

Pemrograman Basis Data



Nama : Asyafa Ditra Al Hauna
Skema : Pemrograman Basis Data
NIM : 21102116

Website Peminjaman Barang IT

Website peminjaman barang IT merupakan platform berbasis web yang dirancang untuk membantu IT support dalam mengelola peminjaman barang IT dalam sebuah perusahaan. Website ini akan membantu admin dalam mengelola data karyawan yang meminjam, barang IT, dan pencatatan data saat peminjaman dan pengembalian barang IT.

Website ini memiliki dua jenis akses yaitu: karyawan (user) dan IT Support (admin)

User adalah karyawan-karyawan biasa yang tidak berada dalam divisi IT Support. User ini akan memanfaatkan website ini dalam melakukan peminjaman dan pengembalian barang IT.

Admin adalah seorang karyawan dalam divisi IT Support pada suatu perusahaan yang mengelola inventaris barang IT. Admin ini akan menggunakan website untuk memantau pencatatan peminjaman dan pengembalian barang serta mengelola data inventori barang IT perusahaan yang dapat dipinjam oleh karyawan lainnya.

Informasi akun :

- Admin
Username : asyafa@company.com
Password : asyafa123
- User
Username : ditra@company.com
Password : ditra123

1. Tabel user

Field	Tipe Data
id user	int (primary key)
username	varchar(50)
email	varchar(100)
password	varchar(50)
akses	enum('admin', 'user')

2. Tabel Karyawan

Field	Tipe Data
id karyawan	int (primary key)
nama	varchar(255)
pekerjaan	varchar(50)
divisi	varchar(50)
jenis_kelamin	enum('l', 'p')

no_telp	varchar(15)
---------	-------------

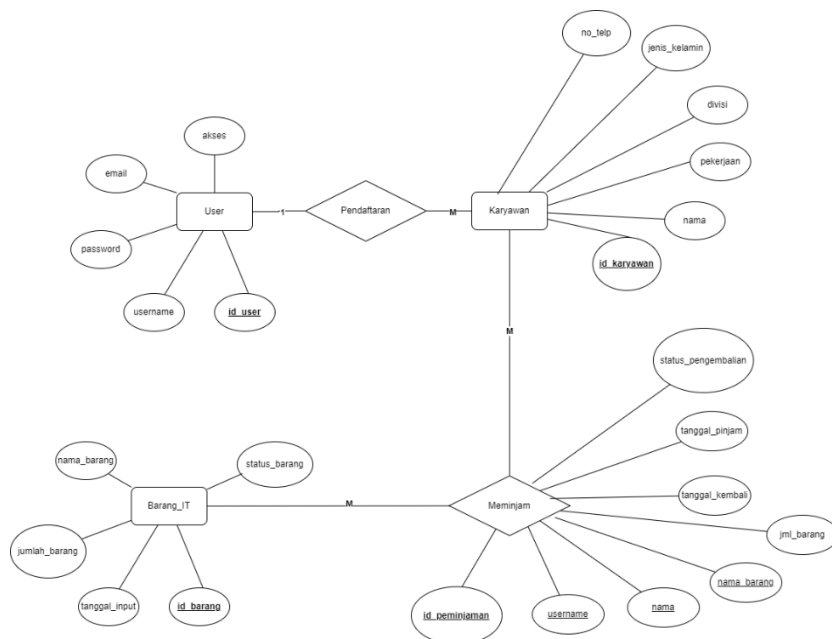
3. Tabel Barang_IT

Field	Type Data
id barang	int (primary key)
nama barang	varchar(100)
jumlah barang	int
status barang	enum('tersedia', 'dipinjam')
tanggal_input	date

4. Tabel Meminjam

Field	Type Data
id_peminjaman	int (primary key)
username	varchar(50) FK
nama	varchar(255) FK
nama barang	varchar(100) FK
tanggal pinjam	datetime
tanggal kembali	datetime
status_pengembalian	enum('dikembalikan', 'belum dikembalikan') default 'belum dikembalikan'

ERD :



Link dokumentasi github: [Ditra02/serkom_basdat](https://github.com/Ditra02/serkom_basdat)

Dokumentasi Praktik

A. Data Definition Language

1. Mengaktifkan database MySQL xampp melalui cmd

```
Select Administrator: XAMPP for Windows - mysql -u root

aditr@ASYAFXCROC c:\xampp
# mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8
Server version: 10.4.25-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

2. Membuat database

```
MariaDB [(none)]> create database peminjaman_barang_it
-> ;
Query OK, 1 row affected (0.011 sec)

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| 21102116_akademik2 |
| 21102116_rentalfilm |
| buku |
| db_akademik |
| db_akademik2 |
| db_akademik_p |
| db_kafe_21102116 |
| db_kepegawaian |
| db_kursus |
| db_perpustakaan |
| db_rentalmobil |
| db_warung |
| dbdemo |
| information_schema |
| latian |
| library management system |
| mysql |
| pegawai |
| peminjaman_barang_it |
| performance_schema |
| pethouse |
| phpmyadmin |
| practice_laravel |
| practice_laravel12 |
| rentalfilm |
| school |
| taiga |
| test |
| tubes |
| tugas_kelompok1 |
+-----+
```

3. Membuat empat tabel dalam database

a. Tabel user

```
MariaDB [peminjaman_barang_it]> create table user (
-> id_user INT AUTO_INCREMENT PRIMARY KEY,
-> username VARCHAR(50) NOT NULL,
-> email VARCHAR(100) NOT NULL,
-> password VARCHAR(50) NOT NULL,
-> akses ENUM('admin', 'user') NOT NULL
-> );
Query OK, 0 rows affected (0.040 sec)
```

```
MariaDB [peminjaman_barang_it]> describe user;
```

Field	Type	Null	Key	Default	Extra
id_user	int(11)	NO	PRI	NULL	auto_increment
username	varchar(50)	NO		NULL	
email	varchar(100)	NO		NULL	
password	varchar(50)	NO		NULL	
akses	enum('admin','user')	NO		NULL	

```
5 rows in set (0.024 sec)
```

b. Tabel karyawan

```
MariaDB [peminjaman_barang_it]> create table karyawan (
-> id_karyawan INT AUTO_INCREMENT PRIMARY KEY,
-> nama VARCHAR(255) NOT NULL,
-> pekerjaan VARCHAR(100) NOT NULL,
-> divisi VARCHAR(50) NOT NULL,
-> jenis_kelamin enum('L', 'P') NOT NULL,
-> no_telp VARCHAR(15) NOT NULL
-> );
Query OK, 0 rows affected (0.047 sec)
```

