

Bab IV Analisis & Perancangan Komponen E-Library

4.1 Deskripsi Global Perangkat Lunak

Perangkat lunak komponen *e-library* merupakan suatu perangkat lunak aplikasi berbasis web berupa komponen dari CMS joomla yang memiliki beberapa fungsi atau layanan seperti yang dimiliki oleh sebuah perpustakaan digital. Perangkat lunak ini dapat digunakan antara lain untuk melakukan pengarsipan secara digital terhadap file-file elektronik yang sebagian besar diperoleh dari internet, menyediakan fasilitas bagi pengguna untuk mengakses file-file yang merupakan koleksi dari perpustakaan. Perangkat lunak ini diharapkan dapat memudahkan *user* dalam mengorganisasi file-file, melakukan pengaksesan, pengelolaan dan pencarian file-file. File-file tersebut bisa berupa artikel-artikel atau jurnal elektronik, pustaka elektronik (*ebook*), file – file multimedia dan lain-lain.

Perangkat lunak komponen *e-library* ini tidak dapat berdiri sendiri untuk dijalankan sebagai suatu program aplikasi. Perangkat lunak ini merupakan ekstensi dari CMS joomla yang berupa komponen sehingga untuk dapat menjalankannya memerlukan suatu aplikasi *web* atau *website* yang menggunakan CMS joomla. Apabila sudah di-*upload* di internet, perangkat lunak ini nantinya dapat di-*download* untuk kemudian di-*install* pada *website-website* yang menggunakan joomla yang ada di internet.

4.1.1 Perspektif Produk

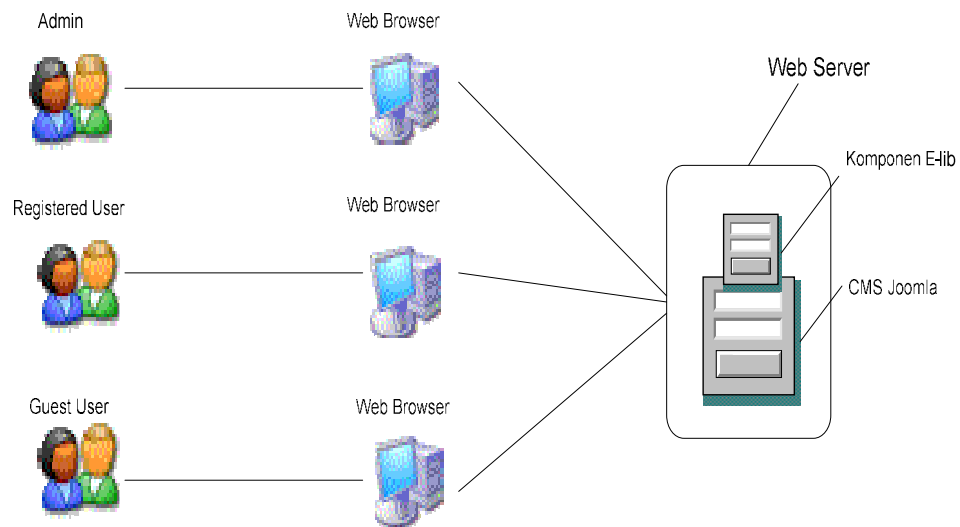
Perangkat lunak komponen *e-library* ini merupakan suatu ekstensi bagi joomla yang dapat menambah fungsionalitas dari CMS joomla. Perangkat lunak ini nantinya akan di-*install* pada *website* yang menggunakan CMS joomla. Jadi

perangkat lunak ini tidak dapat berdiri sendiri tetapi merupakan aplikasi yang terintegrasi dengan CMS Joomla.

Sebagian fasilitas dari komponen *e-library* ini dapat digunakan oleh setiap *user* yang mengunjungi situs tetapi tidak mendaftarkan diri sebagai anggota, sedangkan bagi *user* yang telah melakukan registrasi sebagai anggota dapat menggunakan lebih banyak fasilitas yang tersedia dalam komponen misalnya fasilitas pencarian artikel/pustaka, pengaksesan terhadap keseluruhan artikel/pustaka dan fasilitas untuk mengirimkan artikel/ pustaka elektronik.

Komponen ini merupakan ekstensi dari CMS Joomla dan dibuat dengan menggunakan Joomla sehingga *software* yang digunakan untuk pembuatan komponen ini juga menyesuaikan dengan *software* yang digunakan dalam pengembangan Joomla yaitu menggunakan bahasa pemrograman PHP, basis data MySQL dan *webserver Apache*. Karena merupakan ekstensi dari CMS Joomla, komponen ini nantinya dapat di-*download* dan dapat di-*install* sebagai komponen di setiap *website* yang menggunakan Joomla.

Kaitan antara perangkat lunak komponen *e-library* dengan Joomla, *webserver* dan *web browser* dapat dilihat pada Gambar 11. Komponen *e-library* terintegrasi dengan CMS dan hanya dapat berjalan di atas CMS Joomla dan keduanya berjalan di atas sebuah *web server*. Untuk dapat mengakses komponen tersebut, pengguna harus menggunakan *web browser* yang terkoneksi ke internet. Komponen *e-library* dipasang (di-*install*) pada suatu *web* aplikasi (*website*) yang menggunakan Joomla sebagai suatu *add-on* atau fungsi tambahan yang diaplikasikan pada *website* tersebut.



Gambar 11 Keterkaitan Komponen E-library dengan CMS Joomla, Web Server dan Web Browser

Aplikasi komponen *e-library* ini nantinya dibuat dengan menyediakan 3 jenis halaman akses, yaitu :

1. Halaman admin, yang digunakan untuk melakukan kegiatan administrasi sistem perpustakaan secara keseluruhan, untuk melakukan manajemen data yang ada di dalam basis data.
2. Halaman *guest user*, digunakan untuk pengunjung biasa dalam artian bukan anggota perpustakaan. Halaman ini dapat diakses secara bebas oleh siapa saja yang membuka alamat situs perpustakaan.
3. Halaman *registered user*/anggota perpustakaan, merupakan halaman khusus yang hanya bisa diakses oleh pengguna yang sudah menjadi anggota perpustakaan. Pengunjung biasa / *guest user* tidak akan bisa mengakses halaman ini karena untuk bisa memasuki halaman anggota ini pengunjung harus mempunyai *login name* dan *password*.

