

## BAB II

### TINJAUAN PUSTAKA DAN LANDASAN TEORI

#### 1.1. Tinjauan Pustaka

Dari banyak sekolah yang menyediakan perpustakaan di Indonesia, masih ada yang mengolah data mereka menggunakan sistem manual atau belum terkomputerisasi. Oleh sebab itu sistem administrasi perpustakaan lebih efektif jika menggunakan sistem komputerisasi. Sistem terkomputerisasi perpustakaan sudah banyak dikembangkan sebelumnya dengan metode dan perangkat lunak yang berbeda-beda.

Jenar Kuswidiardi(2012) mengembangkan aplikasi tentang “*Sistem informasi perpustakaan berbasis Microsoft Visual Basic 6.0 dengan Database SQL Server 2000 di perpustakaan SMK YPKK 1 Sleman*”. Pengembangan aplikasi ini menggunakan metode *Research and Development (R&D)*. Aplikasi ini dibangun menggunakan *software Microsoft Visual Basic 6.0 dan Microsoft SQL Server 2000*. Sistem yang dibangun memudahkan petugas perpustakaan dalam mendata siswa yang ingin meminjam buku di perpustakaan tersebut.

Herbowo Tri Nugroho dan Bhekti Pamilih(2011) mengembangkan aplikasi tentang “*Perancangan Sistem Informasi Perpustakaan Pada SMP N 1 Seyegan dengan Menggunakan Java*”. Mereka mengembangkan aplikasi dengan menggunakan metode iterasi dan menggunakan diagram ERD, DFD untuk pemodelan aplikasi yang dibangun dengan menggunakan *software Netbeans 6.1.7* dan menggunakan *MySQL* sebagai basis data. Sistem yang dibangun membuahkan hasil yaitu aplikasi *desktop* yang dapat membantu para anggota perpustakaan dalam mendata siswa yang hendak meminjam buku di perpustakaan tersebut.

Barra Sulthani(2008) mengembangkan aplikasi tentang “*Sistem Informasi Perpustakaan Fakultas Kedokteran dan Kesehatan Universitas Muhammadiyah Jakarta*”. Pengembangan aplikasi ini menggunakan metode Pengumpulan data

seperti wawancara, pengamatan, dan studi pustaka. Untuk pemodelan aplikasi ini, dibangun dengan menggunakan bahasa pemrograman *PHP* dan menggunakan *MySQL* sebagai basis data. Dalam sistemnya menghasilkan sebuah aplikasi *desktop* untuk mempermudah petugas perpustakaan mendata siswa peminjam buku di perpustakaan tersebut.

Dari ketiga penelitian sebelumnya, aplikasi yang dibuat memberikan kemudahan dalam mendata siswa yang ingin meminjam buku pada perpustakaan, sesuai keinginan pengguna. Namun dari ketiga aplikasi yang telah mereka buat, hanya mempermudah petugas perpustakaan dalam mendata peminjam buku. Berdasarkan penelitian yang sudah ada, maka penulis menambah beberapa spesifikasi yang menjadi perbedaan dengan sistem yang telah dikembangkan. Perbedaan tersebut adalah sebagai berikut :

1. Aplikasi dapat mendata siswa yang ingin meminjam buku di perpustakaan tanpa menulis di buku.
2. Aplikasi dapat mendata buku yang ada di perpustakaan.
3. Aplikasi dapat mendata karyawan yang bertugas pada perpustakaan.
4. Aplikasi dapat membuat laporan transaksi pengembalian
5. Aplikasi dapat menampilkan informasi buku(Katalog).
6. Aplikasi dapat memberi notifikasi keterlambatan pengembalian buku.

## **1.2. Landasan Teori**

### **1.2.1. Alat**

Penelitian ini memerlukan alat untuk mendukung berjalannya perancangan dan implementasi aplikasi.

a. Alat yang digunakan adalah perangkat keras dan perangkat lunak.

#### 1. Perangkat keras

Computer / Laptop yang digunakan untuk membuat aplikasi ini adalah:

- Processor : Intel Core i5 2,3 GHz.
- Operating System (OS) : Windows 10 64-bit.
- Installed Memory (RAM) : 4GB.
- Memory (Hardisk) : 500 GB.

Syarat minimum computer / laptop yang digunakan untuk membangun aplikasi ini adalah :

- Processor : 500 MHz Intel Pentium III.
- Operating System (OS) : Windows XP 32-bit.
- Installed Memory (RAM) : 512 MB.
- Memory (Harddisk) : 850 MB dari free disk space (memory yang kosong).

#### 2. Perangkat Lunak

- NetBeans IDE 8.0.2

NetBeans merupakan sebuah software editor untuk mengembangkan beberapa bahasa pemrograman, dan salah satunya adalah bahasa pemrograman Java. Dalam pengembangan aplikasi ini, penulis sangat terbantu dalam proses coding. Didalam perangkat lunak NetBeans terdapat IDE (Integrated Development Environment). Sebuah IDE merupakan ruang lingkup pemrograman yang diintegrasikan kedalam suatu aplikasi perangkat lunak yang menyediakan GUI (Graphic User Interface).

- MySQL Server 5.6

MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional yang didistribusikan secara gratis dibawah lisensi GPL (General Public License). Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tidak boleh dijadikan produk turunan yang bersifat komersial. Untuk mendukung pembuatan aplikasi ini, MySQL digunakan sebagai database server. Selain bersifat gratis, MySQL juga menggunakan query standar yang umum digunakan dalam proses pembuatan database.

### **1.2.2. Bahan**

Dalam penelitian ini bahan yang digunakan adalah:

1. Data didapat dari sekolah berdasarkan penelitian yang berkaitan dengan perpustakaan. Dari data tersebut dapat diperoleh pemahaman mengenai kebutuhan dari aplikasi yang akan dikembangkan.
2. Data karyawan perpustakaan sebagai bahan untuk membuat role pada aplikasi.

### **2.2.3 Prinsip 8 *Golden Rules of Interface Design***

Menurut Ben Shneiderman (1997), Human-Cumputer Interaction (HCI) adalah disiplin ilmu yang berhubungan dengan perancangan, evaluasi, dan implementasi sistem komputer interaktif untuk digunakan oleh manusia, serta studi fenomena-fenomena besar yang berhubungan dengannya. Fokus pada HCI adalah perancangan dan evaluasi antarmuka pemakai (*user interface*). *User interface* adalah bagian sistem komputer yang memungkinkan manusia berinteraksi dengan komputer.

Dalam bukunya yang berjudul "*Designing The User Interface*", Ben Shneiderman (1986) mengemukakan *8 golden rules principle*. *8 golden rules prinsiple* merupakan sebuah aturan yang menetapkan interface agar dapat

menyempurnakan implementasi dalam hal berinteraksi pada manusia. Beberapa poin-poin yang terdapat dalam 8 *golden rules* diantaranya sebagai berikut:

1. *Strive for consistency*

Konsisten terhadap jenis, ukuran, dan spasi *font* merupakan hal yang terpenting untuk membuat sebuah aplikasi yang mempunyai nilai yang baik. Tampilan aplikasi tidak seharusnya rumit, mempunyai warna yang tidak terlalu banyak. *User* biasanya tidak ingin mempelajari hal baru yang membuat waktu mereka terbuang karena menggunakan aplikasi.

2. *Cater to universal usability*

Mengenali kebutuhan *user* yang beragam dan memudahkan dalam melakukan perubahan konten. Mencari perbedaan user pemula dan user ahli, rentang usia, keterbatasan, kemampuan fisik, perbedaan teknologi merupakan panduan dalam merancang *interface*. Menambahkan fitur-fitur untuk *user* pemula seperti *shortcut* untuk *intermediate user* dan untuk *user* ahli membuat tampilan yang kompleks dapat meningkatkan kualitas dari desain *interface*.

3. *Offer informative feedback*

Aplikasi yang baik selalu memberikan timbal balik ketika terjadi sesuatu di dalam aplikasi itu sendiri. Sehingga memudahkan *user* harus melakukan apa dan harus bagaimana.

4. *Design dialogs to yield closure*

Aplikasi yang baik juga harus mempunyai suatu perbedaan suatu tampilan yang dimana bisa memberi tahu pengunjung mereka sedang melakukan apa. Biasanya ketika *user* melakukan aktivitas *log in* pada aplikasi. Tampilan sebelum masuk dan setelah masuk akan berbeda agar lebih memudahkan *user* apakah sudah masuk apa belum.

### 5. *Prevent errors*

Aplikasi yang bisa memberikan solusi kepada para *user* ketika *user* melakukan kesalahan tergolong kepada aplikasi yang baik. Terjadinya kesalahan yang dilakukan oleh *user* harus cepat ditanggapi oleh aplikasi dan memberikan solusi kepada para pengunjung untuk mengatasi kesalahannya tersebut.

### 6. *Permit easy reversal of actions*

Di dalam sebuah aplikasi harus adanya fasilitas untuk *user* bisa mengedit atau menghapus tindakan yang sudah dilakukan sebelumnya. Karena *human error* tidak dapat dihindarkan, maka fasilitas ini tentu wajib dimiliki disetiap aplikasi.

### 7. *Support internal locus of control*

Aplikasi perlu memiliki fasilitas untuk pengguna atau *user* agar dapat dengan bebas bernavigasi dan mengubah informasi akun yang dimilikinya sesuai dengan yang dikehendaki.

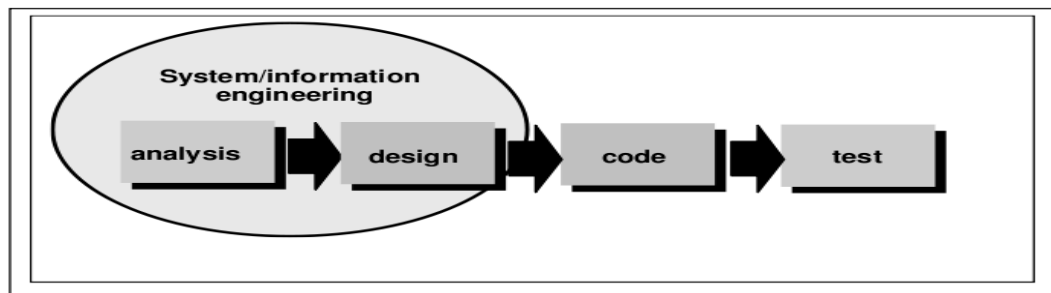
### 8. *Reduce short term memory load*

Keterbatasan manusia dalam mengolah informasi dalam memori jangka pendek membutuhkan antarmuka (*interface*) yang sederhana dan mengurangi pergerakan *window*. Yang dimaksud pergerakan *window* seperti ketika *user* harus mengingat suatu informasi pada satu layar dan informasi tersebut digunakan pada layar lain.

