

Konsultasi 3

Laporan Fitur Aplikasi Inventaris Barang

1. Deskripsi Fitur

Aplikasi ini bertujuan untuk membantu sekolah dalam mengelola inventaris barang dengan fitur utama sebagai berikut:

a. Manajemen Data Barang

Fitur ini memungkinkan pencatatan barang yang dimiliki sekolah, termasuk informasi tentang nama, jumlah, serta mengubah dan menghapus data barang.

b. Peminjaman dan Pengelolaan Barang

Fitur ini digunakan untuk mencatat peminjaman barang oleh siswa atau guru, memastikan bahwa barang dapat dilacak, dan mengetahui siapa yang terakhir menggunakan barang tersebut.

c. Pendataan Ruang Penyimpanan Barang

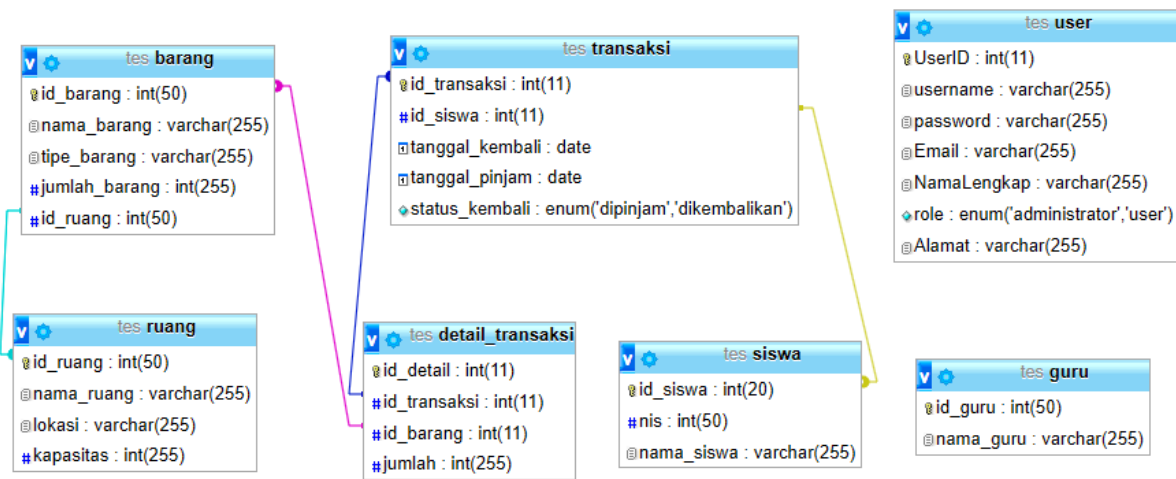
Fitur ini membantu dalam mencatat lokasi penyimpanan barang di sekolah agar lebih terorganisir dan mudah ditemukan.

2. Database

a. Struktur Database

Database ini digunakan untuk mencatat informasi inventaris dan peminjaman barang di lingkungan sekolah. Data yang disimpan mencakup informasi barang, siswa, transaksi peminjaman, dan ruang penyimpanan.

Diagram Relasi Antar Tabel



b. Penjelasan Tabel

1. **barang**

Menyimpan data semua barang yang tersedia untuk dipinjam.

- **id_barang** : ID unik barang
- **nama_barang** : Nama barang
- **tipe_barang** : Jenis barang
- **jumlah_barang** : Jumlah stok barang
- **id_ruang** : Lokasi penyimpanan (relasi ke tabel **ruang**)

2. **ruang**

Menyimpan informasi tentang ruang penyimpanan barang.

- **id_ruang** : ID unik ruang
- **nama_ruang** : Nama ruang
- **lokasi** : Lokasi fisik ruang
- **kapasitas** : Kapasitas maksimal ruang

3. **siswa**

Menyimpan data siswa yang bisa meminjam barang.

- **id_siswa** : ID unik siswa
- **nis** : Nomor induk siswa
- **nama_siswa** : Nama lengkap siswa

4. **transaksi**

Mencatat aktivitas peminjaman barang oleh siswa.

- **id_transaksi** : ID transaksi
- **id_siswa** : Siswa yang meminjam (relasi ke tabel **siswa**)
- **tanggal_pinjam** : Tanggal mulai pinjam
- **tanggal_kembali** : Tanggal rencana pengembalian
- **status_kembali** : Status pengembalian (dipinjam/dikembalikan)

5. **detail_transaksi**

Menyimpan rincian barang apa saja yang dipinjam dalam satu transaksi.

- **id_detail** : ID detail transaksi

- `id_transaksi` : Relasi ke tabel `transaksi`
- `id_barang` : Barang yang dipinjam (relasi ke `barang`)
- `jumlah` : Jumlah barang yang dipinjam

6. user

Menyimpan data pengguna aplikasi (admin dan user biasa).

- `UserID` : ID unik pengguna
- `username` , `password` : Login pengguna
- `Email` , `NamaLengkap` , `Alamat` : Informasi pengguna
- `role` : Hak akses (`administrator` atau `user`)

7. guru

Menyimpan data guru yang terkait dalam sistem.

- `id_guru` : ID unik guru
- `nama_guru` : Nama guru

3. Rincian Fitur

Fitur: Manejemen Data Barang

a. Screenshot Fitur

Inventaris Sekolah

Kembali

Manajemen Data Barang

Nama Barang

Tipe Barang

Jumlah Barang

-- Pilih Ruang Penyimpanan --

Simpan Barang

Tampilan Data

Data Barang

Nama Barang	Tipe	Jumlah	Lokasi Penyimpanan	Aksi	
Kipas Angin	Elektronik	38	Lab Komputer RPL	Edit	Hapus
AC	Elektronik	10	Lab Komputer RPL	Edit	Hapus
Meja	Peralatan Sekolah	3	Lab Komputer PS	Edit	Hapus
kursi	Peralatan Sekolah	23	Lab Komputer AK	Edit	Hapus
Pensil	Peralatan Sekolah	30	Lab Komputer AP	Edit	Hapus
Sapu	Alat Membersihkan	2	Lab Komputer RPL	Edit	Hapus

b. Kode Program

Front-End (HTML dan PHP)

Gambar 1 (*manajemen.php*) :

```
<!DOCTYPE html>
<html lang="id">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Manajemen Data Barang</title>
  <link rel="stylesheet" href="style4.css">
  <script defer src="script.js"></script>
</head>
<body>
  <nav class="navbar">
    <div class="navbar-container">
      <div class="navbar-brand">
        <h2>Inventaris Sekolah</h2>
      </div>
      <ul class="navbar-menu">
        <li><a href="index.php">Kembali</a></li>
      </ul>
    </div>
  </nav>

  <div class="container">
    <h2>Manajemen Data Barang</h2>
    <form id="barangForm" method="post">
      <input type="hidden" id="editId" name="editId">
      <input type="text" id="namaBarang" name="namaBarang" placeholder="Nama
Barang" required>
      <input type="text" id="tipeBarang" name="tipeBarang" placeholder="Tipe
Barang" required>
```

```

        <input type="number" id="jumlahBarang" name="jumlahBarang"
placeholder="Jumlah Barang" required>

        <select name="id_ruang" required>
            <option value="">-- Pilih Ruang Penyimpanan --</option>
            <?php foreach ($ruangList as $ruang): ?>
                <option value="<?= $ruang['id_ruang'] ?>"><?=
htmlspecialchars($ruang['nama_ruang']) ?></option>
            <?php endforeach; ?>
        </select>

        <button type="submit">Simpan Barang</button>
        <button type="button"
onclick="window.location.href='gabungan3.php'">Tampilan Data</button>
    </form>
</div>
</body>
</html>

