

BAB 6 IMPLEMENTASI

Pada bagian ini akan dijelaskan mengenai implementasi yang dilakukan dalam membangun sistem ini. Bahasan pada sub bab terdiri dari penjelasan spesifikasi lingkungan implementasi baik perangkat keras maupun perangkat lunak, batasan implementasi, penjelasan implementasi sistem dari sisi *server* dan sisi klien, serta implementasi antarmuka dari rancangan pada bab sebelumnya.

6.1 Spesifikasi Lingkungan Implementasi

Pada sub bab ini menjelaskan mengenai spesifikasi dari perangkat keras dan perangkat lunak yang digunakan dalam membangun sistem ini.

6.1.1 Spesifikasi Perangkat Keras

Lingkungan perangkat keras yang digunakan dalam membangun sistem ini dapat dilihat pada Tabel 6.1.

Tabel 6.1 Spesifikasi lingkungan perangkat keras

Laptop Asus A455LB	
Processor	Intel(R) Core(TM) i7-5500U CPU @2.40GHz
Kapasitas RAM	8.00 GB
Kapasitas <i>hardisk</i>	1 TB
Kartu Grafis	Intel HD Graphics
Maksimal Resolusi Layar	1366px x 768px

6.1.2 Spesifikasi Perangkat Lunak

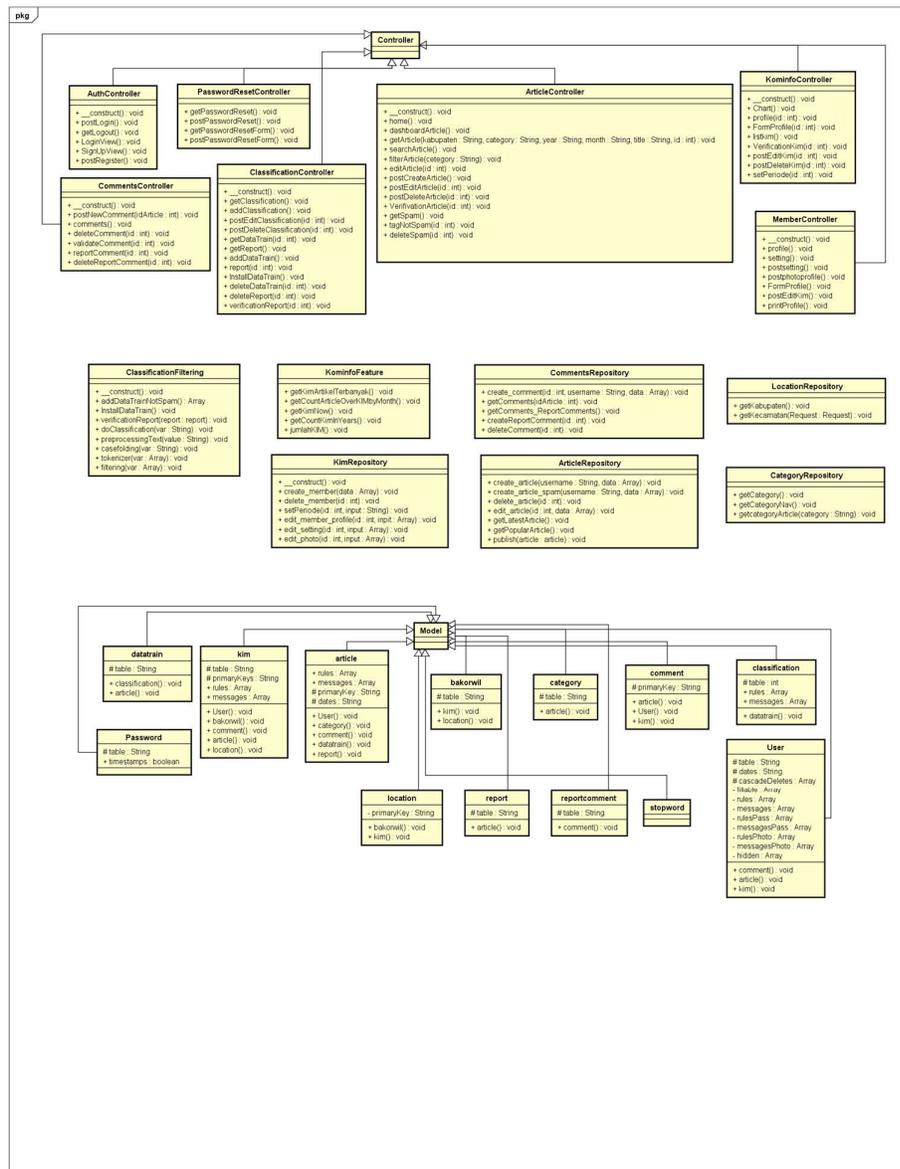
Lingkungan perangkat lunak yang digunakan dalam membangun sistem ini dapat dilihat pada Tabel 6.2

