

# **APLIKASI PEMBUAT BUKU CERITA BERBABIS WEB**



Disusun Oleh:

N a m a : Muhammad Fadhilah Mulyana

NIM : 14523270

**PROGRAM STUDI TEKNIK INFORMATIKA – PROGRAM SARJANA**

**FAKULTAS TEKNOLOGI INDUSTRI**

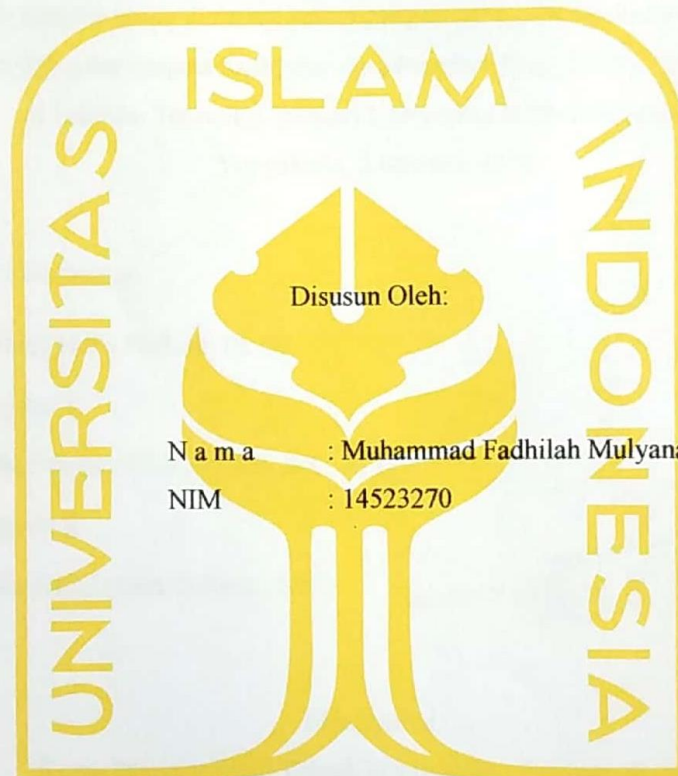
**UNIVERSITAS ISLAM INDONESIA**

**2018**

**HALAMAN PENGESAHAN DOSEN PEMBIMBING**

**APLIKASI PEMBUAT BUKU CERITA BERBASIS WEB**

**TUGAS AKHIR**



Yogyakarta, 21 September 2018

الدعوة الإسلامية  
Pembimbing,  
الإسلام في  
الاندونيسيا

( Dhomas Hatta Fudholi, Ph.D. )

## HALAMAN PENGESAHAN DOSEN PENGUJI

### APLIKASI PEMBUAT BUKU CERITA BERBASIS WEB

#### TUGAS AKHIR

Telah dipertahankan di depan sidang pengujian sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Teknik Informatika di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 3 Oktober 2018

Tim Penguji

Dhomas Hatta Fudholi, Ph.D.

Anggota 1

Ahmad Fathan Hidayatullah, S.T., M.Eng.

Anggota 2

Sheila Nurul Huda, S.Kom., M.Cs.

Mengetahui,

Ketua Program Studi Teknik Informatika – Program Sarjana

Fakultas Teknologi Industri  
Universitas Islam Indonesia



(Haden Teduh Dirgahayu S.T., M.Sc.)



## HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

### HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Muhammad Fadhilah Mulyana

NIM : 14523270

Tugas akhir dengan judul:

### **APLIKASI PEMBUAT BUKU CERITA BERBASIS WEB**

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 21 September 2018



( Muhammad Fadhilah Mulyana )

**HALAMAN PERSEMBAHAN**

## HALAMAN MOTO

لَا يُكَلِّفُ اللَّهُ نَفْسًا إِلَّا وُسْعَهَا

*“Allah tidak akan membebani seseorang melainkan sesuai dengan kemampuannya”*

*(Al-Baqarah : 186)*

وَاصْبِرْ نَفْسَكَ مَعَ الَّذِينَ يَدْعُونَ رَبَّهُمْ بِالْغَدَاةِ وَالْعَشِيِّ  
يُرِيدُونَ وَجْهَهُ

*“Dan bersabarlah kamu bersama-sama dengan orang-orang yang menyeru Tuhan mereka di pagi dan senja hari dengan mengharap keridaan-Nya”*

*(Al-Kahfi : 28)*

## KATA PENGANTAR

### *Assalamu'alaikum Warahmatullahi Wabarakatuh*

*Alhamdulillah*, segala puji bagi Allah ﷻ atas kehadirat-Nya yang telah memberikan rahmat, hidayah, dan karunia-Nya, sehingga laporan tugas akhir dapat terselesaikan. Tak lupa shalawat dan salam selalu terkirimkan kepada Nabi Muhammad ﷺ, yang telah membawa manusia dari zaman jahiliyah menuju jaman terang benderang.

Tugas akhir ini dibuat sebagai salah satu syarat yang harus dipenuhi untuk memperoleh gelar sarjana di Program Studi Teknik Informatika Universitas Islam Indonesia. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan rasa terima kasih kepada:

1. Allah ﷻ yang masih memberikan kenikmatan, yakni nikmat kesehatan, nikmat islam, dan nikmat sunnah.
2. Bapak dan Ibu saya, serta Kakak-kakak saya yang senantiasa mendoakan, dan memberikan dukungan dan semangat dalam pengerjaan tugas akhir ini.
3. Bapak Prof. Dr. Ir. Hari Purnomo selaku Dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
4. Bapak Raden Teguh Dirgahayu selaku Ketua Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.
5. Bapak Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D. selaku Dosen Pembimbing Tugas Akhir yang selama pembuatan tugas akhir telah memberikan pengarahan dan bimbingan.
6. Segenap dosen Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia yang telah membekali penulis dengan ilmu yang bermanfaat selama berkuliah.
7. Segenap teman-teman dari Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia yang kompak dan saling membantu dalam kesusahan dan kemudahan

Penulis menyadari bahwa laporan ini masih belum sempurna. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun demi kesempurnaan Laporan Tugas Akhir ini. Akhir kata, penulis berharap agar laporan ini dapat bermanfaat bagi semua pihak yang bersangkutan dan semoga segala urusan kita selalu dimudahkan Allah ﷻ, aamiin.



*Wassalamu'alaikum Warahmatullahi Wabarakatuh*

Yogyakarta, 21 September 2018

( Muhammad Fadhilah Mulyana )

## SARI

Pendidikan dini yang baik seperti menulis dan membaca merupakan hal yang penting untuk membentuk masyarakat yang baik. Namun kegiatan seperti menulis dan membaca bagi masyarakat Indonesia masih terbilang kurang diminati sehingga dapat mengakibatkan kualitas SDM yang rendah. Solusi yang dapat diberikan dari masalah yang ada yaitu dengan mengembangkan aplikasi yang dapat memberikan sarana bagi masyarakat untuk dapat membuat suatu tulisan agar masyarakat dapat belajar dan dapat terbiasa dalam hal menulis dan membaca. Tujuan dari pengembangan aplikasi ini adalah untuk meningkatkan minat membaca dan menulis bagi masyarakat Indonesia. Aplikasi tersebut dikembangkan dalam bentuk aplikasi web agar lebih efektif dan mudah diakses. Hasil akhir dari penelitian ini berupa sebuah aplikasi berbasis web yang dapat digunakan untuk membuat tulisan dan memiliki fitur untuk menyediakan rekomendasi aktiva atau *asset* berupa gambar yang mendukung dan sesuai dengan teks yang dimasukkan.

Kata kunci: aplikasi web, menulis, membaca, buku cerita.

## GLOSARIUM

<i>Browser</i>	Aplikasi yang digunakan untuk mengakses halaman web dan menampilkan konten web.
<i>Debug</i>	Langkah untuk menelusuri kesalahan kode program
<i>Javascript</i>	Bahasa Script pemrograman yang digunakan untuk mengembangkan aplikasi interaktif berbasis web
<i>Frontend</i>	Sebutan untuk para developer programmer yang menangani masalah perancangan antarmuka, pada sebuah aplikasi
<i>Backend</i>	Sebutan untuk para developer programmer yang menangani masalah dalam pembuatan pengembangan dan logik suatu aplikasi.
<i>Endpoint</i>	URL yang di akses dari suatu API.
<i>HTTP</i>	Protokol jaringan lapisan aplikasi yang digunakan untuk sistem informasi terdistribusi, kolaboratif, dan menggunakan hipermedia
<i>Internet</i>	Seluruh jaringan komputer yang saling terhubung menggunakan standar sistem global Transmission Control Protocol/Internet Protocol Suite (TCP/IP) sebagai protokol pertukaran paket (packet switching communication protocol) untuk melayani miliaran pengguna di seluruh dunia.
<i>JSON</i>	Format pertukaran data
<i>HTML</i>	Sebuah bahasa markup yang digunakan untuk membuat sebuah halaman web, dan menampilkan berbagai informasi
<i>Assets</i>	Aktiva-aktiva atau data inventaris yang disimpan
<i>Offline</i>	Keadaan dimana tidak terkoneksi dengan internet
<i>Online</i>	Keadaan komputer yang terkoneksi/ terhubung ke jaringan Internet.
<i>Web</i>	Halaman informasi yang disediakan melalui jalur internet
<i>Error</i>	Kesalahan yang terdapat dalam code, atau aplikasi yang tidak menghasilkan sesuai perintah

## DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING .....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI .....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	v
HALAMAN PERSEMBAHAN .....	vi
HALAMAN MOTO .....	vii
KATA PENGANTAR.....	viii
SARI .....	x
GLOSARIUM .....	xi
DAFTAR ISI .....	xii
DAFTAR TABEL .....	xv
DAFTAR GAMBAR.....	xvi
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah .....	2
1.4 Tujuan Penelitian .....	3
1.5 Manfaat Penelitian .....	3
1.6 Metode Penelitian .....	3
1.7 Sistematika Penulisan .....	4
<b>BAB II LANDASAN TEORI .....</b>	<b>6</b>
2.1 <i>Review Website</i> Menulis .....	6
2.1.1 StoryJumper.com.....	6
2.1.2 MyStoryBook.com .....	7
2.1.3 Storybird.com .....	8
2.2 Studi Literatur .....	9
2.2.1 Penulisan Buku Cerita Yang Baik.....	10
2.2.2 Rencana Fitur Unggulan.....	11
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>12</b>
3.1 Analisis Kebutuhan .....	12
3.1.1 Perencanaan Sistem.....	12
3.1.2 Analisis Kebutuhan Perangkat Lunak .....	14

3.1.3	Analisis Kebutuhan Perangkat Keras .....	14
3.1.4	Analisis Pengguna .....	15
3.2	<i>Use case</i> Diagram .....	15
3.2.1	Skenario dari <i>Use case</i> Diagram .....	17
3.3	<i>Flowchart</i> Diagram .....	25
3.4	Activity Diagram.....	29
3.4.1	Activity Registrasi Pengunjung.....	29
3.4.2	Activity Login .....	31
3.4.3	Activity Mencari Buku Cerita .....	32
3.4.4	Activity Membaca Buku Cerita.....	32
3.4.5	Activity Mengelola Profil.....	33
3.4.6	Activity Membuat Cerita.....	34
3.4.7	Activity Memberi Saran & Kritik .....	34
3.4.8	Activity Mengelola Buku Cerita .....	35
3.4.9	Activity Mengelola Buku-buku Cerita .....	36
3.4.10	Activity Mengelola Pengguna .....	36
3.4.11	Activity Mengelola Kategori.....	37
3.4.12	Activity Mengelola Asset.....	38
3.4.13	Activity Mengelola Kosa Kata .....	39
3.5	<i>Entity Relationship</i> Diagram .....	40
3.6	Struktur Tabel <i>Database</i> .....	42
3.6.1	Tabel <i>Users</i> .....	42
3.6.2	Tabel <i>Profiles</i> .....	42
3.6.3	Tabel <i>Books</i> .....	42
3.6.4	Tabel <i>Canvas</i> .....	43
3.6.5	Tabel <i>Category</i> .....	43
3.6.6	Tabel Asset .....	44
3.6.7	Tabel <i>Suggestions</i> .....	44
3.6.8	Tabel <i>Stopwords</i> .....	44
3.6.9	Tabel <i>Blockedwords</i> .....	45
3.7	Relational Schemas .....	45
3.8	Perancangan <i>Mockup</i> Antarmuka ( <i>Interface</i> ) .....	46
3.8.1	Halaman <i>Index</i> Sistem.....	46
3.8.2	Halaman Registrasi.....	46

3.8.3	Halaman <i>Login</i> .....	47
3.8.4	Halaman Beranda Anggota .....	47
3.8.5	Halaman Membuat Buku Cerita.....	48
3.8.6	Halaman Semua Buku .....	48
3.8.7	Halaman Memberi Kritik Dan Saran.....	49
3.8.8	Halaman Bantuan .....	49
3.8.9	Halaman Kontak.....	50
3.8.10	Halaman Beranda Admin .....	50
3.8.11	Halaman Manajemen Kategori Admin.....	51
3.8.12	Halaman Asset Admin.....	51
3.8.13	Halaman Manajemen Pengguna Admin.....	52
3.8.14	Halaman Manajemen Buku Admin .....	52
3.8.15	Halaman Manajemen Kosa Kata Admin.....	53
3.8.16	Halaman Kritik & Saran Admin.....	53
BAB IV IMPLEMENTASI DAN PENGUJIAN .....		54
4.1	Implementasi Sistem .....	54
4.1.1	Pencarian Buku.....	54
4.1.2	Membaca Buku.....	56
4.1.3	Membuat Buku Cerita .....	56
4.1.4	Mengunduh Cerita.....	59
4.1.5	Rekomendasi Asset .....	59
4.2	Pengujian.....	64
4.2.1	Pengujian Fungsionalitas ( <i>Black Box</i> ).....	65
4.2.2	Pengujian Usabilitas .....	67
4.3	Kelebihan dan Kekurangan Sistem .....	69
4.3.1	Kelebihan Aplikasi .....	69
4.3.2	Kekurangan Aplikasi .....	70
BAB V KESIMPULAN DAN SARAN .....		71
5.1	Kesimpulan .....	71
5.2	Saran.....	71
DAFTAR PUSTAKA .....		72
LAMPIRAN .....		74

## DAFTAR TABEL

Tabel 2.1 Perbandingan <i>Website</i> Menulis Cerita.....	9
Tabel 3.1 Tabel Gambaran Fitur Yang Tersedia di Aplikasi.....	12
Tabel 3.2 Registrasi Pengunjung (UC-1).....	17
Tabel 3.3 <i>Login</i> (UC-2).....	18
Tabel 3.4 Mencari Buku Cerita (UC-3) .....	18
Tabel 3.5 Membaca Buku Cerita (UC-4).....	19
Tabel 3.6 Mengelola Profil (UC-5).....	20
Tabel 3.7 Membuat Cerita (UC-6).....	20
Tabel 3.8 Memberi Kriti & Saran (UC-7) .....	21
Tabel 3.9 Mengelola Buku Cerita (UC-8) .....	21
Tabel 3.10 Mengelola Buku-buku Cerita (UC-9) .....	22
Tabel 3.11 Mengelola Pengguna (UC-10) .....	23
Tabel 3.12 Mengelola Kategori (UC-11).....	23
Tabel 3.13 Mengelola Asset (UC-12).....	24
Tabel 3.14 Mengelola Kosa Kata (UC-13).....	24
Tabel 3.15 Tabel Kata.....	27
Tabel 3.16 Tabel <i>Users</i> .....	42
Tabel 3.17 Tabel <i>Profiles</i> .....	42
Tabel 3.18 Tabel <i>Books</i> .....	43
Tabel 3.19 Tabel <i>Canvas</i> .....	43
Tabel 3.20 Tabel <i>Categories</i> .....	43
Tabel 3.21 Tabel <i>Assets</i> .....	44
Tabel 3.22 Tabel <i>Suggestion</i> .....	44
Tabel 3.23 Tabel <i>Stopwords</i> .....	44
Tabel 3.24 Tabel <i>Blockedword</i> .....	45
Tabel 4.1 Tabel Pengujian Fungsionalitas .....	65
Tabel 4.2 Tabel Hasil Pengujian Usabilitas .....	68

## DAFTAR GAMBAR

Gambar 3.1 <i>Use case</i> Diagram .....	16
Gambar 3.2 <i>Flowchart</i> Tokenisasi.....	25
Gambar 3.3 <i>Flowchart</i> <i>Stopwords Removal</i> .....	26
Gambar 3.4 <i>Flowchart</i> <i>Stemming</i> pada aplikasi .....	28
Gambar 3.5 <i>Flowchart</i> Mengecek <i>Blockwords</i> .....	29
Gambar 3.6 <i>Activity</i> Diagram Registrasi Pengunjung .....	30
Gambar 3.7 <i>Activity</i> Diagram <i>Login</i> .....	31
Gambar 3.8 <i>Activity</i> Diagram Pencarian Buku .....	32
Gambar 3.9 <i>Activity</i> Diagram Membaca Buku Cerita .....	33
Gambar 3.10 <i>Activity</i> Diagram Mengelola Profil .....	33
Gambar 3.11 <i>Activity</i> Diagram Membuat Cerita .....	34
Gambar 3.12 <i>Activity</i> Diagram Memberi Saran & Kritik .....	35
Gambar 3.13 <i>Activity</i> Diagram Mengelola Buku .....	35
Gambar 3.14 <i>Activity</i> Diagram Mengelola Buku-buku Cerita.....	36
Gambar 3.15 <i>Activity</i> Mengelola Pengguna.....	37
Gambar 3.16 <i>Activity</i> Mengelola Kategori .....	38
Gambar 3.17 <i>Activity</i> Diagram Mengelola Asset .....	39
Gambar 3.18 <i>Activity</i> Diagram Mengelola Kosakata.....	40
Gambar 3.19 <i>Entity Relationship</i> Diagram (ERD) .....	41
Gambar 3.20 Skema Relasi <i>Database</i> .....	45
Gambar 3.21 Halaman <i>Index</i> Sistem Bagi Pengunjung.....	46
Gambar 3.22 Halaman Registrasi Pengguna .....	46
Gambar 3.23 Halaman <i>Login</i> .....	47
Gambar 3.24 Halaman Beranda Anggota .....	47
Gambar 3.25 Halaman Membuat Buku Cerita .....	48
Gambar 3.26 Halaman Semua Buku.....	48
Gambar 3.27 Halaman Memberi Kritik & Saran.....	49
Gambar 3.28 Halaman Bantuan.....	49
Gambar 3.29 Halaman Kontak .....	50
Gambar 3.30 Halaman Beranda Admin.....	50
Gambar 3.31 Halaman Manajemen Kategori Admin .....	51
Gambar 3.32 Halaman Manajemen Asset Admin .....	51



Gambar 3.33 Halaman Manajemen Pengguna Admin .....	52
Gambar 3.34 Halaman Manajemen Buku.....	52
Gambar 3.35 Halaman Manajemen Kosakata.....	53
Gambar 3.36 Halaman Kritik & Saran Untuk Admin .....	53
Gambar 4.1 Halaman <i>Index</i> .....	54
Gambar 4.2 Halaman <i>Index</i> Kotak Pencarian.....	55
Gambar 4.3 Halaman Hasil Pencarian Buku Berhasil .....	55
Gambar 4.4 Halaman Hasil Pencarian Buku Gagal.....	55
Gambar 4.5 Halaman Membaca Buku.....	56
Gambar 4.6 Halaman Beranda Anggota .....	56
Gambar 4.7 Halaman Membuat Cerita .....	57
Gambar 4.8 Fitur Menerapkan Latar Belakang .....	57
Gambar 4.9 Fitur Penambahan Teks.....	58
Gambar 4.10 Penggunaan Fitur Menulis/Menggambar.....	58
Gambar 4.11 Fitur Penambahan Asset.....	59
Gambar 4.12 Halaman Pengunduhan Buku Cerita .....	59
Gambar 4.13 Anggota Memasukkan Teks Cerita.....	60
Gambar 4.14 Script Tokenisasi.....	60
Gambar 4.15 Script <i>Stopwords Removal</i> .....	61
Gambar 4.16 <i>Script Stemming</i> .....	62
Gambar 4.17 <i>Script</i> Pencarian Asset .....	63
Gambar 4.18 <i>Script</i> Pencarian <i>Blockwords</i> .....	64

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Menulis merupakan salah satu kegiatan yang terkadang dianggap sulit dan menjadi permasalahan pada jenjang pendidikan, padahal menulis telah diajarkan di bangku sekolah dasar. Sama halnya dengan membaca, terkadang membaca dianggap sulit dilakukan padahal membaca merupakan referensi untuk menulis. Pada penelitian yang dilakukan oleh *Organisation for Economic Co-operation and Development (OECD)* dalam programnya yaitu *Programme for International Student Assessment (PISA)* di tahun 2009 menyebutkan, bahwa budaya literasi (menulis dan membaca) masyarakat Indonesia memiliki *index* poin yang cukup baik yaitu sebesar 402 poin. Namun pada penelitian yang sama ditahun 2015 Indonesia mengalami penurunan poin menjadi 397. Hal ini menunjukkan bahwa literasi membaca pada masyarakat Indonesia mengalami penurunan dari tahun 2009 dan terbilang masih memiliki poin yang cukup rendah jika dibandingkan dengan negara-negara lain yang memiliki rerata *index* poin sebesar 400.

Tingkatan rendah yang dimiliki masyarakat Indonesia dalam hal menulis dan membaca dapat mengakibatkan keterpurukan pendidikan yang akan menghasilkan kualitas sumber daya manusia yang buruk. Hal ini akan berakibat kepada negara Indonesia itu sendiri karena negara tersebut hanya akan dapat menghasilkan tenaga kerja kasar/buruh. Maka dari itu belajar literasi sangatlah penting dilakukan sejak dini, karena literasi merupakan penentu keberhasilan dalam kegiatan belajar (Kana, 2017). Keterampilan pembelajaran menulis permulaan disajikan bersama dengan membaca permulaan yang biasa sering disebut (Membaca dan Menulis Permulaan). Kemampuan membaca permulaan lebih diorientasikan pada kemampuan membaca tingkat dasar, yaitu mengenal huruf. Sedangkan kemampuan menulis permulaan tidak jauh beda dengan kemampuan membaca permulaan.

