

BAB IV

HASIL DAN ANALISIS PENELITIAN

4.1 Analisis Perancangan Sistem

Unified Modeling Language (UML) adalah bahasa grafis untuk memvisualisasikan, menspesifikasikan, membangun, dan mendokumentasikan elemen dari sistem perangkat lunak pada analisis perancangan sistem. UML menawarkan cara standar untuk menulis cetak biru sistem, termasuk hal-hal konseptual seperti proses bisnis dan fungsi sistem serta hal-hal konkret seperti pernyataan bahasa pemrograman, skema *database*, dan komponen perangkat lunak yang dapat digunakan kembali. UML terdiri dari diagram yang terbagi menjadi tiga kategori yaitu *structure*, *behaviour* dan *interaction*. *Structure* menunjukkan kelas yang membentuk sistem seperti *class diagram*. *Behaviour* menunjukkan bagaimana sistem merespon suatu proses seperti *activity* dan *use case diagram*. *Interaction* menunjukkan interaksi antar objek pada sistem seperti *sequence diagram*. *Class diagram* menampilkan kelas-kelas yang ada pada sistem dengan method (*fungsi*) dan field (*atribut*). *Use case diagram* menampilkan interaksi antar *actor* (pengguna) dalam sistem tertentu. *Sequence diagram* menampilkan urutan pesan yang dikirim antar objek.

Berdasarkan wawancara pada metode pengumpulan data, sistem berjalan Taman Bacaan Masyarakat (TBM) di Komplek Taman Mangu Indah dirancang sesuai dengan kebutuhan pengguna yaitu pengunjung dan pengurus TBM. Sistem berjalan ini sejatinya belum memiliki sistem yang terotomatisasi dengan kata lain proses operasional berjalan manual. Hal ini berdampak terhadap proses pemantauan sirkulasi buku seperti peminjaman dan pengembalian buku sehingga menyebabkan kelalaian baik bagi pengurus dan pengunjung karena minimnya pemberitahuan atau notifikasi

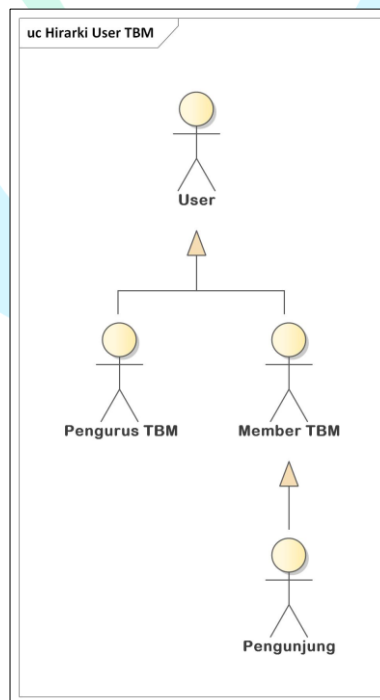
pengembalian buku. Selain itu, minimnya sistem untuk memproses perekapan data buku menyebabkan pemasukan dan pencarian data buku yang tidak akurat serta kurangnya transparansi dan aksesibilitas bagi pengunjung untuk melihat katalog buku yang tersedia di TBM. Maka dari itu diperlukanlah sebuah usulan sistem pengelolaan taman bacaan masyarakat.

4.2 Perancangan Diagram Sistem Usulan

Sistem ini dirancang sebagai sistem pengelolaan taman bacaan masyarakat yang membantu pendataan buku dan sirkulasi buku. Berikut merupakan diagram-diagram berbasis UML yang menggambarkan alur sistem berjalan:

4.2.1 Use Case Diagram

Sistem yang diusulkan memiliki tiga aktor yaitu pengunjung, member dan pengurus. Berikut deskripsi dan hirarki pengguna yang terstruktur tertera pada Gambar 4.1.



Gambar 4. 1 Hirarki User Taman Bacaan Masyarakat

1) Pengunjung

Pengunjung dapat mengakses informasi umum seperti katalog buku, kontak, jadwal operasional dll. Pada katalog buku pengunjung bisa melakukan pencarian buku berdasarkan judul, penulis, ISBN atau kategori serta melihat detail buku tersebut. Selain itu pengunjung dapat melakukan registrasi untuk menjadi member jika ingin melakukan peminjaman buku.

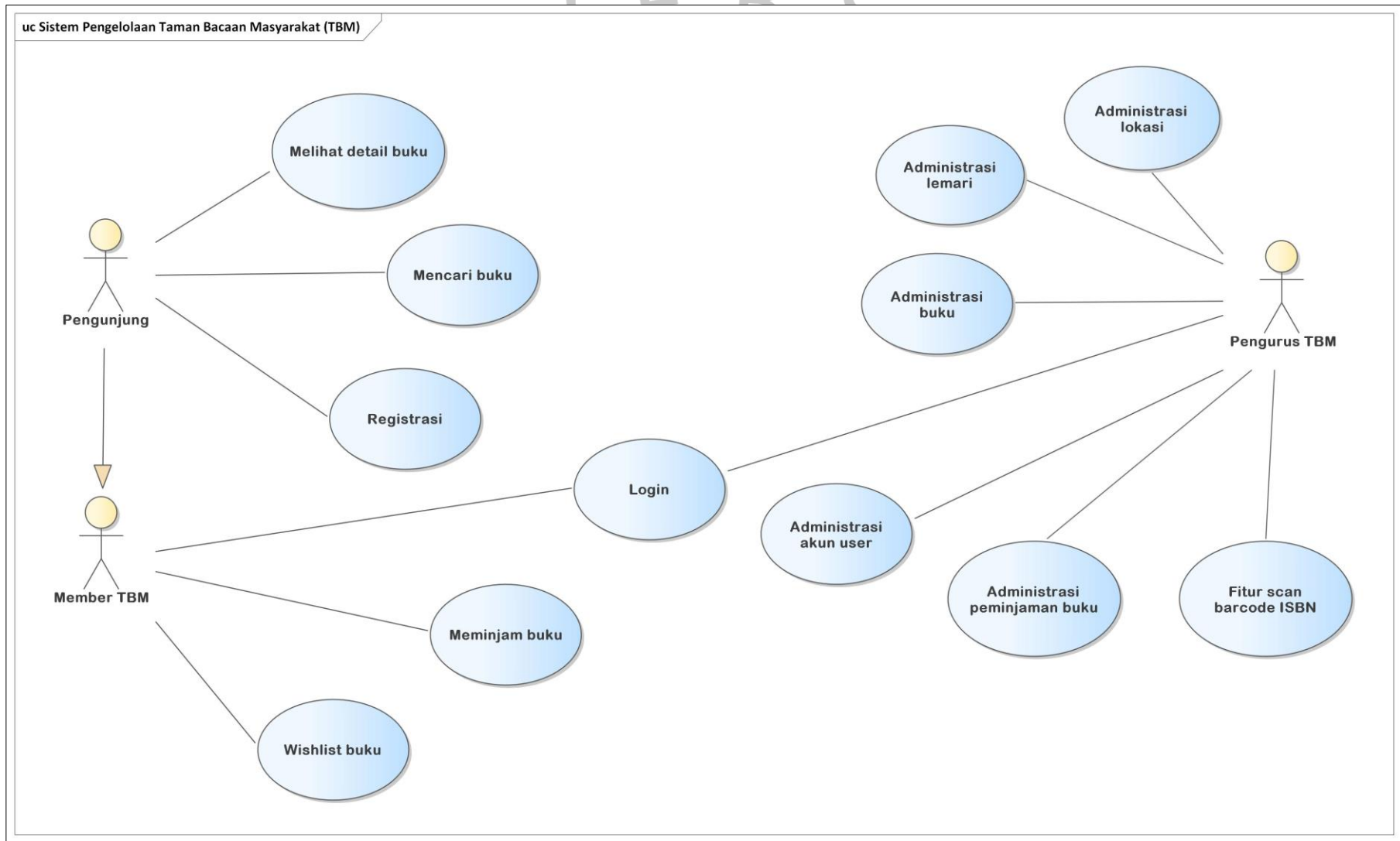
2) Member

Member bisa melakukan peminjaman buku yang nantinya akan dikonfirmasi oleh pengurus, sehingga nanti member bisa mengambil buku tersebut di tempat. Member menentukan berapa lama waktu peminjaman buku, sehingga sistem akan otomatis memberi peringatan pengembalian buku jika sudah jatuh tempo. Member tidak bisa melakukan peminjaman buku lainnya jika masih ada buku yang terpinjam atau belum dikembalikan.

3) Pengurus TBM

Pengguna ini merupakan staf taman baca yang memiliki akses penuh ke sistem sehingga dapat mengelola buku, member serta peminjaman dan pengembalian buku. Agar memudahkan pengurus dalam menambah data buku maka terdapat fitur untuk memindai barcode ISBN buku yang dapat menampilkan data buku dari API *Google Books*.

Sistem ini memfasilitasi interaksi antara pengunjung, member, dan pengurus dalam mengelola taman baca dan layanan peminjaman buku. Hal ini digambarkan dengan jelas dalam *use case diagram* keseluruhan sistem taman bacaan masyarakat yang dapat dilihat pada Gambar 4.2.



Gambar 4. 2 Use Case Diagram Sistem TBM

4.2.2 Use Case Specification

Pada sistem pengelolaan taman baca *use case specification* berfungsi untuk mendeskripsikan sebuah proses yang dicapai aktor atau pengguna pada suatu sistem dengan skenario tertentu. Umumnya *use case specification* berbentuk sebuah tabel yang berisi *cases* (proses), *actor* (pengguna), *pre-conditions* (kondisi sebelum) and *post-conditions* (kondisi sesudah) serta *success and alternative scenario*. Skenario sukses merupakan kondisi dimana proses berjalan sesuai dengan hasil yang diharapkan, sedangkan skenario alternatif merupakan kondisi jika proses menghasilkan sesuatu diluar hasil yang diharapkan contohnya seperti kesalahan *input* pengguna, *input* data yang tidak lengkap dan lain-lain. Berikut merupakan *use case specification* dari sistem taman bacaan masyarakat dapat dilihat pada Tabel 4.1 sampai dengan 4.13.

1) Login

Tabel 4. 1 Use Case Specification Login

<i>Use Case</i>	Login	
<i>Actor</i>	Member dan Pengurus TBM	
<i>Pre-Condition</i>	Sudah melakukan registrasi atau akun sudah terbuat di <i>database</i>	
<i>Post-Condition</i>	Masuk ke dalam halaman sistem sesuai dengan masing-masing <i>role user</i>	
<i>Success Scenario</i>	Aktor	Sistem
	1. Membuka halaman <i>form login</i> kemudian memasukan <i>e-mail</i> dan <i>password</i> yang valid	2. Menampilkan status sukses login, kemudian mengalihkan halaman ke masing-masing <i>role user</i>
<i>Alternative Scenario</i>	<ul style="list-style-type: none"> - Menampilkan status gagal login jika <i>e-mail</i> atau <i>password</i> salah - Menampilkan status harap isi form yang kosong jika input <i>e-mail</i> dan <i>password</i> belum terisi. - Member dapat mengklik link button lupa <i>password</i>, jika ingin <i>me-reset password</i>. 	

2) Registrasi

Tabel 4. 2 Use Case Specification Registrasi Member TBM

Use Case	Registrasi Detail Buku	
Actor	Pengunjung	
Pre-Condition	Berada di halaman registrasi buku	
Post-Condition	Melakukan login sebagai member untuk peminjaman buku.	
Success Scenario	Aktor	Sistem
	1. Mengisi form registrasi member seperti nama lengkap, alamat <i>e-mail</i> , nomor yang bisa dihubungi, alamat rumah dan <i>password</i> .	
		2. Menyimpan data registrasi dan menampilkan status registrasi sukses, kemudian mengalihkan ke halaman aktivasi akun.
	3. Mengklik link aktivasi akun menggunakan <i>e-mail</i> yang diisi sebelumnya.	
		4. Mengalihkan ke halaman <i>e-mail</i> pengunjung agar bisa diaktivasi.
	5. Mengecek ke <i>inbox</i> email pengunjung, kemudian mengklik <i>link button</i> "aktivasi akun".	
		6. Mengalihkan ke halaman login dan menampilkan notifikasi akun berhasil teraktivasi.
Alternative Scenario	<ul style="list-style-type: none"> - Menampilkan status <i>password</i> dan konfirmasi <i>password</i> tidak sesuai jika isi keduanya tidak sama. - Menampilkan status harap isi form yang kosong jika nama lengkap, alamat <i>e-mail</i>, nomor yang bisa dihubungi, alamat rumah atau <i>password</i> belum diisi - Menampilkan status alamat <i>e-mail</i> atau nomor yang bisa dihubungi sudah terdaftar, coba gunakan <i>e-mail</i> atau nomor lain. - Menampilkan <i>link</i> aktivasi sudah kadaluarsa, coba registrasi ulang. 	

3) Mencari Buku

Tabel 4. 3 Use Case Specification Mencari Buku

Use Case	Mencari Buku	
Actor	Pengunjung dan Member TBM	
Pre-Condition	Berada di halaman katalog buku	
Post-Condition	Buku yang dicari akan tampil pada halaman katalog buku	
Success Scenario	Aktor	Sistem
	1. Membuka halaman katalog buku kemudian memasukan data buku yang dicari berdasarkan judul atau pengarang dan lain-lain.	
		2. Melakukan <i>filter</i> buku yang dicari berdasarkan judul atau pengarang, dan menampilkan buku yang dicari.
Alternative Scenario	<ul style="list-style-type: none"> - Menampilkan status buku yang anda cari tidak tersedia jika buku tidak ada di <i>database</i> - Menampilkan status harap isi form yang kosong jika <i>input</i> pencarian buku belum terisi 	

4) Melihat Detail Buku

Tabel 4.4 Use Case Specification Melihat Detail Buku

Use Case	Melihat Detail Buku	
Actor	Pengunjung dan Member TBM	
Pre-Condition	Berada di halaman katalog buku	
Post-Condition	Menampilkan detail buku seperti sinopsis, kategori, status ketersediaan, dan lain-lain.	
Success Scenario	Aktor	Sistem
	1. Memilih salah satu buku pada katalog dan mengklik <i>button</i> "detail".	
		2. Mengalihkan ke halaman detail buku berdasarkan id buku tersebut kemudian menampilkan detail buku seperti sinopsis, kategori, status ketersediaan, dan lain-lain.
Alternative Scenario	<ul style="list-style-type: none"> - Jika member sudah login, maka halaman detail buku akan menampilkan <i>button</i> untuk melakukan peminjaman dan <i>wishlist</i> buku. 	

5) Wishlist Buku

Tabel 4. 2 Use Case Specification Wishlist Buku

Use Case	Wishlist Buku	
Actor	Member TBM	
Pre-Condition	Berada di detail buku yang akan dipinjam	
Post-Condition	Melihat daftar <i>wishlist</i> buku di website	
Success Scenario	Aktor	Sistem
	1. Memilih salah satu buku pada katalog dan mengklik <i>button</i> “detail”.	
		2. Pada detail buku terdapat status <i>button icon</i> hati untuk menyimpan <i>wishlist</i> .
	3. Mengklik <i>button</i> “ <i>wishlist buku</i> ”	
		4. Menyimpan informasi ke tabel <i>wishlist</i> .
	5. Mengklik halaman daftar <i>wishlist</i> untuk melihat buku yang ingin dibaca.	
		6. Menampilkan <i>View</i> halaman daftar <i>wishlist</i> .
Alternative Scenario	- Member dapat me-remove <i>wishlist</i> buku dari daftar <i>wishlist</i> .	

6) Meminjam Buku

Tabel 4. 6 Use Case Specification Meminjam Buku

Use Case	Meminjam Buku	
Actor	Member TBM	
Pre-Condition	Berada di detail buku yang akan dipinjam	
Post-Condition	Melihat daftar peminjaman buku di website	
Success Scenario	Aktor	Sistem
	1. Memilih salah satu buku pada katalog dan mengklik <i>button</i> “detail”.	
		2. Pada detail buku terdapat <i>button</i> untuk meminjam buku
	3. Mengklik <i>button</i> “ <i>pinjam buku</i> ”	
		4. Menyimpan informasi ke tabel <i>peminjaman buku</i> .
	5. Mengklik halaman daftar <i>peminjaman buku</i> untuk melihat status buku	
		6. Menampilkan <i>View</i> halaman daftar <i>peminjaman</i> .
Alternative Scenario	- Member dapat membatalkan peminjaman buku dari daftar peminjaman.	

7) Scan ISBN Buku

Tabel 4. 7 Use Case Specification Scan Barcode ISBN Buku

<i>Use Case</i>	<i>Scan Barcode</i> ISBN Buku	
<i>Actor</i>	Pengurus TBM	
<i>Pre-Condition</i>	Berada di dashboard pengelolaan buku	
<i>Post-Condition</i>	Mengisi informasi buku lain seperti rak dan gudang pada form tambah buku	
<i>Success Scenario</i>	Aktor	Sistem
	1. Mengklik button “scan buku” pada menu navigasi	
		2. Menampilkan halaman tambah buku
	3. Mengklik button “scan barcode ISBN”.	
		4. Menampilkan menu scan barcode dengan opsi scan lewat kamera atau scan lewat gallery
	5. Mengklik button “request camera permission” jika ingin meng-scan dengan kamera.	
		6. Menampilkan opsi scan menggunakan kamera depan atau kamera belakang.
	7. Memilih salah satu opsi kamera, kemudian memindai barcode ISBN pada buku	
		8. Menampilkan status berhasil, dan otomatis mengisi data ISBN, judul buku, pengarang, penerbit dan tahun pada form buku sehingga memudahkan proses perekapan data buku.
<i>Alternative Scenario</i>	<ul style="list-style-type: none"> - Jika <i>browser</i> tidak memiliki akses pada kamera atau gallery, pengurus harus mengkonfigurasi izin <i>browser</i> terlebih dahulu. - Jika <i>barcode</i> ISBN tidak bisa terbaca karena ada buku yang tidak terdaftar pada API <i>Google Books</i> maka pengurus bisa mengetik manual data pada form tambah buku 	

8) Administrasi Buku

Tabel 4. 8 Use Case Specification Administrasi Buku

Use Case	Administrasi Buku	
Actor	Pengurus TBM	
Pre-Condition	Berada di dashboard tabel buku	
Post-Condition	Mengalihkan ke <i>dashboard</i> tabel buku	
Success Scenario	Aktor	Sistem
	1. Pengurus mengklik ke menu navigasi administrasi buku.	
		2. Menampilkan halaman administrasi buku
	3. Pengurus memilih row yang ingin dikelola datanya.	
	4. Pengurus menentukan <i>button</i> pada kolom aksi yaitu <i>button</i> “ubah detail buku, ubah lokasi buku, ubah status buku, <i>request</i> peminjaman by admin”.	
		5. Menampilkan <i>dialog box</i> berdasarkan <i>button</i> pada kolom aksi yang diklik.
	6. Pengurus mengubah data atau melakukan <i>request</i> peminjaman berdasarkan aksi yang dipilih, lalu mengklik simpan.	
		7. Database melakukan <i>query</i> untuk meng- <i>update</i> data buku atau meng- <i>input</i> data peminjaman.
Alternative Scenario	<ul style="list-style-type: none"> - Menampilkan status harap isi form yang kosong jika data buku yang belum diisi. - Menampilkan status buku sudah terdaftar jika memasukan buku dengan ISBN yang sama. - Jika <i>browser</i> tidak memiliki akses pada kamera atau <i>gallery</i>, pengurus harus mengkonfigurasi izin <i>browser</i> terlebih dahulu. 	

9) Administrasi Peminjaman

Tabel 4.9 Use Case Specification Administrasi Peminjaman Buku

Use Case	Administrasi Peminjaman Buku	
Actor	Pengurus TBM	
Pre-Condition	Berada di <i>dashboard</i> peminjaman buku	
Post-Condition	Mengalihkan ke halaman utama katalog buku	
Success Scenario	Aktor	Sistem
	1. Pengurus mengklik <i>button</i> “konfirmasi peminjaman” pada baris buku peminjam	
		2. Menampilkan form peminjaman
	3. Pengurus menentukan jadwal pengembalian buku sesuai dengan kesepakatan member TBM.	
		4. Menyimpan data peminjaman buku dan membarukannya pada <i>dashboard</i> sirkulasi buku dan riwayat peminjaman buku pada member.
	5. Pengurus mengklik <i>button</i> “konfirmasi pengembalian” jika buku sudah dikembalikan member di taman baca.	
		6. Menyimpan data pengembalian buku dan membarukannya pada <i>dashboard</i> sirkulasi buku dan riwayat peminjaman buku pada member.
Alternative Scenario	- Jika terdapat buku yang belum dikembalikan maka sistem akan mengubah status secara otomatis menjadi “terlambat”.	

10) Administrasi Lokasi

Tabel 4. 10 Use Case Specification Administrasi Lokasi Buku

Use Case	Administrasi Lokasi Buku	
Actor	Pengurus TBM	
Pre-Condition	Berada di <i>dashboard</i> lokasi	
Post-Condition	Mengalihkan ke <i>dashboard</i> lokasi	
Success Scenario	Aktor	Sistem
	1. Pengurus memilih button “tambah lokasi/ubah lokasi/ubah status lokasi”. sesuai kebutuhan.	
		2. Menampilkan form sesuai dengan button yang diklik.
	3. Pengurus mengisi data lokasi, kemudian mengklik button “simpan”	
		4. Menyimpan data yang sudah diubah dan menampilkan status lokasi berhasil bertambah.
Alternative Scenario	- Menampilkan status harap isi form yang kosong jika data lokasi yang belum diisi.	

11) Administrasi Lemari

Tabel 4. 11 Use Case Specification Administrasi Lemari Buku

Use Case	Administrasi Lemari Buku	
Actor	Pengurus TBM	
Pre-Condition	Berada di <i>dashboard</i> lemari	
Post-Condition	Mengalihkan ke <i>dashboard</i> lemari	
Success Scenario	Aktor	Sistem
	1. Pengurus mengisi field jumlah lemari, rak dan baris yang ingin diubah dan mengklik simpan.	
		2. Menyimpan data yang sudah diubah dan menampilkan status lokasi berhasil diubah.
Alternative Scenario	- Menampilkan status harap isi form yang kosong jika data lokasi yang belum diisi.	

12) Administrasi User TBM

Tabel 4. 12 Use Case Specification Administrasi User TBM

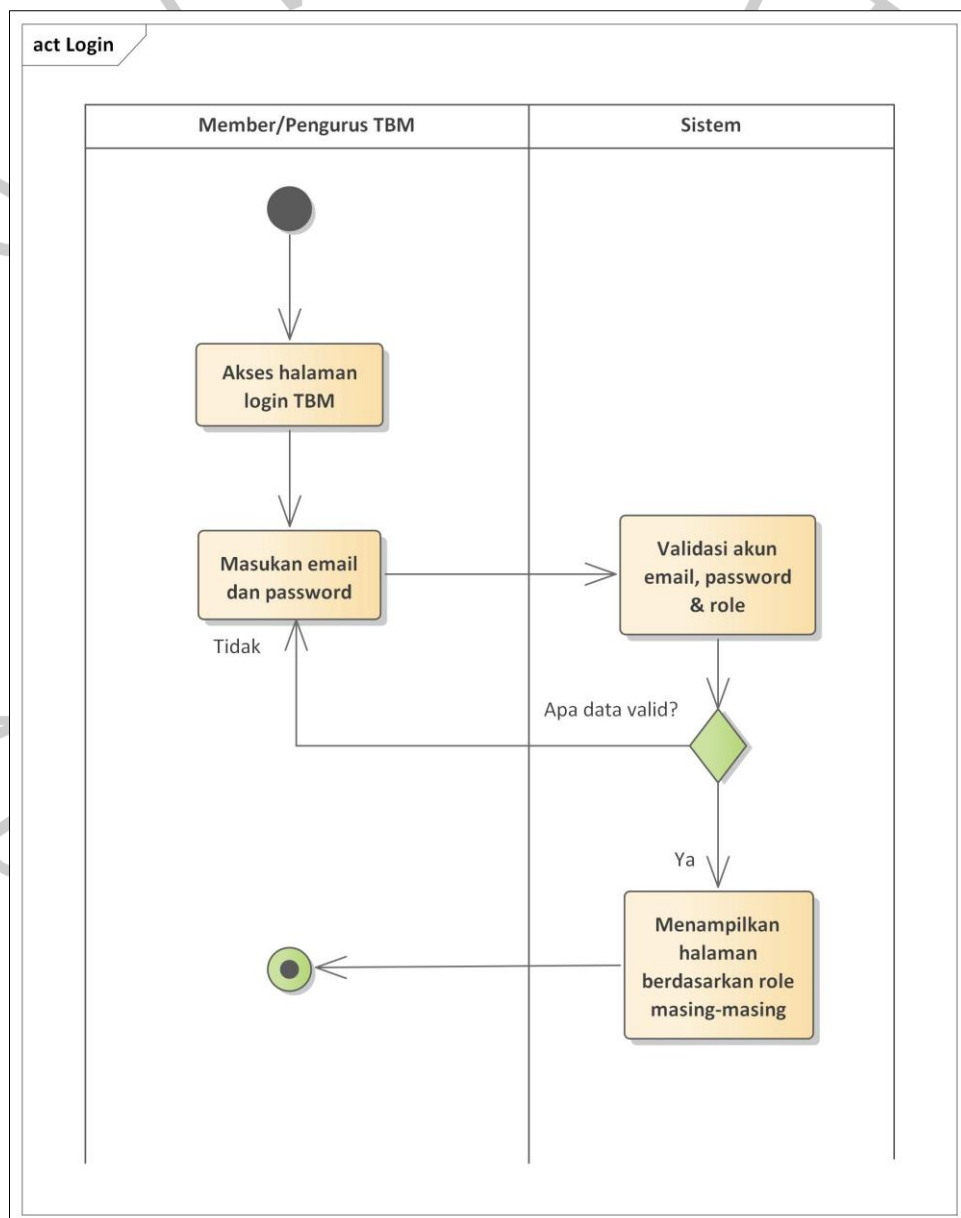
Use Case	Administrasi User TBM	
Actor	Pengurus TBM	
Pre-Condition	Berada di <i>dashboard</i> user	
Post-Condition	Mengalihkan ke <i>dashboard</i> user	
Success Scenario	Aktor	Sistem
	1. Pengurus memilih button “ubah role/reset password/ubah status user”. sesuai kebutuhan.	
		2. Jika button ubah role diklik, sistem akan mengubah role user ke admin dan sebaliknya. Jika button reset password diklik, sistem akan mengubah password ke default ke “123” Jika ubah status user diklik, sistem akan mengubah status user menjadi aktif atau sebaliknya.
Alternative Scenario	-	

4.2.3 Activity Diagram

Peneliti merancang *activity diagram* untuk memvisualisasikan alur kerja dan interaksi antar aktor dalam sistem pengelolaan taman bacaan masyarakat. *Activity diagram* memiliki elemen utama yang mewakili sebuah tindakan yaitu: *Partition* mewakili aktor yang terlibat pada proses bisnis dan digambarkan sebagai persegi panjang yang berisi elemen lainnya, *Action* mewakili langkah atau task yang dilakukan oleh aktor dan digambarkan sebagai persegi panjang dengan siku tumpul, *Initial Flow* mewakili titik awal alur proses bisnis dan digambarkan sebagai lingkaran berisi warna hitam, *Fork/join* merupakan elemen yang digunakan untuk menangani *task* paralel dan digambarkan sebagai garis panjang horizontal atau vertikal, *Final Flow* mewakili titik akhir proses bisnis dan digambarkan sebagai lingkaran dengan garis tepi hitam, sedangkan *Decision* mewakili percabangan berdasarkan kondisi tertentu dan digambarkan sebagai belah ketupat. Berikut rancangan *activity diagram* TBM pada Gambar 4.3 sampai Gambar 4.14.

1) Login

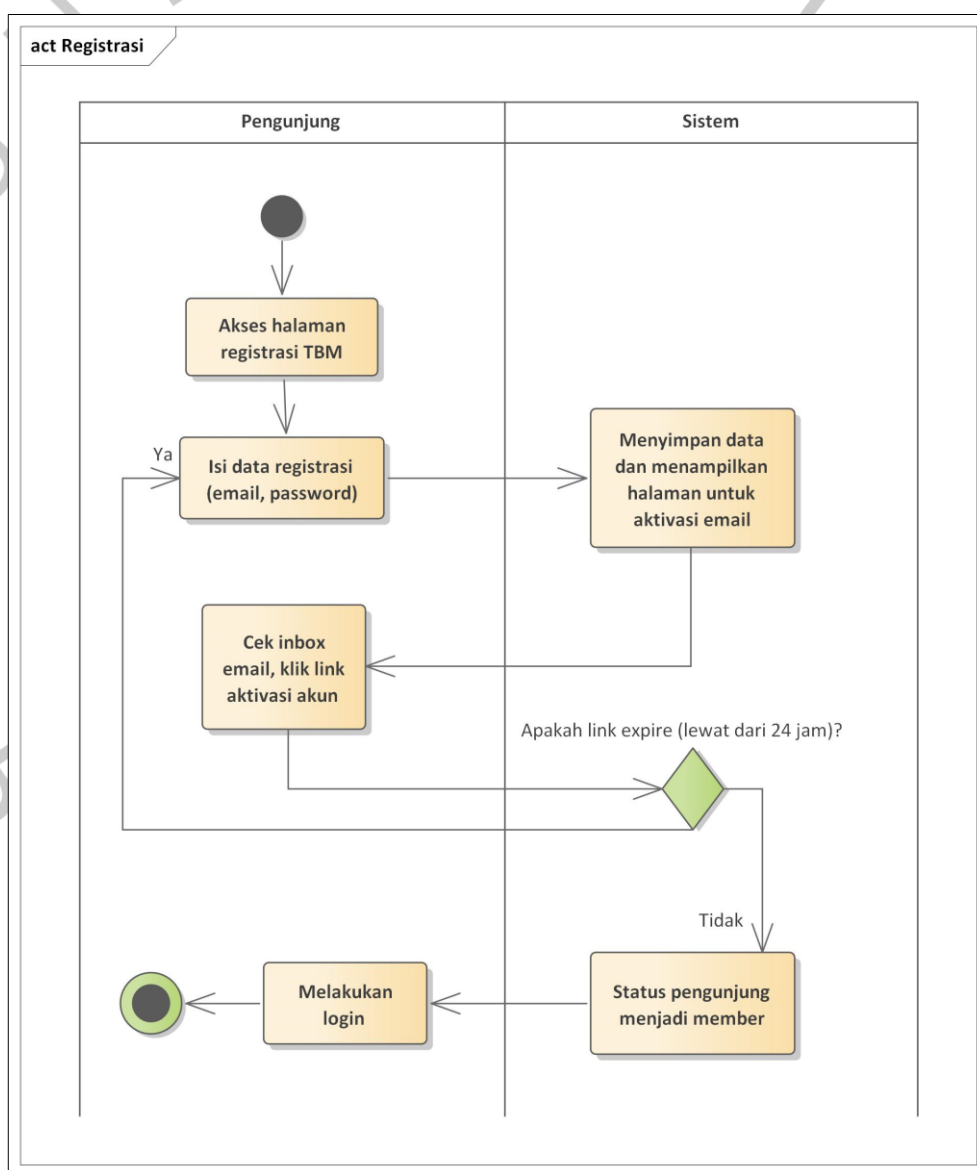
Pada Gambar 4.3 menggambarkan *activity diagram* validasi sistem agar pengguna dapat mengakses sistem utama TBM. Pengguna (pengunjung/pengurus TBM) mengisi *field email* dan *password* pada halaman login. Sistem mengecek *field email* dan *password* apakah sudah sesuai saat diisi oleh pengguna, kemudian mengalihkan pengguna ke halaman utama berdasarkan *field role* masing-masing.



Gambar 4. 3 Activity Diagram Login

2) Registrasi

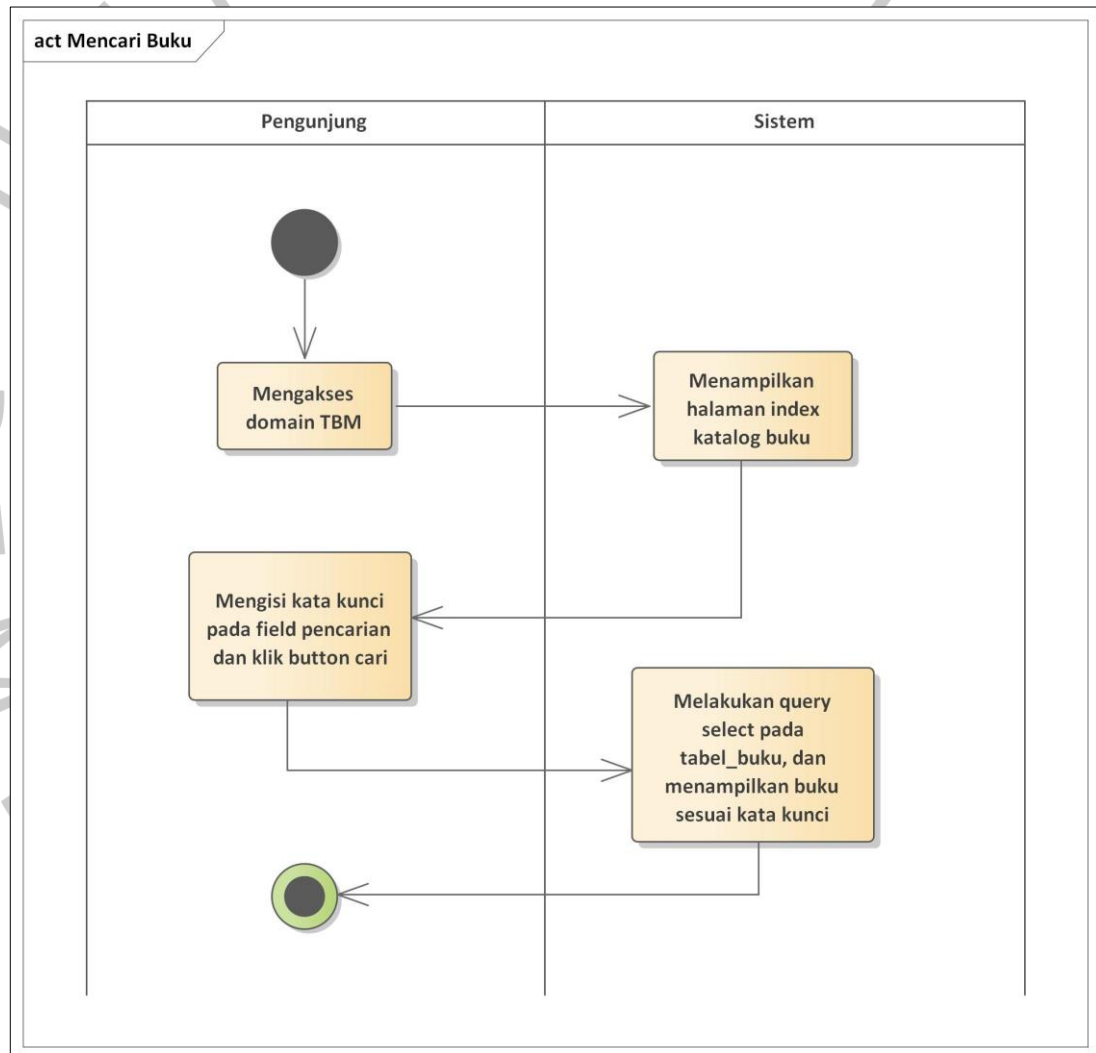
Pada Gambar 4.3 menggambarkan *activity diagram* pendaftaran pengunjung untuk menjadi member TBM. Pengunjung mengisi *field* nama, *email* dan *password form* pada halaman registrasi. Sistem menyimpan *field-field* yang diisi oleh pengunjung dan juga men-*generate* *field link token* dan *timestamp* untuk aktivasi *email*. Pengunjung mengecek *inbox* untuk mengklik *link aktivasi* yang dikirimkan oleh sistem. Jika *timestamp* lewat dari 24 jam maka link akan *expired*.



