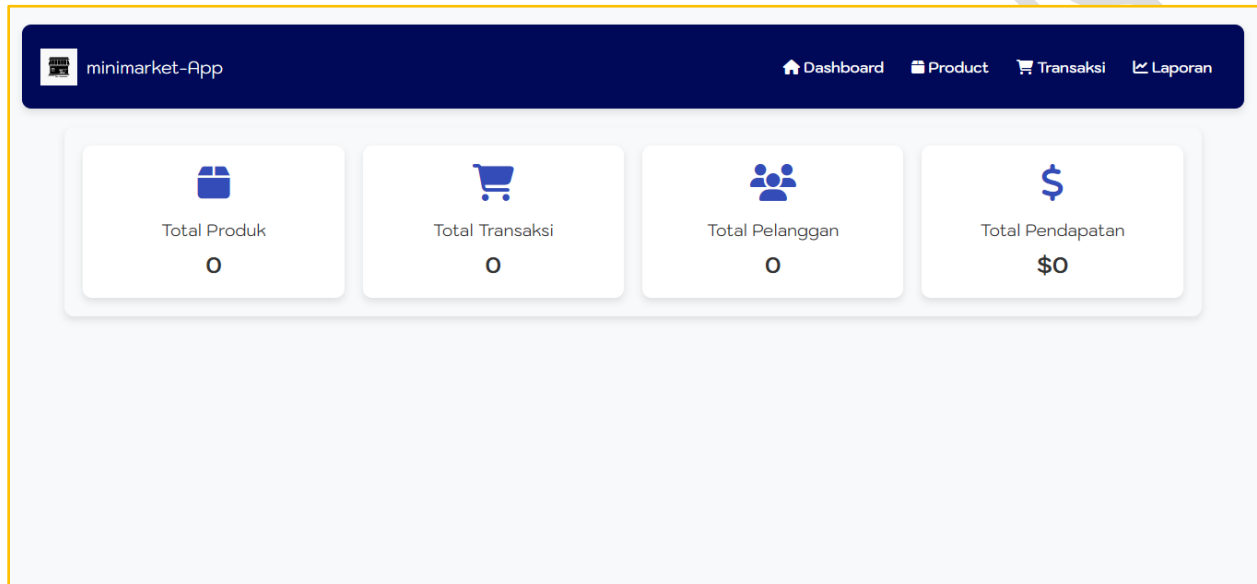


## PEMBAHASAN UKK PAKET 4

### APLIKASI MINIMARKET DENGAN DISKON



Rekayasa Perangkat Lunak - SMK Negeri 1 Kuwus

Manggarai Barat

Nusa Tenggara Timur

2024/2025

## A. SCRIPT HTML (index.html)

Kode lengkap dapat ditampilkan sebagai berikut:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>MiniMarket - Dashboard</title>
  <link rel="stylesheet" href="css/style.css">

  <!-- FontAwesome for Icons -->
  <link rel="stylesheet" href="assets/fonts/font-
awesome/all.min.css">

  <!-- Google Fonts -->
  <link href="assets/fonts/kodchasan/css2.css" rel="stylesheet">

</head>
<body>

  <!-- Navbar -->
  <div class="navbar">
    <div class="logo">
      
      <h3>minimarket-App</h3>
    </div>
    <span class="nav-toggle">&#9776;</span>
    <ul class="menu">
      <li><a href="index.html"><i class="fas fa-home"></i>
Dashboard</a></li>
      <li><a href="barang.html"><i class="fas fa-box"></i>
Barang</a></li>
      <li><a href="transaksi.html"><i class="fas fa-shopping-
cart"></i> Transaksi</a></li>
      <li><a href="report.html"><i class="fas fa-chart-line"></i>
Laporan</a></li>
    </ul>
  </div>
```

```
<!-- Dashboard Content -->
<div class="container_dashboard">
  <div class="card">
    <i class="fas fa-box"></i>
    <h3>Total Produk</h3>
    <div class="count" id="totalProducts">0</div>
  </div>

  <div class="card">
    <i class="fas fa-shopping-cart"></i>
    <h3>Total Transaksi</h3>
    <div class="count" id="totalTransactions">0</div>
  </div>

  <div class="card">
    <i class="fas fa-users"></i>
    <h3>Total Pelanggan</h3>
    <div class="count" id="totalCustomers">0</div>
  </div>

  <div class="card">
    <i class="fas fa-dollar-sign"></i>
    <h3>Total Pendapatan</h3>
    <div class="count" id="totalRevenue">$0</div>
  </div>
</div>

<script src="js/index.js"></script>

</body>
</html>
```

Script HTML di atas adalah template untuk **Dashboard** aplikasi "MiniMarket-App." Berikut adalah penjelasan rinci dari setiap bagiannya:

## 1. Bagian Head (<head>...</head>)

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>MiniMarket - Dashboard</title>
  <link rel="stylesheet" href="css/style.css">

  <!-- FontAwesome for Icons -->
  <link rel="stylesheet" href="assets/fonts/font-
awesome/all.min.css">

  <!-- Google Fonts -->
  <link href="assets/fonts/kodchasan/css2.css" rel="stylesheet">
</head>
```

- **<meta charset="UTF-8">**: Menentukan karakter encoding sebagai UTF-8, agar teks bisa ditampilkan dengan benar, termasuk karakter khusus.
- **<meta name="viewport"...>**: Membuat tampilan responsif, terutama untuk perangkat mobile.
- **<title>**: Menentukan judul halaman pada tab browser ("MiniMarket - Dashboard").
- **<link rel="stylesheet" href="css/style.css">**: Menghubungkan file CSS eksternal untuk styling.
- **FontAwesome (all.min.css)**: Digunakan untuk menampilkan ikon-ikon yang ada pada navbar dan card.
- **Google Fonts (kodchasan/css2.css)**: Mengimpor font khusus (Kodchasan) untuk styling teks.

## 2. Navbar (<div class="navbar">...</div>)

```
<div class="navbar">
  <div class="logo">
    
    <h3>minimarket-App</h3>
  </div>
  <span class="nav-toggle">&#9776;</span>
  <ul class="menu">
    <li><a href="index.html"><i class="fas fa-home"></i>
Dashboard</a></li>
    <li><a href="barang.html"><i class="fas fa-box"></i>
Barang</a></li>
    <li><a href="transaksi.html"><i class="fas fa-shopping-
cart"></i> Transaksi</a></li>
    <li><a href="report.html"><i class="fas fa-chart-line"></i>
Laporan</a></li>
  </ul>
</div>
```

- **<div class="logo">**: Menampilkan logo dan nama aplikasi.
- **<span class="nav-toggle">**: Simbol ☰ (hamburger icon) untuk toggle menu pada tampilan mobile.
- **<ul class="menu">...</ul>**: Daftar menu navigasi:
  - ✓ **Dashboard** (index.html) - Ikon Home
  - ✓ **Barang** (barang.html) - Ikon Box
  - ✓ **Transaksi** (transaksi.html) - Ikon Shopping Cart
  - ✓ **Laporan** (report.html) - Ikon Chart Line

Semua ikon menggunakan FontAwesome (<i class="fas ...">).