```
MariaDB [peminjaman_barang_it]> describe karyawan;
```

Field	Type	Null	Key	Default	Extra
id_karyawan	int(11)	NO	PRI	NULL	auto_increment
nama	varchar(255)	NO		NULL	
pekerjaan	varchar(100)	NO		NULL	
divisi	varchar(50)	NO		NULL	
jenis_kelamin	enum('L','P')	NO		NULL	
no_telp	varchar(15)	NO		NULL	

```
6 rows in set (0.011 sec)
```

c. Tabel barang_it

```
MariaDB [peminjaman_barang_it]> create table barang_it(
-> id_barang INT AUTO_INCREMENT PRIMARY KEY,
-> nama_barang VARCHAR(100) NOT NULL,
-> jumlah_barang INT(40) NOT NULL,
-> status_barang ENUM('tersedia', 'dipinjam') NOT NULL,
-> tanggal_input DATETIME NOT NULL
-> );
Query OK, 0 rows affected (0.038 sec)
```

```
MariaDB [peminjaman_barang_it]> describe barang_it;
```

Field	Type	Null	Key	Default	Extra
id_barang	int(11)	NO	PRI	NULL	auto_increment
nama_barang	varchar(100)	NO		NULL	
jumlah_barang	int(40)	NO		NULL	
status_barang	enum('tersedia','dipinjam')	NO		NULL	
tanggal_input	datetime	NO		NULL	

```
5 rows in set (0.021 sec)
```

d. Tabel meminjam (tabel transaksi)

```

MariaDB [peminjaman_barang_it]> create table meminjam(
  -> id_peminjaman INT AUTO_INCREMENT PRIMARY KEY,
  -> username VARCHAR(50) NOT NULL,
  -> nama VARCHAR(255) NOT NULL,
  -> nama_barang VARCHAR(100) NOT NULL,
  -> tanggal_pinjam DATETIME NOT NULL,
  -> tanggal_kembali DATETIME NOT NULL,
  -> status_pengembalian ENUM('dikembalikan', 'belum dikembalikan') default 'belum dikembalikan'
  -> NOT NULL
  -> );
Query OK, 0 rows affected (0.048 sec)

MariaDB [peminjaman_barang_it]> ALTER TABLE meminjam
  -> ADD jml_barang INT(10) NOT NULL AFTER nama_barang;
Query OK, 0 rows affected (0.016 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [peminjaman_barang_it]> describe meminjam;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_peminjaman | int(11) | NO | PRI | NULL | auto_increment |
| username | varchar(50) | NO | | NULL | |
| nama | varchar(255) | NO | | NULL | |
| nama_barang | varchar(100) | NO | | NULL | |
| jml_barang | int(10) | NO | | NULL | |
| tanggal_pinjam | datetime | NO | | NULL | |
| tanggal_kembali | datetime | NO | | NULL | |
| status_pengembalian | enum('dikembalikan','belum dikembalikan') | NO | | belum dikembalikan | |
+-----+-----+-----+-----+-----+-----+

```

- e. Menampilkan nama tabel yang berhasil dibuat

```

MariaDB [peminjaman_barang_it]> show tables;
+-----+
| Tables_in_peminjaman_barang_it |
+-----+
| barang_it |
| karyawan |
| meminjam |
| user |
+-----+
4 rows in set (0.001 sec)

```

4. Menginputkan data pada tabel yang berhasil dibuat

- a. Tabel user

```

MariaDB [peminjaman_barang_it]> INSERT INTO user (username, email, password, akses)
  -> VALUES
  -> ('asyafa', 'asyafa@company.com', 'asyafa123', 'admin'),
  -> ('ditra', 'ditra@company.com', 'ditra123', 'user'),
  -> ('alhauna', 'alhauna@company.com', 'alhauna123', 'user'),
  -> ('jamal', 'jamal@company.com', 'jamal123', 'user');
Query OK, 4 rows affected (0.011 sec)
Records: 4 Duplicates: 0 Warnings: 0

MariaDB [peminjaman_barang_it]> select * from user;
+-----+-----+-----+-----+-----+
| id_user | username | email | password | akses |
+-----+-----+-----+-----+-----+
| 1 | asyafa | asyafa@company.com | asyafa123 | admin |
| 2 | ditra | ditra@company.com | ditra123 | user |
| 3 | alhauna | alhauna@company.com | alhauna123 | user |
| 4 | jamal | jamal@company.com | jamal123 | user |
+-----+-----+-----+-----+-----+
4 rows in set (0.000 sec)

```

- b. Tabel karyawan

```
MariaDB [peminjaman_barang_it]> INSERT INTO karyawan (nama, pekerjaan, divisi, jenis_kelamin, no_telp)
-> VALUES
-> ('Asyafa arthur', 'pengelola barang it', 'it support', 'L', '081227227910'),
-> ('Ditra james', 'asisten manajer', 'manajemen', 'P', '081227227911'),
-> ('Al Hauna john', 'sekertaris', 'marketing', 'L', '081227227912'),
-> ('Jamal Udin', 'asisten manajer', 'advisor', 'L', '081227227913');
Query OK, 4 rows affected (0.011 sec)
Records: 4 Duplicates: 0 Warnings: 0

MariaDB [peminjaman_barang_it]> select * from karyawan;
+-----+-----+-----+-----+-----+-----+
| id_karyawan | nama          | pekerjaan      | divisi      | jenis_kelamin | no_telp      |
+-----+-----+-----+-----+-----+-----+
| 1           | Asyafa arthur | pengelola barang it | it support   | L             | 081227227910 |
| 2           | Ditra james   | asisten manajer  | manajemen    | P             | 081227227911 |
| 3           | Al Hauna john | sekertaris       | marketing    | L             | 081227227912 |
| 4           | Jamal Udin    | asisten manajer  | advisor      | L             | 081227227913 |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.001 sec)
```

c. Tabel barang_it

```
MariaDB [peminjaman_barang_it]> INSERT INTO barang_it (nama_barang, jumlah_barang, status_barang, tanggal_input)
-> VALUES
-> ('proyektor', 5, 'tersedia', '2024-12-03 15:00:00'),
-> ('monitor', 2, 'tersedia', '2024-12-04 15:00:00'),
-> ('keyboard', 7, 'tersedia', '2024-12-05 15:00:00'),
-> ('kabel hdmi', 10, 'tersedia', '2024-12-06 15:00:00');
Query OK, 4 rows affected (0.005 sec)
Records: 4 Duplicates: 0 Warnings: 0

