**LAPORAN PENGEMBANGAN APLIKASI PEMINJAMAN BUKU PERPUSTAKAAN ONLINE**

Di susun untuk memenuhi tugas

Mata Kuliah: Pemrograman Basis Data

Dosen Pengampu: Ahmad Wilda Y, ST, MT



Oleh:

1. Ilham Jody Bimantara (1741160044)
2. M. Khusuma Dwi N (1741160091)
3. Monalisa Desideria M (1741160011)
4. Yogie Tri Priyo S (1741160040)

**D-IV JARINGAN TELEKOMUNIKASI DIGITAL**

**TEKNIK ELEKTRO**

**POLITEKNIK NEGERI MALANG**

**2020**

**PENGEMBANGAN APLIKASI PEMINJAMAN BUKU PERPUSTAKAAN ONLINE**

Ilham Jody Bimantara1, M. Khusuma Dwi N2, Monalisa Desideria M3, Yogie Tri Priyo S4

Program Studi Jaringan Telekomunikasi Digital

Jurusan Teknik Elektro

Politeknik Negeri Malang

Ilhamjb23@gmail.com1, monalisadesideria12@gmail.com3

***Abstrak*** - **Perancangan basis data merupakan proses membuat desain yang akan mendukung operasional dan tujuan suatu perusahaan. Pemanfaatan basis data pada bidang perpustakaan memungkinkan untuk dapat menyimpan data atau melakukan perubahan dan menampilkan kembali data tersebut dengan cepat dan mudah. Metode**

**penelitian yang digunakan adalah metode Research and Development (R&D). Metode perancangan basis data menggunakan metode DBLC (Data Base Life Cycle) dengan variabel penelitian adalah perancangan basis data perputakaan umum dengan model data relasional. Aspek penelitian meliputi Conceptual Database Design, Logical Database Design dan Physical Database Design. Hasil akhir dari penelitian ini adalah menghasil 9 (sembilan) tipe entitas konsepsual, menghasilkan diagram hubungan entitas dari kesembilan entitas tersebut pada logikal dan menghasilkan rancangan pisikal yang terdiri dari tabel kategori, penerbit, penulis, buku, anggota, detail transaksi, master transaksi, pegawai, rak.**

***Kata Kunci: Basis Data, Data Relasional, DBLC dan Perpustakaan.***

1. **Pengantar**

Penggunaan teknologi informasi yang tepat dapat membantu dalam mengambil keputusan dimana penyimpanan data telah didukung oleh salah satu komponen penting dari teknologi informasi yaitu database (basis data). Basis data didefinisikan sebagai kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dimanipulasi, diambil, dan dicari secara cepat. Perancangan basis data (*database design*) merupakan proses membuat desain yang akan mendukung operasional dan tujuan perusahaan. Sistem database merupakan komponen dasar sistem informasi dari perusahaan besar, sistem pengembangan siklus database secara melekat terkait dengan siklus hidup sistem informasi. Salah satu aspek yang sulit dalam perancangan database adalah kenyataan bahwa perancang, *programmer* dan pemakai akhir cenderung melihat data dengan cara yang berbeda.

Poses perancangan basis data terdiri dari tiga bagian yaitu perancangan basis data konseptual, logikal dan fisikal. Perancangan basis data konseptual adalah proses membangun model data yang digunakan dalam suatu perusahaan, serta terbebas dari semua pertimbangan fisik. Perancangan basis data logikal adalah proses merancang model data yang digunakan dalam suatu perusahaan berdasarkan pada model data yang spesifik, tetapi terbebas dari DBMS (*Database Management System)* tertentu dan pertimbangan fisik lainnya. Perancangan basis data fisikal adalah proses menghasilkan deskripsi implementasi basis data pada penyimpanan sekunder, menggambarkan hubungan dasar, organisasi file, dan indeks yang digunakan untuk mencapai akses yang efisien terhadap data, dan setiap kendala integritas terkait dan langkah-langkah keamanan. *Relational Database* adalah mempresentasikan semua data dalam database sebagai tabel dua dimensi sederhana yang di sebut relasi. Tabel-tabel itu serupa dengan file biasa, namun informasi dalam lebih dari satu file bisa dengan mudah di ekstrak dan di kombinasikan. Kadang kala tabel-tabel ini bisa di anggap sebagai file.