Tabel 6.2 Tabel spesifikasi lingkungan perangkat lunak

Laptop Asus A455LB	
Sistem Operasi	Windows 10 Pro 64-bit
Bahasa Pemrograman	HTML, PHP, CSS, Javascript, SQL
Perangkat Lunak Pendukung	homestead phpMyAdmin Atom DBDesigner Microsoft Word 2016 Google Chrome Astah
<i>Library</i> Pendukung	Jquery Satrawati

Laravelcollective html
 google recaptcha
 maximebf cachecache
 camspiars statistical-classifier
 iatstuti laravel-cascade-soft-deletes

6.2 Implementasi Class Diagram



Gambar 6.1 Implementasi class diagram

Pada Gambar 6.1 digambarkan *class diagram* yang sudah diimplementasikan dengan menggunakan framework laravel. Pada implementasi *class diagram* terdapat *class controller* dan *model* yang berperan sebagai *parent* dari *class* yang

dibuat pada tahap perancangan dan sebagian *class* tidak memiliki *parent class* yang memiliki peran sebagai pendukung proses transaksi pada *controller*.

6.3 Batasan Implementasi

Batasan implementasi pada pembangunan sistem ini adalah sebagai berikut:

1. *Web Server* yang digunakan pada pengembangan sistem ini adalah Nginx versi 1.11.1
2. *Database Management System* yang digunakan pada sistem ini adalah mysql Ver 14.14 Distrib 5.7.17, for Linux (x86_64).
3. Versi bahasa pemrograman PHP yang digunakan dalam membangun sistem ini adalah versi 7.0.8

6.4 Implementasi Menambah Artikel

Pada proses menambah artikel beberapa tahap sebelum artikel masuk disimpan dalam *database*. Terdapat proses klasifikasi untuk mengetahui artikel yang dimasukkan tidak termasuk kategori *spam*. Ketika artikel dikategorikan sebagai *spam*, maka artikel tersebut tetap disimpan di *database*, namun artikel tersebut dimasukkan dalam kategori *spam*.

Terdapat pula proses pengecekan status KIM sebelum melakukan penambahan artikel yang diterjadi pada *middleware*. Ketika status KIM aktif, maka proses penambahan artikel lanjut pada proses klasifikasi, sebaliknya jika status KIM tidak aktif, maka proses penambahan artikel dihentikan dan dikembalikan ke halaman *dashboard*.

Pada penambahan artikel memiliki batas penambahan artikel sejumlah 10. Jumlah tersebut menunjukkan bahwa terdapat artikel sejumlah 10 yang belum diproses oleh KOMINFO. Pembatasan ini digunakan untuk mengantisipasi pembuatan artikel yang berulang-ulang.

Setelah penambahan artikel berhasil, sistem akan mengirim notifikasi kepada *email* KOMINFO, sehingga KOMINFO bisa merespon dengan cepat untuk memverifikasinya. Notifikasi hanya dikirim ketika artikel yang disimpan tidak termasuk *spam*. Pada Tabel 6.3 merupakan kode program dari tambah artikel yang diimplementasikan pada *class* ArticleController dengan nama file ArticleController.php.