```

Gambar 2 (***gabungan3.php***):

```

<nav class="navbar">
    <div class="navbar-container">
        <div class="navbar-brand">
            <h2>Inventaris Sekolah</h2>
        </div>
        <ul class="navbar-menu">
            <li><a href="index.php">Kembali</a></li>
        </ul>
    </div>
</nav>

<div class="container">
    <h2>Data Barang</h2>
    <input type="text" id="searchBarang" onkeyup="searchTable('searchBarang',
'tableBarang')" placeholder="Cari Barang...">
    <table id="tableBarang" border="1">
        <tr>
            <th>Nama Barang</th>
            <th>Tipe</th>
            <th>Jumlah</th>
            <th>Lokasi Penyimpanan</th>
            <th>Aksi</th>
        </tr>
        <?php
            $query = "SELECT barang.id_barang, barang.nama_barang, barang.tipe_barang,
barang.jumlah_barang, ruang.nama_ruang
                    FROM barang
                    LEFT JOIN ruang ON barang.id_ruang = ruang.id_ruang";

            $result = $conn->query($query);

            if (!$result) {

```

```

        die("Error pada query: " . mysqli_error($conn));
    }

    if ($result->num_rows > 0) {
        while ($row = $result->fetch_assoc()) {
            $lokasi = !empty($row['nama_ruang']) ? $row['nama_ruang'] : "Tidak
diketahui";

            echo "<tr>
                <td>{$row['nama_barang']}</td>
                <td>{$row['tipe_barang']}</td>
                <td>{$row['jumlah_barang']}</td>
                <td>{$lokasi}</td>
                <td>
                    <a href='edit_barang.php?id=
{$row['id_barang']}'>Edit</a> |
                    <a href='hapus_barang.php?id={$row['id_barang']}'
onclick='return confirm(\"Yakin ingin menghapus?\")'>Hapus</a>
                </td>
            </tr>";
        }
    } else {
        echo "<tr><td colspan='5'>Tidak ada data barang.</td></tr>";
    }
    ?>
</table>
</div>

```

Back-End (PHP - Proses Manajemen Data)

1. tambah_barang.php

```

<?php
session_start();
if ($_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
    exit();
}

include 'koneksi.php';

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $nama_barang = $_POST['nama_barang'];
    $jumlah_barang = $_POST['jumlah_barang'];

    $sql = "INSERT INTO barang (nama_barang, jumlah_barang) VALUES ('$nama_barang',
'$jumlah_barang')";

    if ($koneksi->query($sql) === TRUE) {
        echo "Barang berhasil ditambahkan!";
        header("Location: tampil_barang.php");
    } else {
        echo "Error: " . $sql . "<br>" . $koneksi->error;
    }
}

```

```

    }
}
?>

<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Tambah Barang</title>
</head>
<body>
    <h2>Tambah Barang</h2>
    <form method="POST">
        <label>Nama Barang:</label>
        <input type="text" name="nama_barang" required><br>
        <label>Jumlah:</label>
        <input type="number" name="jumlah_barang" required><br>
        <input type="submit" value="Simpan">
    </form>
</body>
</html>

```

2. tampil_barang.php

```

<?php
session_start();
if ($_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
    exit();
}

include 'koneksi.php';

$sql = "SELECT barang.id_barang, barang.nama_barang, barang.jumlah_barang
        FROM barang";
$result = $koneksi->query($sql);
?>

<!DOCTYPE html>
<html lang="id">
<head>
<meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Data Barang</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <h2>Daftar Barang</h2>
    <table border="1">
        <tr>
            <th>ID Barang</th>
            <th>Nama Barang</th>

```

```

        <th>Jumlah</th>
        <th>Lokasi</th>
        <th>Aksi</th>
    </tr>
    <?php while ($row = $result->fetch_assoc()) { ?>
    <tr>
        <td><?php echo $row['id_barang']; ?></td>
        <td><?php echo $row['nama_barang']; ?></td>
        <td><?php echo $row['jumlah_barang']; ?></td>
        <td>
            <a href="edit_barang.php?id=<?php echo $row['id_barang']; ?
">Edit</a> |
            <a href="hapus_barang.php?id=<?php echo $row['id_barang']; ?>"
onclick="return confirm('Hapus barang ini?')">Hapus</a>
        </td>
    </tr>
    <?php } ?>
</table>
</body>
</html>

```

3. edit_barang.php

```

<?php
error_reporting(E_ALL);
ini_set('display_errors', 1);

session_start();
if (!isset($_SESSION['role']) || $_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
    exit();
}

include 'koneksi3.php';

if ($conn->connect_error) {
    die("Koneksi database gagal: " . $conn->connect_error);
}

if (!isset($_GET['id']) || !is_numeric($_GET['id'])) {
    die("ID barang tidak valid.");
}

$id = intval($_GET['id']);

$result = $conn->query("SHOW COLUMNS FROM barang LIKE 'id_barang'");
if ($result->num_rows == 0) {
    die("Error: Kolom 'id_barang' tidak ditemukan di tabel barang.");
}

$stmt = $conn->prepare("SELECT * FROM barang WHERE id_barang = ?");
if (!$stmt) {
    die("Query Error: " . $conn->error);
}

```



```

}

$stmt->bind_param("i", $id);
$stmt->execute();
$result = $stmt->get_result();
$row = $result->fetch_assoc();

if (!$row) {
    die("Data barang tidak ditemukan.");
}

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $nama_barang = trim($_POST['nama_barang']);
    $jumlah_barang = intval($_POST['jumlah_barang']);

    if (empty($nama_barang) || $jumlah_barang <= 0) {
        echo "<script>alert('Harap isi semua data dengan benar!');</script>";
    } else {
        $stmt = $conn->prepare("UPDATE barang SET nama_barang=?, jumlah_barang=?
WHERE id_barang=?");
        $stmt->bind_param("sii", $nama_barang, $jumlah_barang, $id);

        if ($stmt->execute()) {
            echo "<script>alert('Data berhasil diperbarui!');
window.location.href='gabungan3.php';</script>";
            exit();
        } else {
            echo "Error updating record: " . $stmt->error;
        }
    }
}
?>

```

```

<!DOCTYPE html>
<html lang="id">
<head>
    <title>Edit Barang</title>
    <link rel="stylesheet" href="style_edit.css">
</head>
<body>
    <form method="POST">
        <label>Nama Barang:</label>
        <input type="text" name="nama_barang" value="<?php echo
htmlspecialchars($row['nama_barang']); ?>" required><br>

        <label>Jumlah:</label>
        <input type="number" name="jumlah_barang" value="<?php echo
$row['jumlah_barang']; ?>" required><br>

        <button type="submit"><a href="gabungan3.php">Update</a></button>
    </form>
</body>
</html>

```

4. hapus_barang.php

```
<?php
session_start();
if (!isset($_SESSION['role']) || $_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
    exit();
}

include 'koneksi3.php';

if (!isset($_GET['id']) || !is_numeric($_GET['id'])) {
    die("ID tidak valid.");
}

$id = intval($_GET['id']);

$stmt = $conn->prepare("DELETE FROM barang WHERE id_barang = ?");
if ($stmt) {
    $stmt->bind_param("i", $id);
    if ($stmt->execute()) {
        echo "<script>alert('Data berhasil dihapus!');
window.location.href='gabungan3.php';</script>";
    } else {
        echo "Error deleting record: " . $stmt->error;
    }
} else {
    echo "Query Error: " . $conn->error;
}
?>
```

c. Analisis Kode

Fitur ini terdiri dari beberapa file untuk mengelola data barang, mulai dari menambah, menampilkan, mengedit, hingga menghapus data barang.

