

LAPORAN TUGAS BESAR MATA KULIAH BASIS DATA

Disusun Oleh :

1. Muhammad Nadhif Athalla (119140209)
2. Arvenda Budi Anggara (119140089)
3. Agosto Hawlai R (119140119)
4. Muhammad Fariz Luthfi (119140199)
5. Cahya Andy Mareza (119140007)

Kelas RC & RD Basis Data



**BASIS DATA
PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK ELEKTRO, INFORMATIKA, DAN SISTEM FISIS
INSTITUT TEKNOLOGI SUMATERA
2021**

DAFTAR ISI

DAFTAR ISI	2
BAB I PENDAHULUAN	4
1.1. Latar Belakang	4
1.2. Rumusan Masalah	4
1.3. Tujuan Pengembangan Aplikasi	4
BAB II RANCANGAN SISTEM	5
2.1. Deskripsi Sistem / Aplikasi	5
2.2. Use Case Diagram	5
2.3. Rancangan Basis Data	6
Rancangan Entity Relationship Diagram (ERD)	6
Mapping Entity Relationship Diagram (ERD) ke Model Relasional	6
Perancangan Fisik (Integrity Constraint)	12
BAB III IMPLEMENTASI DAN PEMBAHASAN	14
3.1 Implementasi Basis data	14
Proses pembuatan tabel majalah	14
Proses pembuatan tabel pelanggan	15
Proses pembuatan tabel pegawai	16
Proses pembuatan tabel penulis	17
Proses pembuatan tabel distributor	17
Proses menghubungkan program dengan basis data	18
3.2 Penjelasan Fitur/Fungsi aplikasi	20
BAB IV KESIMPULAN DAN SARAN	21
DAFTAR PUSTAKA	22
LOG SHEET KEGIATAN KELOMPOK	23
Biodata dan Lingkup Kerja Anggota	24

BAB I

PENDAHULUAN

1.1. Latar Belakang

Basis data adalah kumpulan informasi yang disimpan didalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari data tersebut. Konsep basis data sangat sentral dan diperlukan di dalam dunia komputer karena saat ini semua instansi dan perusahaan pastinya memiliki data yang jumlahnya tidak sedikit. Data-data tersebut melibatkan banyak entitas yang juga bisa terhubung dengan entitas lainnya melalui suatu relationship.

Konsep basis data dipelajari mulai dari desain sederhana hingga yang melibatkan banyak entitas dan relationship, proses pemetaannya menjadi relational diagram, hingga proses implementasinya. Implementasi basis data dapat digunakan pada berbagai sistem terutama yang melibatkan banyak entitas, salah satu implementasinya dapat diterapkan pada permasalahan pendataan dalam sistem produksi majalah yang melibatkan beberapa entitas.

Seperti diketahui, majalah melibatkan beberapa entitas dalam produksinya, beberapa diantaranya adalah entitas majalah itu sendiri, penulis majalah tersebut, pelanggan majalah, pegawai yang menangani pelanggan, dan distributor dalam produksi majalah tersebut. Semua entitas tersebut saling terkait dan memiliki keterhubungan, namun seringkali produksi majalah memiliki kesulitan untuk mengatur data dari entitas-entitas tersebut. Oleh karena itu, dengan kemajuan teknologi dan meningkatnya penggunaan aplikasi untuk melakukan pendataan, konsep basis data yang diimplementasikan dalam program aplikasi dapat digunakan untuk memudahkan pendataan entitas-entitas yang terlibat dalam produksi majalah tersebut.

1.2. Rumusan Masalah

1. Apakah konsep basis data dapat diimplementasikan pada permasalahan sistem pendataan majalah ?
2. Bagaimanakah rancangan basis data dalam bentuk model Entity Relationship Diagram (ERD) pada permasalahan sistem pendataan majalah ?
3. Bagaimanakah implementasi basis data pada berbagai entitas sistem pendataan majalah ?
4. Bagaimanakah sistem basis data yang sudah dibuat dapat terhubung dengan program aplikasi ?
5. Apakah program aplikasi tersebut dapat memudahkan sistem pendataan majalah ?

1.3. Tujuan Pengembangan Aplikasi

Sistem ini pada dasarnya berusaha untuk menggabungkan program dengan basis data yang bertujuan untuk memudahkan pengguna untuk melakukan pendataan dan modifikasi data pada berbagai entitas dalam produksi majalah.