Seorang pengunjung biasa hanya bisa melakukan pencarian buku atau melihat informasi buku, sedangkan anggota dapat melakukan pencarian buku, mengakses

atau membaca buku tersebut serta mengirimkan file-file elektronik (bisa berupa artikel, jurnal, *ebook*, dan lain-lain), sedangkan halaman administrator hanya berhak diakses oleh admin atau seseorang yang telah memiliki kewenangan penuh terhadap sistem perpustakaan.

4.1.2 Fungsionalitas Perangkat Lunak

4.1.2.1 Kebutuhan Fungsional

Perangkat lunak komponen *e-library* ini memiliki beberapa kemampuan / fungsi atau spesifikasi kebutuhan perangkat lunak. Fungsi yang dimiliki ini dapat dikategorikan ke dalam dua kategori yaitu kebutuhan fungsional dan kebutuhan non fungsional. Kebutuhan fungsional merupakan layanan-layanan yang bisa dilakukan sistem. Kebutuhan fungsional dari sistem perangkat lunak komponen *e-library* didasarkan pada definisi, tujuan dan layanan *e-library* adalah sebagai berikut :

1. Fasilitas bagi pengguna untuk melakukan pengarsipan terhadap file – file *e-library* yang meliputi penambahan dan pengklasifikasian file-file *e-library* ke dalam beberapa kategori dan sub kategori.
2. Kemampuan sistem untuk melakukan pemutakhiran (*update*) file-file *e-library* yang telah dimiliki.
3. Fasilitas untuk menarik file/dokumen elektronik dari publikasi dan fasilitas untuk menghapus file-file yang sudah tidak diperlukan lagi hingga file tersebut tidak tersimpan lagi dalam basis data *e-library*.
4. Fasilitas bagi anggota perpustakaan/ *e-library* untuk dapat mengakses file-file yang terdapat dalam perpustakaan.
5. Kemampuan untuk mencari file – file yang ada dalam *e-library* dengan berdasar pada nama pengarang, judul/topik, pengirim *file*, serta kata kunci dalam isi artikel.

6. Fasilitas bagi anggota perpustakaan dapat melakukan *download* terhadap file/artikel/pustaka elektronik pada *e-library*.
7. Fasilitas untuk menentukan hak akses bagi pengguna *e-library*. Ada 3 macam hak akses yang bisa diberikan kepada pengguna perpustakaan, yaitu hak akses sebagai *guest user* (pengguna yang belum melakukan registrasi sebagai anggota *e-library*) dan hak akses sebagai *registered user* (pengguna yang telah melakukan registrasi sebagai anggota *e-library*) serta hak akses sebagai administrator .
8. Fasilitas bagi non anggota perpustakaan untuk mendaftar menjadi anggota perpustakaan elektronik tersebut.
9. Fasilitas login untuk anggota perpustakaan maupun untuk admin.
10. Fasilitas bagi anggota perpustakaan untuk mengirimkan file atau artikel elektronik ke sistem perangkat lunak komponen *e-library*.
11. Fasilitas untuk publikasi *file e-library* dimana file atau artikel yang bisa diterbitkan dan ditampilkan di situs serta diakses di internet hanya file atau artikel yang sesuai dengan kriteria yang telah ditentukan oleh administrator.

4.1.2.2 Kebutuhan Non Fungsional

Selain memiliki beberapa spesifikasi kebutuhan fungsional seperti telah dijelaskan pada bagian diatas, sistem perangkat lunak komponen *e-library* juga memiliki beberapa kebutuhan non fungsional. Kebutuhan non fungsional dari komponen *e-library* adalah :

- Perangkat lunak komponen *e-library* ini harus bisa dijalankan oleh beberapa *software web browser* diantaranya Internet Explorer, Mozilla Firefox dan Netscape Navigator.
- Komponen *e-library* membutuhkan CMS joomla untuk bisa dijalankan.

4.1.3 Karakteristik Pengguna

Karakteristik pengguna ini menggambarkan siapa saja yang menjadi pengguna dari perangkat lunak komponen *e-library*. Karakteristik pengguna sistem komponen *e-library* dapat dilihat pada Tabel 1. berikut ini.

Tabel 1 Karakteristik Pengguna E-Library

Pengguna	Deskripsi Pengguna	Tanggung Jawab	Hak Akses	Tingkat Pendidikan	Tingkat Keterampilan
<i>Guest User</i>	Pengunjung situs yang bukan anggota <i>e-library</i>	-	Akses <i>homepage e-library</i> , akses katalog <i>elibrary</i> (<i>browse artikel</i>)	Sangat Beragam	Sangat Beragam
<i>Registered User</i>	Pengguna <i>e-library</i> yang sudah terdaftar sebagai anggota	-	Akses terhadap seluruh file dalam arsip <i>elibrary</i> , pencarian, <i>upload</i> dan <i>download file</i>	Sangat Beragam	Sangat Beragam
<i>Administrator</i>	Pengguna <i>e-library</i> yang bertugas mengelola dan memiliki kewenangan tertinggi atas semua fasilitas pada <i>e-library</i> .	Beroperasinya situs, hak akses pengguna, keamanan sistem, pengelolaan arsip <i>e-library</i> .	Mengaktifkan atau mematikan <i>server</i> , instalasi sistem, pemberian hak akses, tambah, update, hapus, <i>publish data e-library</i>	Diploma III (minimum)	Menguasai keahlian web master, pemrograman web, & jaringan internet.

4.1.4 Batasan- Batasan

Perangkat lunak komponen *e-library* yang akan dibangun memiliki beberapa batasan yang nantinya akan membatasi lingkup pengembangan perangkat lunak. Beberapa batasan tersebut adalah :

- Aplikasi komponen *e-library* ini bukan sebuah perpustakaan konvensional yang di-online-kan melainkan lebih menyerupai sebuah arsip digital dimana salah satu tujuan utamanya adalah memudahkan pengelolaan dan pengaksesan file-file elektronik yang sebagian besar diperoleh dari internet.
- Perangkat lunak komponen *e-library* tidak dapat digunakan untuk mengakses file-file dokumen yang berformat linux atau unix karena adanya perbedaan sistem file.