1. **Latar Belakang**

Perpustakaan adalah tempat yang melayani kebutuhan manusia dalam memperoleh informasi berbagai macam jenis buku maupun informasi lainnya yang terkait dengan peminjaman buku. Dalam proses operasionalnya, seperti pengadaan buku, pendataan buku, peminjaman buku maupun pengembalian buku dilakukan secara konvensional tidaklah tepat ketika pimpinan sudah menyadari bahwa penerapan teknologi informasi mampu memberikan efisiensi dan efektivitas dari aktivitas pengelolaan perpustakaan. Mengabaikan teknologi informasi dalam pengelolaan perpustakaan juga akan berdampak kurang baiknya pelayanan kepada setiap anggota maupun petugas perpustakaan akan mengalami kesulitan dalam menghasilkan berbagai laporan kepada pimpinan.

Perancangan basis data perpustakaan akan memberikan gambaran sebuah basis data yang disusun dan dirancang sesuai dengan konsep DBLC (*Data Base Life Cycle*). Dimana dalam proses pembuatan database perpustakaan akan dilakukan sesuai dengan tahap atau fase dari DBLC. Fokus pembahasan dalam penelitian adalah mengarah kepada perancangan basis data relasional yang meliputi *Conceptual Database Design, Logical Database Design dan Physical Database Design*. Hasil dari rancangan basis data model relasional khususnya basis data perpustakaan dimaksudkan untuk menjaga integritas data dari setiap tabel yang berrelasi. Keberhasilan dalam membangun sebuah aplikasi tidak terlepas dari sebuah model basis data yang baik. Inilah yang menjadi dasar utama diperlukan sebuah rancangan basis data dengan merapkan mode relasional.

1. **Metode Penelitian**

Metode penelitian yang digunakan adalah metode penelitian dan pengembangan atau yang lebih dikenal dengan *Research and Development* (R&D). Metode perancangan basis data menggunakan metode DBLC (*Data Base Life Cycle*) dengan variabel penelitian adalah perancangan basis data perpustakaan sekolah dengan model data relasional. Aspek penelitian meliputi perancangan basis data konseptual, perancangan basis data logikal dan perancangan basis data fisikal.

1. **Hasil dan Pembahasan**

Perancangan merupakan suatu hal yang sangat penting dalam pembuatan basis data. Permasalahan yang dihadapi pada waktu perancangan adalah bagaimana basis data yang akan dibangun ini dapat memenuhi kebutuhan saat ini dan masa yang akan datang. Untuk itu diperlukan perancangan basis data baik secara fisik maupun secara konseptualnya. Perancangan konseptual akan menunjukkan entity dan relasinya berdasarkan proses yang diinginkan oleh organsisasinya. Untuk menentukan entity dan relasinya perlu dilakukan analisis data tentang informasi yang ada dalam spesifikasi di masa yang akan datang. Metodologi perancangan basis data adalah kumpulan teknik terorganisasi untuk pembuatan rancangan basis data. Teknik terorganisasi ini merupakan kumpulan tahap-tahapan yang memiliki aturan-aturan terurut. Teknik yang digunakan pada perancangan basis data dibagi dalam tiga tahap, yaitu perancangan basis data konseptual (*conseptual database design*), perancangan basis data logikal (*logical database design*) dan perancangan basis data fisikal (*physical database design*).

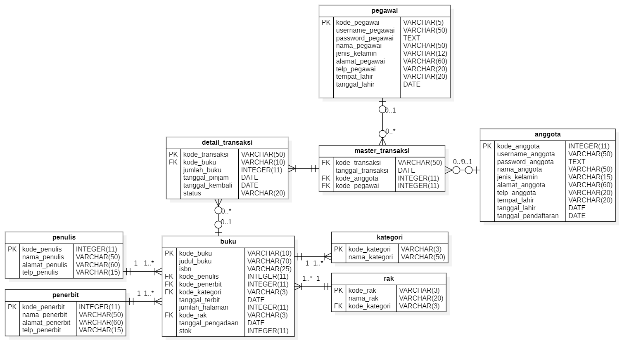
*4.1 Conseptual Database Design*

*Conceptual database design* adalah proses membangun model data yang digunakan di dalam suatu perusahaan, bersifat independent dari semua pertimbangan fisikal. Tahap desain konseptual database yang dimulai dengan membuat model data konseptual dari perusahaan dengan rincian implementasi seperti target DBMS, program aplikasi, bahasa pemrograman, hardware platform, performance dan segala pertimbangan fisikal lain nya (tabel 1 Identifikasi Tipe Entitas).

