



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

ANALISIS DAN PERANCANGAN

3.1 Metode Penelitian

Melalui penelitian ini akan dibuat sebuah aplikasi dengan menggunakan sistem RFID untuk membantu proses *stock opname* perpustakaan. Alat ini akan difungsikan untuk memeriksa kesesuaian buku pada rak dengan data buku yang tercatat pada *database*. Metode pembuatan *prototype model* dipilih untuk membuat dan menganalisis aplikasi yang tepat untuk membantu proses *stock opname* pada perpustakaan Universitas Multimedia Nusantara. Dengan menggunakan metode ini, diharapkan mendapatkan hasil yang lebih terfokus pada masalah utama yang akan diteliti. Tahapan dari perancangan aplikasi ini adalah sebagai berikut.

3.1.1 Pengumpulan Kebutuhan Pengguna

Pertama akan dilakukan pengumpulan kebutuhan dan permasalahan yang ada di perpustakaan mengenai proses *stock opname*. Pada tahap ini akan dikumpulkan seluruh kebutuhan sistem untuk proses *stock opname* buku perpustakaan dan akan dirancang garis besar dari aplikasi *stock opname* yang akan dibuat.

3.1.2 Pembuatan Desain Awal Aplikasi

Dari hasil pengumpulan kebutuhan dan masalah yang ada akan dirancang sebuah aplikasi *stock opname* awal yang memiliki fungsi dan kemampuan untuk menyelesaikan masalah pada proses *stock opname* yang telah dikumpulkan sebelumnya. Pada tahap ini akan ditetapkan fungsi-fungsi yang akan digunakan untuk pembuatan *prototype* aplikasi.

3.1.3 Pembuatan dan Pengujian *Prototype* Aplikasi

Pembuatan aplikasi tahap awal yang akan diuji coba dan dianalisis kekurangan dan kesalahan pada sistem yang telah dibuat. *Prototype* aplikasi dibuat sesuai dengan perencanaan awal dari kebutuhan dan masalah yang telah dirumuskan. Pembuatan *prototype* aplikasi ini bertujuan untuk mengumpulkan dan menguji solusi yang telah dibuat agar dapat dievaluasi dan diperbaiki sehingga aplikasi menjadi lebih baik dan sempurna. *Prototype* aplikasi akan diuji kesesuaiannya dengan kebutuhan dan masalah yang ada. Pengujian *prototype* dilakukan menggunakan simulasi percobaan *stock opname* buku pada perpustakaan Universitas Multimedia Nusantara.

3.1.4 Evaluasi *Prototype*

Setelah *prototype* aplikasi diuji coba, maka selanjutnya proses evaluasi dari masalah dan kekurangan yang ada pada aplikasi. Evaluasi ini untuk mengetahui solusi dari aplikasi yang diberikan sudah memenuhi kriteria dari kebutuhan *stock opname* buku perpustakaan. Jika *prototype* sudah sesuai dengan kebutuhan *stock opname* maka perancangan dan pembuatan aplikasi akan dilanjutkan ketahap selanjutnya. Jika *prototype* belum sesuai dengan kebutuhan pengguna maka perancangan dan pembuatan aplikasi akan diulang dari tahap awal kembali.

3.1.5 Mengkodekan Sistem

Setelah *prototype* aplikasi disetujui maka tahap selanjutnya adalah penyesuaian aplikasi dengan sistem dan sumber daya yang ada. Seluruh komponen dari aplikasi harus disesuaikan dengan sumber daya yang dimiliki pengguna

sehingga penulisan dan pembuatan program harus sesuai dengan format yang diminta pengguna.

3.1.6 Menguji Aplikasi Akhir

Setelah aplikasi sudah menjadi suatu perangkat yang lengkap, aplikasi harus dites sebelum digunakan.

3.1.7 Evaluasi Akhir

Pengguna akan mengevaluasi aplikasi yang sudah jadi dan akan melihat kesesuaian aplikasi dengan yang diharapkan. Jika seluruh kriteria sudah sesuai, maka tahap terakhir dapat dilakukan, namun jika belum maka perlu mengulangi dari tahap mengkodekan sistem.

3.1.8 Implementasi

Aplikasi yang telah diuji dan diterima pengguna siap untuk digunakan dan diimplementasikan di perpustakaan.

3.2 Analisis Masalah

Perpustakaan Universitas Multimedia Nusantara sudah menggunakan teknologi RFID pada sebagian proses operasionalnya. Namun untuk proses *stock opname* buku perpustakaan masih menggunakan cara manual, yaitu dengan melihat satu per satu buku pada rak buku dan mencatatkan hasil pemeriksaannya pada komputer ataupun *laptop*. Sebelumnya, petugas *stock opname* sudah diberikan *database* buku perpustakaan untuk membantu proses *stock opname*. Cara tersebut memiliki masalah yaitu waktu pemeriksaan yang cukup lama dikarenakan jumlah buku yang semakin banyak. Karena pemeriksaan yang lama tersebut maka proses

stock opname hanya dapat dilakukan ketika perpustakaan tutup. Permasalahan lainnya ada pada keakuratan pencatatan. Meskipun jarang terjadi kesalahan namun diperlukan ketelitian pencatatan oleh petugas *stock opname*.

Lalu permasalahan lainnya ada pada perangkat RFID *reader* yang seharusnya dikhususkan untuk proses peminjaman buku, dapat diubah penggunaannya untuk proses *stock opname*.

