

PROJECT DATABASE SYSTEM

**PERANCANGAN PROGRAM ADMINISTRASI K-ONE UNTUK
PERPUSTAKAAN SISWA/SISWI SMA NEGERI 5 TANGERANG**

Ditulis dalam rangka memenuhi penilaian mata kuliah Visual Programming dan
Database System



Disusun Oleh :

Agym Dimas TIAN (00000058149)
Angeline Marcellina Chandrawijaya (00000057862)
Benz Martheen W Jonash (00000059822)
Jericho Alexander Gunawan (00000058345)
Nicholas Suwandi Wijaya (00000057903)
Rheinald Adyatma Rizqi Wibowo (00000058204)

Program Studi: Information Systems

Mata Kuliah: IS341-Database System

**UNIVERSITAS MULTIMEDIA NUSANTARA
TANGERANG
SEMESTER GENAP 2021/2022**

DAFTAR ISI

HALAMAN PENGESAHAN PROPOSAL	4
HALAMAN PENILAIAN REVIEWER	6
BAB I	7
PENDAHULUAN	7
Latar Belakang Masalah	7
Rumusan Masalah	8
Tujuan	9
Profil Perusahaan	9
Proses Aplikasi	10
BAB II	11
Masalah Bisnis	11
System Requirements	11
Functional Requirements	12
Non-Functional Requirements	13
Use Case Diagram	15
Activity Diagram	16
BAB III	23
Proses Normalisasi	23
1NF (First Normal Form)	23
2NF (Second Normal Form)	24
Entity Relationship Diagram (Erd)	26
Flowchart	28
BAB IV	29
4.1 Tabel Relasi	29
4.2 Penjelasan Dan Gambar User Interface Aplikasi	30
4.3 Query Pembuatan Tabel Basis Data	53
4.3.1 Pembuatan Table DataBuku	53
4.3.2 Pembuatan Table Staff	53
4.3.3 Pembuatan Table Peminjaman	53

4.3.4 Pembuatan Table Pengembalian	53
4.3.5 Pembuatan Table Student	54
4.3.8 Memasukan Data Ke dalam Tabel DataBuku	54
4.3.9 Memasukan Data Ke dalam Tabel Staff	54
4.3.10 Memasukan Data Ke dalam Tabel Pengembalian	54
4.3.11 Memasukan Data Ke dalam Tabel Peminjaman	55
4.3.12 Memasukan Data Ke dalam Tabel Student	55
4.3.13 Menampilkan Tabel-Tabel	55
4.3.14 Menghapus Isi Tabel	55
4.3.15 Menghapus Tabel	55
4.4 Advanced Query Basis Data	56
4.5 Kesimpulan	56
Lampiran	58
Source Code Home	58
Source Code Log-In	61
Source Code Forgot Password	62
Source Code Peminjaman	64
Source Code Peminjaman2	68
Source Code Pengembalian	69
Source Code Pengembalian2	72
Source Code Book Data	73
Source Code Data Buku2	77
XI. Source Code Laporan Data Buku	78
XII. Source Code Laporan Data Peminjam	79
XIII. Source Code Report	80
XIV. Source Code Staff2	83
XV. Source Code Student2	84
XVI. Source Code Notes	85

HALAMAN PENGESAHAN PROPOSAL

Dokumen ini telah disetujui sebagai laporan akhir untuk penggerjaan proyek akhir program studi Sistem Informasi mata kuliah Database Systems dengan judul Perancang Program Administrator Perpustakaan K-ONE Siswa/Siswi SMA NEGERI 5 TANGERANG untuk membantu proses pengelolaan data peminjaman buku yang akan menjadi pengguna akhir dari dokumen ini.

Disusun oleh tim *kelompok 5 - K ONE LIBRARY*

NIM	Nama	Program	Tanda Tangan
00000058 149	Agym Dimas Tian	Sistem Informasi	
00000057 862	Angeline Marcellina Chandrawijaya	Sistem Informasi	
00000059 822	Benz Martheen W Jonash	Sistem Informasi	
00000058 345	Jericho Alexander Gunawan	Sistem Informasi	
00000057 903	Nicholas Suwandi Wijaya	Sistem Informasi	

00000058	Rheinald Adyatma Rizqi Wibowo	Sistem Informasi	
204			

Topik	:	Visualisasi Database Administrasi Buku Perpustakaan SMA Negeri 5 Tangerang
Judul Project (Indonesia)	:	Perancang Program Administrasi Perpustakaan K-ONE Siswa/Siswi SMA Negeri 5 Tangerang
<i>Judul Project (Inggris)</i>	:	K-ONE'S LIBRARY ADMINISTRATION PROGRAM FOR NEGERI 5 TANGERANG HIGH SCHOOL LIBRARY ADMINISTRATORS
Pembimbing	:	1. <<LO074887– Jansen Wiratama>> 2. <<LO00079 – Agus Sulaimain>>

Dibimbing Oleh:

Jansen Wiratama

Dosen Sistem Informasi

Tanggal:

HALAMAN PENILAIAN REVIEWER

Topik Final (Diisi oleh Dosen) :

Hasil Review Proposal Skripsi (Diisi oleh Dosen)

Tanggal Review	:	
Keputusan	:	Diterima / Revisi
Paraf Reviewer	:	

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Perpustakaan merupakan sebuah lembaga wajib yang memfasilitasi setiap tingkat pendidikan, mulai dari SD, SMP, SMA sampai ke perguruan tinggi. Perpustakaan juga merupakan tempat bagi para pelajar agar dapat menimba ilmu. Pengertian perpustakaan secara umum merupakan sebuah institusi yang bergerak dalam pengelolaan koleksi yang berupa karya cetak, tulis, atau karya lainnya yang memenuhi standar dan kebutuhan para pelajaranya. Kehadiran perpustakaan pada sekolah – sekolah dapat membantu para pelajaranya untuk memperoleh ilmu apa mereka ingin kuasai secara mudah.

Namun seiring dengan perkembangan teknologi informasi terutama dalam era pandemi ini, para pelajar menginginkan kehidupan yang terstruktur, serba cepat dan juga mudah. Itu semata mereka lalui agar setiap pelaksanaan atau kegiatan yang mereka lakukan dapat berjalan dengan lancar. Pengandalan teknologi terutama sistem informasi bermain peran yang besar dalam mewujudkan perkembangan teknologi masa kini. Terkhususnya para pelajar yang akan menjadi penerus bangsa, sudah sepatutnya memiliki banyak fasilitas yang mendukung teknologi komunikasi agar dapat mendukung dalam proses pembelajarannya.

Seiring dengan itu, latar belakang yang kami gunakan akan mengacu pada referensi dari sebuah jurnal yang sudah beredar sebelumnya pada tahun 2016. Karena mengambil referensi dari jurnal yang sudah beredar dan berjudul “SISTEM INFORMASI ADMINISTRASI PERPUSTAKAAN BERBASIS WEB PADA SMA NEGERI 5 TANGERANG”, maka kami akan menggunakan data permasalahan pada jurnal tersebut yang akan kita pergunakan kembali untuk menjadi sebuah penelitian baru.

Melanjutkan dari referensi jurnal yang kami pergunakan, SMA Negeri 5 Tangerang pada saat itu setelah melalui hasil observasi dari peneliti sebelumnya,

diperlukan adanya perhatian khusus untuk mengatasi permasalahan pada sistem perpustakaannya. Meninjau dari penelitian sebelumnya, itu menggunakan sebuah website yang dipergunakan untuk membantu pada permasalahannya, sedangkan pada kelompok kami akan memberikan usul lain yang dapat diterapkan pada permasalahan yang sama juga. Usul yang kami berikan diharapkan dapat memberikan sebuah penerapan teknologi sistem informasi pada perpustakaan.

Usulan dari kelompok kami yaitu akan merancang sebuah program aplikasi dimana program ini akan dipergunakan oleh bagian internal yakni administrator yang dapat mengakses, mengontrol, dan mengatur data para pelajar yang meminjam atau mengembalikan buku dari perpustakaan SMA Negeri 5 Tangerang. Selanjutnya kami akan membangun sebuah program khusus oleh perusahaan K-ONE yang hanya dapat diakses oleh administrator perpustakaan SMA Negeri 5 Tangerang dengan tujuan memantau, mendata, dan mengontrol.

Dengan memberikan usulan ini, diharapkan dapat membantu dan memudahkan petugas perpustakaan dalam mengontrol sirkulasi peminjaman, pengembalian, dan mengelola buku-buku perpustakaan guna mendapatkan hasil pendataan yang tepat dan akurat.

- Data para pelajar (Nama, student id)
- Data jenis buku yang dipilih berdasarkan kode buku
- Data Staff Admin yang bertugas
- Durasi Peminjaman
- Durasi Pengembalian
- Data Denda

1.2 Rumusan Masalah

Dalam Mendesain Aplikasi K-One untuk Perpustakaan SMA Negeri 5 Tangerang ini, agar tidak menyimpang dari tujuan yang hendak dicapai, maka kami dari kelompok 5 akan membatasi ruang lingkup sebagai berikut:

1. Apakah aplikasi ini merancang sistem database Perpustakaan yang mencakup pendataan dan pengolahan data keanggotaan, koleksi perpustakaan, serta data peminjaman dan pengembalian?

2. Bagaimana fitur input, cetak, ubah, dan hapus data yang terdaftar dalam database Perpustakaan sesuai dengan wewenang atau hak akses yang berbeda bagi setiap administrator dalam memanipulasi data-data?

1.3 Tujuan

Tujuan dari manajemen perpustakaan dalam menyusun dan merancang database buku perpustakaan antara lain:

1. Mengembangkan sistem perpustakaan baru untuk SMA Negeri 5 Tangerang
2. Menghindari kemungkinan admin perpustakaan yang kewalahan dalam melayani Pelajar SMA Negeri 5 Tangerang
3. Mempermudah pencarian informasi buku oleh administrator dengan beberapa permasalahan meliputi Judul, Penulis, Tahun Terbit, dan seterusnya
4. Menata data jenis buku dengan jumlah yang tak terhingga yang dimiliki oleh perpustakaan SMA Negeri 5 Tangerang
5. Menampilkan dan mencetak laporan data perpustakaan dengan harapan memperlancar dan mempermudah proses evaluasi di kemudian harinya.

1.4 Profil Perusahaan

Aplikasi K-One merupakan sebuah program aplikasi yang dirancang secara khusus untuk menyediakan program analisa dan pembacaan database perusahaan yang dapat diakses oleh staff internal. Aplikasi ini memiliki harapan untuk dapat mempermudah, memperlancar, dan meningkatkan tingkat kecepatan, ketepatan, dan efisiensi dalam pendataaan, penghapusan, maupun penggalian database yang diperlukan.

K-One merupakan ide pemikiran kami dari kelompok 5 yang akan dipergunakan untuk dapat menyediakan platform berupa program aplikasi yang dapat mendata, menghapus, melacak, memeriksa informasi terkait dengan para pelajar SMA Negeri 5 Tangerang. Disertai juga dengan informasi buku yang tersedia di perpustakaan SMA Negeri 5 Tangerang dengan menggali database yang tersedia dalam sistem kami. Itu semua akan tertampil secara baik ketika pihak internal atau administrator dari SMA Negeri 5 Tangerang mencarinya dalam kolom pencarian.

Sehingga penggunaan dari aplikasi K-One sebagai penyedia platform pencarian database memiliki harapan besar dalam mendorong efisiensi dan ketepatan dalam menghadapi suatu permasalahan internal.

1.5 Proses Aplikasi

1. Pada tampilan awal program aplikasi K-ONE, pilihan log in akan muncul bagi administrator SMA Negeri 5 Tangerang yang telah melakukan Log-In
2. Setelah proses Log-In sukses dilakukan, administrator dapat mengambil alih program aplikasi K-ONE yang akan menampilkan menu berupa button-button dan fitur.
3. Administrator K-ONE dapat mengakses berbagai fitur yang telah disediakan seperti data diri, data jenis buku yang dipilih, durasi waktu peminjaman buku, data staff admin yang bertugas.

BAB II

ANALISIS MASALAH

2.1 Masalah Bisnis

Berdasarkan latar belakang di atas, dapat diketahui beberapa masalah, antara lain:

1. Diperlukannya pembaharuan dari sistem lama menjadi ke sistem baru yang mengikuti perkembangan teknologi sistem informasi
2. Kinerja perangkat keras yang diperlukan harus mendukung karena hal ini dapat membawa dampak yang besar ke dalam sebuah database
3. Memerlukan kapasitas yang besar untuk menampung database para siswa

2.2 System Requirements

Sebuah system requirements merupakan fase fundamental dari pengembangan sistem. Perusahaan K-ONE merupakan salah satu instansi mandiri yang menjadi sumber yang mencakup semua bidang pengelolaan. Bagian pengelolaan merupakan salah satu bidang penting yang memiliki informasi tentang perkembangan perpustakaan, metode pendataan dan pemeliharaan data di perpustakaan SMA Negeri 5 Tangerang.

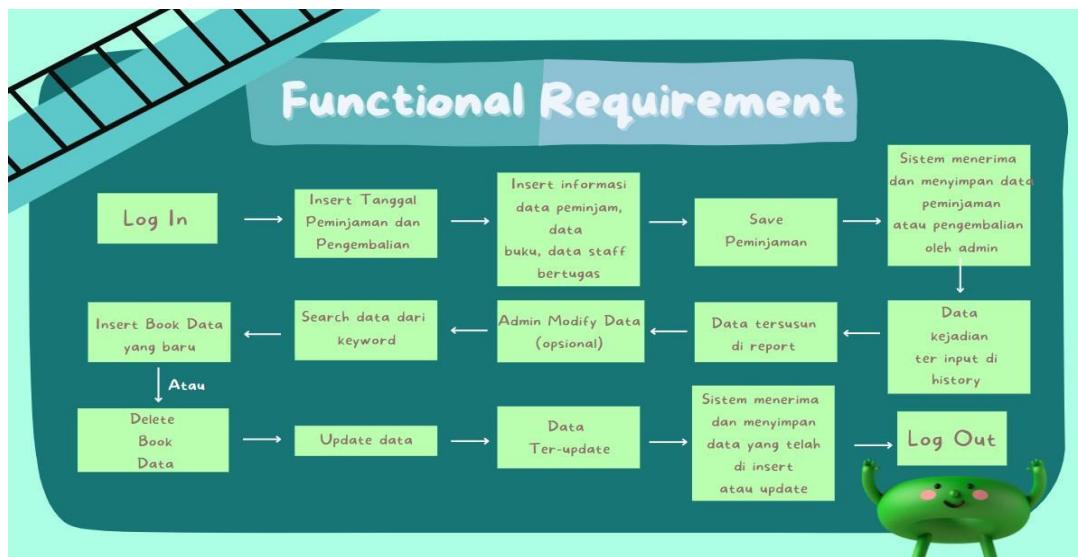
Pelaporan acara menjadi salah satu kegiatan inti di departemen ini. Sering kali terjadi kebingungan dalam pelaporan data dan ketidakefisienan waktu karena harus melakukan rekap data peminjaman buku sebanyak dua kali ataupun lebih. Berbagi data juga sering menjadi kendala mengakibatkan informasi yang kurang up to date. Mengingat sistem berbasis SOA, dapat memaksimalkan aktivitas pelaporan di bidang pengelolaan ini. Oleh karena itu dilakukan penelitian untuk menganalisis kebutuhan, merancang sistem, memastikan konsistensi, dan memastikan memenuhi beberapa kriteria service-oriented architecture dalam system requirements.

Studi ini dimulai dengan data. Analisis kebutuhan menghasilkan use case sebagai definisi dari sistem fungsional persyaratan dan persyaratan tambahan sebagai definisi persyaratan sistem non-fungsional. Perancangan sistem menghasilkan desain artefak, meliputi: sketsa antarmuka pengguna, diagram komunikasi, diagram kelas buku, diagram urutan peminjaman dan pengembalian, dan skema database Pengujian konsistensi mendefinisikan persyaratan menghasilkan nilai RCI (Realtime Compression Interface) 100%, yang berarti mendefinisikan kebutuhan sistem adalah 100% konsisten. Pengujian uji laksus menghasilkan desain yang memenuhi keselarasan bisnis dan memenuhi kriteria dapat digunakan kembali.

2.2.1 Functional Requirements

Dalam mengembangkan K-ONE, persyaratan fungsional alias functional requirements diperlukan untuk mendeterminasikan apabila keseluruhan fungsi dari K-ONE memiliki performa yang baik. K-ONE dapat dikatakan memenuhi persyaratan fungsional apabila memiliki sifat seperti dibawah ini:

- A. Deskripsi proses berjalannya system K-ONE
- B. Rincian Input data – system behaviour – data output antar interaksi user dengan system
- C. Data dapat terkalkulasi oleh system
- D. Data yang dapat tersimpan oleh system
- E. Data dapat termanipulasi oleh system
- F. Mendukung proses bisnis
- G. Mendukung interaksi user



Gambar 2.1. lampiran ilustrasi funtional requirement

Berikut merupakan jenis dari functional requirements dari K-ONE:

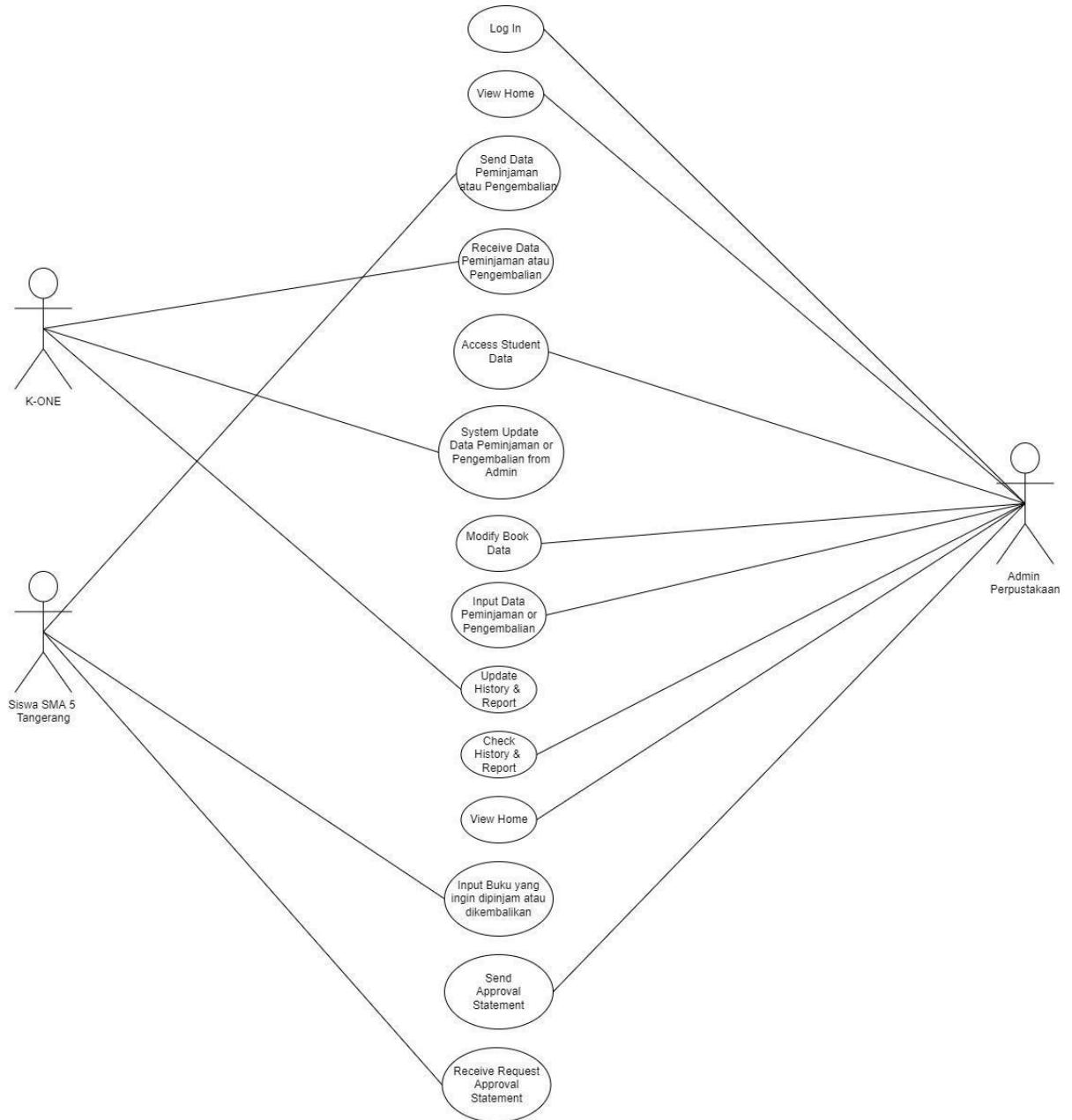
1. Business Requirements: Goal yang ingin dicapai dari project K-ONE, ekspektasi yang dimiliki untuk K-ONE, keuntungan yang dibawa oleh K-ONE, future scope K-ONE
2. User requirements: keterlibatan user dan apa yang user perlukan oleh user K-ONE, aktivitas yang user K-ONE dapat dilakukan dengan system K-ONE, masalah yang dihadapi oleh system, persyaratan yang diperlukan untuk mengakses jenis system K-ONE
3. System require
4. ments: action system K-ONE, spesifikasi software dan hardware untuk K-ONE

2.2.2 Non-Functional Requirements

Non-Functional Requirements memberikan gambaran mengenai hubungan kriteria system K-ONE dengan efektifitas dan performa kerja system alias kinerja, waktu respon untuk update data dari system, waktu respon untuk mengambil Kembali atau retrieve data dari system, mengantisipasi volume dari data, dan juga keamanan yang dipertimbangkan termasuk:

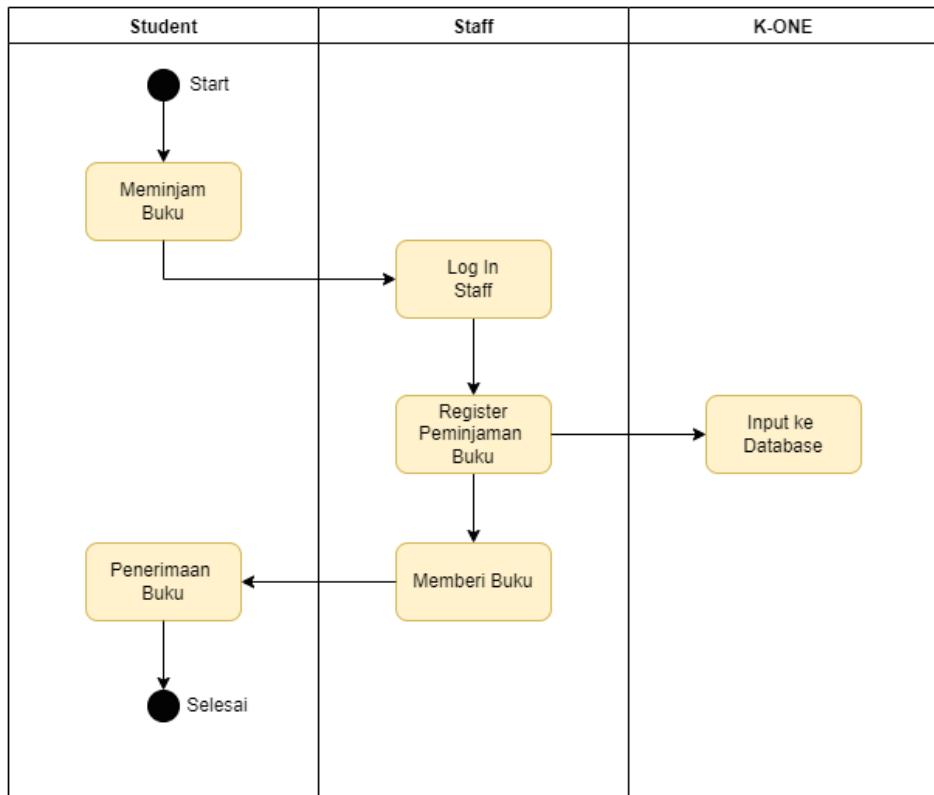
- A. Security Sistem: akan memberikan akses kepada pengguna yang sudah terdaftar, yang bertujuan hanya orang yang mempunyai akses yang dapat menggunakannya.
- B. Performance: Mempunyai batasan mengenai kecepatan operasional sebuah aplikasi
- C. Reliability System: dari aplikasi harus bersifat dapat diandalkan dikarenakan data perpustakaan yang dipegang sangat penting
- D. Availability: Aplikasi ini dapat digunakan setiap 8 jam sehari dan 5 hari dalam seminggu
- E. Maintainability: Memodifikasi perubahan seperti admin baru, penggantian kata sandi, perubahan database yang harus diverifikasi setidaknya sekali setiap harinya secara automatis

