

Laporan Tugas Besar

Pemrograman Berorientasi Objek

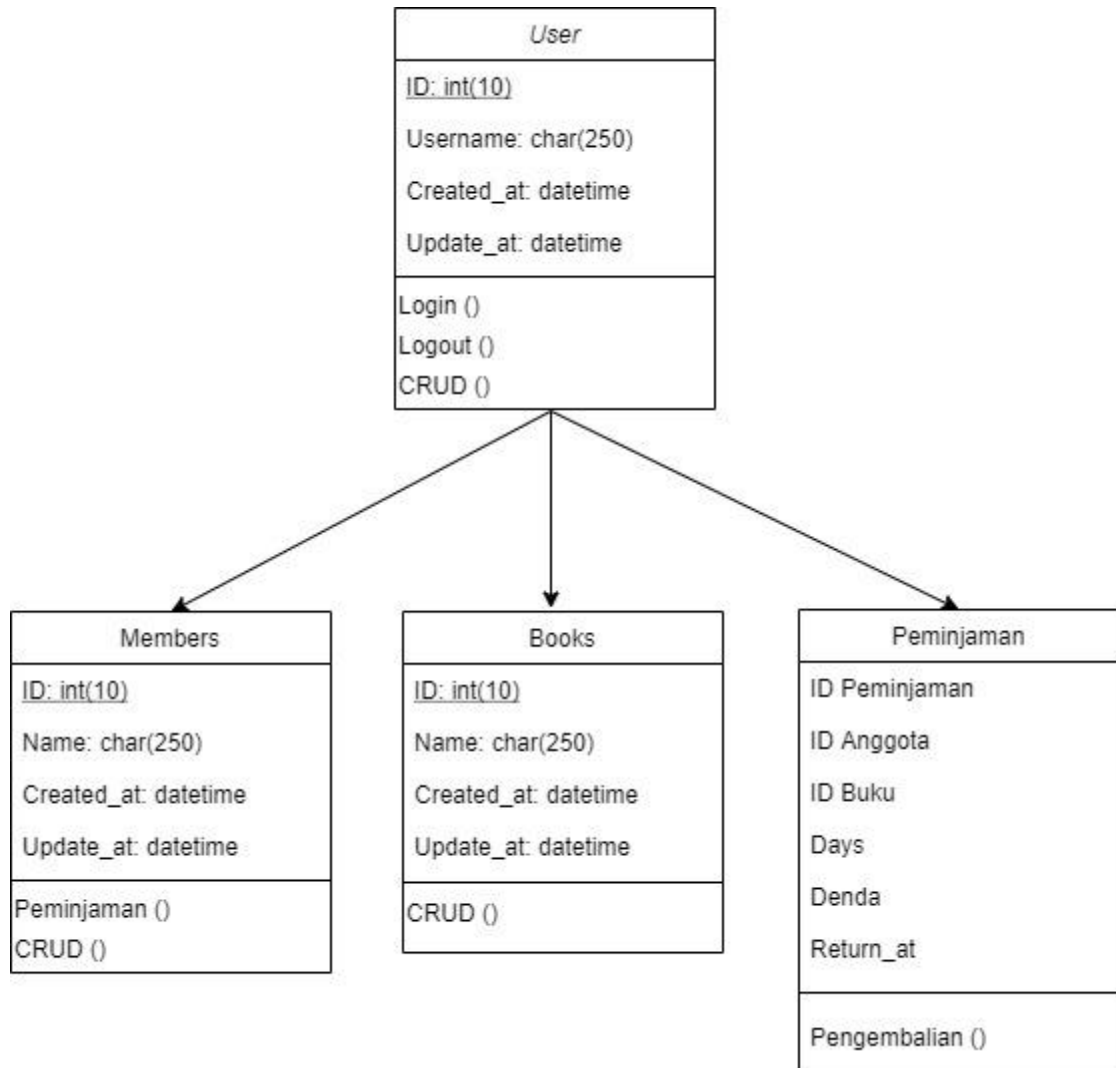
Disusun Oleh :