## **1.3. Alur Penelitian**

Untuk mengembangkan aplikasi dibutuhkan metode atau model pengembangannya, salah satunya adalah Waterfall. Metode Waterfall adalah sebuah metode pengembangan aplikasi dengan pendekatan sekuensial.

Pendekatan model ini terlihat mengalir menurun seperti air terjun (Waterfall) yang dikembangkan oleh Roger S. Pressman (2002) melalui beberapa tahapan. Tahapan pada model Waterfall dalam pembuatan aplikasi dapat dilihat pada gambar 2.1



**Gambar 2.1** Model Waterfall

Berikut penjelasan dari Tahapan-tahapan pada gambar 2.1 :

a. Analisa

Langkah pertama adalah menganalisa semua kebutuhan dari keseluruhan sistem dari sebelum pembuatan hingga perancangan. Pada tahap ini penulis melakukan identifikasi masalah, mulai dari latar belakang perlunya pembuatan forum diskusi online ini, sampai dengan ulasan untuk pemecahan masalah yang sedang dialami.

Proses identifikasi kebutuhan diintensifkan dan difokuskan pada software, seperti bagaimana tampilan user interface, bagaimana rule dan kebutuhan fungsi-fungsi yang digunakan dalam aplikasi.

b. Desain

Pada tahap selanjutnya yang dilakukan penulis adalah mengubah kebutuhan-kebutuhan sebelumnya menjadi representasi kedalam model “blueprint”. Sehingga maksud pembuatan model ini adalah untuk memperoleh pengertian yang lebih baik terhadap aliran data dan kontrol, proses-proses fungsional, tingkah laku operasi dan informasi-informasi yang terkandung didalamnya. Terdiri dari aktifitas utama permodelan proses yang

dimana penulis menggunakan use case diagram dan entity relational database diagram untuk permodelan data dan desain antarmuka di semua rancangan halaman.

c. Coding

Untuk dapat dimengerti oleh mesin, maka penulis melakukan coding atau menerapkan bahasa pemrograman atau bahasa yang dimengerti oleh mesin. Hal ini perlu dilakukan dalam setiap membangun aplikasi. Bahasa pemrograman yang digunakan adalah bahasa pemrograman Java. Dan software yang digunakan penulis dalam menerapkan bahasa pemrograman Java ini adalah Netbeans.

d. Testing

Testing atau pengujian adalah element kritis dari jaminan kualitas perangkat lunak dan mempresentasikan kajian pokok dari spesifikasi desain dan pengkodean. Pada umumnya pengujian dilakukan dengan menguji coba aplikasi secara mandiri dan lapangan. Ada dua metode pendekatan pengujian, pendekatan pertama adalah black box testing dan kedua adalah white box testing. Black box testing menyinggung uji coba yang dilakukan pada interface software. White box testing didasarkan pada pemeriksaan detail procedural.

Pada penelitian ini, metode yang dipakai dalam pengembangan aplikasi ini adalah black box testing. Black box testing atau tes fungsional adalah pengujian yang dilakukan hanya dengan mengamati hasil eksekusi melalui data uji dan memeriksai fungsional dari aplikasi tersebut.

Pengujian program dalam aplikasi web ini dilakukan oleh pengembang dan user yang terlibat untuk memberikan data yang akan di-input. Selain itu juga dilihat bagaimana aplikasi tersebut berjalan. Hal-hal yang menjadi perhatian dalam pengujian adalah sebagai berikut :



- a. Aplikasi dapat mendata siswa yang ingin meminjam buku di perpustakaan tanpa menulis di buku.
- b. Aplikasi dapat mendata buku yang ada di perpustakaan.
- c. Aplikasi dapat mendata karyawan yang bertugas pada perpustakaan.
- d. Aplikasi dapat membuat laporan transaksi pengembalian
- e. Aplikasi dapat menampilkan informasi buku(Katalog).
- f. Aplikasi dapat memberi notifikasi keterlambatan pengembalian buku.

#### **1.4. Analisa Kebutuhan Sistem**

Untuk mengidentifikasi masalah, maka diperlukan analisa. Dari hasil analisa tersebut nantinya akan diperoleh hasil berupa sebuah sistem yang dapat memecahkan masalah. Dari latar belakang dan rumusan masalah yang telah dibahas sebelumnya maka didapatkan analisa kebutuhan sistem sebagai berikut :

1. Sistem dapat digunakan untuk melakukan transaksi peminjaman dan pengembalian buku.
2. Sistem dapat digunakan untuk menyimpan data buku.
3. Sistem dapat digunakan untuk menambah atau menghapus data karyawan.
4. Sistem dapat memberi notifikasi keterlambatan pengembalian buku.
5. Sistem dapat membatasi jumlah buku yang dipinjam(Optional)
6. Sistem dapat mengatur stok buku yang tersedia.
7. Sistem dapat digunakan untuk membuat laporan bulanan.

## 1.5. Teori Pemrograman

### 1. Java

*Java* merupakan bahasa pemrograman yang bersifat umum atau nonspesifik (*general purpose*), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin. Karena fungsionalitasnya yang memungkinkan aplikasi *java* mampu berjalan di beberapa *platform* sistem operasi yang berbeda, *java* dikenal pula dengan slogannya, "Tulis sekali, jalankan di mana pun". Penulis memakai bahasa *java* dalam mengembangkan aplikasi karena *java* dapat dijalankan di beberapa platform atau sistem operasi komputer, sesuai dengan prinsip tulis sekali, jalankan di mana saja. *Java* merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis web.

#### a. Kelebihan Java

##### 1. *Multiplatform*

Kelebihan utama dari *Java* ialah dapat dijalankan di beberapa platform. Dengan kelebihan *java*, programmer cukup menulis sebuah program *Java* dan dikompilasi (diubah, dari bahasa yang dimengerti manusia menjadi bahasa mesin) sekali lalu hasilnya dapat dijalankan di atas beberapa platform tanpa perubahan. Kelebihan *java* memungkinkan sebuah program berbasis *java* dikerjakan di atas operating system Linux tetapi dijalankan dengan baik di atas *Microsoft Windows*. Platform yang didukung adalah *Microsoft Windows*, *Linux*, *Mac OS*, dan *Sun Solaris*.

##### 2. *Perpustakaan kelas yang lengkap*

*Java* terkenal dengan kelengkapan yang sangat memudahkan dalam penggunaan oleh para *programmer* untuk membangun aplikasinya. Kelengkapan perpustakaan ditambah dengan keberadaan komunitas *java* yang terus menerus membuat perpustakaan-perpustakaan baru untuk melingkupi seluruh kebutuhan

pembangunan aplikasi. 3. Memiliki sintaks seperti bahasa pemrograman C++ sehingga menarik banyak programmer C++ untuk pindah ke *Java*.

## **b. Kekurangan Java**

### *1. Tidak Kompatibel*

Masih ada beberapa hal yang tidak kompatibel antara platform satu dengan platform lain. Untuk *J2SE*, *SWT-AWT bridge* yang sampai sekarang tidak berfungsi pada *Mac OS X*.

### *2. Mudah didekompilasi*

Dekompilasi adalah proses membalikkan dari kode jadi menjadi kode sumber. Java merupakan bytecode yang menyimpan banyak atribut bahasa tingkat tinggi, seperti nama-nama kelas, metode, dan tipe data. Hal yang sama juga terjadi pada Microsoft .NET Platform. Dengan demikian, algoritma yang digunakan program lebih sulit disembunyikan dan mudah dibajak.

### *3. Penggunaan memori yang banyak*

Biasanya bukan merupakan masalah bagi pihak yang menggunakan teknologi terbaru (karena trend memori terpasang makin murah), tetapi menjadi masalah bagi mereka yang masih harus berlutut dengan mesin komputer berumur lebih dari 4 tahun.

## **2. MySQL**

*MySQL* adalah sebuah implementasi dari sistem manajemen basis data relasional yang di distribusikan secara gratis dibawah lisensi *GPL (General Public License)*. Setiap pengguna dapat secara bebas menggunakan *MySQL*, namun dengan batasan perangkat lunak tidak boleh dijadikan produk turunan yang bersifat komersial.

Berikut beberapa keistimewaan dari *MySQL* menurut beberapa sumber sebagai bahan pertimbangan:

### 1. *Portabilitas*

*MySQL* dapat berjalan stabil pada berbagai sistem operasi seperti *Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga*, dan masih banyak lagi.

### 2. *Perangkat lunak sumber terbuka (open source)*

*MySQL* didistribusikan sebagai open source sehingga dapat digunakan secara gratis.

### 3. *Multi-user*

*MySQL* dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

### 4. *Performance tuning*

*MySQL* memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak *SQL* per satuan waktu.

### 5. *Ragam tipe data*

*MySQL* memiliki ragam tipe data yang sangat kaya, seperti *signed* or *unsigned integer, float, double, char, text, date, timestamp*, dan lain-lain.

### 6. *Perintah dan Fungsi*

*MySQL* memiliki operator dan fungsi secara penuh yang mendukung perintah *Select* dan *Where* dalam perintah (*query*).

### 7. *Keamanan*

*MySQL* memiliki beberapa lapisan keamanan seperti password yang terenkripsi.

## 8. Skalabilitas dan Pembatasan

*MySQL* mampu menangani basis data dalam skala besar, dengan jumlah *record* lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

## 9. Konektivitas

*MySQL* dapat melakukan koneksi dengan klien menggunakan protokol *TCP/IP*, *Unix Soket (UNIX)*, atau *named pipes (NT)*.

## 10. Lokalisasi

*MySQL* dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahasa Indonesia belum termasuk di dalamnya.

## 11. Antar Muka

*MySQL* memiliki antar muka (*interface*) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi *API (Application Programming Interface)*.

## 12. Klien dan Peralatan

*MySQL* dilengkapi dengan berbagai *tool* yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk online.

## 13. Struktur tabel

*MySQL* memiliki struktur tabel yang lebih fleksibel dalam menangani *ALTER TABLE*, dibandingkan basis data lainnya semacam *PostgreSQL* ataupun *Oracle*.

## 1.6. Perangkat Lunak Pendukung

### a. NetBeans

*NetBeans* merupakan perangkat lunak (*software*) yang digunakan untuk mengembangkan aplikasi, di dalam perangkat lunak terdapat *IDE (Integrated Development Environment)*. Sebuah *IDE* merupakan lingkup pemrograman yang diintegrasikan ke dalam suatu aplikasi perangkat lunak yang menyediakan *GUI (Graphic User Interface)*.