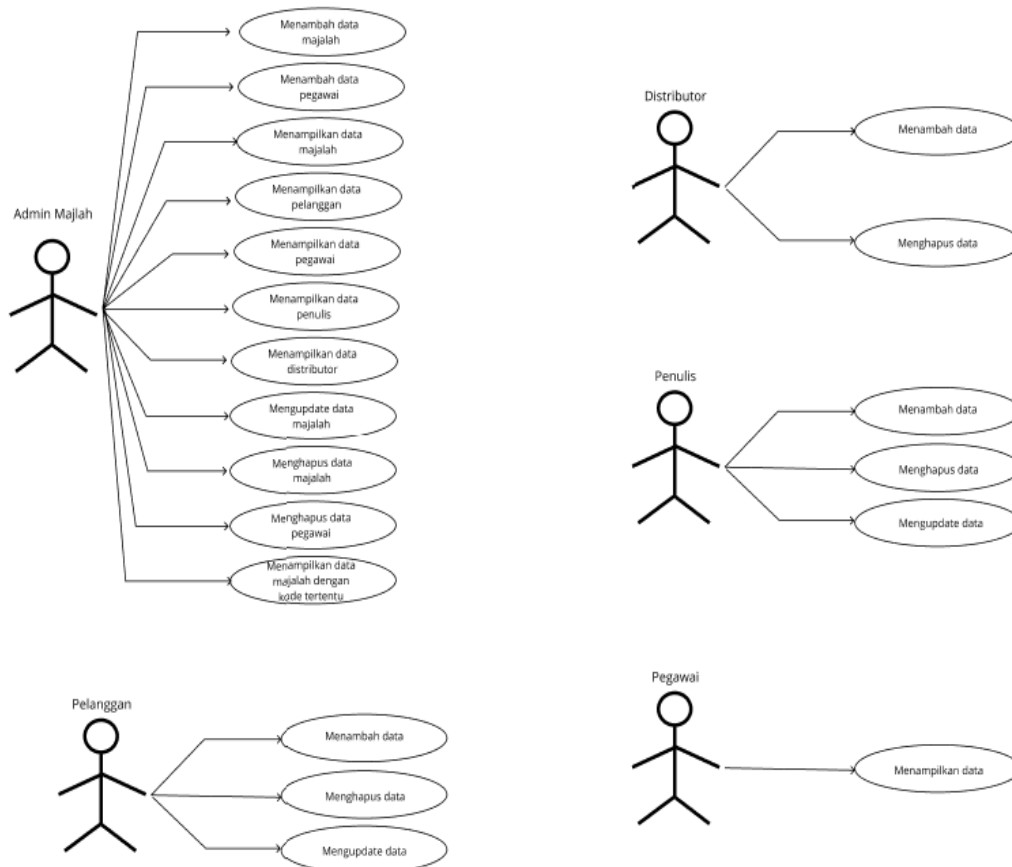
BAB II

RANCANGAN SISTEM

2.1. Deskripsi Sistem / Aplikasi

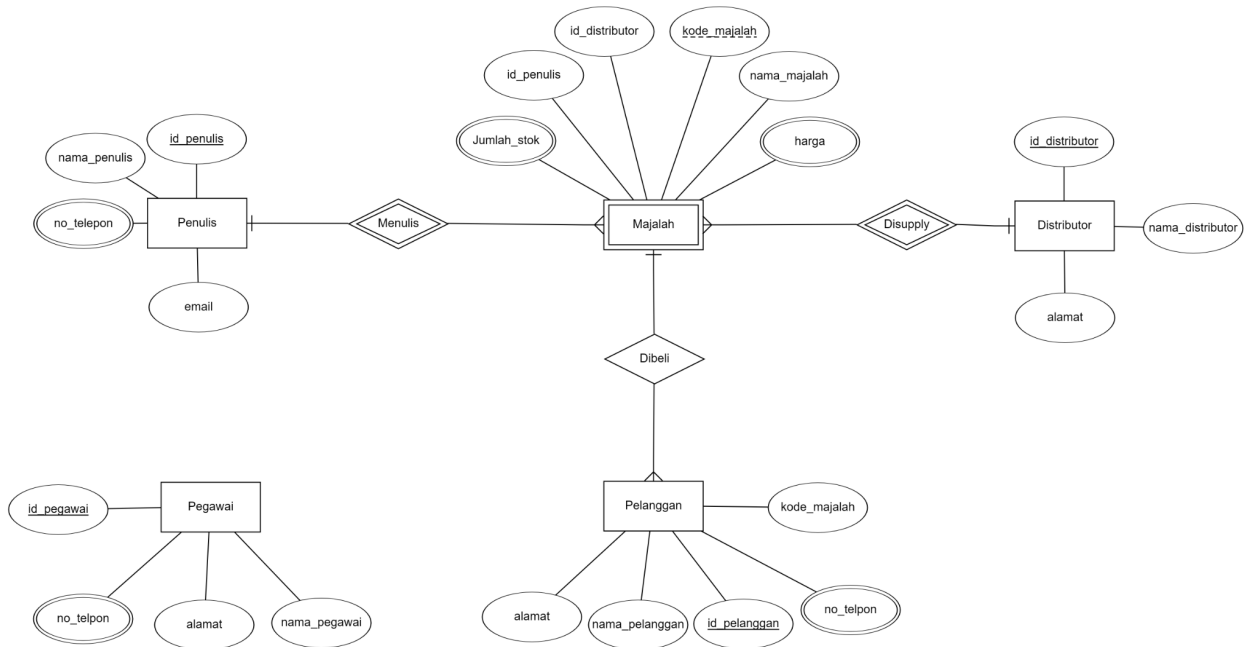
Sistem pada dasarnya menghubungkan program dengan basis data yang bertujuan untuk memudahkan pendataan berbagai entitas dalam sistem produksi majalah. Sistem dibuat dengan bahasa pemrograman python yang belum mengaplikasikan Graphical User Interface (GUI) sehingga hanya bisa berjalan pada Command Line Interface (CLI). Sistem ini dihubungkan dengan struktur basis data sesuai dengan rancangan Entity Relationship Diagram (ERD) model yang telah dikembangkan, yang diaplikasikan dengan menggunakan database MySQL dari server MariaDB.

2.2. Use Case Diagram

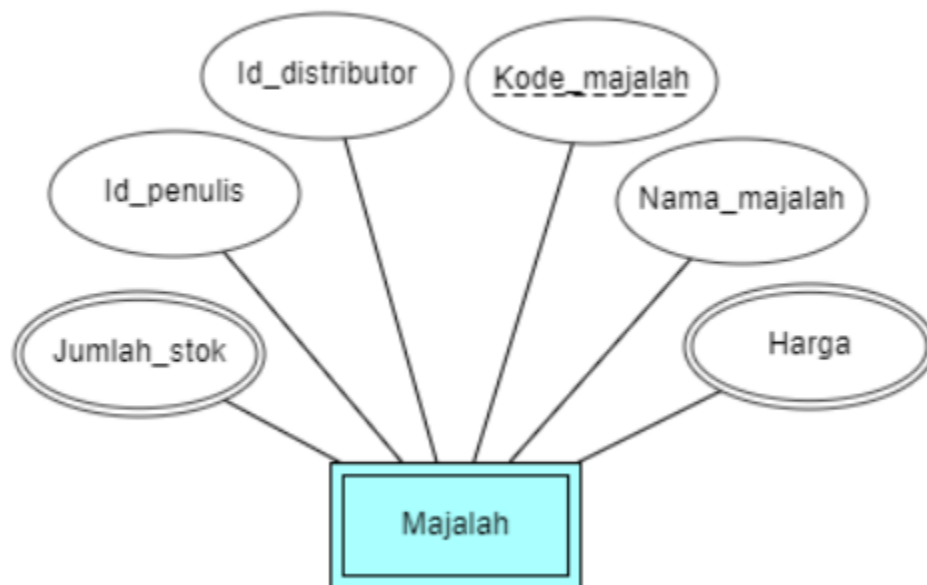


2.3. Rancangan Basis Data

A. Rancangan Entity Relationship Diagram (ERD)



B. Mapping Entity Relationship Diagram (ERD) ke Model Relasional



Majalah

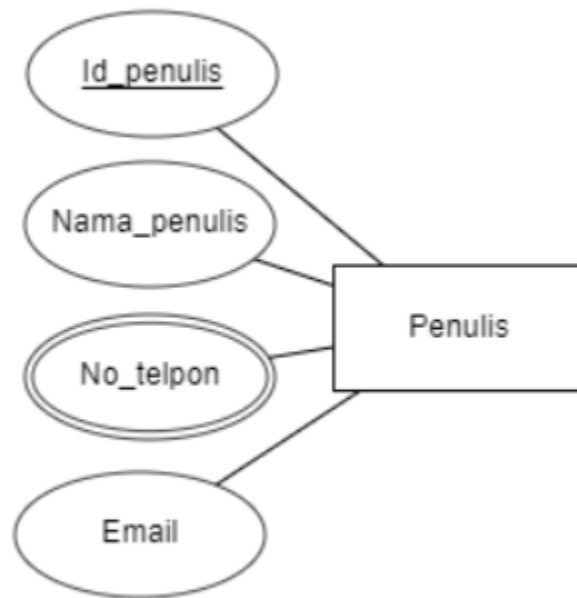
<u>Kode_majalah</u>	Id_penulis	Id_distributor	Nama_majalah
---------------------	------------	----------------	--------------

Jumlah_stok_majalah

<u>Kode_majalah</u>	Jumlah_stok
---------------------	-------------

Harga_majalah

<u>Kode_majalah</u>	Harga_majalah
---------------------	---------------

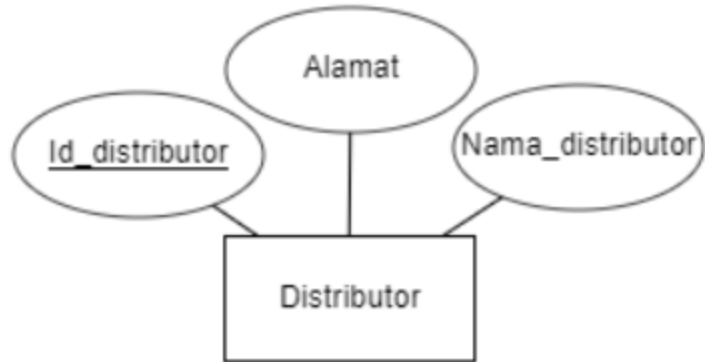


Penulis

<u>Id_penulis</u>	Nama_penulis	Email
-------------------	--------------	-------

No_telpon_penulis

<u>Id_penulis</u>	No_telpon
-------------------	-----------



Distributor

<u>Id_distributor</u>	Alamat	Nama_distributor
-----------------------	--------	------------------