Tabel 6.3 Implementasi menambah artikel

No	Pseudocode
1	public function postCreateArticle()
2	{
3	\$input = Input::all();
4	\$validator = Validator::make(\$input, article::\$rules,
5	article::\$messages);
6	if(\$validator->fails())
	{

Tabel 6.3 Implementasi menambah artikel (lanjutan)

No	Pseudocode
7	return redirect()->back()
8	->withErrors(\$validator)
9	->withInput();
10	}
11	\$user_id=Auth::user()->id;
12	if(\$this->ArticleRepository->checkHasCreated() <= 10){
13	if(\$this->ClassificationFiltering->doClassification(\$input['isi'])!='Spam'){
14	\$this->ArticleRepository->create_article(\$user_id,\$input);
15	\$data = [
16	'namakim' => Auth::user()->kim->nama,
17	'operation' => 'tambah artikel dengan judul
18	'.'. \$input['judul'],
19	'subject' => 'Notifikasi artikel baru'
20];
21	\$this->userMailer->notification(\$data);
22	}else{
23	\$this->ArticleRepository->create_article_spam(\$user_id,\$input);
24	return redirect()->route('ArticleManagerKIM')
25	->with('status', 'success')
26	->with('message', 'Artikel berhasil dibuat.');
27	}else{
28	return redirect()->route('ArticleManagerKIM')
29	->with('status', 'danger')
30	->with('message', 'Terlalu banyak artikel yang belum diproses');
31	}
32	}

6.5 Implementasi Verifikasi Artikel

Pada proses verifikasi artikel dilakukan penambahan *data train* dan publikasi artikel. Artikel yang diverifikasi di masukkan ke dalam *data train* dalam kategori bukan *spam*. Kode program dari tambah verifikasi artikel dapat dilihat pada tabel 6.4 yang diimplementasikan pada *class* ArticleController dengan nama file ArticleController.php.

Tabel 6.4 Implementasi verifikasi artikel

No	Pseudocode
1	public function VerificationArticle(\$id)
2	{
3	\$article= article::find(\$id);
4	// \$this->ClassificationFiltering-
5	>addDataTrainNotSpam(\$article->isi);
6	\$this->ArticleRepository->publish(\$article);
7	return redirect()->back()
8	->with('status', 'success')
9	->with('message', 'Artikel berhasil di publish');
	}

Dalam *method* verifikasi artikel terdapat pemanggilan *method* `addDataTrainNotSpam()` dan *method* `publish()`. Pada *method* `addDataTrainNotSpam()` dilakukan *preprocessing text* yakni *casefolding*, *tokenizer*, *filtering*, *stemming*. Hasil *preprocessing text* kemudian dimasukkan ke dalam *data train*.

Selanjutnya artikel siap untuk diterbitkan dengan memanggil fungsi `publish()`. Pada fungsi ini status artikel diperbarui menjadi 'publish' yang sebelumnya statusnya adalah menunggu.

6.6 Implementasi Middleware

Untuk pengontrolan bagaimana respon sistem ketika pengguna memiliki *role* yang berbeda, pada sistem ini diimplementasikan *middleware* yakni salah satu fitur dari *framework* laravel yang dijalankan sebelum *controller* dijalankan, sehingga pengguna hanya dapat menjalankan fungsi pada *role* mereka. *Middleware* yang terdapat pada sistem dapat dilihat pada Tabel 6.5

