

## BAB VI

### KESIMPULAN DAN SARAN

#### 6.1 Kesimpulan

Berdasarkan dari pembahasan pada bab-bab sebelumnya, maka dapat ditarik kesimpulan dari Tugas Akhir ini yaitu :

1. Telah berhasil dibangun aplikasi *web scraper* situs berita dan peringkas berita berbasis *web* dan *mobile*.
2. Aplikasi KUBERIN dapat mengambil data berita dari 6 situs berita dengan teknik *web scraping* secara otomatis.
3. Aplikasi KUBERIN dapat memberikan notifikasi atau pengesetan status dari aktif menjadi tidak aktif, jika tags dan rules pengambilan berita tidak dapat mengambil berita.
4. Aplikasi KUBERIN berhasil dibangun untuk dapat melakukan peringkasan berita berdasarkan topik yang diinginkan pengguna.
5. Aplikasi KUBERIN berhasil dibangun untuk menampilkan data berita berdasarkan kategori, sehingga memudahkan pengguna dalam mencari berita.
6. Aplikasi KUBERIN berhasil dibangun untuk memberikan kemudahan dalam membaca berita tanpa membuka berbagai macam situs berita.

## 6.2 Saran

Dari proses analisis, perancangan, implementasi hingga pengujian sistem pada pembuatan Tugas Akhir, didapatkan beberapa saran untuk pengembangan lebih lanjut dari perangkat lunak KUBERIN, yaitu :

1. Aplikasi KUBERIN dapat dikembangkan lebih lanjut untuk bagian *summarize* berita agar dapat menghasilkan hasil ringkasan yang baik dan dimengerti oleh pengguna, misalnya dengan mengidentifikasi kemiripan dokumen terlebih dahulu sebelum diringkas.

## DAFTAR PUSTAKA

- Astari, V. P. W., 2014. *Analisis Framing Pemberitaan Penetapan Gubernur Banten Ratu Atut Chosiyah, Sebagai Tersangka Korupsi pada VIVAnews dan Tempo.co Periode 17 - 20 Desember 2013*, Yogyakarta: Universitas Atma Jaya Yogyakarta.
- Aswari, A. F. (2013). *PENGARUH MEDIA ONLINE DAN KETERBUKAAN INFORMASI PUBLIK MELALUI WEBSITE DALAM MENINGKATKAN CITRA PEMERINTAH PROVINSI DKI JAKARTA*. Jakarta: BINUS University.
- digitalocean, 2013. *How To Use Cron To Automate Tasks On a VPS*. [Online]  
Available at:  
<https://www.digitalocean.com/community/tutorials/how-to-use-cron-to-automate-tasks-on-a-vps>  
[Accessed 22 Juli 2015].
- Evan, F. H., Purnomo, Y. S. & Pranowo, 2014. *Pembangunan Perangkat Lunak Peringkat Dokumen dari Banyak Sumber Menggunakan Sentence Scoring dengan Metod TF-IDF*, Yogyakarta: Magister Teknik Informatika Universitas Atma Jaya Yogyakarta.
- Hardiono, K., 2010. *Aplikasi Web Scrapping Untuk Koleksi Konten Utama Halaman Situs*, s.l.: Unisbank.
- Hovy, E. & Lin, C.-Y., 2001. *Automated Text Summarization and the Summarist System*, California: Information Sciences Institute of the University of Southern California.
- IDWebhost, 2007. *SETTING CRONJOB*. [Online]  
Available at:  
<https://blog.idwebhost.com/cpanel/setting-cronjob/>  
[Accessed 21 Juni 2015].
- Indra Warman, M. & Zahni, A., 2013. *REKAYASA WEB UNTUK PEMESANAN HANDPHONE*. *Jurnal Momentum*, Volume 15, pp. 30-38.

- Intan, R. & Defeng, A., 2006. *HARD:Subject-based Search Engine menggunakan TF-IDF dan Jaccard's Coefficient*, Surabaya: Jurusan Teknik Informatika Kristen Petra.
- Josi, A., Abdillah, L. A. & Suryayusra, 2014. Penerapan Teknik *Web scraping* pada Mesin Pencari Artikel Ilmiah. *Jurnal Sistem Informasi*, pp. 159-164.
- Kusumawaty, A., 2012. *APLIKASI PEMESANAN MAKANAN PADA RESTORAN*, Jakarta: Universitas Gunadarma.
- Kuswandi, R. et al., 2007. KONSOLIDASI DATABASE KEPENDUDUKAN NASIONAL. *Jurnal Ssitem Informasi MTI UI*, Volume 3, pp. 28-34.
- Lee, S. & Scheel, 2004. *Mobile Applications : Architecture, Design, and Development*. New Jersey: Pearson Education.
- Pattera, W. A., Noertjahyana, A. & Gunawan, I., 2014. *Pembuatan Aplikasi Driving Alert System Berbasis*, Surabaya: Universitas Kristen Petra.
- Putra, A. D., 2014. *Analisa Hasil Perbandingan Peringkasan Teks Otomatis Untuk Berita Online Menggunakan Metode Term frequency-Inverse Document Frequency Dengan Corpus dan Non Corpus (Studi Kasus BBC Magazine Online)*, Yogyakarta: UIN Sunan Kalijaga.
- Rifangi, M. A., 2012. *Rancang Bangun Aplikasi Dekstop Client Transkrip Nilai pada Domain "krs.amikom.ac.id" dengan Teknik Web scraping*, Yogyakarta: STMIK AMIKOM Yogyakarta.
- Riyadi, D., 2013. *Rancang Bangun Rest Web Service untuk Perbandingan Harga Pengiriman dengan Metode Web Scrapping dan Pemanfaatan API*, Yogyakarta: STMIK AMIKOM Yogyakarta.

Sutanta, E. & Mustofa, K., 2012. *Kebutuhan Web Service untuk Sinkronisasi Data antar Sistem Informasi dalam E-Gov di Pemkab Bantul Yogyakarta. JURTIK STMIK Bandung.*

Utomo, M. S., 2012. *Implementasi PHP sebagai Penghasil Konten Otomatis pada Halaman Situs. Jurnal Teknologi Informasi DINAMIK, pp. 147-153.*

Vargiu, E. & Urru, M., 2012. *Exploiting Web scraping in a collaborative filtering-based approach to web advertising. Artificial Intelligence Research, Volume 2, pp. 44-54.*

Zaira, Z., 2010. *Implementasi Ekstraksi Web untuk Hadits yang diterjemahkan dalam Bahasa Indonesia, Depok: Universitas Indonesia.*



# DPPL

## DESKRIPSI PERANCANGAN PERANGKAT LUNAK

KUBERIN

(Kumpulan Berita Indonesia)

Untuk :


Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

Dipersiapkan Oleh :

Yustinus Andry Setiawan /110706659

Program Studi Teknik Informatika  
Fakultas Teknologi Industri  
Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		<b>DPPL - KUBERIN</b>		1 / 62
		Revisi	A	

## DAFTAR PERUBAHAN

REVISI	DESKRIPSI
A	
B	
C	
D	
E	
F	

INDEKS TGL	-	A	B	C	D	E	F
DITULIS OLEH							
DIPERIKSA OLEH							
DISETUJUI OLEH							



## DAFTAR HALAMAN PERUBAHAN

Halaman	Revisi	Halaman	Revisi



## DAFTAR ISI

1. Pendahuluan.....	7
1.1 Tujuan.....	7
1.2 Lingkup Masalah.....	7
1.3 Definisi, Akronim dan Singkatan.....	8
1.4 Referensi.....	9
2 Perancangan Sistem.....	10
2.1 Perancangan Asitektur.....	10
2.1.1 Web Application.....	10
2.1.2 Mobile Application.....	11
2.2 Perancangan Rinci.....	12
2.2.1 Sequence Diagram KUBERIN Website.....	12
2.2.1.1 Fungsi Login.....	12
2.2.1.2 Fungsi Logout.....	12
2.2.1.3 Fungsi Pengelolaan Data Scraping Berita.....	13
2.2.1.3.1 Fungsi Test Data Scraping.....	13
2.2.1.3.2 Fungsi Simpan Data Scraping.....	14
2.2.1.3.3 Fungsi Tampil Data Scraping.....	14
2.2.1.3.4 Fungsi Update Data Scraping.....	15
2.2.1.4 Fungsi Lihat Data Berita.....	16
2.2.1.5 Fungsi Pengelolaan Kategori.....	16
2.2.1.5.1 Fungsi Tambah Kategori Utama.....	16
2.2.1.5.2 Fungsi Pindah Kategori Berita.....	17
2.2.1.5.3 Fungsi Hubung Kategori Utama dan Kategori Berita.....	17
2.2.1.6 Fungsi Lihat Summary Berita Web.....	18
2.2.2 Sequence Diagram KUBERIN Mobile.....	18
2.2.2.1 Fungsi Tampil Trending Topik.....	18
2.2.2.2 Fungsi Lihat Summary Berita Mobile.....	19
2.2.2.3 Fungsi Tampil Data Berita.....	19
2.2.3 Sequence Diagram KUBERIN Web Scraper.....	20
2.2.3.1 Fungsi Get URL Berita.....	20
2.2.3.2 Fungsi Get Data Berita.....	20
2.2.3.3 Fungsi Cek Tags Tidak Aktif.....	21
2.2.4 Class Diagram.....	22
2.2.4.1 Class Diagram Aplikasi Website.....	22
2.2.4.2 Class Diagram Aplikasi Mobile.....	23
2.2.5 Class Diagram Specific Descriptions KUBERIN Website.....	24
2.2.5.1 Specific Design Class login_view.....	24
2.2.5.2 Specific Design Class scraping_view.....	24
2.2.5.3 Specific Design Class databerita_view.....	24
2.2.5.4 Specific Design Class databeritasearch_view.....	24
2.2.5.5 Specific Design Class datatags_view.....	24
2.2.5.6 Specific Design Class kategori_view.....	24
2.2.5.7 Specific Design Class summarize_view.....	25
2.2.5.8 Specific Design Class summarizebaru_view.....	25
2.2.5.9 Specific Design Class home.....	25
2.2.5.10 Specific Design Class scraping.....	26
2.2.5.11 Specific Design Class croncekaktif.....	27

2.2.5.12	Specific Design Class cronurlberita.....	28
2.2.5.13	Specific Design Class cronberita.....	28
2.2.5.14	Specific Design Class cronberitamenunggu.....	29
2.2.5.15	Specific Design Class databerita.....	30
2.2.5.16	Specific Design Class datatags.....	30
2.2.5.17	Specific Design Class kategori.....	31
2.2.5.18	Specific Design Class summarize.....	31
2.2.5.19	Specific Design Class summarizebaru.....	32
2.2.5.20	Specific Design Class loginmodel.....	32
2.2.5.21	Specific Design Class scrapmodel.....	32
2.2.5.22	Specific Design Class beritamodel.....	34
2.2.5.23	Specific Design Class tagsmodel.....	35
2.2.5.24	Specific Design Class kategorimodel.....	36
2.2.5.25	Specific Design Class summarizemodel.....	37
2.2.6	Class Diagram Specific Descriptions KUBERIN Mobile.....	37
2.2.6.1	Spesific Design Class ListBeritaActivity.....	37
2.2.6.2	Specific Design Class DetailBeritaActivity.....	38
2.2.6.3	Specific Design Class FragmentSummarize.....	38
2.2.6.4	Specific Design Class PilihBeritaActivity.....	39
2.2.6.5	Specific Design Class DetailSummarizeActivity.....	40
2.2.6.6	Specific Design Class FragmentTrendingTopik.....	40
2.2.6.7	Specific Design Class DetailTrendigTopikActivity.....	41
2.2.6.8	Specific Design Class service_kategori.....	41
2.2.6.9	Specific Design Class service_summarize.....	41
2.2.6.10	Specific Design Class service_summarize_new.....	42
2.2.6.11	Specific Design Class service_trending_topik.....	42
2.2.6.12	Specific Design Class berita_model.....	43
2.2.6.13	Specific Design Class summarize_model.....	43
3	PERANCANGAN DATA.....	45
3.1	Dekomposisi Data.....	45
3.1.1	Deskripsi Entitas Tbl_Berita.....	45
3.1.2	Deskripsi Entitas Tbl_Berita.....	45
3.1.3	Deskripsi Entitas Tbl_GetURL.....	46
3.1.4	Deskripsi Entitas Tbl_GetBerita.....	48
3.1.5	Deskripsi Entitas Tbl_KategoriBerita.....	48
3.1.6	Deskripsi Entitas Tbl_KategoriUtama.....	48
3.1.7	Deskripsi Entitas Tbl_Relasi_KatUtama_KatBerita.....	49
3.1.8	Deskripsi Entitas Tbl_Log.....	49
3.1.9	Deskripsi Entitas Tbl_DetailLog.....	50
3.2	Physical Data Model.....	51
4	Deskripsi Perancangan AntarMuka.....	52
4.1	Antarmuka Halaman Login Web.....	52
4.2	Antarmuka Halaman Scraping.....	53
4.3	Antarmuka Halaman Lihat Data Berita.....	54
4.4	Antarmuka Halaman Update Data Tags.....	55
4.5	Antarmuka Halaman Pengelolaan Kategori.....	57
4.6	Antarmuka Halaman Summarize Berita Web.....	59
4.7	Antarmuka Halaman Kategori Berita.....	60
4.8	Antarmuka Halaman Summarize Berita Mobile.....	61
4.9	Antarmuka Halaman Trending Topik.....	62

## DAFTAR GAMBAR

Gambar 2.1 Perancangan Arsitektur KUBERIN Web Application.....	10
Gambar 2.2 Perancangan Arsitektur KUBERIN Mobile Application.....	11
Gambar 2.3 Sequence Diagram : Fungsi Login.....	12
Gambar 2.4 Sequence Diagram : Fungsi Logout.....	12
Gambar 2.5 Sequence Diagram : Fungsi Test Data Scraping.....	13
Gambar 2.6 Sequence Diagram : Fungsi Simpan Data Scraping.....	14
Gambar 2.7 Sequence Diagram : Fungsi Tampil Data Scraping.....	14
Gambar 2.8 Sequence Diagram : Fungsi Update Data Scraping.....	15
Gambar 2.9 Sequence Diagram : Fungsi Lihat Data Berita.....	16
Gambar 2.10 Sequence Diagram : Fungsi Tambah Kategori Utama.....	16
Gambar 2.11 Sequence Diagram : Fungsi Pindah Kategori Berita.....	17
Gambar 2.12 Sequence Diagram : Fungsi Hubung Kategori Utama dan Kategori Berita.....	17
Gambar 2.13 Sequence Diagram : Fungsi Lihat Summary Berita Web.....	18
Gambar 2.14 Sequence Diagram : Fungsi Tampil Trending Topik.....	18
Gambar 2.15 Sequence Diagram : Fungsi Lihat Summary Berita Mobile...	19
Gambar 2.16 Sequence Diagram : Fungsi Lihat Tampil Data Berita.....	19
Gambar 2.17 Sequence Diagram : Fungsi Get URL Berita.....	20
Gambar 2.18 Sequence Diagram : Fungsi Get Data Berita.....	20
Gambar 2.19 Sequence Diagram : Fungsi Cek Tags Tidak Aktif.....	21
Gambar 2.20 Class Diagram Aplikasi Website.....	22
Gambar 2.21 Class Diagram Aplikasi Mobile.....	23
Gambar 3.1 Physical Data Model.....	51
Gambar 4.1 Rancangan Antarmuka Halaman Login Web.....	52
Gambar 4.2 Rancangan Antarmuka Halaman Scraping.....	53
Gambar 4.3 Rancangan Antarmuka Halaman Lihat Data Berita.....	54
Gambar 4.4 Rancangan Antarmuka Halaman Update Data Tags.....	55
Gambar 4.5 Rancangan Antarmuka Halaman Pengelolaan Kategori.....	57
Gambar 4.6 Rancangan Antarmuka Halaman Summarize Berita Web.....	59
Gambar 4.7 Rancangan Antarmuka Halaman Kategori Berita.....	60
Gambar 4.8 Rancangan Antarmuka Halaman Summarize Berita Mobile.....	61
Gambar 4.9 Rancangan Antarmuka Halaman Trending Topik.....	62

## 1. Pendahuluan

### 1.1 Tujuan

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) ini bertujuan untuk mendefinisikan perancangan perangkat lunak yang akan dikembangkan. Dokumen tersebut akan digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap berikutnya.

### 1.2 Lingkup Masalah

Perangkat Lunak KUBERIN dikembangkan dengan tujuan untuk :

1. Melakukan proses web scraping untuk mengambil data berita dari situs berita.
2. Mengelola data berita yang telah diambil dengan teknik web scraping.
3. Mengurangi pencarian data berita.
4. Mengkategorikan berita.
5. Melakukan *summary* berita.
6. Memudahkan user dalam membaca berita.

Perangkat Lunak KUBERIN bisa berjalan pada lingkungan dengan mobile dan web.

Program Studi Teknik Informatika	DPPL – KUBERIN	7 / 62
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

### 1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
DPPL	Deskripsi Perancangan Perangkat Lunak disebut juga Software Design Description (SDD). Merupakan deskripsi dari perancangan produk /perangkat lunak yang akan dikembangkan.
DPPL-KUBERIN-XXX	Kode yang merepresentasikan kebutuhan pada KUBERIN (Kumpulan Berita Indonesia) di mana XXX merupakan nomor fungsi produk.
KUBERIN	Perangkat lunak mobile dan web yang dibangun.
Internet	Internet merupakan istilah umum yang dipakai untuk menunjuk <i>Network</i> global yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World Wide Web.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
Web Scraping	Mendapatkan informasi dalam sebuah website dengan cara mengekstrak seluruh isi data dari sebuah page website.
Website	Sekumpulan halaman informasi yang disediakan melalui jalur internet sehingga bisa diakses di seluruh dunia selama terkoneksi dengan jaringan internet.

Situs Berita	Website yang menampilkan berita
Berita	Laporan dan informasi mengenai sesuatu peristiwa atau kejadian yang sedang terjadi
Summarize Berita	Merangkum beberapa berita atau artikel menjadi satu.

#### 1.4 Referensi

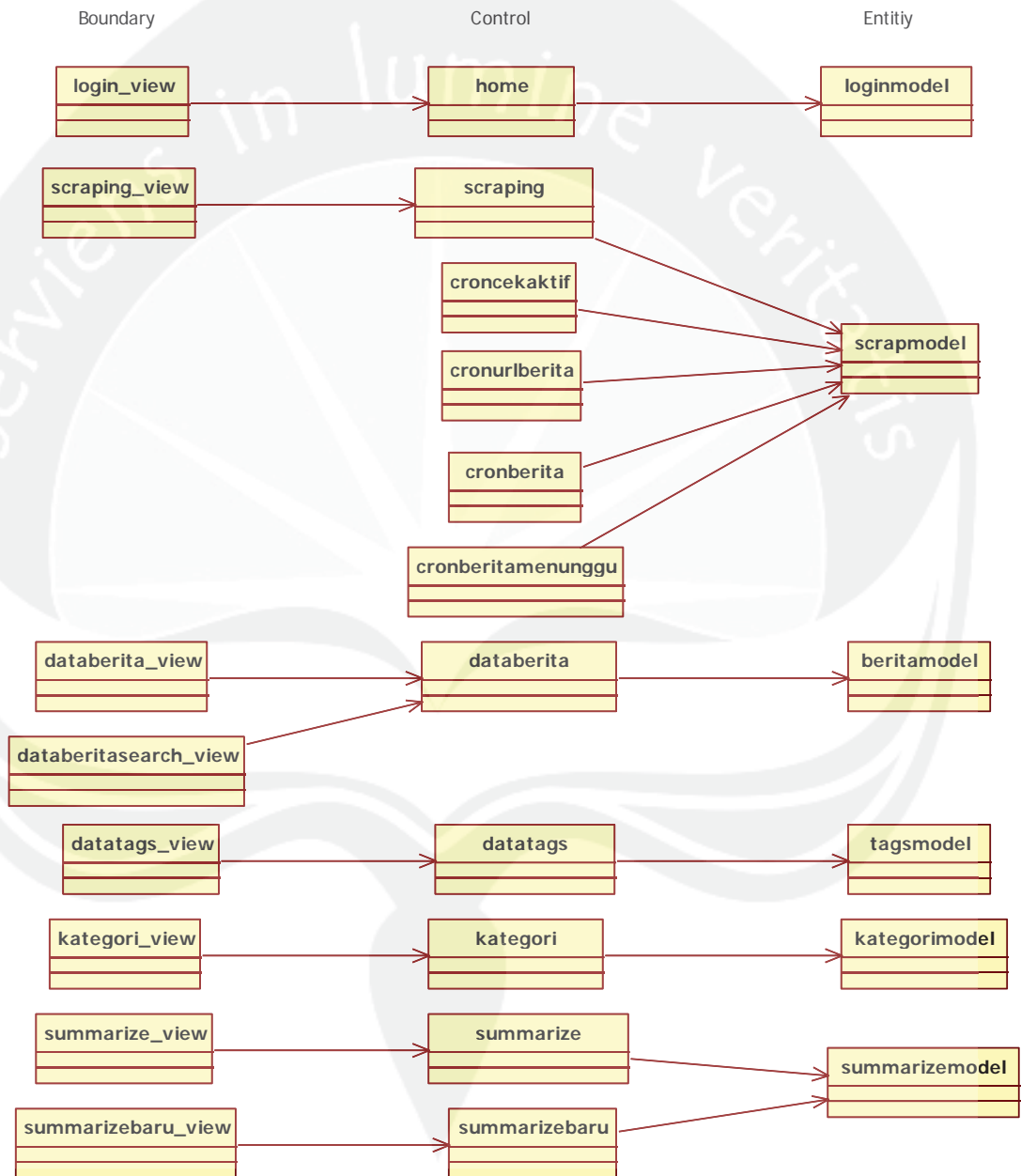
Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Yustinus Andry Setiawan / 6659, *Deskripsi Perancangan Perangkat Lunak (DPPL) MEXTEM*, Universitas Atma Jaya Yogyakarta.
2. Yustinus Andry Setiawan / 6659, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) MEXTEM*, Universitas Atma Jaya Yogyakarta.
3. Yustinus Andry Setiawan / 6659, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) KUBERIN*, Universitas Atma Jaya Yogyakarta.