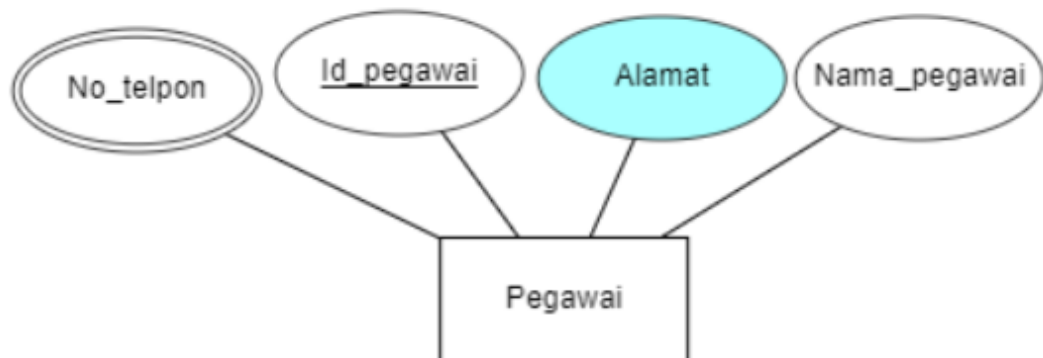


Pelanggan

<u>Id_pelanggan</u>	Nama_pelanggan	Alamat	Kode_majalah
---------------------	----------------	--------	--------------

No_telpon_pelanggan

<u>Id_pelanggan</u>	No_telpon
---------------------	-----------

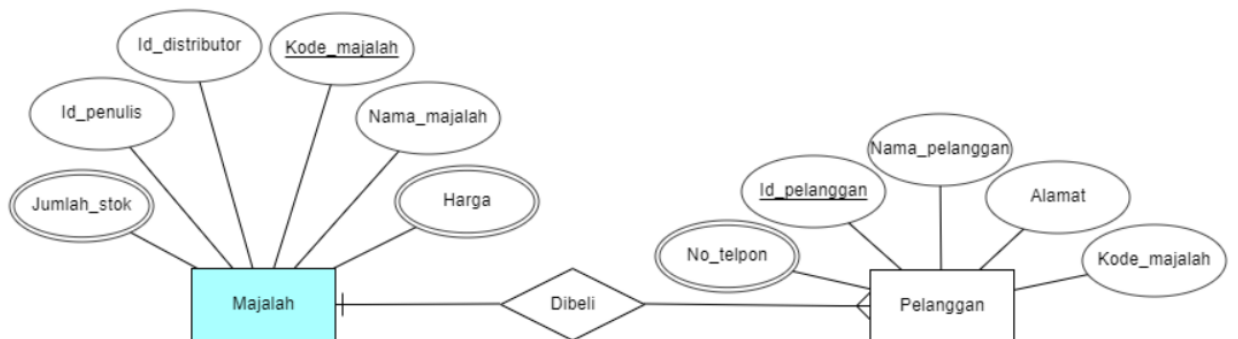


Pegawai

<u>Id_pegawai</u>	Alamat	Nama_pegawai
-------------------	--------	--------------

No_telpon_pegawai

<u>Id_pegawai</u>	No_telpon
-------------------	-----------



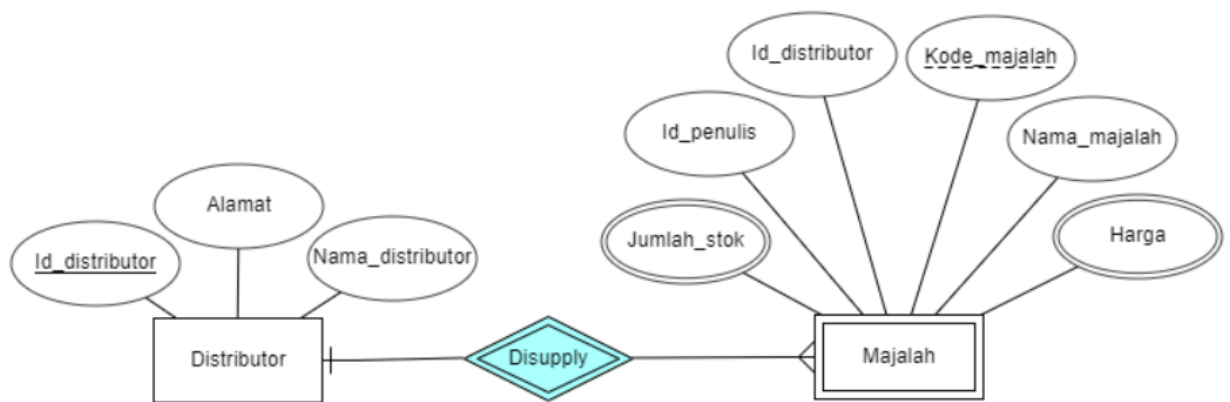
Pelanggan

<u>id_pelanggan</u>	Nama_pelanggan	Alamat	No_telpon	kode_majalah
---------------------	----------------	--------	-----------	--------------



Majalah

<u>Kode_majalah</u>	id_distributor	id_penulis	Nama_majalah	Jumlah_stok	Harga
---------------------	----------------	------------	--------------	-------------	-------



Harga_majalah

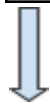
<u>Kode_majalah</u>	Harga
---------------------	-------

Jumlah_stok_majalah

<u>Kode_majalah</u>	Jumlah_stok
---------------------	-------------

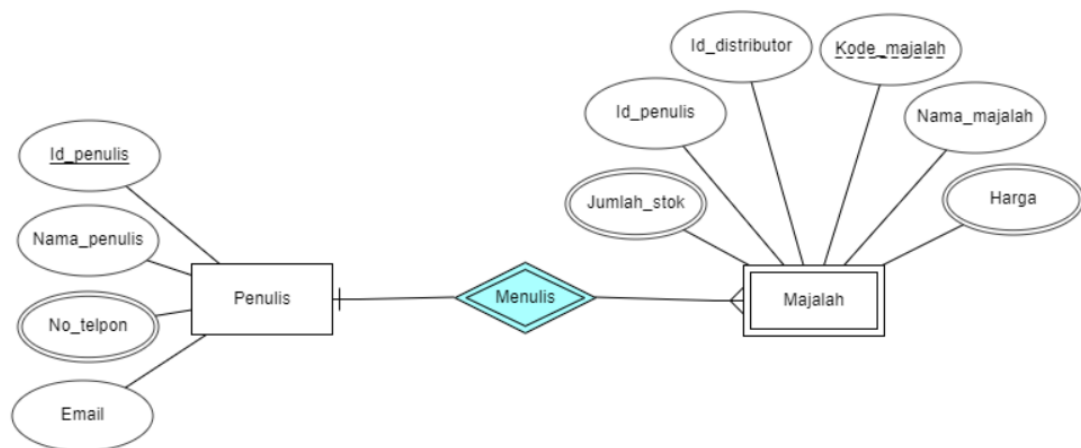
Majalah

<u>Kode_majalah</u>	Nama_majalah	Id_distributor	Id_penulis	Harga	Jumlah_stok
---------------------	--------------	----------------	------------	-------	-------------



Distributor

<u>Id_distributor</u>	Alamat	Nama_distributor
-----------------------	--------	------------------



Majalah

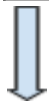
<u>Kode_majalah</u>	Nama_majalah	Id_penulis	Id_distributor	Jumlah_stok	Harga
---------------------	--------------	------------	----------------	-------------	-------

Jumlah_stok_majalah

<u>Kode_majalah</u>	Jumlah_stok
---------------------	-------------

Harga_majalah

<u>Kode_majalah</u>	Harga
---------------------	-------



Penulis

<u>Id_penulis</u>	Nama_penulis	No_telpon	Email
-------------------	--------------	-----------	-------