Umumnya tujuan dari menulis dan membaca adalah meningkatkan kemampuan dalam mengolah informasi yang diperoleh lalu mengungkapkannya melalui tulisan. Selain itu tujuannya ialah agar pembaca dapat lancar dalam membaca dan memahami suatu tulisan. Namun terkadang pembaca cepat merasa jenuh dalam belajar sesuatu. Oleh karena itu untuk meningkatkan semangat dalam menulis dan membaca adalah mengajarkan mereka menulis suatu buku cerita yang berasal dari ide mereka. Sebagaimana menurut pendapat Abbas (2006),

keterampilan menulis adalah kemampuan mengungkapkan gagasan, pendapat, dan perasaan kepada pihak lain dengan melalui bahasa tulisan. Namun hanya saja saat ini media yang mudah digunakan untuk menulis dan membaca suatu cerita masih sedikit untuk ditemui.

Berdasarkan penjelasan diatas dibutuhkan suatu media yang dapat memberikan wadah bagi masyarakat untuk dapat menulis suatu cerita. Media tersebut direncanakan berupa aplikasi berbasis *web*. Aplikasi tersebut memberikan wadah untuk menulis suatu cerita dan wadah untuk membaca cerita. Aplikasi tersebut juga menyediakan aktiva-aktiva atau *asset* berupa gambar-gambar yang dapat mendukung jalannya suatu cerita yang dituliskan. Untuk memudahkan dalam menuliskan cerita maka terdapat fitur yang memberi rekomendasi gambar atau aktiva yang cocok dengan cerita yang dituliskan. Aplikasi akan secara otomatis mendeteksi teks yang diketikan dan menampilkan gambar apa saja yang tepat untuk jalan cerita yang dituliskan. Hasil cerita yang telah selesai dituliskan dapat dibagikan dan dibaca oleh semua orang yang terdaftar pada aplikasi, sehingga masing-masing pengguna baik dari kalangan kanak-kanak maupun kalangan dewasa dapat saling berbagi suatu cerita. Dengan adanya aplikasi ini, diharapkan dapat meningkatkan semangat dalam menulis dan membaca dari kalangan anak-anak hingga dewasa, juga dapat meningkatkan budaya literasi masyarakat Indonesia.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, maka permasalahan yang dapat dirumuskan adalah bagaimana merancang aplikasi berbasis *web* yang dapat memberikan kemudahan dalam membuat suatu buku cerita dengan menyediakan aktiva atau *assets* berupa gambar ilustrasi yang dapat mendukung dalam proses pembuatan cerita.

## 1.3 Batasan Masalah

Dalam penelitian tugas akhir ini dibuat beberapa batasan masalah agar penelitian dapat terfokus pada masalah yang akan diselesaikan, yaitu:

- a. Aplikasi yang dibangun berbasis *website*.
- b. Aktiva-aktiva atau *asset* yang disediakan berasal dan diatur oleh admin.
- c. Aplikasi yang dibangun ditujukan seluruh kalangan masyarakat Indonesia, baik dari kalangan anak-anak hingga orang dewasa.
- d. Buku dapat di ekspor hanya dengan format PDF.

#### 1.4 Tujuan Penelitian

Tujuan dari penelitian tugas akhir ini adalah membangun aplikasi berbasis internet yang dapat memberikan wadah untuk dapat menulis dan membaca cerita dengan mudah dan memberikan aktiva-aktiva yang mendukung dalam proses pembuatan cerita.

#### 1.5 Manfaat Penelitian

Manfaat yang diharapkan dari penilitan ini adalah:

- a. Bagi Masyarakat:
  1. Masyarakat Indonesia memiliki semangat dan senang dalam hal menulis dan membaca
  2. Budaya literasi Indonesia menjadi semakin baik.
  3. Memicu anak-anak untuk dapat mengembangkan karya-karya tulisan berdasarkan ide-ide mereka.
  4. Menanamkan nilai-nilai kehidupan pada anak-anak pelajar.
- b. Bagi Pengguna:
  1. Memberikan wadah untuk dapat berkreasi dalam membuat suatu tulisan.
  2. Memberikan wadah dalam mencari tulisan bercerita.
- c. Bagi Penulis:
  1. Memberikan pengetahuan baru bagi penulis dalam proses pengembangan
  2. Turut ikut serta dalam mengembangkan pendidikan bangsa

#### 1.6 Metode Penelitian

Dalam penelitian tugas akhir ini, metodologi penelitian yang dilakukan dalam pengembangan aplikasi ini yakni sebagai berikut:

##### a. Pengumpulan Data

Mencari dan mengumpulkan informasi serta data dari beberapa sumber seperti jurnal dan *website* yang berhubungan dengan materi pada tugas akhir ini. Pada tahap ini juga dilakukan review aplikasi serupa yang sudah ada.

##### b. Analisis Kebutuhan

Tahap yang dilakukan untuk mengidentifikasi masalah dan kebutuhan dalam perancangan aplikasi yang meliputi analisis masukan, keluaran, proses dan antarmuka dari aplikasi.

c. Desain Perancangan Sistem

Tahap perancangan aplikasi meliputi, Bahasa pemrograman yang digunakan yaitu PHP dan Javascript, desain *database*, *use case diagram*, dan desain antarmuka yang berupa *mockup* dari aplikasi *website* yang akan dibangun.

d. Implementasi

Implementasi atau pengkodean, yaitu tahap dalam membangun aplikasi dengan cara pengkodean yang menggunakan Bahasa Pemrograman PHP dan Javascript.

e. Pengujian

Pada tahap ini dilakukan pengujian terhadap aplikasi khususnya terhadap fitur-fitur dari aplikasi dan terhadap penggunaan aplikasi untuk mengetahui apakah aplikasi telah berjalan dengan baik dan sesuai harapan.

## 1.7 Sistematika Penulisan

Sistematika penulisan laporan penelitian tugas akhir ini terdiri dari beberapa bab yang akan diuraikan sebagai berikut:

### **BAB I PENDAHULUAN**

Bab satu menjelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penulisan.

### **BAB II LANDASAN TEORI**

Bab dua menjelaskan tentang review dari beberapa *website* yang memiliki tujuan yang sama dengan aplikasi yang akan dibuat. Bab dua ini juga menjelaskan tentang tabel perbandingan antara aplikasi dan menjelaskan teori dasar dalam menulis buku cerita yang baik. Pada bab ini juga akan menjelaskan fitur pembeda dari aplikasi lain, yaitu fitur rekomendasi *asset*, dan studi literatur yang mendukung dalam pengembangan aplikasi pada penelitian ini.

### **BAB III METODOLOGI PENELITIAN**

Bab tiga berisi tentang analisis sistem berdasarkan uraian dari bab satu, review dari *website* pembuat buku cerita. Pada bab ini berisi perancangan dari aplikasi berdasarkan hasil analisis yang dilakukan, perancangan desain *database*, perancangan *use case diagram*, *activity diagram*, *flowchart* cara kerja algoritma, dan perancangan antarmuka.

### **BAB IV IMPLEMENTASI DAN PENGUJIAN APLIKASI**

Bab ini berisi tentang implementasi atau pengkodean aplikasi yang dibuat. Aplikasi yang dibuat berbasis *website* dan berdasarkan hasil perancangan pada bab tiga. Pada bab ini juga berisi tentang pengujian yang dilakukan untuk mengetahui kelebihan dan kekurangan sistem.

## **BAB V KESIMPULAN DAN SARAN**

Pada bab ini berisis kesimpulan dari pembahasan bab-bab sebelumnya dan saran untuk perbaikan dan pengembangan aplikasi yang mungkin dapat dilakukan di masa yang akan datang.

## BAB II LANDASAN TEORI

### 2.1 *Review Website Menulis*

Terdapat beberapa *website* yang dapat menulis buku cerita. *Website* tersebut dapat dijadikan sebagai tolak ukur dalam pembuatan aplikasi pembuat buku cerita pada penelitian tugas akhir ini. Berikut merupakan *review website* tersebut.

#### 2.1.1 **StoryJumper.com**

Storyjumper merupakan situs yang memberikan para guru, pelajar, para orang tua, dan penulis suatu *tools* yang intuitif untuk menulis dan mengilustrasikan suatu cerita. Tujuan dari Storyjumper ialah menginspirasi siapa saja yang ingin menulis sebuah cerita yang memiliki ilustrasi (Storyjumper, 2018). Storyjumper tidak memberikan batas kepada siapa saja untuk dapat menulis dan juga membaca, sehingga siapa saja yang terdaftar dapat membaca setiap cerita yang dituliskan oleh orang lain. Namun situs Storyjumper tidak menyediakan berbagai bahasa untuk digunakan, sehingga setiap pengguna harus mengetahui bahasa Inggris untuk dapat menggunakannya.

Beberapa fitur yang terdapat pada Storyjumper adalah sebagai berikut:

- a. Di halaman utama terdapat Kolom Pencarian yang memberikan cara kepada pengguna untuk dapat mencari buku cerita dengan kata kunci dari pencarian.
- b. Pada halaman utama pada pengguna yang sedang *login* menampilkan 4 bagian, yakni *My Personal Books* yang menampilkan hasil buku cerita yang telah ditulis oleh pengguna yang sedang *login*, *Enjoy These Books* yang menampilkan buku-buku cerita pilihan yang memiliki rating paling tinggi, *Recently Published Author* yang menampilkan buku cerita yang paling terbaru, dan bagian terakhir adalah *Books I Like* yang menampilkan buku-buku cerita yang di favoritkan pengguna yang sedang *login*.
- c. Terdapat 2 jenis pengguna yakni pengguna yang terdaftar (Anggota) dan pengguna yang tidak terdaftar (Pengunjung).

##### 1. Pengunjung

Pengunjung dapat mencari dan membaca buku cerita yang ditulis. Tampilan yang diberikan Storyjumper pada pengunjung saat membuka situs yakni menampilkan kolom

pencarian, menampilkan fitur-fitur dari Storyjumper, dan menampilkan kategori dari buku-buku cerita yang ada.

## 2. Anggota

Pengguna dapat mendaftar menjadi anggota dengan memilih *Sign in with Google*, *Login with Facebook*, atau dengan cara mendaftarkan data diri pada formulir yang disediakan pada Storyjumper. Dengan terdaptarnya pengguna sebagai anggota, pengguna tersebut dapat membaca, menulis, memberikan *like* pada buku cerita, memberikan komentar pada buku cerita dan dapat membeli buku sesuai dengan harga yang telah ditentukan. Saat proses pembuatan cerita, Storyjumper menyediakan berbagai fitur menarik, seperti pengguna dapat memberikan teks pada cerita, properti yang mendukung jalannya cerita, latar belakang yang banyak, dan fitur untuk menambahkan suara yang dapat menjadi pengguna sebagai narator dalam cerita.

### 2.1.2 MyStoryBook.com

Mystorybook merupakan situs gratis yang memberikan pengguna atau pelajar untuk dapat membuat buku cerita sendiri, mendesainnya dan membaginya secara online. Terdapat juga pilihan untuk mencetak buku cerita (Liwacloud.wordpress, 2016). Tujuan dari Mystorybook adalah menyediakan lingkungan yang kreatif dan artistic, memberikan konsep yang terfokus pada kanak-kanak pada gambar dan teks, dan menyediakan *tools* bagi anak-anak untuk dapat berkreasi dengan cara mereka sendiri.

Mystorybook memiliki fitur yang hamper sama seperti Storyjumper, hanya saja Mystorybook sedikit lebih terfokus pada anak-anak karena memiliki tampilan yang berwarna dan lebih menarik. Berikut adalah fitur-fitur yang diberikan oleh Mystorybook:

- a. Saat masuk ke situs web Mystorybook.com, pengguna sudah dapat langsung membuat buku cerita tanpa perlu mendaftar sebagai anggota.
- b. Mystorybook menyediakan berbagai macam pilihan properti, seperti karakter, gambar, teks dan lukisan.
- c. Mystorybook menyediakan pilihan untuk dapat mencetak buku cerita dalam bentuk PDF, sehingga pengguna dapat membaca buku cerita dalam keadaan offline. Namun untuk menggunakan fitur ini, pengguna mengharuskan membayar \$5 untuk setiap buku cerita yang ingin dicetak.



### 2.1.3 Storybird.com

Storybird merupakan situs *visual storytelling* yang memberikan siapa saja untuk dapat membuat cerita visual dalam beberapa detik. Storybird memberikan berbagai macam *artwork* dari *animators* dan *illustrator* dari setiap penjuru dunia untuk dapat menginspirasi penulis dalam menulis suatu cerita dengan *artwork* yang unik (Storybird, 2018). Storybird juga merupakan situs komunitas yang artinya pengguna dapat saling mengikuti dengan memilih pilihan *Subscribe*. Berikut ini beberapa fitur-fitur yang terdapat pada Storybird:

- a. Storybird memiliki banyak kategori, dan pengguna dapat mencari cerita berdasarkan umur dan kategori dari cerita. Pengguna juga dapat mencari cerita berdasarkan *tags* yang dimiliki suatu cerita.
- b. Storybird memiliki 3 format cerita yakni; *Longform Book* untuk membuat buku yang memiliki cerita yang panjang (*multi-chapter*), *Picture Book* untuk membuat cerita yang memiliki halaman yang banyak (*multi-page*), dan *Poem* untuk membuat cerita dengan satu gambar (*single-page*). Umumnya untuk pilihan *Poem* digunakan dengan menulis suatu puisi atau kutipan dengan ilustrasi gambar yang mendukung.
- c. Pada halaman utama bagi pengguna yang belum terdaftar, pengguna dapat langsung membaca cerita, dan mencari cerita dari berbagai macam kategori.
- d. Jika pengguna telah terdaftar, maka pada halaman utama akan ditampilkan *dashboard* untuk pengguna yang menampilkan cerita pengguna yang belum terselesaikan, pengguna-pengguna yang telah di *subscribe*, dan cerita pengguna yang telah tersimpan.
- e. Pada bagian pembuatan cerita, awalnya pengguna akan diminta untuk memilih tipe art atau gaya seni cerita dari gaya-gaya seni yang telah disediakan, lalu pengguna harus memilih format cerita apa yang akan dibuat, setelah itu pengguna akan diberikan berbagai macam gambar ilustrasi untuk mendukung cerita sesuai dengan gaya seni yang dipilih.

Tabel 2.1 merupakan table *review* perbandingan antara 3 *website* yang telah dijelaskan di atas. Yang menjadi perbandingan ialah fitur-fitur utama dan fitur-fitur pendukung yang terdapat pada *website* dengan yang tidak terdapat pada *website*.

Tabel 2.1 Perbandingan *Website* Menulis Cerita

	<b>Storyjumper</b>	<b>Mystorybook</b>	<b>Storybird</b>
<b>Pengguna</b>			
Registrasi pengguna	Ada	Ada	Ada
Notifikasi	Tidak ada	Tidak ada	Ada
Halaman akun	*Ada	Tidak ada	*Ada
Edit akun	*Ada	Tidak ada	*Ada
Buat cerita	*Ada	Ada	*Ada
Baca cerita	Ada	Ada	Ada
Subscribe pengguna lain	Tidak ada	Tidak ada	*Ada
<b>Pencarian cerita</b>			
Kotak pencarian	Ada	Tidak ada	Ada
Kategori cerita	Ada	Tidak ada	Ada
<b>Halaman membuat cerita</b>			
Menambah teks	*Ada	Ada	*Ada
Menambahkan gambar/properti	*Ada	Ada	*Ada
Menggambar pada cerita	Tidak ada	Ada	Tidak ada
Simpan cerita	*Ada	*Ada	*Ada
Cetak cerita	*Ada	*Ada	*Ada
<b>Kelebihan lain</b>			
Bahasa Indonesia	Tidak ada	Tidak ada	Tidak ada
Tutorial	Ada	Ada	*ada
Iklan	Tidak ada	Tidak ada	Tidak ada

\*) setelah terdaftar dan melakukan *login*

## 2.2 Studi Literatur

Studi Literatur adalah mencari referensi teori yang mendukung untuk kasus permasalahan pada penelitian ini. Kegiatan yang berkenaan dengan pengumpulan data, membaca dan mengolah referensi ilmu.

### 2.2.1 Penulisan Buku Cerita Yang Baik

Dalam penulisan buku cerita terdapat beberapa faktor yang mempengaruhi baik buruknya suatu cerita tersebut. Selain jalan cerita yang baik, faktor lain seperti gambar juga menjadi pendukung untuk jalannya suatu cerita, dan masih banyak lagi faktor-faktor pendukung lainnya. Berikut adalah penjelasan faktor-faktor pendukung tersebut:

a. Penulisan cerita

Materi cerita yang dimaksud adalah antara lain panjang pendeknya suatu bacaan, kepadatan informasi, derajat kompleksitas materi, familiaritas kosakata, dan karakter fisik bacaan (Sentari, 2002).

b. Materi cerita

Cerita yang menarik akan mempengaruhi pemahaman membaca dan akan membuat pembaca semakin mudah menguasai bacaan. Pada sebuah penelitian menunjukkan bahwa siswa cenderung untuk *me-recall* informasi dari bacaan yang menarik tetapi tidak penting, daripada *me-recall* informasi yang tidak menarik tetapi penting (Sentari, 2002).

c. Presentasi cerita

Presentasi cerita yang dimaksud yakni bagaimana suatu cerita diberikan kepada pembaca. Hal ini menyangkut pada tampilan latar, warna latar, jenis huruf, dan warna huruf. Menurut Soetopo (2007), kesulitan-kesulitan yang timbul pada saat membaca disebabkan oleh tampilan atau desain dari bacaan tersebut, sehingga pembaca sulit untuk mengelola informasi pada bacaan. Desain yang menyangkut pembubuhan warna pada rancangan akan memberikan kesan tampak menarik, tidak membosankan karena tampilan bacaan yang baik akan berdampak positif bagi pembaca dan dapat menjadikan membaca sebagai kegiatan yang menyenangkan serta tidak menimbulkan perasaan cemas atau ambigu (Soetopo, 1997).

Faktor-faktor tersebut merupakan hal-hal yang perlu ditekankan pada saat menuliskan sebuah cerita. Maka dari itu aplikasi yang dibuat harus dapat memberikan dukungan dalam menangani faktor-faktor tersebut. Aplikasi yang dibuat harus dapat memperkenankan pengguna agar dapat menuliskan teks sesuai keinginan pengguna, memberikan warna-warna yang menarik, dan dapat memberikan *asset* yang menarik dan mendukung dalam cerita yang dituliskan.

### 2.2.2 Rencana Fitur Unggulan

Selain berbahasa Indonesia, sebagai pembeda antara aplikasi yang akan dibangun dengan aplikasi serupa yang telah ada, diberikan fitur tambahan yaitu fitur rekomendasi *asset*. Fitur tersebut terpilih karena dalam mengembangkan imajinasi diperlukan lingkungan yang baik sebagai pemicu dalam meningkatkan daya imajinasi pengguna.

Probowati (2014) dalam penelitiannya menjelaskan tentang bagaimana meningkatkan kemampuan menulis puisi melalui kegiatan sugesti imajinasi pada anak. Dari hasil penelitiannya dapat diketahui bahwa telah terjadi peningkatan hasil belajar dalam pembelajaran menulis puisi pada siswa kelas V SDN Blimbing 3 Malang dengan menerapkan metode sugesti imajinasi. Berdasarkan hasil penelitian tersebut dapat diketahui bahwa sugesti atau rekomendasi dalam pembelajaran dapat meningkatkan daya imajinasi anak didiknya sesuai dengan perkembangannya.

Imajinasi anak bergantung pada lingkungan dimana anak bertempat tinggal, maka dalam hal ini lingkungan anak dapat digambarkan pada sistem aplikasi. Ketika lingkungan aplikasi baik maka hal ini akan memicu imajinasi anak. Dengan adanya fitur rekomendasi *asset* yang berbasis pada tulisan yang ditulis maka imajinasi anak akan berkembang dan dapat membantu dalam penulisan cerita.

Dalam mewujudkan fitur rekomendasi *asset*, akan diterapkan *text pre-processing* untuk mengelola data teks yang dituliskan oleh pengguna. *Text pre-processing* atau pemrosesan teks merupakan suatu proses perubahan data berupa teks yang belum terstruktur menjadi data yang terstruktur sesuai dengan kebutuhan. Tahap *pre-processing* atau praproses data merupakan proses untuk mempersiapkan data mentah sebelum dilakukan proses lain (Siti, 2016). Intinya, *pre-processing* adalah mengubah teks menjadi *term index* yang tujuannya adalah untuk menghasilkan sebuah *set term index* yang dapat mewakili suatu dokumen. Dengan menggunakan metode *text pre-processing* diharapkan dapat memudahkan dalam pencarian *asset* dari hasil *index* yang telah diproses. Untuk selengkapnya tentang cara kerja fitur rekomendasi *asset* akan dijelaskan lebih lengkap pada Bab 3.

## BAB III METODOLOGI PENELITIAN

### 3.1 Analisis Kebutuhan

Analisis kebutuhan merupakan tahap untuk proses pengumpulan data yang dibutuhkan untuk dapat digunakan sebagai pendukung aplikasi yang akan dibuat.

#### 3.1.1 Perencanaan Sistem

Pada tahap ini dilakukan identifikasi data dan informasi terhadap apa yang dibutuhkan pada sistem dalam merancang sebuah aplikasi. Data yang digunakan adalah data dari *review* aplikasi yang serupa dengan aplikasi yang direncanakan akan dibangun. Dari hasil *review* aplikasi diambil poin-poin penting terkait fitur yang terdapat pada aplikasi serupa dan dijadikan poin fitur yang mungkin akan dibangun dengan aplikasi yang direncanakan.

Berdasarkan hasil perencanaan yang dilakukan diperoleh gambaran dalam pembuatan sistem aplikasi pembuat buku cerita berbasis web yang ditampilkan pada Tabel 3.1.

Tabel 3.1 menampilkan fitur-fitur yang diambil dari Tabel 2.1 yang merupakan hasil *review website* aplikasi menulis cerita. Menentukan fitur-fitur tersebut berdasarkan dari fitur *website* aplikasi menulis yang umum digunakan seperti membuat cerita, membaca cerita, berbahasa Indonesia, menambahkan teks, menggambar pada cerita, menambahkan gambar/*asset* dan mencetak cerita. Fitur yang tersedia yakni fitur notifikasi, kategori cerita, dan *subscribe* pada pengguna lain. Juga terdapat fitur rekomendasi *asset* yang akan memberikan rekomendasi *asset* untuk mendukung jalannya suatu cerita yang dituliskan.

