



**PERANCANGAN SISTEM APLIKASI MEDIA INFORMASI INTERNAL
PADA PT. SEMEN INDONESIA**



Oleh :

ANAK AGUNG ANGGA WIJAYA

16.41010.0123

FAKULTAS TEKNOLOGI DAN INFORMATIKA

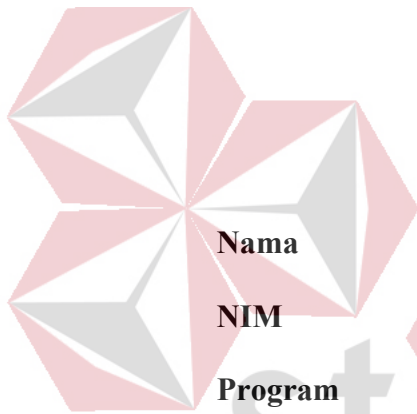
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA

2019

**PERANCANGAN SISTEM APLIKASI MEDIA INFORMASI INTERNAL
PADA PT. SEMEN INDONESIA**

Diajukan sebagian salah satu syarat untuk menyelesaikan

Program Sarjana



Disusun Oleh:

Nama : ANAK AGUNG ANGGAWIJAYA

NIM : 16410100123

Program : S1 (Strata Satu)

Jurusan : Sistem Informasi

FAKULTAS TEKNOLOGI DAN INFORMATIKA

INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA

2019

LEMBAR PENGESAHAN
PERANCANGAN SISTEM APLIKASI MEDIA INFORMASI INTERNAL
PADA PT.SEMEN INDONESIA

Laporan Kerja Praktik oleh
Anak Agung Angga Wijaya
Nim : 16410100123
Telah dipriksa, diuji dan disetujui

Surabaya, Juli 2019

Disetujui :

Pembimbing

Penyelia


Dr. Anjik Sukmaaji, S.Kom., M.Eng.
NIDN. 0731057301


Muhammad Ramzi, S.Kom

 Mengetahui,

Ketua Program Studi S1 Sistem Informasi




Dr. Anjik Sukmaaji, S.Kom., M.Eng.
NIDN. 0731057301



“a luta continua a vitória é certa”

INSTITUT BISNIS
DAN INFORMATIKA

stikom
SURABAYA



*Ku persembahkan kepada
Keluargaku yang ku sayangi,
Beserta semua teman dan sahabat yang selalu
Mendukungku.*

INSTITUT BISNIS
DAN INFORMATIKA

stikom
SURABAYA

**SURAT PERNYATAAN
PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH**

Sebagai mahasiswa Institut Bisnis dan Informatika Stikom Surabaya, saya :

Nama : Anak Agung Angga Wijaya
Nim : 16410100123
Program Studi : S1 Sistem Informasi
Fakultas : Fakultas Teknologi dan Informatika
Jenis Karya : Laporan Kerja Praktek
Judul Karya : **PERANCANGAN SISTEM APLIKASI
MEDIA INFORMASI INTERNAL PADA PT.
SEMEN INDONESIA**

Menyatakan dengan sesungguhnya bahwa:

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, saya menyetujui memberikan kepada Institut Bisnis dan Informatika Stikom Surabaya Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalti Free Right*) atas seluruh isi/ sebagian karya ilmiah saya tersebut di atas untuk disimpan, diahlimeidiakan dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta
2. Karya tersebut di atas adalah karya asli saya, bukan plagiat baik sebagian maupun keseluruhan, Kutipan karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka saya bersedia untuk menerima pencabut terhadap gelar kerjasama yang telah diberikan kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, Juli 2019

Yang menvatakan



Anak Agung Angga Wijaya
NIM : 16410100123

ABSTRAK

Aplikasi SMILE terdahulu memiliki masalah yang cukup berarti, yaitu *load* halaman yang lambat karena *resource* yang di *load* besar dan banyak. Sehingga menyebabkan pegawai dari PT Semen Indonesia enggan untuk menggunakan aplikasi tersebut. Aplikasi tersebut jadi kurang bisa memberikan manfaat karena *user* yang sedikit.

Solusi dari masalah tersebut adalah perlunya pembuatan ulang aplikasi SMILE dikarenakan aplikasi SMILE yang terdahulu sudah tergolong *outdated*. Walaupun masih memaksakan untuk melakukan *upgrade* berdasar *codebase* versi lama sudah tidak memungkinkan karena tidak dapat mengikuti perkembangan dikemudian hari (*legacy code*). Untuk aplikasi SMILE yang baru harus membawa teknologi baru dan menambahkan *plugins* yang bisa membuat proses *load resource* lebih cepat dan efisien.

Berdasarkan permasalahan yang ada dan solusi yang saya sarankan, membuat ulang aplikasi dengan ReactJS. Karena ReactJS bisa dapat mengatasi *load resource* lebih efisien dan cepat. Dan dukungan *plugin* yang begitu banyak sehingga dapat memudahkan pembuatan aplikasi .

Kata kunci: *web*, media informasi, semen indonesia

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa atas segala nikmat yang diberikan sehingga penulis dapat melaksanakan kerja praktik dan menyelesaikan pembuatan laporan dari kerja praktik ini. Laporan ini disusun berdasarkan kerja praktik dan hasil studi yang dilakukan selama lebih kurang satu bulan di PT Sinergi Informatika Semen Indonesia kota Gresik.

Kerja Praktik ini membahas tentang pembuatan aplikasi media informasi internal pada PT Sinergi Informatika Semen Indonesia kota Surabaya yang berfungsi sebagai portal berita atau media informasi internal dan semua kegiatan yang berkaitan pada PT Semen Indonesia

Penyelesaian laporan kerja praktik ini tidak terlepas dari bantuan berbagai pihak yang telah memberikan banyak masukan, nasihat, saran, kritik dan dukungan moral maupun materil kepada penulis. Oleh karena itu penulis menyampaikan rasa terima kasih kepada:

1. Papa dan mamaku tercinta serta keluarga besarku yang selalu mendoakan, mendukung, dan memberikan semangat di setiap langkah dan aktifitas penulis.
2. Bapak Prof. Dr. Budi Jatmiko, M.Pd. selaku rektor Institut Bisnis dan Informatika Stikom Surabaya yang telah mengesahkan dan memberikan kesempatan secara resmi dalam melakukan kerja praktik.
3. Bapak Dr. Anjik Sukmaaji, S.Kom., M.Eng selaku Kepala Program Studi Sistem Informasi Institut Bisnis dan Informatika Stikom Surabaya serta dosen pembimbing dalam kegiatan kerja praktik yang telah memberikan izin kepada penulis untuk melakukan kerja praktik.

4. Bapak Muhammad Ramzy selaku pegawai PT Sinergi Informatika Semen Indonesia yang telah memberikan dukungan serta kesempatan dalam melakukan kerja praktik kepada penulis.
5. Teman-teman tercinta yang memberikan bantuan dan dukungannya dalam penyusunan proposal ini.
6. Pihak-pihak lain yang tidak disebutkan satu-persatu yang telah memberikan bantuan dan dukungan kepada penulis.

Semoga Tuhan YME memberikan balasan yang setimpal kepada semua pihak yang telah memberikan bantuan, bimbingan, dan nasehat dalam proses kerja praktik ini.

Penulis menyadari bahwa kerja praktik ini yang dikerjakan masih banyak terdapat kekurangan, sehingga kritik yang bersifat membangun dan saran dari semua pihak sangatlah diharapkan agar aplikasi ini dapat diperbaiki menjadi lebih baik lagi dikemudian hari. Semoga laporan kerja praktik ini dapat diterima dan bermanfaat bagi penulis dan semua pihak.

Surabaya, Juni 2019

Penulis

DAFTAR ISI

	Halaman
ABSTRAK	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL	xiv
DAFTAR GAMBAR	xvi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan	2
1.5 Manfaat	3
1.6 Sistematika Penulisan	3
BAB II GAMBARAN UMUM INSTANSI	5
2.1 Latar Belakang Perusahaan	5
2.2 Identitas Instansi	6
2.3 Sejarah Perusahaan	6
2.4 Visi dan Misi Instansi	7
2.5 Jumlah Pelanggan	8
2.6 Struktur Organisasi	8
BAB III LANDASAN TEORI	9
3.1 React JS	9
3.2 Apache	9
3.3 PWA (<i>Progressive Web App</i>)	10

3.4	Rest API.....	11
3.5	JWT (<i>Json Web Token</i>).....	12
3.6	React-Redux.....	13
3.7	<i>Localstorage</i>	15
BAB IV DESKRIPSI PEKERJAAN		16
4.1	Analisis dan Desain Sistem.....	16
4.2	Perancangan Sistem	17
4.2.1	<i>Login</i>	19
4.2.2	Mengelola Sinergi	20
4.2.3	Mengelola Video	21
4.2.4	Mengelola <i>Event</i>	22
4.2.5	Mengelola Artikel	23
4.2.6	Mengelola Pengumuman.....	24
4.2.7	Mengelola <i>Newsfeed</i>	25
4.2.8	Mengelola <i>Data Master</i>	26
4.2.9	Mengelola <i>Newspaper</i>	27
4.2.10	Melihat SMI <i>Gallery</i>	28
4.2.11	Melihat SMI <i>Activity</i>	29
4.2.12	Melihat SMI <i>News</i>	30
4.3	<i>Flow Of Event</i>	31
4.3.1	<i>Flow Of Event Login</i>	31
4.3.2	<i>Flow Of Event Mengelola Video</i>	32
4.3.3	<i>Flow Of Event Mengelola Sinergi</i>	33
4.3.4	<i>Flow Of Event Mengelola Event</i>	34
4.3.5	<i>Flow Of Event Mengelola Artikel</i>	35
4.3.6	<i>Flow Of Event Mengelola Pengumuman</i>	36

4.3.7	<i>Flow Of Event Mengelola Newsfeed</i>	37
4.3.8	<i>Flow Of Event Mengelola Data Master</i>	38
4.3.9	<i>Flow Of Event Mengelola Newspaper</i>	39
4.3.10	<i>Flow of Event Melihat Sosial Media</i>	40
4.3.11	<i>Flow of Event Melihat Gallery</i>	40
4.3.12	<i>Flow of Event Melihat Pengumuman</i>	41
4.3.13	<i>Flow of Event Melihat Event</i>	41
4.3.14	<i>Flow of Event Melihat Artikel</i>	42
4.3.15	<i>Flow of Event Melihat Newspaper</i>	42
4.3.16	<i>Flow of Event Melihat Newsfeed</i>	43
4.4	<i>Sequence Diagram</i>	43
4.4.1	<i>Login Admin</i>	43
4.4.2	<i>Admin Announcement</i>	44
4.4.3	<i>Admin Article</i>	44
4.4.4	<i>Admin Category</i>	45
4.4.5	<i>Admin Event</i>	45
4.4.6	<i>Admin Link</i>	46
4.4.7	<i>Admin Newsfeed</i>	46
4.4.8	<i>Admin Newspaper</i>	47
4.4.9	<i>Admin Opco</i>	47
4.4.10	<i>Admin Role</i>	48
4.4.11	<i>Admin Sinergi</i>	48
4.4.12	<i>Admin Testimoni</i>	49
4.4.13	<i>Admin User</i>	49
4.4.14	<i>Login User</i>	50
4.4.15	<i>User SMI Activity</i>	50

4.4.16	<i>User SMI News</i>	51
4.4.17	<i>User SMI Gallery</i>	51
4.5	<i>Class Diagram</i>	52
4.5.1	<i>p_user</i>	53
4.5.2	<i>Login</i>	53
4.5.3	<i>Comment Artikel</i>	53
4.5.4	<i>Latest video</i>	54
4.5.5	<i>Home</i>	54
4.5.6	<i>Halaman SMI News</i>	54
4.5.7	<i>Halaman SMI Activity</i>	54
4.5.8	<i>Halaman Gallery</i>	55
4.5.9	<i>Gallery</i>	55
4.5.10	<i>Event</i>	55
4.5.11	<i>Announcement</i>	56
4.5.12	<i>Koran</i>	56
4.5.13	<i>Halaman Artikel</i>	56
4.5.14	<i>Artikel</i>	57
4.6	<i>Kebutuhan Sistem</i>	57
4.6.1	<i>ReactJS</i>	57
4.6.2	<i>Codeigniter</i>	64
BAB V PENUTUP		68
LAMPIRAN		71

DAFTAR TABEL

	Halaman
Tabel 1.1 <i>Flow of Event Login</i>	31
Tabel 1.2 <i>Flow of Event Mengelola Video</i>	32
Tabel 1.3 <i>Flow of Event Mengelola Sinergi</i>	33
Tabel 1.4 <i>Flow of Event Mengelola Event</i>	34
Tabel 1.5 <i>Flow of Event Mengelola Artikel</i>	35
Tabel 1.6 <i>Flow of Event Mengelola Pengumuman</i>	36
Tabel 1.7 <i>Flow of Event Mengelola Newsfeed</i>	37
Tabel 1.8 <i>Flow of Event Mengelola Data Master</i>	38
Tabel 1.9 <i>Flow of Event Mengelola Newspaper</i>	39
Tabel 1.10 <i>Flow of Event Melihat Sosial Media</i>	40
Tabel 1.11 <i>Flow of Event Melihat Gallery</i>	40
Tabel 1.12 <i>Flow of Event</i>	41
Tabel 1.13 <i>Flow of Event Melihat Event</i>	41
Tabel 1.14 <i>Flow of Event melihat Artikel</i>	42
Tabel 1.15 <i>Flow of Event Melihat Newspaper</i>	42
Tabel 1.16 <i>Flow of Event Melihat Newsfeed</i>	43
Tabel 1.17 <i>Class Diagram p_user</i>	53
Tabel 1.18 <i>Class Diagram Login</i>	53
Tabel 1.19 <i>Class Diagram comment_artikel</i>	53
Tabel 1.20 <i>Class Diagram latest_video</i>	54
Tabel 1.21 <i>Class Diagram home</i>	54
Tabel 1.22 <i>Class Diagram halaman_sminews</i>	54

Tabel 1.23 <i>Class Diagram</i> halaman_smiactivity	54
Tabel 1.24 <i>Class Diagram</i> halaman_gallery	55
Tabel 1.25 <i>Class Diagram</i> gallery	55
Tabel 1.26 <i>Class Diagram</i> event	55
Tabel 1.27 <i>Class Diagram</i> announcement	56
Tabel 1.28 <i>Class Diagram</i> koran.....	56
Tabel 1.29 <i>Class Diagram</i> halaman_artikel.....	56
Tabel 1.30 <i>Class Diagram</i> artikel.....	57



DAFTAR GAMBAR

	Halaman
Gambar 2.1 PT. Semen Indonesia.....	5
Gambar 2.2 Peta PT. Sinergi Informatika Semen Indonesia	6
Gambar 4.1 <i>Use Case</i> Diagram Aplikasi.....	18
Gambar 4.2 <i>Activity</i> Diagram <i>Login</i>	19
Gambar 4.3 <i>Activity</i> Diagram Mengelola Sinergi	20
Gambar 4.4 <i>Activity</i> Diagram Mengelola Video.....	21
Gambar 4.5 <i>Activity</i> Diagram Mengelola <i>Event</i>	22
Gambar 4.6 <i>Activity</i> Diagram Mengelola Artikel	23
Gambar 4.7 <i>Activity</i> Diagram Mengelola Pengumuman.....	24
Gambar 4.8 <i>Activity</i> Diagram Mengelola <i>Newsfeed</i>	25
Gambar 4.9 <i>Activity</i> Diagram Mengelola Data <i>Master</i>	26
Gambar 4.10 <i>Activity</i> Diagram Mengelola <i>Newspaper</i>	27
Gambar 4.11 <i>Activity</i> Diagram Melihat SMI <i>Gallery</i>	28
Gambar 4.12 <i>Activity</i> Diagram Melihat SMI <i>Activity</i>	29
Gambar 4.13 <i>Activity</i> Diagram Melihat SMI <i>News</i>	30
Gambar 4.14 <i>Sequence</i> <i>Login</i> Admin	43
Gambar 4.15 <i>Sequence</i> <i>Announcement</i>	44
Gambar 4.16 <i>Sequence</i> Admin Article	44
Gambar 4.17 <i>Sequence</i> Admin Category	45
Gambar 4.18 <i>Sequence</i> Admin Event	45
Gambar 4.19 <i>Sequence</i> Admin Link.....	46
Gambar 4.20 <i>Sequence</i> Admin <i>Newsfeed</i>	46

Gambar 4.21 Sequence Admin Newspaper	47
Gambar 4.22 Sequence Admin Opco	47
Gambar 4.23 Sequence Admin Role	48
Gambar 4.24 Sequence Admin Sinergi	48
Gambar 4.25 Sequence Admin Testimoni	49
Gambar 4.26 Sequence Admin User	49
Gambar 4.27 Sequence Login User	50
Gambar 4.28 Sequence SMI Activity	50
Gambar 4.29 Sequence SMI News	51
Gambar 4.30 Sequence SMI Gallery	51
Gambar 4.31 <i>Class Diagram</i>	52
Gambar 4.32 <i>Store</i>	58
Gambar 4.33 <i>Reducer</i>	58
Gambar 4.34 <i>Action</i>	59
Gambar 4.35 <i>Lazy Load</i>	60
Gambar 4.36 <i>Progressive Image</i>	61
Gambar 4.37 <i>LocalStorage</i>	62
Gambar 4.38 <i>localsotrage</i>	62
Gambar 4.39 <i>Axios</i>	63
Gambar 4.40 <i>JWT</i>	64
Gambar 4.41 <i>Datatables</i>	65
Gambar 4.42 <i>RestAPI</i>	66
Gambar 4. 43 <i>CKEditor</i>	67

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Surat Balasan Instansi/Perusahaan	71
Lampiran 2. Form KP-5 (Acuan Kerja)	72
Lampiran 3. Form KP-5 Garis Besar Rencana Kerja Mingguan	73
Lampiran 4. Form KP-6 Log Harian dan Catatan Perubahan Acuan Kerja.....	74
Lampiran 5. Form KP-7 Kehadiran Kerja Praktik	75
Lampiran 6. Kartu Bimbingan Kerja Praktik	76
Lampiran 7. Biodata Penulis	77



BAB I

PENDAHULUAN

1.1 Latar Belakang

PT Sinergi Informatika Semen Indonesia merupakan anak usaha dari PT Semen Indonesia (Parsero) Tbk. Yang bergerak di bidang teknologi informasi dan komunikasi. Resmi berdiri pada tanggal 9 juni 2014, SISI dibentuk berdasar dari pengalaman mengelola ICT Semen Indonesia *Group* selama lebih dari 15 tahun. SISI secara konsisten menunjukan jati dirinya sebagai perusahaan penyedia solusi IT terkemuka di Indonesia dengan terus menemukan dan menciptakan *best practice* dalam pengelolaan bisnis baik bagi perusahaan maupun masyarakat secara umum.

Proses bisnis yang dilakukan oleh PT. Sinergi Informatika Semen Indonesia membuat berbagai kebutuhan dalam bentuk aplikasi untuk pihak internal. Salah satunya adalah aplikasi media informasi internal. Semua kegiatan, berita, dan pengumuman akan di *post* di aplikasi media informasi tersebut.

Tetapi saat ini perfoma *website* sangat kurang, dikarenakan terlalu banyak *load* data dan UI/UX yang kurang efisien dan efektif, serta teknologi yang digunakan bisa dikatakan kuno tidak mengikuti perkembangan jaman. Sehingga menyebabkan *website* tersebut menjadi sangat lambat dalam mengelola atau mengaksesnya. Terkadang tidak bisa dibuka sama sekali tergantung juga dengan jaringan internet yang ada.

Dengan kondisi yang terjadi saat ini, maka perlu adanya solusi untuk mengatasi masalah tersebut. Solusi dari permasalahan tersebut adalah perlu dibangun aplikasi Media Informasi Internal pada PT. Sinergi Informatika Semen Indonesia dengan menggunakan teknologi yang terbaru dan dapat dijalankan di semua *platform*.

1.2 Rumusan Masalah

Berdasar latar belakang yang dijabarkan diatas, masalah yang ada pada rancang bangun aplikasi media informasi internal pada PT. Sinergi Informatika Semen Indonesia. Dapat dirumuskan sebagai berikut: Bagaimana merancang aplikasi media informasi internal dengan perfoma yang maksimal serta bisa dijalankan di berbagai *platform*?

1.3 Batasan Masalah

Berdasarkan penjelasan pada latar belakang maka dibuatlah batasan masalah agar pembahasan masalah tidak melebar. Batasan masalah sebagai berikut:

- a. Aplikasi Berbasis WEB
- b. Aplikasi dibangun menggunakan *framework* ReactJS
- c. Aplikasi dibangun menggunakan *framework* Code Igniter 3
- d. Aplikasi dibangun dengan php 5.6
- e. *Database* yang digunakan yaitu MySql

1.4 Tujuan

Berdasarkan latar belakang dan rumusan masalah, maka tujuan dari kerja praktik ini adalah membangun aplikasi media informasi internal pada PT. Sinergi Informatika Semen Indonesia.

1.5 Manfaat

Manfaat yang diharapkan dengan dibangunnya aplikasi ini sebagai berikut:

- a. *Multi platform*.
- b. Menggunakan *Progressive Web Apps* sehingga dapat dikontrol mengenai performa *website* tersebut.
- c. Integrasi pada setiap anak perusahaan PT. Semen Indonesia.

1.6 Sistematika Penulisan

Untuk memberikan gambaran menyeluruh terhadap masalah yang dibahas, maka sistematika penulisan dibagi ke dalam beberapa bab yaitu:

BAB I

PENDAHULUAN

Pada bab ini menjelaskan tentang latar belakang dari hal-hal yang berhubungan dengan perusahaan, rumusan masalah, batasan masalah, tujuan yang ingin dicapai, manfaat yang diperoleh dengan adanya aplikasi yang telah dibuat, serta sistematika penulisan dari proposal.

BAB II

GAMBARAN UMUM INSTANSI

Bab ini menjelaskan tentang PT. Sinergi Informasi Semen Indonesia, mulai dari visi & misi perusahaan, dan stuktur organisasi.

BAB III

LANDASAN TEORI

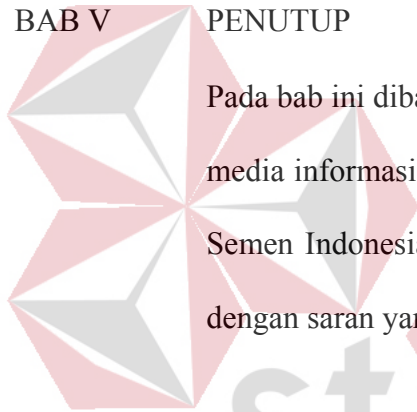
Pada bab ini membahas tentang teori-teori yang dianggap berhubungan dengan kerja praktik yang dilakukan, dimana teori-teori tersebut akan menjadi acuan untuk penyelesaian masalah.