No_telpon_penulis

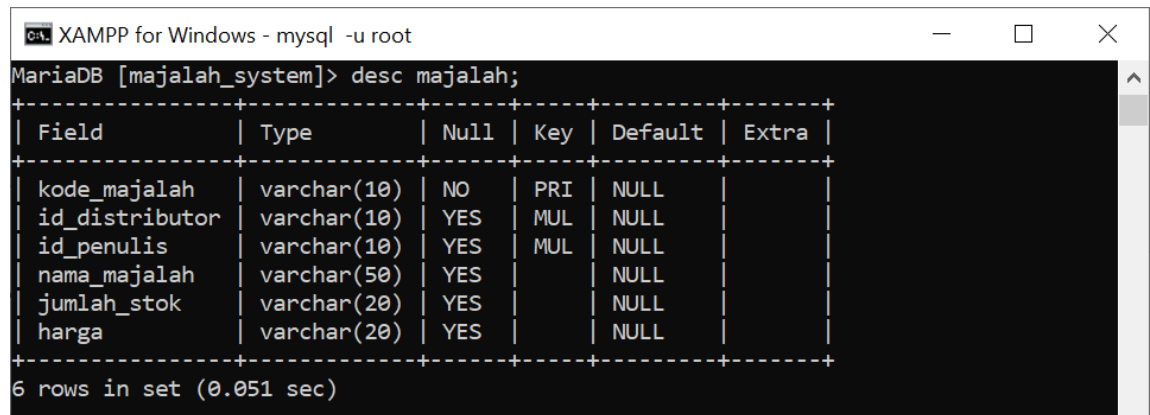
<u>Id_penulis</u>	No_telpon
-------------------	-----------

C. Perancangan Fisik (Integrity Constraint)

Perancangan Fisik (Integrity Constraint) pada basis data yang dirancang dengan ERD kemudian ditransformasikan menjadi bentuk relasional dapat dilihat pada admin MySQL dari aplikasi XAMPP yang digunakan untuk mengimplementasikan rancangan basis data sistem pendataan majalah, bentuknya adalah sebagai berikut :

1. Proses pembuatan tabel majalah :

```
create table majalah(  
  kode_majalah varchar(10) PRIMARY KEY,  
  id_distributor varchar(10),  
  id_penulis varchar(10),  
  nama_majalah varchar(50),  
  jumlah_stok varchar(20),  
  harga varchar(20),  
  FOREIGN KEY (id_distributor)  
  REFERENCES distributor(id_distributor)  
  ON DELETE RESTRICT  
  ON UPDATE CASCADE  
)ENGINE = InnoDB;
```



The screenshot shows a terminal window titled "XAMPP for Windows - mysql -u root". The prompt is "MariaDB [majalah_system]>". The command entered is "desc majalah;". The output is a table with 7 columns: Field, Type, Null, Key, Default, and Extra. The data rows are as follows:

Field	Type	Null	Key	Default	Extra
kode_majalah	varchar(10)	NO	PRI	NULL	
id_distributor	varchar(10)	YES	MUL	NULL	
id_penulis	varchar(10)	YES	MUL	NULL	
nama_majalah	varchar(50)	YES		NULL	
jumlah_stok	varchar(20)	YES		NULL	
harga	varchar(20)	YES		NULL	

At the bottom of the output, it says "6 rows in set (0.051 sec)".

2. Proses pembuatan tabel pelanggan :

```
create table pelanggan(  
  id_pelanggan varchar(10) PRIMARY KEY,  
  kode_majalah varchar(10),  
  nama_pelanggan varchar(50),  
  no_telepon varchar(20),  
  alamat char(50),  
  FOREIGN KEY (kode_majalah)  
  REFERENCES majalah(kode_majalah)  
  ON DELETE RESTRICT  
  ON UPDATE CASCADE  
)ENGINE = InnoDB;
```

```
C:\XAMPP for Windows - mysql -u root
MariaDB [majalah_system]> desc pelanggan;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_pelanggan   | varchar(10)    | NO   | PRI | NULL    |       |
| kode_majalah   | varchar(10)    | YES  | MUL | NULL    |       |
| nama_pelanggan | varchar(50)    | YES  |     | NULL    |       |
| no_telepon     | varchar(20)    | YES  |     | NULL    |       |
| alamat         | char(50)       | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.047 sec)
```

3. Proses pembuatan tabel pegawai :

```
create table pegawai(
id_pegawai varchar(10),
nama_pegawai varchar(25),
alamat varchar(50),
no_telpon varchar(20),
primary key (id_pegawai)
)ENGINE = InnoDB;
```

```
C:\XAMPP for Windows - mysql -u root
MariaDB [majalah_system]> desc pegawai;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_pegawai     | varchar(10)    | NO   | PRI | NULL    |       |
| nama_pegawai   | varchar(25)    | YES  |     | NULL    |       |
| alamat         | varchar(50)    | YES  |     | NULL    |       |
| no_telpon      | varchar(20)    | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.068 sec)
```

4. Proses pembuatan tabel penulis :

```
create table penulis(
id_penulis varchar(10),
nama_penulis varchar(50),
email varchar(50),
no_telpon varchar(20),
PRIMARY KEY (id_penulis)
)ENGINE = InnoDB;
```

```

c:\xampp> XAMPP for Windows - mysql -u root
MariaDB [majalah_system]> desc penulis;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_penulis | varchar(10)   | NO   | PRI | NULL    |       |
| nama_penulis | varchar(50)  | YES  |     | NULL    |       |
| email      | varchar(50)  | YES  |     | NULL    |       |
| no_telpon  | varchar(20)  | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.046 sec)

```

5. Proses pembuatan tabel distribusi :

```

create table distributor(
id_distributor varchar(10) NOT NULL,
nama_distributor char(30) NOT NULL,
alamat char(30) NOT NULL,
PRIMARY KEY (id_distributor)
)ENGINE = InnoDB;

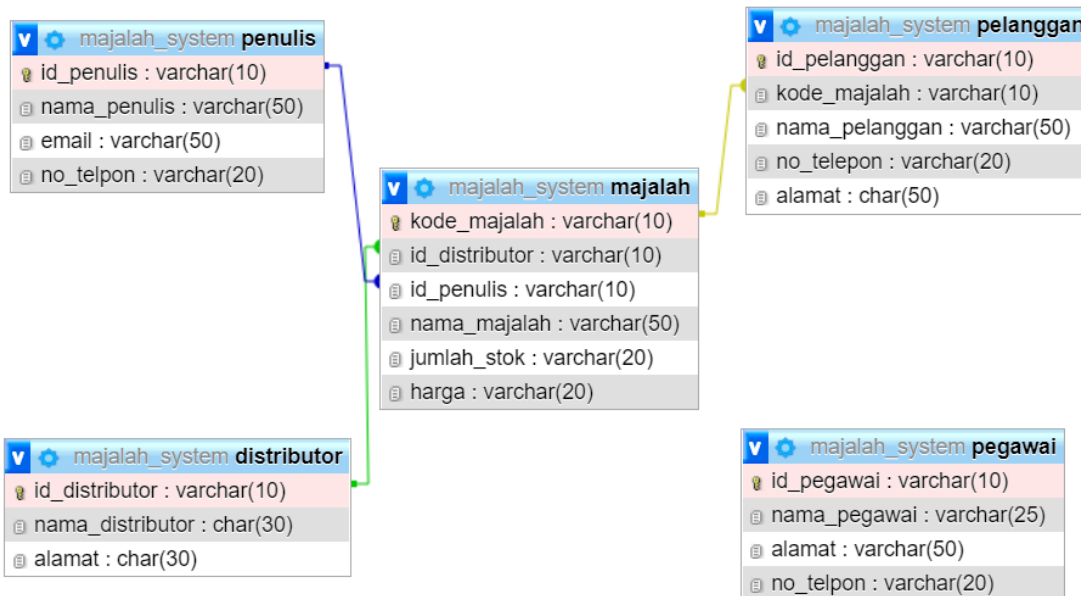
```

```

c:\xampp> XAMPP for Windows - mysql -u root
MariaDB [majalah_system]> desc distributor;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_distributor | varchar(10)   | NO   | PRI | NULL    |       |
| nama_distributor | char(30)      | NO   |     | NULL    |       |
| alamat        | char(30)      | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.078 sec)

