Laporan Proyek: Sistem Aplikasi Point of Sale (POS) Sederhana

# I. Pendahuluan

## 1.1 Latar Belakang

Dalam operasional bisnis makanan dan minuman, kecepatan dan akurasi pencatatan pesanan adalah kunci untuk kepuasan pelanggan dan efisiensi internal. Proyek ini dikembangkan untuk mengatasi keterbatasan sistem pencatatan manual atau desktop-lama yang kaku. Permasalahan utama adalah kurangnya interaktivitas dan kebutuhan page reload untuk setiap perubahan keranjang. Solusi yang diberikan adalah membangun aplikasi POS berbasis web yang dinamis, memungkinkan pelayan atau kasir mencatat pesanan, memfilter menu, dan mengkonfirmasi transaksi secara real-time tanpa interruption.

## 1.2 Tujuan Proyek

Tujuan utama proyek ini adalah:

* Menyediakan layanan interaktif untuk mencatat pesanan secara real-time menggunakan AJAX.
* Meningkatkan efisiensi proses ordering dengan fitur penambahan dan pengurangan item keranjang yang dinamis.
* Memberikan solusi praktis untuk penyimpanan data transaksi sementara dan riwayat pemesanan menggunakan PHP Session.

## 1.3 Ruang Lingkup Proyek

Ruang lingkup proyek ini mencakup:

* **Termasuk**: Implementasi tampilan menu dinamis, manajemen keranjang (tambah/kurang/hapus) melalui AJAX, proses konfirmasi pesanan dengan detail pelanggan, penyimpanan transaksi ke PHP Session ($\_SESSION['orders'] dan $\_SESSION['order\_menu']), dan tampilan riwayat pesanan.
* **Tidak Termasuk**: Integrasi dengan database relasional (MySQL/PostgreSQL), user authentication (login/otorisasi), manajemen stok, fitur split bill, dan deployment pada lingkungan produksi.

# II. Analisis Kebutuhan

## 2.1 Kebutuhan Sistem

* Kebutuhan Fungsional:
  1. Aplikasi harus dapat memfilter dan menampilkan menu berdasarkan kategori tanpa page reload.
  2. Aplikasi harus dapat menambah item ke keranjang dan meningkatkan kuantitasnya melalui tombol "Tambah" (input formulir ).
  3. Aplikasi harus dapat mengubah kuantitas item (tambah/kurang) dan menghapus item dari keranjang melalui AJAX.
  4. Aplikasi harus melakukan perhitungan total harga secara akurat dan real-time (perhitungan data ).
  5. Aplikasi harus menyimpan data keranjang dan data transaksi (orders, order\_menu) ke PHP Session (penyimpanan data ).
* Kebutuhan Non-Fungsional:

1. Aplikasi harus memberikan feedback visual yang jelas menggunakan notifikasi kustom (hijau/merah).
2. Antarmuka pengguna harus intuitif dan responsif.

## 2.2 Kebutuhan Perangkat Keras dan Perangkat Lunak

Berikut kebutuhan perangkat keras dan perangkat lunak yang dibutuhkan selama system ini dikembangkan.

* **Perangkat Keras:** Laptop Pengembang, Perangkat Testing (Browser Desktop), Server Lokal (MAMP)
* **Perangkat Lunak:** Sistem Operasi, Editor Kode (Visual Studio Code), Browser Modern (Chrome), Lingkungan Server (PHP 7.4+), Library (jQuery 3.x).

# III. Desain Sistem

## 3.1 Arsitektur Sistem

Jelaskan struktur arsitektur aplikasi yang dikembangkan, seperti:

* Flowchart Melakukan Order :