## 3. Dashboard Content (<div class="container\_dashboard">...</div>)

```
<div class="container_dashboard">
  <div class="card">
    <i class="fas fa-box"></i>
    <h3>Total Produk</h3>
    <div class="count" id="totalProducts">0</div>
  </div>

  <div class="card">
```

```
<i class="fas fa-shopping-cart"></i>
<h3>Total Transaksi</h3>
<div class="count" id="totalTransactions">0</div>
</div>

<div class="card">
  <i class="fas fa-users"></i>
  <h3>Total Pelanggan</h3>
  <div class="count" id="totalCustomers">0</div>
</div>

<div class="card">
  <i class="fas fa-dollar-sign"></i>
  <h3>Total Pendapatan</h3>
  <div class="count" id="totalRevenue">$0</div>
</div>
</div>
```

Bagian ini berisi **Card Dashboard** yang menampilkan ringkasan data:

- **Total Produk** (id="totalProducts") dengan ikon Box.
- **Total Transaksi** (id="totalTransactions") dengan ikon Shopping Cart.
- **Total Pelanggan** (id="totalCustomers") dengan ikon Users.
- **Total Pendapatan** (id="totalRevenue") dengan ikon Dollar Sign.

Masing-masing menggunakan struktur:

- **<i class="fas ...">**: Menampilkan ikon FontAwesome.
- **<h3>**: Judul ringkasan (misalnya, Total Produk).
- **<div class="count" id="...">**: Nilai data yang saat ini diset ke 0. Angka ini bisa di-update menggunakan JavaScript.

#### 4. Script JavaScript (<script src="js/index.js"></script>)

Menautkan file JavaScript eksternal (index.js) yang akan mengelola:

- **Interaksi pengguna** (misalnya, toggle navbar).
- **Update data dashboard** (misalnya, mengambil data dari database melalui API dengan AJAX atau FETCH).

#### Fitur yang Perlu Diperhatikan:

1. **Responsif**: Menggunakan <meta name="viewport"> dan hamburger icon (nav-toggle) untuk mobile.
2. **Navigasi Dinamis**: Menggunakan FontAwesome untuk ikon navigasi yang menarik.
3. **Card Dashboard**: Desain yang rapi untuk menampilkan ringkasan data.
4. **Modular**: Menggunakan file eksternal untuk CSS (style.css) dan JS (index.js), sehingga mudah dikelola.

## B. SCRIPT CSS (style.css)

Kode lengkap dapat ditampilkan sebagai berikut:

```
body {
  font-family: "Kodchasan", serif;
  font-weight: 400;
  font-style: normal;
  margin: 20px;
  background-color: #f8f9fa;
  color: #333;
}

/* Navbar Styling */
.navbar {
  background-color: #000957;
  color: white;
  padding: 15px 20px;
  display: flex;
  align-items: center;
  justify-content: space-between;
  border-radius: 10px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
}

.navbar img {
  height: 40px;
}

.logo {
  display: flex;
  align-items: center;
  justify-content: space-between;
  gap: 10px;
}

.navbar .menu {
  list-style: none;
  display: flex;
  margin: 0;
  padding: 0;
}

.navbar .menu li {
  margin: 0 15px;
}
```



```
.navbar .menu li a {
    text-decoration: none;
    color: white;
    font-weight: 600;
    transition: 0.3s;
}

.navbar .menu li a:hover {
    text-decoration: underline;
    color: #f4f4f4;
}

/* Container */
.container_dashboard {
    width: 90%;
    margin: 20px auto;
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
    padding: 20px;
    gap: 20px;
    border-radius: 10px;
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
}

/* Container */
.container {
    width: 50%;
    background: #fff;
    margin: 20px auto;
    padding: 20px;
    border-radius: 10px;
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
}

/* Button Styling */
.btn {
    padding: 6px 10px;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    font-size: 14px;
}
```

```
}  
.btn-edit {  
    background-color: #024CAA;  
    color: white;  
}  
.btn-delete {  
    background-color: #C62E2E;  
    color: white;  
}  
.btn-cancel {  
    background-color: #FF6500;  
    color: white;  
}  
.btn-add {  
    background-color: #0B2F9F;  
    color: white;  
}  
.btn-green {  
    background-color: teal;  
    color: white;  
}  
  
.btn i {  
    margin-right: 5px;  
}  
  
/* Dashboard Cards */  
.card {  
    background: white;  
    padding: 20px;  
    border-radius: 10px;  
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);  
    text-align: center;  
}  
.card h3 {  
    margin: 10px 0;  
    font-size: 18px;  
}  
.card i {  
    font-size: 40px;  
    margin-bottom: 10px;
```

```
        color: #344CB7;
    }
    .card .count {
        font-size: 24px;
        font-weight: bold;
        color: #333;
    }

    /* Table Styling */
    table {
        width: 100%;
        border-collapse: collapse;
        background: #F8FAE5;
        box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
        border-radius: 8px;
        overflow: hidden;
        margin-bottom: 10px;
    }
    table {
        width: 100%;
        border-collapse: collapse;
        margin-top: 10px;
    }
    th, td {
        padding: 12px;
        border: 1px solid #ddd;
        text-align: left;
    }
    th {
        background-color: #344CB7;
        color: white;
    }
    td i {
        margin-right: 5px;
        color: whitesmoke;
    }

    /* CSS Add Barang*/
    form {
```

```
    max-width: 100%;
    background: #f9f9f9;
    padding: 20px;
    border-radius: 8px;
    box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);
}
label {
    display: block;
    margin-top: 10px;
}
input, select {
    width: 98%;
    padding: 8px;
    margin-top: 5px;
    border: 1px solid #ccc;
    border-radius: 5px;
}

.button {
    padding: 10px 20px;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    margin-top: 15px;
    margin-bottom: 15px;
}

.rightButton {
    text-align: right;
}

/* Responsive Navbar */
.nav-toggle {
    display: none;
    font-size: 24px;
    cursor: pointer;
}

@media (max-width: 768px) {
    .navbar .menu {
```

```
        display: none;
        flex-direction: column;
        position: absolute;
        top: 60px;
        right: 0;
        gap: 5;
        background-color: #344CB7;
        width: 100%;
        text-align: left;
        padding: 10px 0;
    }
    .navbar .menu.active {
        display: flex;
    }
    .nav-toggle {
        display: block;
    }
    table {
        font-size: 14px;
    }
}

/* Responsive Container */
@media (max-width: 600px) {
    .container, .container_dashboard {
        width: 95%;
        padding: 15px;
    }
}

/* Responsive Table */
.table-container {
    overflow-x: auto;
}
```

Script CSS di atas digunakan untuk **styling** halaman dashboard **MiniMarket-App** yang telah dijelaskan sebelumnya. Berikut adalah penjelasan terperinci dari setiap bagian:

## 1. Global Styling

```
body {  
  font-family: "Kodchasan", serif;  
  font-weight: 400;  
  font-style: normal;  
  margin: 20px;  
  background-color: #f8f9fa;  
  color: #333;  
}
```

- **Font:** Menggunakan **Kodchasan** dari Google Fonts.
- **Background:** Warna abu terang (#f8f9fa).
- **Text Color:** Warna teks utama adalah abu gelap (#333).
- **Margin:** Memberikan jarak 20px di seluruh sisi halaman.