Tabel 3.1 Tabel Gambaran Fitur Yang Tersedia di Aplikasi

	<b>Rencana Fitur</b>	<b>Keterangan</b>
<b>Pengguna</b>		
Registrasi pengguna	Ada	
Notifikasi	Tidak ada	
Halaman akun	*Ada	Halaman profil dan Home bagi pengguna
Edit akun	*Ada	

Buat cerita	*Ada	Halaman yang berisi fitur aplikasi untuk membuat cerita
Baca cerita	Ada	
Subscribe pengguna lain	Tidak ada	
<b>Pencarian cerita</b>		
Kotak pencarian	Ada	
Kategori cerita	Tidak ada	
<b>Halaman membuat cerita</b>		
Menambah teks	*Ada	
Menambahkan gambar/properti	*Ada	
Menggambar pada cerita	*Ada	
Simpan cerita	*Ada	
Cetak cerita	*Ada	
<b>Kelebihan lain</b>		
Bahasa Indonesia	Tidak ada	
Tutorial	Ada	
Iklan	Tidak ada	
<b>Fitur tambahan</b>		
Rekomendasi Asset	*Ada	Memberikan rekomendasi <i>asset</i> ketika pengguna menuliskan teks cerita
Cek kata pada cerita	*Ada	Mengecek setiap kata jika terdapat kata yang masuk ke dalam kata yang terlarang
Menambahkan <i>asset</i>	*Ada	*ada
Iklan	Tidak ada	Tidak ada

\*) setelah menjadi anggota dan melakukan *login*

### 3.1.2 Analisis Kebutuhan Perangkat Lunak

Analisis perangkat lunak terdiri dari spesifikasi minimum perangkat lunak yang digunakan dalam membangun dan mengimplementasikan aplikasi pada penelitian ini. Kebutuhan-kebutuhan tersebut yaitu:

- a. Sistem Operasi Windows 2003, XP, 7, 8, 10
- b. Laragon 3.1.8.
- c. PHP 7.1
- d. Visual Code 1.72.2
- e. *Framework* Laravel 5.6
- f. *Library* Fabric.js 2.4.1

### 3.1.3 Analisis Kebutuhan Perangkat Keras

Analisis perangkat lunak terdiri dari spesifikasi minimum perangkat keras yang dibutuhkan untuk menjalankan aplikasi yang mendukung dalam membangun dan mengimplementasikan aplikasi pada penelitian ini. Kebutuhan-kebutuhan tersebut terdiri dari:

- a. Perangkat Keras Aplikasi pada bagian *Backend*
  - Processor : Minimum 2.8 GHz
  - Memori : 256 MB RAM
  - Hardisk : Minimum 350 MB
- b. Perangkat Keras Aplikasi *Frontend*
  - Resolusi : 1280 x 720
  - Refresh rate: 60 Hz
  - Bit depth : 8-bit

### 3.1.4 Analisis Pengguna

Pengguna yang dapat menggunakan aplikasi ini adalah sebagai berikut:

a. Pengunjung

Pengunjung adalah pengguna yang tidak atau belum terdaftar pada aplikasi, dan tidak melakukan *login* akun. Pengunjung dapat hanya dapat mencari dan membaca buku cerita yang telah dituliskan oleh pengguna yang terdaftar.

b. Anggota

Anggota adalah pengguna yang telah melakukan registrasi. Registrasi dilakukan dengan mengisi data *form* apa aplikasi yang terdiri dari data nama, *email*, password, *username* dan alamat tempat tinggal. Setelah berhasil melakukan pendaftaran, selain dapat mencari dan membaca buku cerita, anggota juga dapat menggunakan fitur-fitur yang lebih seperti menulis cerita jika setelah melakukan *login* akun. *Login* dilakukan dengan anggota yang akan menampilkan halaman akun yang menampilkan buku-buku cerita yang telah dibuat dan memiliki pilihan untuk mencetak buku-buku cerita tersebut.

c. Admin

Admin adalah yang mengatur pengelolaan pengguna, *asset*, kategori dari *asset*, dan buku. Admin dapat menambahkan, menghapus, dan mengedit *asset*. Admin juga dapat menambah, menghapus, dan mengedit kategori *asset*. Kategori *asset* yang dimaksud adalah setiap kategori memiliki kategori, sehingga setiap kategori dapat memiliki banyak *asset*. Admin juga dapat mengelola pengguna dan buku dari pengguna, namun admin dalam hal ini hanya dapat menghapus pengguna atau buku. Admin tidak dapat mengedit data pengguna atau buku cerita dari pengguna.

## 3.2 Use case Diagram

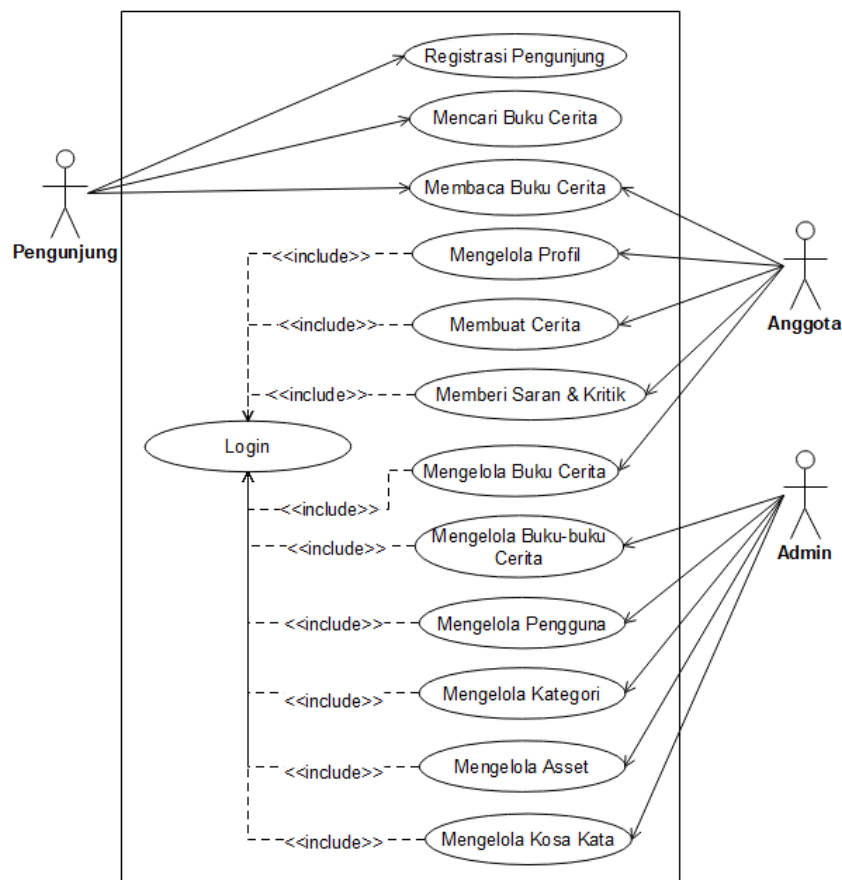
*Use case diagram* merupakan diagram yang digunakan untuk menggambarkan tugas yang dikerjakan oleh sistem dan aktor. Merupakan rancangan diagram *use case* yang dibuat dalam penelitian ini, terdapat dua actor yang berinteraksi dengan sistem aplikasi, yaitu pengunjung dan anggota.

Pengunjung yang mengunjungi *website* aplikasi dapat melakukan registrasi atau mendaftar untuk menjadi anggota, atau dapat langsung mencari buku dengan melakukan pencarian buku cerita. Registrasi adalah pendaftaran dengan mengisi *form* dengan data diri dari pengguna yang disediakan oleh sistem. Setelah terdaftar maka pengunjung telah menjadi anggota.



Pengunjung yang sudah terdaftar menjadi anggota, dapat melakukan *login*, kelola profil, menulis buku cerita, membaca buku cerita, dan mengelola buku cerita yang telah dibuat. Anggota dapat mengelola profil seperti mengubah alamat, foto profil, dan *username*. Jika anggota memilih untuk menulis suatu cerita, maka anggota dapat memilih pilihan “Buat Buku Cerita” dan aplikasi membawa anggota ke halaman membuat buku cerita. Pada halaman membuat buku cerita fitur-fitur yang ditawarkan yaitu mengubah latar belakang, menambahkan teks, menambahkan asset, dan dapat menggambar seperti menggunakan pensil. Setelah selesai membuat suatu cerita, anggota dapat menyimpan dan mempublikasikan buku tersebut agar dapat dibaca secara online oleh pengguna yang lain. Jika merasa terjadi kesalahan, pengguna juga dapat menghapus atau memperbarui buku cerita yang telah tersimpan.

Admin dapat mengelola user yakni admin dapat menghapus user yang diinginkan. Admin juga dapat mengelola buku cerita dan mengelola asset. Admin dapat menambahkan, mengedit atau menghapus suatu asset yang diinginkan. Admin juga dapat mengelola buku cerita, yakni membaca buku cerita dari pengguna dan menghapus buku cerita dari pengguna. Desain *use case diagram* dapat dilihat pada Gambar 3.1.



Gambar 3.1 Use case Diagram

### 3.2.1 Skenario dari *Use case Diagram*

#### 1. Registrasi Pengunjung (UC-1)

Nama use case : Registrasi Pengunjung

Aktor : Pengunjung

Pre-kondisi : Aktor belum memiliki akun

Post-kondisi : Aktor terdaftar sebagai anggota

Tabel 3.2 merupakan skenario *use case* registrasi pengunjung. Awal pengunjung mengakses aplikasi akan menemui halaman *index* yang terdiri dari beberapa tombol. Untuk melakukan pendaftaran/registrasi maka pengunjung harus memilih tombol “daftar” lalu mengisi form yang disediakan. Jika telah selesai maka sistem akan menyimpan data anggota dan akan menampilkan pada halaman beranda.

Tabel 3.2 Registrasi Pengunjung (UC-1)

Pengunjung	Sistem
1. Pengunjung memilih menu registrasi	
	2. Masuk ke halaman registrasi pengguna
3. Memasukkan data diri	
4. Klik daftar	
	5. Menambahkan anggota ke basisdata
	6. Menampilkan pesan pendaftaran berhasil
	7. Menampilkan halaman beranda

#### 2. *Login* (UC-2)

Nama use case : *Login*

Aktor : Anggota

Pre-kondisi : Aktor belum masuk ke sistem

Post-kondisi : Aktor masuk ke sistem pada halaman beranda

Tabel 3.3 merupakan skenario *use case login*. Pada *use case* ini aktor terlebih dahulu memilih halaman *login*, lalu memasukan data *login*. Jika data terdapat pada database maka sistem akan mengarahkan ke halaman beranda. Jika tidak terdapat data pada database maka sistem akan menampilkan info *error*.

Tabel 3.3 *Login* (UC-2)

<b>Pengunjung</b>	<b>Sistem</b>
1. Pilih menu <i>login</i> dengan menekan tombol <i>login</i>	
	2. Masuk ke halaman form <i>login</i>
3. Mengisi form <i>login</i> dengan memasukan email dan password	
4. Klik tombol masuk	
	5. Melakukan pengecekan pada anggota <i>database</i>
	6. Jika data benar akan diarahkan ke halaman beranda
	7. Jika data salah maka akan kembali ke halaman form <i>login</i>

## 3. Mencari Buku Cerita (UC-3)

Nama use case : Mencari Buku Cerita

Aktor : Pengunjung/Anggota

Pre-kondisi : Aktor belum melakukan pencarian

Post-kondisi : Aktor di halaman pencarian buku cerita

Tabel 3.4 merupakan skenario *use case* registrasi mencari buku cerita. Pada *use case* aktor dapat mencari buku dengan mengisi judul buku pada kotak pencarian. Jika terdapat data pada *database* maka sistem akan menampilkan buku yang dicari, jika tidak terdapat data maka sistem akan menampilkan pesan info pencarian gagal.

Tabel 3.4 Mencari Buku Cerita (UC-3)

<b>Pengunjung</b>	<b>Sistem</b>
1. Memasukan kata pencarian pada kotak pencarian	
	2. Mencari apakah terdapat judul buku dengan kata kunci pada <i>database</i>
	3. Jika terdapat data yang dicari, maka akan diarahkan pada halaman pencarian buku
	4. Jika tidak terdapat data maka akan mengarahkan ke halaman pencarian dengan notifikasi buku tidak ditemukan.

#### 4. Membaca Buku Cerita (UC-4)

- Nama use case : Membaca Buku Cerita  
 Aktor : Pengunjung/Anggota  
 Pre-kondisi : Aktor belum memilih buku cerita  
 Post-kondisi : Aktor di halaman membaca buku cerita

Tabel 3.5 merupakan skenario *use case* membaca buku cerita. Pada *use case* ini di asumsikan telah terdapat buku cerita yang telah selesai dibuat. Aktor dapat memilih buku cerita dan sistem akan memuat buku cerita tersebut untuk dapat dibaca.

Tabel 3.5 Membaca Buku Cerita (UC-4)

Pengunjung	Sistem
1. Memilih buku cerita yang akan dibaca	
	2. Mengarahkan ke halaman membaca dan memuat buku cerita

#### 5. Mengelola Profil (UC-5)

- Nama use case : Mengelola Profil  
 Aktor : Anggota  
 Pre-kondisi : Aktor telah melakukan *login*  
 Post-kondisi : Terjadi perubahan data profil dari anggota

Tabel 3.6 merupakan skenario *use case* mengelola profil. Pada *use case* ini actor telah melakukan *login* dan berada pada halaman profil. Pada halama profil terdapat tombol untuk mengedit data profil dan data password. Sistem akan menampilkan modal form untuk mengisi data profil ketika tombol “edit profil” ditekan.

Tabel 3.6 Mengelola Profil (UC-5)

<b>Pengunjung</b>	<b>Sistem</b>
1. Memilih halaman profil	
	2. Menampilkan halaman profil
3. Memilih tombol edit profil	
	4. Menampilkan form edit profil.
5. Memperbarui data profil	
6. Klik tombol simpan	
	7. Memperbarui data.

## 6. Membuat Cerita

Nama use case : Membuat Cerita

Aktor : Anggota

Pre-kondisi : Aktor telah melakukan *login*

Post-kondisi : Anggota menambahkan buku cerita

Tabel 3.7 merupakan skenario *use case* membuat cerita. Setelah melakukan pendaftaran dan masuk pada halaman membuat cerita. aktor dapat langsung memulai membuat cerita dengan menggunakan fitur-fitur yang ada, seperti menggambar, menulis teks, dan menambahkan asset. Jika telah selesai aktor dapat menyimpan cerita untuk dapat membacanya dilain waktu.

Tabel 3.7 Membuat Cerita (UC-6)

<b>Pengunjung</b>	<b>Sistem</b>
1. Memilih halaman membuat cerita	
	2. Menampilkan halaman membuat cerita
3. Menulis suatu cerita.	
4. Menyimpan cerita dengan menekan tombol “simpan cerita”.	
	5. Menyimpan data buku di <i>database</i>
	6. Menampilkan info berhasil menyimpan.

## 7. Memberi Saran & Kritik

Nama use case : Memberi Saran & Kritik

Aktor : Anggota

Pre-kondisi : Aktor telah melakukan *login*

Post-kondisi : Anggota mengirimkan kritik & saran

Tabel 3.8 merupakan skenario *use case* memberi kritik & saran. Di asumsikan aktor telah terdaftar dan telah melakukan *login*. Aktor dapat memberi kritik & saran kepada admin dari sistem dengan cara mengisi form yang disediakan pada halaman kritik & saran. Data akan disimpan pada database dan akan ditampilkan pada halaman admin.

Tabel 3.8 Memberi Kritis & Saran (UC-7)

Pengunjung	Sistem
1. Memilih halaman form kritik & saran	
	2. Menampilkan halaman form kritik & saran
3. Mengisi form kritik & saran	
4. Menekan tombol kirim	
	5. Menyimpan data kritik & saran ke <i>database</i>

## 8. Mengelola Buku Cerita

Nama use case : Mengelola Buku Cerita

Aktor : Anggota

Pre-kondisi : Aktor telah melakukan *login*

Post-kondisi : Terjadi perubahan data buku cerita

Tabel 3.9 merupakan skenario *use case* mengelola buku cerita. Anggota dapat kembali, menghapus, atau mencetak dalam bentuk pdf.

Tabel 3.9 Mengelola Buku Cerita (UC-8)

Pengunjung	Sistem
1. Memilih pilihan dari halaman “buku saya”	
	2. Menjalankan perintah baik itu mengedit, menghapus, atau mencetak buku cerita.

### 9. Mengelola Buku-buku Cerita

Nama use case : Mengelola Buku-buku Cerita

Aktor : Admin

Pre-kondisi : Aktor telah melakukan *login*

Post-kondisi : Terjadi perubahan data buku cerita

Tabel 3.10 merupakan skenario *use case* mengelola buku-buku cerita. Admin dapat mengelola setiap buku-buku yang telah dibuat oleh anggota. Admin dapat menyetujui cerita untuk dipublikasi, menghapus buku dari buku-buku yang ada dan dapat mencetak buku milik pengguna, dan membaca buku dari pengguna.

Tabel 3.10 Mengelola Buku-buku Cerita (UC-9)

Pengunjung	Sistem
1. Memilih menu halaman “manajemen buku”	
	2. Menampilkan halaman “manajemen buku”
3. Memilih pilihan pada halaman “manajemen buku”	
	4. Menjalankan perintah menghapus, membaca ataupun mencetak.

### 10. Mengelola Pengguna

Nama use case : Mengelola Pengguna

Aktor : Admin

Pre-kondisi : Aktor telah melakukan *login*

Post-kondisi : Terjadi perubahan pada *database*

Tabel 3.11 merupakan skenario *use case* mengelola pengguna. Yang dimaksud mengelola pengguna yaitu admin dapat menghapus anggota tanpa sepengetahuan dari anggota, jika diketahui anggota telah melanggar dalam penggunaan sistem.

Tabel 3.11 Mengelola Pengguna (UC-10)

Pengunjung	Sistem
1. Memilih menu halaman “manajemen pengguna”	
	2. Menampilkan halaman “manajemen pengguna”
3. Memilih pilihan pada halaman “manajemen pengguna”	
	4. Menjalankan perintah dari pilihan admin.

## 11. Mengelola Kategori

Nama use case : Mengelola Kategori

Aktor : Admin

Pre-kondisi : Aktor telah melakukan *login*

Post-kondisi : Terjadi perubahan data kategori

Tabel 3.12 merupakan skenario *use case* mengelola kategori. Yang dimaksud mengelola kategori yaitu admin dapat menambah, menghapus atau mengedit data kategori. Kategori dalam hal ini adalah kategori dari asset, karena masing-masing asset ditentukan adalah bagian dari suatu kategori.

Tabel 3.12 Mengelola Kategori (UC-11)

Pengunjung	Sistem
1. Memilih menu halaman “manajemen kategori”	
	2. Menampilkan halaman “manajemen kategori”
3. Memilih pilihan pada halaman “manajemen kategori”	
	4. Menjalankan perintah dari pilihan admin, baik itu menghapus, membuat atau mengedit.

## 12. Mengelola Asset

Nama use case : Mengelola Asset

Aktor : Admin

Pre-kondisi : Aktor telah melakukan *login*

Post-kondisi : Terjadi perubahan pada data asset



Tabel 3.13 merupakan scenario use case mengelola asset. Yang dimaksud mengelola pengguna yaitu admin dapat menghapus anggota tanpa sepengetahuan dari anggota, jika diketahui anggota telah melanggar dalam penggunaan sistem.

Tabel 3.13 Mengelola Asset (UC-12)

Pengujung	Sistem
1. Memilih menu halaman “manajemen asset”	
	2. Menampilkan halaman “manajemen asset”
3. Memilih pilihan pada halaman “manajemen asset”	
	4. Menjalankan perintah dari pilihan admin, baik itu menghapus, membuat atau mengedit.

### 13. Mengelola Kosa Kata

Nama use case : Mengelola Kosa Kata

Aktor : Admin

Pre-kondisi : Aktor telah melakukan *login*

Post-kondisi : Terjadi perubahan data kosa kata pada *database*

Tabel 3.14 merupakan skenario *use case* mengelola kosa kata. Yang dimaksud mengelola kosa kata yaitu admin dapat menambahkan atau menghapus kosa kata pada *database*. Data pada tabel kosa kata dalam *database* akan digunakan untuk memfilter kata-kata pilihan yang tidak pantas pada buku cerita.

Tabel 3.14 Mengelola Kosa Kata (UC-13)

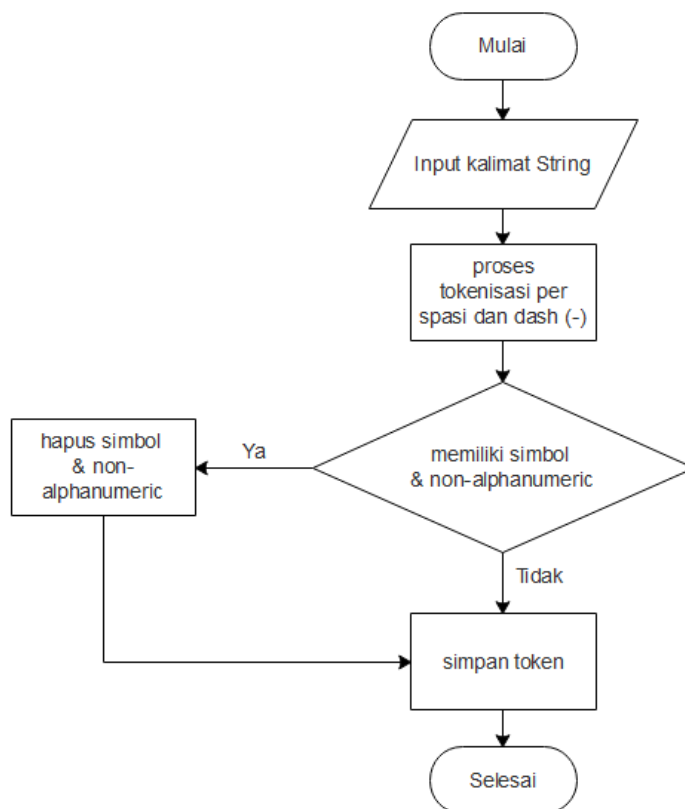
Pengujung	Sistem
1. Memilih menu halaman “manajemen kosa kata”	
	2. Menampilkan halaman “manajemen kosa kata”
3. Memilih pilihan pada halaman “manajemen kosa kata”	
	4. Menjalankan perintah dari pilihan admin.

### 3.3 Flowchart Diagram

Diagram *flowchart* digunakan untuk menggambarkan suatu algoritma, alur atau proses. Dalam penelitian ini *flowchart* diagram digunakan untuk menggambarkan alur algoritma dari metode *text pre-processing* yang diterapkan pada aplikasi.

