

**LAPORAN HASIL PENGUJIAN
APLIKASI TOKO SEDERHANA**

PERIODE: SEMESTER GENAP 2024/2025



NAMA KELOMPOK:

- | | |
|----------------------------|---------------|
| 1. ALFINO KURNIAWAN | (24104410012) |
| 2. NESSA NOVIANTI | (24104410007) |
| 3. ENSTIKA ELINDASARI | (24104410006) |
| 4. MUHAMMAD NAUFAL FITRA S | (24104410003) |
| 5. FAIZAL RAMADHAN | (24104410033) |

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM BALITAR
2024/2025**

KATA PENGANTAR

Puji dan syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa karena atas rahmat dan karunia-Nya, penulis dapat menyelesaikan laporan ini yang berjudul "Laporan Pembuatan Aplikasi Toko Sederhana" dengan lancar dan tepat waktu.

Laporan ini disusun sebagai bentuk dokumentasi dan pertanggungjawaban atas proses perancangan dan pembuatan aplikasi kasir toko yang bertujuan untuk membantu mempermudah proses transaksi penjualan, pencatatan stok barang, serta pencetakan struk pada suatu toko. Aplikasi ini dirancang agar dapat digunakan dengan mudah oleh pemilik usaha kecil maupun menengah sebagai solusi digitalisasi sistem kasir.

Penulis menyadari bahwa dalam penyusunan laporan ini tidak terlepas dari bimbingan, dukungan, dan bantuan dari berbagai pihak. Oleh karena itu, penulis menyampaikan ucapan terima kasih kepada:

1. Sri Lestanti, S.kom,M,t selaku pembimbing yang telah memberikan arahan dan masukan selama proses pembuatan aplikasi dan penulisan laporan ini.
2. Mareta Kusuma Ayu yang telah memberikan kesempatan dan fasilitas dalam pelaksanaan proyek ini.
3. Teman-teman dan semua pihak yang telah membantu secara langsung maupun tidak langsung dalam menyelesaikan laporan ini.

Penulis menyadari bahwa laporan ini masih jauh dari sempurna. Oleh karena itu, kritik dan saran yang membangun sangat penulis harapkan guna penyempurnaan laporan di masa yang akan datang.

Akhir kata, semoga laporan ini dapat bermanfaat bagi pembaca dan pihak-pihak yang membutuhkan.

Blitar, 1 Juli 2025

Kelompok 4

DAFTAR ISI

KATA PENGANTAR.....	ii
DAFTAR ISI.....	iii
BAB I.....	v
PENDAHULUAN	v
1.1 Latar Belakang	v
1.2 Rumusan Masalah.....	v
1.3 Tujuan	v
1.4 Ruang Lingkup.....	v
BAB II	1
LANDASAN TEORI	1
2.1 Bahasa CSS	1
2.2 Bahasa HTML.....	1
2.3 Bahasa JS	2
BAB III.....	3
PELAKSANAAN OBSERVASI.....	3
3.1 TUJUAN	3
3.2 LOKASI OBSERVASI.....	3
3.3 RANCANGAN APLIKASI.....	3
3.4 PENJELASAN FITUR APLIKASI.....	3
3.5 KOMPONEN LOGIKA YANG DIGUNAKAN.....	3
3.6 SIMULASI.....	4
BAB IV	5
HASIL APLIKASI.....	5
4.1 Identitas Aplikasi	5
4.2 Deskripsi Singkat Aplikasi	5
4.3 Fitur-Fitur Utama	5
4.5 Kekurangan dan Saran Perbaikan	5
4.7 Hasil Pengujian	5
4.8 Implementasi Penggunaan Aplikasi.....	6
4.9 Coding.....	6
E. Tampilan Aplikasi	14
J. Kesimpulan.....	15
BAB IV	vi
PENUTUP.....	vi

4.1 Kesimpulan	vi
4.2 Saran	vi
DAFTAR PUSTAKA.....	vii

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan perkembangan teknologi informasi yang semakin pesat, kebutuhan akan sistem yang dapat membantu efisiensi operasional usaha menjadi semakin penting. Salah satu sistem yang kini banyak dibutuhkan oleh pelaku usaha, khususnya di sektor perdagangan, adalah aplikasi kasir. Sistem kasir konvensional yang masih menggunakan pencatatan manual seringkali menimbulkan kendala seperti kesalahan perhitungan, kesulitan dalam mengelola stok barang, dan lamanya proses pembuatan laporan penjualan.