Gambar 4. 4 Activity Diagram Registrasi

3) Mencari Buku

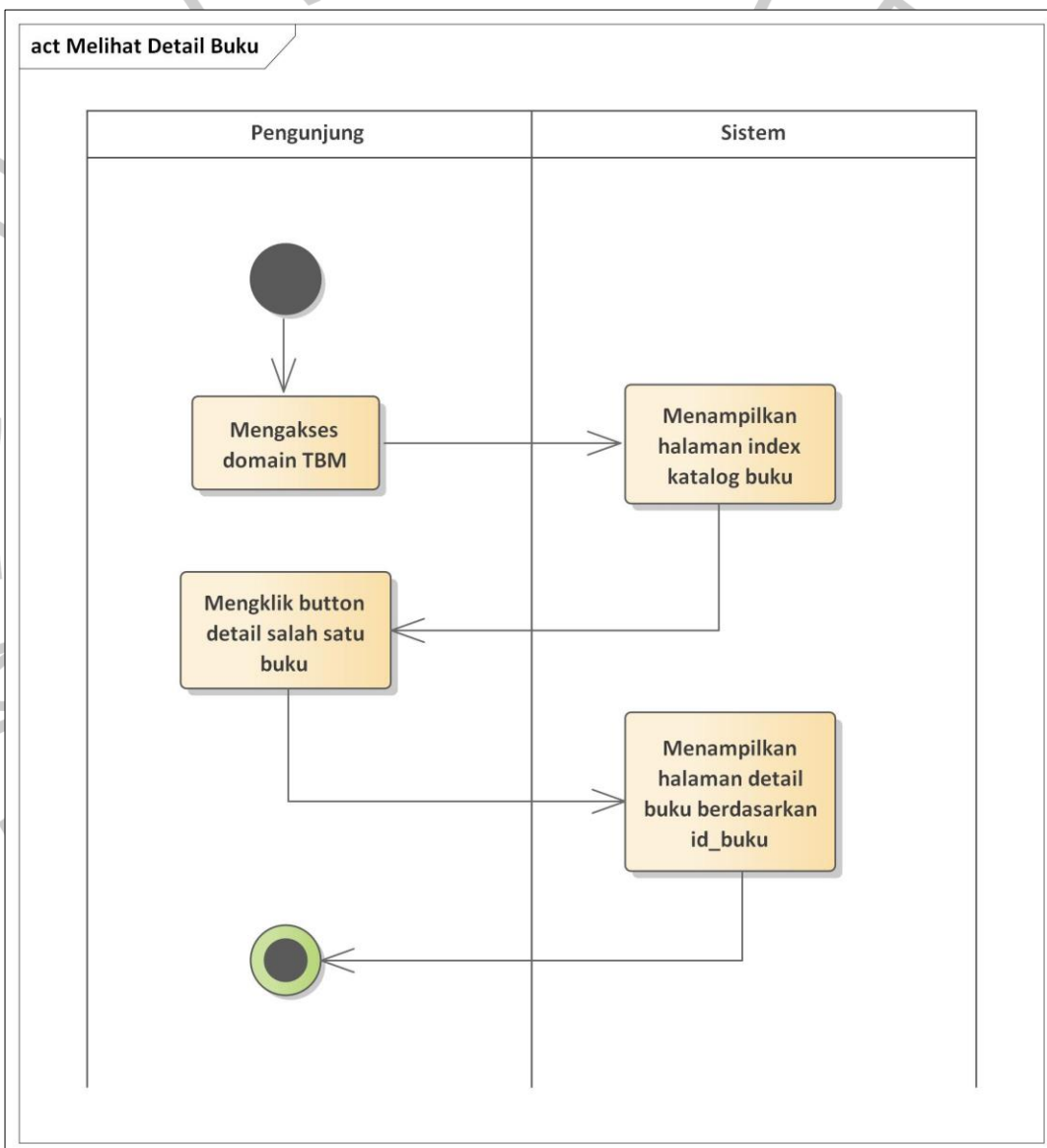
Pada Gambar 4.5. menggambarkan *activity diagram* proses pencarian buku. Pengunjung mengetik url *website* TBM, lalu sistem menampilkan halaman utama katalog buku. Pada halaman tersebut terdapat form pencarian untuk mencari buku berdasarkan kata kunci seperti judul, penulis, dll. Jika sudah input kata kunci selanjutnya pengunjung mengklik *button* mencari. Sistem akan melakukan *query select* untuk melakukan filter terhadap tabel buku dan menampilkan data buku sesuai dengan kata kunci yang dicari.



Gambar 4. 5 Activity Diagram Mencari Buku

4) Melihat Detail Buku

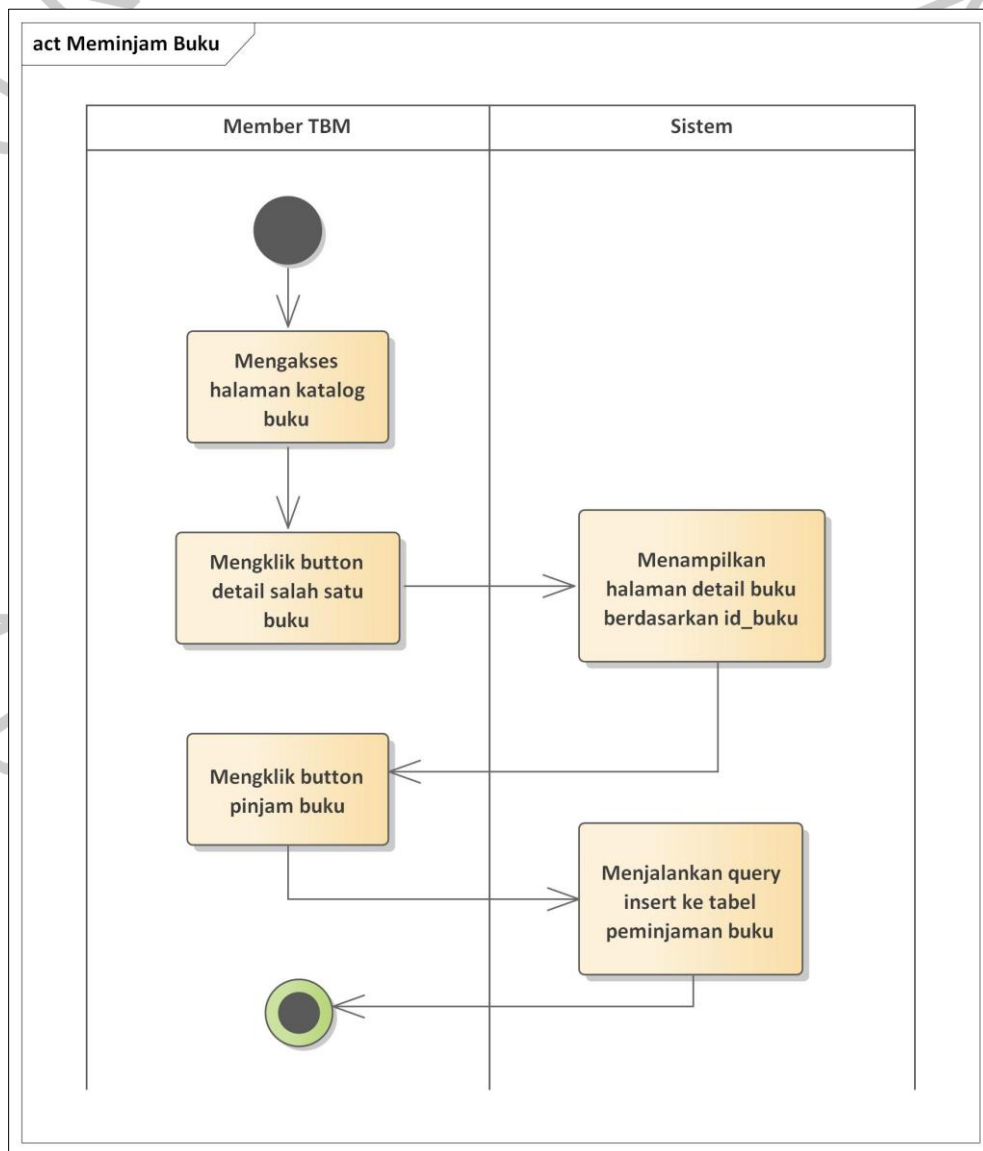
Pada Gambar 4.6. menggambarkan *activity diagram* proses pengunjung untuk melihat detail lengkap informasi buku seperti sinopsis, tahun terbit, dll. Pengunjung mengetik url *website* TBM, lalu sistem menampilkan halaman utama katalog buku. Member mengklik *button* detail buku pada salah satu buku, kemudian sistem menampilkan halaman detail buku berdasarkan *id_buku*.



Gambar 4. 6 Activity Diagram Melihat Detail Buku

5) Meminjam Buku

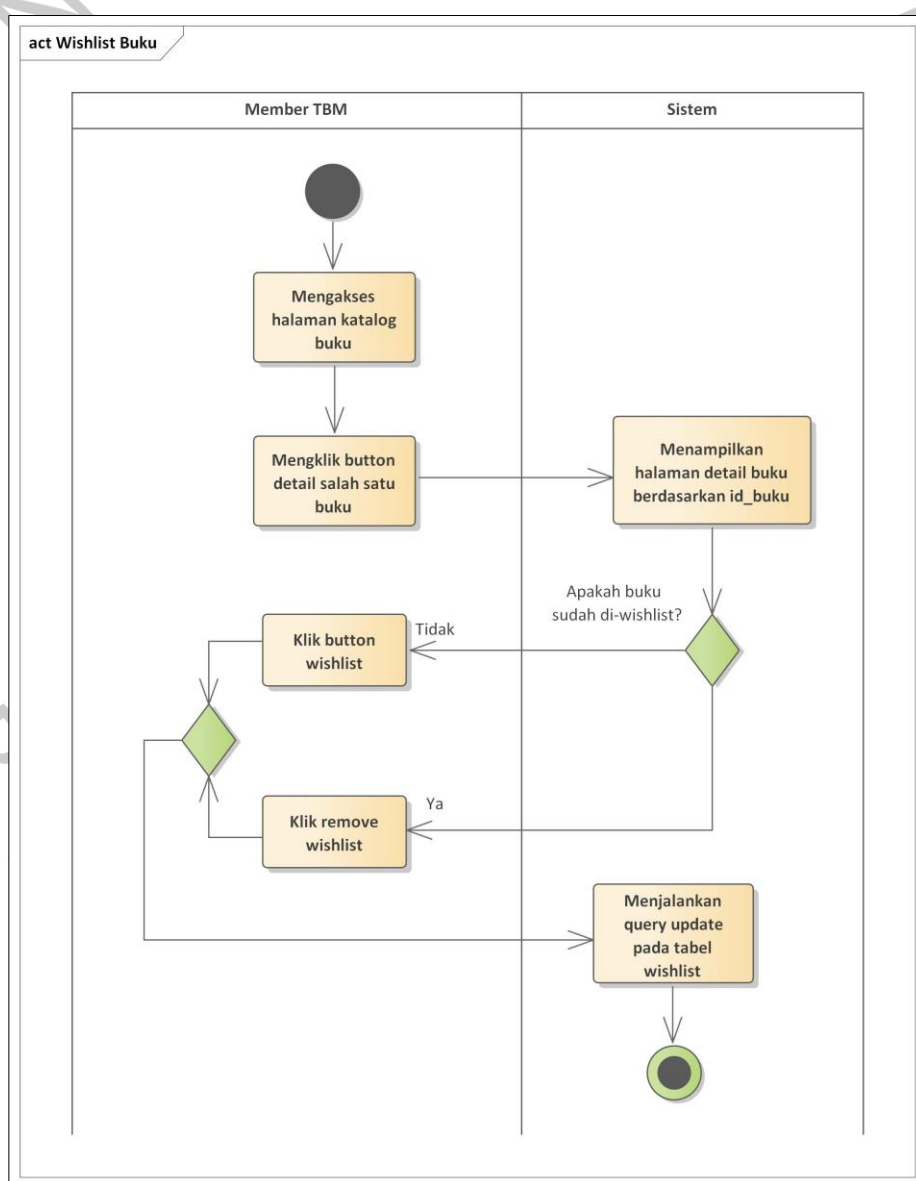
Pada Gambar 4.7. menggambarkan *activity diagram* proses member TBM untuk melakukan *request* peminjaman buku. Pada halaman katalog buku, member mengklik *button* detail buku pada salah satu buku, kemudian sistem menampilkan halaman detail buku berdasarkan *id_buku*. Pada halaman detail buku tersebut terdapat *button* untuk pinjam buku, member mengklik *button* tersebut. Sistem kemudian menjalankan *query* untuk *insert* peminjaman buku. Pengunjung melaporkan permohonan peminjaman buku ke pengurus untuk disetujui.



Gambar 4. 7 Activity Diagram Meminjam Buku

6) Wishlist Buku

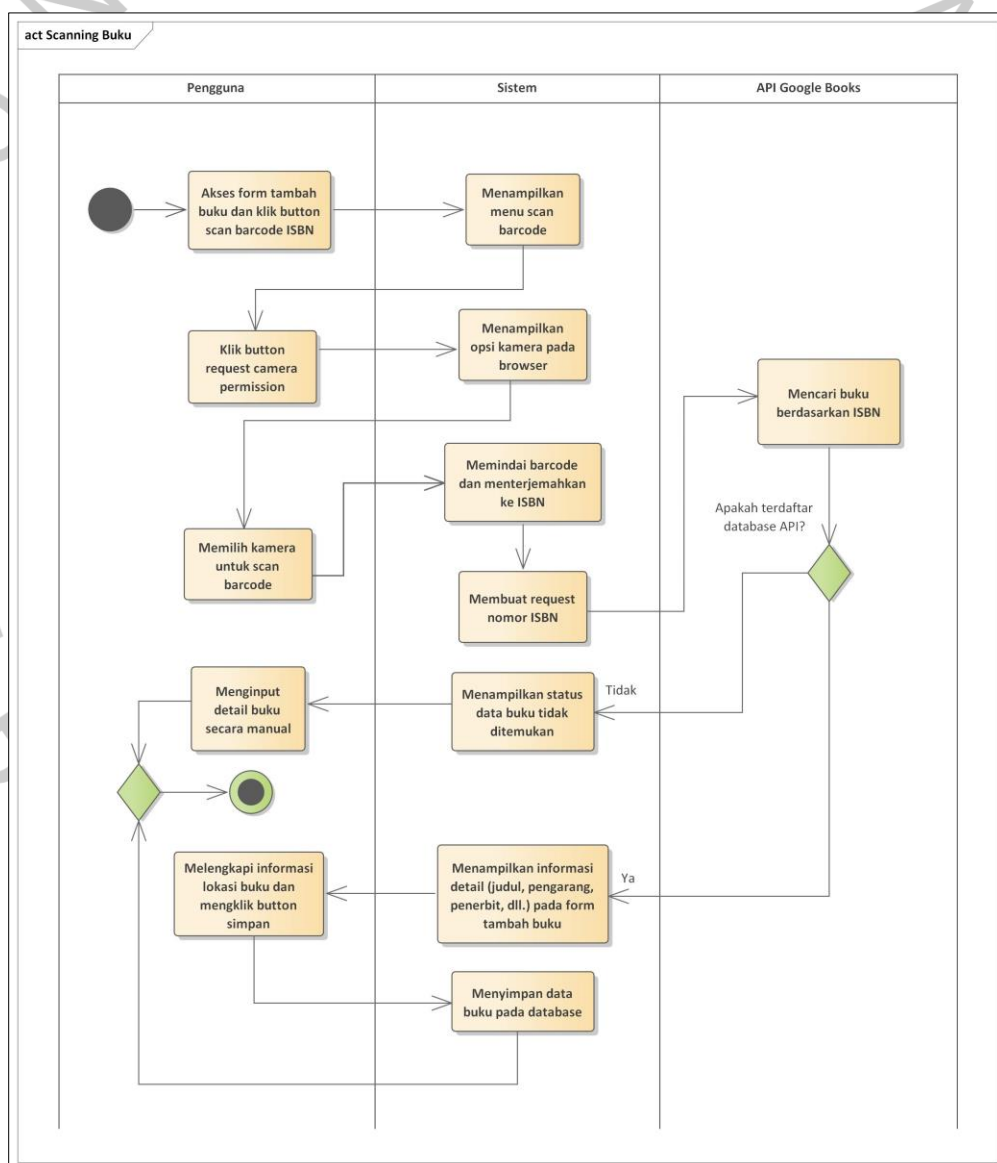
Pada Gambar 4.8. menggambarkan *activity diagram* sirkulasi buku yaitu peminjaman dan pengembalian. Member mengklik *button* pinjam buku pada halaman detail, kemudian sistem menyimpan dan memberi pesan kepada pengurus pada halaman administrasi peminjaman. Pengurus mengatur tanggal pengembalian buku melalui *date picker* dan mengklik *button* konfirmasi peminjaman. Buku yang sudah dipinjam tidak akan ditampilkan ke dalam halaman katalog buku. Jika buku sudah dikembalikan ke TBM maka pengurus mengklik *button* pengembalian.



Gambar 4. 8 Activity Diagram Wishlist Buku

7) Scan ISBN Buku

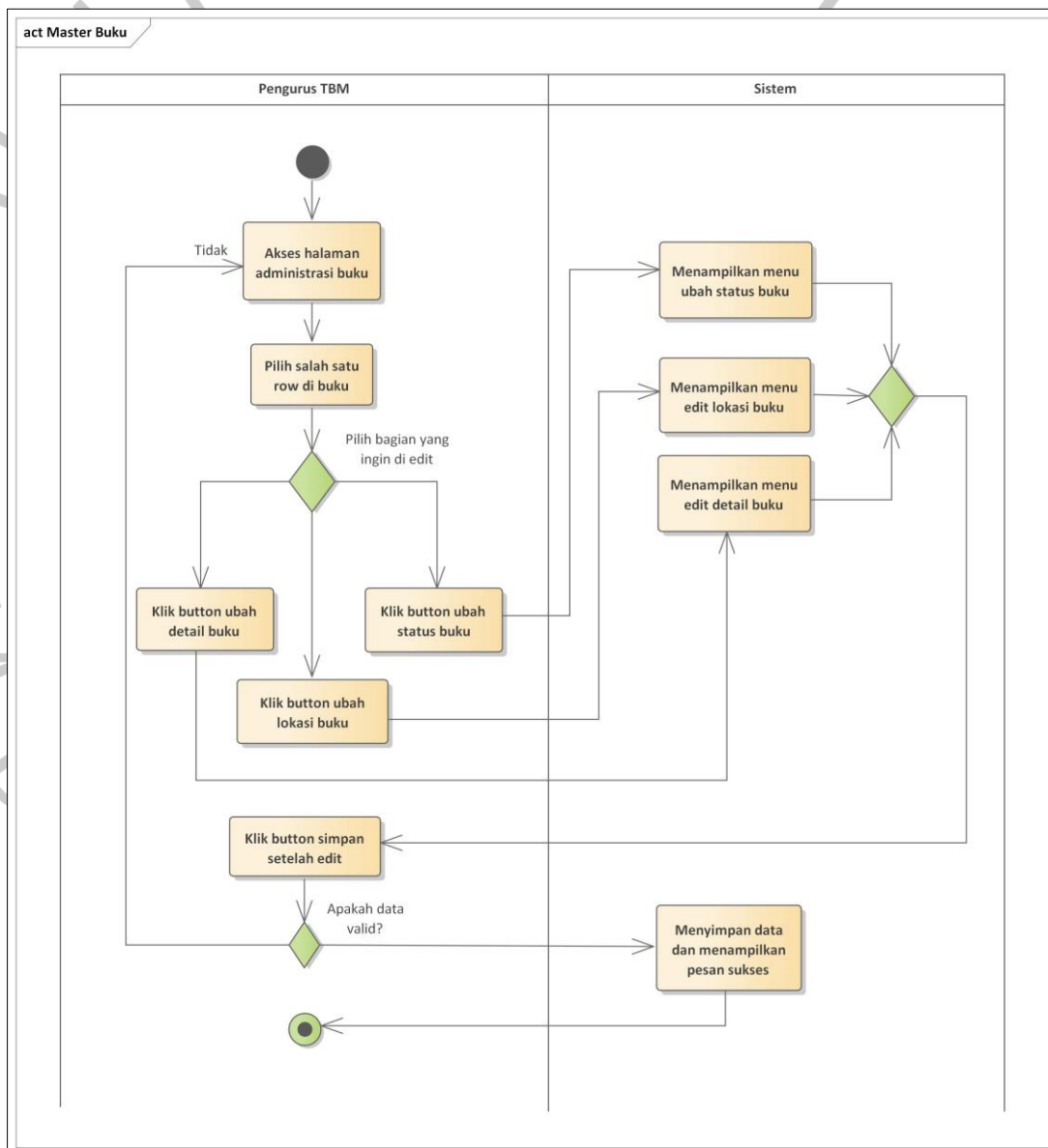
Pada Gambar 4.9 menggambarkan *activity diagram* dari fitur sistem TBM di bagian administrasi buku yang dapat menambahkan data buku melalui pemindaian *Barcode* untuk mendapatkan data dari *Google Books API*. Kamera menterjemahkan kode garis ke ISBN. Sistem menyimpan variabel ISBN dan menambahkannya ke url link API *Google Books* untuk mengecek ada atau tidaknya buku. Jika ada sistem akan mengekstrak data JSON ke *field* informasi buku mengenai judul, sampul, pengarang, penerbit dll. Jika tidak ada, maka pengurus masih bisa melakukan mengisi data buku secara manual.



Gambar 4. 9 Activity Diagram Scan Buku

8) Administrasi Buku

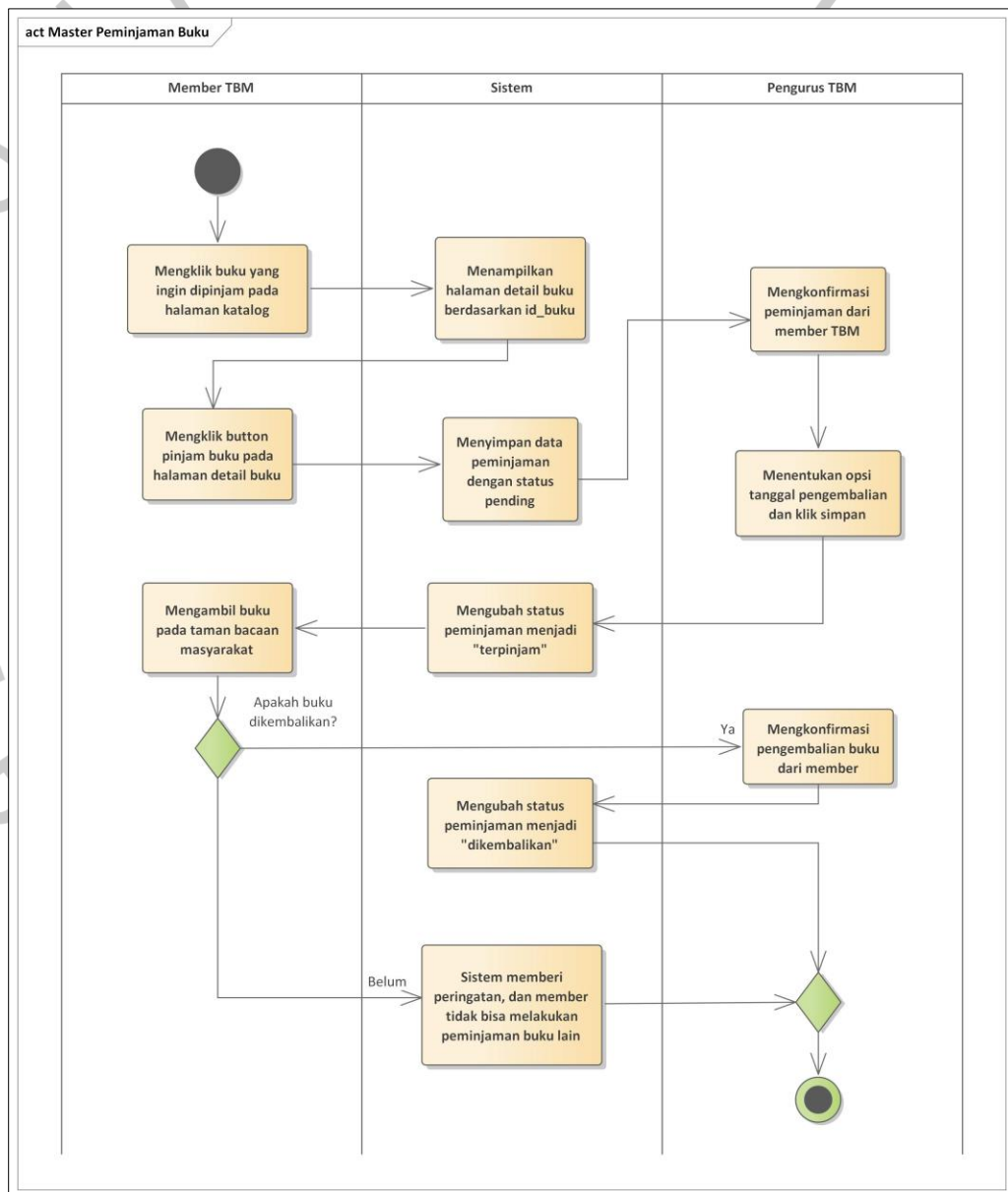
Pada Gambar 4.10 menggambarkan *activity diagram* dari sistem TBM di bagian administrasi buku yang dapat mengubah status, lokasi dan detail buku. Halaman administrasi buku menampilkan tabel daftar buku yang telah ditambahkan baik dari proses *scan* ISBN atau manual. Kolom aksi pada tabel terdapat button yang menampilkan *dialog box* ubah detail buku seperti judul, penulis, dll., ubah lokasi seperti lemari, rak, dll. serta status buku yang berisi status hilang/rusak atau *publish*.



Gambar 4. 10 Activity Diagram Administrasi Buku

9) Administrasi Peminjaman

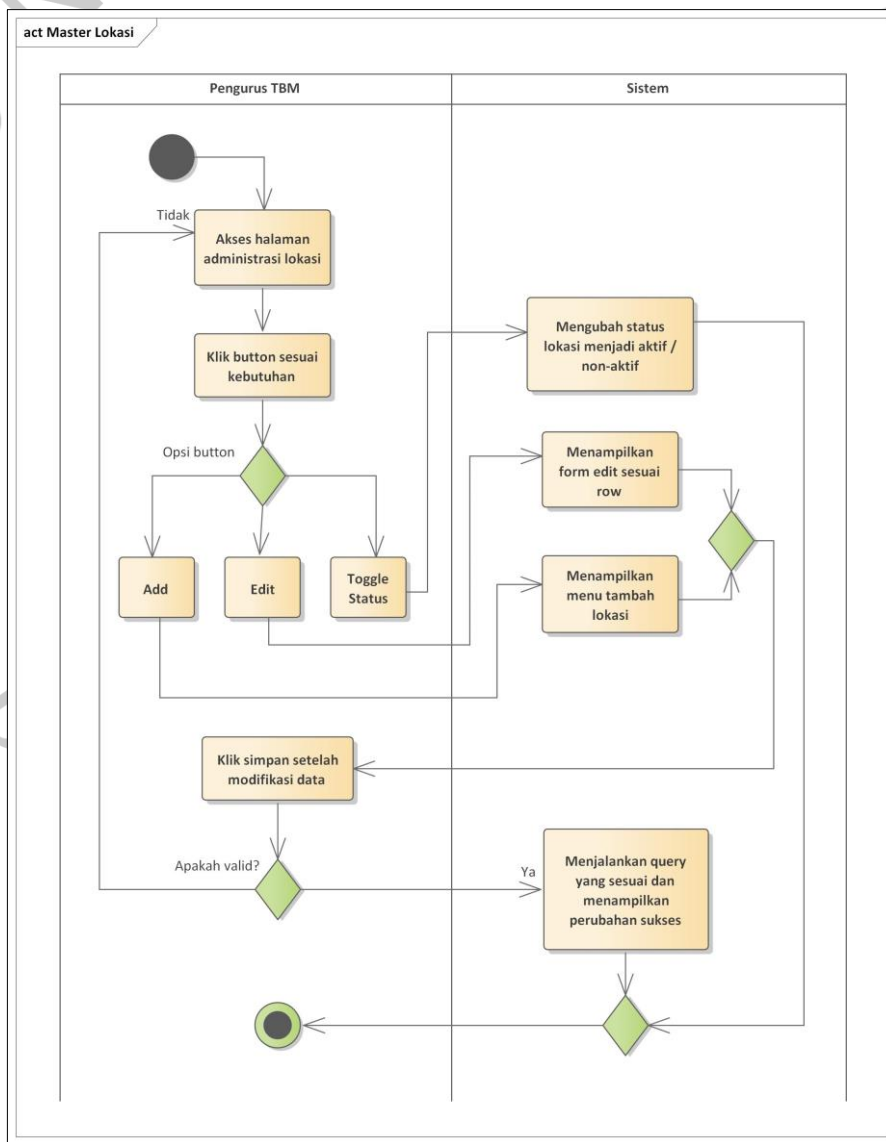
Pada Gambar 4.11 menggambarkan *activity diagram* untuk administrasi peminjaman dan pengembalian buku. Member mengklik *button* pinjam buku pada halaman detail, kemudian sistem menyimpan dan memberi pesan kepada pengurus pada halaman administrasi peminjaman. Pengurus mengatur tanggal pengembalian buku melalui date picker dan mengklik *button* konfirmasi peminjaman. Jika buku sudah dikembalikan ke TBM maka pengurus mengklik *button* pengembalian.



Gambar 4. 11 Activity Diagram Administrasi Peminjaman

10) Administrasi Lokasi

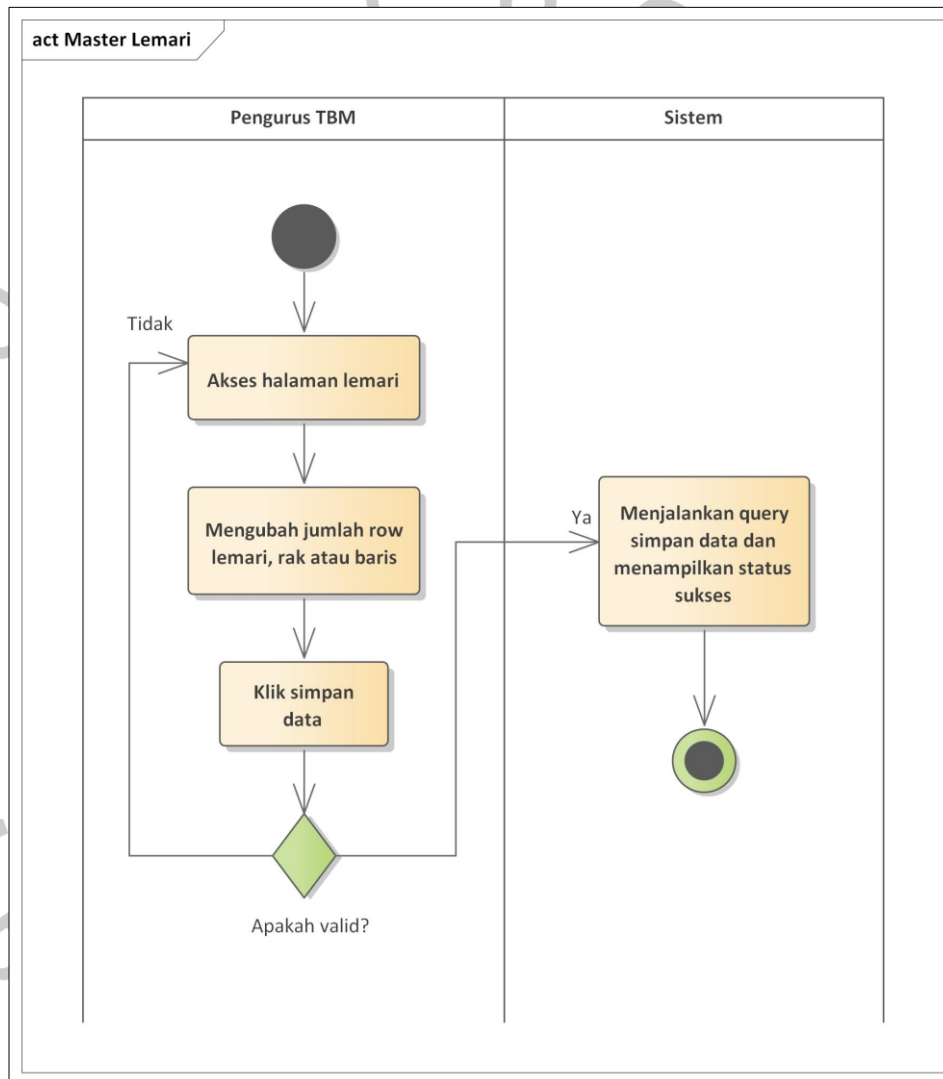
Pada Gambar 4.12 menggambarkan *activity diagram* dari sistem TBM di bagian administrasi buku yang dapat mengelola lokasi sebuah buku. Pada halaman tersebut terbagi menjadi tiga *button* yang menampilkan kotak dialog yaitu tambah data lokasi, ubah data lokasi, *toggle* aktivasi lokasi. Pengurus mengisi *field* kode lokasi, nama lokasi dan keterangan pada kotak dialog tambah data. *Field* yang sama juga digunakan pada kotak dialog ubah lokasi jika ingin mengubah data lokasi. *Toggle* aktivasi dapat menonaktifkan atau mengaktifkan kembali lokasi.



Gambar 4. 12 Activity Diagram Administrasi Lokasi

11) Administrasi Lemari

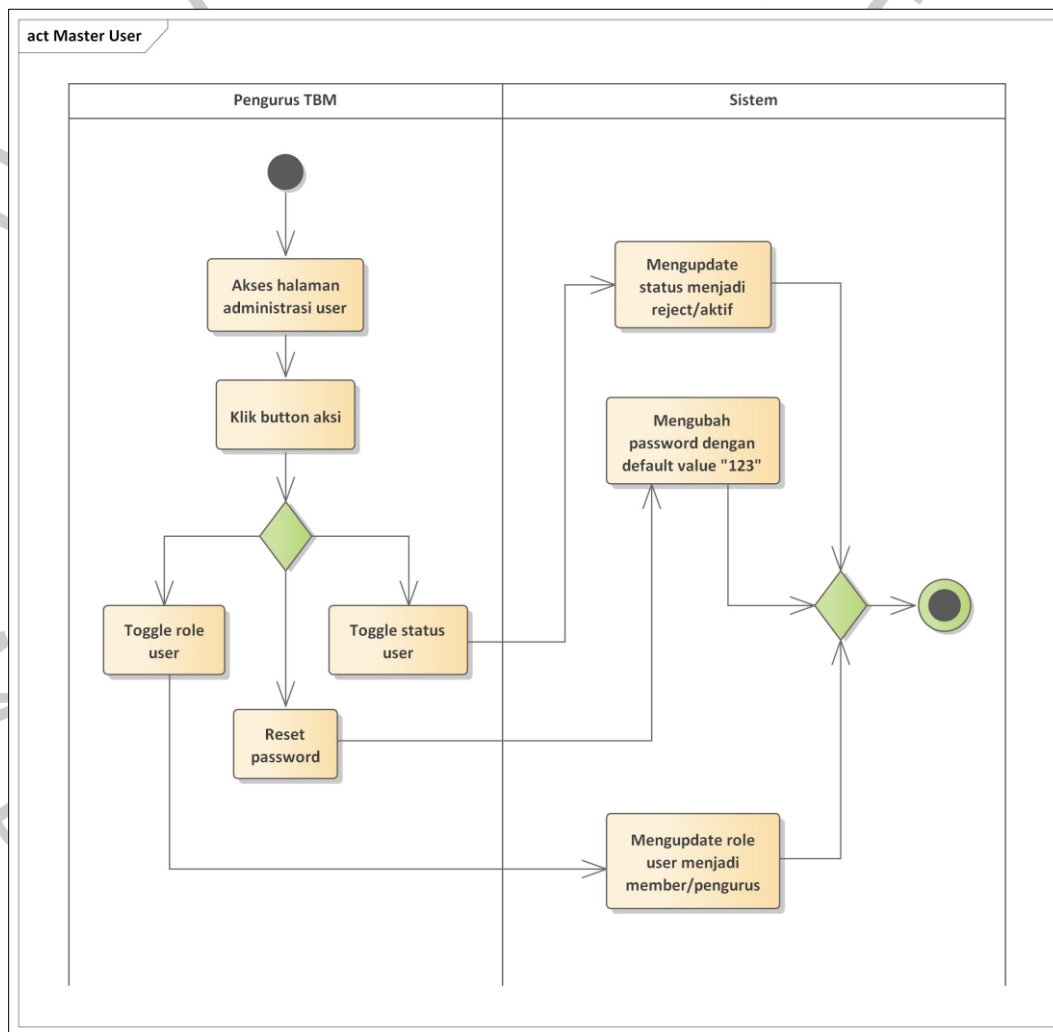
Pada Gambar 4.13 menggambarkan *activity diagram* dari sistem TBM di bagian administrasi lemari untuk mengubah jumlah maksimal lemari, rak, atau baris di suatu TBM jika suatu saat kedepannya ada penambahan pada objek tersebut.