#### a. Desain *Flowchart* Tokenisasi

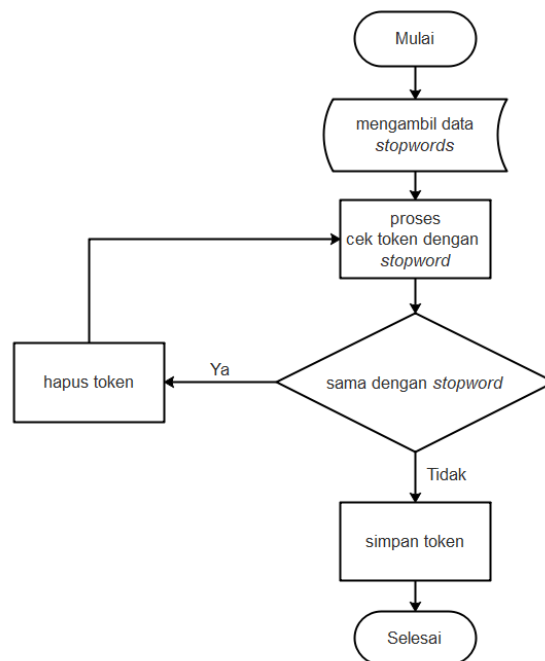
Tokenisasi adalah proses memilah suatu kalimat menjadi kata per kata. Umumnya kalimat yang berupa tipe string dipotong dengan batasan spasi. Setelah melakukan tokenisasi, kata yang didapatkan harus dibersihkan terlebih dahulu dari white space dan non-alphanumeric karakter sebelum digunakan agar tidak terjadi kesalahan dalam proses mining data teks. Gambar *flowchart* seperti pada Gambar 3.2 merupakan alur dari algoritma tokenisasi.



Gambar 3.2 *Flowchart* Tokenisasi

b. Desain *Flowchart Stopwords Removal*

*Stopwords removal* dapat diartikan yaitu menghapus kata-kata yang tidak penting. Setiap kata yang ditoken akan diperiksa apakah termasuk dari kata-kata yang tidak penting. Kata-kata yang tidak penting telah ditentukan dan disimpan pada suatu tabel di *database*. Kata-kata tersebut dapat disebut sebagai *stoplist*. Pada penelitian ini, data *stoplist* yang digunakan merupakan data dari hasil penelitian oleh Tala (2003), yang berupa kumpulan kata *stopword* seperti; ini, itu, yang, ke, dalam, dan sebagainya. Proses *stopword* yakni jika terdapat kata *stopword* pada data teks yang telah menjadi token, maka data tersebut akan dibuang. Sebaliknya jika data tersebut tidak termasuk ke dalam kata-kata *stopword* maka data tersebut akan diproses. Gambal *flowchart* alur algoritma *stopwords removal* terlihat seperti pada Gambar 3.3.



Gambar 3.3 *Flowchart Stopwords Removal*

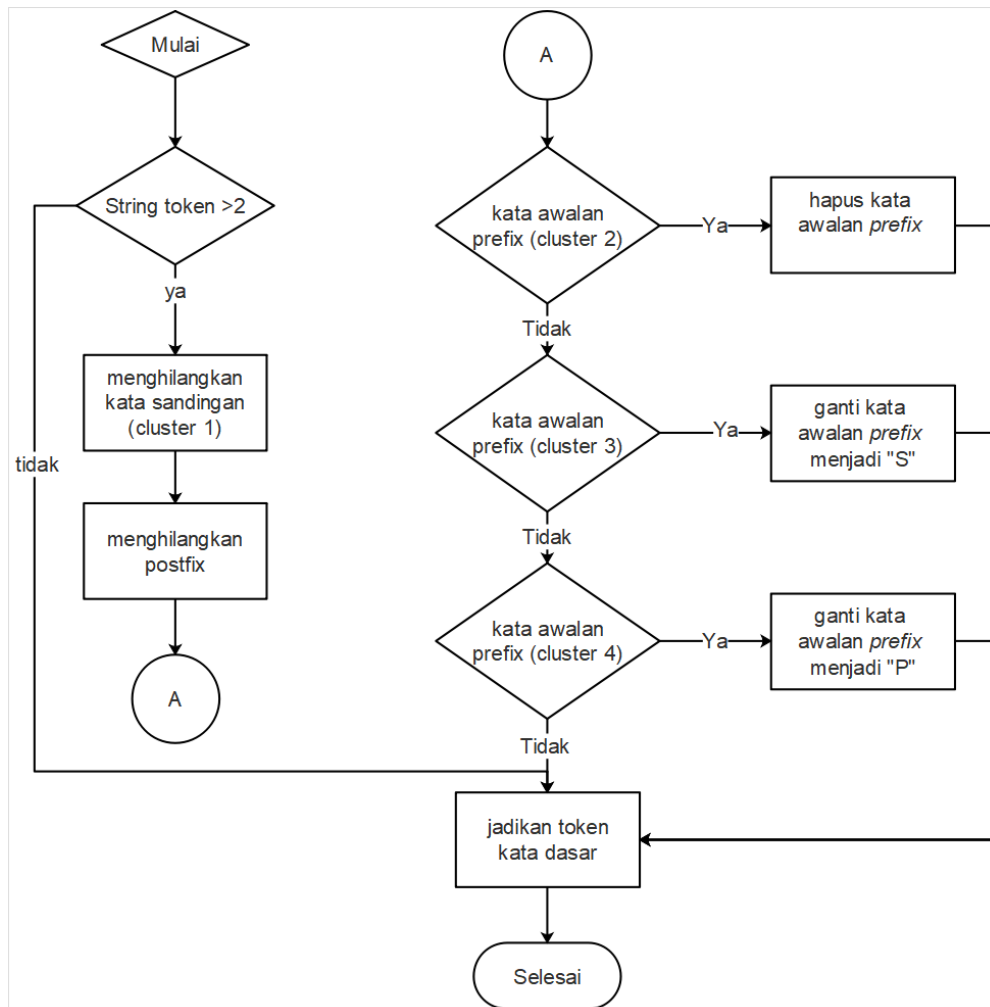
c. *Flowchart Stemming*

*Stemming* merupakan proses pembentukan kata dasar. Hal ini dilakukan untuk memecah kata untuk menjadikan sebagai kata dasar untuk menghindari ketidakcocokan pada saat mencari data teks. Pada penelitian ini akan digunakan algoritma *stemming* dari Tala (2003) yang menggunakan Bahasa Indonesia. *Stemming* Tala menggunakan algoritma *rule based stemming* seperti halnya algoritma Porter *stemming* dalam bahasa Inggris. Menurut Irfan, Apriana, dan Fadilla (2017) algoritma *stemming* Tala dapat digunakan bertujuan untuk

mempercepat waktu implementasi dan diharapkan performa sistem tetap stabil walaupun data dokumen teks bertambah terus. Pada penelitian ini telah ditentukan apa saja yang menjadi kata-kata sandingan dan juga kata-kata *prefix* yang kemudian nanti akan digunakan untuk mencari kata dasar. Berikut adalah pembagian kata sandingan dan kata *prefix* dalam bentuk *cluster*, dan gambaran *flowchart* prosesnya yang dapat dilihat pada Tabel 3.15 dan Gambar 3.4.

Tabel 3.15 Tabel Kata

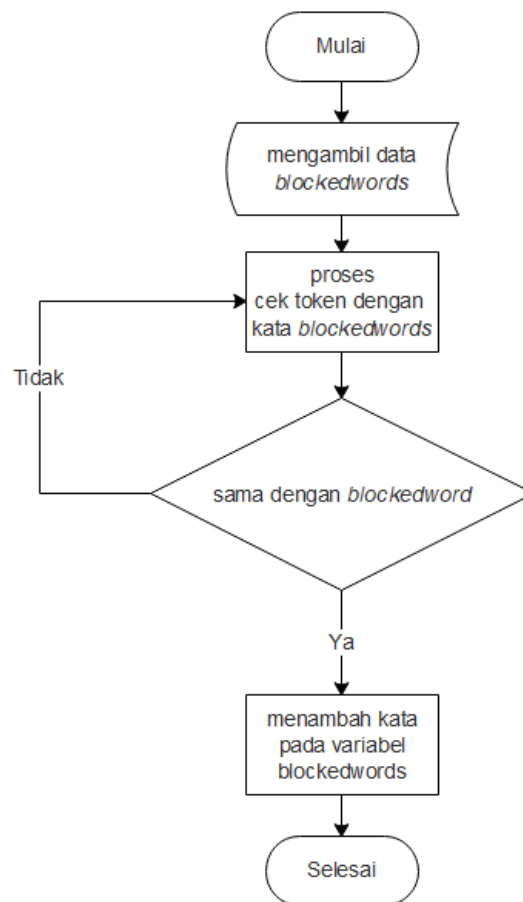
<b>Kata sandingan Diubah menjadi = “ “ (Cluster 1)</b>	<b>Kata prefix yang diubah menjadi = “ ” (Cluster 2)</b>	<b>Kata prefix yang diubah menjadi = “S” (Cluster 3)</b>	<b>Kata prefix yang diubah menjadi = “P” (Cluster 4)</b>
<b>lah</b>	meng	menya	mema
<b>kah</b>	mem	menyi	memi
<b>pun</b>	me	menye	memu
<b>ku</b>	peng	meny	meme
<b>mu</b>	pen	penya	memo
<b>nya</b>	pem	penyi	pema
<b>kan</b>	di	penyu	pepi
<b>an</b>	ter	penye	pemu
<b>i</b>	ke	penyo	peme
	ber	peny	
	bel		
	be		
	per		
	pel		
	pe		



Gambar 3.4 *Flowchart Stemming* pada aplikasi

d. *Flowchart Mengecek Blockedwords*

Pada proses ini dilakukan untuk mengecek apakah terdapat kata yang dilarang yang telah ditentukan oleh admin pada aplikasi. Jika terdapat kata yang dilarang maka kata tersebut akan disimpan dan akan ditampilkan pada admin di halaman manajemen buku milik admin. Gambar 3.5 berikut merupakan *flowchart* proses mengecek *blockedwords*.



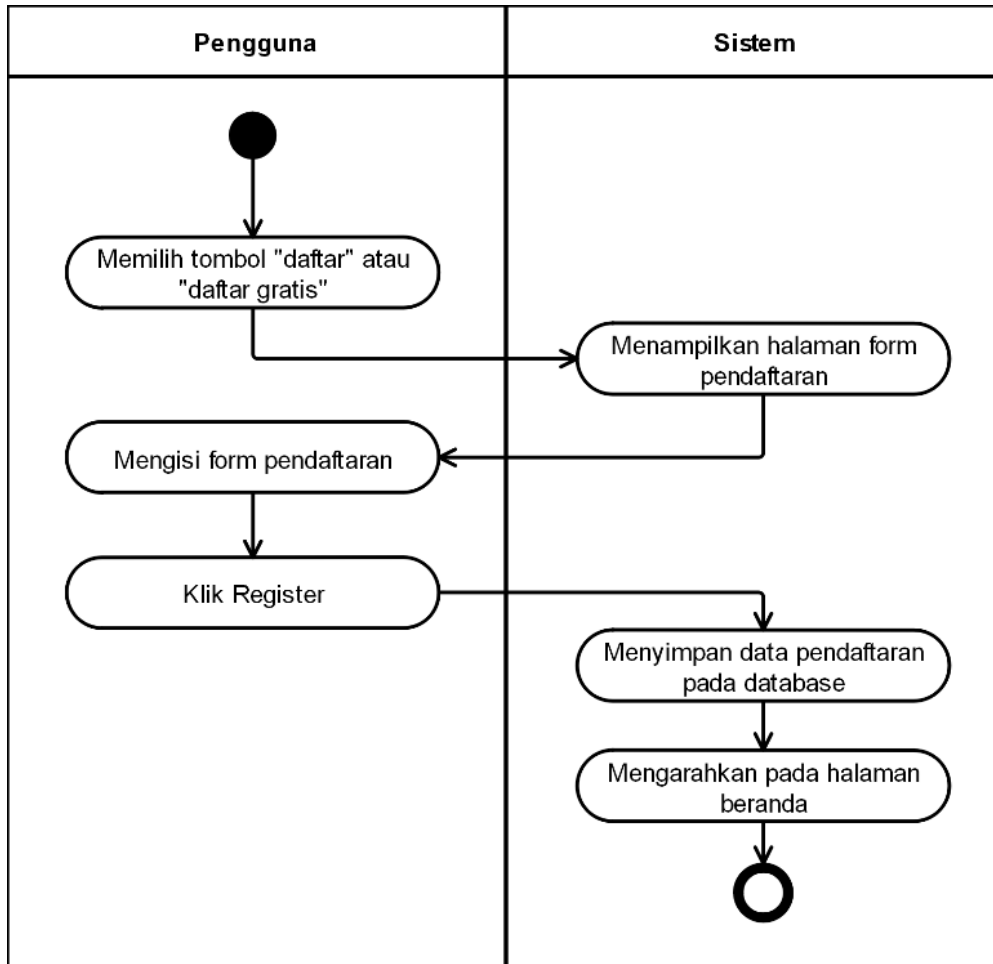
Gambar 3.5 *Flowchart Mengecek Blockwords*

### 3.4 Activity Diagram

*Activity* diagram dibuat untuk menggambarkan aliran aktivitas yang terjadi pada sistem di setiap *use case* yang ada. Pada desain *activity* diagram dibawah ini dibentuk berdasarkan desain *use case* yang sebelumnya telah dibentuk. Berikut merupakan desain *activity* diagram yang digunakan pada sistem.

#### 3.4.1 *Activity* Registrasi Pengunjung

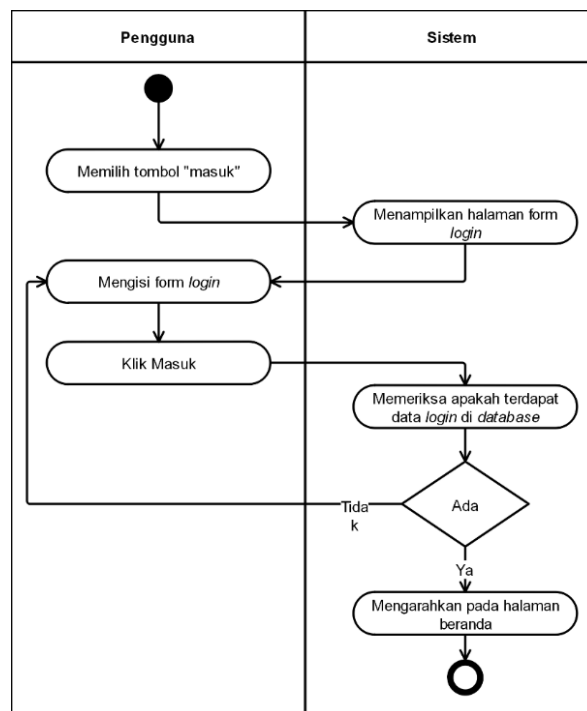
Gambar 3.6 merupakan desain gambar *activity* diagram dari *use case* registrasi pengunjung. Pengguna yang mengakses halaman awal dari sistem dapat melakukan pendaftaran dengan menekan tombol “daftar” atau tombol “daftar gratis”. Setelah itu sistem akan mengarahkan pada halaman yang terdapat form pendaftaran. Pengunjung memasukan data pendaftaran lalu menekan tombol “register”, kemudian sistem akan menyimpan data pendaftaran pada *database* dan akan mengarahkan anggota ke halaman beranda.



Gambar 3.6 *Activity* Diagram Registrasi Pengunjung

### 3.4.2 Activity Login

Gambar 3.7 merupakan gambar *activity* diagram dari *use case login*. Pengunjung yang telah terdaftar sebagai anggota dapat melakukan *login* dengan memilih tombol *login*, lalu sistem akan mengarahkan pada halaman *form login*, lalu anggota dapat mengisi data *login* dan menekan tombol masuk, lalu sistem akan mengecek data *login* pada *database*. Jika terdapat data *login* maka anggota akan diarahkan ke halaman beranda, jika tidak maka anggota akan diarahkan kembali ke halaman *form login*.

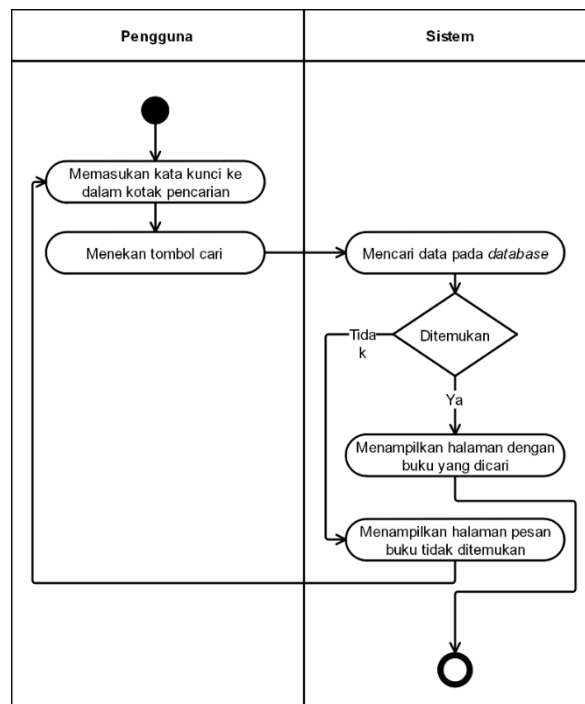


Gambar 3.7 Activity Diagram Login



### 3.4.3 Activity Mencari Buku Cerita

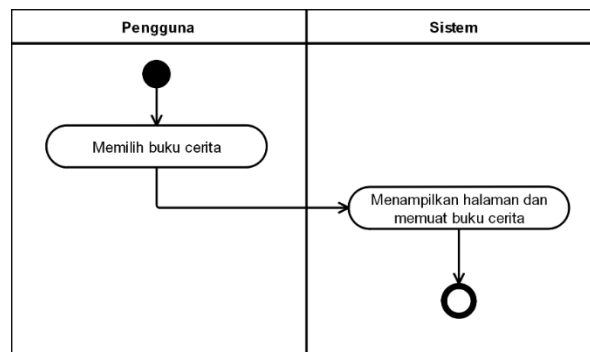
Gambar 3.8 adalah bentuk desain *activity* diagram dari *use case* mencari buku cerita. Dalam hal ini pengguna hanya tinggal memasukan kata kunci pada kotak pencarian, kemudian sistem akan mencari kata tersebut pada tabel buku dalam *database*. Jika data ditemukan maka akan menampilkan halaman berisi pencarian buku, namun jika tidak maka akan menampilkan halaman dengan pesan bahwa buku tidak ditemukan.



Gambar 3.8 Activity Diagram Pencarian Buku

### 3.4.4 Activity Membaca Buku Cerita

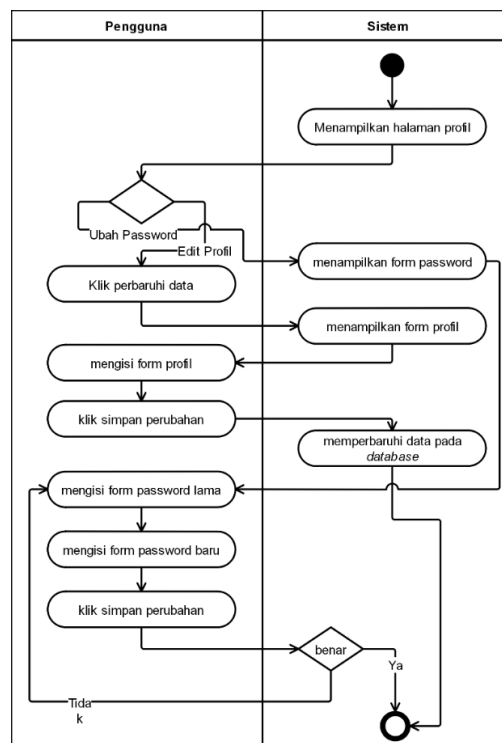
Gambar 3.9 adalah desain *activity* diagram dari *use case* membaca buku cerita. Aktivitas pada *use case* ini baik anggota ataupun pengunjung dapat membaca buku hanya dengan mengklik link dari buku cerita tersebut. Kemudian sistem akan memuat buku cerita.



Gambar 3.9 Activity Diagram Membaca Buku Cerita

### 3.4.5 Activity Mengelola Profil

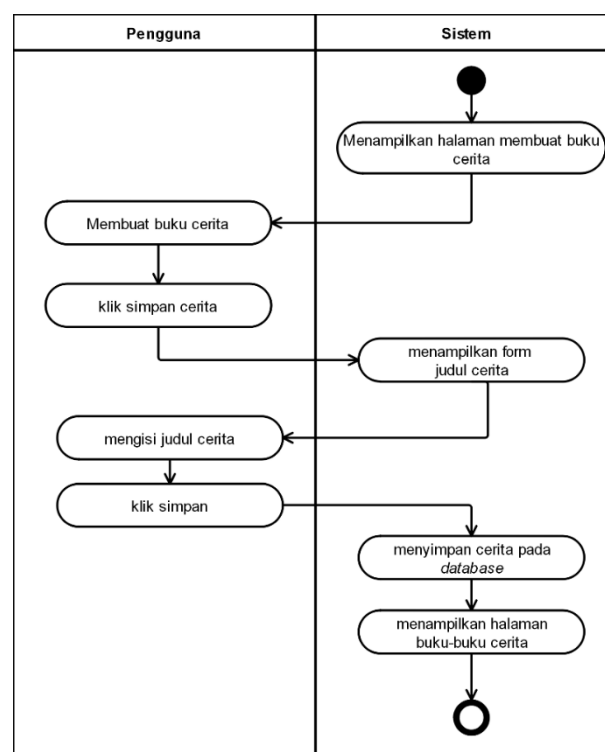
Gambar 3.10 adalah gambar *activity* diagram dari *use case* mengelola profil. Dalam hal ini diasumsikan anggota telah *login*, kemudian memilih halaman profil. Jika ingin mengubah data profil maka anggota dapat memilih tombol perbaruhi data, lalu sistem akan menampilkan form profil dan anggota dapat mengubah data. Jika ingin mengubah *password* maka anggota dapat memilih tombol ubah *password*, lalu sistem akan menampilkan form *password*. Anggota harus memasukan *password* lama terlebih dahulu kemudian *password* baru. Jika *password* lama benar, maka perubahan *password* berhasil dilakukan.



Gambar 3.10 Activity Diagram Mengelola Profil

### 3.4.6 Activity Membuat Cerita

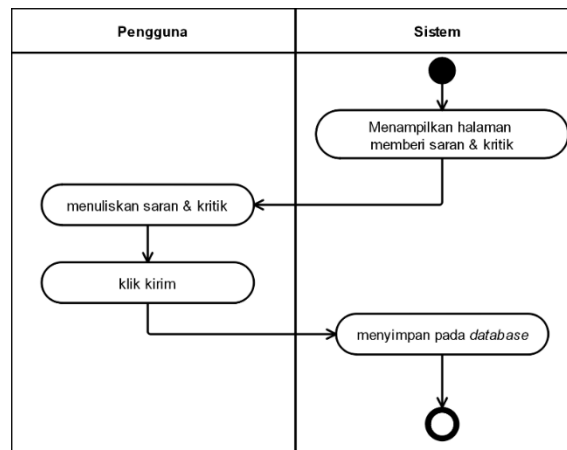
Gambar 3.11 merupakan *activity* diagram dari *use case* membuat cerita. Saat memasuki halaman membuat cerita, anggota langsung dapat membuat cerita dengan menggunakan fitur-fitur yang ada, seperti menuliskan teks, menggambar, ataupun memberikan asset. Jika telah selesai anggota dapat menyimpan buku cerita dengan memberikan judul buku cerita, kemudian sistem akan menyimpan data buku cerita, lalu anggota akan diarahkan ke halaman buku-buku cerita yang telah dibuat.