2.3 Use Case Diagram

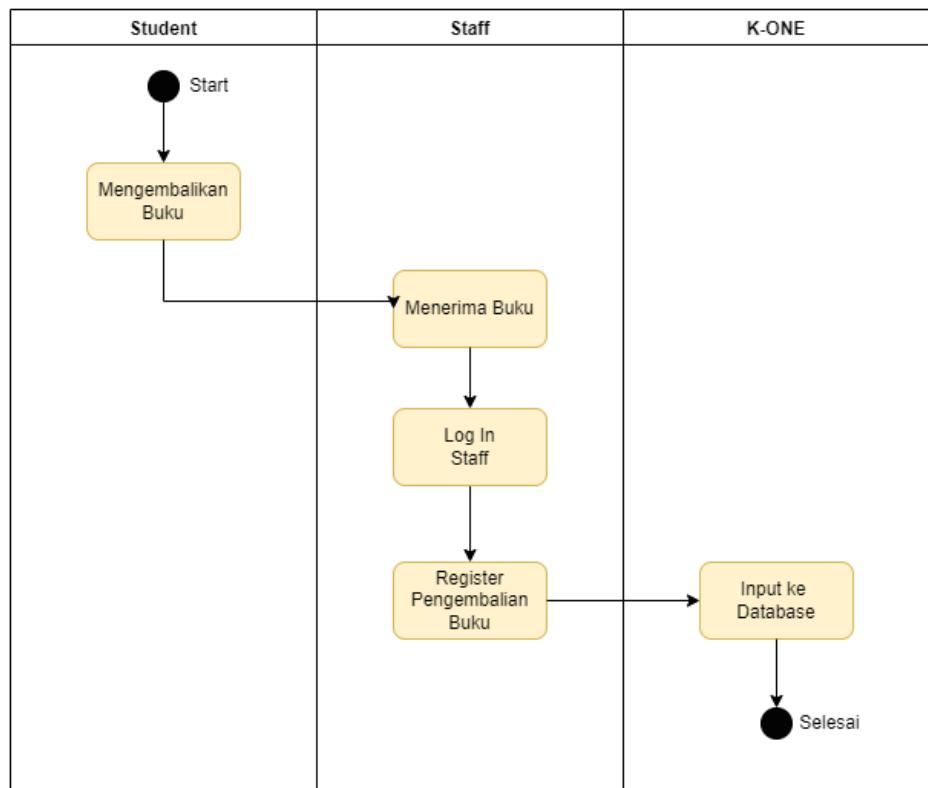


Gambar 2.2 Use Case Diagram

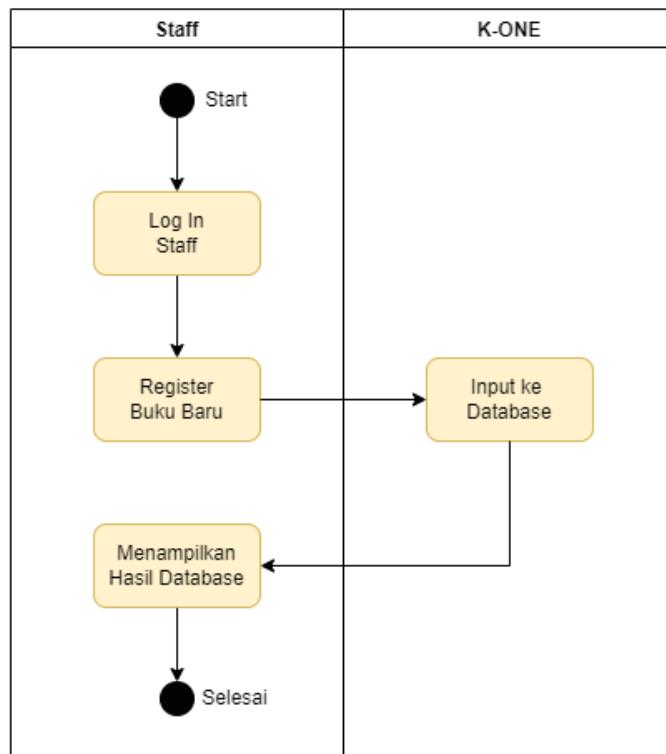
2.4 Activity Diagram



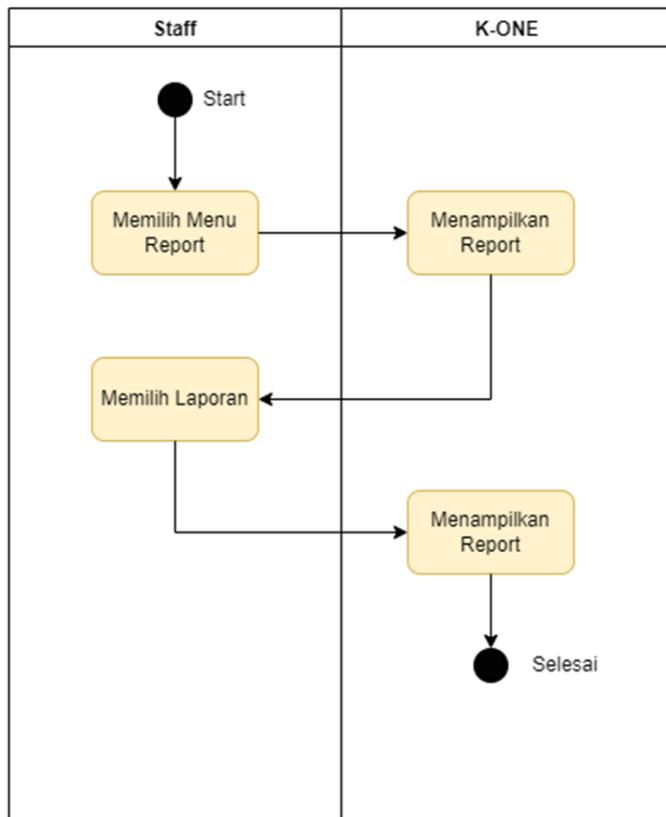
Gambar 2.3. Activity Diagram Peminjaman Buku



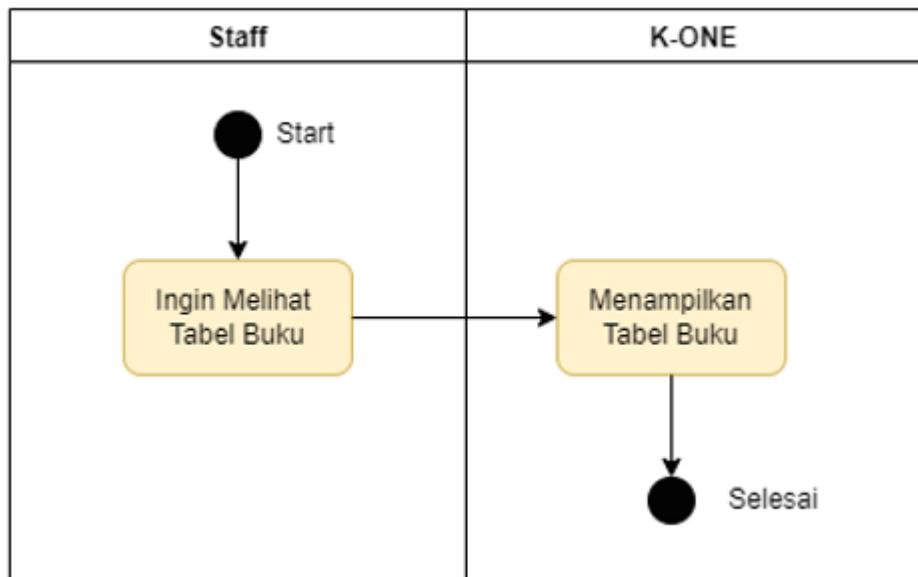
Gambar 2.4. Activity Diagram Pengembalian Buku



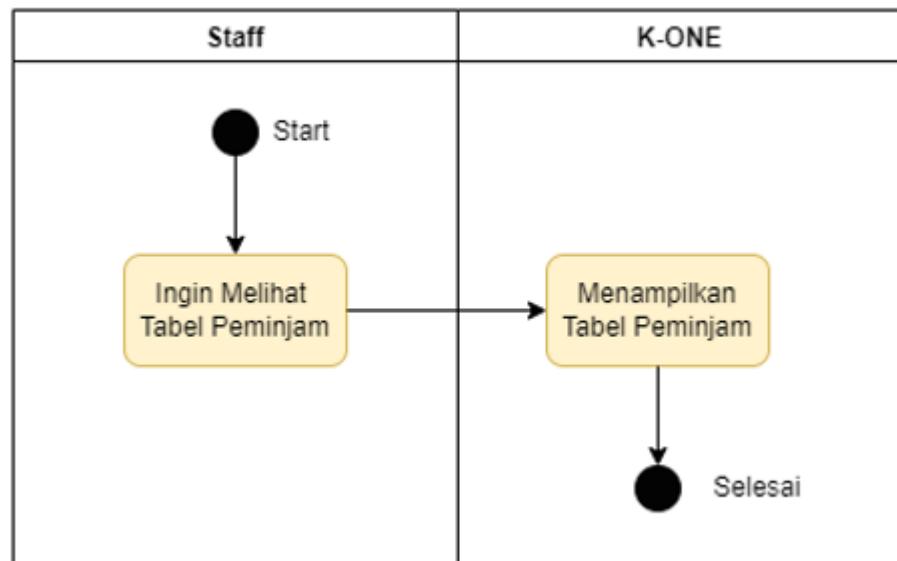
Gambar 2.5. Activity Diagram Pendaftaran Buku Baru



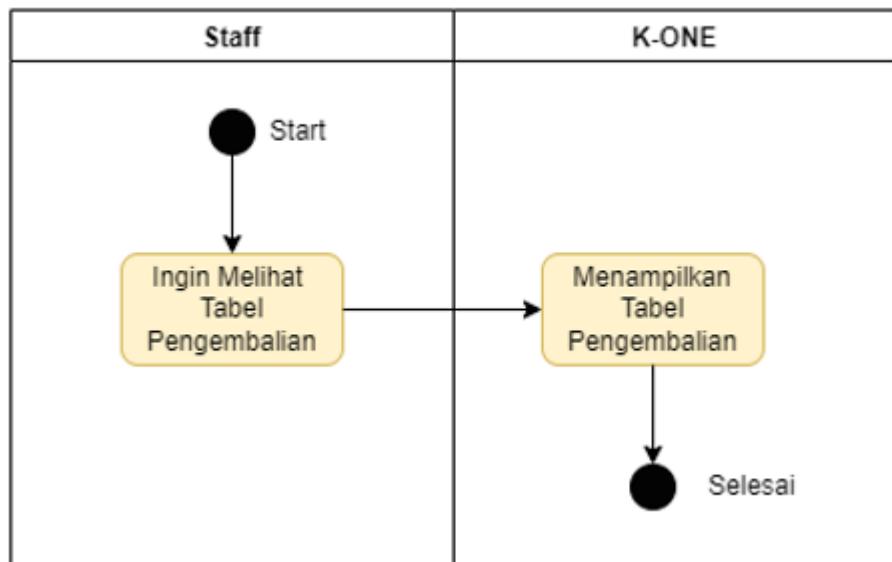
Gambar 2.6. Activity Diagram Laporan Data Perpustakaan



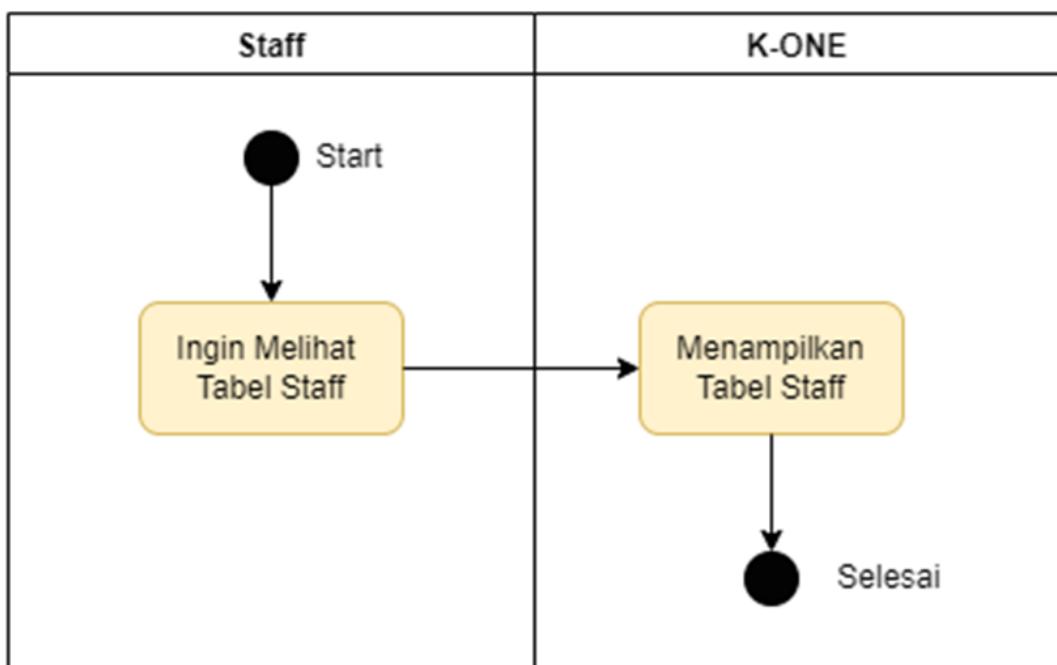
Gambar 2.7. Activity Diagram Review Data Buku



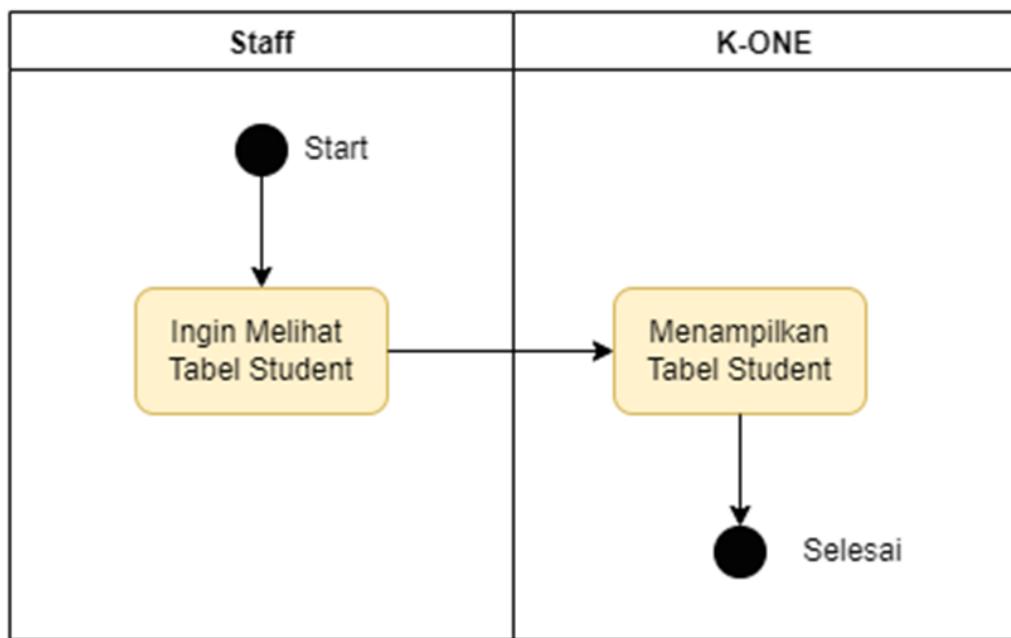
Gambar 2.8. Activity Diagram Review Data Peminjam



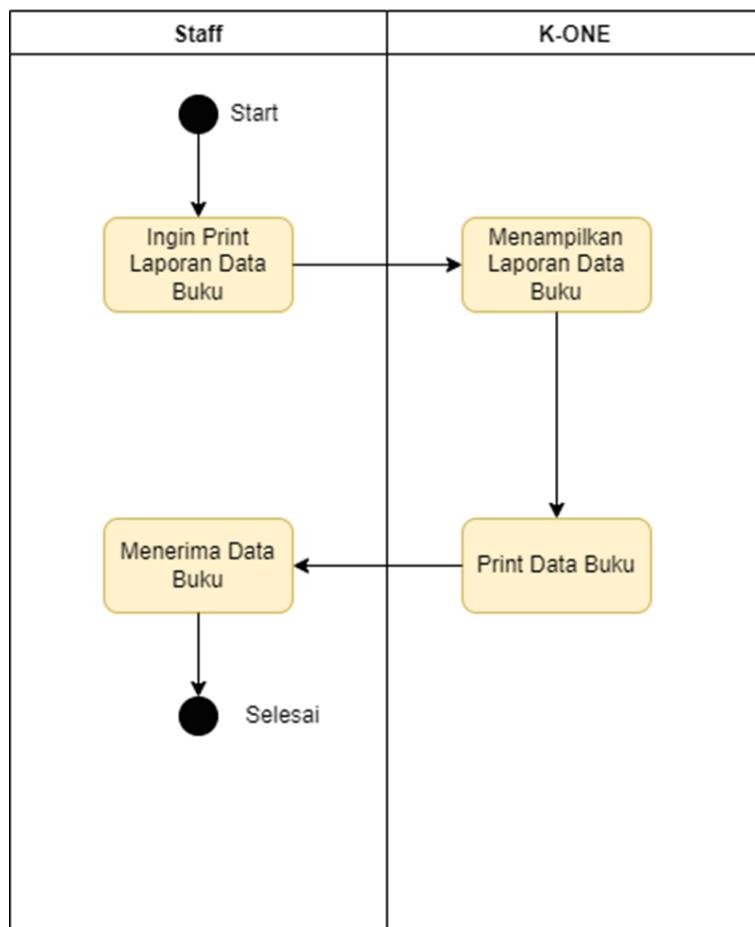
Gambar 2.9. Activity Diagram Review Data Pengembalian



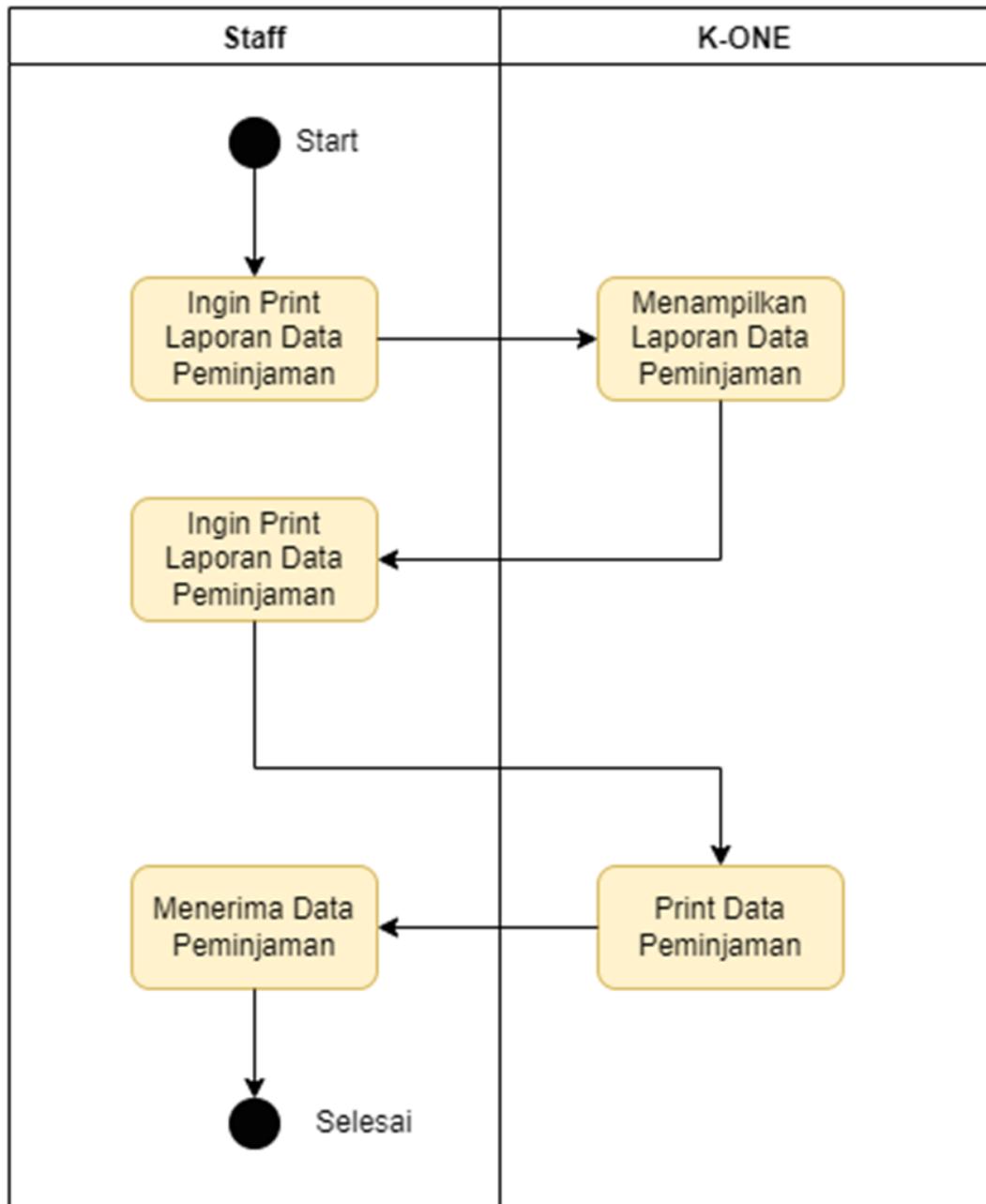
Gambar 2.10. Activity Diagram Review Data Staff



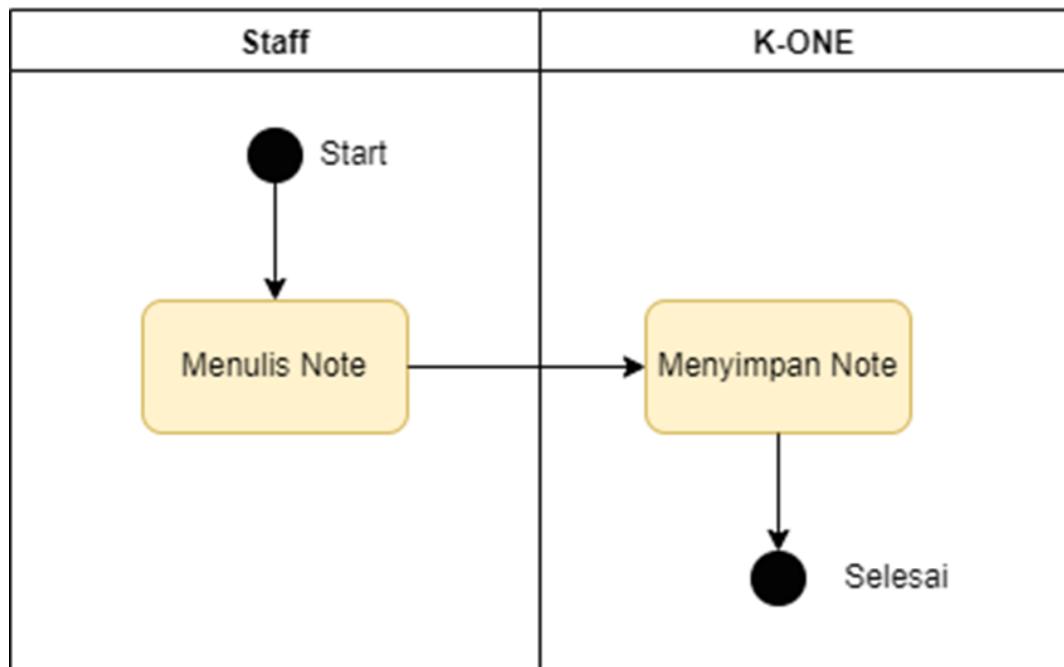
Gambar 2.11. Activity Diagram Review Data Student