## 2 Perancangan Sistem

### 2.1 Perancangan Asitektur

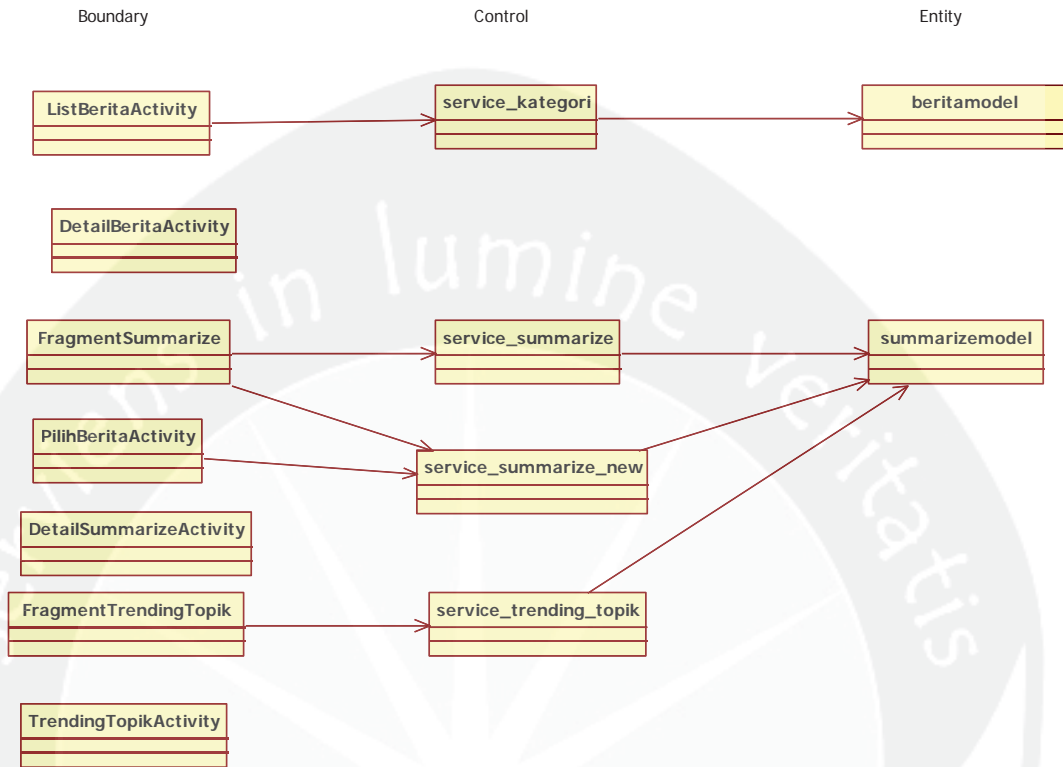
#### 2.1.1 Web Application



**Gambar 2.1 Perancangan Arsitektur KUBERIN Web Application**



## 2.1.2 Mobile Application

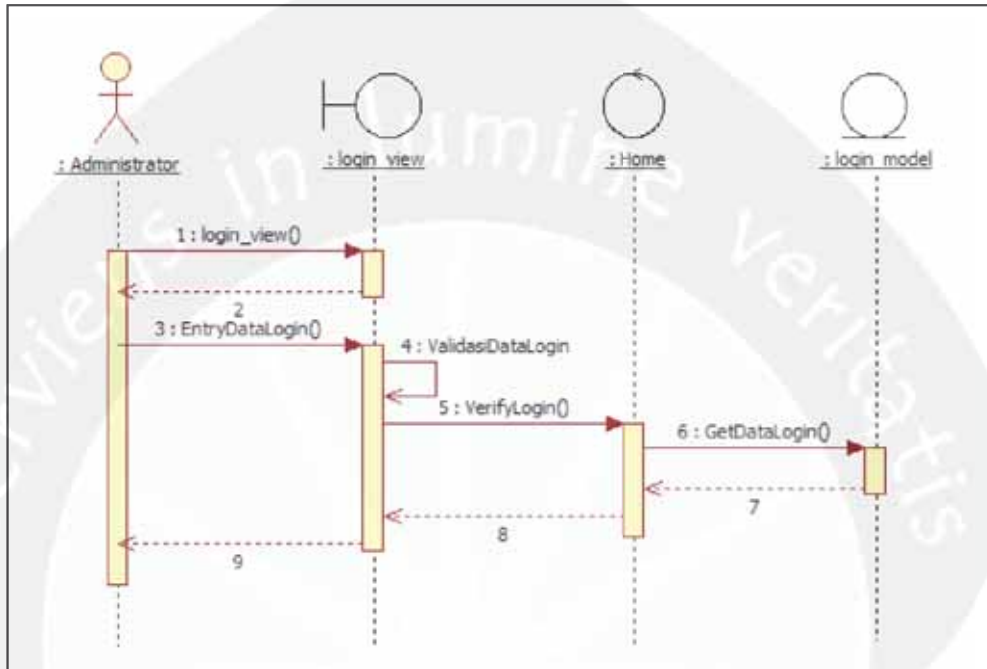


Gambar 2.2 Perancangan Arsitektur KUBERIN Mobile Application

## 2.2 Perancangan Rinci

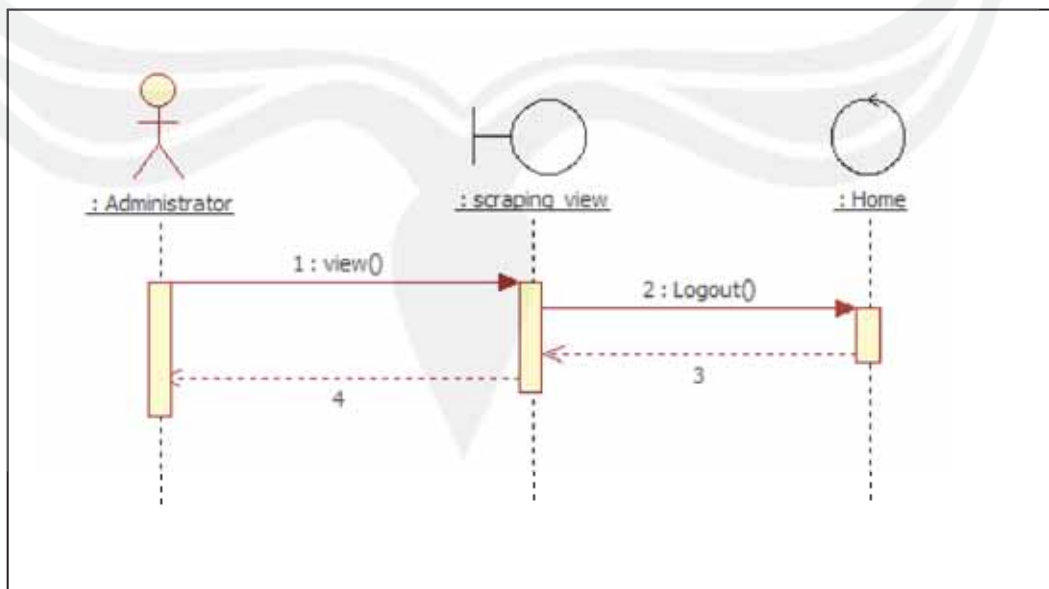
### 2.2.1 Sequence Diagram KUBERIN Website

#### 2.2.1.1 Fungsi Login



Gambar 2.3 Sequence Diagram Fungsi Login

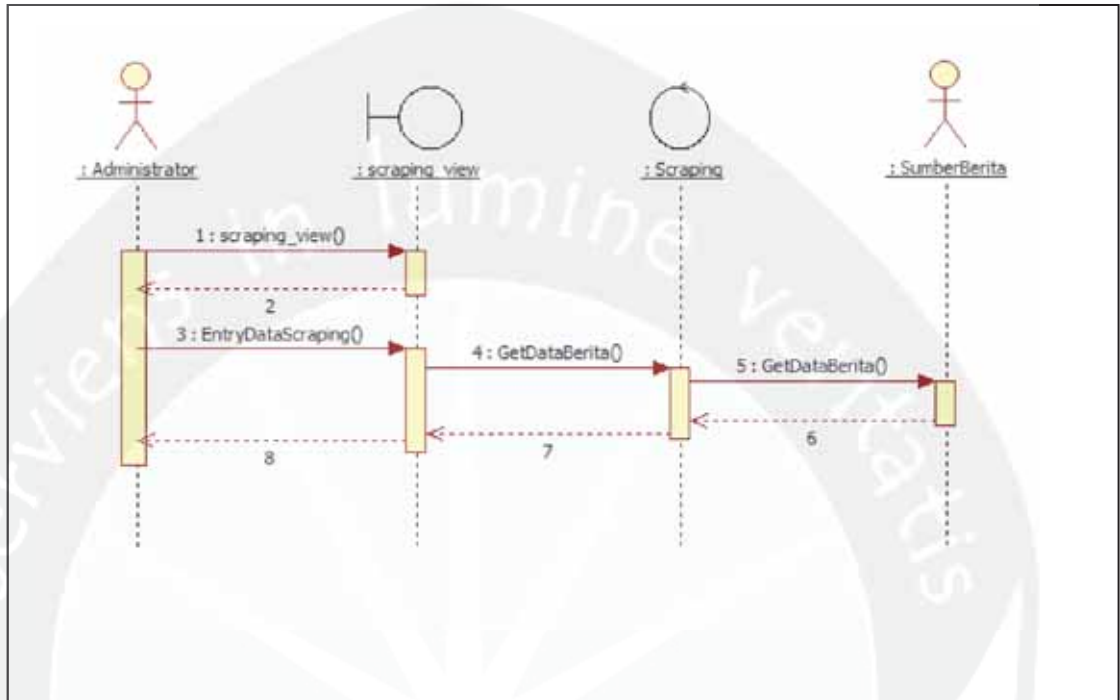
#### 2.2.1.2 Fungsi Logout



Gambar 2.4 Sequence Diagram Fungsi Logout

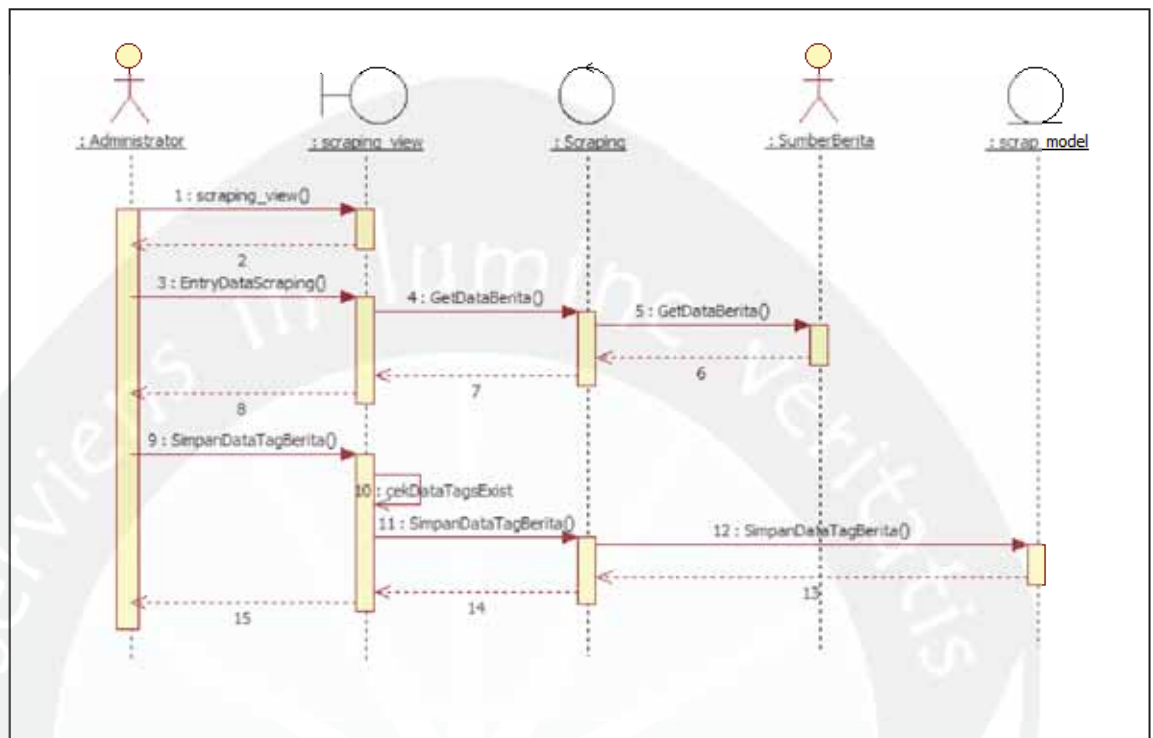
### 2.2.1.3 Fungsi Pengelolaan Data Scraping Berita

#### 2.2.1.3.1 Fungsi Test Data Scraping



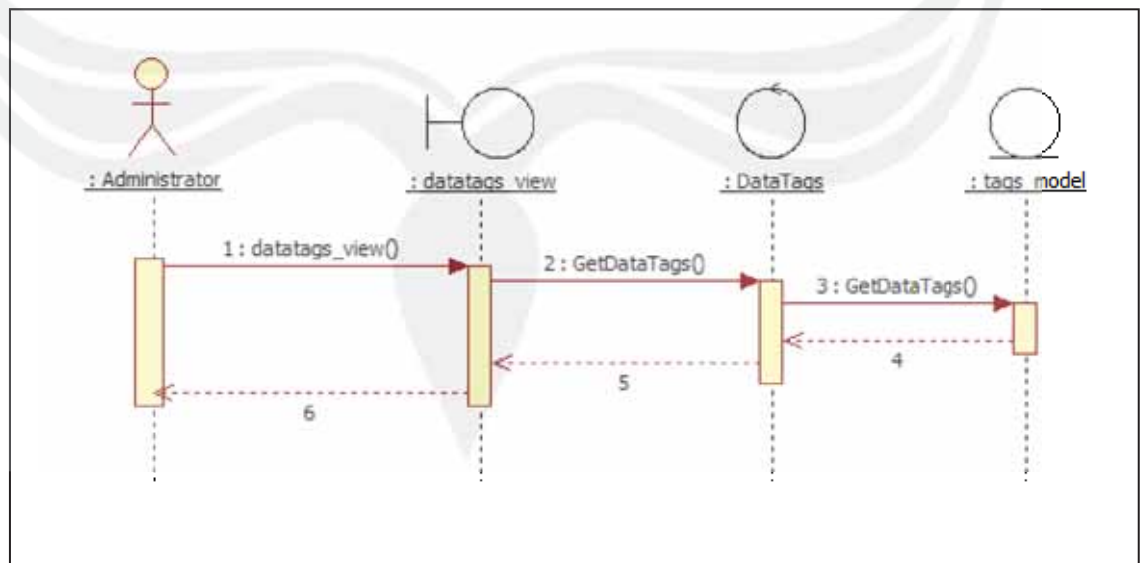
Gambar 2.5 Sequence Diagram Fungsi Test Data Scraping

### 2.2.1.3.2 Fungsi Simpan Data Scraping



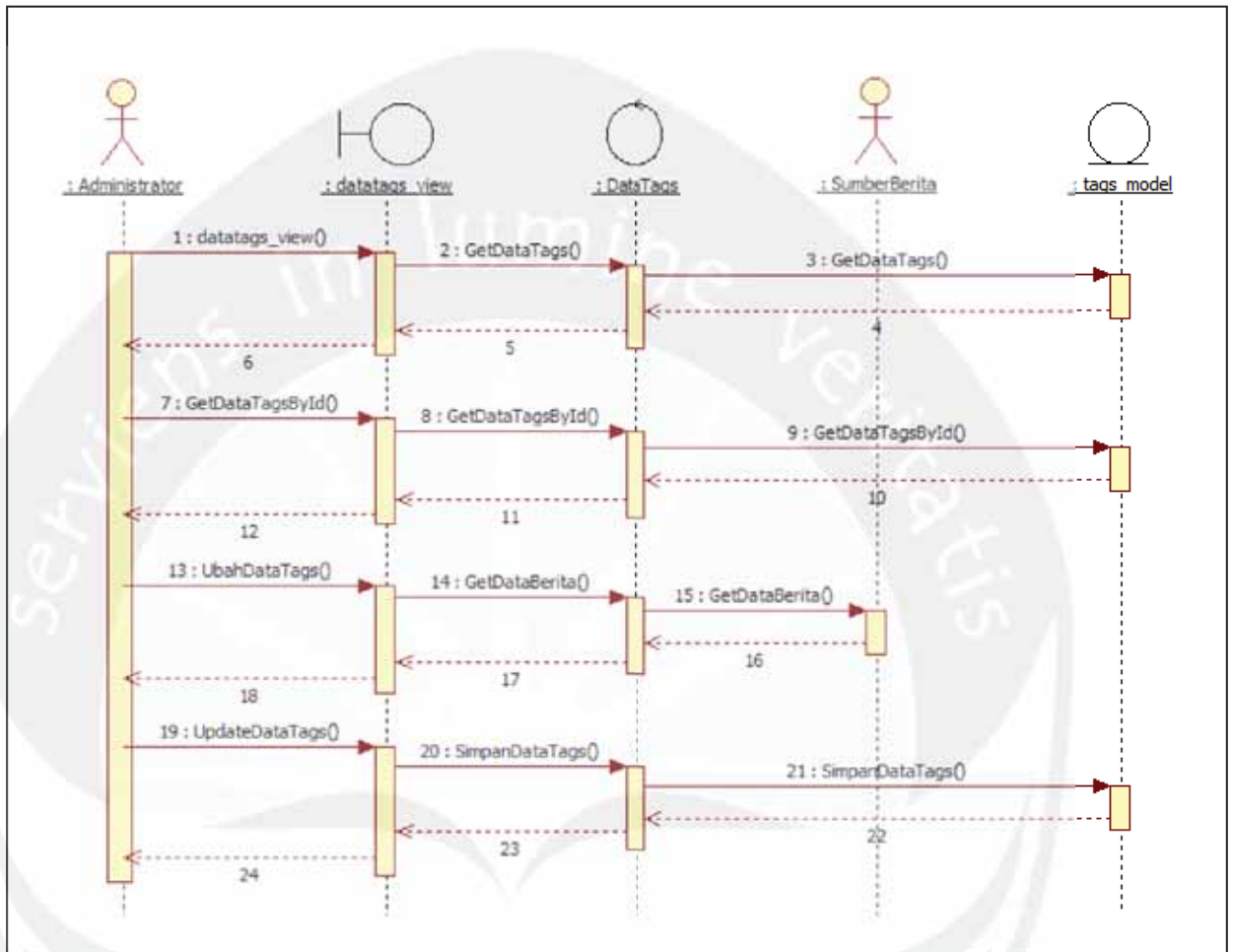
Gambar 2.6 Sequence Diagram Fungsi Simpan Data Scraping