Tabel 1. Identifikasi Tipe Entitas

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Entity** | **Keterangan Entity** | **Kegiatan** |
| 1 | Kategori Buku | Merupakan Entitas yang berisi informasi mengenai data kategori dari suatu buku. | Pengelompokan data buku berdasarkan kategori dan satu kategori buku bisa terdiri dari beberapa buku. |
| 2 | Penerbit Buku | Merupakan Entitas yang berisi informasi mengenai data penerbit buku. | Pengelompokan data buku berdasarkan penerbit dan satu penerbit buku bisa terdiri dari beberapa buku. |
| 3 | Penulis Buku | Merupakan Entitas yang berisi informasi mengenai data penulis buku. | Pengelompokan data buku berdasarkan penulis dan satu penulis buku bisa terdiri dari beberapa buku. |
| 4 | Buku | Merupakan Entitas yang berisi informasi mengenai data buku. | Setiap buku dapat dipinjamkan kepada satu atau beberapa anggota |
| 5 | Anggota | Merupakan Entitas yang berisi informasi mengenai data anggota. | Anggota dapat melakukan beberapa kali peminjaman buku |
| 6 | Master Transaksi | Merupakan Entitas yang berisi informasi mengenai data transaksi peminjaman dan pengembalian. | Dapat menyimpan hanya satu jenis data transaksi. |
| 7 | Detil Transaksi | Merupakan Entitas yang berisi informasi mengenai data transaksi peminjaman dan pengembalian secara detil. | Satu transaksi bisa terdiri dari satu buku atau beberapa buku. |
| 8. | Pegawai | Merupakan entitas yang berisi informasi mengenai data pegawai | Pegawai dapat melakukan penambahan, pengubahan, dan penghapusan data pada database |
| 9. | rak | Merupakan Entitas yang berisi informasi mengenai data rak. | Rak menunjukkan lokasi buku |

Identifikasi tipe rasional bertujuan menentukan hubungan-hubungan penting yang ada antara jenis-jenis entitas yang telah diidentifikasikan sebelumnya (gambar 1. E-R Diagram Konseptual).



Gambar 1. E-R Diagram Konseptual

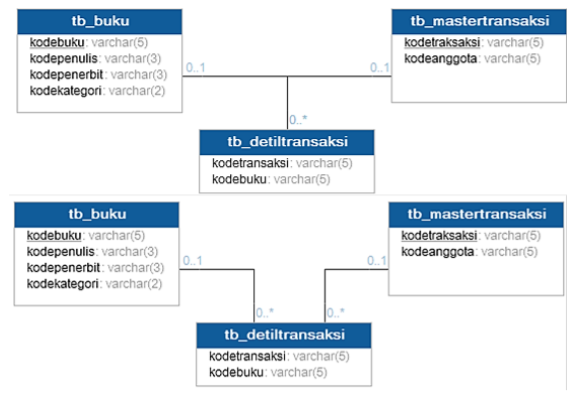
Domain adalah seluruh kemungkinan nilai yang dapat diberikan kesuatu atribut. Memberi nama domain yang sesuai dengan nilai yang akan dimiliki domain tersebut. Domain menentukan tipe data dari nilai yang akan membentuk domain dan menentukan format dari domain (tabel 2 Tabel Attribute Domain).