Gambar 4. 13 Activity Diagram Administrasi Lemari

12) Administrasi User

Pada Gambar 4.14 menggambarkan *activity diagram* dari sistem TBM di bagian administrasi user untuk mengelola data pengguna tersebut. Pada halaman tersebut terbagi menjadi tiga *button* yaitu *toggle role* dapat mengubah role member menjadi pengurus, *reset password* mengubah *password default* menjadi “123” dan *toggle* aktivasi dapat menonaktifkan atau mengaktifkan kembali pengguna.



Gambar 4. 14 Activity Diagram Administrasi User

4.2.4 Sequence Diagram

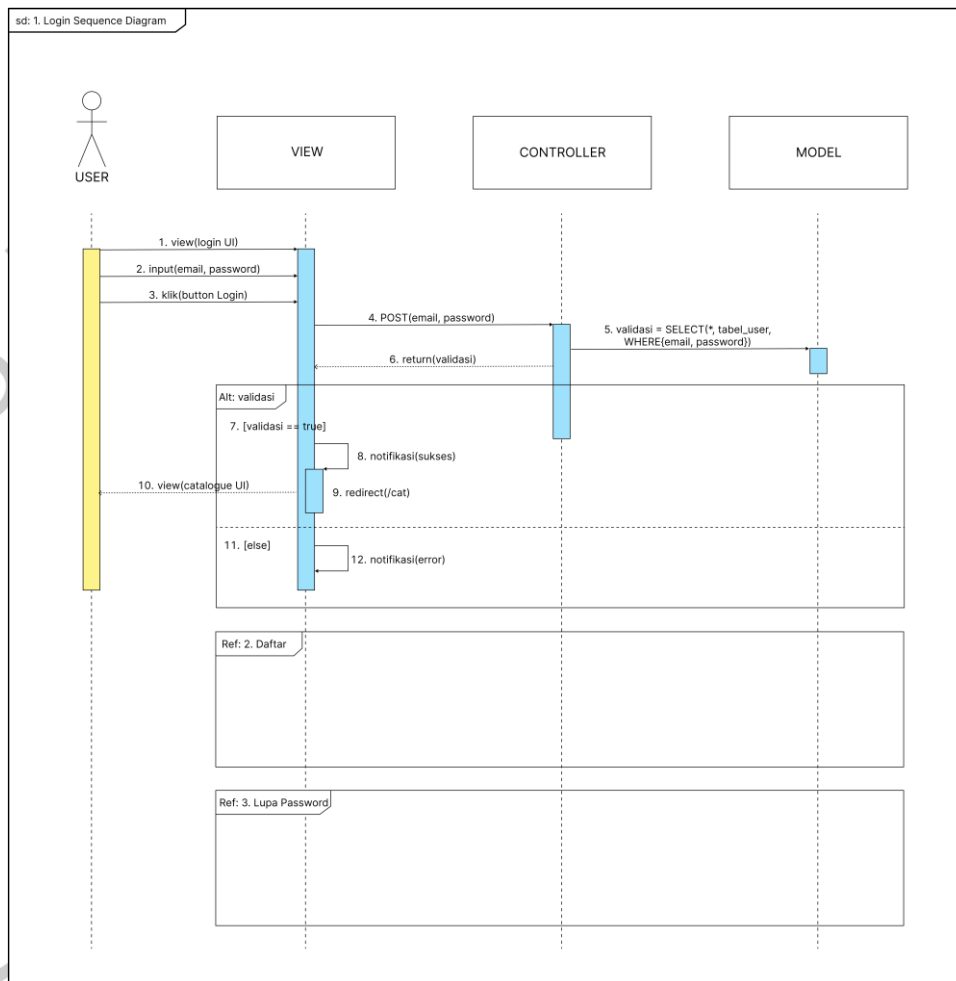
Pada sistem pengelolaan taman baca *sequence diagram* berfungsi untuk memperlihatkan interaksi antar objek atau komponen pada sistem secara berurutan dalam skenario tertentu. Elemen utama yang terdapat pada *sequence diagram* yaitu *actor*, *lifeline* dan *message*. *Actor* menunjukkan entitas eksternal di luar sistem seperti member atau pengurus taman baca. *Lifeline* menunjukkan objek atau kelas internal pada dalam sistem seperti buku, peminjaman dll. *Message* menunjukkan komunikasi antar objek yang digambarkan dengan arah panah.

Sequence diagram pada sistem taman baca ini menggunakan struktur *Model*, *View* dan *Controller* (MVC) dari *Framework CodeIgniter*. MVC menjadi objek utama pada *sequence diagram* sistem taman baca. *Model* berfungsi untuk mengakses dan mengelola data pada basis data. *View* berfungsi untuk menampilkan data dan informasi pada pengguna. *Controller* berfungsi untuk mengendalikan *request* mengambil, menyimpan atau mengubah data dari *Model* dan menampilkan *response*-nya ke *View*. Berikut merupakan *sequence diagram* sistem pengelolaan taman bacaan masyarakat pada Gambar 4.15. hingga Gambar 4.38.

1) Login

Pada Gambar 4.15 menggambarkan *activity diagram* validasi sistem agar pengguna dapat mengakses sistem utama TBM. Pengguna mengakses halaman *View* Login, lalu memasukan *field email* dan *password*, serta mengklik *button* login. *View* Login mengirim request *field email* dan *password* dengan metode POST ke *Controller*. *Controller* melakukan *query* validasi (*email*, *password*, dan status) dari data tabel_user yang diambil dari *Model*. *Controller* menerima status *return* validasi. Notifikasi sukses akan ditampilkan ke *View* Login, dan *session* akan aktif. Jika validasi sukses *View* Login akan

redirect ke halaman Index atau utama sesuai dengan role. Jika validasi gagal akan menampilkan notifikasi gagal, *email* atau *password* tidak sesuai.



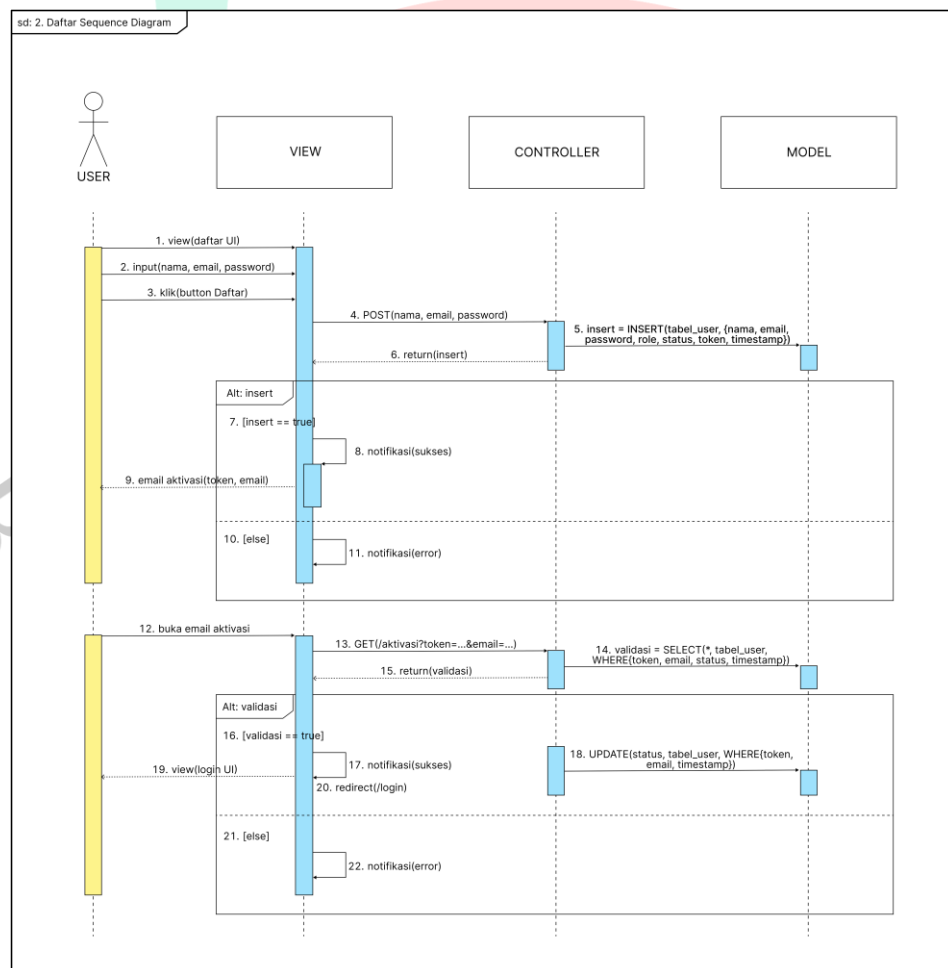
Gambar 4. 15 Sequence Diagram Login

2) Registrasi

Pada Gambar 4.16 menggambarkan *sequence diagram* registrasi dan aktivasi agar pengunjung menjadi member TBM. Pengguna mengakses halaman *View* Pendaftaran, lalu memasukan *field nama, email* dan *password*, serta mengklik *button* daftar. *View* Pendaftaran mengirim *request field* nama, *email* dan *password* dengan metode *POST* ke *Controller*. *Controller* melakukan *query insert (email, nama, password, role, status, token* dan *timestamp)* dari

data tabel_user yang diambil dari Model. Jika query sudah masuk, View Pendaftaran akan menampilkan notifikasi sukses dan harap cek inbox email.

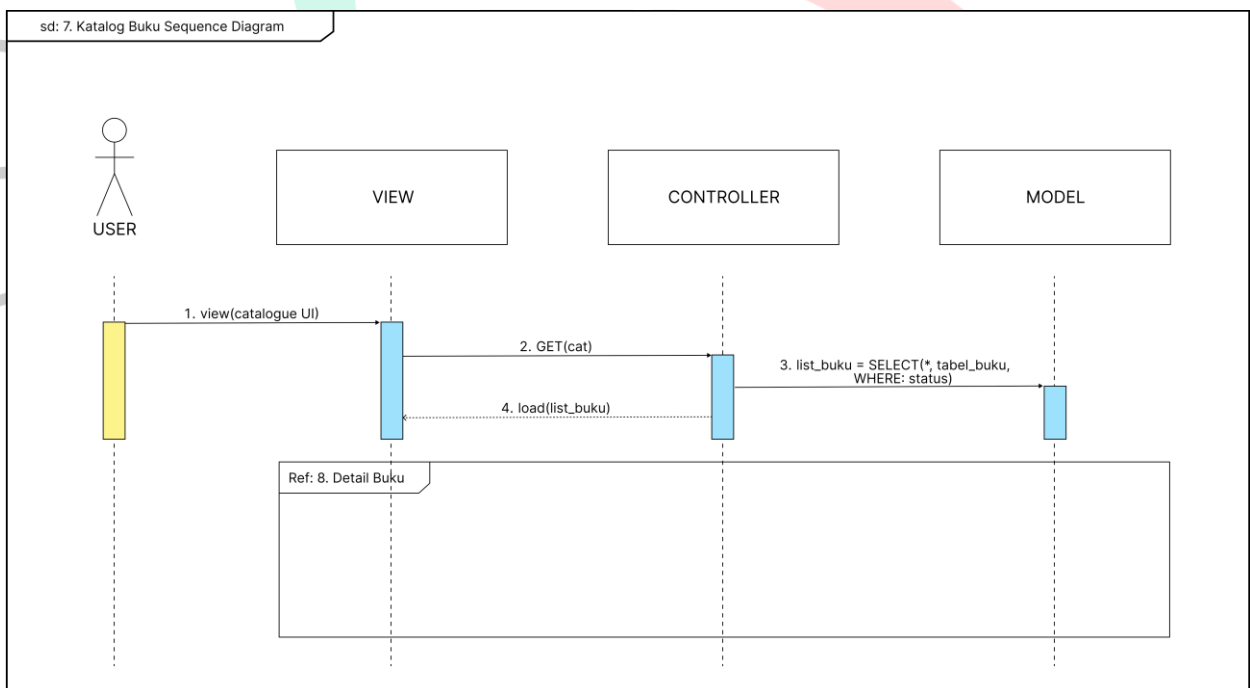
Pengunjung mengklik link pada inbox email untuk mengirim request email dan token dengan metode GET ke Controller. Token merupakan kode unik yang ter-generate otomatis saat query insert untuk dicocokkan bersamaan dengan email. Jika token dan email sesuai maka pengguna akan aktif. Timestamp juga digunakan juga untuk mengaktifasi pengguna dalam kurun waktu 24 jam, jika link tidak diaktivasi atau sudah lewat dari 24 jam maka link kadaluarsa. Jika validasi sukses link inbox dari email akan redirect ke halaman Login.



Gambar 4. 16 Sequence Diagram Registrasi

3) Mencari Buku

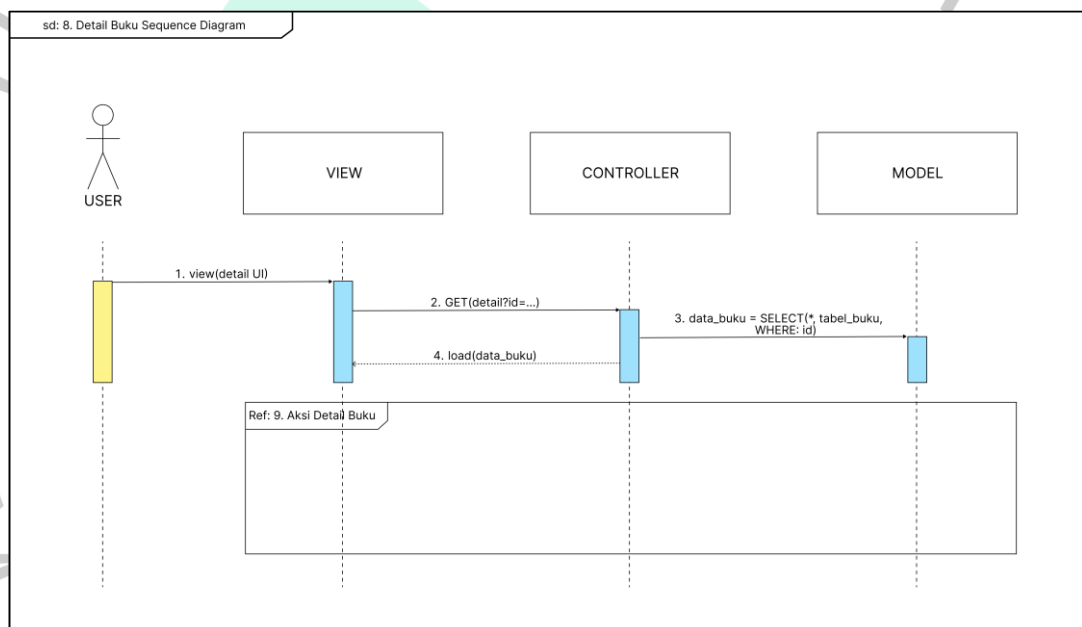
Pada Gambar 4.17 menggambarkan *sequence diagram* halaman index katalog buku yang diakses oleh Member. *View* Daftar Katalog Buku mengirim *request* metode GET halaman katalog. *Controller* mengirim *query select* from tabel buku dari *Model*. Selanjutnya *query* menampilkan data buku (cover, judul, pengarang, dll.) pada halaman *View* Katalog. Jika Member ingin mencari buku maka Member akan mengisi form pencarian, lalu *request* metode POST dijalankan sesuai dengan 'kata_kunci' pencarian ke *Controller*. *Controller* mengirim *query select by* 'kata_kunci' from tabel_buku dari *Model*. Selanjutnya *query* menampilkan data buku yang telah ter-filter kata kunci pada halaman *View* Daftar Katalog Buku.



Gambar 4. 17 Sequence Diagram Mencari Buku

4) Melihat Detail Buku

Pada Gambar 4.18 menggambarkan *sequence diagram* halaman detail buku harus diakses oleh Member jika ingin melakukan peminjaman atau menyimpan wishlist pada buku. Member mengakses *View* Detail Buku dari mengklik salah satu buku halaman katalog. *Controller* menerima request dengan metode GET untuk field *id_buku*. *Model* menerima *query* dari *Controller* untuk *select by id* dari tabel buku. Selanjutnya result menampilkan halaman detail buku berdasarkan *id_buku*.



Gambar 4. 18 Sequence Diagram Detail Buku

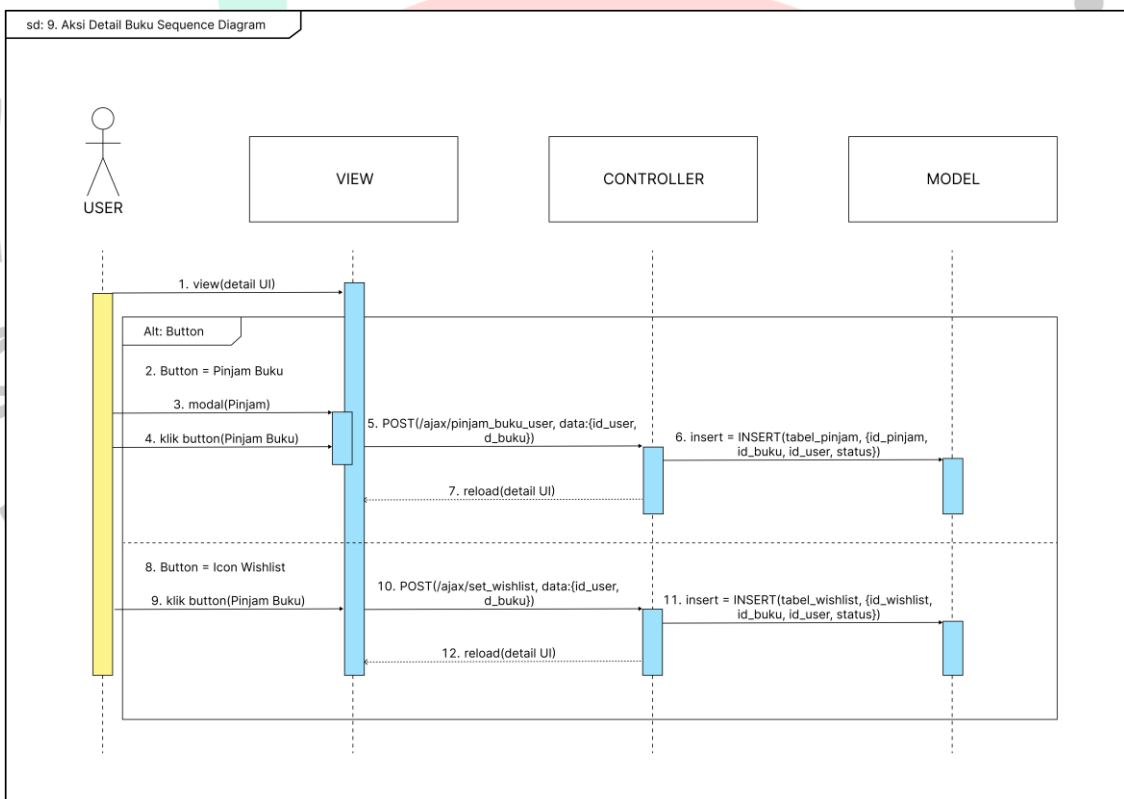
5) Member Pinjam Buku

Pada Gambar 4.19 jika Member ingin melakukan peminjaman maka harus berada pada *View* Detail Buku terlebih dahulu, lalu mengklik button pinjam buku. *Controller* akan menerima request dengan metode POST yang berisi *id_buku*, *id_user*, *tanggal_pinjam*. *Model* akan menerima query dari *Controller* yang berisi perintah *insert id_pinjam*, *id_buku*, *id_user*, *status* dll. ke tabel *pinjam*. Selanjutnya result akan ditampilkan di halaman daftar

pinjam dan administrasi peminjaman. Member harus menghubungi Pengurus untuk mengkonfirmasi peminjaman dan untuk menentukan berapa lama pengembalian.

6) Member Wishlist Buku

Pada Gambar 4.19 jika Member ingin melakukan wishlist buku maka harus berada pada *View* Detail Buku terlebih dahulu, lalu mengklik button wishlist buku. *Controller* akan menerima request dengan metode POST yang berisi id_buku, id_user. *Model* akan menerima *query* dari *Controller* yang berisi perintah insert id_wishlist, id_buku, id_user, status dll. ke tabel_wishlist. Selanjutnya result akan ditampilkan di halaman daftar wishlist.

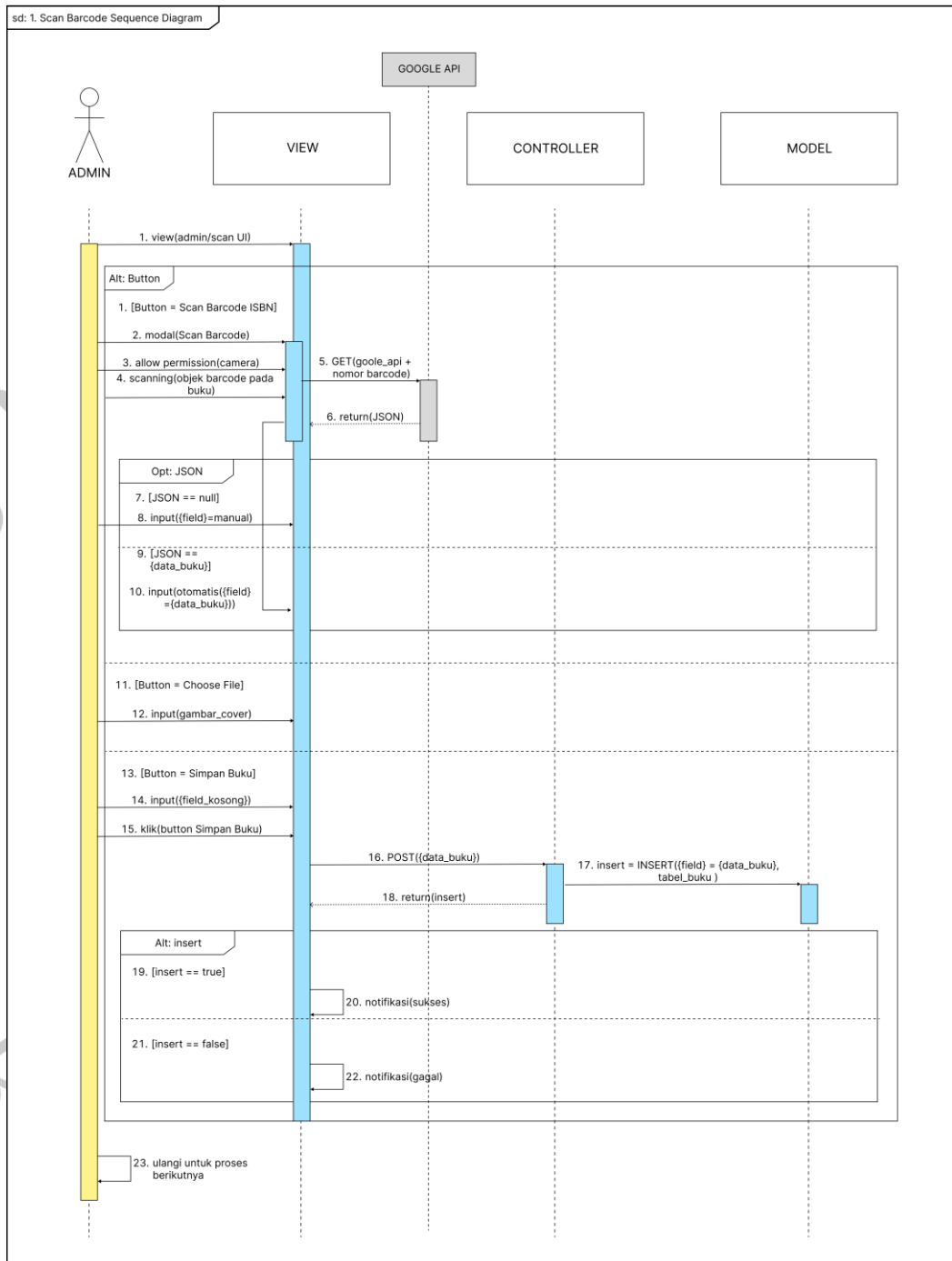


Gambar 4. 19 Sequence Diagram Aksi Detail Buku

7) Administrasi Buku – Input Buku Metode Scan

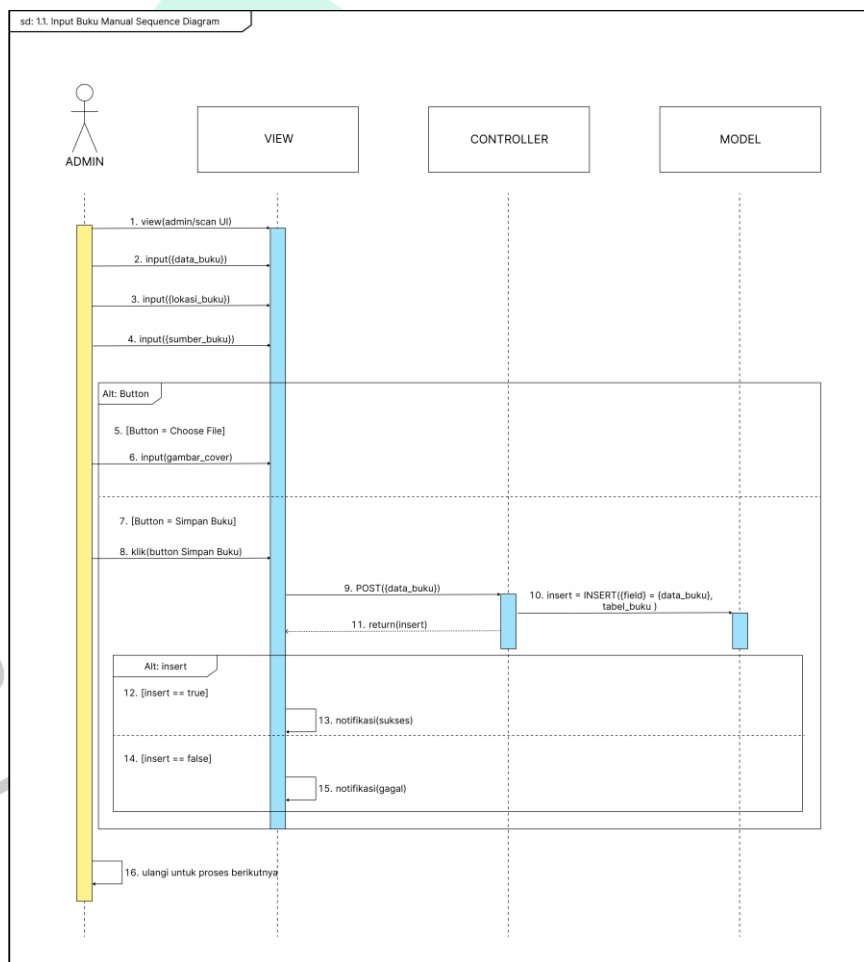
Pada Gambar 4.20 menggambarkan *sequence diagram* dialog box pemindaian barcode ISBN pada buku yang terhubung ke *Google Books* API untuk memudahkan proses input data buku. Pengurus berada di Halaman *View Scan* dan mengklik *button scan* untuk *trigger dialog box barcode scanner*. Pengurus mengarahkan kamera kepada *barcode* di belakang buku ke dalam kotak kamera agar membaca ISBN. Ketika *View Scan* berhasil membaca *barcode*, maka halaman tersebut akan menjalankan mekanisme *request* melalui metode GET yaitu dengan mengisi variabel *api_link* = "https://www.googleapis.com/books/v1/volumes?q=isbn:" sesuai dengan *field* ISBN yang terisi saat berhasil *ter-scan*.

Output yang dihasilkan dari *request* API tersebut adalah JSON data buku dengan ISBN yang bersangkutan dan terisi secara otomatis pada *field* judul, pengarang, penerbit, tahun, cover buku (jika ada), resume dll. pada Halaman *View Scan*. Selain itu pengurus memilih *dropdown* pada lokasi buku dan memilih sumber donasi. Ketika data sudah siap disimpan, pengurus mengklik *button* simpan buku pada halaman tersebut untuk mengirim *request* metode POST data buku ke *Controller*. *Model* kemudian menerima *query insert* data buku dari *Controller* ke *tabel_buku*.



Gambar 4. 20 Sequence Diagram Input Buku Metode Scan

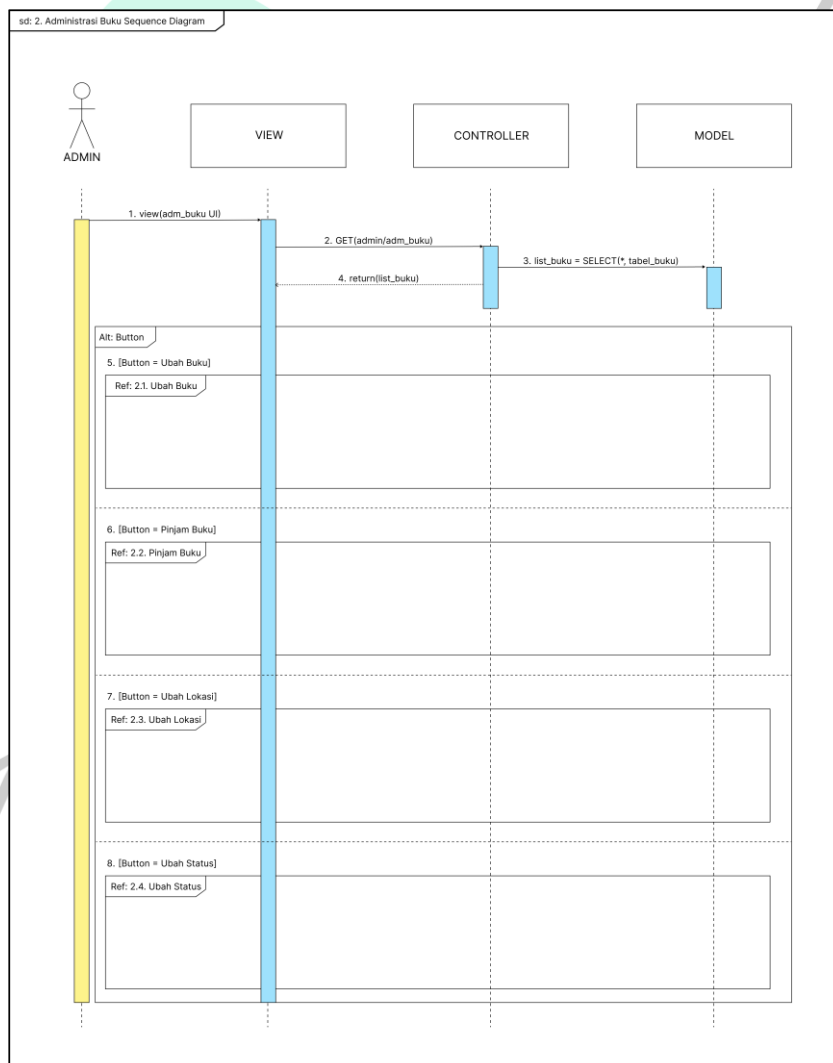
Pada Gambar 4.21 jika buku yang telah ter-*scan* tidak terdaftar di *Google Books API*. Pengurus tetap pada halaman *View Scan Buku* dan langsung mengisi *field* judul, pengarang, penerbit, tahun, resume dll. Selain itu pengurus memilih *dropdown* pada lokasi buku dan memilih sumber donasi. Ketika data sudah siap disimpan, pengurus mengklik button simpan buku pada halaman tersebut untuk mengirim *request* metode *POST* data buku ke *Controller*. *Model* kemudian menerima *query insert* data buku dari *Controller* ke tabel buku.



Gambar 4. 21 Sequence Diagram Input Buku Metode Manual

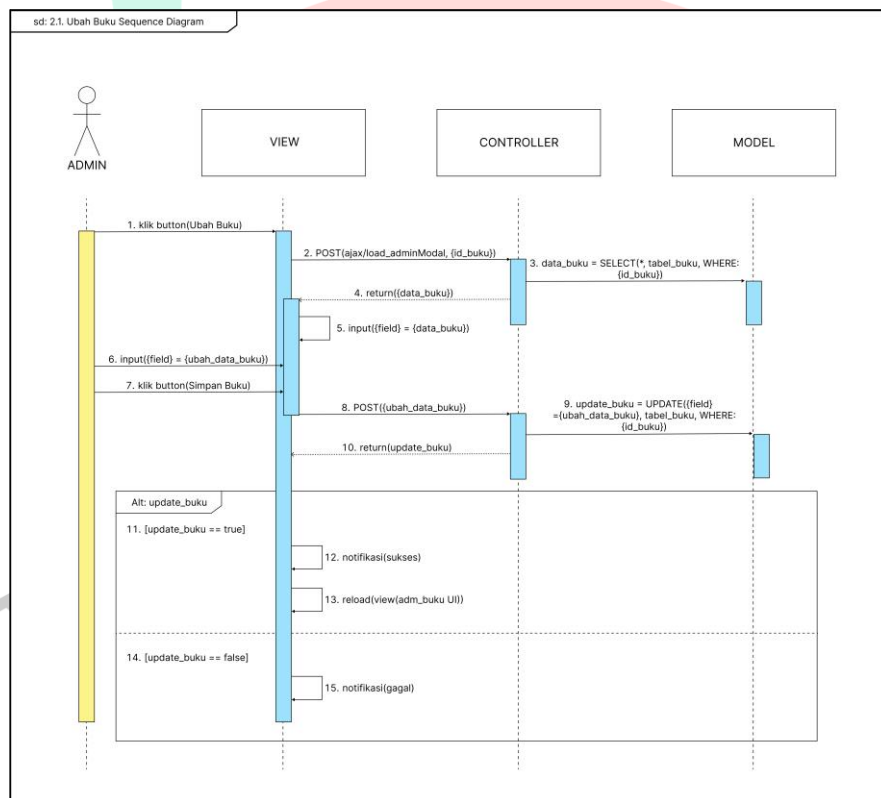
8) Administrasi Buku

Pada Gambar 4.22 s.d. 4.26 menggambarkan *sequence diagram* halaman utama administrasi buku yang berisi tabel dengan kolom gambar buku, informasi buku (judul, pengarang, penerbit, dll.), status buku dan aksi buku. Aksi buku memiliki empat button untuk mengelola data buku yaitu button pertama untuk mengubah informasi buku, button kedua untuk melakukan peminjaman buku dari sisi pengurus, button ketiga mengubah lokasi penyimpanan buku dan button keempat untuk mengubah status buku.



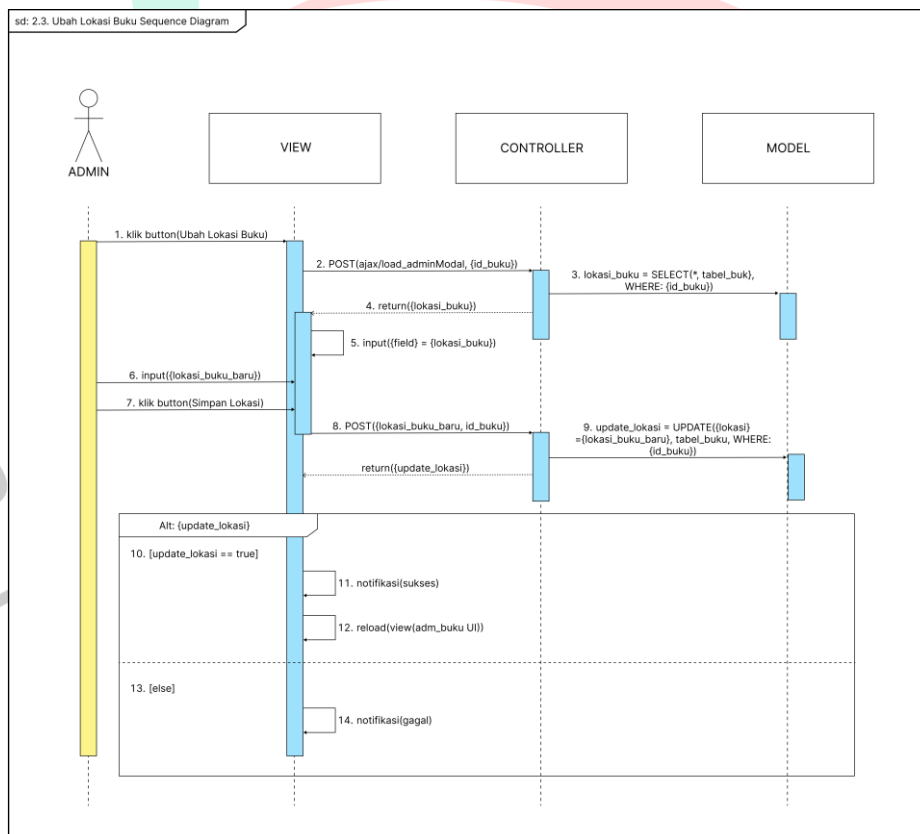
Gambar 4. 22 Sequence Diagram Administrasi Buku

Pada Gambar 4.23 menggambarkan *sequence diagram* dialog box untuk mengubah detail buku seperti judul, pengarang, tahun, cover buku dll. Pengurus mengakses halaman *View* Administrasi Buku setelah login. Pada halaman tersebut Pengurus memilih salah satu *row* pada buku yang detailnya ingin diubah, lalu mengklik button pertama pada kolom aksi. *Button* pertama menampilkan kotak dialog untuk mengubah *field* seperti judul, pengarang, tahun, cover buku dll. Pengurus mengubah *field* yang ingin diubah, saat data sudah siap disimpan, pengurus mengklik button simpan. Button simpan berisi perintah untuk mengirim *request* metode POST data buku ke *Controller*. *Model* kemudian menerima *query update* data buku dari *Controller* ke *tabel_buku*.



