah dibuat.

1.6 Manfaat Penelitian

Manfaat dari penelitian ini adalah :

1. Terbentuknya sebuah sistem penyaringan *email*.
2. Dapat mengetahui kelebihan dan kekurangan dari algoritma Boyer Moore dalam menyaring *email*.
3. Sebagai bahan studi kasus bagi pembaca dan acuan bagi mahasiswa, terutama bagi yang ingin melakukan penelitian sejenis.

BAB II

KAJIAN PUSTAKA DAN LANDASAN TEORI

2.1 Kajian Pustaka

Penelitian yang relevan dapat diperoleh dengan mencari referensi seperti jurnal, tesis, dan penelitian terdahulu seperti berikut:

Eza Rahmanita (2014: 15) dalam penelitian “Pencarian *String* Menggunakan Algoritma Boyer Moore pada Dokumen” menyatakan proses pencarian merupakan salah satu kegiatan penting dalam pemrosesan data. Proses ini dapat menghabiskan waktu dalam ruang pencarian yang besar sehingga diperlukan suatu teknik pencarian yang efisien. Algoritma Boyer Moore merupakan suatu solusi pencarian yang efisien dapat melakukan perbandingan *pattern* mulai dari kanan ke kiri. Jika terjadi ketidakcocokan *string* dari kanan *pattern* maka ketidakcocokan akan membantu menggerakkan *pattern* tersebut dengan jarak yang lebih jauh. Gerakan melompat ini akan memberikan informasi berapa banyak *pattern* harus digeser untuk mencocokan karakter terakhir yang cocok dengan kemunculan awal *pattern*. Artinya, akan lebih signifikan dalam mengurangi proses perbandingan, jika bisa melompati atau tidak melakukan perbandingan karakter yang diprediksi akan gagal. Algoritma Boyer Moore mempunyai keunggulan dalam waktu menemukan *pattern* yang akan dicari dalam ukuran *file* yang lebih besar. Pada *file* berekstensi .txt dengan *file size* 4.625 byte dengan varian *keyword* berbeda. Varian *keyword* yang sedikit dapat ditempuh dengan waktu lebih cepat pada *pattern* ‘a’ yaitu 0,228 detik dengan banyak *pattern* ditemukan 685. Pada *file* berekstensi .doc dengan *file size*

39.936 byte, pada *pattern* ‘yang’ dapat diproses dengan waktu 0,542 detik dengan ditemukannya *pattern* sebanyak 18. Sedangkan pada *file* berekstensi .pdf optimalisasi dari *pattern matching* yang diproses adalah 0,103 detik dengan *file size* 15.804 byte dan banyak *pattern* = 1 untuk pencarian *pattern* ‘suatu’.

Baranwal, *et al.*, (2018: 37) dalam penelitian “*Spam Filtration using Boyer Moore Algorithm and Naïve Method*” mengembangkan *classifier* Naïve Bayes klasik dengan teknik pembelajaran *non-mesin* dan algoritma pencocokan pola. Setelah menganalisis kinerja model yang diusulkan, dapat disimpulkan bahwa algoritma Boyer Moore meningkatkan kinerja dalam bentuk akurasi proses klasifikasi.

Shukor Bin Abd Razak dan Ahmad Fahrulrazie Bin Mohamad (2013: 353) dalam penelitian “*Identification of Spam Email Based on Information from Email Header*” menyatakan bahwa ada banyak teknik deteksi *spam* yang telah diusulkan dan diimplementasikan. *Spammer* dapat menggunakan teknik-teknik canggih untuk menghindari sistem penyaringan. Tidak cukup untuk mendeteksi berbagai jenis pesan *spam* dengan melihat informasi di badan *email*. Berdasarkan informasi yang ada di *header* menjadi pendekatan yang populer di kalangan peneliti untuk mengidentifikasi fitur potensial yang dapat digunakan untuk mendeteksi perilaku *spam*.

Rahim, *et al.*, (2017: 4) dalam penelitian “*Visual Approach of Searching Process using Boyer-Moore Algorithm*” menyatakan bahwa proses visualisasi dari algoritma Boyer Moore memudahkan peneliti atau pelajar untuk mempelajari bagaimana cara kerja algoritma Boyer Moore, dan dalam pengembangan aplikasi

akan lebih mudah untuk membuat fungsi pencarian kata dan dapat diimplementasikan ke dalam banyak proses pencarian.

Fince Tinus Wuruwu dan Rila Mandala (2016: 42) dalam penelitian “Perbandingan Algoritma Knuth Morris Pratt dan Boyer Moore dalam Pencocokan *String* pada Aplikasi Kamus Bahasa Nias” menyimpulkan pada perbandingan algoritma Knuth Morris Pratt dan Boyer Moore yang lebih cepat dalam waktu akses pencocokan *string* adalah algoritma Boyer.

Pranit Chettri dan Chinmoy Kar (2016: 26) dalam penelitian “*Comparative Study between Various Pattern Matching Algorithms*” meneliti berbagai algoritma pencarian teks yaitu algoritma Knuth-Morris Pratt, algoritma Pencarian Naïve dan algoritma Boyer Moore dengan menyediakan *input* teks berbagai ukuran dan menganalisis perilaku mereka pada *input* variabel. Setelah menganalisis dan melakukan studi pada algoritma, hasilnya menyatakan bahwa algoritma Boyer Moore bekerja dengan sangat baik dan efisien daripada yang lainnya ketika berhadapan dengan set data yang lebih besar.

Srinivas, *et al.*, (2018), dalam penelitian “*Study of String Matching Algorithm*” menyimpulkan algoritma Knuth Morris Pratt bergerak maju hanya dalam urutan *input* dan membutuhkan ruang tambahan. Algoritma Rabin Karp digunakan untuk mengidentifikasi plagiarisme dan membutuhkan ruang tambahan untuk pencocokan. Algoritma Boyer Moore sangat cepat dalam urutan yang luas, Boyer Moore menghindari banyak perbandingan yang tidak berguna dengan pola pencarian yang secara mendasar berkaitan dengan konten; kompleksitas *running case* terbaiknya adalah *sub linear*. Hasil menunjukkan bahwa Boyer Moore adalah

algoritma terbaik untuk mengatasi masalah pencocokan *string* dalam kasus-kasus biasa, dan Rabin Karp adalah pilihan yang layak untuk beberapa kasus tertentu, misalnya ketika pola dan teks sangat kecil.

Vina Sagita dan Maria Irmina Prasetyowati (2013: 31) dalam penelitian “Studi Perbandingan Implementasi Algoritma Boyer Moore, Turbo Boyer Moore, dan *Tuned* Boyer-Moore dalam Pencarian *String*” menyatakan bahwa algoritma Boyer Moore memiliki waktu pencarian tercepat dari tiga varian Boyer Moore yang telah ditentukan. Algoritma Turbo Boyer Moore merupakan algoritma tercepat kedua dan yang paling lambat adalah *Tuned* Boyer Moore.

Berdasarkan hasil uji performa dari penelitian yang relevan menetapkan algoritma Boyer Moore untuk diimplementasikan pada sistem penyaringan *email* di dalam penelitian ini. Kecepatan pencocokan adalah faktor yang utama dikarenakan data *string* pada *header* berjumlah besar.

2.2 Dasar Teori

2.2.1 Algoritma

Lamhot Sitorus (2015: 2) algoritma berasal dari kata *al-kwarizmi* yang terdapat di buku Abu Ja’far Muhammad Ibnu Musa Al-Kwarizmi, seorang ahli Matematika dari Persia dengan judul bukunya “Aljabar wal Muqabala”. Menurut Donald E. Knuth, algoritma yang baik memiliki kriteria sebagai berikut:

a. *Input*

Suatu algoritma harus memiliki 0 (nol) atau lebih masukan (*input*). Artinya, suatu algoritma itu dimungkinkan tidak memiliki masukan secara langsung dari pengguna tetapi dapat juga memiliki beberapa masukan. Algoritma yang tidak

memiliki masukan secara langsung dari pengguna, maka semua data dapat diinisialisasikan atau dibangkitkan dalam algoritma.

b. *Output*

Suatu algoritma harus memiliki satu atau lebih *output*. Suatu algoritma yang tidak memiliki keluaran (*output*) adalah suatu algoritma yang sia-sia, yang tidak perlu dilakukan. Algoritma dibuat untuk tujuan menghasilkan suatu yang diinginkan, yaitu berupa hasil keluaran.

c. *Finiteness*

Setiap pekerjaan yang dikerjakan pasti berhenti. Demikian juga algoritma harus dapat dijamin akan berhenti setelah melakukan sejumlah langkah proses.

d. *Definiteness*

Tidak menimbulkan makna ganda (*ambiguous*). Setiap baris aksi/ pernyataan dalam suatu algoritma harus pasti, artinya tidak menimbulkan penafsiran lain bagi setiap pembaca algoritma, sehingga memberikan *output* yang sesuai dengan yang diharapkan oleh pengguna.

e. *Effectiveness*

Langkah-langkah algoritma dikerjakan dalam waktu yang wajar. Suatu algoritma tidak terdapat suatu aksi yang tidak perlu dilakukan. Setiap aksi akan memerlukan waktu eksekusi, padahal aksi tersebut jelas tidak berpengaruh atau tidak ada gunanya. Misalnya aksi $X \leftarrow X + 0$. Aksi ini jelas tidak ada pengaruh dan tidak ada gunanya karena $X + 0$ akan menghasilkan bilangan X juga yang berarti tidak berguna. Jadi tidak perlu dilakukan karena sia-sia.

Lamhot Sitorus (2015: 5) secara umum, algoritma terdiri dari sekuensial (*sequential*), tes kondisi atau percabangan (*branching*) dan perulangan (*looping*).

1. Algoritma Sekuensial (*Sequential*)

Dalam kehidupan sehari-hari, algoritma yang sering dilakukan adalah algoritma sekuensial. Algoritma sekuensial adalah langkah-langkah atau aksi-aksi yang dilakukan secara berurutan sesuai dengan urutan penulisnya. Jika urutan penulisannya diubah, maka kemungkinan akan memberikan hasil akhir yang berbeda pula. Misalkan algoritma memiliki 4 baris aksi, yaitu t1, t2, t3 dan t4, maka semua aksi akan dilaksanakan secara berurutan mulai aksi t1 hingga t4. Setelah selesai mengerjakan aksi t1, maka aksi t2 akan dilaksanakan. Setelah selesai mengerjakan aksi t2, maka aksi t3 akan dilaksanakan. Setelah selesai mengerjakan aksi t3, maka aksi t4 akan dilaksanakan. Setelah selesai mengerjakan aksi t4, baru algoritma berhenti karena aksi t4 merupakan aksi terakhir.

2. Algoritma Percabangan (*Branching*)

Ada kalanya suatu kegiatan tertentu akan dilaksanakan atau tidak akan dilakukan karena tergantung situasi atau kondisi tertentu. Demikian juga dalam algoritma ada kalanya satu atau beberapa aksi akan dikerjakan atau tidak akan dikerjakan tergantung kondisi tertentu, akan dikerjakan jika kondisi tertentu dipenuhi. Percabangan ini hanya mengerjakan satu aksi dari dua atau lebih pilihan yang diberikan.

3. Algoritma Perulangan (*Looping*)

Banyak kegiatan sehari-hari yang harus dilakukan secara berulang-ulang, misalnya makan berjalan, naik tangga dan lain-lain. Dalam membuat algoritma juga mengenal perulangan, yaitu melaksanakan satu atau beberapa aksi secara berulang-ulang sesuai dengan kebutuhan atau kondisi. Salah satu kelebihan komputer dibandingkan dengan manusia adalah melaksanakan pekerjaan secara berulang-ulang tanpa mengenal istilah lelah atau bosan. Manusia dalam melaksanakan pekerjaan yang sama secara berulang-ulang akan mengalami kelelahan dan bosan yang cenderung akan melaksanakan kesalahan.

2.2.2 Algoritma Boyer Moore

Algoritma Boyer Moore diperkenalkan oleh Bob Boyer dan J.S. Moore pada tahun 1977, pada algoritma Boyer Moore pencocokan kata dimulai dari karakter terakhir kata kunci menuju karakter awalnya, jika terjadi perbedaan antara karakter terakhir kata kunci dengan kata yang dicocokkan maka karakter-karakter dalam potongan kata yang dicocokkan tadi akan diperiksa satu per satu, hal ini dimaksudkan untuk mendeteksi apakah ada karakter dalam potongan kata tersebut yang sama dengan karakter yang ada pada kata kunci (Eza Rahmanita, 2014: 17) .

Rahim, *et al.*, (2017: 2) langkah-langkah pencarian *pattern* pada teks menggunakan algoritma Boyer Moore sebagai berikut:

- a. Pertama, membutuhkan tabel yang mempunyai nilai *Match Heuristic* (MH), dan *Occurrence Heuristic* (OH) untuk menentukan jumlah perpindahan yang akan dilakukan pada suatu *pattern* jika ada yang tidak sesuai pada karakter dalam proses pencocokan karakter terhadap karakter teks.

- b. Jika dalam proses perbandingan ada ketidakcocokan antara karakter pada *pattern* dengan karakter pada teks dan karakter pada teks yang tidak cocok terdapat pada karakter *pattern* maka pergeseran dilakukan dengan menggunakan nilai terbesar dari nilai MH dan OH.
- c. Jika terjadi ketidakcocokan namun karakter yang tidak cocok pada teks tidak terdapat pada *pattern* maka *pattern* bergeser sebanyak panjang *pattern*.
- d. Jika karakter dalam teks cocok dengan karakter pada *pattern* maka karakter mencocokan karakter sebelah kiri yang belum dicocokan sampai karakter *pattern* yang paling kiri.
- e. Jika semua karakter memiliki kecocokan maka *pattern* telah ditemukan pada teks.

Contoh cara kerja algoritma Boyer Moore ini adalah sebagai berikut:

- a. Deklarasi teks dan *pattern*.

Teks: ABDBCADDABC

Pattern: ABC

Tabel 2.1 Nilai MH dan OH

<i>Index</i>	0	1	2	
Karakter	A	B	C	*
MH	2	1	0	3
OH	1			

Pada tabel 2.1 dapat dijelaskan bahwa hanya karakter unik pada *pattern* yang diambil pada tabel, *index* dimulai dari 0, nilai OH yaitu 1 dan nilai MH di

setiap karakter pada *pattern* diperoleh dengan cara panjang *pattern* dikurangi *index* dan dikurangi 1. Nilai MH dan OH merupakan jumlah perpindahan atau pergeseran *pattern* ke kanan, jika karakter pada teks yang tidak cocok terdapat pada *pattern* maka nilai MH dan OH dibandingkan dan diambil nilai terbesarnya. Untuk karakter yang tidak ada pada *pattern* maka nilai MH sama dengan panjang *pattern*.

b. Langkah pertama

<i>Index</i>	0	1	2	3	4	5	6	7	8	9	10	11
Teks	A	B	D	B	A	C	D	B	D	A	B	C
<i>Pattern</i>	A	B	C									

Karakter paling kanan dari *pattern* yaitu "C" tidak cocok dengan karakter yang sejajar pada teks yaitu "D". karakter "D" tidak terdapat pada *pattern* maka *pattern* berpindah sebanyak panjang *pattern*. Panjang *pattern* diketahui yaitu 3 maka *pattern* bergeser ke kanan sebanyak 3 karakter, dari *index* 2 menuju *index* 5.

c. Langkah kedua

<i>Index</i>	0	1	2	3	4	5	6	7	8	9	10	11
Teks	A	B	D	B	A	C	D	B	D	A	B	C
<i>Pattern</i>				A	B	C						

Karakter paling kanan dari *pattern* yaitu "C" cocok dengan karakter yang sejajar pada teks yaitu "C". Jika terjadi kecocokan maka langkah selanjutnya mencocokan karakter sebelah kiri dari karakter *pattern* yang sudah cocok. Karakter yang selanjutnya dicocokan yaitu karakter "B" terhadap karakter yang sejajar pada teks yaitu "A". ternyata hasilnya terjadi ketidakcocokan, karakter "A" yang terdapat pada *pattern* memiliki nilai MH yaitu 2. Jika karakter teks yang tidak cocok terdapat

pada *pattern* maka dilakukan perbandingan nilai MH dan OH. Diambil nilai yang paling besar diantara keduanya. Hasilnya diperoleh nilai terbesar yaitu 2 maka *pattern* bergeser ke kanan sebanyak 2 karakter, dari *index* 5 menuju *index* 7.

d. Langkah ketiga

<i>Index</i>	0	1	2	3	4	5	6	7	8	9	10	11
Teks	A	B	D	B	A	C	D	B	D	A	B	C
<i>Pattern</i>						A	B	C				

Karakter paling kanan dari *pattern* yaitu "C" tidak cocok dengan karakter yang sejajar pada teks yaitu "B". karakter "B" terdapat pada *pattern*. Jika karakter teks yang tidak cocok terdapat pada *pattern* maka dilakukan perbandingan nilai MH dan OH. Karakter "B" yang terdapat pada *pattern* memiliki nilai MH yaitu 1. Diambil nilai yang paling besar diantara keduanya. Hasilnya diperoleh nilai terbesar yaitu 1 maka *pattern* bergeser ke kanan sebanyak 1 karakter, dari *index* 7 menuju *index* 8.

e. Langkah keempat

<i>Index</i>	0	1	2	3	4	5	6	7	8	9	10	11
Teks	A	B	D	B	A	C	D	B	D	A	B	C
<i>Pattern</i>							A	B	C			

Karakter paling kanan dari *pattern* yaitu "C" tidak cocok dengan karakter yang sejajar pada teks yaitu "D". karakter "D" tidak terdapat pada *pattern* maka *pattern* berpindah sebanyak panjang *pattern*. Panjang *pattern* diketahui yaitu 3 maka *pattern* bergeser ke kanan sebanyak 3 karakter, dari *index* 8 menuju *index* 11.

f. Langkah kelima

<i>Index</i>	0	1	2	3	4	5	6	7	8	9	10	11
Teks	A	B	D	B	A	C	D	B	D	A	B	C
<i>Pattern</i>										A	B	C

Karakter paling kanan dari *pattern* yaitu "C" cocok dengan karakter yang sejajar pada teks yaitu "C". Jika terjadi kecocokan maka langkah selanjutnya mencocokan karakter sebelah kiri dari karakter *pattern* yang sudah cocok. Karakter yang selanjutnya dicocokan yaitu karakter pada *pattern* yaitu "B" terhadap karakter yang sejajar pada teks yaitu "B". Ternyata hasilnya terjadi kecocokan. Jika terjadi kecocokan maka langkah selanjutnya mencocokan karakter sebelah kiri dari karakter *pattern* yang sudah cocok. Karakter yang selanjutnya dicocokan karakter pada *pattern* yaitu "A" terhadap karakter yang sejajar pada teks yaitu "A". Ternyata hasilnya terjadi kecocokan. Semua karakter *pattern* sudah dicocokan terhadap karakter yang sejajar pada teks dan hasilnya semua karakter pada *pattern* cocok. Jadi dapat diambil kesimpulan bahwa *pattern* ditemukan pada teks.

2.2.3 *Email*

Electronic mail atau biasa disingkat sebagai *email*, merupakan sebuah metode untuk mengirimkan pesan dalam bentuk digital, pesan ini biasanya dikirimkan melalui medium internet, sebuah pesan elektronis tediri dari isi, alamat pengirim, dan alamat-alamat yang dituju (Edy Victor Haryanto, 2012: 127).

Sistem *email* beroperasi di atas jaringan berbasis pada model *store and forward*. Sistem ini mengaplikasikan sebuah sistem *server email* yang menerima, meneruskan, mengirimkan, serta menyimpan pesan-pesan pengguna. Pengguna

hanya perlu menghubungkan PC ke dalam jaringan (Edy Victor Haryanto, 2012: 127).

Mengirim sebuah *email* dari alamat *email* yang satu ke alamat *email* yang lain digunakan sebuah *protocol* (aturan), yaitu *Simple Mail Transfer Protocol* (SMTP). SMTP telah menjadi aturan dasar yang disepakati untuk mengirim *email* (Edy Victor Haryanto, 2012: 128).

2.2.4 Internet Message Access Protocol (IMAP)

IMAP merupakan metode akses *email*, yaitu dirancang untuk memungkinkan pengguna untuk menyimpan pesan secara permanen di *server*, IMAP menyelesaikan masalah *backup* administrator sistem dan memungkinkan pengguna untuk mengakses semua pesan dari tempat mana pun di dunia (selain pembatasan *firewall*), IMAP juga memiliki implementasi yang lebih luas dari koneksi yang dijamin TLS, membuat IMAP aman untuk meningkatkan kinerja dan memungkinkan pengguna untuk bekerja dengan *mailbox* mereka sementara tidak terhubung ke *server email*, sebagian besar aplikasi *email* dengan dukungan IMAP *caching mailbox* dan pesan yang diunduh di *hard drive* lokal (Haycox, et al., 2009: 11).

IMAP mendukung banyak *folder* dan menyimpan informasi status pesan (baik pesan belum dibaca, dibalas, atau dihapus) di *server*. Ini berarti bahwa pengguna yang mengakses penyimpanan pesan mereka dari lokasi yang berbeda, dengan kemungkinan *email client* yang berbeda, akan disajikan dengan tampilan pesan mereka yang konsisten dan terbaru. IMAP juga mendukung pencarian sisi

server, sehingga aplikasi *client* tidak perlu mengunduh semua pesan untuk mencari *email* (Haycox, *et al.*, 2009: 11).

2.2.5 Simple Mail Transfer Protocol (SMTP)

Simple Mail Transfer Protocol (SMTP) adalah protokol standar untuk pengiriman *email* di internet, SMTP adalah protokol yang cukup sederhana, berbasis teks dan protokol ini menyebutkan satu atau lebih penerima *email* untuk kemudian diverifikasi, jika penerima *email* valid, *email* akan segera dikirim (Edy Victor Haryanto, 2012: 130).

SMTP adalah protokol teks *line-oriented* yang berjalan melalui TCP, yang membuatnya mudah untuk memecahkan kode transkrip SMTP dan untuk memulai sesi SMTP menggunakan klien telnet biasa yang ditemukan di hampir semua komputer (Haycox, *et al.*, 2009: 11). Namun, karena terbatas pada kemampuannya untuk antrian pesan di sisi penerima, biasanya akan menggunakan salah satu dari dua protokol ini, yaitu POP3 (*Post Office Protocol* versi 3) dan IMAP (*Internet Message Access Protocol*) yang memungkinkan pengguna menyimpan pesan dalam kotak surat (*mailbox*) *server* dan mengunduh secara berkala dari *server* (Edy Victor Haryanto, 2012: 131).

2.2.6 HTML

HTML merupakan singkatan dari *Hypertext Markup Language*. Singkatan ini terdiri dari 3 komponen kata, yakni: *Hypertext*, *Markup* dan *Language*. Kata *Hypertext* dari HTML menekankan pengertian *text* yang lebih dari sekedar teks ('*hyper*'- *text*), maksudnya selain berfungsi sebagai teks biasa, sebuah teks di dalam

HTML juga bisa berfungsi sebagai penghubung ke halaman lain atau dikenal dengan istilah *link* (Andre Pratama, 2015: 1).

Kata kedua dari singkatan HTML adalah *Markup*. *Markup* dapat diterjemahkan sebagai tanda atau penanda (bahasa Inggris: *mark*), di dalam HTML terdapat tanda-tanda khusus. Tanda ini diperlukan untuk mengatur format dan membuat struktur halaman *web* (Andre Pratama, 2015: 1).

Bagian terakhir dari HTML adalah *Language*. Istilah *language* jika diterjemahkan berarti: bahasa, HTML tidak menggunakan ‘*Programming Language*’, tetapi hanya ‘*Language*’ saja, hal ini secara tidak langsung menyatakan bahwa HTML bukanlah sebuah bahasa pemrograman. HTML tidak memiliki struktur dasar seperti variabel, kondisi *IF*, *function*, atau *class* seperti layaknya sebuah bahasa pemrograman komputer (Andre Pratama, 2015: 1).

2.2.7 Hypertext Preprocessor (PHP)

PHP merupakan singkatan dari PHP: *Hypertext Preprocessor*, singkatan ini disebut singkatan rekursif, yakni permainan kata dimana kepanjangannya juga terdiri dari singkatan PHP itu sendiri, yakni PHP: *Hypertext Preprocessor*. *Hypertext Preprocessor* bisa diterjemahkan sebagai ‘pemroses *hypertext*’, atau ‘pemroses HTML’. Dalam pengertian paling sederhana, PHP adalah bahasa pemrograman *web* yang digunakan untuk men-*generate* atau menghasilkan kode HTML. PHP termasuk Bahasa pemrograman *server side*, karena termasuk bahasa pemrograman berbasis *server*, kita harus menjalankan kode-kode PHP dari sebuah *server*, dengan kata lain, kode PHP tidak bisa dijalankan tanpa *server* (Andre Pratama, 2016:4).

2.2.8 *Model-View-Controller (MVC)*

Yudho Yudhanto dan Helmi Adi Prasetyo (2018: 6) *Model-View-Controller* (MVC) adalah sebuah metode untuk membuat sebuah aplikasi dengan memisahkan data (*Model*) dari tampilan (*View*) dan cara bagaimana memprosesnya (*Controller*).

- a. *Model* mewakili struktur data. Biasanya model berisi fungsi-fungsi yang membantu seseorang dalam pengelolaan basis data, seperti memasukkan data ke basis data, pembaruan data, dan lain-lain.
- b. *View* adalah bagian yang mengatur tampilan ke pengguna.
- c. *Controller* merupakan bagian yang menjembatani *model* dan *view*. *Controller* berisi perintah-perintah yang berfungsi untuk memproses suatu data dan mengirimkannya ke halaman *web*.

2.2.9 *Framework Laravel*

Laravel adalah *framework* pengembangan web MVC yang ditulis dalam bahasa pemrograman PHP, Laravel telah dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi baik biaya pengembangan awal dan biaya pemeliharaan berkelanjutan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi, dengan memberikan sintaksis ekspresif yang jelas dan serangkaian fungsionalitas inti yang akan menghemat waktu implementasi. Laravel telah dirancang sebagai *framework* yang minimalis dan fungsionalitas, sehingga lebih mudah untuk memahami basis kode yang lebih kecil dan Laravel mengimplementasikan solusi dengan cara yang bersih, sederhana, dan elegan (Shawn McCool 2012: 3).

Laravel menawarkan modularitas kode, hal ini dicapai melalui kombinasi *driver* dan sistem *bundle*-nya, *driver* memungkinkan pengguna mudah untuk mengubah dan memperluas fungsi *caching*, *session*, *database*, dan *authentication*. Dengan menggunakan *bundle*, pengguna dapat mengemas segala jenis kode untuk digunakan kembali sendiri atau untuk disediakan ke seluruh komunitas Laravel (Shawn McCool 2012: 3).

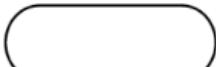
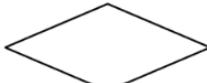
2.2.10 MySQL

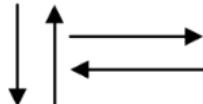
MySQL adalah salah satu aplikasi RDBMS (*Relational Database Management System*) pengertian sederhana RDBMS adalah aplikasi *database* yang menggunakan prinsip relasional, MySQL bersifat gratis dan *open source*, artinya setiap orang boleh menggunakan dan mengembangkan aplikasi ini, namun walaupun gratis, MySQL di-*support* oleh ribuan *programmer* dari seluruh dunia (Andre Pratama, 2017:2).

2.2.11 Flowchart

Lamhot Sitorus (2015: 14) *flowchart* berfungsi untuk menggambarkan urutan logika dari suatu prosedur pemecahan masalah supaya sebuah algoritma yang terstruktur mudah dipahami oleh orang lain (khususnya *programmer* yang bertugas mengimplementasikan program), sehingga *flowchart* merupakan langkah-langkah penyelesaian masalah yang dituliskan dalam simbol-simbol tertentu. Simbol-simbol yang digunakan untuk menggambarkan algoritma dalam bentuk diagram alir dan kegunaan dari simbol-simbol yang bersangkutan dapat dilihat pada tabel 2.2.

Tabel 2.2 Simbol-Simbol Flowchart

No	Simbol	Nama	Fungsi
1		<i>Terminal</i>	Menyatakan permulaan atau akhir suatu program
2		<i>Input / Output</i>	Menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya
3		<i>Process</i>	Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer
4		<i>Decision</i>	Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban: ya / tidak
5		<i>Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
6		<i>Offline Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda
7		<i>Predefined Process</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
8		<i>Punched Card</i>	Menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu

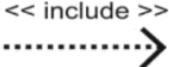
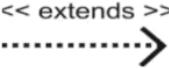
No	Simbol	Nama	Fungsi
9		Document	Mencetak keluaran dalam bentuk dokumen (melalui printer)
10		Flow	Menyatakan jalannya arus suatu proses

2.2.12 Use Case

Uus Rusmawan (2019: 72) *use case* adalah teknik untuk merekam persyaratan fungsional sebuah sistem, *use case* mendeskripsikan interaksi tipikal antara para pengguna sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sistem digunakan. Dua hal penting dalam *use case* adalah skenario dan aktor. Skenario adalah rangkaian langkah-langkah yang menjelaskan sebuah interaksi antara seorang pengguna dengan sebuah sistem. secara umum simbol-simbol *use case* dapat dilihat pada tabel 2.3.

Tabel 2.3 Simbol – Simbol *Use Case*

Simbol	Nama	Keterangan
	Actor	Seseorang atau apa saja yang berhubungan dengan sistem yang sedang dibangun.
	Use case	Menggambarkan bagaimana seseorang menggunakan sistem
	Relasi asosiasi	Relasi yang dipakai untuk menunjukkan hubungan antara <i>actor</i> dan <i>use case</i>

Simbol	Nama	Keterangan
	Relasi <i>include</i>	Memungkinkan fungsionalitas yang disediakan oleh <i>use case</i> lainnya
	Relasi <i>extend</i>	Memungkinkan suatu <i>use case</i> secara <i>optional</i> menggunakan fungsionalitas yang disediakan oleh <i>use case</i> lainnya

2.2.13 ISO 9126

Pressman (2014: 418) standar ISO 9126 dikembangkan dalam usaha identifikasi atribut-atribut kualitas perangkat lunak komputer. Standar ISO 9126 mengidentifikasi 6 atribut-atribut kualitas perangkat lunak, yang dijabarkan sebagai berikut:

- Fungsionalitas. Derajat tentang bagaimana perangkat lunak memenuhi kebutuhan yang telah ditetapkan sebelumnya dan memiliki sub-sub atribut berikut ini: kecocokan, akurasi, interoperabilitas, kesesuaian dan keamanan.
- Keandalan. Jumlah waktu penggunaan perangkat lunak yang tersedia dan memiliki sub atribut-sub atribut: kematangan, toleransi kesalahan, kemampuan untuk melakukan pemulihan.
- Kemudahan penggunaan. Derajat tentang bagaimana kemudahan perangkat lunak digunakan, dimana hal ini seringkali diindikasikan menggunakan sub atribut-sub atribut berikut ini: kemudahan untuk dipahami, kemudahan untuk dipelajari, operabilitas.

- d. Efisiensi. Derajat penggunaan sumber daya sistem secara optimal, dimana hal ini diindikasikan oleh sub atribut-subs atribut berikut ini: perilaku waktu, perilaku sumber daya.
- e. Kemudahan pemeliharaan. Kemudahan yang menentukan tentang bagaimana perbaikan-perbaikan mungkin dilakukan pada suatu perangkat lunak, dimana hal ini diindikasikan menggunakan sub atribut-subs atribut berikut ini: kemampuan untuk dilakukan analisis, kemampuan untuk dilakukan perubahan, hal-hal yang berkaitan dengan stabilitas, serta kemampuan untuk dilakukan pengujian
- f. Portabilitas. Kemudahan bagaimana perangkat lunak dapat dipindahkan dari suatu lingkungan operasional ke lingkungan operasional yang lainnya, yang hal ini diidentifikasi menggunakan sub atribut-subs atribut: kemampuan untuk beradaptasi, kemampuan untuk dipasangkan, kesesuaian, kemampuan untuk digantikan.

BAB III

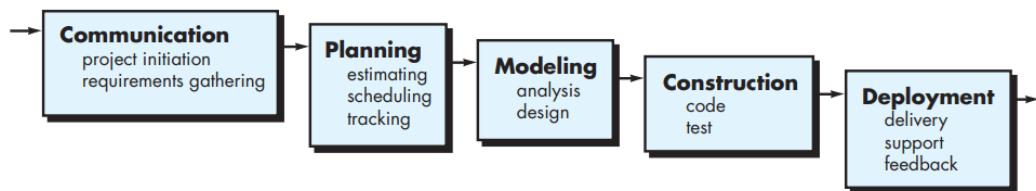
METODE PENELITIAN

3.1 Waktu dan Tempat Pelaksanaan

Penelitian dilaksanakan di Jurusan Teknik Elektro Fakultas Teknik Universitas Negeri Semarang. Ada pun pelaksanaan dimulai pada bulan Februari - November 2019.

3.2 Desain Penelitian

Metode penelitian yang digunakan dalam penelitian ini adalah metode *classic life cycle* yang biasa disebut dengan metode *waterfall*. Metode *classic life cycle* merupakan sebuah pendekatan sistematis dan berurutan untuk pengembangan perangkat lunak yang dimulai dengan spesifikasi kebutuhan pelanggan dan berkembang melalui *communication*, *planning*, *modeling*, *construction*, dan *deployment*, serta berkelanjutan dalam mendukung perangkat lunak yang sudah selesai (Roger S. Pressman & Bruce R. Maxim, 2014: 41). Tahapan dalam model *classic life cycle* ditunjukan pada gambar 3.1.



Sumber: (Roger S. Pressman & Bruce R. Maxim, 2014: 42)

Gambar 3.1. Model *Classic Life Cycle*

3.2.1 *Communication*

Pada tahap *communication* dilakukan observasi permasalahan yang dialami pengguna dari berbagai macam *provider email*. Menganalisis penyebab masalah yang terjadi dan dampak pada pengguna pada berbagai macam *provider email*. Diperoleh hasil observasi bahwa *provider email* dapat salah memasukan data pada *blacklist* di sistem penyaringan *email* yang dimilikinya seperti Outlook terhadap Zoho pada tahun 2019. Outlook memblokir rentang IP yang dimiliki Zoho yaitu 135.84.80.192/26 dan 136.143.188.0/2. Sistem penyaringan *email* dapat terjadi kesalahan dalam menandai *email* sebagai *spam* yaitu jika *email* yang dikirim dari alamat *email* yang memiliki *host* menggunakan gTLD yang belum memiliki reputasi atau memiliki reputasi yang buruk.

3.2.2 *Planning*

Pada tahap ini digambarkan estimasi waktu dan penjadwalan pembuatan sistem penyaringan *email*, hingga tahap akhir yaitu *deployment*.