Oleh karena itu, dibutuhkan sebuah solusi digital berupa aplikasi kasir toko yang mampu mendukung aktivitas transaksi secara lebih cepat, akurat, dan efisien. Aplikasi kasir ini tidak hanya bertugas mencatat penjualan, tetapi juga mengelola data barang, stok, dan menyajikan laporan keuangan secara otomatis.

Melalui laporan ini, penulis ingin memaparkan proses perancangan dan pembuatan aplikasi kasir toko yang dirancang untuk menjawab kebutuhan tersebut, khususnya bagi pelaku usaha mikro, kecil, dan menengah (UMKM).

1.2 Rumusan Masalah

Adapun permasalahan yang melatarbelakangi pembuatan aplikasi ini adalah sebagai berikut:

- Bagaimana merancang aplikasi kasir toko yang mudah digunakan dan efisien dalam membantu proses transaksi penjualan?
- Bagaimana sistem ini dapat mengelola data barang, stok, dan cetak struk?
- Apa saja fitur yang dibutuhkan untuk mendukung operasional toko secara optimal?

1.3 Tujuan

Tujuan dari pembuatan aplikasi kasir toko ini antara lain:

- Membuat sebuah aplikasi kasir yang dapat mempermudah proses transaksi penjualan di toko.
- Menyediakan sistem pengelolaan stok barang dan laporan keuangan secara otomatis.
- Menghadirkan antarmuka pengguna yang sederhana, efektif, dan mudah dipahami oleh pengguna non-teknis.

1.4 Ruang Lingkup

Aplikasi kasir yang dibuat dalam proyek ini memiliki ruang lingkup sebagai berikut:

- Mengelola jenis produk dan stok barang.
- Aplikasi bersifat desktop/mobile/web (sesuaikan sesuai platform yang digunakan).
- Tidak mencakup sistem pembayaran online atau integrasi dengan perangkat keras seperti barcode scanner (jika tidak disertakan)

BAB II

LANDASAN TEORI

2.1 Bahasa CSS

a. Pengertian

CSS adalah *Cascading Style Sheet*. CSS digunakan untuk memberikan warna, *style*, ukuran, dan juga digunakan untuk mengatur posisi pada elemen-elemen di halaman website, sehingga membuat halaman website tampak lebih menarik saat dilihat. CSS bukanlah bahasa pemrograman. CSS adalah bahasa *style sheet* yang dirancang untuk memungkinkan pemisahan antara presentasi dan konten, sehingga memungkinkan beberapa halaman website menggunakan kode CSS yang sama, namun menampilkan tampilan yang berbeda, sesuai kode yang telah dibuat.

b. Fungsi

1. Mempercepat Loading halaman website
2. Memudahkan Pengelolaan Kode Program
3. Memberikan Variasi pada tampilan halaman website
4. Membuat halaman website tampil rapi

c. Jenis CSS

1. Inline CSS
2. Internal CSS
3. External CSS

2.2 Bahasa HTML

a. Pengertian

HTML adalah kependekan dari *Hypertext Markup Language* yang merupakan sebuah bahasa *markup*. HTML adalah kode untuk membuat struktur halaman suatu *website* yang menarik, saling terhubung satu dengan yang lainnya, dan yang pasti dapat diakses melalui internet.

Awalnya html ditemukan oleh Tim Berners-Lee pada tahun 1991. HTML adalah solusi untuk membantu ilmuwan dalam mengakses dokumen, namun kini html semakin berkembang pesat di dunia pemrograman web.

Bagi pemula yang ingin memulai belajar pemrograman web dan bingung belajar bahasa markup apa, HTML adalah jawabannya. HTML dianggap mudah dipahami oleh orang awam karena tag dasar html mudah dibaca dan diimplementasikan.

b. Fungsi

1. Berfungsi untuk membuat website
2. Berfungsi sebagai Hyperlink
3. Sebagai pondasi website
4. Untuk menambahkan multimedia pada website
5. Sebagai penanda teks dan bagian website

2.3 Bahasa JS

a. Pengertian

JavaScript adalah bahasa pemrograman yang dipakai untuk meningkatkan interaktivitas website. Bahasa ini dirancang agar pengembang dapat menambahkan fitur-fitur dinamis seperti pop up, slideshow, efek hover, hingga aplikasi web yang tergolong kompleks. Bahasa pemrograman ini sendiri pertama kali diperkenalkan oleh Brendan Eich pada tahun 1995 di Netscape Communications. Awalnya, bahasa ini memiliki nama “LiveScript” dan ditujukan untuk penggunaan web browser internal perusahaan saja. Namun, JavaScript mengalami peningkatan bersama dengan browser populer lain seperti Chrome dan Firefox. Bahkan, saat ini JavaScript masih terus dikembangkan untuk mengkompilasi kode bit menjadi kode mesin asli, yakni JavaScript V8.