## 2. Navbar Styling

```
.navbar {  
  background-color: #000957;  
  color: white;  
  padding: 15px 20px;  
  display: flex;  
  align-items: center;  
  justify-content: space-between;  
  border-radius: 10px;  
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);  
}
```

- **Background Color:** Biru gelap (#000957).
- **Flexbox Layout:** Digunakan untuk tata letak horizontal yang rapi.
- **Border Radius:** Membuat sudut melengkung (10px).
- **Box Shadow:** Memberi efek bayangan lembut.

```
.navbar img {  
  height: 40px;  
}  
.logo {  
  display: flex;  
  align-items: center;  
  justify-content: space-between;  
  gap: 10px;  
}
```

Logo:

- **Gambar logo** diatur tingginya 40px.
- **Teks logo** diposisikan rapi dengan **flexbox** dan jarak 10px di antaranya.

```
.navbar .menu {  
  list-style: none;  
  display: flex;  
  margin: 0;  
  padding: 0;  
}  
.navbar .menu li {  
  margin: 0 15px;  
}  
.navbar .menu li a {  
  text-decoration: none;  
  color: white;  
  font-weight: 600;  
  transition: 0.3s;  
}  
.navbar .menu li a:hover {  
  text-decoration: underline;  
  color: #f4f4f4;  
}
```

#### Menu Navbar:

- Horizontal Flexbox Layout.
- Link memiliki animasi hover (underline dan warna lebih terang).
- Jarak Antar Item: 15px di antara tiap menu.

### 3. Container Styling

```
.container_dashboard {  
  width: 90%;  
  margin: 20px auto;  
  display: grid;  
  grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));  
  padding: 20px;  
  gap: 20px;  
  border-radius: 10px;  
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);  
}
```

- **Grid Layout:** Menggunakan `grid-template-columns` untuk tata letak dinamis yang responsif.
- **Card** akan otomatis menyesuaikan lebar (min 250px) dan menjadi 1 kolom saat layar kecil.
- **Padding dan Jarak Antar Card:** 20px.

```
.container {  
  width: 50%;  
  background: #fff;  
  margin: 20px auto;  
  padding: 20px;  
  border-radius: 10px;  
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);  
}
```

#### Form Container:

- Lebar 50% dari layar.
- Tengah dengan `margin: 20px auto`.
- Background Putih (#fff) dengan bayangan lembut.

#### 4. Button Styling

```
.btn {  
  padding: 6px 10px;  
  border: none;  
  border-radius: 5px;  
  cursor: pointer;  
  font-size: 14px;  
}
```

- **Padding dan Border Radius:** Memberikan efek tombol bulat lembut.
- **Cursor** berubah jadi `pointer` saat dihover.

#### Warna Berdasarkan Fungsi:

```
.btn-edit { background-color: #024CAA; color: white; }  
.btn-delete { background-color: #C62E2E; color: white; }  
.btn-cancel { background-color: #FF6500; color: white; }  
.btn-add { background-color: #0B2F9F; color: white; }  
.btn-green { background-color: teal; color: white; }
```

Setiap tombol memiliki warna spesifik untuk mempermudah identifikasi aksi:

- **Edit:** Biru Tua



- **Delete:** Merah
- **Cancel:** Orange
- **Add:** Biru Lebih Gelap
- **Green:** Hijau Tua

## 5. Dashboard Cards

```
.card {  
  background: white;  
  padding: 20px;  
  border-radius: 10px;  
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);  
  text-align: center;  
}  
.card h3 {  
  margin: 10px 0;  
  font-size: 18px;  
}  
.card i {  
  font-size: 40px;  
  margin-bottom: 10px;  
  color: #344CB7;  
}  
.card .count {  
  font-size: 24px;  
  font-weight: bold;  
  color: #333;  
}
```

Card memiliki:

- **Background Putih** dengan **Box Shadow** lembut.
- **Ikon Besar** (font-size: 40px) dengan warna biru (#344CB7).
- **Angka Count** tampil besar dan tebal.

## 6. Table Styling

```
table {  
  width: 100%;  
  border-collapse: collapse;  
  background: #F8FAE5;  
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);  
  border-radius: 8px;  
  overflow: hidden;  
}
```

```
th {
  background-color: #344CB7;
  color: white;
}
th, td {
  padding: 12px;
  border: 1px solid #ddd;
  text-align: left;
}
```

- Tabel Responsif dengan overflow: hidden.
- Header (<th>) berwarna biru gelap dengan teks putih.
- Padding yang Nyaman (12px) untuk konten.

## 7. Form Styling

```
form {
  max-width: 100%;
  background: #f9f9f9;
  padding: 20px;
  border-radius: 8px;
  box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);
}
input, select {
  width: 98%;
  padding: 8px;
  border: 1px solid #ccc;
  border-radius: 5px;
}
```

- Form Background warna abu terang (#f9f9f9).
- Input dan Select memiliki border lembut dan padding yang nyaman.

## 8. Responsive Navbar & Container

```
.nav-toggle {
  display: none;
}
@media (max-width: 768px) {
  .navbar .menu {
    display: none;
    flex-direction: column;
    position: absolute;
    top: 60px;
  }
}
```

```
    right: 0;
    background-color: #344CB7;
    width: 100%;
    text-align: left;
    padding: 10px 0;
  }
  .navbar .menu.active {
    display: flex;
  }
  .nav-toggle {
    display: block;
  }
}
```

- Hamburger Icon (.nav-toggle) ditampilkan saat layar kurang dari 768px.
- Menu Navbar menjadi vertikal dengan animasi **slide** dari kanan.

### C. SCRIPT HTML (barang.html)

Kode lengkap dapat ditampilkan sebagai berikut:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>MiniMarket Produk</title>
  <link rel="stylesheet" href="css/style.css">

  <!-- FontAwesome for Icons -->
  <link rel="stylesheet" href="assets/fonts/font-
awesome/all.min.css">

  <!-- Google Fonts -->
  <link href="assets/fonts/kodchasan/css2.css" rel="stylesheet">

</head>
<body>
  <!-- Navbar -->
  <div class="navbar">
    <div class="logo">
      
      <h3>minimarket-App</h3>
    </div>
    <span class="nav-toggle">&#9776;</span>
    <ul class="menu">
      <li><a href="index.html"><i class="fas fa-home"></i>
Dashboard</a></li>
      <li><a href="barang.html"><i class="fas fa-box"></i>
Barang</a></li>
      <li><a href="transaksi.html"><i class="fas fa-shopping-
cart"></i> Transaksi</a></li>
      <li><a href="report.html"><i class="fas fa-chart-line"></i>
Laporan</a></li>
    </ul>
  </div>

  <div class="container">
```

```
<div id="form-barang">
  <h2 id="judul"><i class="fas fa-edit"></i> Tambah
Barang</h2>
  <form id="barangForm">
    <input type="hidden" id="id_barang">

    <label for="nama_barang"> Nama Barang:</label>
    <input type="text" id="nama_barang" required>

    <label for="kategori">Kategori:</label>
    <input type="text" id="kategori" required>

    <label for="harga">Harga:</label>
    <input type="number" id="harga" required>

    <label for="diskon">Diskon:</label>
    <input type="number" id="diskon" required>

    <label for="stok">Stok:</label>
    <input type="number" id="stok" required>

    <button class="button btn-add" type="submit">Simpan
Produk</button>
    <button type="button" class="button btn-cancel"
onclick="window.location.href='barang.html'">Kembali</button>
  </form>
</div>

<div id="display-barang">
  <h2><i class="fas fa-box"></i> Daftar Produk</h2>
  <button class="button btn-add" id="addBarang">Tambah
Produk</button>
  <table id="productBarang" class="table-container">
    <thead>
      <tr>
        <th><i class="fas fa-tag"></i> Nama</th>
        <th><i class="fas fa-list"></i> Kategori</th>
        <th><i class="fas fa-dollar-sign"></i>
Harga</th>
        <th><i class="fas fa-boxes"></i> Stok</th>
      </tr>
    </thead>
  </table>
</div>
```