Tabel 2. Tabel Attribute Domain

|  |  |  |
| --- | --- | --- |
| **Entity Name** | **Attribute** | **Domain** |
| Kategori  Penerbit | kode\_kategori | String dengan panjang maksimal 3 karakter {alk,dra,fik,…} |
| nama\_kategori | String dengan panjang maksimal 50 karakter |
| Penerbit | kode\_penerbit | Int dengan panjang maksimal 11 karakter {1,2,3,…} |
| nama\_penerbit | String dengan panjang maksimal 50 karakter |
| alamat\_penerbit | String dengan panjang maksimal 60 karakter |
| telp\_penerbit | String dengan panjang maksimal 15 karakter |
| Penulis | kode\_penulis | int dengan panjang maksimal 11 karakter {1,2,3,…} |
| nama\_penulis | String dengan panjang maksimal 50 karakter |
| alamat\_penulis | String dengan panjang maksimal 60 karakter |
| telp\_penulis | String dengan panjang maksimal 15 karakter |
| Buku | kode\_buku | String dengan panjang maksimal 10 karakter  {Bk\_1, Bk\_2, Bk\_3,…} |
| kode\_penerbit | Int dengan panjang maksimal 11 karakter {1,2,3,…} |
| kode\_penulis | Int dengan panjang maksimal 11 karakter {1,2,3,…} |
| judul\_buku | String dengan panjang maksimal 70 karakter |
| Isbn | String dengan panjang maksimal 25 karakter |
| tanggal\_terbit | Tanggal dengan panjang maksimal 10 karakter {yyyy-mm-dd} |
| jumlah\_halaman | Int dengan panjang maksimal 11 karakter {1,2,3,…} |
| kode\_kategori | String dengan panjang maksimal 3 karakter {alk,dra,fik,…} |
| kode\_rak | String dengan panjang maksimal 3 karakter |
| stok | Int dengan panjang maksimal 11 karakter {1,2,3,…} |
| Tanggal\_pengadaan | Tanggal dengan panjang maksimal 10 karakter {yyyy-mm-dd} |
| Anggota | kode\_anggota | Int dengan panjang maksimal 11 karakter {1,2,3,…} |
| nama\_anggota | String dengan panjang maksimal 50 karakter |
| jenis\_kelamin | String dengan panjang maksimal 15 karakter |
| alamat\_anggota | String dengan panjang maksimal 60 karakter |
| telp\_anggota | String dengan panjang maksimal 20 karakter |
| tempat\_lahir | String dengan panjang maksimal 20 karakter |
| tanggal\_lahir | Tanggal dengan panjang maksimal 10 karakter {yyyy-mm-dd} |
| usarname\_anggota | String dengan panjang maksimal 50 karakter |
| password\_anggota | Text |
| tanggal\_pendaftaran | Tanggal dengan panjang maksimal 10 karakter {yyyy-mm-dd} |
| Master Transaksi | kode\_traksaksi | String dengan panjang maksimal 50 karakter |
| tanggal\_transaksi | Tanggal dengan panjang maksimal 10 karakter {yyyy-mm-dd} |
| kode\_anggota | Int dengan panjang maksimal 11 karakter {1,2,3,…} |
| kode\_pegawai | Int dengan panjang maksimal 11 karakter {1,2,3,…} |
| Detil Transaksi | kode\_traksaksi | String dengan panjang maksimal 50 karakter |
| kode\_buku | String dengan panjang maksimal 10 karakter  {Bk\_1, Bk\_2, Bk\_3,…} |
| tanggal\_pinjam | Tanggal dengan panjang maksimal 10 karakter {yyyy-mm-dd} |
| tanggal\_kembali | Tanggal dengan panjang maksimal 10 karakter {yyyy-mm-dd} |
| jumlah\_buku | Int dengan panjang maksimal 11 karakter {1,2,3,…} |
| status | String dengan panjang maksimal 20 karakter |
| Pegawai | kode\_pegawai | Int dengan panjang maksimal 11 karakter {1,2,3,…} |
| username\_pegawai | String dengan panjang maksimal 50 karakter |
| password\_pegawai | Text |
| nama\_pegawai | String dengan panjang maksimal 50 karakter |
| jenis\_kelamin | String dengan panjang maksimal 12 karakter |
| alamat\_pegawai | String dengan panjang maksimal 60 karakter |
| telp\_pegawai | String dengan panjang maksimal 20 karakter |
| tempat\_lahir | String dengan panjang maksimal 20 karakter |
| tanggal\_lahir | Tanggal dengan panjang maksimal 10 karakter {yyyy-mm-dd} |
| Rak | kode\_rak | String dengan panjang maksimal 3 karakter  {A1, B2, C3,…} |
| nama\_rak | String dengan panjang maksimal 20 karakter |
| kode\_kategori | String dengan panjang maksimal 3 karakter {alk,dra,fik,…} |

*4.2 Logical Database Design*

Perancangan basis data logikal adalah suatu proses membangun sebuah model dari informasi yang digunakan di perusahaan berdasarkan sebuah model data spesifik. Tujuan dari langkah ini adalah untuk membangun data model logikal lokal dari data model konseptual dengan menggambarkan pandangan khusus dari perusahaan dan kemudian untuk memvalidasikan model ini untuk memastikan bahwa model tersebut benar dan untuk memastikan bahwa model tersebut mendukung transaksi yang diperlukan. Dalam sistem basis data relasional yang akan digunakan, ada hal-hal dalam perancangan basis data konseptual yang tidak bisa diimplementasikan oleh sebab itu, dalam rancangan database relasional perlu diadakan modifikasi, yaitu menghilangkan bagian yang tidak kompatibel dari model data konseptual. Langkah-langkanya antara lain menghilangkan relasi biner *many-to-many*, relasi rekursif *many-to-many*, relasi kompleks dan atribut *multivalued*. Untuk menghilangkan tipe hubungan yang mengandung *many-to-many* (\*.\*). Hubungan tersebut dipecah dengan mengidentifikasi sebuah entitas baru dan mengganti hubungannya dengan *one-to-many* (1.\*) sehingga menghilangkan hubungan *many-to-many*.