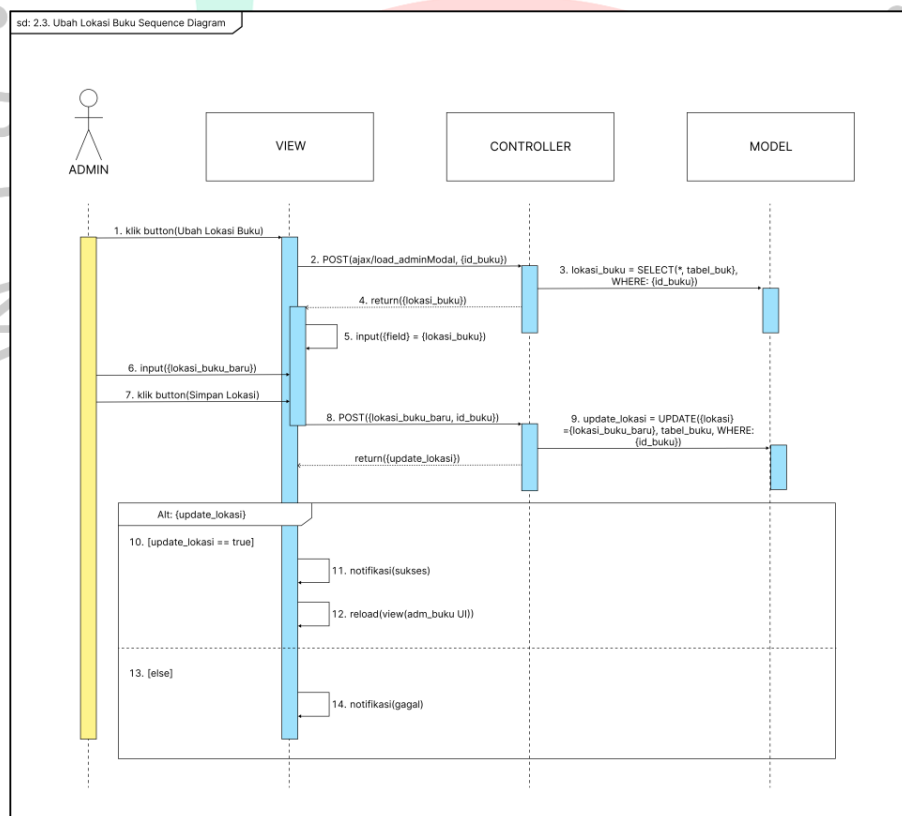
Gambar 4. 23 Sequence Diagram Ubah Detail Buku

Pada Gambar 4.24 menggambarkan *sequence diagram* dialog box untuk mengubah posisi/tempat buku di taman baca seperti lokasi, lemari, rak dan baris. Pengurus mengakses halaman *View* Administrasi Buku setelah login. Pada halaman tersebut Pengurus memilih salah satu row pada buku yang lokasinya ingin diubah, lalu mengklik *button* ketiga pada kolom aksi. *Button* ketiga menampilkan kotak dialog untuk mengubah *dropdown* seperti lokasi, lemari, rak atau baris. Pengurus mengubah *dropdown* yang ingin diubah, saat data sudah siap disimpan, pengurus mengklik *button* simpan. *Button* simpan berisi perintah untuk mengirim *request* metode *POST* data buku ke *Controller*. *Model* kemudian menerima *query update* data buku dari *Controller* ke *tabel_buku*.



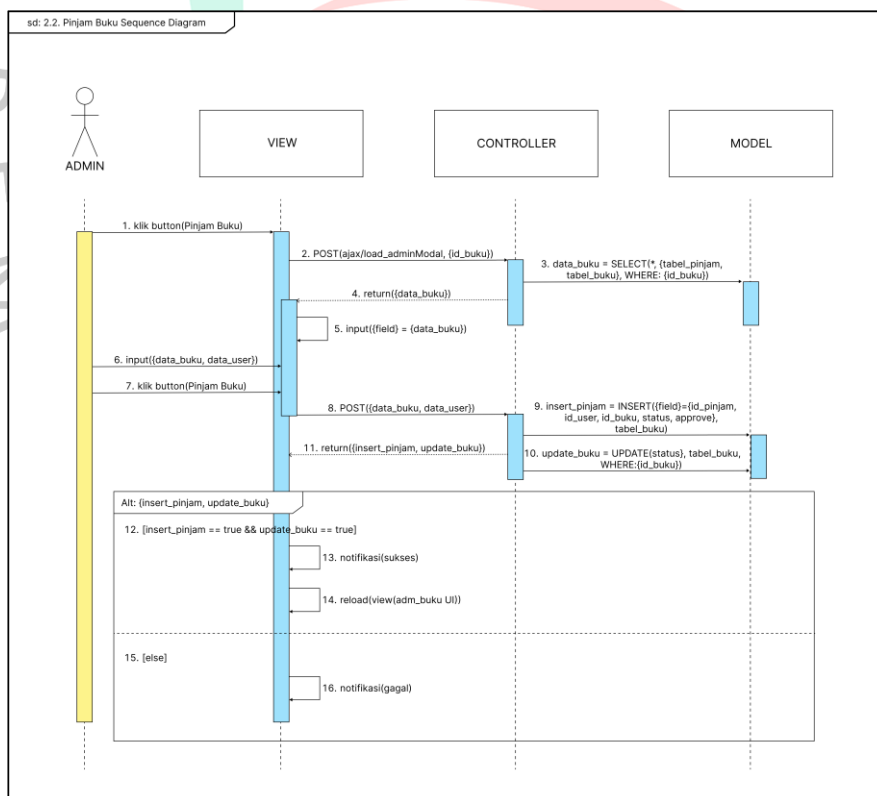
Gambar 4. 24 Sequence Diagram Ubah Lokasi Buku

Pada Gambar 4.25 menggambarkan *sequence diagram* dialog box untuk mengubah status buku. Pengurus mengakses halaman *View* Administrasi Buku setelah login. Pada halaman tersebut Pengurus memilih salah satu row pada buku yang statusnya ingin diubah, lalu mengklik *button* keempat pada kolom aksi. *Button* keempat menampilkan kotak dialog untuk mengubah *dropdown* status seperti buku hilang/rusak, buku dihibahkan, atau buku di-*publish*. Pengurus mengubah *dropdown* yang ingin diubah, saat data sudah siap disimpan, pengurus mengklik button simpan. *Button* simpan berisi perintah untuk mengirim *request* metode *POST* data buku ke *Controller*. *Model* kemudian menerima *query update* data buku dari *Controller* ke *tabel_buku*.



Gambar 4. 25 Sequence Diagram Ubah Status Buku

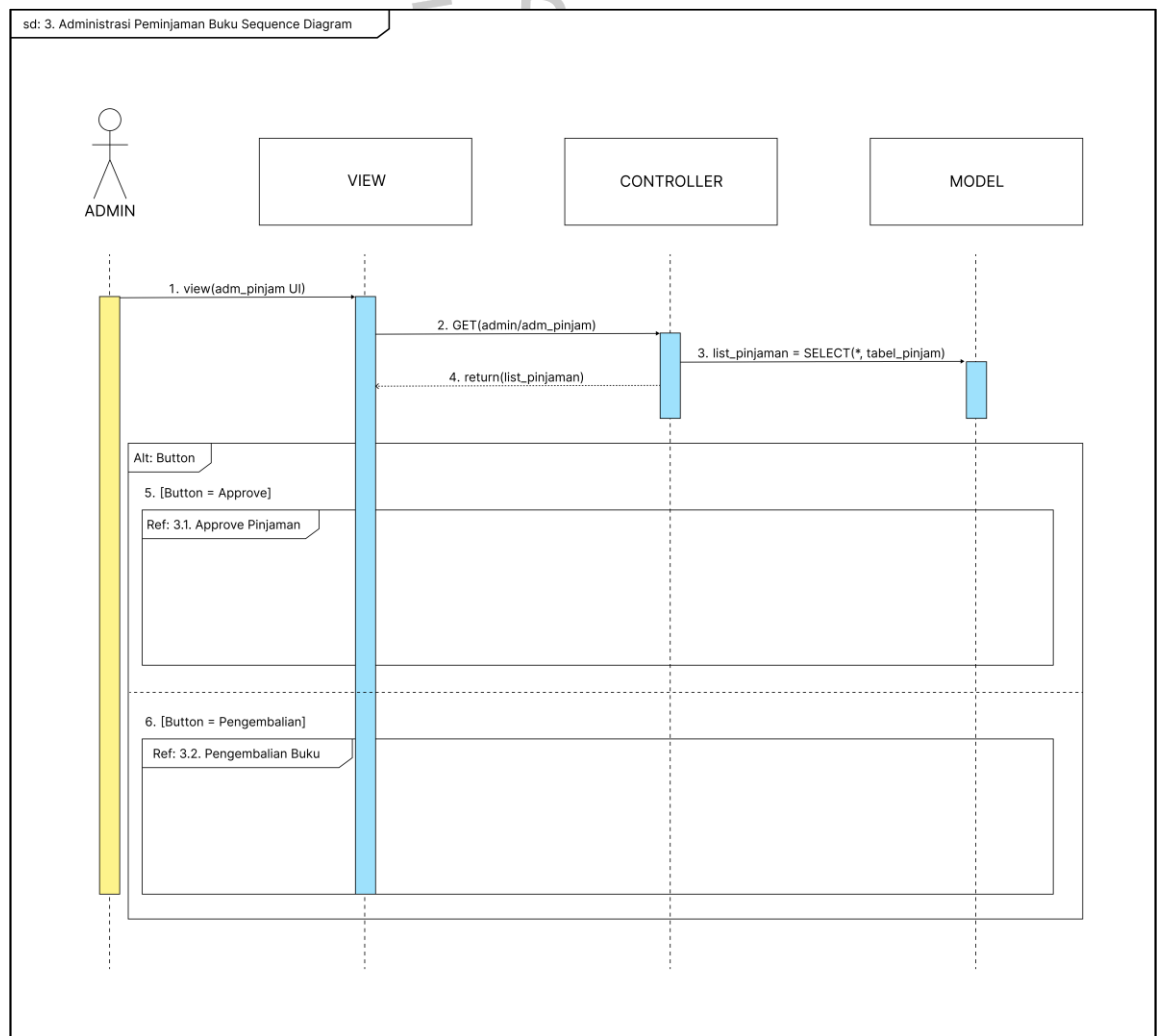
Pada Gambar 4.26 menggambarkan *sequence diagram* untuk membuat request peminjaman buku dari sisi pengurus. Member mengakses halaman *View* Administrasi Buku setelah login. Pada halaman tersebut Pengurus mengklik *button* ketiga pada kolom aksi ke row buku yang dipinjam. *Button* ketiga menampilkan kotak dialog peminjaman yang berisi email member, nama, dan tanggal pengembalian. Pengurus mengisi field email dari member yang sudah terdaftar dan sistem akan menampilkan nama lengkap member tersebut, lalu menentukan tanggal pengembalian dengan *datepicker*. Pengurus mengklik *button* simpan untuk mengirim *request* metode POST data pinjam buku (*id_pinjam*, *id_buku*, *id_user*, dll.) ke *Controller*. *Model* kemudian menerima *query insert* data pinjam buku dari *Controller* ke *tabel_pinjam*.



Gambar 4. 26 Sequence Diagram Pinjam Buku Lewat Pengurus

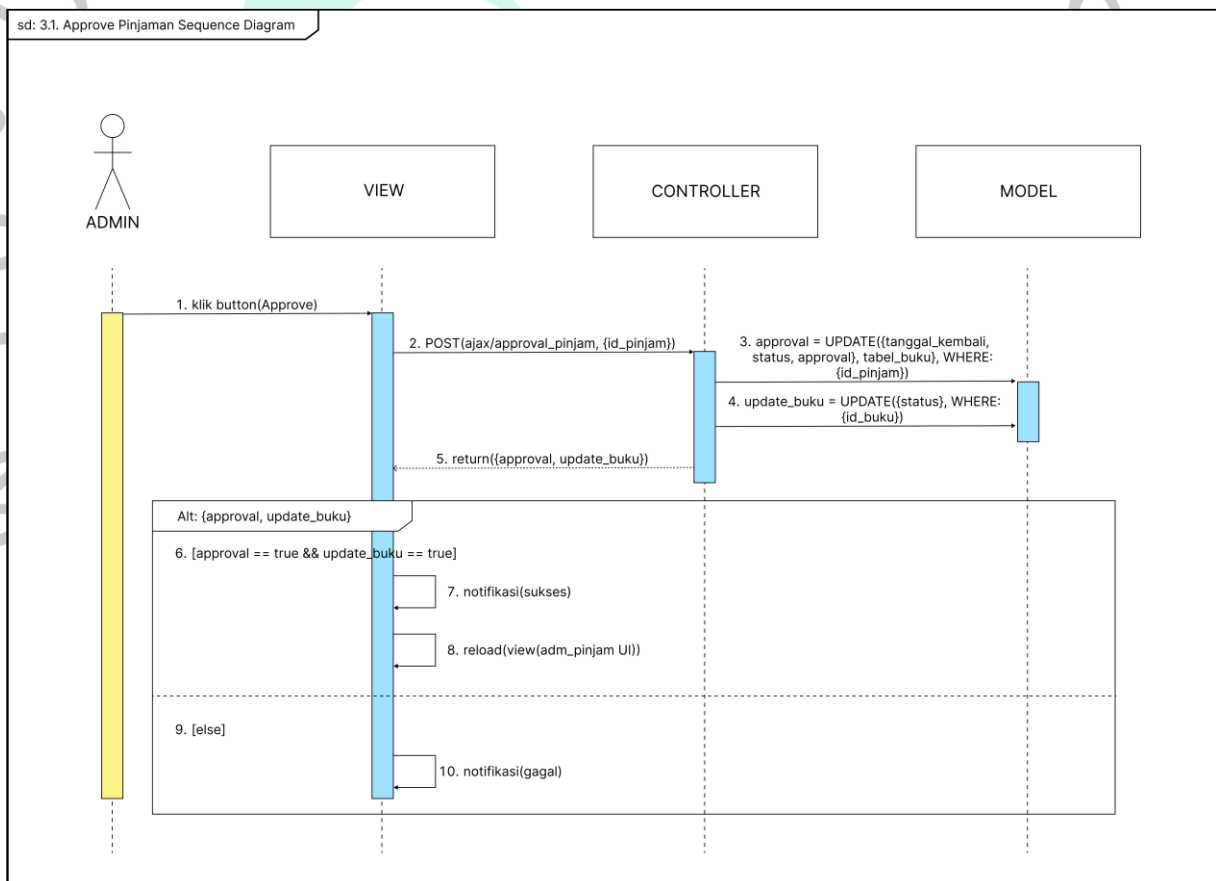
9) Administrasi Peminjaman

Pada Gambar 4.27 s.d. 4.29 menggambarkan *sequence diagram* halaman administrasi peminjaman buku serta aksi *button approve* (pinjam) dan *return* (kembali) buku. Sumber peminjaman yang ditampilkan berasal dari pengurus langsung atau dari permohonan member.



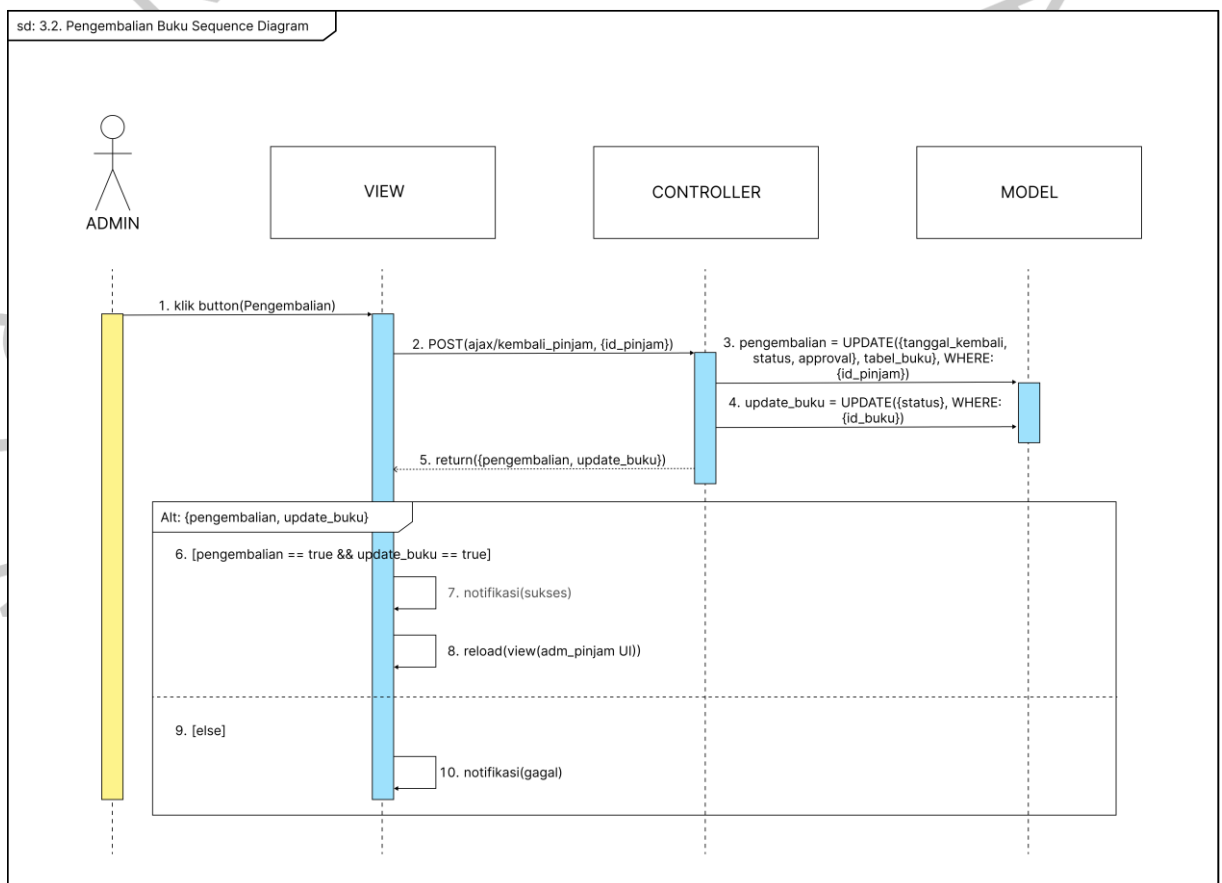
Gambar 4. 27 Sequence Diagram Administrasi Peminjaman

Jika pengurus ingin mengkonfirmasi peminjaman dari member, pengurus memilih *datepicker* pada salah satu row peminjaman dan menentukan tanggal pengembalian buku tersebut. Ketika tanggal sudah ditentukan proses selanjutnya mengklik button *approve*. *Controller* akan menerima *request* dari *button approve* dengan metode POST yang berisi *id_pinjam*, *id_user*, *id_buku* dan status. *Model* akan menerima *query* dari *Controller* yang berisi perintah *update* *id_pinjam*, *id_user*, *id_buku* dan status ke *tabel_pinjam*. Selanjutnya *button approve* akan berubah menjadi *button pengembalian*.



Gambar 4. 28 Sequence Diagram Approve Peminjaman Buku

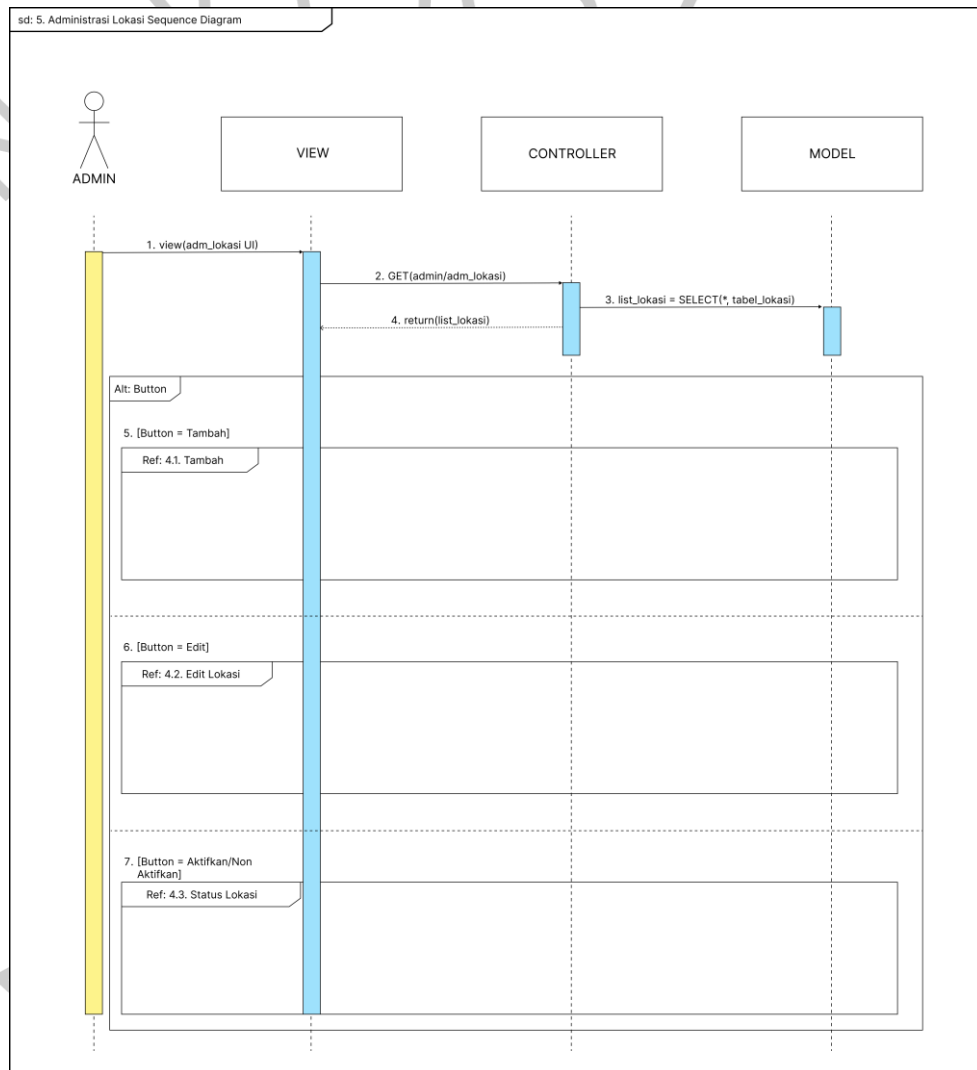
Jika pengurus ingin mengkonfirmasi pengembalian dari member, maka pengurus mengklik button pengembalian. *Controller* akan menerima *request* dari *button pengembalian* dengan metode POST yang berisi *id_pinjam*, *id_user*, *id_buku* dan *status*. *Model* akan menerima *query* dari *Controller* yang berisi perintah *update id_pinjam*, *id_user*, *id_buku* dan *status* ke tabel *pinjam*. Selanjutnya *row* tersebut akan hilang dari Administrasi Peminjaman dan buku dapat dipinjam kembali.



Gambar 4. 29 Sequence Diagram Return Peminjaman Buku

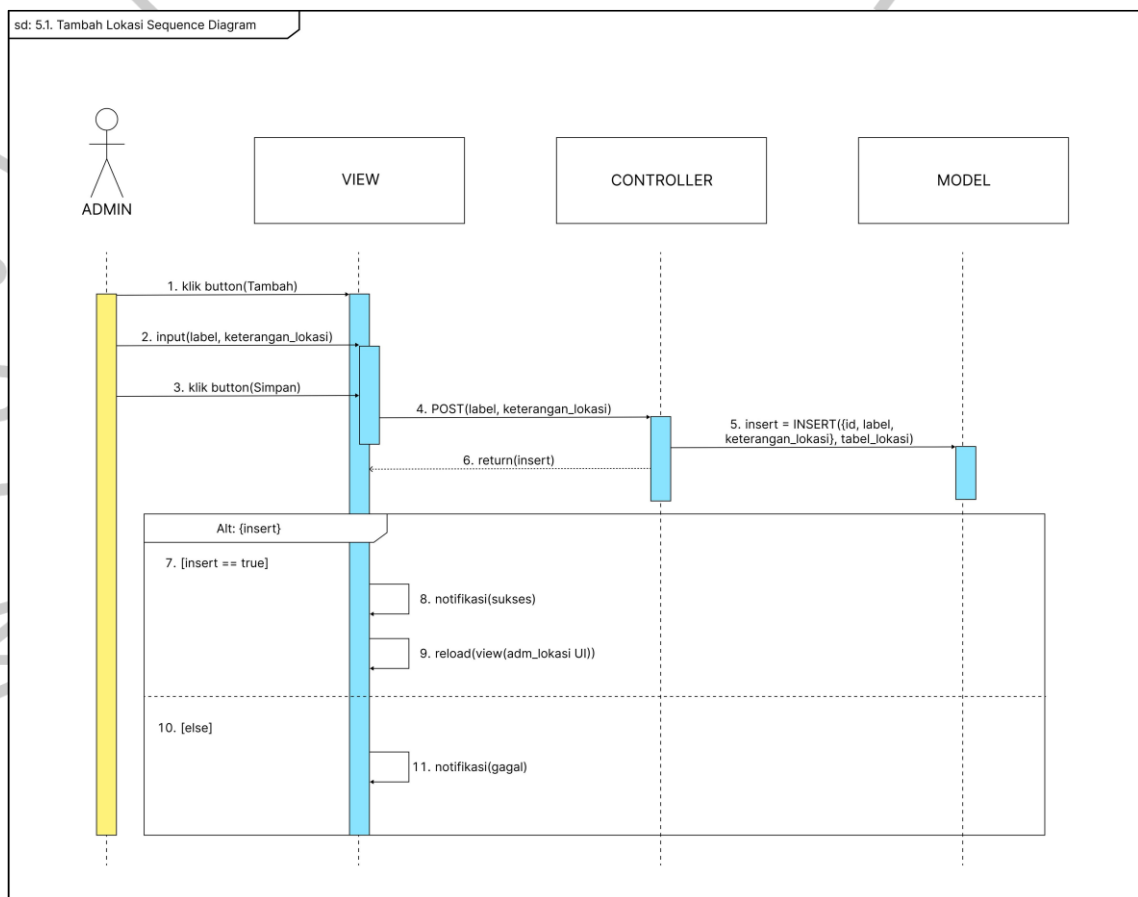
10) Administrasi Lokasi

Pada Gambar 4.30 s.d. 4.33 menggambarkan *sequence diagram* halaman konfigurasi lokasi. Member mengakses halaman *View* Administrasi User setelah login, pada halaman tersebut terdapat tiga *button* yaitu tambah lokasi, ubah lokasi dan *toggle* status lokasi.



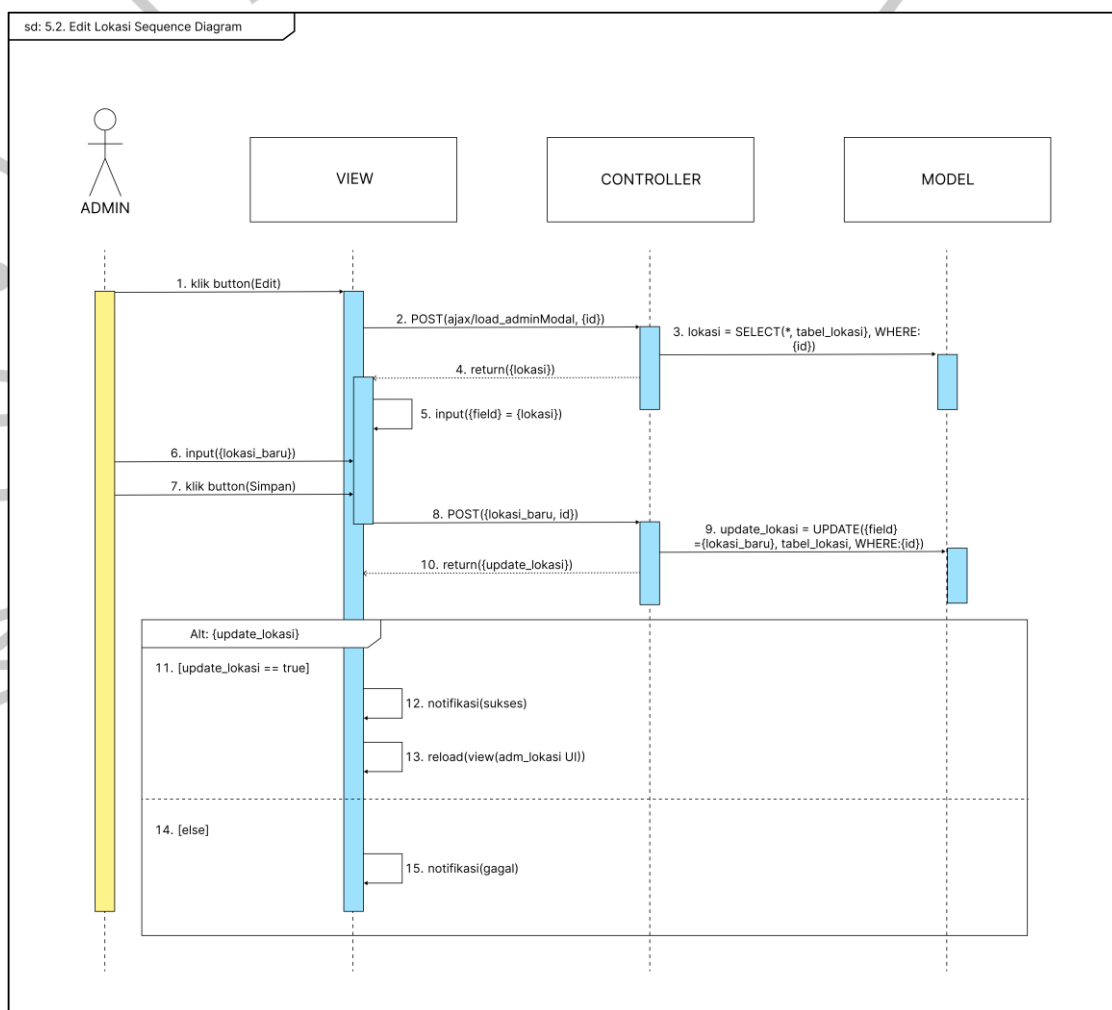
Gambar 4. 30 Sequence Diagram Administrasi Lokasi

Jika Pengurus ingin menambah lokasi baru, maka *button* tambah harus diklik. *Button* tambah menampilkan kotak dialog tambah lokasi yang berisi field label dan keterangan. Saat pengurus mengklik simpan pada kotak dialog tersebut, *Controller* akan menerima *request* dengan metode POST yang berisi *id_lokasi*, label dan keterangan. *Model* akan menerima *query* dari *Controller* yang berisi perintah *insert id_lokasi*, label dan keterangan ke tabel_lokasi.



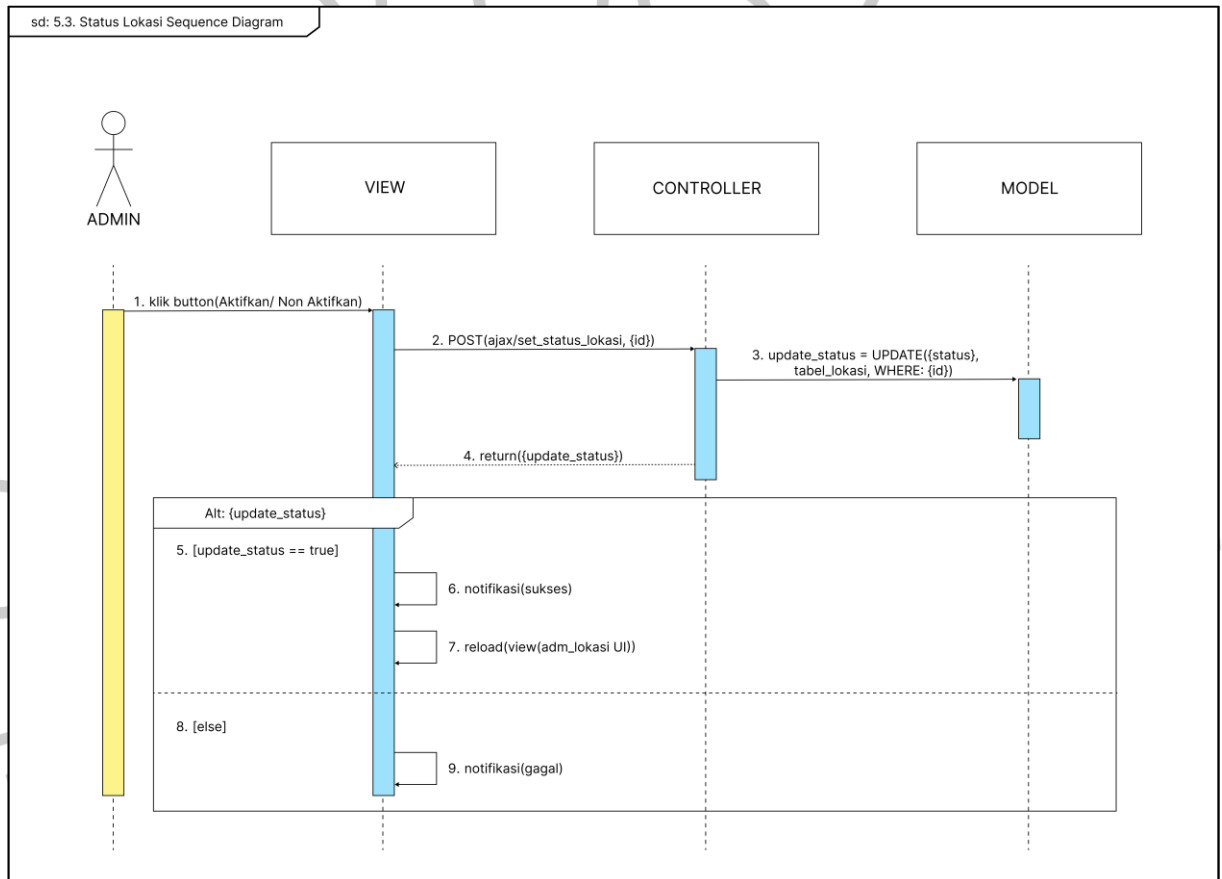
Gambar 4. 31 Sequence Diagram Tambah Lokasi

Jika Pengurus ingin mengubah informasi lokasi, maka *button* ubah harus diklik. *Button* ubah menampilkan kotak dialog ubah lokasi yang berisi field label dan keterangan. Saat pengurus mengklik simpan pada kotak dialog tersebut, *Controller* akan menerima *request* dengan metode POST yang berisi *id_lokasi*, label dan keterangan. *Model* akan menerima *query* dari *Controller* yang berisi perintah *update* label dan keterangan ke *tabel_lokasi*.