Dilihat dari isi RFID *tag* yang digunakan pada perpustakaan Universitas Multimedia Nusantara, *tag* tersebut menyimpan nomor ID buku (nomor induk buku) dan status “*check-in*” dan “*check-out*” pada *memory tag*. Untuk proses *stock opname* menggunakan RFID data ID buku pada *tag* dapat dijadikan acuan identifikasi yang akan diperiksa dengan data pada *database* perpustakaan. Sedangkan untuk status buku pada *tag* untuk saat ini tidak perlu digunakan untuk aplikasi *stock opname*.

Dengan memanfaatkan teknologi RFID yang sebelumnya digunakan untuk proses peminjaman dan keamanan buku, akan diterapkan untuk proses *stock opname*. Dilihat dari kemampuan RFID yang dapat membaca/mengidentifikasi *tag* secara cepat dan akurat (karena proses berjalan secara *device-to-device*, komunikasi antar *tag* dengan *reader*), RFID dapat mempermudah dan mempercepat proses *stock opname*.

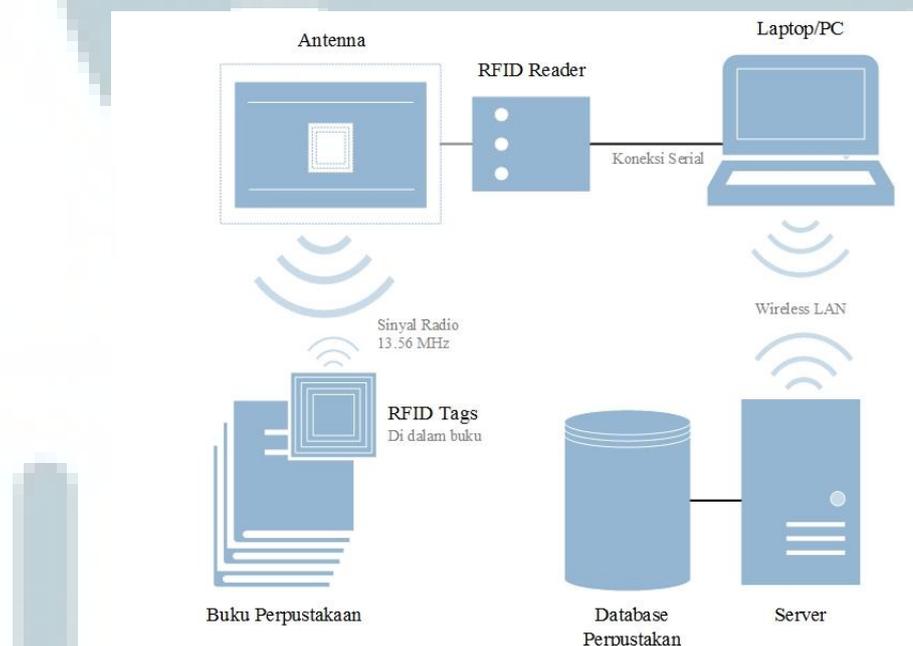
Untuk mengimplementasikan RFID pada proses *stock opname* diperlukan sebuah aplikasi untuk menjembatani antara data pada RFID dengan data yang terdapat pada *database*. Aplikasi tersebut harus dapat memeriksa buku yang ada pada rak dengan daftar buku pada *database*. Lalu aplikasi harus dapat menghasilkan

output langsung secara digital sehingga tidak diperlukan lagi proses *entry* data ke dalam komputer.

3.3 Perancangan dan Pembuatan Aplikasi

3.3.1 Spesifikasi Perangkat

Dalam pembuatan aplikasi Digital Library Stock Opname diperlukan perangkat pendukung untuk membuat aplikasi dan melakukan percobaan. Perlengkapan yang digunakan dalam penelitian ini yaitu, perangkat keras dan perangkat lunak. Gambaran topologi perangkat yang digunakan pada aplikasi Digital Library Stock Opname dapat dilihat pada Gambar 3.1.



Gambar 3.1 Topologi Aplikasi Digital Library Stock Opname

Berikut adalah penjelasan dari spesifikasi perangkat yang digunakan peneliti selama proses perancangan alat dan percobaan.

1. *RFID Reader*

RFID reader merupakan objek utama yang akan diteliti pada penelitian ini. Pada penelitian ini akan menggunakan PROMAG™ AMP600 *Advanced Smart Label Reader/Writer*. Frekuensi yang digunakan adalah *High frequency* sebesar 13.56 MHz (*datasheet* alat terdapat dalam Lampiran 4). *RFID reader* ini akan digunakan untuk membaca 3M™ ISO RFID 50x50mm *Tags*, yang terdapat di dalam buku perpustakaan Universitas Multimedia Nusantara. *Tag* tersebut sudah terlebih dulu dikonfigurasi.

2. *Komputer/ Laptop*

Peneliti menggunakan *personal computer* (PC) untuk merancang aplikasi dan melakukan percobaan. Aplikasi ini dirancang agar dapat digunakan selain untuk PC dan *laptop*, dapat digunakan juga pada perangkat *tablet* yang memiliki sistem operasi Windows 8. Spesifikasi komputer terdapat pada Tabel 3.1.

Tabel 3.1 Spesifikasi Komputer

<i>Processor</i>	Intel® Core™ 2 CPU 6300 @ 1.86 GHz
<i>Memory</i>	3.25 GB
<i>Operating System</i>	Microsoft Windows 8 Pro 32-bit (6.2, build 9200)
<i>Harddisk</i>	512 GB

3. *Connector Serial-to-USB*

Penggunaan *Connector Serial-to-USB* digunakan untuk menghubungkan perangkat RFID *reader* dengan komputer. *Connector Serial-to-USB* yang digunakan adalah USB-to-Serial DB9 RS232 *Adapter* (Prolific chipset).