1. manajemen.php

- Menyediakan **form input barang** dengan kolom: nama, tipe, jumlah, dan pilihan ruang penyimpanan.
- Data dikirim melalui metode **POST** ke backend untuk diproses (terintegrasi melalui script.js / backend tambahan).
- Terdapat tombol untuk melihat data barang (gabungan3.php).
- Menggunakan perulangan **foreach** untuk menampilkan pilihan ruang dari variabel PHP `$ruangList`.

2. gabungan3.php

- Menampilkan **data barang** dalam bentuk tabel.
- Mengambil data dari **tabel** barang **dan** ruang dengan perintah SQL **LEFT JOIN**.
- Menyediakan fitur **pencarian barang** dan tombol aksi: **Edit** dan **Hapus**.
- Jika nama ruang kosong, ditampilkan sebagai "Tidak diketahui".

3. **tambah_barang.php**

- Hanya bisa diakses oleh pengguna dengan **role administrator**.
- Menyimpan data barang ke database menggunakan perintah `INSERT INTO barang`.
- Setelah data berhasil ditambah, pengguna diarahkan ke `tampil_barang.php`.
- Tidak melibatkan pemilihan ruang penyimpanan (form-nya sederhana).

4. **tampil_barang.php**

- Menampilkan data barang dalam bentuk tabel: ID, nama, jumlah, aksi.
- Mengambil data dari database, namun **query belum lengkap (kurang tanda kutip penutup)**.
- Setiap baris barang dilengkapi dengan tombol **Edit** dan **Hapus**.

5. **edit_barang.php**

- Mengambil data barang berdasarkan ID dari URL.
- Menampilkan form untuk mengubah nama dan jumlah barang.
- Menggunakan **prepared statement** untuk update data (`UPDATE barang SET...`).
- Menampilkan pesan sukses dan kembali ke halaman tampilan barang (`gabungan3.php`).

6. **hapus_barang.php**

- Menghapus barang berdasarkan ID yang dikirim lewat URL.
- Setelah berhasil, menampilkan alert dan kembali ke halaman tampilan barang.
- Menggunakan **prepared statement** untuk menghindari query berbahaya.

d. Peran Fitur dalam Database

Fitur **Manajemen Data Barang** ini memiliki peran penting dalam menjaga data barang tetap **terstruktur dan mudah dikelola**. Berikut perannya:

- **Tambah Barang:**

Data barang baru dimasukkan ke tabel `barang`, meliputi nama, tipe, jumlah, dan lokasi penyimpanan (dalam versi lengkap).

- **Tampilkan Barang:**

Menampilkan seluruh isi tabel `barang`, serta menggabungkannya dengan tabel `ruang` untuk mengetahui lokasi penyimpanan.

- **Edit Barang:**

Memungkinkan pengguna untuk memperbarui informasi barang jika ada perubahan, langsung ke tabel `barang`.

- **Hapus Barang:**

Data barang yang sudah tidak digunakan bisa dihapus dari database agar data tetap bersih dan relevan.

Fitur ini mendukung proses **CRUD (Create, Read, Update, Delete)** pada tabel `barang`, dan secara tidak langsung juga mengakses data dari tabel `ruang`. Hal ini membantu pengguna dalam **mengelola stok dan lokasi penyimpanan barang secara efisien**, terutama dalam lingkungan sekolah.

Fitur: Peminjaman dan Pengelolaan Barang

a. Screenshot Fitur

Inventaris Sekolah

Kembali

Nama Siswa:

-- Pilih Siswa --

Tanggal Pinjam:

mm/dd/yyyy

Tanggal Kembali:

mm/dd/yyyy

Barang:

-- Pilih Barang --

Jumlah:

1

Pinjam

Tampilkan Data

Inventaris Sekolah

Kembali

Data Peminjaman

Cari Peminjaman...

Nama Siswa	Tanggal Pinjam	Tanggal Kembali	Barang	Jumlah	Aksi	
Githania Agatha	2025-03-04	2025-03-08	Kipas Angin	3	Edit	Hapus
Adrian	2025-03-01	2025-03-04	Kipas Angin	2	Edit	Hapus
Kristian	2025-03-14	2025-03-15	Meja	2	Edit	Hapus

b. Kode Program

Front-End (HTML dan PHP)

Gambar 1 (*pinjam_barang.php*) :

```
<?php
session_start();
if ($_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
    exit();
}

include 'koneksi.php';

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $id_siswa = $_POST['id_siswa'] ?? null;
    $tanggal_pinjam = $_POST['tanggal_pinjam'] ?? null;
    $tanggal_kembali = $_POST['tanggal_kembali'] ?? null;
```

```

$id_barang = $_POST['barang_id'] ?? null;
$jumlah = $_POST['jumlah'] ?? null;

if ($id_siswa && $tanggal_pinjam && $tanggal_kembali && $id_barang && $jumlah)
{
    $sql_transaksi = "INSERT INTO transaksi (id_siswa, tanggal_pinjam,
tanggal_kembali) VALUES ('$id_siswa', '$tanggal_pinjam', '$tanggal_kembali')";
    if ($koneksi->query($sql_transaksi)) {
        $id_transaksi = $koneksi->insert_id;
        $sql_detail = "INSERT INTO detail_transaksi (id_transaksi, id_barang,
jumlah) VALUES ('$id_transaksi', '$id_barang', '$jumlah')";
        $koneksi->query($sql_detail);
        $koneksi->query("UPDATE barang SET jumlah_barang = jumlah_barang -
$jumlah WHERE id_barang = '$id_barang'");
    }
}
?>

<!DOCTYPE html>
<html lang="id">
<head>
    <title>Peminjaman Barang</title>
    <link rel="stylesheet" href="style2.css">
    <style>
        .btn { padding: 5px 10px; border: none; cursor: pointer; border-radius:
5px; display: inline-block; text-align: center; }
        .btn-detail { background: blue; color: white; text-decoration: none; }
        .btn-edit { background: orange; color: white; }
        .btn-hapus { background: red; color: white; }
    </style>
    <script>
        function tampilkanDetail(siswa, tanggalPinjam, tanggalKembali, barang,
jumlah, status) {
            alert(`Siswa: ${siswa}\nTanggal Pinjam: ${tanggalPinjam}\nTanggal
Kembali: ${tanggalKembali}\nBarang: ${barang}\nJumlah: ${jumlah}\nStatus:
${status}`);
        }
    </script>
</head>
<body>
    <nav class="navbar">
        <div class="navbar-container">
            <div class="navbar-brand">
                <h2>Inventaris Sekolah</h2>
            </div>
            <ul class="navbar-menu">
                <li><a href="index.php">Kembali</a></li>
            </ul>
        </div>
    </nav>

    <form method="POST">