### 2.2.1.3.3 Fungsi Tampil Data Scraping



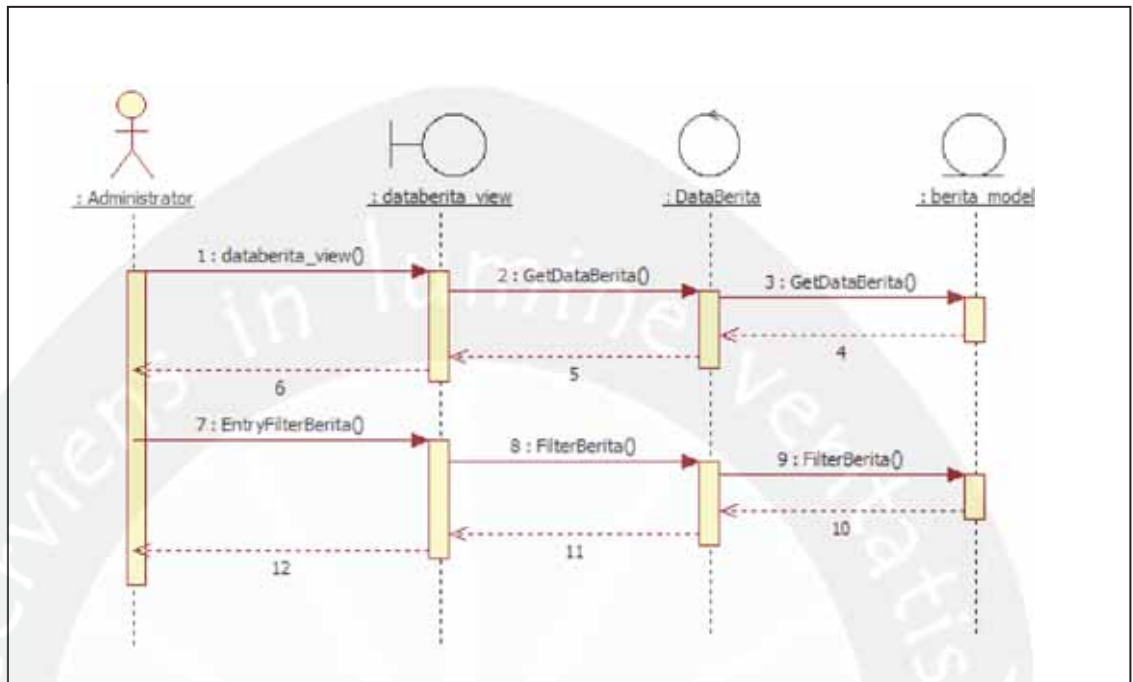
Gambar 2.7 Sequence Diagram Fungsi Tampil Data Scraping

### 2.2.1.3.4 Fungsi Update Data Scraping



Gambar 2.8 Sequence Diagram Fungsi Update Data Scraping

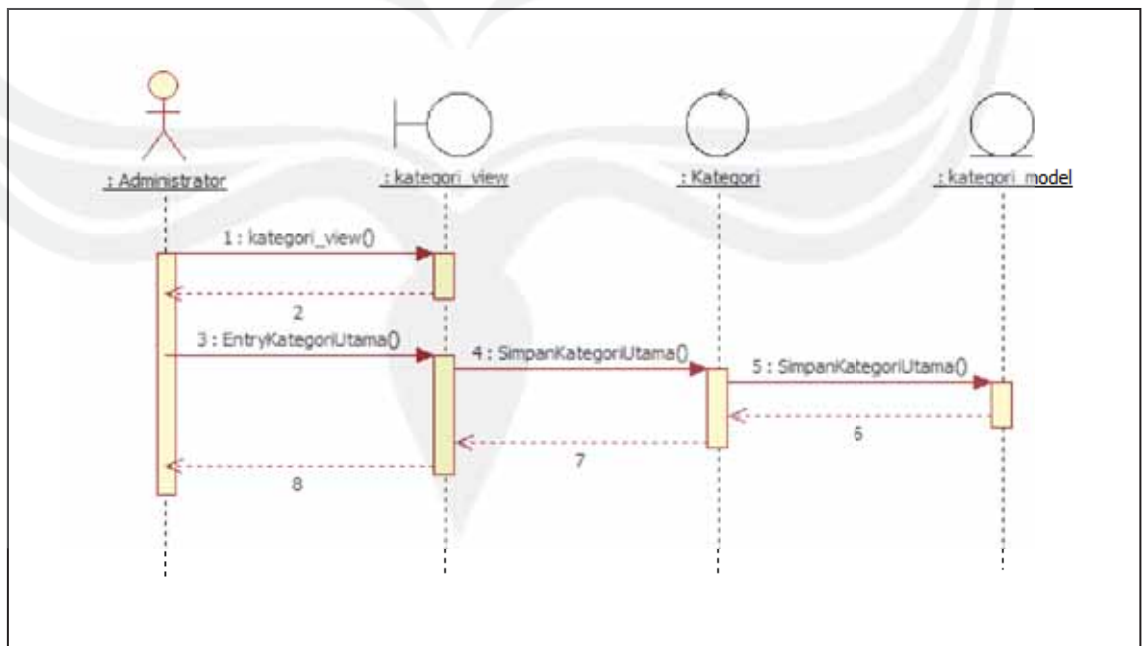
#### 2.2.1.4 Fungsi Lihat Data Berita



Gambar 2.9 Sequence Diagram Fungsi Lihat Data Berita

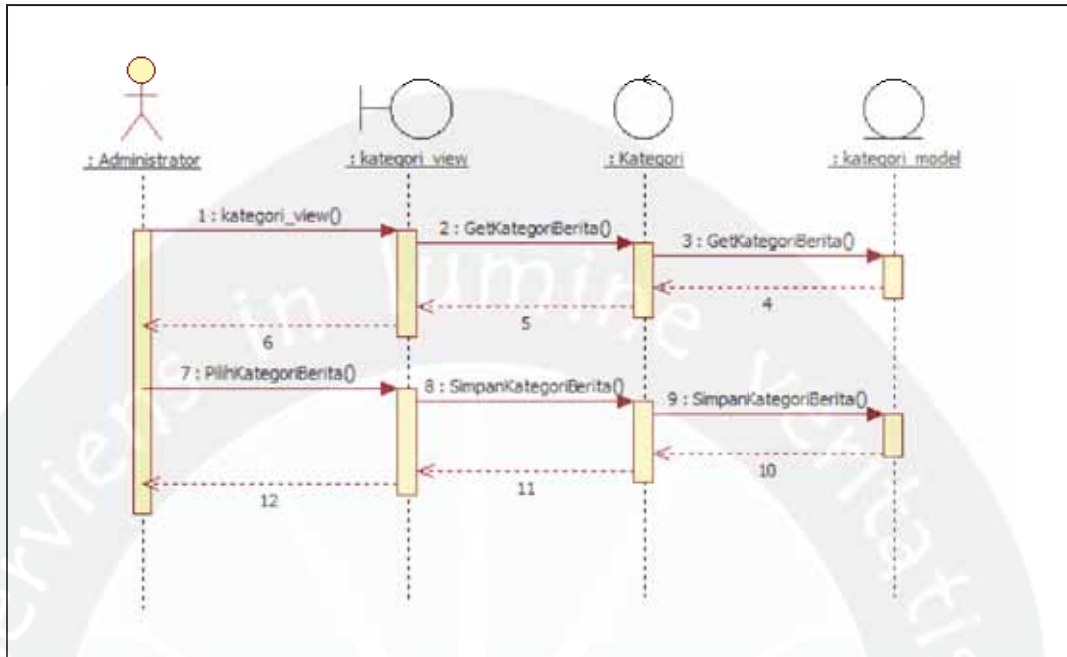
#### 2.2.1.5 Fungsi Pengelolaan Kategori

##### 2.2.1.5.1 Fungsi Tambah Kategori Utama



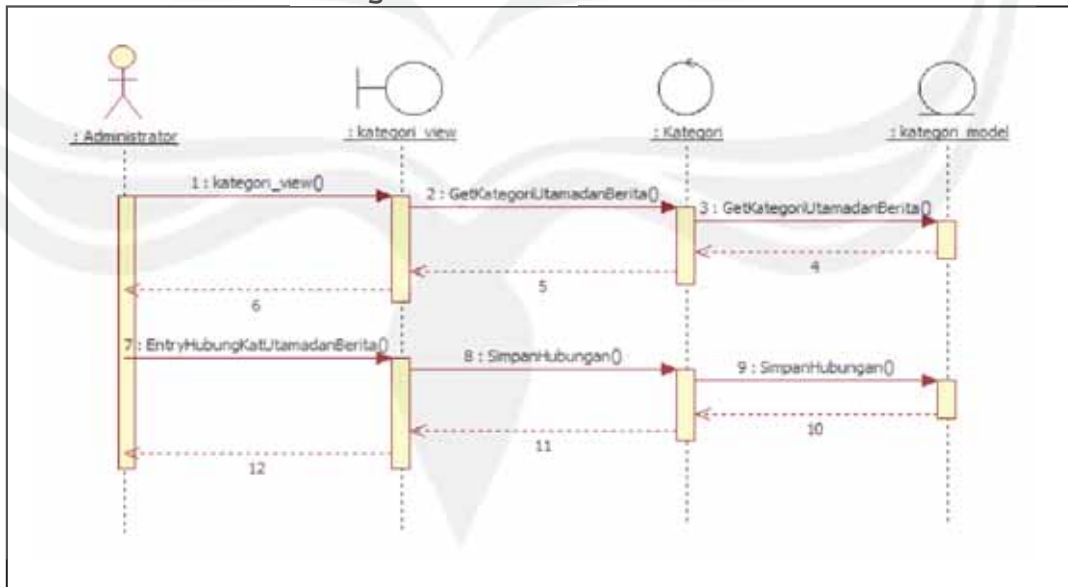
Gambar 2.10 Sequence Diagram Fungsi Tambah Kategori Utama

### 2.2.1.5.2 Fungsi Pindah Kategori Berita



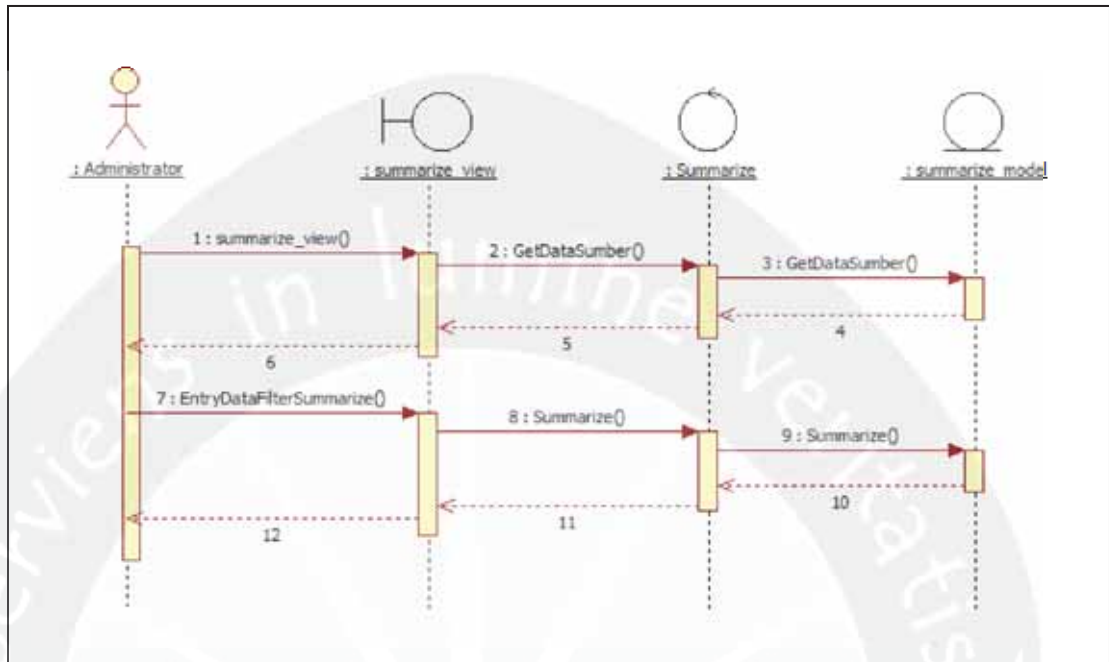
Gambar 2.11 Sequence Diagram Fungsi Pindah Kategori Berita

### 2.2.1.5.3 Fungsi Hubung Kategori Utama dan Kategori Berita



Gambar 2.12 Sequence Diagram Fungsi Hubung Kategori Utama dan Kategori Berita

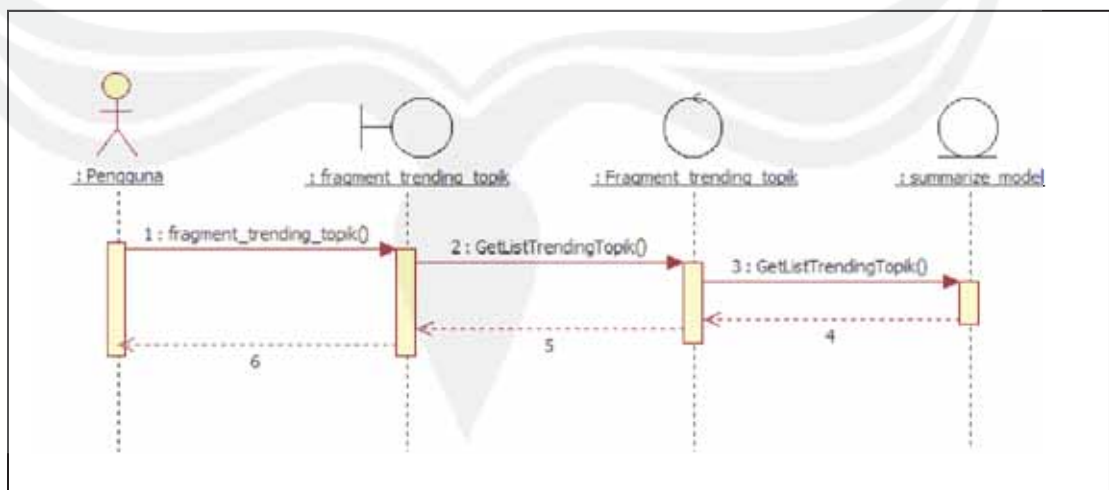
### 2.2.1.6 Fungsi Lihat Summary Berita Web



Gambar 2.13 Sequence Diagram Fungsi Lihat Summary Berita Web

### 2.2.2 Sequence Diagram KUBERIN Mobile

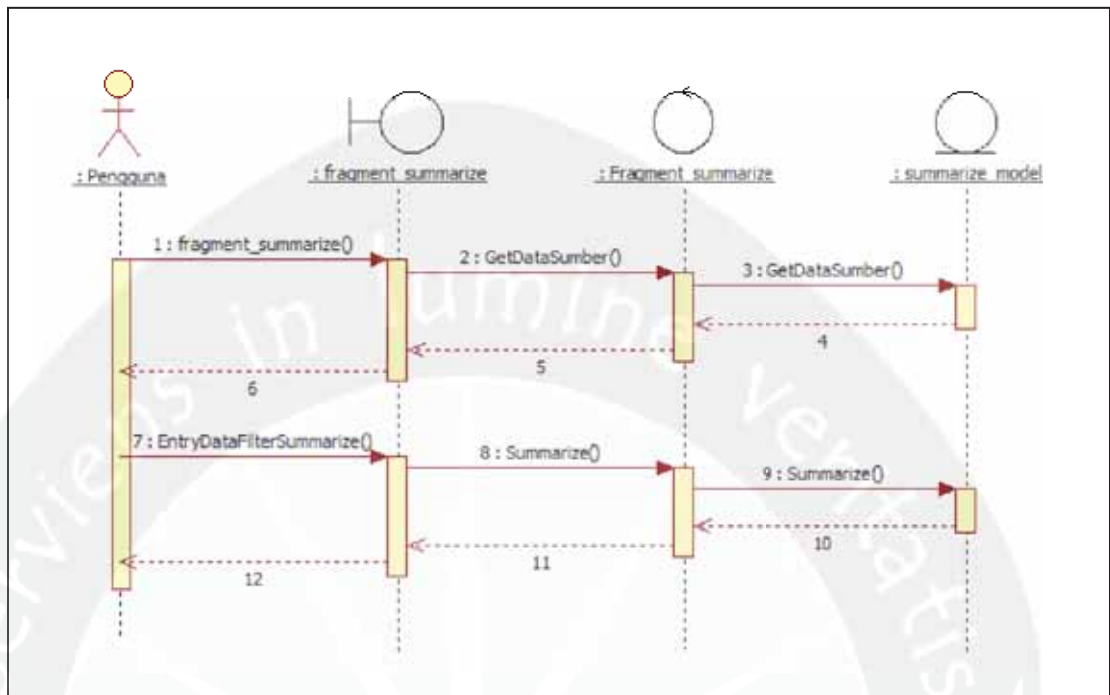
#### 2.2.2.1 Fungsi Tampil Trending Topik



Gambar 2.14 Sequence Diagram Fungsi Tampil Trending Topik

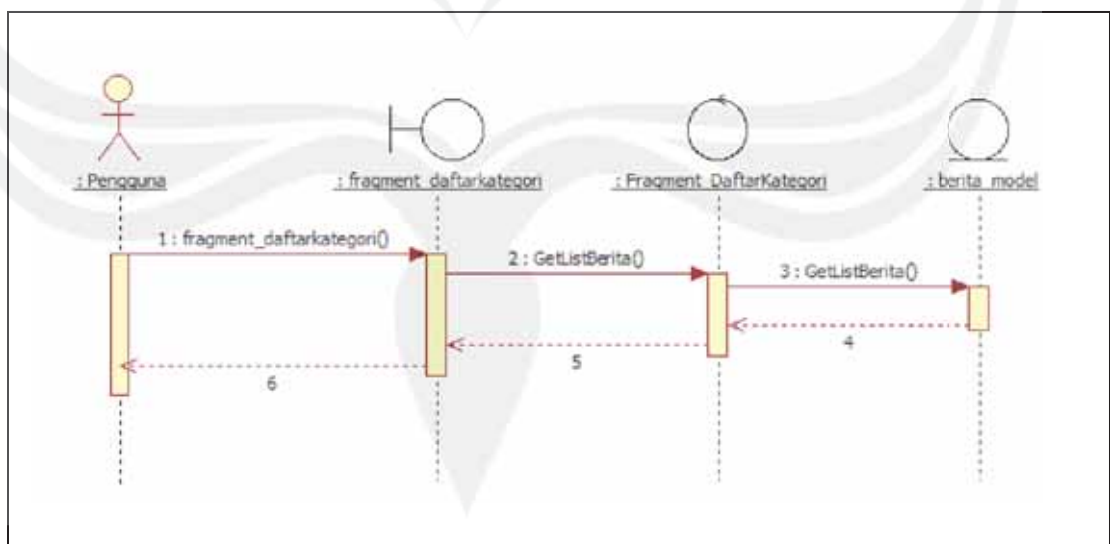


### 2.2.2.2 Fungsi Lihat Summary Berita Mobile



Gambar 2.15 Sequence Diagram Fungsi Lihat Summary Berita Mobile

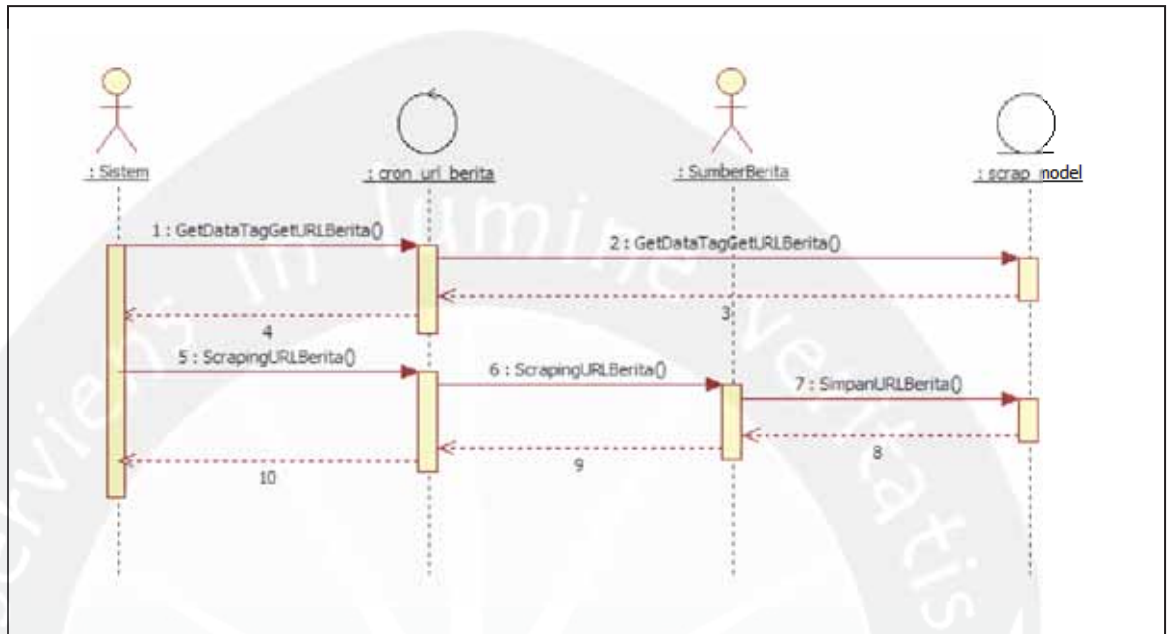
### 2.2.2.3 Fungsi Tampil Data Berita



Gambar 2.16 Sequence Diagram Fungsi Tampil Data Berita

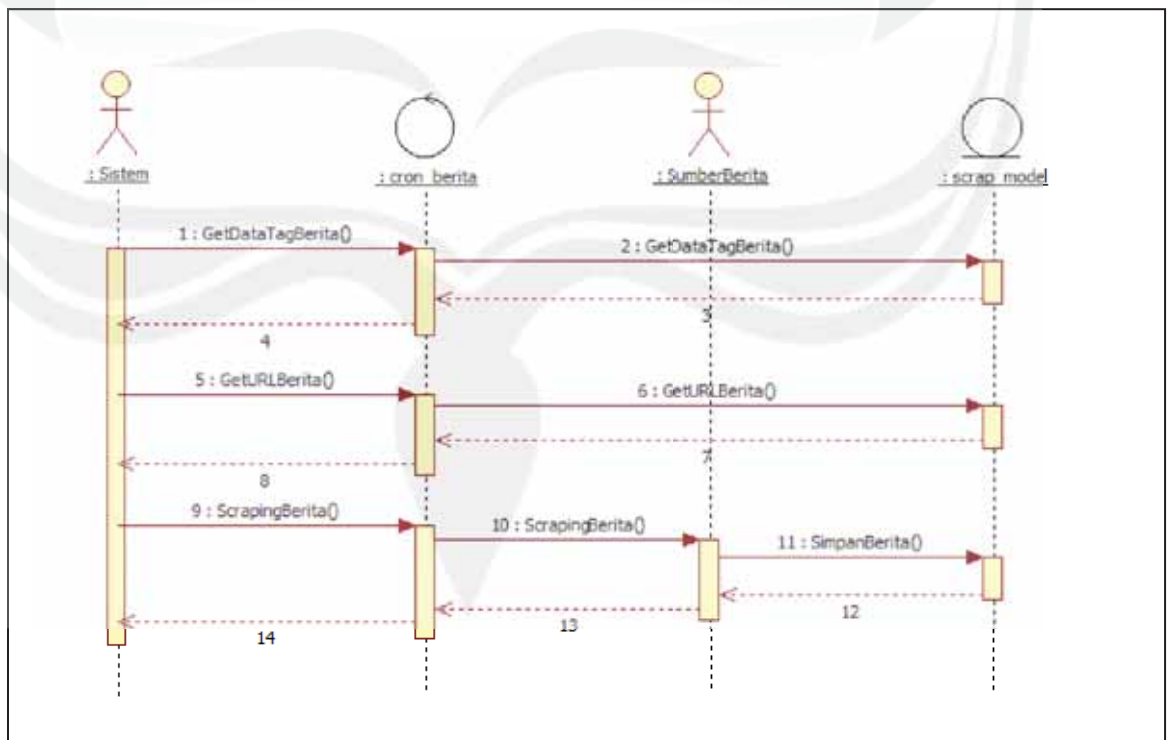
### 2.2.3 Sequence Diagram KUBERIN Web Scraper