4.1.5 Asumsi-Asumsi

Sehubungan dengan pembangunan dan implementasi perangkat lunak aplikasi komponen *e-library* , ada beberapa asumsi yang digunakan yaitu sebagai berikut :

1. Sistem operasi yang akan digunakan sudah tersedia dan dapat berjalan dengan baik.
2. Perangkat lunak CMS joomla dan *web server* yang akan digunakan sudah tersedia dan dapat beroperasi dengan tanpa masalah.
3. File-file, artikel atau pustaka elektronik yang digunakan dalam *e-library* bersifat *free* atau dapat di-download secara gratis dari internet dan tidak ada permasalahan mengenai hak cipta.

4.2 Model Use Case

4.2.1 Definisi Aktor

Tabel 2 Definisi Aktor

No.	Aktor	Deskripsi
1.	<i>Administrator</i>	Pengguna perangkat lunak yang memiliki kewenangan penuh untuk melakukan instalasi, pengaktifan sistem, tambah data file elektronik, hapus data, update, publish serta pemberian hak akses terhadap <i>user</i> .
2.	<i>Guest User</i>	Pengguna perangkat lunak yang hanya melakukan kunjungan terhadap situs dan hanya dapat melakukan akses <i>homepage</i> dan melihat melakukan <i>browse</i> artikel-artikel <i>e-library</i> .
3.	<i>Registered User</i>	Pengguna perangkat lunak yang telah melakukan registrasi sebagai anggota <i>e-library</i> dan memiliki hak untuk melakukan pencarian dan akses terhadap seluruh file artikel/pustaka pada <i>e-library</i> dan hak untuk melakukan <i>upload</i> dan <i>download file</i> .

4.2.2 Definisi Use Case

Tabel 3 Definisi Use Case

No.	Use Case	Deskripsi
1.	Tambah Data E-lib	<i>Administrator</i> melakukan proses penambahan data file <i>e-library</i> ke basis data.
2.	Hapus Data E-lib	<i>Administrator</i> melakukan proses penghapusan file/ artikel/pustaka pada <i>e-library</i> .
3.	Update E-lib	<i>Administrator</i> melakukan proses

No.	Use Case	Deskripsi
		<i>update</i> data file/artikel/pustaka elektronik
4.	Tambah Kategori	<i>Administrator</i> melakukan proses penambahan kategori untuk penggolongan file/artikel/pustaka <i>e-library</i> .
5.	Mengatur hak akses	<i>Administrator</i> memberikan wewenang/hak akses kepada pengguna sistem.
6.	Akses file <i>e-lib</i>	<i>User</i> melakukan akses/view file/artikel/pustaka yang terdapat pada <i>e-library</i>
7.	Mencari file <i>e-lib</i>	<i>User</i> melakukan pencarian terhadap file/artikel/pustaka yang diinginkan dengan memasukkan <i>keyword</i> .
8.	<i>Download file e-lib</i>	<i>User</i> melakukan proses penyimpanan file/data <i>e-lib</i> ke PC <i>User</i> .
9.	Publish data <i>e-lib</i>	<i>Administrator</i> melakukan proses <i>publishing</i> data <i>e-lib</i> .
10.	Registrasi Anggota	<i>User</i> melakukan proses pendaftaran agar memiliki kewenangan sebagai <i>registered user</i> .
11.	Login Anggota	<i>User</i> melakukan pengisian <i>username</i> dan <i>password</i> agar bisa memperoleh hak-hak yang dimiliki oleh <i>registered user</i> .

4.2.3 Skenario Use Case

1.Skenario Use Case Tambah Data E-lib

Identifikasi		
Nomor	1	
Nama	Tambah Data E-lib	
Tujuan	Menambahkan data file artikel e-library ke dalam basis data e-library	
Aktor	User , Administrator	
Skenario Utama		
Kondisi awal	Menampilkan menu utama modus registered user	
Aksi Aktor		Reaksi Sistem
1	User memilih menu browse kategori lalu pilih menu posting artikel	
2		Menampilkan fasilitas tambah data artikel dimana terdapat formulir input data-data artikel
3	User mengisi data artikel dan melakukan upload artikel	
4		Menampilkan pesan data sudah dikirim
5	Admin login pada back-end lalu pilih menu components lalu pilih menu approval	
6		Menampilkan daftar artikel yang membutuhkan persetujuan admin untuk di-publish.

Aksi Aktor		Reaksi Sistem
7.	Admin memberikan persetujuan untuk artikeal yang sesuai dengan criteria	
8		Pada front-end di bagian menu browse category, muncul artikel yang baru ditambahkan.
Skenario Alternatif		
Aksi Aktor		Reaksi Sistem
3	Memasukkan data artikel yang akan ditambahkan	
4		Muncul pesan bahwa data sudah ada dalam basis data
Kondisi akhir	Menampilkan kembali menu utama	

2. Skenario Use Case Hapus data E-lib

Identifikasi		
Nomor		2
Nama		Hapus Data E-lib
Tujuan		Menghilangkan data artikel yang terdapat dalam basis data e-library.
Aktor		Administrator
Skenario Utama		
Kondisi awal	Menampilkan menu utama modus administrator	
Aksi Aktor		Reaksi Sistem
1	Administrator memilih menu component lalu pilih menu..list posting artikel	

Aksi Aktor		Reaksi Sistem
		Menampilkan daftar artikel-artikel yang ada dalam basis data.
3	Menghapus data artikel dengan cara klik pada icon delete	
4		Muncul pesan bahwa data telah dihapus Dan dalam daftar artikel tidak ada lagi data artikel yang dihapus tersebut
5.	Admin masuk ke halaman utama front-end dan memilih menu browse artikel	.
		Artikel yang dihapus tidak muncul lagi pada tampilan browse artikel
Skenario Alternatif		
Aksi Aktor		Reaksi Sistem
3.	mengubah status artikel dari publish menjadi unpublish	
4.	klik pada icon save	
5.		Status artikel berubah menjadi unpublish
6.	admin masuk ke homepage pada front-end dan memilih menu browse artikel	
7.		Artikel tidak muncul lagi pada tampilan browse artikel
8.	Admin keluar dari menu list posting artikel pada back-end	
Kondisi akhir	Menampilkankembali menu utama back-end (modus administrator)	

