



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODE DAN PERANCANGAN APLIKASI

3.1 Metode Penelitian

Aplikasi hasil penelitian ini diimplementasikan pada *smartphone* yang menggunakan sistem operasi Android dan berguna untuk memudahkan pengguna dalam mencari jurnal Universitas Multimedia Nusantara sesuai dengan kebutuhan pengguna. Pada pengerjaannya, metode penelitian yang digunakan, yaitu dengan tahapan sebagai berikut.

1) Studi Pustaka

Studi pustaka merupakan tahap pembelajaran terhadap sumber-sumber yang berkaitan dengan penelitian, seperti jurnal, buku, artikel, dan referensi lainnya, baik berupa media cetak maupun media elektronik.

2) Perancangan dan Pengembangan Aplikasi

Pada tahapan ini, analisis terhadap prosedur yang sesuai untuk penelitian dilakukan agar aplikasi yang menjadi hasil dari penelitian dapat mencapai tujuan. Rancangan yang dibuat kemudian diimplementasikan pada bahasa pemrograman yang sesuai dengan sistem operasi Android, yaitu bahasa pemrograman Java.

3) Uji Coba Aplikasi

Tahapan uji coba dilakukan saat aplikasi dikembangkan. Hal ini bertujuan agar aplikasi dapat dipastikan berjalan dengan baik, dan apabila ada kesalahan, dapat dideteksi.

4) Pengumpulan Sampel dan Validasi Data

Dalam tahapan ini dibutuhkan sejumlah responden untuk mencoba aplikasi yang telah dibuat. Responden yang telah mencoba aplikasi kemudian diberikan sejumlah pertanyaan dalam bentuk kuesioner untuk meminta pendapat responden tentang aplikasi yang dibuat pada penelitian ini.

5) Analisis Sampel Data

Sampel data yang telah dikumpulkan dari para responden kemudian dianalisis untuk memastikan bahwa semua fitur dalam aplikasi dapat berjalan dengan baik dalam melakukan pencarian jurnal Universitas Multimedia Nusantara sesuai dengan *input* dari responden.

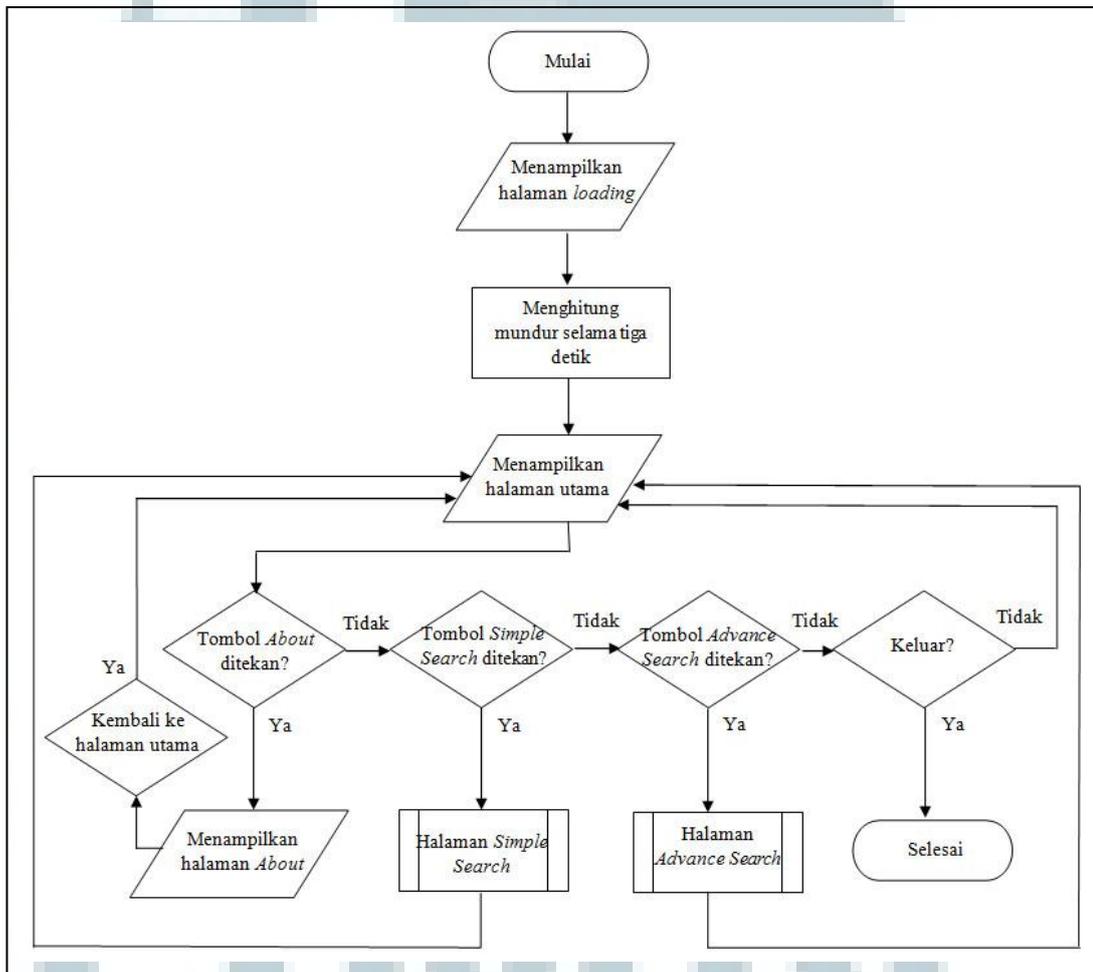
6) Penulisan Laporan

Dokumentasi yang ada pada setiap tahap, hingga penarikan kesimpulan dari hasil analisis data, kemudian disusun untuk dijadikan sebuah laporan penelitian.