b. Fungsi

1. Manipulasi DOM
2. Membuat website interaktif
3. Pengembangan aplikasi web
4. Pengembangan backend
5. Pembuatan game berbasis web

BAB III

PELAKSANAAN OBSERVASI

3.1 TUJUAN

Tujuan dari observasi ini ialah:

- Mengidentifikasi kebutuhan nyata di lingkungan sekitar
- Membuat aplikasi dengan fitur dasar seperti pemilihan, pengulangan, fungsi, dan sorting

3.2 LOKASI OBSERVASI

Tempat Observasi : Toko Oleh-Oleh

Tanggal Observasi : 25 Mei 2025

Permasalahan : Perhitungan manual sering keliru dan memakan waktu

3.3 RANCANGAN APLIKASI

Kami akan membuat rancangan aplikasi toko sederhana menggunakan dengan fitur yang diminta: pemilihan (if/else), pengulangan (loop), fungsi/prosedur, fungsi `hitung_total()`, dan sorting. Aplikasi ini digunakan untuk membantu toko dalam menghitung total belanja pelanggan, menerapkan diskon, serta menampilkan daftar produk secara terurut berdasarkan harga.

3.4 PENJELASAN FITUR APLIKASI

- Menampilkan daftar produk oleh-oleh (sudah di-sorting berdasarkan harga).
- Memungkinkan pengguna memilih produk dan jumlah yang ingin dibeli.
- Menghitung total belanja menggunakan fungsi khusus.
- Memberikan diskon 10% jika total belanja lebih dari Rp100.000 (menggunakan if/else).
- Menampilkan struk belanja akhir.

3.5 KOMPONEN LOGIKA YANG DIGUNAKAN

- Pemilihan (if/else): Digunakan dalam fungsi `hitung_total()` untuk menentukan diskon berdasarkan total belanja
- Pengulangan (loop): Digunakan untuk memproses daftar belanja/pembelian
- Fungsi/prosedur: Beberapa fungsi dibuat, seperti `tampilkan_produk()`, `hitung_total()`, dan `cetak_struk()`.
- Sorting: Untuk mengurutkan Produk berdasarkan harga menggunakan `sorted()`

3.6 SIMULASI

1. Aplikasi menambahkan produk
2. menampilkan daftar produk
3. User memilih produk dan memasukkan jumlah yang ingin dibeli.
4. Proses bisa diulang untuk beberapa produk.
5. Aplikasi menghitung total harga.
6. Diskon diberikan bila memenuhi syarat.
7. Struk ditampilkan di akhir.

BAB IV

HASIL APLIKASI

4.1 Identitas Aplikasi

Nama Aplikasi: Toko Sederhana

Platform: Aplikasi Web

Bahasa Pemrograman: Java Script

4.2 Deskripsi Singkat Aplikasi

Aplikasi ini merupakan toko daring sederhana yang menampilkan berbagai produk makanan/minuman beserta harga dan tombol pembelian. Aplikasi ini bertujuan mempermudah pelanggan untuk melihat dan memilih produk yang ditawarkan oleh toko.

4.3 Fitur-Fitur Utama

1. Tampilan Produk: Gambar produk, nama, dan harga ditampilkan dalam grid responsif
2. Tombol Beli: Setiap produk memiliki tombol "Beli"
3. Desain Responsif: Layout menyesuaikan perangkat (mobile dan desktop)
4. Notifikasi Interaktif (Alert): Saat tombol beli diklik, muncul alert pembelian berhasil

4.4 Kelebihan Aplikasi

1. Tampilan minimalis dan bersih
2. Mudah digunakan (User friendly)
3. Cocok sebagai prototype toko daring

4.5 Kekurangan dan Saran Perbaikan

1. Data produk statis (hardcoded)
Saran: Gunakan backend atau file JSON untuk dinamisasi data
2. Tidak terdapat fitur keranjang belanja
Saran: Tambahkan fitur cart dan checkout sederhana
4. Tombol beli hanya memunculkan alert
Saran: Buat fungsi beli menyimpan transaksi atau terhubung ke database