Gambar 3.11 Activity Diagram Membuat Cerita

### 3.4.7 Activity Memberi Saran & Kritik

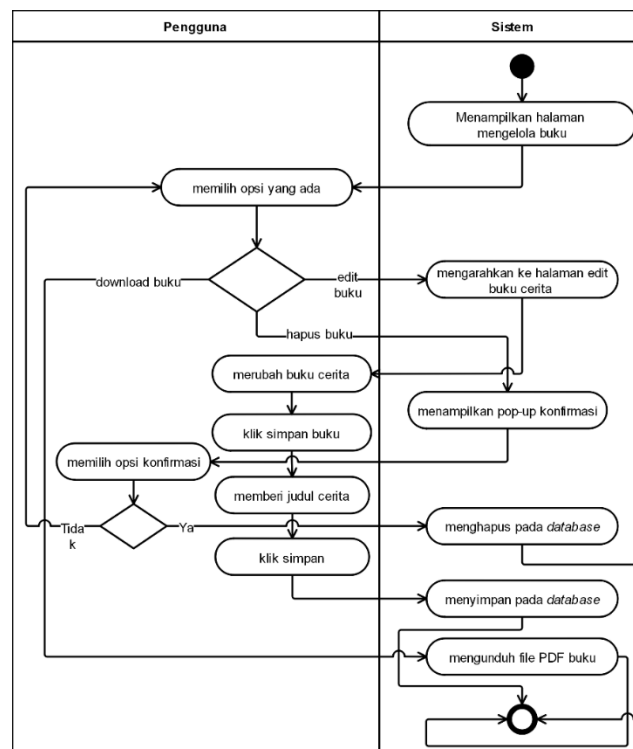
Terlihat pada Gambar 3.12 hanya terdapat sedikit aktivitas sederhana. Ketika sistem menampilkan halaman form kritik & saran anggota langsung dapat memberikan kritik & saran dengan mengisi form yang ada. Jika telah selesai maka anggota dapat menekan tombol kirim, lalu sistem akan menyimpan data ke *database* yang nantinya akan ditampilkan di halaman admin.



Gambar 3.12 Activity Diagram Memberi Saran & Kritik

### 3.4.8 Activity Mengelola Buku Cerita

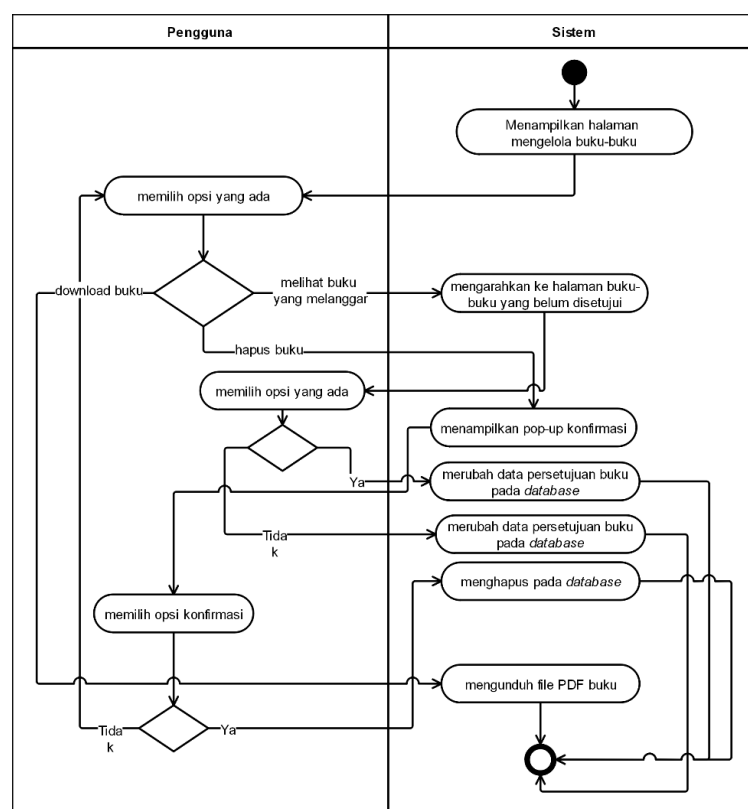
Gambar 3.13 adalah diagram *activity* mengelola buku cerita. Anggota dapat memilih pilihan yaitu, mengedit buku, menghapus buku, atau mengunduh buku. Ketika memilih mengedit buku anggota akan diarahkan ke halaman edit buku. Jika memilih menghapus buku maka akan diberi konfirmasi untuk menghapus buku. Jika memilih mengunduh buku maka sistem akan memberikan unduhan file berupa pdf yang berisi buku cerita.



Gambar 3.13 Activity Diagram Mengelola Buku

### 3.4.9 Activity Mengelola Buku-buku Cerita

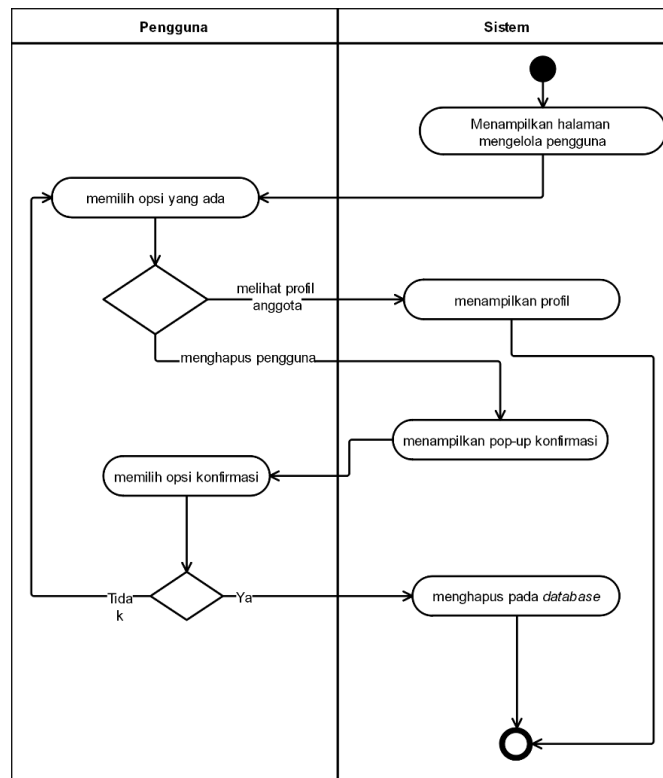
Gambar 3.14 pengguna yang dimaksud adalah admin. Pada *use case* ini admin dapat mengelola buku namun tidak sama dengan anggota mengelola buku. Dalam hal ini admin hanya dapat melihat buku yang memiliki pelanggaran, menghapus buku, atau mengunduh buku. Jika terdapat buku yang melanggar atau terdapat kata-kata yang tidak pantas dalam buku cerita anak-anak maka admin harus mengecek terlebih dahulu apakah buku tersebut layak dipublikasikan.



Gambar 3.14 Activity Diagram Mengelola Buku-buku Cerita

### 3.4.10 Activity Mengelola Pengguna

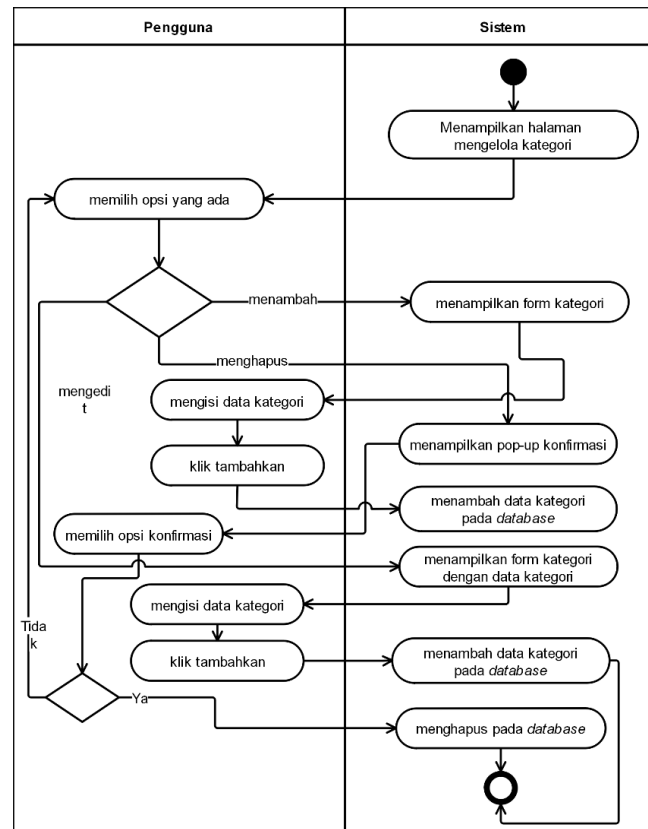
Gambar 3.15 merupakan desain *activity* diagram untuk use case mengelola pengguna. Admin dapat memilih pilihan opsi yang ada seperti, melihat profil anggota atau menghapus anggota juga dianggap anggota telah melanggar peraturan pada aplikasi.



Gambar 3.15 Activity Mengelola Pengguna

### 3.4.11 Activity Mengelola Kategori

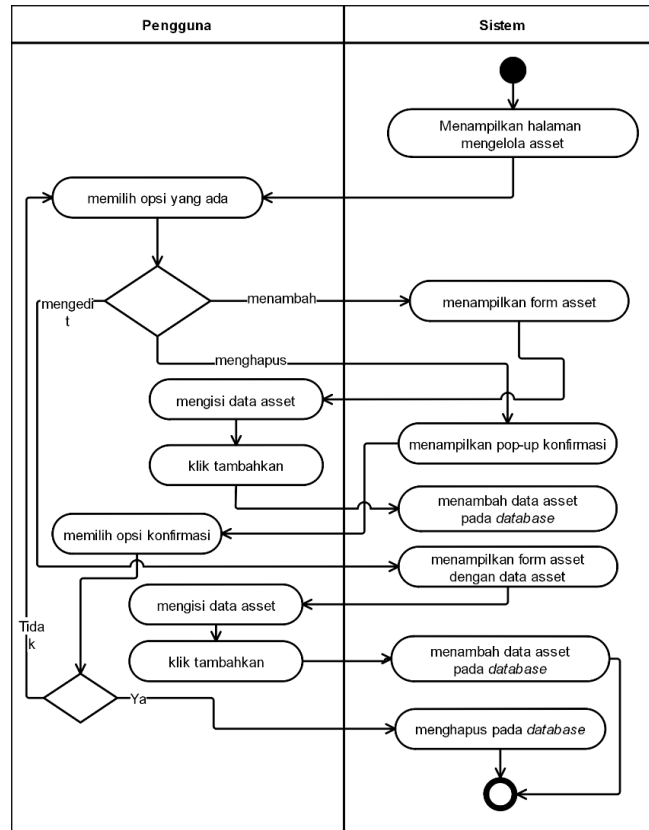
Gambar 3.16 merupakan desain *activity* diagram dari *use case* mengelola kategori. Dalam hal ini admin dapat mengelola kategori seperti menambahkan, mengedit, dan menghapus. Kategori ini nantinya akan digunakan dalam data asset sebagai *foreign key* pada *database*. Ketika admin ingin menambahkan, maka menekan tombol tambah lalu modal form akan tampil. Begitu juga dengan mengedit kategori yang sudah ada. Jika admin ingin menghapus kategori maka akan tampil *pop-up* konfirmasi kepada admin jika ingin benar-benar menghapus kategori.



Gambar 3.16 Activity Mengelola Kategori

### 3.4.12 Activity Mengelola Asset

Pada Gambar 3.17 memperlihatkan aktivitas yang terdapat pada *use case* mengelola asset. Fungsi-fungsi yang ada cukup sama dengan *use case* mengelola kategori. Admin dapat menambah, mengedit, dan menghapus asset. Ketika menambahkan asset, admin harus memilih kategori untuk asset, yang dimana kategori tersebut telah dibuat dan diambil dari *database*, dan admin harus mengunggah file dari asset tersebut. Ketika telah selesai, admin dapat mengklik tambahkan untuk menyimpan data asset ke dalam *database*.

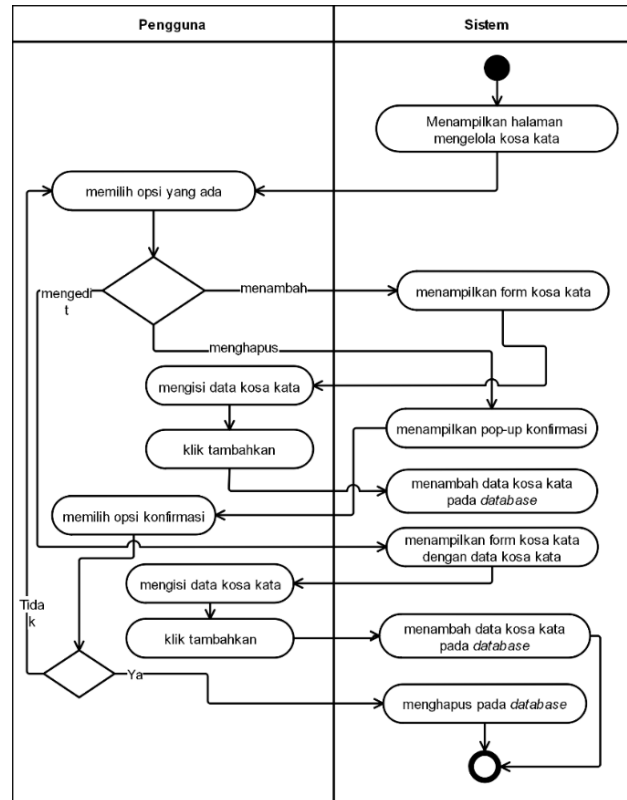


Gambar 3.17 Activity Diagram Mengelola Asset

### 3.4.13 Activity Mengelola Kosa Kata

Pada Gambar 3.18 memperlihatkan aktivitas yang terdapat pada *use case* mengelola kosa kata. Fungsi-fungsi yang ada juga cukup sama dengan *use case* mengelola kategori dan *use case* mengelola asset. Admin dapat menambah, mengedit, dan menghapus asset. Ketika telah selesai, admin dapat mengklik tambahkan untuk menyimpan data asset ke dalam *database*. Data dari kosa kata akan digunakan sebagai kata terlarang dalam cerita. Jika terdapat kata yang terlarang dari suatu cerita, maka cerita akan di periksa oleh admin terlebih dahulu.

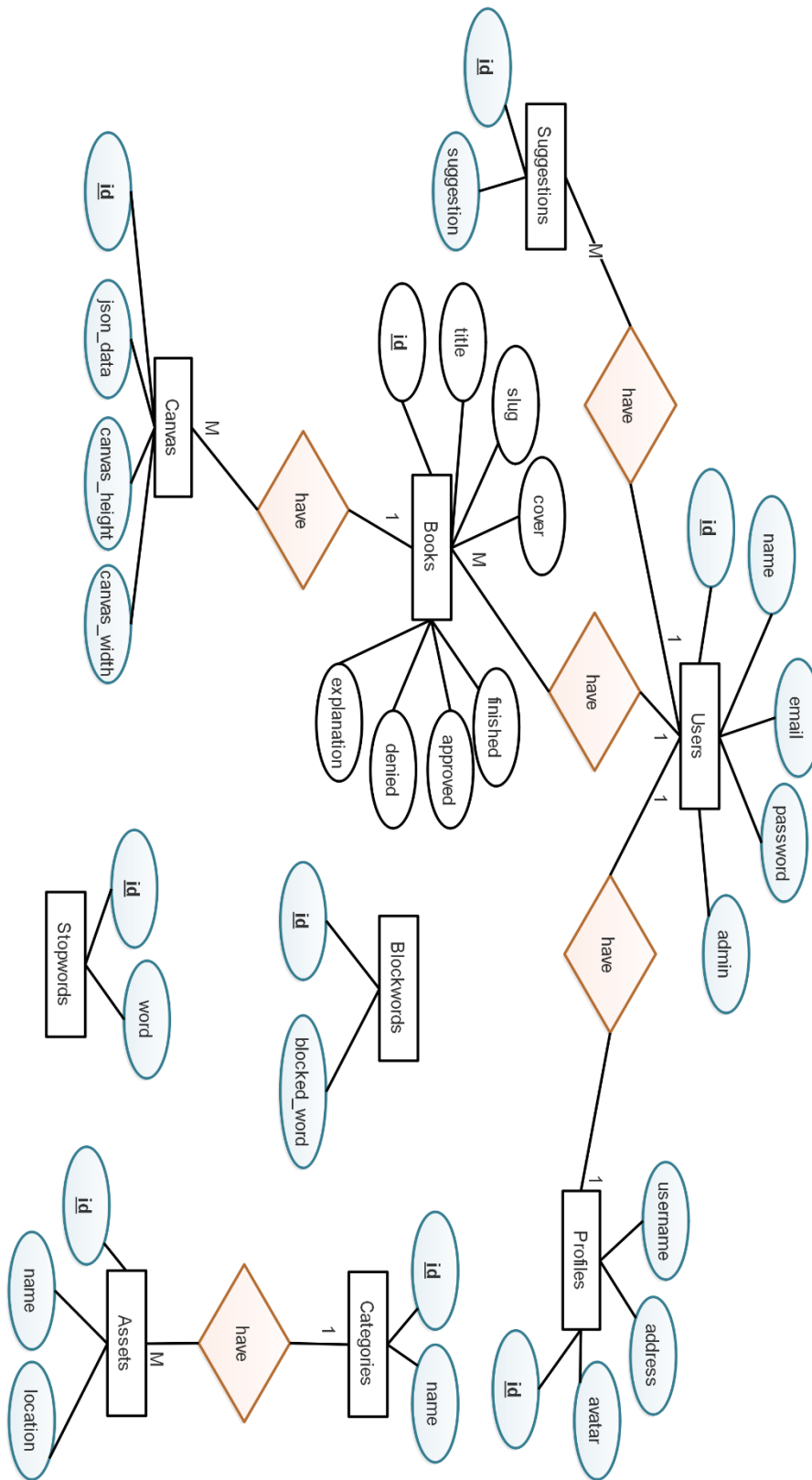




Gambar 3.18 Activity Diagram Mengelola Kosakata

### 3.5 Entity Relationship Diagram

*Entity Relationship* diagram (ERD) digunakan untuk menggambarkan entitas, atribut, relasi dan kardinalitas dalam *database*. Desain yang dibuat didapatkan dari perancangan *use case* diagram dan *activity* diagram. Gambar 3.19 menunjukkan rancangan dari ERD untuk aplikasi yang akan dibangun.



Gambar 3.19 Entity Relationship Diagram (ERD)

### 3.6 Struktur Tabel *Database*

#### 3.6.1 Tabel *Users*

Tabel 3.16 *users* menunjukkan penggunaan Tabel *Users* pada database untuk menyimpan data *credential* seperti data *login*. Tabel ini memiliki *field* *id*, *name*, *email*, *password*, dan *admin*. Saat *login* atau registrasi data yang diproses akan disimpan pada tabel ini.

Tabel 3.16 Tabel *Users*

Nama Kolom	Tipe Data	Keterangan
Id	Integer(10)	<i>Primary key</i>
Name	Varchar(255)	
password	Varchar(255)	
Email	Varchar(255)	
Admin	TinyInt(1)	

#### 3.6.2 Tabel *Profiles*

Tabel 3.17 menunjukkan penggunaan Tabel *Profiles* untuk menyimpan data profil dari anggota. Tabel ini memiliki *field* *id*, *avatar* untuk menyimpan nama dari foto profil, *username* untuk username anggota, *address* untuk menyimpan alamat anggota, dan terdapat *field* *user\_id* untuk menyimpan id anggota dari tabel *users*.

Tabel 3.17 Tabel *Profiles*

Nama Kolom	Tipe Data	Keterangan
Id	Integer(10)	<i>Primary key</i>
avatar	Varchar(255)	
username	Varchar(255)	
address	Varchar(255)	
user_id	TinyInt(11)	<i>Foreign key</i>

#### 3.6.3 Tabel *Books*

Tabel 3.18 menunjukkan penggunaan Tabel *Books* untuk menyimpan data buku. Terdapat *field* *id*, *title* untuk judul buku, *slug* untuk alamat *route*-nya, *finished* untuk penanda apakah selesai atau tidak, *approved* untuk penanda apakah buku diterima atau tidak, *explanation* untuk penjelasan jika buku tertolak, *denied* untuk menandakan bahwa buku tertolak, *cover* untuk data bukunya, *profil\_id* untuk menyimpan id profil sebagai *foreign key*.

Tabel 3.18 Tabel *Books*

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
Id	Integer(10)	<i>Primary key</i>
title	Varchar(255)	
slug	Varchar(255)	
finished	TinyInt(1)	
approved	TinyInt(1)	
denied	TinyInt(1)	
explanation	Varchar(255)	
cover	Longtext	
profil_id	Integer(11)	<i>Foreign key</i>

### 3.6.4 Tabel *Canvas*

Tabel 3.19 menunjukkan penggunaan Tabel *Canvas* untuk menyimpan data *canvas* jika suatu cerita telah dibuat dan ingin disimpan. Di dalamnya terdapat *field id*, *json\_data* untuk menyimpan data *canvas* sebagai data *json*, dan *book\_id* yang menyimpan id buku dari tabel *book* sebagai *foreign key*.

Tabel 3.19 Tabel *Canvas*

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
Id	Integer(10)	<i>Primary key</i>
json_data	Longtext	
book_id	Integer(11)	<i>Foreign key</i>
canvas_height	Varchar(255)	
canvas_width	Varchar(255)	

### 3.6.5 Tabel *Category*

Tabel 3.20 menunjukkan penggunaan Tabel *Categories* untuk menyimpan data kategori. Data kategori nantinya akan digunakan untuk mengklasifikasikan berbagai macam asset. Dalam tabel ini terdapat *field id*, dan *field name* untuk menyimpan nama dari kategori.