### b. IReport

*IReport* merupakan sebuah aplikasi *designer* yang digunakan untuk membuat laporan yang didalamnya terdapat *Library JasperReport*. *Library JasperReport* sendiri merupakan *JAR (Java Library)* yang bersifat *open source* dan dirancang untuk menambahkan kemampuan pelaporan (*reporting capabilities*) pada aplikasi *java*.

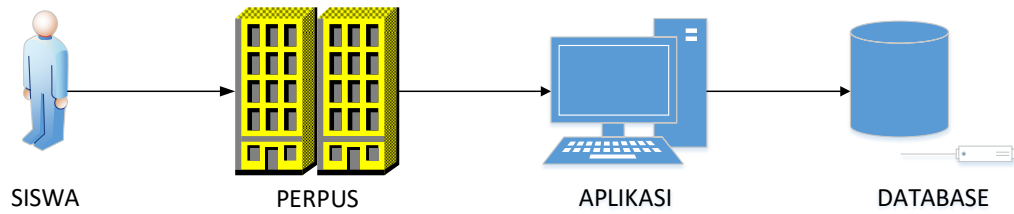
*JasperReport* memiliki sejumlah fitur, antara lain :

1. *Layout* dan desain laporan yang fleksibel.
2. Dapat menampilkan laporan dalam bentuk teks maupun gambar (*chart*).
3. Dapat menghasilkan *report* dalam berbagai format : *html, pdf, rtf, xls, csv*.
4. Dapat menerima data dari berbagai sumber data : *JDBC, Bean Collection, ResultSet, CSV, XML, Hibernate*.

## 1.7. Konsep Pembangunan Sistem

### 1.7.1. Arsitektur Website

Pembangunan sebuah sistem diperlukan perancangan, oleh karena itu penulis membuat gambaran menggunakan *UML* dan *ERD* untuk menunjukkan bagaimana proses sistem akan berjalan.



**Gambar 2.1** Arsitektur Website

a. Siswa

Siswa merupakan orang yang mengunjungi perpustakaan untuk membaca atau meminjam buku.

b. Perpus

Perpustakaan merupakan tempat penyimpanan buku-buku untuk belajar siswa.

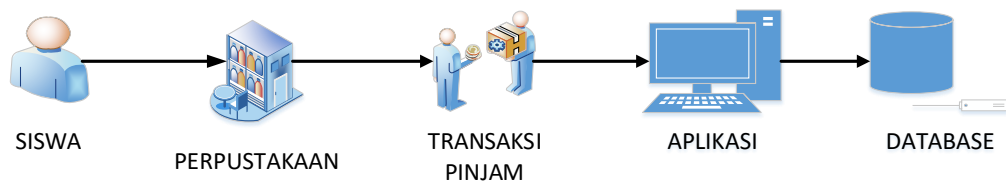
c. Aplikasi

Aplikasi adalah media bagi penjaga perpustakaan untuk mengolah segala jenis data yang ada dalam perpustakaan.

d. Database

Database ini untuk menyimpan data buku, karyawan, maupun transaksi simpan pinjam yang di lakukan siswa.

**1.7.2. Analisis Sistem Transaksi Pinjam**



**Gambar 2.2** Prosedur Transaksi

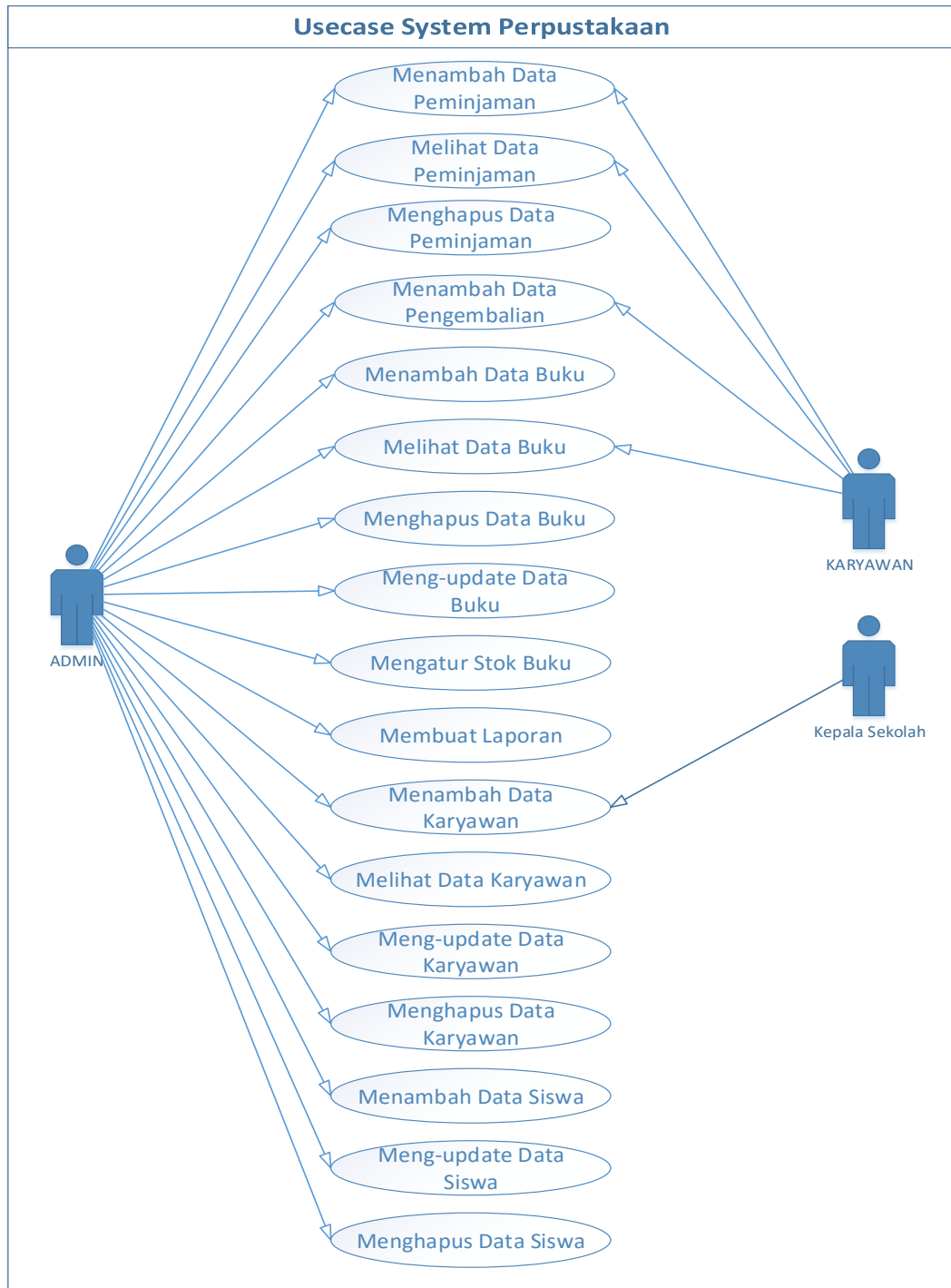
Proses yang terjadi pada sistem transaksi aplikasi Administrasi Perpustakaan SMAN 1 Kasui ini :

1. Siswa mengunjungi perpustakaan untuk meminjam buku.
2. Siswa memilih buku yang ingin dipinjam.
3. Siswa menemui petugas untuk di catat data diri serta buku yang ingin dipinjam.
4. Siswa diberi batas waktu selama tujuh hari peminjaman.



## 2.8. Perancangan Sistem

### a. Usecase Diagram

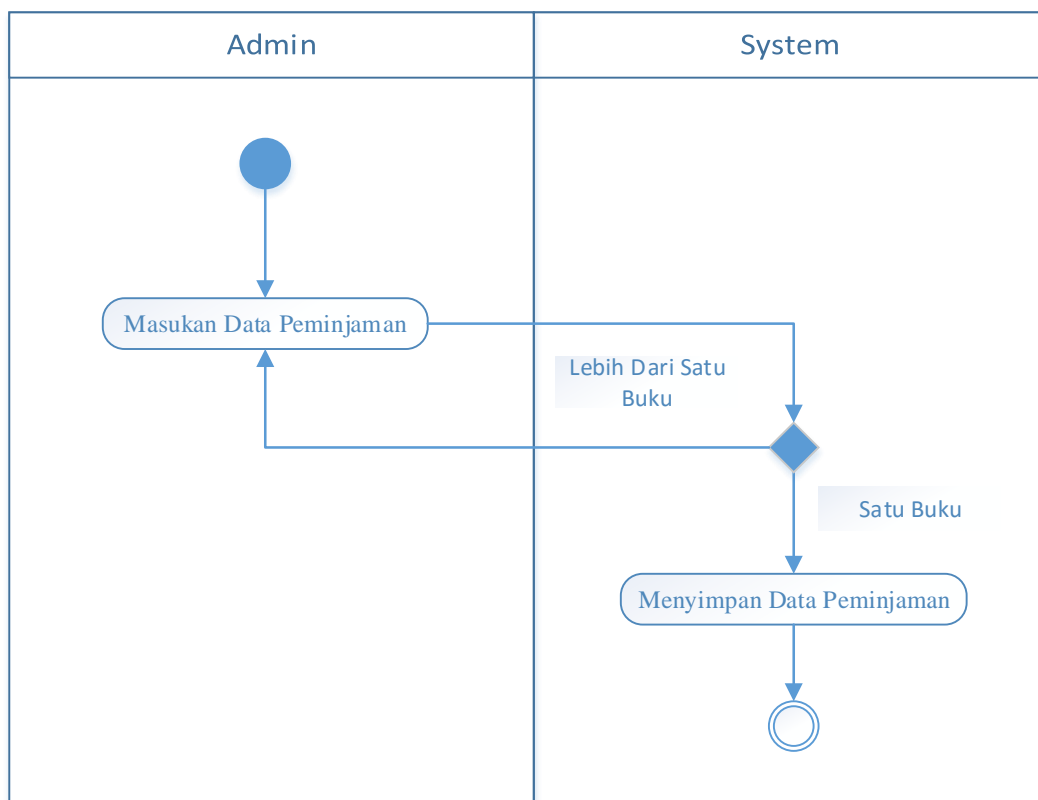


**Gambar 2.3** Usecase Diagram Role Aplikasi

Pada gambar 2.3 dijelaskan bahwa aplikasi memberikan tiga role yaitu admin, karyawan, dan kepala sekolah. Saat pertama menjalankan aplikasi, akan muncul form utama dan user memilih login untuk melakukan verifikasi role yang mereka miliki.