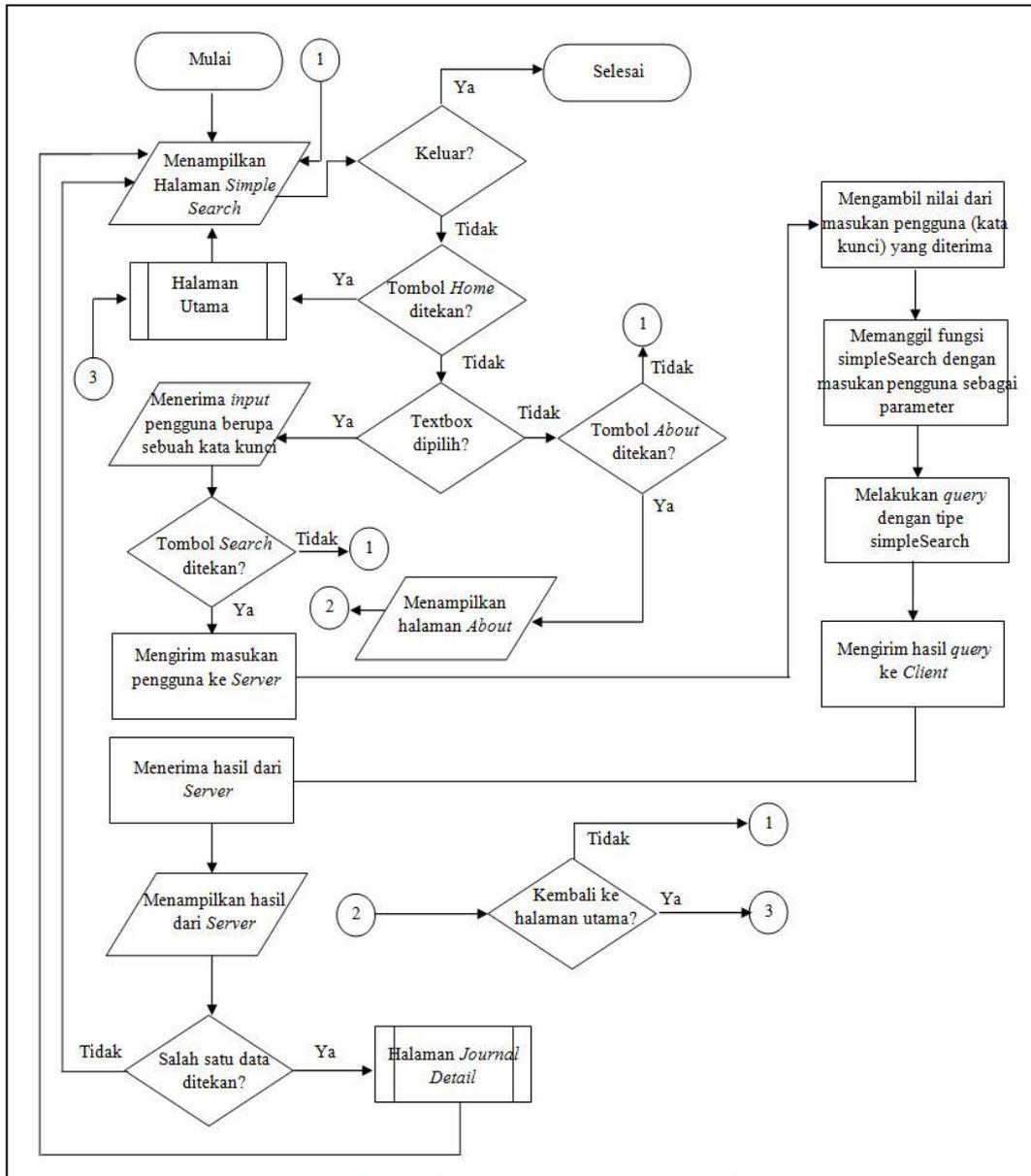
3.2 Perancangan Flowchart Aplikasi

Melalui Gambar 3.1, dapat dilihat jalannya aplikasi yang bermula dari halaman *Loading* hingga selesainya aplikasi. Saat aplikasi dijalankan pertama kali, maka ditampilkan halaman *Loading* berupa *splash screen*. Setelah halaman *Loading* selesai, aplikasi menampilkan halaman utama yang terdiri dari beberapa pilihan yaitu *About*, *Simple Search*, dan *Advance Search*. Jika *About* dipilih, maka ditampilkan halaman *About* yang berisi informasi tentang aplikasi.

Terdapat dua fitur pencarian dalam aplikasi, yaitu *simple search* dan *advance search*. Jika *Simple Search* dipilih, maka ditampilkan halaman *Simple Search* yang berguna untuk melakukan pencarian jurnal dengan memasukkan sebuah kata kunci. Jika *Advance Search* dipilih, maka ditampilkan halaman *Advance Search* yang berguna untuk melakukan pencarian jurnal dengan memasukkan data yang lebih spesifik terkait dengan jurnal yang dicari.