Gambar 4. 32 Sequence Diagram Ubah Lokasi

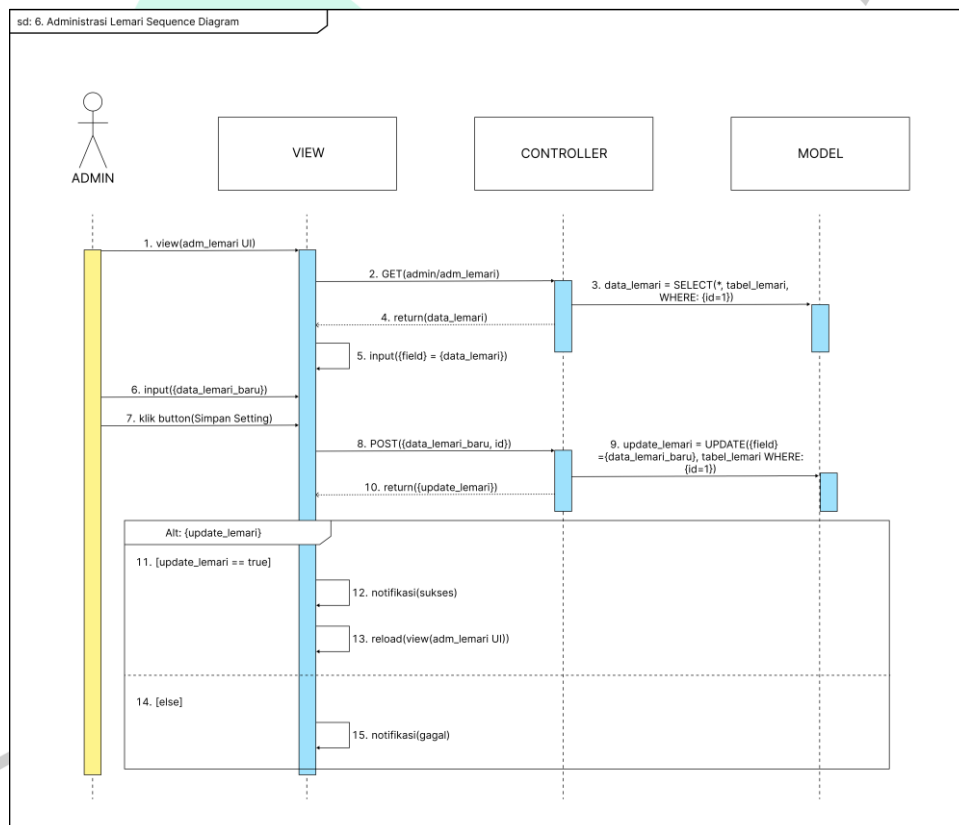
Jika Pengurus ingin mengubah status lokasi menjadi disable atau active, maka *button toggle* status lokasi harus diklik. *Controller* akan menerima *request* dengan metode POST yang berisi *id_lokasi* dan *status*. *Model* akan menerima *query* dari *Controller* yang berisi perintah *update id_lokasi* dan *status* ke *tabel_lokasi*.



Gambar 4. 33 Sequence Diagram Ubah Status Lokasi

11) Administrasi Lemari

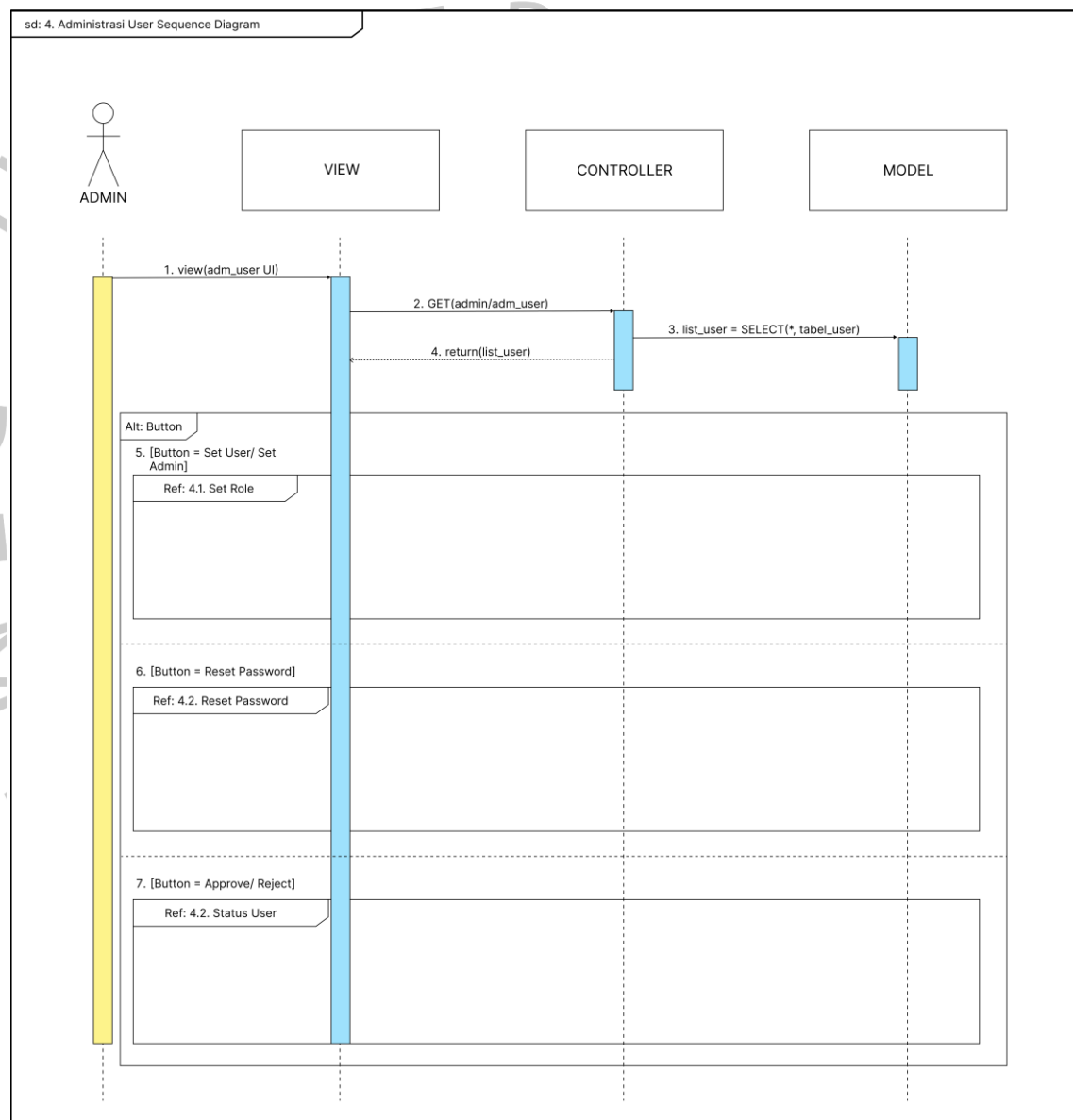
Pada Gambar 4.34 menggambarkan *sequence diagram* halaman konfigurasi jumlah lemari, rak dan baris jika sewaktu-waktu ada penambahan atau pengurangan pada *field* tersebut. Member mengakses halaman *View* Administrasi Lemari setelah login. Pada halaman tersebut Pengurus memilih *field* lemari, rak atau baris yang ingin diubah, lalu mengklik button simpan. *Button* simpan mengirim *request* metode POST data lemari ke *Controller*. *Model* kemudian menerima *query update* data buku dari *Controller* ke tabel_lemari.



Gambar 4. 34 Sequence Diagram Administrasi Lemari

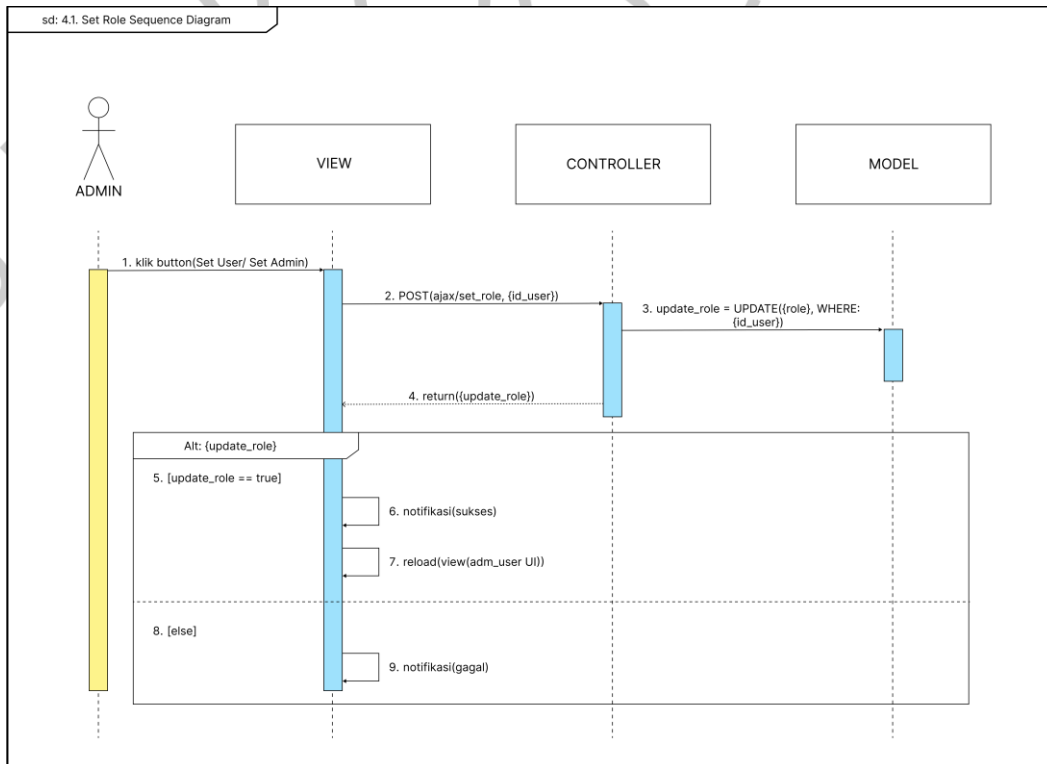
12) Administrasi User

Pada Gambar 4.35 s.d. 4.38 menggambarkan *sequence diagram* halaman administrasi user dan proses yang dilakukan pada halaman tersebut. Member mengakses halaman *View* Administrasi User setelah login, pada kolom aksi terdapat tiga *button* yaitu *toggle set role*, *reset password* dan *toggle status user*.



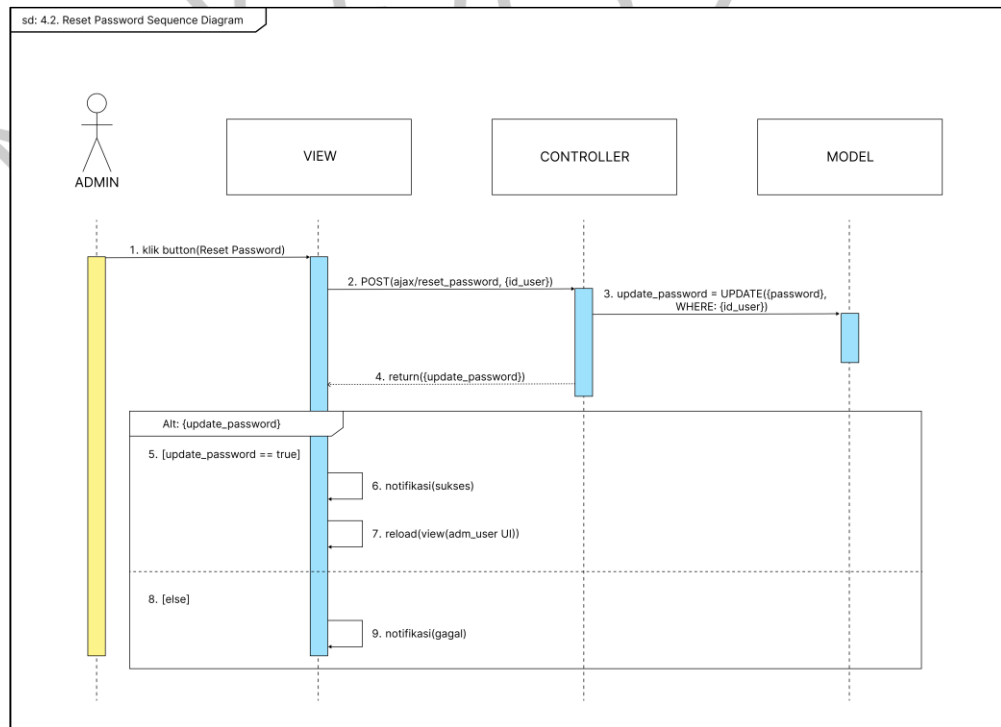
Gambar 4. 35 Sequence Diagram Administrasi User

Jika Pengurus ingin mengubah role member menjadi pengurus atau sebaliknya, maka *button toggle set role* harus diklik. *Controller* akan menerima *request* dengan metode POST yang berisi *id_user* dan *role*. *Model* akan menerima *query* dari *Controller* yang berisi perintah *update id_user* dan *role* ke *tabel_user*.



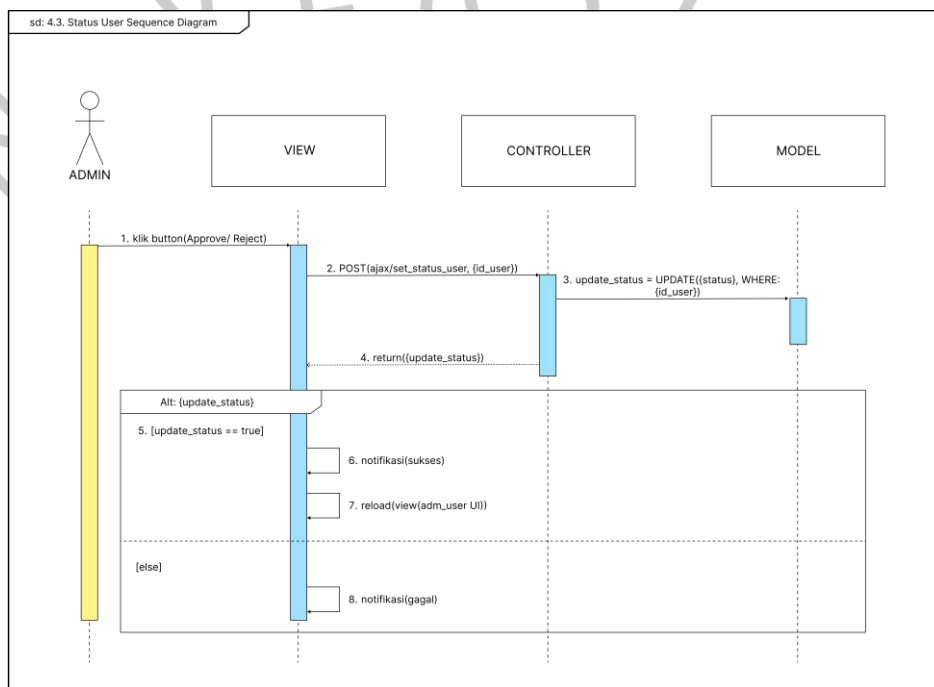
Gambar 4. 36 Sequence Diagram Set Role

Jika Pengurus ingin me-*reset password* menjadi *default*, maka *button reset password* harus diklik. *Controller* akan menerima *request* dengan metode *POST* yang berisi *id_user* dan *password*. *Model* akan menerima *query* dari *Controller* yang berisi perintah *update id_user* dan *password='123'* ke *tabel_user*.



Gambar 4. 37 Sequence Diagram Reset Password

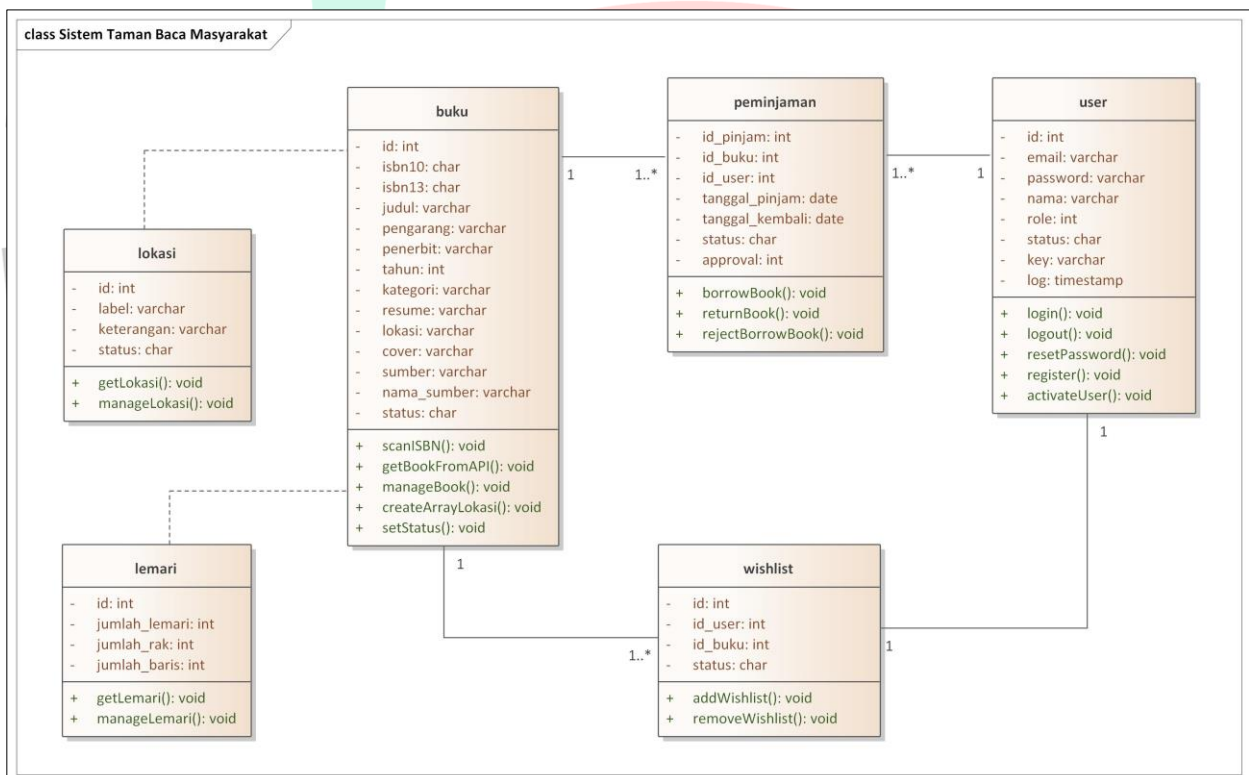
Jika Pengurus ingin mengubah status member menjadi disable atau active, maka *button toggle status user* harus diklik. *Controller* akan menerima *request* dengan metode POST yang berisi *id_user* dan status. *Model* akan menerima *query* dari *Controller* yang berisi perintah *update id_user* dan status ke tabel *user*.



Gambar 4. 38 Sequence Diagram Ubah Status User

4.2.5 Class Diagram

Pada sistem pengelolaan taman baca *class diagram* berfungsi untuk memvisualisasikan kelas dan hubungannya pada sistem yang berbentuk tabel. Kelas pada *class diagram* memiliki atribut *field* yang membangun kelas tersebut beserta tipe *field*. Selain itu kelas juga memiliki metode yang mewakili operasi/fungsi yang dilakukan. Hubungan antar kelas juga divisualisasikan menggunakan kardinalitas yang berbentuk notasi seperti *one-to-one* (1:1), *one-to-many* (1:N), *many-to-many* (N:M) dll. Berikut merupakan *class diagram* dari sistem pengelolaan taman bacaan masyarakat pada Gambar 4.39.



Gambar 4. 39 Class Diagram Sistem TBM

4.2.6 Specification Database Table

Pada sistem pengelolaan taman baca *specification database table* berfungsi untuk mendefinisikan struktur dan karakteristik dari sebuah basis data pada sistem tersebut. Masing-masing tabel memiliki komponen nama *field*, tipe data (*integer, character, date, dll.*), ukuran kolom serta atribut kolom seperti *primary key* dan *foreign key*. Berikut merupakan *specification database table* dari sistem taman bacaan masyarakat dapat dilihat pada Tabel 4.13 sampai dengan 4.18.

1) Tabel Buku

Nama Tabel: `tabel_buku`

Deskripsi: Menyimpan data buku yang ada di taman baca

Struktur: `id_buku+isbn10+isbn13+judul+pengarang+penerbit+tahun+kategori+resume+lokasi+cover+sumber`

+`nama_sumber+status`

Primary Key: `id_buku`

Tabel 4. 13 Specification Database Table "tabel_buku"

No.	Kolom	Tipe	Length	Keterangan
1	<code>id_buku</code>	Integer	11	ID unik untuk setiap buku (<i>Auto-increment</i>)
2	<code>isbn10</code>	Char	10	Nomor ISBN 10 buku
3	<code>isbn13</code>	Char	13	Nomor ISBN 13 buku
4	<code>judul</code>	Varchar	255	Judul buku
5	<code>pengarang</code>	Varchar	255	Nama pengarang buku
6	<code>penerbit</code>	Varchar	255	Nama penerbit buku
7	<code>tahun</code>	Integer	4	Tahun terbit buku
8	<code>kategori</code>	Varchar	255	Kategori buku
9	<code>resume</code>	Varchar	1000	Ringkasan buku
10	<code>lokasi</code>	Varchar	255	Menyimpan array dari gabungan lokasi, lemari, rak dan baris untuk menyimpan tempat buku diletakan.
11	<code>cover</code>	Varchar	255	URL gambar sampul buku
12	<code>sumber</code>	Varchar	255	Sumber perolehan buku
13	<code>nama_sumber</code>	Varchar	255	Nama sumber perolehan buku
14	<code>status</code>	Char	1	Status buku (misalnya: tersedia, dipinjam, rusak)

2) Tabel Lokasi

Nama Tabel: tabel_lokasi

Deskripsi: Menyimpan alamat lemari pada taman baca

Struktur: id_lokasi+label+keterangan+status

Primary Key: id_lokasi

Tabel 4. 3 Specification Database Table "tabel_lokasi"

No.	Kolom	Tipe	Length	Keterangan
1	id_lokasi	Integer	11	ID unik untuk setiap lokasi
2	label	Char	1	Identifikasi unik untuk setiap lokasi misalnya seperti "A" untuk Pos Taman Baca dan "B" untuk Gudang Taman Baca
3	keterangan	Varchar	255	Deskripsi lengkap mengenai lokasi seperti Pos Taman Baca atau Gudang Taman Baca
4	Status	Char	1	Status lokasi (misalnya: aktif, non-aktif)

3) Tabel Lemari

Nama Tabel : tabel_lemari

Deskripsi : Menyimpan alamat lemari pada taman baca

Struktur: id_lemari+jumlah_lemari+jumlah_rak+jumlah_baris

Primary Key: id_lemari

Tabel 4. 4 Specification Database Table "tabel_lemari"

No.	Kolom	Tipe	Length	Keterangan
1	Id_lemari	Integer	11	ID unik untuk setiap lemari
2	jumlah_lemari	Integer	11	Jumlah lemari dalam lokasi
3	jumlah_rak	Integer	11	Jumlah rak dalam setiap lemari
4	jumlah_baris	Integer	11	Jumlah baris dalam setiap rak

4) Tabel Pinjam

Nama Tabel : tabel_pinjam

Deskripsi : Mengelola peminjaman member sampai proses pengembalian buku

Struktur: id_pinjam+id_buku+id_user+tanggal_pinjam+tanggal_kembali+status+approval

Primary Key: id_pinjam

Foreign Key: id_buku, id_user

Tabel 4. 5 Specification Database Table "tabel_pinjam"

No.	Kolom	Tipe	Length	Keterangan
1	id_pinjam	Integer	11	ID unik untuk setiap peminjaman
2	id_buku	Integer	11	ID buku yang dipinjam (<i>Foreign Key</i> ke tabel_buku)
3	id_user	Integer	11	ID pengguna yang meminjam buku (<i>Foreign Key</i> ke tabel_user)
4	tanggal_pinjam	Date	-	Tanggal peminjaman buku
5	tanggal_kembali	Date	-	Tanggal pengembalian buku
6	status	Char	1	Status peminjaman (misalnya: sedang dipinjam, terlambat)
7	approval	Char	1	Status persetujuan peminjaman (misalnya: disetujui, ditolak)

5) Tabel User

Nama Tabel : tabel_user

Deskripsi : Menyimpan pengguna dalam website TBM

Struktur: id_user+email+password+nama+role+status+_key+log

Primary Key: id_user

Tabel 4. 6 Specification Database Table "tabel_user"

No.	Kolom	Tipe	Length	Keterangan
1	id_user	Integer	11	ID unik untuk setiap pengguna
2	email	Varchar	255	<i>Email</i> pengguna
3	password	Varchar	255	<i>Password</i> pengguna (terenkripsi)
4	nama	Varchar	255	Nama lengkap pengguna
5	role	Varchar	255	Peran pengguna (misalnya: member, pengurus)
6	status	Char	1	Status pengguna (misalnya: aktif, non-aktif)
7	_key	Varchar	255	Kunci token aktivasi email
8	log	Timestamp	-	<i>Timestamp</i> untuk menghitung link aktivasi email jika expire atau tidak

6) Tabel Wishlist

Nama Tabel : tabel_wishlist

Deskripsi : Menyimpan alamat lemari pada taman baca

Struktur: id_wishlist+id_user+id_buku+status

Primary Key: id_wishlist

Foreign Key: id_user, id_buku

Tabel 4. 7 Specification Database Table "tabel_wishlist"

No.	Kolom	Tipe	Length	Keterangan
1	Id_wishlist	Integer	11	ID unik untuk setiap wishlist
2	id_user	Integer	11	ID pengguna yang membuat <i>wishlist</i> (<i>Foreign Key</i> ke tabel_user)
3	id_buku	Integer	11	ID buku yang diinginkan (<i>Foreign Key</i> ke tabel_buku)
4	status	Char	1	Status <i>wishlist</i> (misalnya: <i>favorite</i> , <i>removed</i>)

4.3 Perancangan Antar Muka Pengguna

Bagian ini membahas rancangan *User Interface* (UI) atau antar muka bagi pengguna sistem taman baca menggunakan *wireframe*. *Wireframe* menggambarkan struktur atau tata letak elemen-elemen UI seperti menu navigasi, *button*, *form input* dll pada masing-masing halaman sistem. Berikut adalah perancangan antar muka dapat dilihat di Gambar 4.40 sampai dengan Gambar 4.86:

1) Login

Pada Gambar 4.40 menggambarkan UI halaman login pengguna yang dapat mengakses sistem utama TBM. User memasukan *field email* dan *password*, lalu mengklik *button login*. Jika sesuai *value* pada basis data maka menampilkan pesan login berhasil dan sebaliknya akan menampilkan pesan login gagal saat *value* tidak sesuai. Sistem juga mengecek apakah *field* tersebut telah terisi, jika kosong maka akan menampilkan pesan data tidak lengkap.

LOGO	KATALOG	KONTAK	USER
Login			
<input type="text" value="Email"/>			
<input type="password" value="Password"/>			
LOGIN			
Belum punya aku? Daftar di sini Lupa password? Reset password di sini			
About Us Kontak Kami	Wishlist Buku Pinjaman Buku	FAQ Term of Services Privacy Policy	LOGO Alamat Taman Baca Nomor Telepon Alamat Email
Copyright @Tahun Pengembang Aplikasi			

Gambar 4. 40 UI Login

2) Registrasi

Pada Gambar 4.41 menggambarkan UI halaman registrasi pengunjung melakukan pendaftaran agar menjadi member TBM. User memasukan *field*, nama, *email*, *password* dan konfirmasi *password*, lalu mengklik *button* daftar. Sistem mengecek apakah *field* tersebut telah terisi, jika kosong maka akan menampilkan pesan data tidak lengkap. Saat *field* sudah terisi dan tersimpan maka proses selanjutnya akan menampilkan pesan sukses dan harap cek *inbox email*. Jika sudah menerima pesan email dari sistem, pengguna mengklik link tersebut untuk mengaktifasi akun sehingga bisa melakukan login.

The image shows a web page for account registration. At the top left is a 'LOGO' and at the top right are links for 'KATALOG', 'KONTAK', and a 'USER' icon. The main heading is 'Pendaftaran Akun'. Below this are four input fields: 'Nama', 'Email', 'Password', and 'Ulang Password'. A red 'DAFTAR' button is positioned below the fields. The footer contains several sections: 'About Us' and 'Kontak Kami' on the left; 'Wishlist Buku' and 'Peminjaman Buku' in the middle; 'FAQ', 'Term of Services', and 'Privacy Policy' on the right; and a 'LOGO' with contact details (Alamat Taman Baca, Nomor Telepon, Alamat Email) on the far right. A copyright notice 'Copyright ©Tahun Pengembang Aplikasi' is at the bottom center.

Gambar 4. 41 UI Registrasi

3) Kirim Reset Password

Pada Gambar 4.42 menggambarkan UI halaman kirim link ubah *password* bagi pengguna. Halaman tersebut berisi text box untuk mengisi email dari akun yang ingin mengganti *password*. Saat field sudah terisi dan tersimpan maka proses selanjutnya akan menampilkan pesan sukses dan harap cek inbox *email*. Jika sudah menerima pesan email dari sistem, pengguna mengklik *link* tersebut untuk menuju form ubah *password*.

The image shows a web interface for sending a password reset link. At the top left is a 'LOGO'. To the right are links for 'KATALOG' and 'KONTAK', and a 'USER' profile icon. The main heading is 'Kirim Link Reset'. Below this is a text input field labeled 'Email'. Underneath the input field is a red button with the text 'KIRIM LINK'. The footer section contains several links: 'About Us', 'Kontak Kami', 'Wishlist Buku', 'Peminjaman Buku', 'FAQ', 'Term of Services', and 'Privacy Policy'. To the right of these links is another 'LOGO' and contact details: 'Alamat Taman Baca', 'Nomor Telepon', and 'Alamat Email'. At the very bottom, there is a copyright notice: 'Copyright @Tahun Pengembang Aplikasi'.

Gambar 4. 42 UI Kirim Link Reset Password

4) Form Reset Password

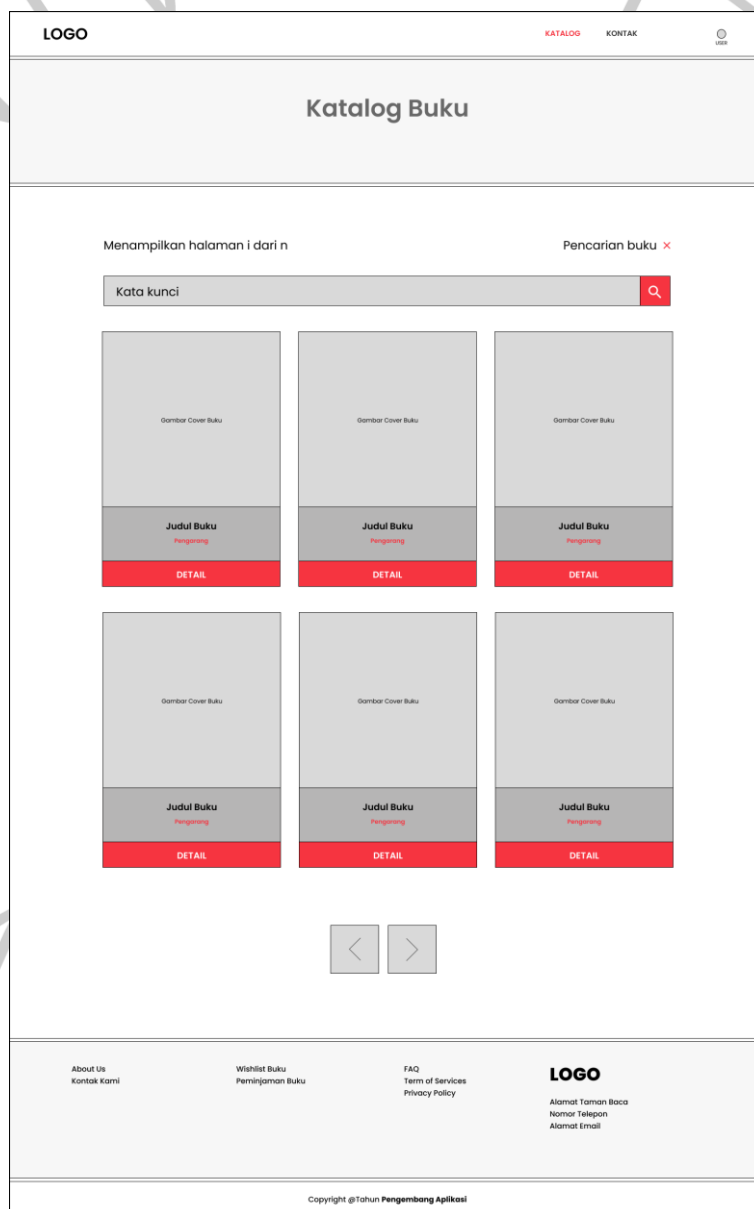
Pada Gambar 4.43 menggambarkan UI halaman form ubah password bagi pengguna. Ketika pelanggan sudah mengklik *link reset password*, maka halaman ubah password akan tampil. Halaman tersebut berisi isian *password* lama, *password* baru dan konfirmasi *password* baru. *Button* ubah *password* akan menyimpan *password* pengguna agar menjadi baru.

The image shows a web interface for changing a password. The page has a header with a logo, navigation links for 'KATALOG' and 'KONTAK', and a user profile icon. A dropdown menu is open, showing options like 'Nama User (Reset Pass)', 'Peminjaman Buku', 'Wishlist', and 'Logout'. The main content area is titled 'Ubah Password' and contains three input fields: 'Password Lama', 'Password Baru', and 'Ulang Password Baru'. Below these fields is a red button labeled 'UBAH PASSWORD'. The footer contains links for 'About Us', 'Kontak Kami', 'Wishlist Buku', 'Peminjaman Buku', 'FAQ', 'Term of Services', and 'Privacy Policy', along with a 'LOGO' and contact information: 'Alamat Taman Baca', 'Nomor Telepon', and 'Alamat Email'. A copyright notice 'Copyright @Tahun Pengembang Aplikasi' is at the bottom center.

Gambar 4. 43 UI Form Reset Password

5) Katalog Buku

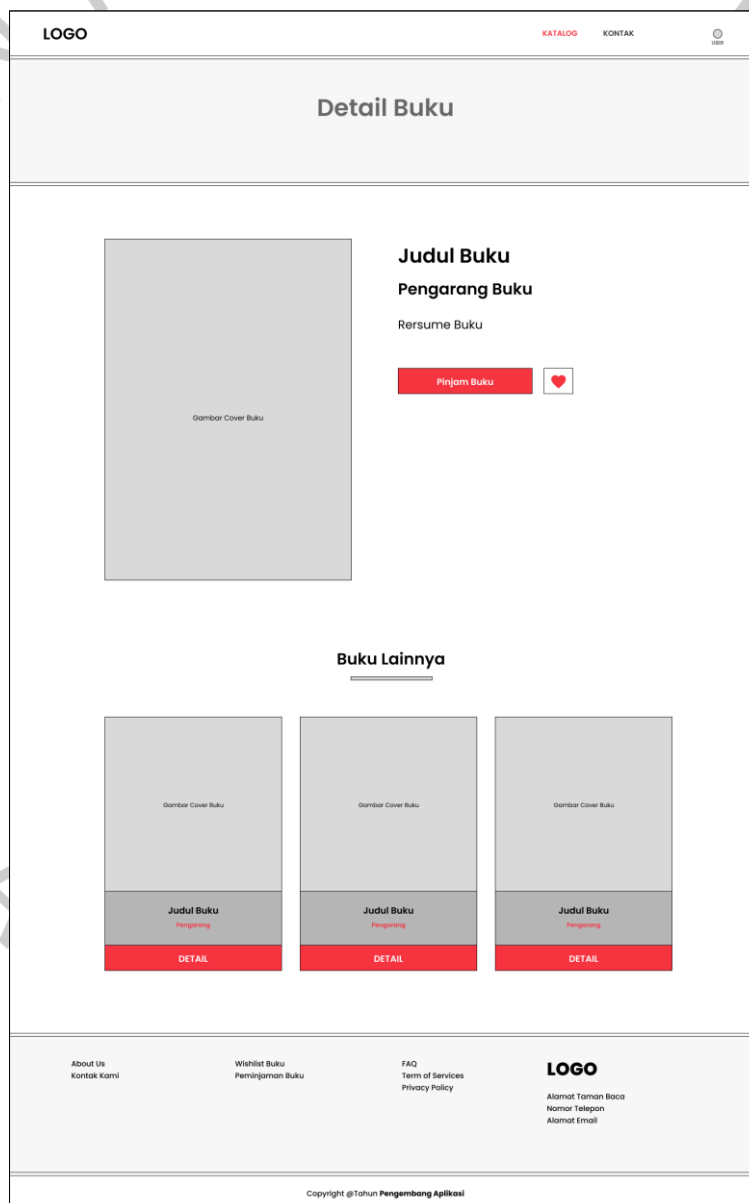
Pada Gambar 4.44 menggambarkan UI daftar buku yang tersedia untuk dipinjam oleh member. Katalog buku menampilkan maksimal sembilan buku per halaman. Pengguna dapat melakukan pencarian di *button* kanan atas daftar buku. Jika mengklik salah satu buku maka akan menampilkan halaman detail buku berdasarkan id-nya.



