

Aplikasi Sistem Manajemen Perpustakaan dengan Penerapan Pemrograman Berorientasi Objek

Angelina Jolie*, Dedrick, Raphael Kurniawan Sugeng, Winona Audrey Lee, Andik Yulianto

Teknologi Informasi, Fakultas Ilmu Komputer, Universitas Internasional Batam, Baloi-Sei Ladi, Jalan Gajah Mada, Tiban Indah, Kecamatan Sekupang, Kota Batam, Kepulauan Riau, 29426, Indonesia
e-mail: 2132018.angelina@uib.edu

Abstrak

Perpustakaan adalah salah satu pilar utama dalam meningkatkan mutu pendidikan di Indonesia. Oleh karena keberadaan yang penting bagi generasi penerus bangsa, Indonesia berusaha untuk meningkatkan jumlah pembangunan sekolah dan perpustakaan agar generasi muda Indonesia saat ini bisa bersaing dengan orang-orang dari negara lain. Namun, jumlah perpustakaan yang meningkat belum tentu akan sejalan dengan efektivitas pelayanan yang diberikan. Banyak perpustakaan di Indonesia yang belum mengikuti perkembangan teknologi dan masih menggunakan sistem pengelolaan manual sehingga proses manajemen dan pelayanan di perpustakaan menjadi tidak efektif. Oleh karena itu, dibuatlah penelitian ini dengan tujuan untuk menghasilkan rancangan aplikasi manajemen dan pengaturan perpustakaan. Metode yang akan digunakan dalam penelitian ini terbagi menjadi beberapa tahapan yaitu perancangan sistem, perancangan class diagram, perancangan basis data MySQL, dan perancangan GUI aplikasi. Melalui proses-proses tersebut dihasilkan sebuah aplikasi manajemen perpustakaan berbasis desktop yang terintegrasi dengan basis data. Dengan adanya aplikasi tersebut, proses pengelolaan, pencatatan dan penyajian data dapat dilakukan dengan cara yang lebih efektif dan efisien.

Kata kunci: Sistem Manajemen, Basis Data, MySQL, Aplikasi GUI, Perpustakaan, Aplikasi Berbasis Desktop

Abstract

Library is one of the most important pillars to improve the quality of Indonesia's education. Due to its importance for the next generation, the government tries to increase the establishment of schools and libraries so that the young generation can compete with people from another countries. However, the increasing number of libraries did not mean that the library service quality will become better. Many library systems in Indonesia still have not kept up with technology and still using the manual system, hence the management process and services are not effective. Because of that, this research is created for the purpose of producing a design for a library management application. The methods used in this research is split into multiple steps which is system planning, class diagram planning, MySQL database planning, and GUI planning. The following processes will produce a library management desktop application that is integrated to a database. With this application, processing, documenting, and deliverance of data can be done in more effective and efficient way.

Keywords: Management System, Database, MySQL, GUI Application, Library, Desktop Applications

Copyright © TELCOMATICS Journal. All rights reserved

I. PENDAHULUAN

Perpustakaan sudah sejak lama dikenal sebagai salah satu pilar utama dalam menyebarkan dan meningkatkan mutu pendidikan di Indonesia, bersama dengan institusi lainnya seperti sekolah dan universitas. Oleh karena keberadaan yang penting bagi generasi penerus bangsa, Indonesia berusaha untuk meningkatkan jumlah pembangunan

sekolah dan perpustakaan agar generasi muda Indonesia saat ini bisa bersaing dengan orang-orang dari negara lain.

Berdasarkan penelitian dari Badan Pusat Statistika, ditemukan bahwa perpustakaan yang terdapat di Indonesia saat ini memiliki jumlah sekitar 10.000 perpustakaan. Bersamaan dengan itu, nilai kegemaran baca penduduk juga meningkat menjadi nilai 59,52 pada tahun 2022

[1]. Peningkatan ini mendorong pemerintah untuk semakin menggiatkan pembangunan institusi pendidikan dan perpustakaan di Indonesia. Perpustakaan yang memiliki fungsi sebagai penyedia sumber informasi dan ilmu pengetahuan diharapkan dapat berperan juga sebagai sarana yang dapat mendidik, meningkatkan kecerdasan dan memberdayakan generasi muda Indonesia agar bisa bersaing dengan orang-orang dari negara lain.