Tabel 3.1 Jadwal Estimasi Waktu

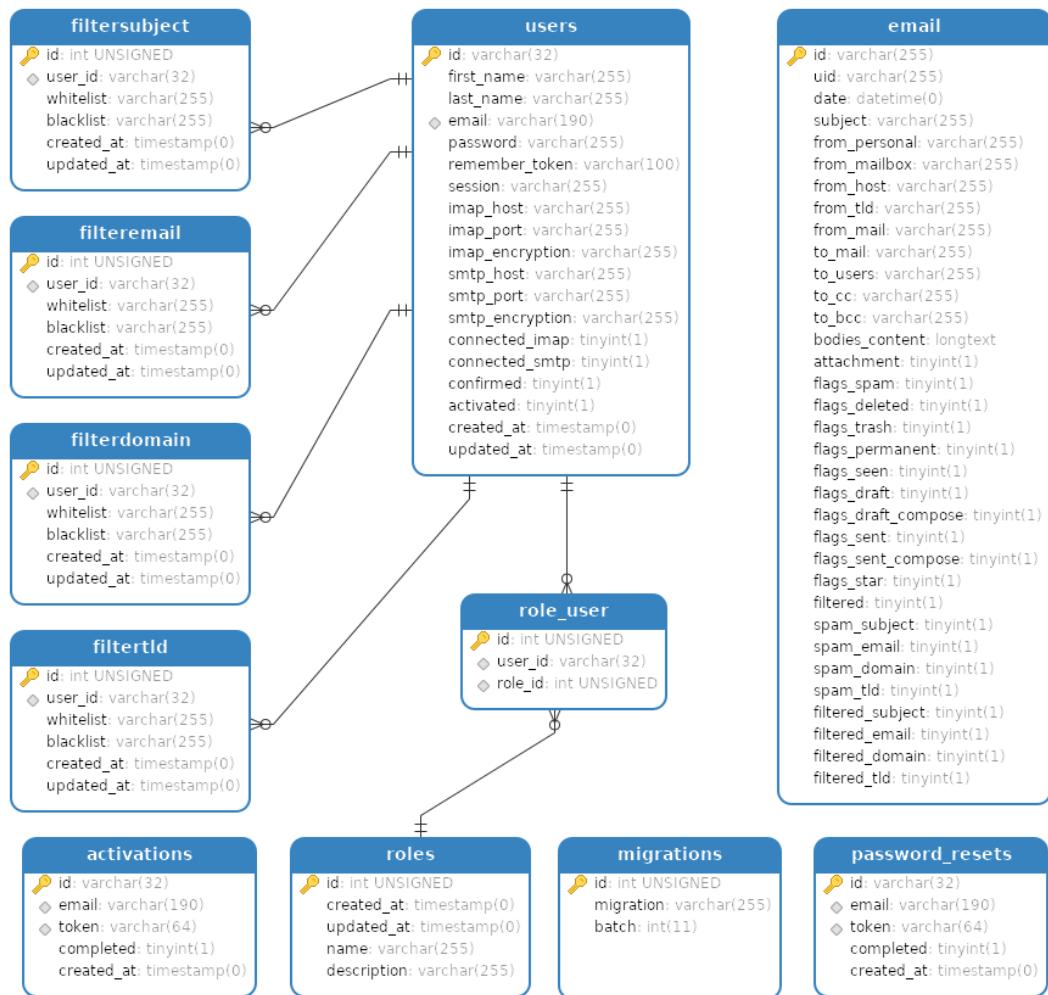
No	Nama Kegiatan	Bulan									
		Feb	Mar	Apr	Mei	Jun	Jul	Ags	Sep	Okt	Nov
1	Observasi <i>Provider Email</i>										
2	Tahap Pemodelan										
3	Tahap Kontruksi (Pengkodean)										
4	Pengujian dan <i>deployment</i>										

3.2.3 *Modelling*

Tahap ini adalah tahap perancangan dan pemodelan arsitektur sistem yang berfokus pada perancangan *database*, arsitektur *software*, tampilan *interface* dan algoritma program. tujuannya untuk lebih memahami gambaran besar dari apa yang akan dikerjakan.

3.2.3.1 Desain *Database*

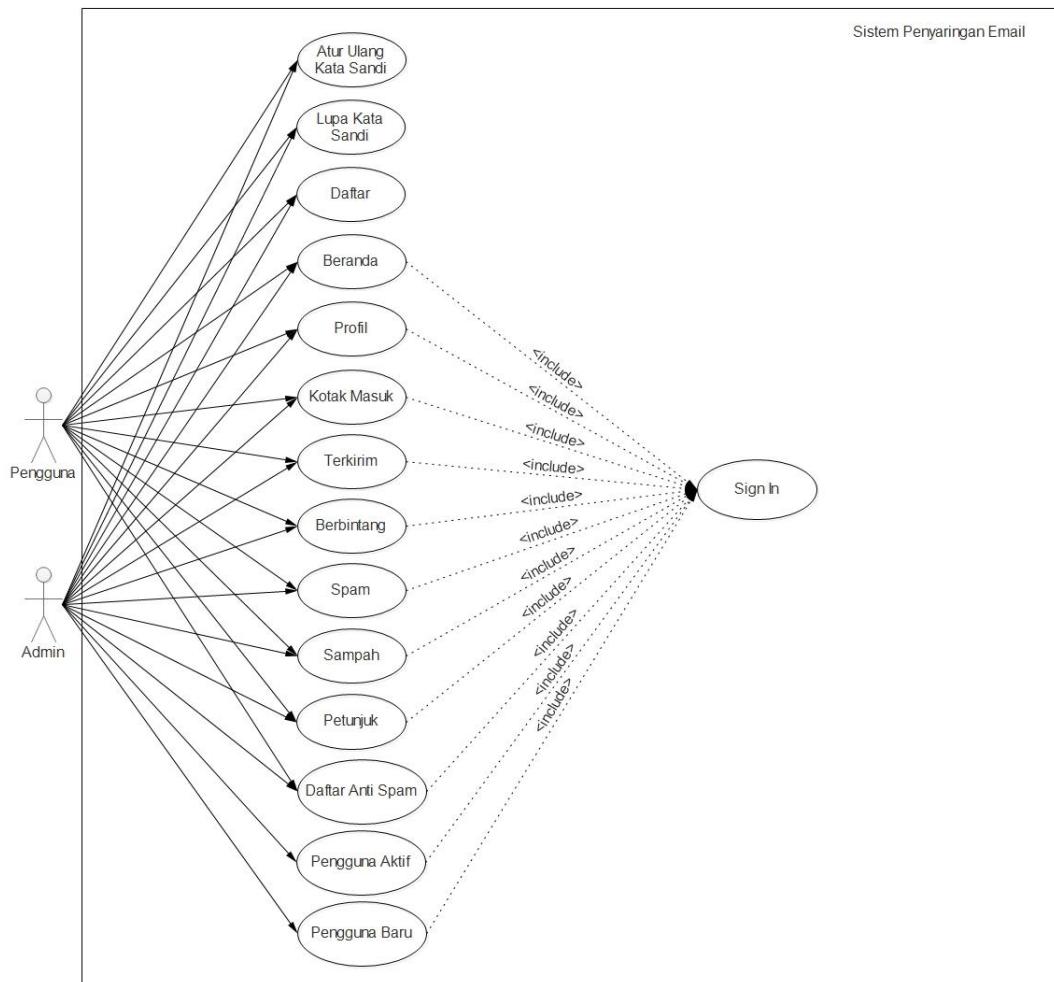
Desain *database* dilakukan dengan cara mengidentifikasi *file-file* yang diperlukan oleh sistem dengan melihat desain model yang telah digambarkan dalam bentuk *Entity Relationship Diagram* (ERD) yang ditentukan melalui teknik normalisasi. Selanjutnya menentukan parameter dari *file database* seperti: tipe *file*, organisasi *file*, akses *file* dan *field kunci* dari *file*. Pada sistem penyaringan *email* memiliki tabel *email* yang strukturnya berdasarkan dari data *email* yang diperoleh dari IMAP. Desain *database* sistem penyaringan *email* dapat dilihat pada gambar 3.2.



Gambar 3.2. Desain Database Sistem Penyaringan Email

3.2.3.2 Use Case Diagram

Use Case Diagram sistem penyaringan *email* yang akan dibuat pada penelitian ini dapat dilihat pada gambar 3.3.



Gambar 3.3. *Use Case Diagram Sistem Penyaringan Email*

Berdasarkan gambar 3.3 *use case* diagram sistem penyaringan *email* dapat diketahui interaksi antara satu atau lebih *actor* dengan sistem yang akan dibuat. Interaksi tersebut dijelaskan pada tabel 3.2.

Tabel 3.2 Definisi *Use Case* Sistem Penyaringan *Email*

No	Keterangan	Interaksi
1	<i>Actor</i> <i>Summary</i> <i>Bussiness Rule</i>	Pengguna Pengguna dapat mengelola <i>email</i> , profil, dan daftar anti <i>spam</i>
		<ol style="list-style-type: none"> 1. Pengguna dapat mengakses halaman masuk 2. Pengguna dapat mengakses halaman daftar 3. Pengguna dapat mengakses halaman lupa kata sandi 4. Pengguna dapat mengakses halaman atur ulang kata sandi 5. Pengguna dapat mengakses beranda 6. Pengguna dapat mengirim <i>email</i> 7. Pengguna dapat melihat isi <i>email</i> 8. Pengguna dapat mencari <i>email</i> 9. Pengguna dapat mengelola data profil 10. Pengguna dapat mengelola daftar anti <i>spam</i>
		Admin
		Pengguna dapat mengelola <i>email</i> , profil, daftar anti <i>spam</i> , dan hak akses pengguna
		<ol style="list-style-type: none"> 1. Pengguna dapat mengakses halaman masuk 2. Pengguna dapat mengakses halaman daftar 3. Pengguna dapat mengakses halaman lupa kata sandi 4. Pengguna dapat mengakses halaman atur ulang kata sandi 5. Pengguna dapat mengakses beranda 6. Pengguna dapat mengirim <i>email</i> 7. Pengguna dapat melihat isi <i>email</i> 8. Pengguna dapat mencari <i>email</i> 9. Pengguna dapat mengelola data profil 10. Pengguna dapat mengelola daftar anti <i>spam</i> 11. Admin dapat mengelola pengguna yang aktif 12. Admin dapat mengelola pengguna baru

3.2.3.3 Desain Interface (*Mockup*)

Mockup dirancang guna memberikan gambaran bagi pengguna mengenai rancangan sistem yang akan dibuat berupa tampilan *layout*. Perancangan *interface* ini dirancang sesuai dengan pembagian hak akses pengguna, yaitu Pengguna dan Admin.

The mockup shows a login form with the title 'Masuk' at the top. It contains fields for 'Alamat Email' and 'Kata Sandi', both with placeholder text. Below these is a 'reCAPTCHA' checkbox. At the bottom are 'Masuk' and 'Lupa kata sandi?' buttons, and a 'Daftar' link at the very bottom.

Gambar 3.4. Halaman Masuk

Pada gambar 3.4 halaman masuk terdapat *input* tipenya adalah *email* dan *password*, ada fitur *reCAPTCHA* untuk memverifikasi bagi pengakses, terdapat satu *button* masuk dan terdapat dua *hyperlink*, *hyperlink* yang pertama mengarahkan ke halaman lupa kata sandi dan *hyperlink* kedua mengarahkan ke halaman daftar. Halaman masuk akan ditampilkan pada tengah halaman. Setiap *input* memiliki validasi *client side* untuk memvalidasi *input* sesuai tipenya.

Lupa Kata Sandi ?

Masukan alamat email anda untuk mengatur ulang kata sandi!

Email

Kembali

Kirimkan

Gambar 3.5. Halaman Lupa Kata Sandi

Pada gambar 3.5 halaman lupa kata sandi terdapat *input* tipenya adalah *email* dan terdapat dua *button* yaitu kembali dan kirimkan. *Button* kembali mengarahkan pengguna ke halaman masuk sedangkan *button* kirimkan akan mengirim *email* lupa kata sandi ke alamat *email* yang dituju.

Daftar

Masukan detail pribadi anda di bawah ini:

Nama Depan

Nama Depan

Nama Depan

Nama Belakang

Alamat Email

Alamat Email

Kata Sandi

Kata Sandi

Ketik Ulang Kata Sandi

Ketik Ulang Kata Sandi

reCAPTCHA

Kembali

Kirimkan

Gambar 3.6. Halaman Daftar

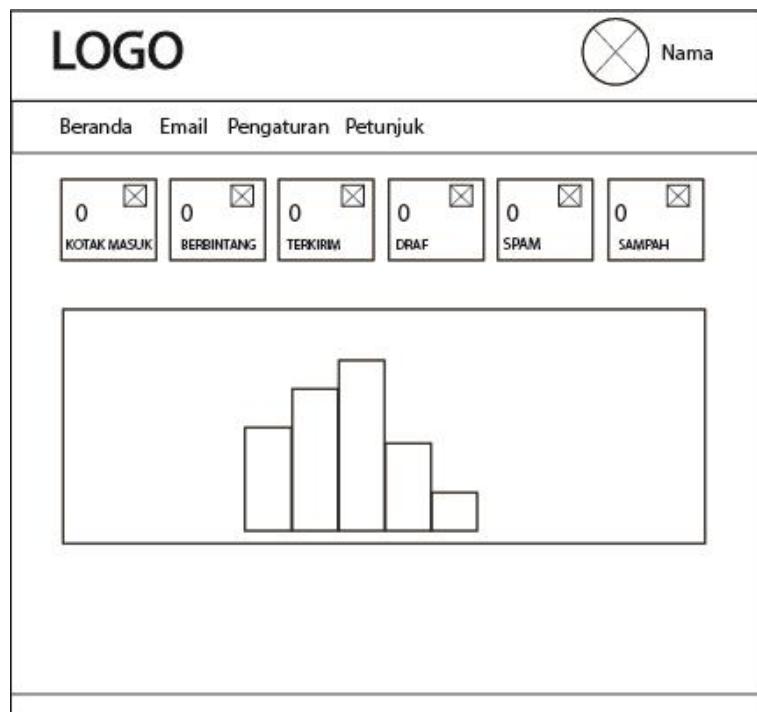
Pada gambar 3.6 halaman daftar terdapat beberapa *input* yaitu *input* nama depan tipenya adalah *text*, *input* nama belakang tipenya adalah *text*, *input* alamat *email* tipenya adalah *email*, *input* kata sandi tipenya adalah *password*, *input* ketik ulang kata sandi tipenya adalah *password*. Terdapat fitur *reCAPTCHA* untuk memverifikasi bagi pengakses. Terdapat dua *button* yaitu kembali dan kirimkan. *Button* kembali mengarahkan pengguna ke halaman masuk sedangkan *button* kirimkan akan mengirim *email* aktivasi akun ke alamat *email* yang dituju. Setiap *input* memiliki validasi *client side* untuk memvalidasi *input* sesuai tipenya.



Gambar 3.7 Halaman Atur Ulang Kata Sandi

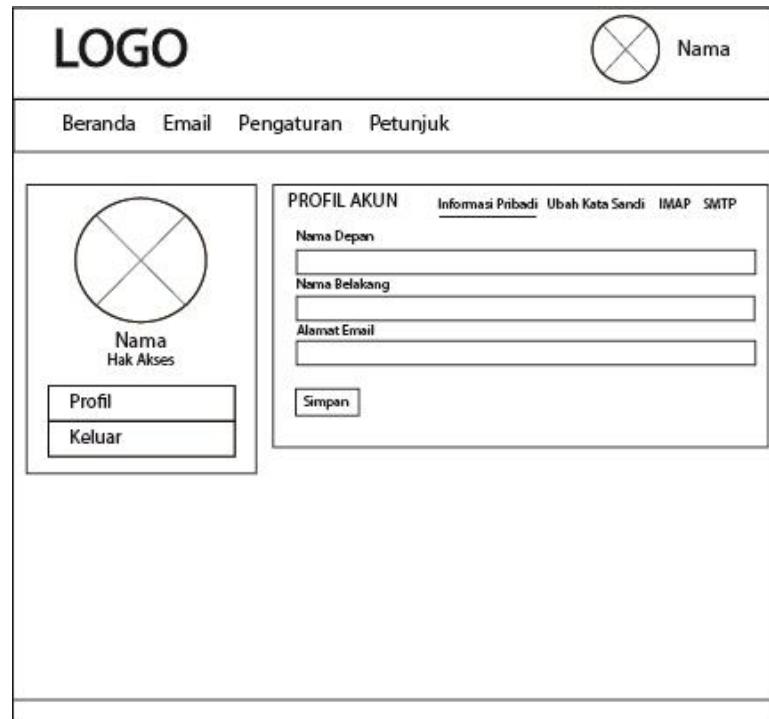
Pada gambar 3.7 halaman atur ulang kata sandi terdapat dua *input* yaitu *input* kata sandi baru tipenya adalah *password* dan *input* ketik ulang kata sandi baru tipenya adalah *password* dan juga terdapat *button* perbaharui yang berfungsi untuk

mengatur ulang kata sandi. Setiap *input* memiliki validasi *client side* untuk memvalidasi *input* sesuai tipenya.



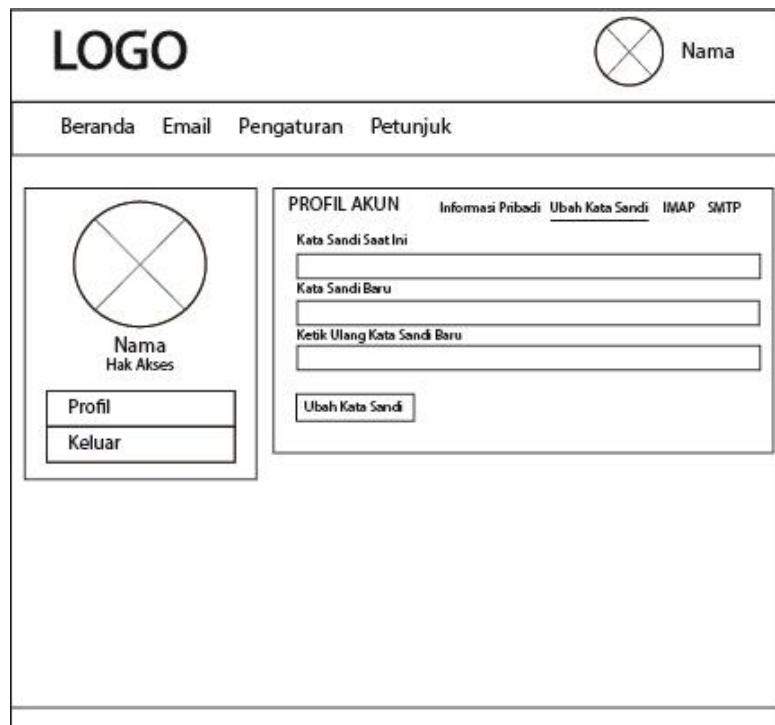
Gambar 3.8. Halaman Beranda

Pada gambar 3.8 bagian paling atas di halaman beranda di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman beranda yang berisi grafik hasil penyaringan *email* setelah sinkronisasi data melalui IMAP.



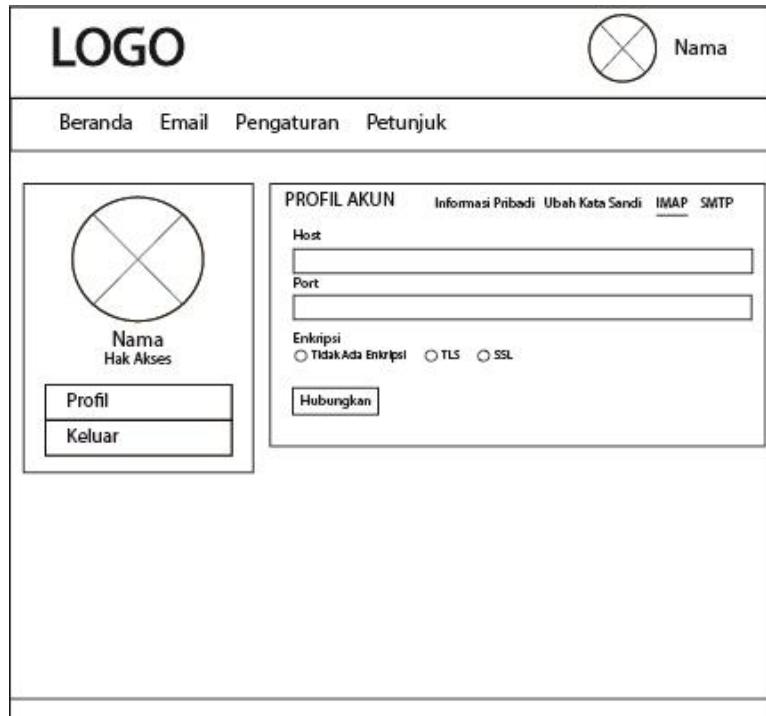
Gambar 3.9. Halaman Informasi Pribadi

Pada gambar 3.9 bagian paling atas di halaman informasi pribadi di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman informasi pribadi yang di bagian kiri terdapat foto profil, nama, hak akses dari pengguna dan *menu* profil dan keluar. Bagian kanan terdapat *form* informasi pribadi yang berisi beberapa *input* yaitu *input* nama depan tipenya adalah *text*, *input* nama belakang tipenya adalah *text* dan *input* alamat *email* tipenya adalah *email*. Setiap *input* memiliki validasi *client side* untuk memvalidasi *input* sesuai tipenya. Di dalam *form* terdapat *button* tipenya adalah *submit* untuk menyimpan data.



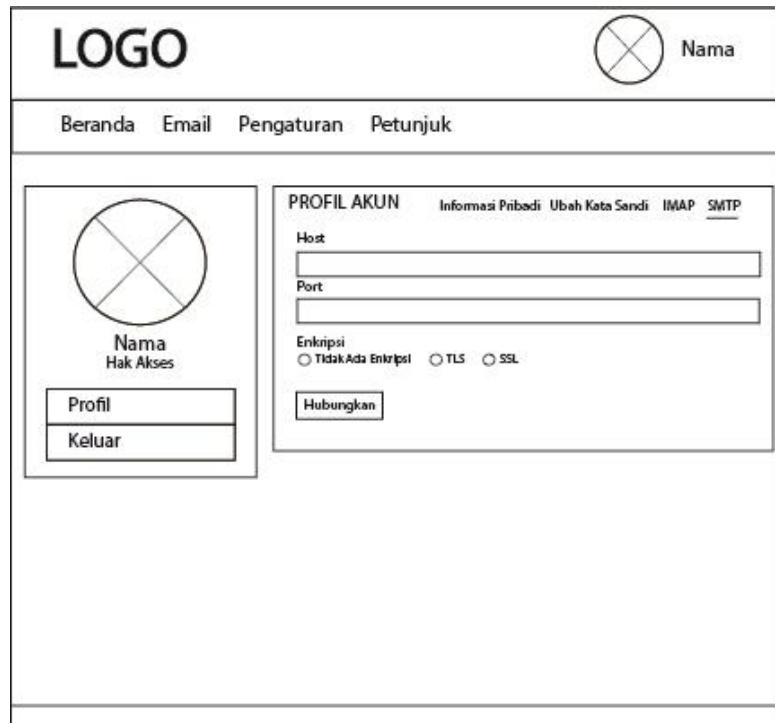
Gambar 3.10. Halaman Ubah Kata Sandi

Pada gambar 3.10 bagian paling atas di halaman ubah kata sandi di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman ubah kata sandi yang di bagian kiri terdapat foto profil, nama, hak akses dari pengguna dan *menu* profil dan keluar. Bagian kanan terdapat *form* ubah kata sandi yang berisi beberapa *input* yaitu *input* kata sandi saat ini tipenya adalah *password*, *input* kata sandi baru tipenya adalah *password* dan *input* ketika ulang kata sandi baru tipenya adalah *password*. Setiap *input* memiliki validasi *client side* untuk memvalidasi *input* sesuai tipenya. Di dalam *form* terdapat *button* tipenya adalah *submit* untuk menyimpan data.



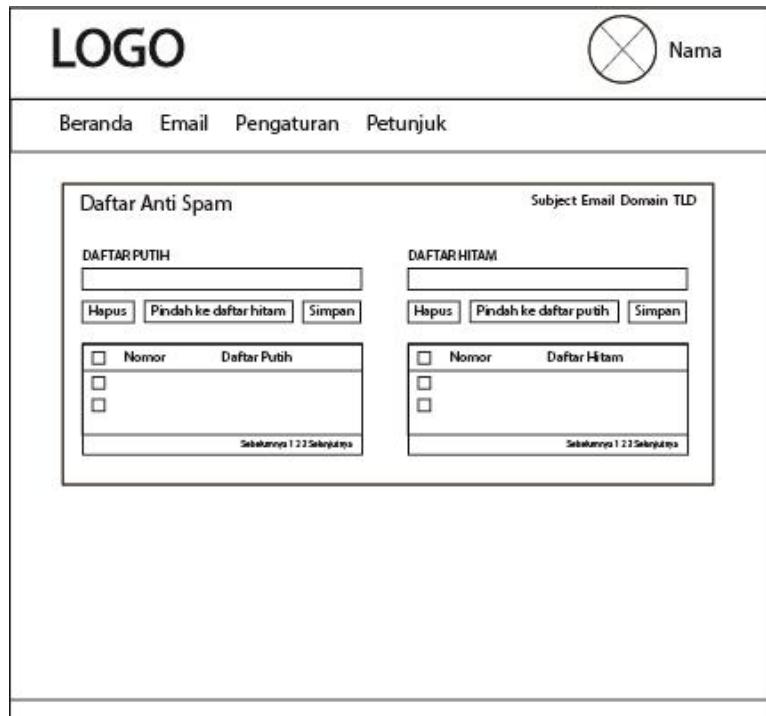
Gambar 3.11. Halaman IMAP

Pada gambar 3.11 bagian paling atas di halaman IMAP di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman IMAP yang di bagian kiri terdapat foto profil, nama, hak akses dari pengguna dan *menu* profil dan keluar. Bagian kanan terdapat *form* IMAP yang berisi beberapa *input* yaitu *host* tipenya adalah *text*, *input port* tipenya adalah *text* dan *input* enkripsi tipenya adalah *radio button* yang memiliki beberapa opsi yaitu tidak ada enkripsi, TLS, dan SSL. Setiap *input* memiliki validasi *client side* untuk memvalidasi *input* sesuai tipenya. Di dalam *form* terdapat *button* hubungkan yang berfungsi untuk menghubungkan IMAP ke *server* sesuai data yang ada pada *form*.



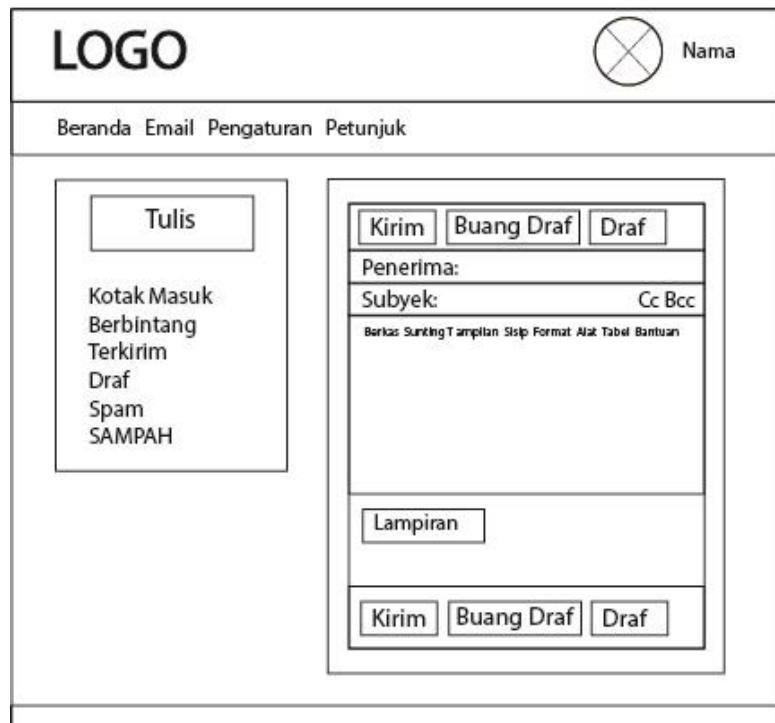
Gambar 3.12. Halaman SMTP

Pada gambar 3.12 bagian paling atas di halaman SMTP di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman SMTP yang di bagian kiri terdapat foto profil, nama, hak akses dari pengguna dan *menu* profil dan keluar. Bagian kanan terdapat *form* SMTP yang berisi beberapa *input* yaitu *host* tipenya adalah *text*, *input port* tipenya adalah *text* dan *input* enkripsi tipenya adalah *radio button* yang memiliki beberapa opsi yaitu tidak ada enkripsi, TLS, dan SSL. Setiap *input* memiliki validasi *client side* untuk memvalidasi *input* sesuai tipenya. Di dalam *form* terdapat *button* hubungkan yang berfungsi untuk menghubungkan SMTP ke *server* sesuai data yang ada pada *form*.



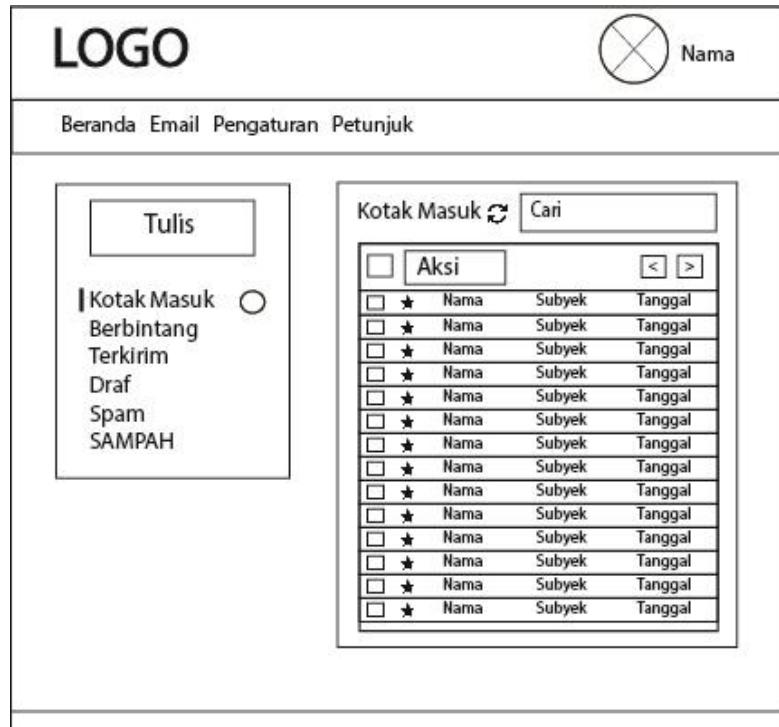
Gambar 3.13. Halaman Daftar Anti *Spam*

Pada gambar 3.13 bagian paling atas di halaman daftar anti *spam* di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman daftar anti *spam* yaitu daftar putih dan daftar hitam. *Form* daftar putih terdiri dari *input* daftar putih tipenya adalah *text*. Tiga *button* yaitu *button* hapus, *button* pindah daftar hitam, dan *button* simpan. Bagian bawah terdapat DataTables *server side* untuk menampilkan data daftar putih. *Form* daftar hitam terdiri dari *input* daftar hitam tipenya adalah *text*. Tiga *button* yaitu *button* hapus, *button* pindah daftar putih, dan *button* simpan. Bagian bawah terdapat DataTables *server side* untuk menampilkan data daftar hitam.



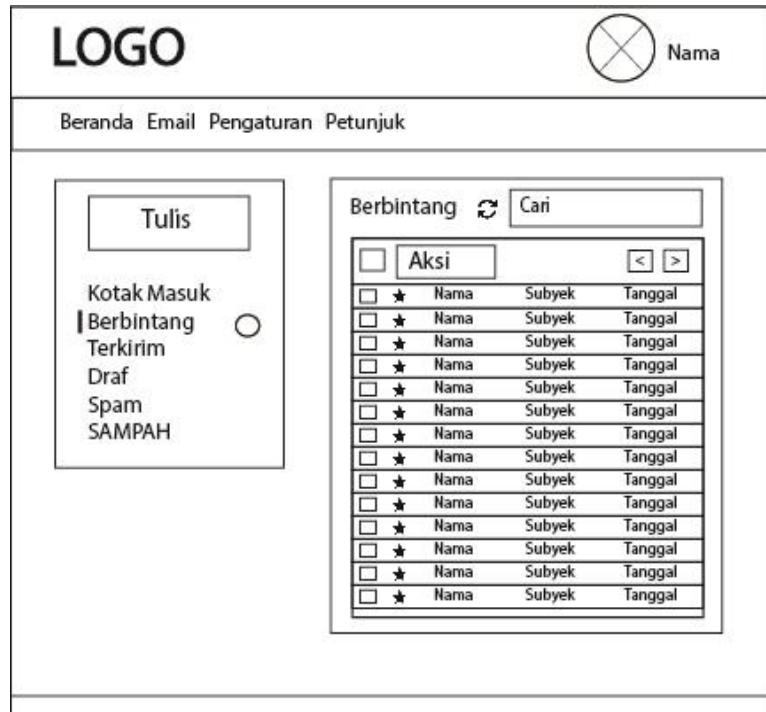
Gambar 3.14. Halaman Tulis

Pada gambar 3.14 bagian paling atas di halaman tulis di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah navbar terdapat fitur inti dari halaman tulis yang di bagian kiri terdapat *sidebar* yang terdiri beberapa *menu* yaitu tulis, kotak masuk, berbintang, terkirim, draf, *spam*, dan sampah. Bagian sebelah kanan terdapat *form* tulis yang berisi *input* penerima tipenya *text*, *input* subyek tipenya *text*, *input* cc tipenya *text*, *input* bcc tipenya *text*, *textarea* untuk isi *email*, *input* lampiran tipenya *file*, *button* kirim tipenya *submit*, *button* buang draf, dan *button* draf tipenya *submit*. Setiap *input* memiliki validasi *client side* untuk memvalidasi *input* sesuai tipenya. Besar *file* lampiran maksimal 25 MB sesuai standar *Multipurpose Internet Mail Extensions* (MIME) pada RFC2045.



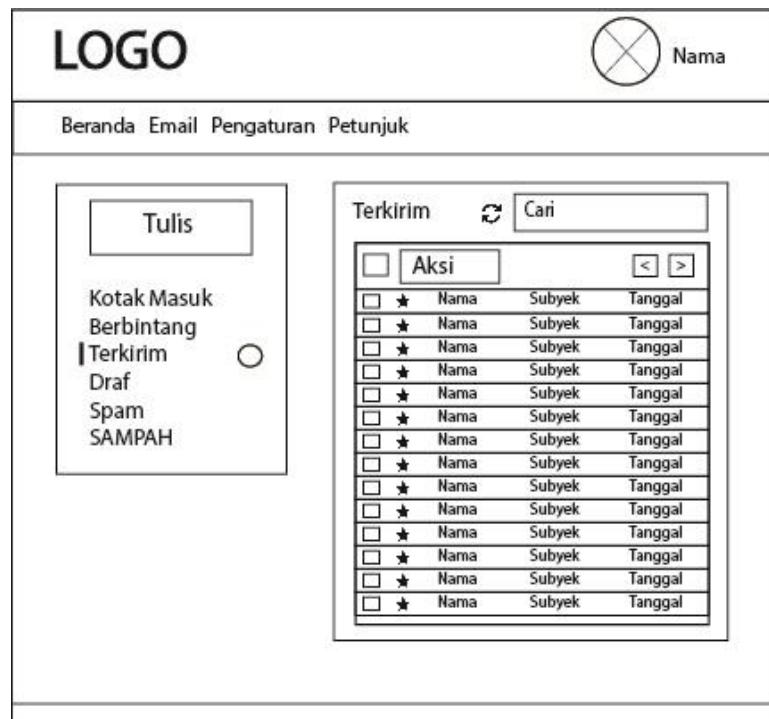
Gambar 3.15. Halaman Kotak Masuk

Pada gambar 3.15 bagian paling atas di halaman kotak masuk di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman tulis yang di bagian kiri terdapat *sidebar* yang terdiri beberapa *menu* yaitu tulis, kotak masuk, berbintang, terkirim, draf, *spam*, dan sampah. Bagian sebelah kanan terdapat tabel yang menampilkan data kotak masuk. Data yang ditampilkan hanya berupa nama, subyek dan tanggal. Pengguna dapat mencari *email* berdasarkan nama atau subyek pada *input* cari. Pengguna dapat memberikan tanda bintang, memberikan tanda sudah dibaca, memberikan tanda belum dibaca, atau menghapus *email* dengan cara memilih *email* melalui *checkbox* di kolom paling kiri pada tabel lalu memilih *menu* pada *dropdown* aksi.