4.6 Tempat Pengujian

Tempat: Toko Oleh-oleh Arjuna

Alamat: Jl.Ir Soekarno No.177, Blitar

Pemilik Toko: Mareta Kusuma Ayu

4.7 Hasil Pengujian

1. Responsif (Desktop): Baik - Grid layout rapi dan mudah dibaca
2. Tombol Beli: Baik - Alert muncul dengan benar saat tombol ditekan
3. Navigasi: Cukup - Tidak ada menu navigasi, hanya 1 halaman utama
4. Keamanan: Rendah - Tidak ada sistem login atau validasi data

4.8 Implementasi Penggunaan Aplikasi



4.9 Coding

a. Connection Js

```
1 // index.js
2 const express = require("express");
3 const mysql = require("mysql2");
4 const multer = require("multer");
5 const path = require("path");
6
7 const app = express();
8 app.use(express.json());
9 app.use(express.urlencoded({ extended: true }));
10 app.use(express.static("public")); // Untuk menyajikan file statis seperti CSS dan gambar
11
12 // Konfigurasi upload gambar
13 const storage = multer.diskStorage({
14   destination: "./public/uploads/",
15   filename: (req, file, cb) => {
16     cb(null, Date.now() + "-" + file.originalname);
17   }
18 });
19 const upload = multer({ storage: storage });
20
21 // Konfigurasi koneksi MySQL
22 const db = mysql.createConnection({
23   host: "localhost",
24   database: "tokosederhana",
25   user: "root",
26   password: "",
27 });
```

```

29 db.connect((err) => {
30   if (err) throw err;
31   console.log("Database connected...");
32   // Buat tabel products jika belum ada
33   db.query(`
34     CREATE TABLE IF NOT EXISTS products (
35       id INT AUTO_INCREMENT PRIMARY KEY,
36       name VARCHAR(100),
37       price INT,
38       stock INT,
39       image VARCHAR(255)
40     )
41   `);
42 });

```

```

44 // Fungsi untuk menambahkan produk
45 function addProductToDB(name, price, stock, imageUrl, callback) {
46   const sql = "INSERT INTO products (name, price, stock, image) VALUES (?, ?, ?, ?)";
47   db.query(sql, [name, price, stock, imageUrl], (err, result) => {
48     if (err) return callback(err);
49     callback(null, result);
50   });
51 }
52
53 // Fungsi untuk mengambil produk
54 function getProductsFromDB(callback) {
55   db.query("SELECT * FROM products", (err, results) => {
56     if (err) return callback(err);
57     callback(null, results);
58   });
59 }
60
61 // Endpoint untuk mengambil daftar produk
62 app.get("/api/products", (req, res) => {
63   getProductsFromDB((err, results) => {
64     if (err) {
65       res.status(500).send("Error retrieving products");
66       return;
67     }
68     res.json(results);
69   });
70 });
71
72 // Endpoint untuk menambahkan produk
73 app.post("/api/add-product", upload.single("image"), (req, res) => {
74   const { name, price, stock } = req.body;
75   const imageUrl = req.file ? `/uploads/${req.file.filename}` : "/uploads/default.jpg";
76   addProductToDB(name, parseInt(price), parseInt(stock), imageUrl, (err, result) => {
77     if (err) {
78       res.status(500).send("Error adding product: " + err.message);
79       return;
80     }
81     res.json({ message: "Product added successfully", id: result.insertId });
82   });
83 });
84 app.get("/", (req, res) => {
85   console.log("Route / accessed");
86   res.send(`
87   `);
88 });
89
90 app.listen(8000, () => {
91   console.log("Server ready on port 8000...");
92 });
93

```

A. HTML

```

Welcome X JS Connection.js {} package.json {} package-lock.json < index.html X JS script
public > < index.html > html > body.bg-gray-100.font-sans > div.min-h-screen.flex.items-center.justify-center >
1 <!DOCTYPE html>
2 <html lang="id">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Aplikasi Toko Sederhana</title>
7   <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
8   <script src="https://cdn.tailwindcss.com"></script>
9   <style>
10     .product-img { width: 150px; height: 100px; object-fit: cover; }
11     #receipt { display: none; }
12     .receipt-container {
13       font-family: 'Courier New', Courier, monospace;
14       max-width: 300px;
15       border: 1px solid #000;
16       padding: 10px;
17       line-height: 1.2;
18       margin: 0 auto;
19     }

```

```

20 .receipt-header { text-align: center; font-size: 14px; }
21 .receipt-line { border-top: 1px dashed #000; margin: 8px 0; }
22 .receipt-item { display: flex; justify-content: space-between; font-size: 12px; }
23 .receipt-total { font-weight: bold; font-size: 14px; }
24 @media print {
25   body, body * { visibility: hidden; }
26   #receipt, #receipt * { visibility: visible; display: block !important; }
27   #receipt {
28     position: absolute;
29     left: 0;
30     top: 0;
31     width: 100%;
32     height: auto;
33   }
34   .receipt-container {
35     border: none;
36     box-shadow: none;
37     margin: 0;
38     padding: 10px;
39     width: 100%;
40     max-width: 300px;
41   }
42   @page { margin: 0; }
43 }
44