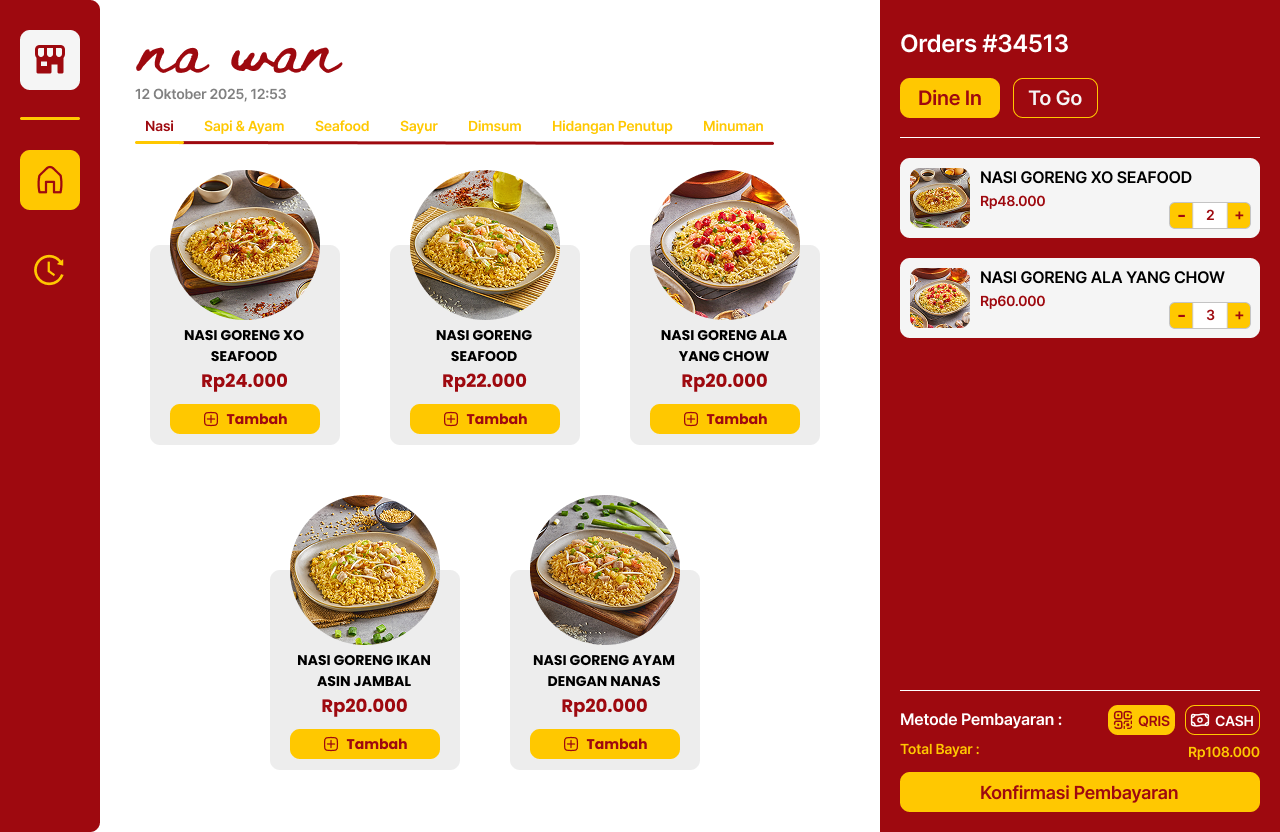
A diagram of a process

AI-generated content may be incorrect.

## 3.2 Desain Antarmuka Pengguna (UI/UX)

Desain Antarmuka pengguna :

* Home Page :



* History Page

A screenshot of a computer

AI-generated content may be incorrect.

# IV. Implementasi Sistem

## 4.1 Teknologi yang Digunakan

Teknologi yang digunakan dalam proyek ini:

* **Bahasa Pemrograman:** HTML, CSS, JavaScript, PHP.
* **Framework/Library:** jQuery.
* **Database Sementara:** PHP Session.
* **Alat Pengembangan:** MAMP (Local Server Environment), Editor Kode Visual Studio Code.

## 4.2 Proses Pengembangan

Proses pengembangan melibatkan implementasi PHP Session sebagai data storage dan penggunaan jQuery untuk membuat aplikasi dinamis.

* 1. **Identifikasi** **Kebutuhan**: Ditentukan bahwa interaksi dinamis ( filter, keranjang) adalah fitur inti.
  2. **Desain UI/UX:** Dibuat wireframe dengan sidebar keranjang statis dan area menu yang dapat diperbarui secara dinamis.
  3. **Implementasi Logika Backend (PHP)**:
     + Manajemen Session: PHP Session digunakan untuk memelihara state keranjang di seluruh permintaan.
     + AJAX Handlers: Dibuat endpoint PHP terpisah (add\_to\_cart.php, update\_cart.php, confirm\_order.php) untuk memproses permintaan POST dari JavaScript.
  4. **Integrasi dengan jQuery**:
     + **Manipulasi DOM**: jQuery digunakan untuk menghapus item keranjang (.fadeOut(), .remove()), memperbarui total harga (.text()), dan memfilter menu (.html(data.html)).
     + **Konsistensi Data**: Digunakan atribut data-menu-id di frontend dan kunci 'menus\_id' di backend untuk memastikan data yang dikirim dan diterima selalu sinkron.

# VII. Kesimpulan

## 7.1 Kesimpulan

Proyek Sistem Aplikasi POS Sederhana ini berhasil mencapai tujuannya dengan menyediakan solusi pencatatan pesanan yang dinamis dan interaktif. Penggunaan PHP Session memungkinkan state transaksi dipertahankan tanpa database relasional, sementara kombinasi jQuery dan AJAX berhasil menghilangkan kebutuhan page reload untuk hampir semua interaksi kritis (tambah/kurang item, filter menu, konfirmasi pesanan).

## 7.2 Rekomendasi

Untuk pengembangan lebih lanjut, direkomendasikan:

1. **Database Relasional**: Migrasi penyimpanan data transaksi (orders, order\_menu) dari PHP Session ke MySQL/MariaDB untuk persistence data yang sejati dan skalabilitas.
2. **Framework CSS**: Penggunaan framework CSS modern (misalnya, Bootstrap atau Tailwind CSS) untuk mempercepat pengembangan UI dan memastikan responsivitas tingkat lanjut.
3. **Aplikasi Multi-Page**: Implementasi fitur user authentication (login/logout) untuk membedakan antara pengguna (misalnya, kasir dan manajer)..

# VIII. Daftar Pustaka

Homepage - Ta-Wan. (2025, April 16). Ta-Wan. <https://www.tawanrestaurant.com/>

*PHP: Documentation*. (n.d.). https://www.php.net/docs.php

OpenJS Foundation - openjsf.org. (n.d.). *JQuery API Documentation*. https://api.jquery.com/

*Homepage - Ta-Wan*. (2025, April 16). Ta-Wan. https://www.tawanrestaurant.com/