MariaDB [peminjaman_barang_it]> select * from barang_it;
+-----+-----+-----+-----+-----+
| id_barang | nama_barang | jumlah_barang | status_barang | tanggal_input |
+-----+-----+-----+-----+-----+
| 1         | proyektor   | 5             | tersedia      | 2024-12-03 15:00:00 |
| 2         | monitor     | 2             | tersedia      | 2024-12-04 15:00:00 |
| 3         | keyboard    | 7             | tersedia      | 2024-12-05 15:00:00 |
| 4         | kabel hdmi  | 10            | tersedia      | 2024-12-06 15:00:00 |
+-----+-----+-----+-----+-----+
4 rows in set (0.001 sec)
```

d. Tabel peminjaman

```
MariaDB [peminjaman_barang_it]> INSERT INTO meminjam (username, nama, nama_barang, jml_barang, tanggal_pinjam, tanggal_kembali, status_pengembalian)
-> VALUES
-> ('ditra', 'Ditra James', 'proyektor', 1, '2024-12-07 14:00:00', '2024-12-08 14:00:00', 'belum dikembalikan'),
-> ('jamal', 'Jamal Udin', 'monitor', 1, '2024-12-08 14:00:00', '2024-12-09 14:00:00', 'belum dikembalikan');
Query OK, 2 rows affected (0.004 sec)
Records: 2 Duplicates: 0 Warnings: 0

MariaDB [peminjaman_barang_it]> select * from meminjam;
+-----+-----+-----+-----+-----+-----+-----+
| id_peminjaman | username | nama          | nama_barang | jml_barang | tanggal_pinjam | tanggal_kembali | status_pengembalian |
+-----+-----+-----+-----+-----+-----+-----+
| 1             | ditra    | Ditra James   | proyektor   | 1          | 2024-12-07 14:00:00 | 2024-12-08 14:00:00 | belum dikembalikan |
| 2             | jamal    | Jamal Udin    | monitor     | 1          | 2024-12-08 14:00:00 | 2024-12-09 14:00:00 | belum dikembalikan |
+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.000 sec)
```

5. Menghubungkan relasi antar tabel dan menambahkan index field

a. Tabel user

Index field pada username diterapkan untuk meningkatkan kecepatan query saat pencarian data dengan username

```
MariaDB [peminjaman_barang_it]> ALTER TABLE user ADD INDEX idx_username (username);
Query OK, 0 rows affected (0.025 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [peminjaman_barang_it]>

MariaDB [peminjaman_barang_it]> describe user;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id_user    | int(11)       | NO   | PRI | NULL    | auto_increment |
| username   | varchar(50)   | NO   | MUL | NULL    |                |
| email      | varchar(100)  | NO   |     | NULL    |                |
| password   | varchar(50)   | NO   |     | NULL    |                |
| akses      | enum('admin','user') | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.018 sec)
```

b. Tabel karyawan

Index field pada nama diterapkan untuk meningkatkan kecepatan query saat pencarian data dengan nama

```
MariaDB [peminjaman_barang_it]> ALTER TABLE karyawan ADD INDEX idx_nama (nama);
Query OK, 0 rows affected (0.012 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [peminjaman_barang_it]>

MariaDB [peminjaman_barang_it]> describe karyawan;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| id_karyawan | int(11)   | NO   | PRI | NULL    | auto_increment |
| nama        | varchar(255) | NO   | MUL | NULL    |              |
| pekerjaan   | varchar(100) | NO   |     | NULL    |              |
| divisi      | varchar(50) | NO   |     | NULL    |              |
| jenis_kelamin | enum('L','P') | NO   |     | NULL    |              |
| no_telp     | varchar(15) | NO   |     | NULL    |              |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.021 sec)
```

c. Tabel barang_it

```
MariaDB [peminjaman_barang_it]> ALTER TABLE barang_it ADD INDEX idx_nama_barang (nama_barang);
Query OK, 0 rows affected (0.023 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [peminjaman_barang_it]>

MariaDB [peminjaman_barang_it]> describe barang_it;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| id_barang  | int(11)   | NO   | PRI | NULL    | auto_increment |
| nama_barang | varchar(100) | NO   | MUL | NULL    |              |
| jumlah_barang | int(40)   | NO   |     | NULL    |              |
| status_barang | enum('tersedia','dipinjam') | NO   |     | NULL    |              |
| tanggal_input | datetime | NO   |     | NULL    |              |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.010 sec)
```

d. Tabel meminjam

1. Menambahkan CONSTRAINT berupa foreign key dari tabel meminjam ke tabel barang_it. Hal ini bertujuan untuk menghubungkan tabel meminjam dengan tabel barang_it.

```
MariaDB [peminjaman_barang_it]> describe meminjam;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| id_peminjaman | int(11)   | NO   | PRI | NULL    | auto_increment |
| username      | varchar(50) | NO   |     | NULL    |              |
| nama          | varchar(255) | NO   |     | NULL    |              |
| nama_barang   | varchar(100) | NO   | MUL | NULL    |              |
| jml_barang    | int(10)   | NO   |     | NULL    |              |
| tanggal_pinjam | datetime | NO   |     | NULL    |              |
| tanggal_kembali | datetime | NO   |     | NULL    |              |
| status_pengembalian | enum('dikembalikan','belum dikembalikan') | NO   |     | belum dikembalikan |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.013 sec)
```

2. Menambahkan CONSTRAINT berupa foreign key dari tabel meminjam ke tabel user. Hal ini bertujuan untuk menghubungkan tabel meminjam dengan tabel user.


```

MariaDB [peminjaman_barang_it]> ALTER TABLE meminjam ADD CONSTRAINT FK_meminjam_user FOREIGN KEY (username) REFERENCES user (username);
Query OK, 2 rows affected (0.075 sec)
Records: 2 Duplicates: 0 Warnings: 0

MariaDB [peminjaman_barang_it]> describe meminjam;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_peminjaman | int(11) | NO | PRI | NULL | auto_increment |
| username | varchar(50) | NO | MUL | NULL | |
| nama | varchar(255) | NO | | NULL | |
| nama_barang | varchar(100) | NO | MUL | NULL | |
| jml_barang | int(10) | NO | | NULL | |
| tanggal_pinjam | datetime | NO | | NULL | |
| tanggal_kembali | datetime | NO | | NULL | |
| status_pengembalian | enum('dikembalikan','belum dikembalikan') | NO | | belum dikembalikan | |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.013 sec)

```

3. Menambahkan CONSTRAINT berupa foreign key dari tabel meminjam ke tabel karyawan. Hal ini bertujuan untuk menghubungkan tabel meminjam dengan tabel karyawan.