Aktifitas yang dapat dilakukan admin antara lain, melakukan pelayanan peminjaman dan pengembalian buku. Kemudian admin juga dapat memasukkan dan menghapus data buku. Lalu admin juga dapat membuat laporan transaksi yang terjadi pada aplikasi. Dan admin dapat menambah dan menghapus data karyawan.

**b. Activity Diagram**

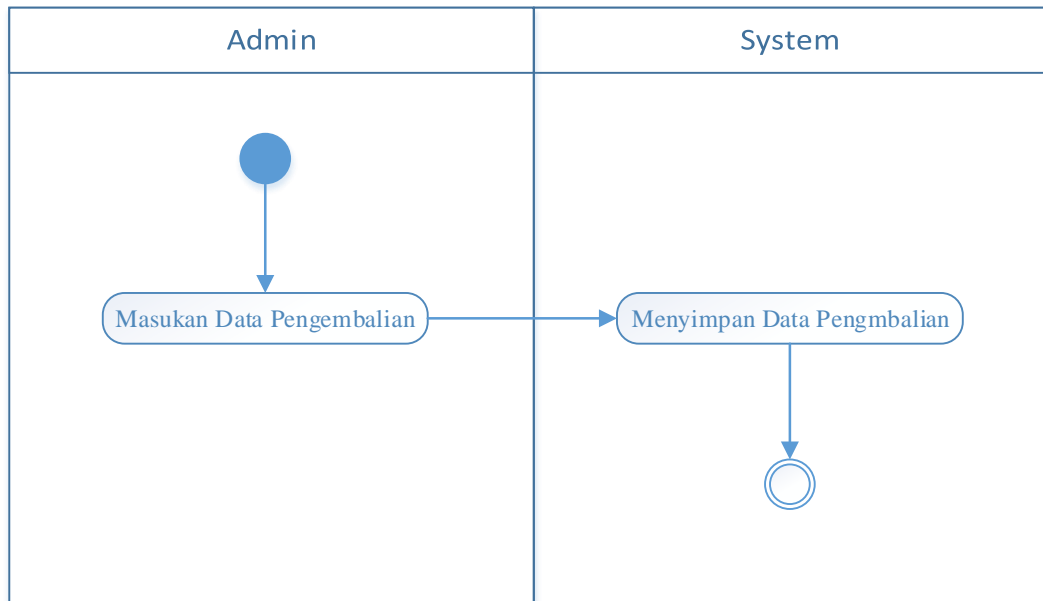


**Gambar 2.4** Peminjaman Buku

Proses yang terjadi pada sistem peminjaman:

1. Siswa datang ke perpustakaan, lalu meminjam buku.

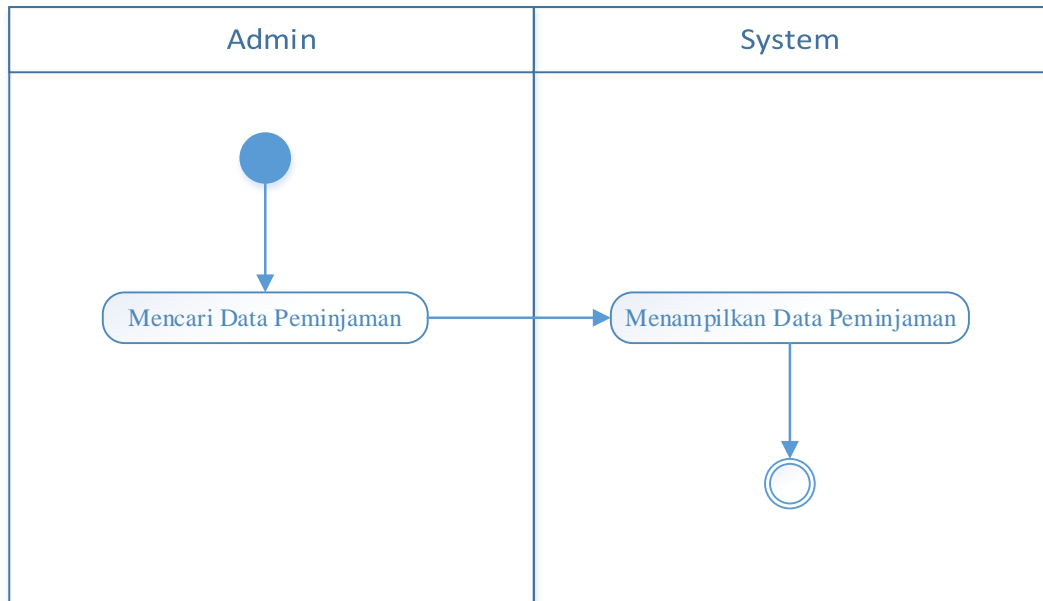
2. Petugas mengisi data peminjaman pada aplikasi.
3. Memasukkan data buku dari peminjam.
4. Data tersimpan.



**Gambar 2.5** Pengembalian Buku

Proses yang terjadi siswa melakukan pengembalian:

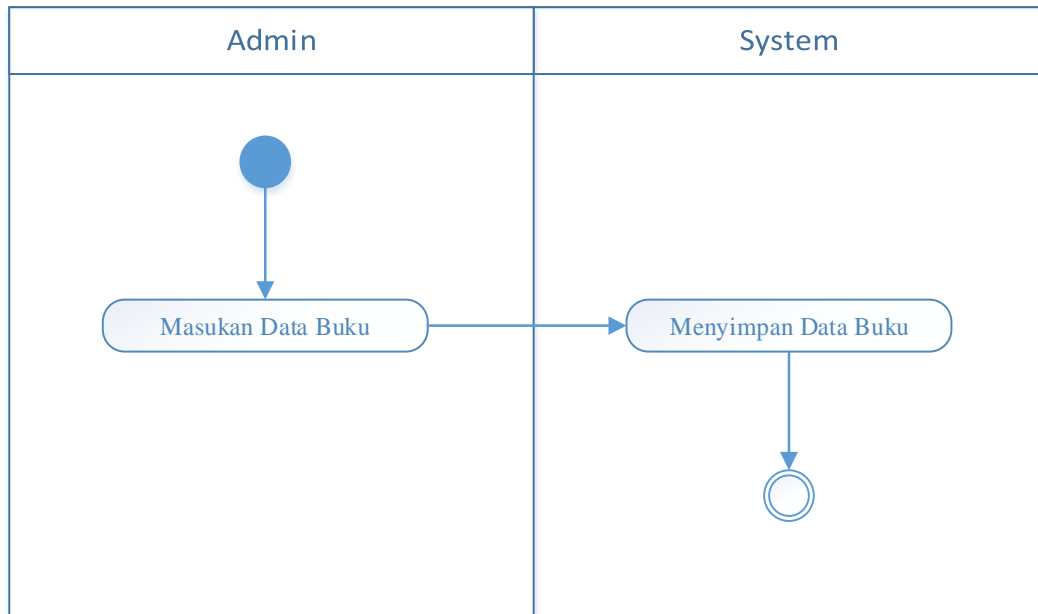
1. Siswa datang ke perpustakaan
2. Siswa membawa buku
3. Siswa menunjukan nis
4. Admin menghapus data peminjaman
5. Buku dikembalikan ditempatnya.



**Gambar 2.6** Mencari Data Peminjaman

Proses yang terjadi saat mencari data peminjaman:

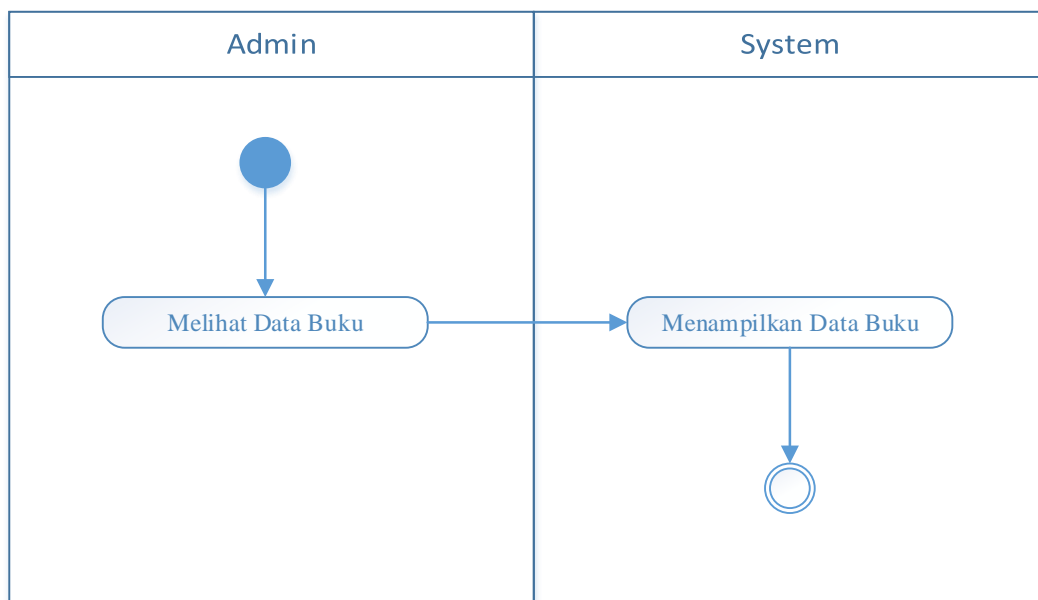
1. Admin mengetikkan nis untuk mencari data.
2. Aplikasi menampilkan data yang diinginkan.