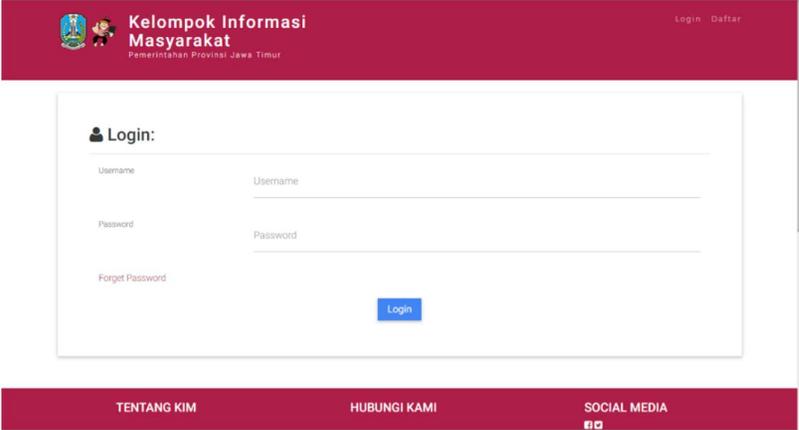
Tabel 6.5 Implementasi middleware

No	Nama Middleware	Nama role	File
1	<i>auth</i>	Pengguna yang telah berhasil <i>login</i>	app/Http/Middleware/Authenticchate.php
2	KOMINFO	Pengguna yang dikenali sebagai KOMINFO setelah berhasil <i>login</i>	app/Http/Middleware/kominfo.php
3	KIM	Pengguna yang dikenali sebagai KIM setelah berhasil <i>login</i>	app/Http/Middleware/validateKim.php
4	<i>guest</i>	Pengguna yang tidak dikenali oleh sistem / tamu.	app/Http/Middleware/RedirectIfAuthenticate.php

6.7 Implementasi Antarmuka

Implementasi antarmuka sistem informasi KIM telah dirancang pada sub bab 5.5. Pada sub bab ini ditampilkan hasil implementasi dari perancangan tersebut.

6.7.1 Halaman *Login*



The screenshot shows the login interface for the Kelompok Informasi Masyarakat (KIM) website. The header is dark red with the KIM logo and name on the left, and 'Login' and 'Daftar' links on the right. The main content area is white and contains a 'Login:' section with two input fields for 'Username' and 'Password'. Below these fields is a 'Forgot Password' link and a blue 'Login' button. The footer is dark red with three navigation links: 'TENTANG KIM', 'HUBUNGI KAMI', and 'SOCIAL MEDIA'.

Gambar 6.2 Implementasi antarmuka *login*

Pada Gambar 6.2 adalah hasil implementasi dari perancangan antarmuka pada Gambar 5.10. Pada halaman ini digunakan pengguna untuk melakukan proses *login*. Ketika pengguna lupa akan *password*, pengguna dapat melakukan *reset password* dengan mengklik *link* lupa *password* yang berada dibawah *form login*. Ketika pengguna belum memiliki akun, pengguna dapat melakukan pendaftaran dengan mengklik link *daftar* di pojok kanan atas. Untuk kembali ke halaman home, pengguna bisa mengklik logo KIM yang berada di pojok kiri atas. Pada halaman *login* juga dapat menampilkan pesan ketika *login* gagal. Ketika proses *login* berhasil, pengguna akan diarahakan ke halaman *dashboard*.

6.7.2 Halaman *Home*

Implementasi antarmuka home dapat dilihat pada Gambar 6.3. Pada halaman ini ditampilkan beberapa artikel. Terdapat slider pada kotak sebelah kiri dan dua kotak sebelah kanan hanya berupa gambar dan judulnya. Pada tampilan kotak ditampilkan artikel-artikel terpopuler. Pada tiap-tiap artikel dapat dilihat KIM yang membuat artikel dan asal KIM tersebut. Terdapat navigasi kategori untuk melakukan pencarian artikel berdasarkan kategori-kategori tertentu. Untuk membaca lebih lengkap dari artikel terkait, pengguna bisa mengklik judul dari artikel atau mengklik link *baca selengkapnya*.


Kelompok Informasi Masyarakat
Kemendagri, Kemendikbud, Kemendagri

[Login](#) [Daftar](#)



Petis Pamekasan



Dodol Temulawak Enak Khas Madiun

Kategori

[Info Kim](#) - [Wisata Kim](#) - [Kuliner Kim](#) - [Inspirasi Kim](#) - [Pengaduan Kim](#) - [Galeri Kim](#) - [Jaringan KIM](#)

Agenda

Artikel terbaru



Pengusaha Sukses Indonesia Asal Ponorogo

2017-02-14

kota Ponorogo melahirkan usahawan sukses, diarahkan Agudakendri, dan ...