```

```

44 </style>
45 </head>
46 <body class="bg-gray-100 font-sans">
47   <div class="min-h-screen flex items-center justify-center">
48     <div class="container mx-auto p-6">
49       <header id="appHeader" class="bg-blue-600 text-white p-4 shadow-md rounded-lg">
50         <div class="container mx-auto">
51           <h1 class="text-2xl font-bold text-center">Aplikasi Toko Sederhana</h1>
52         </div>
53       </header>
54       <div class="input-section id="loginSection">
55         <div class="card mt-10 max-w-md mx-auto bg-white p-6 rounded-lg shadow-md">
56           <h2 class="text-xl font-semibold mb-4 text-center">Login</h2>
57           <div class="grid grid-cols-1 gap-4">
58             <div>
59               <label class="block text-sm font-medium text-gray-700">Username</label>
60               <input type="text" id="username" placeholder="Masukkan Username" class="mt-1 block w-full border-gray-300 rounded-md shadow-sm focus:ring-blue-500 focus:outline-none">
61             </div>
62             <div>
63               <label class="block text-sm font-medium text-gray-700">Password</label>
64               <input type="password" id="password" placeholder="Masukkan Password" class="mt-1 block w-full border-gray-300 rounded-md shadow-sm focus:ring-blue-500 focus:outline-none">
65             </div>
66             <button onclick="login()" class="mt-4 bg-green-600 text-white px-4 py-2 rounded-md hover:bg-green-700 transition w-full">Login</button>
67           <p id="message" class="text-center mt-2"></p>
68         </div>
69       </div>

```

```

70     </div>
71     <div class="input-section" id="tokoSection" style="display: none;">
72         <div class="card mt-10 max-w-4xl mx-auto bg-white p-6 rounded-lg shadow-md">
73             <h2 class="text-xl font-semibold mb-4">Tambah Produk Baru</h2>
74             <div class="grid grid-cols-1 md:grid-cols-4 gap-4">
75                 <div>
76                     <label class="block text-sm font-medium text-gray-700">Nama Produk</label>
77                     <input type="text" id="newProductName" placeholder="Nama Produk"
78                     class="mt-1 block w-full border-gray-300 rounded-md shadow-sm focus:ring-blue-500 focus:border-blue-500">
79                 </div>
80                 <div>
81                     <label class="block text-sm font-medium text-gray-700">Harga Produk</label>
82                     <input type="number" id="newProductPrice" placeholder="Harga Produk"
83                     class="mt-1 block w-full border-gray-300 rounded-md shadow-sm focus:ring-blue-500 focus:border-blue-500">
84                 </div>
85                 <div>
86                     <label class="block text-sm font-medium text-gray-700">Stok Produk</label>
87                     <input type="number" id="newProductStock" placeholder="Stok Produk"
88                     class="mt-1 block w-full border-gray-300 rounded-md shadow-sm focus:ring-blue-500 focus:border-blue-500">
89                 </div>
90                 <div>
91                     <label class="block text-sm font-medium text-gray-700">Foto Produk</label>
92                     <input type="file" id="newProductImage"
93                     accept="image/*" class="mt-1 block w-full text-sm text-gray-500 file:mr-4 file:py-2 file:px-4
94                     file:rounded-md file:border-0 file:text-sm file:font-semibold file:bg-blue-50 file:text-blue-700 hover:file:bg-blue-100">
95                     <img id="imagePreview" class="mt-2 w-24 h-24 object-cover rounded-md hidden alt="Pratinjau">
96                 </div>