Namun meningkatnya jumlah perpustakaan di Indonesia belum tentu berkorelasi dengan efektivitas dan kualitas yang dirasakan oleh para pekerja dan pengguna perpustakaan. Seiring dengan berkembangnya teknologi di dunia, perpustakaan juga didorong untuk mengembangkan dan mengimplementasikan penggunaan teknologi di dalam aktivitas sirkulasi perpustakaan. Menurut penelitian dari International Federation of Library Association and Institution (IFLA), dari 2,7 juta perpustakaan yang ada di dunia, hanya terdapat 373 ribu atau 13% dari total perpustakaan yang telah mengadopsi teknologi dan bisa mengakses internet [2]. Hal ini menunjukkan bahwa adopsi teknologi masih kurang diterapkan di Indonesia. Penerapan teknologi dianjurkan dilakukan dengan tujuan agar perpustakaan dapat bersaing dengan perpustakaan atau sarana penyedia informasi lain yang telah menerapkan teknologi tersebut. Namun penggunaan teknologi tersebut harus didukung dengan sistem implementasi yang tepat, karena peranannya yang akan berpengaruh besar terhadap penurunan ataupun peningkatan kualitas pelayanan yang terdapat di perpustakaan.

Berdasarkan uraian latar belakang tersebut, dilakukanlah penelitian ini yang bertujuan untuk menghasilkan suatu rancangan aplikasi yang dapat membantu perpustakaan untuk meningkatkan kualitas pengelolaan dan pelayanannya. Aplikasi yang dihasilkan diharapkan dapat mengurangi beban operasional dan juga mempermudah aktivitas dan pekerjaan bagi staff yang ada di dalam perpustakaan.

II. TINJAUAN PUSTAKA

A. Object Oriented Programming (OOP)

OOP adalah suatu metode pemrograman aplikasi yang menggunakan beberapa struktur data bernama *object* untuk mengurangi

ketergantungan dengan struktur data lain dan perintah-perintah dalam aplikasi [3].

B. Kelas (Class) dan Objek

Class adalah suatu kerangka yang berisi beberapa perintah operasional atau fungsi dan juga beberapa atribut yang belum terdefiniskan dan belum dijadikan suatu *object* [3]. Sedangkan *Object* adalah kumpulan struktur data yang terbentuk dari suatu kelas dan memiliki atribut-atribut yang terdefinisi dan saling berhubungan dengan perintah operasional atau fungsi [3].

C. Python

Python adalah bahasa pemrograman tingkat tinggi untuk menerjemahkan perintah-perintah yang terdapat dalam kode aplikasi menjadi fungsi yang bisa dijalankan oleh aplikasi. Berbeda dengan bahasa pemrograman yang lain, *Python* menyerupai bahasa yang sering digunakan oleh manusia sehingga dapat dibaca dan diprogram dengan lebih mudah [4].

D. MySQL

MySQL adalah basis data relasional yang dibentuk dari bahasa pemrograman *Structured Query Language* dan merupakan sistem manajemen basis data yang sering digunakan untuk menyimpan data-data dari aplikasi yang bersifat data *table* [5].

E. Inheritance

Inheritance adalah fitur dari *OOP* dimana fungsi-fungsi dari suatu kelas dapat dipinjam dan dipakai di kelas objek lainnya. Penerapan *inheritance* ini dilakukan untuk mengurangi jumlah kode di aplikasi sehingga tidak terjadi duplikasi data dan juga membuat aplikasi dapat berjalan lebih cepat [6].

F. Abstract Class

Abstract class adalah suatu kelas dengan hirarki tertinggi di *OOP* yang memiliki metode dan fungsi, namun tidak memiliki implementasi tertentu dan tidak dapat diubah menjadi *object* kelas. *Abstract class* dapat diterapkan dengan menetapkan *Abstract method* pada suatu kelas [6].

G. XAMPP

XAMPP adalah suatu aplikasi yang menjalankan situs berbasis Apache Webserver untuk menjalankan dan menampilkan data-data

yang terdapat dalam basis data *MySQL* dengan mudah [7].

H. Database (Basis Data)

Database atau basis data adalah sekumpulan data atau informasi terstruktur yang tersimpan di dalam suatu komputer dan biasanya ditulis dalam Bahasa pemrograman *SQL* atau *Structured Query Language* [8].

I. Structured Query Language (SQL)

Structured Query Language adalah bahasa pemrograman yang biasa digunakan di hampir semua basis data berrelasional. *SQL* memberikan akses untuk *query* memanipulasi dan menambahkan data ke dalam basis data [9].