Gambar 4. 44 UI Katalog Buku

6) Detail Buku

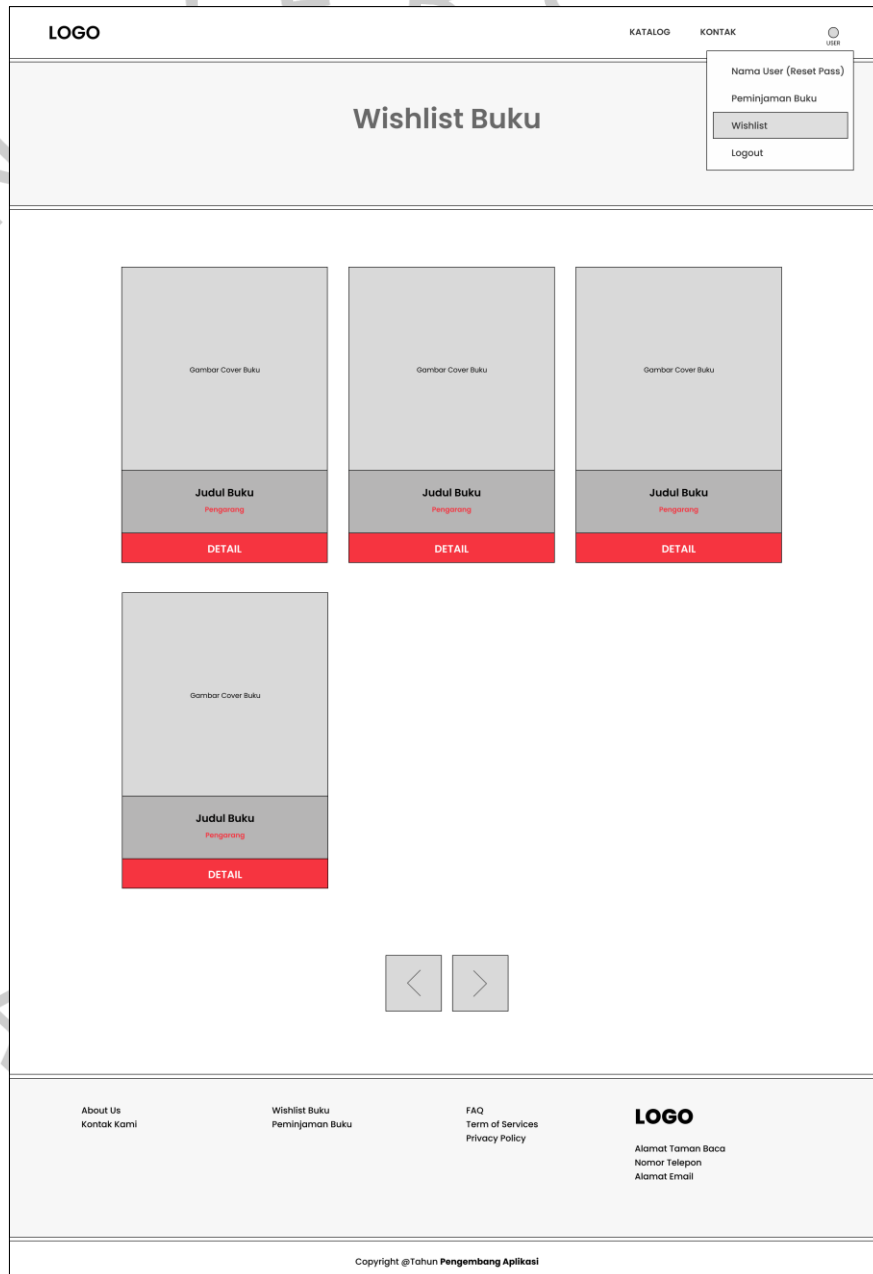
Pada Gambar 4.45 menggambarkan UI detail buku yang halaman detail buku berdasarkan id-nya agar member bisa melakukan *wishlist* atau peminjaman terhadap buku. Detail buku berisi sampul buku, judul, pengarang dan resume buku. Selain itu terdapat *button* pinjam buku dan *button toggle wishlist* untuk bisa menambah atau mengurangi *wishlist* buku.



Gambar 4. 45 UI Detail Buku

7) Wishlist Buku

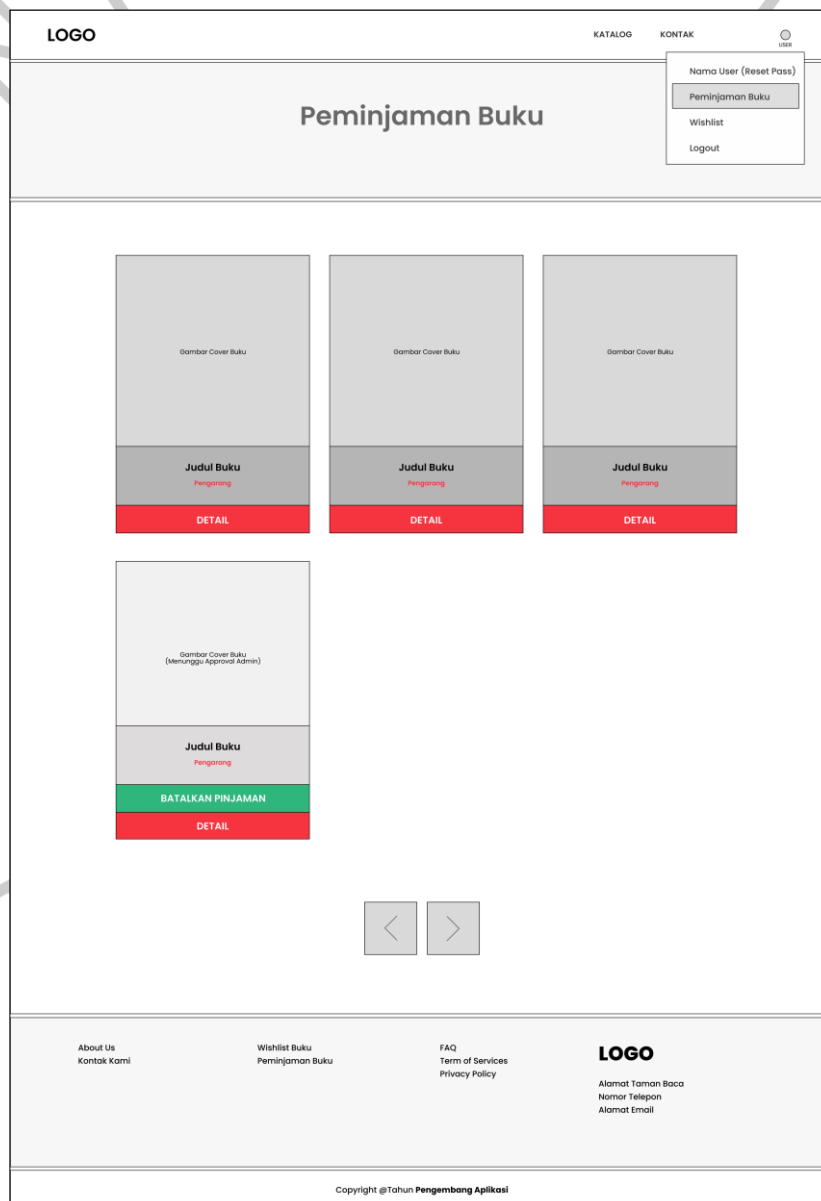
Pada Gambar 4.46 menggambarkan UI daftar buku-buku yang telah difavoritkan atau ditandai oleh member. Halaman *wishlist* buku menampilkan maksimal sembilan buku per halaman.



Gambar 4. 46 UI Wishlist Buku

8) Peminjaman Buku

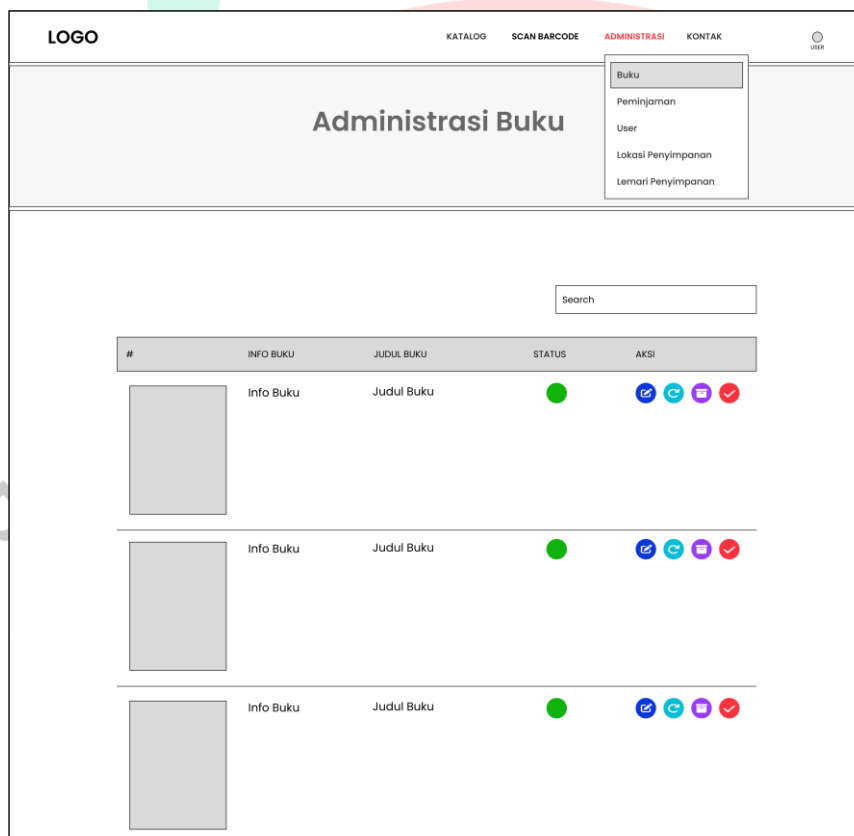
Pada Gambar 4.47 menggambarkan UI halaman daftar buku yang dipinjam oleh member. Halaman peminjaman buku menampilkan maksimal sembilan buku per halaman. Member dapat mengklik *button* batalkan peminjaman jika mengarahkan *cursor* pada salah satu buku.



Gambar 4. 47 UI Peminjaman Buku

9) Administrasi Buku

Pada Gambar 4.48 menggambarkan UI halaman daftar tabel buku yang dikelola oleh pengurus. Halaman administrasi buku menampilkan tabel dengan *field* gambar, informasi buku (ISBN, pengarang & lokasi buku), judul buku, status buku, dan *button* aksi. *Button* aksi pertama adalah *button* untuk mengubah detail buku, *button* aksi kedua adalah *button* untuk peminjaman buku dari sisi admin, *button* ketiga adalah *button* untuk mengubah lokasi buku, serta *button* aksi keempat adalah untuk mengubah status buku. Status buku secara *default* berwarna hijau yang berarti buku ter-*display* di halaman katalog, status buku dapat berwarna merah jika buku hilang/rusak.



Gambar 4. 48 UI Administrasi Buku

10) Administrasi Buku – Add

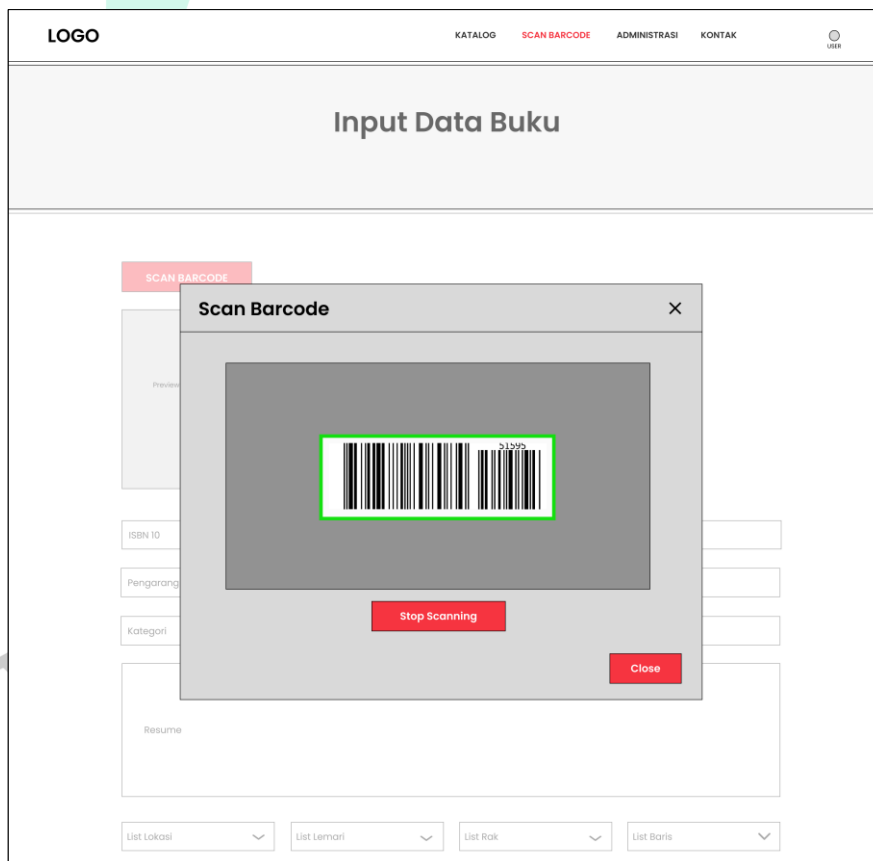
Pada Gambar 4.49 menggambarkan UI halaman input data buku oleh pengurus. *Button scan barcode* akan menampilkan kotak dialog untuk halaman scan buku menggunakan kamera pada *device* tertentu. Di bawah *button scan* terdapat *preView cover* buku dan akan otomatis mengisi *field* seperti ISBN, Judul, Pengarang, Penerbit, Kategori dan Resume jika *scan* berhasil. *Button browse* gambar digunakan untuk mengunggah gambar atau memfoto secara langsung buku di tempat taman baca. Selanjutnya terdapat *dropdown* letak buku yang berisi lokasi, lemari, rak dan baris buku serta *dropdown* sumber donasi.

The screenshot shows a web application interface for adding book data. At the top, there is a navigation bar with 'LOGO', 'KATALOG', 'SCAN BARCODE', 'ADMINISTRASI', 'KONTAK', and a user profile icon. The main heading is 'Input Data Buku'. Below this, there is a red 'SCAN BARCODE' button. Underneath is a 'PreView Cover Buku' placeholder. The form contains several input fields: ISBN 10, ISBN 13, Judul, Pengarang, Penerbit, Tahun, Kategori, and File Cover. A large text area is labeled 'Resume'. At the bottom, there are dropdown menus for 'List Lokasi', 'List Lemari', 'List Rak', 'List Baris', and 'List Sumber Buku', along with a 'Nama Sumber' input field. A red 'SIMPAN BUKU' button is located at the bottom left of the form.

Gambar 4. 49 UI Add Buku

11) Administrasi Buku – Scan

Pada Gambar 4.50 menggambarkan UI kotak dialog *scan* buku oleh pengurus yang tersambung oleh API Google Books. Halaman ini yang berisi *button request camera permission* ketika pertama kali diakses dan akan menampilkan *dropdown select camera*. Saat sudah memilih kamera akan menyala, dan pengurus bisa memposisikan *barcode* buku di dalam kotak merah kamera agar ISBN terbaca. Jika *scan* berhasil akan otomatis menampilkan gambar dan mengisi *field* seperti ISBN, Judul, Pengarang, Penerbit, Kategori dan Resume. Disamping itu jika gagal, *field* ISBN tetap terisi namun akan tampil pesan buku tidak ditemukan.



Gambar 4. 50 UI Scan Buku

12) Administrasi Buku – Ubah Detail

Pada Gambar 4.51 menggambarkan UI kotak dialog untuk mengubah detail mengenai buku seperti ISBN, Judul, Pengarang, Penerbit, Tahun dan *button* unggah gambar *cover* buku terbaru. Terdapat *button* simpan untuk membarukan data dan *close button* untuk menutup kotak dialog ubah detail buku.

The image shows a web application interface for book administration. A modal dialog titled "Administrasi Buku" is open, displaying a form to "Ubah Detail Buku". The form includes the following fields and controls:

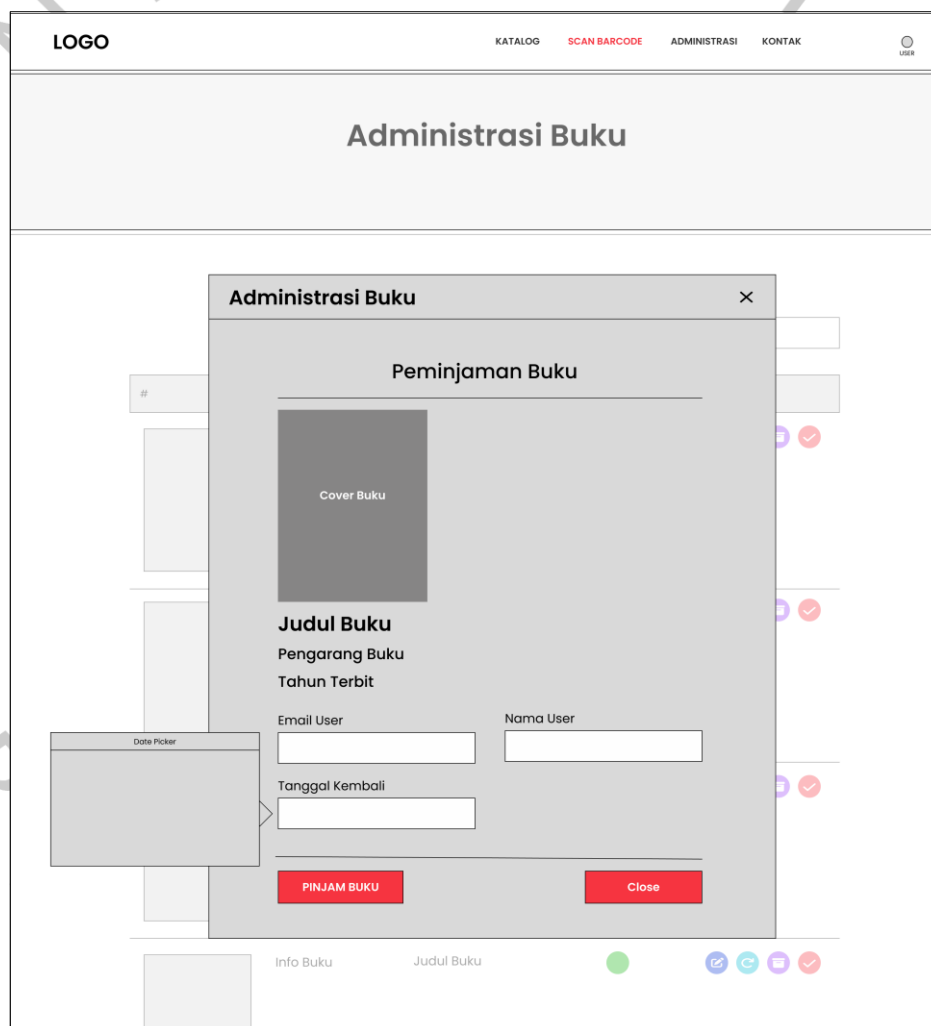
- ISBN 10:
- ISBN 13:
- Judul Buku:
- Pengarang:
- Penerbit:
- Tahun Terbit:
- Cover Buku: with a "Browse" button
- Resume Buku:

At the bottom of the dialog are two red buttons: "Simpan" (Save) and "Close". The background shows a navigation menu with "KATALOG", "SCAN BARCODE", "ADMINISTRASI", and "KONTAK".

Gambar 4. 51 UI Ubah Detail Buku

13) Administrasi Buku – Peminjaman Buku

Pada Gambar 4.52 menggambarkan UI kotak dialog untuk melakukan peminjaman buku dari sisi pengurus secara langsung, sehingga cukup membutuhkan email yang terdaftar dan *date picker* untuk memilih tanggal pengembalian. Terdapat *button* simpan untuk membarukan data dan *close button* untuk menutup kotak dialog peminjaman buku.



Gambar 4. 52 UI Pinjam Buku

14) Administrasi Buku – Ubah Lokasi

Pada Gambar 4.53 menggambarkan UI kotak dialog untuk mengubah lokasi suatu buku berupa *dropdown* lokasi (gudang/taman), lemari, rak dan baris. Terdapat *button* simpan untuk membarukan data dan *close button* untuk menutup kotak dialog ubah detail buku.

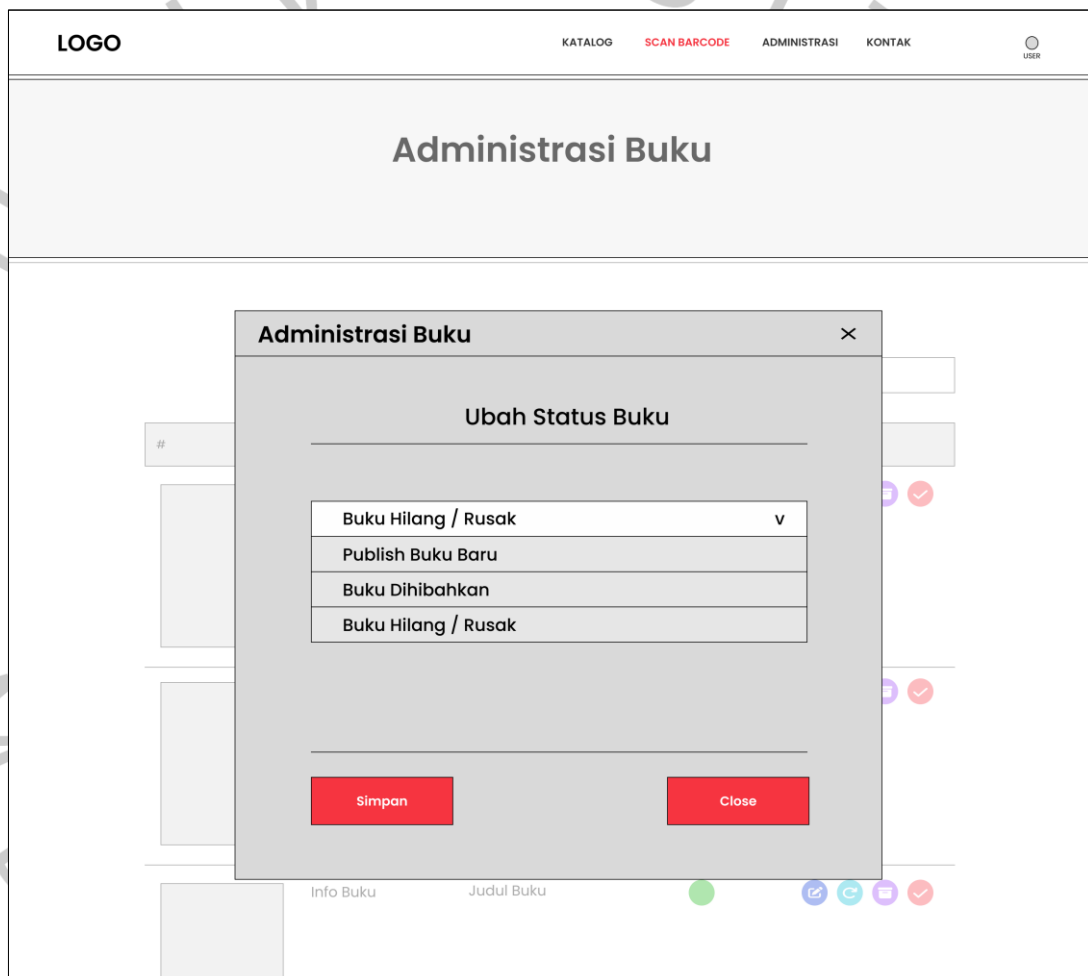
The image shows a web application interface with a navigation bar at the top containing 'LOGO', 'KATALOG', 'SCAN BARCODE', 'ADMINISTRASI', 'KONTAK', and 'USER'. The main content area is titled 'Administrasi Buku'. A modal dialog box titled 'Administrasi Buku' is open, with a sub-title 'Ubah Lokasi Buku'. The dialog box contains the following fields and buttons:

- Judul Buku
- Pengarang
- Tahun Terbit
- Pilih Lokasi (dropdown menu)
- Pilih Lemari (dropdown menu)
- Pilih Rak (dropdown menu)
- Pilih Nomor Baris (dropdown menu)
- Simpan (red button)
- Close (red button)

Gambar 4. 53 UI Ubah Lokasi Buku

15) Administrasi Buku – Ubah Status

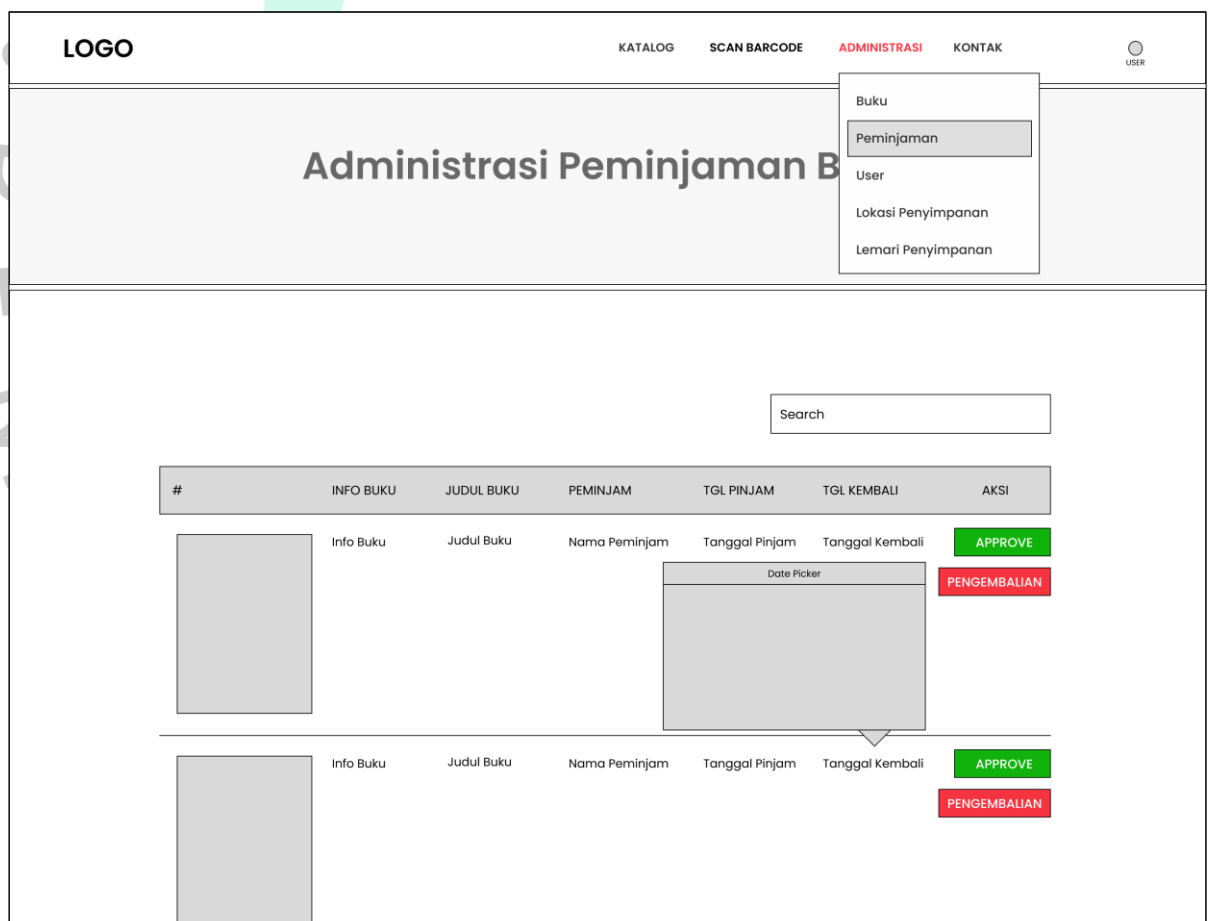
Pada Gambar 4.54 menggambarkan UI kotak dialog untuk mengubah status suatu buku berupa *dropdown* status yang memiliki *value publish*, buku dihibahkan, dan buku hilang/rusak. Terdapat *button* simpan untuk membarukan data dan *close button* untuk menutup kotak dialog ubah detail buku.



Gambar 4. 54 UI Ubah Status Buku

16) Administrasi Peminjaman

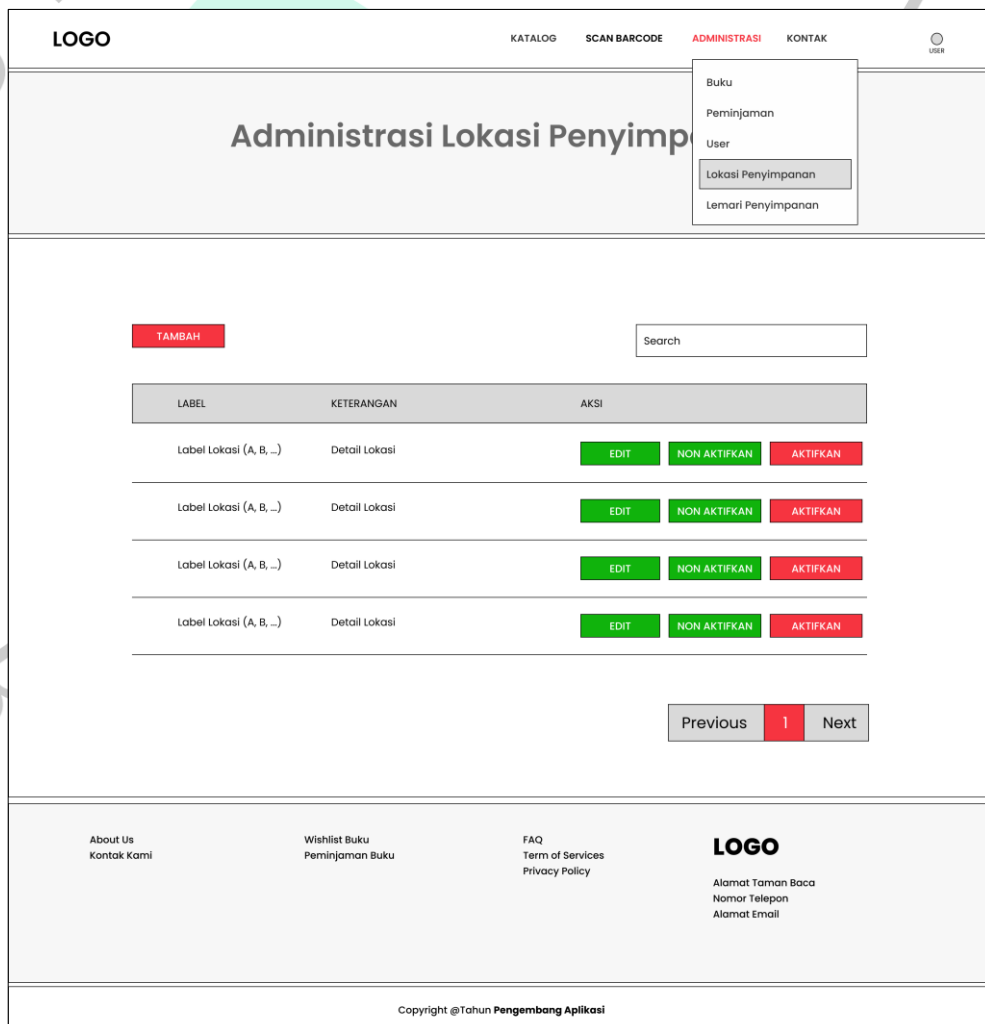
Pada Gambar 4.55 menggambarkan UI halaman daftar peminjaman buku yang dikelola oleh pengurus. Halaman administrasi peminjaman menampilkan tabel dengan *field* gambar, info buku (ISBN, pengarang, dll.), judul buku, peminjam, tanggal pinjam, tanggal pengembalian dan aksi. Pengurus mengatur tanggal pengembalian yang berbentuk *datepicker* dan mengklik *button approve* jika ingin menyetujui peminjaman buku. *Button approve* akan berubah menjadi *button* pengembalian saat proses peminjaman. Jika buku sudah dikembalikan di taman baca, maka pengurus akan mengklik *button* pengembalian.



Gambar 4. 55 UI Administrasi Peminjaman

17) Administrasi Lokasi

Pada Gambar 4.56 menggambarkan UI halaman daftar lokasi buku yang dikelola oleh pengurus. Halaman administrasi lokasi menampilkan tabel dengan *field* label lokasi (A, B, C), keterangan lokasi (Gudang, Taman, dll.) dan aksi yang berisi *button edit* dan *button toggle* aktivasi status. *Button edit* menampilkan dialog box untuk mengubah *field-field* pada lokasi. Terdapat *button* tambah di atas tabel yang menampilkan dialog box untuk menambah *data lokasi* baru.



Gambar 4. 56 UI Administrasi Lokasi Penyimpanan

18) Administrasi Lokasi – Ubah/Tambah

Pada Gambar 4.57 menggambarkan UI *dialog box* aksi menambah atau mengubah *field* label lokasi (A, B, C) dan keterangan lokasi (Gudang, Taman, dll.) pada lokasi penyimpanan taman baca.

The image shows a web application interface for 'Administrasi Lokasi Penyimpanan'. A modal dialog box is open, titled 'Administrasi Lokasi Penyimpanan' with a close button (X). The dialog box contains the text 'Tambah/ Ubah Lokasi Penyimpanan' and two input fields: 'Label' and 'Keterangan Lokasi'. Below the input fields are two buttons: 'Simpan' (Save) and 'Close'. The background shows a table with columns 'TAMBAH', 'LABEL', and 'KETERANGAN'. The 'TAMBAH' column has a red button, and the 'KETERANGAN' column has red buttons. The table has a 'Previous' button, a page number '1', and a 'Next' button. The footer contains links for 'About Us', 'Kontak Kami', 'Wishlist Buku', 'Peminjaman Buku', 'FAQ', 'Term of Services', 'Privacy Policy', and a 'LOGO' with contact information: 'Alamat Taman Baca', 'Nomor Telepon', and 'Alamat Email'. The copyright notice is 'Copyright @Tahun Pengembang Aplikasi'.

Gambar 4. 57 UI Tambah / Ubah Lokasi Penyimpanan

19) Administrasi Lemari

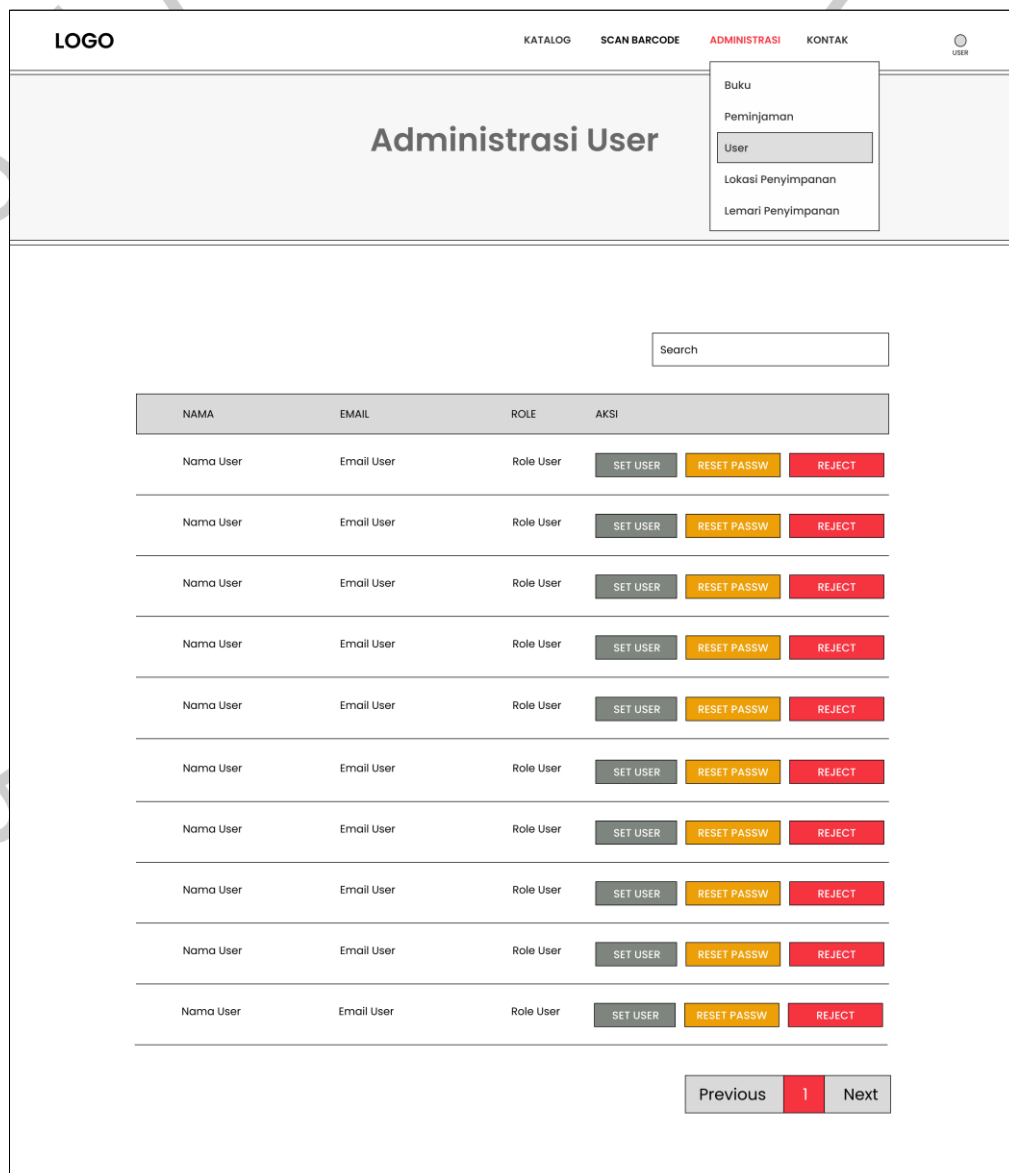
Pada Gambar 4.58 menggambarkan UI halaman daftar lemari buku yang dikelola oleh pengurus. Halaman administrasi lokasi menampilkan *text box* yang berisi jumlah maksimal lemari, rak dan baris pada taman baca yang dikelola pengurus. *Button* simpan digunakan untuk meng-*update* data lemari.