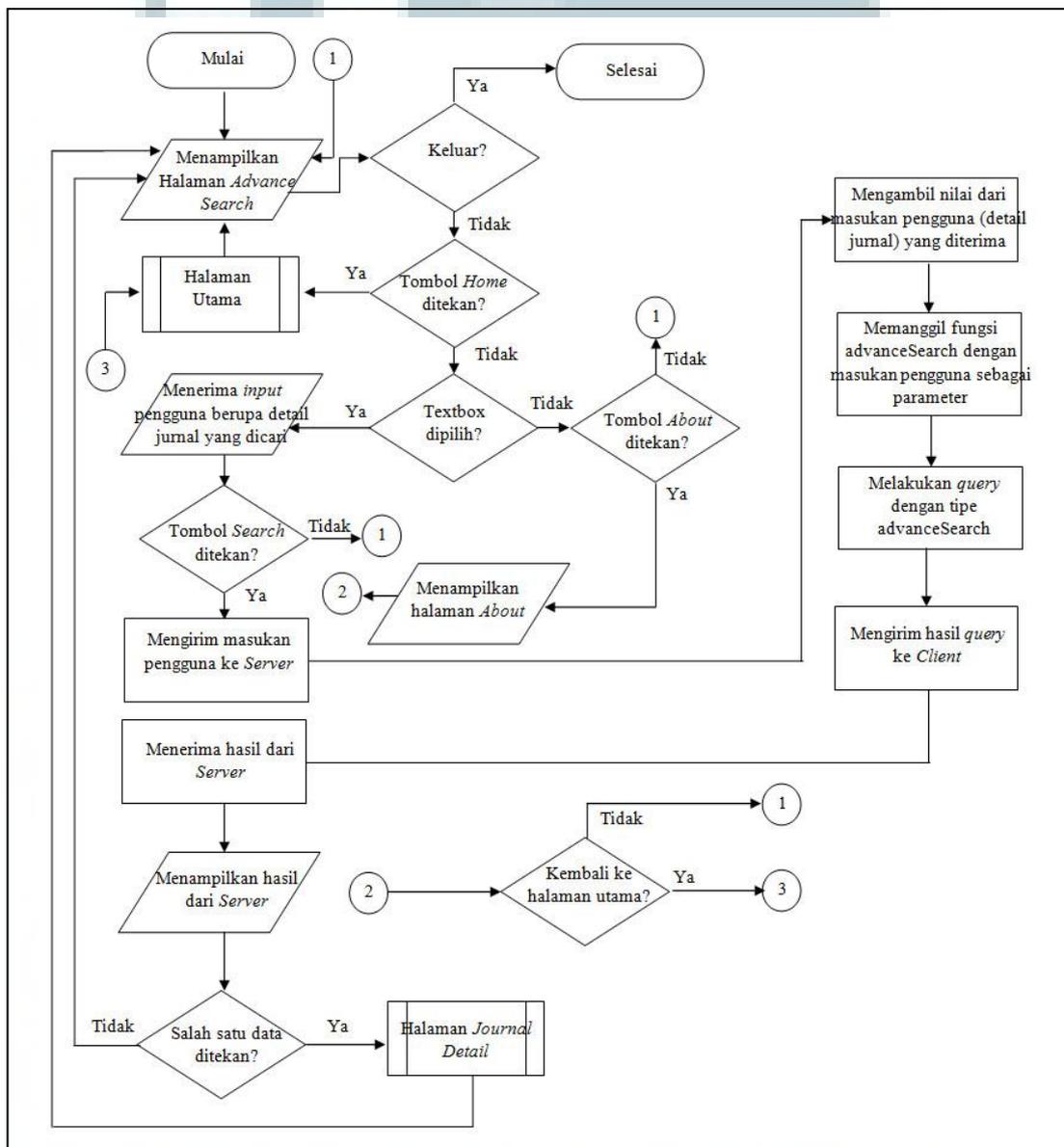
Gambar 3.1 Diagram Alir Halaman Utama



Gambar 3.2 Diagram Alir Halaman *Simple Search*

Gambar 3.2 merupakan diagram alir dari halaman *Simple Search*. Halaman ini merupakan halaman yang digunakan untuk melakukan pencarian *simple*. Pada halaman ini, pengguna memasukkan sebuah kata kunci terkait dengan jurnal yang dicari. Kata kunci tersebut kemudian dikirim ke *server* sebagai parameter. Di *server*, nilai dari kata kunci yang dikirim dari *client* diambil, lalu dijadikan sebagai parameter ketika memanggil fungsi *simpleSearch*. Dalam fungsi

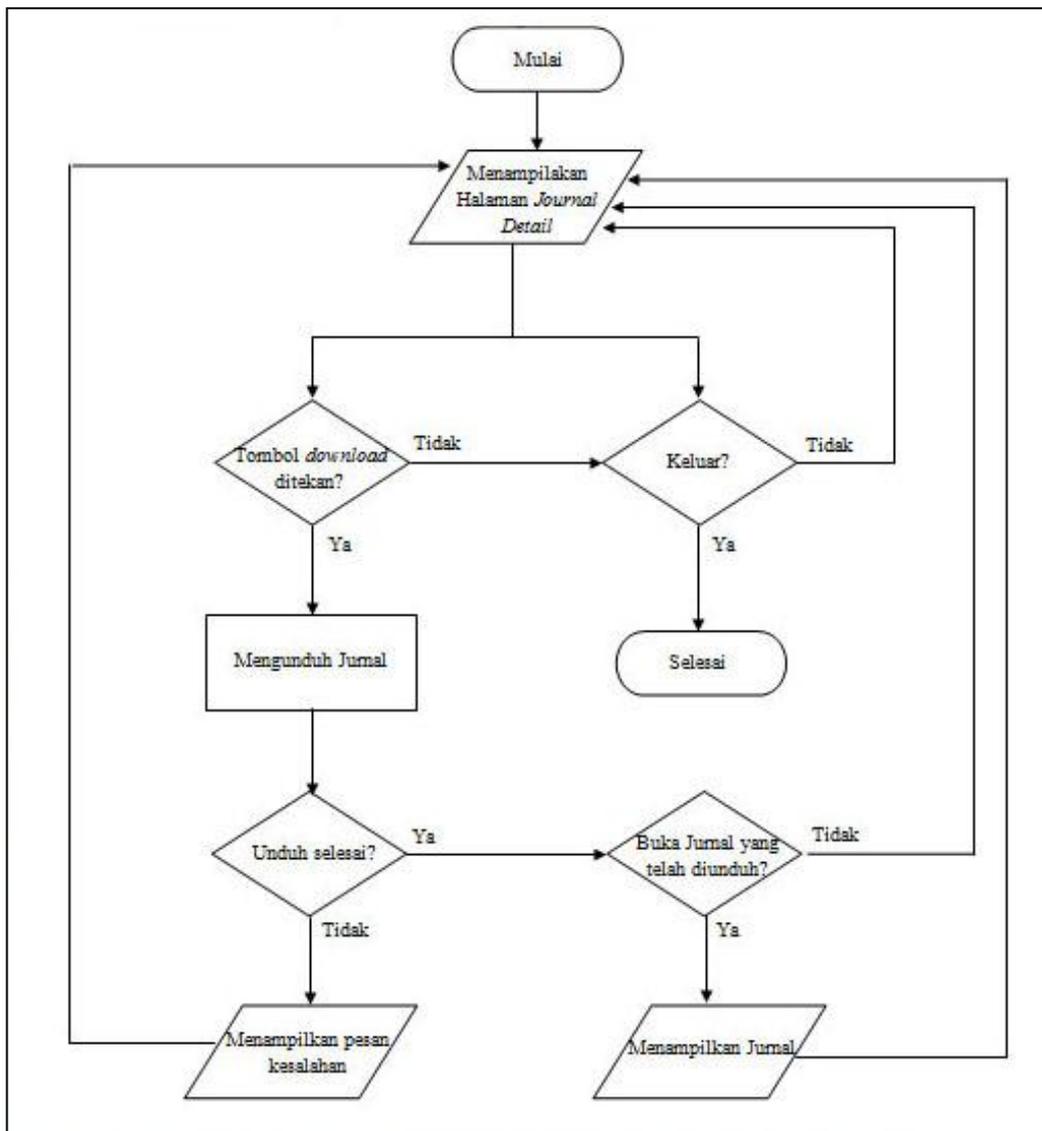
simpleSearch dilakukan *query* dengan SPARQL sesuai dengan kata kunci yang diterima. Hasil *query* tersebut dikembalikan ke aplikasi. Jika jurnal ditemukan, aplikasi menampilkan data jurnal berupa *list view*, atau menampilkan pesan kesalahan apabila jurnal tidak ditemukan. Pada halaman ini, terdapat tombol *home* untuk kembali ke halaman utama dan tombol *about* untuk menampilkan informasi tentang aplikasi.