[Baca selengkapnya](#)



Tanam Apotik Hidup Di Pekarang Rumah Warga Ngunut Jadi Peduli Kesehatan

2017-02-14

Kesehatan adalah harta paling berharga, segala aktivitas akan dicatat dibukukan bila jernih dan sipih terlah. Lebih berketuhanan kesehatan harus selalu diperhatikan. Mesti tak mengalamii sakit, badan harus dijaga agar tetap sehat. ...

[Baca selengkapnya](#)



Dodol Temulawak Enak Khas Madiun

2017-02-14

Dengan banyanya khasiat Temulawak untuk kesehatan, UPPKS Manggis Kamlaten aningsi berinisiasi dengan membuat Dodol Temulawak. Dodol Temulawak buatan Buibu ENK yang terpacung dalam UPPKS Manggis yang berakumulasi di Jalan Mendi Ko. ...

[Baca selengkapnya](#)



ASAL USUL TOPENG MALANGAN

2017-02-09

Wayang Topeng Malangan merupakan tradisi budaya dan religiusitas masyarakat Jawa selatan. Komposisi kerajinan yang dipamerkan oleh Gagayana semasa abad ke-18 M. Inibisa pernah tertitika. ...

[Baca selengkapnya](#)



WISATA ALAM "WATU KONANG"

2017-02-09

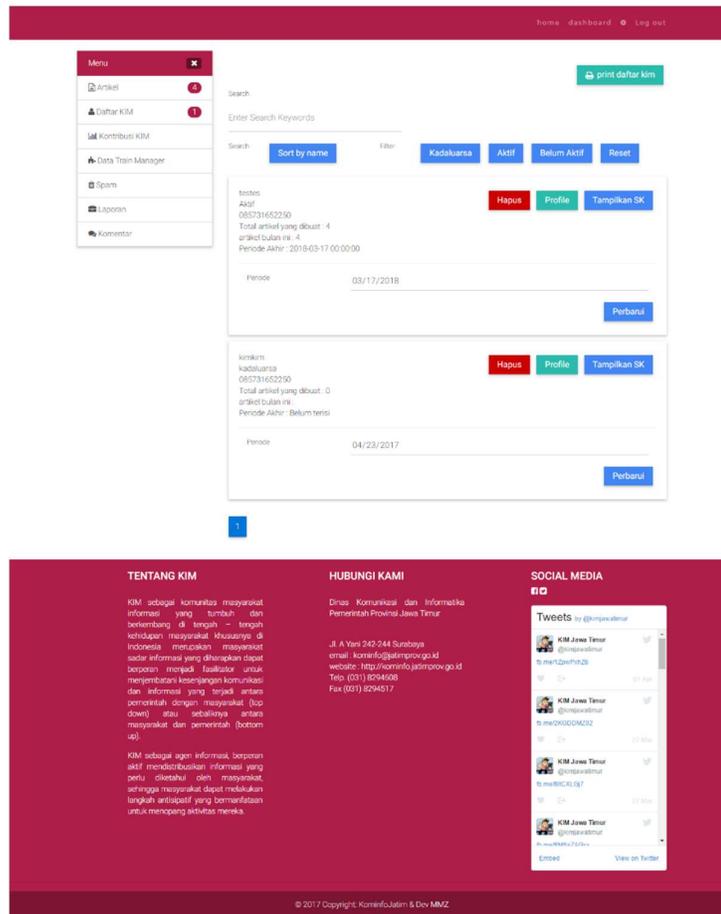
Dikenal satu potensi wisata di desa Karangrejo adalah Watu Konang yang muaranya sangat indah. Airnya yang haribagi sangat baik untuk air minum. Banyak dikalori dari tempat wisata yang lain seperti Sumb. ...