#### 2.2.3.1 Fungsi Get URL Berita



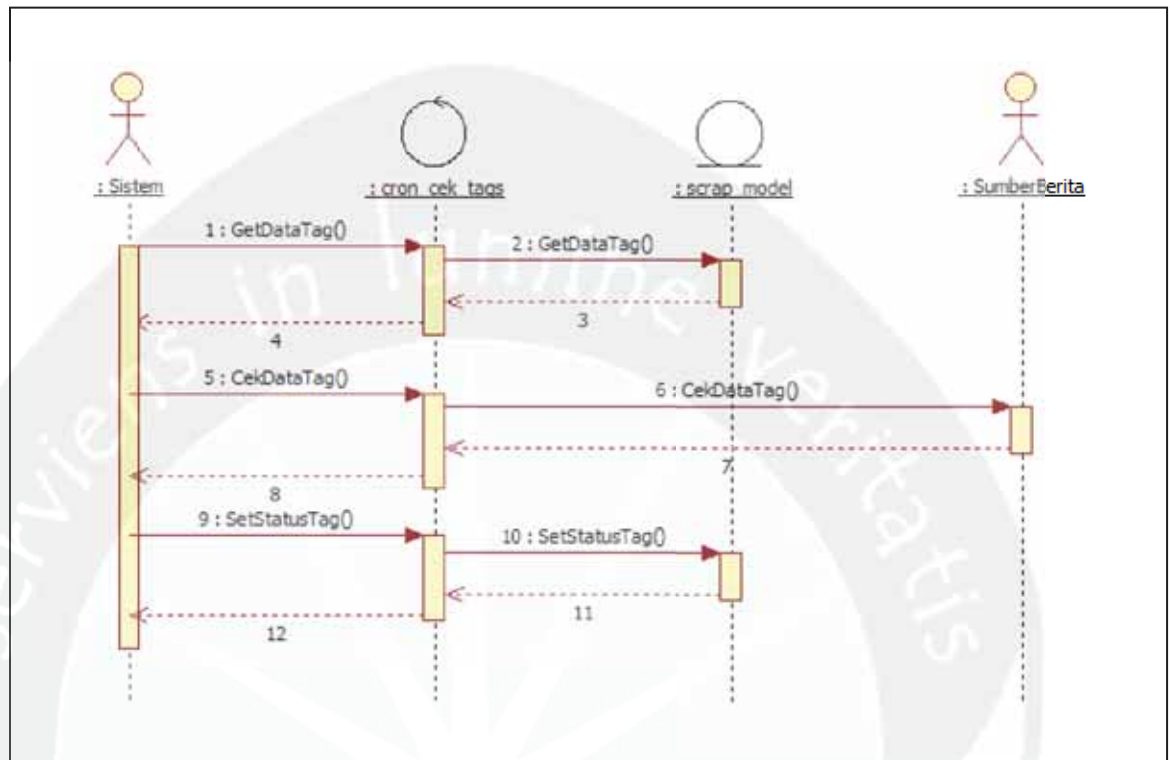
Gambar 2.17 Sequence Diagram Fungsi Get URL Berita

#### 2.2.3.2 Fungsi Get Data Berita



Gambar 2.18 Sequence Diagram Fungsi Get Data Berita

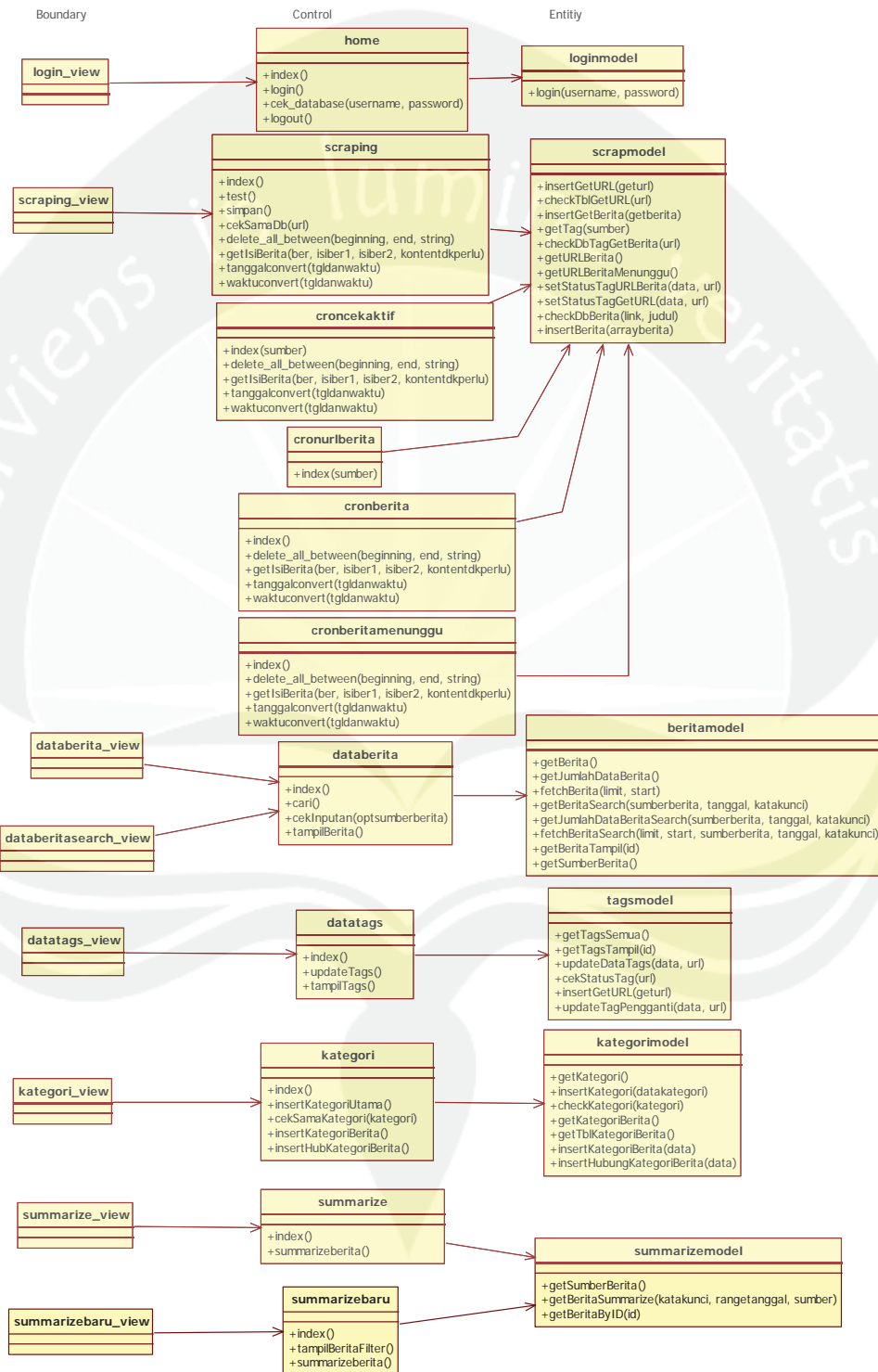
### 2.2.3.3 Fungsi Cek Tags Tidak Aktif



Gambar 2.19 Sequence Diagram Fungsi Cek Tags Tidak Aktif

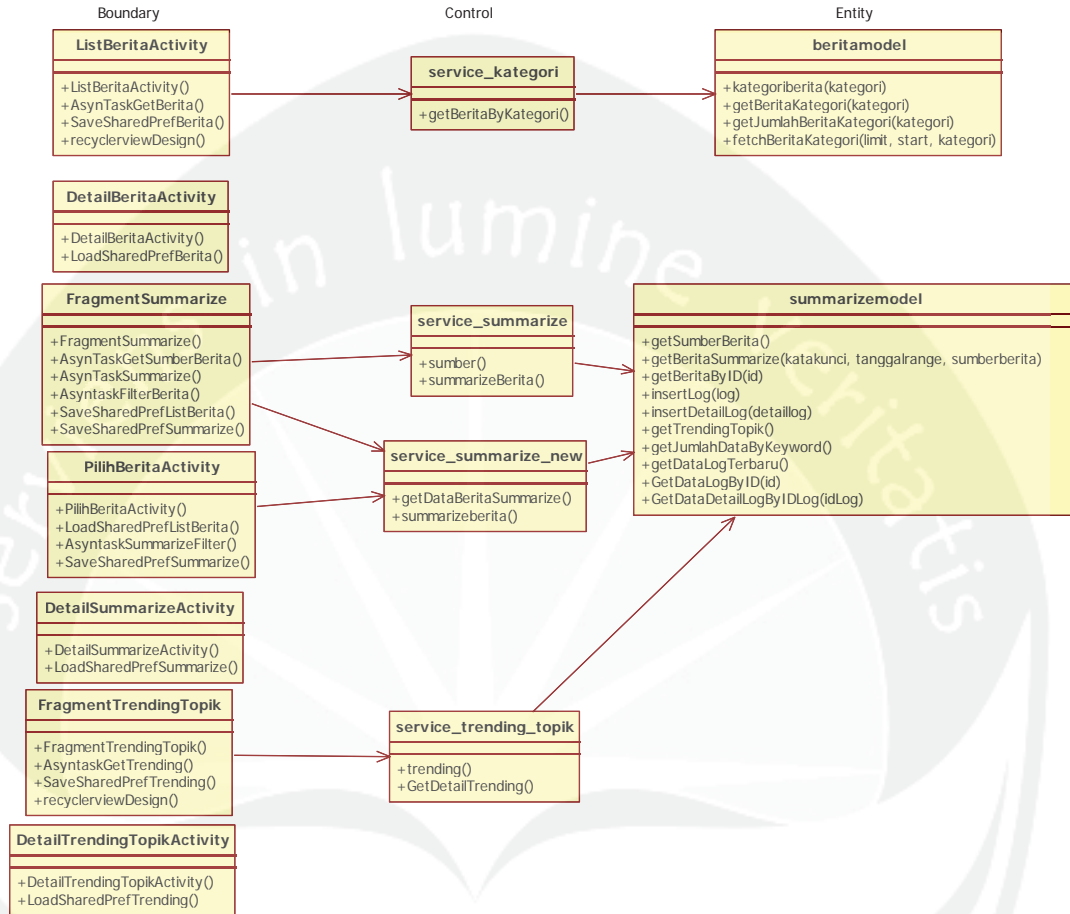
## 2.2.4 Class Diagram

### 2.2.4.1 Class Diagram Aplikasi Website



Gambar 2.20 Class Diagram Aplikasi Website

### 2.2.4.2 Class Diagram Aplikasi Mobile



Gambar 2.21 Class Diagram Aplikasi Mobile

## 2.2.5 Class Diagram Specific Descriptions KUBERIN Website

### 2.2.5.1 Specific Design Class login\_view

login_view	<<boundary>>

### 2.2.5.2 Specific Design Class scraping\_view

scraping_view	<<boundary>>

### 2.2.5.3 Specific Design Class databerita\_view

databerita_view	<<boundary>>

### 2.2.5.4 Specific Design Class databeritasearch\_view

databeritasearch_view	<<boundary>>

### 2.2.5.5 Specific Design Class datatags\_view

datatags_view	<<boundary>>

### 2.2.5.6 Specific Design Class kategori\_view

kategori_view	<<boundary>>

#### 2.2.5.7 Specific Design Class summarize\_view

summarize_view	<<boundary>>

#### 2.2.5.8 Specific Design Class summarizebaru\_view

summarizebaru_view	<<boundary>>

#### 2.2.5.9 Specific Design Class home

Home	<<control>>
<pre>+index() Operasi ini digunakan untuk mengeset view dan mengecek session login. +login() Operasi ini digunakan untuk masuk ke dalam sistem atau login. +cek_database(username,password) Operasi ini digunakan untuk mengecek username dan password yang diinputkan dengan username dan password yang ada di database. +logout() Operasi ini digunakan untuk keluar dari sistem atau logout.</pre>	

### 2.2.5.10 Specific Design Class scraping

Scraping	<<control>>
<pre> +index() Operasi ini digunakan untuk mengeset view dan mengecek session login. +test() Operasi ini digunakan untuk mengetes pengambilan berita atau proses scraping dan hasil test ditampilkan dalam halaman scraping. +simpan() Operasi ini digunakan untuk menyimpan tags yang berhasil di test ke dalam database. +cekSamaDb(url) Operasi ini digunakan untuk mengecek apakah data tags yang diinputkan sama dengan yang ada di database dengan parameter url. +delete_all_between(beginning,end,string) Operasi ini digunakan untuk menghapus string berdasarkan parameter beginning atau awal string, end atau akhir string dan string. +getIsiBerita(ber,isiber1,isiber2,kontentdkperlu) Operasi ini digunakan untuk mengambil isi berita dengan teknik web scraping. Pada fungsi ini terdapat parameter ber, isiber1, isiber2, dan kontentdkperlu. +tanggalconvert(tgldanwaktu) Operasi ini digunakan untuk mengkonversi tanggal dengan parameter tanggaldanwaktu yang di dapatkan dari proses scraping. +waktuconvert(tgldanwaktu) Operasi ini digunakan untuk mengkonversi waktu dengan </pre>	



parameter tanggal dan waktu yang di dapatkan dari proses scraping.

#### 2.2.5.11 Specific Design Class Croncekaktif

Croncekaktif	<<control>>
<pre>+index(sumber)</pre> <p>Operasi ini digunakan untuk mengecek tags berita apakah aktif atau tidak aktif untuk mendapatkan berita. Operasi ini juga mengeset status ketika selesai mengecek statusnya. Pengecekan tags berita berdasarkan sumber berita.</p>	
<pre>+delete_all_between(beginning,end,string)</pre> <p>Operasi ini digunakan untuk menghapus string berdasarkan parameter beginning atau awal string, end atau akhir string dan string.</p>	
<pre>+getIsiBerita(ber,isiber1,isiber2,kontentdkperlu)</pre> <p>Operasi ini digunakan untuk mengambil isi berita dengan teknik web scraping. Pada fungsi ini terdapat parameter ber, isiber1, isiber2, dan kontentdkperlu.</p>	
<pre>+tanggalconvert(tgldanwaktu)</pre> <p>Operasi ini digunakan untuk mengkonversi tanggal dengan parameter tanggal dan waktu yang di dapatkan dari proses scraping.</p>	
<pre>+waktuconvert(tgldanwaktu)</pre> <p>Operasi ini digunakan untuk mengkonversi waktu dengan parameter tanggal dan waktu yang di dapatkan dari proses scraping.</p>	

### 2.2.5.12 Specific Design Class cronurlberita

<b>cronurlberita</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+index(sumber)</pre> <p>Operasi ini digunakan untuk mendapatkan URL Berita dari halaman index situs berita. URL didapatkan dengan proses scraping. Data tags untuk mengambil URL berita yang dipanggil berdasarkan sumber berita.</p>	

### 2.2.5.13 Specific Design Class cronberita

<b>cronberita</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+index()</pre> <p>Operasi ini digunakan untuk mendapatkan data berita yaitu judul, tanggal, waktu, kategori, gambar, isiberita, link, sumberberita, lama waktu untuk mendapatkan berita, dan tanggal mendapatkan berita. Data berita didapatkan dengan proses scraping.</p> <pre>+delete_all_between(beginning,end,string)</pre> <p>Operasi ini digunakan untuk menghapus string berdasarkan parameter beginning atau awal string, end atau akhir string dan string.</p> <pre>+getIsiBerita(ber,isiber1,isiber2,kontentdkperlu)</pre> <p>Operasi ini digunakan untuk mengambil isi berita dengan teknik web scraping. Pada fungsi ini terdapat parameter ber, isiber1, isiber2, dan kontentdkperlu.</p> <pre>+tanggalconvert(tgldanwaktu)</pre> <p>Operasi ini digunakan untuk mengkonversi tanggal dengan parameter tanggal dan waktu yang di dapatkan dari proses scraping.</p>	

+waktuconvert(tgldanwaktu)

Operasi ini digunakan untuk mengkonversi waktu dengan parameter tanggaldanwaktu yang di dapatkan dari proses scraping.

#### 2.2.5.14 Specific Design Class cronberitamenu

cronberitamenu

<<control>>

+index()

Operasi ini digunakan untuk mendapatkan data berita yaitu judul, tanggal, waktu, kategori, gambar, isiberita, link, sumberberita, lama waktu untuk mendapatkan berita, dan tanggal mendapatkan berita. Data berita didapatkan dengan proses scraping.

+delete\_all\_between(beginning,end,string)

Operasi ini digunakan untuk menghapus string berdasarkan parameter beginning atau awal string, end atau akhir string dan string.

+getIsiBerita(ber,isiber1,isiber2,kontentdkperlu)

Operasi ini digunakan untuk mengambil isi berita dengan teknik web scraping. Pada fungsi ini terdapat parameter ber, isiber1, isiber2, dan kontentdkperlu.

+tanggalconvert(tgldanwaktu)

Operasi ini digunakan untuk mengkonversi tanggal dengan parameter tanggaldanwaktu yang di dapatkan dari proses scraping.

+waktuconvert(tgldanwaktu)

Operasi ini digunakan untuk mengkonversi waktu dengan parameter tanggaldanwaktu yang di dapatkan dari proses scraping.