Tabel 3.20 Tabel *Categories*

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Keterangan</b>
Id	Integer(10)	<i>Primary key</i>
name	Varchar(255)	

### 3.6.6 Tabel Asset

Tabel 3.21 menunjukkan penggunaan Tabel *Assets* untuk menyimpan data asset. Data asset nantinya akan digunakan pada saat membuat cerita untuk dimasukkan pada canvas buku cerita. Di dalam tabel ini terdapat *field id*, *name* untuk nama kategori, *location* untuk lokasi penyimpanan asset, dan *category\_id* untuk menyimpan id kategori dari tabel *category* sebagai *foreign key*.

Tabel 3.21 Tabel *Assets*

Nama Kolom	Tipe Data	Keterangan
Id	Integer(10)	<i>Primary key</i>
name	Varchar(255)	
location	Varchar(255)	
category_id	Integer(11)	<i>Foreign key</i>

### 3.6.7 Tabel *Suggestions*

Tabel 3.22 menunjukkan penggunaan Tabel *Suggestion* untuk menyimpan data kritik & saran dari anggota. Di dalamnya terdapat *field id*, *suggestion* untuk menyimpan data kritik & saran, *user\_id* untuk menyimpan id anggota dari tabel *user* sebagai *foreign key*.

Tabel 3.22 Tabel *Suggestion*

Nama Kolom	Tipe Data	Keterangan
Id	Integer(10)	<i>Primary key</i>
suggestion	Varchar(255)	
user_id	Integer(11)	<i>Foreign key</i>

### 3.6.8 Tabel *Stopwords*

Tabel 3.23 menunjukkan penggunaan Tabel *Stopwords* untuk menyimpan data *stopword* atau *stoplist*. Data tersebut nanti akan digunakan untuk mengecek teks dari suatu cerita untuk memberikan rekomendasi asset. Dalam tabel ini terdapat *field id* dan *field words* untuk data kata yang *irrelevant*.

Tabel 3.23 Tabel *Stopwords*

Nama Kolom	Tipe Data	Keterangan
Id	Integer(10)	<i>Primary key</i>
words	Varchar(255)	

### 3.6.9 Tabel *Blockedwords*

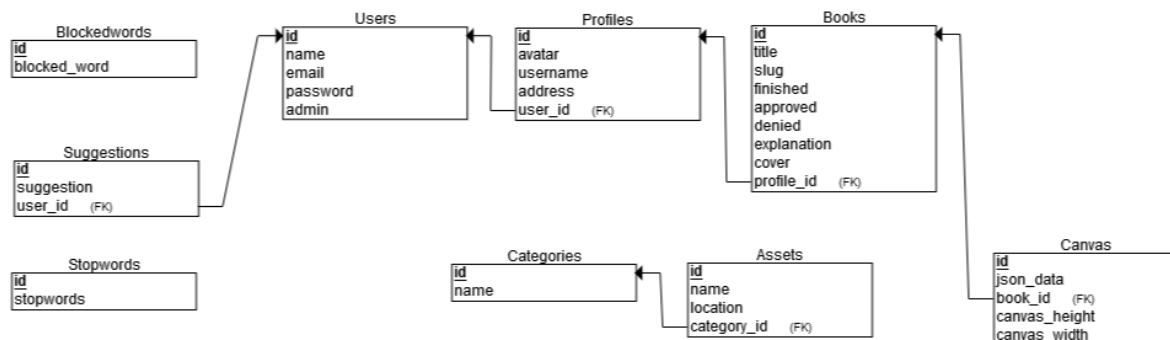
Tabel 3.24 digunakan untuk menyimpan data kosa kata yang terlarang. Kosa kata ini dapat dikelola oleh admin. Admin dapat memilih kosa kata apa saja yang mungkin tidak pantas untuk berada dalam cerita. Dalam tabel ini terdapat *field id*, dan *field blocked\_word* untuk menyimpan data kata pilihan dari admin.

Tabel 3.24 Tabel *Blockedword*

Nama Kolom	Tipe Data	Keterangan
Id	Integer(10)	<i>Primary key</i>
blocked_word	Varchar(255)	

### 3.7 Relational Schemas

Gambar 3.20 menunjukkan relasi basisdata yang digunakan pada sistem yang akan dibangun. Dalam desain basisdata yang digunakan menghasilkan 9 tabel yang beberapa memiliki relasi antara satu dengan yang lain. Terdapat tabel *users* yang memiliki relasi *one-to-one* pada tabel *profiles* yang berarti setiap pengguna memiliki satu data profil pada tabel *profiles*. Tabel *users* juga memiliki relasi *one-to-many* pada tabel *books* dan tabel *suggestions*, artinya pengguna dapat memiliki banyak buku dan dapat mengirimkan banyak saran & kritik. Tabel *books* memiliki relasi *one-to-many* pada tabel *canvas*, artinya setiap buku dapat memiliki banyak data *canvas* yang tersimpan. Tabel *assets* memiliki relasi *one-to-many* pada tabel *categories*, dimana setiap asset merupakan bagian dari satu kategori, dan satu kategori dapat memiliki banyak asset dalam kategori tersebut. Juga terdapat tabel *stopwords* yang berdiri sendiri tanpa relasi yang bertugas menyimpan kata *stopwords* yang nanti akan digunakan pada fitur rekomendasi.

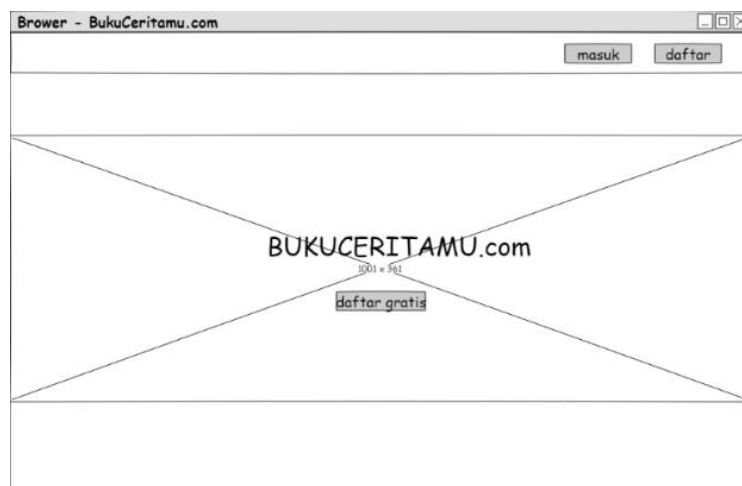


Gambar 3.20 Skema Relasi *Database*

### 3.8 Perancangan *Mockup* Antarmuka (*Interface*)

#### 3.8.1 Halaman Index Sistem

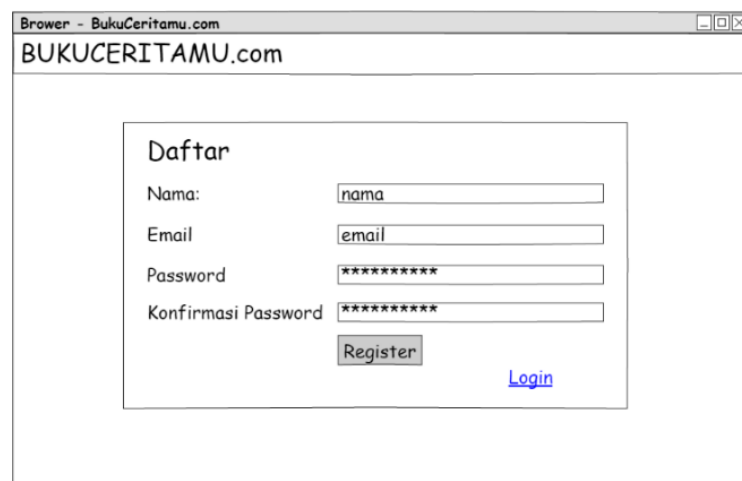
Gambar 3.21 merupakan rancangan *mockup* halaman *index* bagi pengguna yang belum terdaftar atau pengunjung. Menampilkan *landing page* dengan background gambar dan tombol link untuk menuju halaman pendaftaran dan halaman *login*.



Gambar 3.21 Halaman *Index* Sistem Bagi Pengunjung

#### 3.8.2 Halaman Registrasi

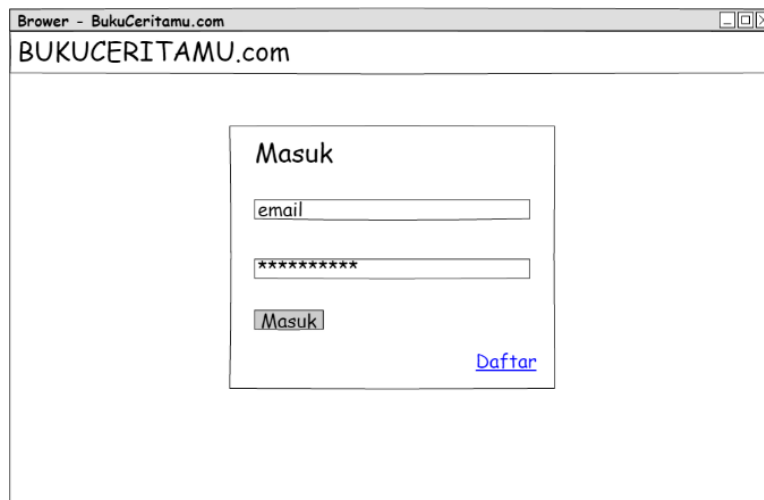
Gambar 3.22 adalah rancangan rancangan dari halaman registrasi pengguna yang belum terdaftar. Halaman ini menampilkan form untuk memasukan data pengguna.

A screenshot of a web browser window titled "Browser - BukuCeritamu.com". The page displays a registration form titled "Daftar". The form includes four input fields: "Nama:" with the placeholder "nama", "Email" with the placeholder "email", "Password" with "\*\*\*\*\*", and "Konfirmasi Password" with "\*\*\*\*\*". Below the fields is a "Register" button and a blue "Login" link.

Gambar 3.22 Halaman Registrasi Pengguna

### 3.8.3 Halaman *Login*

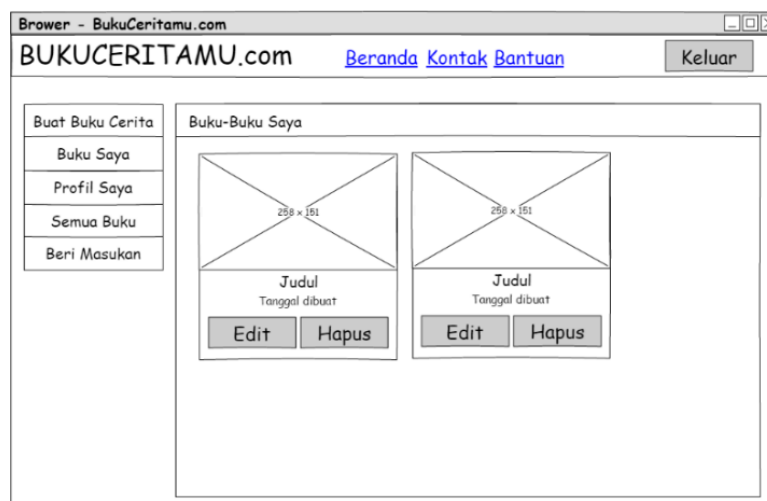
Gambar 3.23 merupakan rancangan halaman *login* pengguna yang telah terdaftar atau disebut anggota. Ketika telah masuk maka anggota dapat langsung menggunakan fitur aplikasi seperti membuat buku cerita ataupun membaca buku cerita.



Gambar 3.23 Halaman Login

### 3.8.4 Halaman Beranda Anggota

Gambar 3.24 adalah rancangan antarmuka dari halaman beranda anggota. Ketika anggota telah benar memasukkan data *login*, maka halaman ini yang akan ditampilkan.

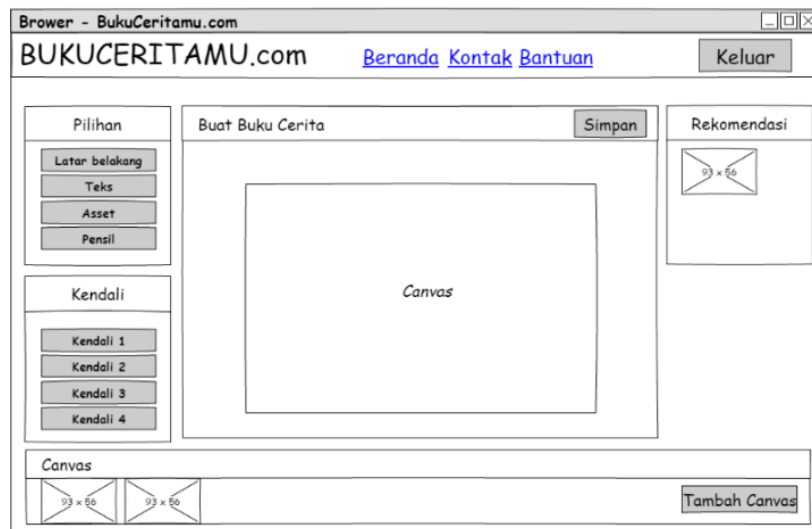


Gambar 3.24 Halaman Beranda Anggota



### 3.8.5 Halaman Membuat Buku Cerita

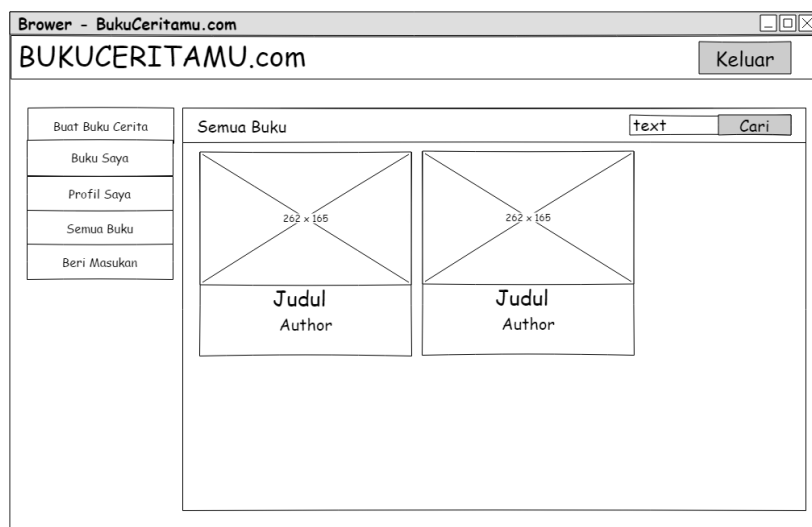
Gambar 3.25 adalah rancangan antarmuka halaman membuat cerita. Halaman ini menampilkan *canvas* buku cerita dan beserta fitur-fitur dalam menulis cerita.



Gambar 3.25 Halaman Membuat Buku Cerita

### 3.8.6 Halaman Semua Buku

Gambar 3.26 adalah rancangan untuk halaman semua buku. Halaman ini menampilkan semua buku yang telah dipublikasikan oleh anggota, juga terdapat kotak pencarian untuk mencari buku.



Gambar 3.26 Halaman Semua Buku

### 3.8.7 Halaman Memberi Kritik Dan Saran

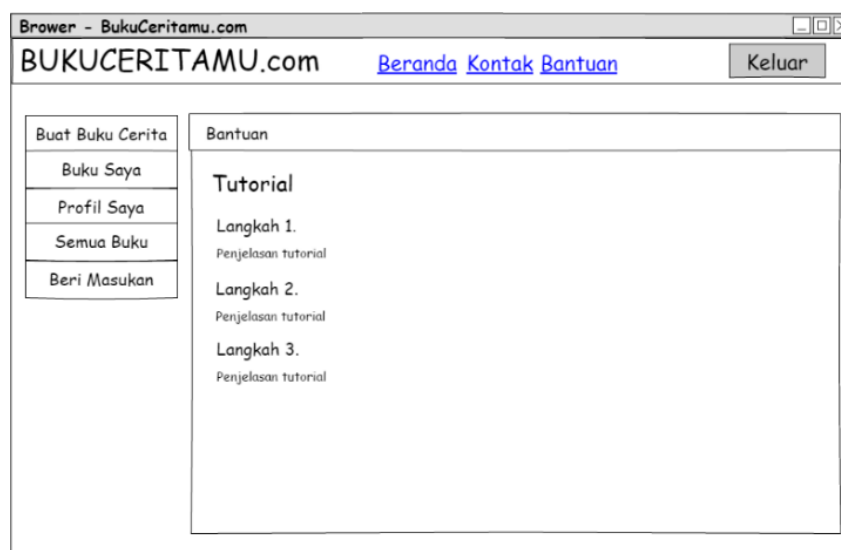
Gambar 3.27 adalah rancangan halaman memberi kritik dan saran. Halaman ini hanya berisi form sederhana yang mana anggota dapat mengirimkan data dan akan dibaca oleh admin.



Gambar 3.27 Halaman Memberi Kritik & Saran

### 3.8.8 Halaman Bantuan

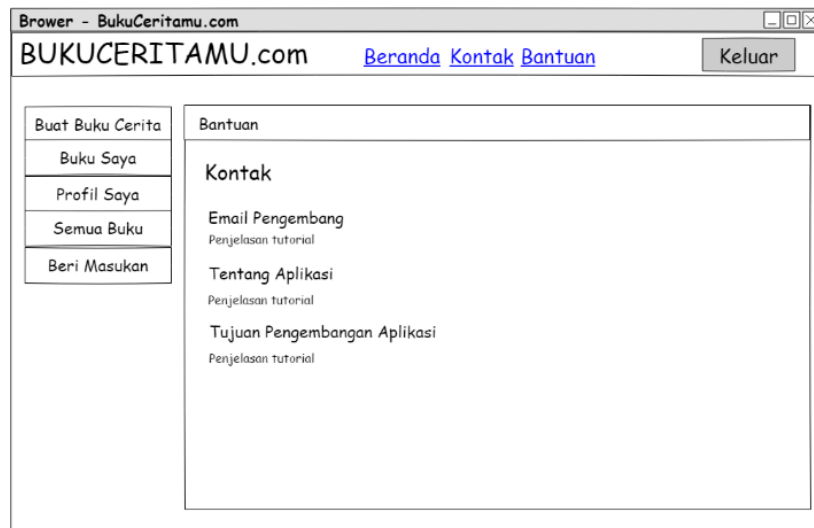
Gambar 3.28 adalah rancangan halaman bantuan. Halaman ini memberikan langkah-langkah tutorial dalam membuat buku.



Gambar 3.28 Halaman Bantuan

### 3.8.9 Halaman Kontak

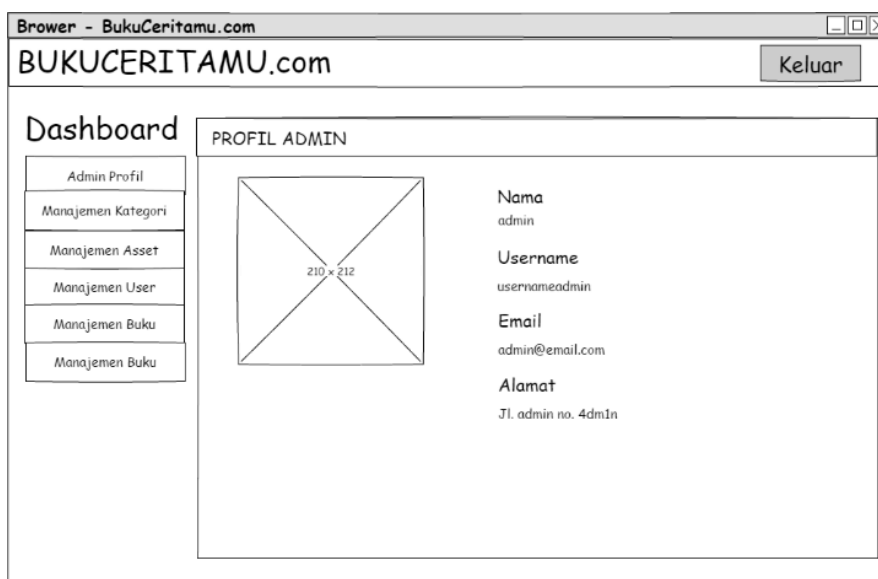
Gambar 3.29 adalah rancangan halaman kontak yang bersisi kontak pengembang, dan tentang aplikasi.



Gambar 3.29 Halaman Kontak

### 3.8.10 Halaman Beranda Admin

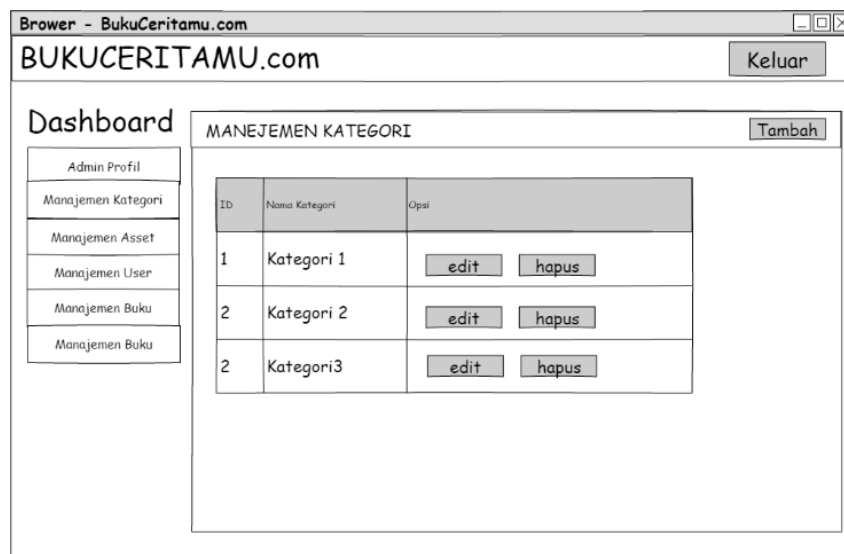
Gambar 3.30 adalah rancangan halaman *index* atau beranda dari admin. Halaman beranda admin ketika diakses langsung menunjukkan info profil dari admin.



Gambar 3.30 Halaman Beranda Admin

### 3.8.11 Halaman Manajemen Kategori Admin

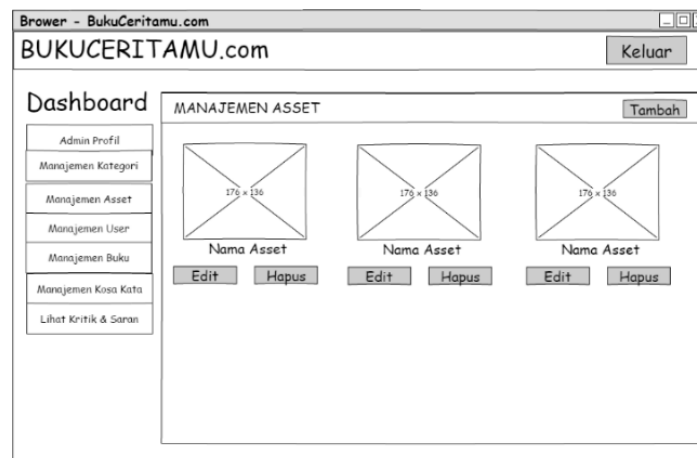
Gambar 3.31 adalah rancangan halaman manajemen kategori untuk admin. Halaman ini menampilkan kategori yang ada dalam bentuk tabel, dan admin dapat mengelola kategori seperti menambah, mengedit ataupun menghapus.