BAB IV DESKRIPSI PEKERJAAN

Bab ini menguraikan tentang langkah-langkah yang digunakan untuk pembuatan sistem yang digunakan untuk penyelesaian masalah yang membahas keseluruhan desain input, proses, dan output dari sistem. Pada bab ini juga membahas tentang implementasi dari perancangan yang telah dilakukan dalam pembuatan aplikasi media informasi internal pada PT. Sinergi Informatika Semen Indonesia.

BAB V PENUTUP

Pada bab ini dibahas mengenai kesimpulan dari pembuatan aplikasi media informasi internal berbasis web pada PT. Sinergi Informatika Semen Indonesia terkait dengan tujuan dan permasalahan, beserta dengan saran yang bermanfaat untuk pengembangan aplikasi ini.



stikom
SURABAYA

BAB II

GAMBARAN UMUM INSTANSI

2.1 Latar Belakang Perusahaan



Gambar 2.1 PT. Semen Indonesia

PT. Sinergi Informatika Semen Indonesia merupakan perusahaan yang bergerak pada bidang Teknologi Informasi di kota Gresik dan seluruh Indonesia. Perusahaan ini merupakan anak cabang dari PT. Semen Indonesia.

2.2 Identitas Instansi

Nama Instansi : PT. Sinergi Informatika Semen Indonesia

Alamat : Jl. Veteran, Gresik Jawa Timur, 61122

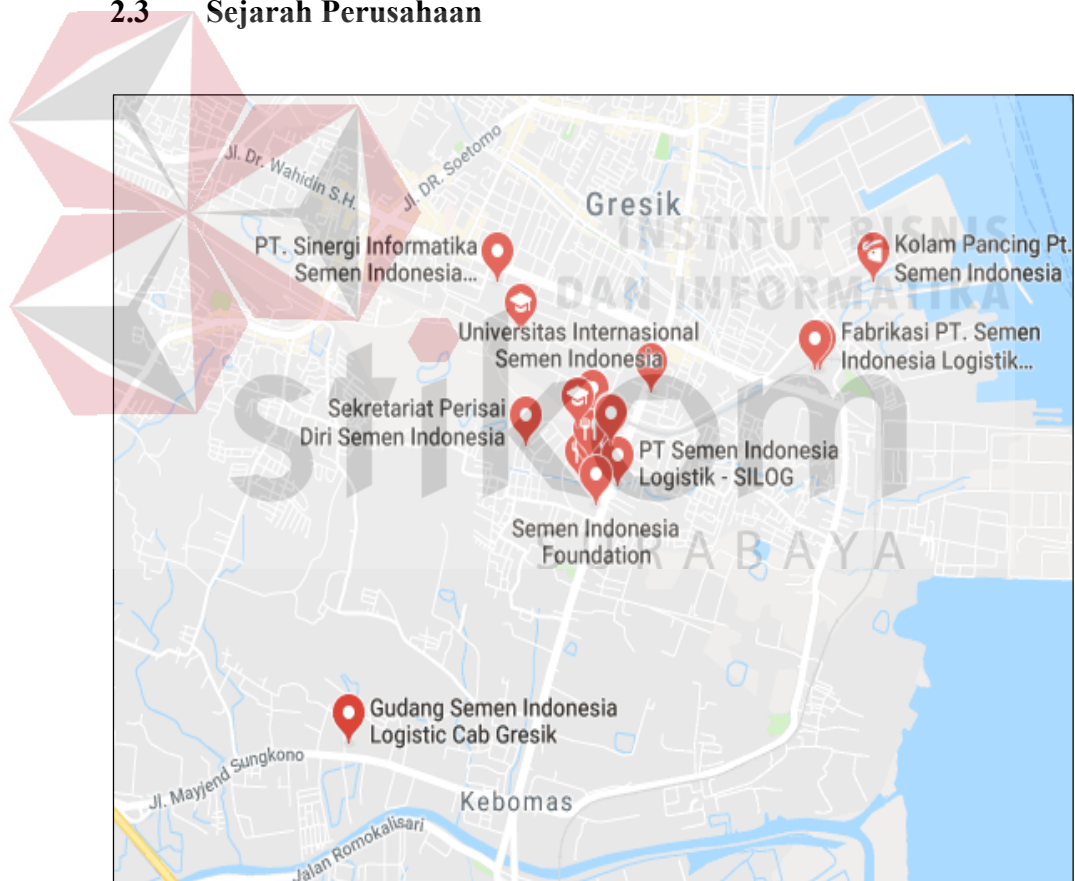
No. Telepon : (021) 52137111

No. Fax : -

Website : www.sisi.id

Email : ptsisi@sisi.id

2.3 Sejarah Perusahaan



Gambar 2.2 Peta PT. Sinergi Informatika Semen Indonesia

2014 : PT Sinergi Informatika Semen Indonesia (SISI) merupakan anak usaha dari PT Semen Indonesia (Persero) Tbk. yang bergerak di bidang teknologi informasi dan komunikasi.

Resmi berdiri pada tanggal 9 Juni 2014, SISI dibentuk berdasar dari pengalaman mengelola ICT Semen Indonesia Group selama lebih dari 15 tahun. SISI secara konsisten menunjukkan jati dirinya sebagai perusahaan penyedia solusi IT terkemuka di Indonesia dengan terus menemukan dan menciptakan *best practice* dalam pengelolaan bisnis baik bagi perusahaan maupun masyarakat secara umum.

2.4 Visi dan Misi Instansi

Visi

To Be the Leading of Information & Communication Technology Company in South East Asia

Misi

1. Mengembangkan solusi ICT dan model bisnis yang inovatif untuk peningkatan daya saing dunia bisnis dan kemudahan bagi masyarakat umum.
2. Mengembangkan sistem manajemen untuk pengembangan, penyediaan dan pengelolaan layanan ICT yang unggul dan berorientasi kepada kepuasan pelanggan.

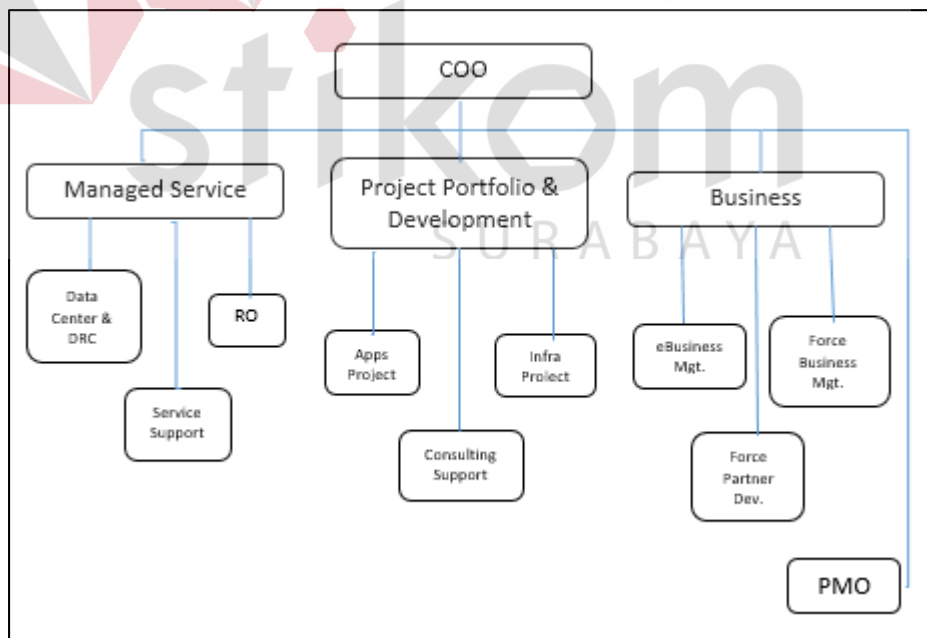
3. Berkontribusi terhadap pengembangan kualitas SDM di bidang ICT di Indonesia.
4. Meningkatkan *value* perusahaan secara berkesinambungan

2.5 Jumlah Pelanggan

Pelayanan Teknologi Informasi dan Komunikasi pada PT. Sinergi Informatika Semen Indonesia tidak hanya terbatas pada PT. Semen Indonesia, daerah Gresik dan sekitarnya. Total pihak yang telah menjadi pelanggan PT. Sinergi Informatika Semen Indonesia mencapai 24 perusahaan berskala nasional dan akan terus bertambah lagi.

2.6 Struktur Organisasi

Struktur organisasi pada PT. Sinergi Informatika Semen Indonesia bagian COO adalah sebagai berikut.



Gambar 2.1 Struktur Organisasi Bagian COO

BAB III

LANDASAN TEORI

3.1 React JS

ReactJS merupakan *library* JavaScript yang bersifat *open source* dan dikelola oleh perusahaan IT ternama didunia seperti Facebook dan Instagram bersama dengan komunitas pengembang dari seluruh dunia. *Framework* ini digunakan secara luas untuk mengembangkan tampilan antarmuka pada sebuah aplikasi web. (Web, 2016)

Sudah banyak alat atau aplikasi yang diciptakan dan digunakan oleh para pengembang, namun tidak banyak yang benar-benar bisa merevolusi cara kerja sebuah tim pengembang, salah satunya yang bisa merevolusi alur kerja adalah ReactJS. (Web, 2016)

Berikut ini merupakan kelebihan dari react JS yaitu :

1. ReactJS sangat efisien
2. *Library* Javascript
3. Sangat bagus untuk *Search Engine Optimization*
4. Fokus pada antarmuka

3.2 Apache

Apache HTTP *server* adalah perangkat lunak dengan *platform operating system* (OS) yang mendukung *multi-tasking*, dan menyediakan layanan untuk aplikasi lain yang terhubung ke dalamnya, seperti web *browser*. Apache pertama kali dikembangkan untuk bekerja dengan sistem operasi Linux/Unix, tetapi

kemudian diadaptasi untuk bekerja di bawah sistem lain, termasuk Windows dan Mac. (Aziz & Tampati, 2015).

Apache mempunyai beberapa kelebihan dibandingkan dengan web *server* lain. Berikut ini kelebihan dari web *server* Apache menurut (Ariata, 2019) :

1. *Open-source* dan gratis, bahkan untuk tujuan komersial.
2. *Software* yang andal dan stabil.
3. *Patch* keamanan yang terus-menerus diperbarui.
4. Fleksibel karena memiliki struktur berbasis modul.
5. Kemudahan konfigurasi dan tidak sulit bagi pemula.
6. Lintas *platform* (dapat berfungsi baik di *server* Unix maupun Windows).
7. Dapat digunakan di situs WordPress.
8. Komunitasnya besar dan memudahkan pengguna jika menemukan masalah.

3.3 PWA (*Progressive Web App*)

Progressive Web Apps (PWA) adalah konsep pengalaman pengguna yang menggabungkan bagian terbaik web dan bagian terbaik *native apps*. PWA berguna bagi pengguna sejak pertama membuka halaman sebuah web dengan konsep PWA, dan seiring dengan pengguna menggunakan aplikasi web lebih banyak lagi, aplikasi akan menjadi semakin *powerful*. Aplikasi dapat dimuat dengan cepat, bahkan dalam kondisi internet yang kurang baik, bisa mengirim *push notifications*, punya ikon aplikasi di *home screen*, dan bisa berjalan dalam mode layar penuh. (Adi, Akbar, & Khotimah, 2017)

PWA sepenuhnya mengandalkan *browser* pengguna dan teknologi yang ada didalamnya. Sampai saat studi ini ditulis, sudah ada 73,61% dari seluruh

browser di seluruh dunia yang mendukung fitur *service worker*, seperti Mozilla Firefox, Google Chrome, Chrome for Android dan Opera, sementara Edge dan Safari belum mendukung fitur ini. (Adi, Akbar, & Khotimah, 2017)

3.4 Rest API

REST menurut Sinha et al dan Zhou et al merupakan gaya arsitektur dalam mendesain sebuah web *service* di mana desain REST memiliki *resource* yang dapat diakses melalui sebuah alamat HTTP URL yang *unique*. REST juga memungkinkan klien dapat melakukan *request* melalui *protokol* HTTP dengan mudah menggunakan URI seperti pada penelitian Kurniawan. Masing-masing alamat URL mengacu kepada kumpulan program yang akan dieksekusi dan akan mengembalikan pesan kepada pengirim perintah. (Rahman, 2013)

REST mengirimkan perintah yang akan dikerjakan oleh *server* menggunakan metode-metode HTTP *request method* yang disebut verb. Mengacu pada penelitian Lee dan Rahman terdapat delapan HTTP *request method*, yaitu *GET*, *POST*, *PUT*, *DELETE*, *OPTIONS*, *HEAD*, *TRACE*, dan *CONNECT*. Dalam penggunaan API REST hanya menggunakan empat dari metode-metode tersebut, yaitu: *GET*, *POST*, *PUT*, dan *DELETE*. Pesan yang diterima dari *server* berupa kode HTTP berhasil atau gagal di dalam header dan isi pesan hasil pengolahan program itu sendiri. Berikut adalah kode HTTP yang sering digunakan dalam penggunaan REST API

4.5.1. 200 OK

Perintah yang dikirim ke *server* benar dan berhasil dijalankan.

4.5.2. 400 Bad Request

Perintah yang dikirim ke *server* berisi isian yang salah.

4.5.3. 401 *Unauthorized*

Pengirim perintah mengirimkan kode kunci yang salah.

4.5.4. 403 *Forbidden*

Pengirim perintah tidak memiliki hak akses ke dalam *resource* yang dituju

4.5.5. 404 *Not Found*

Resource yang dituju tidak ditemukan dalam *server*.

4.5.6. 429 *Too Many Requests*

Pengirim perintah mengakses mencapai/melebihi dari *limit* yang telah ditentukan dari batas waktu tertentu.

4.5.7. 500 *Internal Server Error*

Server atau potongan program dalam *resource* mengalami kesalahan (Rahman, 2013)

3.5 **JWT (*Json Web Token*)**

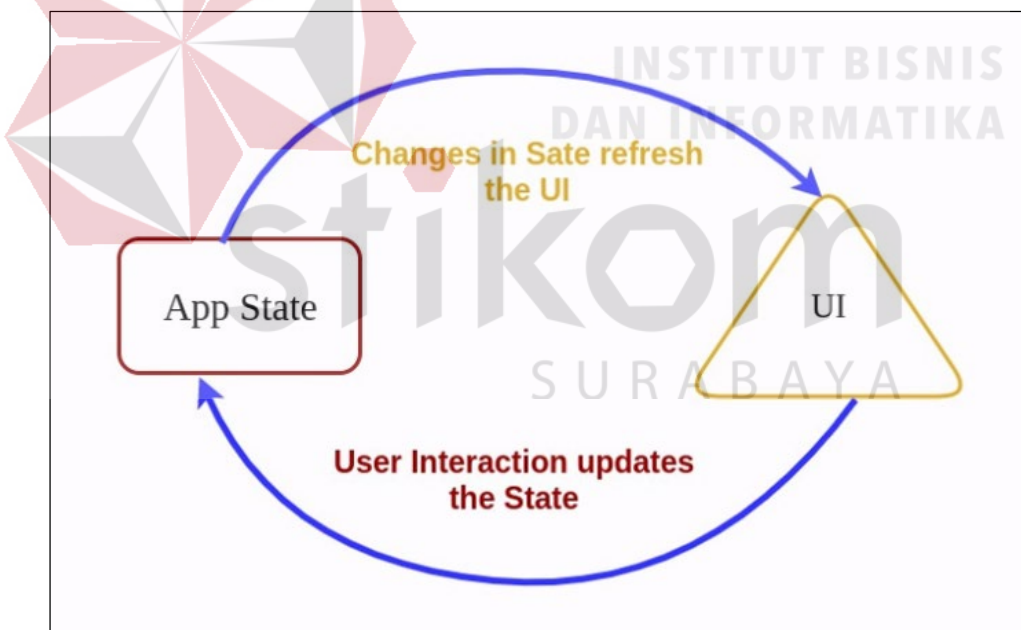
JWT ini adalah sebuah token berbentuk *string* panjang yang sangat *random* yang gunanya sendiri untuk melakukan sistem Otentikasi dan Pertukaran Informasi. Umumnya untuk melakukan login tidak seperti pada aplikasi website biasa dimana kita menggunakan *session* untuk mengingat siapa yang sedang *login*. Tapi didalam API sendiri kita menggunakan konsep JWT atau dibacanya sebagai "jot". (Salma, 2017)

JWT mengamankan informasi menjadi sebuah klaim yang di *encode* ke dalam bentuk JSON dan menjadi *payload* dari JSON Web Signature (JWS). Alasan penggunaan autentikasi JWT yang berbasis *token* adalah adanya *expiration*. Ukuran JWT yang kecil memungkinkannya dapat dikirimkan melalui

URL, parameter HTTP *POST*, atau *header* HTTP *POST*. *Token* disimpan disisi klien sebagai *cookies* sehingga *server* tidak harus menyimpan *token*. Karena *token* sudah ditandatangani secara digital maka dapat diketahui siapa yang melakukan *request*, karena isi dari tanda tangan pada JWT merupakan gabungan dari isi *header* dan *payload*, jika terjadi perubahan maka *signature* akan menjadi tidak valid. (Putra, Bhawiyuga, & Data, 2018)

3.6 React-Redux

Redux adalah pustaka *state management* yang dapat Anda kaitkan dengan pustaka JavaScript apa pun, dan bukan hanya Bereaksi Namun, ia bekerja sangat baik dengan Bereaksi karena sifat fungsional React. (Manjunath, 2018)



Gambar 3.1 Penjelasan redux terhadap UI

Berikut ini adalah skenario yang mungkin Anda temui saat bekerja dengan React dan mengapa harus menggunakan redux :

1. Anda sedang membangun aplikasi berukuran sedang, dan Anda memiliki komponen yang terbagi rapi menjadi *container component* dan *presentational component*.
2. *Container component* menangani keadaan dan kemudian meneruskannya ke *presentational component*. Mereka mengurus pembuatan panggilan API, mengambil data dari sumber data, memproses data, dan kemudian mengatur negara. *Presentational component* menerima *props* dan mengembalikan *representasi UI*.
3. Ketika anda akan menulis komponen baru, tidak selalu jelas di mana menempatkan negara. Anda dapat membiarkan negara bagian dari wadah yang merupakan induk langsung dari komponen *presentasional*. Lebih baik lagi, Anda dapat memindahkan *state* lebih tinggi dalam *hierarki* sehingga *state* dapat diakses ke beberapa komponen *presentasional*.
4. Ketika aplikasi tumbuh, Anda melihat bahwa *state* tersebar di semua tempat. Ketika suatu komponen perlu mengakses keadaan yang tidak segera memiliki akses. Anda akan mencoba untuk mengangkat *state* ke *root component* terdekat.
5. Setelah *refactoring* dan pembersihan yang konstan, Anda akan berakhir dengan sebagian besar tempat *state management* di bagian atas hirarki komponen.
6. Akhirnya, Anda memutuskan bahwa itu ide yang baik untuk membiarkan komponen di bagian atas menangani *state* secara *global* dan kemudian mengesampingkan semuanya. Setiap komponen lain dapat berlangganan *props* yang mereka butuhkan dan mengabaikan sisanya. (Manjunath, 2018)

3.7 *Localstorage*

Local storage merupakan fungsi di javascript yang cara kerjanya sama seperti *cookie* (data disimpan dalam *browser*) tapi memiliki beberapa kelebihan. *Local storage* dapat menyimpan data lebih aman dan lebih besar dibandingkan dengan *cookie*. *Local storage* juga dapat menyimpan data lebih dari 5 MB tanpa harus membebani performa *browser*. Banyak hal yang bisa dilakukan saat membangun aplikasi web dengan menggunakan fungsi ini dan dapat mempermudah anda dalam membuat atau membangun sebuah *website* yang lebih *responsif* di sisi *client*. (Agus, 2016).



BAB IV

DESKRIPSI PEKERJAAN

4.1 Analisis dan Desain Sistem

Berdasarkan hasil observasi dan wawancara di PT.Sinergi Informatika Semen Indonesia ditemukan permasalahan pada portal berita internal PT. Semen Indonesia. Portal berita tersebut bernama SMILE. Permasalahan yang terjadi pada *website* portal berita SMILE meliputi , tidak *responsive*-nya website portar berita , tidak multi *platform* , *load* data membutuhkan waktu yang lama sehingga pengguna atau user dari pihak civitas PT.Semen Indonesia banyak yang enggak menggunakan portal berita tersebut.

Aplikasi portal berita SMILE berbasis web dan dikombinasikan menggunakan PWA (*Progressive Web App*) , hal itu ditujukan agar aplikasi dapat digunakan di berbagai perangkat. Aplikasi portal berita SMILE ini membuat proses pengumuman terhadap karyawan menjadi lebih efektif dan efisien serta menampilkan berita – berita mengenai hal apa saja yang terjadi di kantor PT. Semen Indonesia terhadap seluruh karyawannya.

Kebutuhan Fungsional :

1. Sistem dapat menentukan Hak akses (*Roles*) secara otomatis antara admin dan *user*.
2. Sistem memungkinkan admin untuk menambah,merubah,menghapus,dan melihat *content* yang ada di aplikasi.
3. Sistem hanya mengizinkan user untuk melihat *content* yang ada.