J. Graphical User Interface (GUI)

GUI adalah Suatu tampilan yang digunakan untuk memudahkan pengguna atau user suatu aplikasi untuk bergerak antar aplikasi atau menjalankan suatu perintah didalam aplikasi dengan mudah [10].

K. Visual Studio Code

Visual Studio Code adalah sebuah aplikasi yang digunakan untuk membantu pengguna atau programmer untuk memprogram suatu aplikasi dan membuatnya menjadi suatu aplikasi yang dapat digunakan. *Visual Studio Code* dapat di integrasi dengan berbagai macam bahasa pemrograman seperti *Python* [11].

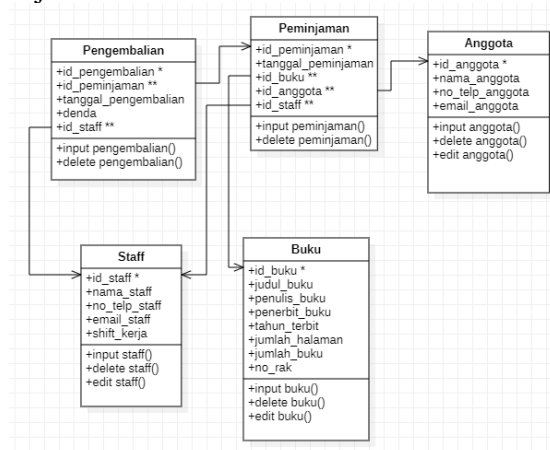
L. Tkinter

Tkinter adalah suatu peralatan tambahan yang ada dalam bahasa pemrograman *Python* untuk membuat suatu *GUI* aplikasi yang menggunakan bahasa pemrograman *Python*. Dengan aplikasi ini, *GUI* dapat dibuat oleh programmer untuk membantu *user* dalam menggunakan aplikasi [12].

akan tertanam dalam server lokal dan tidak bisa diakses oleh *user* yang berada di tempat lain. *User* akan mengakses aplikasi utama melalui komputer di perpustakaan. Kemudian *user* akan mengakses aplikasi dan mengirimkan perintah untuk mengambil data dari basis data yang ada di komputer. Jika basis data memiliki data yang diminta oleh *user*, maka basis data akan mengirimkan kembali data tersebut ke dalam aplikasi agar dapat dilihat oleh *user*. Jika basis data tidak memiliki data yang diminta oleh *user*, maka basis data tidak dapat menampilkan data yang diinginkan oleh *user* dan akan muncul pesan bahwa data tersebut tidak terdapat dalam *database*. Kemudian aplikasi akan meminta *user* untuk meminta data lain yang sudah ada.

A. Perancangan Class Diagram

Perancangan *class diagram* dilakukan untuk menjelaskan struktur dari sistem perpustakaan yang akan dikembangkan ditunjukkan pada Gambar 1. *Class diagram* umumnya terdiri dari tiga komponen yaitu komponen atas yang berisi nama kelas, komponen tengah yang terdiri dari atas atribut suatu kelas, dan komponen bawah yang berisi metode yang bisa dilakukan oleh objek kelas.



Gambar 1. Class Diagram Sistem Perpustakaan

III. METODE PENELITIAN

Perancangan sistem manajemen perpustakaan akan berbasis *desktop* dengan mengintegrasikan *database*, penggunaan peralatan elektronik (komputer atau laptop), dan proses yang terkomputerisasi. *Desktop based* dalam hal ini berarti aplikasi yang dibuat hanya akan memberikan pelayanan informasi perpustakaan di tempat secara langsung. Sistem

Pada *class diagram* yang telah dibuat, terdapat lima kelas yang terdiri dari kelas buku, anggota, staff, peminjaman dan pengembalian.

Pada Kelas Buku terdapat delapan atribut sebagai berikut:

1. Id Buku, berisi ID identifikasi yang membedakan satu buku dengan buku lainnya. Dengan adanya ID buku, staff

perpustakaan dapat menemukan buku yang diinginkan dengan cara yang lebih mudah dan cepat.