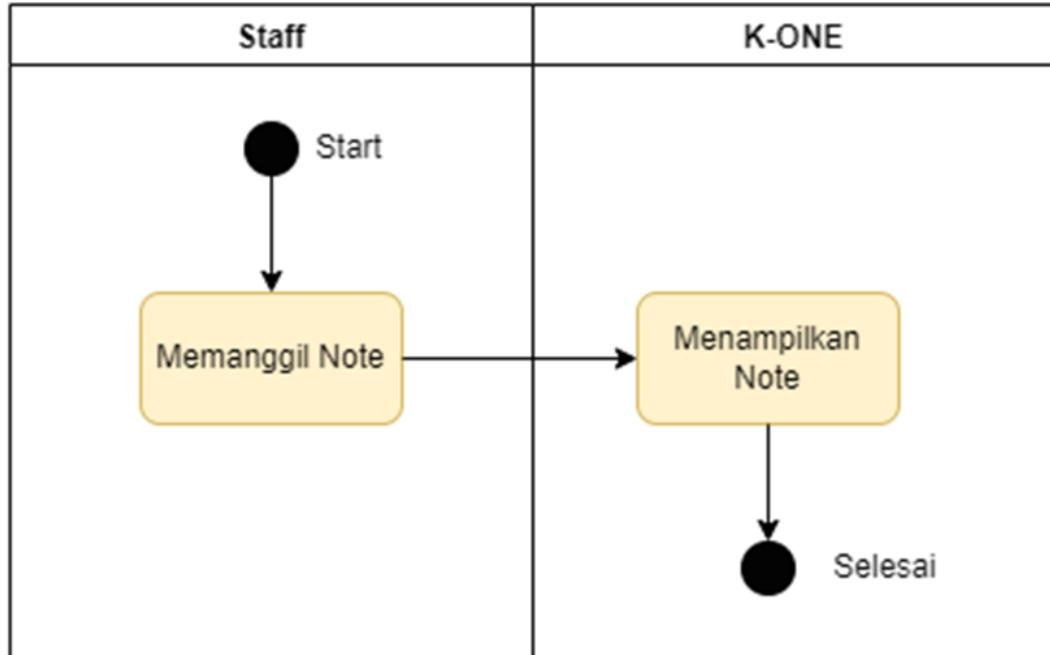
Gambar 2.12. Activity Diagram Print Data Buku



Gambar 2.13. Activity Diagram Print Data Peminjam



Gambar 2.14. Activity Diagram Menulis Note



Gambar 2.15. Activity Diagram Memanggil Note

BAB III

PEMBAHASAN

3.1 Proses Normalisasi

3.1.1 UNF (Unnormalized Form)

- Perpustakaan (Id_Student, Nama_Student, Alamat, No_Hp, {No_Pinjam, Tanggal_Pinjam, No_Kembali, Tanggal_Kembali, Jumlah_Denda, Id_Buku, Kategori, Judul_Buku, Penerbit, Kuantitas} Id_Staff, Nama_Staff, Gender_Staff, Username, Password_Staff)

Id_Student	Nama_Student	Alamat	No_Hp	No_Pinjam	Tanggal_Pinjam	No_Kembali	Tanggal_Kembali	Jumlah_Denda	Id_Buku
801	Anton	Jl. Boulevard Raya 1 Blok CA No.1	08123456789	0001	1 Februari 2022	7001	31 April 2022	Rp. 20.000	00432
802	Budil	Jl. Kampung Rambutan No.9	08165895574	0002	20 Maret 2022	7002	27 Maret 2022	-	00744
803	Budul	Jl. Kursi Goyang No 2	08187659884	0003	17 Maret 2022	7003	18 Maret 2022	-	00124
Kategori	Judul_buku	Penerbit	Kuantitas	Id_Staff	Nama_Staff	Gender_Staff	Username	Password_Staff	
Ensiklopedia	Ensiklopedia Nusantara I	Erlangga	2	10001	Angel	P	AngelFly	Fly842	
Novel Ringan	Laskar Pelangi	Gramedia	1	10002	Agim	L	Larith	Coklat30	
Buku Pelajaran	Ilmu Pengetahuan Sosial Kelas -11	DimasBook	1	10003	Alvin	L	veen444	Vin444	

Gambar 3.1 Perpustakaan Unnormalized Form

(tabel atas dan bawah merupakan 1 tabel dengan 1 row yang panjang, bukan 2 tabel)

3.1.2 1NF (First Normal Form)

- Perpustakaan (Id_Student, Nama_Student, Alamat, No_Hp, Id_Staff, Nama_Staff, Gender_Staff, Username, Password_Staff)

Id_Student	Nama_Student	Alamat	No_Hp	No_Pinjam	Tanggal_Pinjam	No_Kembali	Tanggal_Kembali	Jumlah_Denda
801	Anton	Jl. Boulevard Raya 1 Blok CA No.1	08123456789	0001	1 Februari 2022	7001	31 April 2022	Rp. 20.000
801	Anton	Jl. Boulevard Raya 1 Blok CA No.1	08123456789	0001	1 Februari 2022	7001	31 April 2022	Rp. 20.000
802	Budil	Jl. Kampung Rambutan No.9	08165895574	0002	20 Maret 2022	7002	27 Maret 2022	-
803	Budul	Jl. Kursi Goyang No 2	08187659884	0003	17 Maret 2022	7003	18 Maret 2022	-

Gambar 3.2 Perpustakaan First Normalized Form

- Peminjaman Buku (No_Pinjam, Tanggal_Pinjam, No_Kembali, Tanggal_Kembali, Jumlah_Denda, Id_Buku, Kategori, Judul_Buku, Penerbit, Kuantitas)

Id_Buku	Kategori	Judul_buku	Penerbit	Kuantitas	Id_Staff	Nama_Staff	Gender_Staff	Username	Password_Staff
00432	Ensiklopedia	Ensiklopedia Nusantara I	Erlangga	2	10001	Angel	P	AngelFly	Fly842
00433	Ensiklopedia	Ensiklopedia Nusantara II	Erlangga	2	10001	Angel	P	AngelFly	Fly842
00744	Novel Ringan	Laskar Pelangi	Gramedia	1	10002	Agim	L	Larith	Coklat30
00124	Buku Pelajaran	Ilmu Pengetahuan Sosial Kelas -11	DimasBook	1	10003	Alvin	L	veen444	Vin444

Gambar 3.3 Peminjaman Buku First Normalized Form

3.1.3 2NF (Second Normal Form)

- Student (Id_Student, Nama_Student, Alamat, No_HP, No_Pinjam)

Id_Student	Nama_Student	Alamat	No_Hp
801	Anton	Jl. Boulevard Raya 1 Blok CA No.1	08123456789
801	Anton	Jl. Boulevard Raya 1 Blok CA No.1	08123456789
802	Budil	Jl. Kampung Rambutan No.9	08165895574
803	Budul	Jl. Kursi Goyang No 2	08187659884

Gambar 3.4 Student Second Normalized Form

- Staff (Id_Staff, Nama_Staff, Gender_Staff, Username, Password_Staff)

Id_Staff	Nama_Staff	Gender_Staff	Username	Password_Staff
10001	Angel	P	AngelFly	Fly842
10001	Angel	P	AngelFly	Fly842
10002	Agim	L	Larith	Coklat30
10003	Alvin	L	veen444	Vin444

Gambar 3.5 Staff Second Normalized Form

- Peminjaman (No_Pinjam, Tanggal_Pinjam)

No_Pinjam	Tanggal_pinjam
0001	1 Februari 2022
0001	1 Februari 2022
0002	20 Maret 2022
0003	17 Maret 2022

Gambar 3.6 Peminjaman Second Normalized Form

- Pengembalian (No_Kembali, Tanggal_Kembali, Jumlah_Denda)

No_Kembali	Tanggal_kembali	Jumlah_denda
7001	31 April 2022	Rp. 20.000
7001	31 April 2022	Rp. 20.000
7002	27 Maret 2022	-
7003	18 Maret 2022	-

Gambar 3.7 Pengembalian Second Normalized Form

- Buku (Id_Buku, Kategori, Judul_Buku, Penerbit, Kuantitas)

Id_Buku	Kategori	Judul_buku	Penerbit	Kuantitas
00432	Ensiklopedia	Ensiklopedia Nusantara I	Erlangga	2
00433	Ensiklopedia	Ensiklopedia Nusantara II	Erlangga	2
00744	Novel Ringan	Laskar Pelangi	Gramedia	1
00124	Buku Pelajaran	Ilmu Pengetahuan Sosial Kelas -11	DimasBook	1

Gambar 3.8 Buku Second Normalized Form

3.1.4 3NF (Third Normal Form)

- Student (Id_Student, Nama_Student, Alamat, No_HP, No_Pinjam, No_Kembali)

Id_Student	Nama_Student	Alamat	No_Hp	No_Kembali	No_Pinjam
801	Anton	Jl. Boulevard Raya 1 Blok CA No.1	08123456789	7001	0001
802	Budil	Jl. Kampung Rambutan No.9	08165895574	7002	0002
803	Budul	Jl. Kursi Goyang No 2	08187659884	7003	0003

Gambar 3.9 Student Third Normalized Form

- Staff (Id_Staff, Nama_Staff, Gender_Staff, Username, Password_Staff)

Id_Staff	Nama_Staff	Gender_Staff	Username	Password_Staff
10001	Angel	P	AngelFly	*****
10002	Agim	L	Larith	*****
10003	Alvin	L	veen444	*****

Gambar 3.10 Staff Third Normalized Form

- Peminjaman (No_Pinjam, Tanggal_Pinjam, Id_Buku, Id_Staff)

No_Pinjam	Tanggal_pinjam	Id_Staff	Id_Buku
0001	1 Februari 2022	10001	00432
0002	20 Maret 2022	10002	00744
0003	17 Maret 2022	10003	00124

Gambar 3.11 Peminjaman Third Normalized Form

- Pengembalian (No_Kembali, Tanggal_Kembali, Id_Buku, Jumlah_Denda)

No_Kembali	Tanggal_kembali	Id_Staff	Id_Buku	Jumlah_denda
7001	31 April 2022	10001	00432	Rp. 20.000
7002	27 Maret 2022	10002	00744	-
7003	18 Maret 2022	10003	00124	-

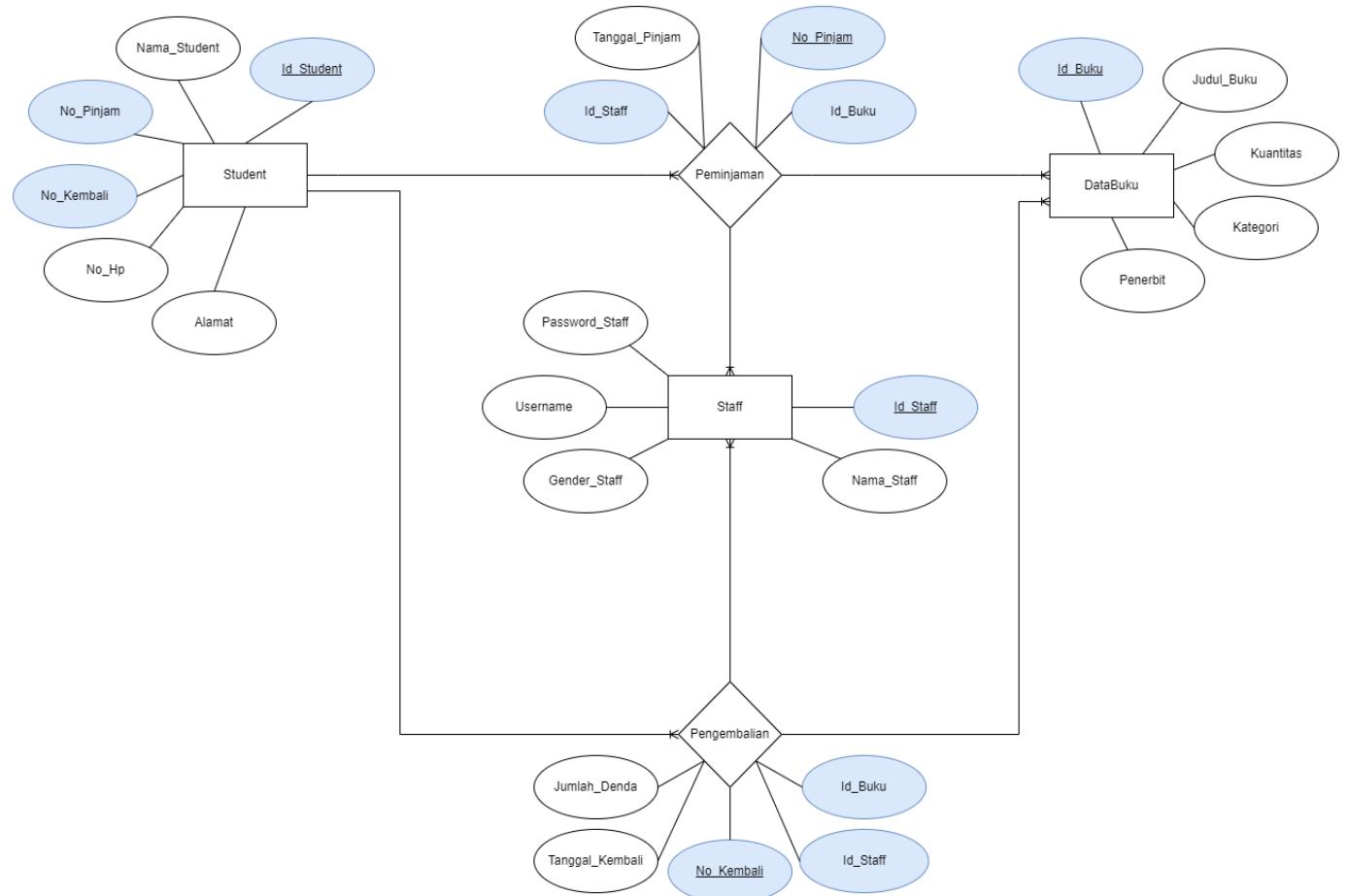
Gambar 3.12 Pengembalian Third Normalized Form

- Buku (Id_Buku, Kategori, Judul_Buku, Penerbit, Kuantitas)

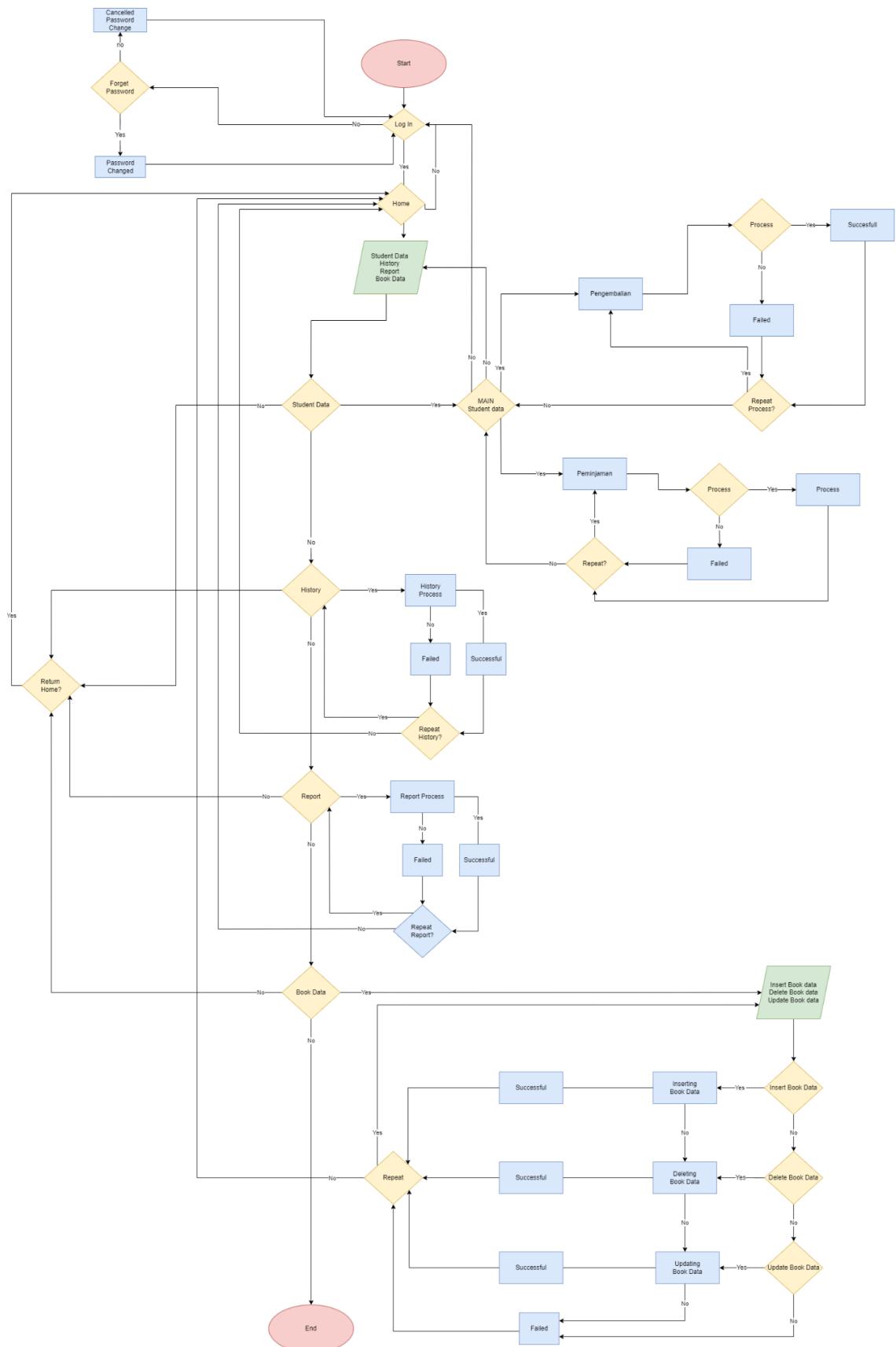
Id_Buku	Judul_buku	Kuantitas	Kategori	Penerbit
00432	Ensiklopedia Nusantara I	2	Ensiklopedia	Erlangga
00433	Ensiklopedia Nusantara II	2	Ensiklopedia	Erlangga
00744	Laskar Pelangi	1	Novel Ringan	Gramedia
00124	Ilmu Pengetahuan Sosial Kelas -11	1	Buku Pelajaran	DimasBook