[Baca selengkapnya](#)

TENTANG KIM
HUBUNGI KAMI
SOCIAL MEDIA

Gambar 6.3 Implementasi antarmuka home

6.7.3 Halaman *Dashboard* (KOMINFO) Daftar KIM



Gambar 6.4 Implementasi antarmuka *dashboard* (KOMINFO)

Pada Gambar 6.4 adalah hasil implementasi dari perancangan antarmuka *dashboard* yang bisa dilihat pada Gambar 5.1. Terdapat navigasi yang berada di samping kiri untuk mengakses fitur-fitur lain dari *website* KIM sesuai otorisasi pengguna. Untuk penampilan data digunakan *list*. Implementasi menggunakan *list* dibanding penggunaan tabel dikarenakan untuk memberikan tampilan yang lebih *responsive* pada pengguna *mobile*. Terdapat fitur-fitur di tiap-tiap *list* yang digunakan untuk melakukan operasi pada data terkait. Di antar *list* terdapat fitur *search* dan *filter* untuk mempermudah pencarian dan seleksi data. Di bagian bawah *list* terdapat pagination yang digunakan untuk menampilkan data yang lain, karena pada satu halaman terdapat batasan *list* yang ditampilkan sehingga halaman tidak terlalu panjang. Berikut penjelasan tombol-tombol yang terdapat di tiap tiap *list*.

a) Tombol *Setujui*

Digunakan untuk menyetujui KIM yang baru mendaftar.

b) Tombol *Delete*

Digunakan untuk menghapus KIM terkait. Ketika pengguna menekan tombol *delete* akan ditampilkan jendela konfirmasi untuk mengkonfirmasi operasi terkait.

c) Tombol *Profile*

Digunakan untuk menampilkan informasi KIM terkait. KOMINFO memiliki hak akses untuk masuk ke setiap *profile* pengguna.

d) Tombol Tampilkan SK

Digunakan untuk menampilkan SK yang diupload oleh KIM terkait.

Pada tiap tiap *list* juga terdapat informasi dasar terkait KIM tersebut dan terdapat *form* untuk mengubah masa aktif suatu KIM. ketika KIM melebihi masa aktif, akan ditampilkan status “kadaluarsa”, ketika belum disetujui oleh KOMINFO maka status yang ditampilkan berupa “belum aktif”, ketika sudah disetujui dan masa aktif belum habis maka status yang ditampilkan berupa “aktif”.

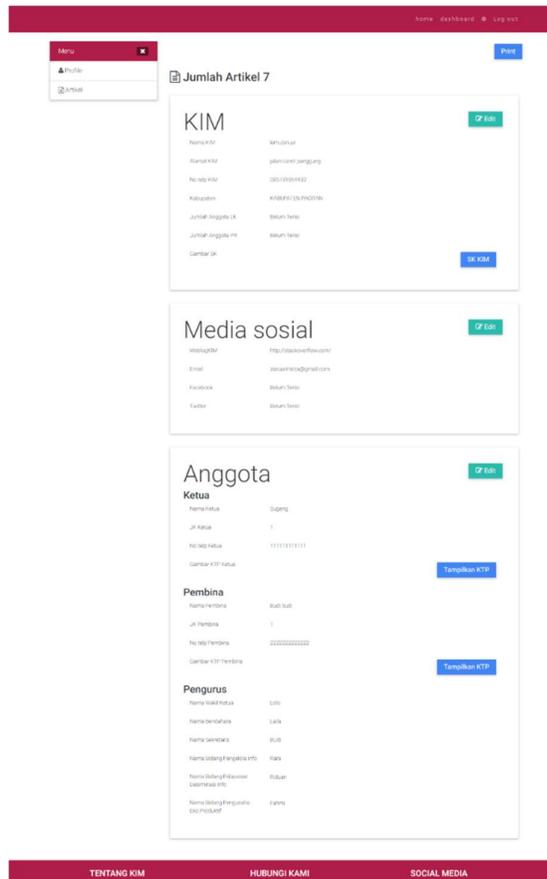
6.7.4 Halaman Baca Artikel Lengkap

Pada Gambar 6.5 adalah implementasi dari halaman baca artikel lengkap. Pada halaman ini ditampilkan konten lengkap dari artike terkait. Di bawah artikel terdapat *form* komentar untuk melakukan komentar pada artikel terkait. Di sisi kanan terdapat navigasi yang berisikan artikel terpopuler dan artikel terbaru. Di atas artikel terdapat navigasi kategori untuk membuka artikel berdasarkan kategorinya. Terdapat informasi KIM yang membuat artikel beserta asal daerahnya dan juga informasi kategori dari artikel terkait.