Gambar 3.3 Diagram Alir Halaman Advance Search

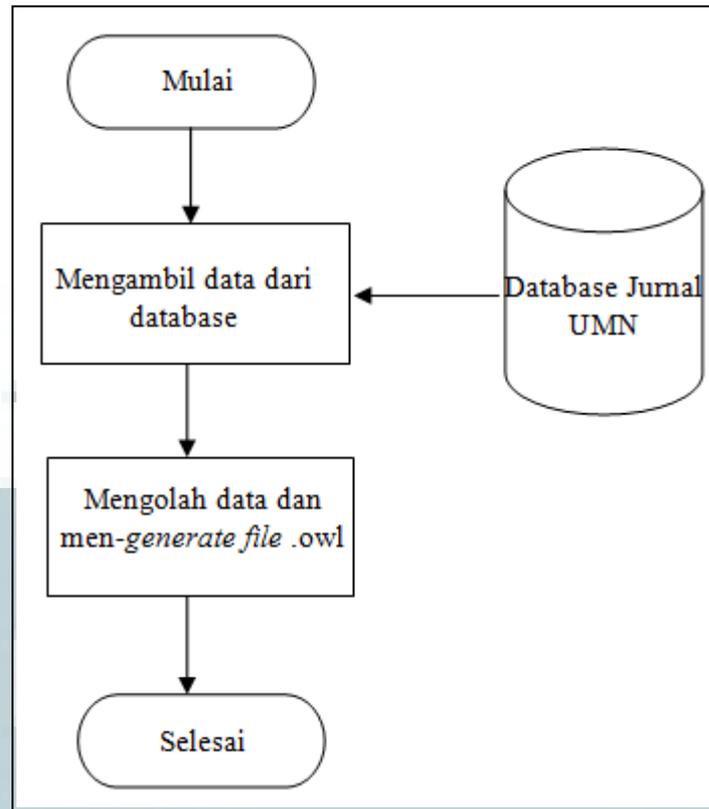
Gambar 3.3 merupakan diagram alir dari halaman *Advance Search*. Halaman ini merupakan halaman yang digunakan untuk melakukan pencarian *advance*. Pada halaman ini, pengguna memasukkan data yang lebih spesifik terkait dengan jurnal yang dicari, hal ini berbeda dengan *simple search* yang hanya memasukkan sebuah kata kunci. Data tersebut kemudian dikirim ke *server* sebagai parameter. Di *server*, nilai dari data yang dikirim dari *client* diambil, lalu dijadikan sebagai parameter ketika memanggil fungsi *advanceSearch*. Hasil *query* tersebut dikembalikan ke aplikasi. Aplikasi menampilkan data jurnal berupa *list view* jika jurnal ditemukan, atau menampilkan pesan kesalahan apabila jurnal tidak ditemukan. Pada halaman ini, terdapat tombol *home* untuk kembali ke halaman utama dan tombol *about* untuk menampilkan informasi tentang aplikasi.

Gambar 3.4 merupakan diagram alir dari halaman *Journal Detail*. Pada halaman ini, ditampilkan informasi jurnal yang dipilih dari halaman *simple search* dan *advance search*, seperti judul, penulis, penerbit, *keyword*, *abstract*, dan lain-lain. Pengguna juga dapat mengunduh jurnal dengan menekan tombol *download*. *File* yang diunduh disimpan dalam media penyimpanan yang ada pada *smartphone*. Setelah proses unduh selesai, pengguna dapat membuka jurnal langsung dari *device* tersebut sesuai dengan jenis *file* yang diunduh.



Gambar 3.4 Diagram Alir Halaman *Journal Detail*

Data yang ditampilkan pada halaman *Journal Detail* merupakan data yang diambil dari sebuah *file resource* dengan format *.owl*. *File* tersebut dibuat secara otomatis dengan mengambil data dari *database* jurnal Universitas Multimedia Nusantara, lalu memasukkannya ke dalam sebuah *file* dengan struktur ontologi. Diagram alir dari proses tersebut dapat dilihat pada Gambar 3.5.

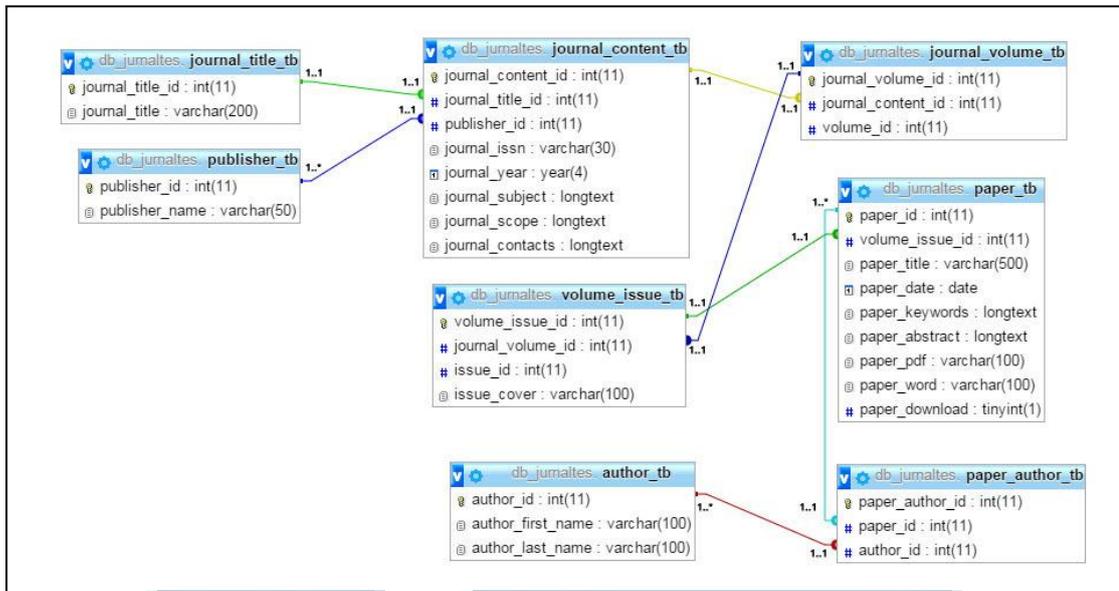


Gambar 3.5 Diagram Alir Proses Pembuatan *File Resource*

Proses pembuatan *file resource* disimpan pada sebuah *file* dengan format .php dan dieksekusi di *server*. Ketika *admin* jurnal Universitas Multimedia Nusantara memasukkan data baru atau mengubah atau menghapus data di *database* jurnal UMN, maka *file* tersebut dipanggil sehingga *file resource* secara otomatis dibuat kembali.