Gambar 3.13 Pengembalian Third Normalized Form

3.2 Entity Relationship Diagram (Erd)



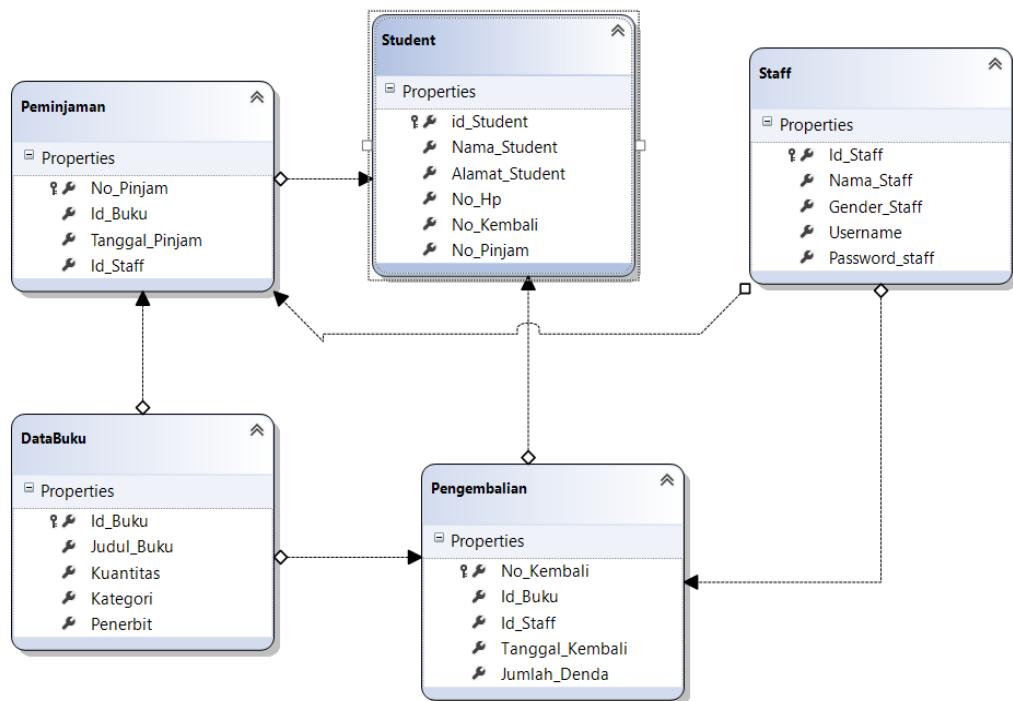
3.3 Flow Chart



BAB IV

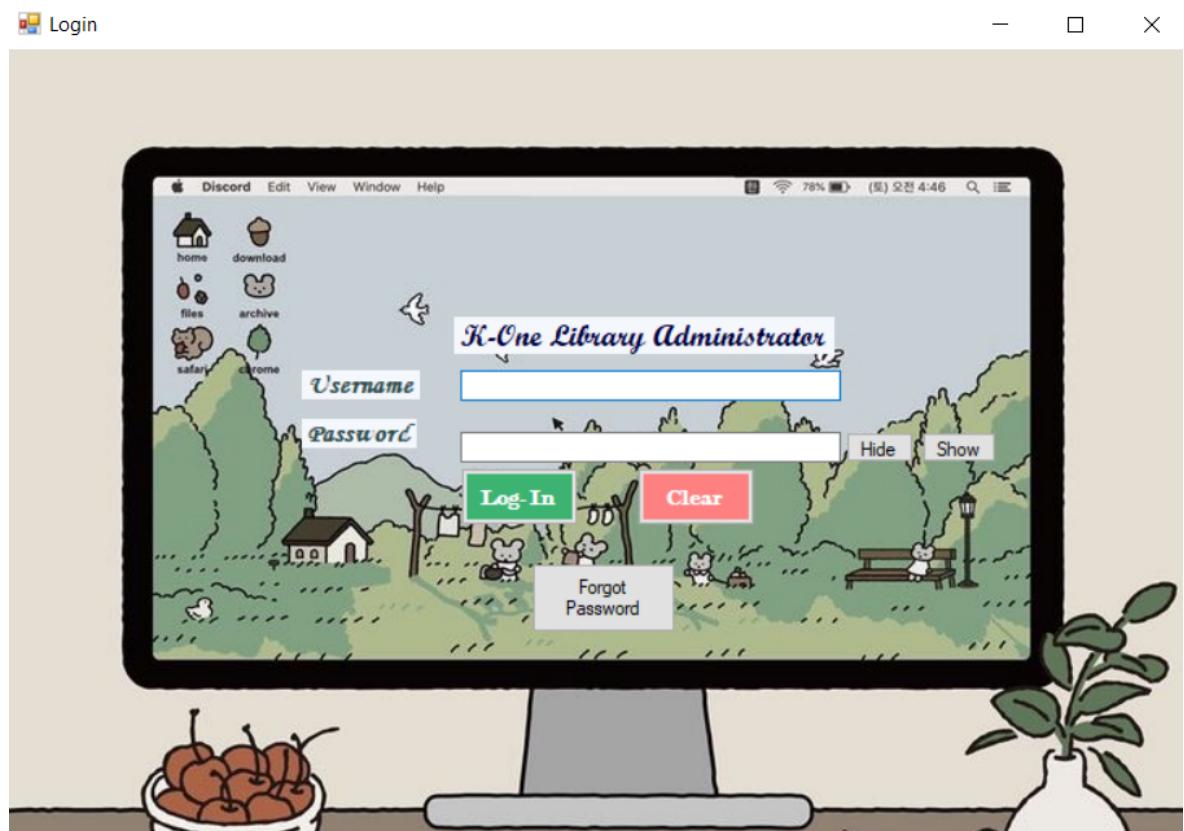
HASIL DAN KESIMPULAN

4.1 Tabel Relasi



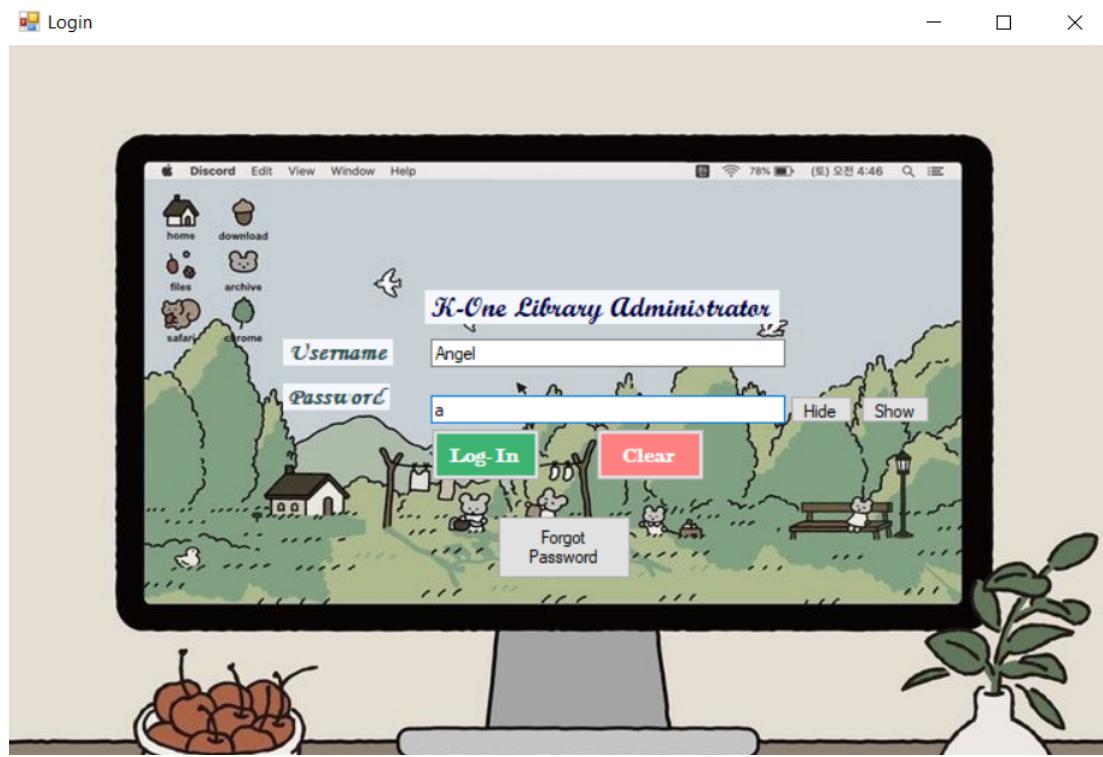
Gambar 4.1. Tabel Relasi

4.2 Penjelasan Dan Gambar User Interface Aplikasi



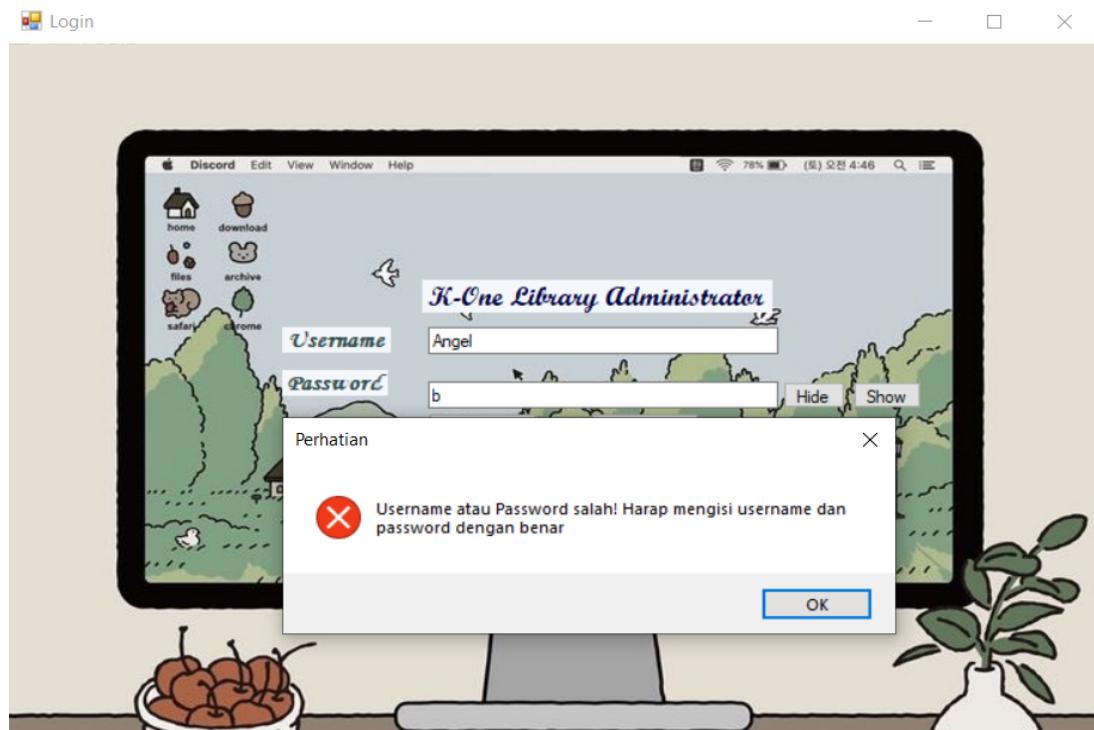
Gambar 4.1. Tampilan log-in administrator

Administrator perpustakaan SMA 5 Tangerang yang bertugas memiliki akses untuk masuk ke program aplikasi k-one untuk mendukung pendataan keluar masuk buku dari perpustakaan. Berikut yang ditampilkan di laman log-in merupakan laman yang dapat digunakan bagi para administrator untuk memverifikasi dirinya dengan cara memasukkan username dan kata sandi akses yang benar.



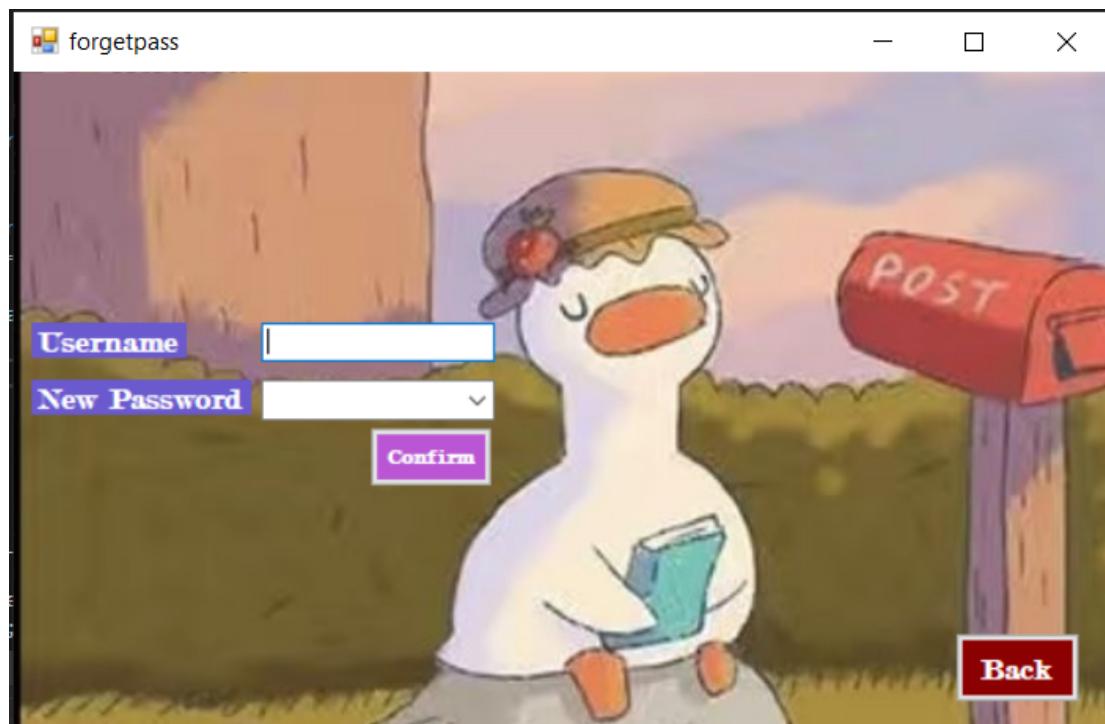
Gambar 4.2. Administrator memasukkan username dan password yang sudah tercatat dalam database log-in.

Nantinya aplikasi akan membaca character yang telah dimasukkan. Apabila sudah benar maka administrator akan di redirect langsung ke homepage dan apabila username atau password masih salah, maka administrator akan mendapatkan notifikasi pop-up berupa *password* yang dimasukkan tidak tepat.



Gambar 4.3 pop-up notifikasi password tidak tepat

Apabila administrator menginput username atau password yang tidak sesuai dengan data di database, maka akan muncul pop-up “perhatian” yang memberikan notifikasi bahwa username atau password yang telah dimasukkan salah



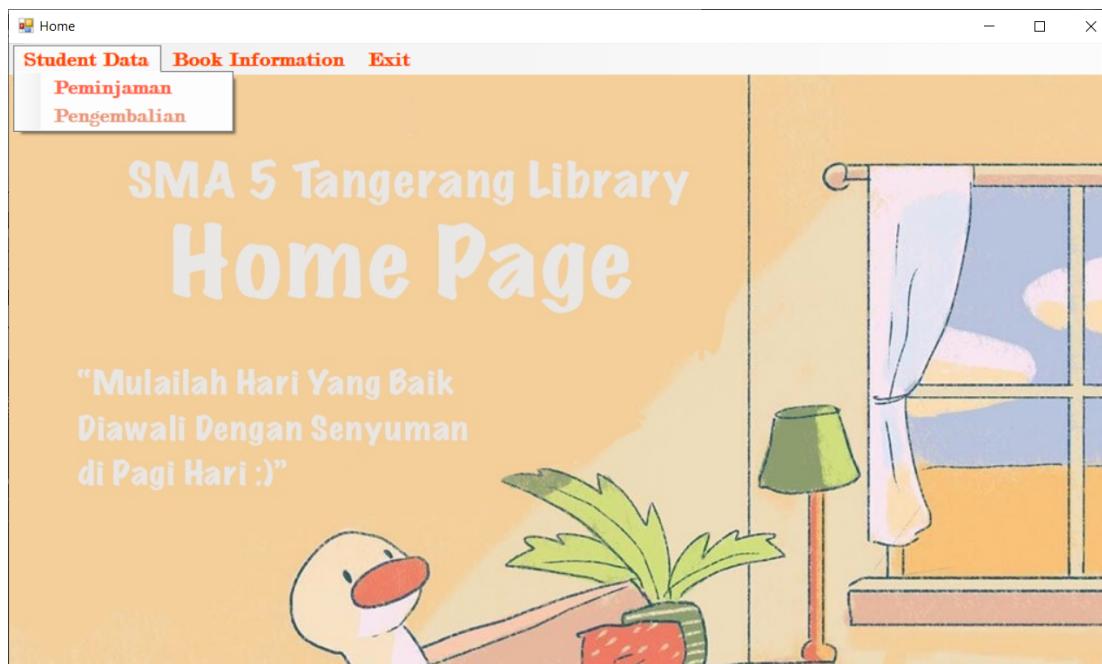
Gambar 4.4. laman reset password

Pada laman ini, administrator yang tidak mengingat password mereka dapat melakukan perubahan password melalui laman ini dimana password baru nya akan terinput ke dalam database sehingga administrator dapat mengakses program aplikasi kembali



Gambar 4.5. laman homepage administrator

Setelah administrator memasukkan input username dan password yang tepat, admin akan di re-direct menuju halaman homepage k-one. Halaman homepage merupakan laman yang digunakan untuk mengakses student data, book information, dan exit.



Gambar 4.6. laman student data

Pada laman student data, terdapat dua pilihan bagi administrator dimana mereka dapat mengakses forum penginputan peminjaman dan forum penginputan pengembalian buku.

	Id_Buku	Judul_Buku	Kuantitas	Kategori	Penerbit
1	00124	Ilmu Pengetahua...	1	Buku Pelajaran	DimasBook
	00432	Ensiklopedia Nus...	2	Ensiklopedia	Erlangga
	00744	Laskar Pelangi	1	Novel Ringan	Gramedia

Gambar 4.7. laman input peminjaman buku

Pada laman ini, administrator dapat menginput data yang diperlukan untuk memenuhi kriteria pencatatan peminjaman buku. Administrator diwajibkan untuk mengisi semua kolom data, berikut merupakan penjelasannya.

No Pinjam

ID Buku: 00124

Tanggal Pinjam: Sunday, June 5.

Id Staff: 10002

Id Student: 0001

Generate

Gambar 4.8 laman re-direct setelah save peminjaman

Setelah administrator menginput data yang wajib diisi untuk mendata aktivitas peminjaman siswa, administrator dapat men-generate angka acak 5 digit dimana setiap angka generate nya dipastikan berbeda.

No Pinjam

49603

ID Buku: 00124

Tanggal Pinjam: Sunday, June 5.

Id Staff: 10002

Id Student: 0001

Generate

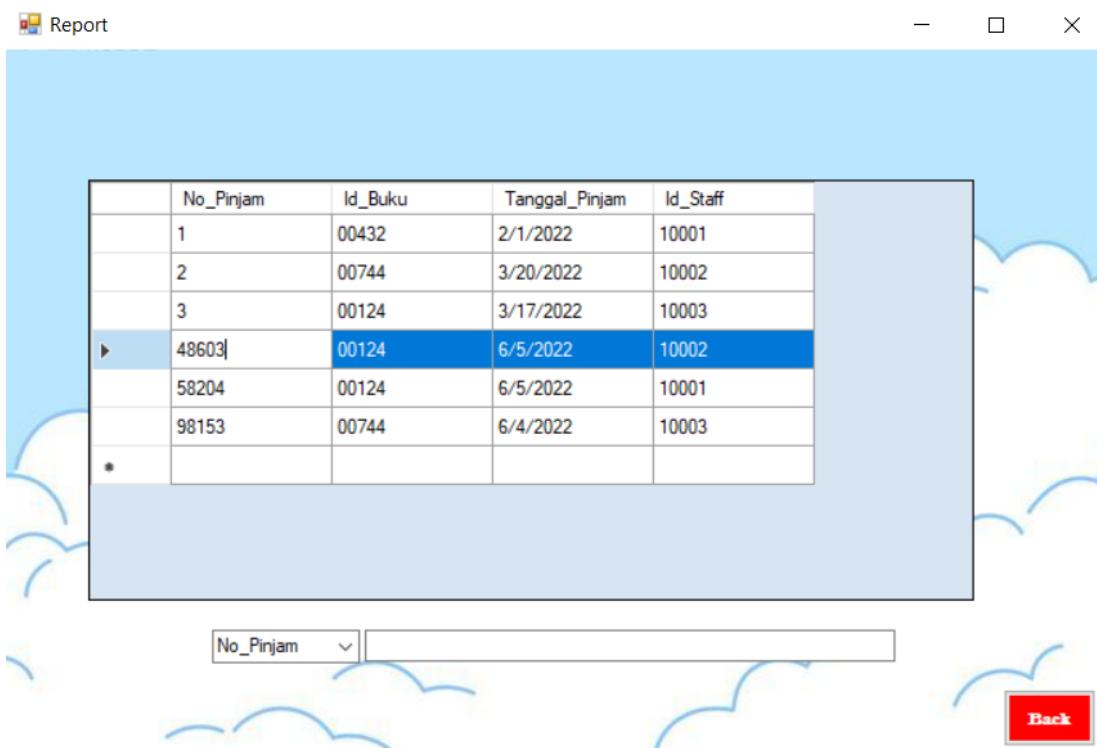
Gambar 4.9. laman peminjaman yang menunjukkan contoh tampilan angka acak.

Berikut merupakan tampilan dari angka acak yang dihasilkan oleh generator. Angka ini yang selalu berbeda digunakan dengan tujuan mendata setiap peminjaman dengan nomor unik yang berbeda.



Gambar 4.10. laman peminjaman yang menunjukkan pop up save berhasil.

Pada proses ini, administrator telah berhasil melakukan penyimpanan atas data peminjaman yang telah di-input ke dalam database. Tombol search hanyalah tombol opsional untuk mencari keywords yang ingin ditemukan oleh administrator.



Gambar 4.11. laman report

Laman ini akan muncul dengan otomatis ketika administrator telah melakukan save peminjaman. laman report ini digunakan untuk memastikan kembali apabila data yang telah dimasukkan oleh administrator sudah benar apa adanya.

No_Pinjam	Id_Buku	Tanggal_Pinjam	Id_Staff	Id_Student
1	00432	2/1/2022	10001	801
2	00744	3/20/2022	10002	802
3	00124	3/17/2022	10003	803
48603	00124	6/5/2022	10002	1

Gambar 4.12. laman pengembalian buku perpustakaan SMA 5 Tangerang.

Pada laman ini, tampilan menunjukkan format pengisian yang wajib di-input oleh administrator atas segala aktifitas yang berhubungan dengan pengembalian buku perpustakaan oleh siswa atau siswi SMA 5 Tangerang. Data meliputi nomor identitas buku, nomor identitas administrator, tanggal buku dikembalikan, jumlah denda, dan nomor identitas siswa atau siswi.

The screenshot shows a form titled "Pengembalian" (Return) with a background illustration of a cartoon character. The form fields include:

- No Kembali: 26481 (highlighted in red)
- Id Buku: 00432
- Id Staff: 10001
- Tanggal Kembali: Sunday , June 5, ✓
- Jumlah Denda: 0
- Id Student: 0001
- Buttons: "Generate" (disabled), "Save Pengembalian" (green), "Clear Search" (orange)

Gambar 4.13. laman menunjukkan hasil angka acak oleh generator.

Setelah mengisi semua data yang wajib di-input, administrator dapat melakuka generate random numbers yang dipastikan tidak akan pernah menghasilkan acakan angka yang sama dengan tujuan membuat kode unik untuk setiap peminjam yang ingin dan akan di-input.

The screenshot shows a "Halaman Pengembalian" (Return Page) with a table of borrowed books and a return form. The table has columns: No_Pinjam, Id_Buku, Tanggal_Pinjam, Id_Staff, and Id_Student. The data is as follows:

No_Pinjam	Id_Buku	Tanggal_Pinjam	Id_Staff	Id_Student
1	00432	2/1/2022	10001	801
2	00744	3/20/2022	10002	802
3	00124	3/17/2022	10003	803
48603	00124	6/5/2022	10002	1

The return form fields are identical to the one in Figure 4.13, with "No Kembali" set to 26481. A modal dialog box is open, displaying "Save Pengembalian Berhasil" (Return saved successfully) with an "OK" button.

Gambar 4.14. laman penyimpanan pengembalian sukses.

Dalam halaman ini, setelah administrator melakukan save pengembalian atas data yang telah dipastikan, administrator akan mendapatkan notifikasi pop-up bahwa

save pengembalian telah berhasil dan akan masuk secara otomatis kedalam database.



Gambar 4.15. ilustrasi tampilan pilihan book information.

Para administrator yang ingin mengakses book information akan mendapatkan pilihan data yakni data view, report, book data, laporan, dan notes.



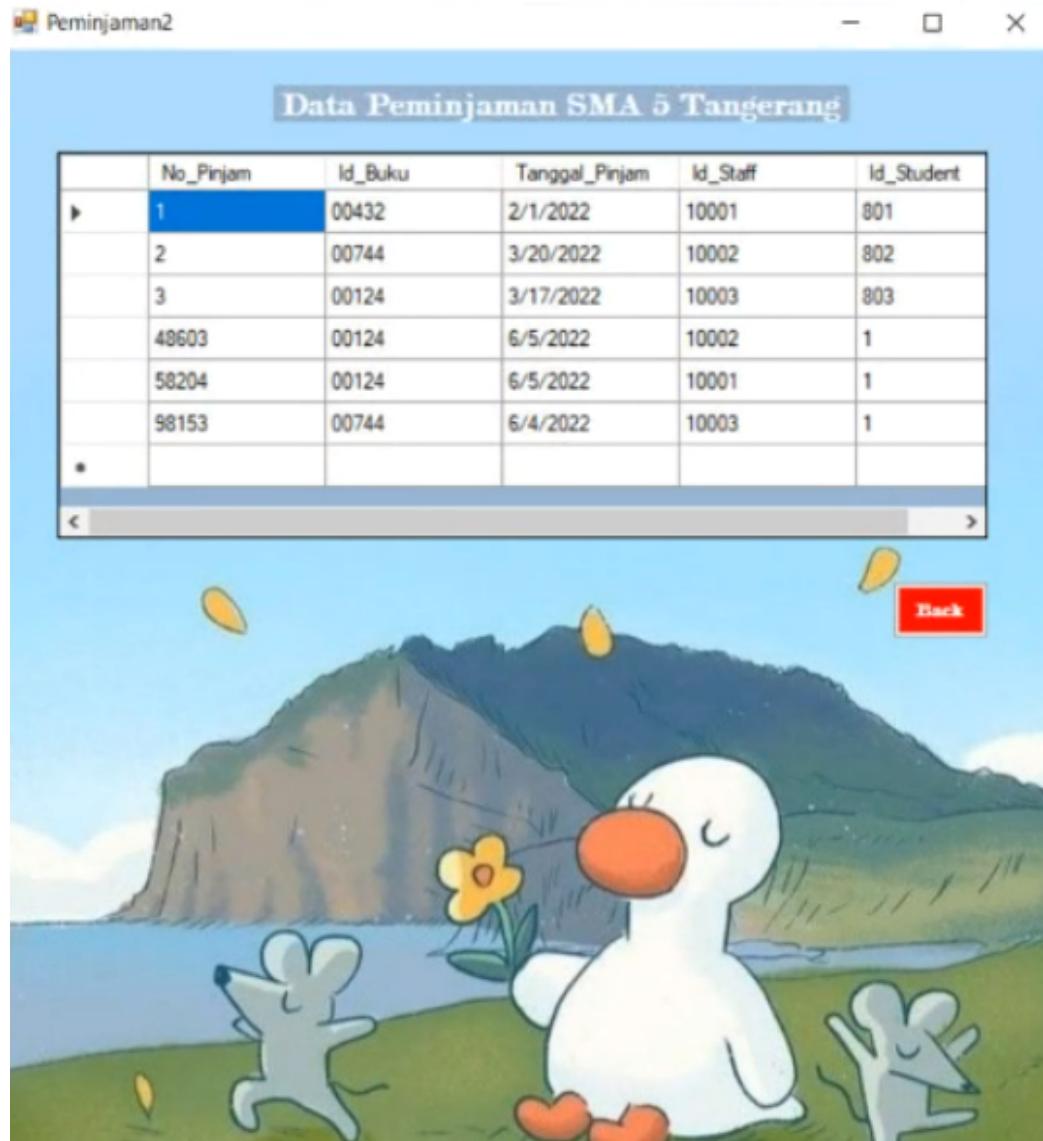
Gambar 4.16. laman pilihan data view.

Pada tampilan ini menunjukkan pilihan opsi yang dapat diakses dari data view yakni data buku, peminjaman, pengembalian, staff, dan student.

