

# **RANCANG BANGUN APLIKASI BACABUKU!**

## **LAPORAN TUGAS BESAR**

Dibuat untuk memenuhi salah satu tugas Praktikum Pengembangan Berorientasi  
Objek

Oleh :

**Ilham Syahidatul Rajab (1806043)**

**Tania Agusviani Wahidah (1806052)**



**PROGRAM STUDI TEKNIK INFORMATIKA**

**SEKOLAH TINGGI TEKNOLOGI GARUT**

**2021**

## **ABSTRAK**

**Abstrak** – Dewasa ini membaca buku tidak hanya dilakukan secara fisik, namun bisa juga dengan ebook. Masih banyak orang yang belum bisa mengakses ebook dengan mudah. Oleh karena itulah aplikasi BacaBuku! dibuat. Tujuan dari pembuatan aplikasi ini adalah untuk memudahkan orang-orang untuk mengakses ebook serta sebagai salah satu bentuk digitalisasi. Hasil dari pembuatan aplikasi ini yaitu aplikasi berjalan dengan baik dari menampilkan data, memasukkan data, dan menghapus data. Kesimpulan yang diperoleh diantaranya mempermudah pencarian buku.

**Kata Kunci :** *Aplikasi, Buku, PHP, Web*

## **KATA PENGANTAR**

Puji syukur alhamdulillah kami panjatkan ke hadirat Tuhan Yang Maha Esa, karena telah melimpahkan rahmat-Nya berupa kesempatan dan pengetahuan sehingga laporan ini bisa selesai tepat pada waktunya.

Laporan ini sudah selesai disusun secara maksimal dengan bantuan dari berbagai pihak sehingga bisa memperlancar pembuatan laporan ini. Untuk itu kami menyampaikan banyak terima kasih kepada semua pihak yang sudah ikut berkontribusi di dalam pembuatan laporan ini.

Kami berharap semoga laporan ini bisa menambah pengetahuan para pembaca. Namun terlepas dari itu, kami memahami bahwa laporan ini masih jauh dari kata sempurna, sehingga kami sangat mengharapkan kritik serta saran yang bersifat membangun demi terciptanya laporan selanjutnya yang lebih baik lagi.

Garut, 16 Januari 2021

Penyusun

## DAFTAR ISI

<b>ABSTRAK .....</b>	<b>i</b>
<b>KATA PENGANTAR.....</b>	<b>ii</b>
<b>DAFTAR ISI.....</b>	<b>iii</b>
<b>DAFTAR LAMPIRAN .....</b>	<b>iv</b>
<b>1. PENDAHULUAN.....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Tujuan Pembuatan Aplikasi .....	1
1.3. Ruang Lingkup Aplikasi.....	1
1.4. Sistematika Penulisan.....	1
<b>2. LANDASAN TEORI.....</b>	<b>3</b>
<b>3. METODOLOGI PEKERJAAN.....</b>	<b>5</b>
<b>4. HASIL DAN PEMBAHASAN .....</b>	<b>7</b>
4.2. Proses Aplikasi BacaBuku!.....	19
<b>5. KESIMPULAN DAN SARAN .....</b>	<b>25</b>
5.1. Kesimpulan .....	25
5.2. Saran .....	25
<b>DAFTAR PUSTAKA .....</b>	<b>26</b>

## DAFTAR LAMPIRAN

Gambar 1. Metode Waterfall .....	5
Gambar 2. Use Case BacaBuku! .....	7
Gambar 3. Activity Diagram (Login) .....	12
Gambar 4. Activity Diagram (Baca Buku) .....	12
Gambar 5. Activity Diagram (Tambah Buku) .....	13
Gambar 6. Activity Diagram (Hapus Buku) .....	13
Gambar 7. Activity Diagram (Edit Buku) .....	14
Gambar 8. Sequence Diagram (Login) .....	14
Gambar 9. Sequence Diagram (Baca Buku) .....	15
Gambar 10. Sequence Diagram (Tambah Buku) .....	15
Gambar 11. Sequence Diagram (Hapus Buku) .....	15
Gambar 12. Sequence Diagram (Edit Buku) .....	16
Gambar 13. Collaboration Diagram (Login) .....	16
Gambar 14. Collaboration Diagram (Baca Buku) .....	16
Gambar 15. Collaboration Diagram (Tambah Buku) .....	17
Gambar 16. Collaboration Diagram (Hapus Buku) .....	17
Gambar 17. Collaboration Diagram (Edit Buku) .....	17
Gambar 18. Class Diagram (Login) .....	18
Gambar 19. Class Diagram (Baca Buku) .....	18
Gambar 20. Class Diagram (Tambah Buku) .....	18
Gambar 21. Class Diagram (Hapus Buku) .....	19
Gambar 22. Class Diagram (Edit Buku) .....	19
Gambar 23. Tampilan Awal .....	20
Gambar 24. Tampilan Lihat Buku .....	20
Gambar 25. Tampilan Login Admin .....	21
Gambar 26. Tampilan Halaman Admin .....	21
Gambar 27. Tampilan Input Buku .....	22
Gambar 28. Tampilan Edit Penulis .....	22
Gambar 29. Tampilan Daftar Penulis .....	23
Gambar 30. Tampilan Edit Buku .....	24

# **1. PENDAHULUAN**

## **1.1. Latar Belakang**

Di zaman sekarang ini hampir seluruh aktivitas dilakukan menggunakan media elektronik dan tentunya di sertai dengan internet. Bahkan dalam kegiatan baca membaca buku pun sekarang hampir seluruh aktivitasnya menggunakan media internet.

Maka dari itu dengan dibuatnya aplikasi BacaBuku! supaya dapat mempermudah aktivitas dalam membaca buku dan tentu di dalamnya itu terdapat berbagai jenis buku tidak hanya satu jenis tetapi ada beberapa jenis seperti novel, komik, sains dan banyak lainnya. Jadi aplikasi BacaBuku! dapat di pakai oleh semua kalangan pembaca. Dan tentunya juga aplikasi ini akan membantu dalam menyimpan beberapa koleksi buku dengan baik. Semoga dengan adanya aplikasi BacaBuku! Dapat meningkatkan minat baca untuk semua orang yang awalnya tidak suka membaca.

## **1.2. Tujuan Pembuatan Aplikasi**

Adapun tujuan pembuatan aplikasi ini adalah sebagai berikut.

1. Untuk sebagai salah satu langkah buku.
2. Untuk mempermudah membaca buku secara gratis dan legal.

## **1.3. Ruang Lingkup Aplikasi**

Adapun yang menjadi batasan-batasan dalam membuat aplikasi ini adalah sebagai berikut.

1. *User* hanya bisa membaca buku tanpa bisa memasukkan buku.
2. Admin bisa menambahkan, mengedit, dan menghapus buku.
3. Maksimal ukuran file buku yang bisa diunggah adalah 20 *megabyte*.