```

```

<label>Nama Siswa:</label>
<select name="id_siswa" required>
    <option value="">-- Pilih Siswa --</option>
    <?php
        $result = $koneksi->query("SELECT * FROM siswa");
        while ($row = $result->fetch_assoc()) {
            echo "<option value='{ $row['id_siswa']}'>{$row['nama_siswa']}
</option>";
        }
    ?>
</select><br>
<label>Tanggal Pinjam:</label>
<input type="date" name="tanggal_pinjam" required><br>
<label>Tanggal Kembali:</label>
<input type="date" name="tanggal_kembali" required><br>

<label>Barang:</label>
<select name="barang_id" required>
    <option value="">-- Pilih Barang --</option>
    <?php
        $result = $koneksi->query("SELECT * FROM barang WHERE jumlah_barang >
0");
        while ($row = $result->fetch_assoc()) {
            echo "<option value='{ $row['id_barang']}'>{$row['nama_barang']}
(Stok: { $row['jumlah_barang']})</option>";
        }
    ?>
</select><br>

<label>Jumlah:</label>
<input type="number" name="jumlah" min="1" max="100" value="1" required>
<br>

<input type="submit" value="Pinjam" class="btn btn-detail">
<a href="gabungan4.php" class="btn btn-detail">Tampilan Data</a>
</form>

</body>
</html>

```

Gambar 2 (***gabungan4.php***) :

```

<?php
session_start();

include 'koneksi3.php';
if (!$conn) {
    die("Koneksi database gagal: " . mysqli_connect_error());
}

if (!isset($_SESSION['role']) || $_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
    exit();
}

```



```

        <th>Aksi</th>
    </tr>
    <?php
        $query = "SELECT transaksi.id_transaksi, siswa.nama_siswa,
transaksi.tanggal_pinjam, transaksi.tanggal_kembali, barang.nama_barang,
detail_transaksi.jumlah
                FROM transaksi
                JOIN siswa ON transaksi.id_siswa = siswa.id_siswa
                JOIN detail_transaksi ON transaksi.id_transaksi =
detail_transaksi.id_transaksi
                JOIN barang ON detail_transaksi.id_barang = barang.id_barang";

        $result = $conn->query($query);

        if ($result->num_rows > 0) {
            while ($row = $result->fetch_assoc()) {
                echo "<tr>
                    <td>{$row['nama_siswa']}

```

Back-End (PHP - Proses Peminjaman)

1. pinjam_barang.php (*terdapat dibagian atas pinjam_barang.php*)

```

<?php
session_start();
if ($_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
    exit();
}

include 'koneksi.php';

```



```

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $id_siswa = $_POST['id_siswa'] ?? null;
    $tanggal_pinjam = $_POST['tanggal_pinjam'] ?? null;
    $tanggal_kembali = $_POST['tanggal_kembali'] ?? null;
    $id_barang = $_POST['barang_id'] ?? null;
    $jumlah = $_POST['jumlah'] ?? null;

    if ($id_siswa && $tanggal_pinjam && $tanggal_kembali && $id_barang && $jumlah)
    {
        $sql_transaksi = "INSERT INTO transaksi (id_siswa, tanggal_pinjam,
tanggal_kembali) VALUES ('$id_siswa', '$tanggal_pinjam', '$tanggal_kembali')";
        if ($koneksi->query($sql_transaksi)) {
            $id_transaksi = $koneksi->insert_id;
            $sql_detail = "INSERT INTO detail_transaksi (id_transaksi, id_barang,
jumlah) VALUES ('$id_transaksi', '$id_barang', '$jumlah')";
            $koneksi->query($sql_detail);
            $koneksi->query("UPDATE barang SET jumlah_barang = jumlah_barang -
$jumlah WHERE id_barang = '$id_barang'");
        }
    }
}
?>

```

2. edit_peminjaman.php

```

<?php
session_start();
if ($_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
    exit();
}

include 'koneksi.php';

if (!isset($_GET['id'])) {
    die("ID Transaksi tidak ditemukan!");
}

$id_transaksi = $_GET['id'];

$query = "SELECT transaksi.*, detail_transaksi.id_barang, detail_transaksi.jumlah
FROM transaksi
JOIN detail_transaksi ON transaksi.id_transaksi =
detail_transaksi.id_transaksi
WHERE transaksi.id_transaksi = '$id_transaksi'";
$result = $koneksi->query($query);
$transaksi = $result->fetch_assoc();

if (!$transaksi) {
    die("Data tidak ditemukan!");
}

```

```

$id_siswa = $transaksi['id_siswa'];
$tanggal_pinjam = $transaksi['tanggal_pinjam'];
$tanggal_kembali = $transaksi['tanggal_kembali'];
$id_barang = $transaksi['id_barang'];
$jumlah_sebelumnya = $transaksi['jumlah'];

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $id_siswa = $_POST['id_siswa'];
    $tanggal_pinjam = $_POST['tanggal_pinjam'];
    $tanggal_kembali = $_POST['tanggal_kembali'];
    $id_barang_baru = $_POST['id_barang'];
    $jumlah_baru = $_POST['jumlah'];

    $koneksi->begin_transaction();

    try {
        $koneksi->query("UPDATE barang SET jumlah_barang = jumlah_barang +
$jumlah_sebelumnya WHERE id_barang = '$id_barang'");

        $sql_transaksi = "UPDATE transaksi
                        SET id_siswa='$id_siswa',
tanggal_pinjam='$tanggal_pinjam', tanggal_kembali='$tanggal_kembali'
                        WHERE id_transaksi='$id_transaksi'";
        $koneksi->query($sql_transaksi);

        $sql_detail = "UPDATE detail_transaksi
                        SET id_barang='$id_barang_baru', jumlah='$jumlah_baru'
                        WHERE id_transaksi='$id_transaksi'";
        $koneksi->query($sql_detail);

        $koneksi->query("UPDATE barang SET jumlah_barang = jumlah_barang -
$jumlah_baru WHERE id_barang = '$id_barang_baru'");

        $koneksi->commit();

        $id_barang = $id_barang_baru;
        $jumlah_sebelumnya = $jumlah_baru;

        echo "<p style='color: green;'>Data berhasil diperbarui!</p>";
        header("Location: gabungan4.php");
        exit();

    } catch (Exception $e) {
        echo "<p style='color:red;'>Terjadi kesalahan: " . $e->getMessage() . "
</p>";
        $koneksi->rollback();
    }
}
?>

<!DOCTYPE html>
<html lang="id">
<head>