4. *Software Programming*

Perancangan aplikasi Digital Library Stock Opname menggunakan bahasa pemrograman Visual C# dan menggunakan *software* Microsoft Visual Studio Express 2013 for Windows Desktop.

5. *Database*

Database yang digunakan untuk percobaan aplikasi Digital Library Stock Opname adalah dengan menggunakan MySQL. *Software* penghubung antara *database* dengan aplikasi (*database connector*) menggunakan MySQL Connector .net versi 6.6.6. Untuk pembuatan *database* peneliti menggunakan *software* XAMPP for Windows versi 1.8.3.

Database yang digunakan adalah *database* perpustakaan Universitas Multimedia Nusantara. Meskipun *Database* Universitas Multimedia Nusantara menggunakan Oracle dan pada pengujian aplikasi ini menggunakan MySQL, pemindahan penggunaan *database* dapat dilakukan dengan mudah dan cepat karena struktur datanya sudah sama. Struktur *database* perpustakaan Universitas Multimedia Nusantara yang digunakan pada aplikasi ini adalah sebagai berikut.

a. Tabel Judul

Berisi Informasi judul-judul dari buku yang ada di perpustakaan. Tabel judul memiliki ID_JUDUL yang merupakan penghubung antara judul buku dengan

nomor Induk buku. Buku yang sama memiliki informasi buku dan ID_JUDUL yang sama. Tabel judul juga akan menghubungkan tabel pengarang dan tabel t_ddc. Data yang digunakan aplikasi pada tabel ini adalah dari kolom JUDUL dan THN_TERBIT.

b. Tabel Koleksi

Berisi nomor induk tiap buku perpustakaan. Nomor induk merupakan nomor unik dari setiap buku (tidak ada buku yang memiliki nomor induk yang sama). Nomor induk tersebut juga tersimpan pada *user data block* pada RFID tag. Tabel koleksi akan menghubungkan dengan tabel pinjam, rusak dan hilang. Data yang digunakan aplikasi pada tabel ini adalah dari kolom NO_INDUK, data tersebut yang akan dicocokkan dengan data yang tersimpan pada RFID tag.

c. Tabel Pengarang

Berisi nama pengarang yang akan dihubungkan dengan ID_JUDUL. Data yang digunakan aplikasi pada tabel ini adalah dari kolom NAMA_PENGARANG.

d. Tabel t_ddc

Berisi nomor DDC setiap buku perpustakaan. ID_DDC menghubungkan tabel judul dengan tabel t_DDC. Data yang digunakan aplikasi pada tabel ini adalah dari kolom DDC.

e. Tabel Pinjam

Berisi data buku-buku yang dipinjam. Dengan mengetahui isi tabel ini, dapat dibuat *table view* yang berisi buku-buku yang sedang tidak dipinjam.

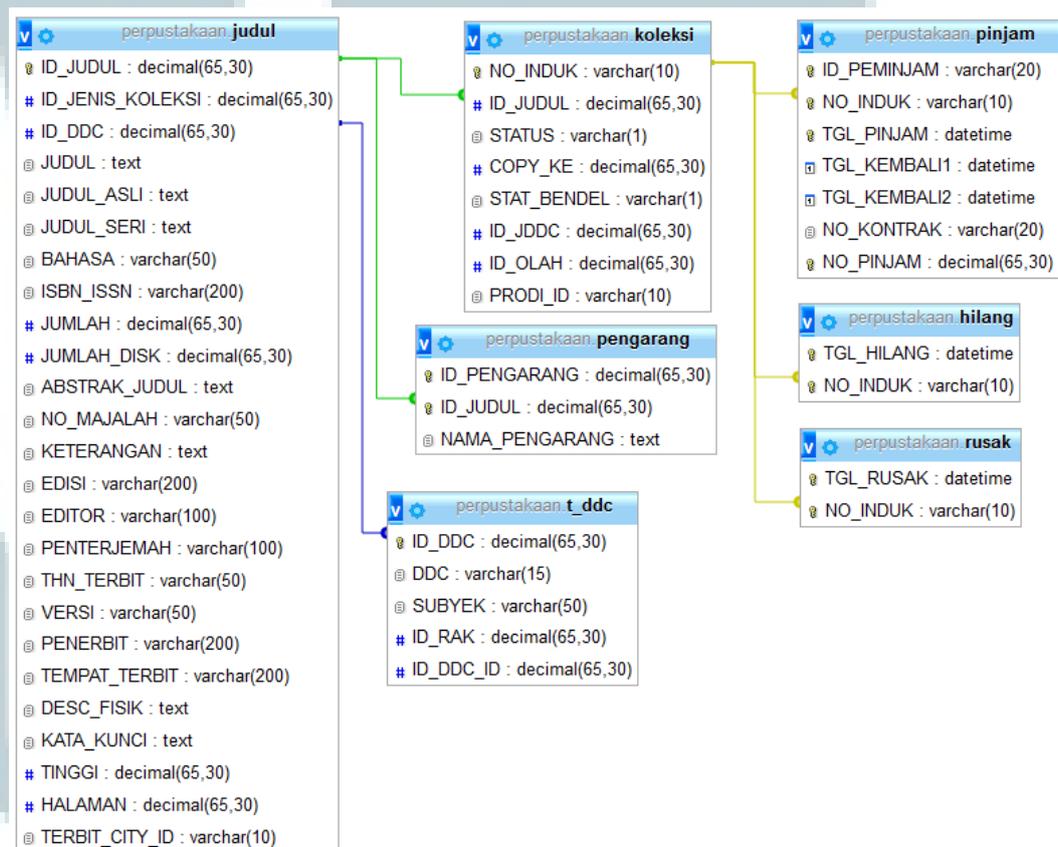
f. Tabel Hilang

Berisi data buku-buku yang hilang. Dengan mengetahui isi tabel ini, dapat dibuat *table view* yang berisi buku-buku yang tidak hilang.

g. Tabel Rusak

Berisi data buku-buku yang rusak. Dengan mengetahui isi tabel ini, dapat dibuat *table view* yang berisi buku-buku yang tidak rusak.