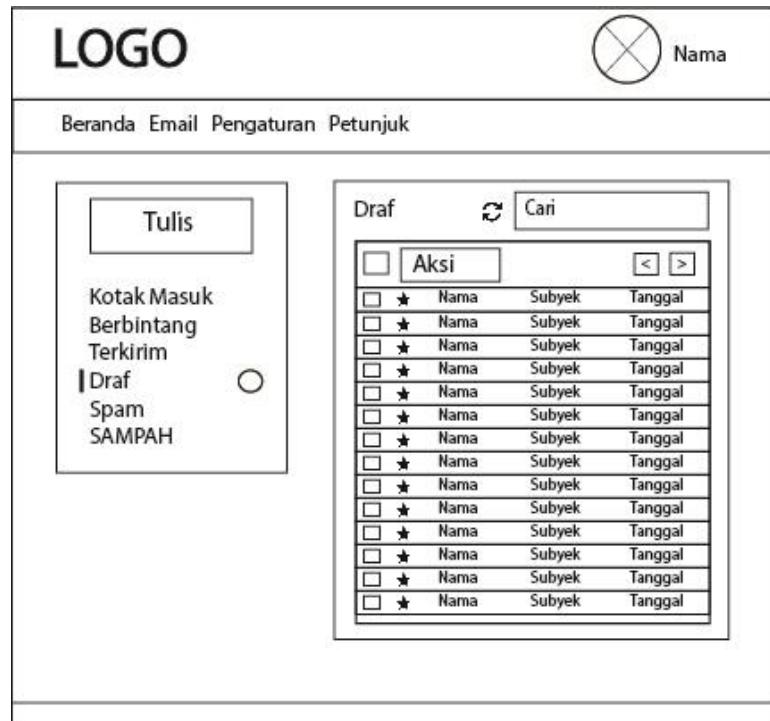
Gambar 3.16. Halaman Berbintang

Pada gambar 3.16 bagian paling atas di halaman berbintang di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman tulis yang di bagian kiri tedapat *sidebar* yang terdiri beberapa *menu* yaitu tulis, kotak masuk, berbintang, terkirim, draf, *spam*, dan sampah. Bagian sebelah kanan terdapat tabel yang menampilkan data *email* berbintang. Data yang ditampilkan hanya berupa nama, subyek dan tanggal. Pengguna dapat mencari *email* berdasarkan nama atau subyek pada *input* cari. Pengguna dapat memberikan tanda tidak berbintang, memberikan tanda sudah dibaca, memberikan tanda belum dibaca, atau menghapus *email* dengan cara memilih *email* melalui *checkbox* di kolom paling kiri pada tabel lalu memilih *menu* pada *dropdown* aksi.



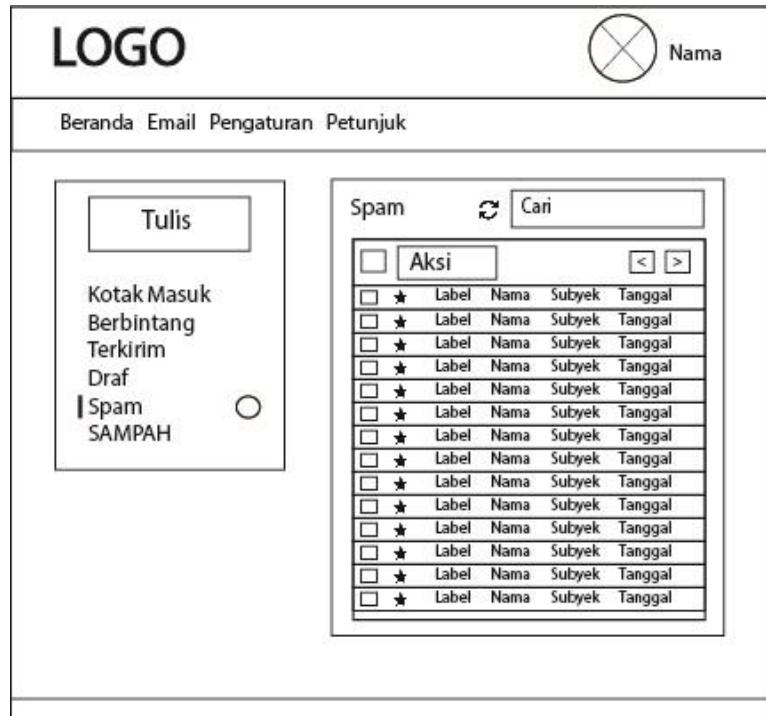
Gambar 3.17. Halaman Terkirim

Pada gambar 3.17 bagian paling atas di halaman terkirim di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman tulis yang di bagian kiri tedapat *sidebar* yang terdiri beberapa *menu* yaitu tulis, kotak masuk, berbintang, terkirim, draf, *spam*, dan sampah. Bagian sebelah kanan terdapat tabel yang menampilkan data *email* terkirim. Data yang ditampilkan hanya berupa nama, subyek dan tanggal. Pengguna dapat mencari *email* berdasarkan nama atau subyek pada *input* cari. Pengguna dapat memberikan tanda bintang, memberikan tanda sudah dibaca, memberikan tanda belum dibaca, atau menghapus *email* dengan cara memilih *email* melalui *checkbox* di kolom paling kiri pada tabel lalu memilih *menu* pada *dropdown* aksi.



Gambar 3.18. Halaman Draf

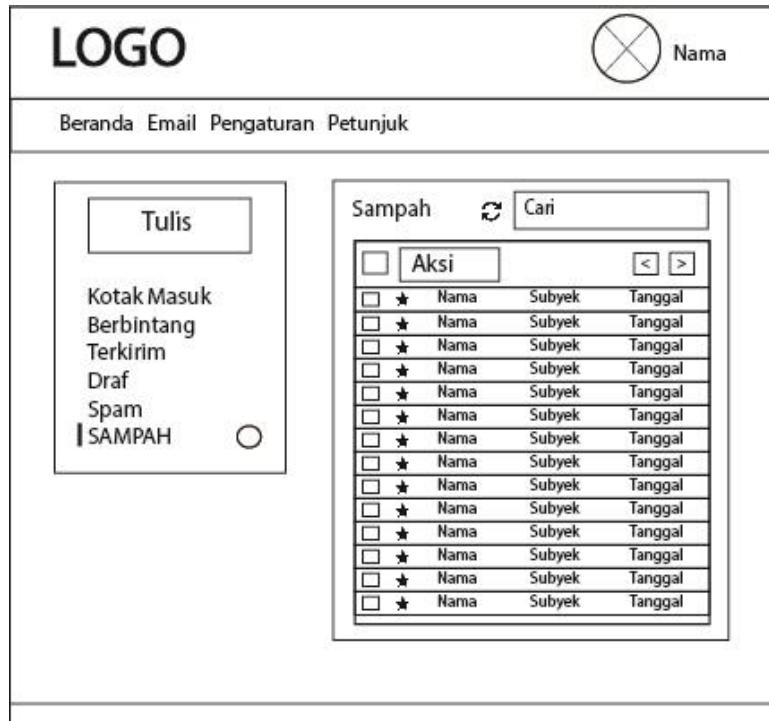
Pada gambar 3.18 bagian paling atas di halaman draf di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman tulis yang di bagian kiri tedapat *sidebar* yang terdiri beberapa *menu* yaitu tulis, kotak masuk, berbintang, terkirim, draf, *spam*, dan sampah. Bagian sebelah kanan terdapat tabel yang menampilkan data draf. Data yang ditampilkan hanya berupa nama, subyek dan tanggal. Pengguna dapat mencari *email* berdasarkan nama atau subyek pada *input* cari. Pengguna dapat memberikan tanda bintang, memberikan tanda sudah dibaca, memberikan tanda belum dibaca, atau menghapus *email* dengan cara memilih *email* melalui *checkbox* di kolom paling kiri pada tabel lalu memilih *menu* pada *dropdown* aksi.



Gambar 3.19. Halaman *Spam*

Pada gambar 3.19 bagian paling atas di halaman *spam* di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu beranda*, *email pengaturan*, dan *petunjuk*. Bagian bawah *navbar* terdapat fitur inti dari halaman tulis yang di bagian kiri tedapat sidebar yang terdiri beberapa *menu* yaitu tulis, kotak masuk, berbintang, terkirim, draf, *spam*, dan sampah. Bagian sebelah kanan terdapat tabel yang menampilkan data *email spam*. Data yang ditampilkan hanya berupa label, nama, subyek dan tanggal. Pengguna dapat mencari *email* berdasarkan nama atau subyek pada *input* cari. Pengguna dapat memberikan tanda bintang, memberikan tanda sudah dibaca, memberikan tanda belum dibaca, atau menghapus

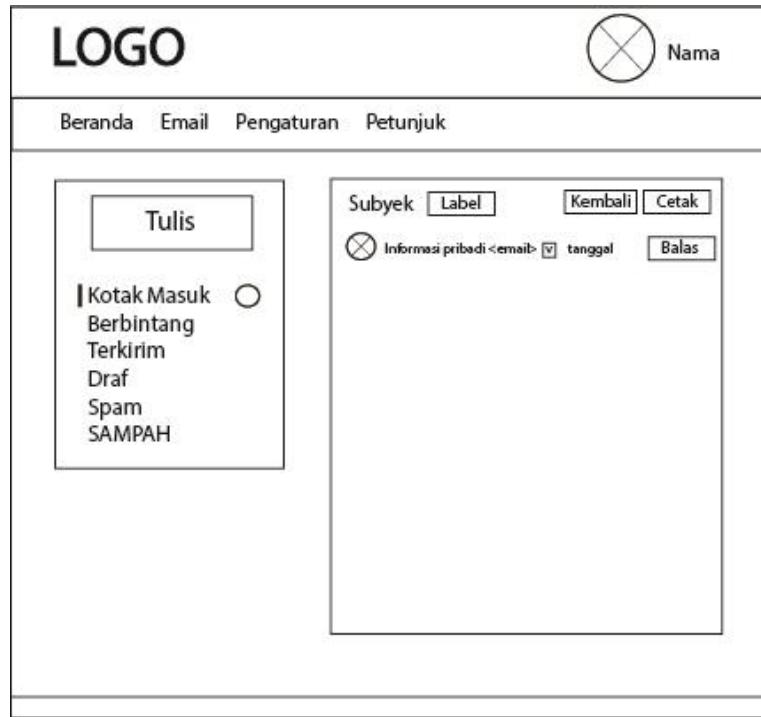
email dengan cara memilih *email* melalui *checkbox* di kolom paling kiri pada tabel lalu memilih *menu* pada *dropdown* aksi.



Gambar 3.20. Halaman Sampah

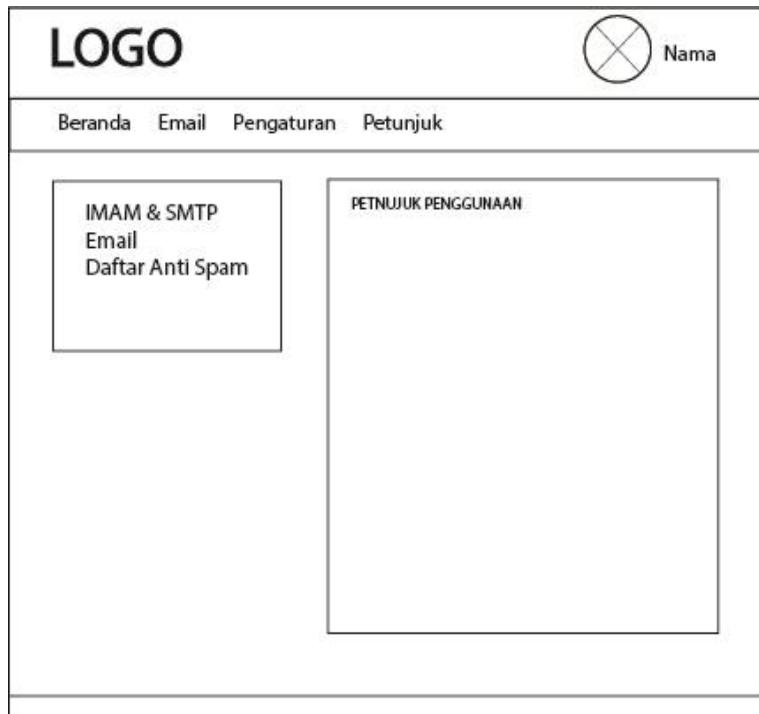
Pada gambar 3.20 bagian paling atas di halaman sampah di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman tulis yang di bagian kiri tedapat *sidebar* yang terdiri beberapa *menu* yaitu tulis, kotak masuk, berbintang, terkirim, draf, *spam*, dan sampah. Bagian sebelah kanan terdapat tabel yang menampilkan data *email* sampah. Data yang ditampilkan hanya berupa label, nama, subyek dan tanggal. Pengguna dapat mencari *email* berdasarkan nama atau subyek pada *input* cari. Pengguna dapat memberikan tanda bintang, memberikan tanda sudah dibaca, memberikan tanda belum dibaca, atau menghapus

email selamanya dengan cara memilih *email* melalui *checkbox* di kolom paling kiri pada tabel lalu memilih *menu* pada *dropdown* aksi.



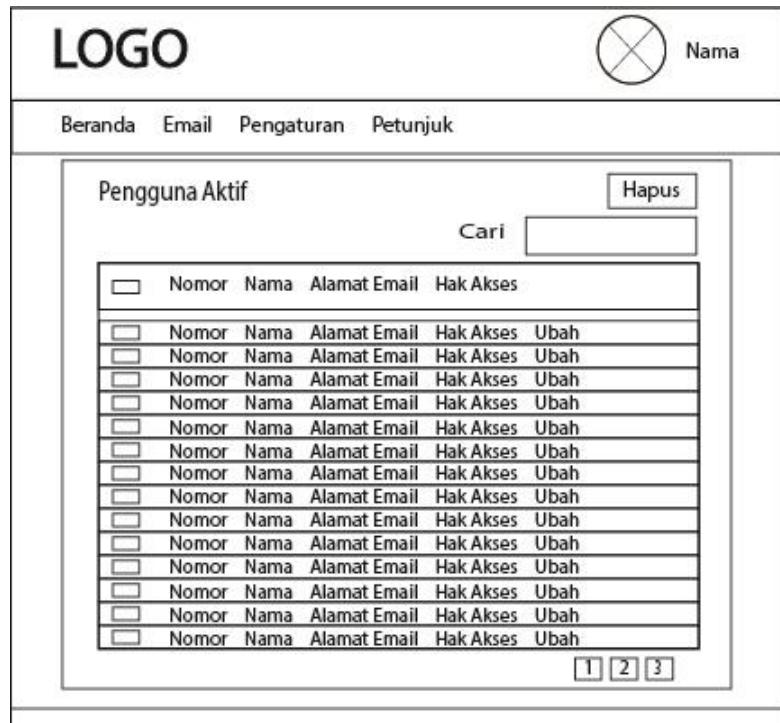
Gambar 3.21. Halaman Lihat *Email*

Pada gambar 3.21 bagian paling atas di halaman lihat *email* di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman tulis yang di bagian kiri terdapat *sidebar* yang terdiri beberapa *menu* yaitu tulis, kotak masuk, berbintang, terkirim, draf, *spam*, dan sampah. Bagian sebelah kanan terdapat tabel yang menampilkan isi *email*. Bagian atas ada subyek *email* dan label, nama pengirim *email*, tanggal, *button* kembali, *button* cetak, dan *button* balas.



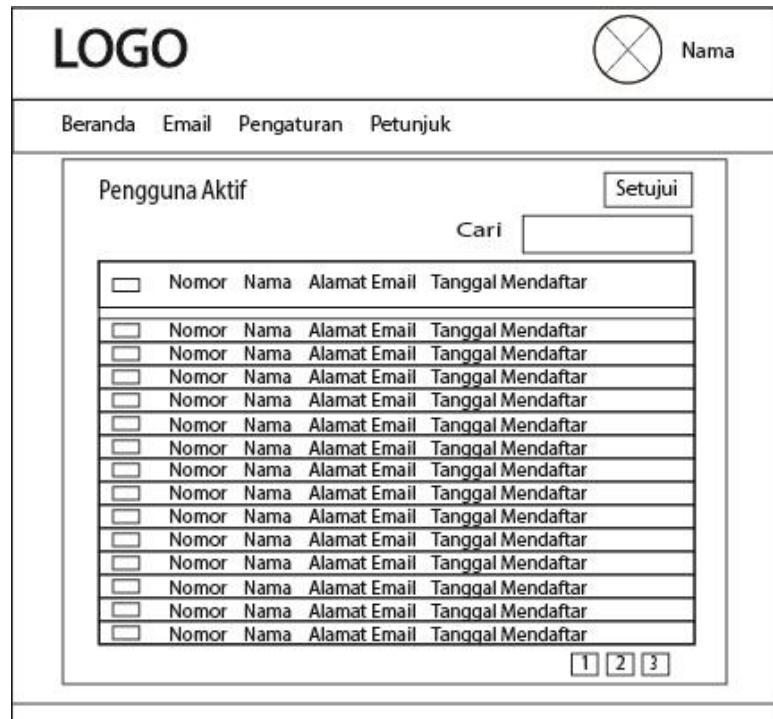
Gambar 3.22 Halaman Petunjuk

Pada gambar 3.22 bagian paling atas di halaman petunjuk di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman tulis yang di bagian kiri tedapat *sidebar* yang terdiri beberapa *menu* yaitu IMAP & SMTP, *email*, dan daftar anti *spam*. Bagian sebelah kanan terdapat informasi petunjuk penggunaan sistem.



Gambar 3.23. Halaman Pengguna Aktif

Pada gambar 3.23 bagian paling atas di halaman pengguna aktif di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu beranda*, *email pengaturan*, dan *petunjuk*. Bagian bawah *navbar* terdapat fitur inti dari pengguna aktif yaitu DataTables *server side* yang menampilkan data pengguna aktif. Data yang ditampilkan yaitu nomor, nama, alamat *email*, dan hak akses. Halaman pengguna aktif hanya dapat diakses oleh pengguna yang memiliki hak akses sebagai admin. Admin dapat mencari data pengguna lain berdasarkan kolom tabel yang ada. Admin dapat menghapus pengguna lain dengan cara memilih *checkbox* pada tabel dan menekan *button hapus*.



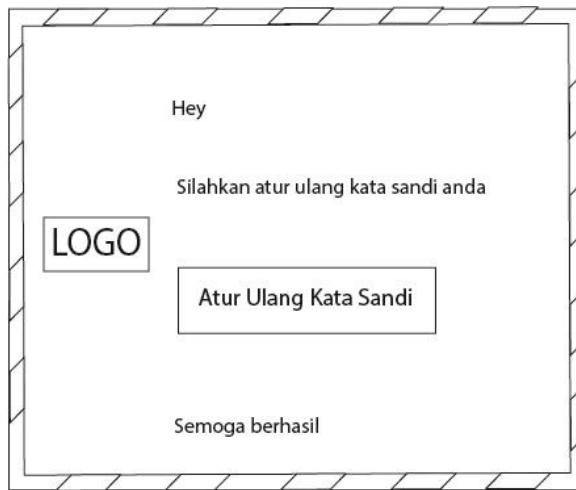
Gambar 3.24. Halaman Pengguna Baru

Pada gambar 3.24 bagian paling atas di halaman pengguna baru di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu beranda*, *email pengaturan*, dan *petunjuk*. Bagian bawah *navbar* terdapat fitur inti dari pengguna aktif yaitu DataTables *server side* yang menampilkan data pengguna baru. Data yang ditampilkan yaitu nomor, nama, alamat *email*, dan tanggal mendaftar. Halaman pengguna baru hanya dapat diakses oleh pengguna yang memiliki hak akses sebagai admin. Admin dapat mencari data pengguna lain berdasarkan kolom tabel yang ada. Admin dapat menyetujui pengguna lain dengan cara memilih *checkbox* pada tabel dan menekan *button setujui*.

The screenshot shows a user interface for editing a user profile. At the top left is a placeholder 'LOGO'. To its right is a circular icon containing a large 'X' and the word 'Nama' next to it. Below this is a horizontal navigation bar with four items: 'Beranda', 'Email', 'Pengaturan', and 'Petunjuk'. The main content area is titled 'FORM UBAH DATA PENGGUNA'. Inside, there are five input fields: 'Nama Depan', 'Nama Belakang', 'Alamat Email', 'Tanggal mendaftar', and 'Hak Akses'. At the bottom of this section are two buttons: 'Kembali' on the left and 'Simpan' on the right.

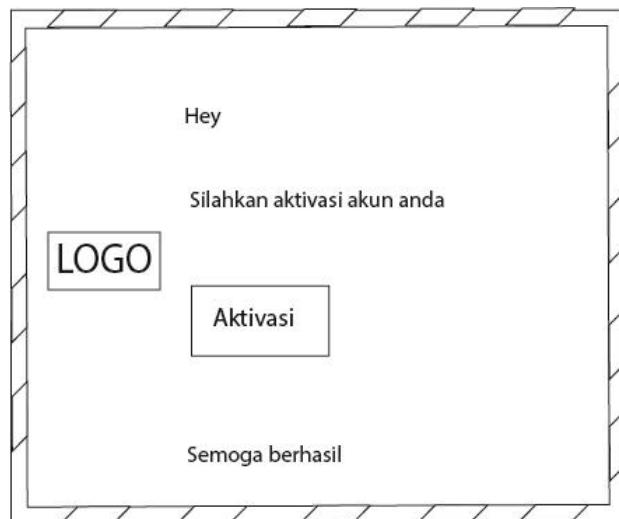
Gambar 3.25. Halaman Ubah Data Pengguna

Pada gambar 3.25 bagian paling atas di halaman ubah data pengguna di sebelah kiri terdapat logo dan di sebelah kanan terdapat gambar dan nama pengguna. Bagian di bawahnya terdapat *navbar* yang berisi beberapa *menu* yaitu *menu* beranda, *email* pengaturan, dan petunjuk. Bagian bawah *navbar* terdapat fitur inti dari halaman ubah pengguna yaitu *form* ubah data pengguna. Di dalam *form* terdapat *input* nama depan tipenya *text*, *input* nama belakang tipenya *text*, *input* alamat *email* tipenya *email*, *input* tanggal mendaftar tipenya *text*, *select* hak akses, *button* kembali, dan *button* simpan. Halaman ubah data pengguna hanya dapat diakses oleh pengguna yang memiliki hak akses sebagai admin. Admin hanya dapat merubah hak akses pengguna lain.



Gambar 3.26. *Email* Atur Ulang Kata Sandi

Pada gambar 3.26 *email* atur ulang kata sandi di dalamnya terdapat *button* atur ulang kata sandi, jika ditekan maka pengguna akan dialihkan ke halaman atur ulang kata sandi.

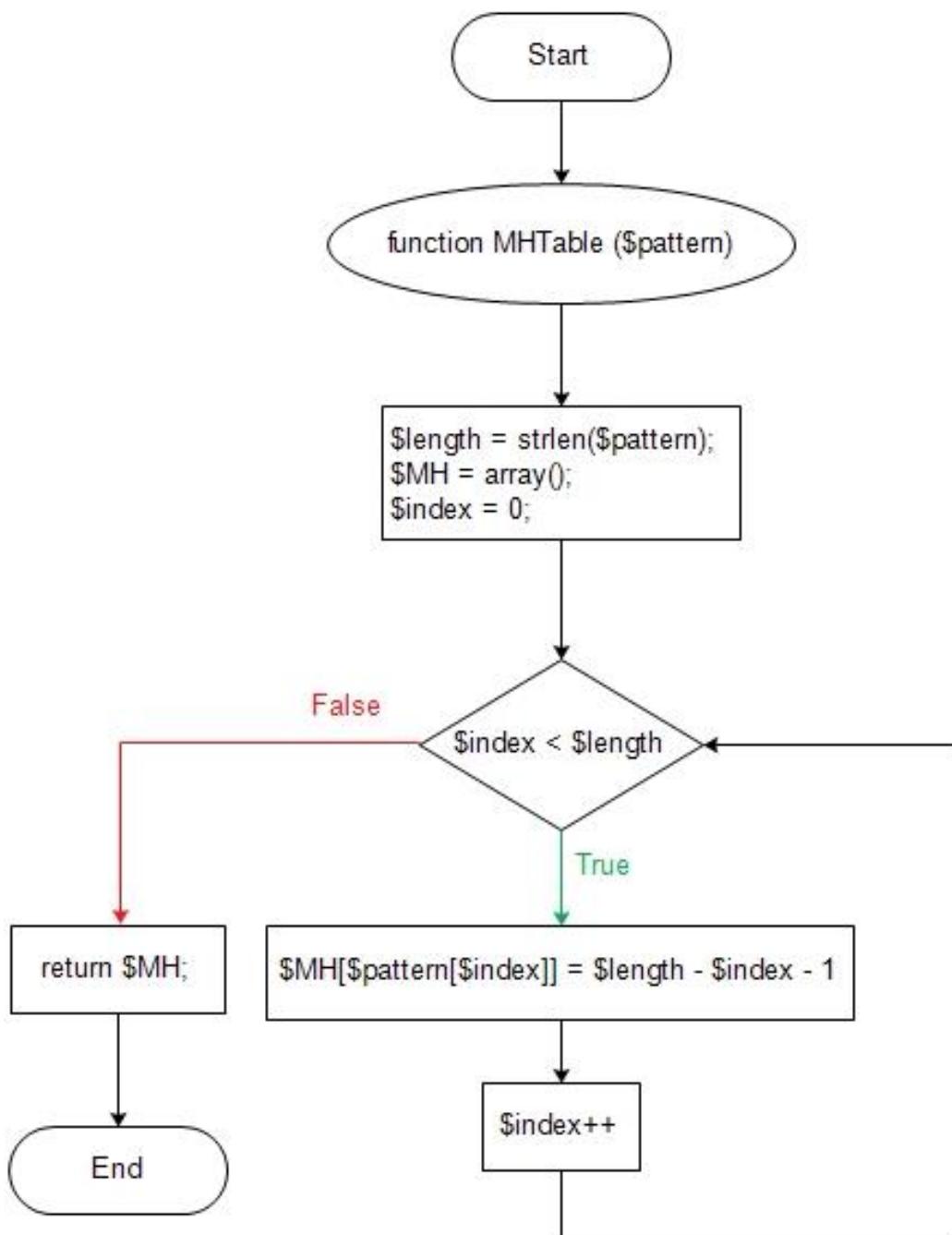


Gambar 3.27. *Email* Aktivasi Akun

Pada gambar 3.27 *email* aktivasi akun di dalamnya terdapat *button* atur ulang kata sandi, jika ditekan maka pengguna akan dialihkan ke halaman masuk setelah akunya teraktivasi.

3.2.4 *Construction*

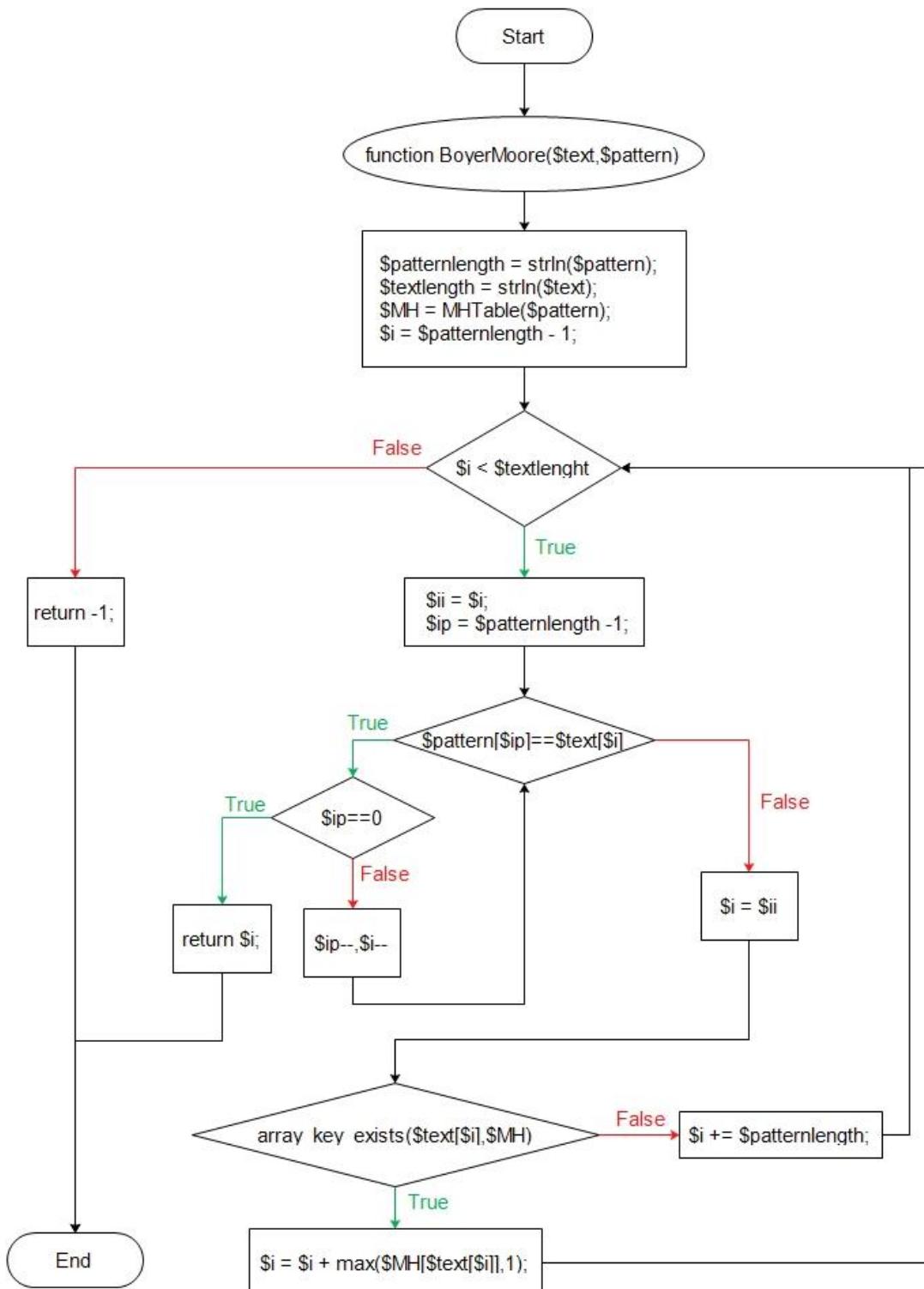
Tahapan *Construction* ini merupakan proses penerjemahan bentuk desain menjadi kode atau bentuk atau bahasa yang dapat dibaca oleh mesin. Setelah pengkodean selesai, pengkodean menggunakan bahasa pemrograman PHP dengan *framework* Laravel 5.8. *Text Editor* yang digunakan yaitu *Visual Studio Code*. Untuk desain tampilan atau *front end* menggunakan Bootstrap. Pada tahap ini dilakukan pengkodean algoritma Boyer Moore, algoritma Boyer Moore dibuat terdiri dari dua *function*, *function* pertama adalah *function* untuk menentukan nilai MH dari *pattern*, diagram alurnya dapat dilihat pada gambar 3.28. *Function* kedua adalah *function* untuk pencocokan *pattern* pada teks berdasarkan nilai MH yang sudah didapatkan dari *pattern*, diagram alirnya dapat dilihat pada gambar 3.29.



Gambar 3.28. *Function MHTable*

Function MHTable akan menghasilkan nilai MH dalam bentuk *array*, tahap adalah menghitung panjang *pattern* yang nilainya dimasukan pada variabel *length*, mendeklarasikan variabel MH sebagai *array*, dan memberikan nilai 0 pada variabel

index. Tahap selanjutnya adalah percabangan, di tahap percabangan akan membandingkan nilai antara variabel *index* dan variabel *length*. Jika nilai variabel *index* kurang dari variabel *length* maka ke tahapan selanjutnya yaitu perhitungan nilai variabel MH. Perhitungan nilai MH yaitu variabel *length* dikurangi nilai variabel *index* dan dikurangi 1. Tahap selanjutnya adalah menambahkan nilai 1 pada variabel *index*, setelah ditambahkan kembali ke tahap percabangan yang sebelumnya, jika nilai variabel *index* kurang dari nilai variabel *length* maka ke tahap perhitungan nilai MH seperti sebelumnya, jika nilai variabel *index* tidak kurang dari nilai variabel *length* maka proses selanjutnya adalah mencetak nilai MH.



Gambar 3.29. Function BoyerMoore

Function BoyerMoore akan menghasilkan nilai -1 jika *pattern* tidak ada yang cocok terhadap teks, jika ada kecocokan pada *pattern* terhadap teks maka akan menghasilkan nilai variabel i bernilai lebih dari -1. variabel i yang dihasilkan merupakan posisi karakter pada teks yang cocok terhadap *pattern*. Nilai yang dihasilkan dari *function* BoyerMoore akan menentukan *email* itu *spam* atau tidak. Terdapat 4 kolom pada tabel *email* di *database* sebagai tempat untuk menyimpan nilai dari *function* BoyerMoore yaitu kolom spam_subject, spam_email, spam_email dan spam_tld. Jika *output* = -1 maka nilai yang akan disimpan adalah 0 sedangkan jika *output* bukan -1 maka nilai yang akan disimpan adalah 1.

Kolom spam_subject tempat menyimpan nilai yang didapatkan berdasarkan hasil pencocokan *subject* pada pesan *email*. kolom spam_email tempat menyimpan nilai yang didapatkan berdasarkan hasil pencocokan alamat *email* pada pesan *email*. Kolom spam_domain tempat menyimpan nilai yang didapatkan berdasarkan hasil pencocokan nama *domain* pada pesan *email*. Kolom spam_tld tempat menyimpan nilai yang didapatkan berdasarkan hasil pencocokan TLD pada pesan *email*. Jika 4 kolom tersebut tidak ada yang memiliki nilai 1 maka *email* akan dianggap bukan *spam*, dan *email* tersebut akan ditampilkan pada *folder inbox*. Tetapi jika salah satu dari 4 kolom tersebut ada yang memiliki nilai 1 maka akan dianggap sebagai *email spam*, dan *email* tersebut akan ditampilkan pada *folder spam*.

3.2.5 Deployment

Tahap *Deployment* dilakukan menjalankan sistem pada *Virtual Private Server* (VPS) agar dapat diakses secara *online*, dikarenakan sistem ini menggunakan *framework* Laravel maka untuk mengunggah *repository project* ke

VPS menggunakan GIT. *Web server* menggunakan Nginx. Jika sistem sudah dapat diakses secara *online* maka dapat dilakukan pengujian *white box* dan *black box*.

3.3 Alat dan Bahan Penelitian

Alat dan bahan yang digunakan sebagai penunjang penelitian antara lain:

3.3.1 *Hardware*

Hardware yang digunakan untuk membuat sistem penyaringan *email* yaitu menggunakan laptop Dell Inspiron N4050, untuk spesifikasi laptop dapat dilihat pada tabel 3.3.