```

```

<title>Edit Peminjaman</title>
<link rel="stylesheet" href="style_edit2.css">
</head>
<body>
    <form method="POST">
        <label>Nama Siswa:</label>
        <select name="id_siswa" required>
            <?php
                $siswa_result = $koneksi->query("SELECT * FROM siswa");
                while ($row = $siswa_result->fetch_assoc()) {
                    $selected = ($row['id_siswa'] == $id_siswa) ? "selected" : "";
                    echo "<option value='{ $row['id_siswa']}' '$selected>
{$row['nama_siswa']}'</option>";
                }
            ?>
        </select><br>

        <label>Tanggal Pinjam:</label>
        <input type="date" name="tanggal_pinjam" value="<?php echo $tanggal_pinjam;
?>" required><br>

        <label>Tanggal Kembali:</label>
        <input type="date" name="tanggal_kembali" value="<?php echo
$tanggal_kembali; ?>" required><br>

        <label>Barang:</label>
        <select name="id_barang" required>
            <?php
                $barang_result = $koneksi->query("SELECT * FROM barang");
                while ($row = $barang_result->fetch_assoc()) {
                    $selected = ($row['id_barang'] == $id_barang) ? "selected" : "";
                    echo "<option value='{ $row['id_barang']}' '$selected>
{$row['nama_barang']}' - Stok: {$row['jumlah_barang']}'</option>";
                }
            ?>
        </select><br>

        <label>Jumlah:</label>
        <input type="number" name="jumlah" min="1" max="100" value="<?php echo
$jumlah_sebelumnya; ?>" required><br>

        <input type="submit" value="Update Peminjaman">
    </form>
</body>
</html>

```

3. hapus_peminjaman.php

```

<?php
session_start();
if ($_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
}

```

```

        exit();
    }

    include 'koneksi.php';

    if (isset($_GET['id'])) {
        $id_transaksi = $_GET['id'];

        $koneksi->query("DELETE FROM detail_transaksi WHERE id_transaksi = '$id_transaksi'");

        $query = "DELETE FROM transaksi WHERE id_transaksi = '$id_transaksi'";
        if ($koneksi->query($query)) {
            echo "<script>alert('Peminjaman berhasil dihapus.');"
window.location='gabungan4.php';</script>";
        } else {
            echo "<p style='color: red;'>Gagal menghapus data!</p>";
        }
    }
?>

```

c. Analisis Kode

Fitur **Peminjaman Barang** terdiri dari beberapa file utama yang saling terintegrasi untuk menjalankan proses peminjaman, pengeditan, penghapusan, dan penampilan data peminjaman secara utuh.

1. **pinjam_barang.php**

- Halaman utama untuk **input transaksi peminjaman**.
- Form berisi pilihan siswa, tanggal pinjam & kembali, barang yang tersedia, dan jumlah yang dipinjam.
- Setelah form dikirim:
 - Data masuk ke tabel `transaksi`.
 - ID transaksi digunakan untuk menambah data ke tabel `detail_transaksi`.
 - Stok barang di tabel `barang` otomatis dikurangi.
- Hanya bisa diakses oleh pengguna yang telah login sebagai **administrator**.

2. **gabungan4.php**

- Menampilkan **data peminjaman barang** secara lengkap.
- Mengambil dan menggabungkan data dari **tabel** `transaksi`, `detail_transaksi`, `barang`, dan `siswa`.
- Menampilkan informasi: nama siswa, tanggal pinjam & kembali, nama barang, dan jumlah.
- Fitur pencarian dinamis menggunakan **JavaScript**.
- Menyediakan tombol aksi untuk **Edit** dan **Hapus** data.

3. **edit_peminjaman.php**

- Mengambil data transaksi berdasarkan `id_transaksi` dari URL.
- Menampilkan form dengan data yang sudah ada.
- Menggunakan **transaksi database** agar perubahan lebih aman.

- Proses update:
 - Stok lama barang dikembalikan terlebih dahulu.
 - Data pada `transaksi` dan `detail_transaksi` diperbarui.
 - Stok baru barang dikurangi sesuai jumlah baru.
- Setelah selesai, pengguna diarahkan kembali ke `gabungan4.php`.

4. `hapus_peminjaman.php`

- Menghapus data peminjaman berdasarkan `id_transaksi`.
- Proses dilakukan dalam dua tahap:
 - Menghapus dari `detail_transaksi`.
 - Menghapus dari `transaksi`.
- Menampilkan alert saat data berhasil dihapus.

d. Peran Fitur dalam Database

Fitur **Peminjaman Barang** berfungsi untuk mencatat, menampilkan, mengubah, dan menghapus data peminjaman yang dilakukan oleh siswa. Setiap file berperan dalam pembaruan data pada beberapa tabel penting, yaitu `transaksi`, `detail_transaksi`, `barang`, dan `siswa`.

- **`pinjam_barang.php`**
 - Menyimpan data utama ke tabel `transaksi` (data siswa & tanggal).
 - Menyimpan rincian barang ke `detail_transaksi`.
 - Mengurangi stok barang langsung dari tabel `barang`.
- **`gabungan4.php`**
 - Mengambil data dari beberapa tabel (`transaksi`, `detail_transaksi`, `barang`, `siswa`) untuk ditampilkan.
 - Mempermudah admin melihat data peminjaman secara menyeluruh.
- **`edit_peminjaman.php`**
 - Memperbarui data peminjaman.
 - Mengembalikan stok lama, lalu mengurangi stok baru berdasarkan perubahan.
 - Mengubah isi `transaksi` dan `detail_transaksi`.
- **`hapus_peminjaman.php`**
 - Menghapus data dari `detail_transaksi` dan `transaksi`.
 - **Stok barang tidak dikembalikan**, ini bisa menyebabkan data stok tidak akurat dan perlu diperbaiki di versi selanjutnya.

Fitur: Pendataan Ruang Penyimpanan Barang

a. Screenshot Fitur

Inventaris Sekolah

Kembali

Nama Ruang:

Lokasi:

Kapasitas:

Tambah Ruang

Tampilan Data

Inventaris Sekolah

Kembali

Data Ruang Penyimpanan

Lab Komputer RPL

Detail Ruangan

Nama Ruang:

Lab Komputer RPL

Lokasi:

Gedung B Lantai 1

Kapasitas:

255

Barang di Ruangan:

Kipas Angin

AC

Sapu

Edit

Hapus

b. Kode Program

Front-End (HTML dan PHP)

Gambar 1 (*list_ruang.php*) :

```
<!DOCTYPE html>
<html lang="id">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Manajemen Ruang</title>
  <link rel="stylesheet" href="style_list_ruang.css">
  <style>
    .btn {
      padding: 8px 15px;
      border: none;
      border-radius: 5px;
    }
  </style>
</head>
<body>
  <div>
    <div>
      <div>Inventaris Sekolah</div>
      <div>Kembali</div>
    </div>
    <div>
      <div>
        <div>Nama Ruang:</div>
        <div></div>
        <div>Lokasi:</div>
        <div></div>
        <div>Kapasitas:</div>
        <div></div>
        <div>Tambah Ruang</div>
        <div>Tampilan Data</div>
      </div>
    </div>
  </div>
</body>
</html>
```

```

        background-color: #3498db;
        color: white;
        cursor: pointer;
        margin-right: 10px;
        text-decoration: none;
        display: inline-block;
    }

    .btn:hover {
        background-color: #2980b9;
    }
</style>
</head>
<body>

    <nav class="navbar">
        <div class="navbar-container">
            <div class="navbar-brand">
                <h2>Inventaris Sekolah</h2>
            </div>
            <ul class="navbar-menu">
                <li><a href="index.php">Kembali</a></li>
            </ul>
        </div>
    </nav>

    <div class="container">
        <form method="POST">
            <label>Nama Ruang:</label>
            <input type="text" name="nama_ruang" required><br>

            <label>Lokasi:</label>
            <input type="text" name="lokasi" required><br>

            <label>Kapasitas:</label>
            <input type="number" name="kapasitas" min="1" required><br>

            <button type="submit" class="btn">Tambah Ruang</button>
            <a href="gabungan5.php" class="btn">Tampilan Data</a>
        </form>
    </div>

</body>
</html>