Dodi Devrian Andrianto	119140023
Ayu Feblira Gersy	119140102
Cahya Putri Antika	119140084



Program Studi Teknik Informatika
Jurusan Teknik Elektro, Informatika, dan Sistem Fisis
Institut Teknologi Sumatera
Lampung Selatan
2021

Diagram Kelas System Management Library



Dari diagram diatas dapat dilihat bahwa pertama kali yang harus dilakukan adalah menginputkan username dan password seperti yang tertera pada tabel user, user yang dimaksud adalah admin perpustakaan. Setelah itu admin dapat memilih untuk masuk kedalam tabel book atau peminjaman dimana admin dapat mengubah data yang terdapat didalam tabel-tabel tersebut.

Penjelasan Methode

Book

```
19 @dispatch()
20 def getAll(self):
21     query = "SELECT * FROM books"
22     if self.execQ(query):
23         return self._cnx.getConnection().fetchall()
24     else:
25         print("Failed to GET ALL data Buku")
26
27 def delete(self, id):
28     query = "DELETE FROM books where id = %s"
29     if self.execQ(query, (id,)):
30         print("===== Buku berhasil dihapus =====")
31     else:
32         print("===== FAILED TO DELETE Buku =====")
33
34 @dispatch(str)
35 def getAll(self, searchValue):
36     query = "SELECT * FROM books WHERE name LIKE %s"
37     if self.execQ(query, (searchValue,)):
38         return self._cnx.getConnection().fetchall()
39     else:
40         print("===== Failed to get data Buku =====")
```

Member

```
19 @dispatch()
20 def getAll(self):
21     query = "SELECT * FROM members"
22     if self.execQ(query):
23         return self._cnx.getConnection().fetchall()
24     else:
25         print("Failed to GET ALL data MEMBER")
26
27 def delete(self, id):
28     query = "DELETE FROM members where id = %s"
29     if self.execQ(query, (id,)):
30         print("===== MEMBER berhasil dihapus =====")
31     else:
32         print("===== FAILED TO DELETE MEMBER =====")
33
34 @dispatch(str)
35 def getAll(self, searchValue):
36     query = "SELECT * FROM members WHERE name LIKE %s"
37     if self.execQ(query, (searchValue,)):
38         return self._cnx.getConnection().fetchall()
39     else:
40         print("===== Failed to get data MEMBER =====")
```

Model

```
13     # overriding
14     @dispatch(str)
15     def execQ(self, query):
16         try:
17             self._cnx.getConnection().execute(query)
18             return True
19         except:
20             return False
21
22     @dispatch(str, tuple)
23     def execQ(self, query, payload):
24         try:
25             self._cnx.getConnection().execute(query, payload)
26             self._cnx.getDB().commit()
27             return True
28         except:
29             return False
```

Peminjaman

```
19     @dispatch()
20     def getAll(self):
21         query = "SELECT * FROM peminjaman where returned_at is null"
22         if self.execQ(query):
23             return self._cnx.getConnection().fetchall()
24         else:
25             print("Failed to GET ALL data PEMINJAMAN")
26
27     def delete(self, id):
28         query = "DELETE FROM peminjaman where id = %s"
29         if self.execQ(query, (id,)):
30             print("===== PEMINJAMAN berhasil dihapus =====")
31         else:
32             print("===== FAILED TO DELETE PEMINJAMAN =====")
33
34     @dispatch(str)
35     def getAll(self, searchValue):
36         query = "SELECT * FROM peminjaman WHERE name LIKE %s"
37         if self.execQ(query, (searchValue,)):
38             return self._cnx.getConnection().fetchall()
39         else:
40             print("===== Failed to get data PEMINJAMAN =====")
```