#### 2.2.5.15 Specific Design Class databerita

<b>Databerita</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+index() Operasi ini digunakan untuk mengeset view, menampilkan data berita dalam bentuk pagination, dan mengecek session login.  +cari() Operasi ini digunakan untuk memfilter data berita dan menampilkannya ke dalam bentuk tabel pagination.  +cekInputan() Operasi ini digunakan untuk mengecek inputan filter berita.  +tampilBerita() Operasi ini digunakan untuk menampilkan detail data berita.</pre>	

#### 2.2.5.16 Specific Design Class datatags

<b>Datatags</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+index() Operasi ini digunakan untuk mengeset view, menampilkan data tags, dan mengecek session login.  +updateTags Operasi ini digunakan untuk mengupdate data tags. Sebelum mengupdate tags harus mengetes tags terlebih dahulu baru bisa di update.  +tampilTags Operasi ini digunakan untuk mendapatkan data tags dari database dan menampilkan tags ke dalam form.</pre>	

### 2.2.5.17 Specific Design Class kategori

<b>Kategori</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+index() Operasi ini digunakan untuk mengeset view dan mengecek session login. +insertKategoriUtama() Operasi ini digunakan untuk menambahkan kategori utama ke dalam database. +cekSamaKategori(kategori) Operasi ini digunakan untuk mengecek data kategori utama dengan data kategori utama yang ada di database. +insertKategoriBerita() Operasi ini digunakan untuk menambahkan kategori berita yang didapat dari tabel berita ke tabel kategori berita. +insertHubKategoriBerita() Operasi ini digunakan untuk menambahkan relasi antara kategori berita dan kategori utama di database.</pre>	

### 2.2.5.18 Specific Design Class summarize

<b>Summarize</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+index() Operasi ini digunakan untuk mengeset view dan mengecek session login. +summarizeberita() Operasi ini digunakan untuk melakukan summarize berita dan menampilkannya pada halaman summarize.</pre>	

#### 2.2.5.19 Specific Design Class summarizebaru

<b>summarizebaru</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+index() Operasi ini digunakan untuk mengeset view dan mengecek session login. +tampilBeritaFilter() Operasi ini digunakan untuk mendapatkan data berita berdasarkan inputan filter berita oleh user. +summarizeberita() Operasi ini digunakan untuk melakukan summarize berita dan menampilkannya pada halaman summarize.</pre>	

#### 2.2.5.20 Specific Design Class loginmodel

<b>Loginmodel</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<pre>+login(username, password) Operasi ini digunakan untuk mengambil data login yang mempunyai username dan password sesuai parameter.</pre>	

#### 2.2.5.21 Specific Design Class scrapmodel

<b>Scrapmodel</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<pre>+insertGetURL(geturl) Operasi ini digunakan untuk memasukkan data taggeturl ke database dengan parameter array geturl. +checkTblGetURL(url) Operasi ini digunakan untuk mengecek inputan geturl apakah ada di database atau tidak.</pre>	

```
+insertGetBerita(getberita)
```

Operasi ini digunakan untuk memasukkan data taggetberita ke database dengan parameter array getberita.

```
+getTag(sumber)
```

Operasi ini digunakan untuk mengambil data tag dari database berdasarkan sumber berita.

```
+checkTblTagGetBerita(url)
```

Operasi ini digunakan untuk mengecek apakah data taggetberita sudah ada di database atau belum.

```
+getURLBerita()
```

Operasi ini digunakan untuk mendapatkan URL berita dari database.

```
+getURLBeritaMenunggu()
```

Operasi ini digunakan untuk mendapatkan URL berita dari database.

```
+setStatusTagURLBerita(data,url)
```

Operasi ini digunakan untuk mengeset status taggetberita atau update status dengan parameter url dan array data.

```
+setStatusTagGetURL(data,url)
```

Operasi ini digunakan untuk mengeset status taggeturl atau update status dengan parameter url dan array data.

```
+checkDbBerita(link,judul)
```

Operasi ini digunakan untuk mengecek apakah data berita sudah ada di database atau belum dengan parameter judul dan link.

```
+insertBerita(berita)
```

Operasi ini digunakan untuk memasukkan data berita ke dalam database dengan parameter array berita.

### 2.2.5.22 Specific Design Class beritamodel

<b>beritamodel</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<pre> +getBerita() Operasi ini digunakan untuk mendapatkan data berita dari database. +getJumlahBerita() Operasi ini digunakan untuk menghitung jumlah berita pada database. +fetchBerita(limit, start) Operasi ini digunakan untuk mengambil data berita dari database dengan batas pengambilan data dan awal data diambil. +getBeritaSearch(sumberberita, tanggal, katakunci) Operasi ini digunakan untuk mengambil data berita dari database berdasarkan filter berita yaitu sumberberita, tanggal, dan katakunci. +getJumlahBeritaSearch(sumberberita, tanggal, katakunci) Operasi ini digunakan untuk menghitung jumlah berita dari database berdasarkan filter berita yaitu sumberberita, tanggal, dan katakunci. +fetchBeritaSearch(limit, start, sumberberita, tanggal) Operasi ini digunakan untuk mengambil data berita dari database dengan batas pengambilan data dan awal data diambil berdasarkan filter berita yaitu sumberberita, tanggal, dan katakunci. +getBeritaTampil(id) Operasi ini digunakan untuk mengambil data detail berita dari database berdasarkan id. </pre>	



```
+getSumberBerita()
```

Operasi ini digunakan untuk mengambil sumber berita dari database.

#### 2.2.5.23 Specific Design Class tagsmodel

Tagsmodel	<<entity>>
<pre>+getTagsSemua() Operasi ini digunakan untuk mengambil data tags dari database.  +getTagsTampil(id) Operasi ini digunakan untuk mengambil data detail tags dari database berdasarkan id.  +updateDataTags(data, url) Operasi ini digunakan untuk mengupdate data tags pada database dengan parameter url dan array data.  +cekStatusTag(url) Operasi ini digunakan untuk mengecek status tag apakah aktif atau tidak dengan parameter url.  +insertGetURL(geturl) Operasi ini digunakan untuk memasukkan data taggeturl ke database dengan parameter array geturl.  +updateTagPengganti(data,url) Operasi ini digunakan untuk mengupdate data tag lama dengan pergantian status menjadi false dan id_tag_pengganti dengan id_geturl yang baru. Parameter yang digunakan adalah array data dan url.</pre>	

#### 2.2.5.24 Specific Design Class kategorimodel

<b>kategorimodel</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<pre>+getKategori() Operasi ini digunakan untuk mengambil kategori utama pada tabel kategori di database. +insertKategori(datakategori) Operasi ini digunakan untuk menambahkan kategori utama baru pada tabel kategori dengan parameter array datakategori. +checkKategori(kategori) Operasi ini digunakan untuk mengecek inputan kategori utama apakah sudah ada di database atau belum dengan parameter kategori. +getKategoriBerita() Operasi ini digunakan untuk mengambil kategori berita pada tabel berita yang tidak ada di tabel kategori berita. +getTblKategoriBerita() Operasi ini digunakan untuk mengambil kategori berita pada tabel kategori berita yang belum mempunyai relasi dengan tabel kategori. +insertKategoriBerita(data) Operasi ini digunakan untuk menambahkan data kategori berita pada tabel kategori berita dengan parameter array data. +insertHubungKategoriBerita(data) Operasi ini digunakan untuk menghubungkan atau merelasikan tabel kategori berita dan tabel kategori dengan parameter array data.</pre>	

### 2.2.5.25 Specific Design Class summarizemodel

<b>summarizemodel</b>	<b>&lt;&lt;entity&gt;&gt;</b>
<pre>+getSumberBerita() Operasi ini digunakan untuk mengambil sumber berita pada database. +getBeritaSummarize(katakunci, rangetanggal, sumberberita ) Operasi ini digunakan untuk mendapatkan data berita berdasarkan filter berita yaitu kata kunci, range tanggal, dan sumber berita. +getBeritaByID(id) Operasi ini digunakan untuk mengambil detail berita dari database berdasarkan id berita</pre>	

### 2.2.6 Class Diagram Specific Descriptions KUBERIN Mobile

#### 2.2.6.1 Spesific Design Class ListBeritaActivity

<b>ListBeritaActivity</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+ListBeritaActivity() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +AsyncTaskGetBerita() Operasi ini digunakan untuk mengambil berita melalui web service dan berjalan pada background. +SaveSharedPrefBerita() Operasi ini digunakan untuk menyimpan shared preference data berita yang didapatkan melalui AsyncTask.</pre>	

```
+recyclerViewDesign()
```

Operasi ini digunakan untuk membuat recyclerview.

#### 2.2.6.2 Specific Design Class DetailBeritaActivity

<b>DetailBeritaActivity</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+DetailBeritaActivity() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +LoadSharedPrefBerita() Operasi ini digunakan untuk mengambil data berita yang sudah disimpan pada shared preference atau memori sementara device.</pre>	

#### 2.2.6.3 Specific Design Class FragmentSummarize

<b>FragmentSummarize</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+FragmentSummarize() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +AsyncTaskGetSumberBerita() Operasi ini digunakan untuk mengambil data sumber berita melalui web service dan berjalan pada background. +AsyncTaskSummarize() Operasi ini digunakan untuk summarize berita dengan metode POST melalui web service dan berjalan pada background. Hasil nilai balikan dari operasi ini adalah data summary berita dan berita terkait.</pre>	

+AsyncTaskFilter()

Operasi ini digunakan untuk memfilter berita dengan mengirimkan data menggunakan metode POST melalui web service dan berjalan pada background. Hasil nilai balikan berupa JSON data berita dan ditampilkan dalam recyclerview.

+SaveSharedPrefListBerita ()

Operasi ini digunakan untuk menyimpan shared preference data list berita yang didapatkan melalui AsyncTask.

+SaveSharedPrefSummarize()

Operasi ini digunakan untuk menyimpan shared preference data summarize yang didapatkan melalui AsyncTask.

#### 2.2.6.4 Specific Design Class PilihBeritaActivity

PilihBeritaActivity	<<boundary>>
<p>+PilihBeritaActivity()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+LoadSharedPrefListBerita()</p> <p>Operasi ini digunakan untuk mengambil data berita yang sudah disimpan pada shared preference atau memori sementara device.</p> <p>+AsyntaskSummarizeFilter()</p> <p>Operasi ini digunakan untuk summarize berita setelah memilih data berita. Operasi ini menggunakan metode POST melalui web service dan berjalan pada background. Hasil nilai balikan dari operasi ini adalah data summary berita dan berita terkait.</p>	

+SaveSharedPrefSummarize()

Operasi ini digunakan untuk menyimpan shared preference data summarize yang didapatkan melalui AsyncTask.

#### 2.2.6.5 Specific Design Class DetailSummarizeActivity

<b>DetailSummarizeActivity</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
+DetailSummarizeActivity() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+LoadSharedPrefSummarize() Operasi ini digunakan untuk mengambil data summary dan berita terkait yang sudah disimpan pada shared preference atau memori sementara device.	

#### 2.2.6.6 Specific Design Class FragmentTrendingTopik

<b>FragmentTrendingTopik</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
+FragmentTrendingTopik() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+AsyncTaskGetTrending() Operasi ini digunakan untuk mengambil data summary melalui web service dan berjalan pada background.	
+SaveSharedPrefTrending() Operasi ini digunakan untuk menyimpan shared preference data summary yang didapatkan melalui AsyncTask.	
+recyclerviewDesign() Operasi ini digunakan untuk membuat recyclerview.	

### 2.2.6.7 Specific Design Class DetailTrendigTopikActivity

<b>DetailTrendigTopikActivity</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>+DetailTrendingTopikActivity() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +LoadSharedPrefTrending() Operasi ini digunakan untuk mengambil data summary berita yang sudah disimpan pada shared preference atau memori sementara device.</pre>	

### 2.2.6.8 Specific Design Class service\_kategori

<b>Service_kategori</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+getBeritaByKategori() Operasi ini digunakan untuk mengambil data berita dari database dan menampilkan dalam bentuk json.</pre>	

### 2.2.6.9 Specific Design Class service\_summarize

<b>Service_summarize</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+sumber() Operasi ini digunakan untuk mengambil sumber berita dari database dan menampilkan dalam bentuk json. +summarizeBerita() Operasi ini digunakan untuk melakukan summarize berita dan menampilkan hasilnya dalam bentuk json. Selain melakukan summarize, operasi tersebut juga digunakan untuk menyimpan data log dan data detail log ke dalam database.</pre>	

#### 2.2.6.10 Specific Design Class service\_summarize\_new

<b>Service_summarize</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+getDataBeritaSummarize() Operasi ini digunakan untuk mendapatkan data berita berdasarkan inputan filter yang telah diisi oleh pengguna. Data berita kemudian dikirimkan dalam bentuk json untuk ditampilkan.</pre> <pre>+summarizeBerita() Operasi ini digunakan untuk melakukan summarize berita dan menampilkan hasilnya dalam bentuk json. Selain melakukan summarize, operasi tersebut juga digunakan untuk menyimpan data log dan data detail log ke dalam database.</pre>	

#### 2.2.6.11 Specific Design Class service\_trending\_topik

<b>Service_trending_topik</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+trending() Operasi ini digunakan untuk mengambil data trending topik atau hasil summary berupa keyword dan idLog dari database dan dikirim dalam json untuk ditampilkan.</pre> <pre>+getDetailTrending () Operasi ini digunakan untuk menampilkan detail trending topik dari database berdasarkan id log.</pre>	



### 2.2.6.12 Specific Design Class `berita_model`

<code>Berita_model</code>	<code>&lt;&lt;entity&gt;&gt;</code>
<pre>+kategoriberita(kategori) Operasi ini digunakan untuk mendapatkan/mengambil kategori berita dari parameter kategori utama. +getBeritaKategori(kategori) Operasi ini digunakan untuk mengambil data berita pada database berdasarkan kategori berita. +getJumlahBeritaKategori(kategori) Operasi ini digunakan untuk mengambil jumlah data berita pada database berdasarkan kategori berita. +fetchBeritaKategori(limit, start, kategori) Operasi ini digunakan untuk mengambil data berita dari database dengan batas pengambilan data dan awal data diambil berdasarkan kategori berita.</pre>	

### 2.2.6.13 Specific Design Class `summarize_model`

<code>summarize_model</code>	<code>&lt;&lt;entity&gt;&gt;</code>
<pre>+getSumberBerita() Operasi ini digunakan untuk mengambil sumber berita pada database. +getBeritaSummarize(katakunci, tanggalrange, sumberberita) Operasi ini digunakan untuk mendapatkan data berita berdasarkan filter berita yaitu kata kunci, range tanggal, dan sumber berita. +getBeritaByID(id) Operasi ini digunakan untuk mengambil detail berita</pre>	

dari database berdasarkan id berita.

```
+insertLog(log)
```

Operasi ini digunakan untuk memasukan data Log Summary ke dalam database dengan parameter array log.

```
+insertDetailLog(detaillog)
```

Operasi ini digunakan untuk memasukan data detail Log Summary ke dalam database dengan parameter array detaillog.

```
+getTrendingTopik()
```

Operasi ini digunakan untuk mengambil data trending topik di dalam database.

```
+getJumlahDataByKeyword()
```

Operasi ini digunakan untuk mengambil data trending topik berdasarkan jumlah data dalam satu keyword.

```
+getDataLogTerbaru()
```

Operasi ini digunakan untuk mengambil data trending topik berdasarkan tanggal dan waktu terbaru dalam satu keyword.

```
+GetDataLogByID(id)
```

Operasi ini digunakan untuk mengambil data log dari database berdasarkan id log.

```
+GetDataDetailLogByIDLog(idLog)
```

Operasi ini digunakan untuk mengambil data detail log dari database berdasarkan id log.

### 3 PERANCANGAN DATA

#### 3.1 Dekomposisi Data

3.1.1 Deskripsi Entitas Tbl_Berita			
Nama	Tipe	Panjang	Keterangan
id_user	Integer	-	id_user, Primary Key
Nama	Variable Character	70	Nama dari pengguna/user
Username	Variable Character	50	Username dari pengguna/user
Password	Variable Character	50	Password dari pengguna/user

3.1.2 Deskripsi Entitas Tbl_Berita			
Nama	Tipe	Panjang	Keterangan
id_berita	Integer	-	id_berita, Primary Key
Judul	Text	-	Judul dari berita
Tanggal	Date	-	Tanggal dari berita
Waktu	Time	-	Waktu dari berita
kategori	Variable Character	45	Kategori dari berita
gambar	Text	-	Gambar dari berita
isi_berita	Text	-	Berita
link	Text	-	Link dari berita
sumberberita	Variable Character	255	Sumber dari berita/nama website berita
timer	Float	-	Waktu untuk mendapatkan berita/lamanya scraping
datetime_scraping	DateTime	-	Tanggal dan waktu proses scraping

### 3.1.3 Deskripsi Entitas Tbl\_GetURL

Nama	Tipe	Panjang	Keterangan
id_geturl	Integer	-	id_geturl, Primary Key
url	Text	-	Url dari situs berita yang akan di ambil datanya/di scrap
taglink1	Text	-	Tag terluar untuk mengambil link berita
taglink2	Text	-	Tag didalam taglink1 untuk mengambil link berita
taglink3	Text	-	Tag didalam taglink2 untuk mengambil link berita
judulberita1	Text	-	Tag judulberita yang paling luar untuk mengambil judul berita
judulberita2	Text	-	Tag di dalam tag judulberita1 untuk mengambil judul berita
kategoriberita1	Text	-	Tag kategoriberita yang paling luar untuk mengambil kategori berita
kategoriberita2	Text	-	Tag di dalam tag kategoriberita1 untuk mengambil kategori berita
tanggalberita1	Text	-	Tag tanggalberita yang paling luar untuk

			mengambil tanggal dan waktu berita
tanggalberita2	Text	-	Tag di dalam tag tanggalberita1 untuk mengambil tanggal dan waktu berita
gambarberita1	Text	-	Tag gambarberita yang paling luar untuk mengambil gambar berita
gambarberita2	Text	-	Tag di dalam tag gambarberita1 untuk mengambil gambar berita
isiberita1	Text	-	Tag isiberita yang paling luar untuk mengambil berita
isiberita2	Text	-	Tag di dalam tag isiberita1 untuk mengambil berita
buangkotonten	Text	-	Tag untuk membuang konten yang tidak perlu
status_tag	Variable Character	45	Status tag berita, yaitu aktif dan tidak
sumber_berita	Variable Character	255	Sumber dari berita/nama website berita
tgl_tdkaktif	Datetime	-	Tanggal saat tag tidak aktif

3.1.4 Deskripsi Entitas Tbl_GetBerita			
Nama	Tipe	Penjang	Keterangan
id_getberita	Integer	-	id_getberita, Primary Key
id_geturl	Integer	-	id_geturk, foreign key dari tabel getURL
url_berita	Text	-	url untuk mengambil data berita
Status	Variable Character	45	Status berita apakah sudah diambil atau belum, yaitu sudah, belum, dan menunggu.

3.1.5 Deskripsi Entitas Tbl_KategoriBerita			
Nama	Tipe	Panjang	Keterangan
id_kategoriberita	Integer	-	id_kategoriberita, Primary Key
kategori_berita	Variable Character	45	Kategori dari Berita

3.1.6 Deskripsi Entitas Tbl_KategoriUtama			
Nama	Tipe	Panjang	Keterangan
id_kategoriutama	Integer	-	id_kategoriutama, primary key
kategori_utama	Variable Character	20	Kategori Utama dari berita

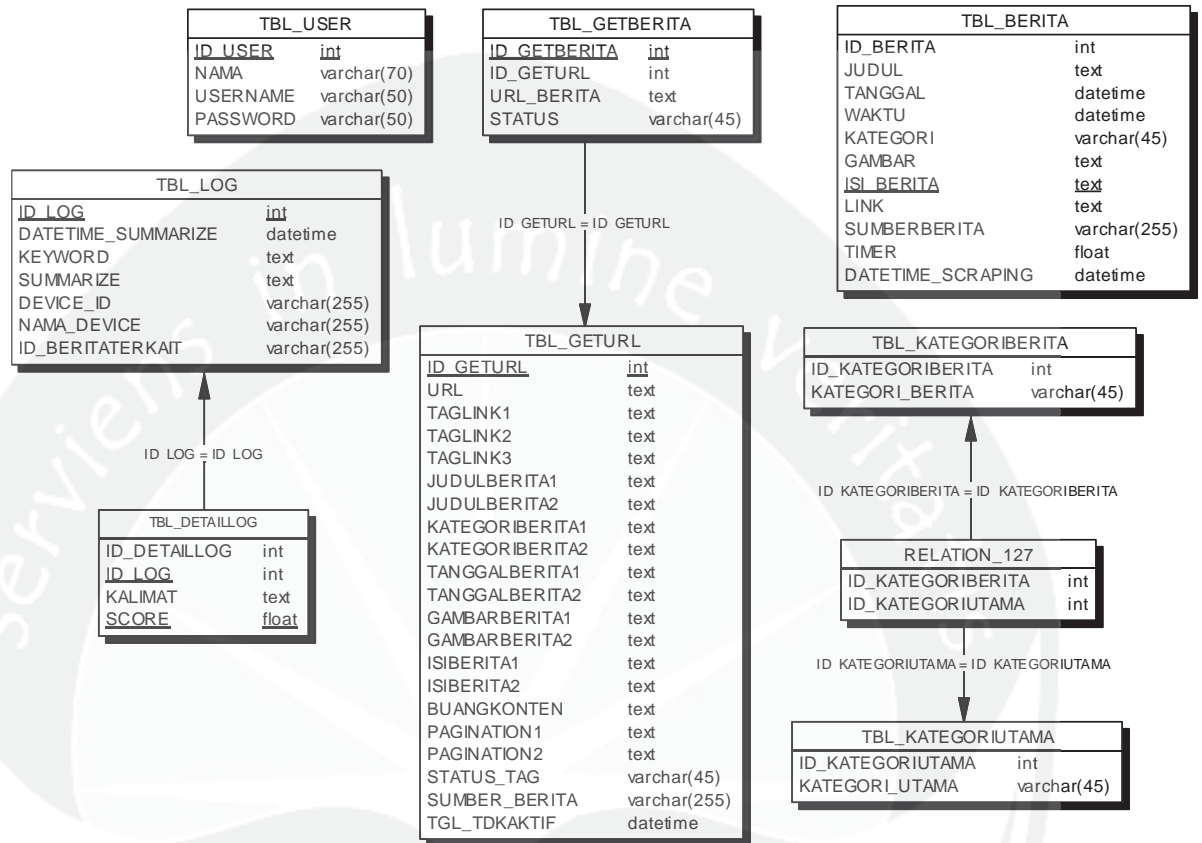
3.1.7 Deskripsi Entitas Tbl_Relasi_KatUtama_KatBerita			
Nama	Tipe	Panjang	Keterangan
id_kategoriutama	Integer	-	id_kategoriutama, Primary Key, Foreign key dari tabel kategori utama
id_kategoriiberita	Integer	-	id_kategoriiberita, Primary Key, Foreign key dari tabel kategori berita

