



IMPLEMENTASI KARTU *MEMBER* PADA APLIKASI *POINT OF SALES* BERBASIS *WEB* DALAM MENDUKUNG PROGRAM PKH (STUDI KASUS : TOKO MENIKWATI)

Dikky Wahyudi ^{1*}, Suprihadi ²

^{1*,2} Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Kota Salatiga, Provinsi Jawa Tengah, Indonesia.

Email: 672019097@student.uksw.edu ^{1*}, suprihadi@uksw.edu ²

Histori Artikel:

Dikirim 22 Juni 2023; *Diterima dalam bentuk revisi* 5 Juli 2023; *Diterima* 8 Agustus 2023; *Diterbitkan* 10 September 2023. Semua hak dilindungi oleh Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) STMIK Indonesia Banda Aceh.

Abstrak

Toko merupakan tempat dimana transaksi jual beli antara penjual dan pembeli. Namun tidak semua orang dapat membeli barang yang dibutuhkan dalam memenuhi kebutuhan sehari-hari, khususnya bagi keluarga yang menyanggah status keluarga miskin. Hal ini tentu saja dapat mengakibatkan berbagai permasalahan bahkan sampai melakukan tindakan kriminal demi memenuhi kebutuhan pribadi masing-masing. Disamping itu salah satu tujuan pemerintah adalah mengurangi angka kemiskinan yang semakin meningkat, berdasarkan inti permasalahan tersebut maka tujuan penelitian ini berfokus pada mengimplementasikan kartu member pada aplikasi point of sales berbasis web dalam mendukung program PKH (Studi Kasus : Toko Menikwati). Diharapkan dengan adanya aplikasi ini dapat membantu mensejahterakan perekonomian bagi keluarga yang menyanggah status keluarga miskin, membantu menanggulangi kemiskinan dan mendukung serta mensukseskan program PKH. Dalam membangun aplikasi point of sales ini metode penelitian yang digunakan adalah model waterfall, bahasa pemrograman yang digunakan adalah PHP, JavaScript, Ajax serta Framework Laravel. Untuk pengujian aplikasi menggunakan User Acceptance Test (UAT) dengan metode pengujian black box testing.

Kata Kunci: Point of Sales; PHP; Laravel; Kartu Member.

Abstract

Store is a place where buying and selling transactions between sellers and buyers. However not everyone can buy the things needed to fulfil of their daily needs, especially for families bear status of poor families. This raises various problems even do criminal acts in order to fulfil their needed. On the other hand one of the government's goals is reduce the poverty rate. Based on the core of problem, this research focuses on implementing member cards on web-based point of sales applications to support of the PKH program (Case Study : Menikwati Store). It is hoped this application can help prosper the economoy for families with poor family status, overcome poverty, and support the PKH program to successfully. The development of this point of sales application use waterfall method, with the programming language used is PHP, JavaScript, Ajax and Laravel Framework. For application testing using User Acceptance Test (UAT) with the black box testing.

Keyword: Point of Sales; PHP; Laravel; Member Card.