The image shows a web application interface for 'Administrasi Lemari Penyimpanan'. At the top, there is a navigation bar with a 'LOGO' on the left and menu items 'KATALOG', 'SCAN BARCODE', 'ADMINISTRASI', and 'KONTAK' on the right. A user profile icon labeled 'USER' is also present. A dropdown menu is open under 'ADMINISTRASI', showing options: 'Buku', 'Peminjaman', 'User', 'Lokasi Penyimpanan', and 'Lemari Penyimpanan'. The main content area features three text input fields labeled 'Jumlah Lemari', 'Jumlah Rak', and 'Jumlah Baris'. Below these fields is a red button labeled 'SIMPAN SETTING'. The footer contains several links: 'About Us', 'Kontak Kami', 'Wishlist Buku', 'Peminjaman Buku', 'FAQ', 'Term of Services', and 'Privacy Policy'. On the right side of the footer, there is a 'LOGO' and contact information: 'Alamat Taman Baca', 'Nomor Telepon', and 'Alamat Email'. At the very bottom, a copyright notice reads 'Copyright @Tahun Pengembang Aplikasi'.

Gambar 4. 58 UI Administrasi Lemari Penyimpanan

20) Administrasi User

Pada Gambar 4.59 menggambarkan UI halaman daftar pengguna yang berada pada sistem taman baca. Halaman administrasi pengguna menampilkan tabel yang berisi data pengguna dengan *field* nama, *email*, *role* dan aksi yang berisi button toggle role set user ke pengurus maupun member, *reset password* dan *toggle reject* atau *enable* pengguna.



Gambar 4. 59 UI Administrasi User

4.4 Perancangan Implementasi

Bagian ini membahas hasil implementasi rancangan pada sistem dan pengujian *black box*. Hasil implementasi rancangan memberi gambaran bahwa Sistem Informasi Taman Baca berbasis *Barcode ISBN* dengan *Google Books API* pada Komplek Taman Mangu Indah yang telah dikembangkan sesuai dengan kebutuhan. Disamping itu, pengujian *black box* merupakan tabel uji fungsional untuk memastikan setiap komponen pada sistem telah berfungsi dengan benar.

4.4.1 Hasil Implementasi Rancangan

Berikut adalah hasil implementasi rancangan dapat dilihat di Gambar 4.60 sampai dengan Gambar 4.86:

1) Login

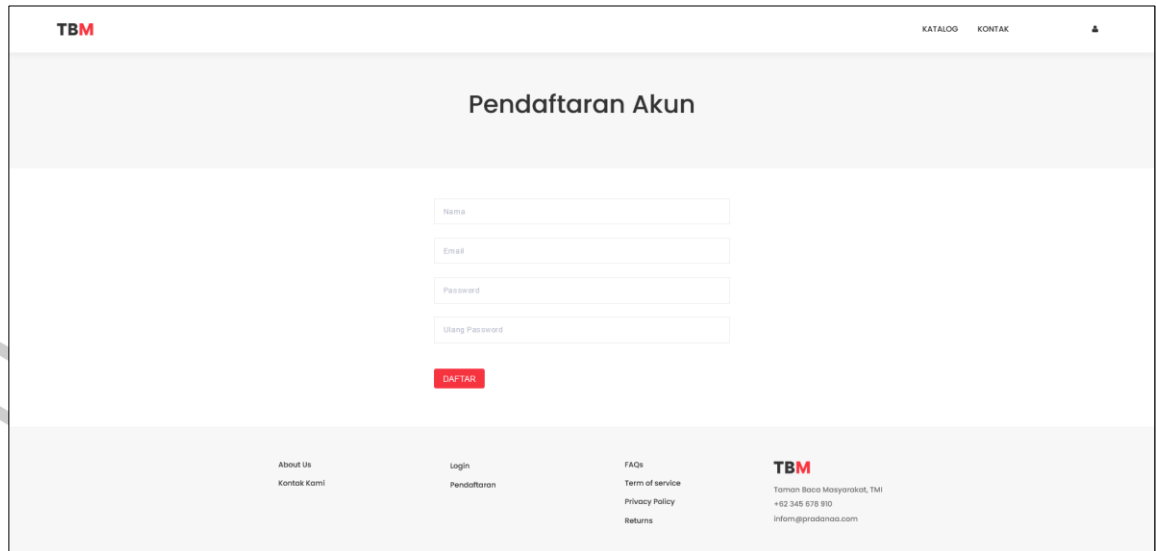
Pada Gambar 4.60 merupakan hasil implementasi halaman login pengguna yang dapat mengakses sistem utama TBM.

The image shows a web browser window displaying the login page of the TBM (Taman Baca Masyarakat) system. The page has a clean, modern design with a light gray background. At the top left, the TBM logo is visible. To the right of the logo, there are links for 'KATALOG' and 'KONTAK'. The main heading of the page is 'Login'. Below the heading, there is a form with two input fields. The first field is for the email address, which is pre-filled with 'alfianpradana4@gmail.com'. The second field is for the password, which is masked with three dots. Below the form, there is a red button labeled 'LOGIN'. Underneath the button, there are two links: 'Belum punya akun? Daftar di sini' and 'Lupa password? Reset password di sini'. At the bottom of the page, there is a footer with a grid of links. The first column contains 'About Us' and 'Kontak Kami'. The second column contains 'Login' and 'Pendaftaran'. The third column contains 'FAQs', 'Term of service', 'Privacy Policy', and 'Returns'. The fourth column contains the TBM logo, 'Taman Baca Masyarakat, TMI', '+62 345 678 910', and 'infom@pradanaa.com'.

Gambar 4. 60 Hasil Login

2) Registrasi

Pada Gambar 4.61 merupakan hasil implementasi halaman registrasi pengunjung yang melakukan pendaftaran agar menjadi member TBM

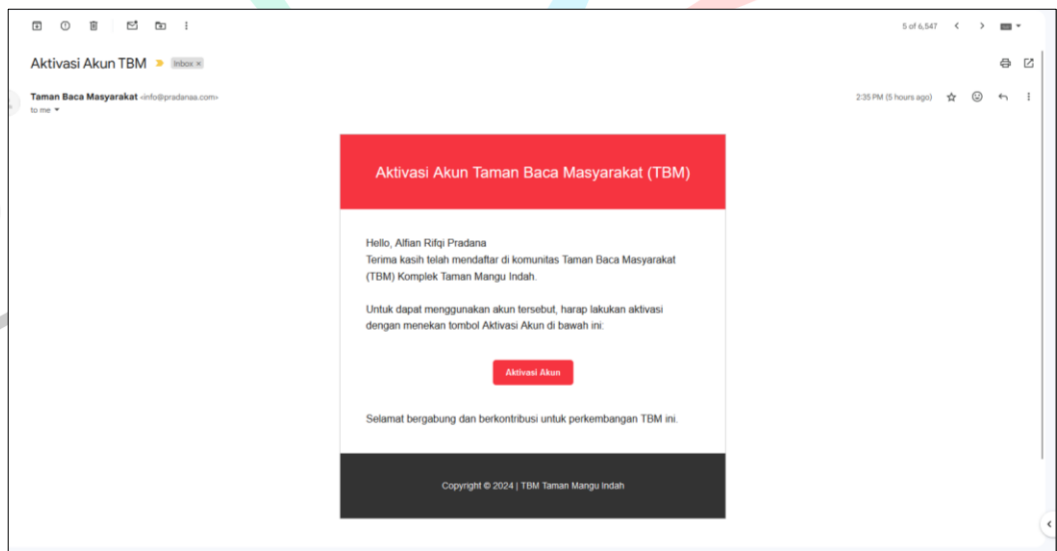


The screenshot shows a web page titled "Pendaftaran Akun" (Account Registration) for TBM. The page features a registration form with the following fields: "Nama" (Name), "Email", "Password", and "Ulang Password" (Repeat Password). A red "DAFTAR" (Register) button is positioned below the form. The footer contains navigation links for "About Us", "Kontak Kami", "Login", "Pendaftaran", "FAQs", "Term of service", "Privacy Policy", and "Returns". Contact information for TBM is also provided, including the address "Taman Baca Masyarakat, TMI", phone number "+62 345 678 910", and email "info@pradana.com".

Gambar 4. 61 Hasil Registrasi

3) Inbox Registrasi

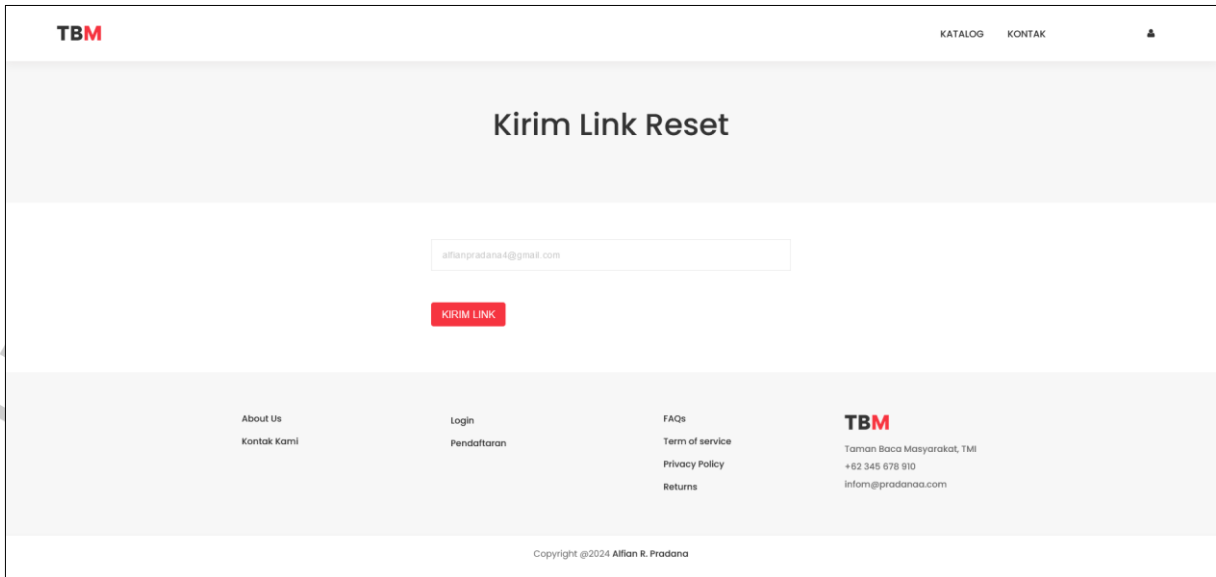
Pada Gambar 4.62 merupakan hasil implementasi pesan yang dikirimkan oleh sistem ketika pengunjung melakukan aktivasi member



Gambar 4. 62 Hasil Inbox Registrasi

4) Form Email Reset Password

Pada Gambar 4.63 merupakan hasil implementasi form email yang diisi oleh member ketika ingin melakukan *reset password*.

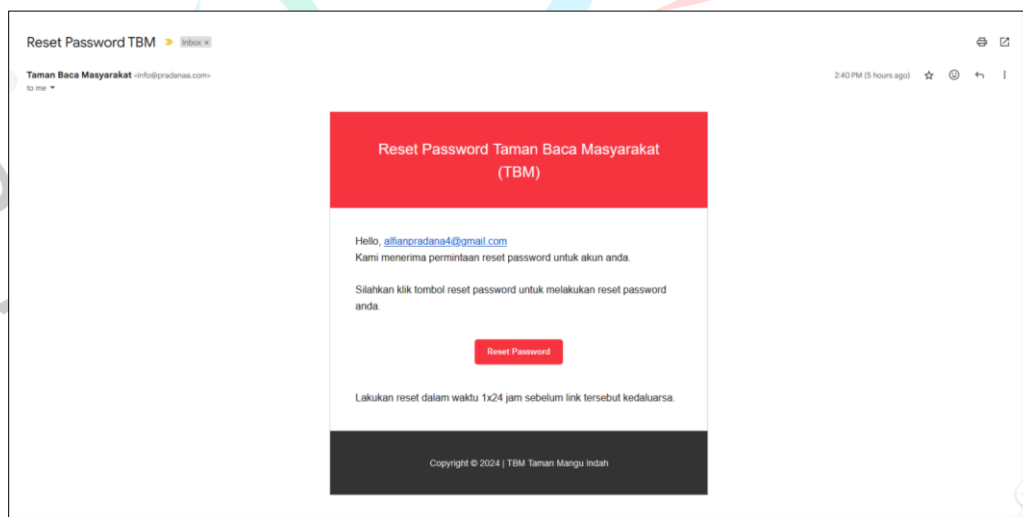


The screenshot shows a web form titled "Kirim Link Reset" on the TBM website. At the top left is the TBM logo, and at the top right are links for "KATALOG" and "KONTAK". The form itself consists of a text input field containing the email address "aflianpradana4@gmail.com" and a red button labeled "KIRIM LINK". Below the form is a footer section with navigation links: "About Us", "Kontak Kami", "Login", "Pendaftaran", "FAQs", "Term of service", "Privacy Policy", and "Returns". On the right side of the footer is the TBM logo and contact information: "Taman Baca Masyarakat, TBM", "+62 345 678 910", and "inform@pradana.com". At the very bottom, it says "Copyright ©2024 Aflian R. Pradana".

Gambar 4. 63 Hasil Form Email Reset Password

5) Inbox Reset Password

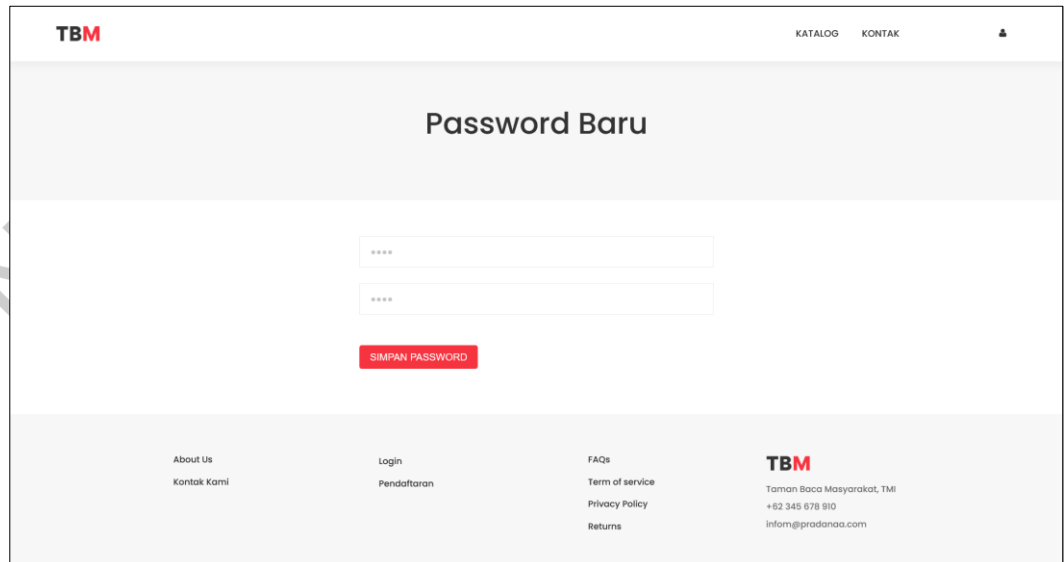
Pada Gambar 4.64 merupakan hasil implementasi pesan yang dikirimkan oleh sistem ketika pengunjung melakukan *reset password* member.



Gambar 4. 64 Hasil Inbox Reset Password

6) Form Reset Password

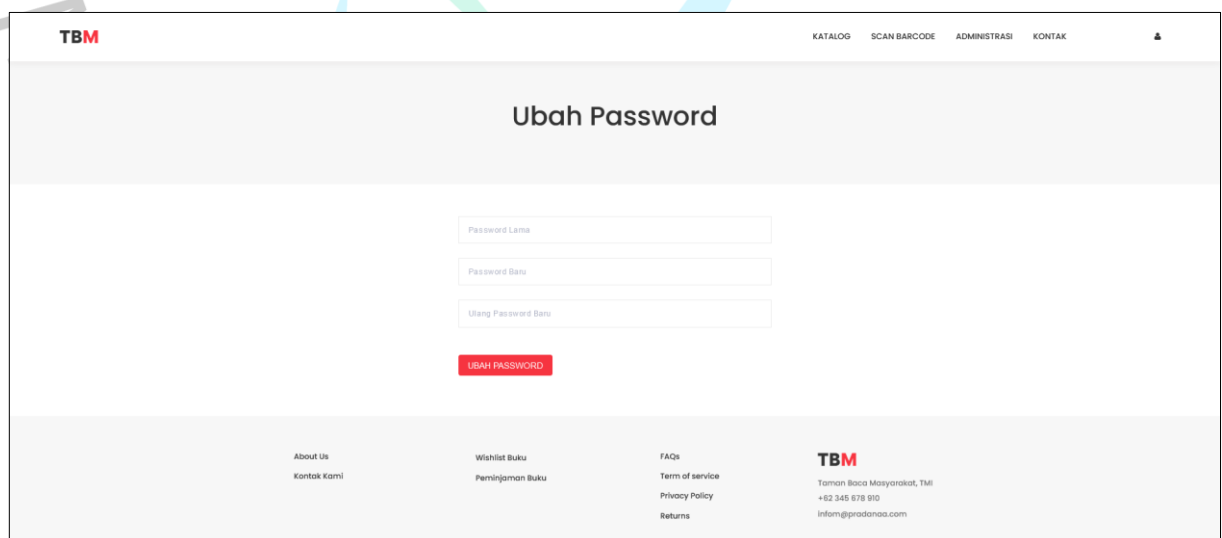
Pada Gambar 4.65 merupakan hasil implementasi *form reset password* yang diisi oleh member ketika sudah mengklik link di *inbox reset password*.



Gambar 4. 65 Hasil Form Reset Password

7) Form Ubah Password

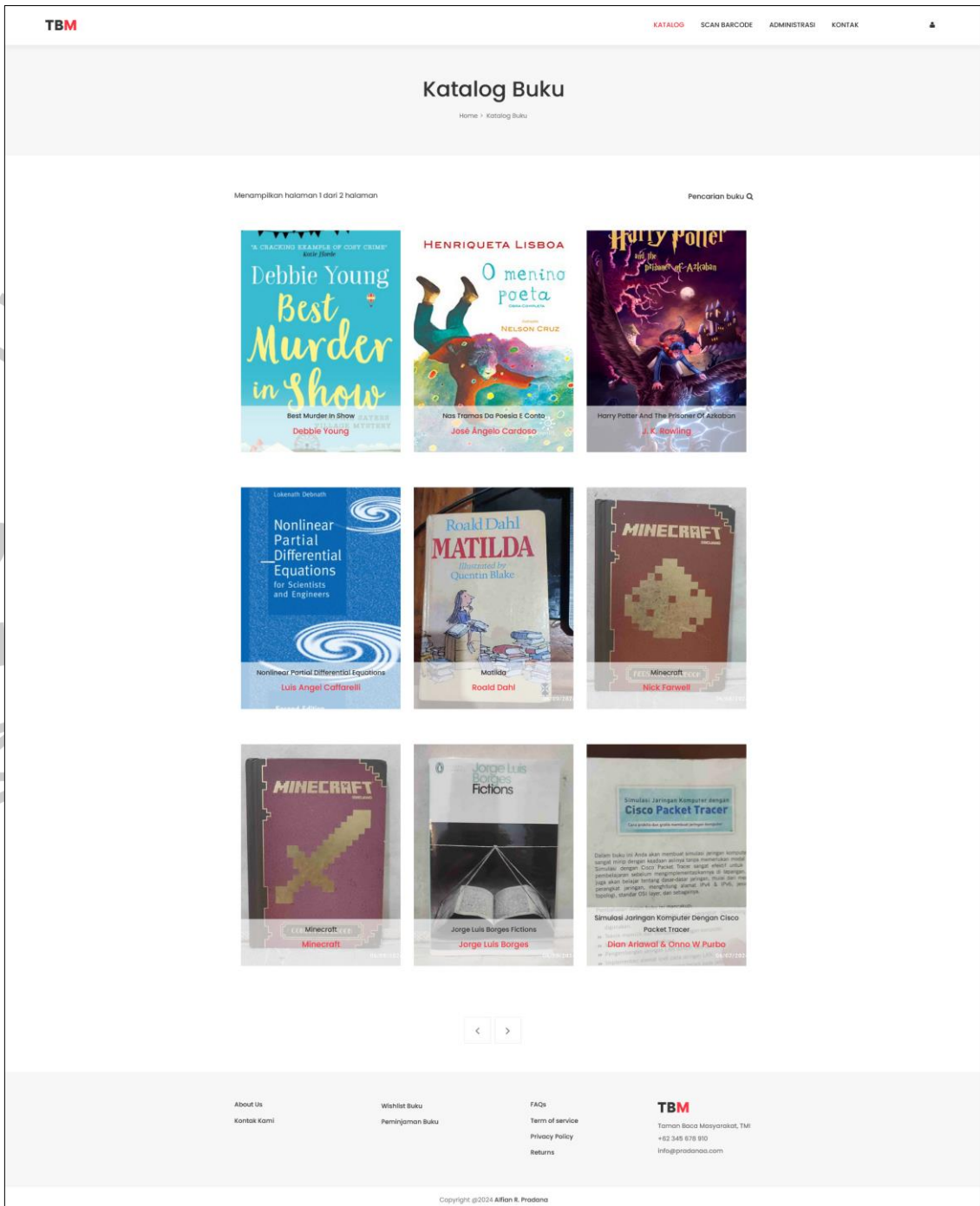
Pada Gambar 4.66 merupakan hasil implementasi *form ubah password* yang dimana member harus melakukan login terlebih dahulu.



Gambar 4. 66 Hasil Form Ubah Password

8) Member – Katalog Buku

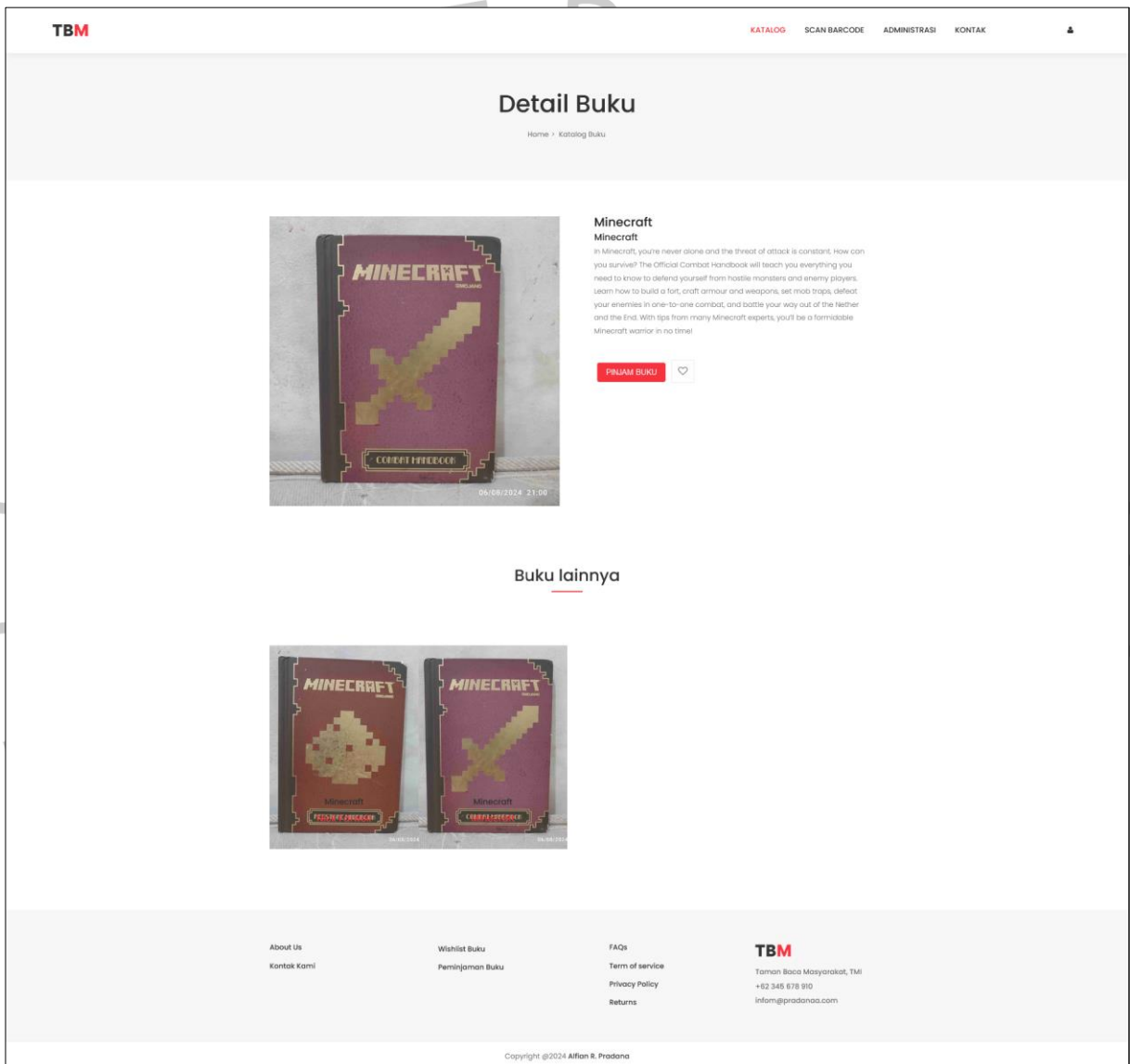
Pada Gambar 4.67 merupakan hasil implementasi daftar buku yang telah diinput oleh pengurus dan tersedia untuk dipinjam oleh member.



Gambar 4. 67 Hasil Katalog Buku

9) Member – Detail Buku

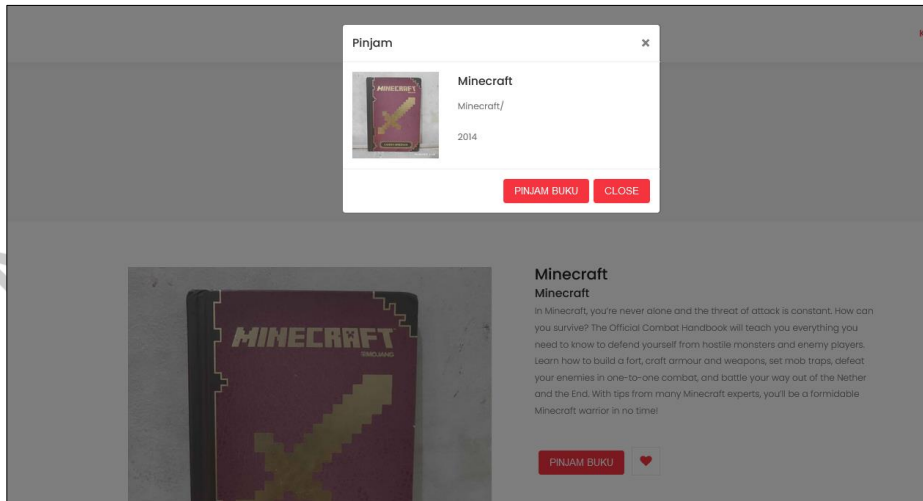
Pada Gambar 4.68 merupakan hasil implementasi detail buku yang halaman detail buku berdasarkan id-nya agar member bisa melakukan *wishlist* atau peminjaman terhadap buku.



Gambar 4. 68 Hasil Detail Buku

10) Member – Kotak Dialog Pinjam

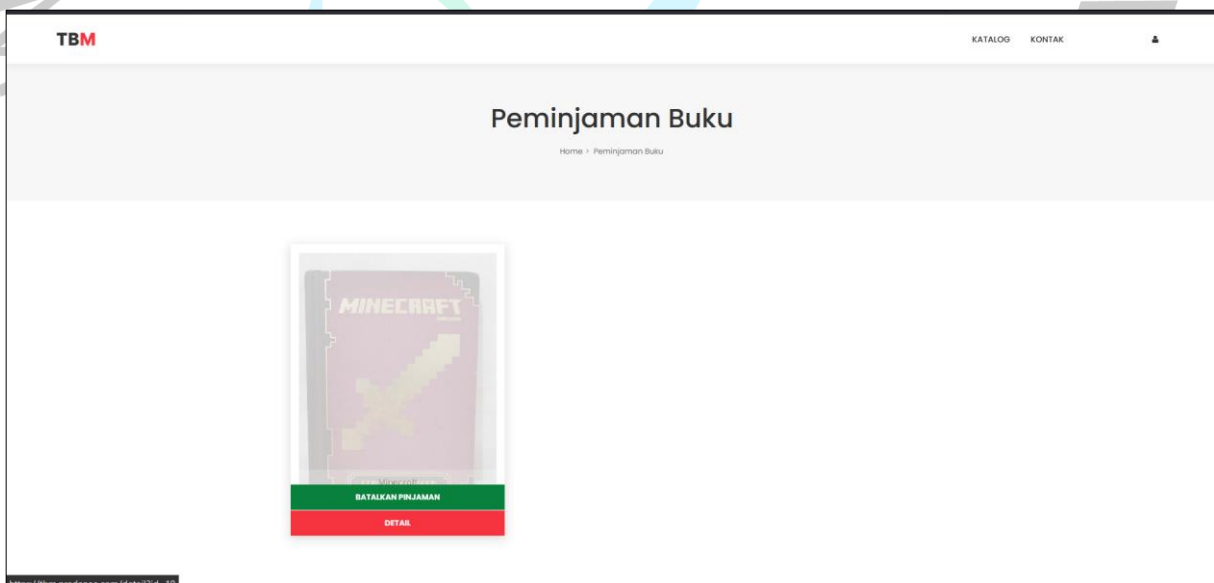
Pada Gambar 4.69 merupakan hasil implementasi kotak dialog konfirmasi untuk peminjaman buku di halaman detail buku.



Gambar 4. 69 Hasil Kotak Dialog Pinjam

11) Member – Daftar Pinjam

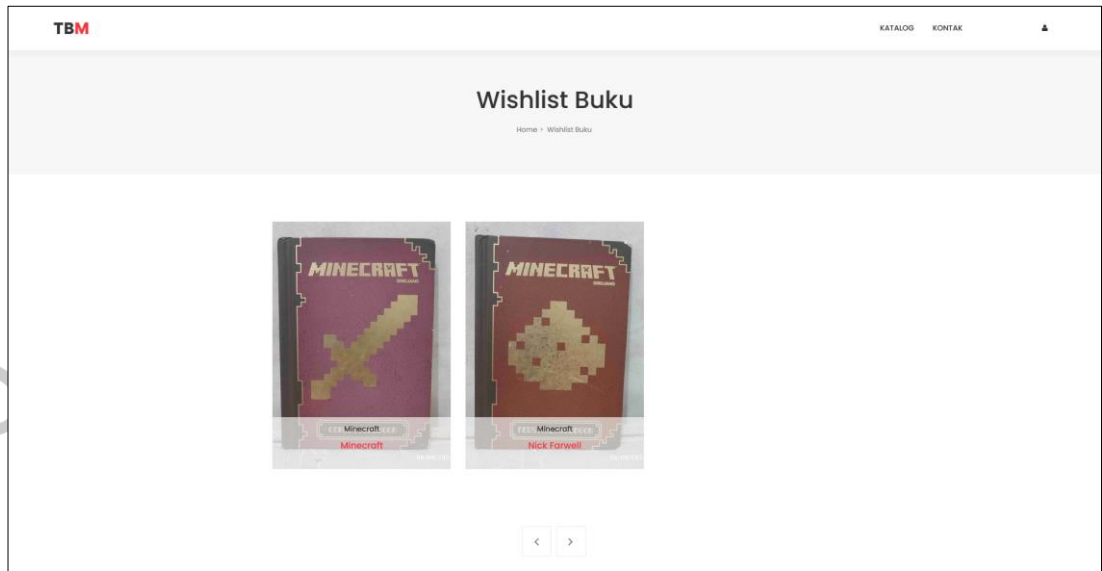
Pada Gambar 4.70 merupakan hasil implementasi halaman daftar buku yang dipinjam oleh member. Member dapat melakukan aksi untuk melihat detail atau melakukan pembatalan peminjaman.



Gambar 4. 70 Hasil Daftar Peminjaman Buku

12) Member – Wishlist

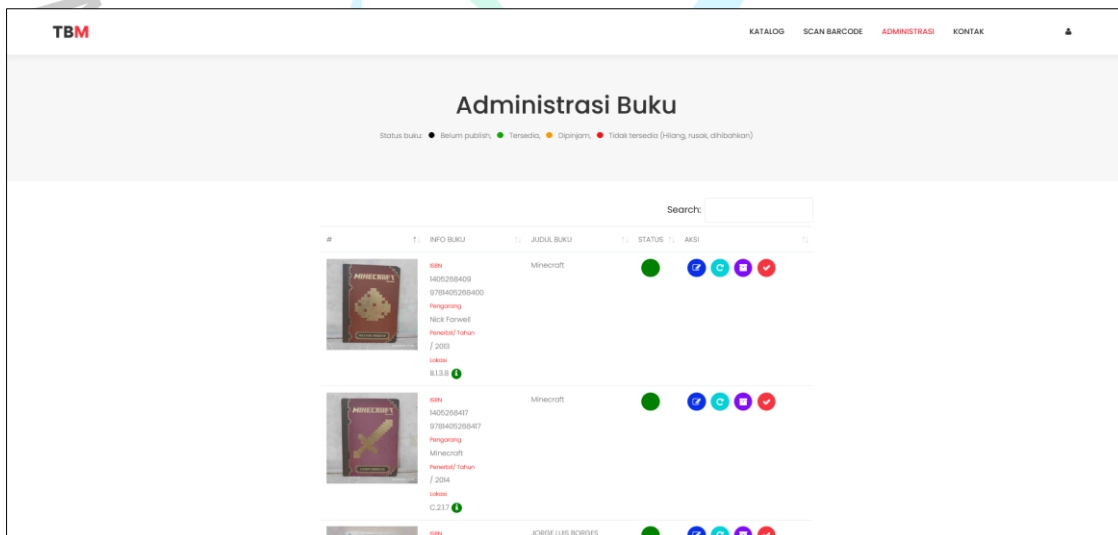
Pada Gambar 4.71 merupakan hasil implementasi daftar buku-buku yang telah difavoritkan atau ditandai oleh member



Gambar 4. 71 Hasil Daftar Wishlist

13) Administrasi Buku

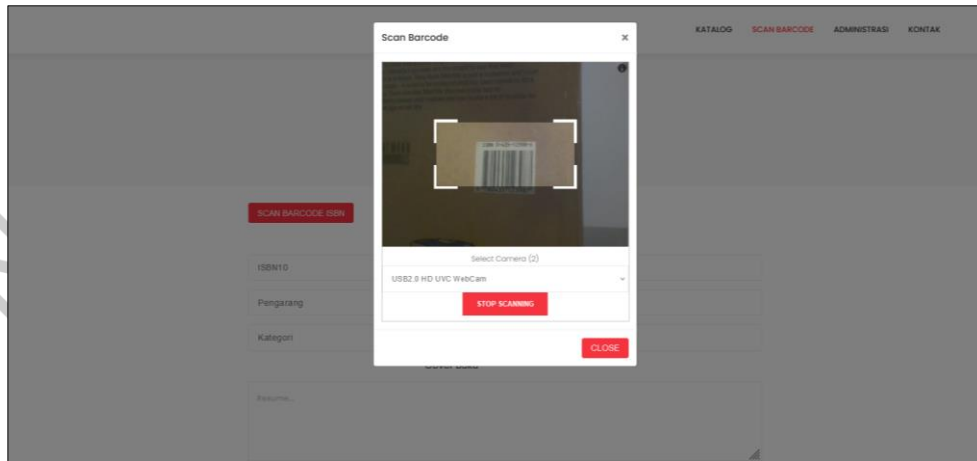
Pada Gambar 4.72 merupakan hasil implementasi halaman daftar tabel buku yang dikelola oleh pengurus.