```

Gambar 2 (***gabungan5.php***) :

```

<?php
session_start();
include 'koneksi3.php';
if (!$conn) {
    die("Koneksi database gagal: " . mysqli_connect_error());
}

```

```

}
if (!isset($_SESSION['role']) || $_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
    exit();
}
?>

<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Gabungan Data Inventaris</title>
    <link rel="stylesheet" href="style_gabungan2.css">
    <script>
        function showDetails(id_ruang) {
            fetch(`get_ruang_detail.php?id=${id_ruang}`)
                .then(response => response.json())
                .then(data => {
                    document.getElementById('detailNamaRuang').innerText =
data.nama_ruang;
                    document.getElementById('detailLokasi').innerText =
data.lokasi;
                    document.getElementById('detailKapasitas').innerText =
data.kapasitas;
                    document.getElementById('detailBarang').innerHTML =
data.barang;

                    document.getElementById('detailAksi').innerHTML = `
                        <a href="edit_ruang.php?id=${id_ruang}" class="btn-
edit">Edit</a>
                        <a href="hapus_ruang.php?id=${id_ruang}" class="btn-hapus"
onclick="return confirm('Yakin ingin menghapus ruang ini?')">Hapus</a>
                    `;
                })
                .catch(error => console.error('Error:', error));
        }
    </script>
    <style>
        .btn-edit, .btn-hapus {
            display: inline-block;
            padding: 8px 16px;
            border-radius: 8px;
            text-decoration: none;
            color: white;
            font-weight: 600;
            margin-top: 10px;
            margin-right: 10px;
            transition: background 0.3s;
        }

        .btn-edit {
            background-color: #28a745;

```



```

    }

    .btn-edit:hover {
        background-color: #218838;
    }

    .btn-hapus {
        background-color: #dc3545;
    }

    .btn-hapus:hover {
        background-color: #c82333;
    }
</style>
</head>
<body>
<nav class="navbar">
    <div class="navbar-container">
        <div class="navbar-brand">
            <h2>Inventaris Sekolah</h2>
        </div>
        <ul class="navbar-menu">
            <li><a href="index.php">Kembali</a></li>
        </ul>
    </div>
</nav>

<div class="container">
    <h2>Data Ruang Penyimpanan</h2>
    <select id="ruangSelect" onchange="showDetails(this.value)">
        <option value="">Pilih Ruang...</option>
        <?php
            $result = $conn->query("SELECT * FROM ruang");
            while ($row = $result->fetch_assoc()) {
                echo "<option value='{$row['id_ruang']}'>{$row['nama_ruang']}";
            }
        ?>
    </select>

    <h3>Detail Ruangan</h3>
    <p><strong>Nama Ruang:</strong> <span id="detailNamaRuang"></span></p>
    <p><strong>Lokasi:</strong> <span id="detailLokasi"></span></p>
    <p><strong>Kapasitas:</strong> <span id="detailKapasitas"></span></p>
    <h4>Barang di Ruangan:</h4>
    <ul id="detailBarang"></ul>

    <div id="detailAksi"></div>
</div>
</body>
</html>

```

Back-End (PHP - Proses Peminjaman)

1. list_ruang.php (*terdapat dibagian atas list_ruang.php*)

```
<?php
session_start();
if ($_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
    exit();
}

include 'koneksi.php';

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $nama_ruang = $_POST['nama_ruang'] ?? null;
    $lokasi = $_POST['lokasi'] ?? null;
    $kapasitas = $_POST['kapasitas'] ?? null;

    if ($nama_ruang && $lokasi && $kapasitas) {
        $sql_insert = "INSERT INTO ruang (nama_ruang, lokasi, kapasitas)
            VALUES ('$nama_ruang', '$lokasi', '$kapasitas')";

        if (mysqli_query($koneksi, $sql_insert)) {
            echo "<p style='color: green;'>Ruang berhasil ditambahkan!</p>";
            header("Refresh: 1");
        } else {
            echo "<p style='color: red;'>Gagal menambahkan ruang: " .
mysqli_error($koneksi) . "</p>";
        }
    } else {
        echo "<p style='color: red;'>Harap isi semua bidang formulir.</p>";
    }
}

$query = "SELECT * FROM ruang";
$result = mysqli_query($koneksi, $query);

if (!$result) {
    die("Query gagal: " . mysqli_error($koneksi));
}
?>
```

2. get_ruang_detail.php

```
<?php
include 'koneksi3.php';

if (isset($_GET['id'])) {
    $id_ruang = $_GET['id'];
    $query = $conn->query("SELECT * FROM ruang WHERE id_ruang = '$id_ruang'");

    if (!$query) {
```

```

        die(json_encode(['error' => 'Query gagal: ' . $conn->error]));
    }

    if ($query->num_rows > 0) {
        $data = $query->fetch_assoc();

        $barangQuery = $conn->query("SELECT nama_barang FROM barang WHERE id_ruang
= '$id_ruang'");
        $barangList = "";
        while ($barang = $barangQuery->fetch_assoc()) {
            $barangList .= "<li>{$barang['nama_barang']}</li>";
        }

        echo json_encode([
            'nama_ruang' => $data['nama_ruang'],
            'lokasi' => $data['lokasi'],
            'kapasitas' => $data['kapasitas'],
            'barang' => $barangList
        ]);
    } else {
        echo json_encode(['error' => 'Data tidak ditemukan']);
    }
} else {
    echo json_encode(['error' => 'ID tidak diberikan']);
}
?>

```

3. edit_ruang.php

```

<?php
session_start();
if ($_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
    exit();
}

include 'koneksi.php';

$row = [];

if (isset($_GET['id'])) {
    $id_ruang = intval($_GET['id']);
    $query = "SELECT * FROM ruang WHERE id_ruang=$id_ruang";
    $result = mysqli_query($koneksi, $query);

    if (!$result) {
        die("Query gagal: " . mysqli_error($koneksi));
    }

    $row = mysqli_fetch_assoc($result);
    if (!$row) {
        die("Data tidak ditemukan.");
    }
}

```

```

}

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $id_ruang = intval($_POST['id_ruang']);
    $nama_ruang = mysqli_real_escape_string($koneksi, $_POST['nama_ruang']);
    $lokasi = mysqli_real_escape_string($koneksi, $_POST['lokasi']);
    $kapasitas = intval($_POST['kapasitas']);

    $query = "UPDATE ruang
              SET nama_ruang='$nama_ruang', lokasi='$lokasi',
kapasitas='$kapasitas'
              WHERE id_ruang=$id_ruang";

    if (mysqli_query($koneksi, $query)) {
        echo "Data berhasil diperbarui!";
        header("Location: gabungan5.php");
        exit();
    } else {
        echo "Error: " . mysqli_error($koneksi);
    }
}
?>

<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Edit Ruangan</title>
    <link rel="stylesheet" href="style_edit.css">
</head>
<body>

<form method="post">
    <input type="hidden" name="id_ruang" value="<?= isset($row['id_ruang']) ?
$row['id_ruang'] : '' ?>">

    <label>Nama Ruang:</label>
    <input type="text" name="nama_ruang" value="<?= isset($row['nama_ruang']) ?
$row['nama_ruang'] : '' ?>" required<br>

    <label>Lokasi:</label>
    <input type="text" name="lokasi" value="<?= isset($row['lokasi']) ?
$row['lokasi'] : '' ?>" required<br>

    <label>Kapasitas:</label>
    <input type="number" name="kapasitas" value="<?= isset($row['kapasitas']) ?
$row['kapasitas'] : '' ?>" required<br>

    <button type="submit">Update</button>
</form>

</body>