1. Pendahuluan

Pada era sekarang teknologi informasi berkembang sangat pesat, sehingga banyak teknologi yang dihasilkan dalam membantu meringankan beban kerja manusia. Yang dulunya menggunakan sistem konvensional sekarang beralih menggunakan sistem terkomputerisasi contohnya toko. Toko merupakan tempat dalam melakukan proses transaksi jual beli barang. Saat seseorang pergi ke sebuah toko pastilah seseorang tersebut ingin membeli suatu barang untuk kebutuhan dan disanalah terjadi proses transaksi antara penjual dengan pembeli. Toko Menikwati sebagai toko yang menyediakan kebutuhan sehari-hari dipilih sebagai studi kasus karena memiliki permasalahan dalam hal pengolahan data dan proses transaksi dimana kegiatan yang telah disebutkan sebelumnya masih menggunakan sistem konvensional atau manual sehingga toko Menikwati ingin menggunakan aplikasi berupa aplikasi *point of sales* sebagai solusi yang dapat memecahkan permasalahan toko. POS (*Point of Sales*) merupakan sistem aplikasi yang diterapkan pada bisnis minimarket atau pertokoan untuk menangani pengolahan data transaksi pembelian (*purchases*), transaksi penjualan eceran (*retails*), transaksi retur pembelian (*purchase returns*), dan laporan transaksi (*reporting*) yang secara umum dibutuhkan dalam pengambilan keputusan strategis oleh para pebisnis swalayan, organisasi, maupun perusahaan skala kecil dan menengah [1]. Dipilihnya aplikasi *point of sales* dikarenakan dapat memberikan pelayanan yang lebih baik kepada pelanggan dalam hal perhitungan harga dan jumlah barang yang dibeli menjadi lebih cepat, sehingga tidak lagi bergantung pada pencatatan manual serta memberikan informasi secara terperinci atas setiap transaksi yang terjadi, selain itu aplikasi *point of sales* memiliki keuntungan lainnya yaitu dapat dengan mudah melakukan pengolahan data secara digital dan cepat guna menghindari hilangnya data [2].

Namun aplikasi yang akan dibangun tidak seperti aplikasi *point of sales* biasanya, terdapat fitur khusus yaitu kartu *member* bagi keluarga yang menyandang status keluarga miskin sehingga saat hendak berbelanja dan memiliki kartu member akan mendapat potongan harga pada hari yang telah ditentukan, hal ini bertujuan untuk membantu mensejahterakan perekonomian bagi keluarga miskin dan mendukung program PKH. PKH sendiri merupakan program bantuan sosial bersyarat yang artinya setiap masyarakat yang menjadi peserta sasaran program memiliki kewajiban yang ditentukan kementerian sosial sebagai syarat penerima bantuan [3].

Dalam pembangunan aplikasi *point of sales* ini pastinya menggunakan teknologi yang canggih dan modern. Laravel adalah sebuah framework web berbasis PHP (bahasa skrip yang dapat ditanamkan ke HTML (bahasa standar web yang dikelola penggunaannya oleh W3C (*World Wide Web Consortium*)) berupa tag-tag yang menyusun setiap elemen dari website [4]) dan diterapkan dalam situs web. Pengertian lainnya PHP merupakan bahasa pemrograman *script* yang membuat dokumen HTML secara *on the fly* dieksekusi di web server dan dokumen HTML yang dibuat menggunakan teks editor atau editor HTML [5]) yang open-source dan tidak berbayar, diciptakan oleh Taylor Otwell dan diperuntukkan untuk pengembangan aplikasi web yang menggunakan pola MVC. Struktur pola MVC pada laravel sedikit berbeda pada struktur pola MVC pada umumnya. Di laravel terdapat routing yang menjembatani antara request dari user dan controller. Jadi controller tidak langsung menerima request tersebut [6]. Dipilihnya *framework* laravel dikarenakan *framework* laravel memiliki keunggulan tersendiri yang menjadikannya lebih baik dari pada *framework* lainnya, berikut ini merupakan kelebihan dari laravel yaitu, *performance* lebih cepat, *reload* data lebih stabil, memiliki keamanan data, menggunakan fitur canggih seperti *blade* menggunakan konsep HMVC (*Hierarchical Model, View, Controller*), tersedianya *library-library* yang sudah siap untuk digunakan dan adanya fitur pengelolaan migration untuk pembuatan skema tabel pada basis data [7].