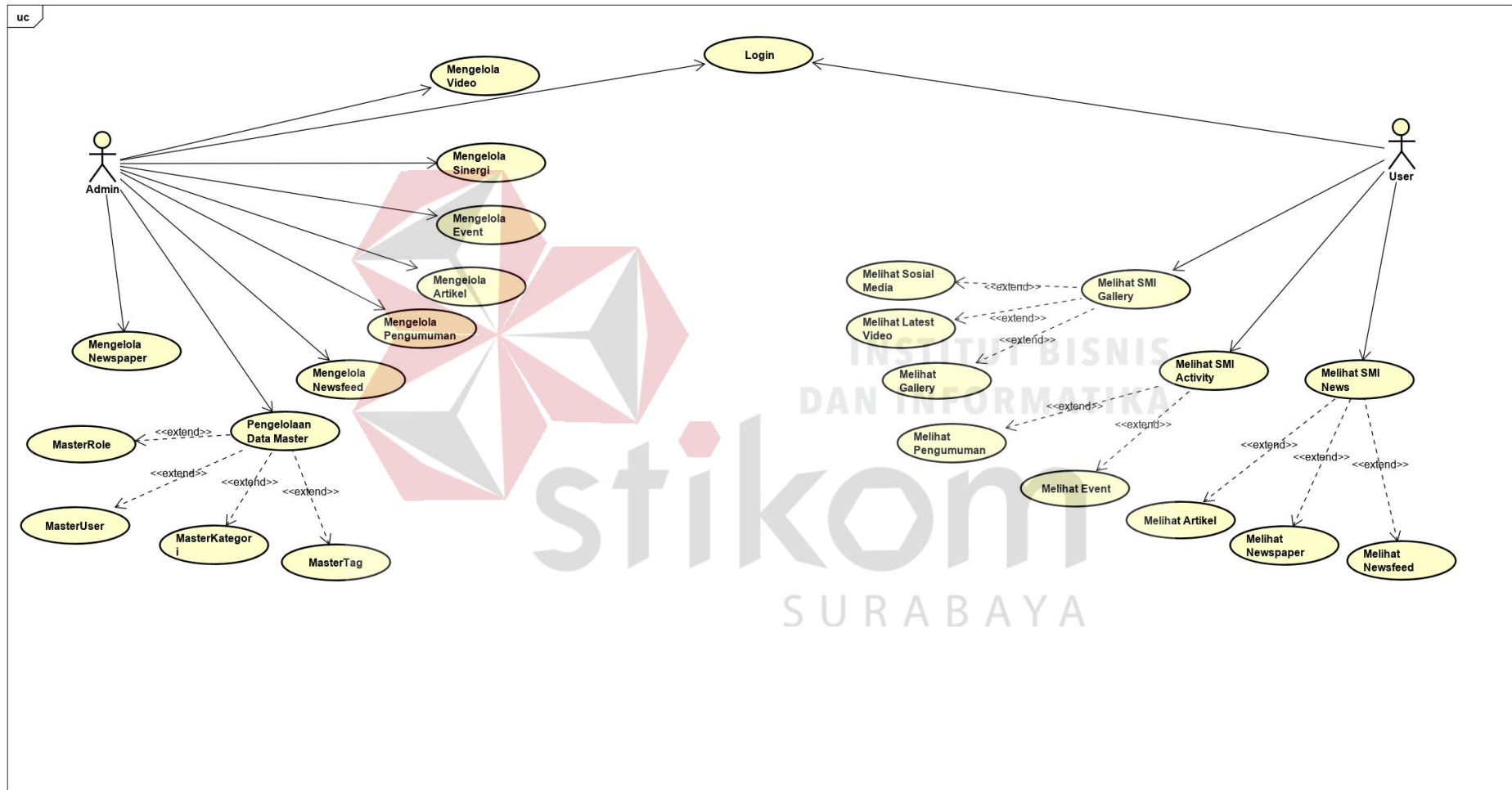
Kebutuhan Non-fungsional :

1. *Frontend system* hanya bisa diakses oleh pegawai semen Indonesia group.
2. Majalah/Koran dapat diunduh secara digital di aplikasi.
3. User dapat memberikan komentar ke artikel yang telah dibuat.

4.2 Perancangan Sistem

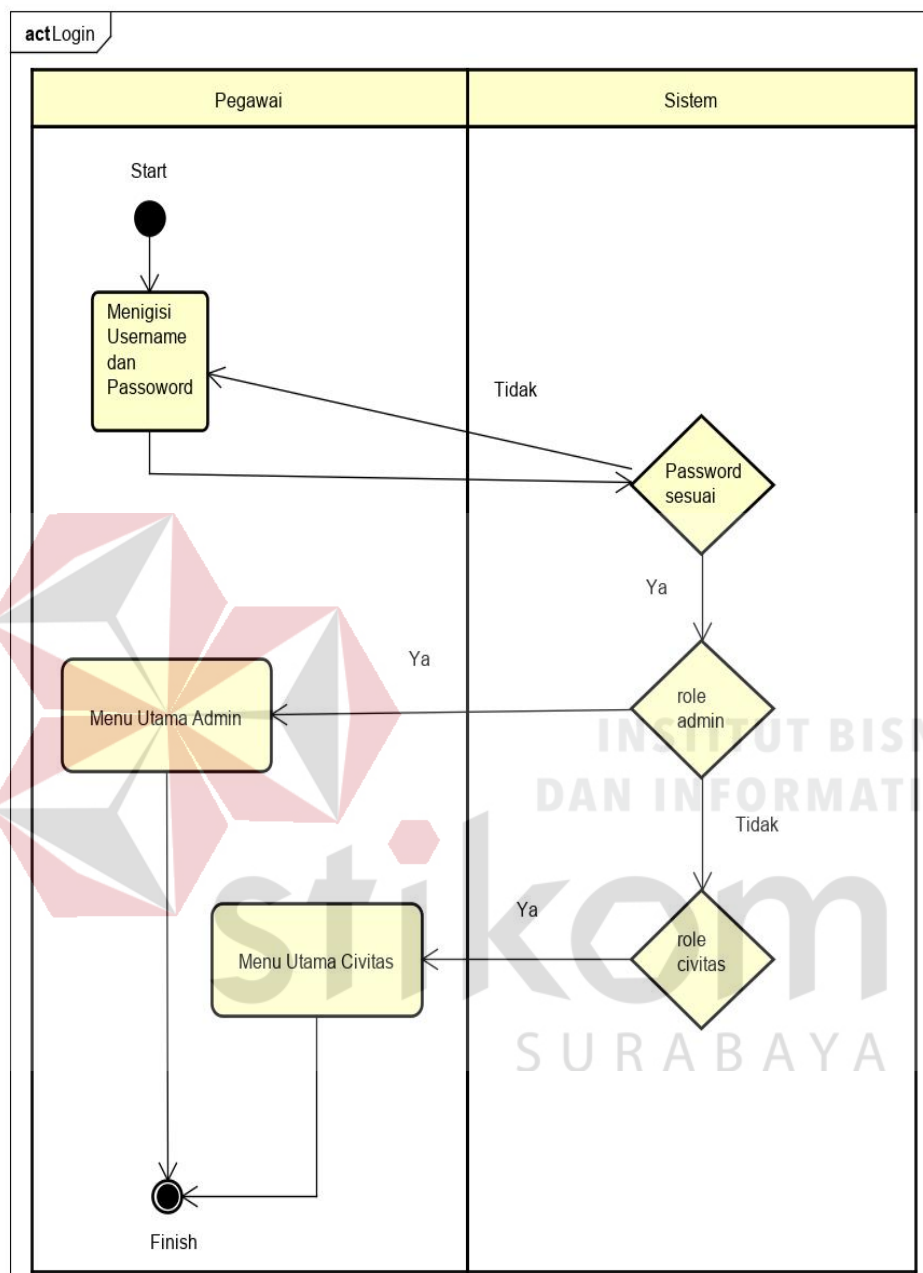
Berikut ini merupakan *use case* Aplikasi portal berita internal SMILE. Pada *use case* ini terdapat 2 aktor yaitu Admin dan *User*. *User* tersebut merupakan pegawai dari PT.Semen Indonesia. Terdapat 23 proses bisnis yang terdapat pada *use case* tersebut. Proses bisnis itu terdiri dari *master* mengelola video, *master* mengelola sinergi, *master* mengelola event, *master* mengelola artikel, *master* mengelola pengumuman, *master* mengelola *newsfeed*, *master* mengelola *newspaper* dan pengelolaan data *master* admin seperti *master role* , *master user*, *master kategori* dan *master tag*.

Hal yang disebutkan tersebut merupakan bagian untuk admin , sedangkan untuk bagian *user* sebagai berikut ini , melihat SMI *gallery*, melihat SMI *activity* dan melihat SMI *news*. Namun tidak berhenti sampai disitu, didalam SMI *gallery* terdapat melihat media social , *latest video*, *gallery*. Pada SMI *activity user* dapat melihat pengumuman dan *event*. Sedangkan untuk SMI *news user* dapat melihat artikel, *newspaper* dan *newsfeed*. Untuk lebih jelasnya hubungan antara *actor* dan proses bisnis dapat dilihat pada Gambar 4.1.



Gambar 4.1 Use Case Diagram Aplikasi

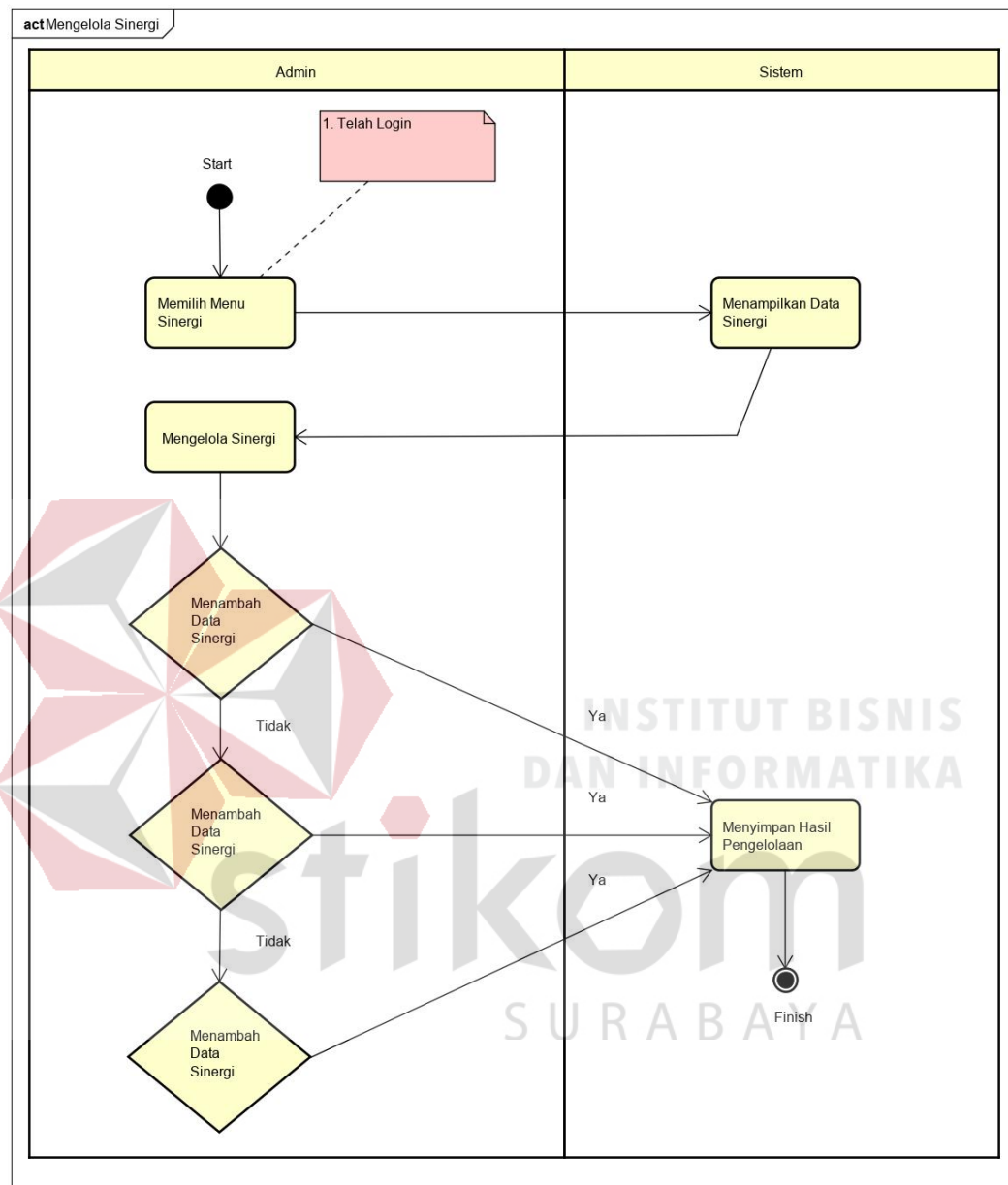
4.2.1 Login



Gambar 4.2 Activity Diagram Login

Gambar 4.2 menggambarkan proses aktivitas *login*. Gambar tersebut menjelaskan bahwa terdapat validasi *username* dan *password* kemudian terdapat pembagian *role* sesuai dengan bagian masing-masing.

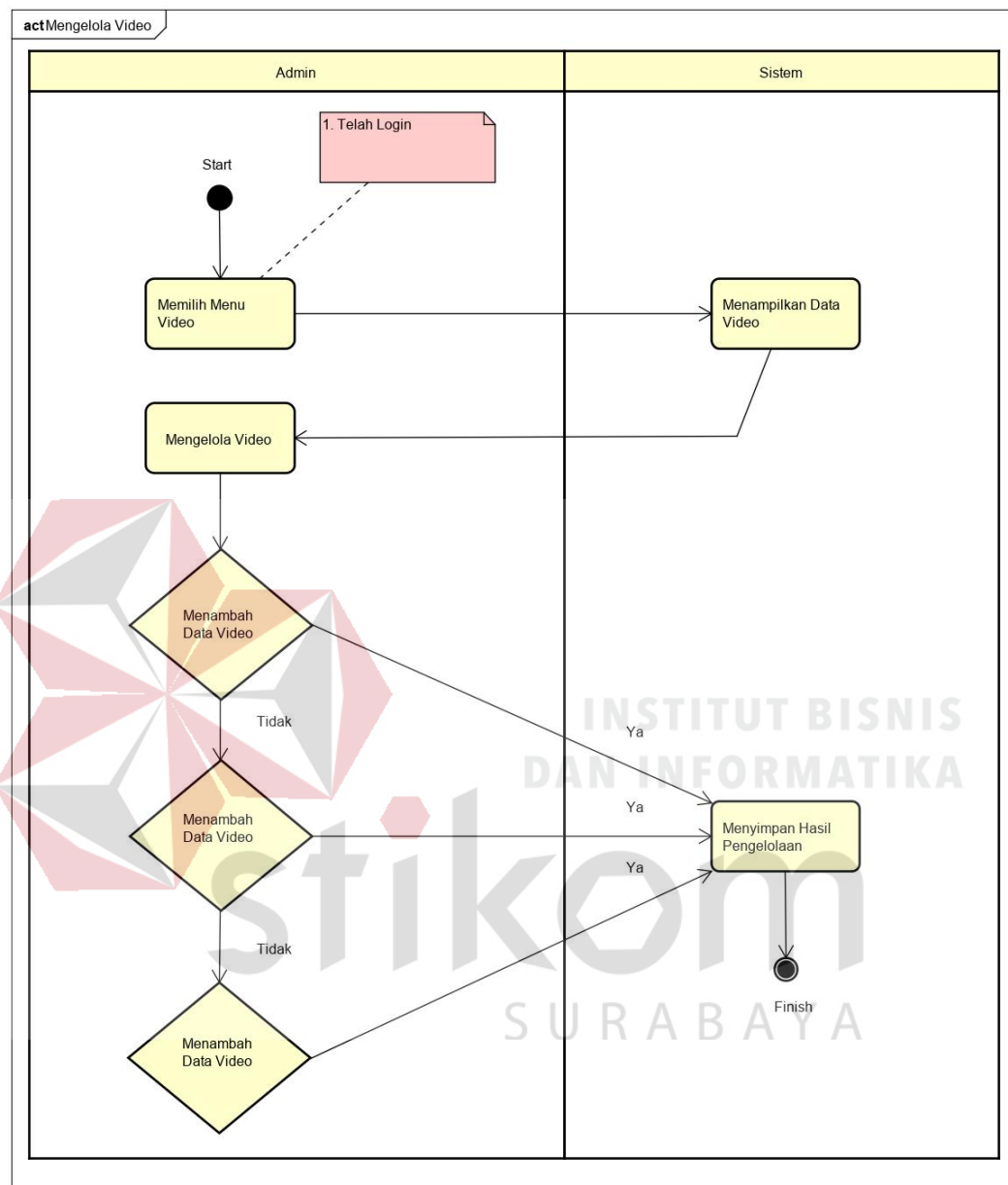
4.2.2 Mengelola Sinergi



Gambar 4.3 *Activity Diagram* Mengelola Sinergi

Gambar 4.3 menggambarkan proses aktivitas mengelola sinergi. Gambar tersebut menjelaskan bahwa sebelum mengelola sinergi harus terlebih dahulu melakukan *login* dari bagian admin. Dalam pengelolaan sinergi dapat dilakukan penambahan, perubahan atau penghapusan data sinergi.

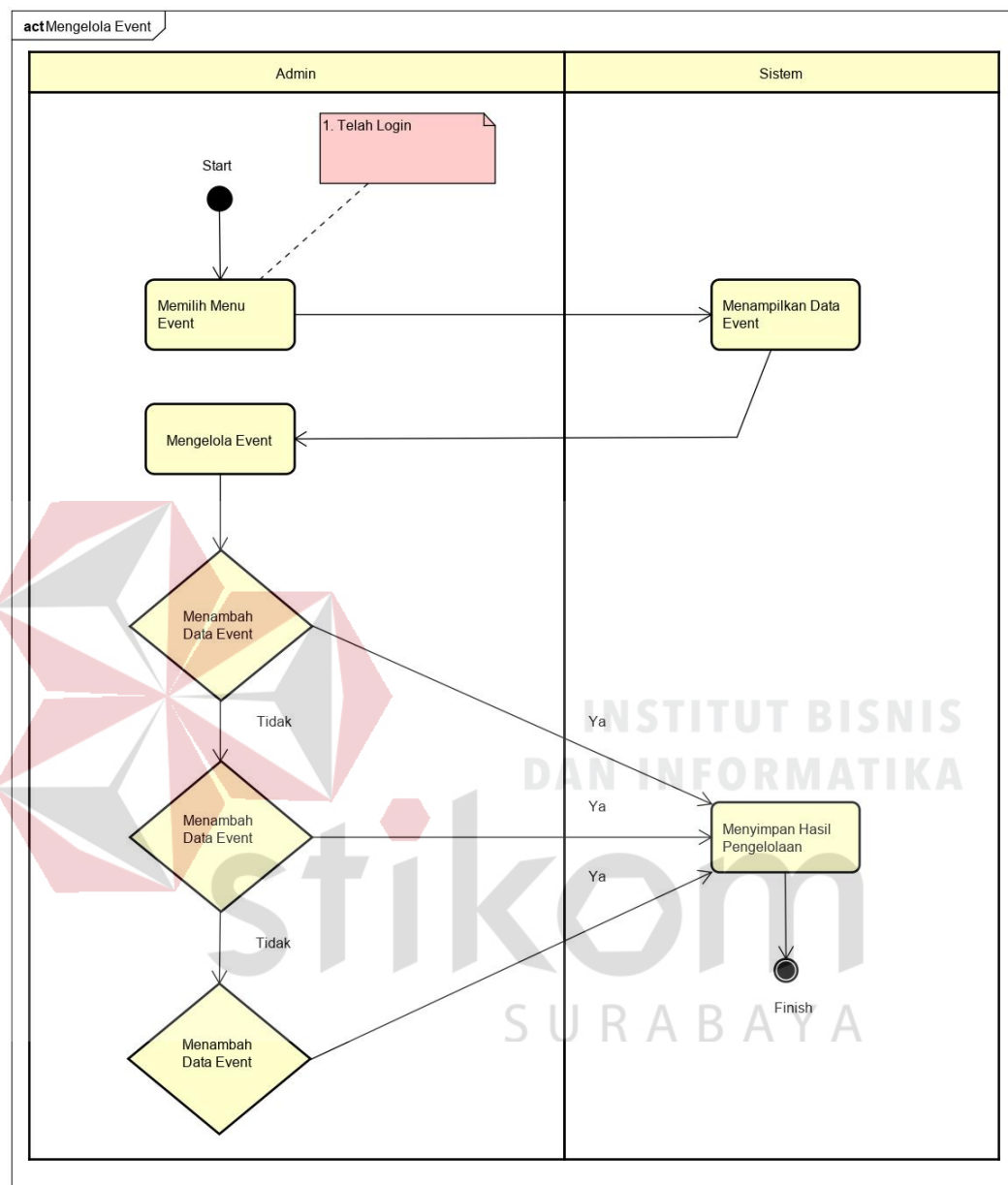
4.2.3 Mengelola Video



Gambar 4.4 Activity Diagram Mengelola Video

Gambar 4.4 menggambarkan proses aktivitas mengelola admin. Gambar tersebut menjelaskan bahwa sebelum mengelola video harus terlebih dahulu melakukan *login* dari bagian Admin. Dalam pengelolaan video dapat dilakukan penambahan, perubahan atau penghapusan data video.