Relasi dari tabel-tabel *database* perpustakaan terdapat pada Gambar 3.2, sebagai berikut.



Gambar 3.2 Struktur Relasi *Database* Perpustakaan

Dari seluruh tabel pada *database* perpustakaan (Gambar 3.2) akan dibuat tabel *view* yang diberi nama *stockopname* dan akan digunakan pada aplikasi. Tabel *stockopname* merupakan *tabel view*, hasil dari *query* yang digunakan untuk menampilkan beberapa kolom pada tabel-tabel yang diinginkan. Tabel *stockopname* dibuat dengan menghilangkan data-data buku yang tidak diperlukan dalam proses *stock opname*. Tabel *stockopname* hanya berisi data buku yang tidak dipinjam, tidak rusak maupun yang tidak hilang dan hanya berisi data-data yang dibutuhkan pada proses *stock opname* (nomor induk, nomor DDC, judul buku, nama pengarang dan tahun terbit).

Data-data pada tabel *stockopname* berasal dari beberapa tabel, yaitu kolom NO_INDUK dari tabel koleksi, DDC dari tabel t_ddc, kolom JUDUL dan THN_TERBIT dari tabel judul dan kolom NAMA_PENGARANG dari tabel pengarang. Struktur tabel *stockopname* adalah sebagai berikut (Tabel 3.2).

Tabel 3.2 Struktur Tabel *stockopname* pada *Database* Perpustakaan

Column	Type	Null	Default
NO_INDUK	varchar(10)	No	
DDC	varchar(15)	No	
JUDUL	text	No	
NAMA_PENGARANG	text	Yes	NULL
THN_PENERBIT	varchar(50)	Yes	NULL

3.2.2 Perancangan Awal Aplikasi

Perancangan dan pembuatan aplikasi Digital Library Stock Opname dimulai dari menentukan fungsi dan rancangan dasar dari aplikasi, perancangan program dasar, implementasi dengan *RFID reader*, implementasi dengan *database* dan pengujian alat. Berikut penjelasan tahapan perancangan dan pengembangan aplikasi Digital Library Stock Opname.

1. Perancangan Dasar Aplikasi

Pada tahap awal perancangan, peneliti menentukan fungsi-fungsi dari aplikasi Digital Library Stock Opname. Fungsi utama dari aplikasi ini adalah untuk pemeriksaan buku (*stock opname*), selain itu ditambahkan fungsi lainnya seperti pencarian buku dan pemeriksaan posisi dan urutan buku pada rak perpustakaan.

2. Pembuatan Program Dasar

Aplikasi Digital Library Stock Opname terdiri dari dua *form*, yaitu *main form* dan *form settings*. Pada proses awal aplikasi dijalankan aplikasi akan menampilkan *main form* yang berisi tombol-tombol pilihan fungsi aplikasi. Selain itu *main form* berfungsi untuk menampilkan hasil informasi dari aplikasi. Informasi buku pada *main form* akan ditampilkan dalam bentuk tabel sedangkan informasi mengenai aplikasi akan ditampilkan pada *statusbar*. Untuk *form settings* menyediakan pilihan pengaturan *database*. Berikut adalah sketsa antarmuka dari aplikasi yang akan dibuat.

a. *Main Form*

Merupakan tampilan awal dari aplikasi ini. Seluruh kegiatan dan proses aplikasi berjalan pada *main form*. Tahap awal memulai aplikasi adalah

menghubungkan aplikasi dengan perangkat RFID sehingga dibutuhkan fungsi koneksi dengan perangkat RFID. *Main form* memiliki fungsi untuk memeriksa buku, mengurutkan buku dan mencari buku yang telah di-*scan*. Untuk menjalankan fungsi pemeriksaan buku maka dibutuhkan tombol untuk memulai pemeriksaan buku (*scan button*). Hasil dari pemeriksaan buku akan ditampilkan pada tabel. Tabel tersebut berisi informasi buku dan hasil pemeriksaan posisi buku. Untuk menyimpan data hasil *stock opname* dibutuhkan fungsi *save*. Jika terjadi kegagalan koneksi dengan *database* maka dibutuhkan menu pengaturan *database*.

Dari beberapa analisis di atas maka sketsa antarmuka pada *main form* dapat dilihat pada Gambar 3.3.

Scan		
Save		
Settings		
	Informasi Buku	Hasil Pemeriksaan Posisi Buku
Exit		

Gambar 3.3 Sketsa Tampilan *Main Form*

b. *Form Settings*

Pembuatan *form settings* bertujuan untuk pengaturan aplikasi. Masalah yang sering terjadi adalah pengaturan *database*. Pada *form settings* akan disediakan *textbox* untuk memasukkan nama *server*, nama *database*, nama *table*, *username* dan *password*. Sehingga sketsa antarmuka pada *form settings* dapat dilihat pada Gambar 3.4.