```

MariaDB [peminjaman_barang_it]> ALTER TABLE meminjam ADD CONSTRAINT FK_meminjam_karyawan FOREIGN KEY (nama) REFERENCES karyawan (nama);
Query OK, 2 rows affected (0.071 sec)
Records: 2 Duplicates: 0 Warnings: 0

MariaDB [peminjaman_barang_it]> describe meminjam;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_peminjaman | int(11) | NO | PRI | NULL | auto_increment |
| username | varchar(50) | NO | MUL | NULL | |
| nama | varchar(255) | NO | MUL | NULL | |
| nama_barang | varchar(100) | NO | MUL | NULL | |
| jml_barang | int(10) | NO | | NULL | |
| tanggal_pinjam | datetime | NO | | NULL | |
| tanggal_kembali | datetime | NO | | NULL | |
| status_pengembalian | enum('dikembalikan','belum dikembalikan') | NO | | belum dikembalikan | |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.018 sec)

```

B. Data Manipulation Language

1. INSERT

```

MariaDB [peminjaman_barang_it]> INSERT INTO user (username, email, password, akses)
-> VALUES
-> ('aji', 'aji@company.com', 'aji123', 'user');
Query OK, 1 row affected (0.005 sec)

MariaDB [peminjaman_barang_it]> select * from user;
+-----+-----+-----+-----+-----+
| id_user | username | email | password | akses |
+-----+-----+-----+-----+-----+
| 1 | asyafa | asyafa@company.com | asyafa123 | admin |
| 2 | ditra | ditra@company.com | ditra123 | user |
| 3 | alhauna | alhauna@company.com | alhauna123 | user |
| 4 | jamal | jamal@company.com | jamal123 | user |
| 5 | aji | aji@company.com | aji123 | user |
+-----+-----+-----+-----+-----+
5 rows in set (0.000 sec)

```

2. UPDATE

```

MariaDB [peminjaman_barang_it]> UPDATE user
-> set username = 'al_hauna', email = 'al_hauna@company.com', password = 'al_hauna123', akses = 'admin'
-> WHERE id_user = 3;
Query OK, 1 row affected (0.004 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [peminjaman_barang_it]> select * from user;
+-----+-----+-----+-----+-----+
| id_user | username | email | password | akses |
+-----+-----+-----+-----+-----+
| 1 | asyafa | asyafa@company.com | asyafa123 | admin |
| 2 | ditra | ditra@company.com | ditra123 | user |
| 3 | al_hauna | al_hauna@company.com | al_hauna123 | admin |
| 4 | jamal | jamal@company.com | jamal123 | user |
| 5 | aji | aji@company.com | aji123 | user |
+-----+-----+-----+-----+-----+
5 rows in set (0.000 sec)

```

3. SELECT

```
MariaDB [peminjaman_barang_it]> select * from user;
```

id_user	username	email	password	akses
1	asyafa	asyafa@company.com	asyafa123	admin
2	ditra	ditra@company.com	ditra123	user
3	al_hauna	al_hauna@company.com	al_hauna123	admin
4	jamal	jamal@company.com	jamal123	user
5	aji	aji@company.com	aji123	user

```
5 rows in set (0.000 sec)

MariaDB [peminjaman_barang_it]>
```

4. DELETE

```
MariaDB [peminjaman_barang_it]> DELETE FROM user
-> WHERE id_user = 5;
Query OK, 1 row affected (0.010 sec)

MariaDB [peminjaman_barang_it]> select * from user;
```

id_user	username	email	password	akses
1	asyafa	asyafa@company.com	asyafa123	admin
2	ditra	ditra@company.com	ditra123	user
3	al_hauna	al_hauna@company.com	al_hauna123	admin
4	jamal	jamal@company.com	jamal123	user

```
4 rows in set (0.000 sec)
```

C. Transaction Control Language

1. Prosedur

Prosedur GetUserByAccess berguna untuk menampilkan semua pengguna yang terdapat dalam tabel user berdasarkan hak akses nya. Hak akses ini akan diinputkan sebagai parameter kondisi sehingga dapat menampilkan user yang memiliki akses user biasa dan admin.

```
MariaDB [peminjaman_barang_it]> CREATE PROCEDURE GetUserByAccess(IN access_right VARCHAR(10))
-> BEGIN
-> SELECT * from user WHERE akses = access_right;
-> END //
Query OK, 0 rows affected (0.018 sec)

MariaDB [peminjaman_barang_it]> _

MariaDB [peminjaman_barang_it]> DELIMITER ;
MariaDB [peminjaman_barang_it]> CALL GetUserByAccess('admin');
```

id_user	username	email	password	akses
1	asyafa	asyafa@company.com	asyafa123	admin
3	al_hauna	al_hauna@company.com	al_hauna123	admin

```
2 rows in set (0.001 sec)

Query OK, 0 rows affected (0.018 sec)

MariaDB [peminjaman_barang_it]> CALL GetUserByAccess('user');
```

id_user	username	email	password	akses
2	ditra	ditra@company.com	ditra123	user
4	jamal	jamal@company.com	jamal123	user

```
2 rows in set (0.000 sec)

Query OK, 0 rows affected (0.008 sec)
```

Procedure GetStatusBarangByNamaBarang berguna untuk menampilkan status barang yang terdapat dalam tabel barang_it berdasarkan namanya. Nama barang akan

diinputkan sebagai parameter kondisi sehingga dapat menampilkan nama dan status barang yang sesuai.

```
MariaDB [peminjaman_barang_it]> DELIMITER //
MariaDB [peminjaman_barang_it]> CREATE PROCEDURE GetStatusBarangByNamaBarang(IN nama_brg VARCHAR(100))
-> BEGIN
-> SELECT nama_barang, status_barang FROM barang_it WHERE nama_barang = nama_brg;
-> END //
Query OK, 0 rows affected (0.007 sec)

MariaDB [peminjaman_barang_it]> CALL GetStatusBarangByNamaBarang('proyektor');
+-----+-----+
| nama_barang | status_barang |
+-----+-----+
| proyektor   | tersedia      |
+-----+-----+
1 row in set (0.002 sec)

Query OK, 0 rows affected (0.008 sec)

MariaDB [peminjaman_barang_it]> CALL GetStatusBarangByNamaBarang('monitor');
+-----+-----+
| nama_barang | status_barang |
+-----+-----+
| monitor     | tersedia      |
+-----+-----+
1 row in set (0.001 sec)

Query OK, 0 rows affected (0.005 sec)
```

Procedur GetDataKaryawan berguna untuk melihat semua data seorang karyawan berdasarkan namanya. Nama lengkap karyawan akan dijadikan sebagai parameter kondisi dalam mencari data yang sesuai.