User

```
19 @dispatch()
20 def getAll(self):
21     query = "SELECT * FROM users"
22     if self.execQ(query):
23         return self._cnx.getConnection().fetchall()
24     else:
25         print("Failed to GET ALL data admin")
26
27 def delete(self, id):
28     query = "DELETE FROM users where id = %s"
29     if self.execQ(query, (id,)):
30         print("===== ADMIN berhasil dihapus =====")
31     else:
32         print("===== FAILED TO DELETE ADMIN =====")
33
34 @dispatch(str)
35 def getAll(self, searchValue):
36     query = "SELECT * FROM users WHERE username LIKE %s"
37     if self.execQ(query, (searchValue,)):
38         return self._cnx.getConnection().fetchall()
39     else:
40         print("===== Failed to get data admin =====")
```

```
42 # overriding
43 @dispatch(str)
44 def get(self, id):
45     query = "SELECT * FROM users where id=%s"
46     if self.execQ(query, (id,)):
47         return self._cnx.getConnection().fetch()
48     else:
49         return False
50
51 @dispatch(str, str)
52 def get(self, username, password):
53     query="SELECT * FROM users where username=%s and password=%s"
54     if self.execQ(query, (username, password,)):
55         data = self._cnx.getConnection().fetchall()
56         if len(data) < 1:
57             return False
58         return data[0]
59     else:
60         return False
```

Panduan Pengguna

System management library dibuat untuk mempermudah pustakwan yang berperan sebagai admin dalam system yang telah kami rancang, untuk melakukan management perpustakaan dalam lingkup menambahkan, menghapus, serta mengedit admin apabila terjadi penambahan, pengurangan, serta kesalahan data admin. Selain admin, terdapat list anggota yang berfungsi sama terdapat fungsi tambah, hapus, dan edit anggota. Kemudian list buku, list peminjaman, serta pengembalian.

Terdapat Langkah – Langkah yang dapat dilakukan oleh pengguna :

1. Diminta untuk login akun dengan memasukkan username admin dan password admin
2. Tampil 6 pilihan anda diminta untuk dapat memilih, admin, anggota, buku, pinjaman, pengembalian, dan logout
3. Jika memilih admin maka akan muncul siapa saja admin yang mengelola system management perpustakaan.
4. Di dalam list admin anda dapat memilih menambahkan admin, menghapus admin, mengedit admin, mencari data admin.
5. Jika memilih anggota akan muncul siapa saja anggota dalam system management perpustakaan.
6. Di dalam list anggota anda dapat memilih menambahkan anggota, menghapus anggota, mengedit anggota, mencari data anggota.
7. Jika memilih buku akan muncul nama buku dan id buku yang terdapat dalam system management perpustakaan
8. Di dalam list buku anda dapat memilih menambahkan nama buku, menghapus nama buku, mengedit nama buku, mencari nama buku.
9. Jika memilih pinjaman anda diminta untuk input id buku, id anggota, jangka waktu peminjaman, biaya yang dikeluarkan jika melebihi jangka waktu.
10. Jika memilih pengembalian maka akan muncul data buku yang telah dipinjam, sebagai admin diminta untuk menginputkan id peminjaman. Jika telah menginputkan id peminjaman maka akan secara otomatis menampilkan denda apabila sudah melebihi batas peminjaman buku. Dan akan muncul tidak ada denda jika pengembalian tidak melebihi batas peminjaman.
11. Jika logout berarti anda telah memilih untuk keluar dari system.

Biodata Anggota dan Lingkup Kerja

Nama : Dodi Devrian Andrianto
NIM : 119140023
TTL : 25 Desember 2000
Alamat : Yukum Jaya, Lampung Tengah
Prodi : Teknik Informatika
Lingkup Kerja : Membuat alur kerja aplikasi, membuat video demo aplikasi

Nama : Ayu Feblira Gersy
NIM : 119140102
TTL : Bandar Lampung, 01 Februari 2001
Alamat : Jalan Imam Bonjol Gg.Ratu Agung II no.53/40
Prodi : Teknik Informatika
Lingkup kerja : Membuat frontend, membuat diagram kelas, membuat laporan

Nama : Cahya Putri Antika
NIM : 119140084
TTL : Ambarawa, 19 Oktober 2000
Alamat : Jl H.M Ghardi Ambarawa Barat
Ambarawa Kab.Pringsewu
Prodi : Teknik Informatika
Lingkup Kerja : Membuat sql connection, membuat data dalam file business serta membuat laporan