## **1.4. Sistematika Penulisan**

Laporan kerja praktik ini disusun dengan sistematika sebagai berikut :

1. **PENDAHULUAN**, berisi latar belakang, tujuan pekerjaan, ruang lingkup aplikasi, serta sistematika penulisan;

2. **LANDASAN TEORI**, berisi kutipan definisi, konsep, teori dan metode terkait tujuan pekerjaan yang bersumber dari sejumlah pustaka;
3. **METODOLOGI PEKERJAAN**, berisi penjelasan tentang tahapan kerja berikut aktivitas dan teknik yang digunakan, serta pelaksana pekerjaan, waktu, tempat, dan sumber daya yang digunakan, dengan memperhatikan landasan teori;
4. **HASIL DAN PEMBAHASAN**, berisi penjelasan tentang hasil pekerjaan, pengetahuan dan keterampilan empiris yang diperoleh, serta bagaimana kearifan lokal diperhatikan di dalam pekerjaan dengan merujuk kepada pustaka;
5. **KESIMPULAN DAN SARAN**, berisi ringkasan penelitian berdasarkan hasil dan pembahasan, serta rekomendasi pekerjaan terkait kelemahan kemampuan mahasiswa atau hambatan di tempat kerja.

## 2. LANDASAN TEORI

Aplikasi berasal dari kata *application* yang artinya penerapan, lamaran, penggunaan. Secara istilah aplikasi adalah program siap pakai yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain dan dapat digunakan oleh sasaran yang dituju. (Arif, Wanda, & Masudi, 2013)

Pengertian Aplikasi Web Pada awalnya aplikasi web dibangun dengan menggunakan bahasa yang disebut HTML (*HyperText Markup Language*). Pada perkembangan berikutnya, sejumlah skrip dan objek dikembangkan untuk memperluas kemampuan HTML seperti PHP dan ASP skrip dan Applet pada objek. Aplikasi Web dapat dibagi menjadi dua jenis yaitu aplikasi web statis dan dinamis. Web statis dibentuk dengan menggunakan HTML. Kekurangan aplikasi seperti ini terletak pada keharusan untuk memelihara program secara terus menerus untuk mengikuti setiap perkembangan yang terjadi. Kelemahan ini diatasi oleh model aplikasi web dinamis. Pada aplikasi web dinamis, perubahan informasi dalam halaman web dilakukan tanpa perubahan program tetapi melalui perubahan data. Sebagai implementasi, aplikasi web dapat dikoneksikan ke basis data sehingga perubahan informasi dapat dilakukan oleh operator dan tidak menjadi tanggung jawab dari webmaster. Arsitektur aplikasi web meliputi klien, web server, middleware dan basis data. Klien berinteraksi dengan web server. Secara internal, web server berkomunikasi dengan middleware dan middleware yang berkomunikasi dengan basis data. Contoh middleware adalah PHP dan ASP. Pada mekanisme aplikasi web dinamis, terjadi tambahan proses yaitu server menerjemahkan kode PHP menjadi kode HTML. Kode PHP yang diterjemahkan oleh mesin PHP yang akan diterima oleh klien. (Arif, Wanda, & Masudi, 2013)

PHP adalah *tool* untuk pembuatan halaman web dinamis. Pada awalnya PHP merupakan kependekan dari *Personal Home Page* (Situs Personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama FI (*Form Interpreted*), yang wujudnya berupa sekumpulan script yang digunakan untuk mengolah data *form* dari web. Saat ini PHP adalah singkatan dari PHP: Hypertext Preprocessor, sebuah kepanjangan rekursif, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri: PHP: *Hypertext Preprocessor*. (Lutfi, 2017)



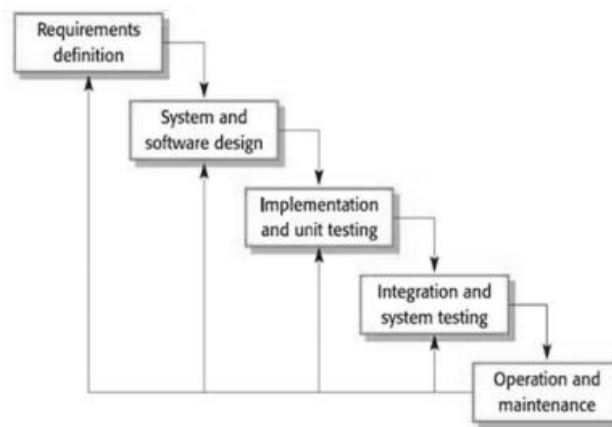
*Website* yaitu kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (hyperlink). (Krisnayani, Arthana, & Darmawiguna, 2016)

UML merupakan singkatan dari “*Unified Modelling Language*” yaitu suatu metode permodelan secara visual untuk sarana perancangan sistem berorientasi objek, atau definisi UML yaitu sebagai suatu bahasa yang sudah menjadi standar pada visualisasi, perancangan dan juga pendokumentasian sistem *software*. CRUD merupakan akronim dari *Create*, *Read*, *Update*, dan *Delete*.

1. *Create* dapat diartikan membuat atau menambahkan data menggunakan aplikasi yang sudah dibuat, ketika kita menambahkan data di aplikasi nantinya akan otomatis akan masuk ke dalam *database* yang sudah dibuat.
2. *Read* yaitu menampilkan atau membaca data yang bisa dilihat menggunakan aplikasi data mana yang ingin ditampilkan maka akan tampil.
3. *Update* diartikan mengubah, kita tidak perlu repot ketika ingin mengubah data, cukup menggunakan aplikasi maka akan terubah dan *database* otomatis akan terubah.
4. *Delete* diartikan menghapus, fitur ini digunakan untuk menghapus data yang tidak diperlukan. Kita tidak perlu repot ketika ingin menghapus data, tidak perlu menghapus data lewat *database*.

### 3. METODOLOGI PEKERJAAN

Metode penelitian yang diterapkan pada penelitian ini adalah dengan pengembangan Metode *Waterfall*. Metode *Waterfall* merupakan model pengembangan sistem informasi yang sistematis dan sekuensial. Metode *Waterfall* memiliki tahapan-tahapan sebagai berikut :



**Gambar 1. Metode Waterfall**

#### 1. *Requirements analysis and definition*

Layanan sistem, kendala, dan tujuan ditetapkan oleh hasil konsultasi dengan pengguna yang kemudian didefinisikan secara rinci dan berfungsi sebagai spesifikasi sistem.

#### 2. *System and software design*

Tahapan perancangan sistem mengalokasikan kebutuhan-kebutuhan sistem baik perangkat keras maupun perangkat lunak dengan membentuk arsitektur sistem secara keseluruhan. Perancangan perangkat lunak melibatkan identifikasi dan penggambaran abstraksi sistem dasar perangkat lunak dan hubungannya.

#### 3. *Implementation and unit testing*

Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program atau unit program. Pengujian melibatkan verifikasi bahwa setiap unit memenuhi spesifikasinya.

#### 4. *Integration and system testing*

Unit-unit individu program atau program digabung dan diuji sebagai sebuah sistem lengkap untuk memastikan apakah sesuai dengan kebutuhan perangkat lunak atau tidak. Setelah pengujian, perangkat lunak dapat dikirimkan ke *customer*.

5. *Operation and maintenance*

Biasanya (walaupun tidak selalu), tahapan ini merupakan tahapan yang paling panjang. Sistem dipasang dan digunakan secara nyata. *Maintenance* melibatkan pembetulan kesalahan yang tidak ditemukan pada tahapan-tahapan sebelumnya, meningkatkan implementasi dari unit sistem, dan meningkatkan layanan sistem sebagai kebutuhan baru.