Pada penelitian yang berjudul “Perancangan Sistem Informasi POS (*Point Of Sales*) Berbasis Web Menggunakan *Framework Codeigniter* pada Pasar Swalayan”, membahas tentang bagaimana merancang sistem informasi *Point Of Sales* yang mudah digunakan pengguna khususnya para pengusaha menengah di daerah pasar swalayan kabupaten karawang yang rata-rata masih menggunakan sistem manual, sehingga dengan adanya aplikasi POS ini dapat memberikan solusi kepada pengusaha pasar swalayan dalam hal pengolahan data. Penelitian ini menggunakan bahasa pemrograman PHP, *Framework Codeigniter*, serta XAMPP sebagai server sebagai server yang berdiri sendiri (*localhost*), yang terdiri

beberapa program antara lain : Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP sendiri merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl [8]. Dari hasil penelitian didapatkan sebanyak 24 respon (21,82%) Sangat Setuju (SS), 63 respon (57,27%) Setuju (S), 23 respon (20,91%) Netral (N), sedangkan sisanya Tidak Setuju (TS) dan Sangat Tidak Setuju (STS) tidak ada yang respon [9].

Pada penelitian yang berjudul “Rancang Bangun Aplikasi *Point of Sales Distro Management System* dengan Menggunakan *Framework* React Native“, membahas tentang bagaimana merancang aplikasi *Point Of Sales* yang mudah digunakan pegawai distro dimana pelaku bisnis distro ini masih menjalankan proses bisnisnya secara manual, diharapkan dengan adanya aplikasi POS ini pegawai distro lebih efektif dan efisien dalam menjalankan bisnisnya. Dalam penelitian ini metode yang digunakan adalah *classic life cycle*, *Framework* yang digunakan React Native, serta pengujianya menggunakan *Black Box Testing*. Dari hasil penelitian disimpulkan bahwa aplikasi berhasil dibuat dan telah disesuaikan dengan kebutuhan pihak distro [10].

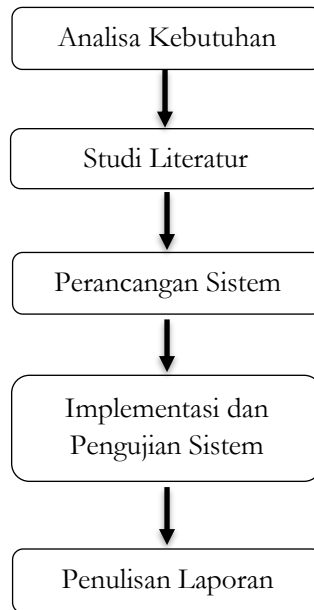
Pada penelitian yang berjudul “Perancangan Aplikasi *Point Of Sales* Pandita Coffee Berbasis Web Dengan Kombinasi Model Model SDLC *Waterfall*”, membahas tentang bagaimana merancang aplikasi *Point Of Sales* yang dapat membantu pegawai Pandita Coffee. Dijelaskan juga aplikasi POS Pandita Coffee ini mampu mendukung proses transaksi lebih cepat, mendata semua transaksi secara lengkap, serta laporan penjualan yang dibutuhkan secara digital atau terkomputerisasi. Metode penelitian yang dipakai adalah SDLC dengan model *waterfall* [11].

Pada penelitian yang berjudul “Perancangan dan Penerapan Sistem Penjualan Berbasis Web pada Restoran Tengkleng Mbah Ninik”, membahas tentang bagaimana merancang dan menerapkan sistem penjualan berbasis web pada restoran mbah ninik, dimana sistem yang dijalankan masih menggunakan sistem yang manual. Metode penelitian yang digunakan dalam penelitian ini adalah SDLC model *waterfall*, serta menggunakan teknik pengujian *black box testing*, *white box testing*, dan *user acceptance*. Dari hasil penelitian, aplikasi dapat membantu dalam manajemen data, meningkatkan pengolahan data, mempercepat proses transaksi, pembuatan laporan, serta meningkatkan kinerja di restoran Tengkleng Mbah Ninik Kudus [12].