2. Judul Buku, berisi nama judul buku yang akan disimpan sebagai *object*.
3. Penulis Buku, berisi nama penulis yang mengarang suatu buku.
4. Penerbit Buku, berisi nama penerbit yang menerbitkan buku tersebut.
5. Tahun Terbit, berisi tahun dimana buku tersebut diterbitkan.
6. Jumlah Halaman, yaitu atribut yang berisi data jumlah halaman dari buku yang disimpan.
7. Jumlah Buku, yaitu stok buku yang terdapat di perpustakaan pada waktu tersebut. Jumlah buku ini akan berubah-ubah sesuai dengan proses peminjaman dan pengembalian yang terjadi.
8. No. Rak, yaitu penomoran rak yang menunjukkan lokasi buku yang disimpan. Dengan demikian buku dapat didapatkan secara mudah karena lokasinya diketahui secara jelas.

Sedangkan untuk metode yang digunakan pada kelas buku antara lain:

1. Input Buku, yang akan digunakan untuk memasukkan data-data buku baru yang terdapat di perpustakaan.
2. *Delete* Buku, yang akan digunakan untuk menghapus buku yang sudah dimasukkan.
3. Edit Buku, yang akan digunakan untuk mengubah informasi suatu buku.

Pada kelas buku terdapat delapan atribut sebagai berikut:

1. Id Anggota, berisi ID identifikasi anggota perpustakaan yang bersifat unik dan dapat membedakan antar satu anggota dengan anggota lain. ID ini akan didapatkan oleh pengunjung perpustakaan yang telah mendaftar sebagai anggota.
2. Nama Anggota, berisi nama lengkap dari anggota yang terdaftar di perpustakaan.
3. No Telp Anggota, berisi nomor telepon anggota agar bisa dihubungi oleh pihak perpustakaan.
4. Email Anggota, berisi informasi tentang email yang dimiliki oleh anggota.

Sedangkan untuk metode yang digunakan pada kelas buku antara lain:

1. Input Anggota, yang akan digunakan untuk memasukkan identitas profil pengunjung yang ingin mendaftar sebagai anggota perpustakaan.
2. *Delete* Anggota, yang akan digunakan untuk menghapus anggota yang sudah tidak aktif atau anggota yang ingin keluar dari keanggotaan perpustakaan.
3. Edit Anggota, yang akan digunakan untuk mengubah informasi anggota.

Beberapa atribut yang terdapat dalam kelas staff antara lain sebagai berikut:

1. Id Staff, berisi ID identifikasi staff perpustakaan yang bersifat unik dan dapat membedakan antar satu staff dengan staff lain.
2. Nama Staff, berisi nama lengkap dari staff yang bekerja di perpustakaan.
3. No Telp Staff, berisi nomor telepon staff yang bekerja di perpustakaan.
4. Email Staff, berisi informasi tentang email yang dimiliki oleh staff.
5. Shift kerja, berisi informasi waktu kerja staff, yaitu pagi atau sore.

Kelas Staff juga memiliki beberapa metode seperti berikut:

1. Input Staff, yang akan digunakan untuk memasukkan identitas profil staff baru di perpustakaan.
2. *Delete* Staff, yang akan digunakan untuk menghapus staff yang sudah tidak bekerja di perpustakaan.
3. Edit Staff, yang akan digunakan untuk mengubah informasi staff.

Beberapa atribut yang terdapat dalam kelas peminjaman antara lain sebagai berikut:

1. Id Peminjaman, berisi ID identifikasi proses transaksi peminjaman yang dilakukan oleh anggota.
2. Tanggal Peminjaman, berisi tanggal waktu peminjaman.
3. Id Buku, berisi ID buku yang dipinjam.
4. Id Staff, berisi ID staff yang menginput proses peminjaman.
5. Id Anggota, berisi ID anggota yang melakukan peminjaman.

Berikut adalah metode yang digunakan pada kelas peminjaman:

1. Input Peminjaman, yang digunakan untuk menginput atau memasukkan data-

data yang dibutuhkan dalam proses peminjaman yang dilakukan.

2. *Delete* Peminjaman, yang digunakan untuk menghapus proses peminjaman yang sudah dilakukan.

Beberapa atribut yang terdapat dalam kelas pengembalian antara lain sebagai berikut:

1. Id Pengembalianz, berisi ID identifikasi proses transaksi pengembalian yang dilakukan oleh anggota.
2. Id Peminjaman, berisi ID peminjaman yang dilakukan
3. Tanggal Pengembalian, berisi tanggal waktu pengembalian.
4. Denda, berisi biaya denda yang harus dibayar anggota jika terlambat melakukan pengembalian
5. Id Staff, berisi ID staff yang menginput proses pengembalian.