Tabel 3.3 Spesifikasi Laptop Dell N4050

No	<i>Hardware</i>	Spesifikasi
1	CPU	Intel Core i3 2310M 2.10GHz Sandy Bridge 32nm Technology
2	RAM	6,00GB Dual-Channel DDR3 665MHz (9-9-9-24)
3	<i>Motherboard</i>	Dell Inc. 02JCHC (CPU 1)
4	<i>Graphics</i>	Generic PnP Monitor (1366x768 60Hz) Intel HD Graphics 3000 (Dell)
5	<i>Storage</i>	465GB Western Digital WDC WD5000BPVT-75HXZT1 (SATA)

Hardware yang digunakan untuk *deployment* sistem penyaringan *email* yaitu menggunakan VPS (*Virtual Private Server*). *Virtual Private Server* (VPS) adalah *server* pribadi yang keseluruhan *resource* hanya digunakan oleh satu pengguna saja dan tidak dipengaruhi oleh pengguna lain. Pengguna dapat mengelola secara penuh semua konfigurasi dan *resource* yang ada pada VPS dan melakukan apapun yang diinginkan, untuk spesifikasi VPS yang digunakan dapat dilihat pada tabel 3.4.

Tabel 3.4 Spesifikasi VPS

No	<i>Hardware</i>	<i>Spesifikasi</i>
1	CPU	<i>3 Core Processor (7.2 GHz)</i>
2	RAM	3 GB
3	<i>Storage</i>	40 GB SSD
4	<i>Bandwidth</i>	1000 GB

3.3.2 *Software*

Software yang digunakan untuk pengembangan sistem penyaringan *email* pada penelitian ini dapat dilihat pada tabel 3.5.

Tabel 3.5 *Software* yang digunakan pada penelitian

No	Nama	Keterangan
1	OS Ubuntu 18.04 <i>Desktop</i>	Sistem operasi pada laptop Dell N4050
2	OS Ubuntu 18.04 <i>Server</i>	Sitem operasi pada VPS
3	<i>Visual Studio Code</i>	<i>Source Code Editor</i>
4	<i>Framework</i> Laravel	<i>Framework</i> bahasa pemrograman PHP yang dirilis dibawah lisensi MIT, dibangun dengan konsep MVC (<i>Model View Controller</i>)
5	PHP 7.3	Bahasa Pemrograman
6	Node. JS	Node.js adalah perangkat lunak yang didesain untuk mengembangkan aplikasi berbasis <i>web</i> dan ditulis dalam sintaks bahasa pemrograman JavaScript.
7	Nginx	<i>server</i> HTTP dan Proxy
8	MySQL	<i>Database</i>
9	Valet Linux	Lingkungan pengembangan Laravel untuk Linux yang minimalis
10	Weblex Laravel IMAP	IMAP <i>Library</i> for Laravel

No	Nama	Keterangan
11	Navicat Premium	Software untuk <i>remote database</i> secara GUI, dimana pengguna bisa mengkonfigurasi <i>database</i> secara grafik atau dengan secara instan tanpa menggunakan <i>script</i> .
12	GIT	Perangkat lunak pengontrol versi atau proyek manajemen kode perangkat lunak

3.4 Teknik Pengumpulan Data

Teknik pengumpulan data pada penelitian ini menggunakan uji performa algoritma, *white box* dan *black box* berdasarkan standar ISO 9126. Analisis standar ISO 9126 pada aspek *functionality*, *portability*, *efficiency*, dan *usability*. Dalam pengujian dibutuhkan instrumen penelitian. Instrumen dalam penelitian ini digunakan untuk membantu dalam melakukan analisis kualitas dari perangkat lunak yang dikembangkan. Instrumen yang digunakan pada penelitian ini antara lain:

3.4.1 Instrumen Performa Algoritma

Pengujian pada aspek ini dilakukan dengan mencatat *load time* dari penyaringan *email spam* yang menggunakan algoritma Boyer Moore.

3.4.2 Instrumen White Box

Dalam *white box testing* terdapat beberapa tahapan yaitu *basic path testing*, dan *cyclometric complexity*. Tahapan awal adalah *basic path testing* membuat *flow graph* berdasarkan *source code* program. Setelah membuat *flow graph* selanjutnya adalah menghitung jumlah *edge* dan *node* yang ada pada *flow graph*. Jumlah *edge* dan *node* yang ada di masukkan ke dalam rumus pencari *cyclometric complexity*. Setelah diketahui nilai *cyclometric complexity* maka dapat diketahui jumlah maksimal *independent path*. Setelah *independent path* diperoleh maka langkah selanjutnya dilakukan *test case*.

3.4.3 Instrumen *Black Box*

3.4.3.1 Instrumen *Functionality*

Instrumen pengujian aspek *functionality* berupa *checklist* daftar fungsi pada sistem yang dijabarkan dari prosedur-prosedur penggunaan program yang telah divalidasi.

3.4.3.2 Instrumen *Portability*

Pengujian pada aspek *portability* dilakukan dengan cara mengoperasikan sistem penyaringan *email* pada beberapa jenis *browser*. Pengujian akan dilakukan menggunakan alat [browsershots.org](#).

3.4.3.3 Instrumen *Efficiency*

Proses pengujian dilakukan dengan bantuan beberapa alat. Alat yang dipakai dalam pengujian adalah Yslow dan PageSpeed.

3.4.3.4 Instrumen *Usability*

Instrumen yang digunakan dalam melakukan analisis aspek *usability* menggunakan *System Usability Scale* (SUS). Penyarataan-pernyataan dalam instrumen pengujian SUS dapat dilihat pada tabel 3.6.

Tabel 3.6. Instrumen Pengujian *System Usability Scale* (SUS)

No	Pernyataan
1	Saya berpikir akan menggunakan sistem ini lagi.
2	Saya merasa sistem ini rumit untuk digunakan.
3	Saya merasa sistem ini mudah untuk digunakan
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.

No	Pernyataan
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat
8	Saya merasa sistem ini membingungkan
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.

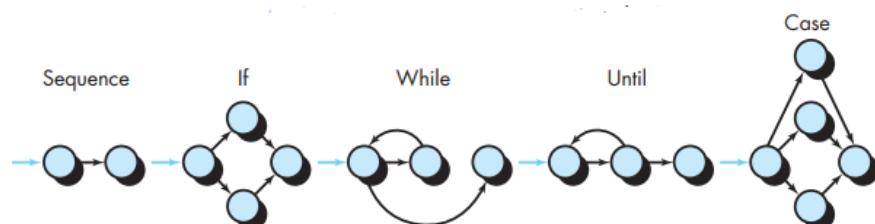
Sumber: (Zahra Sharfina dan Harry Budi Santoso, 2016: 147)

3.5 Teknik Analisis Data

Teknik analisis data yang digunakan adalah sebagai berikut:

3.5.1 Analisis *Basic Path Testing*

Langkah pengujian ini dilakukan dengan menggambarkan alur jalannya program ke dalam graf alur (*flow graph*), untuk menggambarkan jalannya program digunakan *node* dan *edges*. (Roger S. Pressman & Bruce R. Maxim, 2014: 501) *node* digunakan untuk menggambarkan sebuah proses dan keputusan suatu program yang berjalan, sedangkan *edges* digunakan untuk menggambarkan aliran program sesuai dengan *node* yang digambarkan. Untuk menggambarkan alur kontrol dalam graf alur (*flow graph*) diperlukan sebuah notasi sederhana, berikut merupakan notasi yang digunakan dalam menggambarkan proses dalam graf alur :



Gambar 3.30 Notasi *Flow Graph*

Sumber: (Roger S. Pressman & Bruce R. Maxim, 2014: 501)

Basic path testing bertujuan untuk mengetahui jalannya alur program selain itu dapat digunakan untuk mengetahui adanya kesalahan suatu kode pada program. Untuk mengetahui *basic path testing* diterapkan.

3.5.2 Analisis *Cyclomatic Complexity*

Selain *basic path testing* tahap lainnya adalah *cyclomatic complexity* merupakan suatu tahap untuk mengetahui banyaknya *path* (alur program) yang ada pada graf alur (*flow graph*). (Roger S. Pressman & Bruce R. Maxim, 2014: 503) dalam penerapan graf alur (*flow graph*) *cyclomatic complexity* mendefinisikan jumlah *independent path* dan menyediakan batas atas untuk sejumlah uji coba yang harus dilakukan, hal ini bertujuan untuk memastikan seluruh perintah dalam graf alur telah dieksekusi semua, sedikitnya satu kali. (Roger S. Pressman & Bruce R. Maxim, 2014: 502) *independent path* sendiri merupakan alur program yang berisi sedikitnya satu kumpulan perintah pemrosesan pada program atau suatu kondisi baru pada program. Untuk mencari berapa banyak *path* yang ada pada *flow graph* yang sudah digambarkan dapat diketahui dengan menggunakan perhitungan seperti dibawah ini:

$$V(G) = E - N + 2$$

Keterangan

$$V(G) = \text{Cylomatic complexity}$$

E = jumlah *edge* pada graf

N = jumlah *node* pada graf

Cyclomatic complexity berpengaruh pada kualitas *software*, ada hubungan antara *cyclomatic complexity* dan resiko dalam suatu prosedur, hubungan tersebut dapat dilihat padat tabel 3.7.

Tabel 3.7 Hubungan *Cyclomatic Complexity* dengan Resiko

<i>Cyclomatic Complexity</i>	Evaluasi Resiko
1-10	Sederhana, risiko rendah
11-20	Kompleksitas sedang, risiko sedang
21-50	Kompleks, berisiko tinggi
> 50	Risiko sangat tinggi

Sumber: (Sachhidanand, 2011: 32)

3.5.3 Analisis *Independent Path*

Independent path adalah jalur yang melalui program yang memperkenalkan setidaknya satu rangkaian pernyataan pemrosesan baru atau kondisi baru. Ketika dinyatakan dalam *flow graph*, *independent path* harus bergerak sepanjang setidaknya satu *edge* yang belum dilintasi sebelum jalur ditentukan (Roger S. Pressman & Bruce R. Maxim, 2014: 502).

3.5.4 Analisis *Test Case*

(Roger S. Pressman & Bruce R. Maxim, 2014: 504) berikut cara melakukan *test case*:

1. Dengan menggunakan desain atau kode sebagai dasar, gambarkan *flow graph* yang sesuai.
2. Tentukan *cyclometric complexity* dari *flow graph*.
3. Tentukan sebuah set dari *independent path* secara linier.

4. Mempersiapkan *test case* yang akan memaksa pelaksanaan setiap jalur di set dasar.

3.5.5 Analisis *Functionality*

Analisis untuk aspek *functionality* dilakukan dengan teknik deskriptif yaitu menganalisis persentase *functionality* dan hasil perhitungan skor persentase untuk masing-masing penilaian. Pada lembar validasi jawaban setiap *item* pertanyaan menggunakan skala guttman. Menurut Sugiyono (2017: 96), skala pengukuran dengan tipe ini akan didapat jawaban yang tegas yaitu ya-tidak, benar-salah, pernah-tidak pernah, positif-negatif. Jawaban dapat dibuat dalam bentuk *checklist* dengan skor tinggi satu dan skor rendah nol. Hasil dari pengujian tersebut selanjutnya dilakukan analisis menggunakan perhitungan berikut:

$$\text{Persentase } functionality = \frac{\sum x}{n} \times 100\%$$

Keterangan:

n: Jumlah semua fungsi yang ada

$\sum x$: Jumlah fungsi yang berjalan dengan baik

Hasil dari perhitungan tersebut mengindikasikan persentase banyaknya fitur yang berjalan dengan baik atau tidak berjalan baik. Pada pengujian aspek *functionality*.

3.5.6 Analisis *Portability*

Analisis untuk aspek *portability* dilakukan dengan melakukan observasi fungsionalitas aplikasi padabagai macam *browser* yang berbeda untuk

mengetahui apakah sistem ini dapat berjalan atau tidak. Selanjutnya dilakukan perhitungan skor persentase hasil pengujian menggunakan rumus berikut:

$$\text{Persentase portability} = \frac{\text{Jumlah browser yang sesuai}}{\text{Jumlah browser yang diujicoba}} \times 100\%$$

3.5.7 Analisis *Efficiency*

Analisis aspek *efficiency* pada penelitian ini dilakukan pengujian menggunakan parameter dari YSlow dan PageSpeed. Analisis aspek *efficiency* dilakukan dengan melihat hasil pengujian dari YSlow dan PageSpeed pada komponen besarnya *bytes* data dokumen, jumlah HTTP *request* dan *score/grade* akhir.

3.5.8 Analisis *Usability*

Pengujian aspek *usability* dilakukan pada mahasiswa yang masih aktif program studi pendidikan teknik informatika dan komputer angkatan 2015. Untuk responden yang diteliti tidaklah semua karena keterbatasan tenaga dan waktu maka menggunakan teknik sampling. (Sugiyono, 2017:81) teknik *sampling* merupakan teknik pengambilan sampel untuk menentukan sampel yang akan digunakan dalam penelitian. Responden penelitiannya adalah mahasiswa program studi pendidikan teknik informatika dan komputer angkatan 2015 di universitas negeri semarang yang masih aktif di bulan Oktober tahun 2019 berjumlah 52 orang. Pada teknik *sampling* dibutuhkan tingkat kesalahan atau *margin of error* yang dimulai dari 1%, 5%, atau 10% (Fajri Ismail, 2018: 47).

Menghitung jumlah sampel yang diteliti dapat menggunakan rumus slovin:

$$n = \frac{N}{1 + Ne^2}$$

n = ukuran sampel

N = ukuran populasi

e = taraf kesalahan pengambilan sampel yang dapat ditoleransi

Diketahui bahwa ukuran populasi yaitu berjumlah 52, dan taraf kesalahan yang digunakan adalah 10%, maka ukuran sampelnya adalah:

$$n = \frac{N}{1 + Ne^2}$$

$$n = \frac{52}{1 + 52 \times 0,1^2}$$

$$n = \frac{52}{1,52}$$

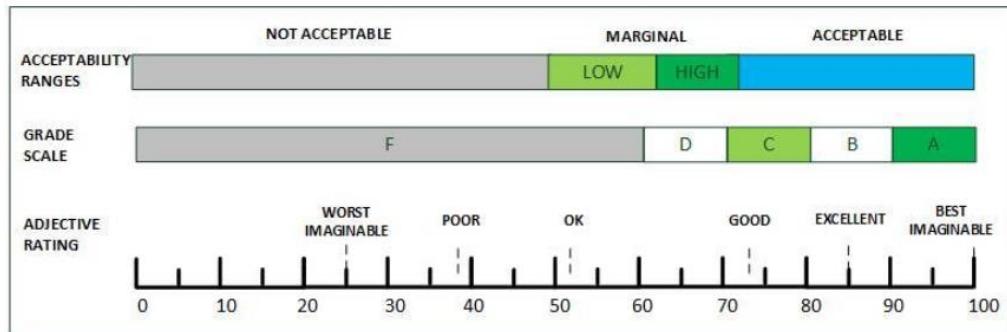
n = 34,21 (35 Orang)

Dalam pengujian aspek *usability* menggunakan *System Usability Scale* (SUS). SUS terdiri dari 10 pernyataan dengan menggunakan skala likert 1 sampai 5 dengan rincian sangat tidak setuju berbobot 1, tidak setuju berbobot 2, netral berbobot 3, setuju berbobot 4, dan sangat setuju berbobot 5. Pada pernyataan bernomor ganjil, skor tiap pernyataan dihitung dengan cara bobot tiap pernyataan dikurangi 1. Pernyataan bernomor genap, skor dihitung dengan cara 5 dikurangi bobot tiap pernyataan. Total skor didapatkan dengan menjumlahkan seluruh skor tiap pernyataan (bernomor genap maupun ganjil). Berikutnya menghitung hasil

akhir yaitu rerata dari semua total skor yang sudah dikalikan dengan 2,5 (Ardiansyah dan Muhammad Imam Ghazali, 2016: 215).

SUS dalam menentukan hasil perhitungan penilaian terdapat tiga sudut pandang yaitu *acceptability*, *grade scale*, dan *adjective rating*. *Acceptability* terdapat tiga tingkatan yang terdiri dari *not acceptable*, *marginal* (rendah dan tinggi), dan *acceptable*. Sedangkan *grade scale* terdiri dari A, B, C, D dan F. Untuk *adjective rating* lebih banyak tingkatan yaitu *worst imaginable*, *poor*, *ok*, *good*, *excelent* dan *best imanginable*. Dari ketiga penilaian SUS seperti yang diperlihatkan pada gambar 3.31 bahwa *acceptability* digunakan untuk melihat tingkat penerimaan pengguna terhadap perangkat lunak, *grade scale* untuk melihat tingkatan (*grade*) perangkat lunak, dan *adjective rating* untuk melihat *rating* dari perangkat lunak yang dihasilkan. Selain dari ketiga cara tersebut SUS memiliki cara lain dalam melakukan penentuan hasil peneilaian yaitu dengan cara SUS *score percentile rank*. Penentuan hasil penilaian berdasarkan SUS *score percentile rank* dilakukan secara umum berdasarkan hasil perhitungan penilaian pengguna. SUS *score percentile rank* memiliki perbedaan dengan *acceptability*, *grade scale*, *adjective rating* yang dikelompokkan menjadi tiga kategori (Ependi, *et al.*, 2019: 71). Berikut adalah ketentuan penentuan penilaian pada SUS *score percentile rank*:

- a) *Grade A* : dengan skor $\geq 80,3$.
- b) *Grade B* : dengan skor ≥ 74 dan $< 80,3$.
- c) *Grade C* : dengan skor ≥ 68 dan < 74 .
- d) *Grade D* : dengan skor ≥ 51 dan < 68 .
- e) *Grade F* : dengan skor lebih < 51 .



Gambar 3.31 Penilaian System Usability Scale (SUS)

Sumber: (Ependi, et. al., 2019:71)

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Penelitian

Sistem penyaringan *email* dibangun menggunakan *framework back-end* Laravel yang menggunakan bahasa pemrograman PHP agar mudah dikembangkan dikarenakan komunitasnya yang besar. *Database* yang digunakan adalah MySQL. *Framework front-end* yang digunakan adalah Bootstrap.

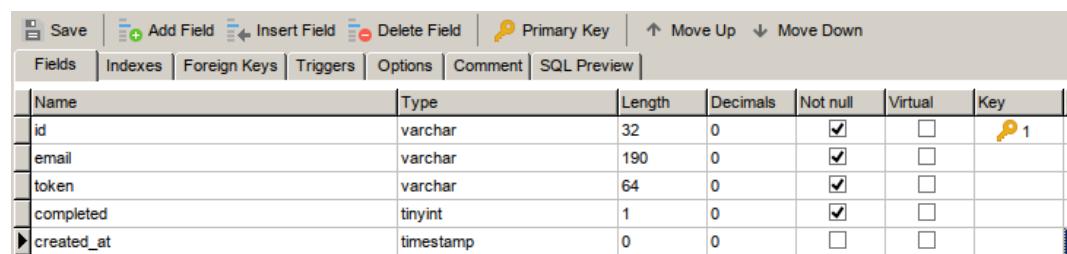
4.1.1 Hasil Desain Sistem

Hasil desain sistem terbagi menjadi dua, yaitu desain *database* dan desain tampilan.

4.1.1.1 Hasil Desain *Database*

a) Tabel *Activations*

Gambar 4.1 tabel *activations* yang digunakan untuk meyimpan data pengguna baru yang ingin mengaktifasi akun, di dalam tabel *activation* terdapat kolom *token*, isi kolom token bersifat unik yang berfungsi untuk mengidentifikasi permintaan aktivasi akun pengguna.



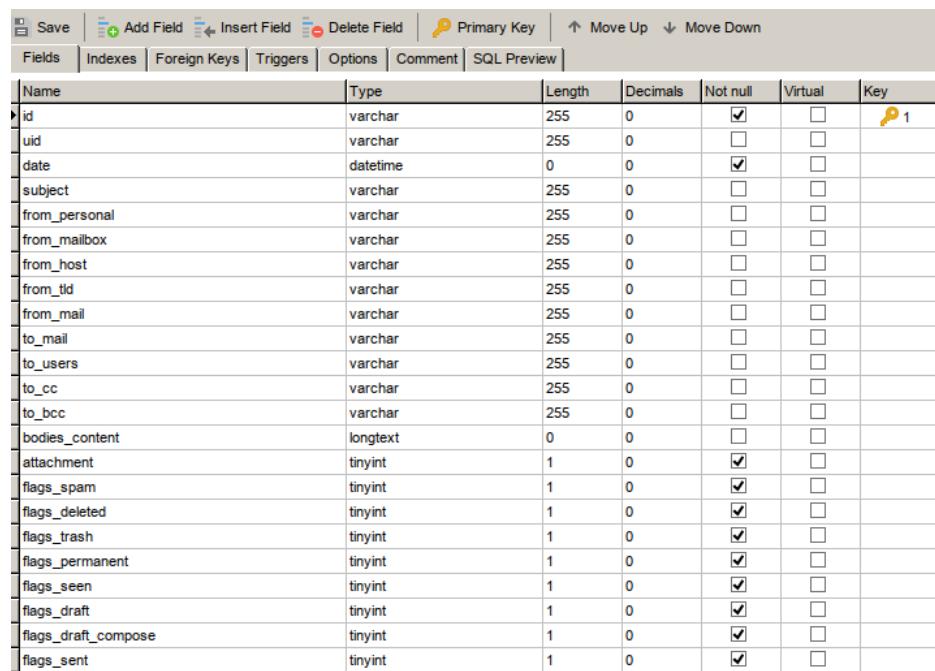
The screenshot shows a database schema editor interface with the following table structure:

Name	Type	Length	Decimals	Not null	Virtual	Key
<code>id</code>	<code>varchar</code>	32	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
<code>email</code>	<code>varchar</code>	190	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
<code>token</code>	<code>varchar</code>	64	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
<code>completed</code>	<code>tinyint</code>	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
<code>created_at</code>	<code>timestamp</code>	0	0	<input type="checkbox"/>	<input type="checkbox"/>	

Gambar 4.1. Hasil Tabel *Activations*

b) Tabel *Emails*

Gambar 4.2 tabel *emails* yang digunakan untuk menyimpan data-data *email* pengguna yang didapatkan dari *provider* melalui IMAP.



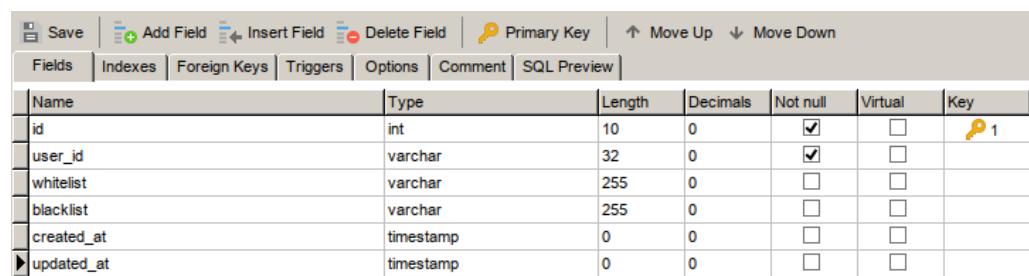
The screenshot shows a database schema editor interface with a toolbar at the top containing Save, Add Field, Insert Field, Delete Field, Primary Key, Move Up, Move Down, Fields, Indexes, Foreign Keys, Triggers, Options, Comment, and SQL Preview buttons. Below the toolbar is a table definition:

Name	Type	Length	Decimals	Not null	Virtual	Key
id	varchar	255	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
uid	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
date	datetime	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
subject	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
from_personal	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
from_mailbox	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
from_host	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
from_tld	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
from_mail	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
to_mail	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
to_users	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
to_cc	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
to_bcc	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
bodies_content	longtext	0	0	<input type="checkbox"/>	<input type="checkbox"/>	
attachment	tinyint	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
flags_spam	tinyint	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
flags_deleted	tinyint	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
flags_trash	tinyint	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
flags_permanent	tinyint	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
flags_seen	tinyint	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
flags_draft	tinyint	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
flags_draft_compose	tinyint	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
flags_sent	tinyint	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Gambar 4.2. Hasil Tabel *Emails*

c) Tabel Filterdomain

Gambar 4.3 tabel filterdomain yang digunakan untuk menyimpan data pengaturan penyaringan *email* yang berdasarkan parameter *domain*.



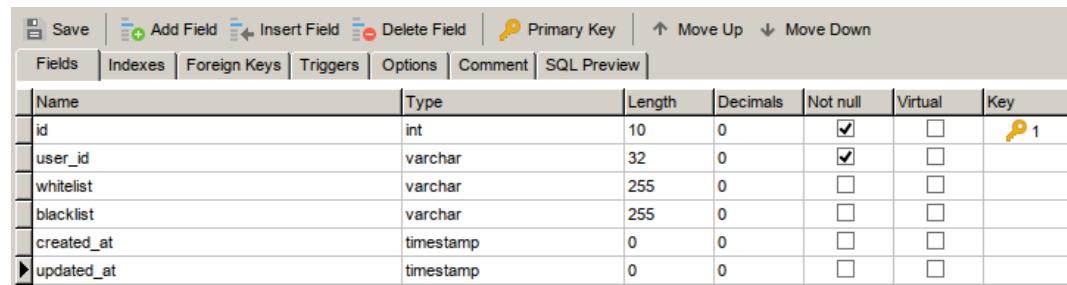
The screenshot shows a database schema editor interface with a toolbar at the top containing Save, Add Field, Insert Field, Delete Field, Primary Key, Move Up, Move Down, Fields, Indexes, Foreign Keys, Triggers, Options, Comment, and SQL Preview buttons. Below the toolbar is a table definition:

Name	Type	Length	Decimals	Not null	Virtual	Key
id	int	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
user_id	varchar	32	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
whitelist	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
blacklist	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
created_at	timestamp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	
updated_at	timestamp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	

Gambar 4.3. Hasil Tabel Filterdomain

d) Tabel Filteremail

Gambar 4.4 tabel filteremail yang digunakan untuk menyimpan data pengaturan penyaringan *email* yang berdasarkan parameter alamat *email*.

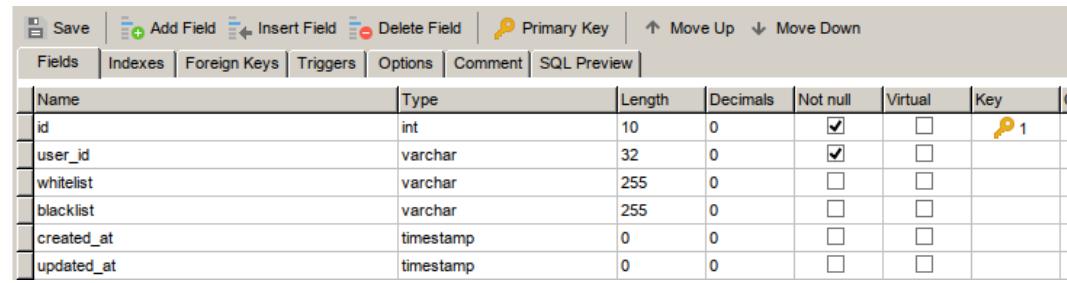


Name	Type	Length	Decimals	Not null	Virtual	Key
<code>id</code>	int	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
<code>user_id</code>	varchar	32	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
<code>whitelist</code>	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
<code>blacklist</code>	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
<code>created_at</code>	timestamp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	
<code>updated_at</code>	timestamp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	

Gambar 4.4. Hasil Tabel Filteremail

e) Tabel Filtersubject

Gambar 4.5 tabel filtersubject yang digunakan untuk menyimpan data pengaturan penyaringan *email* yang berdasarkan parameter subyek.



Name	Type	Length	Decimals	Not null	Virtual	Key
<code>id</code>	int	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
<code>user_id</code>	varchar	32	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
<code>whitelist</code>	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
<code>blacklist</code>	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
<code>created_at</code>	timestamp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	
<code>updated_at</code>	timestamp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	

Gambar 4.5. Hasil Tabel Filtersubject

f) Tabel FilterId

Gambar 4.6 tabel filterId yang digunakan untuk menyimpan data pengaturan penyaringan *email* yang berdasarkan parameter TLD (*Top Level Domain*).

Name	Type	Length	Decimals	Not null	Virtual	Key
id	int	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
user_id	varchar	32	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
whitelist	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
blacklist	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
created_at	timestamp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	
updated_at	timestamp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	

Gambar 4.6. Hasil Tabel FilterId

g) Tabel *Migrations*

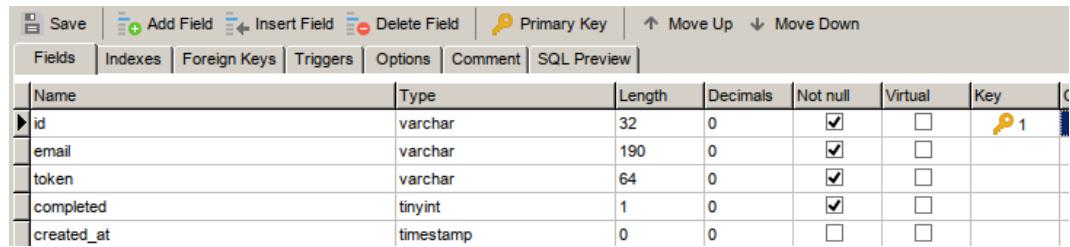
Gambar 4.7 tabel *migrations* yang digunakan untuk menyimpan data *migration*. *Migration* adalah sebuah fitur yang ada pada laravel, *migration* merupakan *control version system* untuk *database*, dengan menggunakan *migration* laravel dapat mempermudah mengelola *database*.

Name	Type	Length	Decimals	Not null	Virtual	Key
id	int	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
migration	varchar	255	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
batch	int	11	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Gambar 4.7. Hasil Tabel *Migrations*

h) Tabel Password_resets

Gambar 4.8 tabel password_resets yang digunakan untuk menyimpan data permintaan pengaturan ulang kata sandi oleh pengguna.



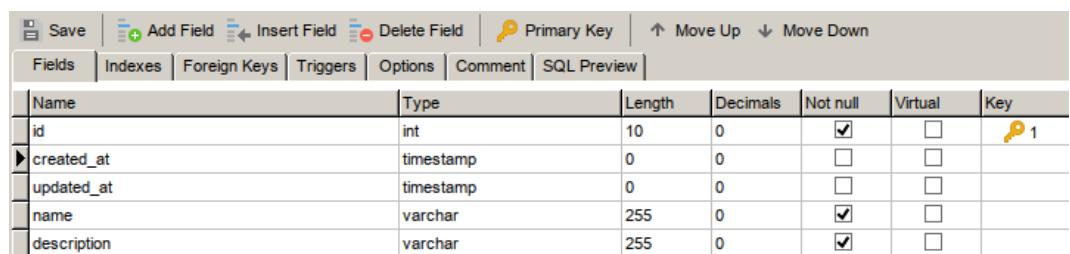
The screenshot shows the 'Fields' tab of the MySQL Workbench table configuration interface. The table structure is as follows:

Name	Type	Length	Decimals	Not null	Virtual	Key
id	varchar	32	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
email	varchar	190	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
token	varchar	64	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
completed	tinyint	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
created_at	timestamp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	

Gambar 4.8. Hasil Tabel Password_resets

i) Tabel *Roles*

Gambar 4.9 tabel *roles* yang digunakan untuk menyimpan data jenis hak akses pengguna di dalam sistem.



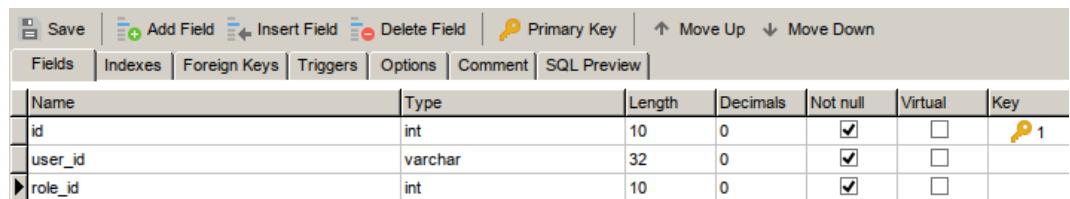
The screenshot shows the 'Fields' tab of the MySQL Workbench table configuration interface. The table structure is as follows:

Name	Type	Length	Decimals	Not null	Virtual	Key
id	int	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
created_at	timestamp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	
updated_at	timestamp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	
name	varchar	255	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
description	varchar	255	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Gambar 4.9. Hasil Tabel *Roles*

j) Tabel Role_user

Gambar 4.10 tabel role_user yang digunakan untuk menghubungkan tabel *users* dan *roles*.



The screenshot shows the 'Fields' tab of the MySQL Workbench table configuration interface. The table structure is as follows:

Name	Type	Length	Decimals	Not null	Virtual	Key
id	int	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
user_id	varchar	32	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
role_id	int	10	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Gambar 4.10. Hasil Tabel Role_user

k) Tabel *Users*

Gambar 4.11 tabel *users* yang digunakan untuk menyimpan data pengguna sistem.

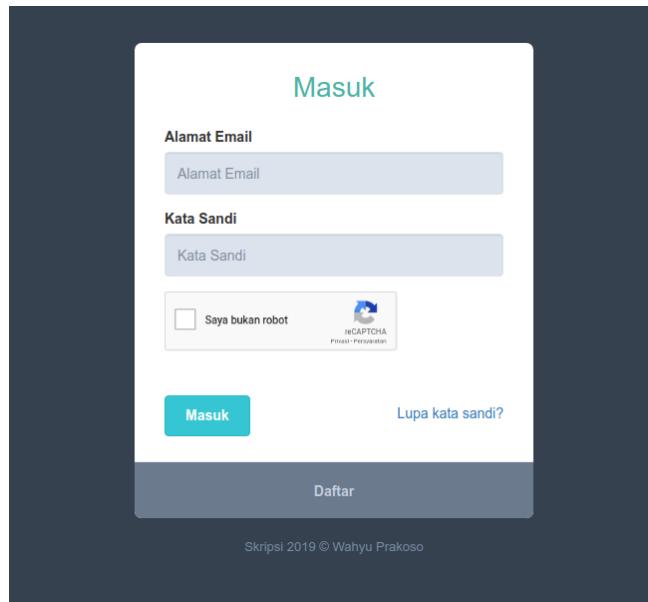
Name	Type	Length	Decimals	Not null	Virtual	Key
id	varchar	32	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1
first_name	varchar	255	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
last_name	varchar	255	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
email	varchar	190	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
password	varchar	255	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
remember_token	varchar	100	0	<input type="checkbox"/>	<input type="checkbox"/>	
session	varchar	255	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
imap_host	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
imap_port	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
imap_encryption	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
smtp_host	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
smtp_port	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
smtp_encryption	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>	
connected_imap	tinyint	1	0	<input type="checkbox"/>	<input type="checkbox"/>	
connected_smtp	tinyint	1	0	<input type="checkbox"/>	<input type="checkbox"/>	
confirmed	tinyint	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
activated	tinyint	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
created_at	timestamp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	
updated_at	timestamp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	

Gambar 4.11. Hasil Tabel *Users*

4.1.1.2 Hasil Desain Tampilan

a) Hasil Tampilan Halaman Masuk

Ditunjukan pada gambar 4.12 merupakan tampilan halaman masuk ketika pengguna ingin mengakses ke sistem. di dalam halaman masuk terdapat *hyperlink* daftar yang berfungsi untuk mengarahkan pengguna ke halaman daftar agar pengguna baru bisa mendaftar akun, serta terdapat *hyperlink* lupa kata sandi yang berfungsi untuk mengarahkan pengguna menuju halaman lupa kata sandi untuk melakukan permintaan pengaturan ulang kata sandi. Untuk masuk ke halaman beranda pengguna harus mengisi lengkap alamat *email*, kata sandi dan mencentang *recaptcha* lalu tekan tombol masuk.