Pada penelitian yang berjudul “Sistem Informasi Manajemen Bengkel Modul *Point of Sales* Berbasis Web”, membahas tentang bagaimana membangun sistem informasi manajemen (SIM) bengkel modul *point of sales* berbasis web, dikarenakan proses bisnis secara konvensional atau manual masih banyak ditemukan, sehingga tidak menutup kemungkinan bahwa akan terjadinya kerusakan atau hilangnya data laporan atau transaksi penjualan suku cadang pada bengkel kendaraan. Dalam penelitian ini metode yang digunakan adalah SDLC model *waterfall*, bahasa pemrograman yang dipakai adalah php, serta *framework* laravel. Hasil penelitian disebutkan bahwa aplikasi dapat digunakan oleh pihak bengkel dan fitur-fitur yang ada telah sesuai dengan kebutuhan bengkel sehingga dapat mempermudah pemilik dan pegawai dalam memperoleh informasi yang akurat, cepat, dan efektif serta membantu dalam pengambilan keputusan manajemen bengkel [13]. Berdasarkan latar belakang yang telah diuraikan, maka dilakukan penelitian yang bertujuan untuk mengimplementasikan kartu *member* pada aplikasi *point of sales* berbasis *web* dalam mendukung program PKH (studi kasus: toko menikwati) dengan tujuan untuk membantu mensejahterakan perekonomian keluarga miskin dan mendukung program PKH.

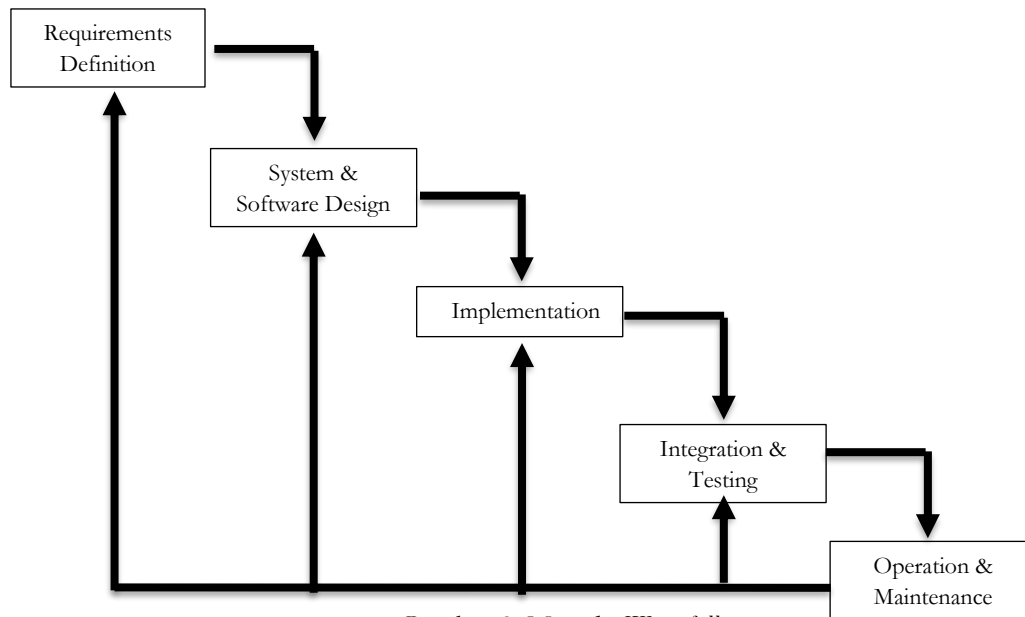
2. Metode Penelitian

Pada pembangunan aplikasi *point of sales* akan dilakukan beberapa tahapan penelitian, berikut merupakan tahapan-tahapan penelitian yang akan dilakukan untuk mendukung keberhasilan penelitian.