Berikut adalah metode yang digunakan pada kelas pengembalian:

1. Input Pengembalian, yang digunakan untuk menginput atau memasukkan data-data proses pengembalian buku.
2. *Delete* Peminjaman, yang digunakan untuk menghapus proses pengembalian.

B. Perancangan Database MySQL

Selain rancangan sistem dan kelas-kelas yang terdapat di dalamnya, aplikasi ini juga membutuhkan database untuk menyimpan data-data. Basis data ini akan diakses dengan aplikasi lain yang bernama XAMPP. Dengan adanya XAMPP, basis data dapat dikelola di *localhost* dan diakses melalui *browser* tanpa memerlukan internet.

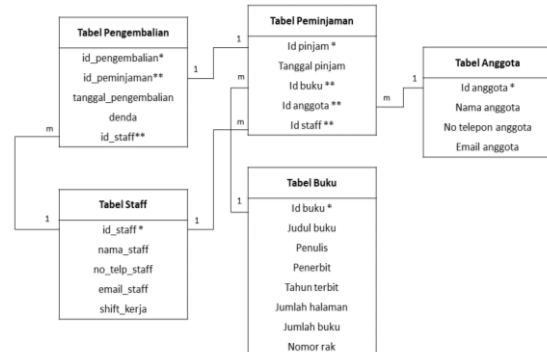
Perancangan *database* ini diawali dengan merancang model data konseptual yang terdiri dari identifikasi entitas, atribut dan relasi yang terjadi di antara entitas-entitas yang ada. Hubungan-hubungan antara ketiga komponen tersebut akan digambarkan dalam suatu *entity relationship diagram (ERD)* seperti yang ditunjukkan pada Gambar 3.

Dalam penerapan basis data relasional, basis data konseptual memiliki beberapa hal yang tidak bisa diimplementasikan. Oleh karena itu, model data logikal yang telah dirancang kemudian dimodifikasi lagi agar bagian-bagian yang tidak kompatibel dalam model data

konseptual bisa dihilangkan. Proses tersebut biasa disebut juga dengan tahapan normalisasi. Model basis data logikal yang telah dinormalisasi kemudian akan diterapkan dalam sistem manajemen basis data *MySQL* ditunjukkan pada Gambar 3.



Gambar 2. *Entity Relationship Diagram* dari Sistem Perpustakaan



Gambar 3. *ERD* Hasil Normalisasi

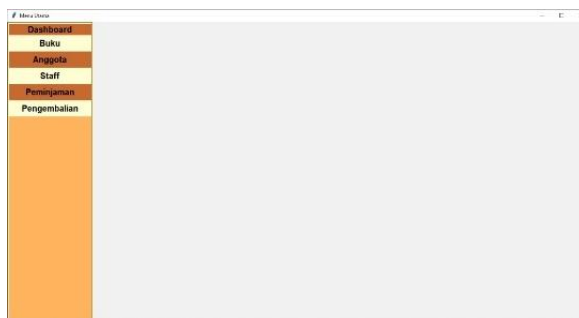
Database yang telah dirancang memiliki lima tabel yang disesuaikan dengan kelas-kelas yang sudah dibuat. Tabel tersebut terdiri dari tabel buku, tabel anggota, tabel staff, tabel peminjaman, dan tabel pengembalian. Tanpa adanya tabel-tabel ini, aplikasi tidak dapat berjalan dengan baik karena tidak adanya media penyimpanan informasi.

C. Perancangan GUI Aplikasi

Aplikasi ini akan dirancang menggunakan bahasa pemrograman *Python* dengan penerapan konsep *OOP (Object Oriented Programming)*.

Perancangan *GUI* ini juga menggunakan *tkinter* yang merupakan suatu *library* atau pustaka yang biasanya sudah dibundel dalam *Python*. *Tk Interface* dapat digunakan untuk membuat berbagai objek *user interface* seperti *frame*, *label*, *button*, dan *entry*.

Tampilan pertama pada aplikasi ini adalah *dashboard* seperti yang ditunjukkan pada Gambar 4. Perancangan *dashboard* utama akan menggunakan dua *frame*. *Frame* ini digunakan untuk menampung komponen-komponen *tkinter* lainnya. *Frame* sebelah kiri akan digunakan untuk menampilkan *side menu* yang terdiri dari *label* untuk *dashboard* serta *button* untuk buku (*book*), anggota (*member*), staff, peminjaman, dan pengembalian. Tampilan *frame* kanan ini akan berubah ketika pengguna mengklik salah satu tombol yang terdapat pada *list menu frame* kiri.