```

                <th><i class="fa-solid fa-percent"></i>
Diskon</th>
                <th>Actions</th>
            </tr>
        </thead>
        <tbody id="barangTable">
            <!-- Barang rows will be inserted here by
JavaScript -->
        </tbody>
    </table>
</div>

</div>

<script src="js/jquery-3.7.1.min.js"></script>
<script src="js/barang/barang.js"></script>
</body>
</html>

```

Script HTML di atas adalah halaman produk untuk MiniMarket-App yang menampilkan formulir penambahan produk dan tabel daftar produk. Berikut adalah penjelasan terperinci dari setiap bagian:

### 1. Bagian <head>

```

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>MiniMarket Produk</title>
    <link rel="stylesheet" href="css/style.css">

    <!-- FontAwesome for Icons -->
    <link rel="stylesheet" href="assets/fonts/font-
awesome/all.min.css">

    <!-- Google Fonts -->
    <link href="assets/fonts/kodchasan/css2.css" rel="stylesheet">
</head>

```

Penjelasan:

- `<meta charset="UTF-8">`: Mengatur karakter encoding menjadi UTF-8 untuk mendukung berbagai karakter.
- `<meta name="viewport" content="width=device-width, initial-scale=1.0">`: Membuat halaman responsif pada berbagai ukuran layar, terutama **mobile**.
- `<title>`: Memberi judul halaman **"MiniMarket Produk"** yang muncul di tab browser.
- `<link rel="stylesheet" href="css/style.css">`: Menghubungkan halaman dengan file CSS eksternal (style.css) untuk styling.
- **FontAwesome**:
  - Menggunakan **FontAwesome** untuk ikon pada navigasi dan tabel (all.min.css).
- **Google Fonts**:
  - Menggunakan **Kodchasan** dari Google Fonts untuk tampilan teks yang lebih menarik.

## 2. Navbar (`<div class="navbar">`)

```
<div class="navbar">
  <div class="logo">
    
    <h3>minimarket-App</h3>
  </div>
  <span class="nav-toggle">&#9776;</span>
  <ul class="menu">
    <li><a href="index.html"><i class="fas fa-home"></i>
Dashboard</a></li>
    <li><a href="barang.html"><i class="fas fa-box"></i>
Barang</a></li>
    <li><a href="transaksi.html"><i class="fas fa-shopping-
cart"></i> Transaksi</a></li>
    <li><a href="report.html"><i class="fas fa-chart-line"></i>
Laporan</a></li>
  </ul>
</div>
```

### Penjelasan:

- **Logo**:
  - Gambar Logo (img/logo.png) diikuti dengan teks judul: "minimarket-App".
- **Hamburger Menu** (`<span class="nav-toggle">`):

- Menggunakan karakter &#9776; (≡) untuk **icon menu** pada layar **mobile**.
- **Menu Navigasi (<ul class="menu">):**
  - **Menu Link dengan ikon FontAwesome:**
    - **Dashboard:** <i class="fas fa-home"></i>
    - **Barang:** <i class="fas fa-box"></i>
    - **Transaksi:** <i class="fas fa-shopping-cart"></i>
    - **Laporan:** <i class="fas fa-chart-line"></i>
- **Navigasi Responsif:**
  - Akan berubah menjadi **dropdown menu** pada **mobile view** (dikendalikan oleh CSS & JS).

### 3. Form Tambah Produk (<div id="form-barang">)

```
<div id="form-barang">
  <h2 id="judul"><i class="fas fa-edit"></i> Tambah Barang</h2>
  <form id="barangForm">
    <input type="hidden" id="id_barang">

    <label for="nama_barang"> Nama Barang:</label>
    <input type="text" id="nama_barang" required>

    <label for="kategori">Kategori:</label>
    <input type="text" id="kategori" required>

    <label for="harga">Harga:</label>
    <input type="number" id="harga" required>

    <label for="diskon">Diskon:</label>
    <input type="number" id="diskon" required>

    <label for="stok">Stok:</label>
    <input type="number" id="stok" required>

    <button class="button btn-add" type="submit">Simpan
Produk</button>
    <button type="button" class="button btn-cancel"
onclick="window.location.href='barang.html'">Kembali</button>
  </form>
</div>
```

**Penjelasan:**



- **Judul Form:** Menggunakan ikon edit (`<i class="fas fa-edit"></i>`) dengan teks "Tambah Barang".
- **Form Input:**
  - Nama Barang (text), Kategori (text), Harga (number), Diskon (number), Stok (number).
- **Button:**
  - Simpan Produk: Tombol submit dengan class .btn-add.
  - Kembali: Tombol cancel yang redirect ke halaman barang.html.

#### 4. Daftar Produk (`<div id="display-barang">`)

```
<div id="display-barang">
  <h2><i class="fas fa-box"></i> Daftar Produk</h2>
  <button class="button btn-add" id="addBarang">Tambah
Produk</button>
  <table id="productBarang" class="table-container">
    <thead>
      <tr>
        <th><i class="fas fa-tag"></i> Nama</th>
        <th><i class="fas fa-list"></i> Kategori</th>
        <th><i class="fas fa-dollar-sign"></i> Harga</th>
        <th><i class="fas fa-boxes"></i> Stok</th>
        <th><i class="fa-solid fa-percent"></i> Diskon</th>
        <th>Actions</th>
      </tr>
    </thead>
    <tbody id="barangTable">
      <!-- Barang rows will be inserted here by JavaScript -->
    </tbody>
  </table>
</div>
```

#### Penjelasan:

- **Judul Tabel:** Menggunakan ikon box (`<i class="fas fa-box"></i>`) dengan teks "Daftar Produk".
- **Button Tambah Produk:**
  - Memiliki ID addBarang untuk trigger modal atau form baru.
- **Tabel Produk (`<table id="productBarang">`):**
  - Kolom Header dengan ikon FontAwesome:
    - Nama (fa-tag), Kategori (fa-list), Harga (fa-dollar-sign), Stok (fa-boxes), Diskon (fa-percent).
  - Actions: Tempat untuk tombol Edit dan Delete.

## 5. JavaScript Include