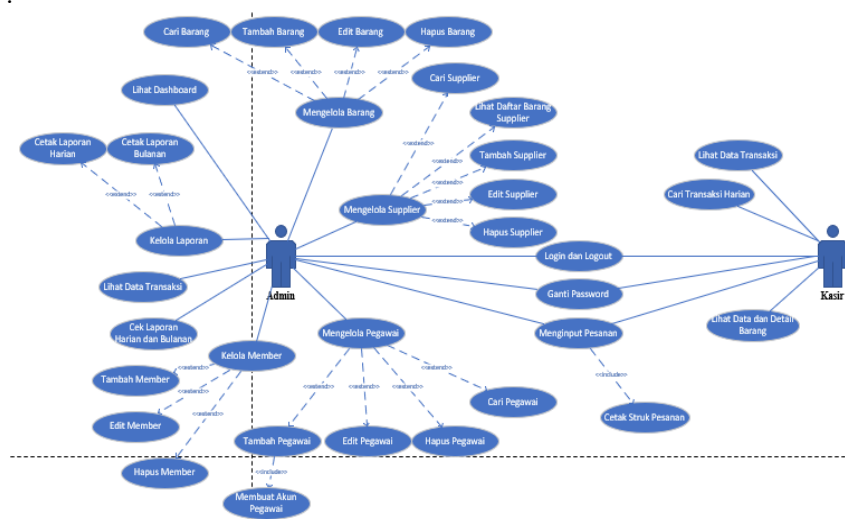
Gambar 1. Tahapan Penelitian

Tahap pertama analisa kebutuhan. Pada tahap ini peneliti akan mengumpulkan data-data yang dibutuhkan untuk menunjang pembangunan aplikasi, sehingga aplikasi yang nantinya akan digunakan sesuai dengan apa yang diharapkan dan yang di inginkan oleh pemilik toko. Tahap kedua adalah studi literatur. Pada tahap ini peneliti akan membaca serta mengumpulkan data-data berupa *literature review* atau penelitian sebelumnya yang serupa dan yang berkaitan dengan topik penelitian yang diambil, untuk mendukung keberhasilan penelitian. Tahap ketiga adalah perancangan sistem. Pada tahap ini peneliti mulai merancang gambaran dan model sistem yang nantinya akan di implementasikan menjadi sebuah aplikasi. Gambaran serta model yang dimaksud adalah model apa yang dipakai untuk membangun sistem, pada pembangunan aplikasi *point of sales* ini model yang dipakai adalah SDLC model *waterfall* serta perancangan sistem menggunakan UML (*Unified Modeling Language*). Tahap keempat adalah implementasi dan pengujian sistem. Pada tahap ini peneliti mengimplementasikan hasil rancangan sistem yang telah dibuat ke dalam bentuk kode-kode program yang nantinya akan menjadi suatu aplikasi dan akan diuji menggunakan *white box testing* atau *black box testing*. Pada kasus ini metode pengujian sistem yang digunakan adalah *black box testing*. Tahap kelima adalah penulisan laporan. Pada tahap ini peneliti melakukan dokumentasi atas penelitian yang telah dilakukan, mulai dari tahap awal hingga tahap akhir penelitian dalam bentuk tulisan ilmiah. Metode penelitian yang digunakan dalam membangun aplikasi *point of sales* adalah model *waterfall*.