Gambar 4. Dashboard Utama

Ketika pengguna mengklik *button* buku (*book*) yang terdapat pada *side menu*, tampilan *frame* kanan akan berubah dan berpindah ke halaman baru. Pada halaman baru ini akan terdapat tulisan “Dashboard Buku” yang terbuat dari komponen *label* seperti Gambar 5. Kemudian di bawahnya akan terdapat tiga tombol untuk input buku, *delete* buku, dan edit informasi buku.



Gambar 5. Menu Buku

Saat tombol input buku diklik, *frame* kanan akan berpindah ke halaman baru seperti yang ditunjukkan pada Gambar 6. Halaman ini memuat *label* tulisan “Menu Daftar Buku”, delapan *entry* yang akan digunakan untuk menerima input dari *user*, satu tombol untuk kembali ke menu buku serta satu tombol *insert* buku yang ketika diklik akan memasukkan data-data yang telah diisi pada kolom *entry* ke dalam *database*. Rancangan *GUI* ini juga menggunakan *message box* yang akan menampilkan *message* setelah suatu *button* diklik.



Gambar 6. Halaman Input Buku

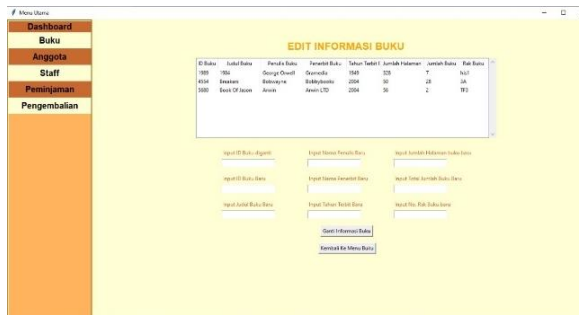
Sedangkan saat tombol *delete* buku diklik, *frame* kanannya akan memuat tampilan halaman yang berbeda. Halaman tersebut akan memuat *label* tulisan dan tabel yang terbuat dari *widget treeview* untuk menampilkan koleksi buku di perpustakaan. Halaman ini juga memuat satu *entry* untuk menerima hasil input pengguna yaitu id buku yang ingin dihapus, satu tombol untuk kembali ke menu buku, dan satu tombol *delete* buku. Rancangan tampilan halaman *delete* ditunjukkan pada Gambar 7.



Gambar 7. Halaman Delete Buku

Tombol terakhir yang terdapat pada *frame* kanan menu buku adalah tombol edit informasi buku. Ketika tombol ini diklik, halaman *frame*

kanan akan memuat satu label tulisan, tabel yang sama seperti tabel pada halaman *delete* buku, *entry* untuk menerima inputan pengguna mengenai data-data yang ingin diubah, satu tombol untuk kembali ke menu buku serta satu tombol untuk mengganti informasi buku yang terdapat pada bagian bawahnya. Tampilan halaman *Edit* ditunjukkan pada Gambar 8.

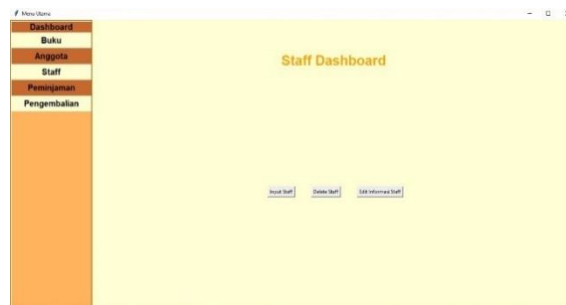


Gambar 8. Halaman *Edit* Buku

Halaman menu anggota (*member*) (Gambar 9) dan staff (Gambar 10) juga terbentuk dengan komponen-komponen yang sama dengan komponen pada halaman buku. Hal yang membedakannya adalah data-data atribut yang perlu diinput dan data tabel yang digunakan.