Gambar 4.12. Hasil Tampilan Halaman Masuk

b) Hasil Tampilan Halaman Daftar

Ditunjukan pada gambar 4.13 merupakan tampilan halaman daftar ketika pengguna baru ingin mendaftar agar dapat mengakses sistem. pengguna harus mengisi data pribadi seperti nama depan, nama belakang, alamat *email*, kata sandi, dan ketik ulang kata sandi. Data pribadi pengguna harus sesuai dengan data pribadi yang ada pada *provider email* yang akan digunakan dan dihubungkan. Data terpenting yang harus sama yaitu alamat *email* dan kata sandi. Setelah mengisi lengkap data pribadi tekan tombol kirimkan, setelah itu sistem akan mengirim *email* untuk mengaktivasi akun ke alamat *email* pengguna.

The screenshot shows a registration form titled "Daftar". The form asks for personal details: "Nama Depan" (First Name), "Nama Belakang" (Last Name), "Alamat Email" (Email Address), and "Kata Sandi" (Password). It also includes a password confirmation field "Ketik Ulang Kata Sandi" (Repeat Password) and a CAPTCHA section with the text "Saya bukan robot" (I am not a robot) and a reCAPTCHA checkbox. At the bottom are "Kembali" (Back) and "Kirimkan" (Send) buttons.

Gambar 4.13. Hasil Tampilan Halaman Daftar

c) Hasil Tampilan Halaman Lupa Kata Sandi

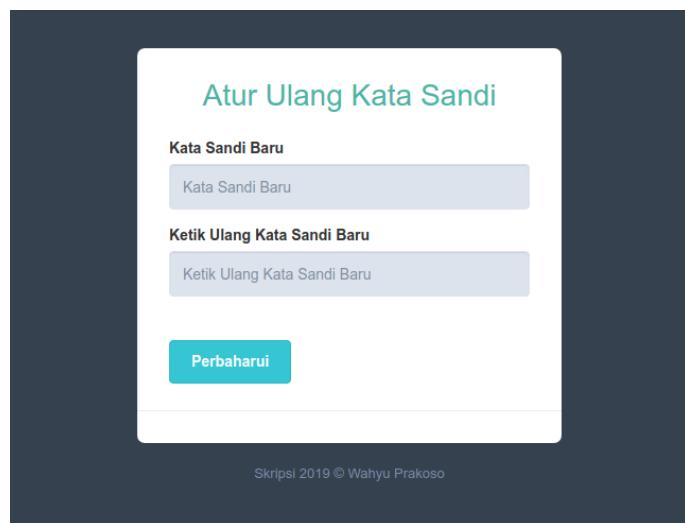
Jika pengguna ingin masuk tetapi lupa kata sandi, pengguna dapat mengatur ulang kata sandi di lupa kata sandi di halaman atur ulang kata sandi yang tampilannya seperti gambar 4.14. Pengguna hanya memasukan data alamat *email* yang terdaftar dan tekan tombol kirimkan. Pengguna akan menerima *email* untuk mengatur ulang kata sandi, pengguna diharapkan untuk memeriksa kotak masuk *email*, hasil tampilan *email* atur ulang kata sandi dapat dilihat pada gambar 4.34.

The screenshot shows a password reset form titled "Lupa Kata Sandi ?". It asks for the email address "Masukan alamat email anda untuk mengatur ulang kata sandi!" and provides a "Email" input field. At the bottom are "Kembali" (Back) and "Kirimkan" (Send) buttons.

Gambar 4.14. Hasil Tampilan Halaman Lupa Kata Sandi

d) Hasil Tampilan Halaman Atur Ulang Kata Sandi

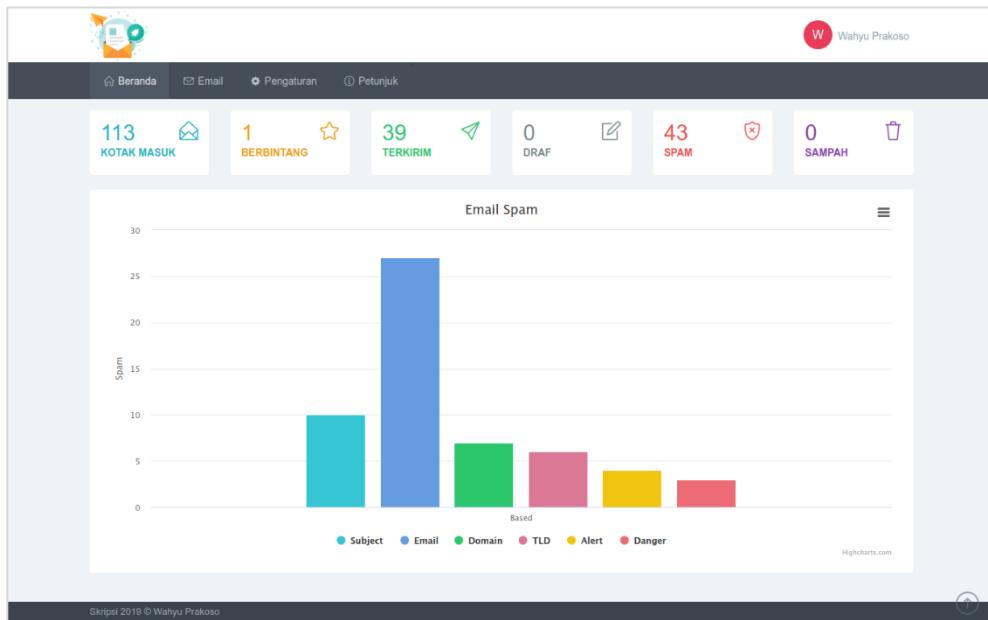
Setelah mendapatkan *email* atur ulang kata sandi dan pengguna menekan tombol atur ulang kata sandi seperti pada gambar 4.34 maka pengguna akan dialihkan ke halaman atur ulang kata sandi seperti gambar 4.15. Pada halaman ini pengguna hanya perlu mengisi kata sandi baru, ketik ulang kata sandi baru dan tekan tombol perbaharui.



Gambar 4.15. Hasil Tampilan Halaman Atur Ulang Kata Sandi

e) Hasil Tampilan Halaman Beranda

Halaman pengguna masuk maka pengguna akan dialihkan ke halaman beranda seperti pada gambar 4.16. Di halaman beranda tersajikan jumlah *email* susai *folder* yang ada yaitu kotak masuk, berbintang, terkirim, draf, *spam* dan sampah. Di halaman beranda juga terdapat grafik yang menampilkan jumlah *email* yang dikategorikan sebagai *spam* sesuai parameter yang ada. Tanda *email spam* berdasarkan parameter dapat dilihat pada tabel 4.1.



Gambar 4.16. Hasil Tampilan Halaman Beranda

f) Hasil Tampilan Informasi Pribadi

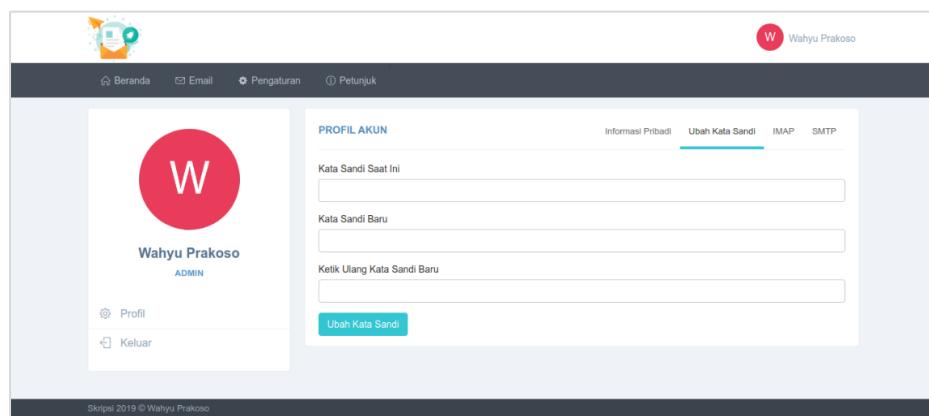
Pada gambar 4.17 halaman informasi pribadi, pengguna dapat merubah nama depan dan nama belakang, alamat *email* tidak bias dirubah.

The screenshot shows the 'PROFIL AKUN' (Personal Account) page. On the left, there is a circular profile picture with a red 'W' and the name 'Wahyu Prakoso' below it, with 'ADMIN' status. There are also 'Profil' and 'Keluar' buttons. The main form has tabs: 'Informasi Pribadi' (selected), 'Ubah Kata Sandi', 'IMAP', and 'SMTP'. The 'Informasi Pribadi' tab contains fields for 'Nama Depan' (Wahyu), 'Nama Belakang' (Prakoso), and 'Alamat Email' (mr.wahyuprakoso.developer@gmail.com). A 'Simpan' (Save) button is at the bottom.

Gambar 4.17. Hasil Tampilan Halaman Informasi Pribadi

g) Hasil Tampilan Halaman Ubah Kata Sandi

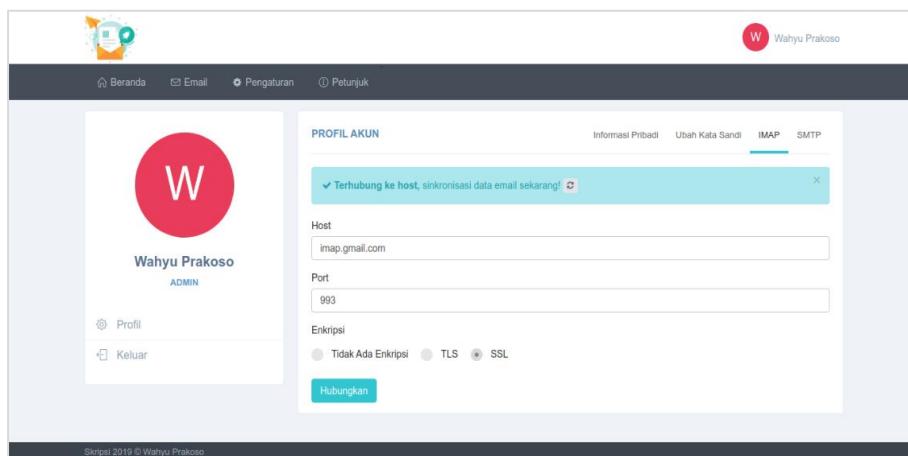
Pengguna dapat merubah kata sandi melalui halaman ubah kata sandi seperti pada gambar 4.18, pengguna hanya perlu memasukan kata sandi yang sedang dipakai, memasukan kata sandi yang baru, mengetik ulang kata sandi baru dan menekan tombol perbaharui.



Gambar 4.18. Hasil Tampilan Halaman Ubah Kata Sandi

h) Hasil Tampilan Halaman IMAP

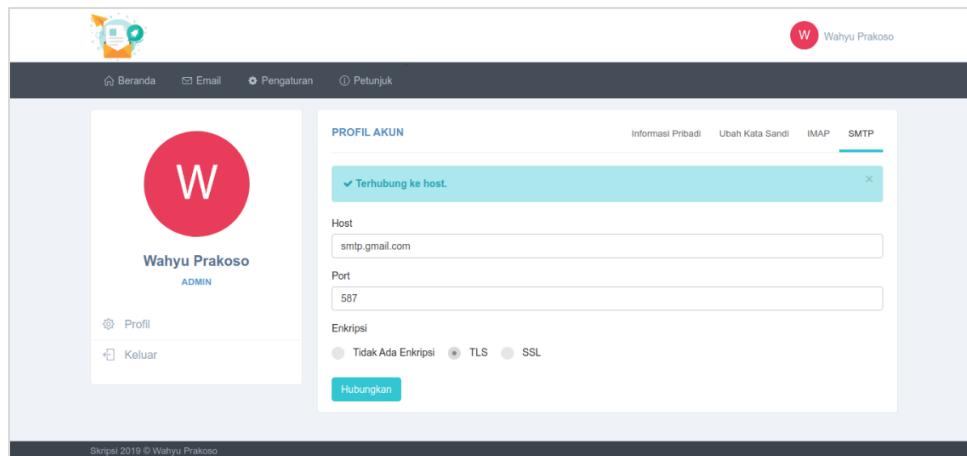
Pada gambar 4.18 halaman IMAP terdapat *form* yang harus dilengkapi yaitu *host*, *port*, dan enkripsi. Pengguna harus melengkapi *form* agar menghubungkan data *email* dari *provider*.



Gambar 4.19. Hasil Tampilan Halaman IMAP

i) Hasil Tampilan Halaman SMTP

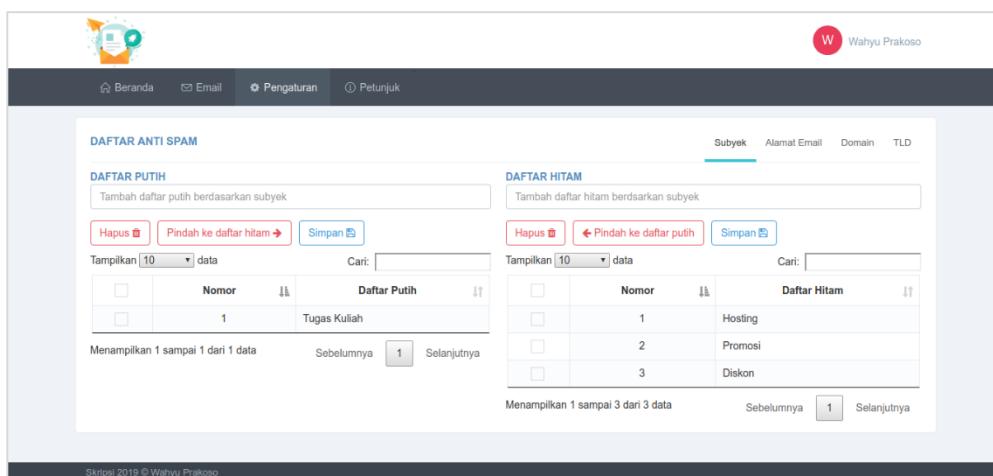
Pada gambar 4.20 halaman SMTP terdapat *form* yang harus dilengkapi yaitu *host*, *port*, dan enkripsi. Pengguna harus melengkapi *form* agar menghubungkan data *email* dari *provider*.



Gambar 4.20. Hasil Tampilan Halaman SMTP

j) Hasil Tampilan Halaman Daftar Anti *Spam*

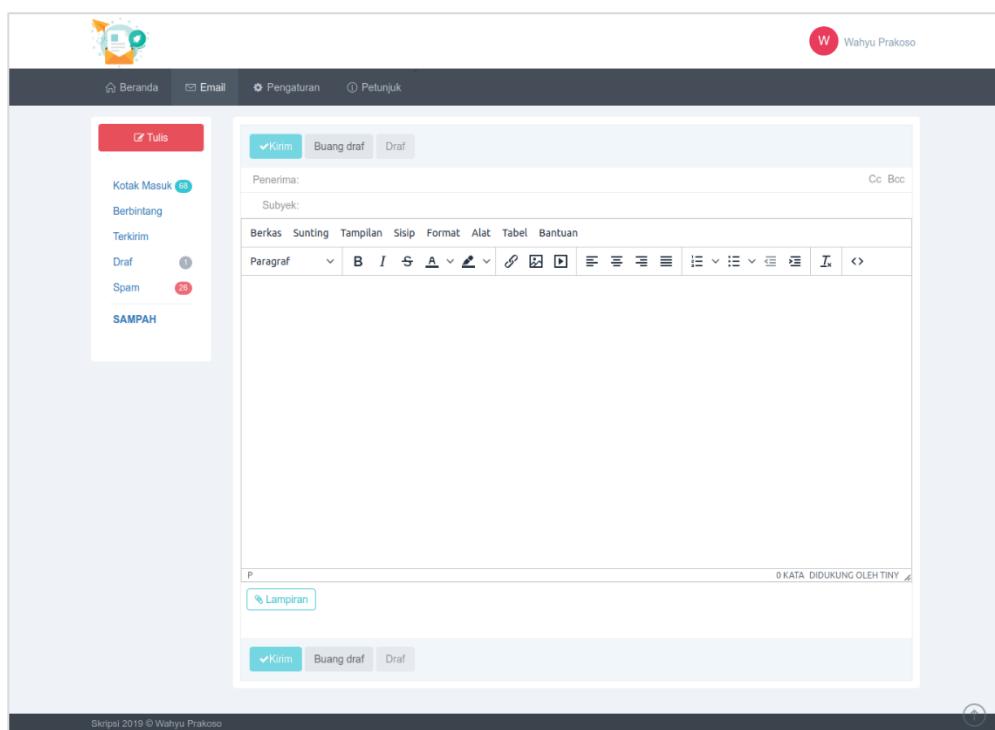
Pada gambar 4.21 halaman daftar anti *spam* pengguna dapat mengatur bagaimana caranya dalam penyaringan *email*. Penyaringan *email* berdasarkan pada subyek, alamat *email*, *domain*, *Top Level Domain* (TLD).



Gambar 4.21. Hasil Tampilan Halaman Daftar Anti *Spam*

k) Hasil Tampilan Halaman Tulis

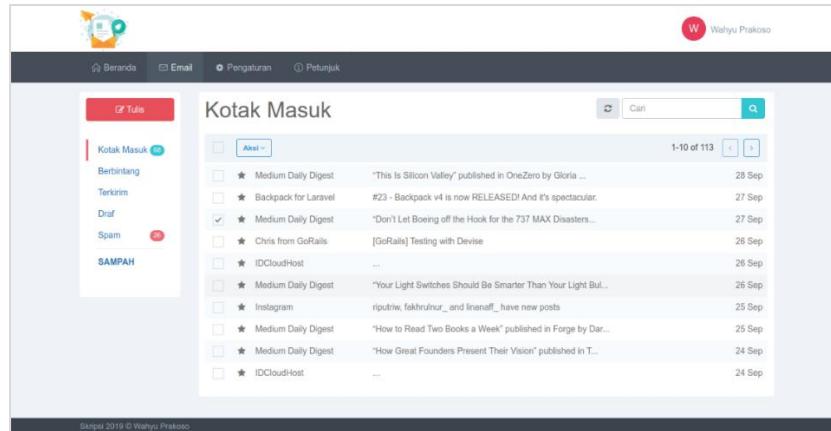
Jika pengguna ingin mengirim *email* maka pengguna dapat menulisnya pada halaman tulis seperti gambar 4.22. Masukan alamat *email* yang dituju, subyek dan isi *email*. Isi *email* tidak hanya berupa teks saja, pengguna dapat mengirim html *email* sesuai desain yang diinginkan. Pengguna dapat mengirim lampiran berkas maksimal 25 MB.



Gambar 4.22. Hasil Tampilan Halaman Tulis

l) Hasil Tampilan Halaman Kotak Masuk

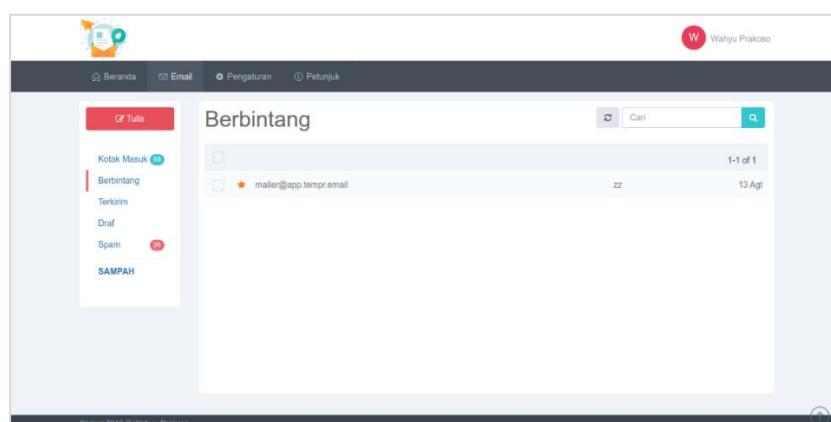
Pada gambar 4.23 halaman kotak masuk menampilkan data-data *email* yang masuk ke pengguna. Data-data *email* tidak ditampilkan semuanya hanya nama depan, nama belakang, subyek dan tanggal pengiriman *email*.



Gambar 4.23. Hasil Tampilan Halaman Kotak Masuk

m) Hasil Tampilan Halaman Berbintang

Pada gambar 4.24 halaman berbintang menampilkan data-data *email* yang masuk ke pengguna yang diberi tanda bintang. Data-data *email* tidak ditampilkan semuanya hanya nama depan, nama belakang, subyek dan tanggal pengiriman *email*.



Gambar 4.24. Hasil Tampilan Halaman Berbintang

n) Hasil Tampilan Halaman Terkirim

Pada gambar 4.25 halaman terkirim menampilkan data-data *email* terkirim.

Data-data *email* tidak ditampilkan semuanya hanya nama depan, nama belakang, subyek dan tanggal pengiriman *email*.

	Subject	Date
<input type="checkbox"/>	★ WahyuPrakoso Test SMTP	20 Agt
<input type="checkbox"/>	★ Email Panel Activate your account now!	14 Agt
<input type="checkbox"/>	★ Email Panel Hei Wahyu, Reset password	13 Agt
<input type="checkbox"/>	★ Email Panel Hei Wahyu, Reset password	13 Agt
<input type="checkbox"/>	★ Email Panel Hei Wahyu, Reset password	13 Agt
<input type="checkbox"/>	★ Email Panel Hei Wahyu, Reset password	13 Agt
<input type="checkbox"/>	★ Email Panel Hei Wahyu, Reset password	13 Agt
<input type="checkbox"/>	★ Email Panel Hei Wahyu, Atur Ulang Sandi AKun myCMA Sekarang !	09 Agt
<input type="checkbox"/>	★ Email Panel Hei Wahyu, Atur Ulang Sandi AKun myCMA Sekarang !	09 Agt

Gambar 4.25. Hasil Tampilan Halaman Terkirim

o) Hasil Tampilan Halaman Draf

Pada gambar 4.26 halaman draf menampilkan data-data *email* draf. Data-data *email* tidak ditampilkan semuanya hanya nama depan, nama belakang, subyek dan tanggal pengiriman *email*.

	Subject	Date
<input type="checkbox"/>	★ Wahyu Prakoso draf 1	5 detik yang lalu

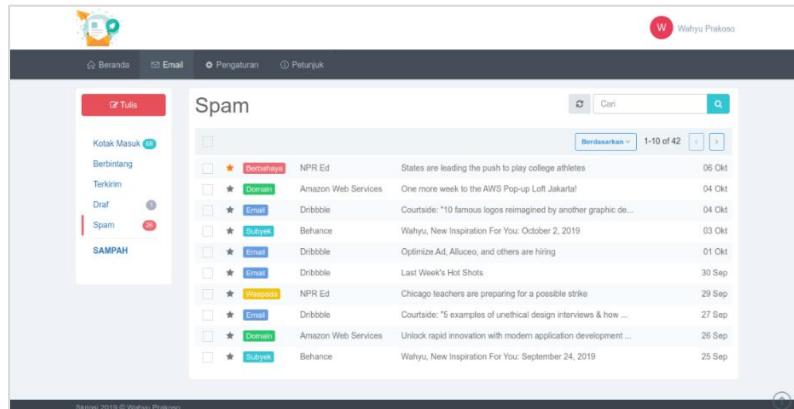
Gambar 4.26. Hasil Tampilan Halaman Draf

p) Hasil Tampilan Halaman *Spam*

Pada gambar 4.27 halaman *spam* menampilkan data-data *email* yang dianggap *spam* sesuai penyaringan *email* yang telah diatur oleh pengguna. Data-data *email* tidak ditampilkan semuanya hanya nama depan, nama belakang, subyek dan tanggal pengiriman *email*. Setiap *email spam* akan diberi tanda berdasarkan parameter untuk mempermudah pengguna dalam memilah kembali *email spam*, informasi detail tanda tersebut bisa dilihat pada tabel 4.1.

Tabel 4.1 Tanda *Email Spam*

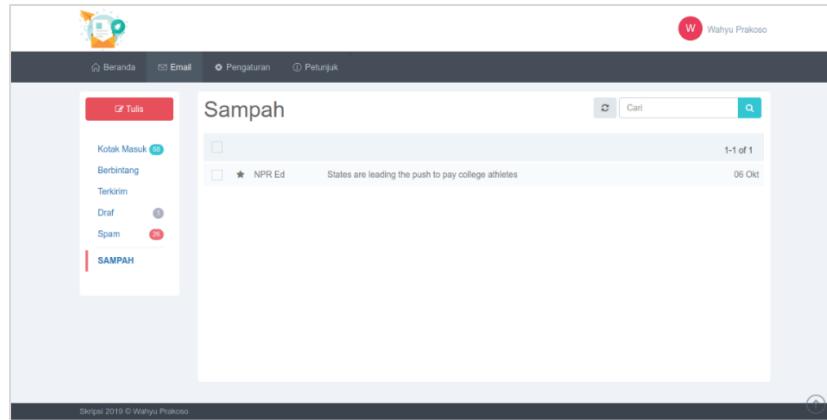
Tanda	Keterangan
Subyek	<i>Email</i> dianggap <i>spam</i> berdasarkan parameter subyek.
Email	<i>Email</i> dianggap <i>spam</i> berdasarkan parameter alamat <i>email</i> .
Domain	<i>Email</i> dianggap <i>spam</i> berdasarkan parameter <i>domain</i> .
TLD	<i>Email</i> dianggap <i>spam</i> berdasarkan parameter TLD.
Waspada	<i>Email</i> dianggap <i>spam</i> berdasarkan 2 parameter.
Berbahaya	<i>Email</i> dianggap <i>spam</i> berdasarkan lebih dari 2 parameter.



Gambar 4.27. Hasil Tampilan Halaman *Spam*

q) Hasil Tampilan Halaman Sampah

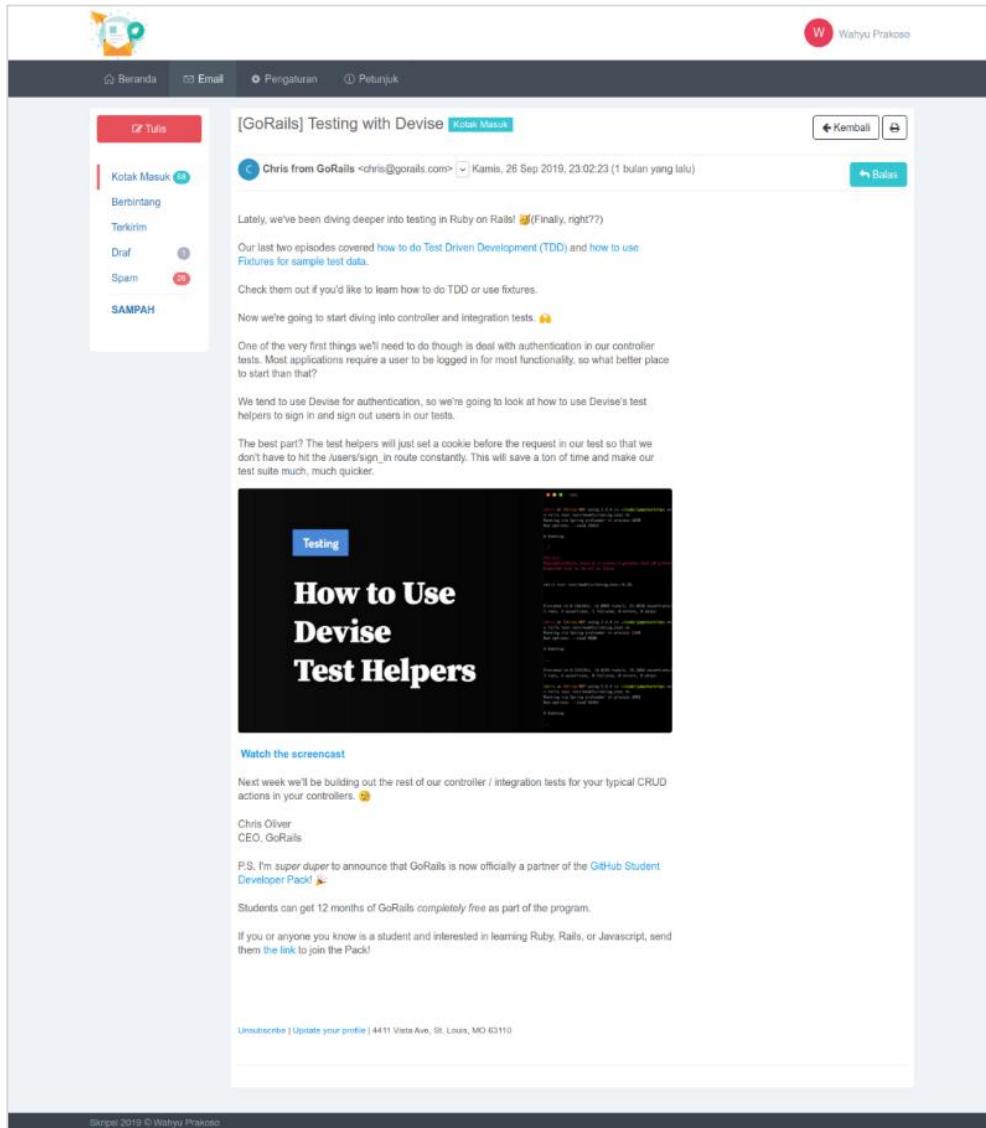
Pada gambar 4.28 halaman sampah menampilkan data-data *email* yang telah dihapus oleh pengguna. Data-data *email* tidak ditampilkan semuanya hanya nama depan, nama belakang, subyek dan tanggal pengiriman *email*.



Gambar 4.28. Hasil Tampilan Halaman Sampah

r) Hasil Tampilan Halaman Lihat *Email*

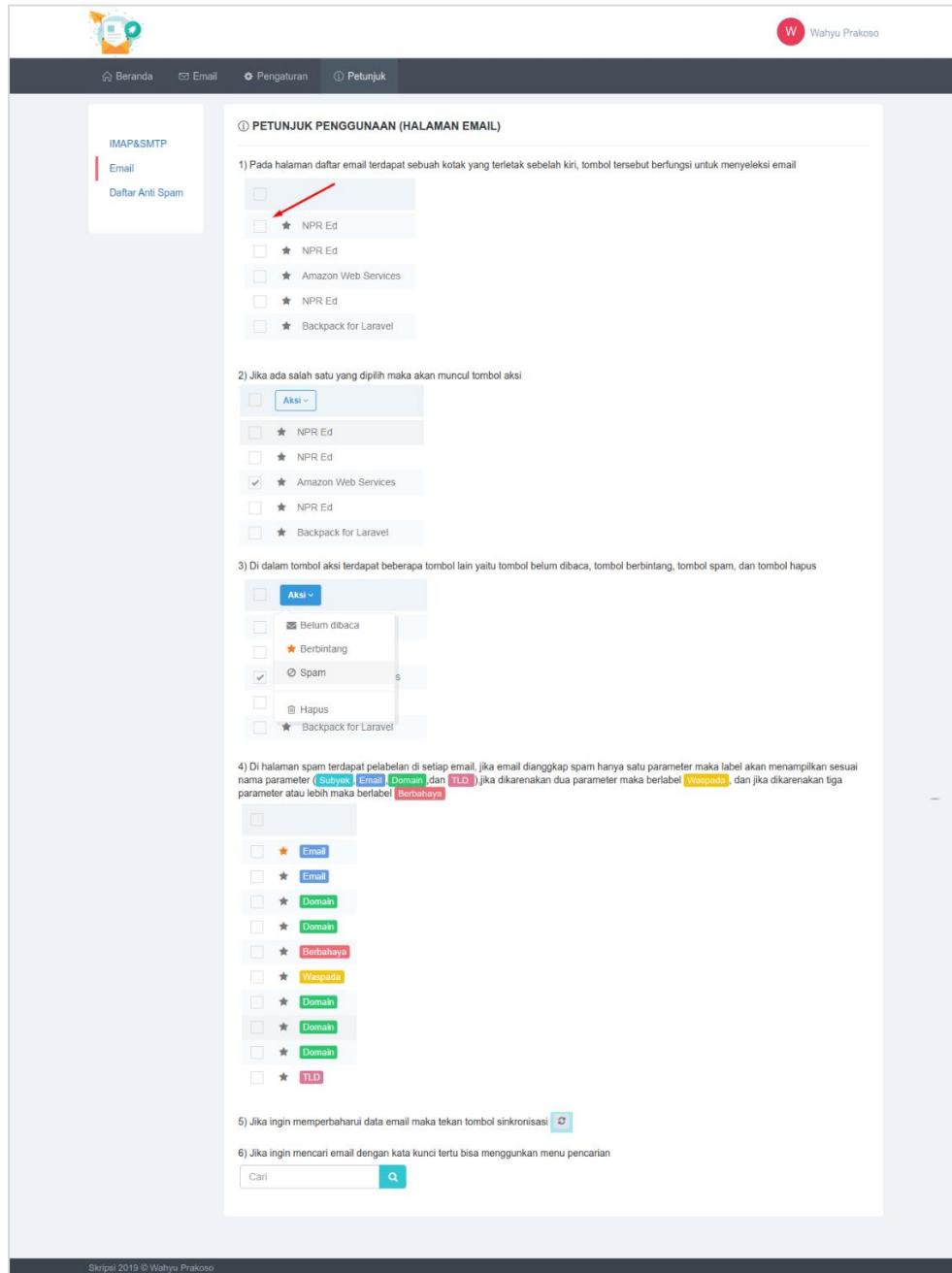
Pada gambar 4.29 halaman ini untuk menampilkan isi *email* serta data *header* seperti nama pengirim *email*, alamat *email* pengirim dan tanggal pengiriman *email*. Terdapat tombol cetak *email*, balas *email* dan kembali.



Gambar 4.29. Hasil Tampilan Halaman Lihat Email

s) Hasil Tampilan Halaman Petunjuk

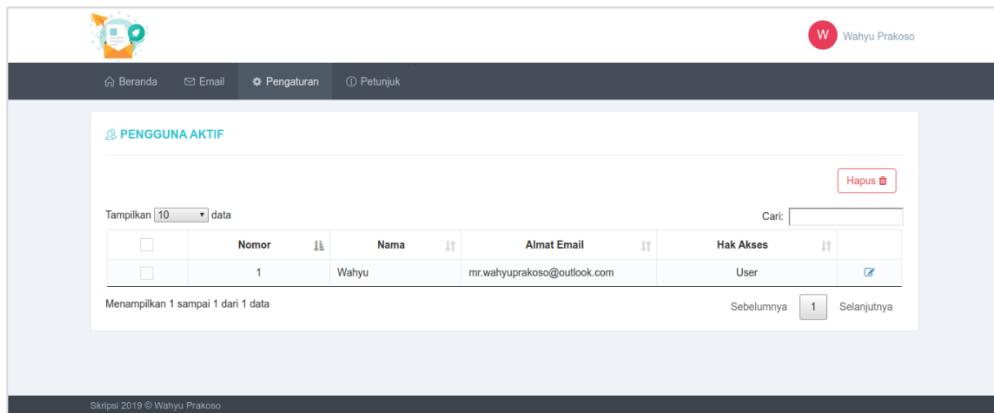
Pada gambar 4.30 halaman petunjuk berfungsi untuk memberikan petunjuk penggunaan sistem, terdapat petunjuk pengaturan IMAP dan SMTP serta petunjuk penggunaan fitur-fitur pada halaman *email* dan penyaringan *email*.