```
<script src="js/jquery-3.7.1.min.js"></script>  
<script src="js/barang/barang.js"></script>
```

### Penjelasan:

- **jquery-3.7.1.min.js**: Memasukkan **jQuery** untuk manipulasi DOM yang lebih mudah.
- **barang.js**:
  - File JavaScript kustom untuk fungsi tambah, edit, hapus, dan manajemen produk.
  - **AJAX Request** ke backend untuk **CRUD Produk**.

## D. SCRIPT HTML (transaksi.html)

Kode lengkap dapat ditampilkan sebagai berikut:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>MiniMarket - Transaksi</title>
  <link rel="stylesheet" href="css/style.css">

  <!-- FontAwesome for Icons -->
  <link rel="stylesheet" href="assets/fonts/font-
awesome/all.min.css">

  <!-- Google Fonts -->
  <link href="assets/fonts/kodchasan/css2.css" rel="stylesheet">

</head>
<body>

  <!-- Navbar -->
  <div class="navbar">
    <div class="logo">
      
      <h3>minimarket-App</h3>
    </div>
    <span class="nav-toggle">&#9776;</span>
    <ul class="menu">
      <li><a href="index.html"><i class="fas fa-home"></i>
Dashboard</a></li>
      <li><a href="barang.html"><i class="fas fa-box"></i>
Barang</a></li>
      <li><a href="transaksi.html"><i class="fas fa-shopping-
cart"></i> Transaksi</a></li>
      <li><a href="report.html"><i class="fas fa-chart-line"></i>
Laporan</a></li>
    </ul>
  </div>
```

```
<div class="container">
  <div id="form-transaksi">
    <h2 id="judul"><i class="fas fa-edit"></i> Form
Transaksi</h2>
    <form id="transaksiForm">
      <select id="id_barang" required> </select>
      <label for="stok">Jumlah barang:</label>
      <input type="number" id="jumlah" placeholder="Jumlah"
required>
      <button type="submit" class="button btn-green">Tambah
Transaksi</button>
      <button type="button" class="button btn-cancel"
onclick="window.location.href='transaksi.html'">Kembali</button>
    </form>
  </div>
  <div id="display-transaksi">
    <h2><i class="fas fa-shopping-cart"></i> Daftar
Transaksi</h2>
    <button class="button btn-add" id="addTransaksi">Mulai
Transaksi</button>
    <table>
      <thead>
        <tr>
          <th>Nama Barang</th>
          <th>Jumlah</th>
          <th>Total Harga</th>
          <th>Tanggal</th>
          <th>Aksi</th>
        </tr>
      </thead>
      <tbody id="transaksiTable">
        <!-- Data akan diisi oleh JavaScript -->
      </tbody>
    </table>
  </div>

</div>

<script src="js/jquery-3.7.1.min.js"></script>
<script src="js/transaksi/transaksi.js"></script>
```

```
</body>  
</html>
```

Script HTML di atas adalah halaman transaksi untuk MiniMarket-App yang berfungsi untuk mengelola proses transaksi seperti menambah item ke transaksi dan melihat daftar transaksi yang sudah dilakukan. Berikut adalah penjelasan lengkap dari setiap bagiannya:

## 1. Bagian <head>

```
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-  
scale=1.0">  
  <title>MiniMarket - Transaksi</title>  
  <link rel="stylesheet" href="css/style.css">  
  
  <!-- FontAwesome for Icons -->  
  <link rel="stylesheet" href="assets/fonts/font-  
awesome/all.min.css">  
  
  <!-- Google Fonts -->  
  <link href="assets/fonts/kodchasan/css2.css" rel="stylesheet">  
</head>
```

### Penjelasan:

- **<meta charset="UTF-8">**: Mengatur karakter encoding menjadi UTF-8 untuk mendukung berbagai karakter.
- **<meta name="viewport" content="width=device-width, initial-scale=1.0">**: Membuat halaman responsif agar tampil dengan baik di perangkat mobile.
- **<title>**: Memberi judul halaman "MiniMarket - Transaksi" yang muncul di tab browser.
- **<link rel="stylesheet" href="css/style.css">**:
  - Menghubungkan dengan file CSS eksternal (style.css) untuk styling tampilan.
- **FontAwesome**:
  - Digunakan untuk ikon navigasi dan ikon dalam form.
- **Google Fonts**:
  - Menggunakan font Kodchasan untuk tampilan teks yang lebih menarik.

## 2. Navbar (<div class="navbar">)

```
<div class="navbar">
  <div class="logo">
    
    <h3>minimarket-App</h3>
  </div>
  <span class="nav-toggle">&#9776;</span>
  <ul class="menu">
    <li><a href="index.html"><i class="fas fa-home"></i>
Dashboard</a></li>
    <li><a href="barang.html"><i class="fas fa-box"></i>
Barang</a></li>
    <li><a href="transaksi.html"><i class="fas fa-shopping-
cart"></i> Transaksi</a></li>
    <li><a href="report.html"><i class="fas fa-chart-line"></i>
Laporan</a></li>
  </ul>
</div>
```

### Penjelasan:

- Logo dan Judul:
  - Gambar Logo (img/logo.png) diikuti dengan teks judul: "minimarket-App".
- Hamburger Menu (<span class="nav-toggle">):
  - Menggunakan karakter &#9776; (≡) untuk ikon menu pada layar mobile.
- Menu Navigasi (<ul class="menu">):
  - Menu Link dengan ikon FontAwesome:
    - Dashboard: <i class="fas fa-home"></i>
    - Barang: <i class="fas fa-box"></i>
    - Transaksi: <i class="fas fa-shopping-cart"></i>
    - Laporan: <i class="fas fa-chart-line"></i>
- Navigasi Responsif:
  - Dapat berubah menjadi dropdown menu pada mobile view (dikendalikan oleh CSS dan JS).

## 3. Form Transaksi (<div id="form-transaksi">)

```
<div id="form-transaksi">
  <h2 id="judul"><i class="fas fa-edit"></i> Form Transaksi</h2>
  <form id="transaksiForm">
```

```
<select id="id_barang" required> </select>
<label for="stok">Jumlah barang:</label>
<input type="number" id="jumlah" placeholder="Jumlah" required>
<button type="submit" class="button btn-green">Tambah
Transaksi</button>
<button type="button" class="button btn-cancel"
onclick="window.location.href='transaksi.html'">Kembali</button>
</form>
</div>
```

### Penjelasan:

- **Judul Form:**
  - Menggunakan ikon edit (`<i class="fas fa-edit"></i>`) dengan teks "Form Transaksi".
- **Form Input:**
  - `<select id="id_barang">`:
    - Dropdown untuk memilih barang yang akan ditransaksikan.
    - Data barang akan diisi oleh JavaScript menggunakan AJAX dari database.
  - Jumlah Barang (`<input type="number" id="jumlah">`):
    - Input jumlah barang yang akan dibeli.
- **Button:**
  - Tambah Transaksi:
    - Tombol submit dengan class `.btn-green`.
  - Kembali:
    - Tombol cancel yang redirect ke halaman transaksi.html.

### 4. Daftar Transaksi (`<div id="display-transaksi">`)

```
<div id="display-transaksi">
  <h2><i class="fas fa-shopping-cart"></i> Daftar Transaksi</h2>
  <button class="button btn-add" id="addTransaksi">Mulai
Transaksi</button>
  <table>
    <thead>
      <tr>
        <th>Nama Barang</th>
        <th>Jumlah</th>
        <th>Total Harga</th>
```

```
        <th>Tanggal</th>
        <th>Aksi</th>
    </tr>
</thead>
<tbody id="transaksiTable">
    <!-- Data akan diisi oleh JavaScript -->
</tbody>
</table>
</div>
```

#### Penjelasan:

- **Judul Tabel:**
  - Menggunakan ikon keranjang belanja (`<i class="fas fa-shopping-cart"></i>`) dengan teks "Daftar Transaksi".
- **Button Mulai Transaksi:**
  - Memiliki ID `addTransaksi` untuk menampilkan form transaksi.
- **Tabel Transaksi:**
  - Kolom:
    - Nama Barang: Nama produk yang dibeli.
    - Jumlah: Kuantitas barang yang dibeli.
    - Total Harga: Harga total dari jumlah barang.
    - Tanggal: Waktu transaksi dilakukan.
    - Aksi: Tempat untuk tombol Edit dan Hapus.
  - Body Tabel (`<tbody id="transaksiTable">`):
    - Data transaksi akan diisi secara dinamis oleh JavaScript dari AJAX request.