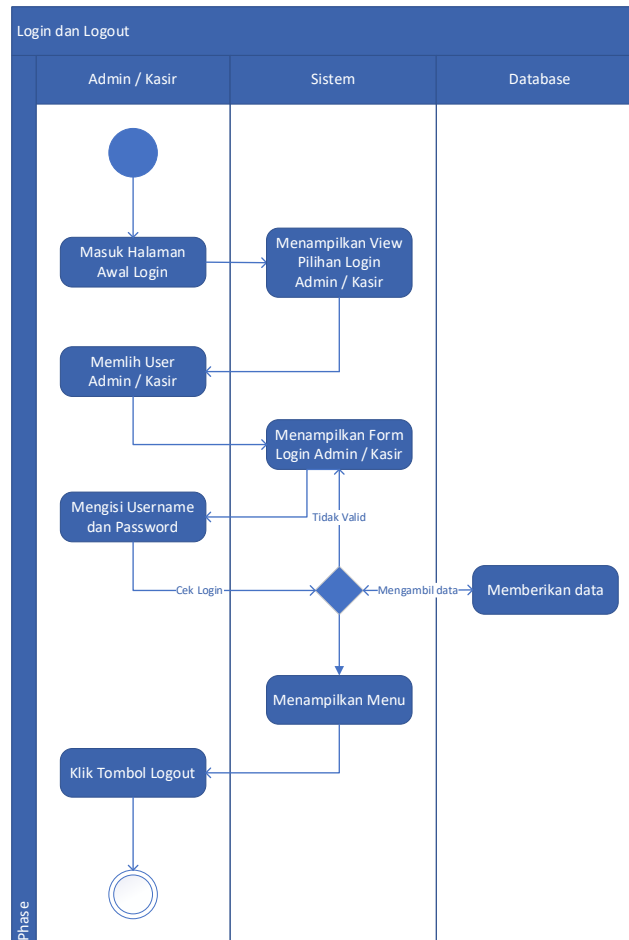
Gambar 2. Metode *Waterfall*

Dipilihnya model *waterfall* ini tidak terlepas dari kelebihanannya yakni kualitas dari sistem yang dihasilkan akan baik karena dalam pelaksanaannya dilakukan secara bertahap serta proses pengembangan model fase *one by one*, sehingga meminimalisir kesalahan yang mungkin terjadi [14]. Cocok digunakan untuk proyek skala kecil, menengah, maupun besar. Dimulai dari mendefinisikan kebutuhan yang diperlukan sistem, mendesain arsitektur sistem yang dibuat, mengimplementasikan desain yang sebelumnya telah dibuat kedalam bentuk kode-kode program, mengintegrasikan sub sistem serta menguji modul-modul sub sistem, dan melakukan pemeliharaan atas sistem yang dibuat. Sistem dirancang menggunakan UML (*Unified Modelling Language*). UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung [15]. Terdapat empat diagram yang digunakan yaitu *use case diagram*, *activity diagram*, *class diagram*, dan *deployment diagram*. Berikut adalah diagram-diagram yang sudah disebutkan sebelumnya :



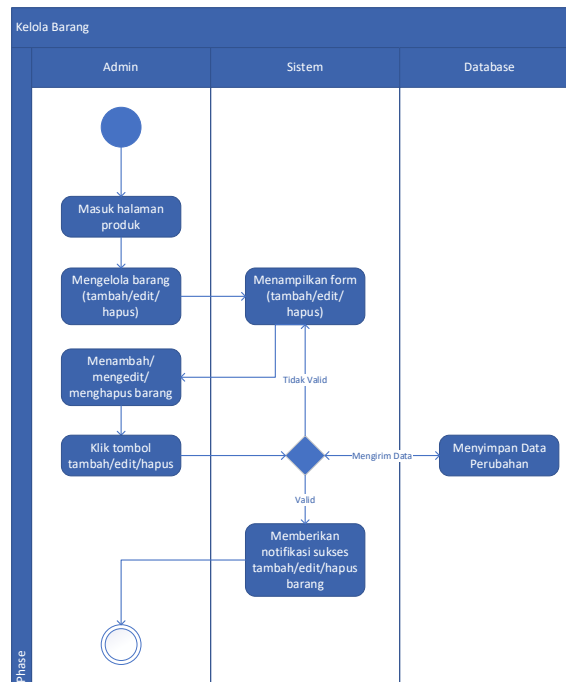
Gambar 3. *Use Case Diagram*

Pada gambar 3 *use case diagram* diatas menjelaskan apa saja yang aktor (admin dan kasir) lakukan dalam bentuk bulatan-bulatan aksi, admin sebagai aktor utama dapat mengelola barang, mengelola pegawai, mengelola supplier, mengelola member, mengelola laporan, mengelola transaksi, melihat dashboard. Sedangkan pada sisi kasir dapat menginput pesanan, melihat data dan detail barang, mencetak struk transaksi, serta melihat data transaksi. Pada *activity diagram* menjelaskan bagaimana tiap aktivitas yang terjadi pada aktor dilakukan, dari mulai aktivitas, *decision* yang mungkin terjadi, sampai dimana berakhirnya aktivitas tersebut, sehingga *activity diagram* ini menjelaskan secara detail bagaimana aktor bekerja di setiap aktivitas yang dilakukan.