3.3 Entity Relationship Diagram

Dalam penelitian ini digunakan *database* jurnal Universitas Multimedia Nusantara yang sudah ada sebelum penelitian ini dilakukan. Pada Gambar 3.6, dapat dilihat *entity relationship diagram* dari *database* jurnal Universitas Multimedia Nusantara.



Gambar 3.6 ERD Database Jurnal UMN

Seperti yang terlihat pada Gambar 3.5, *file resource* yang digunakan dalam penelitian ini dibuat secara otomatis dengan mengambil data dari *database* tersebut.

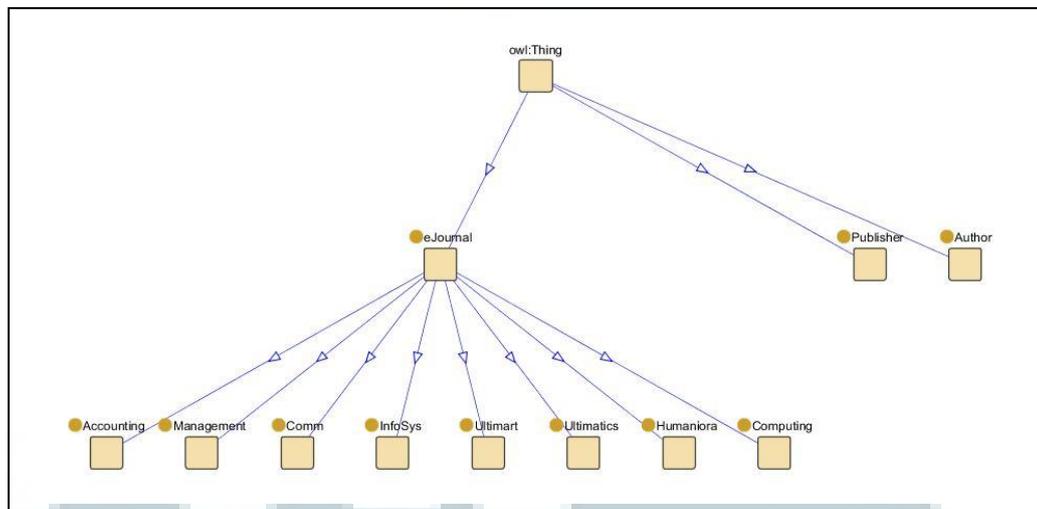
3.4 Model Ontologi yang Digunakan

Rancangan ontologi yang diterapkan untuk *file resource* memiliki definisi-definisi *class*, *subclass*, *object property*, dan *datatype property*. Definisi tersebut diuraikan sebagai berikut.

3.4.1 Definisi Class dan Subclass

Terdapat tiga *class* utama yang digunakan dalam ontologi, yaitu *class* Author, *class* eJournal, dan *class* Publisher. *Class* eJournal mempunyai *subclass* yang berisi Accounting, Comm, Computing, Humaniora, InfoSys, Management,

Ultimart, dan Ultimatics. Pendefinisian *class* dan *subclass* dapat dilihat pada Gambar 3.7.



Gambar 3.7 Class dan Subclass

3.4.2 Definisi Object Property

Terdapat empat *object property* yang digunakan dalam ontologi, yaitu *hasAuthor*, *hasPublisher*, *isAuthorOf*, dan *isPublisherOf*. Pendefinisian *object property* dapat dilihat pada Gambar 3.8.

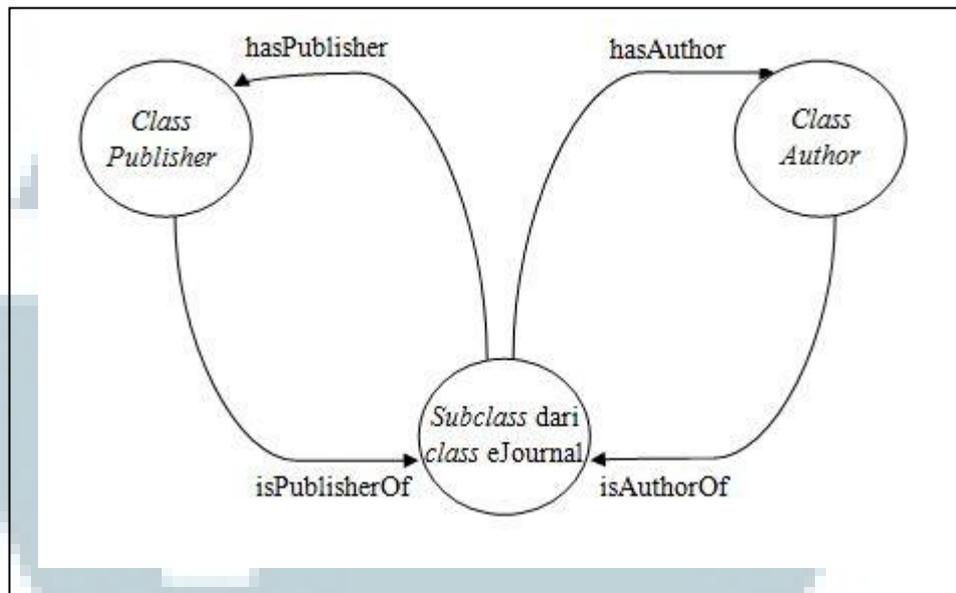
hasAuthor \equiv isAuthorOf

hasPublisher \equiv isPublisherOf

Gambar 3.8 Object Property

Property *hasAuthor* merupakan *inverse property* dari *isAuthorOf* dan *property* *hasPublisher* merupakan *inverse property* dari *isPublisherOf*. Dengan adanya fungsi *inverse property*, maka nilai-nilai dari suatu properti secara otomatis terisi ketika *inverse* dari properti tersebut diberi nilai. *Object property* tersebut menghubungkan *subclass* dari *class* *eJournal* dengan *class* *Publisher* dan

class Author. *Object property* *hasAuthor* dan *isAuthorOf* menghubungkan *subclass* dari *class eJournal* dengan *class Author*. *Object property* *hasPublisher* dan *isPublisherOf* menghubungkan *subclass* dari *class eJournal* dengan *class Publisher*. Hubungan antar *class* tersebut digambarkan pada Gambar 3.9.



Gambar 3.9 Relasi Antar *Class*

3.4.3 Definisi Datatype Property

Terdapat sepuluh *datatype property* yang digunakan dalam ontologi, yaitu *hasAbstract*, *hasAuthor*, *hasISSN*, *hasKeyword*, *hasLink*, *hasNomor*, *hasPublisherName*, *hasTitle*, *hasVolume*, dan *hasYear*. Pendefinisian *datatype property* dapat dilihat pada Gambar 3.10.