3. Skenario Use Case Update E-Lib

Identifikasi		
Nomor	3	
Nama	Update Data E-lib	
Tujuan	Melakukan update terhadap data –library yang sudah di-publish	
Aktor	Administrator	
Skenario Utama		
Kondisi awal	Menampilkan menu utama modus administrator	
Aksi Aktor		Reaksi Sistem
1	Stelah login, administrator memilih menu component lalu pilih list posting artikel	
2		Menampilkan daftar artikel yang telah di-publish
3	Memilih artikel yang akan dihapus lalu klik pada icon delete	
4		Menampilkan pesan bahwa file sudah dihapus dan file tidak muncul lagi pada daftar artikel
5.	Administrator melakukan upload file artikel baru yang menggantikan file yang dihapus tersebut	
6		Menampilkan pesan bahwa file sudah ter-upload
7.	Administrator melakukan publish terhadap artikel pengganti tersebut melalui menu pada banck-end	
8.		Artikel artikel yang sudah di-publish muncul pada tampilan menu browse artikel pada front-end.

Skenario Alternatif		
Aksi Aktor		Reaksi Sistem
3.	Administrator memilih file artikel untuk di-unpublish	Artikel yang di-unpublish tidak muncul lagi pada tampilan menu browse artikel pada front-end.
4.	Administrator keluar dari menu list posting artikel	
Kondisi akhir		Sistem menampilkan menu utama back-end

4. Skenario Use Case Tambah Kategori

Identifikasi		
Nomor		4
Nama		Tambah Kategori
Tujuan		Menambah jenis kategori yang terdapat pada daftar kategori artikel
Aktor		Administrator
Skenario Utama		
Kondisi awal		Menampilkan menu utama modus administrator
Aksi Aktor		Reaksi Sistem
1	Administrator memilih menu kategori dan sub kategori	
2		Menampilkan fasilitas untuk tambah kategori

Aksi Aktor		Reaksi Sistem
3	Melakukan proses tambah kategori dengan mengisi kategori baru pada form tambah kategori	
4		Menampilkan pesan bahwa kategori sudah ditambahkan.
5	Masuk ke halaman front-end dan memilih menu browse artikel	
		Kategori baru muncul pada tampilan menu browse artikel
Skenario Alternatif		
Aksi Aktor		Reaksi Sistem
4.		Muncul pesan bahwa kategori ada
5.	Admin keluar dari menu kategori dan subkategori	
Kondisi akhir	Sistem menampilkan kembali menu utama back-end	

5. Skenario Use Case login

Identifikasi	
Nomor	5
Nama	Login anggota
Tujuan	Masuk ke halaman modus registered user pada front-end
Aktor	Registered user
Skenario Utama	
Kondisi awal	Menampilkan menu utama homepage pada front-end
Aksi Aktor	
Reaksi Sistem	
1	Registered user melakukan login dengan mengisi username dan password pada form login

Aksi Aktor		Reaksi Sistem
2		Sistem menampilkan pesan bahwa login telah berhasil.
3		Muncul tampilan halaman utama modus registered user
Skenario Alternatif		
Aksi Aktor		Reaksi Sistem
2		Sistem menampilkan pesan bahwa login gagal
Kondisi akhir	Sistem menampilkan kembali halaman utama homepage pada front-end	

6. Skenario Use Case Mencari Data

Identifikasi		
Nomor		6
Nama		Mencari data
Tujuan		Melakukan pencarian dengan tujuan menemukan file yang dicari yang sesuai dengan kriteria pencarian yang diberikan.
Aktor		Registered User
Skenario Utama		
Kondisi awal		Menampilkan menu utama modus registered user di front-end
Aksi Aktor		Reaksi Sistem
1	User melakukan klik pada menu pencarian	
2		Menampilkan halaman pencarian e-library
3.	User memilih kategori dan subkategori dari file yang dicari	
4.	User memilih kriteria pencarian dan mengisikan kata kunci untuk pencarian lalu tekan tombol cari	

Aksi Aktor		Reaksi Sistem
		Sistem akan menampilkan hasil pencarian
Skenario Alternatif		
3	User memilih kategori dari file yang ingin dicari lalu tekan tombol cari	
		Sistem akan menampilkan hasil pencarian
4.	Kembali ke menu utama	
Kondisi akhir	Sistem menampilkan kembali menu utama modus registered user	

7. Skenario Use case Browse Data

Identifikasi		
Nomor	7	
Nama	Browse Data E-lib	
Tujuan	Melakukan penelusuran / browsing terhadap artikel-artikel yang terdapat dalam basis data e-library.	
Aktor	User (guest user, registered user atau administrator)	
Skenario Utama		
Kondisi awal	Menampilkan menu utama modus guest user	
Aksi Aktor		Reaksi Sistem
1	User memilih menu browse artikel	
2		Sistem menampilkan daftar artikel-artikel yang terdapat di dalam basis data e-library
3	Memilih kategori dan sub kategori dari artikel – artikel yang diinginkan	

Aksi Aktor		Reaksi Sistem
4		Menampilkan daftar artikel yang sesuai dengan kategori dan sub kategori yang telah dipilih.
5	User kembali ke menu utama pada homepage	
Kondisi akhir		Menampilkan kembali menu utama homepage pada front-end.