Gambar 6.5 Implementasi antarmuka baca artikel Lengkap

6.7.5 Halaman Profile



Gambar 6.6 Implementasi antarmuka *profile*

Pada Gambar 6.6 adalah hasil implementasi antarmuka *profile*. Halaman ini digunakan untuk menampilkan *profile* dari suatu KIM. Terdapat tiga bagian informasi KIM yakni informasi dasar KIM, informasi media sosial KIM, dan informasi keanggotaan. Setiap bagian terdapat tombol *edit* untuk memudahkan pengguna ketika ingin melakukan perubahan. Karena keterbatasan layar vertical, maka disediakanlah tombol *edit* untuk setiap bagian. Terdapat informasi jumlah artikel yang dibuat oleh KIM terkait di atas informasi *profile* KIM. Pengguna dapat mencetak informasi-informasi KIM dengan mengklik tombol print di bagian atas samping kanan.

6.7.6 Halaman *Dashboard* Artikel (KOMINFO)

Hasil implementasi antarmuka *dashboard* artikel (KOMINFO) dapat dilihat pada Gambar 6.7. Pada halaman ini terdapat data data artikel dalam bentuk *list*. Setiap *list* nya memiliki fungsi fungsi dalam bentuk tombol yang digunakan untuk

melakukan operasi terhadap artikel terkait. Berikut penjelasan tombol-tombol pada *list* artikel.

a) Tombol *publish*

Tombol *publish* digunakan untuk mempublikasi artikel terkait. Ketika artikel dipublikasi, artikel terkait juga dimasukkan ke dalam data train pada kategori bukan *spam*.

b) Tombol *Delete*

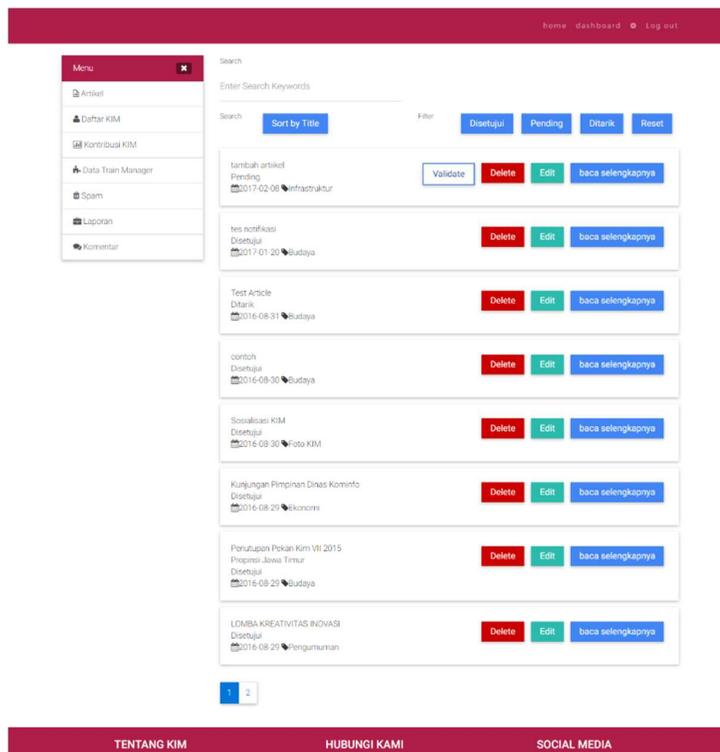
Tombol *delete* digunakan untuk menghapus artikel terkait. Bisa dilakukan setelah atau sebelum artikel dipublikasikan. Ketika pengguna menekan tombol *delete*, akan ditampilkan jendela konfirmasi untuk mengkonfirmasi operasi hapus artikel.

c) Tombol *Edit*

Tombol *edit* digunakan untuk mengubah artikel terkait. Ketika pengguna menekan tombol *edit*, pengguna akan diarahakan ke halaman *edit* artikel.

d) Tombol Baca selengkapnya

Tombol *baca selengkapnya* digunakan untuk masuk ke halaman baca artikel lengkap sesuai artikel yang dipilih.



Gambar 6.7 Implementasi antarmuka *dashboard* artikel (KOMINFO)