Gambar 2. Hubungan Buku Dengan Transaksi

Validasi relasi-relasi menggunakan normalisasi untuk meminimalkan kemungkinan terjadinya data rangkap, menghindari data yang tidak konsisten terutama bila dilakukan penambahan atau penghapusan data sebagai akibat karena adanya data yang rangkap, dan untuk menjamin bahwa identitas tabel secara tunggal sebagai determinan semua atribut. Berikut ini adalah normalisasi basis data perpustakaan:

1. Bentuk unnormal

{Kode\_buku, judul\_buku, isbn, tanggal\_terbit, jumlah\_halaman, nama\_kategori, nama\_rak, stok, nama\_penerbit, alamat\_penerbit, telp\_penerbit, nama\_penulis, alamat\_penulis, telp\_penulis, nama\_anggota, jenis\_kelamin, alamat\_anggota, telp\_anggota, tempat\_lahir, tanggal\_lahir, nama\_pegawai, jenis\_kelamin, alamat\_pegawai, telp\_pegawai, tempat\_lahir, tanggal\_lahir, tgl\_transaksi, tgl\_pinjam, tgl\_kembali, jumlah\_buku, status}

2. Bentuk Normal Pertama,

Langkah berikutnya adalah dengan cara memisahkan atribut-atribut yang nilainya sama akan ditulis hanya satu kali.

Tabel buku {\*Kode\_buku, judul\_buku, isbn, tanggal\_terbit, jumlah\_halaman, nama\_kategori, nama\_rak, stok, nama\_penerbit, alamat\_penerbit, telp\_penerbit, nama\_penulis, alamat\_penulis, telp\_penulis, nama\_anggota, jenis\_kelamin, alamat\_anggota, telp\_anggota, tempat\_lahir, tanggal\_lahir, nama\_pegawai, jenis\_kelamin, alamat\_pegawai, telp\_pegawai, tempat\_lahir, tanggal\_lahir}

Tabel transaksi {tgltransaksi, tglpinjam, tglkembali, jumlahbuku, status}

3. Bentuk Normal Kedua

Langkah selanjutnya adalah dengan cara menentukan ketergantungan fungsional.

Tabel buku {\*Kode\_buku, judul\_buku, isbn, tanggal\_terbit, jumlah\_halaman, \*\*kode\_kategori, \*\*kode\_penerbit, \*\*kode\_penulis,\*\*kode\_rak, stok, tanggal\_pengadaan}

Tabel kategori {\*kode\_kategori, nama\_kategori}

Tabel rak {\*kode\_rak, nama\_rak, \*\*kode\_kategori}

Tabel penerbit {\*kode\_penerbit, nama\_penerbit, alamat\_penerbit, telp\_penerbit}

Tabel penulis {\*kode\_penulis, nama\_penulis, alamat\_penulis, telp\_penulis}

Tabel anggota {\*kode\_anggota, nama\_anggota, jenis\_kelamin, alamat\_anggota, telp\_anggota, tempat\_lahir, tanggal\_lahir, username\_anggota, password\_anggota, tanggal\_pendaftaran}

alamat\_penulis, telp\_penulis}

Tabel pegawai {\*kode\_pegawai, nama\_pegawai, jenis\_kelamin, alamat\_pegawai, telp\_pegawai, tempat\_lahir, tanggal\_lahir, username\_pegawai, password\_pegawai}

alamat\_penulis, telp\_penulis}

Tabel anggota {\*kode\_anggota, nama\_anggota, jenis\_kelamin, alamat\_anggota, telp\_anggota, tempat\_lahir, tanggal\_lahir, username\_anggota, password\_anggota, tanggal\_pendaftaran}

Tabel mastertransaksi {\*kode\_transaksi, tgl\_transaksi, \*\*kode\_anggota, \*\*kode\_pegawai}

Tabel detiltransaksi (\*\*kode\_transaksi, \*\*kode\_buku, tgl\_pinjam, tgl\_kembali, jumlah\_buku, status}

*4.3. Physical Database Design*

Perancangan database secara fisik merupakan tahapan untuk mengimplementasikan hasil perancangan database secara logis menjadi tersimpan secara fisik pada media penyimpanan eksternal sesuai dengan *Database Management System* (DBMS) yang digunakan. Dapat disimpulkan bahwa proses perancangan fisik merupakan transformasi dari perancangan logis terhadap jenis DBMS yang digunakan sehingga dapat disimpan secara fisik pada media penyimpanan. *My Structured Query Language* (MySQL) merupakan pilihan DBMS yang tepat untuk mendukung aplikasi basis data yang dapat dilakukan kapanpun sebelum menuju desain logical asalkan terdapat cukup informasi mengenai kebutuhan sistem.