8. Skenario Use Case Download File E-Lib

Identifikasi		
Nomor		8
Nama		Download file e-lib
Tujuan		Melakukan download terhadap fil eartikel yang diinginkan agar artikel tersebut dapatdibaca atau disimpan disimpan ke tempat yang diinginkan.
Aktor		<i>Registered User</i>
Skenario Utama		
Kondisi awal		Menampilkan menu utama modus registered user di front-end
Aksi Aktor		Reaksi Sistem
1	Registered user melakukan klik pada menu browse artikel	
2		Menampilkan halaman browse artikel
3.	Registered user memilih file artikel yang ingin di-download dengan cara klik dua kali pada artikel tersebut	
4.		Menampilkan halaman download file/artikel

Aksi Aktor		Reaksi Sistem
5.	Registered user melakukan download dengan klik pada tombol download	
6.		Sistem menampilkan file artikel yang di-download sehingga bisa dibaca atau disimpan
Skenario Alternatif		
1	Registered user terlebih dahulu melakukan pencarian file dengan langkah – langkah seperti pada skenario mencari data	
2.		Sistem akan menampilkan hasil pencarian
3.	Registered user memilih file artikel yang ingin di-download dengan cara klik dua kali pada artikel tersebut	
4		Menampilkan halaman download file
5.	Registered user memilih file artikel yang ingin di-download dengan cara klik dua kali pada artikel tersebut	
6.		Sistem menampilkan file artikel yang di-download sehingga bisa dibaca atau disimpan
Kondisi akhir		Sistem menampilkan kembali menu browse artikel

4.2.4 Diagram Use Case



Gambar 12 Diagram Use Case Komponen E-Library

4.3 Pemodelan Kelas Tahap Analisis

Pemodelan kelas pada tahap analisis bertujuan untuk dapat memperoleh suatu model kelas analisis yang tepat untuk sistem berdasarkan pada spesifikasi kebutuhan sistem dan *use case*. Model yang diperoleh pada tahap ini akan menjadi dasar pada tahap perancangan. Proses pemodelan kelas terdiri dari dua langkah yaitu :

1. Identifikasi kelas
2. Identifikasi interaksi antar kelas

4.3.1 Identifikasi Kelas

Pada tahap ini identifikasi yang dilakukan meliputi identifikasi tipe dari masing-masing kelas. Dalam tahapan analisis kelas-kelas dikelompokkan ke dalam 3 tipe yaitu :

- *Boundary*
Merupakan tipe kelas yang berisi antarmuka sistem dengan pengguna system.
- *Control*
Merupakan tipe kelas yang memiliki fungsionalitas tertentu dan dapat mengontrol interaksi dengan kelas lain.
- *Entity*
Merupakan tipe kelas yang dapat menyimpan data atau informasi ke dalam suatu tempat penyimpanan (*storage*).

Kelas-kelas yang terlibat dalam pengembangan perangkat lunak komponen *e-library* tidak hanya kelas-kelas baru yang dibuat sendiri, melainkan juga meliputi beberapa kelas yang berasal dari CMS Joomla.

Hasil identifikasi kelas dalam tahap analisis untuk perangkat lunak *e-library* terbagi ke dalam dua tabel yaitu tabel 4. dan tabel 5. Tabel 4. berisi kelas –kelas

baru yang bukan berasal dari CMS Joomla yang digunakan untuk pengembangan komponen *e-library*, sedangkan tabel 5. berisi daftar kelas-kelas milik CMS Joomla yang terlibat dalam pembangunan komponen *e-library*.

Tabel 4 Identifikasi Kelas-Kelas Baru Tahap Analisis

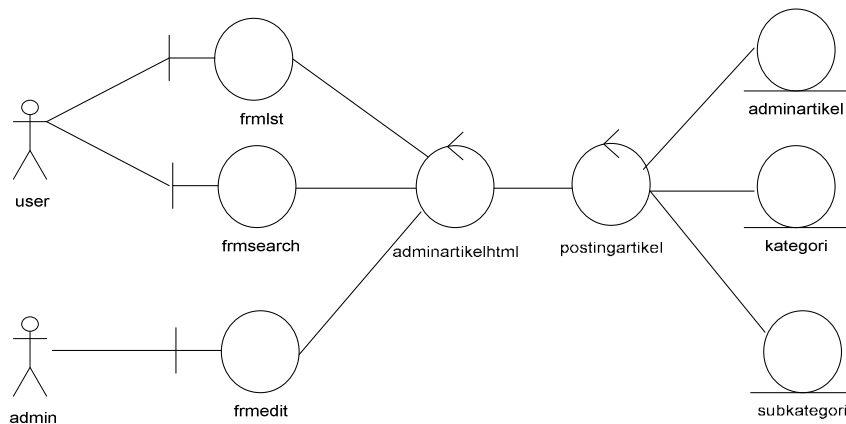
No.	Kelas	Tipe/Jenis Kelas	Definisi
1.	Frmlst	<i>Boundary</i>	Kelas ini berperan untuk menjembatani antara <i>user</i> aplikasi agar <i>user</i> dapat mengakses isi <i>e-library</i> .
2.	Frmedit	<i>Boundary</i>	Kelas ini berfungsi sebagai antarmuka antara <i>admin</i> dan aplikasi agar admin dapat akses dan edit isi <i>e-library</i> .
3.	Frmsearch	<i>Boundary</i>	Kelas ini berfungsi sebagai antarmuka bagi <i>user</i> dalam melakukan pencarian.
4.	admin.artikel.html	<i>Control</i>	Kelas ini antara lain berperan dalam menampilkan daftar artikel atau hasil <i>query</i> terhadap isi basis data .
5.	admin.artikel	<i>Entity</i>	Kelas ini berisi atribut dan <i>method</i> yang berkaitan dengan operasi terhadap artikel.
6.	Kategori	<i>Entity</i>	Kelas ini berisi atribut dan <i>method</i> yang berkaitan dengan operasi terhadap kategori.
7.	subkategori	<i>Entity</i>	Kelas ini berisi atribut dan <i>method</i> yang berkaitan dengan operasi terhadap sub kategori.
8.	postingartikel	<i>Control</i>	Kelas ini berperan dalam koneksi ke basis data, proses <i>upload</i> , <i>download</i> serta pengaksesan artikel yang sudah dikirim.