```
MariaDB [peminjaman_barang_it]> DELIMITER //
MariaDB [peminjaman_barang_it]> CREATE PROCEDURE GetDataKaryawan(IN nama_lengkap VARCHAR(255))
-> BEGIN
-> SELECT * FROM karyawan WHERE nama = nama_lengkap;
-> END //
Query OK, 0 rows affected (0.008 sec)

MariaDB [peminjaman_barang_it]> CALL GetDataKaryawan('Ditra James');
+-----+-----+-----+-----+-----+-----+
| id_karyawan | nama      | pekerjaan      | divisi      | jenis_kelamin | no_telp      |
+-----+-----+-----+-----+-----+-----+
| 2          | Ditra James | asisten manajer | manajemen   | P              | 081227227911 |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.002 sec)

Query OK, 0 rows affected (0.009 sec)
```

2. Function

Fungsi GetUserAkses berguna untuk langsung menampilkan akses sebuah user berdasarkan username yang dijadikan parameter.

```
MariaDB [peminjaman_barang_it]> DELIMITER //
MariaDB [peminjaman_barang_it]> CREATE FUNCTION GetUserAkses(username_parameter VARCHAR(50)) RETURNS ENUM
('admin', 'user')
-> BEGIN
-> DECLARE akses_pengguna ENUM('admin', 'user');
-> SELECT akses INTO akses_pengguna FROM user WHERE username = username_parameter;
-> RETURN akses_pengguna;
-> END //
Query OK, 0 rows affected (0.015 sec)

MariaDB [peminjaman_barang_it]> DELIMITER ;
MariaDB [peminjaman_barang_it]> SELECT GetUserAkses('ditra');
+-----+
| GetUserAkses('ditra') |
+-----+
| user                   |
+-----+
1 row in set (0.001 sec)
```

Fungsi GetSpecificBarangCount berguna untuk melihat jumlah barang tersedia berdasarkan namanya. Dimana nama barang akan dijadikan sebagai parameter untuk data yang sesuai

```

MariaDB [peminjaman_barang_it]> DELIMITER //
MariaDB [peminjaman_barang_it]> CREATE FUNCTION GetSpecificBarangCount(nama_brg VARCHAR(100)) RETURNS INT
--
--> BEGIN
--> DECLARE jml_barang INT;
--> SELECT jumlah_barang INTO jml_barang FROM barang_it WHERE nama_barang = nama_brg;
--> RETURN jml_barang;
--> END //
Query OK, 0 rows affected (0.016 sec)

MariaDB [peminjaman_barang_it]> DELIMITER ;
MariaDB [peminjaman_barang_it]> SELECT GetSpecificBarangCount('proyektor');
+-----+
| GetSpecificBarangCount('proyektor') |
+-----+
| 5 |
+-----+
1 row in set (0.001 sec)

```

Fungsi GetPhoneByName berguna untuk menampilkan no telp karyawan berdasarkan nama. Nama lengkap karyawan akan dijadikan sebagai parameter untuk mencari data yang sesuai

```

MariaDB [peminjaman_barang_it]> DELIMITER //
MariaDB [peminjaman_barang_it]> CREATE FUNCTION GetPhoneByName(nama_lengkap VARCHAR(100)) RETURNS VARCHAR
(15)
--
--> BEGIN
--> DECLARE phone VARCHAR(15);
--> SELECT no_telp INTO phone FROM karyawan WHERE nama = nama_lengkap;
--> RETURN phone;
--> END //
Query OK, 0 rows affected (0.014 sec)

MariaDB [peminjaman_barang_it]> DELIMITER ;
MariaDB [peminjaman_barang_it]> SELECT GetPhoneByName('Ditra james');
+-----+
| GetPhoneByName('Ditra james') |
+-----+
| 081227227911 |
+-----+
1 row in set (0.001 sec)

```

Fungsi GetStatusBarang berguna untuk menampilkan status barang yang ada dalam tabel peminjaman berdasarkan namanya. Nama barang akan dijadikan sebagai parameter untuk mencari data yang sesuai.

```

MariaDB [peminjaman_barang_it]> CREATE FUNCTION GetStatusBarang(nama_brg VARCHAR(100))
--
--> RETURNS ENUM('dikembalikan', 'belum dikembalikan')
--> BEGIN
--> DECLARE status ENUM('dikembalikan', 'belum dikembalikan');
--> SELECT status_pengembalian INTO status FROM meminjam WHERE nama_barang = nama_brg;
--> RETURN status;
--> END //
Query OK, 0 rows affected (0.015 sec)

MariaDB [peminjaman_barang_it]> DELIMITER ;
MariaDB [peminjaman_barang_it]> SELECT GetStatusBarang('proyektor');
+-----+
| GetStatusBarang('proyektor') |
+-----+
| belum dikembalikan |
+-----+
1 row in set (0.001 sec)

```

3. Trigger

Trigger CheckAccessBeforeInsert, berguna untuk mengecek data yang akan diinputkan ke tabel user, khususnya untuk data aksesnya dimana hanya boleh bernilai 'admin' atau 'user'.

```

MariaDB [peminjaman_barang_it]> DELIMITER //
MariaDB [peminjaman_barang_it]> CREATE TRIGGER CheckAccessBeforeInsert
-> BEFORE INSERT ON user
-> FOR EACH ROW
-> BEGIN
-> IF NEW.akses NOT IN ('admin', 'user') THEN
-> SIGNAL SQLSTATE '22000'
-> SET MESSAGE_TEXT = 'Hak akses yang dimasukkan tidak valid. Hak akses hanya dapat berisi "admin" atau "user"
.';
-> END IF;
-> END //
Query OK, 0 rows affected (0.008 sec)

MariaDB [peminjaman_barang_it]> DELIMITER ;
MariaDB [peminjaman_barang_it]> INSERT INTO user (username, email, password, akses)
-> VALUES
-> ('jamal', 'jamal@company.com', 'jamal123', 'boss');
ERROR 1644 (22000): Hak akses yang dimasukkan tidak valid. Hak akses hanya dapat berisi "admin" atau "user".
MariaDB [peminjaman_barang_it]>

```

Trigger CheckCountGreaterThanZero, berguna untuk mengecek data barang it yang akan diinputkan ke dalam tabel barang_it, khususnya mengecek data jumlah barang haruslah lebih besar dari 0.