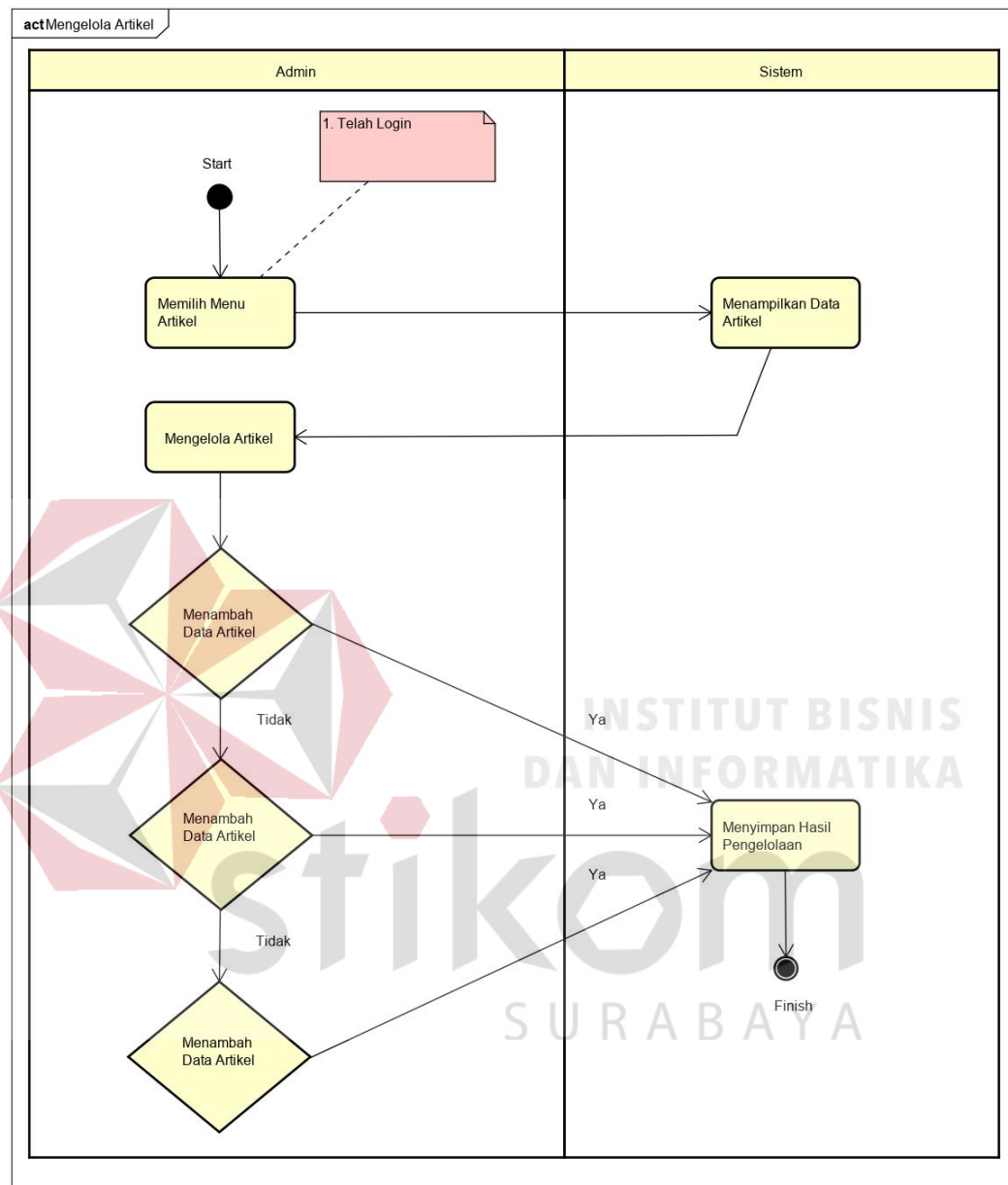
4.2.4 Mengelola *Event*



Gambar 4.5 Activity Diagram Mengelola Event

Gambar 4.5 menggambarkan proses aktivitas mengelola *event*. Gambar tersebut menjelaskan bahwa sebelum mengelola sinergi harus terlebih dahulu melakukan *login* dari bagian admin. Dalam pengelolaan *event* dapat dilakukan penambahan, perubahan atau penghapusan data *event*.

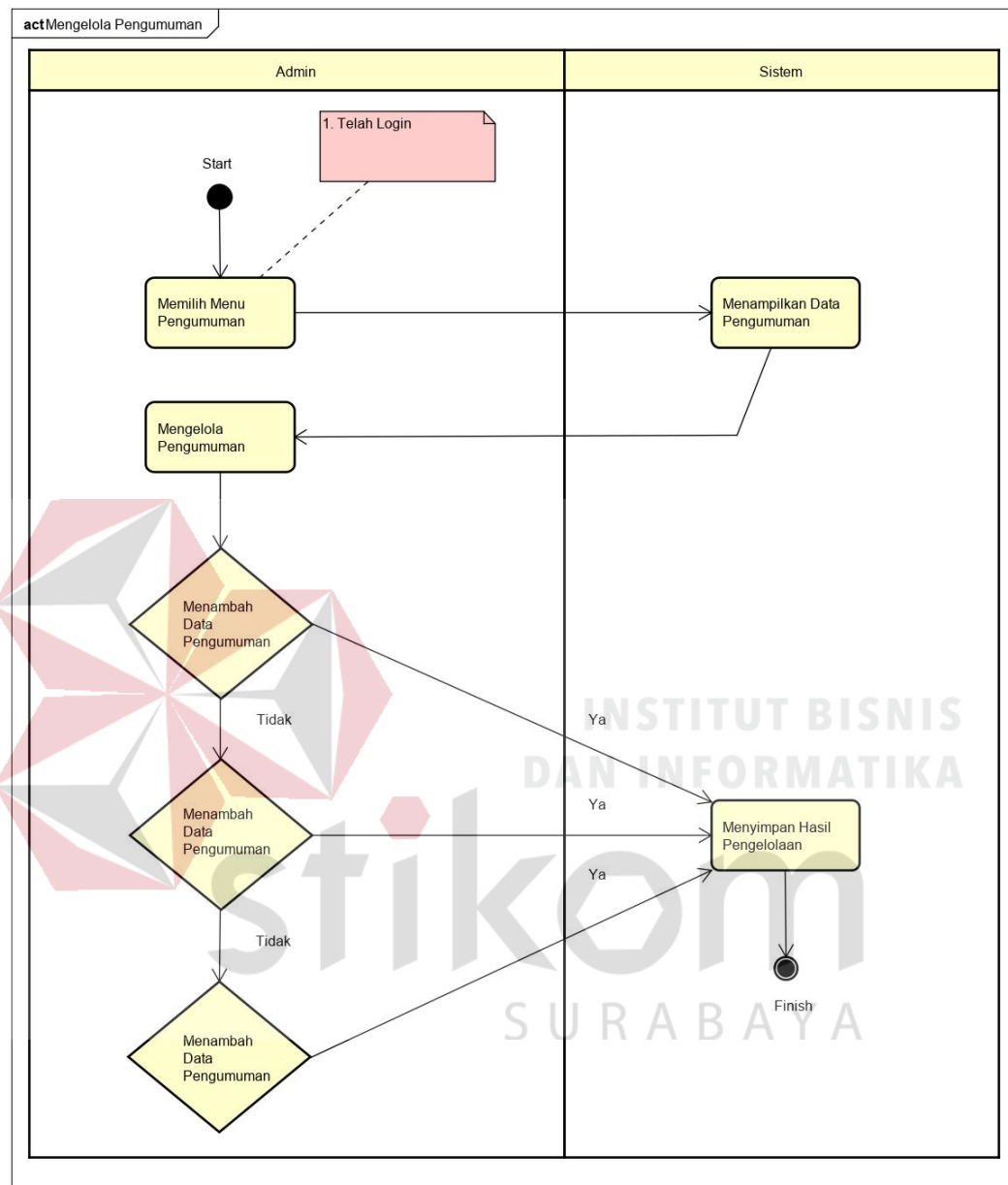
4.2.5 Mengelola Artikel



Gambar 4.6 *Activity Diagram* Mengelola Artikel

Gambar 4.6 menggambarkan proses aktivitas mengelola artikel. Gambar tersebut menjelaskan bahwa sebelum mengelola artikel harus terlebih dahulu melakukan *login* dari bagian Admin. Dalam pengelolaan artikel dapat dilakukan penambahan, perubahan atau penghapusan data artikel

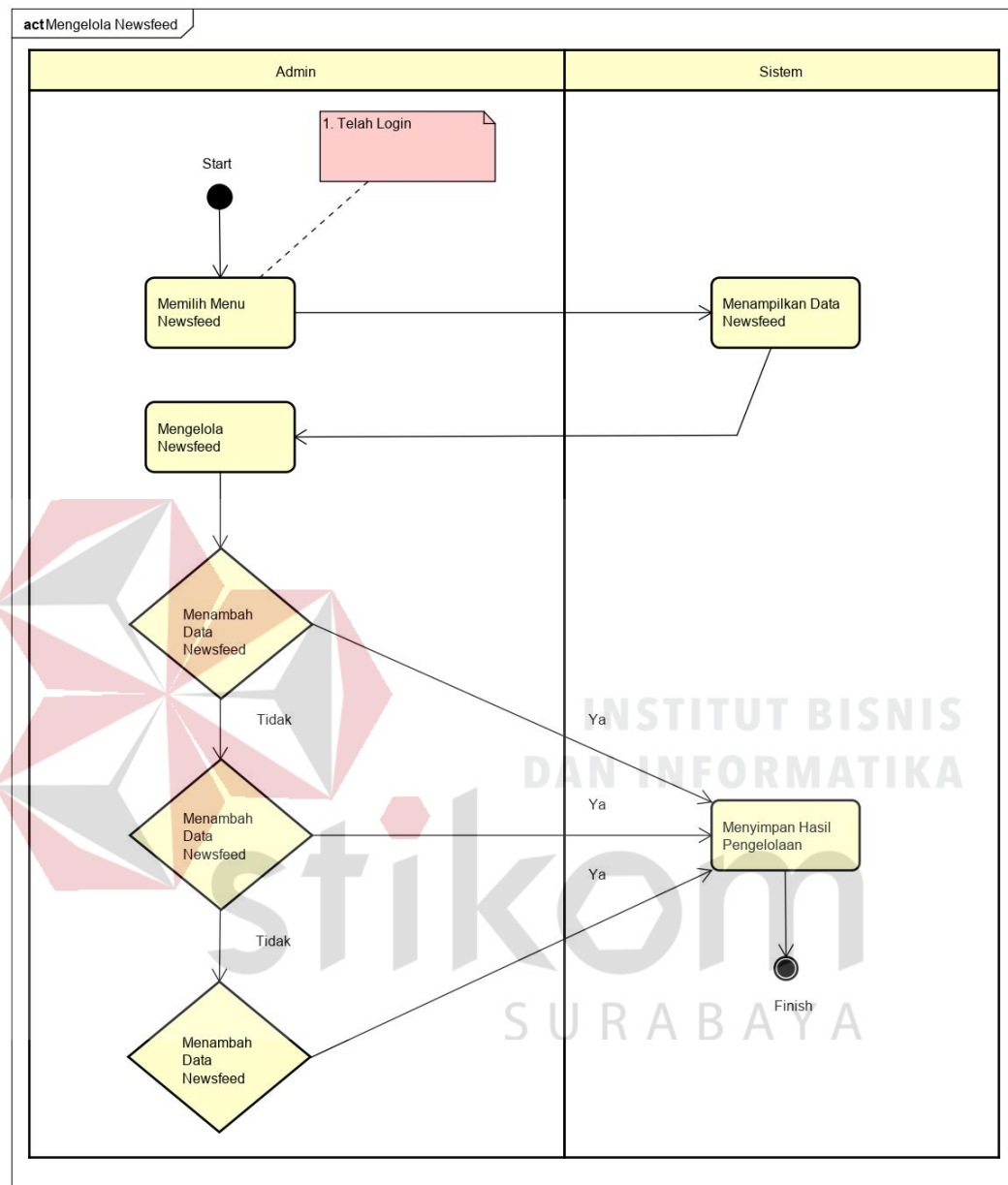
4.2.6 Mengelola Pengumuman



Gambar 4.7 Activity Diagram Mengelola Pengumuman

Gambar 4.7 menggambarkan proses aktivitas mengelola pengumuman. Gambar tersebut menjelaskan bahwa sebelum mengelola pengumuman harus terlebih dahulu melakukan *login* dari bagian admin. Dalam pengelolaan pengumuman dapat dilakukan penambahan, perubahan atau penghapusan data pengumuman.

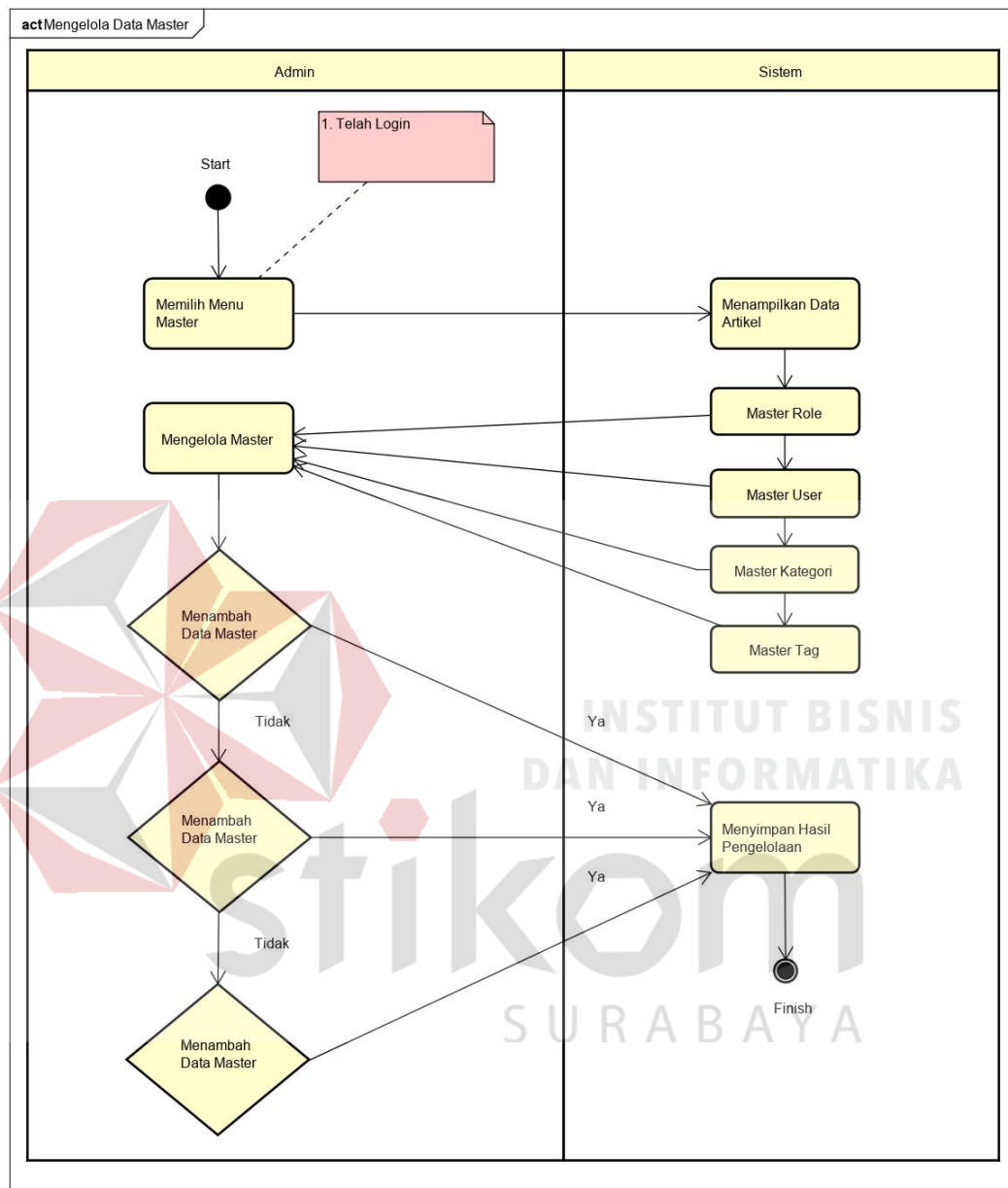
4.2.7 Mengelola *Newsfeed*



Gambar 4.8 *Activity Diagram* Mengelola *Newsfeed*

Gambar 4.8 menggambarkan proses aktivitas mengelola *newsfeed*. Gambar tersebut menjelaskan bahwa sebelum mengelola *newsfeed* harus terlebih dahulu melakukan *login* dari bagian admin. Dalam pengelolaan *newsfeed* dapat dilakukan penambahan, perubahan atau penghapusan data *newsfeed*.

4.2.8 Mengelola Data *Master*

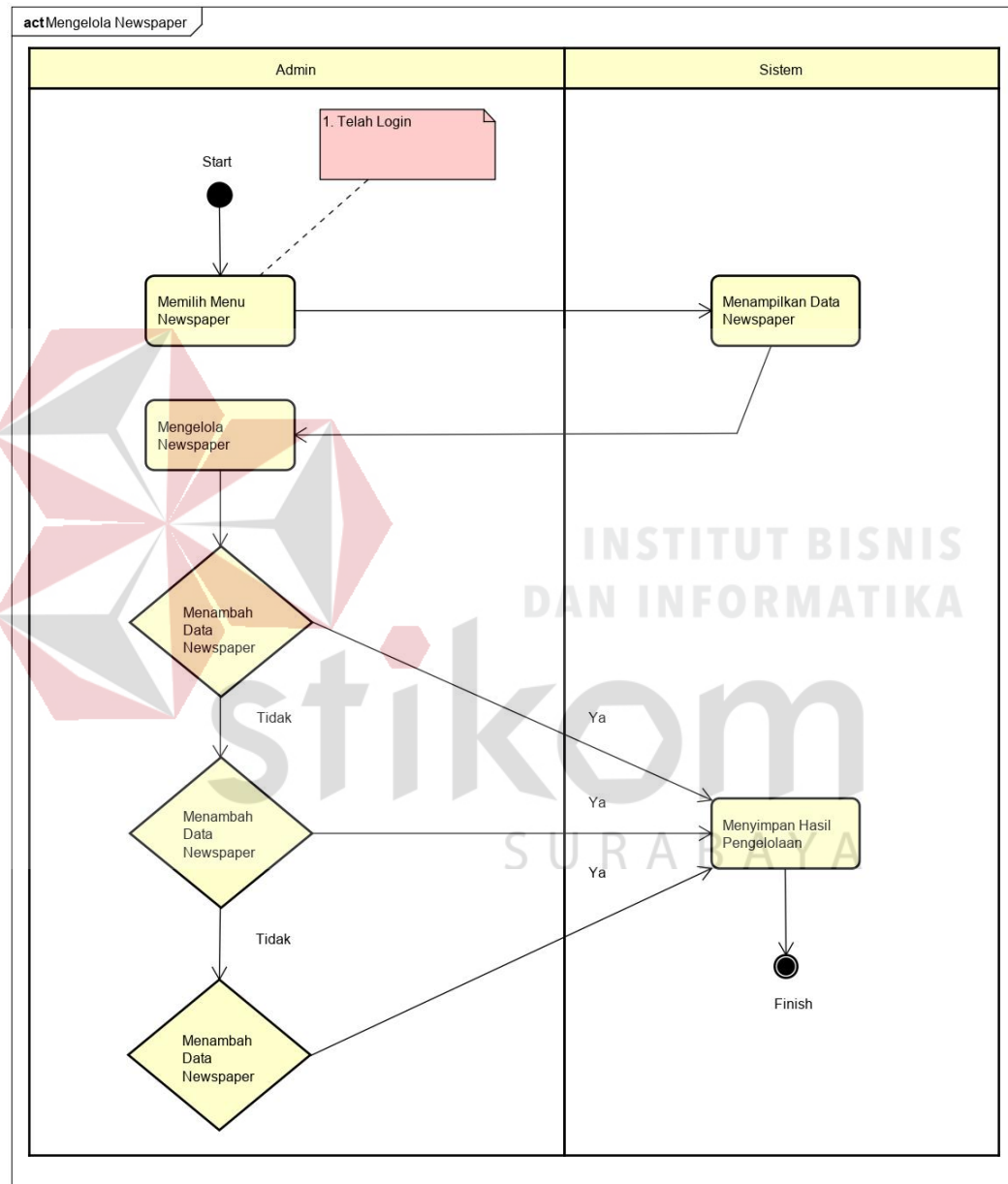


Gambar 4.9 *Activity Diagram Mengelola Data Master*

Gambar 4.9 menggambarkan proses aktivitas mengelola data *master*. Gambar tersebut menjelaskan bahwa sebelum mengelola data *master* harus terlebih dahulu melakukan *login* dari bagian admin. Dalam pengelolaan data *master* dapat dilakukan penambahan, perubahan atau penghapusan data *master*. Dalam

mengelola data *master* , admin mempunyai akses untuk mengelola *master role*, *master user*, *master kategori* dan *master tag* pada aplikasi media internal SMILE.

4.2.9 Mengelola Newspaper



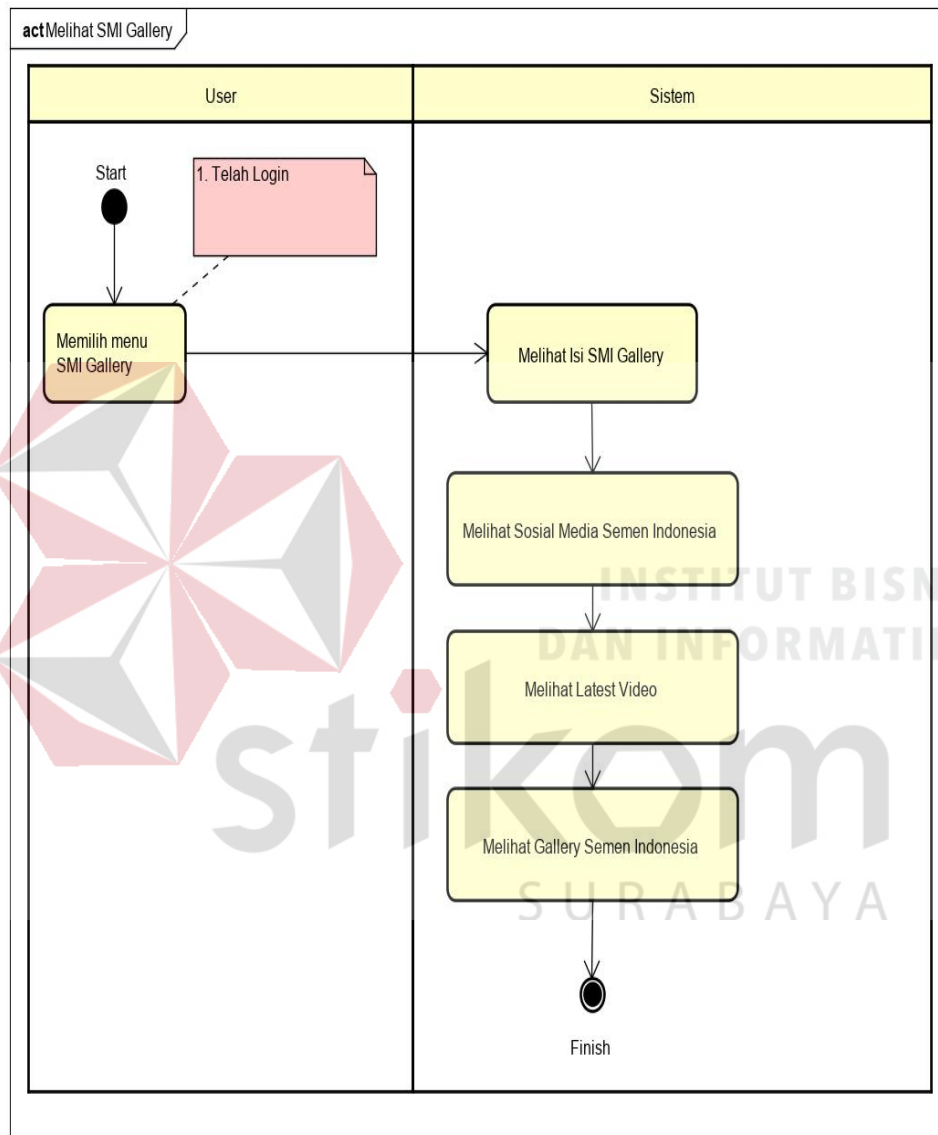
Gambar 4.10 Activity Diagram Mengelola Newspaper

Gambar 4.10 menggambarkan proses aktivitas mengelola *newspaper*.