```

```
</html>
```

4. hapus_ruang.php

```
<?php
session_start();
if ($_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
    exit();
}

include 'koneksi.php';

if (isset($_GET['id'])) {
    $id_ruang = $_GET['id'];
    $query = "DELETE FROM ruang WHERE id_ruang=$id_ruang";

    if (mysqli_query($koneksi, $query)) {
        echo "Data berhasil dihapus!";
        header("Location: gabungan5.php");
        exit();
    } else {
        echo "Error: " . mysqli_error($koneksi);
    }
}
?>
```

c. Analisis Kode

Fitur **Pendataan Ruang Penyimpanan** bertugas mengelola informasi ruang tempat penyimpanan barang, mulai dari input, tampil detail, edit, dan hapus data ruang.

1. list_ruang.php

- Form input untuk **menambahkan ruang baru** ke database.
- Data yang diinput: nama ruang, lokasi, dan kapasitas.
- Setelah disubmit, data disimpan ke tabel `ruang`.
- Hanya bisa diakses oleh user dengan role **administrator**.
- Tersedia tombol navigasi ke halaman `gabungan5.php`.

2. gabungan5.php

- Menampilkan **daftar nama ruang** dalam bentuk dropdown.
- Setelah memilih ruang, sistem menampilkan detail ruang: nama, lokasi, kapasitas, dan daftar barang yang ada di ruang tersebut.
- Fitur **tombol Edit dan Hapus** otomatis muncul setelah ruang dipilih.
- Mengambil data dari tabel `ruang` dan `barang` menggunakan AJAX (`get_ruang_detail.php`).

3. get_ruang_detail.php

- File backend yang digunakan oleh AJAX.
- Menerima `id_ruang`, lalu mengambil data ruang & barang berdasarkan ID tersebut.

- Menyusun data barang ke dalam format list HTML.
- Mengembalikan data dalam format **JSON** agar bisa ditampilkan langsung di halaman `gabungan5.php`.

4. `edit_ruang.php`

- Menampilkan form **untuk mengedit data ruang** yang dipilih.
- Menarik data lama berdasarkan `id_ruang` dari URL.
- Setelah form dikirim, data `nama_ruang`, `lokasi`, dan `kapasitas` akan diperbarui di database.
- Setelah berhasil, pengguna langsung diarahkan ke `gabungan5.php`.

5. `hapus_ruang.php`

- Menghapus data ruang berdasarkan `id_ruang` yang dikirim via URL.
- Setelah berhasil dihapus, sistem mengarahkan ulang ke `gabungan5.php`.
- Tidak melakukan validasi apakah ruang masih memiliki barang aktif atau tidak.

d. Peran Fitur dalam Database

Fitur **Pendataan Ruang Penyimpanan** berfungsi untuk mencatat, memperbarui, menampilkan, dan menghapus data ruang. Setiap ruang digunakan untuk menyimpan data barang pada aplikasi inventaris.

- **`list_ruang.php`**
 - Menyimpan data ruang ke tabel `ruang`.
 - Kolom yang diisi: `nama_ruang`, `lokasi`, `kapasitas`.
- **`gabungan5.php`**
 - Mengambil data dari tabel `ruang` untuk dropdown dan detail.
 - Menampilkan barang di ruang tersebut berdasarkan data pada tabel `barang` (relasi: `barang.id_ruang`).
- **`get_ruang_detail.php`**
 - Mengambil satu baris dari tabel `ruang` berdasarkan `id_ruang`.
 - Mengambil nama barang dari tabel `barang` yang memiliki `id_ruang` sesuai.
 - Mengembalikan data ruang dan daftar barang dalam format JSON.
- **`edit_ruang.php`**
 - Memperbarui data di tabel `ruang` berdasarkan `id_ruang`.
- **`hapus_ruang.php`**
 - Menghapus data ruang dari tabel `ruang`.
 - Tidak menghapus data barang yang terkait, sehingga barang tetap ada di database walau ruang sudah dihapus.

4. Relasi

1. Manajemen Data Barang

Relasi:

- **barang.id_ruang → ruang.id_ruang**
→ Satu ruang dapat menyimpan banyak barang.

Penjelasan Tambahan:

- Lokasi penyimpanan ditentukan berdasarkan ID ruang, dan ditampilkan di kolom “Lokasi Penyimpanan”.

Analisis Query:

```
SELECT barang.nama_barang, barang.tipe_barang, barang.jumlah, ruang.nama_ruang
FROM barang JOIN ruang ON barang.id_ruang = ruang.id_ruang;
```

Menampilkan nama barang, tipe, jumlah, dan lokasi ruang penyimpanan.

2. Fitur Peminjaman Barang

Relasi:

- **transaksi.id_siswa → siswa.id_siswa**
→ Satu siswa dapat melakukan banyak transaksi.
- **detail_transaksi.id_transaksi → transaksi.id_transaksi**
→ Satu transaksi bisa memuat banyak item/barang.
- **detail_transaksi.id_barang → barang.id_barang**
→ Satu barang bisa dipinjam oleh banyak siswa melalui beberapa transaksi.

Analisis Query:

```
SELECT siswa.nama, barang.nama_barang, transaksi.tanggal_pinjam FROM transaksi
JOIN siswa ON transaksi.id_siswa = siswa.id_siswa JOIN detail_transaksi ON
transaksi.id_transaksi = detail_transaksi.id_transaksi JOIN barang ON
detail_transaksi.id_barang = barang.id_barang;
```

Menampilkan nama siswa, barang yang dipinjam, dan tanggal peminjaman.

3. Pendataan Ruang Penyimpanan

Relasi:

- **ruang.id_ruang → barang.id_ruang**
→ Satu ruang menyimpan banyak barang.

Analisis Query:

```
SELECT ruang.nama_ruang, ruang.keterangan, COUNT(barang.id_barang) AS jumlah_barang
FROM ruang LEFT JOIN barang ON ruang.id_ruang = barang.id_ruang GROUP BY
```

```
ruang.id_ruang;
```

Menampilkan nama ruang, keterangannya, dan jumlah barang yang tersimpan di dalamnya.

5. Agregasi

1. Fitur Manajemen Data Barang

Agregasi:

- **COUNT**: Menghitung jumlah barang yang tersedia di setiap ruang.
- **SUM**: Menghitung total jumlah stok barang di seluruh ruang.

Contoh Query:

```
SELECT ruang.nama_ruang, COUNT(barang.id_barang) AS jumlah_barang,  
SUM(barang.jumlah_barang) AS total_stok  
FROM ruang  
LEFT JOIN barang ON ruang.id_ruang = barang.id_ruang  
GROUP BY ruang.id_ruang;
```

Tujuan Agregasi:

- **COUNT(barang.id_barang)**: Untuk mengetahui berapa banyak jenis barang yang ada di setiap ruang penyimpanan.
- **SUM(barang.jumlah_barang)**: Untuk mendapatkan total stok barang yang tersedia di seluruh ruang, memberikan gambaran tentang ketersediaan barang.