Tabel 5 Identifikasi Kelas-Kelas dari CMS Joomla pada Tahap Analisis

No.	Kelas	Tipe/Jenis Kelas	Definisi
1.	toolbar.artikel. html	<i>Boundary</i>	Kelas ini berperan untuk menampilkan menu-menu yang berkaitan dengan proses menampilkan data
2.	Index	<i>Control</i>	Kelas ini berfungsi untuk inialisasi aplikasi <i>e-library</i> .
3.	pagination	<i>Boundary</i>	Kelas ini berperan dalam menyediakan antarmuka untuk paginasi konten.
4.	database	<i>Control</i>	Kelas ini antara lain berperan dalam koneksi ke <i>server database</i> , dan berperan dalam pengolahan data-data dalam basis data joomla.
5.	Object	<i>Control</i>	Kelas ini berperan dalam memberikan dukungan dari <i>core joomla</i> untuk beberapa fungsi dalam <i>e-library</i> misalnya untuk <i>error message</i> .
6.	Html	<i>Boundary</i>	Kelas ini menyediakan antarmuka untuk <i>parsing</i> dan menampilkan dokumen html.
7.	mainframe	<i>Control</i>	Kelas ini berperan dalam proses inialisasi session <i>user</i> .
8.	JDatabase	<i>Entity</i>	Kelas ini berfungsi untuk meyimpan data-data yang berkaitan dengan operasi terhadap <i>user</i> .
9.	JApplication	<i>Control</i>	Kelas ini berperan dalam pengambilan <i>session</i>

4.3.2 Interaksi antar Kelas

Diagram interaksi antar kelas pada tahap analisis dapat dilihat pada Gambar 13. Diagram ini hanya menggambarkan semua kelas-kelas baru yang belum ada pada CMS joomla.



Gambar 13 Interaksi antar Kelas

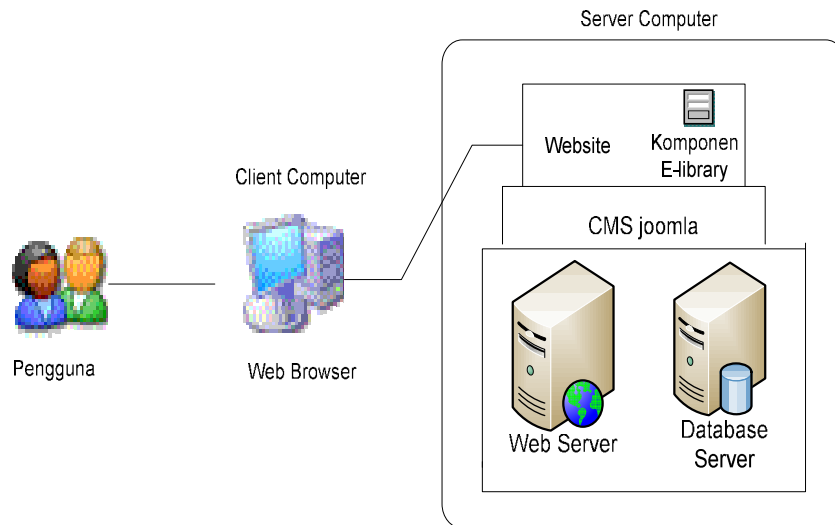
Pada Gambar 13. terlihat bahwa *user* mengakses artikel dengan melalui antar muka *frm1st* dan antarmuka *frmsearch*, sedangkan *admin* mengakses atau melakukan edit artikel dengan melalui antarmuka yang terdapat pada kelas *frmedit*. Perintah pengaksesan atau pencarian yang diterima oleh kelas *boundary frm1st* dan *frmsearch* maupun perintah dari admin yang mditerima melalui keals *frmedit* akan diteruskan ke kelas *control adminartikelhtml*. Kelas *control* ini yang bertanggungjawab terhadap pengaksesan entity baik entity artikel, kategori maupun sub kategori.

4.4 Perancangan

Tahap perancangan merupakan salah satu tahap dalam pembangunan perangkat lunak yang dilaksanakan setelah tahap analisa. Perancangan yang dilakukan meliputi perancangan arsitektur, perancangan kelas diagram, perancangan data dan perancangan antarmuka.

4.4.1 Perancangan Arsitektur Sistem

Perancangan arsitektur yang dilakukan adalah merancang struktur dari sistem yang terdiri dari beberapa komponen atau elemen perangkat lunak, hubungan antar elemen perangkat lunak serta antarmuka antara elemen dengan pengguna.



Gambar 14 Arsitektur Sistem

Pada Gambar 14. terlihat bahwa arsitektur sistem yang digunakan secara garis besar adalah arsitektur *client server*, dimana pengguna (*client computer*) mengakses *website* dengan menggunakan perangkat lunak *web browser*. Perangkat lunak aplikasi komponen *e-library* bisa diakses melalui *website* yang berbasis CMS joomla. Untuk dapat menjalankan CMS joomla terlebih dahulu harus menyediakan perangkat lunak *web server* maupun *database server* pada komputer *server*. Perangkat lunak *database server* merupakan penyedia layanan-layanan pengelolaan basis data mysql yang digunakan pada *e-library* dan joomla. Perangkat lunak *web server* antara lain berperan untuk mengatur komunikasi antara komputer *client* dengan aplikasi-aplikasi di komputer *server*.