Gambar tersebut menjelaskan bahwa sebelum mengelola *newspaper* harus terlebih

dahulu melakukan *login* dari bagian admin. Dalam pengelolaan *newspaper* dapat dilakukan penambahan, perubahan atau penghapusan data *newspaper*.

4.2.10 Melihat SMI Gallery

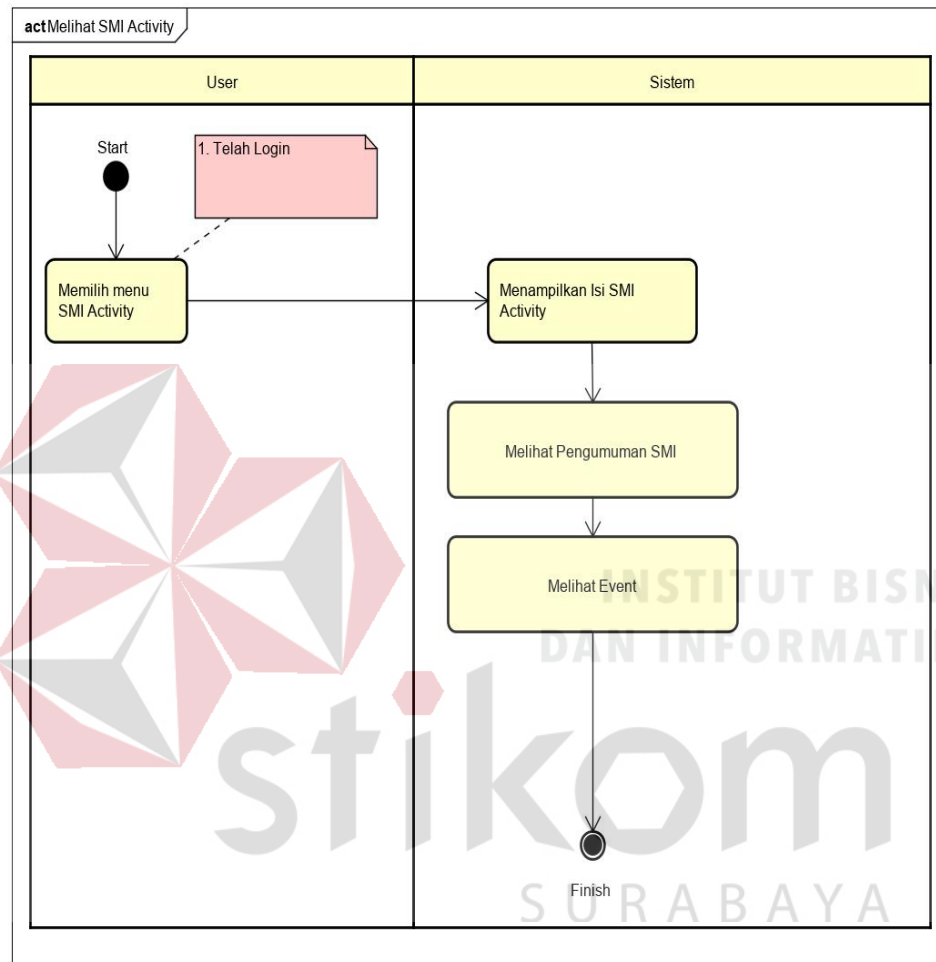


Gambar 4.11 *Activity Diagram* Melihat SMI Gallery

Gambar 4.11 menggambarkan proses aktivitas untuk melihat SMI Gallery. Gambar tersebut menjelaskan bahwa sebelum dapat melihat SMI Gallery harus terlebih dahulu melakukan *login* dari bagian User. Dalam melihat SMI Gallery,

user dapat melihat social media PT. Semen Indonesia, melihat video dan juga dapat melihat gallery dari PT. Semen Indonesia.

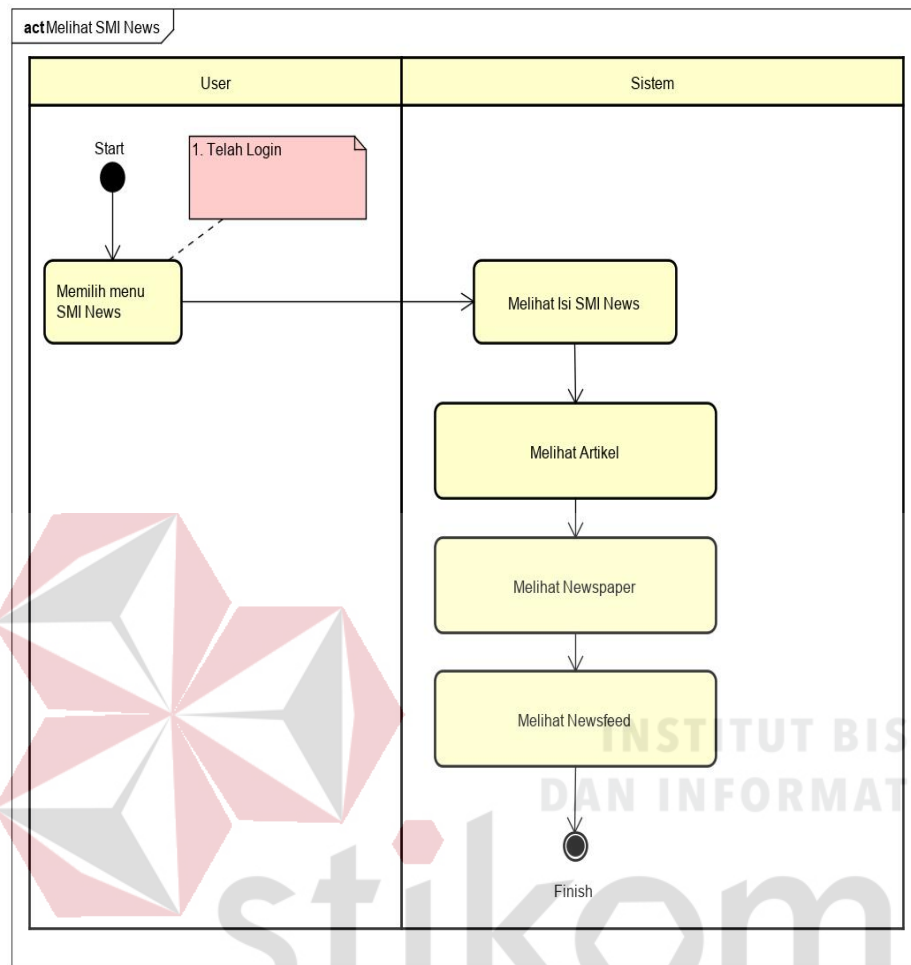
4.2.11 Melihat SMI Activity



Gambar 4.12 Activity Diagram Melihat SMI Activity

Gambar 4.12 menggambarkan proses aktivitas untuk melihat SMI Activity. Gambar tersebut menjelaskan bahwa sebelum dapat melihat SMI Activity harus terlebih dahulu melakukan *login* dari bagian *user*. Dalam bagian SMI Activity, *user* dapat melihat pengumuman yang ditujukan untuk karyawan PT. Semen Indonesia dan dapat melihat *event* yang sedang ada pada perusahaan.

4.2.12 Melihat SMI News



Gambar 4.13 Activity Diagram Melihat SMI News

Gambar 4.13 menggambarkan proses aktivitas untuk melihat SMI News. Gambar tersebut menjelaskan bahwa sebelum dapat melihat SMI News harus terlebih dahulu melakukan *login* dari bagian *user*. Dalam bagian SMI News, *user* dapat melihat artikel yang pada SMILE, lalu *user* juga dapat melihat *newspaper* yang disediakan oleh PT. Semen Indonesia dalam bentuk pdf dan juga dapat melihat *newsfeed*.

4.3 *Flow Of Event*

Berikut merupakan *flow of event* dari aplikasi media informasi internal (SMILE) pada PT. Sinergi Informatika Semen Indonesia yang akan dibuat.

4.3.1 *Flow Of Event Login*

Tabel 1.1 *Flow of Event Login*

Deskripsi	Login awal untuk masuk ke panel admin	
Kondisi Awal	Pegawai sudah memiliki akun yang terdaftar	
Kondisi Akhir	Pegawai dapat masuk sesuai <i>role</i> nya	
Aliran Kejadian Utama		
No.	User	Sistem
1.	User memasukkan <i>username</i> dan <i>password</i> yang telah dimiliki	Sistem melakukan validasi <i>username</i> dan <i>password</i> , jika benar sistem akan menampilkan halaman utama dari aplikasi sesuai <i>role</i> yang dimiliki. Jika salah akan kembali ke halaman <i>login</i> dengan <i>notifikasi username</i> atau <i>password</i> salah

Tabel 1.1 *Flow of event login* menjelaskan bagaimana alur *login user* berdasarkan *role* yang dimiliki pada aplikasi media informasi internal (SMILE).

4.3.2 Flow Of Event Mengelola Video

Tabel 1.2 *Flow of Event* Mengelola Video

Deskripsi	Use case untuk mengelola video	
Kondisi Awal	Pegawai sudah login aplikasi	
Kondisi Akhir	Pegawai dapat mengelola video	
Aliran Kejadian Utama		
No.	User	Sistem
1.	User memilih menu video pada menu aplikasi	Menampilkan halaman video
2.	User memilih pengelolaan yang akan dilakukan	1. User memilih Entry data, sistem akan menampilkan form entry video. 2. User memilih edit data, sistem akan menampilkan form yang berisi data yang sudah ada. 3. User memilih lihat, sistem akan menampilkan data detail video 4. User memilih delete, sistem akan menghapus data video yang dipilih pada aplikasi

Tabel 1.2 *Flow of event* mengelola video menjelaskan bagaimana alur proses pengelolaan data video pada aplikasi media informasi internal (SMILE).

4.3.3 Flow Of Event Mengelola Sinergi

Tabel 1.3 *Flow of Event* Mengelola Sinergi

Deskripsi	Use case untuk mengelola sinergi	
Kondisi Awal	Pegawai sudah login aplikasi	
Kondisi Akhir	Pegawai dapat mengelola sinergi	
Aliran Kejadian Utama		
No.	User	Sistem
1.	User memilih menu sinergi pada menu aplikasi	Menampilkan halaman sinergi
2.	User memilih pengelolaan yang akan dilakukan	1. User memilih Entry data, sistem akan menampilkan form entry sinergi. 2. User memilih edit data, sistem akan menampilkan form yang berisi data yang sudah ada. 3. User memilih lihat, sistem akan menampilkan data detail sinergi 4. User memilih delete, sistem akan menghapus data sinergi yang dipilih pada aplikasi

Tabel 1.3 *Flow of event* mengelola video menjelaskan bagaimana alur proses pengelolaan data video pada aplikasi media informasi internal (SMILE).

4.3.4 Flow Of Event Mengelola Event

Tabel 1.4 *Flow of Event* Mengelola Event

Deskripsi		<i>Use case</i> untuk mengelola sinergi
Kondisi Awal		Pegawai sudah <i>login</i> aplikasi
Kondisi Akhir		Pegawai dapat mengelola sinergi
Aliran Kejadian Utama		
No.	User	Sistem
1.	User memilih menu sinergi pada menu aplikasi	Menampilkan halaman sinergi
2.	User memilih pengelolaan yang akan dilakukan	5. User memilih <i>Entry</i> data, sistem akan menampilkan <i>form entry</i> sinergi. 6. User memilih edit data, sistem akan menampilkan <i>form</i> yang berisi data yang sudah ada. 7. User memilih lihat, sistem akan menampilkan data detail sinergi 8. User memilih <i>delete</i> , sistem akan menghapus data sinergi yang dipilih pada aplikasi

Tabel 1.4 *Flow of event* mengelola event menjelaskan bagaimana alur proses pengelolaan data event pada aplikasi media informasi internal (SMILE).

4.3.5 Flow Of Event Mengelola Artikel

Tabel 1.5 *Flow of Event* Mengelola Artikel

Deskripsi	Use case untuk mengelola artikel	
Kondisi Awal	Pegawai sudah login aplikasi	
Kondisi Akhir	Pegawai dapat mengelola artikel	
Aliran Kejadian Utama		
No.	User	Sistem
1.	User memilih menu artikel pada menu aplikasi	Menampilkan halaman artikel
2.	User memilih pengelolaan yang akan dilakukan	1. User memilih Entry data, sistem akan menampilkan form entry artikel. 2. User memilih edit data, sistem akan menampilkan form yang berisi data yang sudah ada. 3. User memilih lihat, sistem akan menampilkan data detail artikel 4. User memilih delete, sistem akan menghapus data artikel yang dipilih pada aplikasi

Tabel 1.5 *Flow of event* mengelola artikel menjelaskan bagaimana alur proses pengelolaan data artikel pada aplikasi media informasi internal (SMILE).

4.3.6 Flow Of Event Mengelola Pengumuman

Tabel 1.6 *Flow of Event* Mengelola Pengumuman

Deskripsi		<i>Use case</i> untuk mengelola pengumuman
Kondisi Awal		Pegawai sudah <i>login</i> aplikasi
Kondisi Akhir		Pegawai dapat mengelola pengumuman
Aliran Kejadian Utama		
No.	User	Sistem
1.	User memilih menu pengumuman pada menu aplikasi	Menampilkan halaman pengumuman
2.	User memilih pengelolaan yang akan dilakukan	1. User memilih <i>Entry</i> data, sistem akan menampilkan <i>form entry</i> pengumuman. 2. User memilih <i>edit</i> data, sistem akan menampilkan <i>form</i> yang berisi data yang sudah ada. 3. User memilih lihat, sistem akan menampilkan data detail pengumuman 4. User memilih <i>delete</i>, sistem akan menghapus data pengumuman yang dipilih pada aplikasi

Tabel 1.6 *Flow of event* mengelola pengumuman menjelaskan bagaimana alur proses pengelolaan data pengumuman pada aplikasi media informasi internal (SMILE).

4.3.7 Flow Of Event Mengelola Newsfeed

Tabel 1.7 Flow of Event Mengelola Newsfeed

Deskripsi	<i>Use case</i> untuk mengelola <i>newsfeed</i>	
Kondisi Awal	Pegawai sudah <i>login</i> aplikasi	
Kondisi Akhir	Pegawai dapat mengelola <i>newsfeed</i>	
Aliran Kejadian Utama		
No.	User	Sistem
1.	<i>User</i> memilih menu <i>newsfeed</i> pada menu aplikasi	Menampilkan halaman <i>newsfeed</i>
2.	<i>User</i> memilih pengelolaan yang akan dilakukan	1. <i>User</i> memilih <i>Entry</i> data, sistem akan menampilkan <i>form entry newsfeed</i>. 2. <i>User</i> memilih <i>edit</i> data, sistem akan menampilkan <i>form</i> yang berisi data yang sudah ada. 3. <i>User</i> memilih lihat, sistem akan menampilkan data detail <i>newsfeed</i> 4. <i>User</i> memilih <i>delete</i>, sistem akan menghapus data <i>newsfeed</i> yang dipilih pada aplikasi

Tabel 1.7 Flow of event mengelola *newsfeed* menjelaskan bagaimana alur proses pengelolaan data *newsfeed* pada aplikasi media informasi internal (SMILE).

4.3.8 Flow Of Event Mengelola Data Master

Tabel 1.8 *Flow of Event* Mengelola Data Master

Deskripsi	Use case untuk mengelola data <i>master</i>	
Kondisi Awal	Pegawai sudah <i>login</i> aplikasi	
Kondisi Akhir	Pegawai dapat mengelola data <i>master</i>	
Aliran Kejadian Utama		
No.	User	Sistem
1.	<i>User</i> memilih menu data <i>master</i> pada menu aplikasi	Menampilkan halaman data master
2.	<i>User</i> memilih pengelolaan yang akan dilakukan	1. <i>User</i> memilih <i>Entry</i> data, sistem akan menampilkan <i>form entry</i> data <i>master</i>. 2. <i>User</i> memilih <i>edit</i> data, sistem akan menampilkan <i>form</i> yang berisi data yang sudah ada. 3. <i>User</i> memilih lihat, sistem akan menampilkan data detail data <i>master</i> 4. <i>User</i> memilih <i>delete</i>, sistem akan menghapus data <i>master</i> yang dipilih pada aplikasi

Tabel 1.8 *Flow of event* mengelola *master* menjelaskan bagaimana alur proses pengelolaan data *user* dan *role user* pada aplikasi media informasi internal (SMILE).

4.3.9 Flow Of Event Mengelola Newspaper

Tabel 1.9 Flow of Event Mengelola Newspaper

Deskripsi		<i>Use case</i> untuk mengelola <i>newspaper</i>
Kondisi Awal		Pegawai sudah <i>login</i> aplikasi
Kondisi Akhir		Pegawai dapat mengelola <i>newspaper</i>
Aliran Kejadian Utama		
No.	User	Sistem
1.	<i>User</i> memilih menu <i>newspaper</i> pada menu aplikasi	Menampilkan halaman data <i>newspaper</i>
2.	<i>User</i> memilih pengelolaan yang akan dilakukan	1. <i>User</i> memilih <i>Entry</i> data, sistem akan menampilkan <i>form entry newspaper</i>. 2. <i>User</i> memilih <i>edit</i> data, sistem akan menampilkan <i>form</i> yang berisi data yang sudah ada. 3. <i>User</i> memilih lihat, sistem akan menampilkan data detail <i>newspaper</i> 4. <i>User</i> memilih <i>delete</i>, sistem akan menghapus data <i>newspaper</i> yang dipilih pada aplikasi

Tabel 1.9 Flow of event mengelola *newspaper* menjelaskan bagaimana alur proses pengelolaan data *newspaper* pada aplikasi media informasi internal (SMILE).

4.3.10 *Flow of Event* Melihat Sosial Media

Tabel 1.10 *Flow of Event* Melihat Sosial Media

Deskripsi		<i>Use case</i> untuk melihat sosial media
Kondisi Awal		Pegawai sudah <i>login</i> aplikasi
Kondisi Akhir		Pegawai dapat melihat sosial media
Aliran Kejadian Utama		
No.	User	Sistem
1.	<i>User</i> memilih menu sosial media pada menu aplikasi	Menampilkan halaman media sosial

Tabel 1.10 *Flow of event* melihat sosial media menjelaskan bagaimana alur proses *user* melihat media sosial video pada aplikasi media informasi internal (SMILE).

4.3.11 *Flow of Event* Melihat Gallery

Tabel 1.11 *Flow of Event* Melihat Gallery

Deskripsi		<i>Use case</i> untuk melihat <i>gallery</i>
Kondisi Awal		Pegawai sudah <i>login</i> aplikasi
Kondisi Akhir		Pegawai dapat melihat <i>gallery</i>
Aliran Kejadian Utama		
No.	User	Sistem
1.	<i>User</i> memilih menu <i>gallery</i> pada menu aplikasi	Menampilkan halaman <i>gallery</i>

Tabel 1.11 *Flow of event* melihat *gallery* menjelaskan bagaimana alur proses *user* melihat *list gallery* pada aplikasi media informasi internal (SMILE).

4.3.12 *Flow of Event* Melihat Pengumuman

Tabel 1.12 *Flow of Event* Melihat Pngumuman

Deskripsi		<i>Use case</i> untuk melihat pengumuman
Kondisi Awal		Pegawai sudah <i>login</i> aplikasi
Kondisi Akhir		Pegawai dapat melihat pengumuman
Aliran Kejadian Utama		
No.	User	Sistem
1.	<i>User</i> memilih menu pengumuman pada menu aplikasi	Menampilkan halaman pengumuman

Tabel 1.12 *Flow of event* melihat pengumuman video menjelaskan bagaimana alur proses *user* melihat pengumuman pada aplikasi media informasi internal (SMILE).

4.3.13 *Flow of Event* Melihat Event

Tabel 1.13 *Flow of Event* Melihat Event

Deskripsi		<i>Use case</i> untuk melihat event
Kondisi Awal		Pegawai sudah <i>login</i> aplikasi
Kondisi Akhir		Pegawai dapat melihat event
Aliran Kejadian Utama		
No.	User	Sistem
1.	<i>User</i> memilih menu event pada menu aplikasi	Menampilkan halaman event

Tabel 1.13 *Flow of event* melihat event menjelaskan bagaimana alur proses *user* melihat *list event* pada aplikasi media informasi internal (SMILE).

4.3.14 *Flow of Event* Melihat Artikel

Tabel 1.14 *Flow of Event* melihat Artikel

Deskripsi	Use case untuk melihat artikel	
Kondisi Awal	Pegawai sudah login aplikasi	
Kondisi Akhir	Pegawai dapat melihat artikel	
Aliran Kejadian Utama		
No.	User	Sistem
1.	User memilih menu artikel pada menu aplikasi	Menampilkan halaman artikel

Tabel 1.14 *Flow of event* melihat artikel menjelaskan bagaimana alur proses user melihat artikel pada aplikasi media informasi internal (SMILE).

4.3.15 *Flow of Event* Melihat Newspaper

Tabel 1.15 *Flow of Event* Melihat Newspaper

Deskripsi	Use <i>case</i> untuk melihat <i>newspaper</i>	
Kondisi Awal	Pegawai sudah <i>login</i> aplikasi	
Kondisi Akhir	Pegawai dapat melihat <i>newspaper</i>	
Aliran Kejadian Utama		
No.	User	Sistem
1.	<i>User</i> memilih menu <i>newspaper</i> pada menu aplikasi	Menampilkan halaman <i>newspaper</i>

Tabel 1.15 *Flow of event* mengelola video menjelaskan bagaimana alur proses pengelolaan data video pada aplikasi media informasi internal (SMILE).