Gambar 3.10 *Datatype Property*

Masing-masing *property* memiliki *range* dan *type* yang berbeda, seperti yang terlihat pada Tabel 3.1.

Tabel 3.1 *Range dan Type dari Property*

Property	Range	Type
hasAbstract	<i>Single String</i>	<i>Datatype Property</i>
hasAuthorName	<i>Multiple Author</i>	<i>Object Property</i>
hasISSN	<i>Single String</i>	<i>Datatype Property</i>
hasKeyword	<i>Single String</i>	<i>Datatype Property</i>
hasLink	<i>Single String</i>	<i>Datatype Property</i>
hasNomor	<i>Single Integer</i>	<i>Datatype Property</i>
hasPublisherName	<i>Single Publisher</i>	<i>Object Property</i>
hasTitle	<i>Single String</i>	<i>Datatype Property</i>
hasVolume	<i>Single Integer</i>	<i>Datatype Property</i>
hasYear	<i>Single Integer</i>	<i>Datatype Property</i>

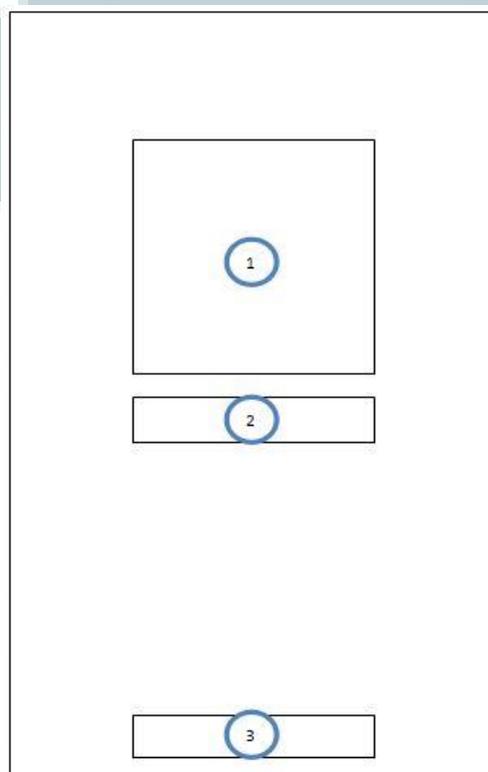
Properti hasAbstract, hasISSN, hasKeyword, hasLink, dan hasTitle memiliki *range single string*, artinya properti-properti ini memiliki *value* sebuah *string*. Properti hasNomor, hasVolume, dan has Year memiliki *range single*

integer, artinya properti-properti ini memiliki *value* sebuah *integer*. Properti *hasPublisherName* memiliki *range single publisher*, artinya properti ini memiliki nilai sebuah *object* dari *class publisher*. Properti *hasAuthorName* memiliki *range multiple author*, artinya properti ini memiliki nilai satu atau lebih *object* dari *class author*.

3.5 Rancangan User Interface Aplikasi

Aplikasi yang dikembangkan memiliki enam halaman, yaitu halaman *loading*, halaman utama, halaman *simple search*, halaman *advance search*, halaman *about*, dan halaman *journal detail*. Rancangan tampilan dan penjelasan dari halaman-halaman tersebut, yaitu sebagai berikut.

1) Halaman *Loading*



Gambar 3.11 Rancangan Tampilan Halaman *Loading*

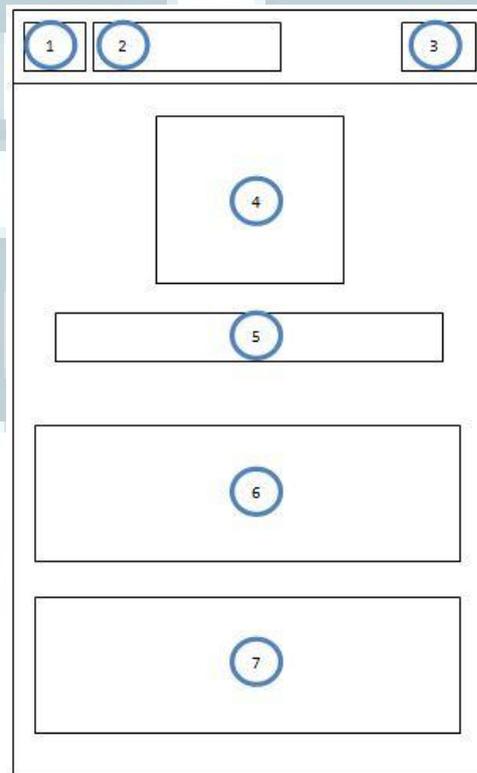
Gambar 3.11 merupakan rancangan tampilan dari halaman *loading* yang ditampilkan selama tiga detik saat pertama kali aplikasi dijalankan, kemudian secara otomatis diarahkan ke halaman utama setelah tiga detik. Berikut keterangan untuk rancangan tampilan halaman *loading*.

Nomor 1 : *Image view* yang memuat logo aplikasi.

Nomor 2 : *Text view* yang berisi nama aplikasi.

Nomor 3 : *Progress bar* dari halaman *loading*.

2) Halaman Utama



Gambar 3.12 Rancangan Tampilan Halaman Utama

Pada Gambar 3.12, dapat dilihat rancangan tampilan halaman utama. Pada halaman ini, terdapat dua tombol yang dapat digunakan pengguna untuk melakukan pencarian dengan tipe pencarian yang berbeda, yaitu *simple search*

dan *advance search*. Berikut keterangan untuk rancangan tampilan halaman utama.

Nomor 1 : Bagian dari *action bar* yang berupa logo aplikasi.

Nomor 2 : Bagian dari *action bar* yang berupa nama halaman.

Nomor 3 : Bagian dari *action bar* yang berupa *icon* “i” yang mengarah ke halaman *about*.