```

Integrity Constraint dari pembuatan tabel-tabel tersebut :



BAB III

IMPLEMENTASI DAN PEMBAHASAN

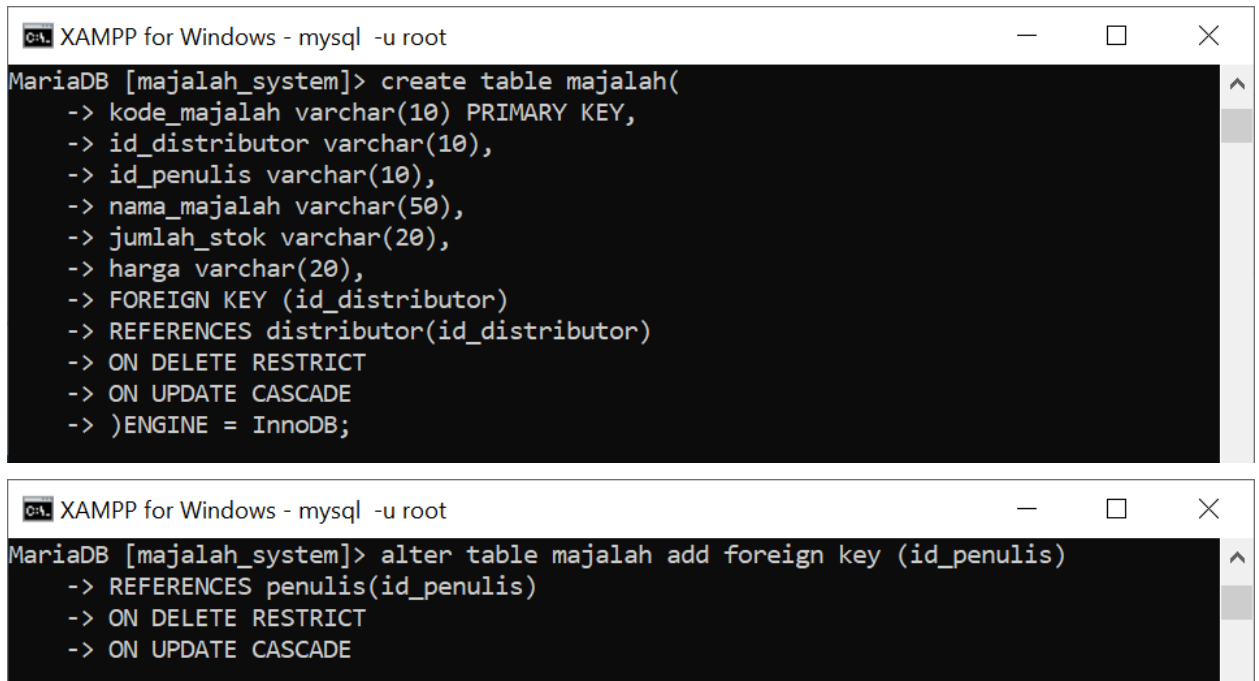
3.1 Implementasi Basis data

A. Proses pembuatan tabel majalah

Untuk membuat tabel majalah, kami menggunakan perintah create table lalu membuat isi tabelnya dengan kolom yang bertipe data, dan mendeklarasikan primary key dari tabel tersebut dengan perintah :

```
create table majalah(  
kode_majalah varchar(10) PRIMARY KEY,  
id_distributor varchar(10),  
id_penulis varchar(10),  
nama_majalah varchar(50),  
jumlah_stok varchar(20),  
harga varchar(20),  
FOREIGN KEY (id_distributor)  
REFERENCES distributor(id_distributor)  
ON DELETE RESTRICT  
ON UPDATE CASCADE  
)ENGINE = InnoDB;
```

Kemudian tambahkan primary key pada kolom id_penulis, dan menggunakan perintah desc [nama tabel] untuk menampilkan hasil tabel tersebut, implementasi yang dilakukan bisa terlihat pada gambar berikut.



The image shows two screenshots of a MySQL command prompt window titled "XAMPP for Windows - mysql -u root".

The first screenshot shows the creation of the 'majalah' table:

```
MariaDB [majalah_system]> create table majalah(  
-> kode_majalah varchar(10) PRIMARY KEY,  
-> id_distributor varchar(10),  
-> id_penulis varchar(10),  
-> nama_majalah varchar(50),  
-> jumlah_stok varchar(20),  
-> harga varchar(20),  
-> FOREIGN KEY (id_distributor)  
-> REFERENCES distributor(id_distributor)  
-> ON DELETE RESTRICT  
-> ON UPDATE CASCADE  
-> )ENGINE = InnoDB;
```

The second screenshot shows the addition of a foreign key to the 'id_penulis' column:

```
MariaDB [majalah_system]> alter table majalah add foreign key (id_penulis)  
-> REFERENCES penulis(id_penulis)  
-> ON DELETE RESTRICT  
-> ON UPDATE CASCADE
```

```
XAMPP for Windows - mysql -u root
MariaDB [majalah_system]> desc majalah;
```

Field	Type	Null	Key	Default	Extra
kode_majalah	varchar(10)	NO	PRI	NULL	
id_distributor	varchar(10)	YES	MUL	NULL	
id_penulis	varchar(10)	YES	MUL	NULL	
nama_majalah	varchar(50)	YES		NULL	
jumlah_stok	varchar(20)	YES		NULL	
harga	varchar(20)	YES		NULL	

```
6 rows in set (0.051 sec)
```

B. Proses pembuatan tabel pelanggan

Untuk membuat tabel pelanggan, kami menggunakan perintah create table lalu membuat isi tabelnya dengan kolom yang bertipe data, dan mendeklarasikan primary key dan foreign key dari tabel tersebut dengan perintah :

```
create table pelanggan(
id_pelanggan varchar(10) PRIMARY KEY,
kode_majalah varchar(10),
nama_pelanggan varchar(50),
no_telepon varchar(20),
alamat char(50),
FOREIGN KEY (kode_majalah)
REFERENCES majalah(kode_majalah)
ON DELETE RESTRICT
ON UPDATE CASCADE
)ENGINE = InnoDB;
```

Kemudian gunakan perintah desc [nama tabel] untuk menampilkan hasil tabel tersebut, implementasi yang dilakukan bisa terlihat pada gambar berikut.

```
XAMPP for Windows - mysql -u root
MariaDB [majalah_system]> create table pelanggan(
-> id_pelanggan varchar(10) PRIMARY KEY,
-> kode_majalah varchar(10),
-> nama_pelanggan varchar(50),
-> no_telepon varchar(20),
-> alamat char(50),
-> FOREIGN KEY (kode_majalah)
-> REFERENCES majalah(kode_majalah)
-> ON DELETE RESTRICT
-> ON UPDATE CASCADE
-> )ENGINE = InnoDB;
```

```
XAMPP for Windows - mysql -u root
MariaDB [majalah_system]> desc pelanggan;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_pelanggan   | varchar(10)    | NO   | PRI | NULL    |       |
| kode_majalah   | varchar(10)    | YES  | MUL | NULL    |       |
| nama_pelanggan | varchar(50)    | YES  |     | NULL    |       |
| no_telepon     | varchar(20)    | YES  |     | NULL    |       |
| alamat         | char(50)       | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.047 sec)
```

C. Proses pembuatan tabel pegawai

Untuk membuat tabel pegawai, kami menggunakan perintah create table lalu membuat isi tabelnya dengan kolom yang bertipe data, dan mendeklarasikan primary key dari tabel tersebut dengan perintah :

```
create table pegawai(
id_pegawai varchar(10),
nama_pegawai varchar(25),
alamat varchar(50),
no_telpon varchar(20),
primary key (id_pegawai)
)ENGINE = InnoDB;
```