4.3.16 Flow of Event Melihat Newsfeed

Tabel 1.16 Flow of Event Melihat Newsfeed

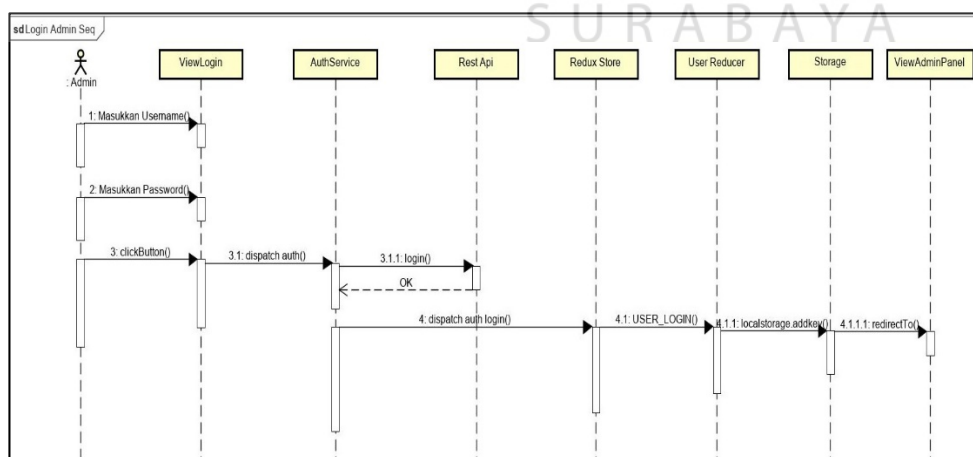
Deskripsi	Use case untuk melihat <i>newsfeed</i>	
Kondisi Awal	Pegawai sudah <i>login</i> aplikasi	
Kondisi Akhir	Pegawai dapat melihat <i>newsfeed</i>	
Aliran Kejadian Utama		
No.	User	Sistem
1.	<i>User</i> memilih menu <i>newsfeed</i> pada menu aplikasi	Menampilkan halaman <i>newsfeed</i>

Tabel 1.16 Flow of event melihat newsfeed menjelaskan bagaimana alur proses user melihat newsfeed pada aplikasi media informasi internal (SMILE).

4.4 Sequence Diagram

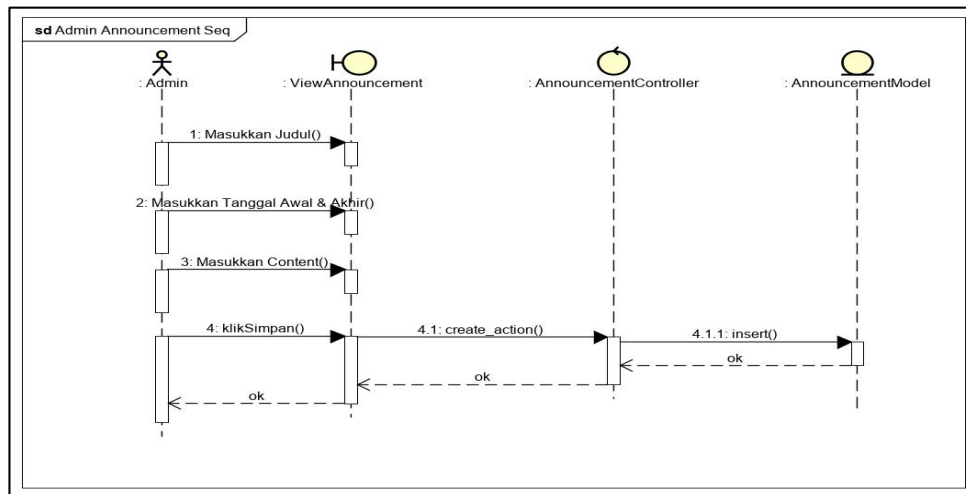
Berikut merupakan Sequence Diagram dari aplikasi media informasi internal (SMILE) pada PT. Sinergi Informatika Semen Indonesia yang akan dibuat.