```

```

97     </div>
98     <button onclick="addProduct()" class="mt-4 bg-green-600 text-white px-4 py-2 rounded-md hover:bg-green-700 transition w-full md:w-auto">Tambah Produk</button>
99 </div>
100 </div>
101 <div class="list-section" id="daftarSection" style="display: none;">
102     <div class="card mt-10 max-w-4xl mx-auto bg-white p-6 rounded-lg shadow-md">
103         <h2 class="text-xl font-semibold mb-4">Daftar Produk</h2>
104         <div id="productList" class="grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 lg:grid-cols-4 gap-6">
105             <p id="noProducts" class="text-gray-500 text-center col-span-full" style="display: none;">Belum ada produk yang ditambahkan.</p>
106         </div>
107         <div class="mt-6 text-center">
108             <button onclick="calculateTotal()" class="bg-blue-600 text-white px-4 py-2 rounded-md hover:bg-blue-700 transition mr-4">Cetak Struk</button>
109             <button onclick="refreshPage()" class="bg-green-600 text-white px-4 py-2 rounded-md hover:bg-green-700 transition">Refresh</button>
110         </div>
111     </div>
112 </div>
113 <div id="receipt" class="mt-6 mx-auto">
114     <div class="receipt-container">
115         <div class="receipt-header">
116             <h2 class="font-bold">Toko Sederhana</h2>
117             <p>Jl. Ir Soekarno No.177, Blitar</p>
118             <p>Telp: (021) 123-4567</p>
119             <p>Tanggal: <span id="receiptDate"></span></p>
120         </div>

```

```

121         <div class="receipt-line"></div>
122         <div id="receiptDetails" class="space-y-1"></div>
123         <div class="receipt-line"></div>
124         <div class="receipt-total">
125             <div class="receipt-item">
126                 <span>Total</span>
127                 <span id="total"></span>
128             </div>
129             <div class="receipt-item">
130                 <span>Diskon (10% untuk ≥ Rp 100,000)</span>
131                 <span id="discount"></span>
132             </div>
133             <div class="receipt-item">
134                 <span>Total Akhir</span>
135                 <span id="finalTotal"></span>
136             </div>
137         </div>
138         <div class="receipt-line"></div>
139         <div class="receipt-header">
140             <p>Terima Kasih Telah Berbelanja!</p>
141             <p>Barang yang sudah dibeli tidak dapat ditukar/dikembalikan</p>
142         </div>
143     </div>
144 </div>
145 </div>
146 </div>
147 <script src="script.js"></script>
148 </body>
149 </html>

```

B. STYLE.CSS

```
1  .card {
2    background: white;
3    padding: 20px;
4    border-radius: 15px;
5    box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
6  }
7
8  .input-section, .list-section {
9    width: 100%;
10   max-width: 800px;
11 }
12
13 #appHeader {
14   width: 100%;
15   margin-bottom: 20px;
16   border-radius: 15px;
17   overflow: hidden;
18 }
19
20
21 #message {
22   text-align: center;
23   margin-top: 10px;
24   color: #dc3545;
25 }
26
27
28 #loginSection .card {
29   margin-top: 50px;
30 }
```

C. SCRIPT.JS

```
1  let products = [];
2  let isLoggedIn = false;
3
4  $(document).ready(function() {
5    loadProducts();
6    $('#appHeader').hide();
7    $('#tokoSection').hide();
8    $('#daftarSection').hide();
9    $('#receipt').hide();
10 });
11
12 // Pratinjau gambar
13 $('#newProductImage').change(function() {
14   const file = this.files[0];
15   if (file) {
16     const reader = new FileReader();
17     reader.onload = function(e) {
18       $('#imagePreview').attr('src', e.target.result).removeClass('hidden');
19     };
20     reader.readAsDataURL(file);
21   }
22 });
```

```

24 function login() {
25     const username = $('#username').val();
26     const password = $('#password').val();
27     const message = $('#message');
28     const loginSection = $('#loginSection');
29     const tokoSection = $('#tokoSection');
30     const daftarSection = $('#daftarSection');
31     const header = $('#appHeader');
32
33     const validUsername = "admin";
34     const validPassword = "cobatanya";
35
36     if (username === validUsername && password === validPassword) {
37         message.text('Login berhasil!').css('color', '#28a745');
38         isLoggedIn = true;
39         loginSection.hide();
40         tokoSection.show();
41         daftarSection.show();
42         header.show();
43     } else {
44         message.text('Username atau password salah!').css('color', '#dc3545');
45     }

```

```

47     setTimeout(() => {
48         $('#username').val('');
49         $('#password').val('');
50         message.text('');
51     }, 2000);
52 }
53
54 function addProduct() {
55     if (!isLoggedIn) {
56         alert("Silakan login terlebih dahulu!");
57         return;
58     }
59     const name = $('#newProductName').val();
60     const price = parseInt($('#newProductPrice').val());
61     const stock = parseInt($('#newProductStock').val());
62     const imageFile = $('#newProductImage')[0].files[0];
63
64     if (name && price && !isNaN(price) && stock >= 0 && !isNaN(stock)) {
65         const formData = new FormData();
66         formData.append('name', name);
67         formData.append('price', price);
68         formData.append('stock', stock);
69         formData.append('image', imageFile);
70
71         $.ajax({