Kemudian gunakan perintah desc [nama tabel] untuk menampilkan hasil tabel tersebut, implementasi yang dilakukan bisa terlihat pada gambar berikut.

```
XAMPP for Windows - mysql -u root
MariaDB [majalah_system]> create table pegawai(
-> id_pegawai varchar(10),
-> nama_pegawai varchar(25),
-> alamat varchar(50),
-> no_telpon varchar(20),
-> primary key (id_pegawai)
-> )ENGINE = InnoDB;
```

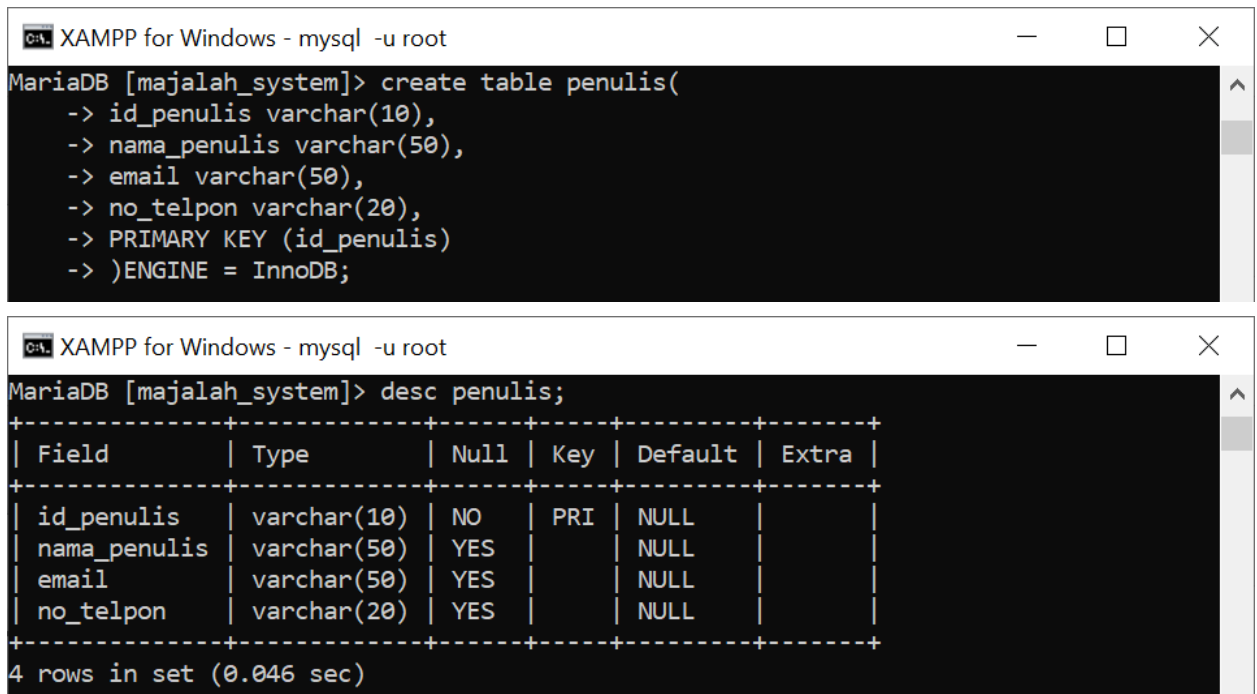
```
XAMPP for Windows - mysql -u root
MariaDB [majalah_system]> desc pegawai;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_pegawai     | varchar(10)    | NO   | PRI | NULL    |       |
| nama_pegawai   | varchar(25)    | YES  |     | NULL    |       |
| alamat         | varchar(50)    | YES  |     | NULL    |       |
| no_telpon      | varchar(20)    | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.068 sec)
```


D. Proses pembuatan tabel penulis

Untuk membuat tabel penulis, kami menggunakan perintah create table lalu membuat isi tabelnya dengan kolom yang bertipe data, dan mendeklarasikan primary key dari tabel tersebut dengan perintah :

```
create table penulis(  
id_penulis varchar(10),  
nama_penulis varchar(50),  
email varchar(50),  
no_telpon varchar(20),  
PRIMARY KEY (id_penulis)  
)ENGINE = InnoDB;
```

Kemudian gunakan perintah desc [nama tabel] untuk menampilkan hasil tabel tersebut, implementasi yang dilakukan bisa terlihat pada gambar berikut.



```
XAMPP for Windows - mysql -u root  
MariaDB [majalah_system]> create table penulis(  
-> id_penulis varchar(10),  
-> nama_penulis varchar(50),  
-> email varchar(50),  
-> no_telpon varchar(20),  
-> PRIMARY KEY (id_penulis)  
-> )ENGINE = InnoDB;
```

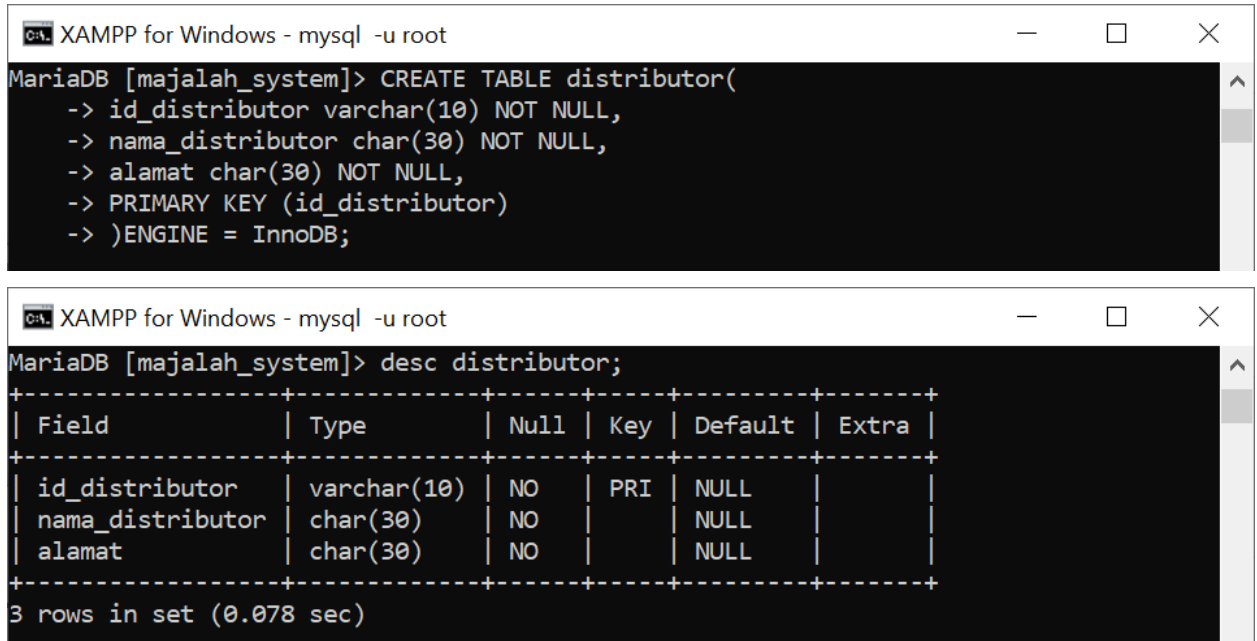
```
XAMPP for Windows - mysql -u root  
MariaDB [majalah_system]> desc penulis;  
+-----+-----+-----+-----+-----+-----+  
| Field      | Type      | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| id_penulis | varchar(10) | NO   | PRI | NULL    |       |  
| nama_penulis | varchar(50) | YES  |     | NULL    |       |  
| email      | varchar(50) | YES  |     | NULL    |       |  
| no_telpon  | varchar(20) | YES  |     | NULL    |       |  
+-----+-----+-----+-----+-----+-----+  
4 rows in set (0.046 sec)
```

E. Proses pembuatan tabel distributor

Untuk membuat tabel distributor, kami menggunakan perintah create table lalu membuat isi tabelnya dengan kolom yang bertipe data, dan mendeklarasikan primary key dari tabel tersebut dengan perintah :

```
create table distributor(  
id_distributor varchar(10) NOT NULL,  
nama_distributor char(30) NOT NULL,  
alamat char(30) NOT NULL,  
PRIMARY KEY (id_distributor)  
)ENGINE = InnoDB;
```

Kemudian gunakan perintah desc [nama tabel] untuk menampilkan hasil tabel tersebut, implementasi yang dilakukan bisa terlihat pada gambar berikut.