**Gambar 2.7** Memasukkan Data Buku

Proses yang terjadi saat memasukkan data buku baru :

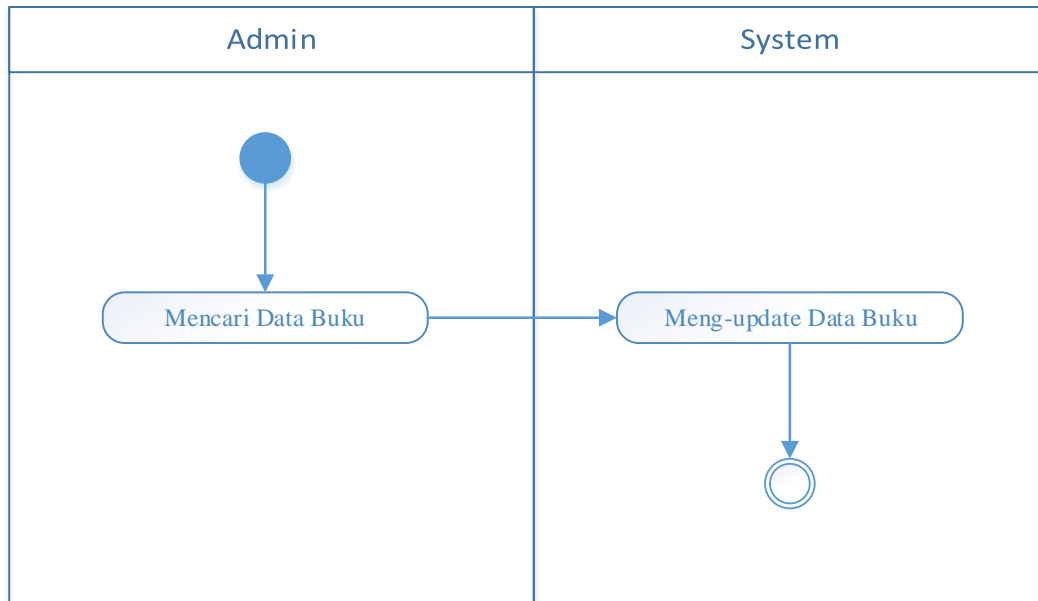
1. Admin memasukkan data buku pada aplikasi
2. Sistem menyimpan data pada database.



**Gambar 2.8** Mencari Data Buku

Proses yang terjadi saat membuat laporan :

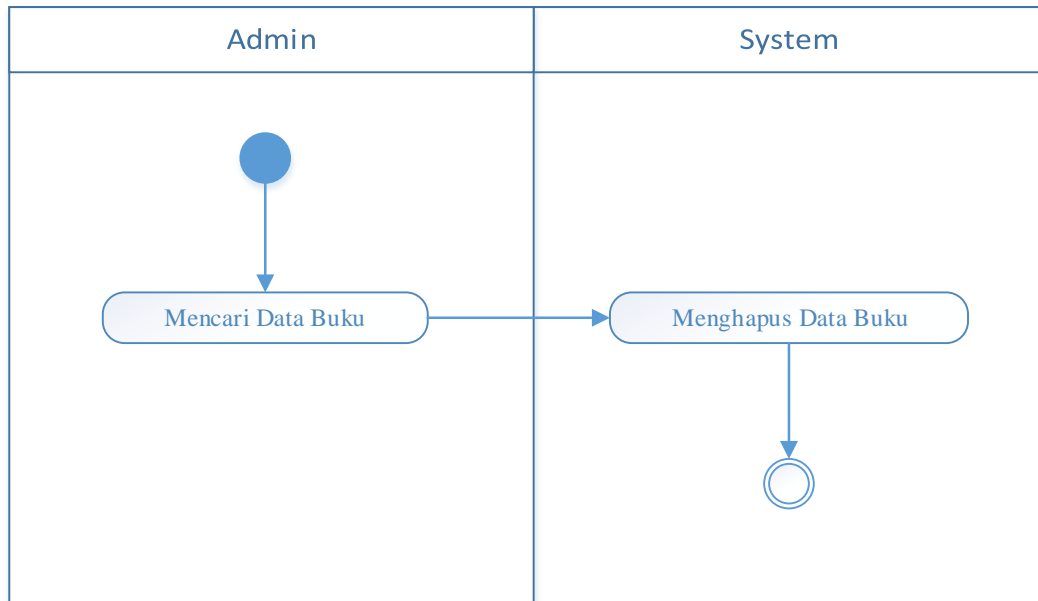
1. Admin mengetikkan nama buku
2. Sistem menampilkan data buku yang diinginkan



**Gambar 2.9** Meng-update Data Buku

Proses yang terjadi saat update data buku :

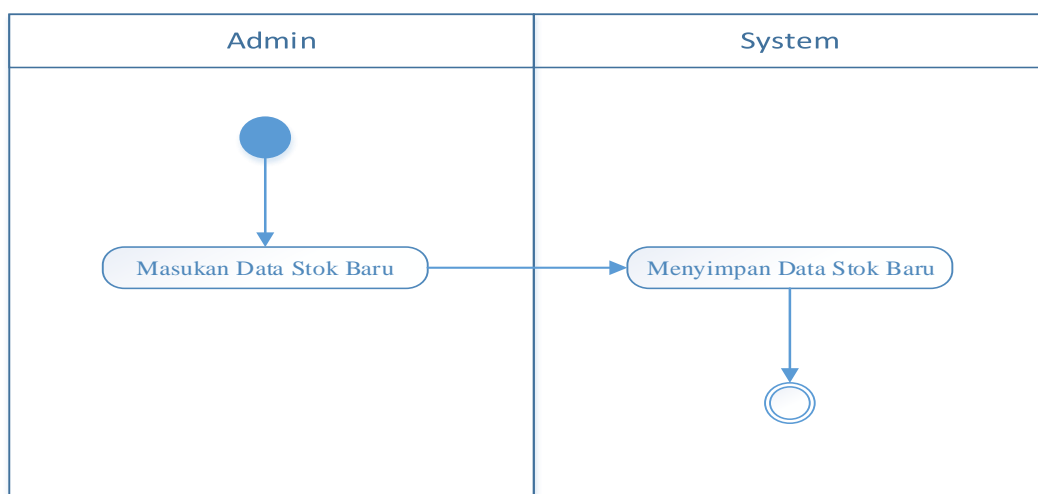
1. Admin mencari data yang ingin di-update
2. Sistem menampilkan data yang di cari
3. Admin memasukkan data baru
4. Sistem menyimpan data



**Gambar 2.10** Menghapus Data Buku

Proses yang terjadi saat menghapus data buku :

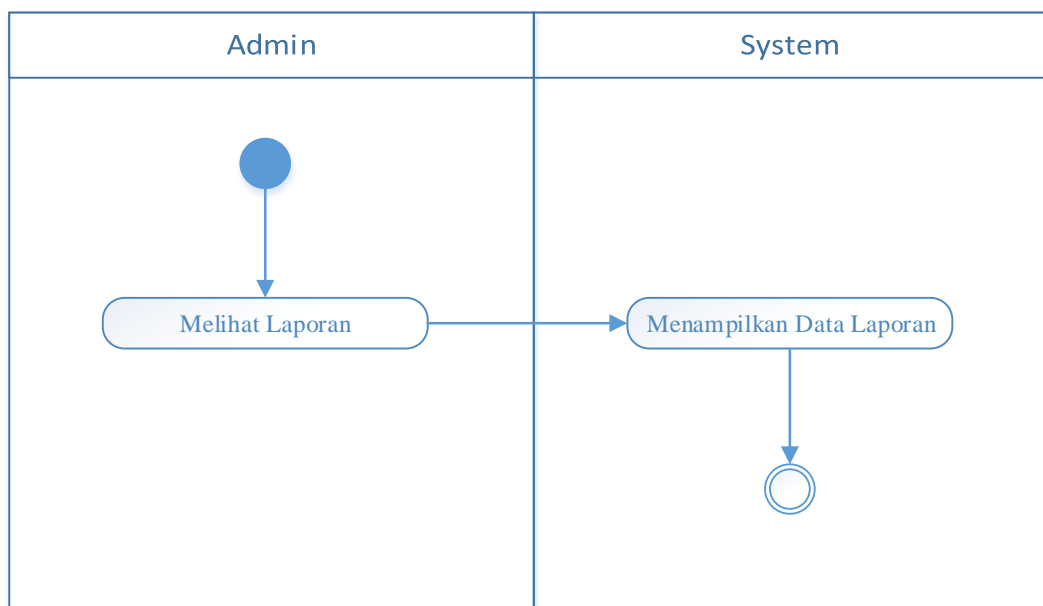
1. Admin mencari data yang ingin dihapus
2. Sistem menampilkan data yang diinginkan admin
3. Admin menghapus data buku.



**Gambar 2.11** Meng-update stok buku

Proses yang terjadi saat meng-update stok buku :

1. Admin mencari data buku yang ingin di-update
2. Sistem menampilkan data
3. Admin mengganti jumlah stok yang tersisa
4. Sistem menyimpan data yang baru.

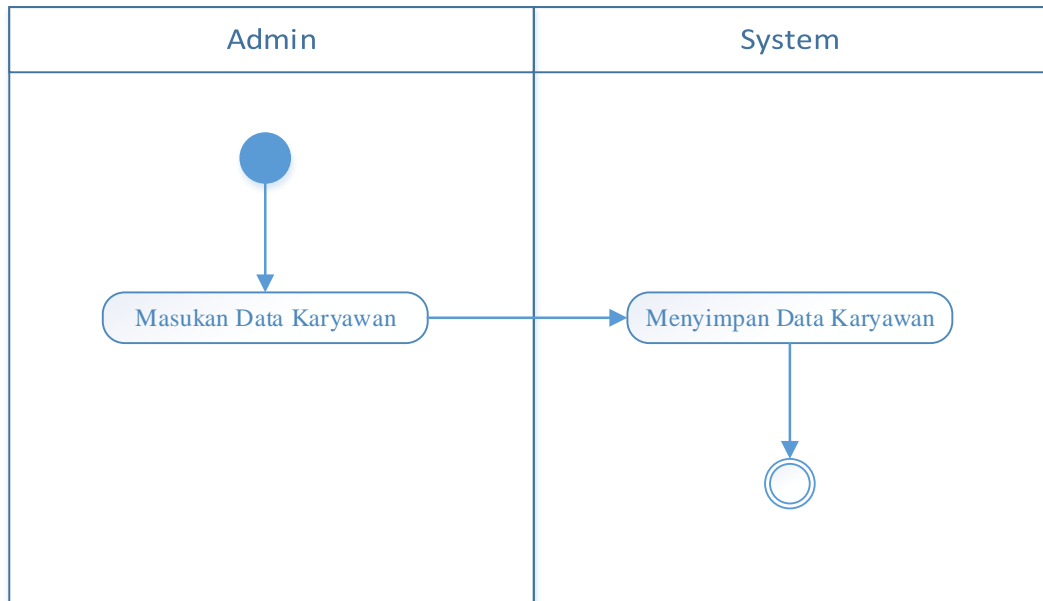


**Gambar 2.12** Membuat Laporan

Proses yang terjadi saat membuat laporan :

1. Sistem menampilkan data transaksi
2. Admin memilih bulan apa yang diinginkan untuk dilaporkan
3. Admin mencetak laporan
4. Sistem menampilkan laporan sesuai bulan yang diinginkan admin