3.1.8 Deskripsi Entitas Tbl_Log			
Nama	Tipe	Panjang	Keterangan
id_log	Integer	-	id_log, Primary Key
datetime_summarize	DateTime	-	Tanggal dan waktu proses summarize
keyword	Text	-	Kata kunci dalam proses summarize
summarize	Text	-	Hasil summarize
device_id	Variable Character	255	ID device android pengguna aplikasi yang melakukan proses summarize
nama_device	Variable Character	255	Nama Device android pengguna aplikasi yang melakukan proses summarize
id_beritaterkait	Variable Character	255	Id berita terkait dengan hasil summary

3.1.9 Deskripsi Entitas Tbl_DetailLog			
Nama	Tipe	Panjang	Keterangan
id_detaillog	Integer	-	id detaillog, Primary Key
id_log	Integer	-	id_log, foreign key dari Tbl_Log
kalimat	Text	-	Kalimat dari hasil summary
score	Float	-	Skor dari kalimat hasil summary



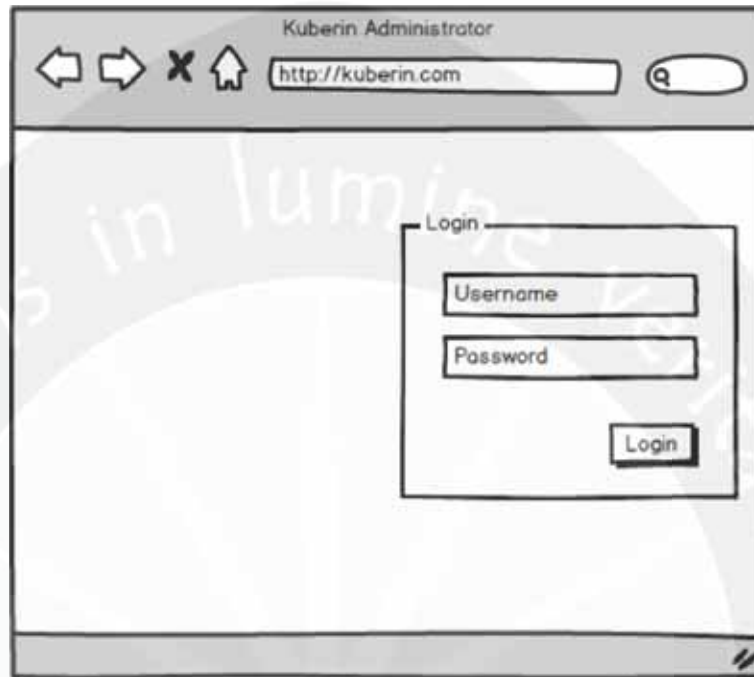
### 3.2 Physical Data Model



Gambar 3.1 Physical Data Model

## 4 Deskripsi Perancangan Antarmuka

### 4.1 Antarmuka Halaman Login Web



Gambar 4.1 Antarmuka Halaman Login Web

Antarmuka ini digunakan untuk melakukan proses login ke dalam sistem aplikasi web. Untuk mendapatkan akses masuk ke dalam sistem, user harus memasukkan username dan password terlebih dahulu pada textbox yang telah disediakan. Pada saat tombol login ditekan, sistem akan mengecek username dan password yang diinputkan dengan data username dan password yang telah tersimpan di database. Jika data username dan password benar atau cocok maka user akan masuk ke dalam sistem, sebaliknya jika username dan password salah atau tidak cocok maka akan muncul pesan peringatan.

Apabila tombol login di tekan maka proses akan dilanjutkan ke antarmuka Scraping Berita.

## 4.2 Antarmuka Halaman Scraping

The screenshot shows a web browser window titled "Kuberin Administrator" with the address "http://kuberin.com". The main content area is divided into three sections:

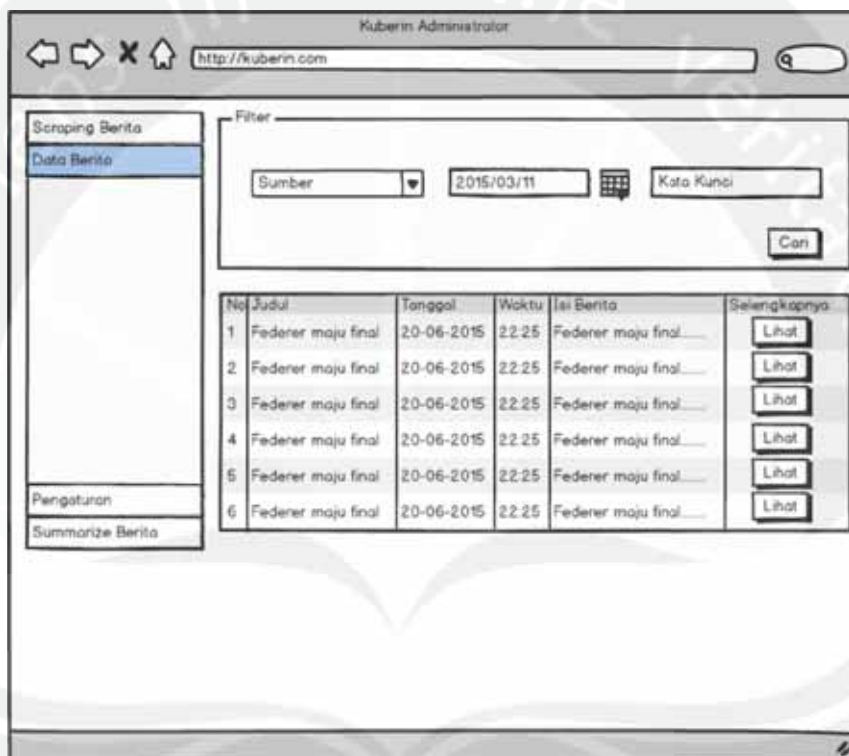
- Input:** A vertical stack of text input fields for defining scraping rules:
  - Sumber Berita
  - Uri Berita
  - Tag Link 1
  - Tag Link 2
  - Tag Link 3
  - Tag Judul 1
  - Tag Judul 2
  - Tag Kategori 1
  - Tag Kategori 2
  - Tag Tanggal 1
  - Tag Tanggal 2
  - Tag Gambar 1
  - Tag Gambar 2
  - Tag Isi 1
  - Tag Isi 2
  - Konten tidak perlu (checkbox)
- Output:** A large empty rectangular box on the right side.
- Buttons:** "Test" and "Simpon" buttons are located at the bottom of the input section.
- Sidebar:** On the left, there is a "Scraping Berita" header and three buttons: "Data Berita", "Pengaturan", and "Summarize Berita".

Gambar 4.2 Antarmuka Halaman Scraping

Antarmuka ini digunakan untuk melakukan tes pengambilan data berita dan untuk menyimpan tags dan rules yang diinputkan. Administrator dapat melakukan tes pengambilan data berita atau scraping dengan memasukkan rules dan tags pada textbox yang tersedia. Klik button test untuk melakukan tes scraping. Hasil dari tes scraping akan tampil pada panel output atau sebelah kanan form. Jika data berhasil didapatkan maka akan tertampil di bagian output, sebaliknya jika tidak maka akan tampil pesan "Data Kosong" di bagian output. Data yang didapatkan meliputi : link, judul, kategori, tanggal, waktu, gambar, dan isi berita. Setelah data dapat berhasil didapatkan, admin bisa melakukan

penyimpanan data tags dan rule dengan memilih tombol simpan. Jika ada salah satu data muncul pesan "Data Kosong" dianjurkan untuk tidak menyimpan data tags dan rules. Data yang telah tersimpan nantinya akan diproses oleh cron tab untuk proses scraping berita.

#### 4.3 Antarmuka Halaman Lihat Data Berita



Gambar 4.3 Antarmuka Halaman Lihat Data Berita

Antarmuka ini digunakan untuk melihat data berita dan filter berita. Administrator dapat melihat data berita berupa tabel dan pagination. Untuk melihat detail berita, admin dapat menekan tombol lihat sesuai dengan berita yang dipilih. Terdapat juga filter berita yang berfungsi untuk memudahkan pencarian data berita. Terdapat form untuk menginputkan data yaitu kata kunci, range tanggal dan sumber berita. Data yang di cari akan

tampil pada tabel dibawahnya. Admin dapat mengisi pada textbox yang tersedia untuk mencari berita. Textbox harus diisi minimal salah satu. Pada saat tombol cari ditekan, sistem akan menampilkan data berita sesuai filter yang diinputkan oleh admin. Jika textbox tidak ada yang diisi maka akan muncul pesan "Data Kosong", sedangkan jika textbox diisi minimal salah satu textbox maka sistem akan memproses data tersebut.

#### 4.4 Antarmuka Halaman Update Data Tags

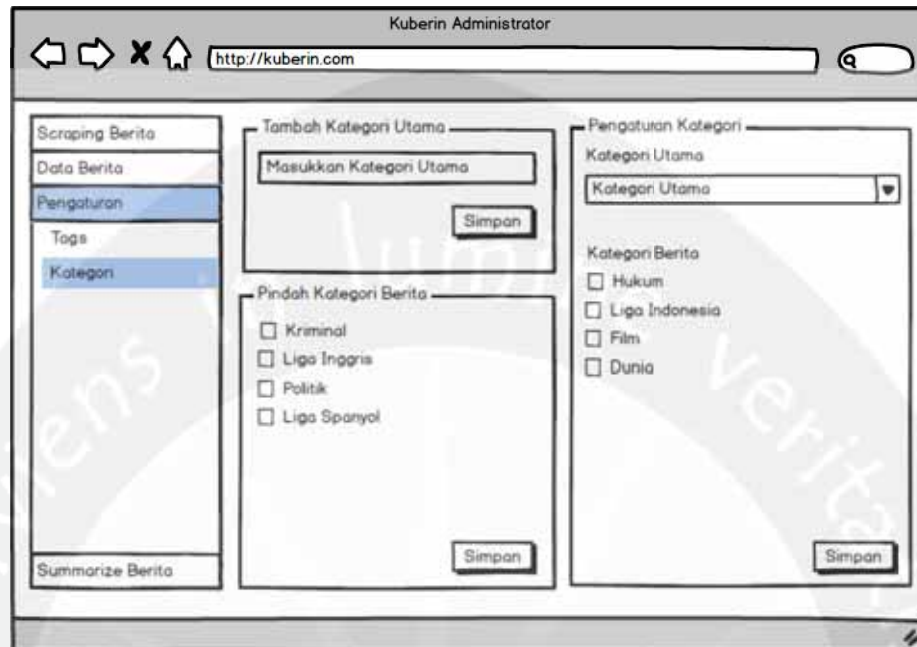


Gambar 4.4 Antarmuka Halaman Update Data Tags

Antarmuka ini digunakan untuk melihat status tags dan untuk mengupdate data tags. Administrator yang bisa melakukan hal tersebut. Pada aplikasi tertampil tabel status tags apakah tags aktif untuk mengambil berita

atau tidak aktif. Admin juga dapat mengupdate data tags dengan menekan tombol edit. Data akan langsung tertampil dalam textbox dan admin dapat mengganti tags yang ingin diupdate. Sama pada halaman antarmuka scraping, terdapat dua button yaitu test dan update. Hanya button update yang membedakan. Button test berfungsi untuk mengetes tags dan rules untuk mengambil data berita. Button update berfungsi untuk mengupdate tags dan rules yang sudah di tes. Ketika button update ditekan maka data tags dan rules yang ada di database akan berubah dan status tag menjadi aktif. Jika ada salah satu data muncul pesan "Data Kosong" dianjurkan untuk tidak mengupdate data tags dan rules.

#### 4.5 Antarmuka Halaman Pengelolaan Kategori



Gambar 4.5 Antarmuka Halaman Pengelolaan Kategori

Antarmuka ini digunakan untuk mengelola kategori. Admin dapat melakukan penambahan kategori utama, pindah kategori berita yang didapatkan ke tabel kategori, dan menghubungkan kategori utama dan kategori berita. Pada tambah kategori utama admin bisa menambahkan kategori utama pada textbox yang telah disediakan. Tekan button simpan untuk menyimpan data kategori utama. Jika data sudah ada di database maka akan muncul pesan data sudah ada di database, sebaliknya jika data belum ada di database maka sistem akan menyimpan data dan muncul pesan data sukses disimpan. Textbox tidak boleh kosong, jika kosong maka akan muncul pesan "Kategori tidak boleh kosong".

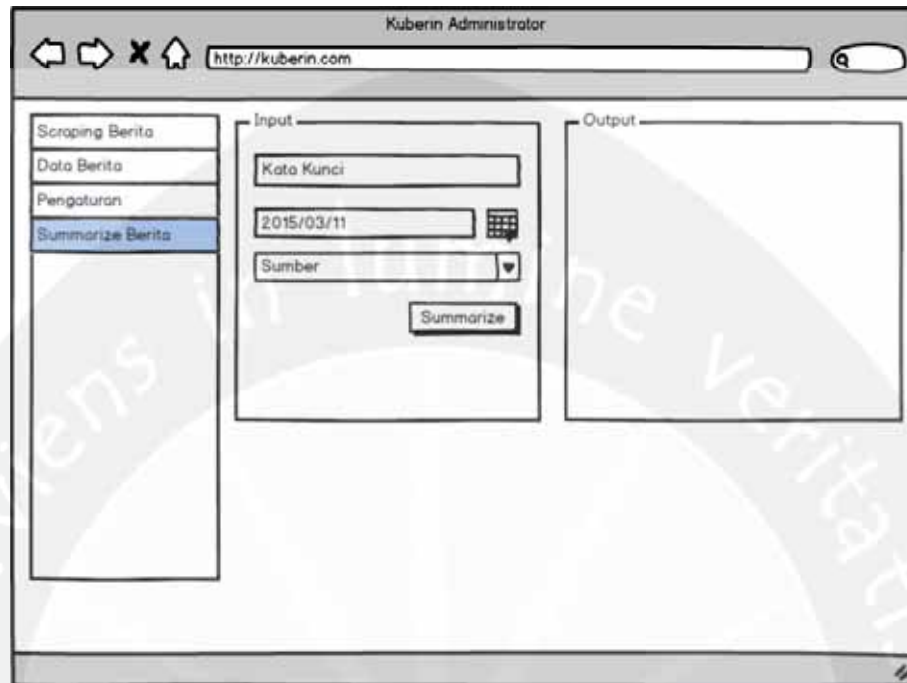
Pada pindah kategori berita terdapat daftar kategori yang di dapatkan dari proses scraping. Kategori tersebut nantinya dihubungkan dengan kategori

utama. Untuk melakukan pemindahan kategori berita hasil scraping ke tabel baru maka admin harus memilih kategori yang akan dipindah berbentuk checkbox dan menekan tombol simpan. Jika admin belum memilih kategori dan langsung menekan button simpan maka akan muncul pesan "Kategori harus dipilih", sebaliknya jika kategori ada yang dipilih maka sistem akan menyimpan data pada database dan muncul pesan sukses.

Hubung kategori utama dan kategori berita ini berfungsi untuk menghubungkan kategori utama dan kategori berita yang didapat dari proses scraping. Admin memilih kategori utama dalam bentuk dropdown list dan memilih kategori berita dalam bentuk checkbox kemudian klik button simpan. Jika salah satu data tidak diisi atau bahkan semuanya tidak diisi maka akan menampilkan pesan harus diisi semua, sebaliknya jika diisi semua maka sistem akan menyimpan data dan menampilkan pesan sukses.



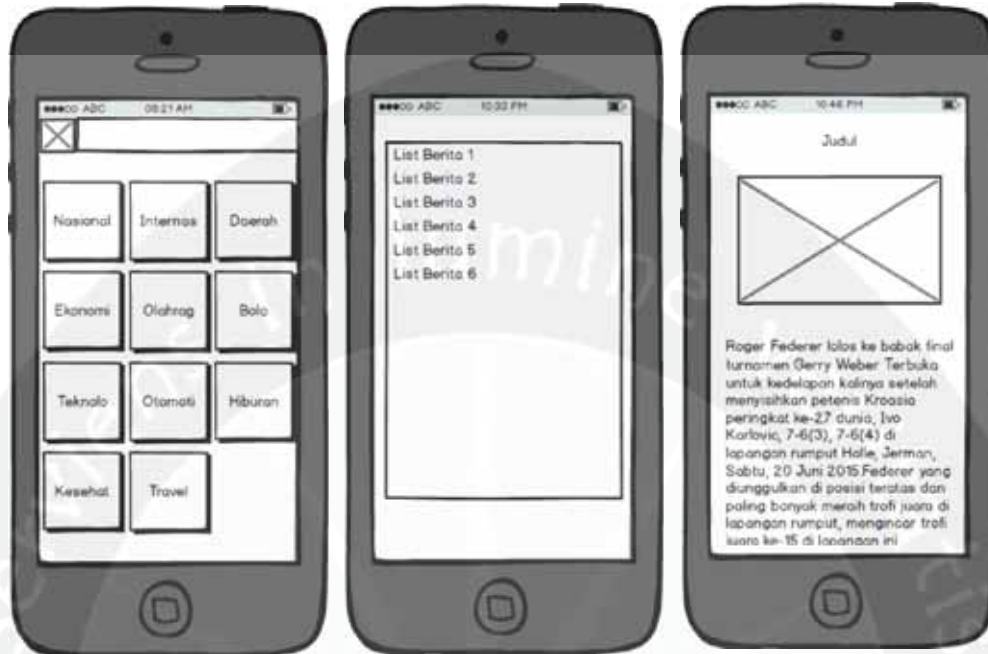
#### 4.6 Antarmuka Halaman Summarize Berita Web



**Gambar 4.6 Antarmuka Halaman Summarize Berita Web**

Antarmuka ini digunakan untuk melakukan summarize berita serta menampilkan hasil summary berita, berita terkait dan skor kalimat. Terdapat textbox untuk memfilter data berita yang akan di ringkas. Textbox tersebut meliputi inputan kata kunci, range tanggal, dan sumber berita. Ketika di klik summarize maka akan muncul hasil ringkasan, berita terkait, dan skor kalimat pada sebelah kanan form. Kata kunci harus diisi tidak boleh kosong, jika tidak maka akan menampilkan pesan "Kata kunci harus diisi".

#### 4.7 Antarmuka Halaman Kategori Berita



**Gambar 4.7 Antarmuka Halaman Kategori Berita**

Antarmuka ini digunakan untuk menampilkan data berita berdasarkan kategori utama. Terdapat pilihan kategori berupa gridview yang dapat dipilih pengguna untuk melihat daftar berita berdasarkan kategori. Setelah salah satu kategori dipilih maka akan muncul list berita berdasarkan kategori. Jika salah satu list berita dipilih maka akan tampil detail berita.

#### 4.8 Antarmuka Halaman Summarize Berita Mobile



**Gambar 4.8 Antarmuka Halaman Summarize Berita Mobile**

Antarmuka ini digunakan untuk melakukan summarize berita. Pengguna dapat memasukkan data berita yang ingin dicari berdasarkan kata kunci, range tanggal, dan sumber. Ketika tombol summarize di tekan maka akan ke halaman selanjutnya untuk menampilkan hasil summary dan berita terkait. Kata kunci harus diisi, jika tidak maka akan muncul pesan "kata kunci harus diisi".