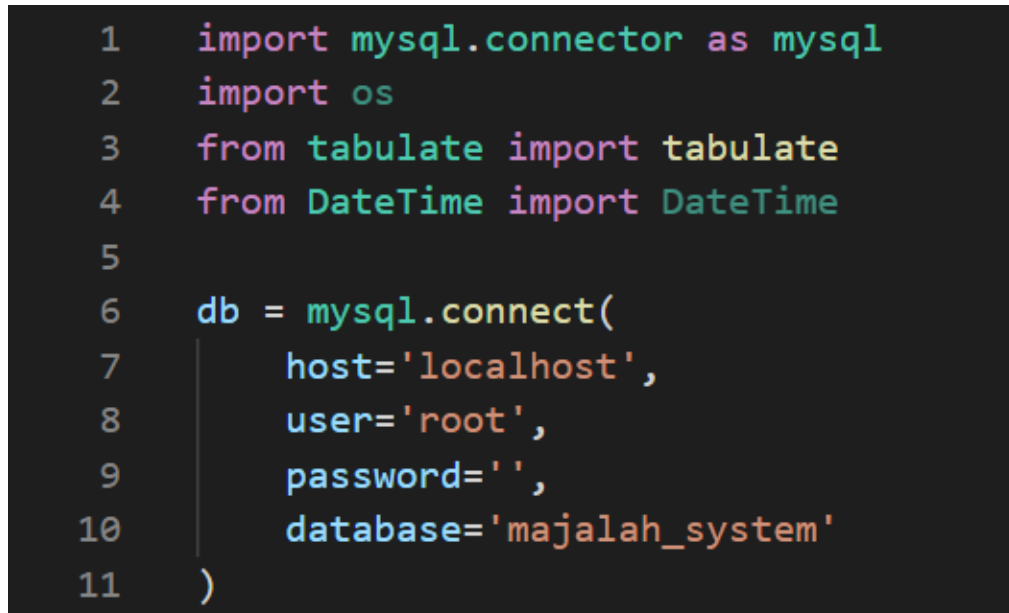
```
XAMPP for Windows - mysql -u root
MariaDB [majalah_system]> CREATE TABLE distributor(
-> id_distributor varchar(10) NOT NULL,
-> nama_distributor char(30) NOT NULL,
-> alamat char(30) NOT NULL,
-> PRIMARY KEY (id_distributor)
-> )ENGINE = InnoDB;

XAMPP for Windows - mysql -u root
MariaDB [majalah_system]> desc distributor;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_distributor | varchar(10)   | NO   | PRI | NULL    |       |
| nama_distributor | char(30)      | NO   |     | NULL    |       |
| alamat         | char(30)      | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.078 sec)
```

F. Proses menghubungkan program dengan basis data

Inti dari pengembangan tugas besar yang dilakukan adalah untuk menghubungkan basis data terstruktur yang sudah dibuat dengan program aplikasi yang diciptakan, proses ini dilakukan dengan melakukan import modul MySQL pada bahasa pemrograman python. Kemudian dilanjutkan dengan mengimplementasikan query yang biasa digunakan pada MySQL di program tersebut. Hasil implementasinya adalah sebagai berikut.

1. Menghubungkan program dengan database



```
1  import mysql.connector as mysql
2  import os
3  from tabulate import tabulate
4  from DateTime import DateTime
5
6  db = mysql.connect(
7      host='localhost',
8      user='root',
9      password='',
10     database='majalah_system'
11 )
```

2. Implementasi query insert pada program :

```
13 def Nambah_data_majalah(db):
14     print("\nInsert data majalah")
15     print("Syarat 1 : ID distributor suatu majalah harus sama dengan daftar ID distributor")
16     print("Syarat 2 : ID penulis suatu majalah harus sama dengan daftar ID penulis")
17     Show_majalah(db)
18     Show_distributor(db)
19     Show_penulis(db)
20     kode_majalah = input("Kode majalah : ")
21     id_distributor = input("ID distributor : ")
22     id_penulis = input("ID penulis : ")
23     nama_majalah = input("Nama majalah : ")
24     jumlah_stok = input("Jumlah stok : ")
25     harga = input("Harga majalah : ")
26
27     val = (kode_majalah, id_distributor, id_penulis, nama_majalah, jumlah_stok, harga)
28     mycursor = db.cursor()
29     sql = "INSERT INTO majalah (kode_majalah, id_distributor, id_penulis, nama_majalah, jumlah_stok, harga) VALUES (%s,%s,%s,%s,%s,%s)"
30     mycursor.execute(sql,val)
31     db.commit()
32     print("{} data berhasil disimpan".format(mycursor.rowcount))
```

3. Implementasi query select all pada program :

```
93 def Show_majalah(db):
94     mycursor = db.cursor()
95     mycursor.execute("SELECT * FROM majalah")
96     myresult = mycursor.fetchall()
97
98     print(tabulate(myresult, headers=["kode_majalah", "id_distributor", "id_penulis", "nama_majalah", "jumlah_stok", "harga"],
```

4. Implementasi query update pada program :

```
160 def Update_data_pelanggan(db):
161     print("\nUpdate data pelanggan")
162     print("Syarat : Kode majalah tidak bisa diubah")
163     Show_pelanggan(db)
164     id_pelanggan = input("ID Pelanggan : ")
165     no_telepon = input("Nomor telepon baru : ")
166     nama_pelanggan = input("Nama baru : ")
167     alamat = input("Alamat baru : ")
168
169     mycursor = db.cursor()
170     val = (no_telepon, nama_pelanggan, alamat, id_pelanggan)
171     sql = "UPDATE pelanggan SET no_telepon = (%s), nama_pelanggan = (%s), alamat = (%s) WHERE id_pelanggan = (%s)"
172     mycursor.execute(sql,val)
173     db.commit()
174     print("{} data berhasil disimpan".format(mycursor.rowcount))
```

5. Implementasi query delete pada program :

```
216 def Delete_data_pegawai(db):
217     print("\nDelete data pegawai")
218     mycursor = db.cursor()
219     Show_pegawai(db)
220     id_pegawai = input("Pilih id pegawai : ")
221     sql = "DELETE FROM pegawai WHERE id_pegawai=%s"
222     val = [id_pegawai]
223     mycursor.execute(sql,val)
224     db.commit()
225     print("{} data berhasil dihapus".format(mycursor.rowcount))
```

6. Implementasi Subquery pada program :

```
296 def Subquery_distributor(db):
297     id_distributor = input("Masukkan id distributor : ")
298     mycursor = db.cursor()
299     val1 = [id_distributor]
300     sql1 = "SELECT * FROM distributor WHERE id_distributor = (%s)"
301     mycursor.execute(sql1, val1)
302     myresult = mycursor.fetchall()
303     print(tabulate(myresult, headers=["id_distributor", "nama_distributor", "alamat"], tablefmt="grid"))
```

7. Implementasi query join pada program :

```
305 def Majalah_distributor(db):
306     mycursor = db.cursor()
307     mycursor.execute("SELECT id_distributor, kode_majalah, nama_majalah,
308     myresult = mycursor.fetchall()
309     print(tabulate(myresult, headers=["id_distributor", "kode_majalah",
    jumlah_stok, harga, nama_distributor, alamat FROM majalah NATURAL JOIN distributor")
    "nama_majalah", "jumlah_stok", "harga", "nama_distributor", "alamat"], tablefmt="grid"))
```