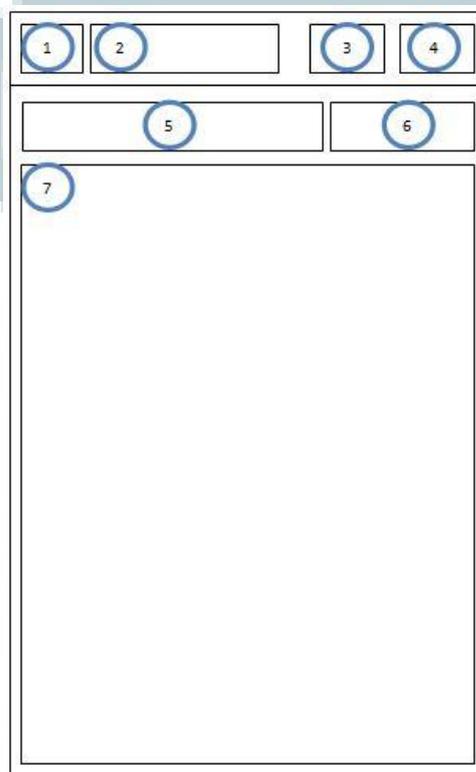
Nomor 4 : *Image view* yang memuat logo aplikasi.

Nomor 5 : *Text view* yang berisi nama aplikasi.

Nomor 6 : *Image button* yang mengarah ke halaman *simple search*.

Nomor 7 : *Image button* yang mengarah ke halaman *advance search*.

3) Halaman *Simple Search*



Gambar 3.13 Rancangan Tampilan Halaman *Simple Search*

Gambar 3.13 merupakan rancangan tampilan halaman *simple search*. Pada halaman ini, pengguna dapat melakukan pencarian dengan memasukkan sebuah kata kunci yang terkait dengan jurnal yang dicari. Berikut keterangan untuk rancangan tampilan halaman *simple search*.

Nomor 1 : Bagian dari *action bar* yang berupa logo aplikasi.

Nomor 2 : Bagian dari *action bar* yang berupa nama halaman.

Nomor 3 : Bagian dari *action bar* yang berupa *icon home* yang mengarah ke halaman utama.

Nomor 4 : Bagian dari *action bar* yang berupa *icon "i"* yang mengarah ke halaman *about*.

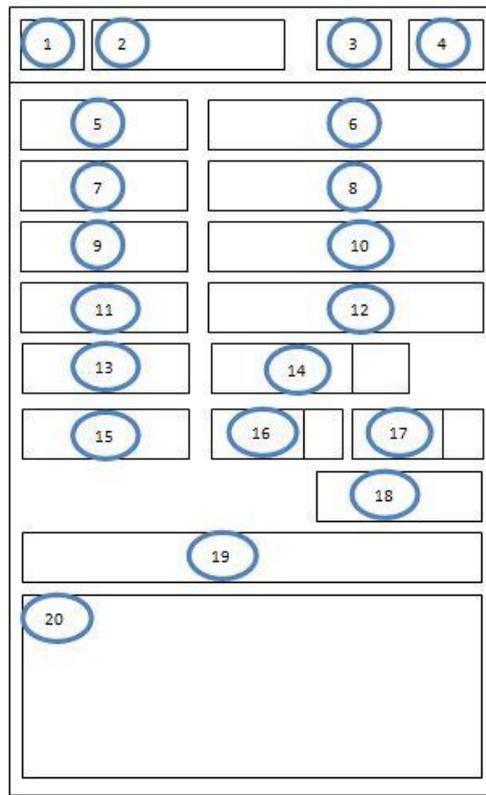
Nomor 5 : *Edit text* untuk memasukkan *keyword*.

Nomor 6 : *Image button* untuk memulai pencarian.

Nomor 7 : *List view* yang menampilkan hasil dari *keyword* yang dimasukkan sebelumnya.

4) Halaman *Advance Search*

Pada Gambar 3.14, dapat dilihat rancangan tampilan halaman *advance search*. Berbeda dengan *simple search*, dengan memilih pencarian tipe *advance search*, pengguna dapat melakukan pencarian dengan memasukkan data yang lebih spesifik terkait dengan jurnal yang dicari, seperti judul, abstrak, kata kunci, penulis, kategori jurnal, dan tahun terbit.



Gambar 3.14 Rancangan Tampilan Halaman *Advance Search*

Berikut keterangan untuk rancangan tampilan halaman *advance search*.

Nomor 1 : Bagian dari *action bar* yang berupa logo aplikasi.

Nomor 2 : Bagian dari *action bar* yang berupa nama halaman.

Nomor 3 : Bagian dari *action bar* yang berupa *icon home* yang mengarah ke halaman utama.

Nomor 4 : Bagian dari *action bar* yang berupa *icon "i"* yang mengarah ke halaman *about*.

Nomor 5 : *Text view* yang berisi kata "*Title*".

Nomor 6 : *Edit text* untuk memasukkan judul.

Nomor 7 : *Text view* yang berisi kata "*Abstract*".

Nomor 8 : *Edit text* untuk memasukkan abstrak.

Nomor 9 : *Text view* yang berisi kata “*Keyword*”.

Nomor 10 : *Edit text* untuk memasukkan *keyword*.

Nomor 11 : *Text view* yang berisi kata “*Author*”.

Nomor 12 : *Edit text* untuk memasukkan penulis.

Nomor 13 : *Text view* yang berisi kata “*Journal*”.

Nomor 14 : *Spinner* yang berisi pilihan kategori jurnal.

Nomor 15 : *Text view* yang berisi kata “*Year*”.

Nomor 16 : *Spinner* yang berisi pilihan tahun awal.

Nomor 17 : *Spinner* yang berisi pilihan tahun akhir.

Nomor 18 : *Image button* yang digunakan untuk memulai pencarian.

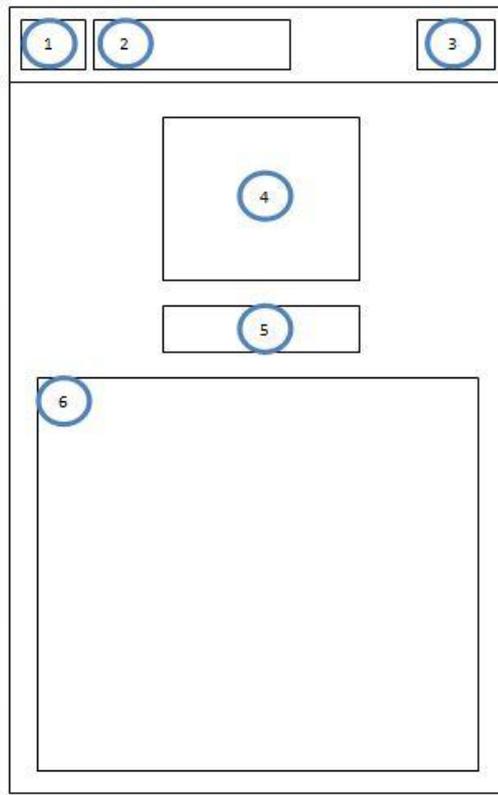
Nomor 19 : *Image button* yang digunakan untuk menyembunyikan pilihan *advance search*.