#### 5. JavaScript Include

```
<script src="js/jquery-3.7.1.min.js"></script>
<script src="js/transaksi/transaksi.js"></script>
```

#### Penjelasan:

- `jquery-3.7.1.min.js`:
  - Memasukkan jQuery untuk mempermudah manipulasi DOM dan AJAX Request.
- `transaksi.js`:
  - File JavaScript kustom untuk mengelola:
    - Pengambilan data barang ke dropdown (`<select id="id_barang">`).
    - Menambah transaksi dan menghitung total harga.
    - Menampilkan daftar transaksi secara dinamis.



- Fungsi Edit dan Hapus transaksi.

RPL-SMAKN1KUNWUS

## E. SCRIPT JS (index.js)

Script selengkapnya dapat dijelaskan sebagai berikut:

### 1. Fungsi Utama:

```
document.querySelector('.nav-toggle').addEventListener('click',  
function () {  
    document.querySelector('.menu').classList.toggle('active');  
});
```

#### Penjelasan:

##### 1. document.querySelector('.nav-toggle'):

- Memilih elemen dengan class nav-toggle.
- Pada script HTML sebelumnya, elemen ini adalah ikon hamburger (≡) yang dipakai untuk toggle menu pada mobile view:

```
<span class="nav-toggle">&#9776;</span>
```

##### 2. .addEventListener('click', function() { ... }):

- Menambahkan event listener untuk event klik.
- Artinya, ketika pengguna mengklik ikon hamburger, fungsi yang berada di dalamnya akan dijalankan.

##### 3. document.querySelector('.menu'):

- Memilih elemen dengan class menu, yaitu daftar navigasi pada navbar:

```
<ul class="menu">  
    <li><a href="index.html"><i class="fas fa-home"></i>  
Dashboard</a></li>  
    <li><a href="barang.html"><i class="fas fa-box"></i>  
Barang</a></li>  
    <li><a href="transaksi.html"><i class="fas fa-shopping-cart"></i>  
Transaksi</a></li>  
    <li><a href="report.html"><i class="fas fa-chart-line"></i>  
Laporan</a></li>  
</ul>
```

##### .classList.toggle('active'):

- Menambah atau menghapus class active pada elemen .menu.
- Jika class active belum ada, maka akan ditambahkan.
- Jika class active sudah ada, maka akan dihapus.

- Hasilnya, menu akan muncul dan menghilang saat ikon hamburger diklik.

## 2. Contoh Penggunaan di CSS:

Agar fungsi ini bekerja, diperlukan **CSS tambahan** untuk mengatur tampilan menu saat active:

```
.menu {  
    display: none;  
}  
  
.menu.active {  
    display: block;  
}
```

### Penjelasan:

- **.menu { display: none; }:**
  - Secara default, menu disembunyikan (terutama untuk tampilan mobile).
- **.menu.active { display: block; }:**
  - Saat class active ditambahkan oleh **JavaScript**:
    - Menu akan ditampilkan (muncul sebagai dropdown atau slide).

## F. SCRIPT JS (barang.js)

Kode selengkapnya dapat dijelaskan sebagai berikut:

```
const myTable = document.getElementById("display-barang");
const myEdit = document.getElementById("form-barang");
const myJudul = document.getElementById("judul");

// Active menu Responsive
document.querySelector('.nav-toggle').addEventListener('click',
function () {
    document.querySelector('.menu').classList.toggle('active');
});

$(document).ready(function() {
    loadData();

    $("#addBarang").click(function() {
        myTable.style.display = "none";
        myEdit.style.display = "block";
        myJudul.innerHTML = "Tambah Barang" ;
    });

    $("#barangForm").submit(function(e) {
        e.preventDefault();
        let id_barang = $("#id_barang").val();
        let nama_barang = $("#nama_barang").val();
        let kategori = $("#kategori").val();
        let harga = $("#harga").val();
        let stok = $("#stok").val();
        let diskon = $("#diskon").val();

        let url = id_barang ? "php/barang/update.php" :
"php/barang/create.php";

        $.post(url, { id_barang, nama_barang, kategori, harga, stok,
diskon }, function(response) {
            if (response === "success") {
                alert("Data tersimpan!")
                $("#barangForm")[0].reset();
                $("#id_barang").val("");
                loadData();
            }
        });
    });
});
```

```
        } else {
            alert("Gagal menyimpan data");
        }
    });
});

function loadData() {
    myTable.style.display = "block";
    myEdit.style.display = "none";
    $.getJSON("php/barang/read.php", function(data) {
        console.log(data);
        let rows = "";
        data.forEach(item => {
            rows += `<tr>
                <td>${item.nama_barang}</td>
                <td>${item.kategori}</td>
                <td>${item.harga}</td>
                <td>${item.stok}</td>
                <td>${item.diskon}</td>
                <td>
                    <button
                        title="Edit"
                        class="btn btn-edit"
                        onclick='editData(${JSON.stringify(item)})'
                    >
                        <i class="fas fa-edit"></i>
                    </button>
                    <button
                        title="Delete"
                        class="btn btn-delete"
                        onclick='deleteData(${item.id_barang})'>
                        <i class="fas fa-trash"> </i>
                    </button>
                </td>
            </tr>`;
        });
        $("#barangTable").html(rows);
    });
}

window.editData = function(item) {
```

```
myTable.style.display = "none";
myEdit.style.display = "block";
myJudul.innerHTML = "Update Barang" ;

$("#id_barang").val(item.id_barang);
$("#nama_barang").val(item.nama_barang);
$("#kategori").val(item.kategori);
$("#harga").val(item.harga);
$("#stok").val(item.stok);
$("#diskon").val(item.diskon);
};

window.deleteData = function(id_barang) {
    if (confirm("Yakin ingin menghapus?")) {
        $.post("php/barang/delete.php", { id_barang },
function(response) {
            if (response === "success") {
                loadData();
            } else {
                alert("Gagal menghapus data");
            }
        });
    }
};
});
```

Script JavaScript/jQuery di atas bertujuan untuk mengelola halaman produk pada aplikasi MiniMarket, yang mencakup:

1. Menampilkan daftar produk.
2. Menambah, mengedit, dan menghapus produk.
3. Menggunakan AJAX untuk berkomunikasi dengan backend (PHP).

Berikut adalah penjelasan rinci dan terstruktur dari masing-masing bagian kode:

### 1. Deklarasi Variabel DOM:

```
const myTable = document.getElementById("display-barang");
const myEdit = document.getElementById("form-barang");
const myJudul = document.getElementById("judul");
```

Penjelasan:

- **myTable:** Memilih elemen `<div id="display-barang">`, yang menampilkan **daftar produk**.
- **myEdit:** Memilih elemen `<div id="form-barang">`, yang berisi **formulir tambah/edit produk**.
- **myJudul:** Memilih elemen `<h2 id="judul">`, untuk **mengubah judul** saat tambah atau edit.

## 2. Menu Responsive:

```
document.querySelector('.nav-toggle').addEventListener('click',  
function () {  
    document.querySelector('.menu').classList.toggle('active');  
});
```

### Penjelasan:

- Toggle class active pada `.menu` saat ikon hamburger (`.nav-toggle`) diklik.
- Berfungsi untuk **menu responsif** pada tampilan mobile.

## 3. Document Ready dengan jQuery:

```
$(document).ready(function() {
```

- Menunggu hingga **DOM sepenuhnya dimuat** sebelum menjalankan kode di dalamnya.
- Memastikan semua elemen **siap digunakan** oleh JavaScript/jQuery.