	Id_Buku	Judul_Buku	Kuantitas	Kategori	Penerbit
▶	00124	Ilmu Pengetahua...	1	Buku Pelajaran	DimasBook
	00432	Ensiklopedia Nus...	2	Ensiklopedia	Erlangga
*	00744	Laskar Pelangi	1	Novel Ringan	Gramedia

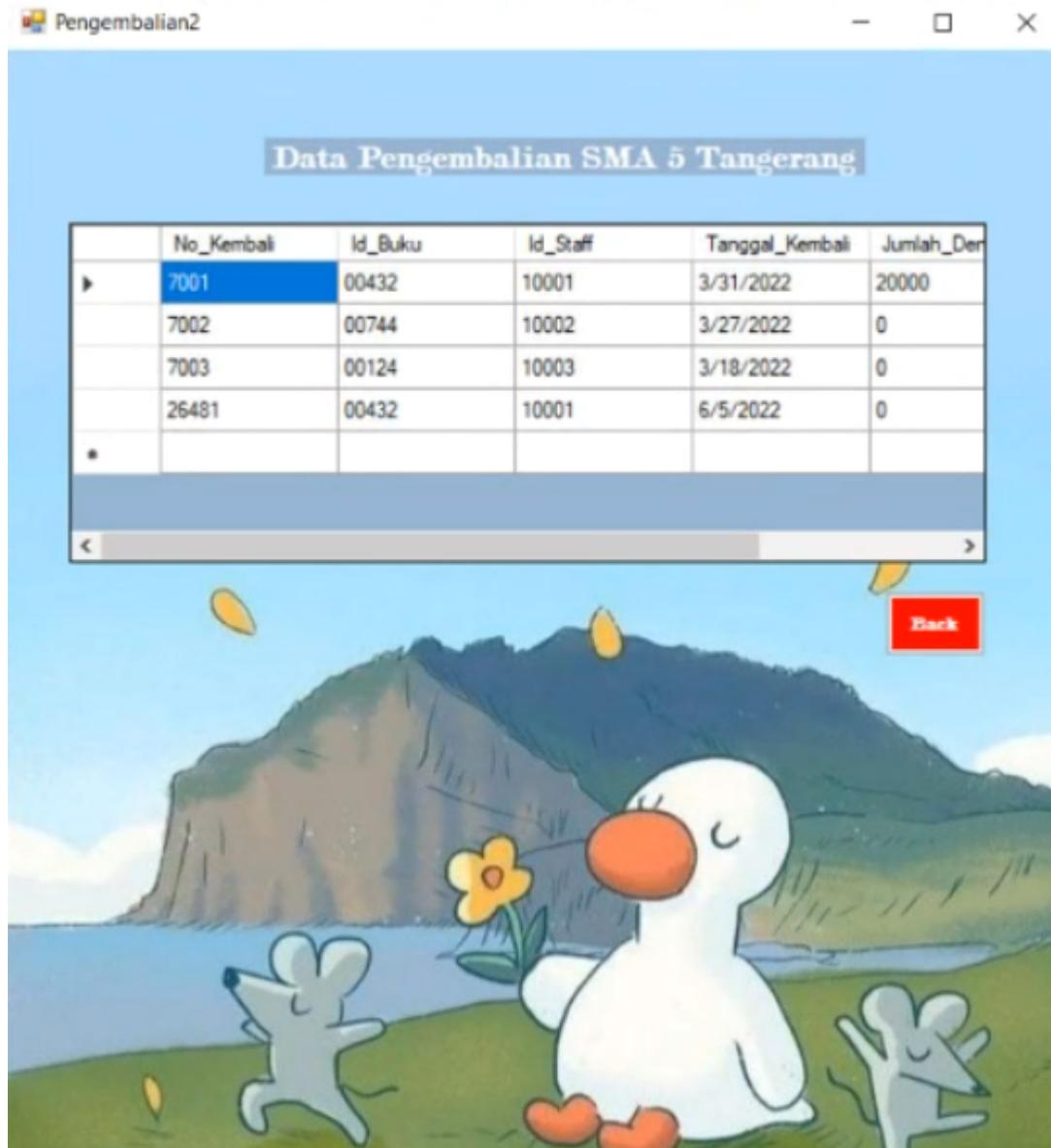
Gambar 4.17. laman data buku perpustakaan.

Pada halaman ini menampilkan data buku yang tersedia di perpustakaan SMA 5 Tangerang yang terdiri atas nomor identitas buku, judul buku, kuantitas, kategori, dan penerbit buku.



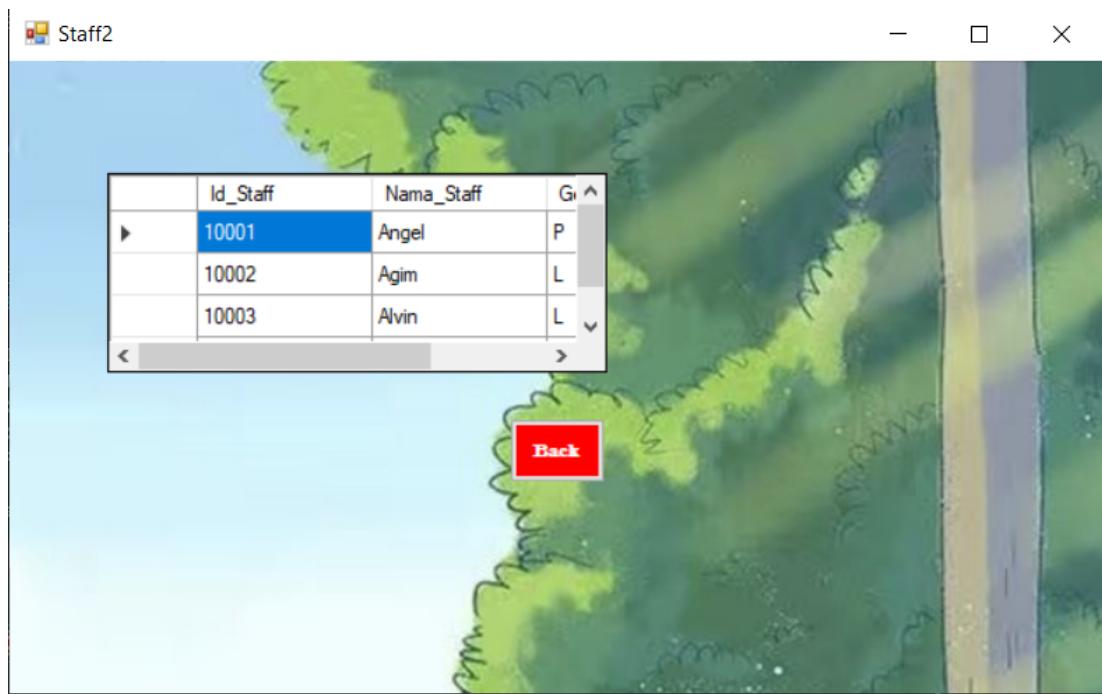
Gambar 4.18. laman data peminjaman buku perpustakaan

Pada halaman ini menampilkan data peminjaman yang terdapat pada SMA 5 Tangerang yang terdiri atas No_Pinjam, id_buku, Tanggal_Pinjam, id_staff, id_student.



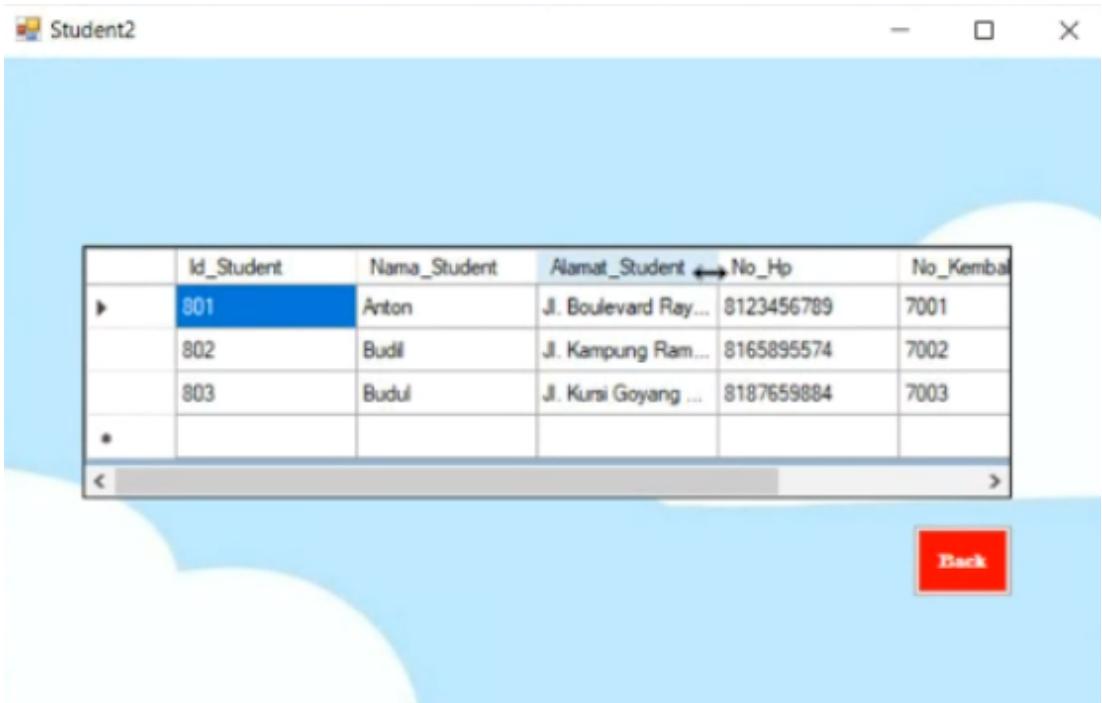
Gambar 4.19. laman yang menampilkan data pengembalian buku siswa SMA 5 Tangerang.

Pada laman ini, data pengembalian buku meliputi nomor pengembalian, id_buku, id_staff, Tanggal_Kembali, Jumlah_Denda.



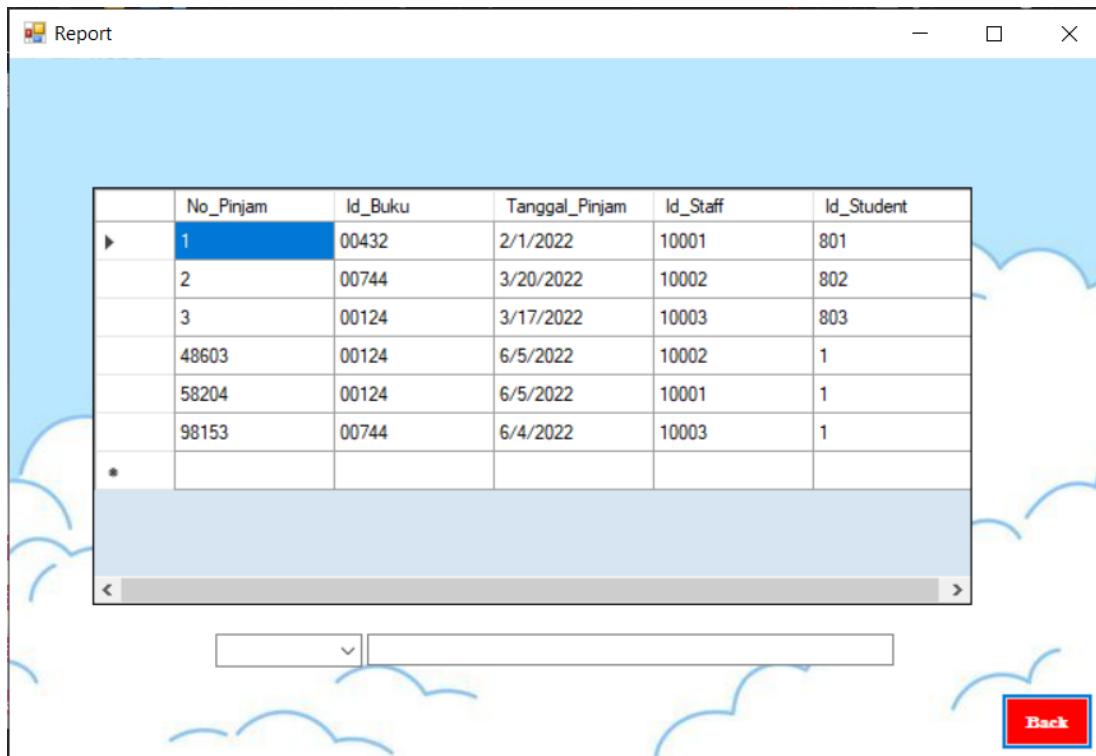
Gambar 4.20. laman yang menunjukkan data staff yang memiliki akses dan bertugas.

Pada laman ini, tampilan menunjukkan informasi dari staff seperti nomor identitas staff, nama staff, dan jenis kelamin dari staff (L/P).



Gambar 4.21. laman yang menunjukkan data informasi siswa dan siswi SMA 5 Tangerang.

Pada laman ini, tampilan menunjukkan data dari siswa dan siswi SMA 5 Tangerang yang pernah melakukan peminjaman dan pengembalian buku dari perpustakaan SMA 5 Tangerang. Data meliputi nomor identitas siswa dan siswi, nama siswa dan siswi, alamat rumah siswa dan siswi, nomor handphone siswa dan siswi, nomor pengembalian buku siswa dan siswi, jumlah denda yang telah diakumulasi berdasarkan waktu telat pengembalian.



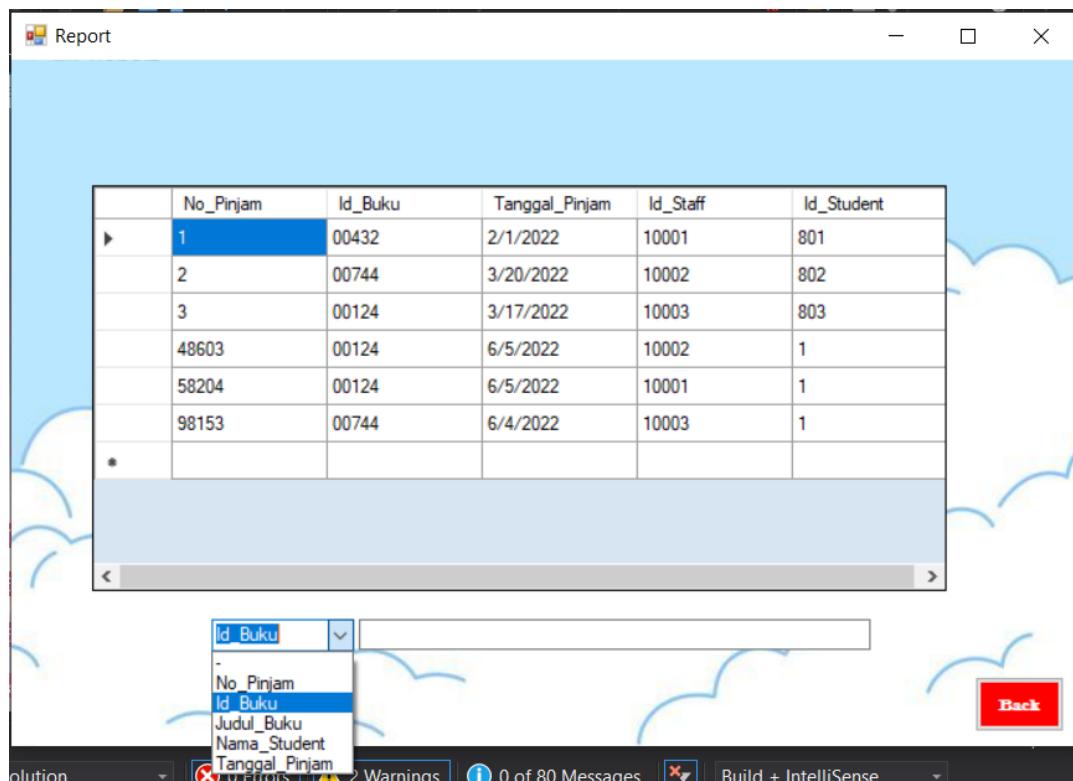
The screenshot shows a Windows-style application window titled "Report". Inside, there is a table with the following data:

No_Pinjam	Id_Buku	Tanggal_Pinjam	Id_Staff	Id_Student
1	00432	2/1/2022	10001	801
2	00744	3/20/2022	10002	802
3	00124	3/17/2022	10003	803
48603	00124	6/5/2022	10002	1
58204	00124	6/5/2022	10001	1
98153	00744	6/4/2022	10003	1
*				

Below the table is a small input field with a dropdown arrow. In the bottom right corner of the window, there is a red button labeled "Back".

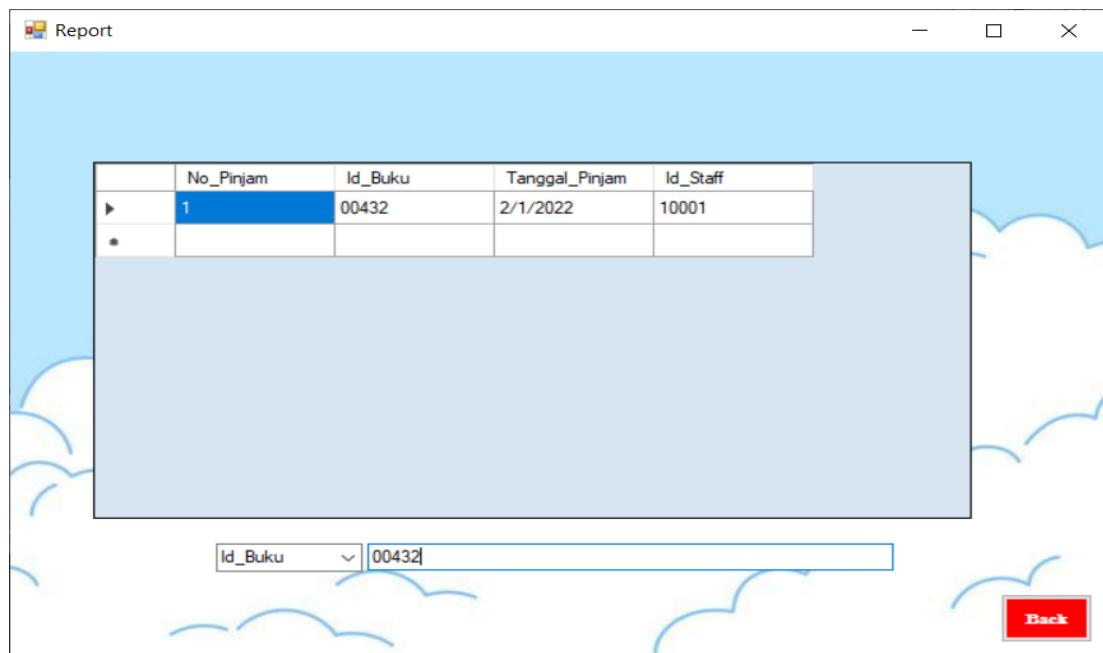
Gambar 4.22. laman tampilan report.

Report data merupakan laman yang muncul secara automatis setelah administrator perpustakaan melakukan penginputan data. Apabila administrator ingin melihat atau mencari data report kembali yang diperlukan, maka mereka tidak perlu menunggu hingga penginputan berikutnya untuk membaca data report kembali.



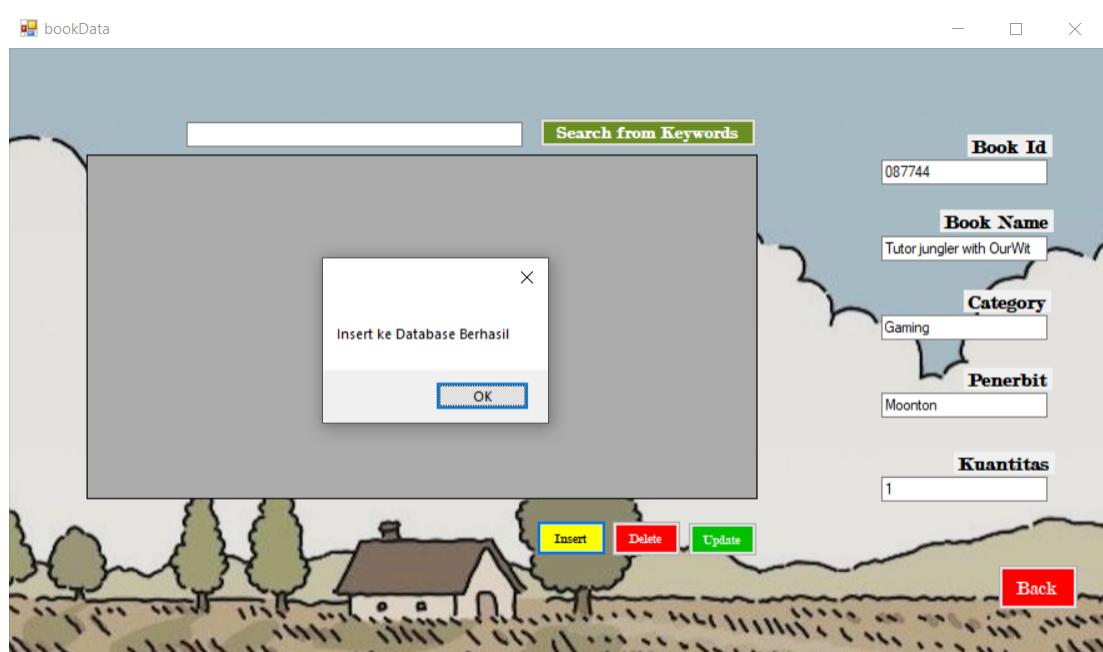
Gambar 4.23. laman yang menunjukkan tampilan report data.

Laman report data juga menawarkan fitur search otomatis melalui keywords, untuk mencari keywords yang diinginkan dalam nomor pinjam, nomor identitas buku, judul buku, nama siswa atau siswi, dan juga tanggal pinjam.



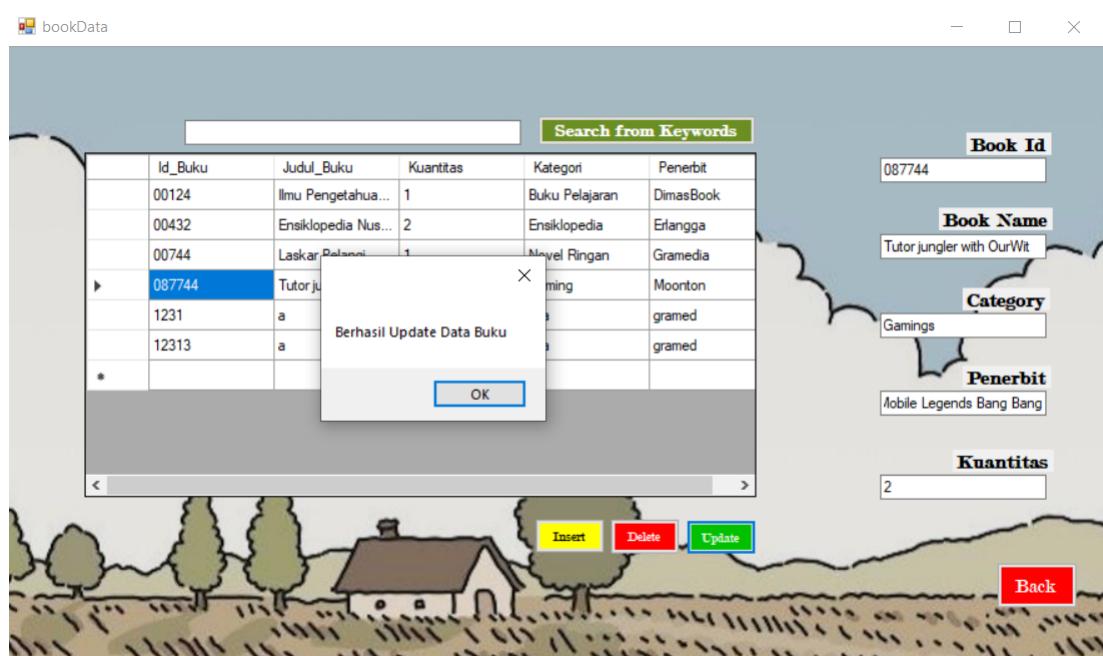
Gambar 4.24. illustrasi yang menunjukkan contoh pencarian search.

Dalam tampilan ini menunjukkan contoh pencarian data report melalui keyword nomor identitas buku yakni 00432, apabila terdapat lebih dari satu peminjaman yang pernah atau sedang meminjam buku 00432, maka administrator dapat melakukan pencarian dari tabel lainnya seperti nomor pinjam, nomor identitas staff, dan lainnya.

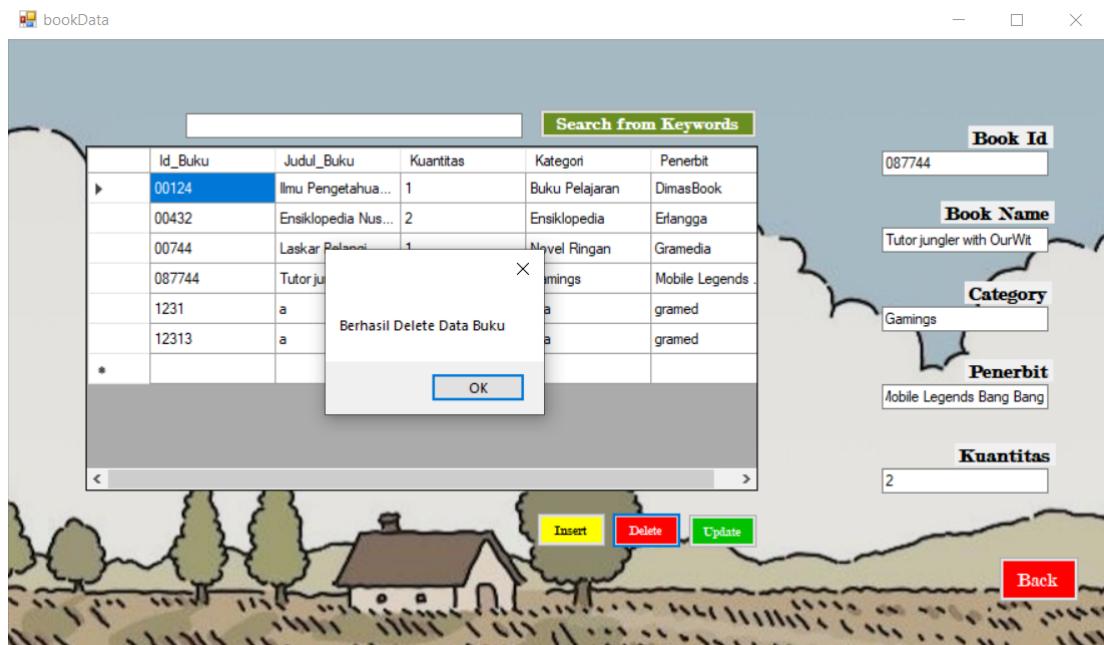


Gambar 4.25. tampilan insert book data sukses.