3.2 Penjelasan Fitur/Fungsi aplikasi

Aplikasi pendataan dalam produksi majalah ini memiliki beberapa fitur yang dapat digunakan, yakni sebagai berikut :

1. Aplikasi ini memungkinkan pengguna untuk menampilkan data entitas yang tersimpan dalam produksi majalah
2. Aplikasi ini memungkinkan pengguna untuk memasukkan data untuk beberapa entitas dalam produksi majalah
3. Aplikasi ini memungkinkan pengguna untuk melakukan pembaruan data untuk beberapa entitas dalam produksi majalah
4. Aplikasi ini memungkinkan pengguna menghapus data entitas yang tersimpan dalam produksi majalah
5. Aplikasi ini memungkinkan pengguna untuk menampilkan data yang diinginkan sesuai dengan ID entitas yang dimasukkan
6. Aplikasi ini memungkinkan pengguna untuk menampilkan data dari dua sampai tiga tabel yang terkait
7. Aplikasi ini terhubung langsung dengan database produksi majalah tersebut sehingga memudahkan pendataan entitas yang dilakukan

BAB IV

KESIMPULAN DAN SARAN

A. Kesimpulan

Kesimpulan yang bisa didapat untuk menjawab rumusan masalah dari pengerjaan tugas besar pembuatan aplikasi sistem pendataan majalah yang terhubung dengan basis data ini adalah sebagai berikut :

1. Konsep basis data dapat diimplementasikan pada permasalahan sistem pendataan majalah
2. Basis data dari sistem pendataan majalah dapat dirancang menggunakan model Entity Relationship Diagram (ERD)
3. Implementasi basis data diawali dari rancangan ERD, transformasi menjadi bentuk relasional, lalu diaplikasikan pada sistem database MySQL
4. Program aplikasi yang dibuat sudah terhubung dengan sistem basis data
5. Program aplikasi dapat memudahkan sistem pendataan majalah

B. Saran

Untuk perbaikan tugas besar kedepannya, kami menerima kritik dan saran seluas-luasnya untuk aplikasi dan laporan pengerjaan tugas besar ini. Saran yang dapat diberikan adalah sebagai berikut :

1. Menurut kami, sebaiknya diadakan minimal 1 kali asistensi berupa pertemuan untuk memudahkan pengerjaan tugas besar.
2. Menurut kami, sebaiknya aplikasi ini melibatkan lebih banyak entitas dan relationship untuk pendataan entitas yang lebih luas
3. Menurut kami, sebaiknya aplikasi ini lebih dikembangkan terutama dalam fungsi perhitungan sehingga memiliki fitur dan manfaat yang lebih luas

DAFTAR PUSTAKA

- I. Gat. (2015). Perancangan Basis Data Perpustakaan Sekolah dengan Menerapkan Model Data Relasional. *Citec Journal*, 2, No. 4(4 agustus-oktober 2015), 304–315.
- II. Latief, M. (2010). Pendekatan Database Untuk Manajemen Data Dalam Meningkatkan Kemampuan Mahasiswa Mengaplikasikan Konsep Basis Data. *Prosiding APTEKINDO*, 6(1), 231–238.
- III. Prasetya, W. S. (2015). Perancangan Model Basis Data Relasional Dengan Metode Database Life Cycle. *Prosiding Seminar Nasional Informatika 2015*, 91–98.

LOG SHEET KEGIATAN KELOMPOK

No	Tanggal	Nama	NIM	Kegiatan
1	21-04-2021	1. Muhammad Nadhif Athalla 2. Augusto Hawlai R 3. Muhammad Fariz Luthfi 4. Cahya Andy Mareza 5. Arvenda Budi Anggara	1. 119140209 2. 119140119 3. 119140199 4. 119140007 5. 119140089	Membuat Rancangan (ERD modelling) basis data

No	Tanggal	Nama	NIM	Kegiatan
2	08-05-2021	1. Muhammad Nadhif Athalla 2. Augusto Hawlai R 3. Muhammad Fariz Luthfi 4. Cahya Andy Mareza 5. Arvenda Budi Anggara	1. 119140209 2. 119140119 3. 119140199 4. 119140007 5. 119140089	Perbaikan ERD dan perancangan bentuk fisik basis data pada MySQL

No	Tanggal	Nama	NIM	Kegiatan
3	11-05-2021	1. Muhammad Nadhif Athalla 2. Augusto Hawlai R 3. Muhammad Fariz Luthfi 4. Cahya Andy Mareza 5. Arvenda Budi Anggara	1. 119140209 2. 119140119 3. 119140199 4. 119140007 5. 119140089	Menghubungkan rancangan basis data pada program dan pembuatan fitur pada program

No	Tanggal	Nama	NIM	Kegiatan
4	18-05-2021	1. Muhammad Nadhif Athalla 2. Augusto Hawlai R 3. Muhammad Fariz Luthfi 4. Cahya Andy Mareza 5. Arvenda Budi Anggara	1. 119140209 2. 119140119 3. 119140199 4. 119140007 5. 119140089	Pembuatan laporan, PPT presentasi, dan persiapan akhir untuk presentasi dan demo program

Biodata dan Lingkup Kerja Anggota

1. Biodata Anggota Pertama



Nama : Muhammad Nadhif Athalla
NIM : 119140209
Prodi : Teknik Informatika
Alamat : Jl. Raden Gunawan Perum GTS blok A no 10, Lampung Selatan
Nomor Telepon : 081368732255
Lingkup Kerja : Koordinator Kelompok, Pembuatan Backend Program, Perancangan ERD, Mentransformasikan ERD ke dalam Bentuk Relasional, Pembuatan Database, Pencarian Referensi dan Pustaka, Pembuatan Laporan, Pembuatan Slide Presentasi, Presentasi Hasil Pengerjaan, dan Mendemokan Program

2. Biodata Anggota Kedua



Nama : Arvenda Budi Anggara
NIM : 119140089
Prodi : Teknik Informatika
Alamat : Perum Korpri Blok E2 No 3, Sukarame, Bandar Lampung
Nomor Telepon : 081369908998

Lingkup Kerja : Pembuatan Database, Perancangan ERD, Pencarian Referensi, Pembuatan Laporan, Pembuatan Slide Presentasi, Presentasi Hasil Pengerjaan

3. Biodata Anggota Ketiga



Nama : Augusto Hawlai Rajagukguk
NIM : 119140119
Prodi : Teknik Informatika
Alamat : Gg Mangga Jati Agung, Lampung Selatan
Nomor Telepon : 0895600999779
Lingkup Kerja : Pembuatan Backend Program, Perancangan ERD, Pembuatan Laporan dan Slide, Pengecekan Kesalahan ERD, Presentasi Hasil Pengerjaan.

4. Biodata Anggota Keempat



Nama : Muhammad Fariz Luthfi
NIM : 119140199
Prodi : Teknik Informatika
Alamat : Jalan Cut Nyak Dien Gang Way Madang 62 Bandar Lampung
Nomor Telepon : 089633607890
Lingkup Kerja : Pembuatan Laporan, Perancangan ERD, Mentransformasikan ERD ke dalam Bentuk Relasional, Pengecekan Kesalahan ERD, Pembuatan Slide Presentasi, dan Presentasi Hasil Pengerjaan.

5. Biodata Anggota Kelima



Nama : Cahya Andy Mareza
NIM : 119140007
Prodi : Teknik Informatika
Alamat : Jl. Pulau Sebesi No.58C, Sukarame, Bandar Lampung
Nomor Telepon : 082278671738
Lingkup Kerja : Perancangan ERD, Pembuatan Database, Pembuatan Laporan dan Slide Presentasi, Pencarian Referensi dan Pustaka, Presentasi Hasil Pengerjaan.