```



```

71     $.ajax({
72         url: '/api/add-product',
73         type: 'POST',
74         data: formData,
75         contentType: false,
76         processData: false,
77         success: function(response) {
78             alert("Produk berhasil ditambahkan!");
79             $('#newProductName').val('');
80             $('#newProductPrice').val('');
81             $('#newProductStock').val('');
82             $('#newProductImage').val('');
83             $('#imagePreview').addClass('hidden');
84             loadProducts();
85         },
86         error: function(err) {
87             alert("Gagal menambahkan produk: " + err.responseText);
88         }
89     });
90 } else {
91     alert('Masukkan nama, harga, dan stok yang valid!');
92 }
93 }

```

```

95 function loadProducts() {
96     $.get('/api/products', function(data) {
97         products = data;
98         renderProducts();
99     }).fail(function(err) {
100         console.error("Error loading products:", err);
101     });
102 }
103
104 function renderProducts() {
105     const productList = $('#productList');
106     productList.empty();
107     if (products.length === 0) {
108         $('#noProducts').show();
109         return;
110     }
111     $('#noProducts').hide();
112     products.sort((a, b) => a.price - b.price);
113     products.forEach(product => {
114         const div = $('

').addClass('bg-white p-4 rounded-lg shadow-md hover:shadow-lg transition');
115         div.html(
116             
117             <h2 class="text-lg font-sembold">${product.name}</h2>
118             <p class="text-gray-600">Harga: Rp ${product.price.toLocaleString('id-ID')}</p>
119             <p class="text-gray-600">Stok: ${product.stock}</p>
120             <input type="number" min="0" max="${product.stock}" value="${product.quantity || 0}" class="quantity w-full p-2 border rounded mt-2" onchange="updateQuantity(${product.id},
121                 this.value)" placeholder="Jumlah">
122         );
123         productList.append(div);
124     });
125 }


```

```

127 function calculateTotal() {
128     if (!isLoggedIn) {
129         alert("Silakan login terlebih dahulu!");
130         return;
131     }
132     let total = 0;
133     let receiptDetails = '';
134     let valid = true;
135     products.forEach(product => {
136         if (product.quantity > product.stock) {
137             valid = false;
138             alert(`Jumlah ${product.name} melebihi stok! Stok tersedia: ${product.stock}`);
139             return;
140         }
141         if (product.quantity > 0) {
142             const subtotal = product.price * product.quantity;
143             receiptDetails += `<div class="receipt-item"><span>${product.name}</span> (${product.quantity})x, Stok: ${product.stock}</span><span>Rp ${subtotal.toLocaleString('id-ID')}</span></div>`;
144             total += subtotal;
145         }
146     });

```

```

148     if (!valid) return;
149
150     const discount = total >= 100000 ? total * 0.1 : 0;
151     const finalTotal = total - discount;
152
153     const today = new Date().toLocaleString('id-ID', {
154         day: '2-digit',
155         month: '2-digit',
156         year: 'numeric',
157         hour: '2-digit',
158         minute: '2-digit'
159     });
160
161     $('#receiptDate').text(today);
162     $('#receiptDetails').html(receiptDetails);
163     $('#total').text(`Rp ${total.toLocaleString('id-ID')}`);
164     $('#discount').text(`Rp ${Math.round(discount).toLocaleString('id-ID')}`);
165     $('#finalTotal').text(`Rp ${Math.round(finalTotal).toLocaleString('id-ID')}`);
166     $('#receipt').show();
167
168     window.print();
169 }

```

```

171 function updateQuantity(id, value) {
172     const product = products.find(p => p.id === id);
173     if (product) {
174         product.quantity = parseInt(value) || 0;
175     }
176 }
177
178 function refreshPage() {
179     if (!isLoggedIn) {
180         alert("Silakan login terlebih dahulu!");
181         return;
182     }
183     products.forEach(p => p.quantity = 0);
184     $('.quantity').val('0');
185     $('#receipt').hide();
186     $('#receiptDetails').empty();
187     $('#total').empty();
188     $('#discount').empty();
189     $('#finalTotal').empty();
190     renderProducts();
191 }