Gambar 3.31 Halaman Manajemen Kategori Admin

### 3.8.12 Halaman Asset Admin

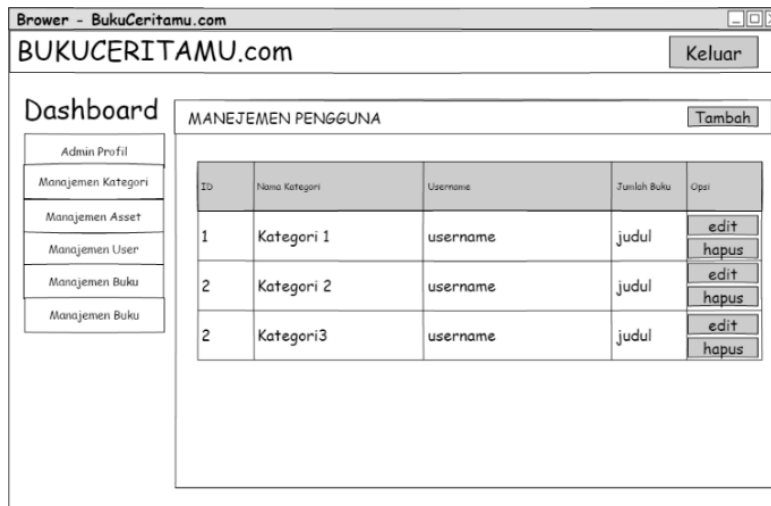
Gambar 3.32 adalah rancangan halaman manajemen pengguna untuk admin. Halaman ini menampilkan para pengguna aplikasi yang telah terdaftar.



Gambar 3.32 Halaman Manajemen Asset Admin

### 3.8.13 Halaman Manajemen Pengguna Admin

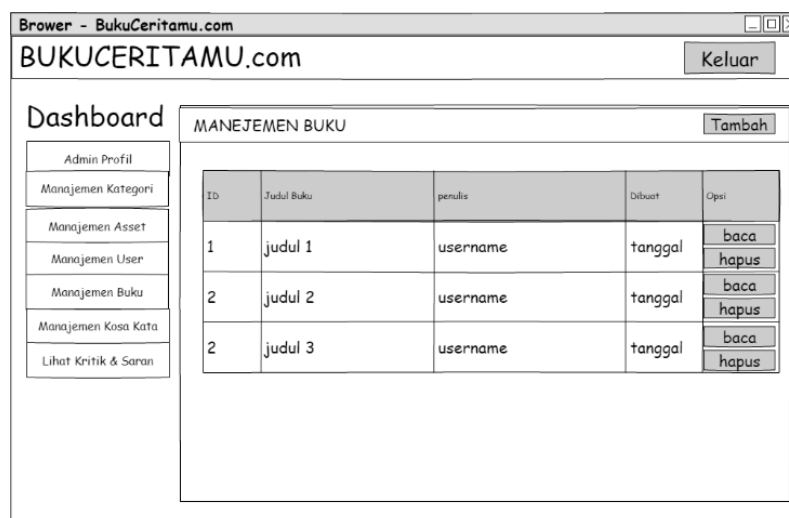
Gambar 3.33 adalah rancangan halaman manajemen pengguna untuk admin. Halaman ini menampilkan para pengguna aplikasi yang telah terdaftar.



Gambar 3.33 Halaman Manajemen Pengguna Admin

### 3.8.14 Halaman Manajemen Buku Admin

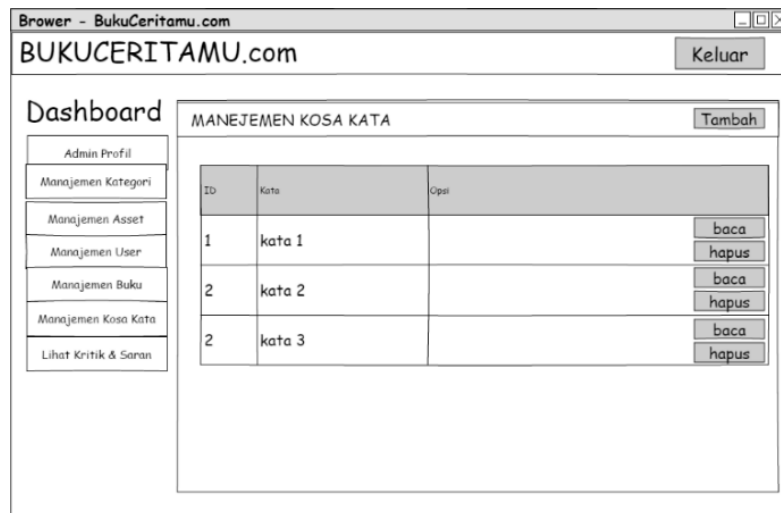
Gambar 3.34 adalah rancangan halaman manajemen buku untuk admin. Halaman ini menampilkan buku-buku yang telah dibuat oleh pengguna.



Gambar 3.34 Halaman Manajemen Buku

### 3.8.15 Halaman Manajemen Kosakata Admin

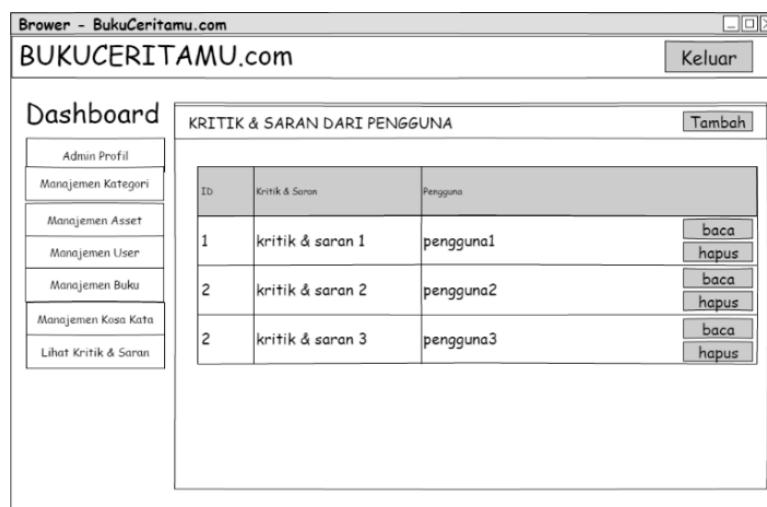
Gambar 3.35 adalah rancangan halaman manajemen kosakata. Halaman ini menampilkan kosakata yang telah di input oleh admin. Kosakata tersebut akan digunakan untuk memfilter buku yang akan dipublikasikan.



Gambar 3.35 Halaman Manajemen Kosakata

### 3.8.16 Halaman Kritik & Saran Admin

Gambar 3.36 adalah rancangan halaman kritik & saran untuk admin. Halaman ini menunjukkan saran-saran dan kritik-kritik yang diberikan oleh pengguna. Admin hanya dapat melihatnya dan tidak dapat mengelolanya.



Gambar 3.36 Halaman Kritik & Saran Untuk Admin

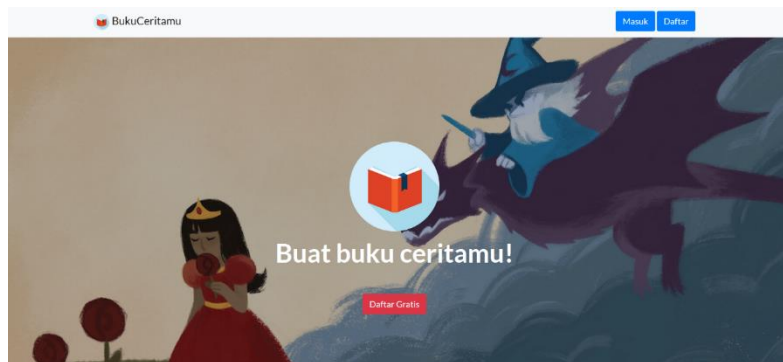
## BAB IV IMPLEMENTASI DAN PENGUJIAN

### 4.1 Implementasi Sistem

Implementasi yaitu pembuatan aplikasi berdasarkan rancangan yang telah dilakukan pada bab sebelumnya. Pada pembahasan implementasi yang dilakukan hanya pada fitur manajemen pada admin dan fitur-fitur anggota saja seperti fitur pencarian buku, membaca cerita, membuat cerita, dan mengunduh cerita. Dalam tahap ini juga dilakukan implementasi penerapan metode *text pre-processing* pada pembuatan fitur rekomendasi asset.

#### 4.1.1 Pencarian Buku

Pengguna yang mengakses aplikasi dapat langsung mencari suatu buku cerita hanya dengan mengetikkan judul cerita. Pengguna dapat menggunakan fitur ini walaupun tidak melakukan pendaftaran terlebih dahulu. Ketika mengakses aplikasi maka tampilan *index* dari aplikasi akan terlihat seperti pada Gambar 4.1



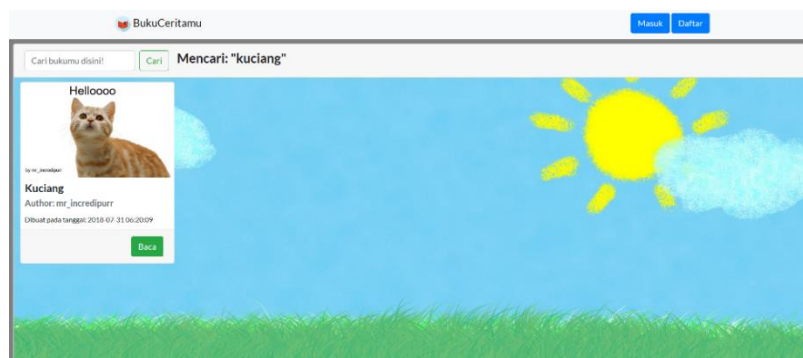
Gambar 4.1 Halaman *Index*

Jika pengguna melakukan *scroll* ke bawah halaman, maka pengguna akan menemukan kotak pencarian buku, dan buku-buku terbaru yang telah dipublikasikan. Kotak pencarian tersebut terlihat pada Gambar 4.2



Gambar 4.2 Halaman *Index* Kotak Pencarian

Jika pengguna ingin mencari buku, maka pengguna harus memasukan judul buku pada kotak pencarian dan tekan tombol cari. Jika terdapat buku maka akan terlihat seperti pada Gambar 4.3, dan jika buku tidak ditemukan maka akan menampilkan notifikasi bahwa tidak terdapat buku pada *database* seperti yang terlihat pada Gambar 4.4.



Gambar 4.3 Halaman Hasil Pencarian Buku Berhasil

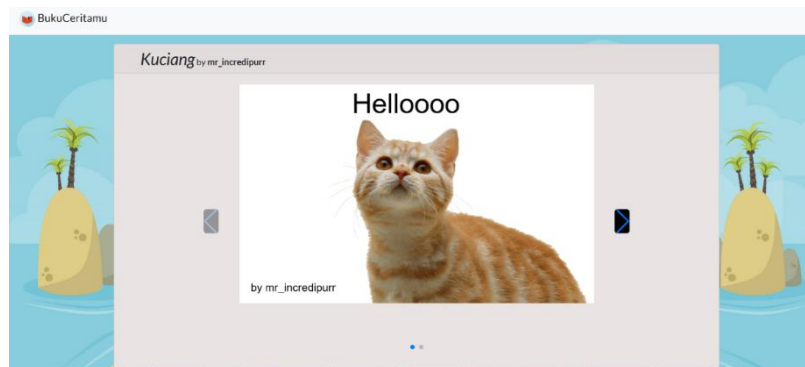


Gambar 4.4 Halaman Hasil Pencarian Buku Gagal



### 4.1.2 Membaca Buku

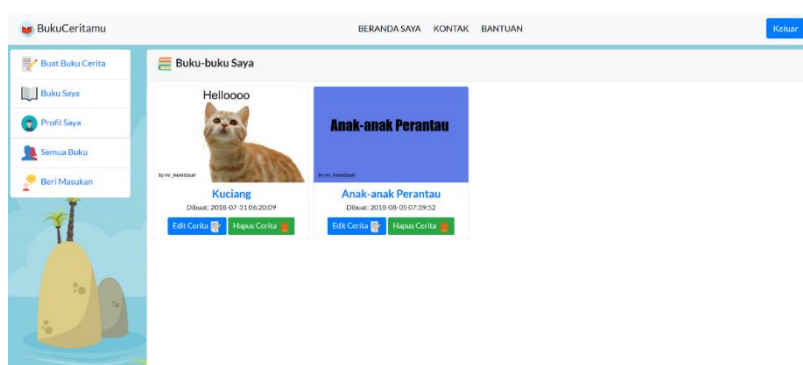
Jika telah melakukan pencarian, pengguna dapat membaca dengan mengklik link dari judul buku tersebut. Buku yang ditampilkan yaitu ditampikan menggunakan elemen *canvas* dengan fitur slide untuk pindah antar canvas. Tampilan buku saat di baca akan seperti pada Gambar 4.5.



Gambar 4.5 Halaman Membaca Buku

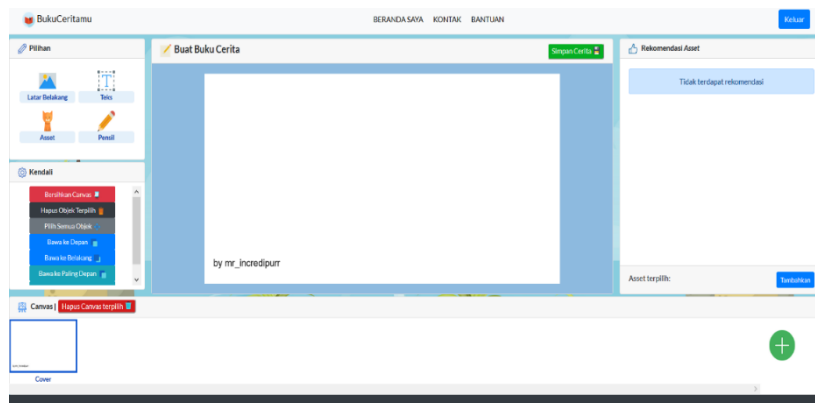
### 4.1.3 Membuat Buku Cerita

Untuk dapat membuat buku cerita, pengguna harus terlebih dahulu mendaftar sebagai anggota. Pengguna harus mengisi form data pengguna yang nantinya akan disimpan pada *database* aplikasi. Jika telah terdaftar, pengguna akan diarahkan pada halaman beranda anggota seperti pada Gambar 4.6.



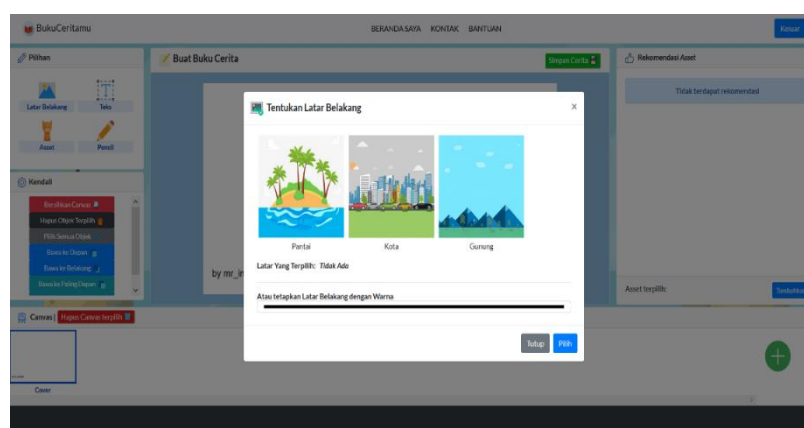
Gambar 4.6 Halaman Beranda Anggota

Pada saat berada pada halaman beranda anggota, aplikasi menampilkan buku-buku yang telah dibuat oleh anggota dengan diberikan pilihan edit cerita dan hapus cerita. Jika anggota ingin membuat cerita maka tinggal memilih pilihan buat buku cerita. Tampilan pada halaman membuat buku cerita akan terlihat seperti pada Gambar 4.7.



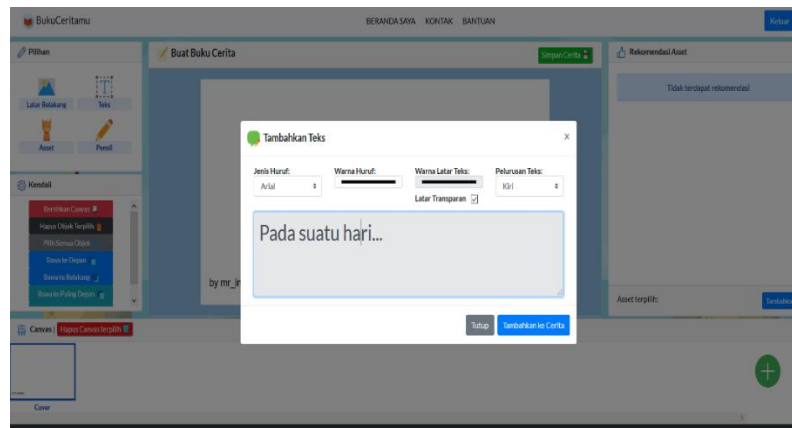
Gambar 4.7 Halaman Membuat Cerita

Pengguna dapat menggunakan fitur-fitur seperti fitur menambahkan teks, menerapkan latar belakang, menambahkan asset, atau menulis dengan pensil. Gambar 4.8 menunjukkan tampilan jendela ketika pengguna memilih pilihan latar belakang. Aplikasi akan menunjukkan gambar-gambar latar belakang yang dapat diterapkan pada canvas. Pengguna dapat memilih gambar sebagai latar belakang atau pengguna dapat memilih warna sebagai latar belakang.



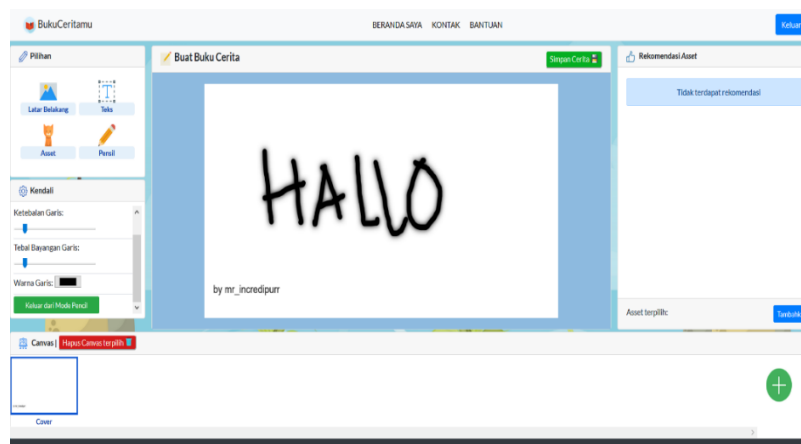
Gambar 4.8 Fitur Menerapkan Latar Belakang

Gambar 4.9 menunjukkan tampilan aplikasi ketika pengguna memilih pilihan teks. Aplikasi akan menampilkan jendela berisi kolom teks yang dapat menerima teks cerita. Pada jendela pengguna dapat mengatur jenis font, warna, dan warna latar belakang dari teks yang dituliskan.



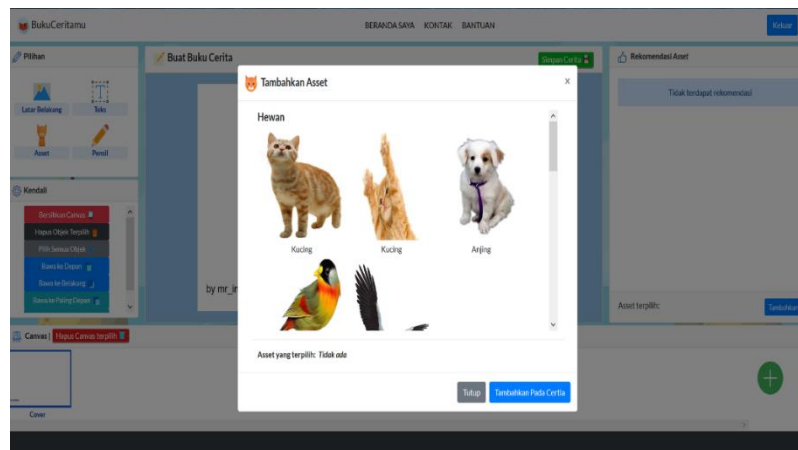
Gambar 4.9 Fitur Penambahan Teks

Gambar 4.10 menunjukkan tampilan saat pengguna menggunakan fitur menggambar pada canvas. Pengguna dapat mengatur warna dan ketebalan pensil yang digunakan untuk menggambar.



Gambar 4.10 Penggunaan Fitur Menulis/Menggambar

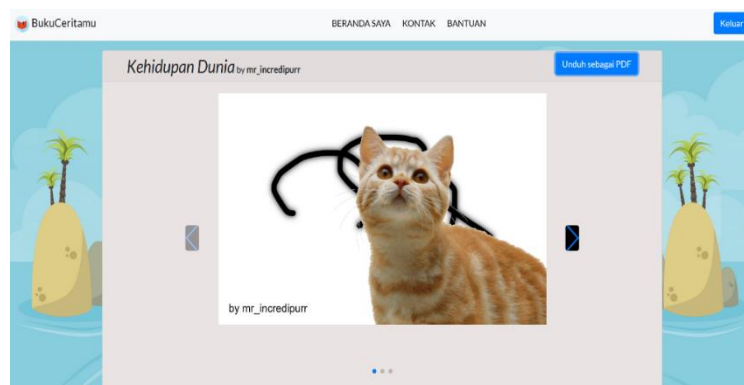
Gambar 4.11 menunjukkan tampilan saat pengguna memilih pilihan tambah asset. Aplikasi akan menampilkan jendela yang berisi gambar-gambar asset yang dapat ditambahkan pada canvas.



Gambar 4.11 Fitur Penambahan Asset

#### 4.1.4 Mengunduh Cerita

Dalam proses pengunduhan suatu cerita, pengguna diharuskan untuk mendaftar terlebih dahulu, karena fitur unduh pada aplikasi hanya diberikan kepada anggota yang telah terdaftar. Cara pengunduhannya yakni dengan membuka halaman membaca, dan memilih tombol unduh sebagai PDF. Halaman ini akan terlihat seperti pada Gambar 4.12.