4.4.1 Login Admin



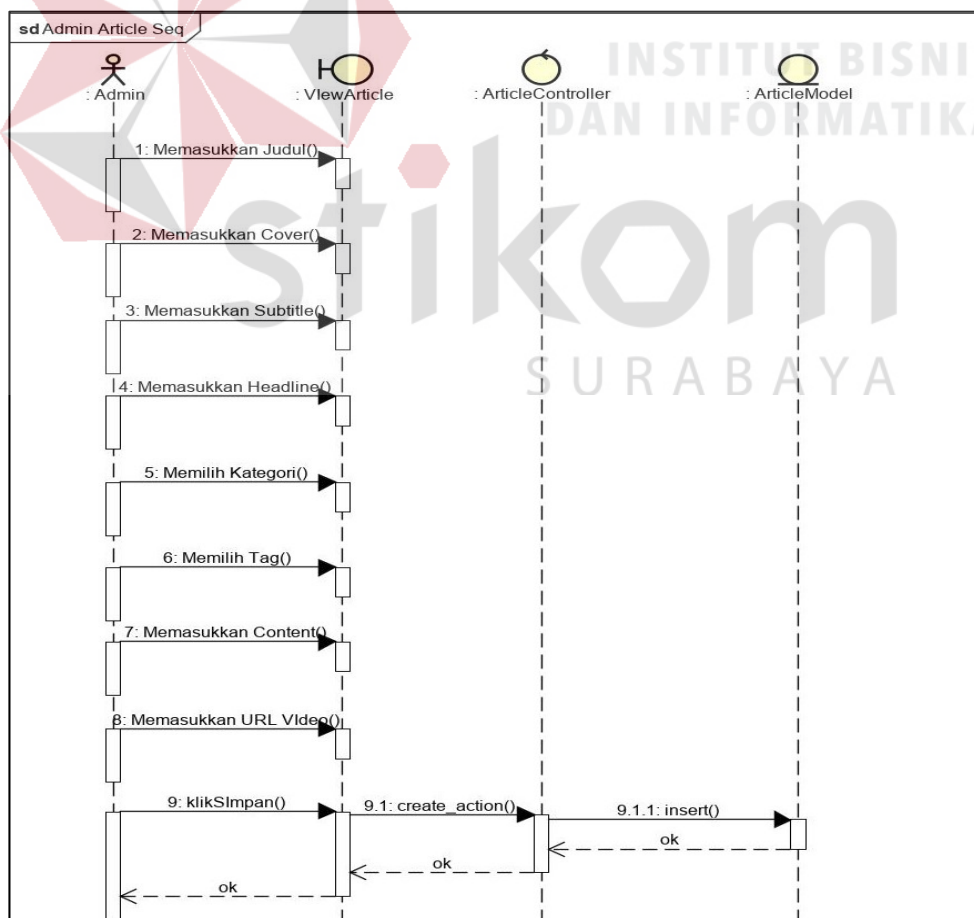
Gambar 4.14 Squence Login Admin

4.4.2 Admin Announcement



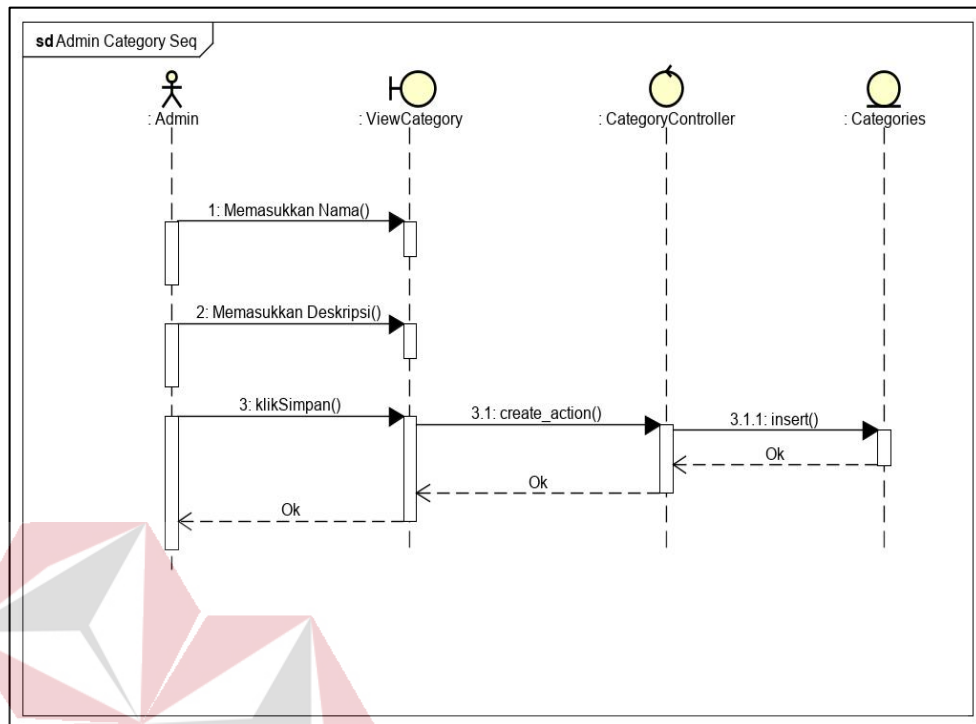
Gambar 4.15 Squence Announcement

4.4.3 Admin Article



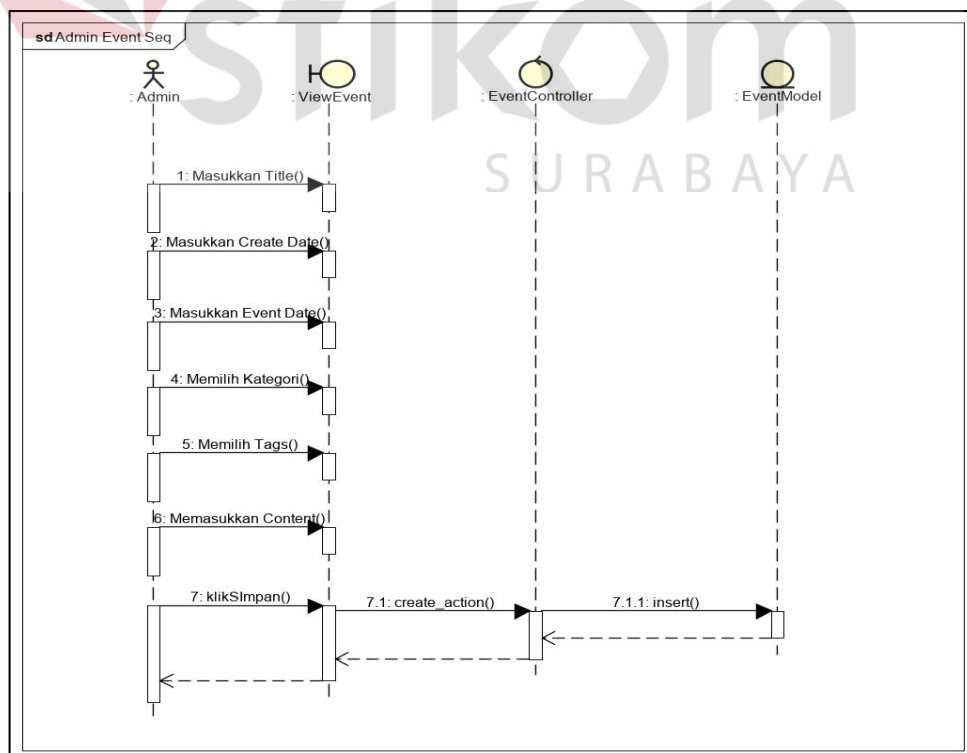
Gambar 4.16 Squence Admin Article

4.4.4 Admin Category



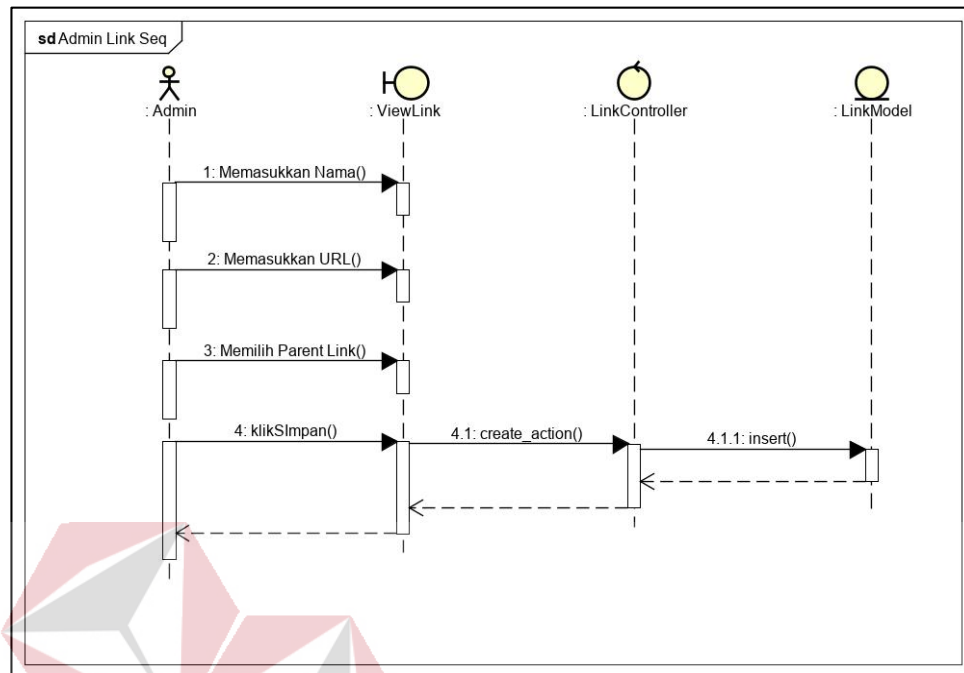
Gambar 4.17 Sequence Admin Category

4.4.5 Admin Event



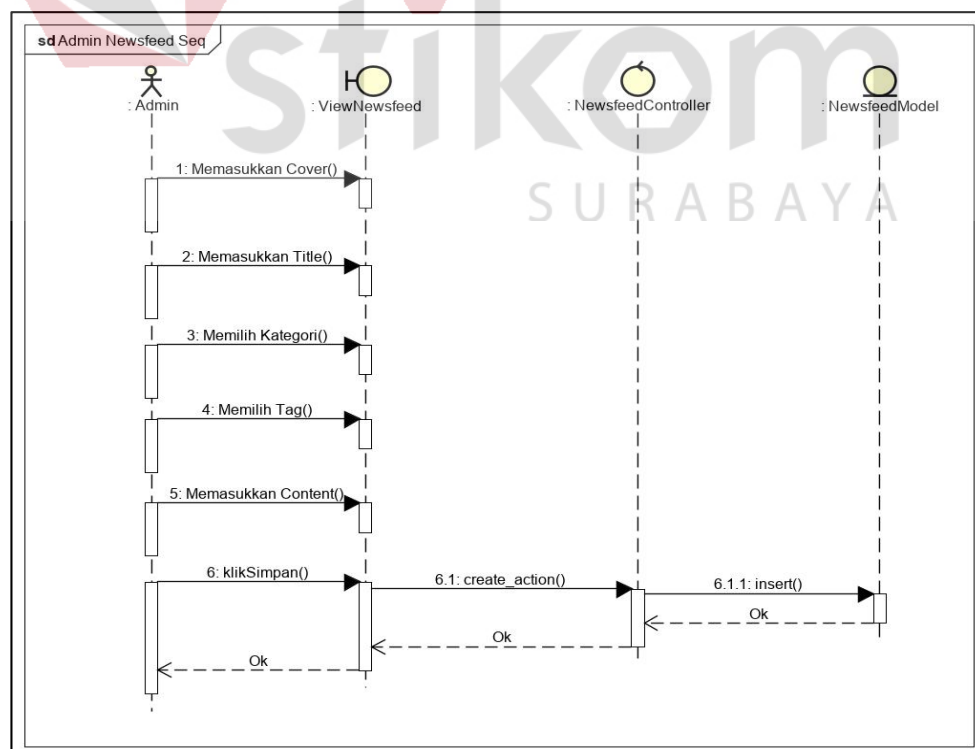
Gambar 4.18 Sequence Admin Event

4.4.6 Admin Link



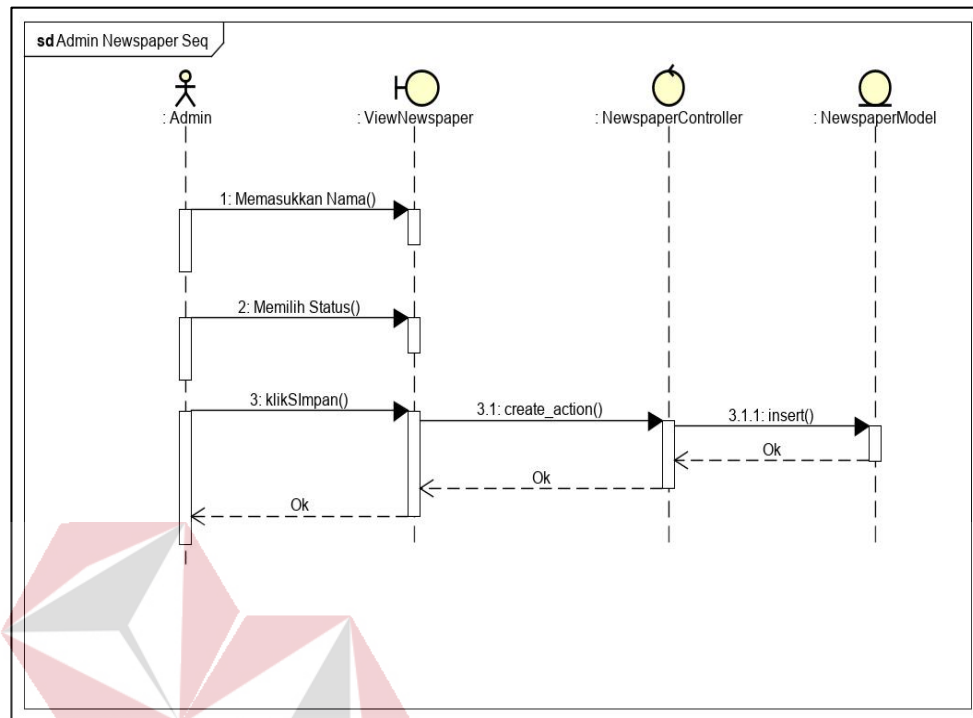
Gambar 4.19 Sequence Admin Link

4.4.7 Admin Newsfeed



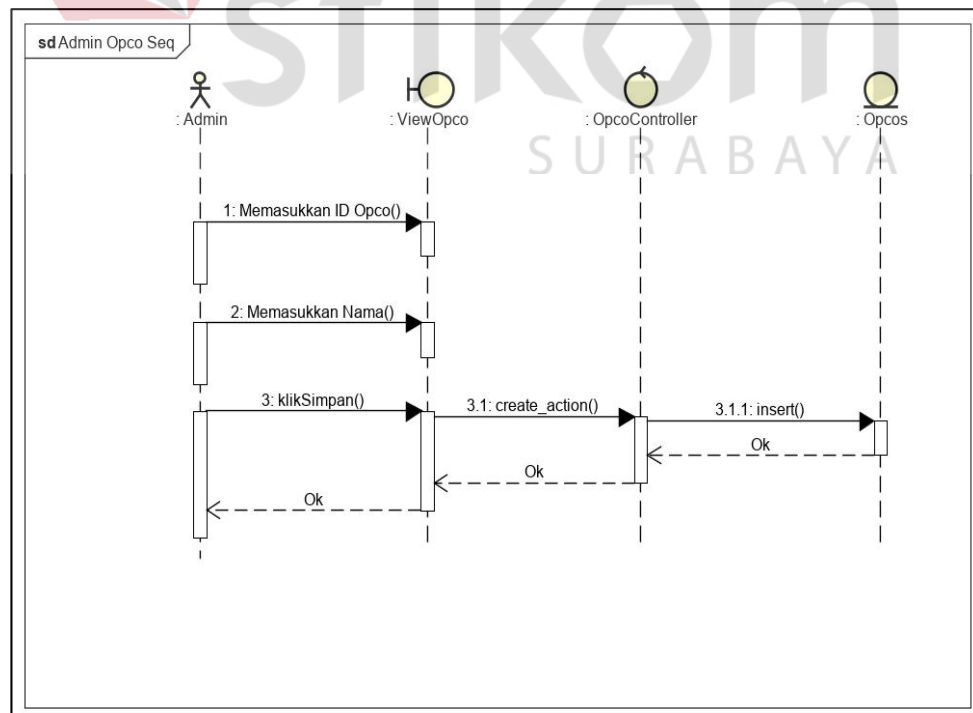
Gambar 4.20 Sequence Admin Newsfeed

4.4.8 Admin Newspaper



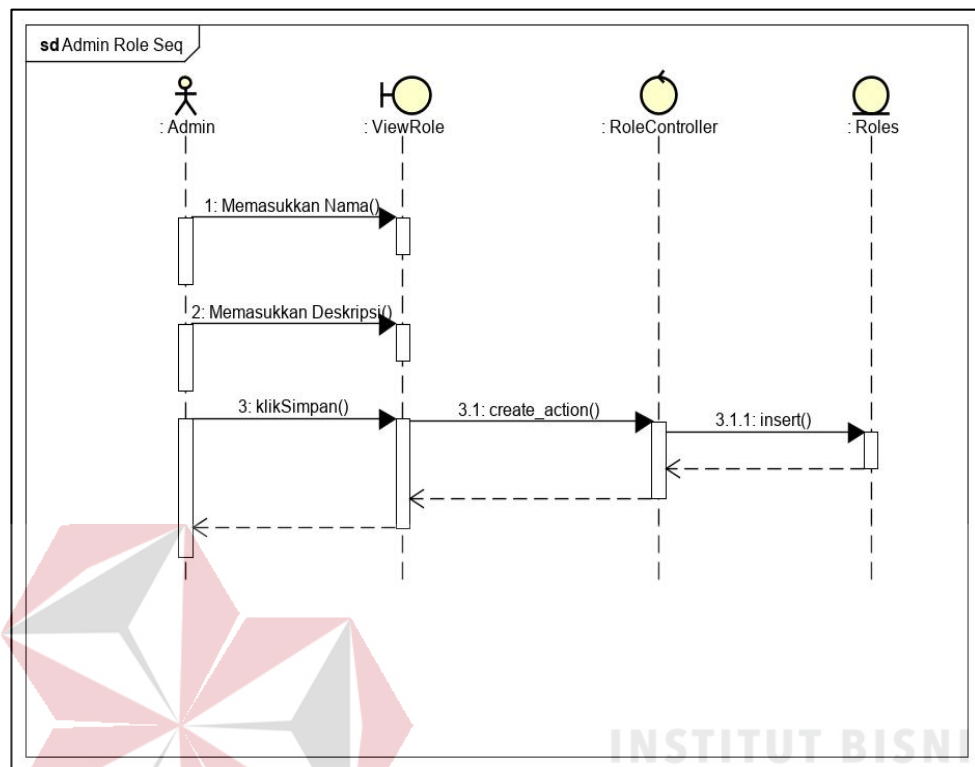
Gambar 4.21 Sequence Admin Newspaper

4.4.9 Admin Opco



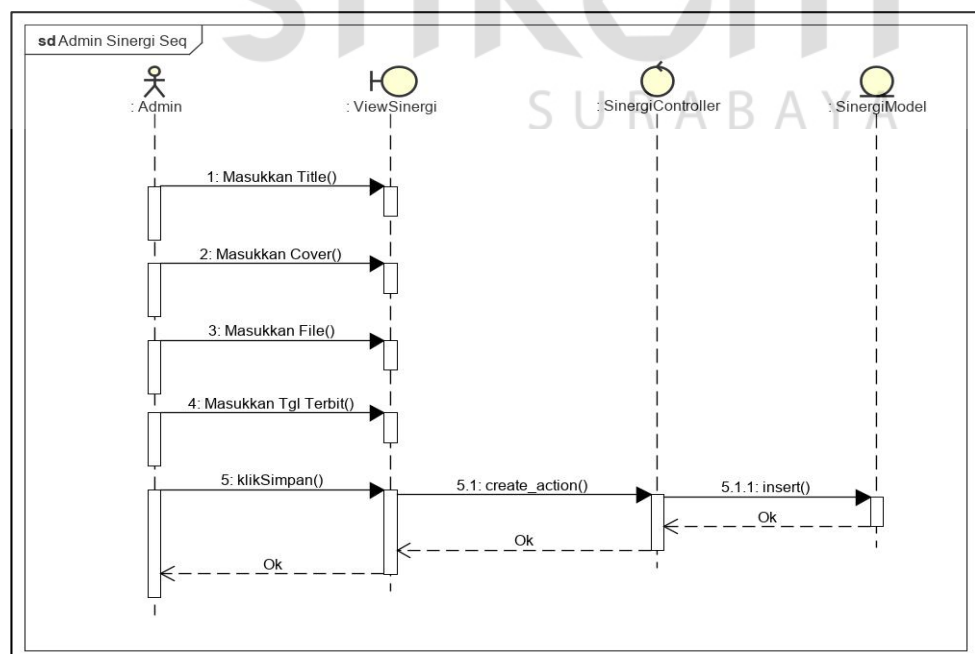
Gambar 4.22 Sequence Admin Opco

4.4.10 Admin Role



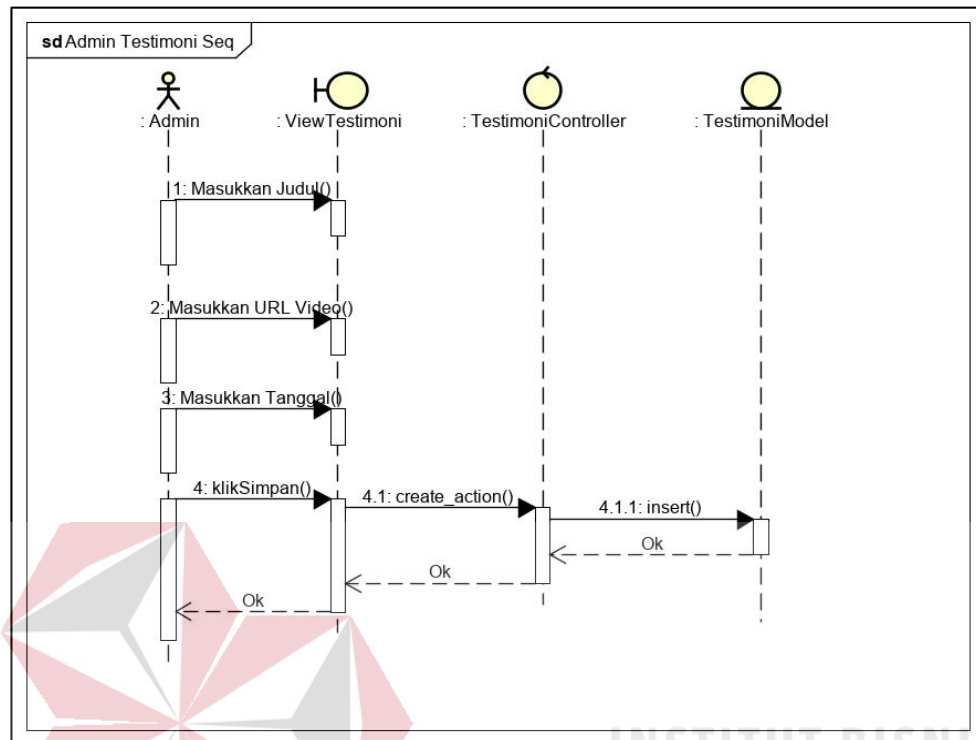
Gambar 4.23 Sequence Admin Role

4.4.11 Admin Sinergi



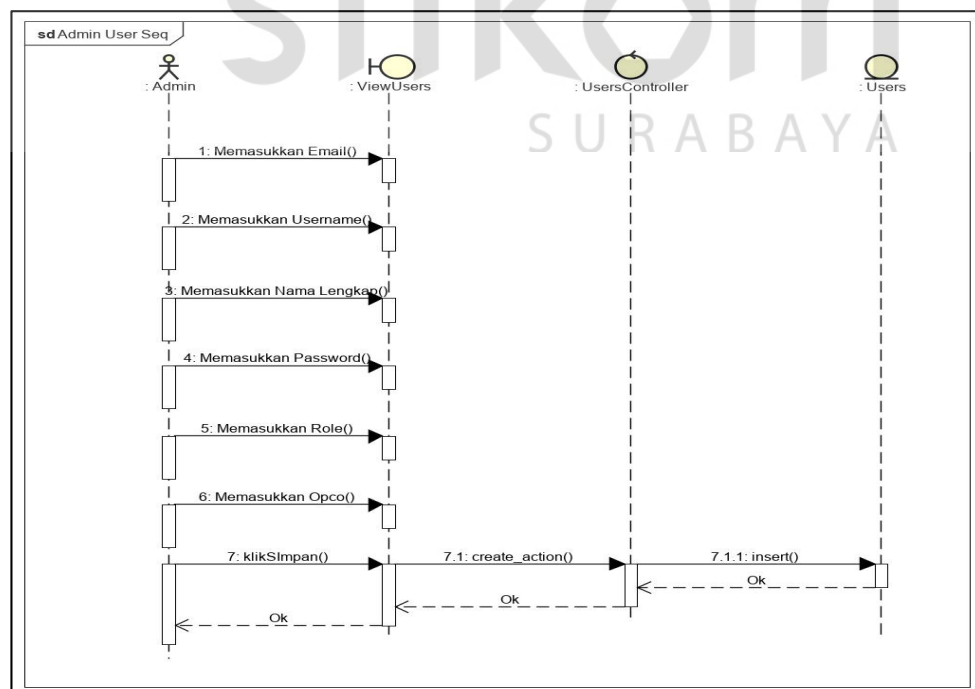
Gambar 4.24 Sequence Admin Sinergi

4.4.12 Admin Testimoni



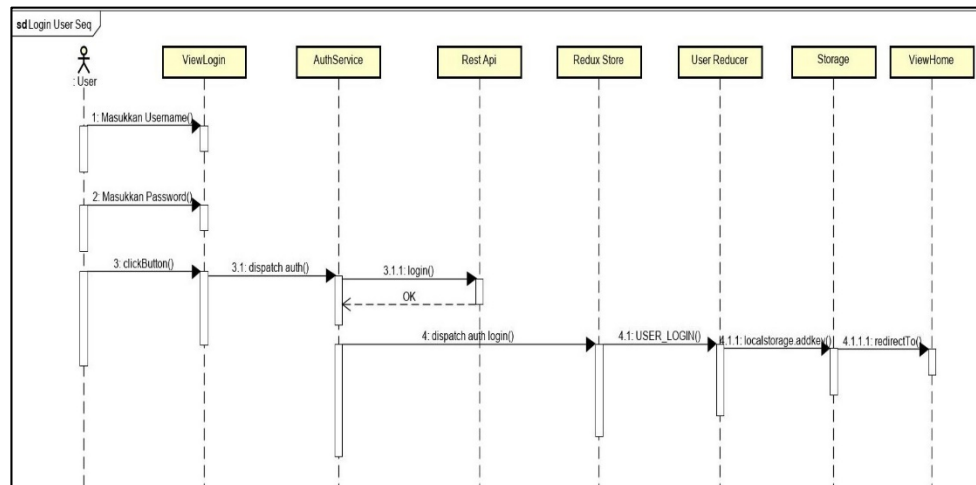
Gambar 4.25 Sequence Admin Testimoni

4.4.13 Admin User



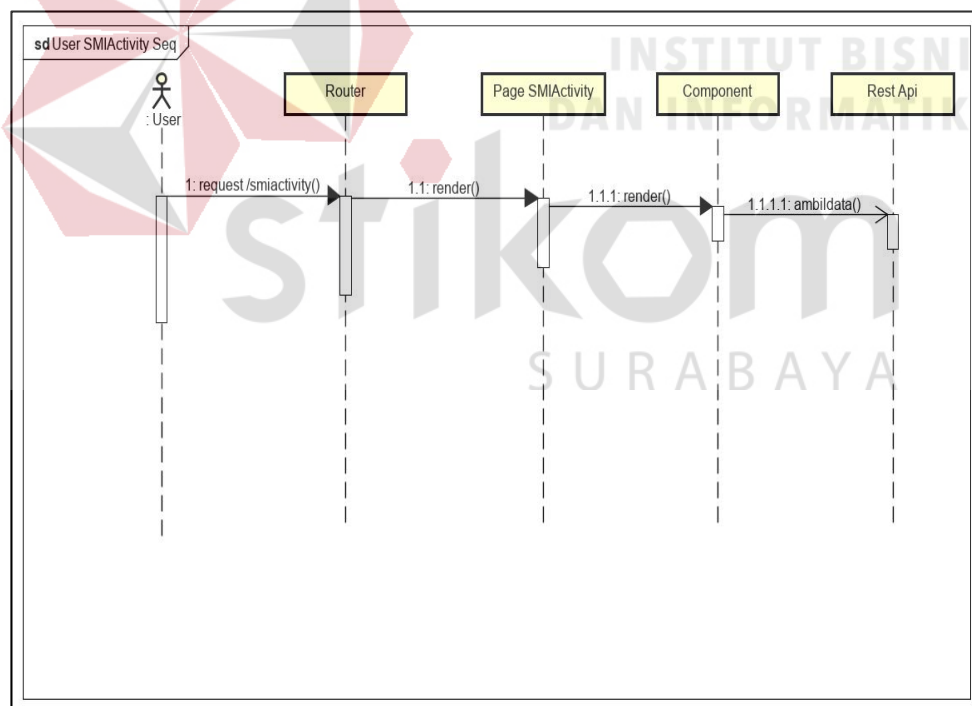
Gambar 4.26 Sequence Admin User

4.4.14 Login User



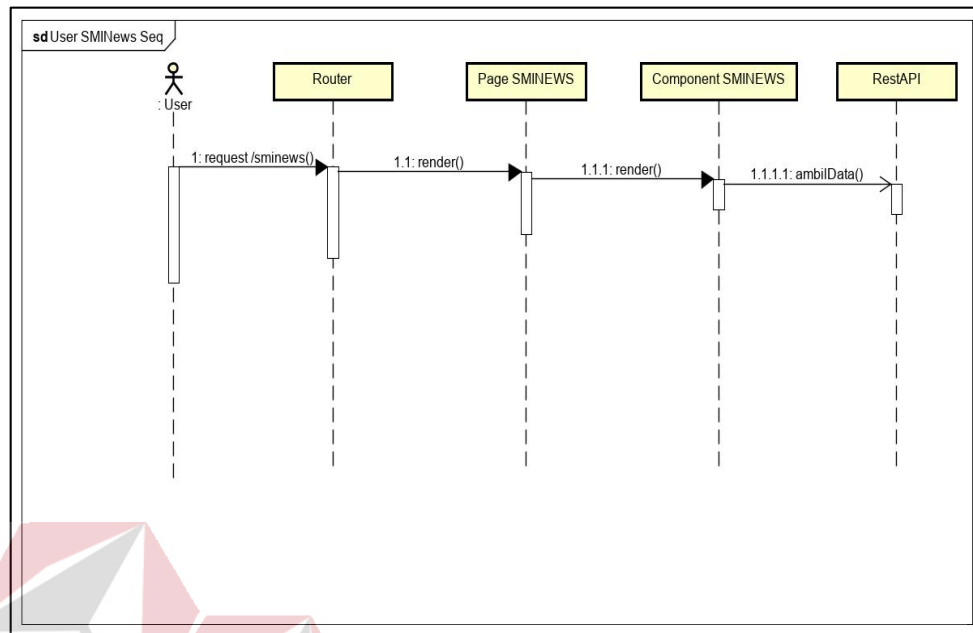
Gambar 4.27 Sequence Login User

4.4.15 User SMI Activity



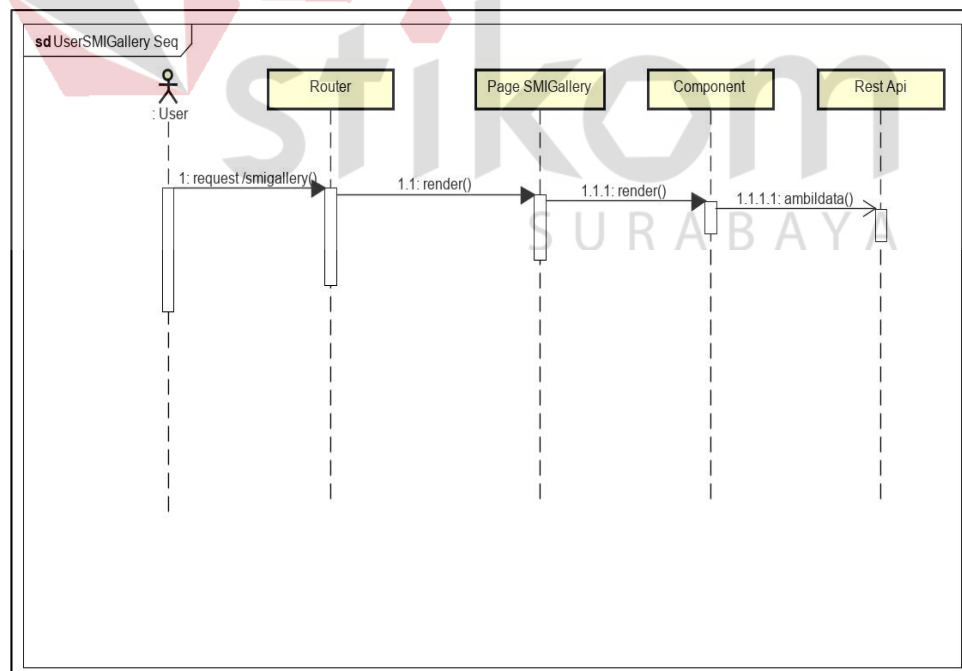
Gambar 4.28 Sequence SMI Activity

4.4.16 User SMI News



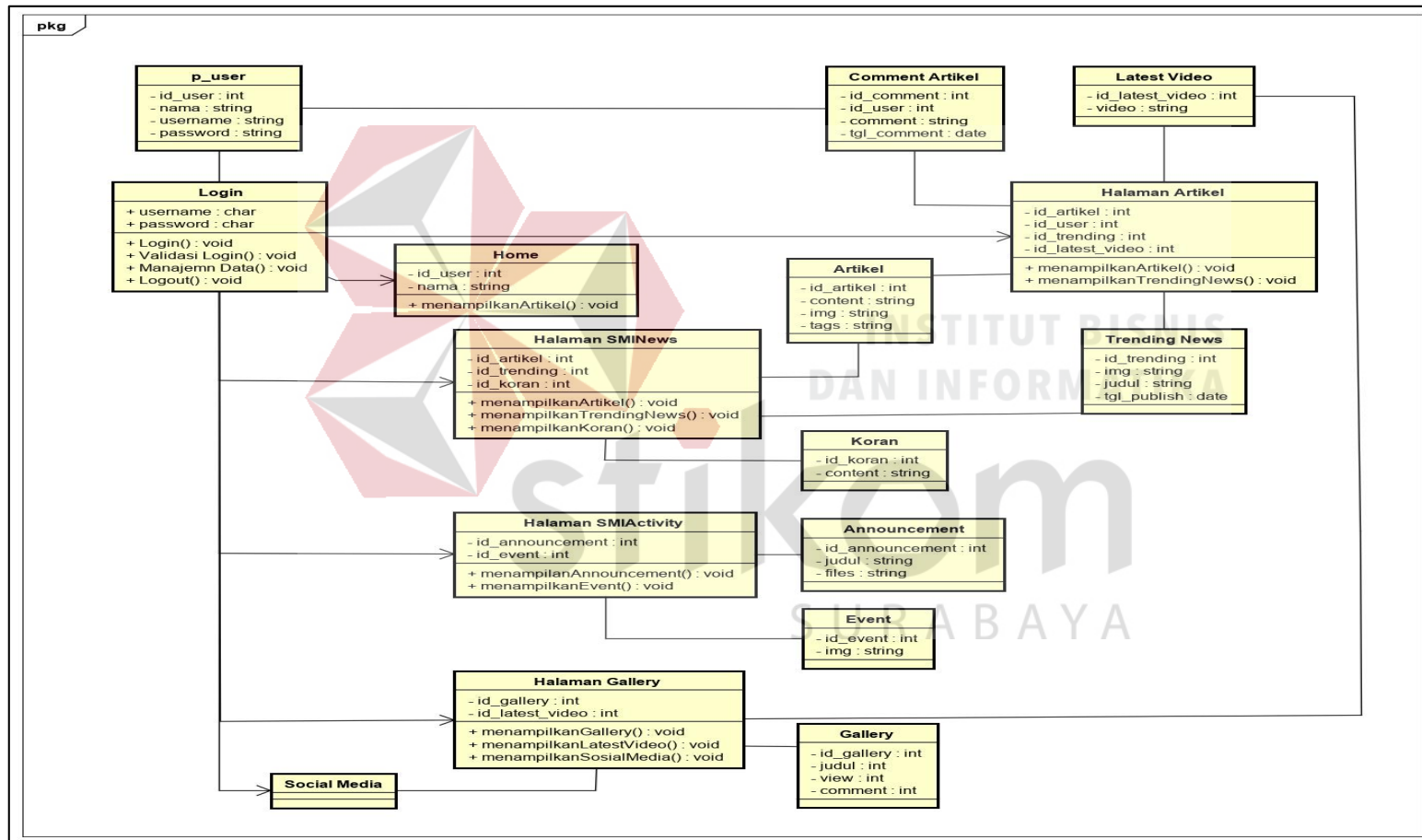
Gambar 4.29 Sequence SMI News

4.4.17 User SMI Gallery



Gambar 4.30 Sequence SMI Gallery

4.5 Class Diagram



Gambar 4.31 Class Diagram

4.5.1 p_user

Tabel 1.17 Class Diagram p_user

<i>Class</i>	p_user
<i>Attributes</i>	• id_user : int • nama : string • username : string • password : string
<i>Method</i>	-

4.5.2 Login

Tabel 1.18 Class Diagram Login

<i>Class</i>	Login
<i>Attributes</i>	• username : string • password : string
<i>Method</i>	• Login : void • Validasi_Login : void • Manajemen_Data : void • Logout : void

4.5.3 Comment Artikel

Tabel 1.19 Class Diagram comment_artikel

<i>Class</i>	comment_artikel
<i>Attributes</i>	• id_comment : int • id_user : int • comment : string • tgl_comment : date
<i>Method</i>	-

4.5.4 Latest video

Tabel 1.20 *Class Diagram* latest_video

<i>Class</i>	latest_video
<i>Attributes</i>	• id_latest_video : int • video : string
<i>Method</i>	-

4.5.5 Home

Tabel 1.21 *Class Diagram* home

<i>Class</i>	Home
<i>Attributes</i>	• id_user : int • nama : string
<i>Method</i>	• menampilkanArtikel : void

4.5.6 Halaman SMI News

Tabel 1.22 *Class Diagram* halaman_sminews

<i>Class</i>	halaman_sminews
<i>Attributes</i>	• id_artikel : int • id_trending : int • id_koran : int
<i>Method</i>	• menampilkanArtikel : void • menampilkanTrendingNews : void • menampilkanKoran : void

4.5.7 Halaman SMI Activity

Tabel 1.23 *Class Diagram* halaman_smiactivity

<i>Class</i>	halaman_smiactivity
<i>Attributes</i>	• id_announcement : int • id_event : int
<i>Method</i>	• menampilkanAnnouncement : void • menampilkanEvent : void

4.5.8 Halaman Gallery

Tabel 1.24 *Class Diagram* halaman_gallery

<i>Class</i>	halaman_gallery
<i>Atributes</i>	• id_gallery : int • id_latest_video : int
<i>Method</i>	• menampilkanGallery : void • menampilkanLatestVideo : void • menampilkanSosialMedia : void

4.5.9 Gallery

Tabel 1.25 *Class Diagram* gallery

<i>Class</i>	Gallery
<i>Atributes</i>	• id_gallery : int • judul : string • view : int
<i>Method</i>	-

4.5.10 Event

Tabel 1.26 *Class Diagram* event

<i>Class</i>	Event
<i>Atributes</i>	• id_event : int • img : string
<i>Method</i>	-

4.5.11 Announcement

Tabel 1.27 Class Diagram announcement

<i>Class</i>	<i>Announcement</i>
<i>Attributes</i>	• id_announcement : int • judul : string • files : string
<i>Method</i>	-

4.5.12 Koran

Tabel 1.28 Class Diagram koran

<i>Class</i>	<i>Koran</i>
<i>Attributes</i>	• id_koran : int • content : string
<i>Method</i>	-

4.5.13 Halaman Artikel

Tabel 1.29 Class Diagram halaman_artikel

<i>Class</i>	<i>halaman_artikel</i>
<i>Attributes</i>	• id_artikel : int • id_user : int • id_latest_video : int
<i>Method</i>	• menampilkanArtikel : void

4.5.14 Artikel

Tabel 1.30 *Class Diagram* artikel

<i>Class</i>	Artikel
<i>Atributes</i>	• id_artikel : int • content : string • image : string • tags : string
<i>Method</i>	-

4.6. Kebutuhan Sistem

A. ReactJS

a. React Redux

Penggunaan Redux dalam React berguna untuk pengambilan alur data yang berada di *state* bisa dilakukan secara *async*. Karena pada umumnya redux digunakan untuk mengambil sesuatu data yang tidak perlu ditunggu . Dan Redux ini juga bias diibaratkan *backend* didalam *frontend* .Redux dibagi menjadi tiga bagian yaitu *Store*, *Reducer*, *Action*.