Input Nama Server
Input Nama Database
Input Username
Input Password
Input Nama Tabel
Check
Save

Gambar 3.4 Sketsa Tampilan *Form Settings*

3. Implementasi dengan *RFID Reader*

Sebelum terhubung dengan *RFID reader*, komputer harus sudah ter-*install driver* PROMAG™ AMP600 *Advanced Smart Label Reader/Writer*. Pada *driver* sudah diberikan fungsi untuk memperoleh data dari hasil pembacaan *RFID*. Proses awal sebelum aplikasi dijalankan adalah pemeriksaan perangkat *RFID*. Aplikasi akan memeriksa koneksi aplikasi dengan perangkat *RFID*, jika sudah terhubung aplikasi dapat dijalankan namun jika tidak aplikasi tidak dapat digunakan.

Pada tahap ini aplikasi sudah dapat membaca nomor induk buku pada *tag* (*ID buku*). *ID buku* ini nantinya akan digunakan sebagai acuan identitas buku yang akan dicocokkan dengan data pada *database*.

4. Implementasi dengan *Database*

Setelah alat telah dapat membaca *ID buku*, data tersebut akan dicocokkan dengan data pada *database* yang sebelumnya telah disimpan di dalam *table* aplikasi. Aplikasi akan membaca urutan proses pembacaan buku, kemudian urutan tersebut yang akan dipakai untuk menentukan posisi buku. Urutan buku secara tidak

langsung sudah terdapat pada *database* yang berbentuk nomor DDC sehingga penempatan dan posisi buku pada rak dapat disusun sesuai nomor DDC.

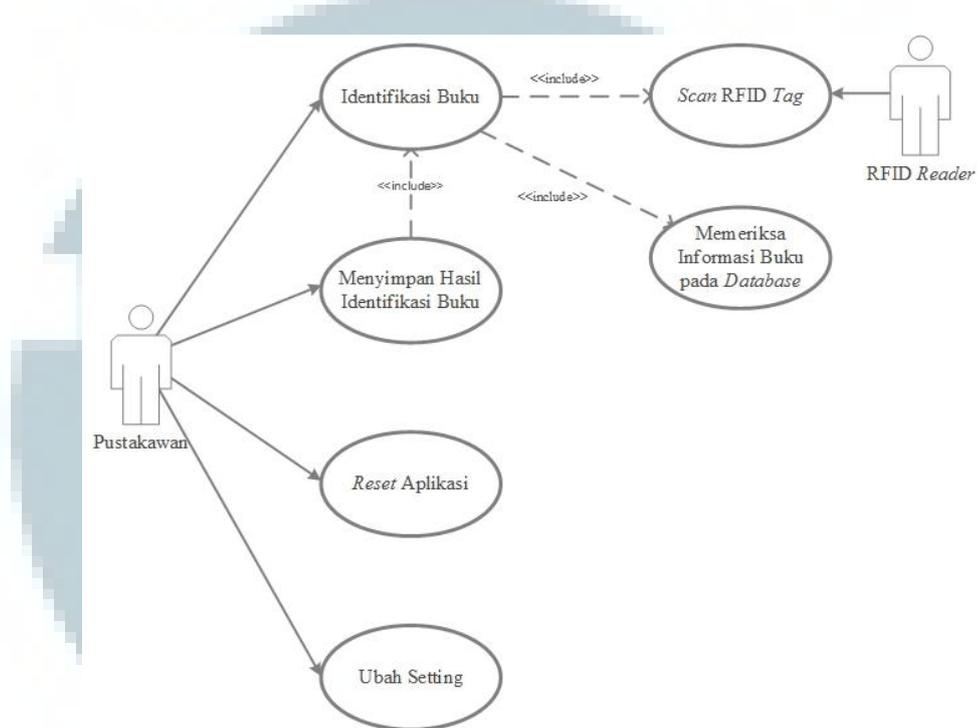
5. Pengujian Alat

Untuk mengetahui kinerja aplikasi, maka akan dilakukan pengujian. Pengujian dilakukan dengan mengambil sampel buku yang disusun seperti pada rak buku perpustakaan, lalu aplikasi mulai melakukan proses *stock opname*. Yang akan diuji adalah kemampuan aplikasi untuk dapat mengidentifikasi seluruh buku yang disediakan dengan benar dan memeriksa posisi dan urutan buku. Proses pengujian akan dilakukan beberapa kali dengan susunan buku yang berbeda.

The image shows a large, light blue watermark of the UMN logo. It consists of a circular emblem containing a stylized figure, with the letters 'UMMN' written in a bold, sans-serif font directly below it.

3.2.3 Cara Kerja Aplikasi

Cara kerja dari aplikasi Digital Library Stock Opname akan dimodelkan dalam *use case diagram* pada Gambar 3.5.