Nomor 20 : *List view* yang berisi hasil pencarian dari *advance search*.

5) Halaman *About*

Halaman *about* merupakan halaman yang menampilkan informasi tentang aplikasi dan penjelasan singkat tentang fitur pencarian yang ada dalam aplikasi.

Rancangan tampilan halaman *about* dapat dilihat pada Gambar 3.15.



Gambar 3.15 Rancangan Tampilan Halaman *About*

Berikut keterangan untuk rancangan tampilan halaman *about*.

Nomor 1 : Bagian dari *action bar* yang berupa logo aplikasi.

Nomor 2 : Bagian dari *action bar* yang berupa nama halaman.

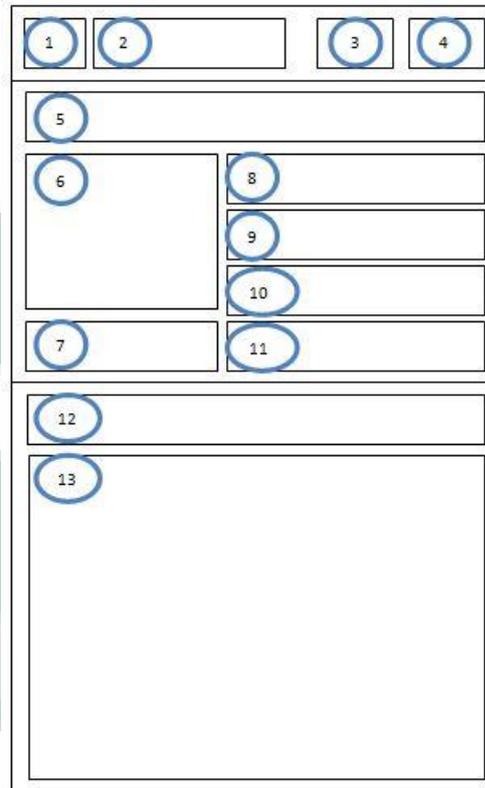
Nomor 3 : Bagian dari *action bar* yang berupa *icon home* yang mengarah ke halaman utama.

Nomor 4 : *Image view* yang memuat logo aplikasi.

Nomor 5 : *Text view* yang berisi nama aplikasi.

Nomor 6 : *Text view* yang berisi informasi tentang aplikasi.

6) Halaman *Journal Detail*



Gambar 3.16 Rancangan Tampilan Halaman *Journal Detail*

Pada Gambar 3.16, dapat dilihat rancangan tampilan halaman *journal detail*. Pada halaman ini, ditampilkan informasi lengkap terkait jurnal yang dipilih dari hasil pencarian *simple search* dan *advance search*. Pengguna juga dapat mengunduh jurnal dengan memilih tombol *download* pada halaman ini. Berikut keterangan untuk rancangan tampilan halaman *journal detail*.

Nomor 1 : Bagian dari *action bar* yang berupa logo aplikasi.

Nomor 2 : Bagian dari *action bar* yang berupa nama halaman.

Nomor 3 : Bagian dari *action bar* yang berupa *icon home* yang mengarah ke halaman utama.

Nomor 4 : Bagian dari *action bar* yang berupa *icon* “i” yang mengarah ke halaman *about*.

Nomor 5 : *Text view* yang berisi judul jurnal.

Nomor 6 : *Image view* yang memuat logo aplikasi.

Nomor 7 : *Image button* yang digunakan untuk mengunduh jurnal.

Nomor 8 : *Text view* yang berisi kategori, tahun, *volume*, dan nomor jurnal.

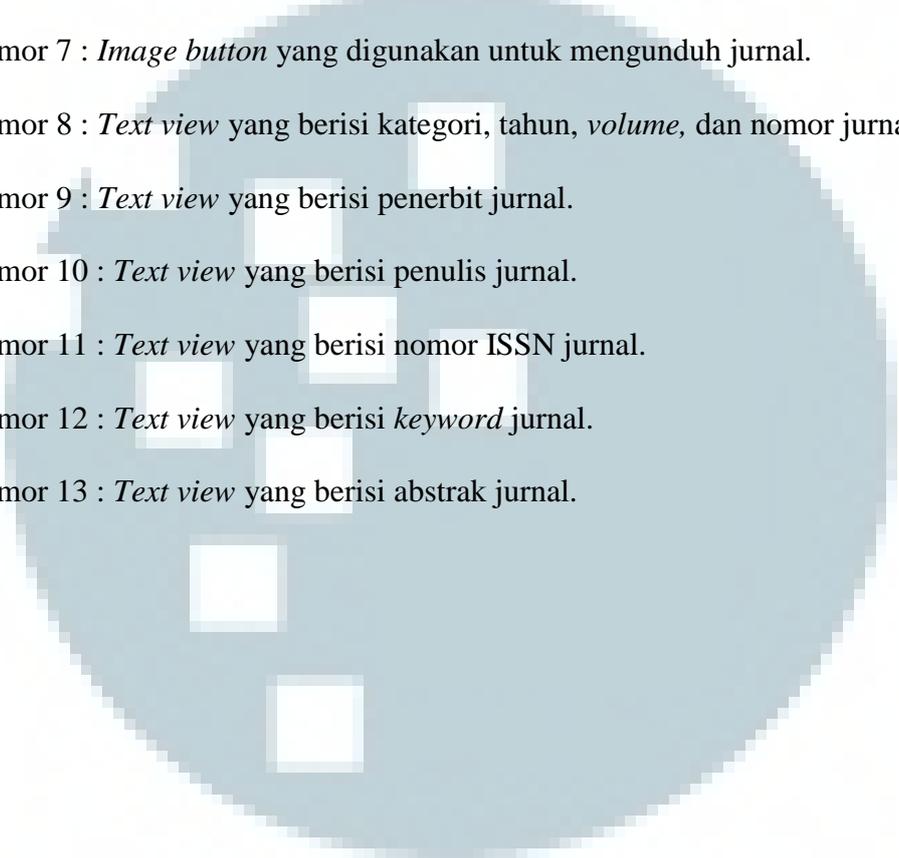
Nomor 9 : *Text view* yang berisi penerbit jurnal.

Nomor 10 : *Text view* yang berisi penulis jurnal.

Nomor 11 : *Text view* yang berisi nomor ISSN jurnal.

Nomor 12 : *Text view* yang berisi *keyword* jurnal.

Nomor 13 : *Text view* yang berisi abstrak jurnal.



UMN