## 4. Memuat Data Produk:

```
loadData();
```

### Penjelasan:

- Memanggil fungsi `loadData()` untuk **menampilkan daftar produk** saat halaman pertama kali dimuat.

## 5. Fungsi `loadData()` - Membaca Data dari Database:

```
function loadData() {  
    myTable.style.display = "block";  
    myEdit.style.display = "none";  
  
    $.getJSON("php/barang/read.php", function(data) {  
        console.log(data);  
        let rows = "";  
        data.forEach(item => {  
            rows += `<tr>  
                <td>${item.nama_barang}</td>
```

```
        <td>${item.kategori}</td>
        <td>${item.harga}</td>
        <td>${item.stok}</td>
        <td>${item.diskon}</td>
        <td>
            <button title="Edit" class="btn btn-edit"
                onclick='editData(${JSON.stringify(item)})'>
                <i class="fas fa-edit"></i>
            </button>
            <button title="Delete" class="btn btn-delete"
                onclick='deleteData(${item.id_barang})'>
                <i class="fas fa-trash"> </i>
            </button>
        </td>
    </tr>`;
});
$("#barangTable").html(rows);
});
}
```

#### Penjelasan:

- Menampilkan myTable dan menyembunyikan myEdit untuk fokus pada daftar produk.
- Mengambil data produk dari php/barang/read.php menggunakan AJAX (JSON).
- Membangun baris tabel secara dinamis dengan forEach() dari data yang diterima.
- Menyematkan tombol Edit dan Delete dengan event handler:
  - editData(): Mengirimkan seluruh data produk ke form untuk diedit.
  - deleteData(): Mengirimkan id\_barang untuk dihapus.

#### 6. Tombol Tambah Produk:

```
$("#addBarang").click(function() {
    myTable.style.display = "none";
    myEdit.style.display = "block";
    myJudul.innerHTML = "Tambah Barang";
});
```

#### Penjelasan:

- Saat tombol "Tambah Produk" diklik:
  - Sembunyikan tabel (myTable).



- Tampilkan form tambah (myEdit).
- Ubah judul menjadi "Tambah Barang".

## 7. Fungsi editData(item) - Edit Produk:

```
window.editData = function(item) {  
    myTable.style.display = "none";  
    myEdit.style.display = "block";  
    myJudul.innerHTML = "Update Barang";  
  
    $("#id_barang").val(item.id_barang);  
    $("#nama_barang").val(item.nama_barang);  
    $("#kategori").val(item.kategori);  
    $("#harga").val(item.harga);  
    $("#stok").val(item.stok);  
    $("#diskon").val(item.diskon);  
};
```

### Penjelasan:

- Mengisi form dengan data produk yang ingin diedit.
- Mengubah judul menjadi "Update Barang".
- Form ditampilkan, dan tabel disembunyikan.

## 8. Fungsi deleteData(id\_barang) - Hapus Produk:

```
window.deleteData = function(id_barang) {  
    if (confirm("Yakin ingin menghapus?")) {  
        $.post("php/barang/delete.php", { id_barang },  
function(response) {  
            if (response === "success") {  
                loadData();  
            } else {  
                alert("Gagal menghapus data");  
            }  
        });  
    }  
};
```

### Penjelasan:

- Menampilkan konfirmasi sebelum menghapus.
- Jika dikonfirmasi, mengirim request ke delete.php dengan id\_barang.

- Jika sukses, `loadData()` dipanggil ulang untuk memperbarui daftar produk.

## 9. Form Submit - Tambah/Update Produk:

```
$("#barangForm").submit(function(e) {
    e.preventDefault();

    let id_barang = $("#id_barang").val();
    let nama_barang = $("#nama_barang").val();
    let kategori = $("#kategori").val();
    let harga = $("#harga").val();
    let stok = $("#stok").val();
    let diskon = $("#diskon").val();

    let url = id_barang ? "php/barang/update.php" :
    "php/barang/create.php";

    $.post(url, { id_barang, nama_barang, kategori, harga, stok, diskon }, function(response) {
        if (response === "success") {
            alert("Data tersimpan!");
            $("#barangForm")[0].reset();
            $("#id_barang").val("");
            loadData();
        } else {
            alert("Gagal menyimpan data");
        }
    });
});
```

### Penjelasan:

- Menangani submit form untuk tambah dan update produk.
- Mencegah reload halaman dengan `e.preventDefault()`.
- Mengambil data dari input form.
- Menentukan URL berdasarkan `id_barang`:
  - Jika ada `id_barang`: Update (`update.php`).
  - Jika kosong: Tambah (`create.php`).
- Mengirim data dengan AJAX `$.post()`.
- Jika sukses:
  - Menampilkan alert "Data tersimpan!".
  - Mereset form.
  - Memanggil `loadData()` untuk memperbarui daftar produk.

**10. Alur Kerja Secara Keseluruhan:**

1. Halaman dimuat, `loadData()` dijalankan untuk menampilkan daftar produk.
2. Klik "Tambah Produk":
  - Form Tambah Barang ditampilkan.
3. Klik "Edit":
  - Form Update Barang ditampilkan dengan data terisi otomatis.
4. Klik "Delete":
  - Konfirmasi sebelum menghapus, kemudian hapus produk.
5. Submit Form:
  - Tambah atau Update produk menggunakan AJAX.
  - Reload daftar produk tanpa refresh halaman.

## G. SCRIPT JS (transaksi.js)

Kode selengkapnya dapat dijelaskan sebagai berikut:

```
const myTable = document.getElementById("display-transaksi");
const myForm = document.getElementById("form-transaksi");

// Active menu Responsive
document.querySelector('.nav-toggle').addEventListener('click',
function () {
    document.querySelector('.menu').classList.toggle('active');
});

$(document).ready(function() {
    loadBarang();
    loadTransaksi();

    $("#addTransaksi").click(function() {
        myTable.style.display = "none";
        myForm.style.display = "block";
    });

    $("#transaksiForm").submit(function(e) {
        e.preventDefault();
        let id_barang = $("#id_barang").val();
        let jumlah = $("#jumlah").val();

        $.post("php/transaksi/create_transaksi.php", { id_barang,
jumlah }, function(response) {
            if (response === "success") {
                $("#transaksiForm")[0].reset();
                loadTransaksi();
            } else if (response === "barang_not_found") {
                alert("Barang tidak ditemukan!");
            } else {
                alert("Gagal menambahkan transaksi!");
            }
        });
    });

    function loadBarang() {
        $.getJSON("php/barang/read.php", function(data) {
```

```
        let options = '<option value="">Pilih Barang</option>';
        data.forEach(item => {
            options += `<option
value="${item.id_barang}">${item.nama_barang}</option>`;
        });
        $("#id_barang").html(options);
    });
}

function loadTransaksi() {
    myTable.style.display = "block";
    myForm.style.display = "none";
    $.getJSON("php/transaksi/read_transaksi.php", function(data) {
        let rows = "";
        data.forEach(item => {
            rows += `<tr>
                <td>${item.nama_barang}</td>
                <td>${item.jumlah}</td>
                <td>${item.total_harga}</td>
                <td>${item.tanggal_transaksi}</td>
                <td>
                    <button
                        class="btn btn-delete"
                        onclick="deleteTransaksi(${item.id_transaks
i})">
                        <i class="fas fa-trash"> </i>
                    </button>
                </td>
            </tr>`;
        });
        $("#transaksiTable").html(rows);
    });
}

window.deleteTransaksi = function(id_transaksi) {
    if (confirm("Yakin ingin menghapus transaksi?")) {
        $.post("php/transaksi/delete_transaksi.php", { id_transaksi
    }, function(response) {
        if (response === "success") {
            loadTransaksi();
        } else {
```

```
        alert("Gagal menghapus transaksi!");
    }
    });
}
};
});
```

Script JavaScript/jQuery di atas digunakan untuk **mengelola transaksi** di aplikasi MiniMarket, yang mencakup:

1. Menampilkan daftar transaksi.
2. Menambah transaksi baru.
3. Menghapus transaksi.
4. Menggunakan AJAX untuk berkomunikasi dengan backend (PHP).