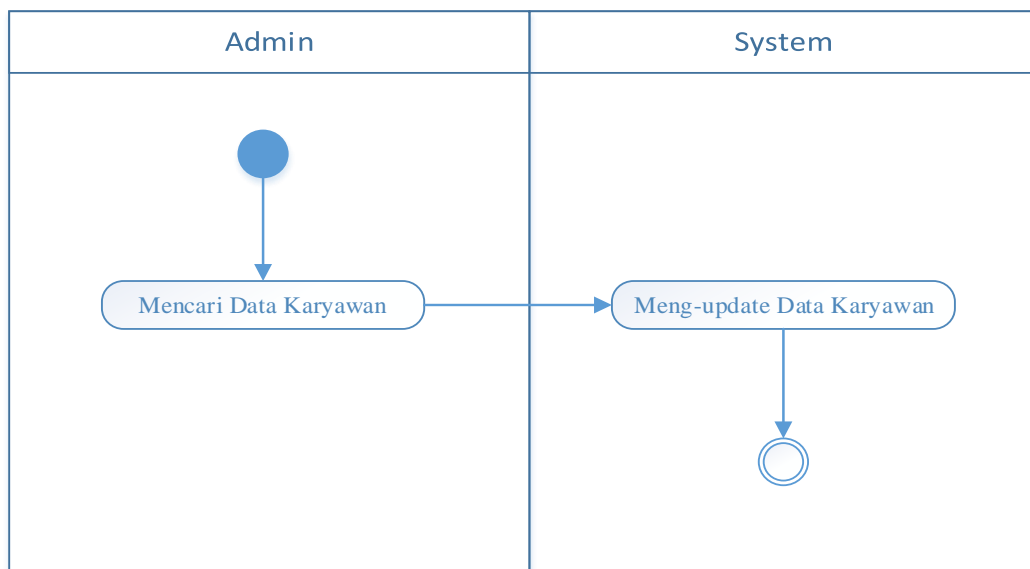




**Gambar 2.13** Memasukkan Data Karyawan

Proses yang terjadi saat memasukkan data karyawan :

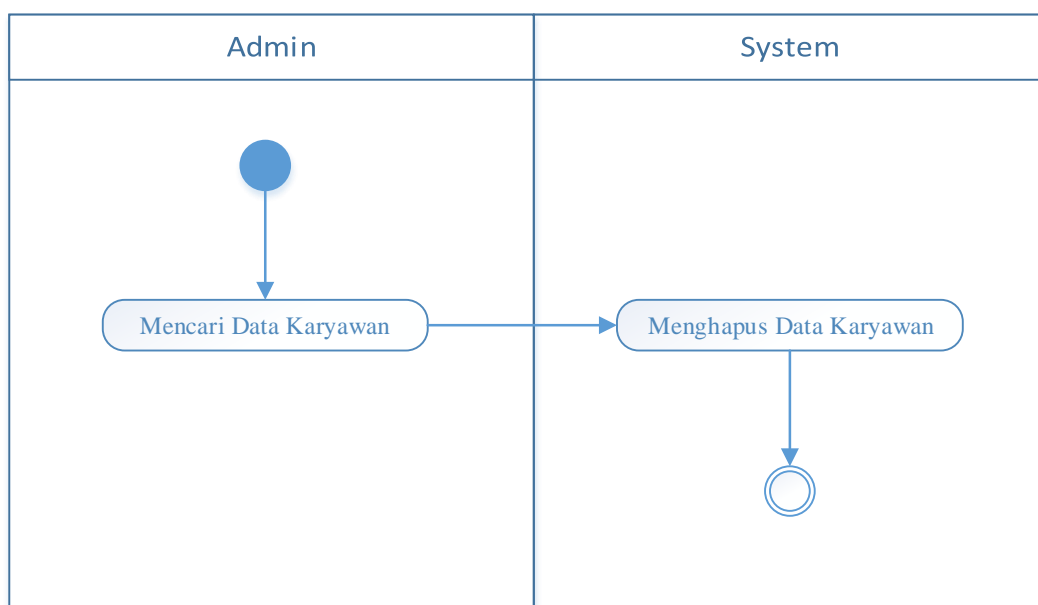
1. Admin meng-input data karyawan
2. Sistem menyimpan data di database



**Gambar 2.14** Meng-update Data Karyawan

Proses yang terjadi saat meng-update data karyawan :

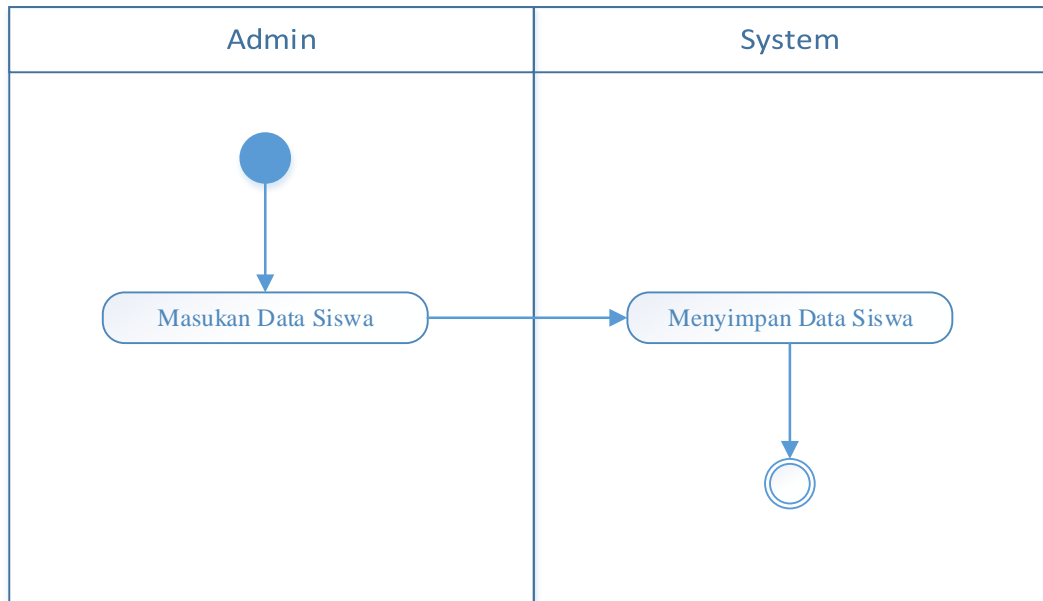
1. Admin mencari data karyawan
2. Sistem menampilkan data yang dicari
3. Admin memasukkan data baru
4. Sistem menyimpan data



**Gambar 2.15** Menghapus Data Karyawan

Proses yang terjadi saat menghapus data karyawan :

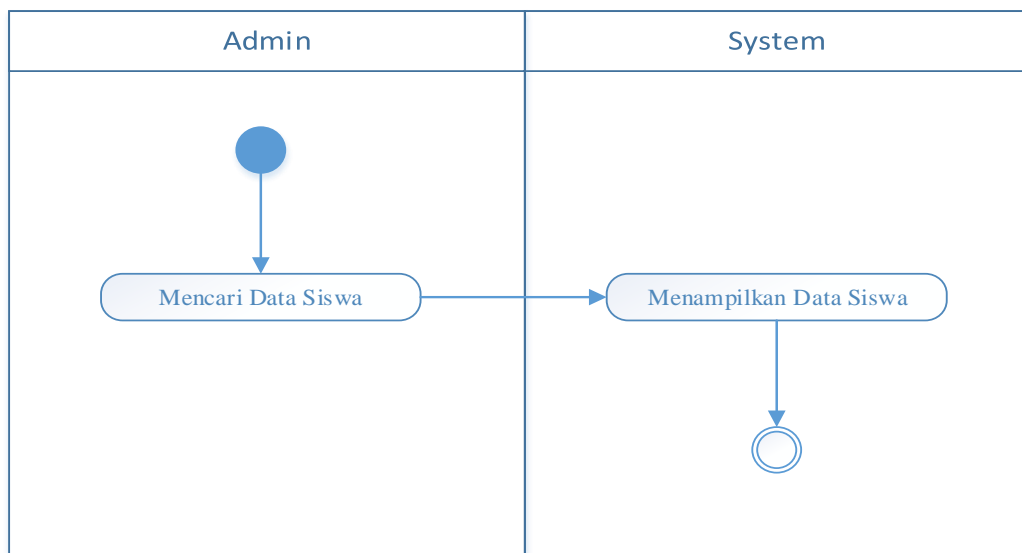
1. Admin mencari data karyawan
2. Sistem menampilkan data karyawan
3. Admin menghapus data karyawan



**Gambar 2.16** Memasukkan Data Siswa

Proses yang terjadi saat memasukkan data siswa :

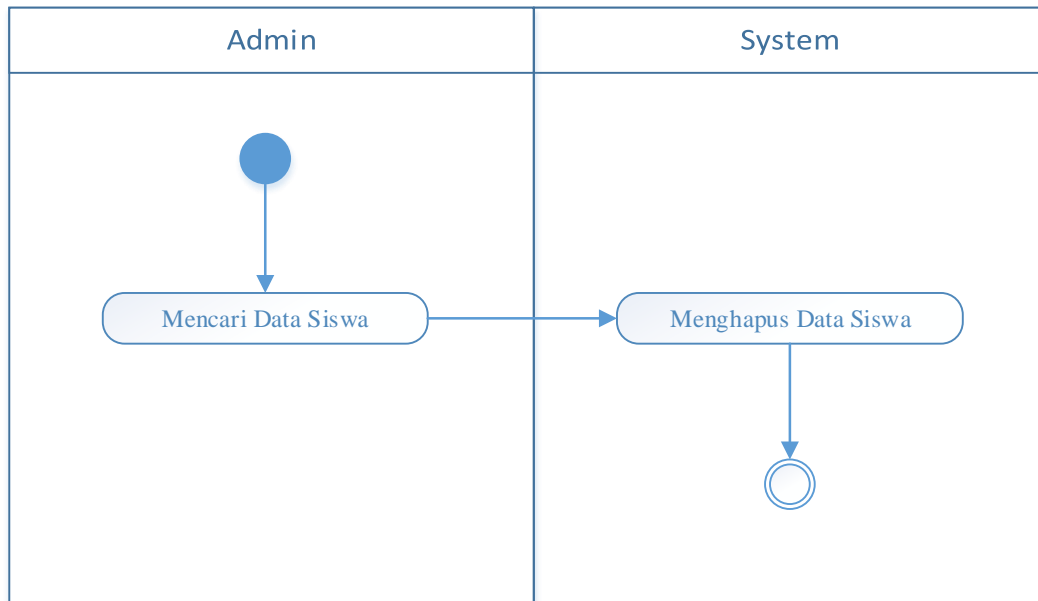
1. Admin memasukkan data
2. Sistem menyimpan data