#### 4.9 Antarmuka Halaman Trending Topik



Gambar 4.9 Antarmuka Halaman Trending Topik

# SKPL

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

KUBERIN

(Kumpulan Berita Indonesia)

Untuk :


Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Yustinus Andry Setiawan / 6659

Program Studi Teknik Informatika - Fakultas Teknologi  
Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika  Fakultas Teknologi	Nomor Dokumen		Halaman
		<i>SKPL-KUBERIN</i>		1/36
		<i>Revisi</i>	A	

**DAFTAR PERUBAHAN**

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiksa oleh								
Disetujui oleh								

### Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi



## DAFTAR ISI

<b>1</b>	<b>Pendahuluan</b>	<b>6</b>
1.1	Tujuan	6
1.2	Lingkup Masalah	6
1.3	Definisi, Akronim dan Singkatan	7
1.4	Referensi	8
1.5	Deskripsi umum (Overview)	8
<b>2</b>	<b>Deskripsi Kebutuhan</b>	<b>9</b>
2.1	Perspektif produk	9
2.2	Fungsi Produk	11
2.3	Karakteristik Pengguna	15
2.4	Batasan-batasan	16
2.5	Asumsi dan Ketergantungan	16
<b>3</b>	<b>Kebutuhan khusus</b>	<b>16</b>
3.1	Kebutuhan antarmuka eksternal	16
3.1.1	Antarmuka pemakai	16
3.1.2	Antarmuka perangkat keras	17
3.1.3	Antarmuka perangkat lunak	17
3.1.4	Antarmuka Komunikasi	18
3.2	Kebutuhan fungsionalitas Perangkat Lunak	19
3.2.1	Use Case Diagram	19
<b>4</b>	<b>Spesifikasi Rinci Kebutuhan</b>	<b>20</b>
4.1	Spesifikasi Kebutuhan Fungsionalitas	20
4.1.1	Use case Spesification : <i>Login</i>	20
4.1.2	Use case Spesification : <i>Logout</i>	21
4.1.3	Use case Spesification : Mengelola Data <i>Scraping</i> Berita	22
4.1.4	Use case Spesification : Melihat Data Berita	24
4.1.5	Use case Spesification : Mengelola Data Kategori	25
4.1.6	Use case Spesification : Melihat <i>Summary</i> Berita Web	28
4.1.7	Use case Spesification : Menampilkan <i>Trending</i> topik	29
4.1.8	Use case Spesification : Melihat <i>Summary</i> Berita Mobile	30
4.1.9	Use case Spesification : Melihat Data Berita Berdasarkan Kategori	31
4.1.10	Use case Spesification : Mengambil <i>URL</i> Berita	32
4.1.11	Use case Spesification : Mengambil Data Berita	34
4.1.12	Use case Spesification : Mengecek <i>Tags</i> Tidak Aktif	35
<b>5</b>	<b>ERD</b>	<b>36</b>

Program Studi Teknik Informatika	SKPL –KUBERIN	4/ 36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		



## Daftar Gambar

Gambar 2.1	Arsitektur Perangkat Lunak KUBERIN .....	10
Gambar 3.1	Use Case Diagram .....	19
Gambar 5.1	ERD.....	36



# 1 Pendahuluan

## 1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak KUBERIN (Kumpulan Berita Indonesia) untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain perangkat lunak dan perangkat keras, dan pengguna) dan atribut (*feature-feature* tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-KUBERIN ini juga mendefinisikan batasan perancangan perangkat lunak.

## 1.2 Lingkup Masalah

Perangkat Lunak KUBERIN dikembangkan dengan tujuan untuk :

1. Melakukan proses web *Scraping* untuk mengambil data berita dari situs berita.
2. Mengelola data berita yang telah diambil dengan teknik web *Scraping*.
3. Mengangani pencarian data berita.
4. Mengkategorikan berita.
5. Melakukan *summary* berita.
6. Memudahkan user dalam membaca berita.

### 1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-KUBERIN-XXX	Kode yang merepresentasikan kebutuhan pada KUBERIN (Kumpulan Berita Indonesia) di mana XXX merupakan nomor fungsi produk.
KUBERIN	Perangkat lunak mobile dan web yang dibangun.
Internet	Internet merupakan istilah umum yang dipakai untuk menunjuk <i>Network</i> global yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World Wide Web.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
Web Scraping	Mendapatkan informasi dalam sebuah website dengan cara mengekstrak seluruh isi data dari sebuah page website.
Website	Sekumpulan halaman informasi yang disediakan melalui jalur internet sehingga bisa diakses di seluruh dunia selama terkoneksi dengan jaringan internet.

Situs Berita	Website yang menampilkan berita
Berita	Laporan dan informasi mengenai sesuatu peristiwa atau kejadian yang sedang terjadi
Summary Berita	Merangkum beberapa berita atau artikel menjadi satu.

#### 1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Yustinus Andry Setiawan / 6659, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) MEXTEM*, Universitas Atma Jaya Yogyakarta.

#### 1.5 Deskripsi umum (Overview)

Secara umum dokumen SKPL ini terbagi atas 3 bagian utama. Bagian utama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak tersebut, definisi, referensi dan deskripsi umum tentang dokumen SKPL ini.

Bagian kedua berisi penjelasan umum tentang perangkat lunak KUBERIN yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak KUBERIN tersebut.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak KUBERIN yang akan dikembangkan.

## 2 Deskripsi Kebutuhan

### 2.1 Perspektif produk

KUBERIN merupakan perangkat lunak yang dikembangkan untuk memudahkan pengguna dalam membaca berita. Terdapat dua sisi user yaitu admin serta pengguna umum. Admin dapat mengatur *URL*, *rule*, pengkategorian berita pada aplikasi web. Pengguna dapat menggunakan aplikasi *mobile* untuk membaca berita dan melakukan *summarize* berita. Sistem ini berjalan pada *mobile* dan web.

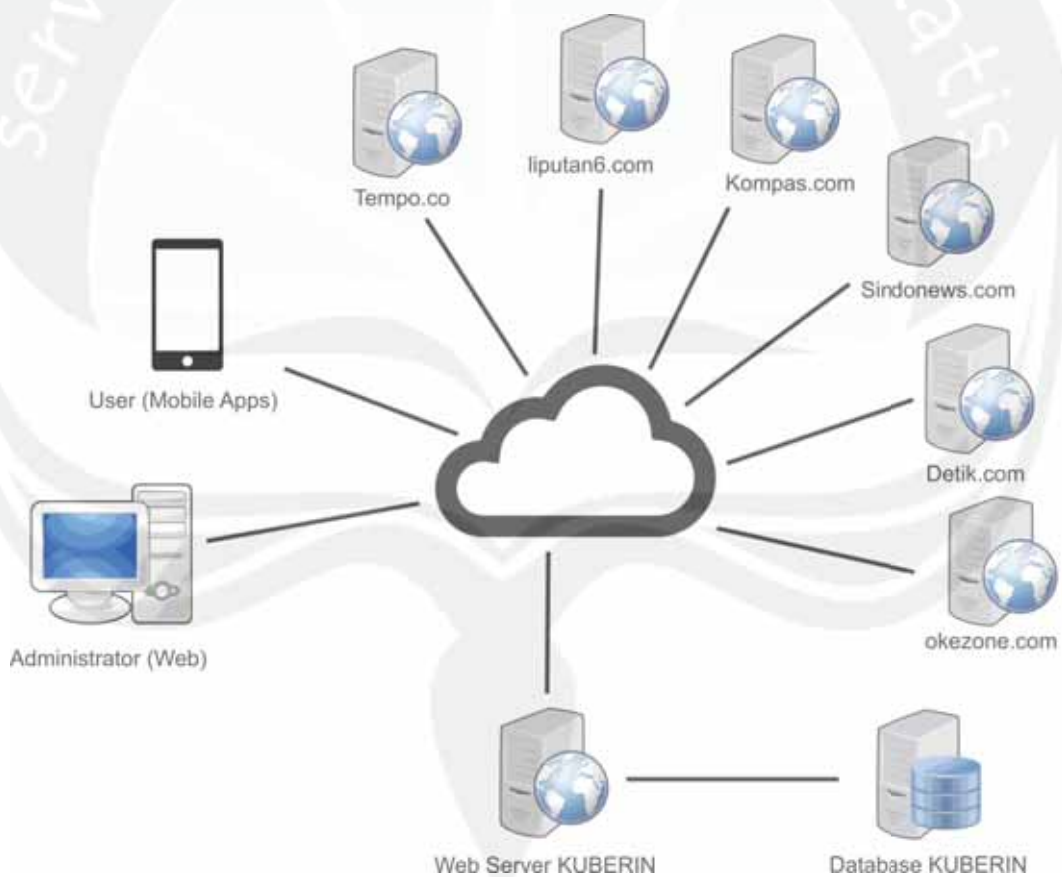
Perangkat lunak desktop KUBERIN ini berjalan pada platform *mobile* Android, untuk web bisa berjalan pada web browser apapun, dan dibuat menggunakan bahasa pemrograman Java dan PHP. Sedangkan untuk lingkungan pemrogramannya menggunakan PHP Storm dan Android Studio. Sedang untuk *database*, akan dipergunakan My SQL, dan kedua aplikasi, desktop dan web, akan mengacu kepada database yang sama.

Pengguna akan berinteraksi dengan aplikasi melalui antarmuka GUI (*Graphical User Interface*) pada aplikasi web maupun *mobile*. Pada aplikasi terlihat pada gambar 1, arsitektur perangkat lunak yang digunakan adalah *client-server*, dimana data disimpan di *server*. Pengguna dapat melakukan pengaksesan data secara *on-line*. Aplikasi *mobile* dapat mengakses data secara *on-line* dengan memanggil *web service* yang terdapat di *web server*. *Web server* mengambil data berita dari situs

Program Studi Teknik Informatika	SKPL –KUBERIN	9/ 36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

berita seperti kompas.com, tempo.co, liputan6.com, okezone.com, detik.com dan sindonews.com dengan teknik web *Scraping* kemudian disimpan dalam *database server*. Data berita tersebut dapat diakses melalui aplikasi web maupun *mobile*.

Inputan data yang dilakukan oleh admin pada aplikasi web akan disimpan dalam *database server*, sehingga jika ada pencarian data, maka data yang diinginkan akan dicari ke *database server* yang selanjutnya dikirimkan ke client yang meminta melalui web server.



**Gambar 2.1** Arsitektur Perangkat lunak KUBERIN

## 2.2 Fungsi Produk

Fungsi produk perangkat lunak KUBERIN adalah sebagai berikut :

### APLIKASI WEB

#### 1. Fungsi *Login* (SKPL-KUBERIN-001).

Merupakan fungsi yang digunakan oleh administrator untuk dapat masuk dalam sistem yang akan digunakan.

#### 2. Fungsi *Logout* (SKPL-KUBERIN-002).

Merupakan fungsi yang digunakan oleh administrator untuk keluar dari sistem atau aplikasi.

#### 3. Fungsi Pengelolaan Data *Scraping* Berita (SKPL-KUBERIN-003).

Merupakan fungsi yang digunakan oleh administrator untuk melakukan pengelolaan data *Scraping* berita yaitu test data *Scraping*, simpan data *Scraping*, tampil data *Scraping* dan edit data *Scraping*.

Fungsi Pengelolaan Data *Scraping* Berita mencakup:

##### a. Fungsi *Test Data Scraping* (SKPL-KUBERIN-003-01).

Merupakan fungsi yang digunakan oleh administrator untuk melakukan *test tag* yang diinputkan, apakah tag tersebut dapat mengambil data atau tidak.

b. Fungsi *Simpan Data Scraping*(**SKPL-KUBERIN-003-02**).

Merupakan fungsi yang digunakan oleh administrator untuk menyimpan data *Scraping* atau *Tags* untuk proses *Scraping*. Data tersebut digunakan untuk mengambil data atau proses web *Scraping* situs berita. Secara otomatis server akan melakukan *Scraping* berita dari situs - situs berita yang diinputkan administrator.

c. Fungsi *Tampil Data Scraping*(**SKPL-KUBERIN-003-03**).

Merupakan fungsi yang digunakan untuk menampilkan data *Tags Scraping* berita. Akan ditampilkan status dari *Tags Scraping* berita yaitu status aktif dan tidak aktif. Data *Tags* tidak aktif ini diperoleh dari pengecekan data *Tags* di database oleh sistem.

d. Fungsi *Update Data Scraping*(**SKPL-KUBERIN-003-04**).

Merupakan fungsi yang digunakan oleh administrator untuk mengupdate data *Tags Scraping*. Data tersebut digunakan untuk mengambil data atau proses web *Scraping* situs berita.

4. Fungsi *Lihat Data Berita*(**SKPL-KUBERIN-004**).

Merupakan fungsi yang digunakan oleh administrator untuk melihat data berita dan filter data berita. Fungsi dari filter data berita adalah untuk pencarian atau filter data



berita berdasarkan inputan sumber berita, tanggal dan kata kunci. Fungsi tampil data berita adalah untuk melihat data berita hasil dari proses *Scraping*. Data berita akan ditampilkan dalam bentuk tabel dan ada button lihat untuk melihat data detail berita.

5. Fungsi *Pengelolaan Kategori* (**SKPL-KUBERIN-005**).

Merupakan fungsi yang digunakan administrator untuk mengatur kategori berita.

Fungsi *Pengelolaan Kategori* mencakup :

a. Fungsi *Tambah Kategori Utama* (**SKPL-KUBERIN-005-01**).

Merupakan fungsi yang digunakan untuk menambah kategori utama. Kategori utama ini digunakan untuk pengelompokan kategori berita.

b. Fungsi *Pindah Kategori Berita* (**SKPL-KUBERIN-005-02**).

Merupakan fungsi yang digunakan untuk mendapatkan kategori pada tabel berita kemudian data kategori dipindahkan ke tabel kategori berita.

c. Fungsi *Hubung Kategori Utama dan Kategori Berita* (**SKPL-KUBERIN-005-03**).

Merupakan fungsi yang digunakan untuk menghubungkan kategori utama dan kategori berita. Proses menghubungkan kategori berita dan kategori utama ini adalah untuk mengelompokkan kategori berita berdasarkan kategori utama.

6. Fungsi *Lihat Summary Berita Web*(**SKPL-KUBERIN-006**).

Merupakan fungsi yang digunakan oleh administrator untuk melakukan *summarize* berita atau peringkasan berita yaitu melihat ringkasan berita berdasarkan inputan kata kunci, *range* tanggal, dan sumber. Kata kunci wajib diisi sedangkan *range* tanggal dan sumber tidak wajib. Hasil yang didapatkan adalah *summary* berita, skor kalimat, dan berita terkait dengan hasil *summary*.

**APLIKASI MOBILE**

7. Fungsi *Tampil Trending Topik* (**SKPL-KUBERIN-007**)

Merupakan fungsi yang digunakan oleh pengguna umum untuk melihat *Trending* topik.

8. Fungsi *Lihat Summary Berita Mobile*(**SKPL-KUBERIN-008**)

Merupakan fungsi yang digunakan oleh pengguna umum untuk *summarize* berita atau peringkasan berita yaitu melihat ringkasan berita berdasarkan inputan kata kunci, *range* tanggal, dan sumber. Kata kunci wajib diisi sedangkan *range* tanggal dan sumber tidak wajib. Hasil yang didapatkan adalah *summary* berita dan berita terkait dengan hasil *summary*.

9. Fungsi *Tampil Data Berita* (**SKPL-KUBERIN-009**).

Merupakan fungsi yang digunakan oleh pengguna untuk melihat data berita sesuai dengan kategori utama.

Program Studi Teknik Informatika	SKPL –KUBERIN	14/ 36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## APLIKASI WEB SCRAPER

### 10. Fungsi *Get URL Berita* (SKPL-KUBERIN-010)

Merupakan fungsi yang digunakan untuk mengambil URL berita pada halaman index berita. Proses pengambilan URL pada index berita dilakukan selama kurang lebih 1-1,5 jam sekali.

### 11. Fungsi *Get Data Berita* (SKPL-KUBERIN-011)

Merupakan fungsi yang digunakan untuk mengambil data berita berdasarkan URL yang telah tersimpan di database. Proses pengambilan berita dilakukan setiap 5 menit sekali dan mengambil 10 URL berita dari database.

### 12. Fungsi *Cek Tags Tidak Aktif* (SKPL-KUBERIN-012)

Merupakan fungsi yang digunakan untuk mengecek apakah tags yang tersimpan dalam database dapat mengambil berita atau tidak. Fungsi ini juga akan mengeset status tags, jika aktif maka statusnya 'aktif' dan jika tidak maka akan terset 'tidak'. Proses cek tags tidak aktif ini akan berjalan setiap jam 13.00-13.30.

## 2.3 Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak KUBERIN adalah sebagai berikut :

1. Memahami pengoperasian Microsoft Windows dan perangkat *mobile* Android.
2. Mengerti tentang internet dan web.
3. Memahami penggunaan aplikasi Kumpulan Berita Indonesia (KUBERIN).

Program Studi Teknik Informatika	SKPL-KUBERIN	15/36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## **2.4 Batasan-batasan**

Batasan-batasan dalam pengembangan perangkat lunak KUBERIN tersebut adalah :

### 1. Kebijaksanaan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak KUBERIN.

### 2. Keterbatasan perangkat keras

Dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

## **2.5 Asumsi dan Ketergantungan**

Sistem ini dapat dijalankan pada perangkat *mobile Smart Phone* yang menggunakan sistem operasi android versi 4.0 keatas, serta aplikasi web broser apapun.

## **3 Kebutuhan khusus**

### **3.1 Kebutuhan antarmuka eksternal**

Kebutuhan antar muka eksternal pada perangkat lunak KUBERIN meliputi kebutuhan antarmuka pemakai, antarmuka perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi.

#### **3.1.1 Antarmuka pemakai**

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam bentuk form-form.

### 3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan dalam perangkat lunak KUBERIN adalah:

1. Perangkat *mobile* Android .
2. Personal Komputer
3. Laptop

### 3.1.3 Antarmuka perangkat lunak

Perangkat lunak yang dibutuhkan untuk mengoperasikan perangkat lunak KUBERIN adalah sebagai berikut :

1. Nama : MySQL

Sumber : MySQL

Sebagai database management *system* (DBMS) yang digunakan untuk menyimpan data di sisi server.

2. Nama : Framework Codeigniter (PHP)

Sumber : Codeigniter

Sebagai bahasa pemrograman yang digunakan dalam membuat aplikasi web dan web service.

3. Nama : PHP Simple HTML DOM Parser

Sumber : Sourceforge.net

Sebagai library yang digunakan untuk melakukan web scraping.

4. Nama : IE/Firefox/Chrome/Opera/dll

Sumber : berbagai sumber

Sebagai aplikasi web browser untuk membuka sistem web.

5. Nama : Apache

Sumber : Apache

Sebagai web server.

6. Nama : Linux Cent OS

Sumber : Cent OS.

Sebagai sistem operasi untuk web server.

7. Nama : Android

Sumber : Google Inc.

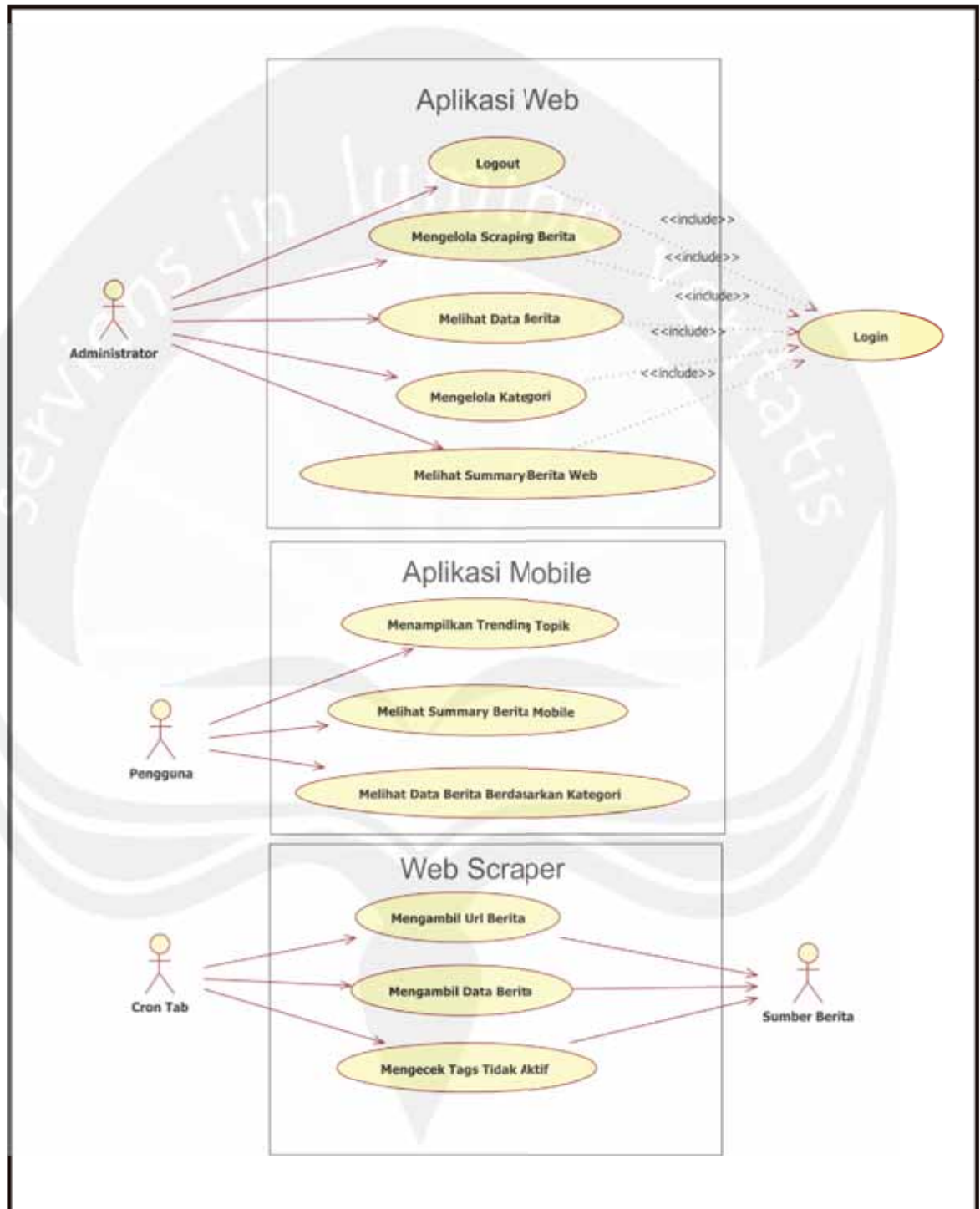
Sebagai sistem operasi untuk perangkat *mobile* yang digunakan untuk menjalankan aplikasi mobile KUBERIN.