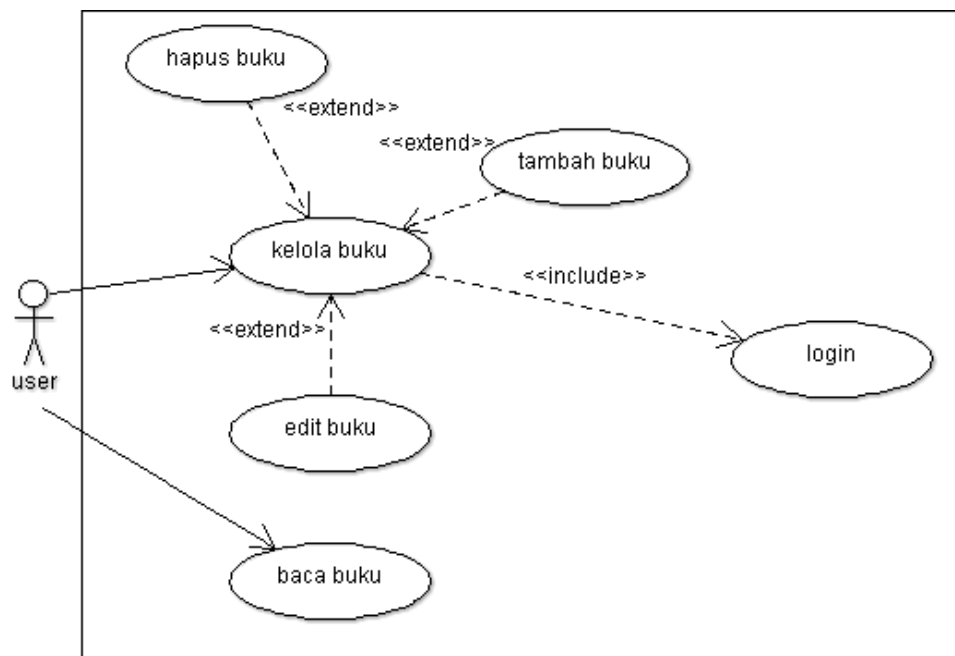
## 4. HASIL DAN PEMBAHASAN

### 4.1. Metode Perancangan Berorientasi Objek

Perancangan merupakan tahapan yang dilakukan setelah tahap analisis. Perancangan berorientasi objek yang digunakan Penulis adalah *Unified Modelling Language* (UML). Diagram perilaku di UML terdiri dari:

#### 4.1.1. Use Case Diagram

Pada *use case diagram* terdapat 1 aktor dan 5 *use case*. Dapat dilihat pada gambar di bawah ini.



**Gambar 2. Use Case BacaBuku!**

Berikut merupakan *scenario* dari *use case* di atas, dapat dilihat pada tabel di bawah ini.

a. Login

Nama Use Case : Login

Type : Primary

Actor : user

Tujuan : Untuk mengidentifikasi user sebelum masuk ke halaman web

Skenario Normal	
Aktor	Sistem
	Sistem meminta email dan password
User memasukkan email dan password	
	Sistem mengonfirmasi email dan password
	Sistem memasukkan user ke web
Skenario alternatif	
	Sistem meminta email dan password
User memasukkan email dan password	
	Sistem mengonfirmasi email dan password
	Menampilkan pesan login tidak valid
User memasukkan email dan password yang valid	
	Sistem memasukkan user ke web

b. Baca Buku

Nama Use Case : Baca Buku

Type : Primary

Actor : user

Tujuan : Untuk baca buku secara online

Skenario Normal	
Aktor	Sistem
User memilih buku yang akan dibaca	
	Buku yang dipilih akan muncul
Skenario alternatif	

User memilih buku yang akan dibaca	
	Sistem menampilkan pesan buku tidak tersedia
User memilih buku yang akan dibaca	
	Buku yang dipilih akan muncul

c. Tambah Buku

Nama Use Case : Tambah Buku

Type : Secondary

Actor : user

Tujuan : Untuk menambah buku

Skenario Normal	
Aktor	Sistem
User menekan tombol tambah	
	Sistem menyediakan form input buku
User mengisi judul, penulis, jumlah halaman, tahun terbit, sinopsis, file maks 20mb, cover	
	Sistem menyimpan data ke database
Skenario alternatif	
User menekan tombol tambah	
	Sistem menyediakan form input buku
User mengisi judul, penulis, jumlah halaman, tahun terbit, sinopsis, file > 20mb, cover	
	Sistem menampilkan pesan error
User mengisi judul, penulis,	

jumlah halaman, tahun terbit, sinopsis, file maks 20mb, cover	
	Sistem menyimpan data ke database

d. Hapus Buku

Nama Use Case : Hapus Buku

Type : Secondary

Actor : user

Tujuan : Untuk menghapus buku

Skenario Normal	
Aktor	Sistem
User menekan tombol “-” / hapus	
	Sistem menampilkan pesan konfirmasi
User menekan tombol ya	
	Sistem menghapus data dari database

e. Edit Buku

Nama Use Case : Edit Buku

Type : Secondary

Actor : user

Tujuan : Untuk memperbarui data buku

Skenario Normal	
Aktor	Sistem
User menekan tombol “edit”	
	Sistem menyediakan form input buku
User mengedit judul, penulis, jumlah halaman, tahun terbit, sinopsis, file maks 20mb, cover	

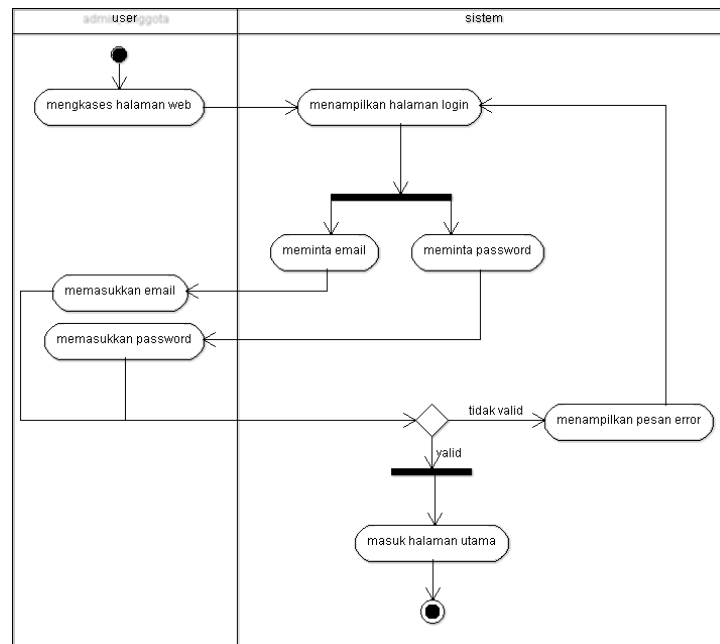
	Sistem memperbarui data ke database
<b>Skenario alternatif</b>	
User menekan tombol “edit”	
	Sistem menyediakan form input buku
User mengedit judul, penulis, jumlah halaman, tahun terbit, sinopsis, file > 20mb, cover	
	Sistem menampilkan pesan error
User mengedit judul, penulis, jumlah halaman, tahun terbit, sinopsis, file maks 20mb, cover	
	Sistem memperbarui data ke database