Tabel 1. Tabel Kategori

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1 | kode\_kategori | VARCHAR | 3 | Kode Kategori |
| 2 | nama\_kategori | VARCHAR | 50 | Nama Kategori |

Tabel 2. Tabel Penerbit

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1 | kode\_penerbit | INT | 11 | Kode penerbit |
| 2 | nama\_penerbit | VARCHAR | 50 | Nama penerbit |
| 3 | alamat\_penerbit | VARCHAR | 60 | Alamat penerbit |
| 4 | telp\_penerbit | VARCHAR | 15 | Telpon penerbit |

Tabel 3 Tabel Penulis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1 | kode\_penulis | INT | 11 | Kode Penulis |
| 2 | nama\_penulis | VARCHAR | 50 | Nama Penulis |
| 3 | alamat\_penulis | VARCHAR | 60 | Alamat Penulis |
| 4 | telp\_penulis | VARCHAR | 15 | Telp Penulis |

Tabel 4 Tabel Buku

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1 | kode\_buku | VARCHAR | 10 | Kode Buku |
| 2 | kode\_penerbit | INT | 11 | Kode penerbit |
| 3 | kode\_penulis | INT | 11 | Kode penulis |
| 4 | judul\_buku | VARCHAR | 70 | Judul buku |
| 5 | Isbn | VARCHAR | 25 | Isbn |
| 6 | tanggal\_terbit | DATE | 10 | Tanggal terbit |
| 7 | jumlah\_halaman | INT | 11 | Jumlah halaman |
| 8 | kode\_kategori | VARCHAR | 3 | Kode kategori |
| 9 | kode\_rak | VARCHAR | 3 | Kode rak |
| 10 | stok | INT | 11 | Stok |
| 11 | tanggal\_pengadaan | DATE | 10 | Tanggal pengadaan |

Tabel 5 Tabel Anggota

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1 | kode\_anggota | INT | 11 | Kode anggota |
| 2 | nama\_anggota | VARCHAR | 50 | Nama anggota |
| 3 | jenis\_kelamin | VARCHAR | 15 | Jenis kelamin |
| 4 | alamat\_anggota | VARCHAR | 60 | Alamat anggota |
| 5 | telp\_anggota | VARCHAR | 20 | Telpon anggota |
| 6 | tempat\_lahir | VARCHAR | 20 | Tempat lahir |
| 7 | tanggal\_lahir | Date | 10 | Tanggal lahir |
| 8 | usarname\_anggota | VARCHAR | 50 | Username anggota |
| 9 | password\_anggota | Text |  | Password anggota |
| 10 | tanggal\_pendaftaran | Date | 10 | Tanggal pendaftaran |

Tabel 6 Tabel Master Transaksi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1 | kode\_traksaksi | VARCHAR | 50 | Kode transaksi |
| 2 | tanggal\_transaksi | DATE | 10 | Tanggal transaksi |
| 3 | kode\_anggota | INT | 11 | Kode anggota |
| 4 | kode\_pegawai | INT | 11 | Kode pegawai |

Tabel 7 Tabel Detail Transaksi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1 | kode\_traksaksi | VARCHAR | 50 | Kode transaksi |
| 2 | kode\_buku | VARCHAR | 10 | Kode buku |
| 3 | tanggal\_pinjam | DATE | 10 | Tanggal pinjam |
| 4 | tanggal\_kembali | DATE | 10 | Tanggal kembali |
| 5 | jumlah\_buku | INT | 11 | Jumlah buku |
| 6 | status | VARCHAR | 20 | status |

Tabel 8 Tabel Pegawai

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1 | kode\_pegawai | Int | 11 | Kode pegawai |
| 2 | username\_pegawai | String | 50 | Username pegawai |
| 3 | password\_pegawai | Text |  | Password pegawai |
| 4 | nama\_pegawai | String | 50 | Nama pegawai |
| 5 | jenis\_kelamin | VARCHAR | 12 | Jenis kelamin |
| 6 | alamat\_pegawai | VARCHAR | 60 | Alamat pegawai |
| 7 | telp\_pegawai | VARCHAR | 20 | Telpon pegawai |
| 8 | tempat\_lahir | VARCHAR | 20 | Tempat lahir |
| 9 | tanggal\_lahir | DATE | 10 | Tanggal lahir |