#### 3.1.4 Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak web dan mobile KUBERIN menggunakan protocol HTTP.

### 3.2 Kebutuhan fungsionalitas Perangkat Lunak

#### 3.2.1 Use Case Diagram



Gambar 3.1 Use Case Diagram KUBERIN

## 4 Spesifikasi Rinci Kebutuhan

### 4.1 Spesifikasi Kebutuhan Fungsionalitas

#### APLIKASI WEB

##### 4.1.1 Use case Spesification : *Login*

###### 1. Brief Description

Use Case ini digunakan oleh aktor untuk memperoleh akses ke sistem. *Login* didasarkan pada sebuah username dari user dan password yang berupa rangkaian karakter.

###### 2. Primary Actor

1. Administrator

###### 3. Supporting Actor

none

###### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan *login*.
2. Sistem menampilkan antarmuka untuk *login*.
3. Aktor memasukkan username dan password.
4. Sistem memeriksa username dan password yang diinputkan aktor.

E-1 Password atau username tidak sesuai.

5. Sistem memberikan akses ke aktor.
6. Use Case ini selesai.

###### 5. Alternative Flow

none

###### 6. Error Flow

E-1 Password atau username tidak sesuai.

1. Sistem menampilkan peringatan bahwa username atau password tidak sesuai.
2. Kembali ke Basic Flow langkah ke 3.

Program Studi Teknik Informatika	SKPL –KUBERIN	20/ 36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		



## 7. PreConditions

none

## 8. PostConditions

1. Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi pada sistem.

### 4.1.2 Use case Spesification : *Logout*

#### 1. Brief Description

Use Case ini digunakan oleh aktor untuk keluar dari sistem.

#### 2. Primary Actor

1. Administrator

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk keluar dari aplikasi.
2. Sistem menampilkan antarmuka untuk *logout*.
3. Aktor memilih untuk *logout*.
4. Sistem menghapus data aktor dari aplikasi.
5. Use Case ini selesai.

#### 5. Alternative Flow

none

#### 6. Error Flow

none

## 7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.

## 8. PostConditions

1. Aktor keluar dari aplikasi.
2. Antarmuka *login* muncul.

Program Studi Teknik Informatika	SKPL –KUBERIN	21/ 36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

#### 4.1.3 Use case Spesification : Mengelola Data Scraping Berita

##### 1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data Scraping berita. Aktor dapat melakukan test Scraping, simpan data Scraping atau Tags untuk Scraping, tampil data Tags Scraping, dan update data Tags Scraping.

##### 2. Primary Actor

1. Administrator

##### 3. Supporting Actor

none

##### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data Scraping berita.
2. Sistem menampilkan antarmuka untuk mengelola data Scraping berita.
3. Sistem menampilkan form inputan untuk menginputkan URL dan Tags rulenya.
4. Sistem memberikan pilihan untuk melakukan test Scraping, simpan Scraping, tampil Scraping atau update Scraping.
5. Aktor memilih untuk melakukan test Scraping.
  - A-1 Aktor memilih untuk simpan Scraping.
  - A-2 Aktor memilih untuk tampil Scraping.
  - A-3 Aktor memilih untuk update Scraping.

6. Aktor menginputkan data *URL* beserta *Tags* rule nya.
7. Aktor meminta sistem untuk melakukan test *Scraping*.
8. Sistem menampilkan data berita.
  - E-1 Data berita yang didapat tidak tepat.
9. Use Case ini selesai.

#### 5. Alternative Flow

A-1 Aktor memilih untuk melakukan simpan *Scraping*.

1. Aktor meminta sistem untuk menyimpan data *Scraping*.
2. Sistem melakukan pengecekan terhadap data yang akan disimpan.

E-2 Data *Scraping* sudah ada.

3. Sistem menyimpan data *Scraping* ke database.
4. Berlanjut ke Basic Flow langkah ke 9.

A-2 Aktor memilih untuk melakukan tampil *Scraping*.

1. Aktor meminta sistem untuk menampilkan data *Scraping*.
2. Sistem menampilkan data *Scraping*.

3. Berlanjut ke Basic Flow langkah ke 9.

A-3 Aktor memilih untuk melakukan update *Scraping*.

1. Aktor meminta sistem untuk mengupdate data *Scraping*.
2. Sistem melakukan update data *Scraping* sesuai inputan user.

3. Berlanjut ke Basic Flow langkah ke 9.

## 6. Error Flow

E-1 Data berita yang didapat tidak tepat.

1. Sistem menampilkan pesan data kosong.
2. Kembali ke Basic Flow langkah ke 6.

E-2 Data *Scraping* sudah ada.

1. Sistem menampilkan pesan bahwa data sudah ada di database.
2. Kembali ke Basic Flow langkah ke 6.

## 7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.

## 8. PostConditions

1. Data *Scraping* di database telah terupdate dan cron job menjalankan data *Scraping* yang tersimpan di database.

### 4.1.4 Use case Spesification : Melihat Data Berita

#### 1. Brief Description

Use Case ini digunakan oleh aktor untuk melihat data berita, yaitu memfilter berita dan menampilkan data berita.

#### 2. Primary Actor

1. Administrator

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melihat data berita.
2. Sistem menampilkan antarmuka tampil data berita.

Program Studi Teknik Informatika	SKPL –KUBERIN	24/ 36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

3. Sistem menampilkan form inputan untuk menginputkan data yang ingin di filter.
4. Sistem menampilkan data berita.
5. Aktor menginputkan data filter berita.
6. Aktor meminta sistem untuk melakukan filter berita.
7. Sistem menampilkan data berita sesuai dengan filter yang diinputkan aktor.

E-1 Data Kosong.

8. Use Case selesai.

#### **5. Alternative Flow**

none

#### **6. Error Flow**

E-1 Data kosong.

1. Sistem memberikan pesan peringatan bahwa data berita kosong.
2. Kembali ke Basic Flow Langkah ke 5.

#### **7. PreConditions**

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki *sistem*.

#### **8. PostConditions**

1. Data berita yang dicari tampil.

### **4.1.5 Use case Spesification : Mengelola Data Kategori**

#### **1. Brief Description**

Use Case ini digunakan oleh aktor untuk mengelola data kategori. Aktor dapat melakukan entry kategori baru, pindah kategori berita, dan hubung kategori utama dan kategori berita.

Program Studi Teknik Informatika	SKPL –KUBERIN	25/ 36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## 2. Primary Actor

1. Administrator

## 3. Supporting Actor

none

## 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data kategori.
2. Sistem menampilkan antarmuka untuk mengelola data kategori.
3. Sistem menampilkan form inputan untuk menginputkan kategori utama baru.
4. Sistem memberikan pilihan untuk melakukan membuat kategori utama, memindah kategori berita, dan menghubungkan kategori utama dan kategori berita.
5. Aktor memilih untuk melakukan buat kategori utama.

A-1 Aktor memilih memindah data kategori berita.

A-2 Aktor memilih menghubungkan kategori utama dan kategori berita.

6. Aktor menginputkan kategori utama baru.
7. Aktor meminta sistem untuk menyimpan data kategori utama yang telah diinputkan.
8. Sistem mengecek data kategori utama yang telah diinputkan.

E-1 Data kategori utama yang diinputkan oleh aktor sudah ada.

9. Sistem menyimpan data kategori utama ke dalam database.
10. Use Case ini selesai.

## 5. Alternative Flow

A-1 Aktor memilih untuk memindah data kategori berita.

1. Sistem menampilkan data kategori berita yang akan dipindah.
2. Aktor memilih atau menginputkan data kategori berita yang akan dipindah.
3. Aktor meminta sistem untuk menyimpan data kategori berita ke tempat yang baru.
4. Sistem melakukan pengecekan terhadap data yang akan disimpan.
5. Sistem menyimpan data kategori berita ke database.
6. Berlanjut ke Basic Flow langkah ke 10.

A-2 Aktor memilih untuk menghubungkan kategori utama dan kategori berita.

1. Sistem menampilkan data kategori berita dan data kategori utama yang akan dihubungkan.
2. Aktor melakukan pemilihan data kategori berita dan kategori utama yang akan dihubungkan.
3. Aktor meminta sistem untuk menyimpan data hubungan kategori berita dan kategori utama.
4. Sistem melakukan pengecekan terhadap data yang akan disimpan.
5. Sistem menyimpan data hubungan kategori utama dan kategori berita ke database.
6. Berlanjut ke Basic Flow langkah ke 10.

## 6. Error Flow

E-1 Data kategori utama yang diinputkan oleh aktor sudah ada.

Program Studi Teknik Informatika	SKPL –KUBERIN	27/ 36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Sistem menampilkan pesan bahwa data kategori utama sudah ada di database.
2. Kembali ke Basic Flow langkah ke 6.

#### 7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.

#### 8. PostConditions

1. Data kategori utama sudah ditambahkan.
2. Data kategori berita sudah dipindahkan.
3. Data kategori utama dan kategori berita sudah dihubungkan.

#### 4.1.6 Use case Spesification : Melihat *Summary* Berita Web

##### 1. Brief Description

Use Case ini digunakan oleh aktor untuk menampilkan *summary* berita, berita terkait, dan *score* kalimat.

##### 2. Primary Actor

1. Administrator

##### 3. Supporting Actor

none

##### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan *summarize* berita.
2. Sistem menampilkan antarmuka untuk *summarize* berita.
3. Sistem menampilkan form inputan untuk menginputkan filter *summarize* berita yaitu kata kunci, range tanggal, dan sumber berita.

Program Studi Teknik Informatika	SKPL –KUBERIN	28/ 36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		



4. Aktor menginputkan filter *summarize* berita.
5. Aktor meminta sistem untuk mencari data berita yang sesuai dengan filter berita.
6. Sistem menampilkan hasil *summarize* berita, berita terkait, dan score kalimat berdasarkan filter berita yang diinputkan.

E-1 Data berita tidak ada.

7. Use Case ini selesai.

#### 5. Alternative Flow

none

#### 6. Error Flow

E-1 Data berita tidak ada.

1. Sistem menampilkan pesan bahwa data berita tidak ada.
2. Kembali ke Basic Flow langkah ke 4.

#### 7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.

#### 8. PostConditions

1. *Summarize* berita sudah dilakukan.

### APLIKASI MOBILE

#### 4.1.7 Use case Spesification : Menampilkan *Trending* topik

##### 1. Brief Description

Use Case ini digunakan oleh aktor melihat *Trending* topik.

##### 2. Primary Actor

1. Pengguna

##### 3. Supporting Actor

none

Program Studi Teknik Informatika	SKPL –KUBERIN	29/ 36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melihat *Trending* topik.
2. Sistem menampilkan antarmuka tampil data *Trending* topik.
3. Aktor memilih *Trending* topik yang diinginkan.
4. Sistem menampilkan data berita sesuai dengan *Trending* topik yang dipilih.
5. Use Case selesai.

#### 5. Alternative Flow

none

#### 6. Error Flow

none

#### 7. PreConditions

1. Aktor telah memasuki *sistem*.

#### 8. PostConditions

1. Aktor melihat data *Trending* topik.

#### 4.1.8 Use case Spesification : Melihat Summary Berita Mobile

##### 1. Brief Description

Use Case ini digunakan oleh aktor untuk menampilkan *summary* berita dan berita terkait.

##### 2. Primary Actor

1. Pengguna

##### 3. Supporting Actor

none

##### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk *summarize* berita.

Program Studi Teknik Informatika	SKPL –KUBERIN	30/ 36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Sistem menampilkan antarmuka *summarize* berita.
3. Sistem menampilkan form inputan untuk menginputkan filter *summarize* berita yaitu kata kunci, range tanggal, dan sumber berita.
4. Aktor menginputkan filter *summarize* berita.
5. Aktor meminta sistem untuk mencari data berita yang sesuai dengan filter berita.
6. Sistem menampilkan hasil *summary* berita dan berita terkait berdasarkan filter berita yang diinputkan dan data berita yang berkaitan dengan hasil *summary*.

E-1 Data berita tidak ada.

7. Use Case selesai.

#### 5. Alternative Flow

none

#### 6. Error Flow

E-1 Data berita tidak ada.

1. Sistem menampilkan pesan bahwa data berita tidak ada.
2. Kembali ke Basic Flow langkah ke 4.

#### 7. PreConditions

1. Aktor telah memasuki sistem.

#### 8. PostConditions

1. *Summarize* berita sudah dilakukan.

### 4.1.9 Use case Spesification : Melihat Data Berita Berdasarkan Kategori

#### 1. Brief Description

Use Case ini digunakan oleh aktor melihat data berita berdasarkan kategori utama.

Program Studi Teknik Informatika	SKPL –KUBERIN	31/ 36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## 2. Primary Actor

1. Pengguna

## 3. Supporting Actor

none

## 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melihat berita berdasarkan kategori utama.
2. Sistem menampilkan antarmuka tampil data berita.
3. Aktor memilih kategori berita yang diinginkan.
4. Sistem menampilkan data berita sesuai dengan kategori yang dipilih.
5. Use Case selesai.

## 5. Alternative Flow

none

## 6. Error Flow

none

## 7. PreConditions

1. Aktor telah memasuki sistem.

## 8. PostConditions

1. Aktor melihat data berita berdasarkan kategori.

## APLIKASI WEB SCRAPER

### 4.1.10 Use case Spesification : Mengambil URL Berita

#### 1. Brief Description

Use Case ini digunakan oleh aktor untuk mengambil URL berita.

Program Studi Teknik Informatika	SKPL –KUBERIN	32/ 36
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

## 2. Primary Actor

1. Cron tab

## 3. Supporting Actor

none

## 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengambilan *URL* berita berdasarkan *Tags* yang disimpan di database.
2. Aktor melakukan pengambilan data *URL* berita berdasarkan pada *Tags* yang disimpan di database.

E-1 Data *URL* berita tidak ada.

3. Aktor menyimpan data *URL* berita yang didapatkan.

E-2 Data *URL* berita sudah ada di database.

4. Use Case ini selesai.

## 5. Alternative Flow

none

## 6. Error Flow

E-1 Data *URL* berita tidak ada.

1. Kembali ke Basic Flow langkah ke 4.

E-2 Data *URL* berita sudah ada di database.

1. Kembali ke Basic Flow langkah ke 4.

## 7. PreConditions

1. Aktor telah memasuki sistem.

## 8. PostConditions

1. Aktor telah melakukan pengambilan *URL* berita.

#### 4.1.11 Use case Spesification : Mengambil Data Berita

##### 1. Brief Description

Use Case ini digunakan oleh aktor untuk mengambil data berita.

##### 2. Primary Actor

1. Cron tab

##### 3. Supporting Actor

none

##### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengambilan data berita berdasarkan *Tags* yang disimpan di database.
2. Aktor melakukan pengambilan data berita berdasarkan pada *Tags* yang disimpan di database.

E-1 Data berita tidak ada.

3. Aktor menyimpan data *URL* berita yang didapatkan.

E-2 Data berita sudah ada di database.

4. Use Case ini selesai.

##### 5. Alternative Flow

none

##### 6. Error Flow

E-1 Data berita tidak ada.

1. Kembali ke Basic Flow langkah ke 4.

E-2 Data berita sudah ada di database.

1. Kembali ke Basic Flow langkah ke 4.

##### 7. PreConditions

1. Aktor telah memasuki sistem.

## 8. PostConditions

1. Aktor telah melakukan pengambilan data berita.

### 4.1.12 Use case Spesification : Mengecek Tags Tidak Aktif

#### 1. Brief Description

Use Case ini digunakan oleh aktor untuk mengecek *Tags* tidak aktif.

#### 2. Primary Actor

1. Cron tab

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengecekan *Tags* tidak aktif.
2. Aktor melakukan pengecekan *Tags* tidak aktif.
3. Aktor mengganti status *Tags* dari aktif menjadi tidak aktif kalau *Tags* tersebut sudah tidak bisa mengambil data.
4. Use Case ini selesai.

#### 5. Alternative Flow

none

#### 6. Error Flow

none

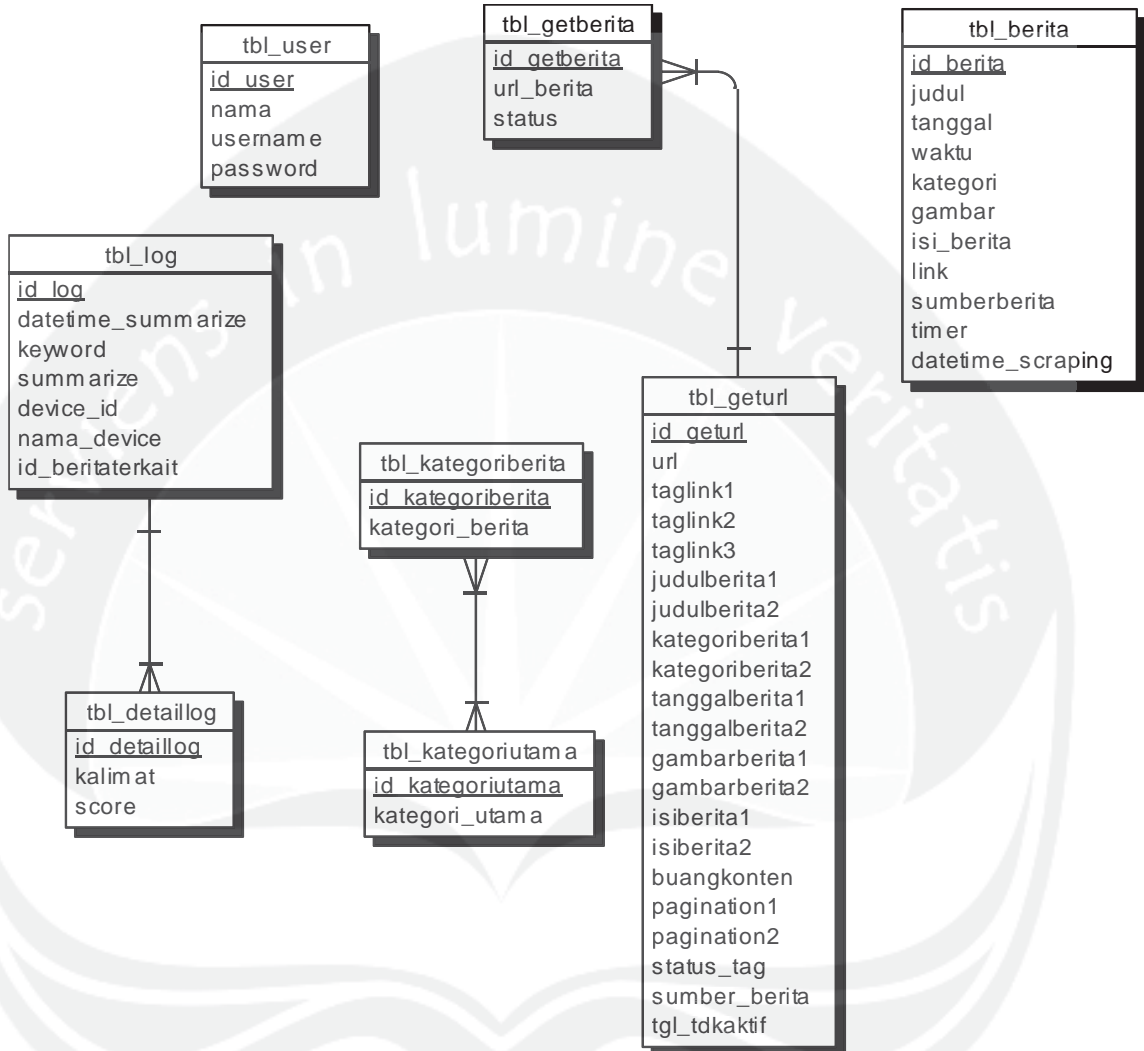
#### 7. PreConditions

1. Aktor telah memasuki sistem.

#### 8. PostConditions

1. Aktor telah melakukan pengecekan *Tags* tidak aktif.

## 5 ERD



Gambar 5.1 Entity Relationship Diagram KUBERIN