#### 4.1.2. Activity Diagram

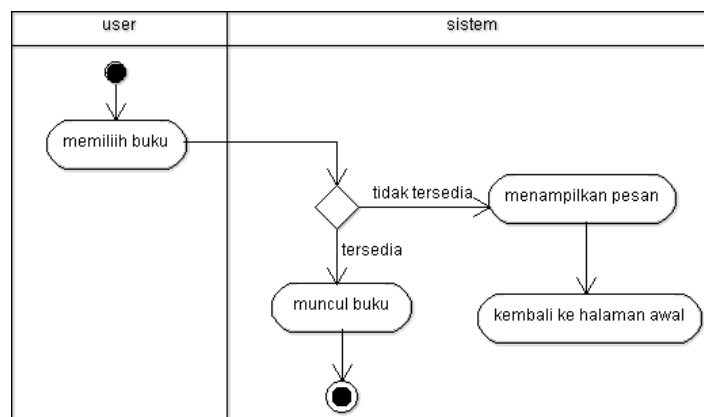
Activity Diagram merupakan rancangan aliran aktivitas atau aliran kerja dalam sebuah sistem yang menjelaskan tentang alir kegiatan dalam program yang sedang dirancang, bagaimana proses alir berawal, keputusan yang mungkin terjadi, dan bagaimana sistem akan berakhir. Activity Diagram pada pembuatan aplikasi ini dapat dilihat pada gambar berikut:

##### a. Login



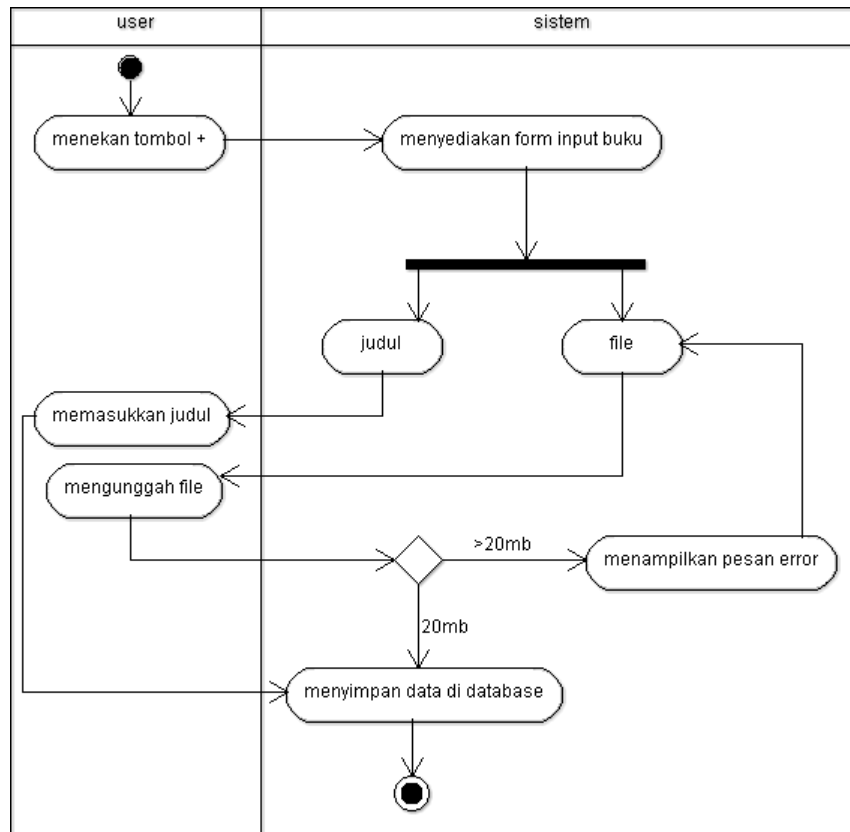
**Gambar 3. Activity Diagram (Login)**

##### b. Baca Buku



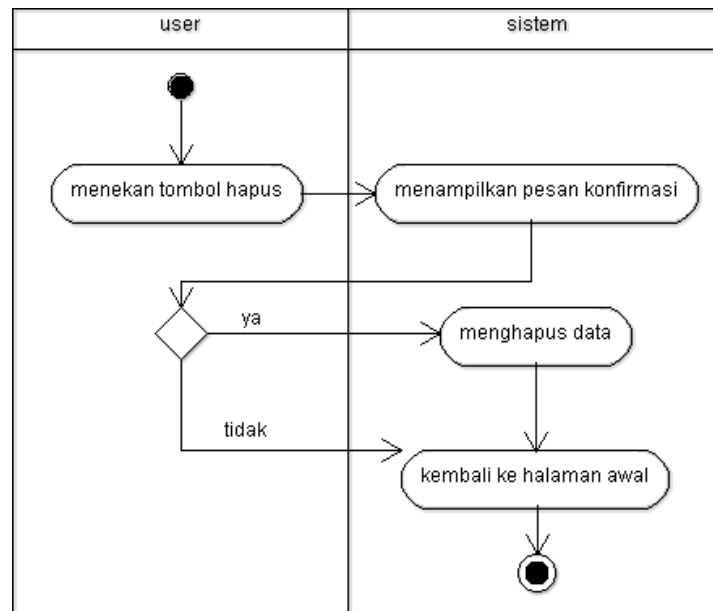
**Gambar 4. Activity Diagram (Baca Buku)**

##### c. Tambah Buku



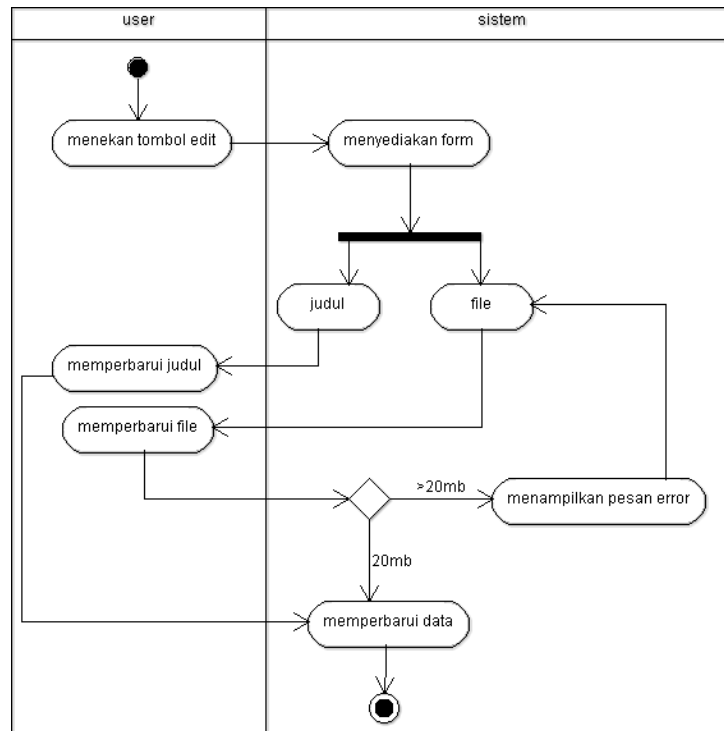
**Gambar 5. Activity Diagram (Tambah Buku)**