```

MariaDB [peminjaman_barang_it]> DELIMITER //
MariaDB [peminjaman_barang_it]> CREATE TRIGGER CheckCountGreaterThanZero
-> BEFORE INSERT ON barang_it
-> FOR EACH ROW
-> BEGIN
-> IF NEW.jumlah_barang < 0 THEN
-> SIGNAL SQLSTATE '42000'
-> SET MESSAGE_TEXT = 'Jumlah barang it tidak boleh < 0';
-> END IF;
-> END //
Query OK, 0 rows affected (0.012 sec)

MariaDB [peminjaman_barang_it]> DELIMITER ;
MariaDB [peminjaman_barang_it]> INSERT INTO barang_it (nama_barang, jumlah_barang, status_barang, tanggal_input)
-> VALUES
-> ('kabel VGA', -1, 'tersedia', '2024-12-09 15:00:00');
ERROR 1644 (42000): Jumlah barang it tidak boleh < 0
MariaDB [peminjaman_barang_it]>

```

Trigger CheckGenderBeforeInsert, berguna untuk mengecek data yang akan dimasukkan dalam tabel karyawan, khususnya data jenis kelamin dimana hanya boleh bernilai "L" atau "P".

```

MariaDB [peminjaman_barang_it]> DELIMITER //
MariaDB [peminjaman_barang_it]> CREATE TRIGGER CheckGenderBeforeInsert
-> BEFORE INSERT ON karyawan
-> FOR EACH ROW
-> BEGIN
-> IF NEW.jenis_kelamin NOT IN ('P', 'L') THEN
-> SIGNAL SQLSTATE '42000'
-> SET MESSAGE_TEXT = 'Jenis kelamin hanya boleh bernilai "P" atau "L".';
-> END IF;
-> END //
Query OK, 0 rows affected (0.013 sec)

MariaDB [peminjaman_barang_it]> DELIMITER ;
MariaDB [peminjaman_barang_it]> INSERT INTO karyawan (nama, pekerjaan, divisi, jenis_kelamin, no_telp)
-> VALUES
-> ('Jamal Udin', 'asisten manajer', 'advisor', 'X', '081227227913');
ERROR 1644 (42000): Jenis kelamin hanya boleh bernilai "P" atau "L".
MariaDB [peminjaman_barang_it]>

```

Trigger CheckBeforeMeminjam, berguna untuk mengecek data yang akan dimasukkan kedalam tabel meminjam khususnya data tanggal kembali dimana tanggal tersebut tidak boleh lebih awal dari tanggal pinjam.

```

MariaDB [peminjaman_barang_it]> DELIMITER //
MariaDB [peminjaman_barang_it]> CREATE TRIGGER CheckBeforeMeminjam
-> BEFORE INSERT ON meminjam
-> FOR EACH ROW
-> BEGIN
-> IF NEW.tanggal_kembali < NEW.tanggal_pinjam THEN
-> SIGNAL SQLSTATE '42000'
-> SET MESSAGE_TEXT = 'Tanggal pengembalian barang tidak bisa lebih awal dari tanggal pinjam.';
-> END IF;
-> END //
Query OK, 0 rows affected (0.020 sec)

MariaDB [peminjaman_barang_it]> DELIMITER ;

MariaDB [peminjaman_barang_it]> INSERT INTO meminjam (username, nama, nama_barang, jml_barang, tanggal_pinjam, tanggal_kembali, status_pengembalian)
-> VALUES
-> ('jamal', 'Jamal Udin', 'mouse', '1', '2024-12-03 15:00:00', '2024-12-02 15:00:00', 'belum dikembalikan');
ERROR 1644 (42000): Tanggal pengembalian barang tidak bisa lebih awal dari tanggal pinjam.
MariaDB [peminjaman_barang_it]>

```

4. Commit

Commit berguna untuk mengonfirmasi transaksi yang terjadi dan menerapkan perubahan yang dilakukan pada query setelah dituliskannya START TRANSACTION, sehingga perubahan data menjadi permanen.

```

MariaDB [peminjaman_barang_it]> DELIMITER //
MariaDB [peminjaman_barang_it]> START TRANSACTION; //
Query OK, 0 rows affected (0.000 sec)

MariaDB [peminjaman_barang_it]> INSERT INTO barang_it (nama_barang, jumlah_barang, status_barang, tanggal_input)
-> VALUES
-> ('kabel sound', 6, 'tersedia', '2024-12-08 15:00:00');
-> //
Query OK, 1 row affected (0.001 sec)

MariaDB [peminjaman_barang_it]> COMMIT //
Query OK, 0 rows affected (0.010 sec)

MariaDB [peminjaman_barang_it]> DELIMITER ;

```

5. Rollback

Rollback berguna untuk mengembalikan kembali perubahan yang dilakukan setelah dituliskannya START TRANSACTION, rollback hanya dapat dijalankan jika perintah COMMIT belum dijalankan.

```

MariaDB [peminjaman_barang_it]> DELIMITER //
MariaDB [peminjaman_barang_it]> START TRANSACTION;
-> INSERT INTO barang_it (nama_barang, jumlah_barang, status_barang, tanggal_input)
-> VALUES
-> ('kabel sound', 6, 'tersedia', '2024-12-08 15:00:00'); //
Query OK, 0 rows affected (0.001 sec)

Query OK, 1 row affected (0.009 sec)

MariaDB [peminjaman_barang_it]> SELECT * FROM barang_it;
-> //
+-----+-----+-----+-----+-----+
| id_barang | nama_barang | jumlah_barang | status_barang | tanggal_input |
+-----+-----+-----+-----+-----+
| 1 | proyektor | 5 | tersedia | 2024-12-03 15:00:00 |
| 2 | monitor | 2 | tersedia | 2024-12-04 15:00:00 |
| 3 | keyboard | 7 | tersedia | 2024-12-05 15:00:00 |
| 4 | kabel hdmi | 10 | tersedia | 2024-12-06 15:00:00 |
| 6 | kabel sound | 6 | tersedia | 2024-12-08 15:00:00 |
+-----+-----+-----+-----+-----+
5 rows in set (0.000 sec)

MariaDB [peminjaman_barang_it]> ROLLBACK; //
Query OK, 0 rows affected (0.002 sec)