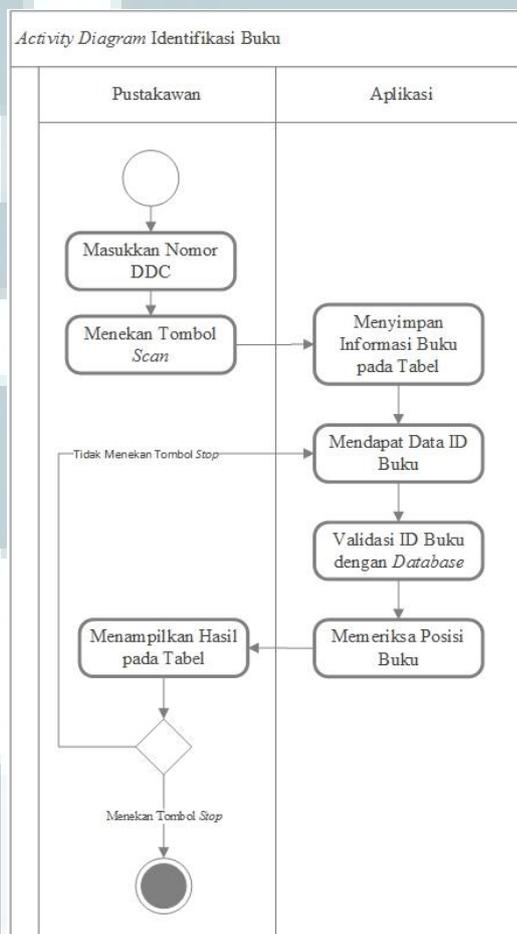
Gambar 3.5 Use Case Diagram

Dengan menggunakan aplikasi Digital Library Stock Opname, petugas *stock opname* dapat mengidentifikasi buku menggunakan RFID lalu hasilnya tersebut dapat disimpan dalam komputer. Aplikasi dapat diatur ulang dan petugas *stock opname* dapat mengubah pengaturan aplikasi secara langsung. Proses identifikasi tersebut membutuhkan hasil pembacaan data *tag* dari *RFID reader* dan data tersebut akan diolah dan diperiksa dengan informasi pada *database*.

Proses aplikasi Digital Library Stock Opname terbagi menjadi beberapa bagian dalam *activity diagram*. Berikut pembagian dan penjelasan lengkap dari cara kerja aplikasi Digital Library Stock Opname.

1. Bagian Identifikasi Buku

Bagian identifikasi buku, digambarkan pada Gambar 3.6 *Activity Diagram Identifikasi Buku*.



Gambar 3.6 *Activity Diagram Identifikasi Buku*

Bagian identifikasi buku dimulai dari memasukkan nomor DDC yang nantinya akan digunakan sebagai acuan posisi buku (Misalkan pengguna

memasukkan nomor DDC 004 maka buku akan diposisikan memiliki nomor DDC yang diawali dengan 004). Selanjutnya dengan menekan tombol *scan*, proses identifikasi akan dimulai. Aplikasi akan melakukan *query* data buku ke *database* lalu hasilnya akan disimpan ke dalam tabel aplikasi. Pada kondisi tersebut seluruh buku masih dianggap hilang/*missing* (karena belum ada *tag* yang teridentifikasi). Lalu saat aplikasi menerima data dari RFID *reader* data akan diolah pada bagian pemrosesan data *tag*, dan akan dihasilkan nomor ID buku yang akan disimpan pada *buffer*.

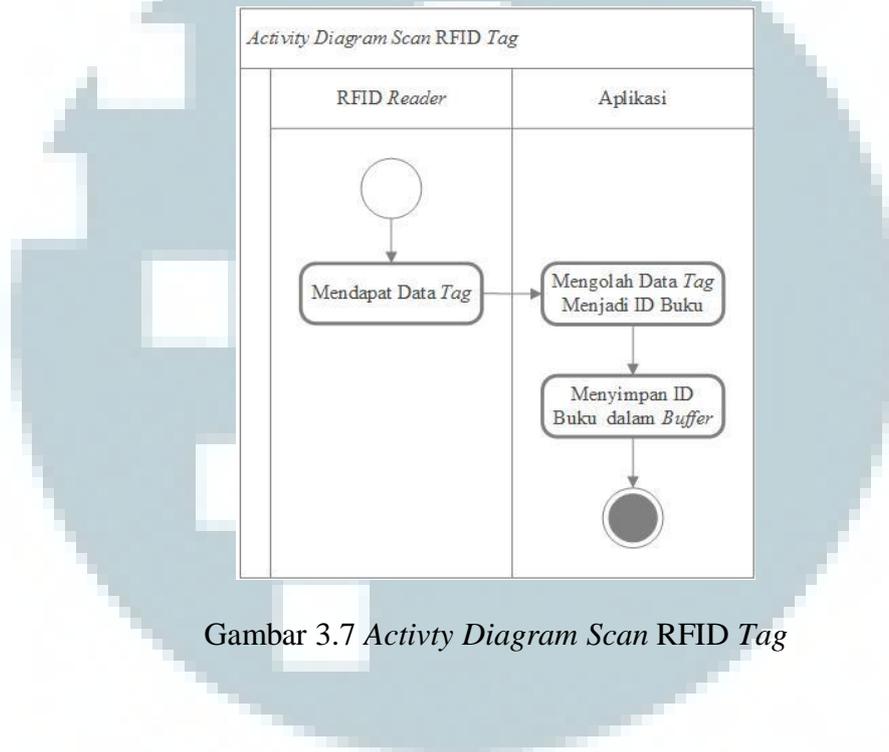
Aplikasi akan mengambil nomor ID buku dari *buffer*, dan akan dibuat *query* yang berisi pemeriksaan ID buku tersebut pada *database*. Setelah *database* mengirimkan hasilnya (berupa informasi buku), aplikasi akan memasukkan pada suatu variabel sebelum ditampilkan pada tabel aplikasi.

Proses selanjutnya aplikasi akan memeriksa posisi buku dengan acuan buku sebelumnya dan nomor DDC yang dimasukkan. Dari data tersebut aplikasi dapat menghasilkan status buku (*correct* maupun *misplaced*). Status tersebut akan digabungkan dengan informasi buku dan akan ditampilkan pada tabel aplikasi.