d. Hapus Buku



**Gambar 6. Activity Diagram (Hapus Buku)**

e. Edit Buku

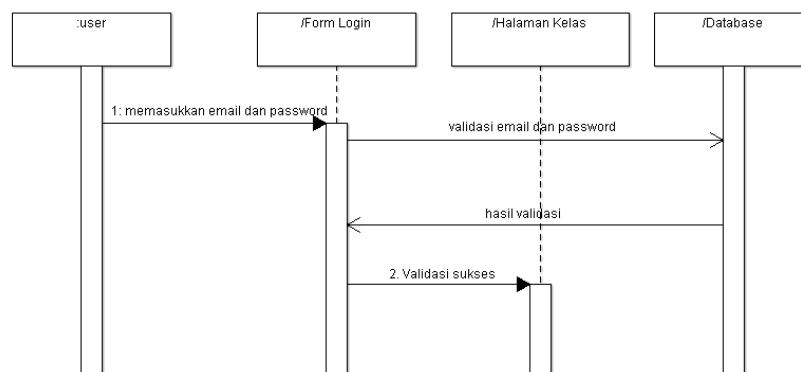


**Gambar 7. Activity Diagram (Edit Buku)**

#### 4.1.3. Sequence Diagram

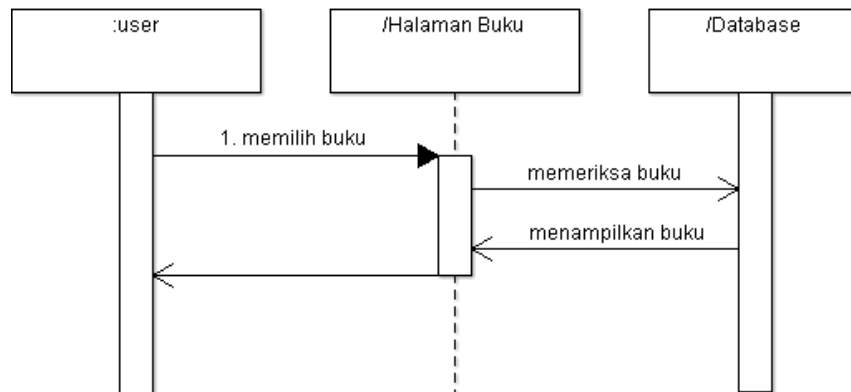
Sequence Diagram yaitu suatu diagram yang menjelaskan tentang interaksi objek dan menunjukkan (memberi tanda atau petunjuk) untuk menjelaskan perilaku pada sebuah skenario dan menggambarkan bagaimana entitas dan sistem berinteraksi, termasuk pesan yang dipakai saat interaksi. Sequence diagram pada pembuatan aplikasi ini dapat dilihat pada gambar berikut:

##### a. Login



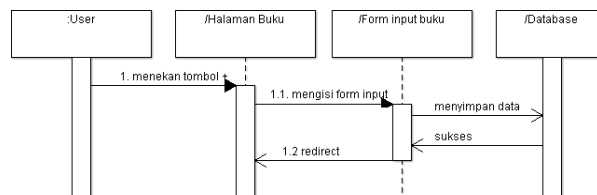
**Gambar 8. Sequence Diagram (Login)**

b. Baca Buku



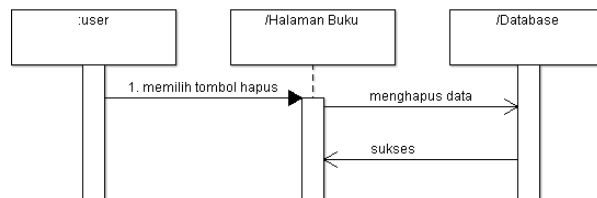
**Gambar 9. Sequence Diagram (Baca Buku)**

c. Tambah Buku



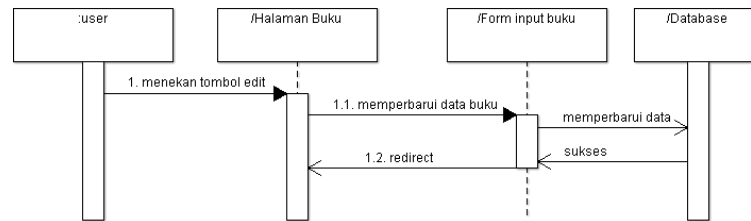
**Gambar 10. Sequence Diagram (Tambah Buku)**

d. Hapus Buku



**Gambar 11. Sequence Diagram (Hapus Buku)**

e. Edit Buku

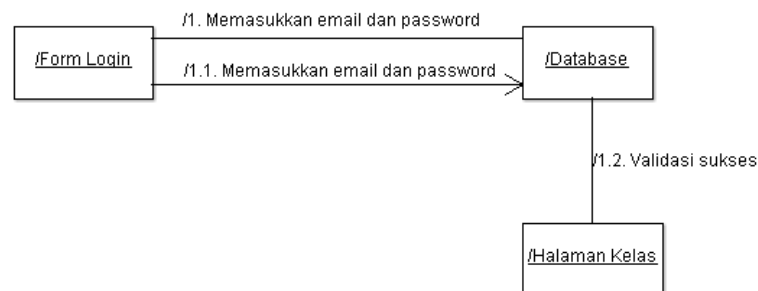


**Gambar 12. Sequence Diagram (Edit Buku)**

#### 4.1.5. Collaboration Diagram

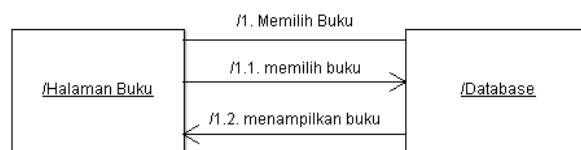
Collaboration Diagram adalah suatu diagram yang memperlihatkan / menampilkan pengorganisasian interaksi yang terdapat disekitar objek (seperti halnya sequence diagram) dan hubungan terhadap yang lainnya. Collaboration Diagram pada pembuatan aplikasi ini sebagai berikut:

a. Login



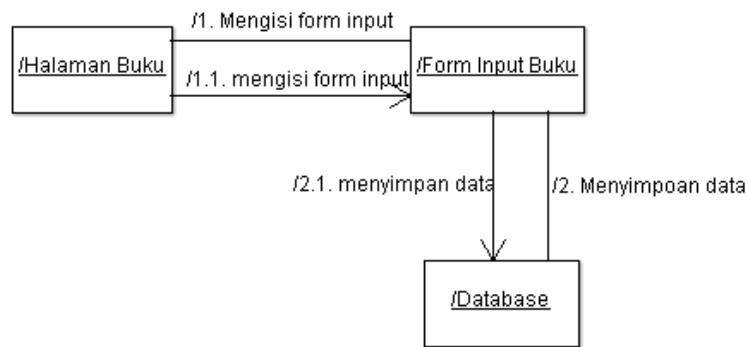
**Gambar 13. Collaboration Diagram (Login)**

b. Baca Buku



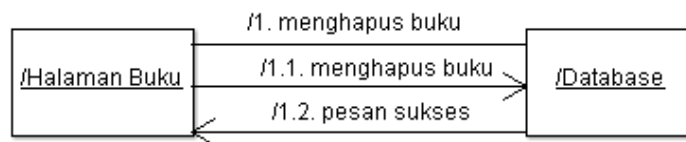
**Gambar 14. Collaboration Diagram (Baca Buku)**