MariaDB [peminjaman_barang_it]> SELECT * FROM barang_it; //
+-----+-----+-----+-----+-----+
| id_barang | nama_barang | jumlah_barang | status_barang | tanggal_input |
+-----+-----+-----+-----+-----+
| 1 | proyektor | 5 | tersedia | 2024-12-03 15:00:00 |
| 2 | monitor | 2 | tersedia | 2024-12-04 15:00:00 |
| 3 | keyboard | 7 | tersedia | 2024-12-05 15:00:00 |
| 4 | kabel hdmi | 10 | tersedia | 2024-12-06 15:00:00 |
+-----+-----+-----+-----+-----+
4 rows in set (0.001 sec)

MariaDB [peminjaman_barang_it]> DELIMITER ;

```

D. Data Query Language

1. Sorting

Menampilkan data user berdasarkan urutan username secara ascending(A-Z)

```
MariaDB [peminjaman_barang_it]> SELECT * FROM user ORDER BY username ASC;
+-----+-----+-----+-----+-----+
| id_user | username | email | password | akses |
+-----+-----+-----+-----+-----+
| 3 | al_hauna | al_hauna@company.com | al_hauna123 | admin |
| 1 | asyafa | asyafa@company.com | asyafa123 | admin |
| 2 | ditra | ditra@company.com | ditra123 | user |
| 4 | jamal | jamal@company.com | jamal123 | user |
+-----+-----+-----+-----+-----+
4 rows in set (0.001 sec)

MariaDB [peminjaman_barang_it]>
```

Menampilkan data meminjam berdasarkan urutan pengembalian secara descending(Z-A)

```
MariaDB [peminjaman_barang_it]> SELECT * FROM meminjam ORDER BY tanggal_kembali DESC;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_peminjaman | username | nama | nama_barang | jml_barang | tanggal_pinjam | tanggal_kembali | status_pengembalian |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 2 | jamal | Jamal Udin | monitor | 1 | 2024-12-08 14:00:00 | 2024-12-09 14:00:00 | belum dikembalikan |
| 1 | ditra | Ditra James | proyektor | 1 | 2024-12-07 14:00:00 | 2024-12-08 14:00:00 | belum dikembalikan |
+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.000 sec)
```

2. Searching

Melakukan searching data user yang usernamenya diawali dengan huruf a.

```
MariaDB [peminjaman_barang_it]> SELECT * FROM user WHERE username LIKE 'a%';
+-----+-----+-----+-----+-----+
| id_user | username | email | password | akses |
+-----+-----+-----+-----+-----+
| 3 | al_hauna | al_hauna@company.com | al_hauna123 | admin |
| 1 | asyafa | asyafa@company.com | asyafa123 | admin |
+-----+-----+-----+-----+-----+
2 rows in set (0.001 sec)
```

Melakukan searching barang it yang jumlah barang nya ada 10 atau nama barangnya Proyektor.

```
MariaDB [peminjaman_barang_it]> SELECT * FROM barang_it WHERE jumlah_barang = 10 OR nama_barang = 'proyektor';
+-----+-----+-----+-----+-----+
| id_barang | nama_barang | jumlah_barang | status_barang | tanggal_input |
+-----+-----+-----+-----+-----+
| 1 | proyektor | 5 | tersedia | 2024-12-03 15:00:00 |
| 4 | kabel hdmi | 10 | tersedia | 2024-12-06 15:00:00 |
+-----+-----+-----+-----+-----+
2 rows in set (0.001 sec)
```

E. Query optimization

1. Pengoptimasian penulisan query dengan menggunakan kolom yang sebelumnya sudah diberi index pada tabel user.

```
MariaDB [peminjaman_barang_it]> SELECT * from user WHERE username LIKE '%dit%';
+-----+-----+-----+-----+-----+
| id_user | username | email | password | akses |
+-----+-----+-----+-----+-----+
| 2 | ditra | ditra@company.com | ditra123 | user |
+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)
```

2. Pengoptimasian penulisan query dengan menggunakan kolom yang sebelumnya sudah diberi index pada tabel barang_it.

```
MariaDB [peminjaman_barang_it]> SELECT * from barang_it WHERE nama_barang LIKE '%kabel%';
+-----+-----+-----+-----+-----+
| id_barang | nama_barang | jumlah_barang | status_barang | tanggal_input |
+-----+-----+-----+-----+-----+
| 4 | kabel hdmi | 10 | tersedia | 2024-12-06 15:00:00 |
+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)
```

3. Pengoptimasian penulisan query dengan menggunakan kolom yang sebelumnya sudah diberi index pada tabel karyawan.