Gambar 4.30. Hasil Tampilan Halaman Petunjuk

t) Hasil Tampilan Halaman Pengguna Aktif

Pada gambar 4.31 halaman pengguna aktif menampilkan data pengguna yang sudah disetujui oleh admin, halaman ini hanya dapat diakses oleh pengguna yang memiliki hak akses sebagai admin.



The screenshot shows a web application interface for managing users. At the top, there is a navigation bar with links for 'Beranda', 'Email', 'Pengaturan', and 'Petunjuk'. A user profile icon for 'Wahyu Prakoso' is also present. Below the navigation bar, the title 'PENGUNA AKTIF' is displayed. A search bar labeled 'Cari:' and a delete button labeled 'Hapus' are visible. A table lists one user record:

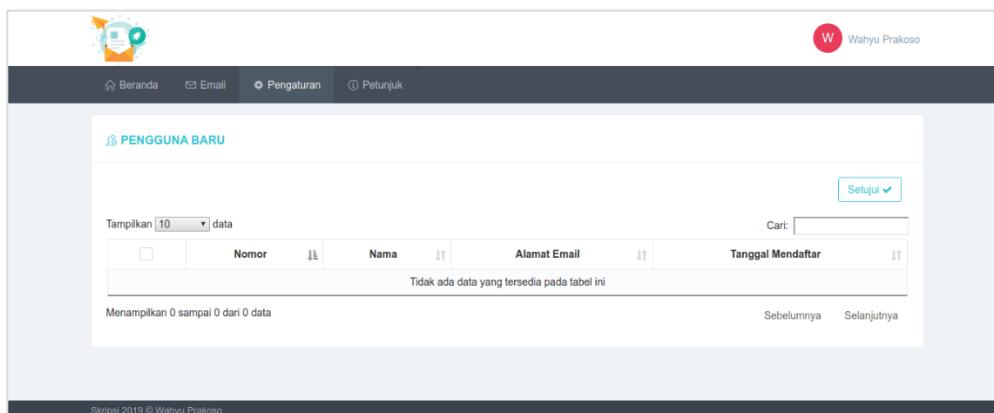
	Nomor	Nama	Alamat Email	Hak Akses
<input type="checkbox"/>	1	Wahyu	mr.wahyuprakoso@outlook.com	User 

Below the table, it says 'Menampilkan 1 sampai 1 dari 1 data'. Navigation buttons 'Sebelumnya' (1), 'Selanjutnya', and 'Skripsi 2019 © Wahyu Prakoso' are at the bottom.

Gambar 4.31. Hasil Tampilan Halaman Pengguna Aktif

u) Hasil Tampilan Halaman Pengguna Baru

Pada gambar 4.32 halaman pengguna baru menampilkan data pengguna yang baru saja mendaftar, halaman ini hanya dapat diakses oleh pengguna yang memiliki hak akses sebagai admin.



The screenshot shows a web application interface for managing users. At the top, there is a navigation bar with links for 'Beranda', 'Email', 'Pengaturan', and 'Petunjuk'. A user profile icon for 'Wahyu Prakoso' is also present. Below the navigation bar, the title 'PENGUNA BARU' is displayed. A search bar labeled 'Cari:' and a 'Setujui' button are visible. A table is shown with the following columns: Nomor, Nama, Alamat Email, and Tanggal Mendaftar. A message 'Tidak ada data yang tersedia pada tabel ini' is displayed above the table. Below the table, it says 'Menampilkan 0 sampai 0 dari 0 data'. Navigation buttons 'Sebelumnya', 'Selanjutnya', and 'Skripsi 2019 © Wahyu Prakoso' are at the bottom.

Gambar 4.32. Hasil Tampilan Halaman Pengguna Baru

v) Hasil Tampilan Halaman Ubah Data Pengguna

Pengguna yang mempunyai hak akses sebagai admin dapat merubah data pengguna sistem melalui halaman ubah data pengguna seperti gambar 4.33, pada halaman tersebut ditampilkan nama depan, nama belakang, alamat *email*,

tanggal mendaftar, dan hak akses. Namun admin hanya dapat merubah hak akses pengguna saja.

FORM UBAH DATA PENGGUNA	
Nama Depan *	Wahyu
Nama Belakang *	Prakoso
Alamat Email *	mr.wahyuprakoso@outlook.com
Tanggal Mendaftar *	14-08-2019 12:17:26
Hak Akses *	User

Gambar 4.33. Hasil Tampilan Halaman Ubah Data Pengguna

w) Hasil Tampilan *Email* Atur Ulang Kata Sandi

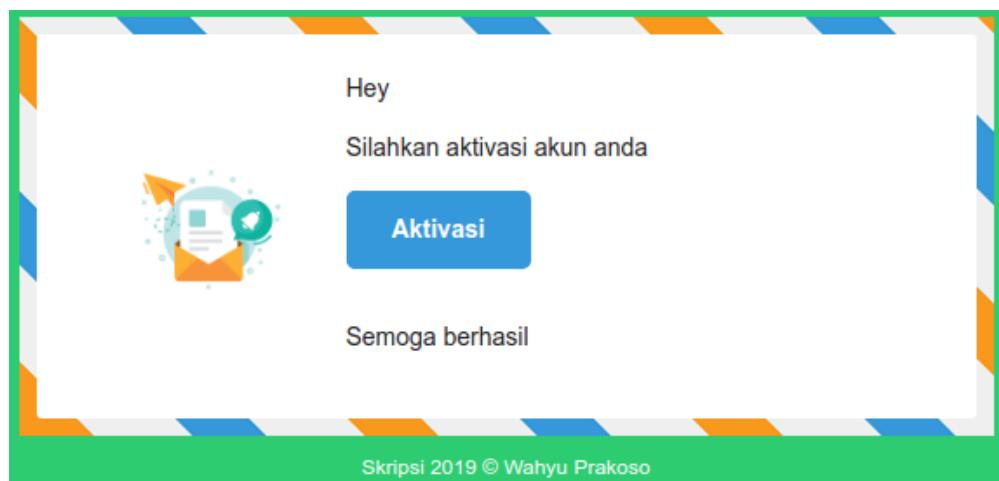
Jika pengguna ingin masuk tetapi lupa kata sandi, pengguna dapat mengatur ulang kata sandi di lupa kata sandi yang diperlihatkan pada gambar 4.14. setelah melengkapi *form* yang ada dan mengirim permintaan atur ulang kata sandi maka sistem akan mengirim *email* ke pengguna, tampilan *email* tersebut seperti gambar 4.34. Pengguna diharapkan untuk memeriksa kotak masuk *email*. Jika sudah mendapatkan *email* atur ulang kata sandi, cukup tekan tombol atur ulang kata sandi yang berwarna biru, setelah tombol tersebut ditekan maka pengguna akan dialihkan ke halaman atur ulang kata sandi yang diperlihatkan pada gambar 4.15.



Gambar 4.34. Hasil Tampilan *Email* Atur Ulang Kata Sandi

x) Hasil Tampilan *Email* Aktivasi Akun

Jika pengguna baru yang sudah mendaftar maka diharapkan memeriksa kotak masuk *email*, pengguna baru akan mendapatkan *email* aktivasi akun dengan tampilan seperti gambar 4.35. untuk mengaktifasi akun cukup dengan menekan tombol aktivasi.



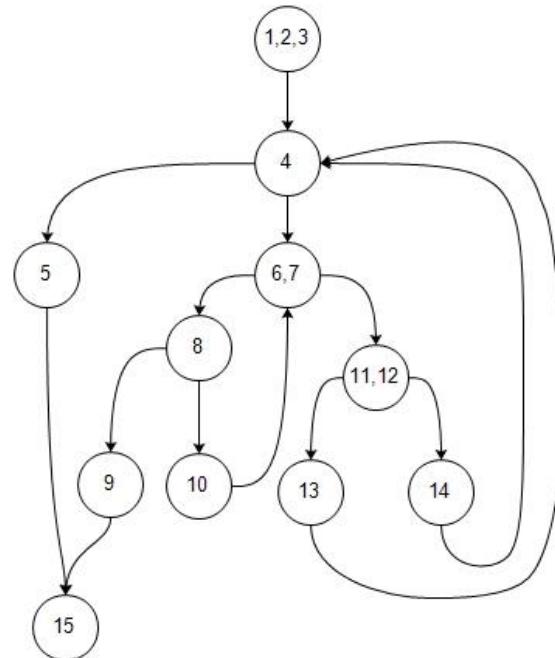
Gambar 4.35. Hasil Tampilan *Email* Aktivasi Akun

4.1.2 Hasil Pengujian

Pengujian sistem terdiri dari pengujian *white box*, pengujian performa algoritma, dan pengujian *black box*.

4.1.2.1 Hasil Pengujian *White Box*

Pengujian *white box* tahap pertama adalah *basic path testing* yaitu membuat *flow graph*. *Flow graph* dibuat berdasarkan *flow chart* algoritma Boyer Moore pada gambar 3.9. berikut tampilan *flow graph* pada gambar 4.36.



Gambar 4.36. *Flow Graph* Algorima Boyer Moore

Dari *flow graph* yang ada pada gambar 4.36 diketahui jumlah *edge* 14 dan *node* 11. Setelah memperoleh jumlah *edge* dan *node* maka bisa dilakukan ke tahapan berikutnya yaitu menghitung nilai *cyclomatic complexity* dengan rumus berikut:

$$V(G) = E - N + 2$$

Keterangan:

$$V(G) = \text{Cylomatic complexity}$$

E = jumlah *edge* pada graf

N = jumlah *node* pada graf

Dari *flow graph* algoritma Boyer Moore dapat dihitung *cyclomatic complexity*:

$$V(G) = 14 - 11 + 2 = 5$$

Jadi *cyclomatic complexity* untuk *flow graph* pada gambar 7.3 adalah 5.

Nilai *cyclomatic complexity* terdapat hubungan dengan resiko dalam suatu prosedur yang dapat dilihat pada tabel 2.1. algoritma Boyer Moore berdasarkan hubungan nilai *cyclomatic complexity* dan resiko termasuk dalam prosedur yang sederhana dan beresiko rendah. Berdasarkan nilai *cyclomatic complexity* maka dapat diketahui jumlah *independent path* yaitu terdiri 5 jalur seperti berikut:

Jalur 1: 1-2-3-4-5-15

Jalur 2: 1-2-3-4-6-7-8-9-15

Jalur 3: 1-2-3-4-6-7-8-10-7-8-9-15

Jalur 4: 1-2-3-4-6-7-11-12-13-4-5-15

Jalur 5: 1-2-3-4-6-7-11-12-14-4-5-15

Tahap selanjutnya yaitu dilakukan *test case* untuk menguji coba sistem sehingga mengetahui apakah hasil yang diharapkan sesuai dengan hasil dari sistem.

a. *Test case* jalur independen 1

Pada *test case* jalur independen 1 yaitu melalui *node* 1-2-3-4-5-15. Skenario yang sesuai dengan jalur independen 1 dapat dilihat pada tabel 4.2. Skenario teks “ABC” dan *pattern* “ABCD” yang diharapkan menhasilkan *output* -1 yang menandakan *pattern* tidak ditemukan pada teks. Hasil skenario *test case* jalur independen 1 dapat dilihat pada tabel 4.3. Jalur independen 1 dimulai dari *node* 1 yaitu terdapat aksi *start*, proses selanjutnya melalui *node* 2 yang terdapat aksi penggunaan *function* BoyerMoore yang memiliki parameter variabel *text* dan *pattern*. Proses selanjutnya melalui *node* 3 yang terdapat aksi deklarasi beberapa variabel yaitu variabel *patternlength* yang di dalamnya terdapat *function* strln untuk menghitung panjang karakter variabel *pattern*, lalu terdapat variabel *textlength* yang di dalamnya terdapat *function* strln untuk menghitung panjang karakter variabel *text*, lalu terdapat variabel *MH* yang berisi *output* dari *function* MHTable dalam bentuk *array*, dan terdapat variabel *i* yang berisi nilai dari variabel *patternlength* dikurangi 1. Proses selanjutnya melalui *node* 4 yang terdapat aksi membandingkan antara variabel *i* terhadap variabel *textlength* yang menghasilkan status *false* sehingga proses selanjutnya melalui *node* 5 yang di dalamnya terdapat aksi mencetak nilai -1. Proses selanjutnya melalui *node* 15 yang terdapat aksi *end*.

Tabel 4.2 Skenario *Test Case* Jalur Independen 1

Skenario	Hasil Yang Diharapkan
Teks: ABC Pattern: ABCD	-1

Tabel 4.3 Hasil Skenario *Test Case* Jalur Independen 1

Node	Aksi	Data
1	<i>Start</i>	
2	<i>Function</i> BoyerMoore(\$text,\$pattern)	<i>Function</i> BoyerMoore("ABC","ABCD")
3	\$patternlength = strlen(\$pattern); \$textlength = strlen(\$text); \$MH = MHTable(\$pattern); \$i = \$patternlength - 1;	\$patternlength = 4; \$textlength = 3; \$MH = ["A" =>3,"B"=>2,"C"=>1, "D"=>0]; \$i = 4 - 1 = 3;
4	\$i < \$textlength	$3 < 3$; <i>Return false</i> ;
5	<i>Return -1</i> ;	-1
15	<i>End</i>	

b. *Test case* jalur independen 2

Pada *test case* jalur independen 2 yaitu melalui *node* 1-2-3-4-6-7-8-9-15.

Skenario yang sesuai dengan jalur independen 2 dapat dilihat pada tabel 4.4.

Skenario teks "A" dan *pattern* "A" yang diharapkan menghasilkan *output* variabel *i* yang menandakan *pattern* ditemukan pada teks. Hasil skenario *test case* jalur independen 2 dapat dilihat pada tabel 4.5. Jalur independen 2 dimulai dari *node* 1 yaitu terdapat aksi *start*, proses selanjutnya melalui *node* 2 yang terdapat aksi

penggunaan *function* BoyerMoore yang memiliki parameter variabel *text* dan *pattern*. Proses selanjutnya melalui *node 3* yang terdapat aksi deklarasi beberapa variabel yaitu variabel patternlength yang di dalamnya terdapat *function* strln untuk menghitung panjang karakter variabel *pattern*, lalu terdapat variabel textlength yang di dalamnya terdapat *function* strln untuk menghitung panjang karakter variabel *text*, lalu terdapat variabel MH yang berisi *output* dari *function* MHTable dalam bentuk *array*, dan terdapat variabel i yang berisi nilai dari variabel patternlength dikurangi 1. Proses selanjutnya melalui *node 4* yang terdapat aksi membandingkan antara variabel i terhadap variabel textlength yang menghasilkan status *true* sehingga proses selanjutnya melalui proses *node 6* yang terdapat aksi deklarasi variabel ii yang memiliki nilai variabel i, terdapat variabel ip yang memiliki nilai variabel patternlength dikurangi 1. Proses selanjutnya melalui *node 7* yang terdapat aksi pencocokan karakter pada variabel *pattern* yang memiliki *index* 0 terhadap variabel *text* yang memiliki *index* 0, pada *node 7* menghasilkan status *true* sehingga proses selanjutnya melalui *node 8* yang terdapat aksi pencocokan variabel ip terhadap nilai 0. Pada *node 8* menghasilkan status *true* sehingga proses selanjutnya melalui *node 9* yang terdapat aksi untuk menyetak nilai variabel i. Proses selanjutnya melalui *node 15* yang terdapat aksi *end*.

Tabel 4.4 Skenario *Test Case* Jalur Independen 2

Skenario	Hasil Yang Diharapkan
Teks: A <i>Pattern</i> : A	\$i

Tabel 4.5 Hasil Skenario *Test Case* Jalur Independen 2

Node	Aksi	Data
1	<i>Start</i>	
2	<i>Function</i> BoyerMoore(\$text,\$pattern)	<i>Function</i> BoyerMoore("A","A")
3	\$patternlength = strlen(\$pattern); \$textlength = strlen(\$text); \$MH = MHTable(\$pattern); \$i = \$patternlength - 1;	\$patternlength = 1; \$textlength = 1; \$MH =[“A”=>0]; \$i = 1 - 1 = 0;
4	\$i < \$textlength	$0 < 1$ <i>Return true;</i>
6	\$ii = \$i; \$ip = \$patternlength - 1;	\$ii = 0; \$ip = 1 - 1 = 0;
7	\$pattern[\$ip] == \$text[\$i]	\$pattern[0] == \$text[0] <i>Return true;</i>
8	\$ip == 0	<i>True</i>
9	<i>Return \$i;</i>	\$i
15	<i>End</i>	

c. *Test case* jalur independen 3

Pada *test case* jalur independen 3 yaitu melalui *node* 1-2-3-4-6-7-8-10-7-8-9-15.

Skenario yang sesuai dengan jalur independen 3 dapat dilihat pada tabel 4.6.

Skenario teks “AA” dan *pattern* “AA” yang diharapkan menghasilkan *output* variabel i yang menandakan *pattern* ditemukan pada teks. Hasil skenario *test case* jalur independen 3 dapat dilihat pada tabel 4.7. Jalur independen 3 dimulai dari *node* 1 yaitu terdapat aksi *start*, proses selanjutnya melalui *node* 2 yang terdapat aksi penggunaan *function* BoyerMoore yang memiliki parameter variabel *text* dan *pattern*. Proses selanjutnya melalui *node* 3 yang terdapat aksi deklarasi beberapa variabel yaitu variabel *patternlength* yang di dalamnya terdapat *function* *strlen* untuk

menghitung panjang karakter variabel *pattern*, lalu terdapat variabel *textlength* yang di dalamnya terdapat *function* *strlen* untuk menghitung panjang karakter variabel *text*, lalu terdapat variabel *MH* yang berisi *output* dari *function* *MHTable* dalam bentuk *array*, dan terdapat variabel *i* yang berisi nilai dari variabel *patternlength* dikurangi 1. Proses selanjutnya melalui *node 4* yang terdapat aksi membandingkan antara variabel *i* terhadap variabel *textlength* yang menghasilkan status *true* sehingga proses selanjutnya melalui proses *node 6* yang terdapat aksi deklarasi variabel *ii* yang memiliki nilai variabel *i*, terdapat variabel *ip* yang memiliki nilai variabel *patternlength* dikurangi 1. Proses selanjutnya melalui *node 7* yang terdapat aksi pencocokan karakter pada variabel *pattern* yang memiliki *index* 1 terhadap variabel *text* yang memiliki *index* 1, pada *node 7* menghasilkan status *true* sehingga proses selanjutnya melalui *node 8* yang terdapat aksi pencocokan variabel *ip* terhadap nilai 0 yang menghasilkan status *false* sehingga proses selanjutnya melalui *node 10* yang terdapat aksi mengurangi nilai variabel *ip* dikurangi 1 dan variabel *i* dikurangi 1. proses selanjutnya kembali ke *node 7* untuk mencocokan karakter pada variabel *pattern* yang memiliki *index* 0 terhadap variabel *text* yang memiliki *index* 0. Pada *node 7* menghasilkan status *true* sehingga proses selanjutnya melalui *node 8* yang terdapat aksi pencocokan variabel *ip* terhadap nilai 0. Pada *node 8* menghasilkan status *true* sehingga proses selanjutnya melalui *node 9* yang terdapat aksi untuk menyetak nilai variabel *i*. Proses selanjutnya melalui *node 15* yang terdapat aksi *end*.

Tabel 4.6 Skenario *Test Case* Jalur Independen 3

Skenario	Hasil Yang Diharapkan
Teks: AA Pattern: AA	\$i

Tabel 4.7 Hasil Skenario *Test Case* Jalur Independen 3

Node	Aksi	Data
1	<i>Start</i>	
2	<i>Function</i> BoyerMoore(\$text,\$pattern)	<i>Function</i> BoyerMoore("AA","AA")
3	\$patternlength = strlen(\$pattern); \$textlength = strlen(\$text); \$MH = MHTable(\$pattern); \$i = \$patternlength - 1;	\$patternlength = 2; \$textlength = 2; \$MH =["A">>0]; \$i = 2 - 1 = 1;
4	\$i < \$textlength	1 < 2 <i>Return true;</i>
6	\$ii = \$i; \$ip = \$patternlength - 1;	\$ii = 1; \$ip = 2 - 1 = 1;
7	\$pattern[\$ip] == \$text[\$i]	\$pattern[1] == \$text[1] <i>Return true;</i>
8	\$ip == 0	<i>False</i>
10	\$ip--,\$i--	\$ip = 1-1=0,\$i = 1-1=0
7	\$pattern[\$ip] == \$text[\$i]	\$pattern[0] == \$text[0] <i>Return true;</i>
8	\$ip == 0	<i>True</i>
9	<i>Return \$i;</i>	\$i
15	<i>End</i>	

d. *Test case jalur independen 4*

Pada *test case* jalur independen 4 yaitu melalui *node* 1-2-3-4-6-7-11-12-13-4-5-15. Skenario yang sesuai dengan jalur independen 4 dapat dilihat pada tabel 4.8. Skenario teks “ABCD” dan *pattern* “BCDE” yang diharapkan menghasilkan *output* variabel -1 yang menandakan *pattern* tidak ditemukan pada teks. Hasil skenario *test case* jalur independen 2 dapat dilihat pada tabel 4.9. Jalur independen 3 dimulai dari *node* 1 yaitu terdapat aksi *start*, proses selanjutnya melalui *node* 2 yang terdapat aksi penggunaan *function* BoyerMoore yang memiliki parameter variabel *text* dan *pattern*. Proses selanjutnya melalui *node* 3 yang terdapat aksi deklarasi beberapa variabel yaitu variabel *patternlength* yang di dalamnya terdapat *function* strln untuk menghitung panjang karakter variabel *pattern*, lalu terdapat variabel *textlength* yang di dalamnya terdapat *function* strln untuk menghitung panjang karakter variabel *text*, lalu terdapat variabel MH yang berisi *output* dari *function* MHTable dalam bentuk *array*, dan terdapat variabel i yang berisi nilai dari variabel *patternlength* dikurangi 1. Proses selanjutnya melalui *node* 4 yang terdapat aksi membandingkan antara variabel i terhadap variabel *textlength* yang menghasilkan status *true* sehingga proses selanjutnya melalui proses *node* 6 yang terdapat aksi deklarasi variabel ii yang memiliki nilai variabel i, terdapat variabel ip yang memiliki nilai variabel *patternlength* dikurangi 1. Proses selanjutnya melalui *node* 7 yang terdapat aksi pencocokan karakter pada variabel *pattern* yang memiliki *index* 3 terhadap variabel *text* yang memiliki *index* 3, pada node 7 menghasilkan status *false* sehingga proses selanjutnya melalui *node* 11 yang terdapat aksi deklarasi variabel i mempunyai nilai variabel ii. Proses selanjutnya melalui *node* 12

yang terdapat aksi mencocokan karakter *text* yang memiliki *index* 3 terhadap *array key* variabel MH. Pada *node* 12 dihasilkan status *true* sehingga proses selanjutnya melalui *node* 13 yang terdapat aksi variabel i ditambah nilai dari nilai tertinggi antara nilai dari *array key* MHTable yang cocok terhadap karakter *text* yang memiliki *index* 3 terhadap 1. Proses selanjutnya melalui *node* 4 yang terdapat aksi membandingkan antara variabel i terhadap variabel textlength yang menghasilkan status *false* sehingga proses selanjutnya melalui *node* 5 yang di dalamnya terdapat aksi mencetak nilai -1. Proses selanjutnya melalui *node* 15 yang terdapat aksi *end*.

Tabel 4.8 Skenario *Test Case* Jalur Independen 4

Skenario	Hasil Yang Diharapkan
Teks: ABCD Pattern: BCDE	-1

Tabel 4.9 Hasil Skenario *Test Case* Jalur Independen 4

Node	Aksi	Data
1	<i>Start</i>	
2	<i>Function</i> BoyerMoore(\$text,\$pattern)	<i>Function</i> BoyerMoore("ABCD","BCDE")
3	\$patternlength = strlen(\$pattern); \$textlength = strlen(\$text); \$MH = MHTable(\$pattern); \$i = \$patternlength - 1;	\$patternlength = 4; \$textlength = 4; \$MH = ["B"=>3,"C"=>2,"D"=>1, "E"=>0]; \$i = 4 - 1 = 3;
4	\$i < \$textlength	3 < 4 <i>Return true</i> ;
6	\$ii = \$i; \$ip = \$patternlength - 1;	\$ii = 3; \$ip = 4 - 1 = 3;
7	\$pattern[\$ip] == \$text[\$i]	\$pattern[3] == \$text[3] <i>Return false</i> ;

Node	Aksi	Data
11	$\$i = \$ii;$	$\$i = 3;$
12	array_key_exists(\$text[\$i], \$MH)	<i>True</i>
13	$\$i = \$i + \max(\$MH[\$text[\$i]], 1)$	$\$i = 3 + \max(1, 1) = 3 + 1 = 4;$
4	$\$i < \$textlength$	$4 < 4$ <i>Return false;</i>
5	<i>Return -1;</i>	-1
15	<i>End</i>	

e. *Test case* jalur independen 5

Pada *test case* jalur independen 5 yaitu melalui *node* 1-2-3-4-6-7-11-12-14-4-5-15. Skenario yang sesuai dengan jalur independen 5 dapat dilihat pada tabel 4.10. Skenario teks “ABCE” dan pattern “ABCD” yang diharapkan menghasilkan *output* variabel -1 yang menandakan *pattern* tidak ditemukan pada teks. Hasil skenario *test case* jalur independen 2 dapat dilihat pada tabel 4.11. Jalur independen 3 dimulai dari *node* 1 yaitu terdapat aksi *start*, proses selanjutnya melalui *node* 2 yang terdapat aksi penggunaan *function* BoyerMoore yang memiliki parameter variabel *text* dan *pattern*. Proses selanjutnya melalui *node* 3 yang terdapat aksi deklarasi beberapa variabel yaitu variabel *patternlength* yang di dalamnya terdapat *function* strln untuk menghitung panjang karakter variabel *pattern*, lalu terdapat variabel *textlength* yang di dalamnya terdapat *function* strln untuk menghitung panjang karakter variabel *text*, lalu terdapat variabel *MH* yang berisi *output* dari *function* MHTable dalam bentuk *array*, dan terdapat variabel *i* yang berisi nilai dari variabel *patternlength* dikurangi 1. Proses selanjutnya melalui *node* 4 yang terdapat aksi membandingkan antara variabel *i* terhadap variabel *textlength* yang menghasilkan

status *true* sehingga proses selanjutnya melalui proses *node* 6 yang terdapat aksi deklarasi variabel *ii* yang memiliki nilai variabel *i*, terdapat variabel *ip* yang memiliki nilai variabel *patternlength* dikurangi 1. Proses selanjutnya melalui *node* 7 yang terdapat aksi pencocokan karakter pada variabel *pattern* yang memiliki *index* 3 terhadap variabel *text* yang memiliki *index* 3, pada *node* 7 menghasilkan status *false* sehingga proses selanjutnya melalui *node* 11 yang terdapat aksi deklarasi variabel *i* mempunyai nilai variabel *ii*. Proses selanjutnya melalui *node* 12 yang terdapat aksi mencocokan karakter *text* yang memiliki *index* 3 terhadap *array key* variabel *MH*. Pada *node* 12 dihasilkan status *false* sehingga proses selanjutnya melalui *node* 14 yang terdapat aksi variabel *i* ditambah variabel *patternlength*. Proses selanjutnya melalui *node* 4 yang terdapat aksi membandingkan antara variabel *i* terhadap variabel *textlength* yang menghasilkan status *false* sehingga proses selanjutnya melalui *node* 5 yang di dalamnya terdapat aksi mencetak nilai -

1. Proses selanjutnya melalui *node* 15 yang terdapat aksi *end*.

Tabel 4.10 Skenario *Test Case* Jalur Independen 5

Skenario	Hasil Yang Diharapkan
Teks: ABCE Pattern: ABCD	-1

Tabel 4.11 Hasil Skenario Test Case Jalur Independen 5

Node	Aksi	Data
1	<i>Start</i>	
2	<i>Function</i> BoyerMoore(\$text,\$pattern)	<i>Function</i> BoyerMoore("ABCE","ABCD")
3	\$patternlength = strlen(\$pattern); \$textlength = strlen(\$text); \$MH = MHTable(\$pattern); \$i = \$patternlength - 1;	\$patternlength = 4; \$textlength = 4; \$MH = ["A"=>3,"B"=>2,"C"=>1, "D"=>0]; \$i = 4 - 1 = 3;
4	\$i < \$textlength	$3 < 4$ <i>Return true;</i>
6	\$ii = \$i; \$ip = \$patternlength - 1;	\$ii = 3; \$ip = 4 - 1 = 3;
7	\$pattern[\$ip] == \$text[\$i]	\$pattern[3] == \$text[3] <i>Return false;</i>
11	\$i = \$ii;	\$i = 3;
12	array_key_exists(\$text[\$i],\$MH)	False
14	\$i += \$patternlength	\$i = \$i + \$patternlength = 3 + 3 = 6
4	\$i < \$textlength	$6 < 4$ <i>Return false;</i>
5	<i>Return -1;</i>	-1
15	<i>End</i>	

4.1.2.2 Hasil Pengujian Performa Algoritma

Pengujian performa algoritma Boyer Moore dilakukan terhadap di setiap parameter. Hasil pengujian performa algoritma berdasarkan parameter subyek dapat dilihat di tabel 4.12. Pengujian menggunakan *pattern adidas*, untuk data teks menggunakan subyek *email*. Pengujian terhadap parameter subyek dilakukan sebanyak 90 kali.

Tabel 4.12 Pengujian Performa Algoritma Penyaringan *Email* Berdasarkan Subyek

Pattern	Teks	Waktu
adidas	11 Ideas That Will Rewire Your Brain Tim Denning	28,85 µs
	Notifikasi keamanan penting	15,97 µs
	Wahyu, New Inspiration For You: October 16, 2019	15,97 µs
	Jangan lewatkan pendaftaran Konferensi AWSome Day Online Daftar Sekarang	15,02 µs
	Wahyu, New Inspiration For You: October 23, 2019	14,07 µs
	Wahyu, New Inspiration For You: October 30, 2019	13,11 µs
	Dua Hari lagi! AWSome Day Online Conference	14,07 µs
	Wahyu, New Inspiration For You: November 6, 2019	12,87 µs
	Wahyu, New Inspiration For You: November 13, 2019	14,07 µs
	=?utf-8?Q?Important=20Updates=20to=20our=20Terms=20off=20Service=20and=20Telemetry=20Services?=	15,02 µs
	=?utf-8?Q?We=20apologize=20and=20have=20rolled=20back=20the=20changes=20announced=20in=20our=20October=202023=20email=C2=A0?=	14,07 µs
	The NPR Student Podcast Challenge is back	12,87 µs
	[GoRails] Testing Active Job	12,87 µs
	Berhasil berlangganan newsletter	14,07 µs
	Perubahan pada Persyaratan Layanan YouTube	13,11 µs
	The mystery (and fear) surrounding office hours	13,11 µs
	How This 26-Year-Old Writer Generates \$1.94 Million a Year Dave Schools in Entrepreneur's Handbook	15,02 µs
	Are the nation's top public universities affordable?	14,07 µs
	Memory-Optimized Droplets are Generally Available	13,11 µs

Pattern	Teks	Waktu
adidas	The Most Underrated Productivity Technique Is Also the Simplest Darius Foroux in Forge	15,97 µs
	=?UTF- 8?B?Q291cnRzaWRlOiAiWW91ciB1bHRpbWF0ZSBndWlkZSB0byBkZXNpZ25pbmcgYSBzdGFuZC1vdXQgVVggcmVzdW1lIClIgYW5k?= more...	15,02 µs
	Last Week's Hot Shots	12,87 µs
	Delivery Status Notification (Failure)	13,11 µs
	YMCA of Greater New York, RapidAPI, and others are hiring	14,07 µs
	=?UTF- 8?B?Q291cnRzaWRlOiAiNSBkZXNpZ24gcHJpbmNpcGxlcyBmb3IgYnVpbGRpbmcgeW91ciBmaXJzdCBwb3J0Zm9saW8gIiBhbmQg?= more...	15,97 µs
	Popular last week on Dribbble	13,83 µs
	Ampush, Clevertech, and others are hiring	13,11 µs
	=?UTF- 8?B?Q291cnRzaWRlOiAiNyBVSSB0b29scyBmb3IgY3JIYXRpbmcgYmV0dGVyIGRpZ2l0YWwgY29sb3IgcGFsZXR0ZXMGliBhbmQg?= more...	15,97 µs
	Popular last week on Dribbble	12,87 µs
	Over the Top, rideOS, and others are hiring	13,11 µs
	=?UTF- 8?B?Q291cnRzaWRlOiAiVGhlIEV2b2x1dGlvbIBvZiBXZWlIgRGVzaWduIHwgUGFydCAxOiBUaGUgMTk5MHMgIiBhbmQgbW9yZeKApge==?=	15,97 µs
	9522 kode Konfirmasi Aktivasi DetikID Kamu	15,02 µs
	Sukses Aktivasi DetikID	14,07 µs
	Akun detikID Login di device baru	13,83 µs

Pattern	Teks	Waktu
adidas	Last Week's Hot Shots	11,92 µs
	Aktivasi Akun Tokopedia	21,93 µs
	Aktivasi Akun Tokopedia	14,07 µs
	Bleacher Report, Hopper, and others are hiring	14,07 µs
	=?UTF- 8?B?Q291cnRzaWRlOiAiVGhlIHVsdlGltYXRIIGd1a WRlIHRvIGHpcmluZyB3aXRoIGRlc2lnbiBleGVyY2lz ZXMGiBhbmQgbW9y?= e...	15,97 µs
	Stop Choosing to Be Less Than You Are Brianna Wiest in Human Parts	15,02 µs
	cynthialvina, fakhrulnur_ and mahar_raniii have new posts	15,02 µs
	Konfirmasikan Pendaftaran Email Anda Segera!	15,02 µs
	How To Walk Into Any Room With Confidence and Poise Anthony Moore in The Post-Grad Survival Guide	15,97 µs
	Notifikasi keamanan penting	14,07 µs
	=?UTF- 8?B?RG9u4oCZdCBUZWFjaCBZb3VyIEtpZCB0byB Db2RlLiBUZWFjaCBuAGVtIHRvIENvbW11bmljYX RlLiB8IEdyZWcgU2F0ZWxs?= fakhrulnur_, febianiys_ and cetakpolaroid_semarang have new posts	13,83 µs
	My Top 4 Patterns for Writing Simple Code Dan Goslen	12,87 µs
	Re: wibu	12,87 µs
	Re: wibu	11,92 µs
	UNGGUL DALAM PERFORMA DENGAN ALPHAEDGE 4D	20,98 µs
	adidas Originals Ozweego by King Push	13,11 µs
	Nikmati Diskon 10% di Order Anda Selanjutnya.	18,12 µs

Pattern	Teks	Waktu
adidas	ZX TORSION dan TORSION X telah dirilis	16,93 µs
	JavaScript Promises: Zero To Hero Plus Cheat Sheet Joshua Saunders in DailyJS	15,97 µs
	Notifikasi keamanan penting	15,97 µs
	Here Are 6 Front-End Challenges to Code Indrek Lasn in Better Programming	15,97 µs
	=?UTF- 8?B?SG93IFRvIEhhbmRsZSBPdGhlciBQZW9wbGXigJlzIEJhZCBNb29kcyBMaWtlIGEgUHJvIHwgTmljayBXaWduYWxsIGluIEJldHQ=?= er Marketing	18,12 µs
	=?utf- 8?Q?=F0=9F=98=8E=C2=A0=20Ada=20Promo=20baru=20loh=20=2D=20Diskon=20Hosting=20sampe=2090%.=20Yakin=20g=20mau=20=3F?=	17,17 µs
	=?utf- 8?Q?=F0=9F=94=A5=C2=A0FLASH=20SALE=20Diskon=20=C2=A090%=20Hosting=20diadakan=20BESOK=21=20Jangan=20Ketinggalan.?=	16,93 µs
	=?utf- 8?Q?=F0=9F=94=A5=C2=A0Hosting=20Diskon=2090%.=20=2D=20Sikat=20Sekarang=21?=	15,02 µs
	=?utf- 8?Q?=E2=9C=A8=C2=A0Mari=20Wujudkan=20Perjuangan=20Kedaulatan=20Data=20bareng=20IDCloudhost?=	15,97 µs
	=?utf- 8?Q?=F0=9F=94=A5=C2=A0FLASH=20SALE=20Diskon=2090%=20Hosting=20Hadir=20lagi=21=20Ikutan=20Yuk=20Jam=202=20Siang=20ini=21?=	15,97 µs

Pattern	Teks	Waktu
adidas	=?utf- 8?Q?=F0=9F=98=8D=C2=A0Mau=20Saldo=20Rp=20 25=20rb=20GRATIS=20=3F=20Yuk=20Ikutan=20Pro monya=21?=	15,97 µs
	=?utf- 8?Q?=F0=9F=94=A5=C2=A0G=20Dapat=20Jatah=20 FLASH=20SALE=2090%=20kemarin=20=3F=20Hari= 20ini=20Terakhir=2C=20Ikut=20Yuk=21?=	18,12 µs
	It's a smartphone life	23,13 µs
	Notifikasi keamanan penting	15,02 µs
	Notifikasi keamanan penting	15,02 µs
	How to Become the Best in the World at Something Tomas Pueyo in Forge	15,02 µs
	Notifikasi keamanan penting	14,07 µs
	Did Google Duplex just pass the Turing Test? Lance Ulanoff	15,02 µs
	11.11 Super Sale: 40% off Creative Cloud	13,83 µs
	Ending soon: 11.11 Super Sale: 40% off Creative Cloud	14,07 µs
	Ending soon: 11.11 Super Sale: 40% off Creative Cloud	15,02 µs
	Don't miss 11.11 Special: Get 40% off Creative Cloud	15,02 µs
	This Is How to Age with Elegance John P. Weiss in Personal Growth	15,97 µs
	3 Foods You Should Throw Away Forever. Tim Rees in Noteworthy - The Journal Blog	14,78 µs
	Six Habits of Generous, Successful People Michael Thompson in Forge	15,02 µs
	How to Find Hidden Cameras in Your Airbnb, and Anywhere Else Fast Company	15,02 µs
	The Trouble With Johnny Depp RollingStone	13,83 µs

Pattern	Teks	Waktu
adidas	Notifikasi keamanan penting	15,97 µs
	6 Ways to Be More Likable by Saying Very Little Michael Thompson in P.S. I Love You	15,97 µs
	6 Mistakes People Make On A First Date Kirstie Taylor in P.S. I Love You	15,02 µs
	Stop Using i++ in Your Loops Devin Soni in Better Programming	13,83 µs
	The Real Cost of Phone Addiction Catherine Price in GEN	15,02 µs
	A Radical Guide to Spending Less Time on Your Phone Ryan Holiday in Forge	15,97 µs
	=?UTF- 8?B?SGVyZeKAmXMgYSBTaW1wbGUgV2F5IHRvIEhvbgQgWW91ciBUZWFTIEFjY291bnRhYmxlIHwgRGF2ZSBCYWlsZXkgaW4gTWFya2U=?= r	15,97 µs
	Do You Make This Mistake In Conversations? John P. Weiss in Personal Growth	15,02 µs
	[GoRails] Continuous Integration (CI) for Rails with GitHub Actions	15,02 µs
	How to Delegate Work So It Actually Gets Done Dave Bailey in Marker	15,02 µs
	=?UTF- 8?B?SW5jb2duaXRvIE1vZGUgV29u4oCZdCBLZWVwIFldXIgQnJvd3NpbmcgUHJpdmF0ZS4gRG8gVGhpcyBJbnN0ZWFrLiB8IEZhc3Q=?= Company	15,97 µs
Total Waktu		1,37 ms

Hasil pengujian performa algoritma berdasarkan parameter *email* dapat dilihat di tabel 4.13. Pengujian terhadap parameter *email* dilakukan sebanyak 20 kali.