- *Store*

```
import {applyMiddleware, createStore, compose} from 'redux'
import logger from 'redux-logger'
import RootReducer from './reducers'
import ReduxThunk from 'redux-thunk'
import {persistStore, persistReducer} from 'redux-persist'
import storage from 'redux-persist/lib/storage'
import hardSet from 'redux-persist/lib/stateReconciler/hardSet'

const persistConfig = {
  key: 'root',
  storage,
  stateReconciler: hardSet,
}

const persistedReducer = persistReducer(persistConfig, RootReducer);

const store = createStore(
  persistedReducer,
  compose(
    applyMiddleware(ReduxThunk, logger)
  )
);
persistStore(store);
export default store;
```

Gambar 4.32 Store

- *Reducer*

```
import {combineReducers} from 'redux'
import Auth from './Auth'
import persistStore from './persistStore'

const RootReducer = combineReducers({Auth, persistStore});

export default RootReducer;
```

Gambar 4.33 Reducer

- *Action*

```
import * as ActionTypes from '../action-types'

export function authLogin(payload){
  return {
    type: ActionTypes.AUTH_LOGIN,
    payload
  }
}

export function authLogout(){
  return {
    type: ActionTypes.AUTH_LOGOUT
  }
}

export function authCheck(){
  return {
    type: ActionTypes.AUTH_CHECK
  }
}
```

Gambar 4.34 *Action*

B. Lazy load

```

<div className="col-md-7">
  {bigHeaderImg.map((anObjectMapped, index) => {
    return (
      <div className="HeaderNews BigImg">
        <LazyLoad>
          <ProgressiveImage
            preview={
              "https://smile.semenindonesia.com/" +
              anObjectMapped.img
            }
            src={
              "https://smile.semenindonesia.com/" +
              anObjectMapped.img
            }
            render={(src, style) => (
              <img
                decoding="async"
                src={src}
                alt=""
                style={style}
              />
            )}
          />
        </LazyLoad>
      </div>
    )
  })
}

```

Gambar 4.35 Lazy Load

Lazy load adalah suatu *plugin* yang berguna untuk menghilangkan penundaan waktu *load* halaman yang diakibatkan karena kita harus men-*load* gambar terlebih dahulu (*Offscreen images*). Oleh karena itu *resource* yang tidak penting akan di *load* ketika dibutuhkan saja. Dengan *plugin* ini kita bisa menghemat *load* suatu halaman.

C. Progressive Image

Suatu *effect (plugin)* yang digunakan untuk memperindah tampilan foto ketika di-*load* kedalam suatu halaman. Jadi ketika terjadi proses *load* gambar maka akan

diambil dulu gambar yang memiliki ukuran kecil dan diberikan effect *blur* namun ketika gambar sudah ter-load versi aslinya *effect blur* tersebut akan dihilangkan.

```

<div className="col-md-7">
  {bigHeaderImg.map((anObjectMapped, index) => {
    return (
      <div className="HeaderNews BigImg">
        <LazyLoad>
          <ProgressiveImage
            preview={
              "https://smile.semenindonesia.com/" +
              anObjectMapped.img
            }
            src={
              "https://smile.semenindonesia.com/" +
              anObjectMapped.img
            }
            render={({src, style}) => (
              <img
                decoding="async"
                src={src}
                alt=""
                style={style}
              />
            )}
          />
        </LazyLoad>
      </div>
    )
  })
}

```

Gambar 4.36 *Progressive Image*

D. Localstorage

Localstorage adalah suatu teknologi HTML5 dimana kita dapat menyimpan sesuatu hal di dalam *browser* kita. Apa yang disimpan di browser kita tidak akan *expired*. *Localstorage* mirip dengan *session* yang ada di PHP. Fungsi *Localstorage* pada proyek ini adalah sebagai media penyimpanan *token*.

```

const authLogin = (state,payload) => {
  console.log(payload);
  const jwtToken = payload.token;
  const user = payload.user[0];
  if (!!payload.user[0].admin){
    localStorage.setItem('is_admin',true);
  } else {
    localStorage.setItem('is_admin',false);
  }
  localStorage.setItem('jwt_token',jwtToken);
  Http.defaults.headers.common['Authorization'] = `Bearer ${jwtToken}`;
  state = Object.assign({}, state, {
    isAuthenticated: true,
    isAdmin: localStorage.getItem('is_admin') === 'true',
    user
  });
  return state;
};

```

Gambar 4.37 LocalStorage



Application	Filter	Key	Value
Manifest		is_admin	false
Service Workers		jwt_token	eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ...
Clear storage		persist:root	{"Auth":{"isAuthenticated":true,"isAdmin":true}}
Storage			
Local Storage			
https://smile1.directoryxx.com			

Gambar 4.38 localsotrage

E. Axios

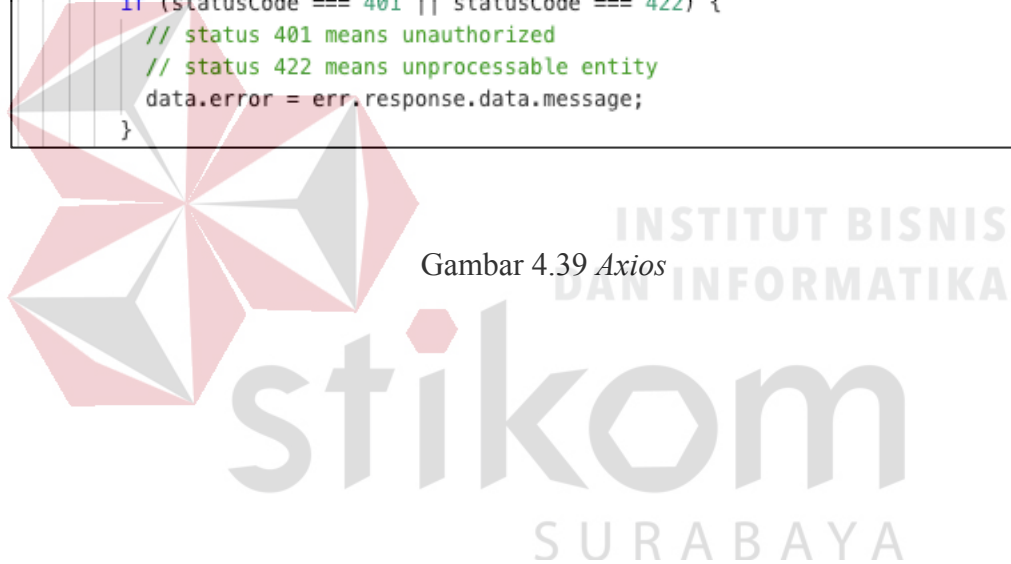
Axios adalah sebuah *library open source* yang saat ini paling *booming* untuk melakukan *request* HTTP karena memiliki banyak kelebihan namun dengan penggunaan yang tidak lebih sulit dibanding yang lain.


```

componentDidMount() {
  Http.defaults.headers.common[
    "Authorization"
  ] = `Bearer ${localStorage.getItem("jwt_token")}`;
  //latest article
  //advertis Follow link (cmd + click)
  Http.get("https://smile-api.directoryxx.com/api/admin/article/latestlimit1")
    .then(res => {
      this.setState({
        bigHeaderImg: res.data.latestlimit1
      });
    })
    .catch(err => {
      const statusCode = err.response.status;
      const data = {
        error: null,
        statusCode
      };
      if (statusCode === 401 || statusCode === 422) {
        // status 401 means unauthorized
        // status 422 means unprocessable entity
        data.error = err.response.data.message;
      }
    })
  }
}

```

Gambar 4.39 Axios



F. Codeigniter

a. JWT

```

public function login_post() {
    $u = $this->post('username'); //Username Posted
    $p = $this->post('password'); //Pasword Posted
    $q = array('username' => $u); //For where query condition
    $kunci = $this->config->item('thekey');
    $invalidLogin = array('status' => 'Invalid Login'); //Respon if login invalid
    $val = $this->M_main->get_user($q)->row(); //Model to get single data row from database
    if ($this->M_main->get_user($q)->num_rows() == 0) {
        $this->response($invalidLogin, 403);
    }
    $match = $val->password; //Get password for user from database
    //if($p == $match){ //Condition if password matched
    if (password_verify($p, $match)) {
        $token['id'] = $val->id; //From here
        $token['username'] = $u;
        $token['opko'] = $val->opko; //From here
        $date = new DateTime();
        $token['iat'] = $date->getTimestamp();
        $token['exp'] = $date->getTimestamp() + 60 * 60 * 5; //To here is to generate token
        // $output['user'] = array($this->M_main->show_data_user($u));
        // $output['token'] = JWT::encode($token,$kunci ); //This is the output token
        $res = new stdClass;
        $res->token = JWT::encode($token, $kunci);
        if ($u == "dev") {
            $res->user = array(
                $this->M_main->show_data_user($u),
            );
        } else {
            $res->user = array(
                $this->M_main->show_data_user($u),
            );
        }
        $this->response($res, 200); //This is the respon if success
    } else {
        $this->response($invalidLogin, 403); //This is the respon if failed
    }
}

```

Gambar 4.40 JWT

JSON Web Token (JWT) adalah suatu standar (RFC 7519) yang digunakan untuk mentransmisikan informasi antar pihak secara aman sebagai objek JSON. Informasi ini dapat diverifikasi dan dipercaya karena ditandatangani secara digital. Dalam proyek ini *token* berguna sebagai suatu media yang berguna untuk *otentikasi*.

b. Datatables

Suatu *plugin* yang memudahkan kita untuk dapat menata suatu tabel dengan rapi karena ketika tabel tersebut ditambahi plugin ini akan otomatis ter-*pagination* dan terdapat kolom *search* secara otomatis.

```
function dataArticle() {
    $('#article-list').DataTable({
        destroy: true,
        serverSide: false,
        processing: false,
        paging: true,
        lengthChange: true,
        searching: true,
        ordering: true,
        autoWidth: true,
        language: {"sEmptyTable": "Data User Kosong"},
        columns: [
            {data: 'no'},
            {data: 'title'},
            {data: 'subtitle'},
            {data: 'created_date'},
            {data: 'posted_date'},
            {data: 'viewed'},
            {data: 'shared_to'},
            {data: 'origin'},
            {data: 'status'},
            {data: 'action'}
        ],
        ajax: {
            url: base_url + 'api/admin/article/ambilsemua',
            type: 'GET',
            beforeSend: function (xhr, settings) {
                xhr.setRequestHeader('Authorization', 'Bearer ' + token);
            },
            dataType: 'JSON',
            dataSrc: function (dt) {
                var data = dt.semua;
                if (data.status == "error") {
                    var return_data = new Array();
                    for (var a = 0; a < 1; a++) {
                        return_data.push({
                            'no': '-',
                            'title': '-',
                            'subtitle': '-',
                            'created_date': '-',
                            'posted_date': '-',
                            'viewed': '-'
                        });
                    }
                }
            }
        }
    });
}
```

Gambar 4.41 *Datatables*

c. RestAPI

RESTAPI adalah salah satu implementasi dari suatu standar yang dimana digunakan untuk perpindahan data antar aplikasi atau system. Dalam restapi ini nantinya akan dibatasi setiap *endpoint*-nya dengan *token* yang telah dibuat

```
<?php defined('BASEPATH') OR exit('No direct script access allowed');

require APPPATH . '/libraries/JWT.php';
require APPPATH . '/libraries/BeforeValidException.php';
require APPPATH . '/libraries/ExpiredException.php';
require APPPATH . '/libraries/SignatureInvalidException.php';
use \Firebase\JWT\JWT;

/**
 * CodeIgniter Rest Controller
 *
 * A fully RESTful server implementation for CodeIgniter using one library, one config file and one
 *
 * @package      CodeIgniter
 * @subpackage   Libraries
 * @category     Libraries
 * @author       Phil Sturgeon, Chris Kacerguis
 * @license      http://philsturgeon.co.uk/code/dbad-license
 * @link         https://github.com/philsturgeon/codeigniter-restserver
 * @version      3.0.0-pre
 */
abstract class REST_Controller extends CI_Controller
{
    /**
     * This defines the rest format.
     *
     * Must be overridden in a controller so that it is set.
     *
     * @var string|null
     */
    protected $rest_format = NULL;

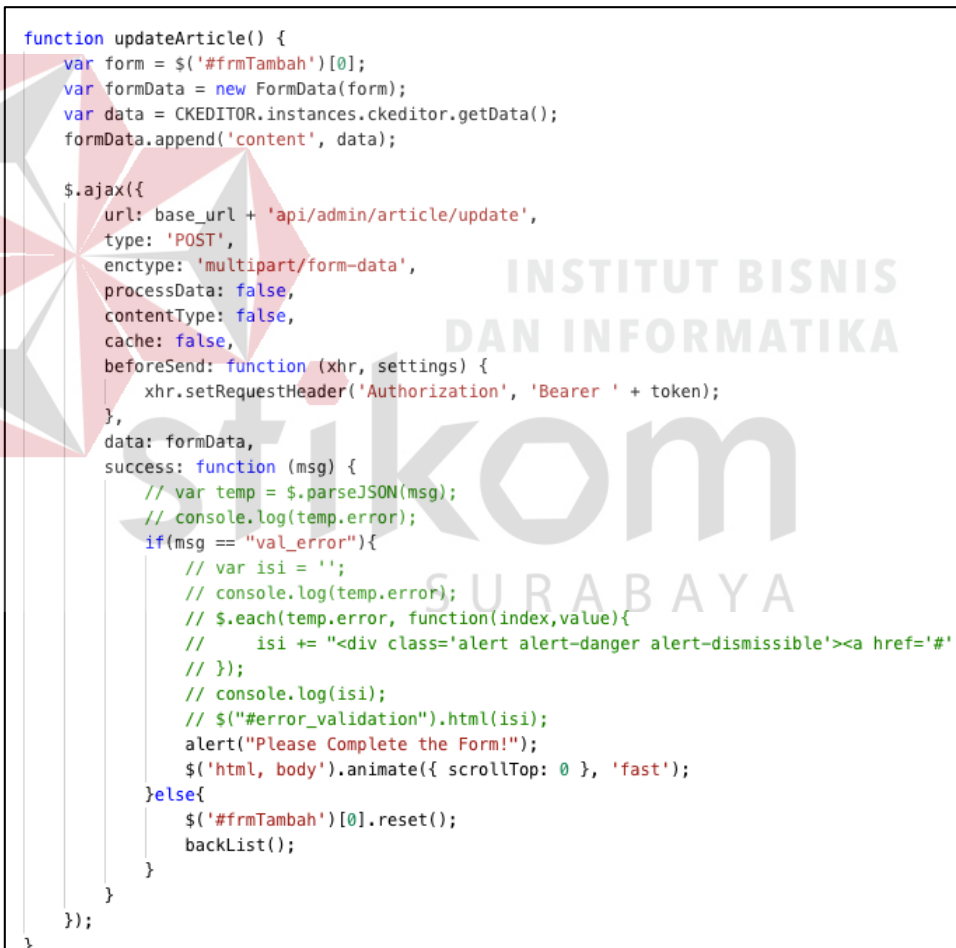
    /**
     * Defines the list of method properties such as limit, log and level
     *
     * @var array
     */
    protected $methods = array();

    /**
     * List of allowed HTTP methods
     */
}
```

Gambar 4.42 RestAPI

d. CKEditor

CKeditor adalah *text editor* yang fleksibel yang dapat digunakan untuk menginput berupa *teks* yang secara otomatis akan di-*convert* menjadi *HTML code*, tidak hanya sebatas itu kita juga bisa menambahkan gambar dan video dengan fitur *drag & drop* kedalam *editor* tersebut. Tentu saja hal ini akan memudahkan pengguna bahkan pada tingkatan yang tidak memiliki pengetahuan seputar programming sekalipun.



```
function updateArticle() {
    var form = $('#frmTambah')[0];
    var formData = new FormData(form);
    var data = CKEDITOR.instances.ckeditor.getData();
    formData.append('content', data);

    $.ajax({
        url: base_url + 'api/admin/article/update',
        type: 'POST',
        enctype: 'multipart/form-data',
        processData: false,
        contentType: false,
        cache: false,
        beforeSend: function (xhr, settings) {
            xhr.setRequestHeader('Authorization', 'Bearer ' + token);
        },
        data: formData,
        success: function (msg) {
            // var temp = $.parseJSON(msg);
            // console.log(temp.error);
            if(msg == "val_error"){
                // var isi = '';
                // console.log(temp.error);
                // $.each(temp.error, function(index,value){
                //     isi += "<div class='alert alert-danger alert-dismissible'><a href='#'
                // });
                // console.log(isi);
                // $("#error_validation").html(isi);
                alert("Please Complete the Form!");
                $('html, body').animate({ scrollTop: 0 }, 'fast');
            }else{
                $('#frmTambah')[0].reset();
                backList();
            }
        }
    });
}
```

Gambar 4. 43 CKEditor

BAB V

PENUTUP

1.1 Kesimpulan

Kesimpulan yang dapat diperoleh dari perancangan sistem aplikasi portal berita dan media informasi (SMILE) Pada PT. Sinergi Informatika Semen Indonesia adalah sebagai berikut :

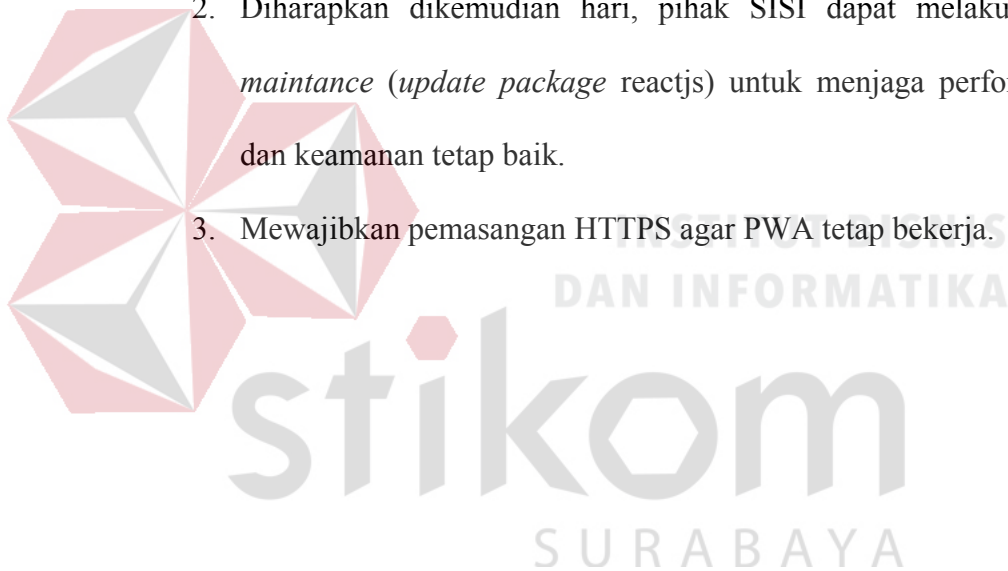
Berdasarkan permintaan penyelia untuk implementasi PWA dan mengurangi *load time* pada halaman. Penggunaan dari ReactJS cukup efektif dan efisien. Jika Aplikasi SMILE terdahulu bisa mencapai 24 detik dalam sekali load sedangkan untuk aplikasi SMILE versi baru hanya mencapai 3 detik. Aplikasi SMILE versi baru berhasil memenuhi keinginan penyelia untuk memangkas *load time*. Aplikasi SMILE versi baru pun sudah berhasil implementasi PWA dan sudah dicoba melalui *smartphone* (android).

Untuk Rest API , JWT berhasil melakukan Access Control terhadap *endpoint* API yang ada. Rest API berhasil melewati *security Testing* karena hanya *token* yang aktif dapat melakukan *insert, read, update, delete* sesuai dengan *role user* masing masing. Maka bisa disimpulkan bahwa aplikasi ini sudah memenuhi kebutuhan penyelia dan bisa masuk ke *production stage*.

1.2 Saran

Berdasarkan aplikasi portal berita dan media informasi (SMILE) pada PT. Sinergi Informatika Semen Indonesia, maka dapat diberikan beberapa saran sebagai berikut :

1. Diharapkan dikemudian hari aplikasi smile dapat menata *layout* dengan baik. Memperhatikan *resource* yang ditampilkan dalam satu halaman dan melakukan *reducing image size* agar *load* pada halaman menjadi lebih rendah.
2. Diharapkan dikemudian hari, pihak SISI dapat melakukan *maintance* (*update package reactjs*) untuk menjaga performa dan keamanan tetap baik.
3. Mewajibkan pemasangan HTTPS agar PWA tetap bekerja.



DAFTAR PUSTAKA

- Adi, L., Akbar, R. J., & Khotimah, W. N. (2017). Platform E-Learning untuk Pembelajaran Pemrograman Web Menggunakan Konsep Progressive Web Apps. *JURNAL TEKNIK ITS Vol. 6*, A579.
- Agus. (2016, February 21). *Menggunakan Local Storage dan Session Storage*. Retrieved June 8, 2019, from mragus: <https://mragus.com/menggunakan-local-storage-dan-session-storage/>
- Ariata. (2019, January 23). *Apa Itu Apache? Pengertian Apache Serta Kelebihan dan Kekurangannya*. Retrieved June 08, 2019, from Hostinger: www.hostinger.co.id/tutorial/apa-itu-apache/
- Aziz, A., & Tampati, T. (2015). Analisis Web Server untuk Pengembangan Hosting Server Institusi: Pbandingan Kinerja Web Server Apache dengan Nginx. *Analisis Web Server untuk Pengembangan Hosting Server Institusi: Pbandingan Kinerja Web Server Apache dengan Nginx*, 13.
- Manjunath. (2018, May 4). *Memulai dengan Redux: Mengapa Redux?* Retrieved June 8, 2019, from envatotuts: <https://code.tutsplus.com/id/tutorials/getting-started-with-redux-why-redux--cms-30349>
- Putra, A. W., Bhawiyuga, A., & Data, M. (2018). Implementasi Autentikasi JSON Web Token (JWT) Sebagai Mekanisme Autentikasi ProtokolMQTTPada Perangkat NodeMCU. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 585.
- Rahman, M. A. (2013). Perancangan dan Implementasi RESTful Web Service untuk Game Sosial Food Merchant Saga pada Perangkat Android. *JURNAL TEKNIK POMITS Vol. 2*, 2 - 4.
- Salma, A. I. (2017, April 4). *Mengenal Konsep JSON Web Token (JWT)*. Retrieved June 8, 2019, from DUMETschool: <https://www.dumetschool.com/blog/mengenal-konsep-json-web-token-jwt>
- Web, J. (2016, August 23). *6 Kelebihan ReactJS dan Alasan Menggunakan ReactJS Untuk Membuat Aplikasi Web*. Retrieved June 4, 2019, from www.jurnalweb.com