```
MariaDB [peminjaman_barang_it]> SELECT * from karyawan WHERE pekerjaan LIKE '%seker%';
+-----+-----+-----+-----+-----+-----+
| id_karyawan | nama | pekerjaan | divisi | jenis_kelamin | no_telp |
+-----+-----+-----+-----+-----+-----+
| 3 | Al Hauna john | sekretaris | marketing | L | 081227227912 |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)
```

Tampilan Website

1. Tampilan Login

Login

Email

Masukkan email

Password

Masukkan password

Login

Kembali ke Home

2. Tampilan register

Form Registrasi Karyawan

Nama Lengkap

Masukkan nama lengkap

Pekerjaan

Masukkan pekerjaan

Divisi

Masukkan divisi

Jenis Kelamin

Pilih jenis kelamin

Nomor Telepon

Masukkan nomor telepon

Daftar

Kembali ke Home

Form Registrasi User

Username

Email

Password

Akses

3. Tampilan Beranda

Website Peminjaman Barang IT

Barang IT

Peminjaman Barang IT

Website ini adalah website Peminjaman Barang IT yang dikelola oleh IT Support sebuah perusahaan.

4. Tampilan Admin

Website Peminjaman Barang IT

Halo, asyafal (Akses: Admin)

Barang IT

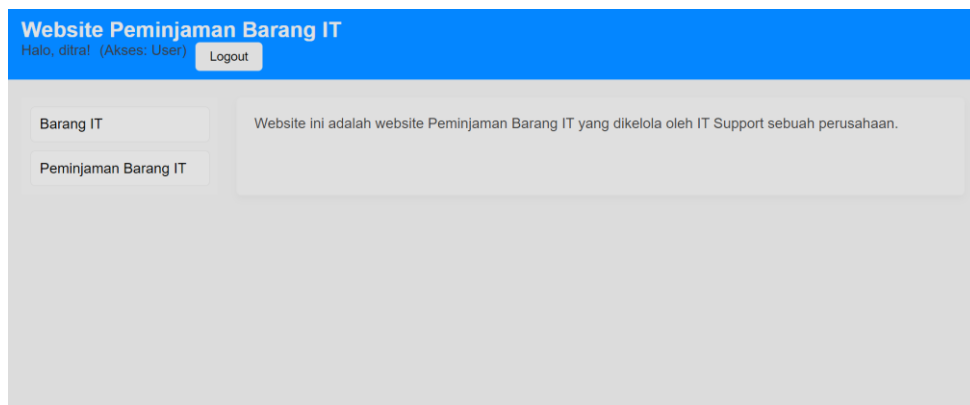
List User

List Karyawan

Peminjaman Barang IT

Website ini adalah website Peminjaman Barang IT yang dikelola oleh IT Support sebuah perusahaan.

5. Tampilan User



6. Tampilan data barang it user

database terkoneksi [Kembali ke Home](#)

Data Barang IT

ID Barang	Nama Barang	Jumlah Barang	Status Barang	Tanggal Input
1	proyektor	5	Tersedia	2024-12-03 15:00:00
2	monitor	2	Tersedia	2024-12-04 15:00:00
9	keyboard	7	Tersedia	2024-12-05 15:00:00
10	kabel HDMI	7	Dipinjam	2024-12-06 15:00:00

7. Tampilan data barang it admin

database terkoneksi [Kembali ke Home](#)

Data Barang IT

[Tambah barang](#)

ID Barang	Nama Barang	Jumlah Barang	Status Barang	Tanggal Input	Aksi
1	proyektor	5	Tersedia	2024-12-03 15:00:00	Edit Hapus
2	monitor	2	Tersedia	2024-12-04 15:00:00	Edit Hapus
9	keyboard	7	Tersedia	2024-12-05 15:00:00	Edit Hapus
10	kabel HDMI	7	Dipinjam	2024-12-06 15:00:00	Edit Hapus

Tambah Barang

Nama Barang

Masukkan nama barang

Jumlah Barang

Masukkan jumlah barang

Status Barang

Pilih status

Tambah Barang

database terkoneksi

Edit Barang

Nama Barang:

proyektor

Jumlah Barang:

5

Status Barang:

Tersedia

Update

8. Tampilan data peminjaman user

database terkoneksi

Data Peminjaman

Tambah peminjaman

ID Peminjaman	Username	Nama	Nama Barang	Jumlah Barang	Tanggal Pinjam	Tanggal Kembali	Status Pengembalian
1	ditra	Ditra James	proyektor	1	2024-12-07 14:00:00	2024-12-08 14:00:00	Belum dikembalikan
2	jamal	Jamal Udin	monitor	1	2024-12-08 14:00:00	2024-12-09 14:00:00	Belum dikembalikan

Tambah Peminjaman

Username

Pilih Username

Nama

Pilih Nama

Nama Barang

Pilih Barang

Jumlah Barang

Masukkan jumlah barang

Tanggal Pinjam

mm/dd/yyyy --:-- --

Tanggal Kembali

mm/dd/yyyy --:-- --

Status Pengembalian

Pilih Status

Tambah Peminjaman

9. Tampilan data peminjaman Admin

database terkoneksi

Data Peminjaman

Tambah peminjaman

ID Peminjaman	Username	Nama	Nama Barang	Jumlah Barang	Tanggal Pinjam	Tanggal Kembali	Status Pengembalian	Aksi
1	ditra	Ditra James	proyektor	1	2024-12-07 14:00:00	2024-12-08 14:00:00	Belum dikembalikan	<div>Edit</div> <div>Hapus</div>
2	jamal	Jamal Udin	monitor	1	2024-12-08 14:00:00	2024-12-09 14:00:00	Belum dikembalikan	<div>Edit</div> <div>Hapus</div>

Tambah Peminjaman

Username

Pilih Username

Nama

Pilih Nama

Nama Barang

Pilih Barang

Jumlah Barang

Masukkan jumlah barang

Tanggal Pinjam

mm/dd/yyyy --:-- --

Tanggal Kembali

mm/dd/yyyy --:-- --

Status Pengembalian

Pilih Status

Tambah Peminjaman

Edit Peminjaman

Username

ditra

Nama

aji barang

Nama Barang

proyektor

Jumlah Barang

1

Tanggal Pinjam

12/07/2024 02:00 PM

Tanggal Kembali

12/08/2024 02:00 PM

Status Pengembalian

Belum Dikembalikan

Simpan Perubahan

10. Tampilan admin data user

database terkoneksi | [Kembali ke Home](#)

Tambah user

ID User	Username	Email	Akses	Aksi
1	asyafa	asyafa@company.com	Admin	Edit Hapus
2	ditra	ditra@company.com	User	Edit Hapus
3	al_hauna	al_hauna@company.com	Admin	Edit Hapus
4	jamal	jamal@company.com	User	Edit Hapus
6	salsa	salsa@company.com	User	Edit Hapus
8	aji	aji@company.com	Admin	Edit Hapus
13	gafasa	gafasa@company.com	User	Edit Hapus

Tambah User

Username

Email

Password

Hak Akses

Pilih Hak Akses ▾

Tambah User

Edit User

Username

Email

Hak Akses

Admin ▾

Simpan Perubahan

11. Tampilan daftar karyawan

database terkoneksi

Kembali ke Home

Daftar Karyawan

Tambah Karyawan

ID Karyawan	Nama	Pekerjaan	Divisi	Jenis Kelamin	No. Telepon	Aksi	
1	Asyafa arthur	pengelola barang it	it support	Laki-laki	081227227910	Edit	Hapus
2	Ditra james	asisten manajer	manajemen	Perempuan	081227227911	Edit	Hapus
3	Al Hauna john	sekertaris	marketing	Laki-laki	081227227912	Edit	Hapus
4	Jamal Udin	asisten manajer	advisor	Laki-laki	081227227913	Edit	Hapus
5	gusdur al majid	cleaning service	kebersihan	Laki-laki	098765432109	Edit	Hapus
6	naruto	direktur	IT	Perempuan	172648364857	Edit	Hapus

Tambah Karyawan

Nama

Masukkan nama karyawan

Pekerjaan

Masukkan pekerjaan karyawan

Divisi

Masukkan divisi karyawan

Jenis Kelamin

Pilih Jenis Kelamin



No. Telepon

Masukkan no telepon karyawan

Tambah Karyawan

Edit Karyawan

Nama

Asyafa arthur

Pekerjaan

pengelola barang it

Divisi

it support

Jenis Kelamin

Laki-laki



No. Telepon

081227227910

Simpan Perubahan