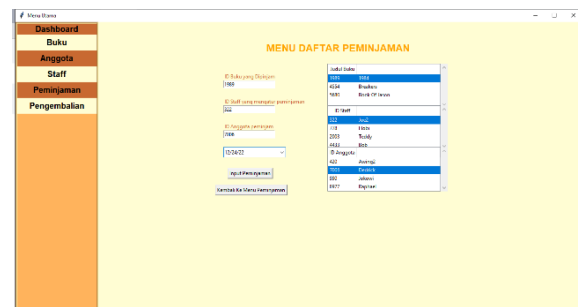


Gambar 9. Menu Anggota

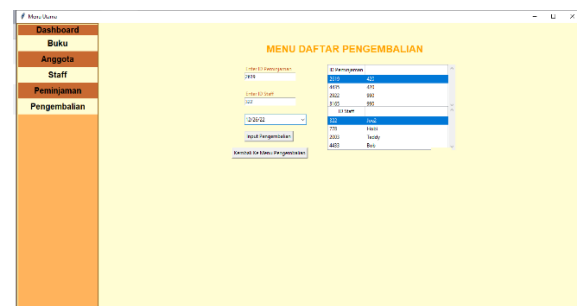


Gambar 10. Menu Staff

Sedangkan komponen pada halaman menu peminjaman pada Gambar 11 memiliki kemiripan dengan komponen menu pengembalian pada Gambar 12. Kedua halaman tersebut sama-sama memuat satu label dan dua tombol yaitu, tombol input dan *delete*. Ketika tombol input pada halaman peminjaman diklik, halaman yang muncul akan memuat beberapa *entry* untuk memasukkan id buku, id staff, id anggota dan juga tanggal peminjaman. Pengguna dapat melihat data-data *list* id tersebut pada tabel yang ditampilkan. Halaman input pengembalian juga memiliki komponen yang sama, hanya saja data-data yang perlu diinput adalah id pengembalian, id staff, dan tanggal pengembalian. Untuk halaman *delete* peminjaman dan pengembalian, komponen yang dimiliki sama dengan komponen yang ada pada halaman *delete* buku.



Gambar 11. Halaman *Input* Peminjaman



Gambar 12. Halaman *Input* Pengembalian

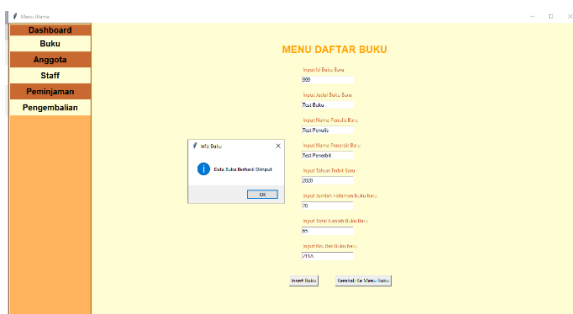
D. Penerapan *Inheritance*

Sesuai dengan pengertiannya pada tinjauan pustaka, *inheritance* adalah suatu kelas yang beberapa *method* atau fungsinya dipinjam dan dipakai oleh kelas lain. Dalam aplikasi yang dirancang, penerapan *inheritance* dilakukan pada lima kelas utama yaitu kelas buku, anggota, staff, peminjaman dan pengembalian. Kelima kelas tersebut akan meminjam fungsi kelas

frame yang isinya berupa satu frame atau kerangka kosong yang ditempatkan pada GUI. Kelas yang diwariskan ini tidak memiliki fungsi lainnya dan hanya digunakan sebagai kerangka penempatan tombol dan label untuk lima kelas utama.

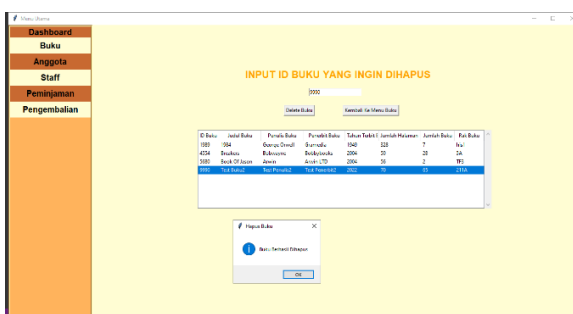
IV. HASIL DAN PEMBAHASAN

Proses pengujian aplikasi dilakukan dengan menjalankan hasil coding pada aplikasi Visual Studio Code. Setelah kode program dijalankan, tampilan GUI dashboard yang telah dirancang akan muncul. Pengujian pertama akan dilakukan dengan menginput data-data buku pada kolom entry halaman input buku. Setelah tombol input diklik, bisa dilihat data-data yang dimasukkan berhasil tersimpan di dalam database. Hasil pengujian input buku ditampilkan pada Gambar 13.