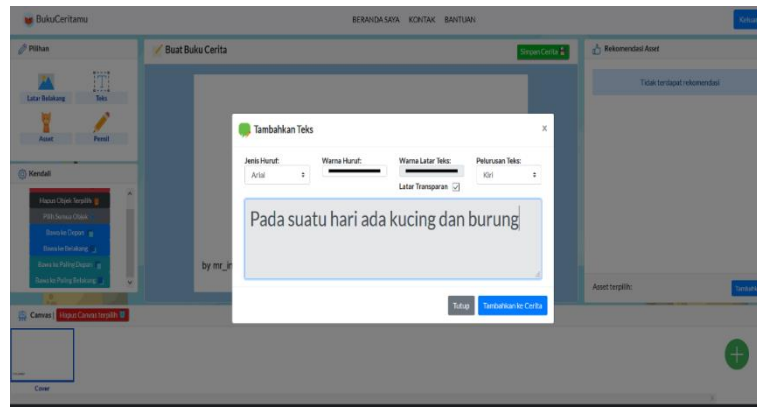


Gambar 4.12 Halaman Pengunduhan Buku Cerita

#### 4.1.5 Rekomendasi Asset

Pada fitur rekomendasi asset digunakan metode *text pre-processing* untuk mengoptimalkan indeks token dan hasil token tersebut akan digunakan untuk mencari asset

yang ada pada *database* untuk dijadikan rekomendasi. String yang belum dijadikan token awalnya di *input* oleh anggota saat menambahkan teks cerita, seperti pada Gambar 4.13.



Gambar 4.13 Anggota Memasukkan Teks Cerita

Kemudian dilakukan tahap awal dari metode *text pre-processing* yaitu tokenisasi. Tokenisasi yang dilakukan yaitu teks tersebut di potong setiap perkata lalu disimpan sebagai token. Namun sebelum disimpan token harus tersebut dibersihkan terlebih dahulu dari spasi atau *whitespace* dan dibersihkan dari segala simbol. Setelah itu akan dicarikan sinonim dari token dan menyimpannya sebagai token lain. Alur *script* dari tokenisasi dapat dilihat pada Gambar 4.14.

```
// ALGORITHM FOR TOKENIZATION
// split by space
var checkText = text.split(/[ -]+/);
// console.log(checkText);
var checkTextForBlockwords = text.split(/[ -]+/);
// check for symbol in it and remove it
for(var i = 0; i < checkText.length; i++){
    checkText[i] = checkText[i].replace(/[\W_\n\r]/g, '').toLowerCase();
// console.log(checkText[i]);
}
// /=====ALGORITHM FOR TOKENIZATION
```

Gambar 4.14 Script Tokenisasi

Setelah dilakukan tokenisasi dan pembersihan spasi dan simbol, kemudian token tersebut akan diproses pada proses *stopwords removal* yaitu menghilangkan kata yang tidak penting. Kata-kata yang tidak penting tersebut telah disimpan pada *database*, kemudian data tersebut disimpan pada variable *array* dan dilakukan perbandingan dengan token. Jika terdapat token yang sama dengan kata yang tidak penting, maka token tersebut akan dibuang. *Script* algoritma *stopwords removal* dapat dilihat pada Gambar 4.15

```

// Get request stopwords and save it to array
$.ajax({
  async: "true",
  crossDomain: 'true',
  method: 'get',
  url: 'http://mytugasakhir.oo/api/get-stopwords',
}).done(function(res, status){
  console.log('Get all stopwords, status: '+status);
  for(var i = 0; i<res.length; i++){
    allStopwords.push(res[i]);
  }
// console.log(allStopwords);
});

// ALGORITHM FOR STOPWORDS REMOVAL

// checkText is the tokens array
// remove all the matching stopwords and switch to empty string
for(var i = 0; i < checkText.length; i++){
  for(var j=0; j< allStopwords.length; j++){
    if(checkText[i].includes(allStopwords[j].stopwords)){
      checkText[i] = "";
    }
  }
}
}

```

Gambar 4.15 Script *Stopwords Removal*

Setelah token telah bersih dari *stopwords* atau *stoplist*, maka dilanjutkan keproses selanjutnya yaitu *stemming*. *Stemming* yang dilakukan yaitu menggunakan algoritma *stemming* Tala (2003) dengan berbasis *rule based*. Kata-kata yang akan digunakan dalam *stemming* ini telah ditentukan seperti pada Tabel 3.15. Alur dari script pemograman pengikuti alur dari *flowchart* pada Gambar 3.4. Penulisan *script* algoritma dapat dilihat pada Gambar 4.16.

```

// ALGORITHM FOR STEMMING WORD
// POST WORDS (Kata Sanding) ===== CLUSTER 1 =====
var collocationWords = ['lah','kah','pun','ku','mu','nya','kan','an','i'];

// PREFIX WORDS SWITCH WITH " " ===== CLUSTER 2 =====
var prefixWordsForNothing = ['meng',
'men','mem','me','peng','pen','pem','di','ter','ke','ber',
'bel','be','per','pel','pe'];

// PREFIX WORDS SWITCH WITH "S" ===== CLUSTER 3 =====
var prefixWordsForS = ['menya', 'menyi',
'menyu','menye','meny','penya','penyi','penyu','penye',
'penyo','peny'];

// PREFIX WORDS SWITCH WITH "P" ===== CLUSTER 4 =====
var prefixWordsForP = ['mema',
'memi','memu','meme','memo','pema','pemi','pemu','peme','pemo'];

for(var i = 0; i < checkText.length; i++){
// Check if words has more charachters than four
if(checkText[i].length > 2){ //BARIS 22
// Remove collocation words
//loop every collocation words
for(var j = 0; j < collocationWords.length; j++){

```



```

// ALGORITHM FOR GIVING ASSET RECOMENDATION
// get all the matching token with asset and make the HTML
if(checkText.length>0){
  for(var i = 0; i < checkText.length; i++){
    for(var j = 0; j < allAssets.length; j++){
      if(allAssets[j].name.toLowerCase().indexOf(checkText[i]) >= 0 &&
checkText[i] != ""){
        allRecommendedAssets.push(allAssets[j]);
        console.log('Ada: '+checkText[i]+' di dalam: '+ allAssets[j].name);
      }
    }
  }
}
// console.log(allRecommendedAssets);
// console.log(allAssets);
$('#recommendLoader').hide();
if(allRecommendedAssets.length>0){
  // make the html
  for(var i = 0; i < allRecommendedAssets.length; i++){
    $('.recommendedAssets').append(`
      <div class="my-div-thumbnail" onclick="getRecommendAsset(this)">
        
        <span class="my-thumbnail-label">`+
allRecommendedAssets[i].name +`</span>
      </div>`);
  }
  $("#recommendLoader").hide();
  $('div.recommendedAssets').show();
}else{
  $("#recommendLoader").hide();
  $("#recommendInfo").show();
}
}
}

```

Gambar 4.17 Script Pencarian Asset

Jika proses *stemming* kata telah selesai maka akan dilanjutkan pada proses pencarian asset. Gambar 4.17 merupakan *script* dari proses pencarian rekomendasi asset. Data token yang ada diperiksa dan dicarikan pada data asset yang ada. Jika terdapat kesamaan maka asset akan disimpan, dan dibuatkan *tag html* pada bagian rekomendasi, dan menampilkan rekomendasi. Jika tidak terdapat rekomendasi, maka akan menampilkan info tidak terdapat rekomendasi asset.

Pada saat yang sama, dilakukan juga pemeriksaan kata yang terlarang. Kata tersebut ditentukan oleh admin dalam sistem, dan akan dijadikan patokan pada cerita yang ada. Pada saat pengguna telah selesai menuliskan cerita, teks yang ada akan diperiksa apakah terdapat kata yang dilarang oleh admin. Jika ada maka kata tersebut akan disimpan dan akan



disampaikan pada admin dalam bentuk keterangan. *Script* proses pencarian kata yang dilarang dapat dilihat pada Gambar 4.18.

```
// Get request blockedwords and save it to array
$.ajax({
  async: "true",
  crossDomain: 'true',
  method: 'get',
  url: 'http://mytugasakhir.oo/api/get-blockedwords',
}).done(function(res, status){
  console.log('Get all blockswords, status: '+status);
  for(var i = 0; i<res.length; i++){
    allBlockedwords.push(res[i]);
  }
  // console.log(allBlockedwords);
});

// get text when users input it
var checkTextForBlockwords = text.split(" ");

for(var i = 0; i < checkTextForBlockwords.length; i++){
  for(var j = 0; j < allBlockedwords.length; j++){

if(checkTextForBlockwords[i].indexOf(allBlockedwords[j].blocked_word.toLowerCase(
)) >= 0 && checkTextForBlockwords[i] != "" ||
allBlockedwords[j].blocked_word.toLowerCase().indexOf(checkTextForBlockwords[i])
>= 0 && checkTextForBlockwords[i] != ""){
    allMatchingBlockedwords.push(checkTextForBlockwords[i]);
  }
}
}
}
```

Gambar 4.18 *Script* Pencarian *Blockwords*

Walau hasil dari proses *text pre-processing* belum sempurna khususnya pada proses *stemming* pada token kata seperti pemotongan kata sanding yang kurang tepat, salah satunya seperti kata dengan akhiran ‘i’ akan terpotong oleh sistem karena aturan yang telah ditentukan, sehingga akan terdapat kesalahan apabila kata yang dideteksi memiliki kata dasar yang berakhiran ‘i’. Contoh dengan kata “lari” maka kata yang di dapatkan adalah “lar”. Namun dalam hal ini sistem tetap dapat mencari asset karena kata yang belum diproses dan sudah diproses tetap digunakan untuk mencari rekomendasi asset, sehingga fitur tersebut dapat tetap berjalan dengan baik.

## 4.2 Pengujian

Pada bagian ini menjelaskan pengujian yang dilakukan yaitu dengan pengujian *blackbox* pada fungsionalitas dan pengujian usabilitas berdasarkan *use case* yang telah ditentukan.

Pengujian ini dilakukan untuk mengetahui apakah sistem aplikasi telah berjalan dengan baik atau tidak.

#### 4.2.1 Pengujian Fungsionalitas (*Black Box*)

Pada tahap ini, pengujian dilakukan untuk melihat apakah fitur-fitur yang direncanakan telah dapat diwujudkan. Hasil pengujian yang dilakukan dapat terlihat pada Tabel 4.1.

Tabel 4.1 Tabel Pengujian Fungsionalitas

No	No. UC	Aktivitas	Hasil	Kesimpulan
1	UC-1	Registrasi Pengunjung	Menampilkan info pendaftaran berhasil	Berhasil
2	UC-2	<i>Login</i>	Menampilkan info berhasil <i>login</i>	Berhasil
3	UC-3	Mencari Buku Cerita	Menampilkan hasil pencarian pengguna	Berhasil
			Menampilkan info hasil pencarian gagal	Berhasil
4	UC-4	Membaca Buku Cerita	Menampilkan Buku Cerita Yang dipilih	Berhasil
5	UC-5	Mengelola Profil	Menampilkan info berhasil	Berhasil
6	UC-6	Membuat Cerita	Menampilkan info berhasil	Berhasil
7	UC-7	Memberi Saran & Kritik	Menampilkan info berhasil	Berhasil

8	UC-8	Mengedit Cerita	Menampilkan info berhasil mengedit	Berhasil
		Mengunduh Cerita	Mengirimkan file PDF pada pengguna	Berhasil
		Menghapus Cerita	Menampilkan konfirmasi terhadap penghapusan	Berhasil
9	UC-9	Memberikan Izin	Menampilkan konfirmasi terhadap pengizinan	Berhasil
		Memberikan komentar	Menampilkan info berhasil	Berhasil
		Menolak Buku	Menampilkan konfirmasi terhadap penolakan	Berhasil
		Menghapus Buku	Menampilkan konfirmasi terhadap penghapusan	Berhasil
10	UC-10	Mengelola Pengguna	Menampilkan konfirmasi terhadap penghapusan	Berhasil
11	UC-11	Menambah Kategori	Menampilkan info berhasil menambah	Berhasil

		Mengedit Kategori	Menampilkan info berhasil mengedit	Berhasil
		Menghapus Kategori	Menampilkan konfirmasi terhadap penghapusan	Berhasil
12	UC-12	Menambah Asset	Menampilkan info berhasil menambah	Berhasil
		Mengedit Asset	Menampilkan info berhasil mengedit	Berhasil
		Menghapus Asset	Menampilkan konfirmasi terhadap penghapusan	Berhasil
13	UC-13	Menambah Kosa Kata	Menampilkan info berhasil menambah	Berhasil
		Mengedit Kosa Kata	Menampilkan info berhasil mengedit	Berhasil
		Menghapus Kosa Kata	Menampilkan info berhasil menambah	Berhasil

#### 4.2.2 Pengujian Usabilitas

Pengujian usabilitas dilakukan pada 10 responden yang memiliki hobi menulis dan membaca.

Tabel 4.2 merupakan hasil pengujian usabilitas menggunakan kuesioner yang dilakukan dan dihitung menggunakan skala likert. Berdasarkan pengujian yang dilakukan didapat nilai persentasi rata-rata yaitu yang berarti sangat sesuai / sangat setuju dari pengguna untuk desain,

fitur dan proses yang ditampilkan dan dijalankan pada aplikasi ini berjalan dengan baik dan sesuai dengan harapan pengguna.

Pada proses pengujian didapatkan juga tanggapan yang diberikan responden terkait tentang aplikasi. Berikut merupakan beberapa tanggapan yang diberikan responden:

- a. Aplikasi yang digunakan terbilang cukup baik karena dapat menyediakan buku-buku cerita yang dapat dibacakan pada anak-anak dengan mudah karena terdapat fitur pencarian buku. Juga pada bagian *download* buku terbilang baik karena buku dapat di *download* agar pengguna dapat membacanya secara *online*.
- b. Masih terdapat kekurangan pada bagian pembuatan cerita yaitu kurangnya pilihan untuk mengatur kanvas tempat pembuatan cerita. Dinilai masih kurang untuk dapat mengatur seperti tidak dapat mengatur ukuran kanvas cerita sesuai dengan kemauan pengguna.
- c. Dalam bagian asset dinilai sudah cukup baik dan dapat mendukung jalannya cerita, hanya saja lebih baik lagi untuk diperbanyak asset yang bisa mendukung cerita agar lebih baik.

Tabel 4.2 Tabel Hasil Pengujian Usabilitas

No.	Pertanyaan	STS	KS	C	S	SS	Persentase
		1	2	3	4	5	
1	Bentuk formulir dan tampilan memasukan tulisan (teks cerita, form <i>login</i> dan pendaftaran) berjalan dengan baik			2	3	5	86%
2	Fitur pendaftaran dan <i>login</i> mudah digunakan dan berjalan dengan baik				4	6	92%
3	Fitur aplikasi membuat cerita (menulis, menggambar, menambah asset) berjalan dengan baik		1	4	2	3	74%
4	Fitur rekomendasi asset berjalan dengan baik		1	5	2	2	70%

5	Fitur mengelola buku (menghapus, mengedit)		1	2	3	4	80%
6	Bentuk tampilan membuat cerita (warna, tulisan, gambar) sesuai dan berjalan dengan baik.			4	4	2	76%
7	Bentuk tampilan membaca cerita berjalan dengan baik		1	3	4	2	74%
8	Letak komponen (link, tombol, dan menu) dapat ditemukan.		1	4	1	4	76%
9	Terdapat informasi jika melakukan kesalahan yang disengaja atau tidak disengaja		1	2	3	4	80%
10	Bahasa yang digunakan dapat dipahami dengan baik			2	3	5	86%
<b>Rerata persentase</b>							<b>79.5%</b>

### 4.3 Kelebihan dan Kekurangan Sistem

Berdasarkan pengujian pada aplikasi diketahui kelebihan dan kekurangan yang didapatkan, yaitu:

#### 4.3.1 Kelebihan Aplikasi

Kelebihan pada Aplikasi Pembuat Buku Cerita Berbasis Web adalah:

- a. Pengguna anggota dapat menggunakan fitur yang cukup untuk dapat membuat buku cerita, contohnya seperti cerita buku komik atau sebagainya.
- b. Aplikasi dapat memberikan rekomendasi *asset* untuk memudahkan pengguna dalam menambahkan *asset* pada cerita.
- c. Pengguna dapat mencari dan membaca buku walau belum melakukan pendaftaran sebagai anggota
- d. Aplikasi menggunakan bahasa Indonesia yang cukup dimengerti.

#### 4.3.2 Kekurangan Aplikasi

Kekurangan pada Aplikasi Pembuat Buku Cerita Berbasis Web adalah:

- a. Masih terdapat fitur yang kurang untuk dapat mengoptimalkan dalam proses membuat cerita, seperti fitur pengaturan kanvas.
- b. Panduan penggunaan dinilai masih kurang untuk menunjukkan tata cara penggunaan dalam membuat cerita.
- c. Panduan penggunaan dinilai masih kurang untuk menunjukkan tata cara penggunaan dalam membuat cerita Aplikasi masih terbatas dapat diakses oleh perangkat *desktop* saja, masih tidak dapat diakses oleh perangkat *mobile*.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Kesimpulan yang didapatkan dari pengembangan dan pengujian Aplikasi Pembuat Buku Cerita Berbasis adalah:

- a. Berdasarkan review pada aplikasi yang sudah ada, dibangun aplikasi yang serupa dengan menggunakan Bahasa Indonesia dan memiliki fitur tambahan berupa fitur rekomendasi asset untuk memudahkan penggunaan aplikasi dalam membuat cerita.
- b. Aplikasi yang dibangun dapat memudahkan pengguna dalam berkreasi membuat suatu tulisan dengan gambar ilustrasi yang mendukung. Pengguna juga dapat membaca cerita yang telah dibuat oleh pengguna lainnya. Sehingga pengguna dapat saling berbagi tulisan cerita.
- c. Aplikasi diharapkan dapat memicu dan meningkatkan minat menulis dan membaca bagi setiap kalangan masyarakat Indonesia dan dapat turut ikut dalam mendukung pendidikan Indonesia.

#### 5.2 Saran

Dari hasil yang didapat dari proses pengujian aplikasi ini masih terdapat beberapa kekurangan, yang dapat digunakan untuk pengembangan aplikasi ini menjadi lebih baik, yaitu:

- a. Memperbaiki desain tampilan agar memberikan *user experience* yang lebih baik, baik dengan perangkat *desktop* maupun perangkat *mobile*.
- b. Menambahkan fitur yang memberikan pengguna keluasaan dalam mengatur canvas tempat menulis cerita.
- c. Memperbaiki desain file unduhan cerita agar dapat memberikan gambar yang lebih jelas.
- d. Menambahkan fitur-fitur seperti suara dan fitur *login* yang lebih mudah seperti *login* via *email* google dan sebagainya.



## DAFTAR PUSTAKA

- Abbas, S. (2006). *Pembelajaran Bahasa Indonesia yang Efektif di Sekolah Dasar*. Jakarta: DIRJENDIKTI.
- Fabric.js. (2018, 8 1). *Fabric.js*. Retrieved from Fabric.js Docs: <http://fabricjs.com/docs/>
- Gaszella, W. P. (2017, September 21). *Makalah Tentang Tokenisasi, Stopword Removal, dan Stemming*. Retrieved from [http://wellapudia.blogspot.com:](http://wellapudia.blogspot.com/)  
<http://wellapudia.blogspot.com/2017/09/makalah-tentang-tokenisasi-stopword.html>
- Irfan Ardhiyanto, A. P. (2017, September 22). *Penerapan Tokenisasi, Stopword Removal, dan Stemming*. Retrieved from [aprianapanca.wordpress.com:](http://aprianapanca.wordpress.com/)  
<https://aprianapanca.wordpress.com/2017/09/22/penerapan-tokenisasi-stopword-removal-dan-stemming/>
- Krisiandi. (2017, 2 30). *PISA dan Daya Baca Bangsa*. Retrieved from Kompas.com: <https://nasional.kompas.com/read/2017/04/30/11135891/pisa.dan.daya.baca.bangsa>
- Kurniawan, B., Effendi, S., & Salim, O. S. (2012). Klasifikasi Konten Berita Dengan Metode Text Mining . *JURNAL DUNIA TEKNOLOGI INFORMASI Vol. 1, No. 1*, 14-19.
- Laravel. (2018, 8 9). *Eloquent: Getting Started*. Retrieved from Laravel: <https://laravel.com/docs/5.6/eloquent>
- Mujilawati, S. (2016). PRE-PROCESSING TEXT MINING PADA DATA TWITTER . *Seminar Nasional Teknologi Informasi dan Komunikasi (SENTIKA)*, 49.
- OECD. (2016). *PISA 2015 Results (Volume I): Excellence and Equity in Education*. Paris: OECD.
- Probowati, T. E. (2014). *Peningkatan Kemampuan Menulis Puisi Melalui Metode Sugesti Imajinasi Lagu Pada Siswa Kelas V SDN Blimbing 3 Malang oleh Titik Eka Probawati*. Malang: Jurusan KSDP, Fakultas Ilmu Pendidikan, Universitas Negeri Malang.
- Saputri, K. (2017). *FAKTOR- FAKTOR YANG MEMPENGARUHI LITERASI ANAK KELAS 1 SD NEGERI 20 BANDA ACEH*. Banda Aceh: Universitas Syiah Kuala.
- Sentari. (2002). *Pengaruh Strategi Belajar Mental Imagery terhadap Recall Bacaan*. Skripsi. Depok: Fakultas Psikologi Universitas Indonesia.
- Siswo, M. U. (Januari 2013). Implementasi Stemmer Tala pada Aplikasi Berbasis Web. *Jurnal Teknologi Informasi DINAMIK Volume 18, No.1.*, 41-45.

- Soetopo, V. A. (1997). *Peran Kepadatan Tampilan Bacaan pada Layar Komputer terhadap Penggunaan Strategi Mendalam (In-Depth Strategy) pada Kegiatan Membaca Siswa-Siswi SMU*. Depok: Fakultas Psikologi Universitas Indonesia.
- Tala, F. Z. (2003). *A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia*. Amsterdam : Universiteit van Amsterdam .
- W3schools. (2018, Agustus 1). *W3schools*. Retrieved from W3schools:  
<https://www.w3schools.com/>
- Yuliangsih. (2014, 12 15). *Literasi Indonesia Sangat Rendah*. Retrieved from Republika.co.id:  
<https://www.republika.co.id/berita/koran/didaktika/14/12/15/ngm3g840-literasi-indonesia-sangat-rendah>

## **LAMPIRAN**