Berikut adalah penjelasan lengkap dan terstruktur dari masing-masing bagian kode:

### 1. Deklarasi Variabel DOM:

```
const myTable = document.getElementById("display-transaksi");
const myForm = document.getElementById("form-transaksi");
```

Penjelasan:

- **myTable**: Memilih elemen `<div id="display-transaksi">`, yang berfungsi untuk **menampilkan daftar transaksi**.
- **myForm**: Memilih elemen `<div id="form-transaksi">`, yang berfungsi untuk **menampilkan form tambah transaksi**.

### 2. Menu Responsive:

```
document.querySelector('.nav-toggle').addEventListener('click',
function () {
    document.querySelector('.menu').classList.toggle('active');
});
```

Penjelasan:

- **Toggle class active** pada `.menu` saat ikon hamburger (`.nav-toggle`) diklik.
- Berfungsi untuk **menu responsif** pada tampilan mobile.

### 3. Document Ready dengan jQuery:

```
$(document).ready(function() {
```

- Menunggu hingga **DOM sepenuhnya dimuat** sebelum menjalankan kode di dalamnya.
- Memastikan semua elemen siap digunakan oleh JavaScript/jQuery.

### 4. Memuat Data Barang dan Transaksi:

```
loadBarang();  
loadTransaksi();
```

Penjelasan:

- **loadBarang()**: Digunakan untuk mengisi select barang pada form transaksi.
- **loadTransaksi()**: Digunakan untuk menampilkan daftar transaksi saat halaman pertama kali dimuat.

### 5. Fungsi loadBarang() - Mengisi Dropdown Barang:

```
function loadBarang() {  
    $.getJSON("php/barang/read.php", function(data) {  
        let options = '<option value="">Pilih Barang</option>';  
        data.forEach(item => {  
            options += `<option  
value="${item.id_barang}">${item.nama_barang}</option>`;  
        });  
        $("#id_barang").html(options);  
    });  
}
```

Penjelasan:

- Mengambil data barang dari `php/barang/read.php` menggunakan **AJAX (JSON)**.
- Membangun **option dropdown** untuk memilih barang saat menambah transaksi.
- Menggunakan **forEach()** untuk menambah **option** berdasarkan data yang diterima.

### 6. Tombol Mulai Transaksi:

```
$("#addTransaksi").click(function() {  
    myTable.style.display = "none";  
    myForm.style.display = "block";  
});
```

**Penjelasan:**

- Saat tombol "Mulai Transaksi" diklik:
  - Sembunyikan tabel (myTable).
  - Tampilkan form (myForm) untuk menambah transaksi baru.

**7. Form Submit- Tambah Transaksi:**

```
$("#transaksiForm").submit(function(e) {  
    e.preventDefault();  
    let id_barang = $("#id_barang").val();  
    let jumlah = $("#jumlah").val();  
  
    $.post("php/transaksi/create_transaksi.php", { id_barang, jumlah },  
function(response) {  
    if (response === "success") {  
        $("#transaksiForm")[0].reset();  
        loadTransaksi();  
    } else if (response === "barang_not_found") {  
        alert("Barang tidak ditemukan!");  
    } else {  
        alert("Gagal menambahkan transaksi!");  
    }  
    });  
});
```

**Penjelasan:**

- Menangani submit form untuk menambah transaksi baru.
- Mencegah reload halaman dengan e.preventDefault().
- Mengambil data dari input form:
  - id\_barang → Barang yang dipilih.
  - jumlah → Jumlah barang yang dibeli.
- Mengirim data dengan AJAX \$.post() ke create\_transaksi.php.
- Respon yang ditangani:
  - success → Transaksi berhasil disimpan:
    - Form di-reset.
    - Daftar transaksi diperbarui dengan loadTransaksi().
  - barang\_not\_found → Menampilkan alert barang tidak ditemukan.
  - Lainnya → Menampilkan alert gagal menambah transaksi.

**8. Fungsi loadTransaksi() - Membaca Data Transaksi:**

```
function loadTransaksi() {
```



```
myTable.style.display = "block";
myForm.style.display = "none";

$.getJSON("php/transaksi/read_transaksi.php", function(data) {
    let rows = "";
    data.forEach(item => {
        rows += `<tr>
            <td>${item.nama_barang}</td>
            <td>${item.jumlah}</td>
            <td>${item.total_harga}</td>
            <td>${item.tanggal_transaksi}</td>
            <td>
                <button
                    class="btn btn-delete"
                    onclick="deleteTransaksi(${item.id_transaksi})"
                >
                    <i class="fas fa-trash"> </i>
                </button>
            </td>
        </tr>`;
    });
    $("#transaksiTable").html(rows);
});
}
```

#### Penjelasan:

- Menampilkan tabel transaksi (myTable) dan menyembunyikan form (myForm).
- Mengambil data transaksi dari php/transaksi/read\_transaksi.php menggunakan AJAX (JSON).
- Membangun baris tabel secara dinamis dengan forEach() dari data yang diterima:
  - nama\_barang → Nama barang yang dibeli.
  - jumlah → Jumlah barang yang dibeli.
  - total\_harga → Total harga (jumlah × harga per barang).
  - tanggal\_transaksi → Waktu transaksi dilakukan.
  - Tombol deleteTransaksi() untuk menghapus transaksi.

#### 9. Fungsi deleteTransaksi(id\_transaksi) - Hapus Transaksi:

```
window.deleteTransaksi = function(id_transaksi) {
    if (confirm("Yakin ingin menghapus transaksi?")) {
```

```
$.post("php/transaksi/delete_transaksi.php", { id_transaksi },  
function(response) {  
    if (response === "success") {  
        loadTransaksi();  
    } else {  
        alert("Gagal menghapus transaksi!");  
    }  
});  
};
```

**Penjelasan:**

- Menampilkan konfirmasi sebelum menghapus.
- Jika dikonfirmasi, mengirim request ke delete\_transaksi.php dengan id\_transaksi.
- Jika sukses, loadTransaksi() dipanggil ulang untuk memperbarui daftar transaksi.
- Jika gagal, menampilkan alert "Gagal menghapus transaksi!".

---

**10. Alur Kerja Secara Keseluruhan:**

1. Halaman dimuat, loadBarang() dan loadTransaksi() dijalankan untuk mengisi dropdown dan menampilkan daftar transaksi.
2. Klik "Mulai Transaksi":
  - Form Tambah Transaksi ditampilkan.
3. Submit Form:
  - Menambah transaksi baru menggunakan AJAX.
  - Reload daftar transaksi tanpa refresh halaman.
4. Klik "Hapus":
  - Konfirmasi sebelum menghapus, kemudian hapus transaksi.