Gambar 4. 72 Administrasi Buku

14) Administrasi Buku – Scan

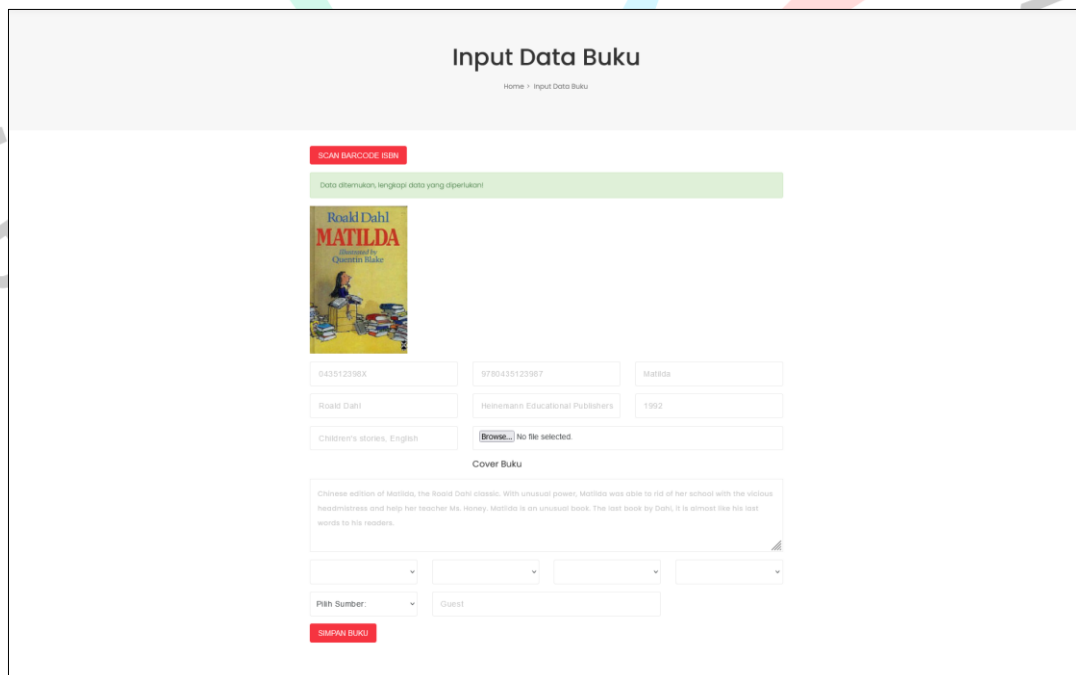
Pada Gambar 4.73 merupakan hasil implementasi kotak dialog *scan* buku oleh pengurus yang tersambung oleh API Google Books.



Gambar 4. 73 Hasil Scan Buku

15) Administrasi Buku – Scan Success

Pada Gambar 4.74 merupakan hasil halaman *input* data buku oleh pengurus jika buku berhasil ditemukan.



Gambar 4. 74 Hasil Scan Buku Sukses

16) Administrasi Buku – Halaman Add Buku

Pada Gambar 4.75 merupakan hasil halaman input data buku secara keseluruhan untuk mengisi Detail Buku (ISBN, Judul, Pengarang, Penerbit, Tahun, Kategori, Resume dan *Cover* Buku), Lokasi Buku (Lokasi, Lemari, Rak dan Baris), dan sumber donasi buku.

The screenshot shows the 'Input Data Buku' page on the TBM website. The page has a header with the TBM logo and navigation links: KATALOG, SCAN BARCODE, ADMINISTRASI, and KONTAK. The main heading is 'Input Data Buku' with a breadcrumb 'Home > Input Data Buku'. Below the heading is a 'SCAN BARCODE ISBN' button and a book icon. The form contains the following fields:

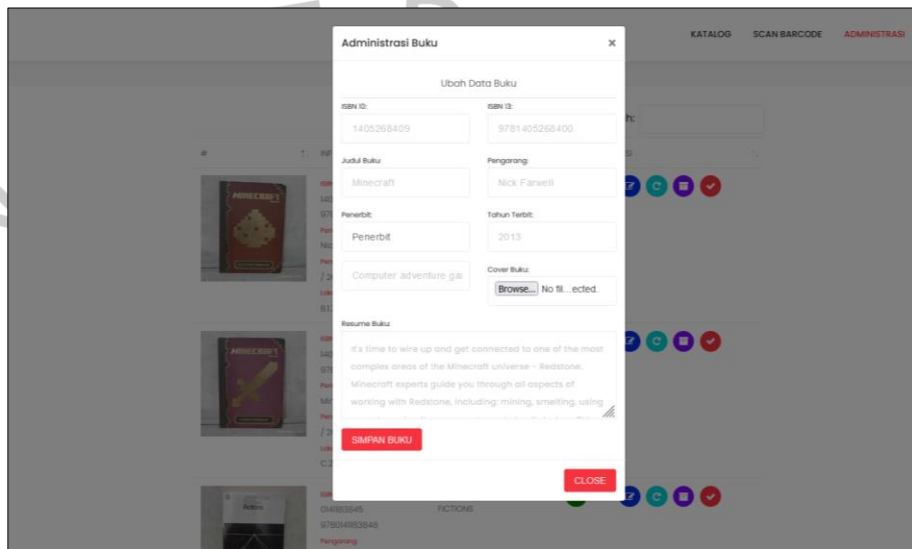
- ISBN10, ISBN13, Judul
- Pengarang, Penerbit, Tahun
- Kategori, a file upload button labeled 'Browse...' with the text 'No file selected.'
- A section for 'Cover Buku' with a text area for 'Resume...'
- Dropdown menus for 'Pilih Lokasi', 'Pilih Lemari', 'Pilih Rak', and 'Pilih Nomor'.
- A dropdown menu for 'Pilih Sumber' with 'Guest' selected.
- A red 'SIMPAN BUKU' button.

The footer contains links for 'About Us', 'Kontak Kami', 'Wishlist Buku', 'Peminjaman Buku', 'FAQs', 'Term of service', 'Privacy Policy', and 'Returns'. The TBM logo and contact information are also present: 'Taman Baca Masyarakat, TMI', '+62 345 678 910', and 'info@pradanaa.com'. A copyright notice at the bottom reads 'Copyright ©2024 Alifan R. Pradana'.

Gambar 4. 75 Hasil Add Buku

17) Administrasi Buku – Edit Detail

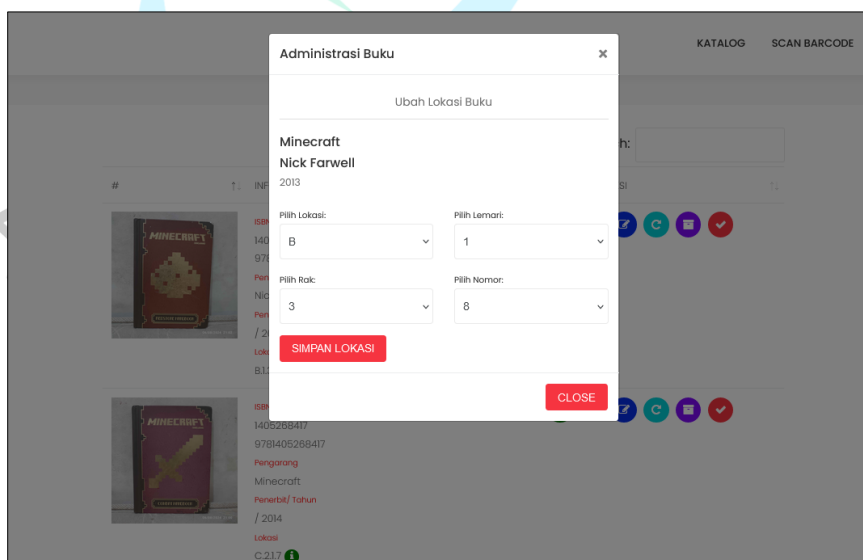
Pada Gambar 4.76 merupakan hasil implementasi kotak dialog untuk mengubah detail mengenai buku seperti ISBN, Judul, Pengarang, Penerbit, Tahun dan *button* unggah gambar *cover* buku terbaru.



Gambar 4. 76 Hasil Edit Detail Buku

18) Administrasi Buku – Edit Lokasi

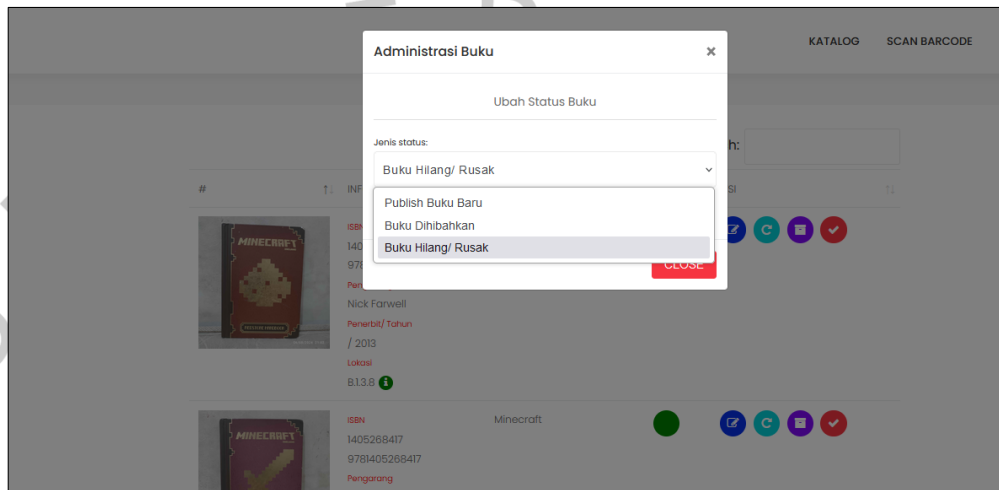
Pada Gambar 4.77 merupakan hasil implementasi kotak dialog untuk mengubah lokasi suatu buku berupa *dropdown* lokasi, lemari, rak dan baris.



Gambar 4. 77 Hasil Edit Lokasi Buku

19) Administrasi Buku – Edit Status

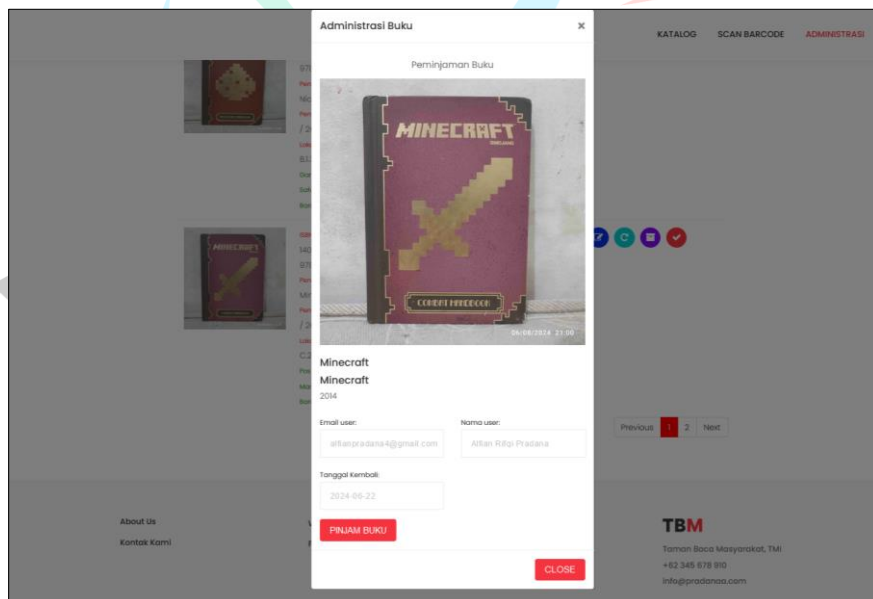
Pada Gambar 4.78 merupakan hasil implementasi kotak dialog untuk mengubah status suatu buku berupa *dropdown* status yang memiliki *value publish*, buku dihibahkan, dan buku hilang/rusak.



Gambar 4. 78 Hasil Edit Status Buku

20) Administrasi Buku – Pinjam

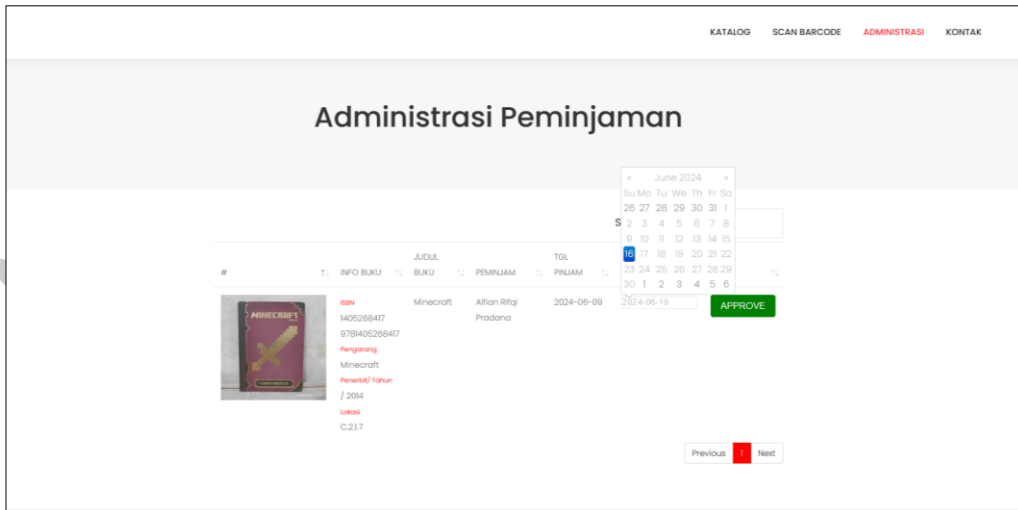
Pada Gambar 4.79 merupakan hasil implementasi kotak dialog untuk melakukan peminjaman buku dari sisi pengurus secara langsung.



Gambar 4. 79 Hasil Pinjam Buku

21) Administrasi Peminjaman – Daftar Peminjaman

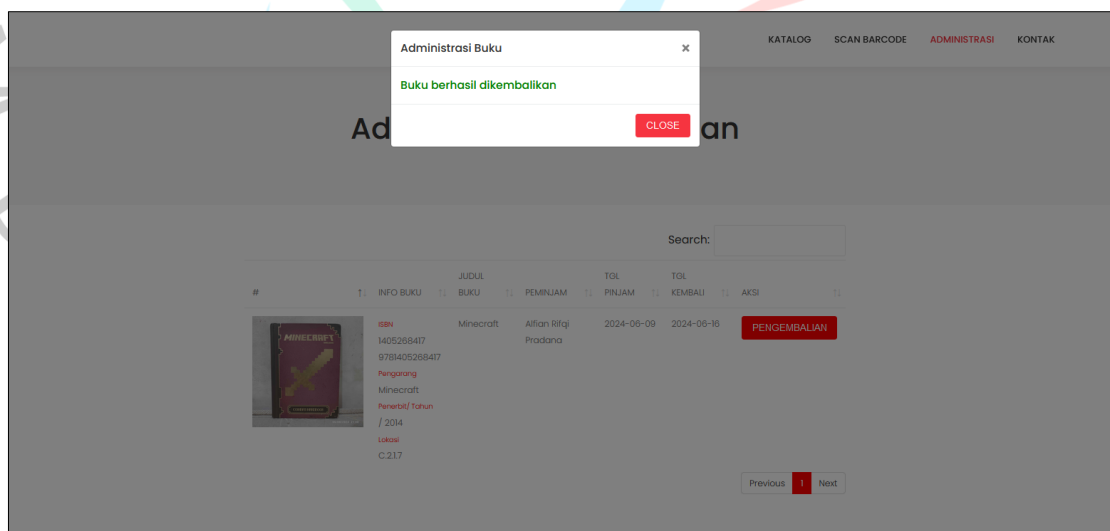
Pada Gambar 4.80 merupakan hasil implementasi halaman daftar peminjaman buku dari member yang dikelola oleh pengurus.



Gambar 4. 80 Hasil Konfirmasi Peminjaman Buku

22) Administrasi Peminjaman – Pengembalian Buku

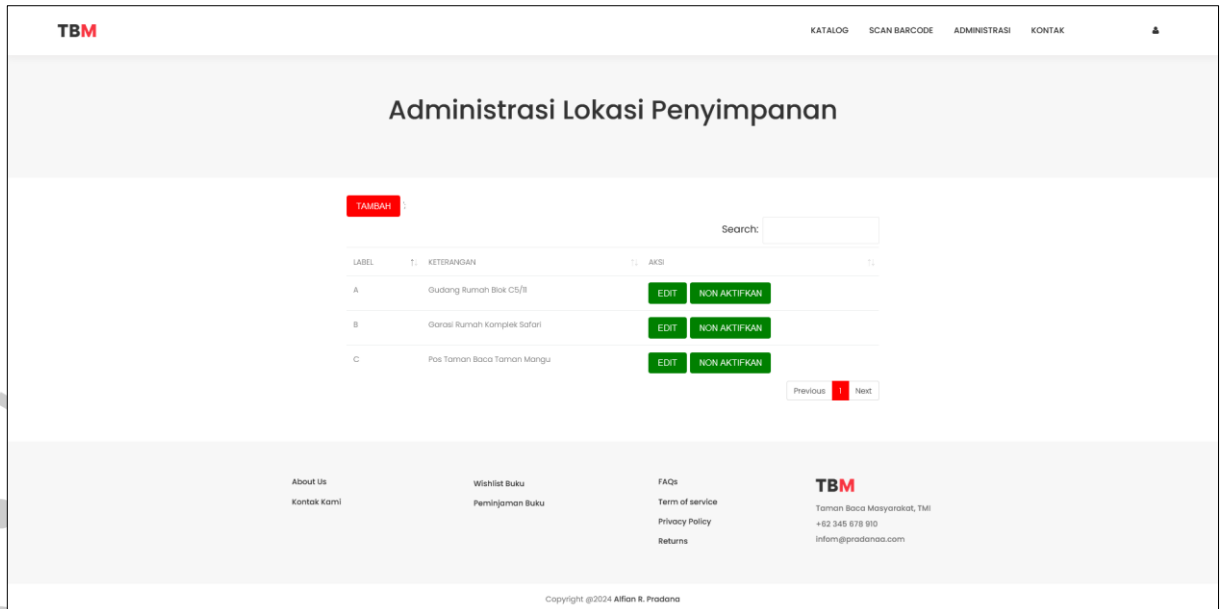
Pada Gambar 4.81 merupakan hasil implementasi *dialog box* jika pengurus mengklik button pengembalian buku saat member selesai membaca.



Gambar 4. 81 Hasil Pengembalian Buku

23) Administrasi Lokasi – Daftar Lokasi

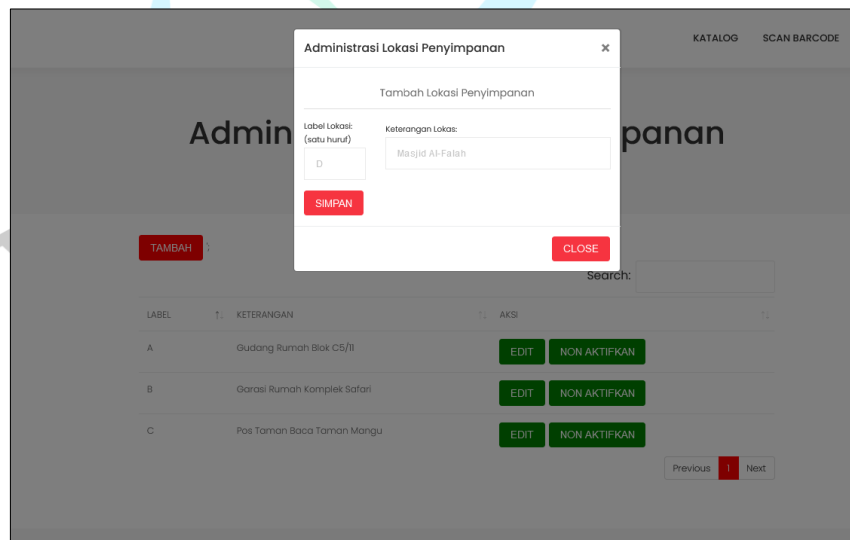
Pada Gambar 4.82 merupakan hasil implementasi halaman konfigurasi lokasi buku yang dikelola oleh pengurus.



Gambar 4. 82 Hasil Halaman Administrasi Lokasi

24) Administrasi Lokasi – Add

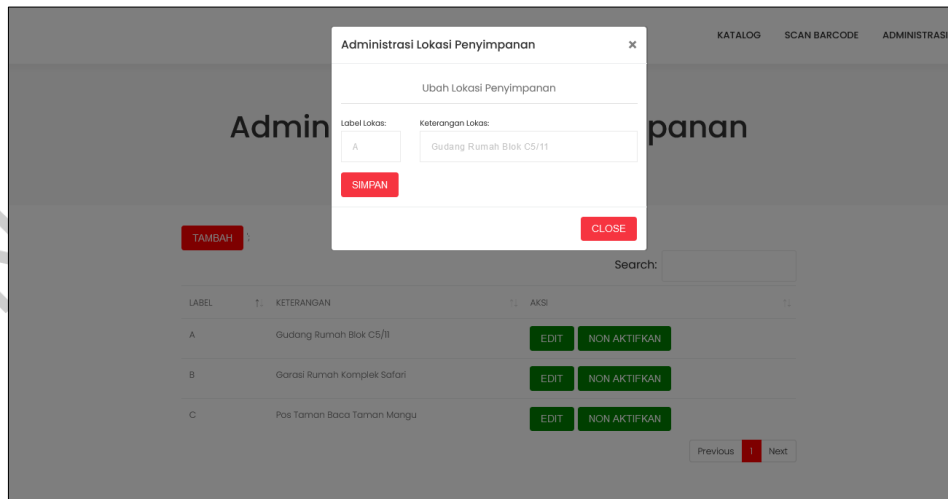
Pada Gambar 4.83 merupakan hasil implementasi kotak dialog *input* lokasi buku oleh pengurus.



Gambar 4. 83 Hasil Add Lokasi

25) Administrasi Lokasi – Edit

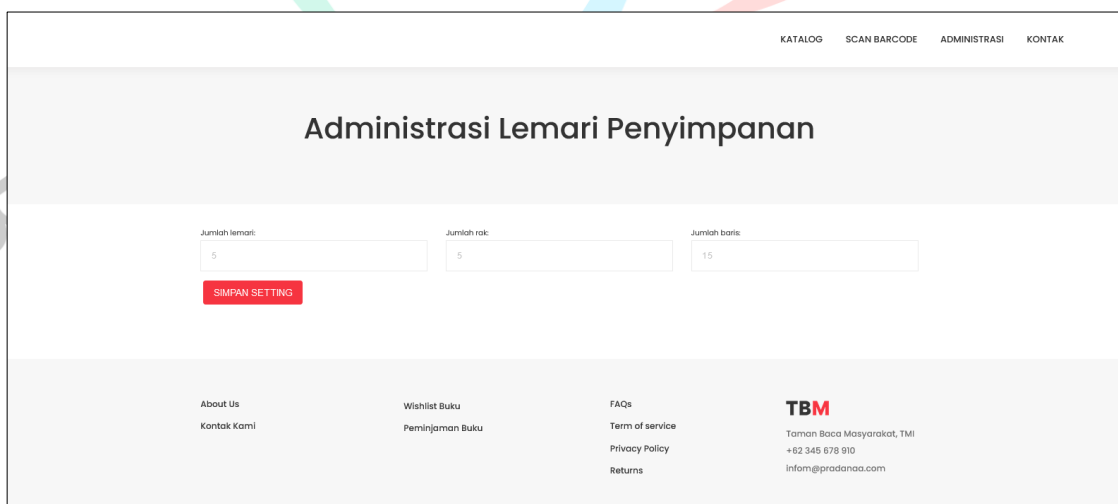
Pada Gambar 4.84 merupakan hasil implementasi kotak dialog *edit* lokasi buku oleh pengurus.



Gambar 4. 84 Hasil Ubah Lokasi

26) Administrasi Lemari

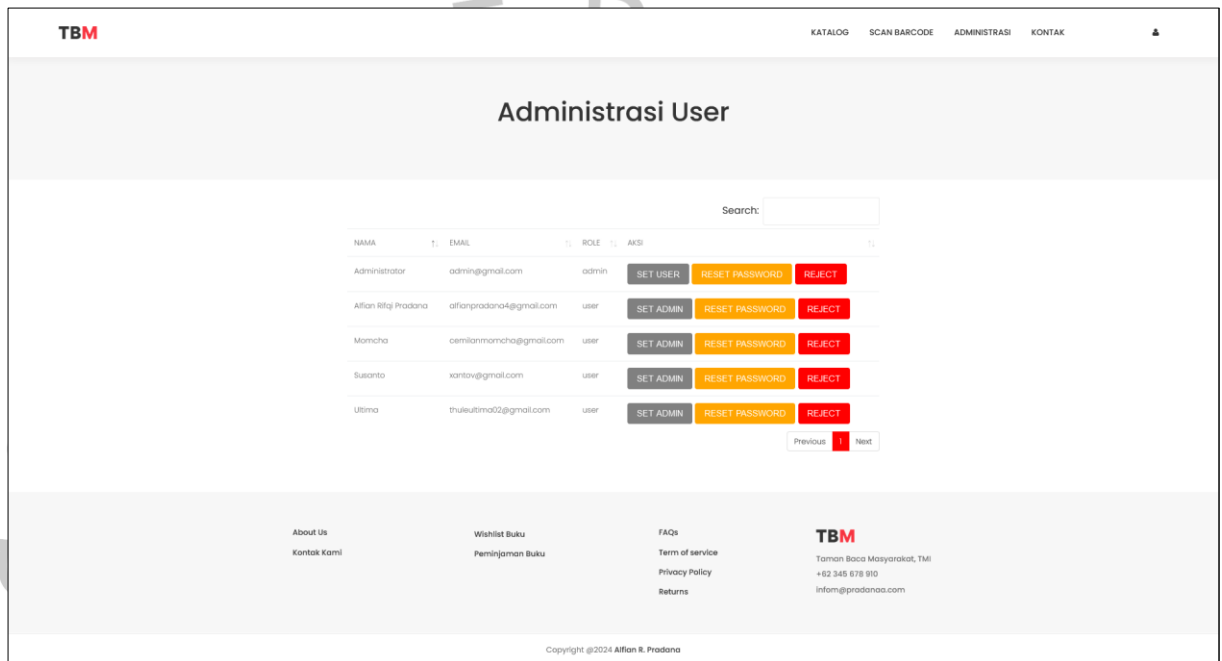
Pada Gambar 4.85 merupakan hasil implementasi halaman konfigurasi jumlah maksimal lemari, rak dan baris buku yang dikelola oleh pengurus.



Gambar 4. 85 Hasil Administrasi Lemari

27) Administrasi User

Pada Gambar 4.86 merupakan hasil implementasi halaman konfigurasi pengguna yang berada pada sistem taman baca agar bisa mengubah *role/level user*, *reset password*, dan mengaktifkan/menonaktifkan *user*.



Gambar 4. 86 Hasil Administrasi User

4.4.2 Pengujian Black Box Testing

Berikut adalah tabel uji *black box* bagi user member dan pengurus dilihat di Tabel 4.19 dan Tabel 4.20:

Tabel 4. 19 Uji *Black box* User Member

<i>No.</i>	<i>Test Case</i>	<i>Test Step</i>	<i>Expected Result</i>	<i>Result</i>	<i>Conclusion</i>
1	Login Member	1. User mengakses halaman login 2. User memasukkan <i>field</i> email dan <i>password</i> 3. User mengklik button login	Menampilkan pesan login sukses dan <i>redirect</i> ke halaman utama sesuai <i>role</i>	<i>Success</i>	<i>Valid</i>
2	Registrasi Member	1. User mengakses halaman daftar 2. User memasukkan <i>field</i> nama, <i>email</i> , <i>password</i> dan konfirmasi <i>password</i> 3. User mengklik button daftar	Menampilkan pesan pendaftaran berhasil harap cek <i>inbox</i> email	<i>Success</i>	<i>Valid</i>
3	Aktivasi Member	1. User mengecek <i>inbox</i> pada <i>email</i> yang terdaftar 2. User mengklik button aktivasi akun	Menampilkan pesan akun berhasil di aktivasi dan <i>redirect</i> ke halaman login	<i>Success</i>	<i>Valid</i>
4	Reset Password	1. User memasukkan <i>field</i> email untuk di-reset <i>password</i> -nya 2. User mengklik button kirim <i>link</i>	Mengirimkan link reset password pada email yang terkait	<i>Success</i>	<i>Valid</i>
5	Form Reset Password	1. User mengecek <i>inbox</i> pada email yang terdaftar 2. User mengklik button <i>reset password</i> 3. User memasukkan <i>field password baru</i> dan konfirmasi <i>password</i>	Menampilkan pesan <i>password</i> berhasil terganti dan <i>redirect</i> ke halaman login	<i>Success</i>	<i>Valid</i>
6	Mencari	1. User mengakses	Menyeleksi pencarian	<i>Success</i>	<i>Valid</i>

	Buku	halaman katalog buku 2. User memasukan kata kunci judul, kategori atau pengarang pada <i>field</i> pencarian 3. User mengklik <i>button</i> icon pencarian	berdasarkan field dan menampilkan buku yang dicari		
7	Melihat Detail Buku	1. User mengakses halaman katalog buku 2. User mengklik button detail buku	Menampilkan informasi buku berdasarkan id pada url	Success	Valid
8	Menambah Wishlist Buku	1. User mengakses halaman detail buku 2. User mengklik button icon wishlist buku	Menambahkan buku pada halaman <i>list wishlist</i>	Success	Valid
9	Menghapus Wishlist Buku	1. User mengakses halaman detail buku 2. User mengklik button icon wishlist buku	Menghapus buku pada halaman <i>list wishlist</i>	Success	Valid
10	Meminjam Buku	1. User mengakses halaman detail buku 2. User mengklik button pinjam buku	Menambahkan peminjaman buku pada halaman peminjaman dan kemudian akan di- <i>approve</i> oleh pengurus	Success	Valid
11	Membatalkan Peminjaman Buku	1. User mengakses halaman list peminjaman buku 2. User mengklik button batalkan peminjaman buku	Menghapus buku pada halaman peminjaman buku	Success	Valid

Tabel 4. 20 Uji *Black box* User Pengurus

No.	Test Case	Test Step	Expected Result	Result	Conclusion
1	Login Pengurus	1. User mengakses halaman login 2. User memasukan <i>field</i> email dan <i>password</i>	Menampilkan pesan login sukses dan <i>redirect</i> ke halaman	Success	Valid

		3. User mengklik button login	utama		
2	Konfirmasi Peminjaman	<p>1. User mengakses halaman administrasi peminjaman</p> <p>2. User mengklik <i>button</i> konfirmasi peminjaman pada salah satu row peminjaman buku</p> <p>3. User memilih tanggal pengembalian dari date picker dan klik button simpan</p>	Menampilkan pesan login sukses dan <i>redirect</i> ke halaman utama sesuai <i>role</i>	<i>Success</i>	<i>Valid</i>
3	Batalkan Peminjaman	<p>1. User mengakses halaman administrasi peminjaman</p> <p>2. User mengklik <i>button</i> konfirmasi peminjaman pada salah satu row peminjaman buku</p>	Menampilkan pesan login sukses dan <i>redirect</i> ke halaman utama sesuai <i>role</i>	<i>Success</i>	<i>Valid</i>
4	Scan Buku	<p>1. User mengakses halaman tambah buku</p> <p>2. User mengklik <i>button scan barcode</i> ISBN</p> <p>3. User mengklik <i>button</i> izinkan kamera</p> <p>4. User memilih kamera, dan memindai <i>barcode</i> di belakang buku.</p> <p>5. User lanjut mengisi lokasi dan lemari buku, lalu klik simpan.</p>	Menampilkan pesan buku berhasil ditemukan dan mengisi <i>field</i> data buku (judul, pengarang, penerbit, dll.) dan menyimpan buku di <i>database</i>	<i>Success</i>	<i>Valid</i>
5	Tambah Buku	<p>1. User mengakses halaman tambah buku</p> <p>2. User mengisi <i>field</i> secara manual data buku (judul, pengarang, penerbit, dll.) jika API tidak menemukan data buku maka.</p> <p>3. User memilih <i>dropdown</i> lokasi, lemari, rak dan baris pada buku.</p> <p>4. User mengklik <i>button</i> simpan.</p>	Menyimpan buku di <i>database</i> dan memberi pesan buku berhasil tersimpan.	<i>Success</i>	<i>Valid</i>
6	Edit Detail Buku	<p>1. User mengakses halaman administrasi buku</p>	Membarukan data buku di <i>database</i> dan memberi	<i>Success</i>	<i>Valid</i>

		<p>2. User mengklik button ubah detail buku pada salah satu row buku</p> <p>3. User mengubah field judul, pengarang, penerbit, dst.</p> <p>4. User mengklik <i>button</i> simpan.</p>	<p>pesan buku berhasil berubah.</p>		
7	Edit Penyimpanan Buku	<p>1. User mengakses halaman administrasi buku</p> <p>2. User mengklik button ubah lokasi pada salah satu row buku</p> <p>3. User mengubah dari dropdown lokasi, lemari, rak atau baris</p> <p>4. User mengklik <i>button</i> simpan.</p>	<p>Membarukan data buku di <i>database</i> dan memberi pesan buku berhasil berubah.</p>	Success	Valid
8	Edit Status Buku	<p>1. User mengakses halaman administrasi buku</p> <p>2. User mengubah dari <i>dropdown</i> status berdasarkan kondisi buku (hilang/rusak/didonasikan)</p> <p>3. User mengklik button ubah status pada salah satu row buku</p>	<p>Membarukan data buku di <i>database</i> dan memberi pesan buku berhasil berubah dan menarik buku dari katalog.</p>	Success	Valid
9	Tambah Lokasi	<p>1. User mengakses halaman administrasi lokasi</p> <p>2. User mengklik <i>button</i> tambah lokasi pada salah satu <i>row</i></p> <p>3. User mengisi <i>field</i> kode lokasi, nama lokasi dan keterangan lokasi.</p> <p>4. User mengklik button simpan</p>	<p>Menyimpan lokasi di <i>database</i> dan memberi pesan lokasi berhasil tersimpan</p>	Success	Valid
10	Ubah Lokasi	<p>1. User mengakses halaman administrasi lokasi</p> <p>2. User mengklik <i>button</i> tambah lokasi pada salah satu <i>row</i></p> <p>3. User mengubah <i>field</i> kode lokasi, nama lokasi</p>	<p>Membarukan data lokasi di <i>database</i> dan memberi pesan lokasi berhasil berubah.</p>	Success	Valid

		dan keterangan lokasi. 4. User mengklik <i>button</i> simpan			
11	Non-Aktifkan Lokasi	1. User mengakses halaman administrasi lokasi 2. User mengklik <i>button</i> non-aktifkan lokasi pada salah satu <i>row</i>	Membarukan data lokasi di <i>database</i> dan memberi pesan lokasi berhasil berubah.	<i>Success</i>	<i>Valid</i>
12	Ubah Jumlah Lemari	1. User mengakses halaman administrasi lemari 2. User mengubah <i>field</i> jumlah lemari, jumlah rak dan jumlah baris. 3. User mengklik <i>button</i> simpan	Membarukan data lemari di <i>database</i> dan memberi pesan lemari berhasil berubah.	<i>Success</i>	<i>Valid</i>
13	Ubah Role User	1. User mengakses halaman administrasi lokasi 2. User mengklik <i>button</i> set role to user/admin pada salah satu <i>row</i>	Membarukan data user di <i>database</i> dan user pesan lemari berhasil berubah.	<i>Success</i>	<i>Valid</i>
14	Reset Password User ke Default	1. User mengakses halaman administrasi lokasi 2. User mengklik <i>button</i> reset password pada salah satu <i>row</i>	Membarukan data user di <i>database</i> dan user pesan lemari berhasil berubah.	<i>Success</i>	<i>Valid</i>
15	Non-Aktifkan User	1. User mengakses halaman administrasi lokasi 2. User mengklik <i>button</i> non-aktifkan user pada salah satu <i>row</i>	Membarukan data user di <i>database</i> dan user pesan lemari berhasil berubah.	<i>Success</i>	<i>Valid</i>