Tabel 4.13 Pengujian Performa Algoritma Penyaringan *Email* Berdasarkan *Email*

Pattern	Teks	Waktu
noreply@medium.com		139,95 μ s
no-reply@accounts.google.com		35,05 μ s
noreply-behance@behance.com		30,04 μ s
aws-apac-marketing@amazon.com		25,99 μ s
info@gitlab.com		26,94 μ s
email@nl.npr.org		24,08 μ s
chris@gorails.com		28,13 μ s
noreply123@id-adidas.com		25,99 μ s
no-reply@youtube.com		25,03 μ s
product@info.digitalocean.com	no-reply@dribbble.com	23,84 μ s
no-reply@dribbble.com	news@sea-adidas.com	22,89 μ s
mailer-daemon@googlemail.com	noreply@medium.com	25,03 μ s
no-reply@detik.com		25,03 μ s
no-reply@mail-		24,8 μ s
accounts.tokopedia.com		
no-reply@mail.instagram.com		24,08 μ s
edm.subscribe@mail.danamon.co.id		24,08 μ s
arminharjianton@gmail.com		21,93 μ s
news@sea-adidas.com		23,84 μ s
care@idcloudhost.com		23,84 μ s
demand@mail.adobe.com		23,84 μ s
Total Waktu		625,37 μs

Hasil pengujian performa algoritma berdasarkan parameter *domain* dapat dilihat di tabel 4.14. Pengujian terhadap parameter *domain* dilakukan sebanyak 15 kali.

Tabel 4.14 Pengujian Performa Algoritma Penyaringan *Email* Berdasarkan *Domain*

Pattern	Teks	Waktu
medium.com	amazon.com idcloudhost.com @dribbble.com	123,98 µs
accounts.google.com		28,13 µs
behance.com		41,01 µs
amazon.com		25,03 µs
gitlab.com		42,2 µs
nl.npr.org		23,13 µs
gorails.com		24,8 µs
id-adidas.com		25,99 µs
youtube.com		24,08 µs
info.digitalocean.com		21,93 µs
dribbble.com		23,13 µs
googlemail.com		23,13 µs
detik.com		24,08 µs
mail-accounts.tokopedia.com		24,08 µs
mail.instagram.com		24,08 µs
Total Waktu		558,14 µs

Hasil pengujian performa algoritma berdasarkan parameter TLD dapat dilihat di tabel 4.15. Pengujian terhadap parameter TLD dilakukan sebanyak 3 kali.

Tabel 4.15 Pengujian Performa Algoritma Penyaringan *Email* Berdasarkan TLD

Pattern	Teks	Waktu
.com	.com	133,04 μ s
.org		29,8 μ s
.co.id		20,98 μ s
Total Waktu		195,98 μs

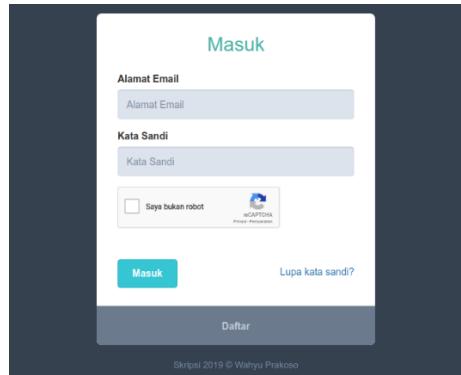
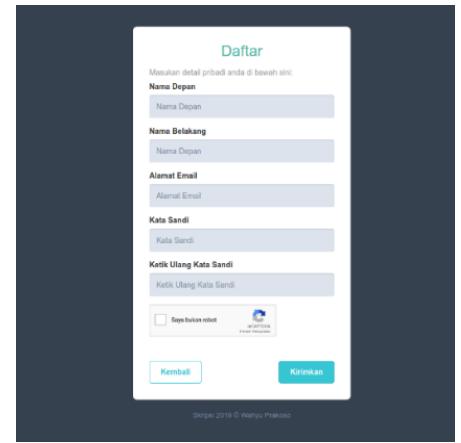
Hasil pengujian performa algoritma Boyer Moore terhadap parameter subyek medapat total 1,37 ms dengan rata-rata 15,17 μ s. Pada parameter *email* mendapat total 625,37 μ s dengan rata-rata 31,60. Pada parameter *domain* mendapat total 558,14 μ s dengan rata-rata 33,52. Pada parameter TLD mendapat total 195,98 μ s dengan rata-rata 61,27. Total keseluruhan semua parameter 2,74 ms dengan rata-rata 20,85 μ s.

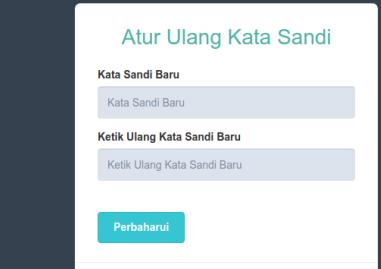
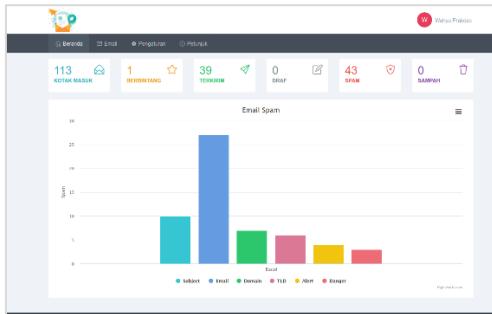
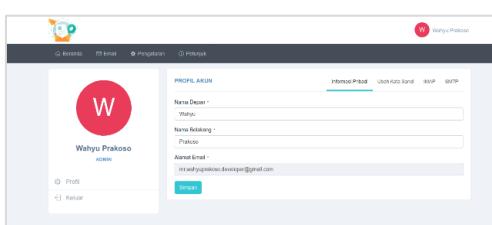
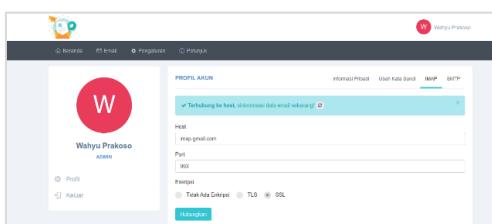
4.1.2.3 Hasil Pengujian *Black Box*

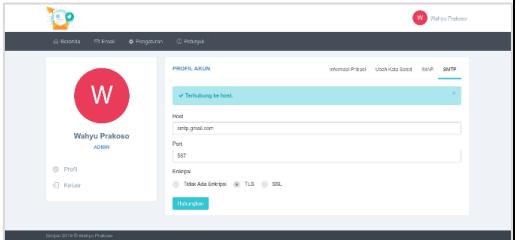
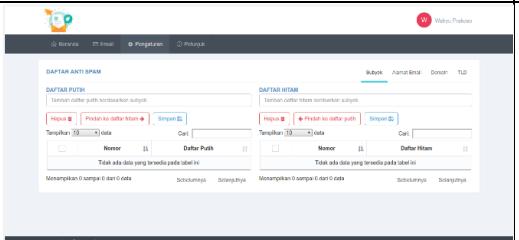
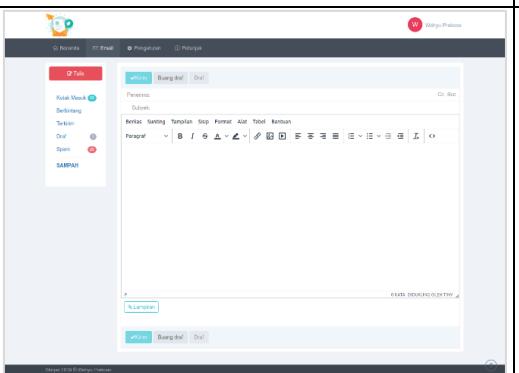
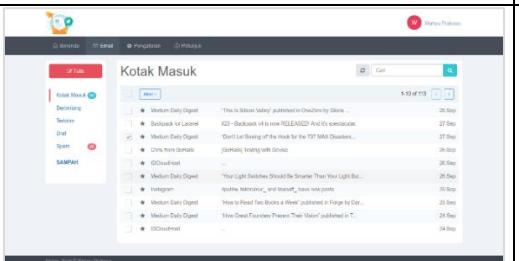
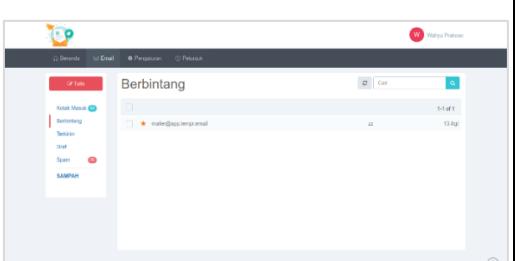
a) Aspek *Functionality*

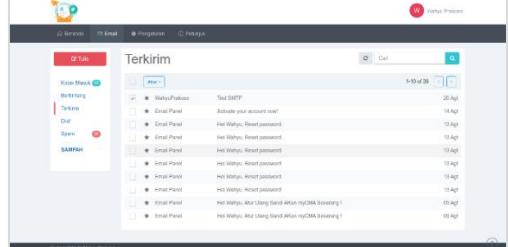
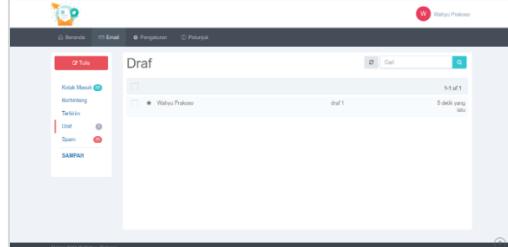
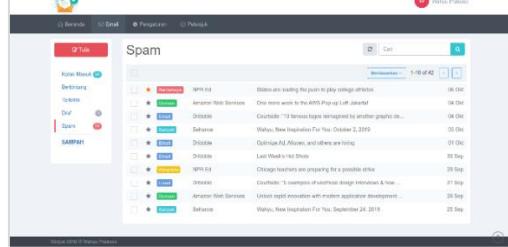
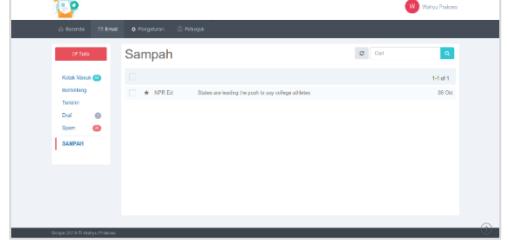
Pengujian sistem dilakukan sesuai standar ISO 9126 pada aspek *functionality, portability, efficiency, dan usability*. Fungsi-fungsi yang ada pada *controller* sistem penyaringan *email* dapat dilihat pada tabel 4.16.

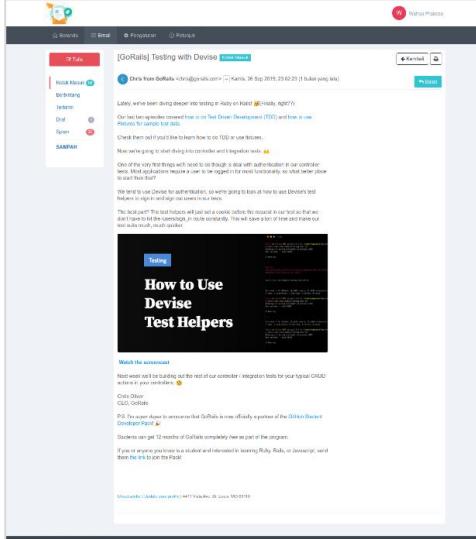
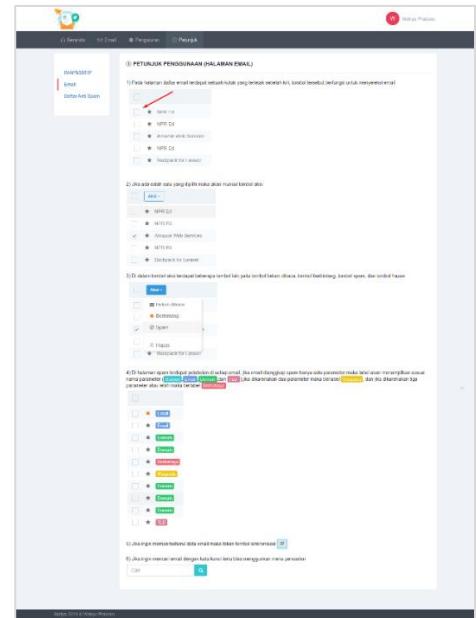
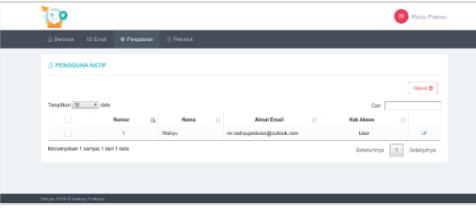
Tabel 4.16 Pengujian Aspek *Functionality*.

Fungsi	Gambar	Status
getSignInPage		Aktif
getRequestResetPassword		Aktif
getSignUpPage		Aktif

Fungsi	Gambar	Status
getResetPassword		Aktif
getDashboard		Aktif
getProfile		Aktif
getProfile		Aktif

Fungsi	Gambar	Status
getProfile		Aktif
getFilterEmail		Aktif
getComposePage		Aktif
getInboxPage		Aktif
getStarredPage		Aktif

Fungsi	Gambar	Status
getSentPage		Aktif
getDraftPage		Aktif
getSpamPage		Aktif
getTrashPage		Aktif

Fungsi	Gambar	Status
getViewInboxPage		Aktif
guidePageEmail		Aktif
getAdminActiveUsers		Aktif

Fungsi	Gambar	Status
getAdminNewUsers		Aktif
getEditAdminActiveUsers		Aktif
sendEmailRequestActivation		Aktif
sendEmailRequestResetPassword		Aktif

Hasil dari pengujian aspek *functionality* tersebut selanjutnya dilakukan analisis menggunakan perhitungan berikut:

$$\text{Persentase } \textit{functionality} = \frac{\Sigma x}{n} \times 100\%$$

$$= \frac{23}{23} \times 100\%$$

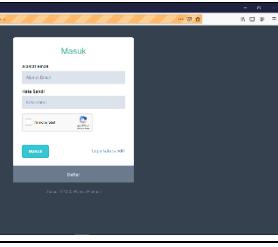
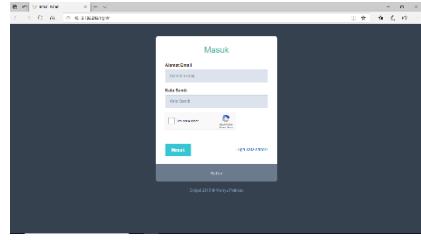
$$= 100\%$$

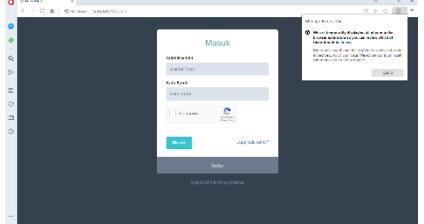
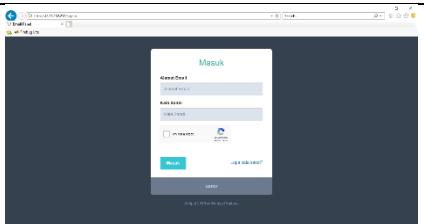
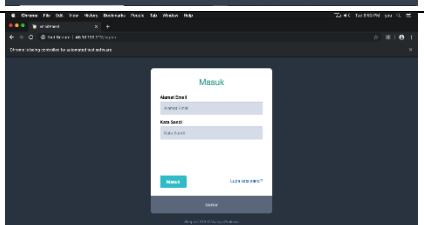
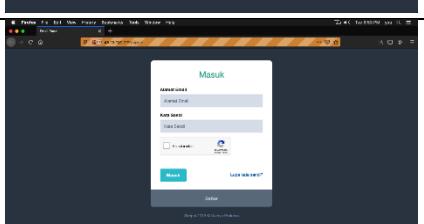
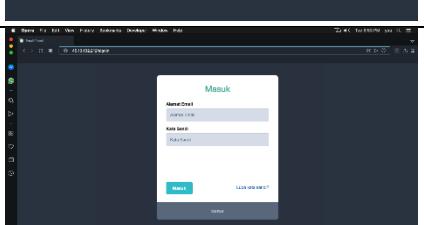
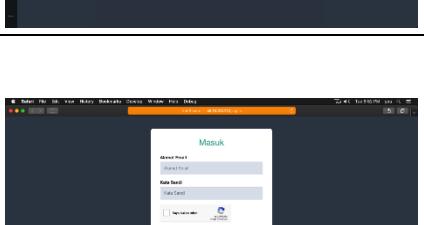
Berdasarkan hasil pengujian *black box*, mendapatkan hasil bahwa pengujian aspek *functionality* dari sistem penyaringan *email* berhasil 100%. Artinya fungsionalitas dan proses yang terdapat pada sistem telah berjalan dengan baik sesuai dengan perancangan sistem yang telah dibuat.

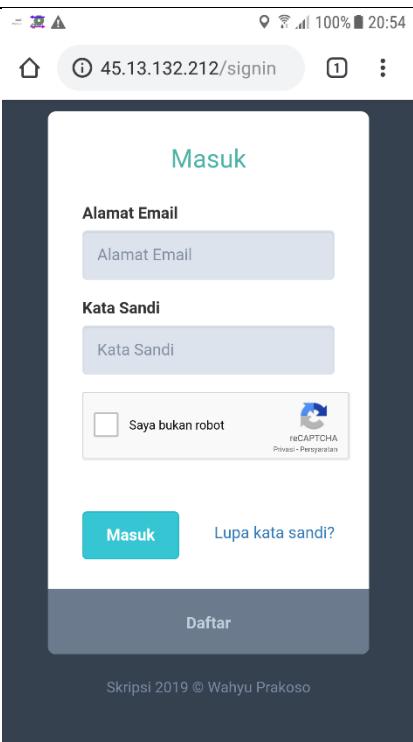
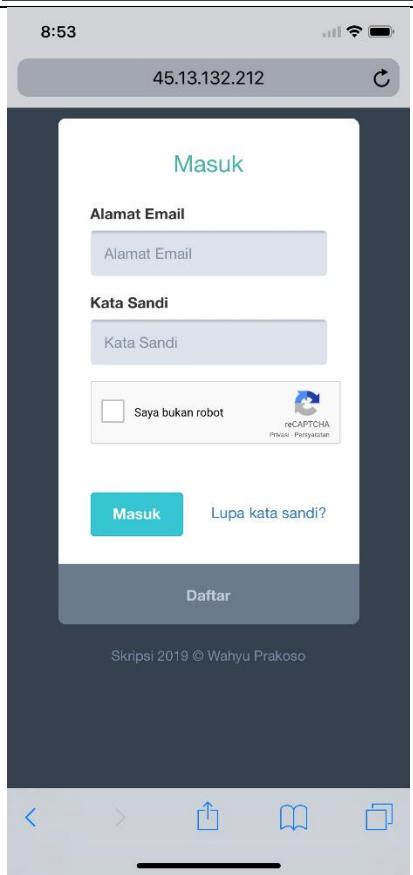
b) Aspek *Portability*

Pada pengujian *black box* aspek *portability* menggunakan alat yaitu browsershots.org. Pada pengujian aspek *portability* dilakukan 11 pengujian melalui beberapa sistem operasi, *browser* dan resolusi yang berbeda. Hasil pengujian dapat dilihat pada tabel 4.17.

Tabel 4.17 Pengujian Apsek *Portability*

Pengujian	Sistem Operasi	Browser	Resolusi Layar	Hasil
1	Windows 10	Chrome versi 78 64 bit	1366x768	
2	Windows 10	Mozilla firefox versi 70	1366x768	
3	Windows 10	Edge versi 18	1366x768	

Pengujian	Sistem Operasi	Browser	Resolusi Layar	Hasil
4	Windows 10	Opera versi 63	1366x768	
5	Windows 10	Internet explorer	1366x768	
6	Mac versi 10.14	Chrome versi 78 64 bit	1366x768	
7	Mac versi 10.14	Mozilla firefox versi 70	1366x768	
8	Mac versi 10.14	Opera versi 60	1366x768	
9	Mac versi 10.14	Safari versi 12	1366x768	

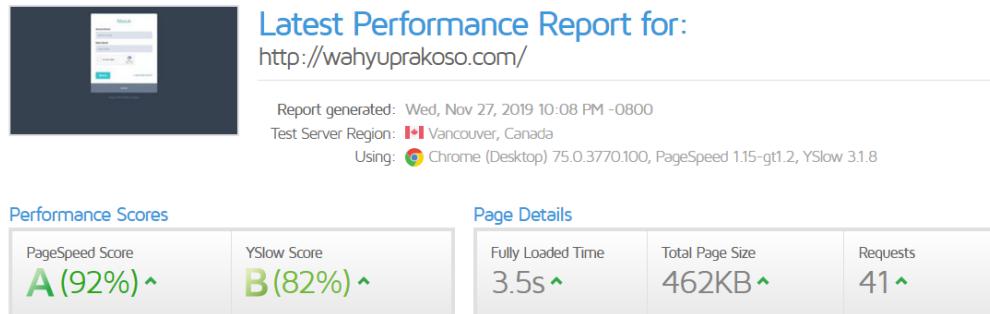
Pengujian	Sistem Operasi	Browser	Resolusi Layar	Hasil
10	Android	Chrome versi 73	1080x1920	
11	Ios	Safari versi 11	1125x2436	

Hasil pengujian aspek *portability* berdasarkan tabel 4.17 sistem dapat berjalan di semua *browser* dan *device*, sehingga skor persentase hasil pengujian adalah:

$$\begin{aligned}\text{Persentase } portability &= \frac{\text{Jumlah browser yang sesuai}}{\text{Jumlah browser yang diujicoba}} \times 100\% \\ &= \frac{11}{11} \times 100\% \\ &= 100\%\end{aligned}$$

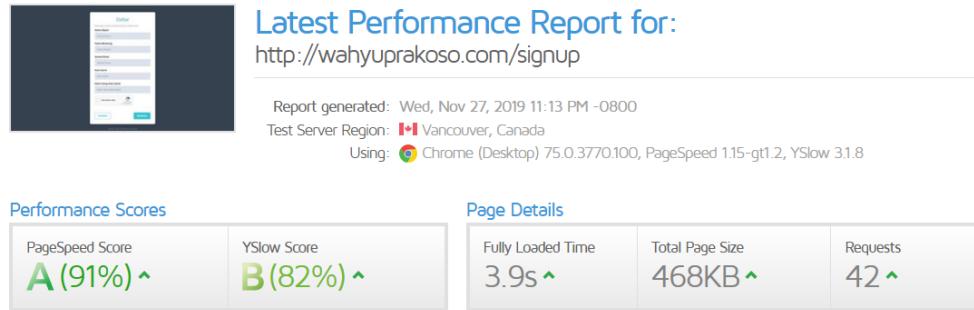
c) Aspek *Efficiency*

Pengujian Aspek *Efficiency* menggunakan alat berbasis *website* yaitu GTMetrix. Pengujian ini dilakukan pada beberapa halaman sistem penyaringan *email*.



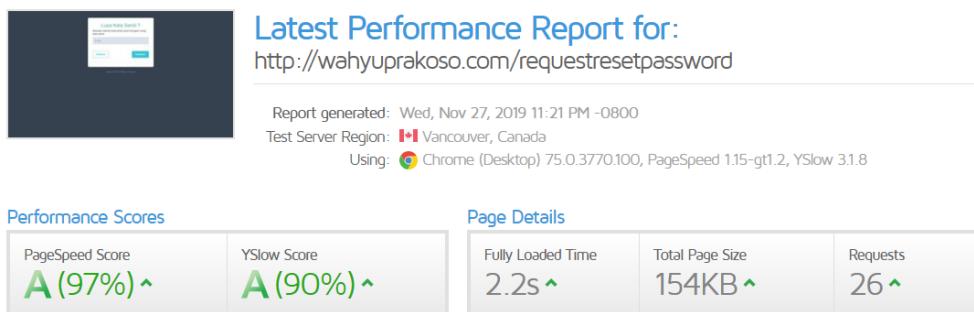
Gambar 4.37. Pengujian Aspek *Efficiency* pada Halaman Masuk

Dari gambar 4.37 dapat dilihat bahwa HTTP *request* yang dilakukan sebanyak 41 kali dengan ukuran *file* 462 KB. *Grade* pengujian berdasarkan PageSpeed adalah A (92%), sedangkan *grade* pengujian berdasarkan YSlow adalah B (82%). Waktu yang dibutuhkan untuk memproses halaman masuk adalah 3,5 detik.



Gambar 4.38. Pengujian Aspek *Efficiency* pada Halaman Daftar

Dari gambar 4.38 dapat dilihat bahwa HTTP *request* yang dilakukan sebanyak 42 kali dengan ukuran file 468 KB. *Grade* pengujian berdasarkan PageSpeed adalah A (91%), sedangkan *grade* pengujian berdasarkan YSlow adalah B (82%). Waktu yang dibutuhkan untuk memproses halaman daftar adalah 3,9 detik.



Gambar 4.39. Pengujian Aspek *Efficiency* pada Halaman Lupa Kata Sandi

Dari gambar 4.39 dapat dilihat bahwa HTTP *request* yang dilakukan sebanyak 26 kali dengan ukuran file 154 KB. *Grade* pengujian berdasarkan PageSpeed adalah A (97%), sedangkan *grade* pengujian berdasarkan YSlow adalah A (90%). Waktu yang dibutuhkan untuk memproses halaman daftar adalah 2,2 detik. Hasil pengujian aspek *efficiency* dapat dilihat pada tabel 4.18.

Tabel 4.18 Pengujian Aspek *Efficiency*.

No	Halaman	Page Load Time (Detik)	Page Size (KB)	PageSpeed Grade	Yslow Grade
1	Masuk	3,5	462	A (92%)	B (82%)
2	Daftar	3,9	468	A (91%)	B (82%)
3	Atur Ulang Kata Sandi	2,2	154	A (97%)	A (90%)
Rata-rata		3,2	362	A (93%)	B (84%)

d) Aspek *Usability*

Hasil pengujian aspek *usability* terhadap mahasiswa Program Studi Pendidikan Teknik Informatika dan Komputer di Universitas Negeri Semarang bisa dilihat pada tabel 4.19. Data responden pada tabel 4.19 dapat dilihat pada lampiran 4 sampai dengan lampiran 39.

Tabel 4.19 Hasil Pengujian Aspek *Usability*

Responden	Skor pada Setiap Pernyataan SUS										Total Skor SUS
	1	2	3	4	5	6	7	8	9	10	
1	10	7,5	7,5	7,5	7,5	7,5	10	10	7,5	5	80
2	10	7,5	7,5	5	10	10	7,5	10	5	2,5	75
3	7,5	10	10	5	7,5	7,5	10	10	7,5	2,5	77,5
4	7,5	10	10	7,5	7,5	10	10	7,5	10	2,5	82,5
5	10	2,5	10	10	10	10	7,5	10	7,5	10	87,5
6	10	10	10	7,5	10	10	7,5	10	10	7,5	92,5
7	10	7,5	10	7,5	10	10	10	10	10	2,5	87,5
8	10	7,5	10	7,5	10	5	10	10	7,5	0	77,5
9	10	7,5	10	7,5	10	10	10	7,5	10	10	92,5
10	7,5	10	7,5	7,5	10	10	7,5	10	7,5	7,5	85
11	7,5	5	7,5	2,5	10	10	7,5	7,5	7,5	2,5	67,5
12	10	10	10	7,5	10	10	10	10	7,5	7,5	92,5
13	10	7,5	7,5	5	10	10	7,5	7,5	10	5	80
14	10	7,5	7,5	7,5	10	7,5	7,5	7,5	10	7,5	82,5
15	10	5	10	5	10	5	7,5	7,5	7,5	5	72,5

Responden	Skor pada Setiap Pernyataan SUS										Total Skor SUS
	1	2	3	4	5	6	7	8	9	10	
16	7,5	7,5	7,5	10	5	7,5	5	7,5	10	0	67,5
17	5	7,5	7,5	2,5	7,5	5	7,5	2,5	2,5	0	47,5
18	7,5	10	10	10	7,5	10	10	10	7,5	10	92,5
19	10	7,5	10	10	10	7,5	7,5	7,5	10	7,5	87,5
20	10	10	10	10	10	10	10	10	10	10	100
21	10	7,5	7,5	0	10	10	10	7,5	7,5	2,5	72,5
22	10	7,5	7,5	7,5	10	7,5	7,5	10	7,5	7,5	82,5
23	5	5	7,5	7,5	10	10	7,5	7,5	7,5	2,5	70
24	7,5	7,5	7,5	7,5	7,5	7,5	10	7,5	10	7,5	80
25	10	7,5	7,5	10	10	10	7,5	10	10	10	92,5
26	5	2,5	7,5	0	10	7,5	5	2,5	5	0	45
27	10	7,5	7,5	7,5	10	10	7,5	7,5	7,5	5	80
28	0	2,5	5	2,5	10	7,5	10	7,5	0	0	45
29	10	10	7,5	7,5	10	10	0	10	7,5	2,5	75
30	10	10	10	5	10	10	7,5	10	10	10	92,5
31	5	2,5	5	2,5	10	7,5	5	2,5	5	0	45
32	7,5	7,5	10	7,5	10	7,5	5	7,5	7,5	7,5	77,5
33	5	7,5	7,5	5	7,5	5	5	5	7,5	5	60
34	10	7,5	10	10	10	10	10	7,5	7,5	10	92,5
35	10	10	10	5	10	10	7,5	10	7,5	5	85
36	10	5	7,5	10	10	7,5	7,5	7,5	7,5	7,5	80
Rerata Total Skor SUS											77,84

Hasil pengujian aspek *usability* berdasarkan tabel 4.19 memperoleh rerata total skor SUS yaitu 77,84. Nilai skor SUS yang telah didapatkan dapat diketahui tiga sudut pandang yaitu *acceptability*, *grade scale*, dan *adjective rating*. Ketentuan penilaian SUS dapat dilihat pada gambar 3.31. Berdasarkan ketentuan penilaian SUS maka pada aspek *usability* sistem termasuk pada kategori *acceptable*, memperoleh *grade C*, dan *adjective rating* pada kategori *excellent*.

4.2 Pembahasan

Sistem penyaringan *email* yang telah dibuat berfungsi untuk mengatasi permasalahan penyaringan *email* dari *provider* yang secara sepihak mengkategorikan *email* sebagai *spam* tanpa pemberitahuan ke pengguna. Sistem penyaringan *email* yang telah dibuat dalam menyaring *email*-nya dengan cara mengolah kembali data *email* yang berada di *folder* kotak masuk dan *spam* sesuai pengaturan penyaringan pengguna pada daftar anti *spam*. Penyaringan *email* berdasarkan pada subyek, alamat *email*, domain, dan *Top Level Domain* (TLD).