Analisis Query:

- Query ini menggabungkan tabel `ruang` dan `barang` menggunakan `LEFT JOIN`, sehingga semua ruang ditampilkan meskipun tidak ada barang di dalamnya.
- Dengan menggunakan `GROUP BY ruang.id_ruang`, hasil query dikelompokkan berdasarkan ID ruang, sehingga setiap ruang hanya muncul sekali dengan total jumlah barang dan stoknya.

Cara Kerja Fitur Terhadap Agregasi:

- Fitur ini memungkinkan pengguna untuk melihat informasi penting tentang barang yang tersedia di setiap ruang, membantu dalam pengelolaan inventaris dan perencanaan ruang penyimpanan yang lebih efisien.

2. Fitur Peminjaman dan Pengelolaan Barang

Agregasi:

- **COUNT**: Menghitung total peminjaman barang oleh siswa.
- **AVG**: Menghitung rata-rata jumlah barang yang dipinjam per transaksi.

Contoh Query:

```
SELECT siswa.nama_siswa, COUNT(transaksi.id_transaksi) AS total_peminjaman,  
AVG(detail_transaksi.jumlah) AS rata_rata_barang  
FROM transaksi  
JOIN siswa ON transaksi.id_siswa = siswa.id_siswa  
JOIN detail_transaksi ON transaksi.id_transaksi = detail_transaksi.id_transaksi  
GROUP BY siswa.id_siswa;
```

Tujuan Agregasi:

- **COUNT(transaksi.id_transaksi)**: Untuk mengetahui berapa banyak transaksi peminjaman yang dilakukan oleh setiap siswa.
- **AVG(detail_transaksi.jumlah)**: Untuk mendapatkan rata-rata jumlah barang yang dipinjam per transaksi, memberikan wawasan tentang pola peminjaman siswa.

Analisis Query:

- Query ini menggabungkan tabel `transaksi`, `siswa`, dan `detail_transaksi` untuk mendapatkan informasi lengkap tentang peminjaman.
- Dengan `GROUP BY siswa.id_siswa`, hasil query dikelompokkan berdasarkan ID siswa, sehingga setiap siswa hanya muncul sekali dengan total peminjaman dan rata-rata jumlah barang yang dipinjam.

Cara Kerja Fitur Terhadap Agregasi:

- Fitur ini memberikan informasi yang berguna bagi administrator untuk memantau aktivitas peminjaman siswa, membantu dalam pengelolaan barang dan perencanaan peminjaman.

3. Fitur Pendataan Ruang Penyimpanan

Agregasi:

- **COUNT**: Menghitung jumlah barang yang ada di setiap ruang.
- **MAX**: Menentukan kapasitas maksimum ruang yang tersedia.

Contoh Query:

```
SELECT ruang.nama_ruang, COUNT(barang.id_barang) AS jumlah_barang,  
MAX(ruang.kapasitas) AS kapasitas_maksimum  
FROM ruang  
LEFT JOIN barang ON ruang.id_ruang = barang.id_ruang  
GROUP BY ruang.id_ruang;
```

Tujuan Agregasi:

- **COUNT(barang.id_barang)**: Untuk mengetahui berapa banyak barang yang ada di setiap ruang penyimpanan.

- **MAX(ruang.kapasitas)**: Untuk mendapatkan kapasitas maksimum dari ruang penyimpanan, membantu dalam memahami seberapa banyak barang yang dapat ditampung.

Analisis Query:

- Query ini menggunakan `LEFT JOIN` untuk memastikan semua ruang ditampilkan, bahkan jika tidak ada barang di dalamnya.
- Dengan `GROUP BY ruang.id_ruang`, hasil query dikelompokkan berdasarkan ID ruang, sehingga setiap ruang hanya muncul sekali dengan jumlah barang dan kapasitas maksimumnya.

Cara Kerja Fitur Terhadap Agregasi:

- Fitur ini memberikan gambaran yang jelas tentang penggunaan ruang penyimpanan dan kapasitasnya, membantu administrator dalam merencanakan pengelolaan ruang dan barang yang lebih baik.

6. Autentikasi

Autentikasi dalam website ini berfungsi untuk membatasi akses ke fitur-fitur tertentu berdasarkan jenis pengguna (user). Namun pada website ini hanya terdapat satu jenis pengguna yang diatur dalam sistem, yaitu **administrator**.

Fitur yang Diproteksi

1. Manajemen Data Barang:

- Hanya dapat diakses oleh pengguna dengan role **administrator**.
- Tujuannya adalah untuk mencegah pengguna biasa mengubah data barang yang sensitif, seperti menambah, mengedit, atau menghapus informasi barang.

2. Peminjaman dan Pengelolaan Barang:

- Fitur ini juga dibatasi hanya untuk **administrator**.
- Mencegah penyalahgunaan akses oleh pengguna biasa yang dapat merusak data peminjaman, seperti mengubah status peminjaman atau menghapus transaksi.

3. Pendataan Ruang Penyimpanan:

- Hanya **administrator** yang dapat menambah, mengedit, atau menghapus data ruang.
- Tujuannya adalah untuk menjaga integritas data ruang penyimpanan, memastikan bahwa informasi tentang lokasi dan kapasitas ruang tidak diubah sembarangan.

4. Fitur Login:

- Terdapat fitur login yang memungkinkan pengguna untuk masuk ke dalam sistem sebagai admin.
- Proses login ini memverifikasi kredensial pengguna (username dan password) sebelum memberikan akses ke fitur-fitur yang dilindungi.
- Hanya pengguna yang berhasil login sebagai administrator yang dapat mengakses fitur-fitur sensitif dalam aplikasi.

Analisis Program

Dalam setiap file PHP yang memerlukan autentikasi, terdapat kode berikut:

```
session_start();
if ($_SESSION['role'] !== 'administrator') {
    header("Location: login.php");
    exit();
}
```

*Penjelasan Kode:

- **session_start()**: Memulai sesi untuk melacak status login pengguna. Ini memungkinkan aplikasi untuk menyimpan informasi tentang pengguna yang sedang aktif.
- **if (\$_SESSION['role'] !== 'administrator')**: Memeriksa apakah pengguna yang sedang login memiliki role sebagai administrator. Jika tidak, akses ke halaman tersebut akan dibatasi.
- **header("Location: login.php")**: Jika pengguna tidak memiliki hak akses, mereka akan diarahkan kembali ke halaman login. Ini mencegah akses tidak sah ke fitur-fitur penting.
- **exit()**: Menghentikan eksekusi script lebih lanjut untuk mencegah akses ke halaman yang dilindungi. Ini memastikan bahwa tidak ada kode lain yang dieksekusi setelah pengalihan.

Dengan cara ini, website memastikan bahwa hanya pengguna yang berwenang yang dapat mengakses dan mengelola data penting, sehingga menjaga keamanan dan integritas sistem inventaris sekolah. Fitur login sebagai admin menambah lapisan keamanan tambahan, memastikan bahwa hanya individu yang memiliki kepentingan yang dapat mengakses fitur-fitur sensitif dalam website.