c. Tambah Buku



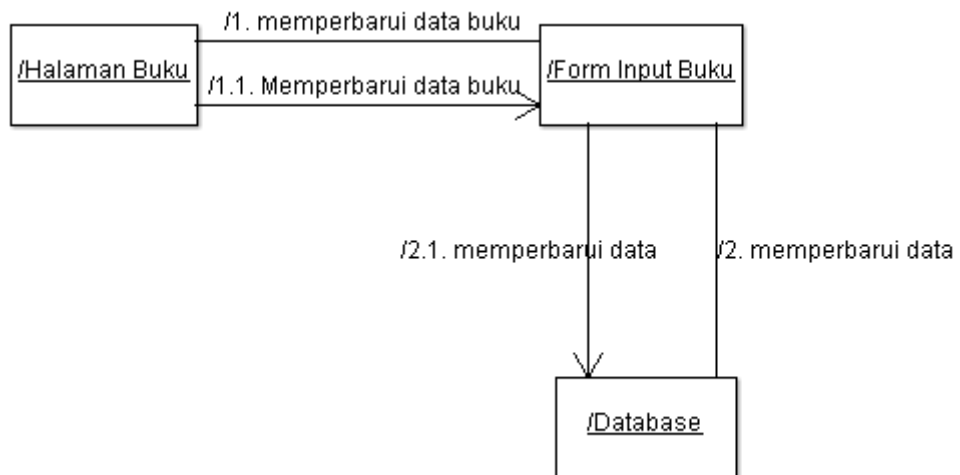
**Gambar 15. Collaboration Diagram (Tambah Buku)**

d. Hapus Buku



**Gambar 16. Collaboration Diagram (Hapus Buku)**

e. Edit Buku

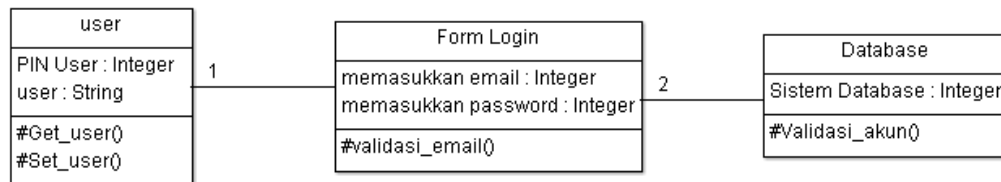


**Gambar 17. Collaboration Diagram (Edit Buku)**

#### 4.1.6. Class Diagram

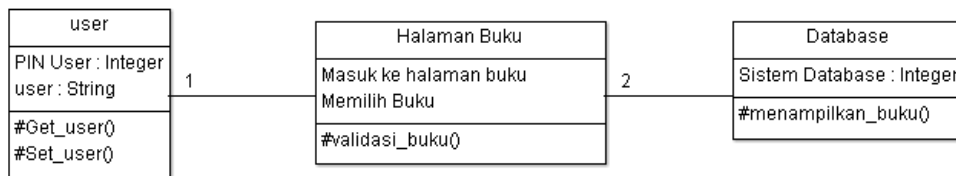
Class Diagram adalah salah satu jenis diagram yang paling berguna di UML, hal ini karena dapat dengan jelas memetakan struktur sistem tertentu dengan memodelkan kelas, atribut, operasi serta hubungan antar objek. Class Diagram pada pembuatan aplikasi ini bisa dilihat gambar di bawah ini:

##### a. Login



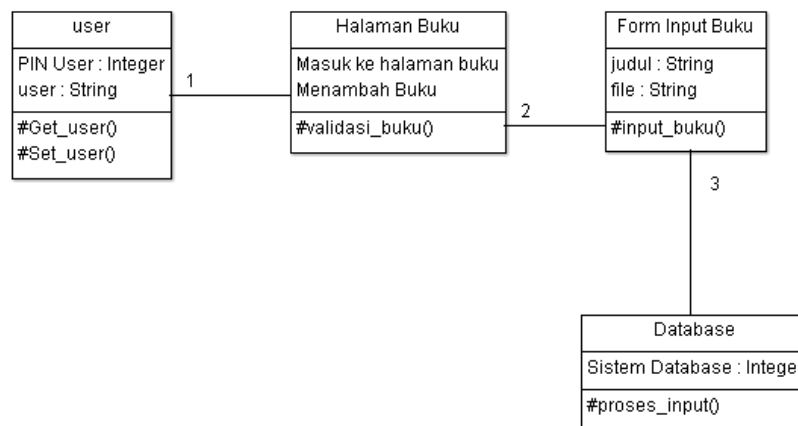
**Gambar 18. Class Diagram (Login)**

##### b. Baca Buku



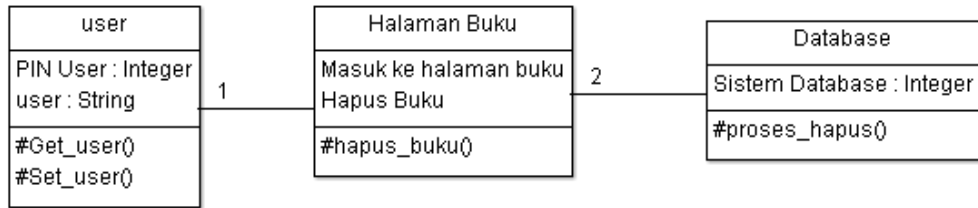
**Gambar 19. Class Diagram (Baca Buku)**

##### d. Tambah Buku



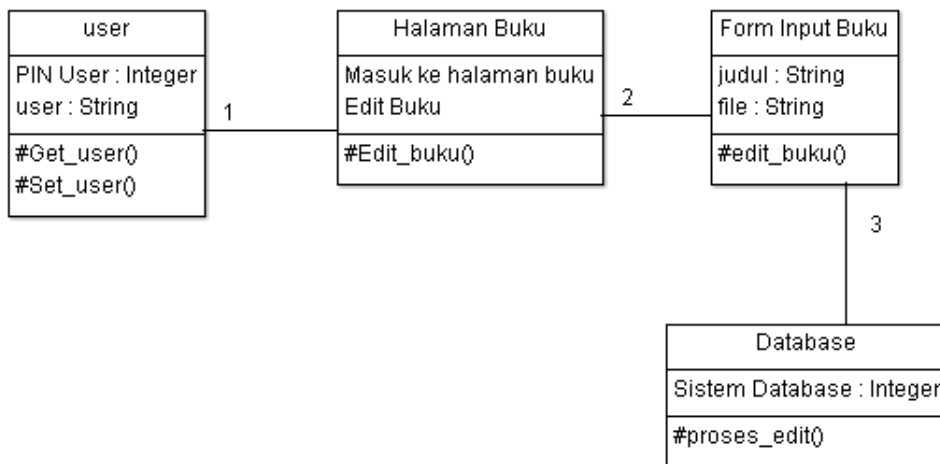
**Gambar 20. Class Diagram (Tambah Buku)**

e. Hapus Buku



**Gambar 21. Class Diagram (Hapus Buku)**

f. Edit Buku



**Gambar 22. Class Diagram (Edit Buku)**

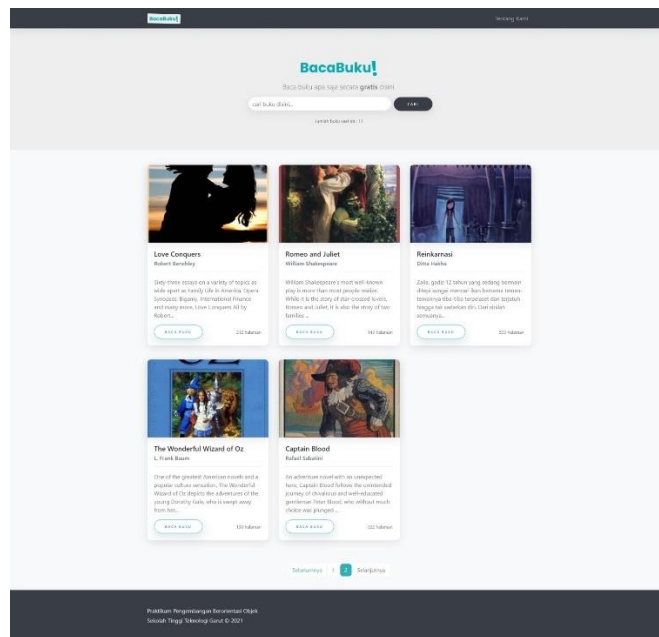
#### 4.2. Proses Aplikasi BacaBuku!