**Gambar 2.17** Mencari Data Siswa

Proses yang terjadi saat mencari data siswa :

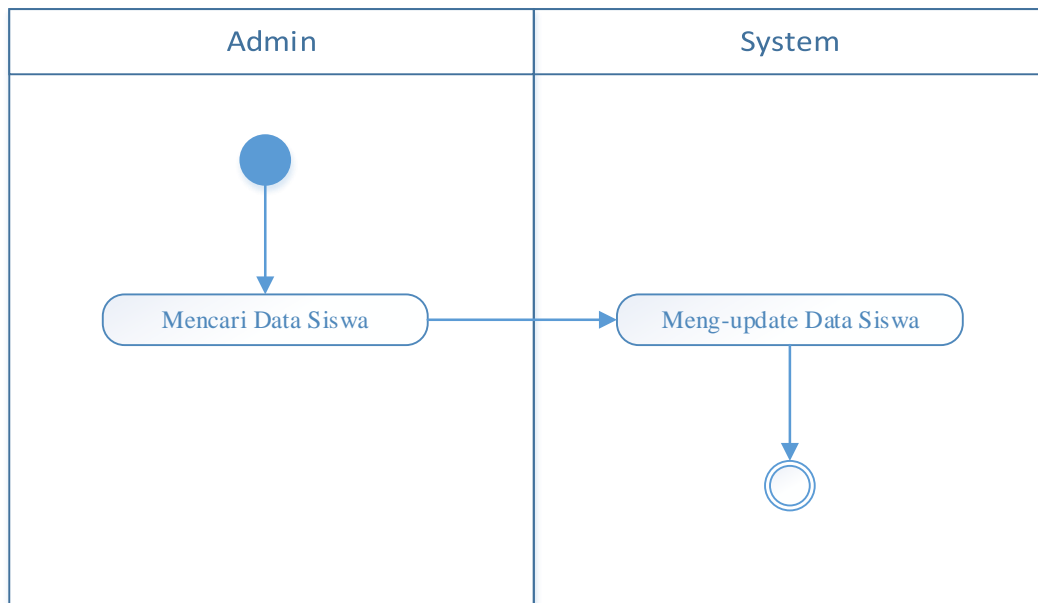
1. Admin mengetikkan nis
2. Sistem menampilkan data siswa



**Gambar 2.18** Menghapus Data Siswa

Proses yang terjadi saat menghapus data siswa :

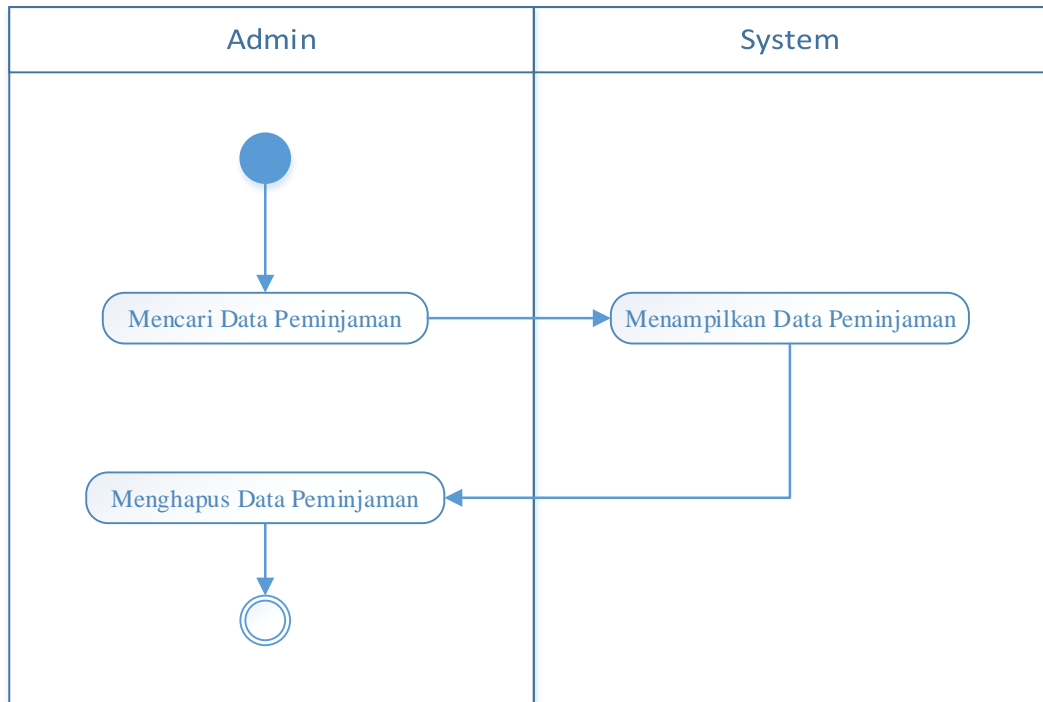
1. Admin mencari data karyawan
2. Sistem menampilkan data
3. Admin menghapus data



**Gambar 2.19** Meng-update Data Siswa

Proses yang terjadi saat meng-update data siswa :

1. Admin mencari data yang ingin di-update
2. Sistem menampilkan data
3. Admin memasukkan data baru
4. Sistem menyimpan data



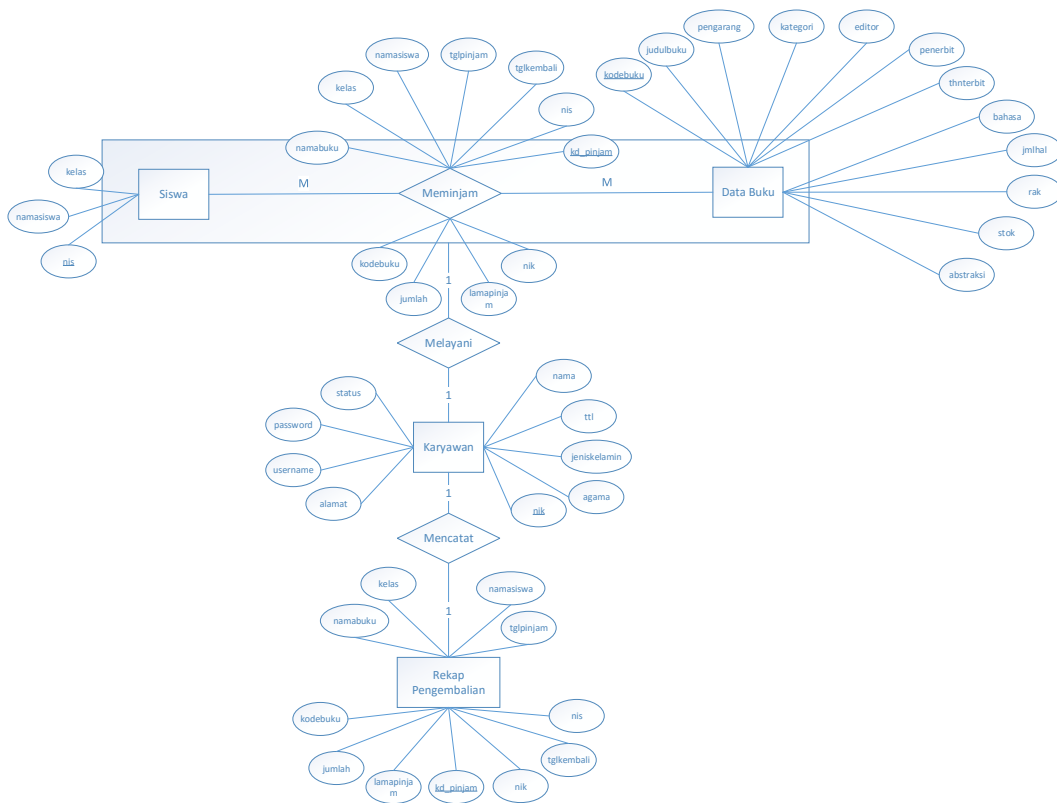
**Gambar 2.20** Menghapus Data Peminjaman

Proses yang terjadi saat menghapus data peminjaman :

1. Admin mencari data peminjaman
2. Sistem menampilkan data
3. Admin menghapus data

### c. ERD Diagram

Secara keseluruhan, sistem yang dirancang dapat dilihat pada ER Diagram dibawah.



**Gambar 2.21** Proses transaksi administrasi perpustakaan

#### 2.8.1. Rancangan User Interface

Rancangan interface atau tampilan yang dirancang ini adalah acuan untuk penulis dalam membangun tampilan. Dengan rancangan ini akan mempermudah penulis untuk mengatur setiap detail yang ada pada website. Adapun rancangan interface ini terbagi menjadi 3 bagian, interface admin, karyawan, dan kepala sekolah. Adapun bagian-bagian itu dapat dilihat pada gambar-gambar dibawah.

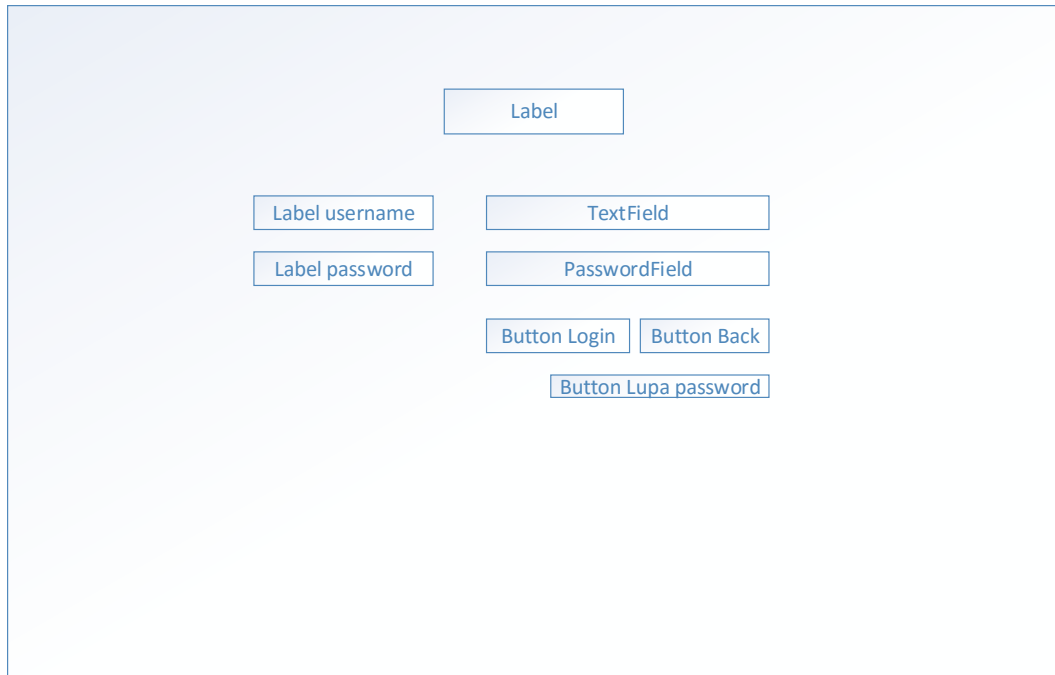


**Gambar 2.22** Rancangan Form Utama

### 1. Form Utama

Form ini akan menampilkan notifikasi pada label notifikasi, yaitu untuk menampilkan pemberitahuan siswa yang telat mengembalikan buku. Kemudian ada label untuk menampilkan nama aplikasi. Ada juga tombol login dan exit, login akan digunakan untuk memanggil form login, dan exit digunakan untuk keluar dari aplikasi.