Pada tampilan ini menunjukkan laman yang disediakan oleh book data. Untuk meng-input data ke dalam database, K-ONE menggunakan metode CRUD (create, read, update, dan delete) untuk insert data ke dalam database. Data meliputi nomor identitas buku, judul buku, kategori buku, penerbit, dan kuantitas buku. Melalui illustrasi ini, administrator telah berhasil memassukkan data yang diketik ke dalam database K-ONE.

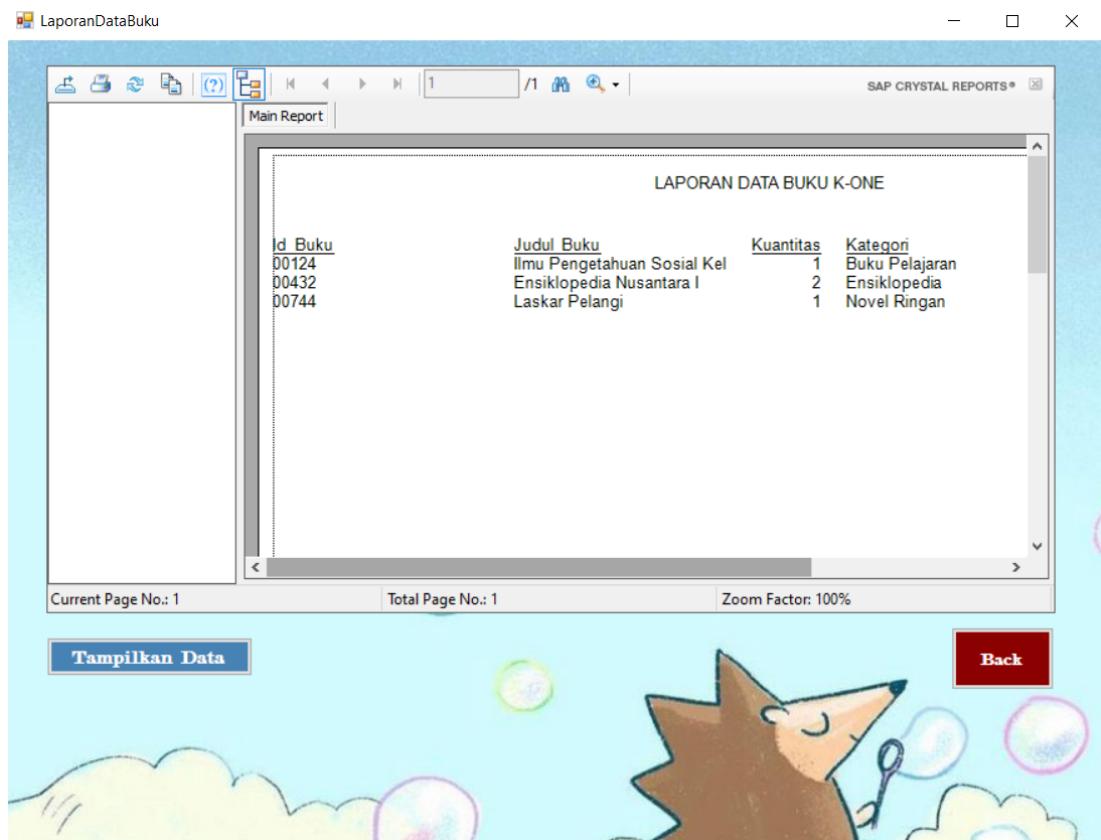
*Gambar 4.26. laman tampilan update sukses ke dalam database.*

Pada tampilan ini menunjukkan bahwa administrator memutuskan untuk melakukan perubahan terhadap data penerbit dan data kuantitas (dapat dibandingkan gambar 4.26 dengan gambar 4.25). Administrator hanya perlu menghapus data yang ingin diganti, dan mengetik kembali data yang ingin dimasukkan lalu menekan update. Apabila data telah berhasil diubah maka akan muncul data telah berhasil di-update.



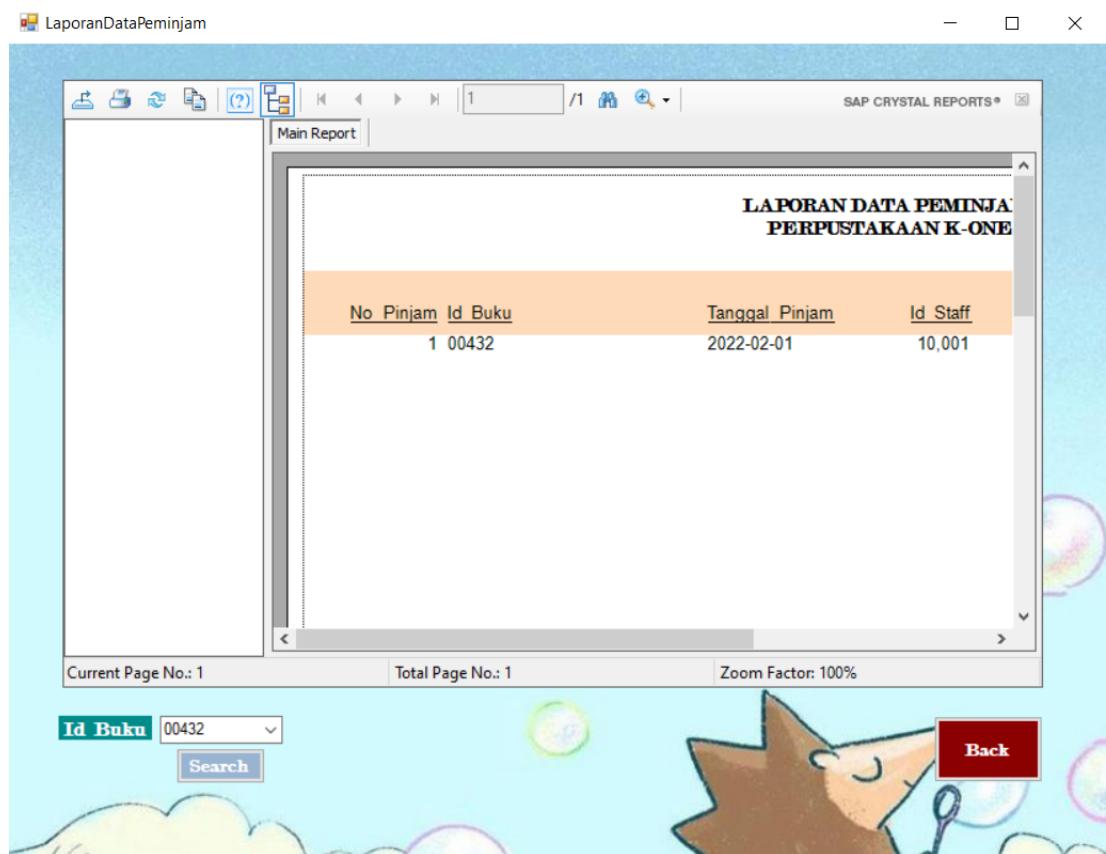
Gambar 4.27. laman menunjukkan penghapusan data sukses.

Pada laman ini tampilan menunjukkan bahwa administrator telah berhasil menghapus data yang diinginkan dari database. Apabila data dalam buku telah berhasil dihapus maka administrator akan mendapatkan notifikasi pop-up bahwa data yang ditunjuk telah berhasil dihapus dari tampilan box dan dari database.



Gambar 4.28. laman yang menunjukkan laporan data buku pada crystal report.

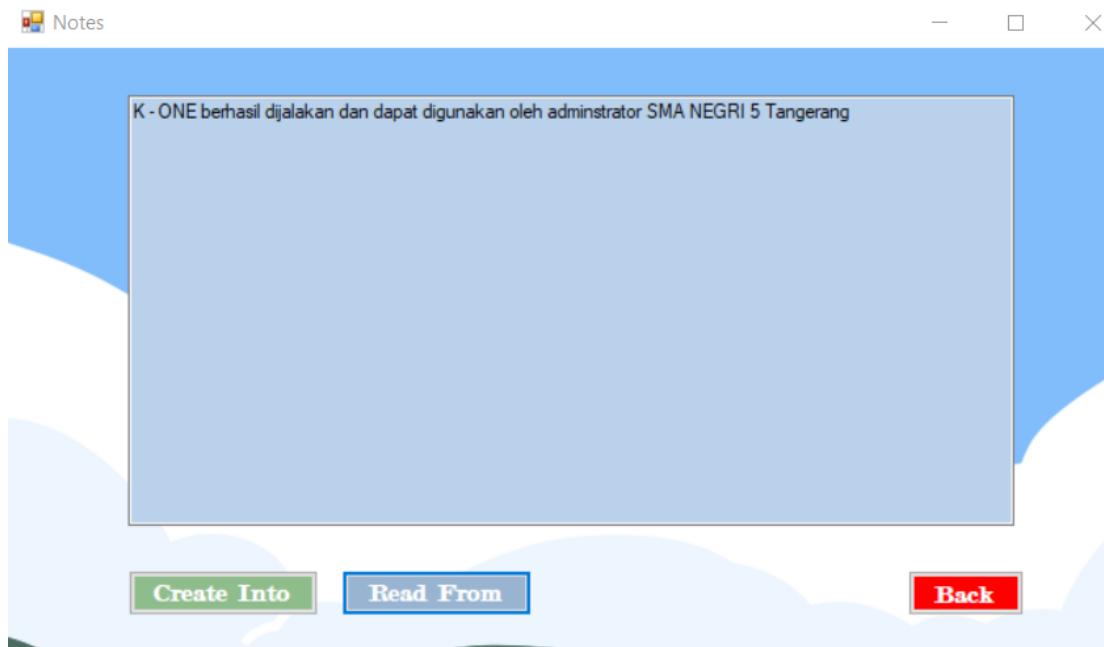
Pada form ini diperuntukkan untuk administrator dapat melihat terlebih dahulu data pada DataBuku crystal report ketika mereka menekan tombol Tampilkan Data. Setelah data - DataBuku telah terlampir pada crystal report, maka administrator dapat mengeprint lampiran data - DataBuku tersebut pada tombol diatas kiri atas. Sehingga lampiran data yang berada pada crystal report tersebut dapat di print sesuai kehendak administrator.



Gambar 4.29.

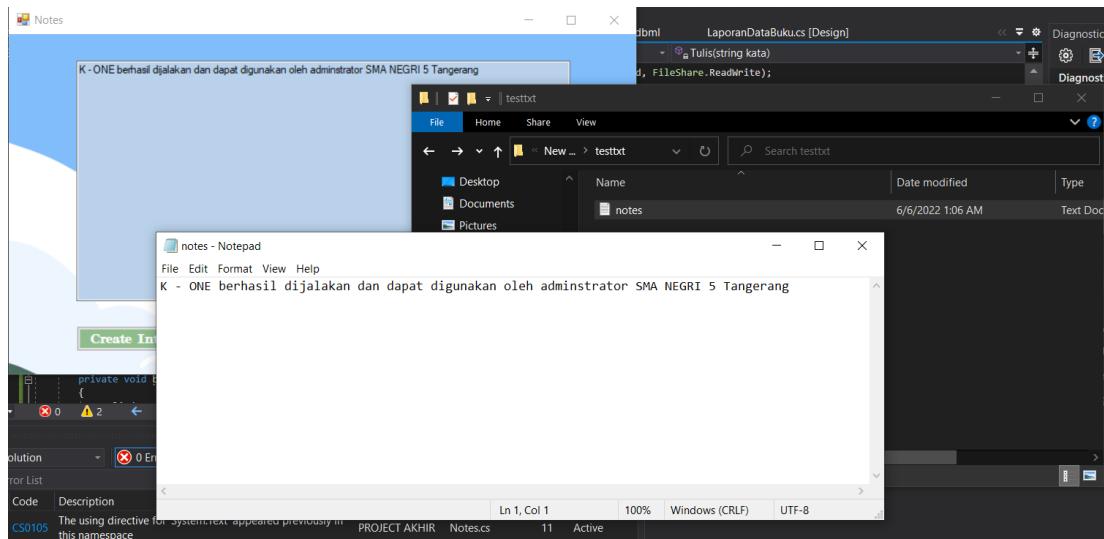
laman yang menunjukkan laporan data peminjaman pada crystal report.

Pada form ini diperuntukkan untuk administrator dapat melihat terlebih dahulu data pada DataPeminjam crystal report. Bedanya dengan crystal report yang sebelumnya, pada crystal report ini secara parameter yang berarti administrator diberikan sebuah tool untuk dapat menfilter data yang hendak mereka tampilkan dan segera di print. Sedangkan crystal report pada data - DataBuku itu secara formula yang berarti tidak ada tool untuk menfilter data, sehingga ketika di print maka akan mengeprint keseluruhan dari data - DataBuku. Untuk cara mengeprint crystal report pada data - DataPeminjam caranya sama seperti crystal report sebelumnya yaitu, meneklik tombol di kiri atas. Sehingga lampiran data yang berada pada crystal report tersebut dapat di print sesuai kehendak administrator.



Gambar 4.30. laman notes atau catatan

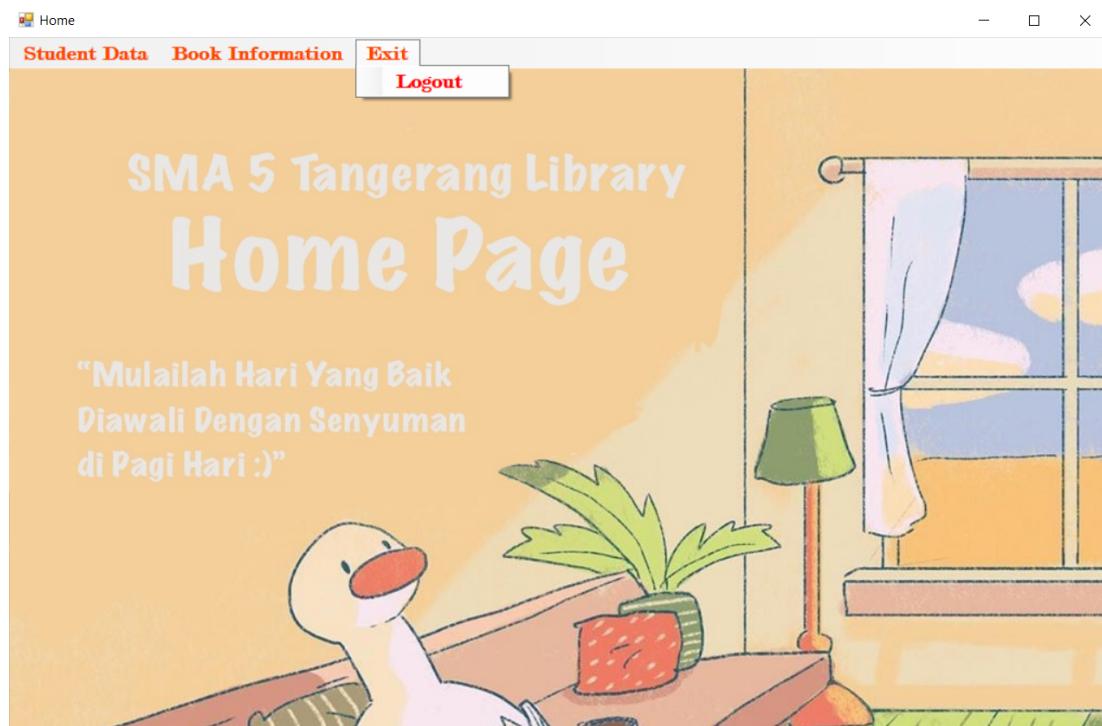
Pada tampilan ini menunjukkan notes atau catatan yang hendak dibuat oleh administrator dengan tujuan komunikasi atau sekedar pengingat. Teks yang dimasukkan ke dalam box akan menjadi file .txt. Tombol create into akan membuat note txt ke dalam folder sql perpustakaan, dan tombol read from digunakan untuk membaca isi dari teks yang telah disimpan.



Gambar 4.31. laman notes atau catatan dalam bentuk txt.

Pada tampilan ini menunjukkan bahwa dengan menekan button create into akan membuat teks yang telah dimasukkan menjadi sebuah file .txt yang dapat dibuka

dalam folder tujuan melalui notepad. Sedangkan read form digunakan untuk membuka file .txt ke dalam box biru.



Gambar 4.32. tampilan exit untuk log out.

Administrator yang sudah menyelesaikan segala tugas yang berkaitan dengan pemasukan, perubahan, dan penghapusan data dapat mengakiri aplikasi K-ONE dengan cara log-out diri mereka demi kenyamanan dan keamanan data.

4.3 Query Pembuatan Tabel Basis Data

```
create database PROJECT;
use PROJECT;
```

4.3.1 Pembuatan Table DataBuku

```
Create table DataBuku(
Id_Buku VARCHAR (50) PRIMARY KEY NOT NULL,
Judul_Buku VARCHAR (50) NOT NULL,
Kuantitas INT NOT NULL,
Kategori VARCHAR (50) NOT NULL,
Penerbit VARCHAR (50) NOT NULL,
)
```

4.3.2 Pembuatan Table Staff

```
CREATE TABLE Staff(
Id_Staff int PRIMARY KEY NOT NULL,
Nama_Staff VARCHAR (50) NOT NULL,
Gender_Staff CHAR (1) NOT NULL,
Username VARCHAR (50) NOT NULL,
Password_staff VARCHAR (50) NOT NULL
)
```

4.3.3 Pembuatan Table Peminjaman

```
CREATE TABLE Peminjaman(
No_Pinjam int PRIMARY KEY NOT NULL,
Id_Buku VARCHAR (50) NOT NULL,
Tanggal_Pinjam DATE NOT NULL,
Id_Staff int NOT NULL,
FOREIGN KEY (Id_Buku) REFERENCES DataBuku ,
FOREIGN KEY (Id_Staff) REFERENCES Staff ,
)
```

4.3.4 Pembuatan Table Pengembalian

```
CREATE TABLE Pengembalian(
No_Kembali INT PRIMARY KEY NOT NULL,
Id_Buku VARCHAR (50) NOT NULL,
Id_Staff int NOT NULL,
Tanggal_Kembali DATE NOT NULL,
Jumlah_Denda INT NOT NULL,
FOREIGN KEY (Id_Buku) REFERENCES DataBuku ,
FOREIGN KEY (Id_Staff) REFERENCES Staff,
)
```

4.3.5 Pembuatan Table Student

```
CREATE TABLE Student(
    id_Student int PRIMARY KEY NOT NULL,
    Nama_Student VARCHAR (50) NOT NULL,
    Alamat_Student VARCHAR (50) NOT NULL,
    No_Hp BIGINT NOT NULL,
    No_Kembali INT NOT NULL,
    No_Pinjam INT NOT NULL,
    FOREIGN KEY (No_Kembali) REFERENCES Pengembalian ,
    FOREIGN KEY (No_Pinjam) REFERENCES Peminjaman ,
)
```

4.3.8 Memasukan Data Ke dalam Tabel DataBuku

```
-- DataBuku
INSERT INTO DataBuku VALUES ('00432','Ensiklopedia Nusantara I','2','Ensiklopedia','Erlangga')
INSERT INTO DataBuku VALUES ('00744','Laskar Pelangi','1','Novel Ringan','Gramedia')
INSERT INTO DataBuku VALUES ('00124','Ilmu Pengetahuan Sosial Kelas -11','1','Buku Pelajaran','DimasBook')
```

4.3.9 Memasukan Data Ke dalam Tabel Staff

```
-- Staff
INSERT INTO Staff VALUES
('10001','Angel','P','AngelFly','Fly842')
INSERT INTO Staff VALUES
('10002','Agim','L','Larith','Coklat30')
INSERT INTO Staff VALUES
('10003','Alvin','L','veen444','Vin444')
```

4.3.10 Memasukan Data Ke dalam Tabel Pengembalian

```
-- Pengembalian
INSERT INTO Pengembalian VALUES
('7001','00432','10001','2022/03/31','20000')
INSERT INTO Pengembalian VALUES
('7002','00744','10002','2022/03/27','-')
INSERT INTO Pengembalian VALUES
('7003','00124','10003','2022/03/18','-')
```

4.3.11 Memasukan Data Ke dalam Tabel Peminjaman

```
-- Peminjaman
```

```

INSERT INTO Peminjaman VALUES
('0001','00432','2/1/2022','10001')
INSERT INTO Peminjaman VALUES
('0002','00744','3/20/2022','10002')
INSERT INTO Peminjaman VALUES
('0003','00124','3/17/2022','10003')

```

4.3.12 Memasukan Data Ke dalam Tabel Student

```

-- Student
INSERT INTO Student VALUES ('801','Anton','Jl. Boulevard Raya
1 Blok CA No.1','08123456789','7001','0001')
INSERT INTO Student VALUES ('802','Budil','Jl. Kampung
Rambutan No.9','08165895574','7002','0002')
INSERT INTO Student VALUES ('803','Budul','Jl. Kursi Goyang No
2','08187659884','7003','0003')

```

4.3.13 Menampilkan Tabel-Tabel

```

SELECT * FROM DataBuku;
SELECT * FROM Staff;
SELECT * FROM Student;
SELECT * FROM Peminjaman;
SELECT * FROM Pengembalian;

```

4.3.14 Menghapus Isi Tabel

```

DELETE FROM DataBuku;
DELETE FROM Staff;
DELETE FROM Student;
DELETE FROM Peminjaman;
DELETE FROM Pengembalian;

```

4.3.15 Menghapus Tabel

```

DROP TABLE Student;
DROP TABLE Pengembalian;
DROP TABLE Peminjaman;
DROP TABLE Staff;
DROP TABLE DataBuku;

```

4.4 Advanced Query Basis Data

```
SELECT Kuantitas, (SELECT SUM(Kuantitas) FROM DataBuku) as  
TotalBuku FROM DataBuku
```

4.5 Kesimpulan

K-ONE merupakan hasil implementasi dari usulan kelompok kami yaitu merancang sebuah program aplikasi yang berbasis library management. Aplikasi ini dapat dipergunakan untuk mengubah dari kinerja perpustakaan SMA Negeri 5 Tangerang dalam menginput data peminjaman dan pengembalian buku dengan cara yang lebih efektif dan efisien dibandingkan secara manual. Meninjau dari penelitian sebelumnya yang menggunakan sebuah website yang bisa menyimpan data ke dalam database, kelompok kami membuatnya dengan menggunakan versi lain. Tampilan dari versi lain yang kami usulkan berupa menggunakan MySql sebagai tempat untuk menyimpan kedalam database dan juga bahasa pemrograman C# sebagai sarana tempat kami memprogram aplikasi ini. Kehadiran dari aplikasi ini akan dipergunakan oleh bagian internal saja yakni administrator yang dapat mengakses, mengontrol, dan mengatur data para pelajar yang meminjam atau mengembalikan buku dari perpustakaan SMA Negeri 5 Tangerang.

Tujuan utama dalam membuat aplikasi ini yaitu dipergunakan sebagai aplikasi yang mengorganisasi koleksi data buku dari sekian banyaknya data untuk keperluan yang bermacam. Sebuah file dari informasi yang diberikan sudah tersusun dengan logis dan berurutan sesuai pada aturan yang bertujuan untuk memfasilitasi akses program aplikasi komputer K-ONE. Seluruh data rekaman dalam database kami saling berkorelasi dan redundansi data harus selalu diperkecil.

Maka berdasarkan hasil analisa kami setelah menjalankan program aplikasi administrasi K-ONE dalam perpustakaan SMA 5 Tangerang yang menunjukkan dapat berjalan dengan semestinya, kelompok kami mendapatkan beberapa analisa :

- a. Terjadi integrasi data files yang sudah tercatat
- b. Pembagian informasi data kepada seluruh administrator perpustakaan dapat membawa banyak keuntungan yang memudahkan
- c. Menyederhanakan penggunaan dari data files

- d. Meminimalisir biaya dari menyimpan dan mengambil data
- e. Tingkat akurasi dan konsistensi
- f. Melindungi data security dari user yang tidak terautorisasii

k-one telah menjadi sebuah program yang memfasilitasi data baru dan lama dengan menawarkan opsi seperti modifikasi dan retrieving di area database.