4.4.2 Perancangan Kelas

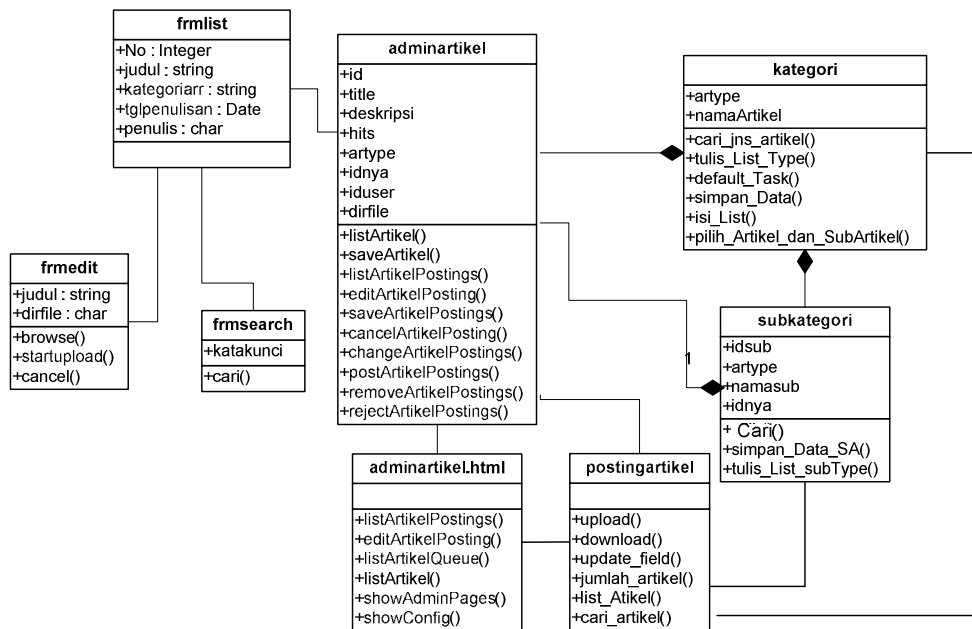
4.4.2.1 Diagram Kelas

Perancangan kelas-kelas yang terdapat pada aplikasi komponen *e-library* ini dilakukan dengan berdasar pada hasil analisis pada sub bab 4.3. Pada kelas-kelas yang bertipe *boundary*, pada tahap perancangan terkadang memerlukan kelas-kelas tambahan untuk mendukung realisasi antarmuka yang diinginkan. Dalam merancang kelas-kelas yang bertipe *control*, perlu mempertimbangkan kompleksitas dari fungsi yang dimiliki. Jika kompleksitas suatu fungsi sangat sederhana, maka kemungkinan dapat diwakili dengan sebuah operasi dari kelas lain yang berkaitan, jadi tidak harus membentuk satu kelas tersendiri.

Pada Gambar 15. terlihat kelas diagram dari perangkat lunak komponen *e-library*. Kelas-kelas yang digambarkan dalam kelas diagram pada gambar 15. ini hanya kelas-kelas baru yang dibuat sendiri, jadi bukan kelas-kelas yang sudah ada di CMS joomla. Kelas diagram yang menggambarkan keterkaitan antara kelas-kelas baru yang dibuat sendiri dengan kelas-kelas pada CMS joomla dapat dilihat pada lampiran D.

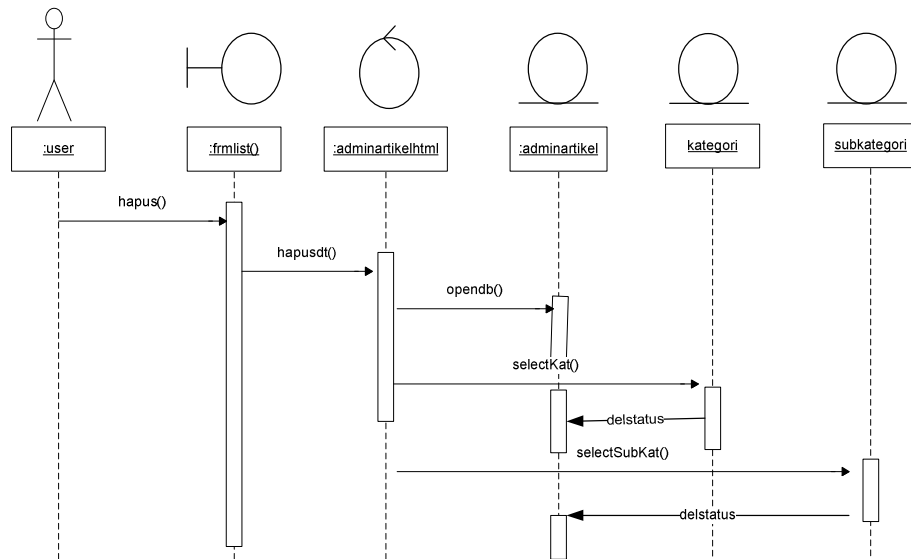
4.4.2.2 Interaksi Antar Kelas

Diagram kelas menggambarkan sistem secara statis, oleh karena itu dalam kelas diagram belum tergambar interaksi antar kelas dimana dapat menggambarkan keadaan sistem secara dinamis. Untuk menggambarkan interaksi antar kelas di dalam sistem dapat digunakan *sequence diagram*. Aspek penting dari *sequence diagram* adalah *sequence diagram* dapat menunjukkan suatu sekuen pesan-pesan yang dikirim diantara objek-objek dan dapat menunjukkan interaksi antar objek [ERI98].



Gambar 15 Perancangan Diagram Kelas

Gambar 16. menunjukkan *sequence diagram* untuk *usecase* hapus data. Pada gambar tersebut terlihat bahwa *sequence diagram* terdiri dari 2 sumbu yaitu sumbu vertikal dan sumbu horizontal. Sumbu vertikal menunjukkan waktu dan sumbu horizontal menunjukkan beberapa objek yang terkait. Beberapa *sequence diagram* yang lain dapat dilihat pada lampiran B.



Gambar 16 Sequence Diagram Use Case Hapus Data

4.4.3 Perancangan Data

Perancangan data merupakan aktivitas untuk menentukan jenis tabel beserta struktur tabel-tabel yang digunakan dalam perangkat lunak komponen *e-library*. Berdasarkan pada hasil analisis, data-data yang digunakan dalam perangkat lunak *e-library* meliputi :

1. Data artikel
2. Data kategori
3. Data subkategori
4. Data user
5. Data session
6. Data group user
7. Data content
8. Data menu
9. Data component

Data-data tersebut di atas harus diimplementasikan dalam bentuk tabel. Beberapa dari data di atas sudah diimplementasikan ke dalam beberapa tabel di dalam basis data Joomla. Tabel-tabel tersebut adalah:

1. Tabel jos_user
Berguna untuk menyimpan data-data pengguna *e-library*, baik *guest user*, *registered user* maupun admin.
2. Tabel jos_session
Digunakan untuk menyimpan data *user* yang sedang melakukan *login* di *e-library*.
3. Tabel jos_core_acl_group
Digunakan untuk menyimpan data group dari *user*.
4. Tabel jos_content
Digunakan untuk menyimpan data isi artikel *e-library*
5. Tabel jos_menu
Digunakan untuk menyimpan data *link* dari menu yang terdapat pada *front-end*
6. Tabel jos_component
Digunakan untuk menyimpan data *link* dari menu yang terdapat pada *back-end*

Enam tabel tersebut di atas sudah ada di dalam basis data joomla, sehingga tidak perlu dibuat lagi. Tabel yang belum ada pada basis data *e-library* yang diperlukan untuk perangkat lunak *e-library* adalah tabel-tabel yang digunakan untuk menyimpan data artikel, data kategori, data sub kategori. Ketiga data di atas diwujudkan ke dalam 3 jenis tabel yaitu sebagai berikut :

1. Tabel jos_jl_jobposting
Tabel ini berisi informasi mengenai artikel-artikel elektronik yang disimpan didalam basis data *e-library*. Penjelasan mengenai struktur data dari tabel ini dapat dilihat pada Tabel 6.

Tabel 6 Struktur Tabel jos_jl_jobposting

No	Nama <i>Field</i>	Tipe	Keterangan
1.	Id	int(4)	Primary key dari tabel jos_artikel
2.	Title	varchar(100)	Judul artikel

No	Nama <i>Field</i>	Tipe	Keterangan
3.	deskripsi	string	Penjelasan singkat mengenai isi artikel
4.	Hits	int(5)	Jumlah pengunjung situs sampai situs terakhir diakses
5.	Artype	varchar(25)0	Jenis kategori dari artikel
6.	idsubkategori	int(4)	Id dari sub kategori
7.	Dirfile	varchar(50)	Lokasi penyimpanan file artikel
8.	Ekstension	Varchar(5)	Ekstension dari file
7.	Iduser	Int(4)	Id dari pengguna
8.	pengarang	Varchar(30)	Nama pengarang artikel

2. Tabel jos_jl_jobtype

Tabel ini berisi informasi mengenai jenis kategori dari suatu artikel yang ada dalam *e-library*. Struktural tabel ini dapat dilihat pada Tabel 7.

Tabel 7 Struktur tabel jos_jl_jobtype

No	Nama <i>Field</i>	Tipe	Keterangan
1.	Artype	int(4)	Id dari kategori artikel
2.	namaartikel	varchar(100)	Judul artikel

3. Tabel jos_jl_subjobtype

Tabel ini berisi informasi mengenai pengklasifikasian kategori ke dalam beberapa sub kategori. Dalam tabel ini terdapat informasi mengenai jumlah sub kategori maupun nama sub kategori. Struktur tabel selengkapnya dapat dilihat di Tabel 8.

Tabel 8 Struktur Tabel jos_jl_subjobtype

No	Nama <i>Field</i>	Tipe	Keterangan
1.	Idsub	int(4)	Primary key dari tabel jos_subkategori
2.	Artype	int(4)	Id dari kategori artikel
3.	namasub	varchar(50)	Nama dari sub kategori
4.	Idnya	int(4)	Id dari sub kategori

4.4.4 Perancangan Antarmuka

Perancangan antarmuka dibuat dengan maksud untuk memberi kemudahan bagi pengguna aplikasi dalam menggunakan sistem atau program aplikasi serta menyediakan sarana untuk komunikasi yang efektif antara sistem dengan pengguna sistem. Beberapa hal yang perlu mendapat perhatian adalah bahwa antarmuka yang dibuat hendaknya memiliki konsistensi desain konten dan memenuhi aspek *user friendly* atau tidak menimbulkan kesulitan bagi pengguna.

Antarmuka pada perangkat lunak komponen *e-library* memiliki 2 modus, yaitu antarmuka yang ditujukan bagi administrator dan antarmuka yang ditujukan bagi *user* baik *registered user* maupun *guest user*. Beberapa desain antarmuka adalah seperti terlihat pada Gambar 17 dan Gambar 18. Gambar 17 menggambarkan desain antarmuka halaman utama (*homepage*) dari aplikasi komponen *e-library*. Pada rancangan antarmuka ini terlihat ada main menu yang terdiri dari menu *home* dan *browse artikel*. Menu *home* digunakan setiap kali ingin kembali ke tampilan halaman *homepage*. Menu *browse artikel* digunakan untuk menuju ke tampilan halaman *browse artikel*, dimana pada halaman ini pengunjung bisa menelusuri koleksi perpustakaan. Gambar 18 memperlihatkan desain antarmuka *Browse Articles*. Pada desain antarmuka *browse artikel* terdapat *pulldown* menu *display*, kriteria dan sub kriteria. Menu *display* digunakan untuk memilih jumlah baris yang akan menampilkan mater/artikel perpustakaan. Di samping menu *display* terdapat *pulldown menu* yang digunakan untuk menentukan kriteria yang diinginkan. Disamping menu tersebut terdapat menu untuk memilih sub kriteria yang sesuai dengan yang diinginkan. Beberapa desain antarmuka yang lain untuk perangkat lunak *e-library* dapat dilihat pada lampiran A.

Banner & Title	
About us	Favorite Contact Us Help
Main Menu <ul style="list-style-type: none"> • Home • Browse Artikel Login Form User name: <input type="text"/> Password : <input type="password"/>	Welcome.....
<input type="button" value="Login"/>	Powered by joomla Copyright @2008

Gambar 17 Desain Antarmuka Halaman Utama (Homepage)

Banner & Title					
About us		Favorite		Contact Us	Help
Main Menu <ul style="list-style-type: none">• Home• Browse Articles	Browse Articles				
	Display	20	Kesehatan	Mengenai Tulang	
	#				
	No. Judul	Kriteria Artikel	Tanggal Kirim	Penulis	
	1 Migrane	Kesehatan	06/04/2008	Mamat	
2 Flu Tulang	Kesehatan	06/04/2008	Mamat		
3 Encok	Kesehatan	06/01/2008	bila		
Powered by joomla					
Copyright @2008					

Gambar 18 Desain Antarmuka Halaman Browse Category