Setelah dianalisa dan dirancang dengan menggunakan metode berorientasi objek dengan alat bantu UML, maka berikut ini adalah aplikasi BacaBuku! dapat dijelaskan pada gambar ilustrasi berikut.



### 4.2.1. Tampilan Utama BacaBuku!

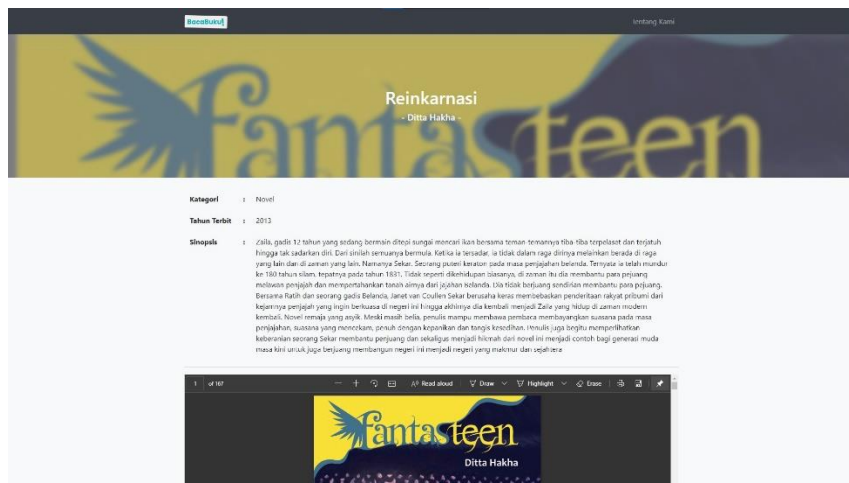
Gambar di bawah ini merupakan tampilan awal pada aplikasi BacaBuku!



**Gambar 23. Tampilan Awal**

### 4.2.2. Tampilan Lihat Buku

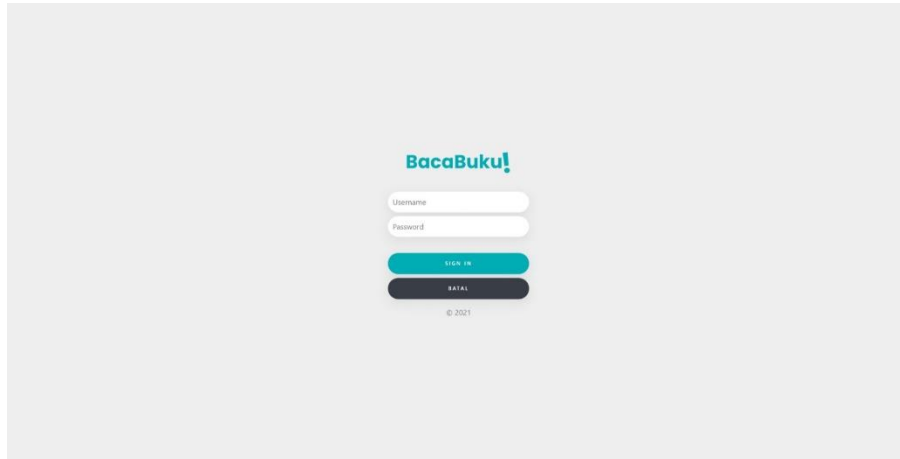
Gambar di bawah ini adalah tampilan buku yang telah kita klik yang kemudian akan muncul mengenai kategori, tahun terbit, dan sinopsis.



**Gambar 24. Tampilan Lihat Buku**

#### 4.2.3. Tampilan Login Admin

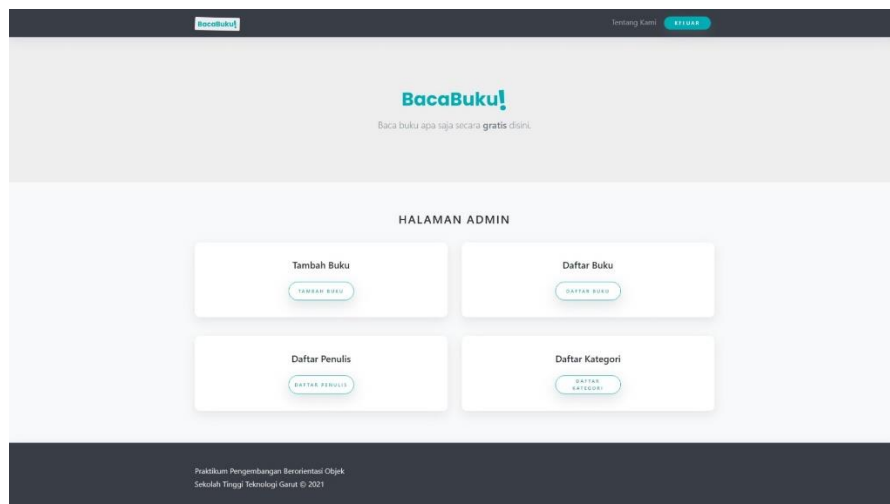
Gambar di bawah ini menunjukkan tampilan Login untuk Admin. Dimana admin dapat melakukan aktivitas seperti menambahkan, mengedit dan menghapus buku.



**Gambar 25. Tampilan Login Admin**

#### 4.2.4. Tampilan Halaman Admin

Pada halaman admin akan menampilkan beberapa pilihan menu seperti tambah buku, daftar buku, daftar penulis, daftar kategori.



**Gambar 26. Tampilan Halaman Admin**

#### 4.2.5. Tampilan Input Buku

Gambar ini adalah tampilan Input Buku yang di dalamnya terdapat form mengenai hal- hal tentang buku yang akan di masukkan.

The screenshot shows the 'TAMBAH BUKU' form in the BacaBuku! application. The form includes the following fields and controls:

- Judul:** A text input field.
- Penulis:** A dropdown menu with a search icon, accompanied by 'LUNAS' and 'MASUKKAN' buttons.
- Kategori:** A dropdown menu with a search icon, accompanied by 'LUNAS' and 'MASUKKAN' buttons.
- Jumlah Halaman:** A text input field.
- Jumlah Halaman:** A text input field.
- Tahun Terbit:** A text input field.
- Tahun Terbit:** A text input field.
- Sinopsis:** A large text area for the synopsis.
- File PDF:** A text input field.
- File PDF:** A text input field with a 'Browse' button.
- Cover:** A text input field.
- Cover:** A text input field with a 'Browse' button.

At the bottom of the form are three buttons: 'SIMPAN DATA' (green), 'BATAL' (red), and 'BATAL' (black). The footer of the application is visible at the bottom.

**Gambar 27. Tampilan Input Buku**

#### 4.2.6. Tampilan Edit Penulis

Gambar ini merupakan tampilan Edit Penulis dan di dalamnya menampilkan edit kategori buku.