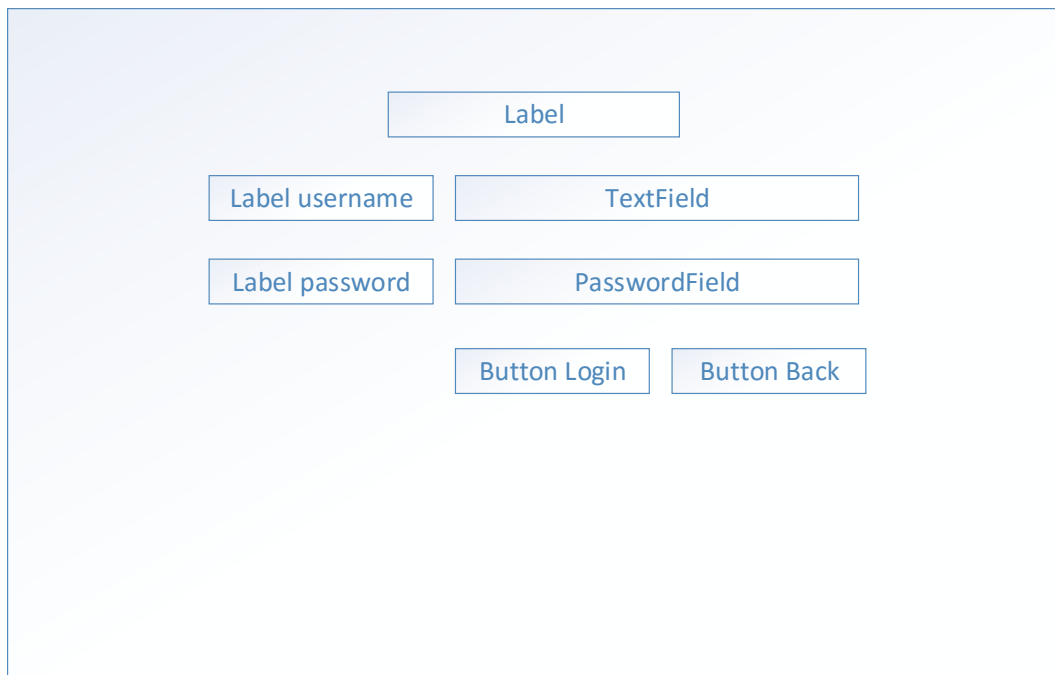




**Gambar 2.23** Rancangan Form Login

## 2. Form Login

Pada form ini akan menampilkan textfield dan passwordfield yang digunakan untuk sistem autentikasi sistem. Tombol login akan digunakan untuk verifikasi akun user, apakah admin atau karyawan biasa. Kemudian ada tombol lupa password yang akan digunakan oleh kepala sekolah untuk membuat akun baru.



**Gambar 2.24** Rancangan Form Login Admin Kepala Sekolah

### 3. Form Login Admin Kepala Sekolah

Form ini akan menampilkan textfield dan passwordfield yang digunakan untuk mengisi username dan password kepala sekolah, lalu digunakan untuk mengakses form register user baru.

Label	
NIK	TextField
Nama	TextField
Tgl lahir	Date Chooser
Jenis Kelamin	ComboBox
Agama	ComboBox
Alamat	TextField
Username	TextField
Password	TextField
Status	TextField
	Button Simpan

**Gambar 2.25** Rancangan Form Register User Baru

#### 4. Form Register User Baru

Form ini akan digunakan user untuk meng-input data karyawan baru, yang juga berguna untuk membuat akun baru. Pada form ini terdapat textfield untuk digunakan meng-input data karyawan baru. Kemudian ada tombol simpan untuk menyimpan data.

**Gambar 2.26** Rancangan Form Admin(Peminjaman)

## 5. Form Admin(Peminjaman)

Form ini akan menampilkan label notifikasi sebagai pemberitahuan keterlambatan pengembalian buku, role yang menunjukkan bahwa yang sedang login admin atau karyawan, tombol logout untuk keluar dari aplikasi, kemudian tab. Disini tab peminjaman akan digunakan untuk meng-input data peminjaman. Pada tab ini berisi textfield untuk mengetikkan data peminjaman, tombol simpan untuk menyimpan data. Dan ada tabel peminjaman untuk menampilkan data peminjaman. Lalu ada tombol cari untuk mencari data peminjaman.

**Gambar 2.27** Rancangan Form Admin(Pengembalian)

#### 6. Form Admin(Pengembalian)

Pada tab pengembalian ini akan berisi textfield yang berisi data peminjam, dan ada tombol hapus untuk menghapus data peminjam. Kemudian tabel peminjaman akan menampilkan data-data peminjaman.

**Gambar 2.28** Rancangan Form Admin, Data Buku, Catalog

## 7. Form Admin, Data Buku, Catalog

Pada form ini akan menampilkan data buku, namun hanya kode, nama, dan stok buku yang akan ditampilkan pada tabel data buku karna hanya akan digunakan untuk mengetahui stok buku yang ingin dipinjam. Kemudian ada textfield dan textarea untuk mengetikkan data buku. Ada tombol simpan untuk menyimpan data buku, lalu tombol hapus dan edit untuk menghapus dan meng-update data buku. Dan juga ada tombol atur stok untuk mengatur stok.

Label Notifikasi				Role	Button Logout
Peminjaman	Pengembalian	Data Buku	Data Karyawan		
Catalog	Laporan				
Tabel Laporan					
		Combobox	Button Print		

**Gambar 2.29** Rancangan Form Admin, Data Buku, Laporan

#### 8. Form Admin, Data Buku, Laporan

Pada tab catalog ini akan menampilkan data transaksi peminjaman, yang akan digunakan oleh admin untuk membuat laporan. Ada combobox untuk memilih bulan sebagai parameter laporan. Lalu ada tombol print untuk mencetak laporan.

Label Notifikasi			Role	Button Logout			
Peminjaman	Pengembalian	Data Buku	Data Karyawan				
NIK	TextField	Button Cari					
Nama	TextField	Tabel Data Karyawan					
Tmpt Tgl Lahir	TextField						
Jenis Kelamin	ComboBox						
Agama	ComboBox						
Alamat	TextField						
Username	TextField						
Password	TextField						
Status	TextField						
					Button Simpan		
					Button Edit		
		Button Hapus					

**Gambar 2.30** Rancangan Form Admin(Data Karyawan)

#### 9. Form Admin(Data Karyawan)

Pada tab ini akan menampilkan table karyawan yang berisi data karyawan, textfield untuk menyetikkan data karyawan, tombol simpan untuk menyimpan data karyawan, tombol hapus untuk menghapus data karyawan, dan tombol cari untuk mencari data karyawan.



**Gambar 2.31** Rancangan Form Admin(Data Siswa)

#### 10. Form Admin(Data Siswa)

Pada tab ini akan menampilkan tabel siswa yang berisi data siswa, textfield untuk mengetikkan data siswa. Tombol simpan akan digunakan untuk menyimpan data siswa, tombol edit mengubah data siswa, dan tombol hapu untuk menghapus data siswa.

**Gambar 2.32** Rancangan Form Karyawan(Peminjaman)

#### 11. Form Karyawan(Peminjaman)

Pada form ini menampilkan label notifikasi yang berguna untuk memberi tahu user tentang keterlambatan pengembalian buku, ada label role untuk memberi tahu apakah admin atau karyawan yang sedang mengakses aplikasi, kemudian tombol logout untuk kembali ke form utama. Kemudian pada tab peminjaman, ada tabel peminjaman, ada textfield untuk mengetikkan data peminjaman, dan ada tombol simpan untuk menyimpan data peminjaman. Serta ada tombol cari untuk mencari data peminjaman.

The image shows a web application interface for the 'Pengembalian' (Return) section. At the top, there is a 'Label Notifikasi' and buttons for 'Role' and 'Button Logout'. Below this, there are three tabs: 'Peminjaman', 'Pengembalian' (which is selected), and 'Data Buku'. The main form area contains several input fields for data entry, each with a corresponding label: 'Kode Pinjam', 'NIS', 'Nama Siswa', 'Kelas', 'Nama Buku', 'Kode Buku', 'Jumlah', 'Tgl Pinjam', 'Tgl Kembali', 'Lama Pinjam', and 'NIK'. There are also two 'TextField' labels without corresponding input boxes. A 'Button Cari' (Search) is located at the top right of the form area, and a 'Button Hapus' (Delete) is at the bottom. A large rectangular area on the right side of the form is labeled 'Tabel Peminjaman'.

**Gambar 2.33** Rancangan Form Karyawan(Pengembalian)

## 12. Form Karyawan(Pengembalian)

Pada tab ini akan menampilkan textfield untuk menampilkan data peminjaman, lalu ada tombol hapus untuk menghapus data peminjaman. Dan ada tombol cari untuk mencari data peminjaman.

The image shows a wireframe of a web application interface. At the top, there is a notification bar labeled "Label Notifikasi". To its right are two buttons: "Role" and "Button Logout". Below this is a tabbed menu with four tabs: "Peminjaman", "Pengembalian", "Data Buku" (which is the active tab), and "Data Karyawan". Inside the "Data Buku" tab, there is a "Button Cari" (Search Button) and a large rectangular area labeled "Tabel Data Buku" (Data Book Table), which is currently empty.

**Gambar 2.34** Rancangan Form Karyawan(Data Buku)

### 13. Form Karyawan(Data Buku)

Pada tab ini akan ditampilkan hanya tabel data buku dan tombol cari.