Selanjutnya dengan hasil analisa yang telah dijabarkan sebelumnya maka mendapat kesimpulan bahwa kemampuan yang diberikan oleh aplikasi K-ONE sebagai usulan dari kelompok kami sudah bisa dipergunakan dengan baik dalam sistem informasi. Seperti dalam membantu dalam pengolahan data - data yang berawal dari sistem lama menjadi ke sistem baru, memudahkan untuk mengawasi segala tindak terjadinya interaksi antara murid dengan administrator, serta membantu dan meringankan pekerjaan para administrator ketika sedang menjalankan tugasnya sebagai pustakawan. Sehingga dalam pengimplementasianya diharapkan untuk aplikasi K-ONE dapat selalu membantu para administrator terkhususnya pada administrator pada SMA Negeri 5 Tangerang.

Lampiran

I. Source Code Home

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace PROJECT_AKHIR
{
    public partial class Home : Form
    {
        public Home()
        {
            InitializeComponent();
        }

        private void
peminjamanToolStripMenuItem_Click(object sender,
EventArgs e)
        {
            // ke peminjaman
            peminjaman panggil = new peminjaman();
            panggil.Show();
            this.Hide();
        }

        private void
pengembalianToolStripMenuItem_Click(object sender,
EventArgs e)
        {
            // ke pengembalian
            pengembalian panggil = new pengembalian();
            panggil.Show();
            this.Hide();
        }

        private void
historyToolStripMenuItem_Click(object sender, EventArgs
e)
        {

            // ke histor
```

```
}

    private void
reportToolStripMenuItem_Click(object sender, EventArgs
e)
{
    // kalo ini ke report
    Report panggil = new Report();
    panggil.Show();
    this.Hide();
}

    private void
bookDataToolStripMenuItem_Click(object sender, EventArgs
e)
{
    bookData panggil = new bookData();
    panggil.Show();
    this.Hide();
}

    private void
logoutToolStripMenuItem_Click(object sender, EventArgs
e)
{
    //logout
    System.Threading.Thread t = new
System.Threading.Thread(new
System.Threading.ThreadStart(OpenLoginForm)); //you
create a new thread
    this.Close(); //you close your current form
(for example a menu)
    t.Start(); //you start the thread
}
public static void OpenLoginForm()
{
    Application.Run(new Login()); //run your new
form
}

    private void
allReportToolStripMenuItem_Click(object sender,
EventArgs e)
{
}
```

```
    private void
studentDataToolStripMenuItem_Click(object sender,
EventArgs e)
{
}

    private void
dataBukuToolStripMenuItem_Click(object sender, EventArgs
e)
{
    DataBuku2 open = new DataBuku2();
    open.Show();
    this.Hide();
}

    private void
peminjamanToolStripMenuItem1_Click(object sender,
EventArgs e)
{
    Peminjaman2 open = new Peminjaman2();
    open.Show();
    this.Hide();
}

    private void
pengembalianToolStripMenuItem1_Click(object sender,
EventArgs e)
{
    Pengembalian2 open = new Pengembalian2();
    open.Show();
    this.Hide();
}

    private void staffToolStripMenuItem_Click(object
sender, EventArgs e)
{
    Staff2 open = new Staff2();
    open.Show();
    this.Hide();
}

    private void
studentToolStripMenuItem_Click(object sender, EventArgs
e)
{
    Student2 open = new Student2();
    open.Show();
    this.Hide();
}
```

```

        }

        private void
laporanDataBukuToolStripMenuItem_Click(object sender,
EventArgs e)
{
    LaporanDataBuku panggil = new
LaporanDataBuku();
    panggil.Show();
    this.Hide();
}

        private void
laporanDataStudentToolStripMenuItem_Click(object sender,
EventArgs e)
{
    LaporanDataPeminjam panggil = new
LaporanDataPeminjam();
    panggil.Show();
    this.Hide();
}

        private void notesToolStripMenuItem_Click(object
sender, EventArgs e)
{
    Notes panggil = new Notes();
    panggil.Show();
    this.Hide();
}
}

```

Penjelasan :

Syntak codingan diatas menunjukkan dan diperuntukkan untuk user dapat memilih sesuai kehendaknya mau dipergunakan sebagai apa aplikasinya.

- **(form) variable = new (form)();**
syntax diatas memanggil form lain dengan nama variable dan untuk ini kurung buka dan kurung tutup di hapus.
- **variable.Show();**
Variable alias nama form lain itu akan ditampilkan setelah menuItem ditekan
- **this.Hide();**
Syntak diatas ini memiliki arti form home ini akan di hidden atau di hilangkan tampilannya, dan selanjutnya diarahkan sesuai pengguna menekan menuItem yang mana. Berikut merupakan lanjutan dari syntax

```

peminjaman panggil = new peminjaman();
panggil.Show();
this.Hide();

```

Adapun syntax lain yang dipergunakan pada form ini yaitu

```

private void
logoutToolStripMenuItem_Click(object sender, EventArgs
e)
{
    //logout
    System.Threading.Thread t = new
System.Threading.Thread(new
System.Threading.ThreadStart(OpenLoginForm)); //you
create a new thread
    this.Close(); //you close your current form
(for example a menu)
    t.Start(); //you start the thread
}
public static void OpenLoginForm()
{
    Application.Run(new Login()); //run your new
form
}

```

Syntax ini dipergunakan untuk ketika pengguna menekan menuItem LogOut, maka mereka akan terlogout akses yang dimiliki pada aplikasi K-One. Kemudian dengan secara otomatis akan diarahkan menuju form Login dan harus mengulang dan mengisi data nama admin (Username) dan berserta password.

Sehingga dari penjelasan ini adapun diantaranya menuItem yang telah disediakan oleh aplikasi K-One meliputi:

- peminjamanToolStripMenuItem_Click = Ini dipergunakan untuk mengarahkan pengguna menuju form peminjaman
- pengembalianToolStripMenuItem_Click = Ini dipergunakan untuk mengarahkan pengguna menuju kepada form pengembalian
- historyToolStripMenuItem_Click = Ini dipergunakan untuk mengarahkan pengguna ke form history
- reportToolStripMenuItem_Click = Ini digunakan untuk mengarahkan pengguna menuju ke form report
- bookDataToolStripMenuItem_Click = Ini digunakan untuk mengarahkan pengguna menuju ke form bookData

- logoutToolStripMenuItem_Click = Ini digunakan untuk mengarahkan pengguna ketika sudah menyelesaikan keseluruhan proses dari menggunakan aplikasi K- ONE, maka setelah di tekan menu item log out ini maka admin akan ter - log out dari aplikasi K - one ini.
- dataBukuToolStripMenuItem_Click = Ini digunakan untuk mengarahkan pengguna menuju form dataBuku
- pengembalianToolStripMenuItem1_Click = Ini digunakan untuk mengarahkan pengguna menuju form pengembalian
- staffToolStripMenuItem_Click = ini digunakan untuk mengarahkan pengguna menuju form staffTool
- studentToolStripMenuItem_Click = Ini digunakan untuk mengarahkan pengguna menuju ke form Student
- laporanDataBukuToolStripMenuItem_Click = Ini digunakan untuk mengarahkan pengguna menuju ke form laporanDataBuku
- laporanDataStudentToolStripMenuItem_Click = Ini digunakan untuk mengarahkan pengguna menuju ke form laporanDataStudent
- notesToolStripMenuItem_Click = Ini digunakan untuk mengarahkan pengguna menuju ke form notes

II. Source Code Log-In

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient; // digunakan untuk dapat
mengakses database mysql

namespace PROJECT_AKHIR
{
    public partial class Login : Form
    {
        public Login()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
    }
}

```

```

        SqlConnection koneksi = new SqlConnection(@"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True");
        SqlDataAdapter sda = new SqlDataAdapter("select
count (*) from Staff where Username = '" + comboBox1.Text + "'"
and Password_staff = '" + textBox2.Text + "'", koneksi);
        DataTable dt = new DataTable();
        sda.Fill(dt);
        if (dt.Rows[0][0].ToString() == "1")
        {
            this.Hide();
            Home panggil = new Home();
            panggil.Show();
        }
        else
        {
            MessageBox.Show("Username atau Password salah!
Harap mengisi username dan password dengan benar",
"Perhatian", MessageBoxButtons.OK, MessageBoxIcon.Error);
        }
    }

    private void button2_Click(object sender, EventArgs e)
    {
        comboBox1.Text = String.Empty;
        textBox2.Text = String.Empty;
    }

    private void label4_Click(object sender, EventArgs e)
    {
        forgetpass panggil = new forgetpass();
        panggil.Show();
        this.Hide();
    }
}
}

```

Penjelasan :

Pada form ini, dipergunakan ketika user mau login ke aplikasi K-ONE, maka form ini adalah tampilan awal dari aplikasi K- ONE.

- using System.Data.SqlClient; // digunakan untuk dapat mengakses database mysql
- sqlConnection koneksi = new SqlConnection(@"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True"); // digunakan untuk dapat mengakses database my sql dengan source code DESKTOP-PSDHS93\SQLEXPRESS, kemudian pada kali ini akan dipergunakan untuk membaca database bernama PROJECT.

Cara kerja form diatas yaitu, menyocokkan data username, beserta password yang telah dibaca sebelumnya pada database PROJECT dan berada pada tabel staff.

- Ketika username, beserta password yang di isikan oleh para user benar, maka form login ini akan di hide, selanjutnya akan ditampilkan form home. Apabila terjadi kesalahan dalam mengisi password maka tampilan peringatan berupa warning seperti “Username atau Password salah! Harap mengisi username dan password dengan benar”
- Terdapat tombol lain yang mengarah kepada form lain yaitu ketika pengguna lupa akan passwordnya, maka akan diarahkan ke form forgotpassword, dan para admin dapat mengganti password mereka disana.

III. Source Code Forgot Password

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace PROJECT_AKHIR
{
    public partial class forgetpass : Form
    {
        public forgetpass()
        {
            InitializeComponent();
        }

        private void button2_Click(object sender,
EventArgs e)
        {
            Home panggil = new Home();
            this.Hide();
            panggil.Show();
        }

        private void button1_Click(object sender,
EventArgs e)
        {
            DataClasses1DataContext db = new
DataClasses1DataContext();
```

```

        var ubah = (from a in db.Staffs
                    where a.Username ==
comboBox1.Text
                    select a).Single();

        ubah.Password_staff = textBox2.Text;

        try
        {
            db.SubmitChanges();
            MessageBox.Show("Berhasil Update
Password Staff");
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString());
        }
    }
}
}

```

Penjelasan :

- Pada form ini dipergunakan untuk admin yang hendak mengganti password untuk dapat melogin kepada aplikasi K-ONE. Biasanya ini dapat dipergunakan oleh admin ketika melupakan passwordnya ataupun hanya sekedar ingin mengganti passwordnya.
- Pertama form ini harus menghadirkan dbml yang sudah dibuat terlebih dahulu, dan pada dbml kali ini bernama DataClasses1DataContext yang dibuat variable pemanggilan yaitu db.
- maka codingan dibawah ini dapat dijalankan,

```

var ubah = (from a in db.Staffs
            where a.Username ==
comboBox1.Text
            select a).Single();

```

```
        ubah.Password_staff = textBox2.Text;
```

- Syntax ini menjelaskan bahwa dbml yang berisikan data tabel staff, dan dimana username itu berada pada combobox1 dan diambil secara single, maka password staff yang berada pada username single tadi akan dirubah pada textBox2.
- Selanjutnya terdapat try and catch, yang berarti ketika berhasil akan menampilkan notifikasi berupa "Berhasil Update Password Staff", sedangkan bila tidak berhasil dikarenakan eror atau terdapat kesalahan

secara sistematis, maka akan memunculkan notifikasi terdapat kesalahan.

IV. Source Code Peminjaman

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;
using System.Globalization;
using System.IO;

namespace PROJECT_AKHIR
{
    public partial class peminjaman : Form
    {
        SqlConnection koneksi = new SqlConnection(@"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True");

        public peminjaman()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            //back

            Home panggil = new Home();
            panggil.Show();
            this.Hide();
        }

        private void Clear_Click(object sender, EventArgs e)
        {

            label3.Text = String.Empty;
            comboBox1.Text = String.Empty;
            comboBox2.Text = String.Empty;
        }
    }
}
```

```
        DataClasses1DataContext db = new
DataClasses1DataContext();
        private void Save_Buku_Click(object sender, EventArgs
e)
    {
        Peminjaman buku = new Peminjaman()
{
            No_Pinjam = Convert.ToInt32 (label3.Text),
            Id_Buku = comboBox2.Text,
            Tanggal_Pinjam =
Convert.ToDateTime(dateTimePicker1.Value),
            Id_Staff = Convert.ToInt32 (comboBox1.Text),
            Id_Student = Convert.ToInt32 (textBox1.Text)

};

        db.Peminjamans.InsertOnSubmit(buku);
        try
        {
            db.SubmitChanges();
            MessageBox.Show("Save Peminjaman Berhasil");
            Report panggil = new Report();
            panggil.Show();

        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString());
        }
    }

    private void peminjaman_Load(object sender, EventArgs
e)
{
    // nampilin data ke gridview
    gridView_data();

    // nampilin data id buku ke combobox
    data_Buku_combo_box();

    // nampilin data id staff ke combobox
    data_idSTAFF_combo_box();
}

// data ke grid view

public void gridView_data()
{
```

```
        string constring = @"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True";
        SqlConnection conn = new SqlConnection(constring);

        SqlCommand cmd = new SqlCommand("SELECT * FROM
DataBuku", conn);
        SqlDataAdapter dataAdapter = new
SqlDataAdapter(cmd);
        DataTable dt = new DataTable();
        dataAdapter.Fill(dt);
        dataGridView1.DataSource = dt;
    }

    // data id buku ke combo box
    public void data_Buku_combo_box()
    {
        SqlConnection conn = new SqlConnection(@"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True");
        SqlCommand cmd = new SqlCommand("SELECT Id_Buku
FROM DataBuku", conn);
        SqlDataAdapter da = new SqlDataAdapter();
        da.SelectCommand = cmd;
        DataTable table1 = new DataTable();
        da.Fill(table1);
        comboBox2.DataSource = table1;
        comboBox2.DisplayMember = "Id_Buku";
        comboBox2.ValueMember = "Id_Buku";
    }

    // data id staff ke combo box
    public void data_idSTAFF_combo_box()
    {
        SqlConnection conn = new SqlConnection(@"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True");
        SqlCommand cmd = new SqlCommand("SELECT Id_Staff
FROM Staff", conn);
        SqlDataAdapter da = new SqlDataAdapter();
        da.SelectCommand = cmd;
        DataTable table1 = new DataTable();
        da.Fill(table1);
        comboBox1.DataSource = table1;
        comboBox1.DisplayMember = "Id_Staff";
        comboBox1.ValueMember = "Id_Staff";
    }
```

```
private void button2_Click(object sender, EventArgs e)
{
    //button search
    DataClasses1DataContext db = new
DataClasses1DataContext();
    IQueryable<DataBuku> cust = db.DataBukus;
    if (!String.IsNullOrEmpty(textBox10.Text))
    {
        cust = cust.Where(c => c.Id_Buku ==
textBox10.Text);
    }
    dataGridView1.DataSource = cust;
}

private void button3_Click(object sender, EventArgs e)
{
    // auto generator angka
    autogen();
}

public void autogen()
{
    string num = "1234567890";
    int len = num.Length;
    string otp = string.Empty;
    int otpdigit = 5;
    string finaldigit;

    int getindex;
    for(int i=0; i < otpdigit; i++)
    {
        do
        {
            getindex = new Random().Next(0, len);
            finaldigit =
num.ToCharArray()[getindex].ToString();
        } while (otp.IndexOf(finaldigit) != -1);
        otp += finaldigit;
    }

    label3.Text = (otp);
}

private void comboBox2_SelectedIndexChanged(object
sender, EventArgs e)
{
}

}
```

```
}
```

Penjelasan:

Pada halaman peminjaman, staff atau admin menghubungkan aplikasi dengan database yang berada pada “DESKTOP-PSDHS93\SQLEXPRESS”. Halaman ini memiliki 5 button dan 1 gridview. diantaranya:

- “button1_Click” = digunakan untuk kembali ke halaman Home.
- “Clear_Click” = Digunakan untuk mengosongkan label 3 (Generator angka), combobox1 (IDBuku), dan combobox2(IDStaff).
- “Save_Buku_Click” = untuk menyimpan peminjaman buku dimana No_Pinjam, Id_Buku, Tanggal_Pinjam, Id_Staff, dan Id_Student akan dimasukan kedalam database. Jika peminjaman berhasil, maka akan ada tulisan “Save Peminjaman Berhasil” lalu memanggil laman report. Jika gagal, maka data tidak akan tersimpan.
- “button3_Click” = untuk mengenerate angka yang berasal dari “ public void autogen”. syntax ini berfungsi untuk mengenerate 5 angka dari 0-9 secara acak.
- “button2_Click” = digunakan untuk mencari data dari database yang akan menampilkan kolom data buku yang dicari.
- “peminjaman_Load” merupakan gridview yang akan menunjukan tabel data yang sudah disave pada database. Selain menampilkan data ke gridview, dalam syntax ini juga digunakan untuk menampilkan atau drop idbuku dan idstaff yang terdaftar pada database. syntax pada “public void gridview_data()” yang berfungsi untuk memanggil data - data tersebut dari database menggunakan bahasa sql.

V. Source Code Peminjaman2

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace PROJECT_AKHIR
{
    public partial class Peminjaman2 : Form
    {
        public Peminjaman2()
```

```

{
    InitializeComponent();
}

private void button1_Click(object sender,
EventArgs e)
{
    //backk
    Home panggil = new Home();
    panggil.Show();
    this.Hide();
}

public void gridview_data()
{
    string constring = @"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True";
    SqlConnection conn = new
SqlConnection(constring);

    SqlCommand cmd = new SqlCommand("SELECT *
FROM Peminjaman", conn);
    SqlDataAdapter dataAdapter = new
SqlDataAdapter(cmd);
    DataTable dt = new DataTable();
    dataAdapter.Fill(dt);
    dataGridView1.DataSource = dt;
}

private void Peminjaman2_Load(object sender,
EventArgs e)
{
    gridview_data();
}
}

```

Penjelasan:

Pada halaman data view bagian Peminjaman, disini staff dan admin dapat melihat tampilan data yang melakukan peminjaman. Kemudian terdapat `public void gridview_data` dan ini mempunyai nilai yang dapat dipanggil berkali - kali, sehingga tidak perlu menulis syntax dengan panjang - panjang cukup `gridview_data` saja, dan kemudian dapat tampil isi dalamnya di `dataGridView1.DataSource`.

Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial yang berguna untuk menghubungkan ke dalam database dan SqlCommand cmd = new SqlCommand("SELECT * FROM Peminjaman", conn); untuk memanggil SELECT semuanya yang dari Peminjaman, dan yang terakhir ada tombol back yang berguna untuk balik ke home.

VI. Source Code Pengembalian

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace PROJECT_AKHIR
{
    public partial class pengembalian : Form
    {
        public pengembalian()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender,
EventArgs e)
        {
            //back
            Home panggil = new Home();
            panggil.Show();
            this.Hide();
        }
        private void Clear_Click(object sender,
EventArgs e)
        {

            label3.Text = String.Empty;
            comboBox1.Text = String.Empty;
            comboBox2.Text = String.Empty;
        }

        private void pengembalian_Load(object sender,
EventArgs e)
```

```

{
    // nampilin data grid view
    gridview_data();
    // nampilin data id buku ke combobox
    data_Buku_combo_box();

    // nampilin data id staff ke combobox
    data_idSTAFF_combo_box();

}

// nampilin data grid view
public void gridview_data()
{
    string constring = @"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True";
    SqlConnection conn = new
SqlConnection(constring);

    SqlCommand cmd = new SqlCommand("SELECT *
FROM Peminjaman", conn);
    SqlDataAdapter dataAdapter = new
SqlDataAdapter(cmd);
    DataTable dt = new DataTable();
    dataAdapter.Fill(dt);
    dataGridView1.DataSource = dt;
}

public void data_Buku_combo_box()
{
    SqlConnection conn = new
SqlConnection(@"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True");
    SqlCommand cmd = new SqlCommand("SELECT
Id_Buku FROM DataBuku", conn);
    SqlDataAdapter da = new SqlDataAdapter();
    da.SelectCommand = cmd;
    DataTable table1 = new DataTable();
    da.Fill(table1);

}

// data id staff ke combo box
public void data_idSTAFF_combo_box()
{
    SqlConnection conn = new
SqlConnection(@"Data

```

```

Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True");
        SqlCommand cmd = new SqlCommand("SELECT
Id_Staff FROM Staff", conn);
        SqlDataAdapter da = new SqlDataAdapter();
        da.SelectCommand = cmd;
        DataTable table1 = new DataTable();
        da.Fill(table1);
    }

    DataClasses1DataContext db = new
DataClasses1DataContext();

    private void Save_Buku_Click(object sender,
EventArgs e)
{
    Pengembalian buku = new Pengembalian()
    {
        No_Kembali =
Convert.ToInt32(label2.Text),
        Id_Buku = comboBox1.Text,
        Id_Staff = Convert.ToInt32
(comboBox2.Text),
        Tanggal_Kembali =
Convert.ToDateTime(dateTimePicker1.Value),
        Jumlah_Denda = Convert.ToInt32
(textBox2.Text),
        Id_Student = Convert.ToInt32
(comboBox3.Text)
    };
    db.Pengembalians.InsertOnSubmit(buku);
    try
    {
        db.SubmitChanges();
        MessageBox.Show("Save Pengembalian
Berhasil");
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString());
    }
}

    private void button2_Click(object sender,
EventArgs e)
{
    //button search
}

```

```
        DataClasses1DataContext db = new
DataClasses1DataContext();
        IQueryable<DataBuku> cust = db.DataBukus;
        if (!String.IsNullOrEmpty(textBox10.Text))
{
            cust = cust.Where(c => c.Id_Buku ==
textBox10.Text);
}
        dataGridView1.DataSource = cust;
}

private void button3_Click(object sender,
EventArgs e)
{
    autogen();
}

public void autogen()
{
    string num = "1234567890";
    int len = num.Length;
    string otp = string.Empty;
    int otpdigit = 5;
    string finaldigit;

    int getindex;
    for (int i = 0; i < otpdigit; i++)
{
    do
    {
        getindex = new Random().Next(0,
len);
        finaldigit =
num.ToCharArray()[getindex].ToString();
    } while (otp.IndexOf(finaldigit) != -1);
    otp += finaldigit;
}

    label2.Text = (otp);
}

}
```

Penjelasan:

Pada halaman pengembalian, staff atau admin menghubungkan aplikasi dengan database yang berada pada “DESKTOP-PSDHS93\SQLEXPRESS”. Halaman ini memiliki 5 button dan 1 gridview. diantaranya:

- “button1_Click” = digunakan untuk kembali ke halaman Home.
- “Clear_Click” = Digunakan untuk mengosongkan label 3 (Generator angka), combobox1 (IDBuku), dan combobox2(IDStaff).
- “Save_Buku_Click” = untuk menyimpan pengembalian buku dimana No_Pinjam, Id_Buku, Tanggal_Pinjam, Id_Staff, dan Id_Student akan dimasukan kedalam database. Jika pengembalian berhasil, maka akan ada tulisan “Save Pengembalian Berhasil” lalu memanggil laman report. Jika gagal, maka data tidak akan tersimpan.
- “button3_Click” = untuk mengenerate angka yang berasal dari “public void autogen”. syntax ini berfungsi untuk mengenerate 5 angka dari 0-9 secara acak.
- “button2_Click” = digunakan untuk mencari data dari database yang akan menampilkan kolom data buku yang dicari.
- “pengembalian_Load” merupakan gridview yang akan menunjukan tabel data yang sudah disave pada database. Selain menampilkan data ke gridview, dalam syntax ini juga digunakan untuk menampilkan atau drop idbuku dan idstaff yang terdaftar pada database. syntax pada “public void gridview_data()” yang berfungsi untuk memanggil data - data tersebut dari database menggunakan bahasa sql.