Jika pengguna tidak menekan tombol *stop* maka proses identifikasi akan diulang dari proses menerima data dari RFID *reader*, namun jika tombol *stop* ditekan maka proses identifikasi selesai.

2. Bagian Pengolahan Data *Tag*

Bagian pengolahan data *tag*, digambarkan pada Gambar 3.7 *Activity Diagram Scan RFID Tag*.

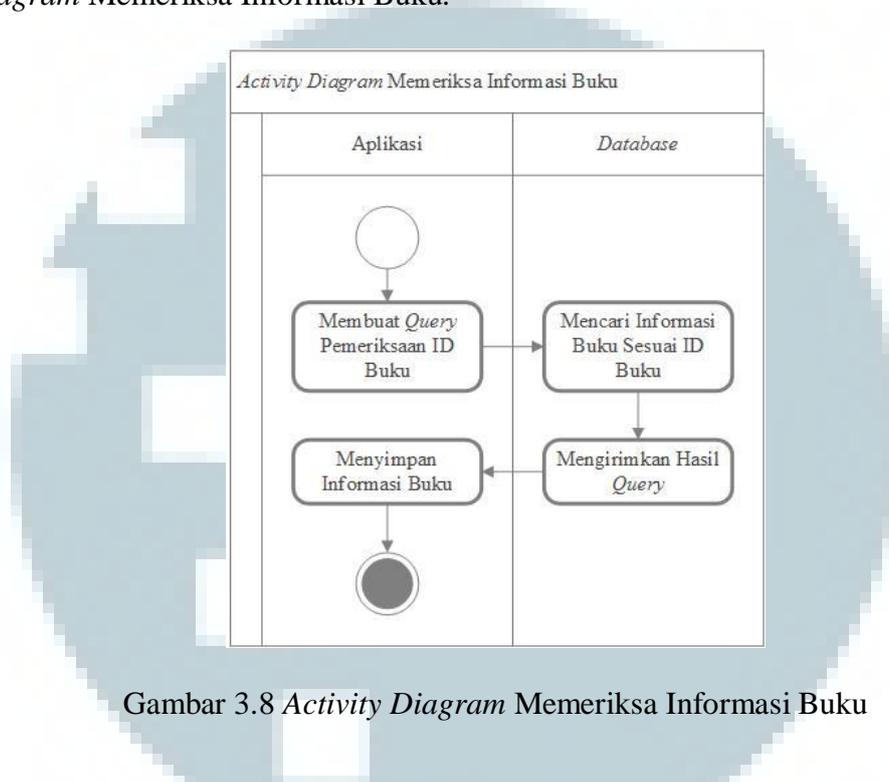


Gambar 3.7 *Activity Diagram Scan RFID Tag*

Pada bagian ini RFID *reader* akan melakukan *scanning tag* pada area bacanya. RFID *reader* akan membaca nomor UID *tag* dan nomor ID buku yang terdapat dalam *user data block*. Selanjutnya RFID *reader* akan mengirimkan data-data tersebut ke aplikasi untuk diolah. Dari pengolahan UID dan *user data block* akan dihasilkan nomor ID buku. ID buku tersebut akan disimpan pada *buffer* untuk proses identifikasi buku.

3. Bagian Pemeriksaan Informasi Buku

Bagian pemeriksaan informasi buku, digambarkan pada Gambar 3.8 *Activity Diagram Memeriksa Informasi Buku*.



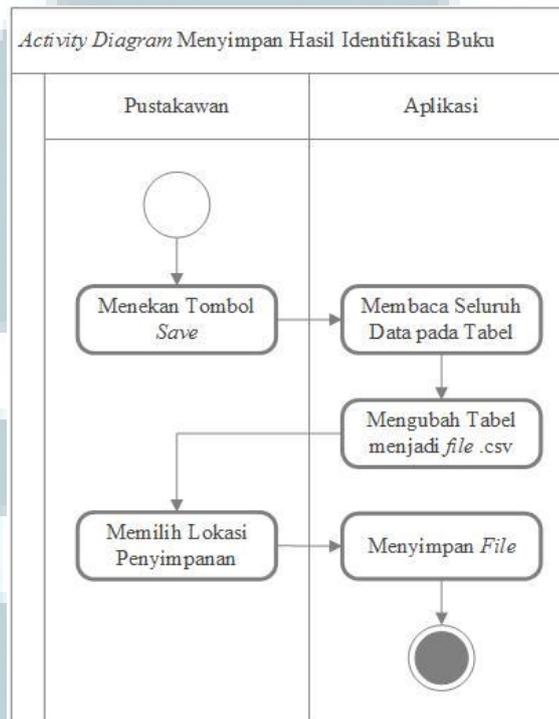
Gambar 3.8 *Activity Diagram Memeriksa Informasi Buku*

ID buku yang didapat akan diperiksa dan dicocokkan dengan data pada *database*. Aplikasi akan membuat *query* pemeriksaan ID buku kemudian pada *database*, *query* akan diproses dan akan dicari data/ informasi yang sesuai dengan ID buku yang diberikan. *Database* yang digunakan adalah *database* stockopname yang hanya berisi data-data buku di dalam rak. Hasil pencarian akan dikirimkan kembali ke aplikasi. Informasi tersebut kemudian akan disimpan pada suatu variabel penampung sebelum dilengkapi dengan data posisi yang akan dihasilkan pada proses identifikasi buku.

4. Bagian Penyimpanan Hasil Identifikasi Buku

Bagian penyimpanan hasil identifikasi buku, digambarkan pada Gambar 3.9

Activity Diagram Menyimpan Hasil Identifikasi Buku.