Tabel 9 Tabel Rak

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1 | kode\_rak | VARCHAR | 3 | Kode rak |
| 2 | nama\_rak | VARCHAR | 20 | Nama rak |
| 3 | kode\_kategori | VARCHAR | 3 | Kode kategori |

*Data Definition Language* (DDL) adalah sebuah bahasa yang digunakan untuk mendefinisikan pendefinisian data. Terdiri dari perintah-perintah untuk membentuk, mengubah atau menghapus tabel beserta kolom-kolom dan type data penyusunnya, serta perintah-perintah untuk menetapkan hubungan dan batasan-batasan data. DDL adalah kumpulan perintah SQL yang digunakan untuk membuat (*create*), mengubah (*alter*) dan menghapus (*drop*) struktur dan definisi tipe data dari objek-objek database.

1. **SQL CREATE TABLE Anggota**

CREATE TABLE `anggota` (

`kode\_anggota` int(11) NOT NULL AUTO\_INCREMENT,

`username\_anggota` varchar(50) NOT NULL,

`password\_anggota` text NOT NULL,

`nama\_anggota` varchar(50) NOT NULL,

`jenis\_kelamin` varchar(15) NOT NULL,

`alamat\_anggota` varchar(60) NOT NULL,

`telp\_anggota` varchar(20) NOT NULL,

`tempat\_lahir` varchar(20) NOT NULL,

`tanggal\_lahir` date NOT NULL,

`tanggal\_pendaftaran` date NOT NULL,

PRIMARY KEY (‘kode\_anggota’)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

1. **SQL CREATE TABLE Pegawai**

CREATE TABLE `pegawai` (

`kode\_pegawai` int(11) NOT NULL AUTO\_INCREMENT,

`username\_pegawai` varchar(50) NOT NULL,

`password\_pegawai` text NOT NULL,

`nama\_pegawai` varchar(50) NOT NULL,

`jenis\_kelamin` varchar(12) NOT NULL,

`alamat\_pegawai` varchar(60) NOT NULL,

`telp\_pegawai` varchar(20) NOT NULL,

`tempat\_lahir` varchar(20) NOT NULL,

`tanggal\_lahir` date NOT NULL,

PRIMARY KEY (‘kode\_pegawai’)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

1. **SQL CREATE TABLE Penerbit**

CREATE TABLE `penerbit` (

`kode\_penerbit` int(11) NOT NULL AUTO\_INCREMENT,

`nama\_penerbit` varchar(50) NOT NULL,

`alamat\_penerbit` varchar(60) NOT NULL,

`telp\_penerbit` varchar(15) NOT NULL,

PRIMARY KEY (‘kode\_penerbit’)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

1. **SQL CREATE TABLE Penulis**

CREATE TABLE `penulis` (

`kode\_penulis` int(11) NOT NULL AUTO\_INCREMENT,

`nama\_penulis` varchar(50) NOT NULL,

`alamat\_penulis` varchar(60) NOT NULL,

`telp\_penulis` varchar(15) NOT NULL,

PRIMARY KEY (‘kode\_penulis’)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

1. **SQL CREATE TABLE Kategori**

CREATE TABLE `kategori` (

`kode\_kategori` varchar(3) NOT NULL,

`nama\_kategori` varchar(50) NOT NULL,

PRIMARY KEY (‘kode\_kategori’)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

1. **SQL CREATE TABLE Rak**

CREATE TABLE `rak` (

`kode\_rak` varchar(3) NOT NULL,

`nama\_rak` varchar(20) NOT NULL,

`kode\_kategori` varchar(3) NOT NULL,

PRIMARY KEY(‘kode\_rak’),

FOREIGN KEY (‘kode\_kategori’) REFERENCES ‘kategori’(‘kode\_kategori’)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

1. **SQL CREATE TABLE Buku**

CREATE TABLE `buku` (

`kode\_buku` varchar(10) NOT NULL,

`judul\_buku` varchar(70) NOT NULL,

`isbn` varchar(25) NOT NULL,

`kode\_penulis` int(11) NOT NULL,

`kode\_penerbit` int(11) NOT NULL,

`kode\_kategori` varchar(3) NOT NULL,

`tanggal\_terbit` date NOT NULL,

`jumlah\_halaman` int(11) NOT NULL,

`kode\_rak` varchar(3) NOT NULL,

`tanggal\_pengadaan` date NOT NULL,

`stok` int(11) NOT NULL,

PRIMARY KEY(‘kode\_buku’),

FOREIGN KEY (‘kode\_penulis’) REFERENCES ‘penulis’(‘kode\_penulis),

FOREIGN KEY (‘kode\_penerbit’) REFERENCES ‘penerbit’(‘kode\_penerbit’),

FOREIGN KEY (‘kode\_kategori’) REFERENCES ‘kategori’(‘kode\_kategori’),

FOREIGN KEY (‘kode\_rak’) REFERENCES ‘rak’(‘kode\_rak’)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