VII. Source Code Pengembalian2

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace PROJECT_AKHIR
{
    public partial class Pengembalian2 : Form
    {
        public Pengembalian2()
        {
            InitializeComponent();
        }
    }
}
```

```

        }

        private void button1_Click(object sender,
EventArgs e)
{
    //backk
    Home panggil = new Home();
    panggil.Show();
    this.Hide();
}

private void Pengembalian2_Load(object sender,
EventArgs e)
{
    gridview_data();
}

public void gridview_data()
{
    string constring = @"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True";
    SqlConnection conn = new
SqlConnection(constring);

    SqlCommand cmd = new SqlCommand("SELECT *
FROM Pengembalian", conn);
    SqlDataAdapter dataAdapter = new
SqlDataAdapter(cmd);
    DataTable dt = new DataTable();
    dataAdapter.Fill(dt);
    dataGridView1.DataSource = dt;
}
}
}

```

Penjelasan:

Halaman data view pada bagian pengembalian ini seorang staff dapat melihat tampilan data yang melakukan peminjaman. Void gridview_data() string constring = @"Data Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial Catalog=PROJECT;Integrated Security=True"; berguna untuk menghubungkan ke dalam database ke dalam Gridview_data. selanjutnya yang terakhir terdapat tombol Back fungsinya untuk kembali ke Home.

VIII. Source Code Book Data

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace PROJECT_AKHIR
{
    public partial class bookData : Form
    {
        public bookData()
        {
            InitializeComponent();
        }

        // ini publioc void ya
        public void gridview_data()
        {
            string constring = @"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True";
            SqlConnection conn = new
SqlConnection(constring);

            SqlCommand cmd = new SqlCommand("SELECT *
FROM DataBuku", conn);
            SqlDataAdapter dataAdapter = new
SqlDataAdapter(cmd);
            DataTable dt = new DataTable();
            dataAdapter.Fill(dt);
            dt.Clear();
            dataAdapter.Fill(dt);
            dataGridView1.DataSource = dt;
        }
    }

    private void button3_Click(object sender,
EventArgs e)
{
    //INSERT
}

```

```
        DataClasses1DataContext db = new
DataClasses1DataContext();
        DataBuku buku = new DataBuku()
{
    Id_Buku = textBox2.Text,
    Judul_Buku = textBox1.Text,
    Kuantitas = Convert.ToInt32
(textBox6.Text),
    Kategori = textBox3.Text,
    Penerbit = textBox4.Text,
};
db.DataBukus.InsertOnSubmit(buku);
try
{
    db.SubmitChanges();
    MessageBox.Show("Insert ke Database
Berhasil");
    gridview_data();

}
catch (Exception ex)
{
    MessageBox.Show(ex.ToString());
}
}

private void button2_Click(object sender,
EventArgs e)
{
    //back

    Home panggil = new Home();
    panggil.Show();
    this.Hide();
}

private void button4_Click(object sender,
EventArgs e)
{
    //update

    DataClasses1DataContext db = new
DataClasses1DataContext();

    var ubah = (from a in db.DataBukus
                where a.Id_Buku == textBox2.Text
                select a).Single();
    ubah.Judul_Buku = textBox1.Text;
    ubah.Kategori = textBox3.Text;
```

```

        ubah.Penerbit = textBox4.Text;
        ubah.Kuantitas =
Convert.ToInt32(textBox6.Text);
        try
        {
            db.SubmitChanges();
            MessageBox.Show("Berhasil Update Data
Buku");
            gridview_data();
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.ToString());
        }
    }

    private void button5_Click(object sender,
EventArgs e)
{
    DataClasses1DataContext db = new
DataClasses1DataContext();
    var delete = (from a in db.DataBukus
              where a.Id_Buku ==
textBox2.Text
              select a).Single();

    db.DataBukus.DeleteOnSubmit(delete);
    try
    {
        db.SubmitChanges();
        MessageBox.Show("Berhasil Delete Data
Buku");
        gridview_data();
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.ToString());
    }
}

private void button1_Click(object sender,
EventArgs e)
{
    //button search
    DataClasses1DataContext db = new
DataClasses1DataContext();
    IQueryable<DataBuku> cust = db.DataBukus;
    if (!String.IsNullOrEmpty(textBox5.Text))
    {
}

```

```
        cust = cust.Where(c => c.Id_Buku ==
textBox5.Text);
    }

    dataGridView1.DataSource = cust;
}

private void
dataGridView1_CellContentClick(object sender,
DataGridViewCellEventArgs e)
{
}
}
```

Penjelasan:

Pertama menggunakan using System.Data.SqlClient; dalam langkah awal menggunakan syntax agar dapat terhubung ke dalam Sql. Setelah itu terdapat grid view data yang berfungsi untuk menampilkan semua isi kolom dari tabel data buku dan nantinya akan ditampilkan ke datagridview1. Setelah itu terdapat tombol search yang berfungsi untuk melakukan create data ke sql. kemudian admin dapat menuliskan Id_Buku, Judul_Buku, Kuantitas, Kategori, dan Penerbit yang dapat kita ambil dari dataclasses1datacontext dan dari tabel databuku.

Dari deskripsi yang telah dituliskan diatas, maka dapat diambil manfaatnya berupa menyamakan sesuai dengan textbox / combobox yang diisi oleh staff. Setelah itu terdapat try dan catch yang berfungsi untuk jika datanya berhasil dimasukkan, maka MessageBox.Show("Insert ke Database Berhasil"); akan dieksekusi. Namun, bila gagal akan masuk ke catch yaitu MessageBox.Show(ex.ToString()); Terdapat juga menu update yang berfungsi untuk memperbarui data yang terdapat di dalam database.

Pada bagian terakhir terdapat tombol Delete yang berfungsi untuk menghapus data dari database. Syntax yang dipakai diantaranya hampir-hampir mirip, namun yang membedakannya ialah pada update dan delete staff harus mempertahankan nilai Primary Key yang nantinya dapat digunakan sebagai nilai update dan nilai delete. Setelah itu ada tombol back yang digunakan sebagai tombol kembali dari halaman sekarang menuju ke halaman home. Lalu yang terakhir adalah tombol search yang digunakan untuk mencari Id_Buku yang nantinya akan tampil di DataGridView yang terletak di bawah tombol search dan text box yang tersedia.

IX. Source Code Data Buku2

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace PROJECT_AKHIR
{
    public partial class DataBuku2 : Form
    {
        public DataBuku2()
        {
            InitializeComponent();
        }

        private void DataBuku2_Load(object sender,
EventArgs e)
        {
            gridview_data();
        }

        public void gridview_data()
        {
            string constring = @"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True";
            SqlConnection conn = new
SqlConnection(constring);

            SqlCommand cmd = new SqlCommand("SELECT *
FROM DataBuku", conn);
            SqlDataAdapter dataAdapter = new
SqlDataAdapter(cmd);
            DataTable dt = new DataTable();
            dataAdapter.Fill(dt);
            dataGridView1.DataSource = dt;
        }

        private void button1_Click(object sender,
EventArgs e)
        {
            //backk
            Home panggil = new Home();
        }
    }
}

```

```
        panggil.Show();
        this.Hide();
    }
}
```

Penjelasan:

Pada halaman data view yang bagian Data Buku, disini staff dan admin dapat melihat tampilan data yang melakukan tampilan data buku yang tersedia di perpustakaan yang terdiri atas nomor identitas buku, judul buku, kuantitas, kategori, dan penerbit buku.

“public void gridView_data” yang mempunyai nilai yang dapat dipanggil berkali - kali, sehingga tidak perlu menulis syntax dengan panjang - panjang cukup gridView_data saja, dan kemudian dapat tampil isi dalamnya di `dataGridView1.DataSource = DESKTOP-PSDHS93\SQLEXPRESS; Initial` yang berguna untuk menghubungkan ke dalam database dan `SqlCommand cmd = new SqlCommand("SELECT * FROM DataBuku", conn);` untuk memanggil SELECT semuanya yang dari Data Buku, dan yang terakhir ada tombol back yang berguna untuk balik ke home.

XI. Source Code Laporan Data Buku

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace PROJECT_AKHIR
{
    public partial class LaporanDataBuku : Form
    {
        public LaporanDataBuku()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender,
EventArgs e)
        {
            CrystalReport3 VCR = new CrystalReport3();
```

```

        crystalReportViewer1.ReportSource = VCR;
    }

    private void crystalReportViewer1_Load(object
sender, EventArgs e)
{
}

private void button2_Click(object sender,
EventArgs e)
{
    Home panggil = new Home();

    panggil.Show();
    this.Hide();
}
}
}

```

Penjelasan :

Pada Source Code Laporan Data Buku, bisa dilihat syntax `private void button1_Click` terdapat Crystalreport3 yang di panggil menggunakan variabel VCR untuk ditampilkan dalam crystal report dengan bantuan `crystalReportViewer1`.

Pada `private void button2_Click` yang merupakan syntax dari button back digunakan untuk mengembalikan administrator ke dalam halaman homepage.

Halaman home di panggil dengan variabel Home dimana ketika “panggil” maka akan show, dan form LaporanDataBuku akan di hide.

XII. Source Code Laporan Data Peminjam

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace PROJECT_AKHIR
{
    public partial class LaporanDataPeminjam : Form

```

```
{  
    public LaporanDataPeminjam()  
    {  
        InitializeComponent();  
    }  
  
    private void LaporanDataPeminjam_Load(object sender,  
EventArgs e)  
    {  
        NoPinjam_combo_box();  
    }  
  
    public void NoPinjam_combo_box()  
    {  
        SqlConnection conn = new SqlConnection(@"Data  
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial  
Catalog=PROJECT;Integrated Security=True");  
        SqlCommand cmd = new SqlCommand("SELECT Id_Buku  
FROM Peminjaman", conn);  
        SqlDataAdapter da = new SqlDataAdapter();  
        da.SelectCommand = cmd;  
        DataTable table1 = new DataTable();  
        da.Fill(table1);  
        comboBox1.DataSource = table1;  
        comboBox1.DisplayMember = "Id_Buku";  
        comboBox1.ValueMember = "Id_Buku";  
    }  
  
    private void button1_Click(object sender, EventArgs e)  
    {  
        CrystalReport4 vcr = new CrystalReport4();  
        vcr.setParameterValue("P_IdBuku", comboBox1.Text);  
        crystalReportViewer1.ReportSource = vcr;  
    }  
  
    private void crystalReportViewer1_Load(object sender,  
EventArgs e)  
    {  
    }  
  
    private void button2_Click(object sender, EventArgs e)  
    {  
        Home panggil = new Home();  
        panggil.Show();  
        this.Hide();  
    }  
}
```

```

    }
}
}
```

Penjelasan :

Pada source code LaporanDataPeminjam terdapat syntax pada `public void NoPinjam_combo_box` yang digunakan untuk menjalankan fungsi combo box dimana isi nomor data buku merupakan nomor yang sudah ada dalam SQL. Administrator akan terhubung dengan koneksi SQL dari desktop yang digunakan dengan tujuan memanggil data id buku dari tabel peminjaman.

Pada `private void button2_Click` yang merupakan syntax dari button back digunakan untuk mengembalikan administrator ke dalam halaman homepage.

Halaman home di panggil dengan variabel Home dimana ketika “panggil” maka akan show, dan form LaporanDataBuku akan di hide.

XIII. Source Code Report

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace PROJECT_AKHIR
{
    public partial class Report : Form
    {
        SqlConnection Con = new SqlConnection(@"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True");

        public Report()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            Home panggil = new Home();
            panggil.Show();
            this.Hide();
        }
}
```

```

    }
}

```

```

public void gridview_data()
{
    string constring = @"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True";
    SqlConnection conn = new SqlConnection(constring);

    SqlCommand cmd = new SqlCommand("SELECT * FROM
Peminjaman", conn);
    SqlDataAdapter dataAdapter = new
SqlDataAdapter(cmd);
    DataTable dt = new DataTable();
    dataAdapter.Fill(dt);
    dataGridView1.DataSource = dt;
}

private void Report_Load(object sender, EventArgs e)
{
    gridview_data();
}
private void txtSearch_TextChanged(object sender,
EventArgs e)
{
    SqlConnection Con = new SqlConnection(@"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True");

    if (cmbColumn.Text == "No_Pinjam")
    {
        SqlDataAdapter sda = new
SqlDataAdapter("SELECT No_Pinjam, Id_Buku, Tanggal_Pinjam,
Id_Staff FROM Peminjaman where No_Pinjam like'" +
txtSearch.Text + "%'",Con);
        DataTable dt = new DataTable();
        sda.Fill(dt);
        dataGridView1.DataSource = dt;
    }
    else if (cmbColumn.Text == "Id_Buku")
    {
        SqlDataAdapter sda = new
SqlDataAdapter("SELECT Id_Buku, No_Pinjam, Tanggal_Pinjam,
Id_Staff FROM Peminjaman where Id_Buku like'" +
txtSearch.Text + "%'", Con);
        DataTable dt = new DataTable();
        sda.Fill(dt);
    }
}

```

```

        dataGridView1.DataSource = dt;
    }

    else if (cmbColumn.Text == "Id_Staff")
    {
        SqlDataAdapter sda = new
SqlDataAdapter("SELECT Id_Staff, No_Pinjam, Id_Buku,
Tanggal_Pinjam FROM Peminjaman where Judul_Buku like'" +
txtSearch.Text + "%'", Con);
        DataTable dt = new DataTable();
        sda.Fill(dt);
        dataGridView1.DataSource = dt;
    }

    else if (cmbColumn.Text == "-")
    {
        SqlDataAdapter sda = new
SqlDataAdapter("SELECT * FROM Peminjaman where Tanggal_Pinjam
like'" + txtSearch.Text + "%'", Con);
        DataTable dt = new DataTable();
        sda.Fill(dt);
        dataGridView1.DataSource = dt;
    }

}
}
}

```

Penjelasan :

Form ini mendeklarasikan `SqlConnection` yang terhubung dengan database pada Sql yang nantinya gunakan pada bagian yang diperlukan. Dengan menggunakan `public void gridview_data` yang berfungsi untuk menampilkan semua data dari tabel Peminjaman dan nantinya ditampilkan ke `datagridView1`.

Terdapat text box dan combo box yang berguna untuk melakukan filter data yang ingin dicari. Dari combo box yang tersedia memasukan nilai nilai yang ingin menjadi kata kunci dari staff nantinya.

Pada text box, terdapat syntax if else yang menjadi kunci utama dalam melakukan pencarian data oleh staff. Contoh salah satunya, Jika combo box = No_Pinjam, maka di datagridviewnya akan menampilkan sesuai dari data yang dicari, dan begitu pula dengan if else yang lainnya.

XIV. Source Code Staff2

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;

```

```
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace PROJECT_AKHIR
{
    public partial class Staff2 : Form
    {
        public Staff2()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender,
EventArgs e)
        {
            //backk
            Home panggil = new Home();
            panggil.Show();
            this.Hide();
        }

        private void Staff2_Load(object sender,
EventArgs e)
        {
            gridview_data();
        }

        public void gridview_data()
        {
            string constring = @"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True";
            SqlConnection conn = new
SqlConnection(constring);

            SqlCommand cmd = new SqlCommand("SELECT
Id_Staff, Nama_Staff, Gender_Staff FROM Staff", conn);
            SqlDataAdapter dataAdapter = new
SqlDataAdapter(cmd);
            DataTable dt = new DataTable();
            dataAdapter.Fill(dt);
            dataGridView1.DataSource = dt;
        }
    }
}
```

```
}
```

Penjelasan:

Halaman data view bagian Staff menunjukkan informasi dari staff, yaitu nomor identitas staff, nama staff, dan jenis kelamin dari staff (L/P).

“`public void gridview_data`”, mempunyai nilai yang dapat dipanggil berkali - kali, sehingga tidak perlu menulis syntax dengan panjang. hanya dengan menggunakan `gridview_data`, dan dapat tampil isi dalamnya pada

```
dataGridView1.DataSource.Source=DESKTOP-PSDHS93\SQLEXP  
RESS;Initial yang berguna untuk menghubungkan ke dalam database dan  
SqlCommand cmd = new SqlCommand("SELECT Id_Staff, Nama_Staff,  
Gender_Staff FROM Staff", conn);  
untuk memanggil SELECT semuanya yang dari data Staff, dan yang terakhir  
ada tombol back yang berguna untuk balik ke home.
```

XV. Source Code Student2

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
using System.Windows.Forms;  
using System.Data.SqlClient;  
  
namespace PROJECT_AKHIR  
{  
    public partial class Student2 : Form  
    {  
        public Student2()  
        {  
            InitializeComponent();  
        }  
  
        private void button1_Click(object sender,  
EventArgs e)  
        {  
            //backk  
            Home panggil = new Home();  
            panggil.Show();  
        }  
    }  
}
```

```
        this.Hide();
    }

    private void Student2_Load(object sender,
EventArgs e)
{
    gridview_data();
}

public void gridview_data()
{
    string constring = @"Data
Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial
Catalog=PROJECT;Integrated Security=True";
    SqlConnection conn = new
SqlConnection(constring);

    SqlCommand cmd = new SqlCommand("SELECT *
FROM Student", conn);
    SqlDataAdapter dataAdapter = new
SqlDataAdapter(cmd);
    DataTable dt = new DataTable();
    dataAdapter.Fill(dt);
    dataGridView1.DataSource = dt;
}
}
```

Penjelasan:

Pada halaman data view bagian Student menunjukkan data dari siswa/siswi yang pernah melakukan peminjaman dan pengembalian buku dari perpustakaan. Data ini meliputi nomor identitas, nama, alamat rumah, nomor handphone, nomor pengembalian buku, dan jumlah denda dari siswa/siswi.

“public void gridview_data”, mempunyai nilai yang dapat dipanggil berkali - kali, sehingga tidak perlu menulis syntax dengan panjang. hanya dengan menggunakan gridview_data, dan dapat tampil isi dalamnya pada

```
“GridView1.DataSource.Source=DESKTOP-PSDHS93\SQLEXPRESS;Initial”
```

yang digunakan untuk menghubungkan kedalam database dan new SqlCommand("SELECT * FROM Student", conn); untuk memanggil SELECT semuanya yang dari data Student, dan yang terakhir ada tombol back yang berguna untuk balik ke home.

XVI. Source Code Notes

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.IO;
using System.Text;

namespace PROJECT_AKHIR
{
    public partial class Notes : Form
    {
        public Notes()
        {
            InitializeComponent();
        }

        private void Notes_Load(object sender, EventArgs e)
        {

        }
        static string Baca()

        {
            FileStream fs = new
FileStream("D:\\testtxt\\notes.txt", FileMode.OpenOrCreate,
FileAccess.Read, FileShare.ReadWrite);
            StreamReader sr = new StreamReader(fs);
            return (sr.ReadToEnd());
            sr.Close();
            fs.Close();
        }

        static void Tulis(string kata)

        {
            FileStream fs = new
FileStream("D:\\testtxt\\notes.txt", FileMode.Append,
FileAccess.Write, FileShare.ReadWrite);
            StreamWriter sw = new StreamWriter(fs);
            sw.WriteLine(kata);
            sw.Close();
            fs.Close();
        }
}
```

```
}

private void button1_Click(object sender, EventArgs e)
{
    Tulis(textBox1.Text);
    textBox1.Clear();
}

private void button2_Click(object sender, EventArgs e)
{
    textBox1.Text = Baca();
}

private void button3_Click(object sender, EventArgs e)
{
    Home panggil = new Home();
    panggil.Show();
    this.Hide();
}

private void textBox1_TextChanged(object sender,
EventArgs e)
{
}

}

}
```

Penjelasan :

- Form notes pertama harus mendeklarasikan using System.IO; dan using System.Text; untuk dapat melakukan write dan read pada file .txt.
- Form ini menggunakan 2 buah static string yang berfungsi untuk melakukan read dan write pada file txt.
- Untuk static string read and write harus dituliskan direktori yang sama sama digunakan untuk mengakses dari file yang ada pada direktori yang digunakan.
- Setelah itu nama dari static void tersebut dapat dituliskan ke dalam button sesuai dengan kebutuhannya masing masing.
- Terakhir adalah tombol back yang digunakan untuk kembali ke form home.

DAFTAR PUSTAKA

- American Society for Quality. (2022). *What is a flowchart?* Excellence Through Quality | ASQ. <https://asq.org/quality-resources/flowchart>
- Biscobing, J. (2019, September 11). *What is entity relationship diagram (ERD)? - Definition* from *WhatIs.com*. SearchDataManagement. <https://www.techtarget.com/searchdatamanagement/definition/entity-relationship-diagram-ERD>
- Department of Computer Science, University of Regina. (2017). *Data modeling and entity relationship diagram (ERD)*. <https://www.cs.uregina.ca/Links/class-info/215/erd/>
- Fauzan, R., Siahaan, D., Rochimah, S., & Triandini, E. (2021). A different approach on automated use case diagram semantic assessment. *International Journal of Intelligent Engineering and Systems*, 14(1), 496-505. <https://doi.org/10.22266/ijies2021.0228.46>
- Firmansyah, B., & Meza, A. (2016). SISTEM INFORMASI ADMINISTRASI PERPUSTAKAAN BERBASIS WEB PADA SMA NEGERI 5 TANGERANG. *JURNAL SISTEM INFORMASI STMIK ANTAR BANGSA*, 1(1), 67-70. <https://media.neliti.com/media/publications/335125-sistem-informasi-administrasi-perpustaka-0effe2a3.pdf>
- GeeksforGeeks. (2020, April 29). *Functional vs non functional requirements*. GeeksforGeeks.org. <https://www.geeksforgeeks.org/functional-vs-non-functional-requirements/>
- Hutauruk, M. K. (2019, November 26). *UML diagram : Use case diagram*. School of Computer Science. <https://socs.binus.ac.id/2019/11/26/uml-diagram-use-case-diagram/>
- Javapoint. (n.d.). *dbms-Normalization*. <https://www.javatpoint.com/dbms-normalization>
- Jayaram, P. (2018, November 27). *What is database normalization in SQL server?* SQL Shack - articles about database auditing, server performance, data recovery, and more. <https://www.sqlshack.com/what-is-database-normalization-in-sql-server/>

- MaryQiu. (2022, May 5). *Database normalization description - Office*. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/en-us/office/troubleshoot/access/database-normalization-description>
- Pradiptamukherjee. (2021, November 5). *What is a flowchart and its types?* GeeksforGeeks. <https://www.geeksforgeeks.org/what-is-a-flowchart-and-its-types/>
- Rahmalia, N. (2021, January 27). *Diagram use case, Teknik Penyederhanaan agar Perancangan Sistem Jadi Mudah.* Glints. <https://glints.com/id/lowongan/use-case-adalah/>
- Sabharwal, S., Kaur, P., & Sibal, R. (2017). Empirical and theoretical validation of a use case diagram complexity metric. *International Journal of Information Technology and Computer Science*, 9(11), 35-47. <https://doi.org/10.5815/ijitcs.2017.11.04>
- Setiyadi, D. (2018). Normalisasi Dalam Perancangan Basis Data Relasional Purchase Order (PO). *INFORMATICS FOR EDUCATORS AND PROFESSIONAL*, 3(1), 67-78. <http://ejournal-binainsani.ac.id/index.php/ITBI/article/view/1046>
- Susilo, G. (2016). KEAMANAN BASIS DATA PADA SISTEM INFORMASI DI ERA GLOBAL. *TRANSFORMASI Jurnal Informasi & Pengembangan Iptek*, 12(2), 78-87. <https://ejournal.stmikbinapatria.ac.id/index.php/JT/article/download/70/41>
- Wijayakusuma, I. G., & Wiranata, I. K. (2018). Perancangan basis data E-library program Studi Matematika FMIPA Universitas Udayana. *Jurnal Matematika*, 8(1), 63. <https://doi.org/10.24843/jmat.2018.v08.i01.p98>

Tugas Anggota:

- Agym Dimas Tian (00000058149)
Membantu membuat codingan aplikasi, Membantu membuat syntax sql,
Membantu merangkai laporan porposal.

- Angeline Marcellina Chandrawijaya (00000057862)
Membantu membuat codingan aplikasi, Membantu membuat syntax sql,
Membantu merangkai laporan porposal, Membuat desain Aplikasi, .

- Benz Martheen W Jonash (00000059822)
Membantu membuat codingan aplikasi, Membantu membuat syntax sql,
Membantu merangkai laporan porposal,

- Jericho Alexander Gunawan (00000058345)
Membantu membuat codingan aplikasi, Membantu membuat syntax sql,
Membantu merangkai laporan porposal, membantu mendesign aplikasi.

- Nicholas Suwandi Wijaya (00000057903)
Membuat codingan aplikasi, Membuat syntax Sql, Membuat fungsi tools
pada aplikasi, Membantu merangkai laporan porposal.

- Rheinald Adyatma Rizqi Wibowo (00000058204)
Membantu membuat codingan aplikasi, Membuat fungsi tools pada aplikasi,
Merangkai dan mengedit laporan porposal, Membantu membuat syntax sql.