Algoritma yang digunakan adalah algoritma Boyer Moore yaitu algoritma pencocokan *pattern*. Algoritma ini merupakan jenis *exact string matching algorithm* yang melakukan pencocokan *string* secara tepat dengan susunan karakter dalam *string* yang dicocokkan memiliki jumlah maupun urutan karakter dalam *string* yang sama. Cara kerja algoritma ini adalah dengan melakukan pencocokan dari kanan ke kiri yaitu memeriksa karakter *pattern* dari kanan ke kiri dimulai dari karakter paling kanan. Berdasarkan hasil pengujian *white box* dapat diperoleh nilai *cyclomatic complexity* yaitu bernilai 5. Berdasarkan nilai *cyclometric complexity* maka dapat diketahui hubungan antara *cyclometric complexity* dan resiko yang dapat dilihat pada tabel 3.7 yaitu algoritma Boyer Moore merupakan algoritma yang sederhana dan memiliki resiko rendah.

Performa algoritma Boyer Moore dapat maksimal atau lebih cepat dalam proses pencocokan jika teks dan *pattern* memiliki jumlah karakter unik yang banyak pada teks atau *pattern* yang panjang. Performa algoritma Boyer Moore pada sistem

penyaringan *email* dapat dikategorikan cepat dikarenakan proses pencocokan teks dan *pattern* dalam waktu kurang dari satu detik yaitu rata-rata 20,85 μ s. Pada penelitian yang relevan menghasilkan beberapa hasil pengujian performa algoritma Boyer Moore yang berbeda-beda. Penelitian pertama mendapatkan hasil rata-rata 55,02 μ s (Fince Tinus Waruwu dan Rila Mandala, 2016: 42). Penelitian kedua mendapatkan hasil rata-rata 1040 μ s (Parenrengi, *et al.*, 2017: 123). Penelitian yang relevan menguji hanya menggunakan data karakter alfabet yang memiliki cakupan 52 karakter unik.

Berdasarkan hasil pengujian dari penelitian yang relevan terlihat perbedaan yang signifikan. Data pengujian pada sistem penyaringan *email* terdiri dari karakter alfabet, angka, dan simbol yang memiliki cakupan 78 karakter unik. Dengan adanya karakter unik yang lebih banyak maka dalam proses pencocokan *pattern* pada teks menggunakan algoritma Boyer Moore memiliki potensi lebih banyak menggunakan nilai panjang *pattern* dalam perpindahannya. Pengujian performa algoritma Boyer Moore pada sistem penyaringan *email* dilakukan menggunakan VPS agar proses pengujian lebih akurat. Dengan penggunaan VPS pengujian hanya berfokus pada sistem penyaringan *email* tanpa ada gangguan sistem lain yang sedang aktif. Spesifikasi VPS yang digunakan terdiri dari 3 *core processor* (7.2 GHz), RAM 3GB dan 40 GB SSD. Penggunaan *hardware* tersebut sangat mempengaruhi hasil dalam pengujian performa algoritma Boyer Moore. Proses pengujian akan dibagi ke 3 *core processor* yang setiap *core* memiliki kemampuan 2,4 GHz. Penggunaan *multi core* ini akan menghasilkan data yang berbeda jika hanya menggunakan *single core*.

Penggunaan algoritma Boyer Moore ternyata kurang sesuai dalam penyaringan *email* jika berdasarkan subyek. Algoritma Boyer Moore berkerja dan berfungsi dengan baik jika pada penyaringan *email* yang berdasarkan alamat *email*, *domain* dan *Top Level Domain* (TLD) dikarenakan teks dan *pattern* sudah pasti berbentuk *character set* ASCII. Sedangkan pada subyek pada *email* dapat berupa ASCII, *encode* UTF-8, dan iconv. Sehingga algoritma Boyer Moore tidak bisa mencocokan *pattern* atau teks berupa karakter spesial seperti *emoji* di dalam *encode* UTF-8.

Sistem penyaringan *email* yang telah dibuat dapat mengatasi jika pengguna mengalami permasalahan pada penyaringan *email* pada *Email Service Provider* (ESP) yang digunakan. Permasalahan seperti kesalahan dalam mengklasifikasikan *email* sebagai *spam*, tidak adanya pengaturan penyaringan *email* untuk pengguna lain dengan hak akses yang berbeda dalam satu *host* yang sama.

Sistem penyaringan *email* yang dibuat menggunakan *framework* Laravel yang bertujuan agar mudah dalam pengembangan, dikarenakan komunitas yang besar. Namun *framework* Laravel memiliki kekurangan dalam pembuatan sistem penyaringan *email* yaitu dalam menangani kebutuhan data secara *real time*. Sehingga pada sistem penyaringan *email* yang telah dibuat masih mempunyai kekurangan yaitu sinkronisasi data *email* melalui IMAP masih dibebankan kepada *client*, sehingga pengguna akan menunggu proses sinkronisasi hingga selesai sebelum mendapatkan data *email* yang baru.

Sinkronisasi IMAP secara *real time* membutuhkan *WebSocket* yang berfungsi untuk mengirim data dari *server* ke *client* yang sedang terhubung melalui

Websocket *protocol* yang sesuai dengan *Request for Comments* (RFC) 6455. Websocket menggunakan *single request* sehingga dapat mengirim *response* ke *client* meskipun *client* tidak mengirim *request*. Pada saat pengujian performa sistem penyaringan *email* menggunakan *monolithic architecture* yaitu *user interface*, *business logic* dan *data access layer* yang langsung terhubung ke *database* pada satu *server*. Untuk meminimalkan beban pada *server* saat sinkronisasi IMAP maka dibutuhkan *microservice architecture* yaitu terdapat *user interface*, *microservice* yang di dalamnya bisa terdapat *business logic* atau *data access layer*, dan *database* pada *server* yang berbeda.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat diambil kesimpulan bahwa:

1. Sistem penyaringan *email* dibuat menggunakan *framework* laravel yang merupakan salah satu *framework* yang memisahkan antara data (*model*) dari tampilan (*view*) dan cara bagaimana memprosesnya (*controller*). Dalam pengembangan menggunakan metode *waterfall* yang terdiri dari *communication, planning, modelling, construction, dan deployment*.
2. Sistem penyaringan *email* dalam mengkategorikan *email* itu sebagai *spam* menggunakan algoritma Boyer Moore yang berdasarkan data *header email* yaitu, subyek, alamat *email, domain, Top Level Domain* (TLD).
3. Performa algoritma Boyer Moore pada sistem penyaringan *email* dapat dikategorikan cepat dikarenakan proses pencocokan teks dan *pattern* dalam waktu kurang dari satu detik yaitu rata-rata 20,85 μ s. Performa algoritma Boyer Moore dapat maksimal atau lebih cepat dalam proses pencocokan jika teks dan *pattern* memiliki jumlah karakter unik yang banyak pada teks atau *pattern* yang panjang. Hasil pengujian *white box* diperoleh nilai *cyclometric complexity* yaitu 5 yang menunjukkan algoritma Boyer Moore merupakan algoritma yang sederhana dan memiliki resiko rendah. Pengujian *black box* memperhatikan beberapa aspek yaitu *functionality*,

portability, efficiency, dan usability. Pada aspek *functionality* memperoleh hasil 100% yang berarti semua fungsi berjalan dan berfungsi dengan baik. Pada aspek *portability* memperoleh hasil 100% yang menunjukan sistem dapat di akses dari berbagai macam *browser* dan *device*. Pada aspek *efficiency* sistem sudah baik karena memperoleh rata-rata *grade* PageSpeed A (93%) dan Yslow B (84%), pada rata-rata *page size* 362 KB dengan *load time* rata-rata 3,2 detik. Pada aspek *usability* memperoleh hasil skor SUS yaitu 77,84. Jika berdasarkan cara penilaian SUS maka pada aspek *usability* sistem termasuk pada kategori *acceptable*, memperoleh *grade* C, dan *adjective rating* pada kategori *excellent*.

5.2 Saran

Berdasarkan penelitian yang telah dilakukan, diajukan saran penelitian lanjutan sebagai berikut:

1. Gunakan *framework* lain yang arsitekturnya dapat menangani dan menyediakan fitur *real time*.
2. Gunakan algoritma lain yang bukan berjenis *exact string matching* untuk penyaringan *email* berdasarkan subyek.
3. Gunakan arsitektur *microservice*.

DAFTAR PUSTAKA

- Ardiansyah, & Ghazali, M. I. (2016). Pengujian Usability User Interface Dan User Experience Aplikasi E-Reader Skripsi Berbasis Hypertext. *JITTER (Jurnal Ilmiah Teknik Informasi Terapan)*, II(3): 213–220.
- Baranwal, A., Gaur G., Bhasker A., & Jain R. (2018). Spam Filtration using Boyer Moore Algorithm and Naïve Method, 180(42): 35–38.
- Chettri, P., & Kar, C. (2016). Comparative Study between Various Pattern Matching Algorithms. *International Journal of Computer Applications*, 26–30.
- Easton, C. R. (2012). ICANN's core principles and the expansion of generic top-level domain names. *International Journal of Law and Information Technology*, 20(4), 273–290. <https://doi.org/10.1093/ijlit/eas013>.
- Ependi, U., Kurniawan, T. B., & Panjaitan, F. (2019). System Usability Scale Vs Heuristic Evaluation: a Review. *Simetris: Jurnal Teknik Mesin, Elektro Dan Ilmu Komputer*, 10(1), 65–74. <https://doi.org/10.24176/simet.v10i1.2725>
- Haryanto, E. D. (2012). *Jaringan Komputer*. Yogyakarta: ANDI.
- Haycox, I., McDonald, A., Bäck, M., Hildebrandt, R., Koetter, P. B., Rusenko, D., & Taylor, C. (2009). *Linux E-mail Set up, maintain, and secure a small office e-mail server*. Birmingham: Packt Publishing Ltd.
- Ismail, F. (2018). *Statistika Untuk Penelitian Pendidikan dan Ilmu-Ilmu Sosial*. Jakarta: Pranamedia Group.
- Laaziri, M., Benmoussa, K., Khoulji, S., & Kerkeb, M. L. (2019). A Comparative study of PHP frameworks performance. *Procedia Manufacturing*, 32, 864–871. <https://doi.org/10.1016/j.promfg.2019.02.295>
- McCool, S. (2012). *Laravel Starter The definitive introduction to the Laravel PHP web development framework*. Birmingham: Packt Publishing Ltd.
- Nonprofit Business Advisor. (2016). With spam filters catching 7 percent of emails, nonprofits need to curate mailing lists. *Nonprofit Business Advisor*, 323: 1–3. <https://doi.org/10.1002/nba.30216>.
- Parenrengi, A. M., Saputra, I. A., & Tajudin, L. M. (2017). Analisis Perbandingan Algoritma Boyer Moore Dan Algoritma Knuth Morris Pratt Pada Aplikasi Tripelka Foodshop Kendari Berbasis Android. *semanTIK*. 3(1): 115-126.

- Pratama, A. (2015). *HTML Uncover Panduan Belajar HTML Untuk Pemula*. Cetakan Pertama. Padang Panjang: DuniaIlkom.
- _____. (2016). *PHP Uncover Panduan Belajar PHP Untuk Pemula*. Cetakan Pertama. Padang Panjang: DuniaIlkom.
- _____. (2017). *MySQL Uncover Panduan Belajar MySQL Dan MariaDB Untuk Pemula*. Cetakan Pertama. Padang Panjang: DuniaIlkom.
- Pressman, R. S., & Maxim, B. R. (2014). *Software Engineering: A Practitioner's Approach*. Edisi Delapan. New York: McGraw-Hill Education.
- Rahim, R., Ahmar, A. S., Ardyanti, A. P., & Nofriansyah, D. (2017). Visual Approach of Searching Process using Boyer-Moore Algorithm. *Journal of Physics: Conference Series*, 930(1). <https://doi.org/10.1088/1742-6596/930/1/012001>.
- Rahmanita, E. (2014). Pencarian String Menggunakan Algoritma Boyer Moore Pada Dokumen. *Jurnal Ilmiah Nero* 1(1): 15-26.
- Ramya. (2019, Mei). *Emails classified as Spam Office 365 - Reasons and recommendations*. Diakses pada 20 Juni 2019, dari <https://help.zoho.com/portal/en/community/topic/emails-classified-as-spam-office-365-reasons-and-recommendations>.
- Rasool, A., Tiwari, A., Singla, G., & Khare, N. (2012). String Matching Methodologies: A Comparative Analysis. (*IJCSIT International Journal of Computer Science and Information Technologies*, 3(2), 3394–3397.
- Razak, S. A. B., & Mohammad, A. F. B. (2014). Identification of spam email based on information from email header. *International Conference on Intelligent Systems Design and Applications*, ISDA, 347–353. <https://doi.org/10.1109/ISDA.2013.6920762>
- Sachhidanand. (2011). Analysis on Software Cyclomatic Complexity. *International Journal of Engineering and Management Research*, 1(1), 30-32.
- Sagita, V., & Prasetyowati, M. I. (2013). Studi Perbandingan Implementasi Algoritma Boyer-Moore, Turbo Boyer-Moore, dan Tuned Boyer-Moore dalam Pencarian String. *Jurnal ULTIMATICS*, 5(1), 31–37. <https://doi.org/10.31937/ti.v5i1.311>
- Sharfina, Z., & Santoso, H. B. (2017). An Indonesian adaptation of the System Usability Scale (SUS). *2016 International Conference on Advanced Computer Science and Information Systems*, ICACSIS 2016, 145–148. <https://doi.org/10.1109/ICACSIS.2016.7872776>

- Sitorus, L. (2015). *Algoritma Dan Pemrograman*. Cetakan Pertama. Yogyakarta: ANDI.
- Srinivas, I. V., Samnani, M. & Shaikh, M. S. (2018). Study of String Matching Algorithm. *IOSR Journal of Computer Engineering*, 32-35.
- Sugiyono. (2017). *Metode Penelitian Kuantitatif, Kualitatif, dan R&D*. Cetakan Ke-26. Bandung: Alfabeta.
- Rusmawan, U. (2019). Teknik Penulisan Tugas Akhir dan Skripsi Pemrograman. Jakarta: Elex Media Komputindo.
- Vezzani, S. (2014). ICANN's New Generic Top-Level Domain Names Dispute Resolution Procedure Viewed Against the Protection of the Public Interest of the Internet Community: Litigation Regarding Health-Related Strings. *The Law And Practice Of International Courts And Tribunals*, 13, 306-346.
- Wuruwu, F. T., & Mandala, R. (2016). Perbandingan Algoritma Knuth Morris Pratt Dan Boyer Moore Dalam Pencocokan String Pada Aplikasi Kamus Bahasa Nias. *Jurnal Ilmiah INFOTEK*, 1(1), 36-43.
- Yudhanto, Y., & Prasetyo H. A. (2018). Panduan Mudah Belajar Framework Laravel. Jakarta: Elex Media Komputindo.

LAMPIRAN

Lampiran 1 Surat Penetapan Dosen Pembimbing Skripsi/Tugas Akhir


**KEPUTUSAN
DEKAN FAKULTAS TEKNIK
UNIVERSITAS NEGERI SEMARANG
Nomor: 67 / FT - UNNES / 2019**
 Tentang
**PENETAPAN DOSEN PEMBIMBING SKRIPSI/TUGAS AKHIR SEMESTER
GASAL/GENAP
TAHUN AKADEMIK 2018/2019**

Menimbang	: Bawa untuk memperlancar mahasiswa Jurusan/Prodi Teknik Elektro/Pend. Teknik Informatika dan Komputer Fakultas Teknik membuat Skripsi/Tugas Akhir, maka perlu menetapkan Dosen-dosen Jurusan/Prodi Teknik Elektro/Pend. Teknik Informatika dan Komputer Fakultas Teknik UNNES untuk menjadi pembimbing.
Mengingat	<ol style="list-style-type: none"> 1. Undang-undang No.20 Tahun 2003 tentang Sistem Pendidikan Nasional (Tambahannya Lembaran Negara RI No.4301, penjelasan atas Lembaran Negara RI Tahun 2003, Nomor 78) 2. Peraturan Rektor No. 21 Tahun 2011 tentang Sistem Informasi Skripsi UNNES 3. SK. Rektor UNNES No. 164/O/2004 tentang Pedoman penyusunan Skripsi/Tugas Akhir Mahasiswa Strata Satu (S1) UNNES; 4. SK Rektor UNNES No.162/O/2004 tentang penyelenggaraan Pendidikan UNNES;
Menimbang	Usulan Ketua Jurusan/Prodi Teknik Elektro/Pend. Teknik Informatika dan Komputer Tanggal 11 Januari 2019
MEMUTUSKAN	
Menetapkan	Menunjuk dan menugaskan kepada:
PERTAMA	Nama : Arief Arfandi, S.T., M.Eng. NIP : 198208242014041001 Pangkat/Golongan : III/b Jabatan Akademik : Asisten Ahli Sebagai Pembimbing Untuk membimbing mahasiswa penyusunan skripsi/Tugas Akhir : Nama : WAHYU PRAKOSO NIM : 5302415028 Jurusan/Prodi : Teknik Elektro/Pend. Teknik Informatika dan Komputer Topik : Implementasi Algoritma Boyer Moore Untuk Penyaringan Email Spam
KEDUA	Keputusan ini mulai berlaku sejak tanggal ditetapkan.

**DITETAPKAN DI : SEMARANG
PADA TANGGAL : 14 Januari 2019**
DEKAN

 Di : Nur Qudus, M.T
 NIP : 196911301994031001 F

Tembusan
 1. Pembantu Dekan Bidang Akademik
 2. Ketua Jurusan
 3. Petinggal

 5302415028
 ... FM-03-AKD-24/Rev. 00 ...

Lampiran 2 Formulir Usulan Topik Skripsi



Formulir Usulan Topik Skripsi
FM-1-AKD-24/rev.00
UNIVERSITAS NEGERI SEMARANG

Usulan topik skripsi ini diajukan oleh:

Nama : WAHYU PRAKOSO
NIM : 5302415028
Jurusan : Teknik Elektro
Program Studi : Pendidikan Teknik Informatika dan Komputer, S1
Topik : Implementasi Algoritma Boyer Moore Untuk Penyaringan Email Spam

Menyetujui
Ketua Jurusan

Dr.-Ing. Dhidik Prastiyanto, S.T.,M.T.
NIP. 19780531200511002

Semarang, 8 Januari 2019

Yang mengajukan,

WAHYU PRAKOSO
NIM. 5302415028



Lampiran 3 Surat Tugas Seminar Proposal Skripsi



**KEMENTERIAN RISET, TEKNOLOGI, DAN PENDIDIKAN TINGGI
UNIVERSITAS NEGERI SEMARANG
FAKULTAS TEKNIK**
Gedung Dekanat Kampus Sekaran Gunungpati Semarang 50229
Telepon/Fax (024) 8508101 - 8508009
Laman : <http://www.ft.unnes.ac.id>, email: ft@mail.unnes.ac.id

SURAT TUGAS
Nomor : 5169 /UN37.1.5/TD.06/2019

Dekan Fakultas Teknik Universitas Negeri Semarang memberi tugas kepada Saudara yang namanya tersebut di bawah ini sebagai Pengaji Seminar Proposal Skripsi Mahasiswa Prodi S1 Pendidikan Teknik Informatika dan Komputer Jurusan Teknik Elektro Fakultas Teknik Universitas Negeri Semarang. Adapun nama-namanya sebagai berikut:

No	Nama / NIP	Pangkat / Golru	Tugas
1	Dr. Ing. Dhidik Prastiyanto, S.T., M.T. 197805312005011002	Penata, III/c	Pengaji 1
2	Aryo Baskoro Utomo, S.T., M.T. 198409092012121002	Penata Muda Tk. I, III/b	Pengaji 2
3	Arief Arfandi, S.T., M.Eng. 198208242014041001	Penata Muda Tk. I, III/b	Pembimbing

untuk menguji mahasiswa :

Nama : WAHYU PRAKOSO
 NIM : 5302415028
 Prodi : S1 Pendidikan Teknik Informatika dan Komputer
 Topik : IMPLEMENTASI ALGORITMA BOYER MOORE UNTUK PENYARINGAN EMAIL SPAM

 Waktu : 23 Mei 2019
 Jam : 10.00-12.00
 Tempat : E11 R. Seminar
 Pakaian : Hitam Putih Jas Almamater

Demikian agar tugas dilaksanakan dengan sebaik-baiknya.



Tembusan :

1. Wakil Dekan Bidang II;
2. Ketua Jurusan TE;
3. Kasubbag Keuangan,
Fakultas Teknik UNNES

Lampiran 4 SUS Responden 1

System Usability Scale (SUS)

Responden : ...

Nama : Dias Fadillah

Prodi : Penda TIK

NIM : 5302415019

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.				✓	
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.		✓			
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat					✓
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.			✓		

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

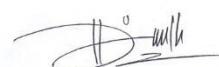
TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 15 Oktober 2019



Dias Fadillah

Lampiran 5 SUS Responden 2

System Usability Scale (SUS)

Responden : ...2...

Nama : ...Malikatul Asna.....

Prodi : ...PTIK.....

NIM : ...S302415011.....

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.			✓		
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat			✓		
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini			✓		
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.				✓	

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 15 Oktober 2019

Malikatul Asna

Lampiran 6 SUS Responden 3

System Usability Scale (SUS)

Responden : 3
 Nama : Ayunita Yahdiani
 Prodi : PTIK
 NIM : 5302415006

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.				✓	
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.			✓		
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.				✓	
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.		✓			
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.				✓	

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang 15 Oktober 2019



(Ayunita Yahdiani)

Lampiran 7 SUS Responden 4

System Usability Scale (SUS)

Responden : 4
 Nama : Hilda Maghfirah
 Prodi : PTIK
 NIM : 5302915016

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.				✓	
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan					✓
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.				✓	
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.		✓			
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat					✓
8	Saya merasa sistem ini membingungkan		✓			
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini					✓
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.				✓	

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

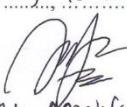
TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 15 Oktober 2019



... (Hilda Maghfirah) ...

Lampiran 8 SUS Responden 5

System Usability Scale (SUS)

Responden : 5
 Nama : BAMBANG SARJIO
 Prodi : PTIK
 NIM : 5302418063

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.				✓	
2	Saya merasa sistem ini rumit untuk digunakan.			✓		
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.	✓				
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.				✓	
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat			✓		
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.	✓				

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

- STJ = Sangat Tidak Setuju
- TJ = Tidak Setuju
- N = Netral
- S = Setuju
- SS = Sangat Setuju

Surabaya, 18 Oktober 2019



BAMBANG SARJIO

Lampiran 9 SUS Responden 6

System Usability Scale (SUS)

Responden : 6
 Nama : Danar Putri Septami
 Prodi : PTK
 NIM : 3302415024

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan					✓
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini					✓
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.		✓			

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Surabaya, 18 Oktober 2019.


Danar Putri Septami

Lampiran 10 SUS Responden 7

System Usability Scale (SUS)

Responden : ...7...
 Nama : Eksi Umayani
 Prodi : PTIK
 NIM : 5302A1S010

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan					✓
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat					✓
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini					✓
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.				✓	

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

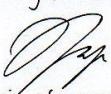
TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 18 Oktober 2019



Eksi Umayani

Lampiran 11 SUS Responden 8

System Usability Scale (SUS)

Responden : 8
 Nama : Fika Adin Laila
 Prodi : Pend. Teknik Informatika
 NIM : 5302415021

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan					✓
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.			✓		
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat					✓
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.					✓

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

..... Semarang, 18 Oktober 2019

Fika Adin Laila
Fika ADIN LA ILA

Lampiran 12 SUS Responden 9

System Usability Scale (SUS)

Responden : ...9...

Nama : Rizki Aji Saputro

Prodi : ~~530241030~~ PTIK

NIM : 530241030

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan					✓
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan		✓			
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini					✓
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.	✓				

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

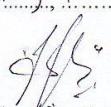
TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 18 Oktober 2019



Rizki...Aji...Saputro....

Lampiran 13 SUS Responden 10

System Usability Scale (SUS)

Responden : 10

Nama : Ahmad Mudzakkir Nurhuda

Prodi : PTIK

NIM : S302415004

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.				✓	
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.				✓	
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.		✓			

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

....., 18 Oktober 2019

AHMAD MUDZAKKIR NURHUDA

Lampiran 14 SUS Responden 11

System Usability Scale (SUS)

Responden : ...!

Nama : Nurul Dhukha

Prodi : PTIK

NIM : 5302415013

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.				✓	
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.				✓	
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.				✓	

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 18 Oktober 2019

Nurul Dhukha

Lampiran 15 SUS Responden 12

System Usability Scale (SUS)

Responden : 12

Nama : Makhfud Zakaria Al Ghazali

Prodi : PTIK

NIM : 5302415045

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.				✓	
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.				✓	
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.		✓			
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.		✓			

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 18 Oktober 2019

Makhfud Zakaria Al Ghazali

Lampiran 16 SUS Responden 13

System Usability Scale (SUS)

Responden : 13....

Nama : Srinata Alif M.....

Prodi : ~~S2~~ PTIK.....

NIM : 5302415031.....

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.			✓		
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini					✓
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.			✓		

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

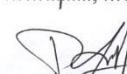
TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 21 Oktober 2019



...Srinata Alif M.....

Lampiran 17 SUS Responden 14

System Usability Scale (SUS)

Responden : 14....

Nama : Fathurrahman Prasetyo Aji....

Prodi : ~~PTIK~~ PTIK.....

NIM : 5302415035.....

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.		✓			
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membungkungkan		✓			
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini					✓
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.		✓			

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 21 Oktober 2019



Fathurrahman Prasetyo Aji

Lampiran 18 SUS Responden 15

System Usability Scale (SUS)

Responden : 15
 Nama : Ressa
 Prodi : PTIK
 NIM : 5302415015

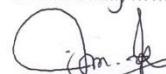
No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.				✓	
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.			✓		
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.			✓		
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan		✓			
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.			✓		

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

- STJ = Sangat Tidak Setuju
- TJ = Tidak Setuju
- N = Netral
- S = Setuju
- SS = Sangat Setuju

Semarang 21 Oktober 2019



Ressa

Lampiran 19 SUS Responden 16

System Usability Scale (SUS)

Responden : 16....

Nama : Ervansah Arif K.....
 Prodi : Pend. Teknik Informatika dan Komputer
 NIM : 5302415025.....

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.				✓	
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.	✓				
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.			✓		
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.		✓			
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat			✓		
8	Saya merasa sistem ini membingungkan		✓			
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini					✓
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.					✓

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 21 Oktober 2019



Ervansah Arif K.

Lampiran 20 SUS Responden 17

System Usability Scale (SUS)

Responden : !7...

Nama : Diah Novianti

Prodi : Pendidikan Teknik Informatika & Komputer

NIM : 5302415018

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.			✓		
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.				✓	
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.			✓		
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan				✓	
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini		✓			
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.					✓

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

Semarang, 21 Oktober 2019

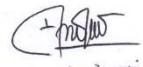
STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju


Diah Novianti.....

Lampiran 21 Responden 18

System Usability Scale (SUS)

Responden : 18...

Nama : Armin Harjianton

Prodi : PTIK

NIM : 5302415008

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.				✓	
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan					✓
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.	✓				
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.				✓	
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat					✓
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.	✓				

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

Semarang, 21 Oktober 2019

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju



Armin Harjianton

Lampiran 22 Responden 19

System Usability Scale (SUS)

Responden : 19....

Nama : Abdil M.A.....

Prodi : PTK.....

NIM : 52020115040.....

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan					✓
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.		✓			
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan		✓			
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini					✓
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.		✓			

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

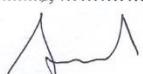
TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 22 Oktober 2019


 (ABDI M.A)

Lampiran 23 SUS Responden 20

System Usability Scale (SUS)

Responden : 20

Nama : Dhea Permatasari

Prodi : PTIK

NIM : 5302415034

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan					✓
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.	✓				
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat					✓
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini					✓
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.	✓				

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 22 Oktober 2019

Dhea Permatasari

Lampiran 24 SUS Responden 21

System Usability Scale (SUS)

Responden : 21

Nama : KEVIN RICO BUDYONO

Prodi : PTIK

NIM : 5302015051

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.					✓
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat					✓
8	Saya merasa sistem ini membingungkan		✓			
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.					✓

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang / 22. 06. 2019

Kevin Rico Budyono

Lampiran 25 SUS Responden 22

System Usability Scale (SUS)

Responden : 22
 Nama : Dewi Wahyudini
 Prodi : PTK
 NIM : 5302415010

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.	✓				
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.		✓			

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

....., 22 Oktober 2019


Dewi Wahyudini

Lampiran 26 SUS Responden 23

System Usability Scale (SUS)

Responden : 23.

Nama : Bayu Dwi Istiyanto.....

Prodi : PTK.....

NIM : 5302415039.....

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.			✓		
2	Saya merasa sistem ini rumit untuk digunakan.			✓		
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan		✓			
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.					✓

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 22 Oktober 2019


 Bayu...Dwi...Istiyanto..

Lampiran 27 SUS Responden 24

System Usability Scale (SUS)

Responden : 24

Nama : Eka Tanti Pratiwi

Prodi : PTIK

NIM : 5302415089

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.				✓	
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.				✓	
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.		✓			
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat					✓
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini					✓
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.		✓			

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 23 Oktober 2019

Eka Tanti Pratiwi

Lampiran 28 Responden 25

System Usability Scale (SUS)

Responden : 25

Nama : Anul Fitriyah M.....

Prodi : PTIK

NIM : 5302415036

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan			✓		
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.	✓				
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat			✓		
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.	✓				

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

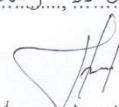
TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 22 Oktober 2019



Anul Fitriyah M.....

Lampiran 29 Responden 26

System Usability Scale (SUS)

Responden : 26...

Nama : Dwita Gladea

Prodi : PTIK

NIM : S30241S017

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.			✓		
2	Saya merasa sistem ini rumit untuk digunakan.				✓	
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.					✓
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.			✓		
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat			✓		
8	Saya merasa sistem ini membingungkan				✓	
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini			✓		
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.					✓

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

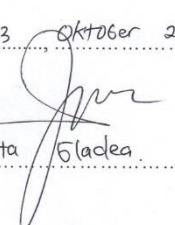
TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

23 Oktober 2019



Dwita Gladea

Lampiran 30 Responden 27

System Usability Scale (SUS)

Responden : 27...

Nama : Rudy Widayoko

Prodi : PTIK

NIM : 530.241.5057

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.				✓	
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan			✓		
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.	✓				
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.				✓	
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat			✓		
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini			✓		
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.		✓			

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

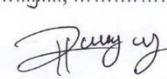
TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 24 October 2019


 Rudy Widayoko

Lampiran 31 Responden 28

System Usability Scale (SUS)

Responden : 28...

Nama : Mutia Diah P.

Prodi : PTIK

NIM : S302415007

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.	✓				
2	Saya merasa sistem ini rumit untuk digunakan.				✓	
3	Saya merasa sistem ini mudah untuk digunakan			✓		
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.				✓	
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.		✓			
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan		✓			
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini	✓				
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.					✓

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

23 Oktober 2019

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

Mutia
Dyah P.

S = Setuju

SS = Sangat Setuju

Lampiran 32 SUS Responden 29

System Usability Scale (SUS)

Responden : 29...

Nama : Bima Aqung Setya Paudi

Prodi : PTIK.....

NIM : 5302415096.....

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat	✓				
8	Saya merasa sistem ini membingungkan					✓
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini		✓			
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.					✓

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

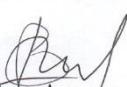
TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 29 Oktober 2019



Bima Aqung S.Pd

Lampiran 33 SUS Responden 30

System Usability Scale (SUS)

Responden : 30...

Nama : Dewi Fatmala Sari

Prodi : PTIK

NIM : 5302415033

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan					✓
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.			✓		
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini					✓
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.	✓				

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

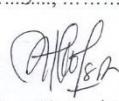
TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 24 Oktober 2019



.....Dewi Fatmala Sari.....

Lampiran 34 SUS Responden 31

System Usability Scale (SUS)

Responden : 31....

Nama : Dhiya... Zaki... Zain.....

Prodi : PTK.....

NIM : 030215038.....

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.			✓		
2	Saya merasa sistem ini rumit untuk digunakan.			✓		
3	Saya merasa sistem ini mudah untuk digunakan			✓		
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.				✓	
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.			✓		✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.		✓			
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat			✓		
8	Saya merasa sistem ini membingungkan				✓	
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini			✓		
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.					✓

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang,..., 25 Oktober 2019



...Dhiya...Zaki...Zain....

Lampiran 35 Responden 32

System Usability Scale (SUS)

Responden : 32...

Nama : Isra Prasetyo.....

Prodi : PTLK

NIM : 3302415058

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.				✓	
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan					✓
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.		✓			
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.		✓			
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat			✓		
8	Saya merasa sistem ini membingungkan		✓			
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.		✓			

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 24 Oktober 2019



Isra Prasetyo

Lampiran 36 SUS Responden 33

System Usability Scale (SUS)

Responden : 33...

Nama : Ahmad. Mafab. Zuliyanto.

Prodi : P.TIK

NIM : 530241 8047

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.			✓		
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.			✓		
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.				✓	
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.			✓		
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat			✓		
8	Saya merasa sistem ini membingungkan			✓		
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini		✓			
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.			✓		

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

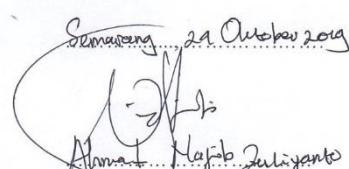
STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 21 Oktober 2019

 Ahmad. Mafab. Zuliyanto

Lampiran 37 SUS Responden 34

System Usability Scale (SUS)

Responden : 34..

Nama : Nur Khaeni Oktavia

Prodi : PTIK

NIM : 5302415001

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan					✓
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.	✓				
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat					✓
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.	✓				

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 25 Oktober 2019

Nur Khaeni Oktavia

Lampiran 38 SUS Responden 35

System Usability Scale (SUS)

Responden : 35...

Nama : Siti Wahyu Ami.....
 Prodi : PTK
 NIM : S30241509

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.	✓				
3	Saya merasa sistem ini mudah untuk digunakan					✓
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.			✓		
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.	✓				
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat			✓		
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.			✓		

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

- STJ = Sangat Tidak Setuju
 TJ = Tidak Setuju
 N = Netral
 S = Setuju
 SS = Sangat Setuju

Semarang, 25 Oktober 2016

Siti Wahyu Ami

Lampiran 39 SUS Responden 36

System Usability Scale (SUS)

Responden : 36...

Nama : Maria Perangin Angin

Prodi : Pend. TIK

NIM : 5302415060

No	Pernyataan	STJ	TJ	N	S	SS
1	Saya berpikir akan menggunakan sistem ini lagi.					✓
2	Saya merasa sistem ini rumit untuk digunakan.		✓			
3	Saya merasa sistem ini mudah untuk digunakan				✓	
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.	✓				
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.					✓
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.		✓			
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat				✓	
8	Saya merasa sistem ini membingungkan	✓				
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini				✓	
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.	✓				

Beri tanda centang (✓) pada pilihan yang tersedia

Keterangan:

STJ = Sangat Tidak Setuju

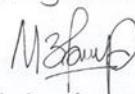
TJ = Tidak Setuju

N = Netral

S = Setuju

SS = Sangat Setuju

Semarang, 24 Okt. 2019


 ...Maria Perangin Angin