```

E. Tampilan Aplikasi

a. Menu Login

Login

Username

Masukkan Username

Password

Masukkan Password

Login

b. Menu Beranda

Aplikasi Toko Sederhana

Tambah Produk Baru

Nama Produk

Nama Produk

Harga Produk

Harga Produk

Stok Produk

Stok Produk


Foto Produk

Choose File

No file chosen

Tambah Produk

Daftar Produk




enteng-enteng

Harga: Rp 40.000

Stok: 30

0



jenang ketan

Harga: Rp 50.000

Stok: 30

0

Cetak Struk

Refresh

c. Menu Cetak Struk

Aplikasi Toko Sederhana

Tambah Produk Baru

Nama Produk

Nama Produk

Harga Produk

Harga Produk

Stok Produk

Stok Produk


Foto Produk

Choose File

No file chosen

Tambah Produk

Daftar Produk




enteng-enteng

Harga: Rp 40.000

Stok: 30

1



jenang ketan

Harga: Rp 50.000

Stok: 30

1

Cetak Struk

Refresh

Toko Sederhana

Jl. Ir. Soekarno No.177, Bilitar

Telp: +62 815-5045-8164

Tanggal: 01/07/2020, 16:16

.....

enteng-enteng (St. Resk: 30) Rp 40.000

jenang ketan (St. Resk: 30) Rp 50.000

Total

Rp 90.000

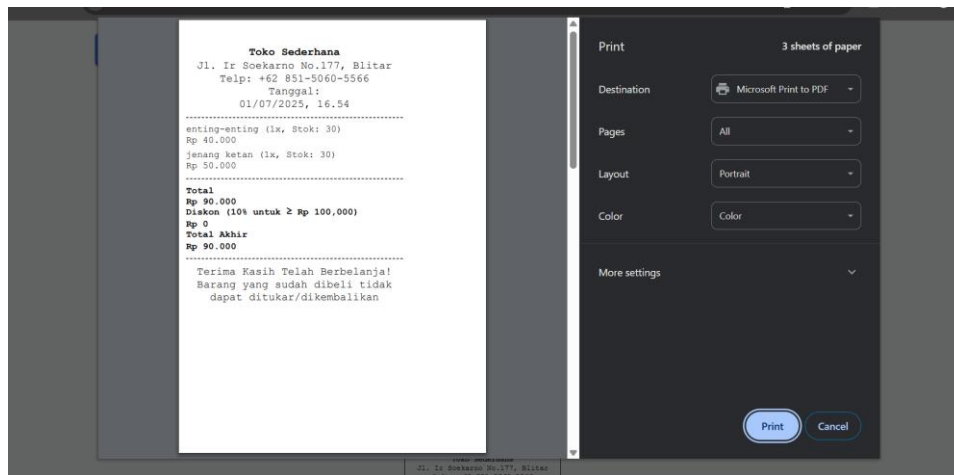
Bayar (tunai) x Rp 100.000 Rp 0

Tunai Akhir Rp 90.000

.....

Terima Kasih Telah Berbelanja!

Sorang yang sudah dibeli tidak dapat di-track/dikembalikan



J. Kesimpulan

Aplikasi toko sederhana ini layak dijadikan prototype awal untuk pengembangan toko online skala kecil. Dengan pengembangan lebih lanjut seperti sistem login, manajemen produk, dan fitur checkout, aplikasi ini berpotensi digunakan secara riil oleh UMKM.

BAB IV

PENUTUP

4.1 Kesimpulan

Aplikasi kasir toko yang telah dibuat berhasil memenuhi kebutuhan dasar dalam mengelola data barang, dan mencetak struk. Aplikasi ini diharapkan dapat membantu pemilik toko dalam meningkatkan efisiensi operasional dan meminimalkan kesalahan perhitungan.

4.2 Saran

Pengembangan lebih lanjut masih diperlukan, seperti integrasi dengan perangkat kasir, serta peningkatan keamanan dan kemudahan penggunaan. Semoga aplikasi ini dapat terus disempurnakan dan bermanfaat bagi pelaku usaha.

DAFTAR PUSTAKA

- Rumahweb. (n.d.). *CSS adalah: Pengertian, fungsi, dan jenis-jenisnya*. Rumahweb Journal. Diakses pada 1 Juli 2025, dari <https://www.rumahweb.com/journal/css-adalah/>
- DomaiNesia. (n.d.). *HTML adalah: Pengertian, fungsi, dan struktur dasar HTML*. DomaiNesia Berita. Diakses pada 1 Juli 2025, dari <https://www.domainesia.com/berita/html-adalah/>
- Dewaweb. (n.d.). *Mengenal JavaScript: Pengertian, fungsi, dan cara kerjanya*. Dewaweb Blog. Diakses pada 1 Juli 2025, dari <https://www.dewaweb.com/blog/mengenal-javascript/>