1. **SQL CREATE TABLE Detail Transaksi**

CREATE TABLE `detail\_transaksi` (

`kode\_transaksi` varchar(50) NOT NULL,

`kode\_buku` varchar(10) NOT NULL,

`jumlah\_buku` int(11) NOT NULL,

`tanggal\_pinjam` date NOT NULL,

`tanggal\_kembali` date NOT NULL,

`status` varchar(20) NOT NULL,

PRIMARY KEY(‘kode\_transaksi’),

FOREIGN KEY (‘kode\_buku’) REFERENCES ‘buku’(‘kode\_buku’)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

1. **SQL CREATE TABLE Master Transaksi**

CREATE TABLE `master\_transaksi` (

`kode\_transaksi` varchar(50) NOT NULL,

`tanggal\_transaksi` date NOT NULL,

`kode\_anggota` int(11) NOT NULL,

`kode\_pegawai` int(11) NOT NULL

FOREIGN KEY (‘kode\_transaksi’) REFERENCES ‘detail\_transaksi’(‘kode\_transaksi’),

FOREIGN KEY (‘kode\_anggota’) REFERENCES ‘anggota’(‘kode\_anggota’),

FOREIGN KEY (‘kode\_pegawai’) REFERENCES ‘pegawai’(‘kode\_pegawai’)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

Hasil dari perancangan basis data ini, dihasilkan 9 tabel basis data dan dimasukkan ke dalam basis data yang bernama perpustakaan. Perancangan basis data yang mengacu kepada model data relasional khususnya basis data perpustakaan dimaksudkan agar dalam setiap tabel yang terdapat didalam database perpustakaan saling memiliki keterkaitan demi menjamin integritas data. Selain itu, model data relasional akan memberikan gambaran yang jelas dan memberikan kemudahan bagi programmer ketika ingin membangun aplikasi pengelolaan data buku perpustakaan sekolah. Keberhasilan dalam membangun sebuah aplikasi tidak terlepas dari sebuah model basis data yang baik. Inilah yang menjadi dasar utama diperlukan sebuah rancangan basis data dengan merapkan mode relasional.

1. **Kesimpulan**

Berdasarkan hasil dari perancangan basis data perputakaan umum dengan model data relasional, maka dapat diambil kesimpulan bahwa rancangan basis data yang dirancang dengan menggunakan metode perancangan database DBLC (*Data Base Life Cycle*) telah menghasilkan bentuk database relational dengan rincian sebagai berikut ini:

1. *Conceptual database design*

Tipe entitas yang diperlukan berjumlah 9 (sembilan) entitas dengan memberikan attribute domain pada setiap nama entitas dan menghasilkan diagram hubungan entitas.

2. *Logical Database Design*

Menghasilkan relasi untuk model data logikal lokal yang mempresentasikan entity, relationship, dan attribute yang telah diidentifikasi sebelumnya.

3. *Physical Database Design*

Perancangan database menggukan database MySQL dengan *Data Definiton Language* adalah bahasa yang digunakan untuk mendefinisikan pendefinisian data. Jumlah tabel dalam basis data perpustakaan ada 9 buah yaitu, kategori, penerbit, penulis, buku, anggota, detail transaksi, master transaksi, pegawai dan rak

4. Menghasilkan database dengan model relasional diperlukan sebagai upaya untuk menjamin dan menjaga integritas data dari setiap tabel yang berhubungan.

5. Kebutuhan untuk menghasilkan sebuah aplikasi yang baik tidak terlepas dari bagaimana sebuah model dari basis data yang digunakan dan oleh karenanya maka perpustakaan umum perlu dibangun dengan model data relasional.

Referensi

1. Gutsy, 2015. Perancangan Basis Data Perputakaan Sekolah dengan Menerapkan Model Data Relasional. Pontianak: STMIK Pontianak
2. Ramez Elmasri, Sam Navathe, 2000. Fundamentals of Database Systems,4rdEdition, Addison Wesley Publishing Company.