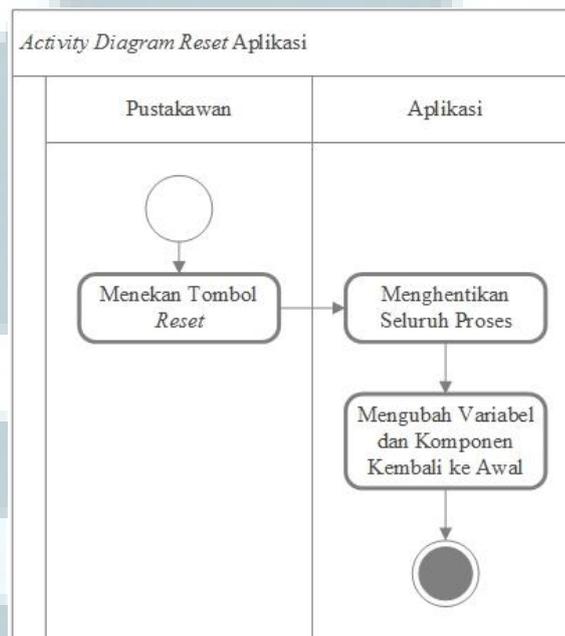


Gambar 3.9 *Activity Diagram* Menyimpan Hasil Identifikasi Buku

Setelah proses identifikasi buku, hasilnya dapat disimpan ke dalam *file* .csv. Proses penyimpanan dimulai dari membaca seluruh data pada tabel aplikasi yang berisi informasi buku dan posisi buku. Kemudian dari tabel tersebut akan diubah menjadi format .csv. Pengguna dapat memilih lokasi penyimpanan *file* pada direktori komputernya.

5. Bagian *Reset* Aplikasi

Bagian *reset* aplikasi, digambarkan pada Gambar 3.10 *Activity Diagram Reset* Aplikasi.

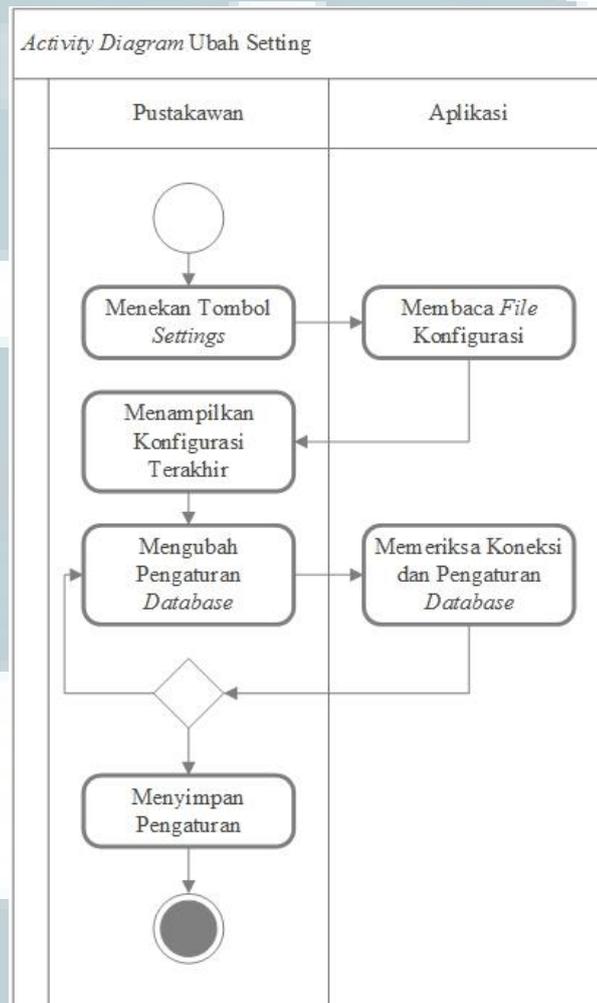


Gambar 3.10 *Activity Diagram Reset* Aplikasi

Fungsi dari *reset* aplikasi digunakan untuk menghapus seluruh data buku pada aplikasi yang telah diidentifikasi dan akan kembali pada kondisi awal aplikasi dijalankan. Prosesnya, setelah pengguna menekan tombol *reset* maka akan muncul kotak dialog untuk memastikan keputusan pengguna. Proses *reset* diawal dengan menghentikan proses-proses yang ada (memutuskan koneksi *database* dan komponen RFID *reader* terhadap aplikasi, menutup seluruh *thread* yang masih berjalan, dan mengubah seluruh status proses menjadi berhenti/*stop*). Lalu seluruh komponen akan di-*load* ulang dan variabel-variabel yang ada akan diisi dengan nilai *default* saat pertama aplikasi dijalankan.

6. Bagian Perubahan Pengaturan Aplikasi

Bagian perubahan pengaturan aplikasi, digambarkan pada Gambar 3.11 *Activity Diagram Ubah Setting*.



Gambar 3.11 *Activity Diagram Ubah Setting*

Proses ini digunakan jika terjadi kegagalan koneksi *database* dengan aplikasi maupun perubahan pada koneksi *database*. Pengguna dapat mengubah nama *server*, nama *database*, nama tabel, *username* dan *password* yang akan digunakan

untuk menghubungkan aplikasi ke *database*. Terdapat pilihan “*check database connection*” untuk memastikan perubahan yang dilakukan berhasil atau tidak. Setelah selesai pengguna dapat menyimpan *file* konfigurasi agar dapat digunakan kembali.