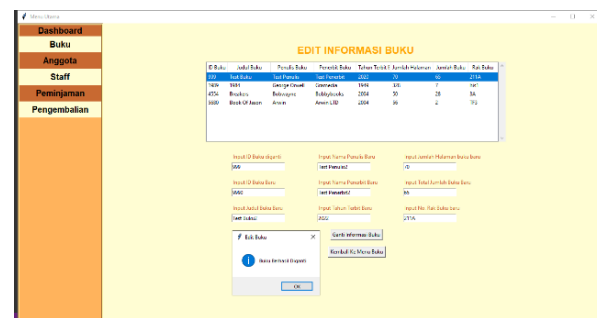
Gambar 13. Proses Pengujian Menginput Buku

Selanjutnya adalah pengujian penghapusan data. Pengujian ini akan dilakukan pada halaman delete buku, dimana penguji akan memasukkan id buku yang ingin dihapus pada kolom entry yang tersedia. Setelah button delete diklik, dapat dilihat bahwa buku dengan id yang dimasukkan sebelumnya berhasil terhapus di database. Hasil pengujian penghapusan buku ditunjukkan pada Gambar 14.



Gambar 14. Proses Pengujian Menghapus Buku

Proses pengujian yang selanjutnya adalah mengedit data. Proses ini dilakukan dengan melakukan perubahan data pada halaman edit informasi buku. Penguji akan memasukkan id buku yang ingin diganti serta data-data baru yang ingin diubah pada kolom entry. Setelah tombol delete diklik, data buku yang dimasukkan awalnya akan mengalami perubahan menjadi data baru sesuai dengan proses pengeditan yang dilakukan oleh penguji. Hasil pengujian edit buku ditampilkan pada Gambar 15.



Gambar 15. Proses Pengujian Mengedit Buku

V. KESIMPULAN

Rancangan aplikasi sistem manajemen perpustakaan berhasil untuk dijalankan. Hal ini dibuktikan dengan proses pengujian untuk menginput, menghapus dan mengedit data-data melalui aplikasi. Data-data yang diinput berhasil tersimpan dalam database, data yang dihapus akan dihilangkan dari database, dan data yang diedit mengalami perubahan. Perubahan data-data tersebut dapat dilihat melalui tabel basis data yang diakses dengan aplikasi XAMPP.

Aplikasi yang telah dirancang ini diharapkan dapat membantu petugas perpustakaan untuk melakukan pekerjaan mereka melalui peraturan sederhana yang dijalankan dalam aplikasi dan cara akses yang mudah dari komputer perpustakaan.

VI. DAFTAR PUSTAKA

[1] Badan Pusat Statistik. (2022). Statistical Yearbook of Indonesia

[2] IFLA. (2022). Library Map Of The World <https://librarymap.ifla.org/>

[3] Raut, Rushikesh S., (2020). Research Paper on Object-Oriented Programming (OOP), International Research Journal of Engineering and Technology (IRJET)

- [4] Van Rossum, G. (2003). An introduction to Python (p. 115). F. L. Drake (Ed.). Bristol: Network Theory Ltd..
- [5] Raj, A., Meghana, P. D. S., Rodrigues, R., & D'Souza, P. (2020). IMPLEMENTATION OF MYSQL IN PYTHON. IJRAR-International Journal of Research and Analytical Reviews (IJRAR), 7(1), 447-451.
- [6] M. Orrú, E. Tempero, M. Marchesi and R. Tonelli. (2015). "How Do Python Programs Use Inheritance? A Replication Study," Asia-Pacific Software Engineering Conference (APSEC), 2015, pp. 309-315, doi: 10.1109/APSEC.2015.51.
- [7] Walia, E. S., & Gill, E. S. K. (2014). A framework for web based student record management system using PHP. International Journal of Computer Science and Mobile Computing, 3(8), 24-33.
- [8] Connell, J. (2003). "What is a database?," Beginning Visual Basic 6 Database Programming, pp. 9–38.
- [9] Powell, G. (2007). "What is SQL?," Oracle Performance Tuning for 10gR2, pp. 91–114.
- [10] Jansen, B. J. (1998). The graphical user interface. ACM SIGCHI Bulletin, 30(2), 22-26.
- [11] Del Sole, A. (2021). Introducing Visual Studio Code. In: Visual Studio Code Distilled. Apress, Berkeley, CA.
- [12] Lacey, N. (2019). "Tkinter GUI" Python by Example: Learning to Program in 150 Challenges. Cambridge University Press, pp110-123.