The screenshot shows the 'EDIT KATEGORI' form in the BacaBuku! application. The form includes the following fields and controls:

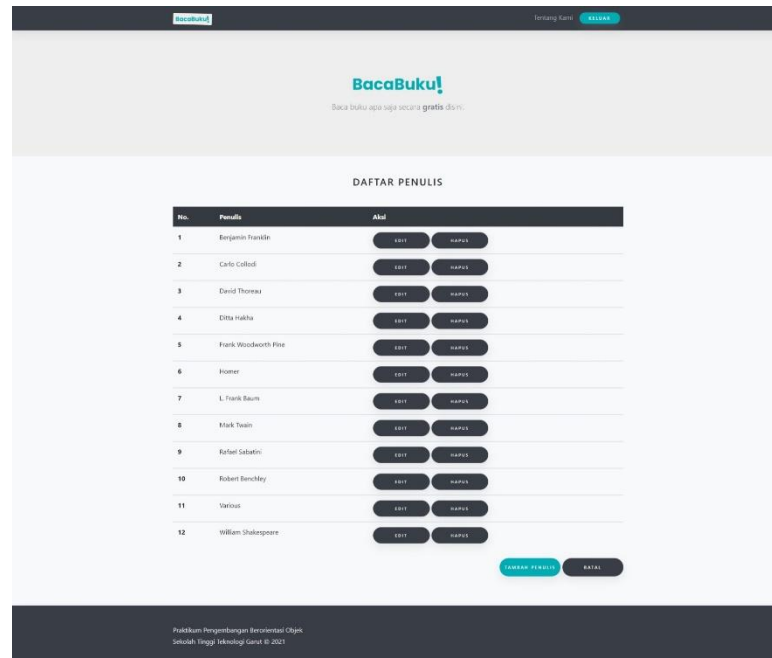
- Kategori:** A text input field with the value 'Biografi'.

At the bottom of the form are two buttons: 'UPDATE DATA' (green) and 'BATAL' (black). The footer of the application is visible at the bottom.

**Gambar 28. Tampilan Edit Penulis**

#### 4.2.7. Tampilan Daftar Penulis

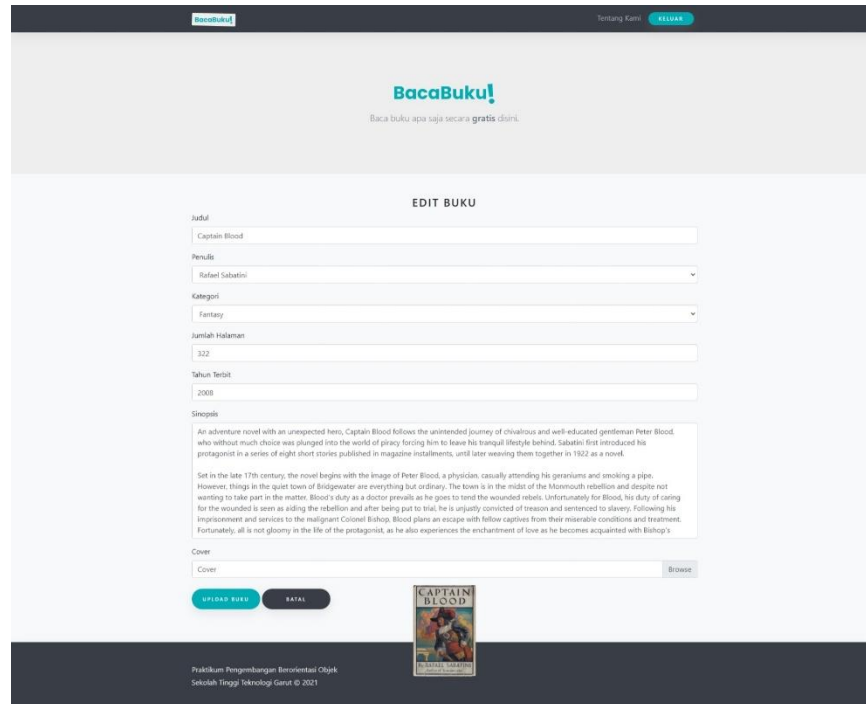
Gambar ini menampilkan daftar dari penulis buku yang sudah diinputkan ke dalam sistem.



**Gambar 29. Tampilan Daftar Penulis**

#### 4.2.8. Tampilan Edit Buku

Pada tampilan edit buku admin dapat mengedit buku yang sudah diinputkan ke dalam aplikasi jika terdapat kesalahan dalam penulisan identitas buku.



**BacaBuku!**  
Baca buku apa saja secara gratis disini.

**EDIT BUKU**

Judul  
Captain Blood

Penulis  
Rafael Sabatini


Kategori  
Fantasy

Jumlah Halaman  
322

Tahun Terbit  
2008

Sinopsis  
An adventure novel with an unexpected hero, Captain Blood follows the unintended journey of chivalrous and well-educated gentleman Peter Blood, who without much choice was plunged into the world of piracy forcing him to leave his tranquil lifestyle behind. Sabatini first introduced his protagonist in a series of eight short stories published in magazine installments, and later weaving them together in 1922 as a novel.  
Set in the late 17th century, the novel begins with the image of Peter Blood, a physician, casually attending his granitum and smoking a pipe. However, things in the quiet town of Bridgewater are everything but ordinary. The town is in the midst of the Monmouth rebellion and despite not wanting to take part in the matter, Blood's duty as a doctor prevails as he goes to tend the wounded rebels. Unfortunately for Blood, his duty of caring for the wounded is seen as aiding the rebellion and after being put to trial, he is unjustly convicted of treason and sentenced to slavery. Following his imprisonment and service to the malignant Colonel Bishop, Blood plans an escape with fellow captives from their miserable conditions and treatment. Fortunately, all is not gloomy in the life of the protagonist, as he also experiences the enchantment of love as he becomes acquainted with Bishop's daughter.

Cover  
Cover



Praktikum Pengembangan Berorientasi Objek  
Sekolah Tinggi Teknologi Garut © 2021

**Gambar 30. Tampilan Edit Buku**

## **5. KESIMPULAN DAN SARAN**

### **5.1. Kesimpulan**

Dalam pembuatan aplikasi ini dilakukan analisa dan perancangan sistem rencana pembuatan aplikasi BacaBuku! dan pembuatan implementasi rancangan sistem. Adapun hal-hal yang dapat disimpulkan adalah :

1. Metode analisa dan desain berorientasi objek sangat membantu dalam menentukan kebutuhan (*requirement*) pada pembuatan *website* rencana pembuatan aplikasi BacaBuku!
2. Aplikasi ini dapat mempermudah kita untuk proses pencarian buku dan membaca buku secara *online*.

### **5.2. Saran**

Aplikasi ini masih banyak kekurangan semoga dapat dikembangkan lagi sehingga aplikasi ini bisa lebih bermanfaat. Dan semoga ke depannya akan ada banyak fitur yang lebih baik.

## DAFTAR PUSTAKA

- Arif, S. N., Wanda, A. P., & Masudi, A. (2013). Aplikasi Administrasi Perpustakaan Berbasis Web SMK Swasta Brigjend Katamso Medan. *Jurnal SAINTIKOM*, 12(1), 25-36.
- Krisnayani, P., Arthana, I. R., & Darmawiguna, I. M. (2016). Analisa Usability Pada Website UNDIKSHA Dengan Menggunakan Metode Heuristic Evaluation. *Kumpulan Artikel Mahasiswa Pendidikan Teknik Informatika (KARMAPATI)*, 5(2).
- Lutfi, A. (2017). Sistem Informasi Akademik Madrasah Aliyah Salafiyah Syafi'iyah Menggunakan PHP dan MySQL. *AiTech*, 3(2), 104 - 112.