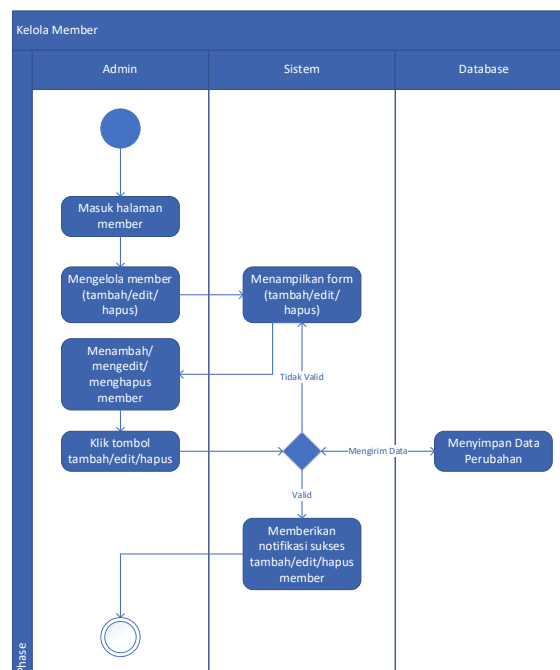
Gambar 4. *Activity Login dan Logout*

Pada gambar 4 dapat dilihat pertama kali admin atau kasir akan masuk ke halaman login awal dan memilih login sebagai admin atau kasir, kemudian sistem akan menampilkan halaman form login admin/kasir, jika admin atau kasir memasukkan *username* dan *password* dengan benar maka admin atau kasir akan diarahkan ke menu utama, jika salah admin atau kasir akan mengisi ulang kembali *username* dan *password* hingga benar.



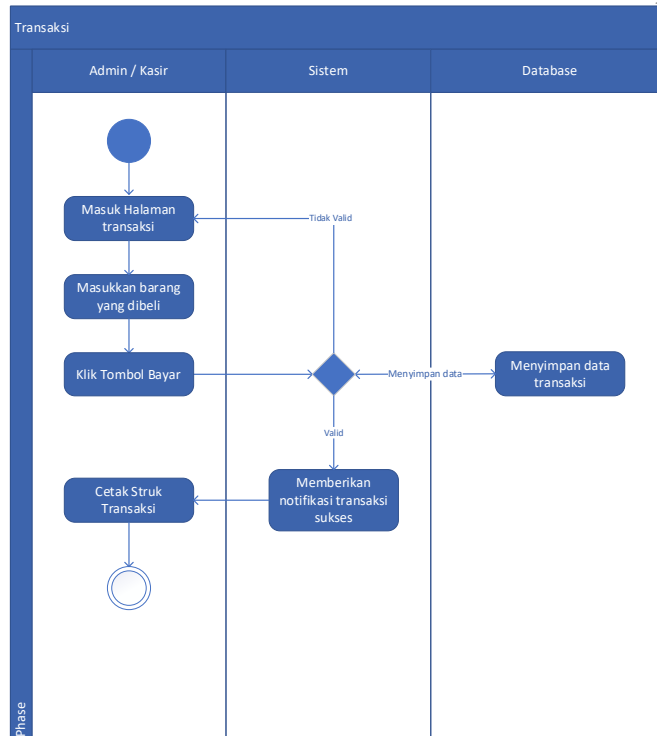
Gambar 5. Activity Kelola Barang

Pada gambar 5 dapat dilihat saat admin ingin mengelola barang, admin terlebih dahulu masuk ke menu data barang, lalu menekan tombol tambah/edit/hapus, setelah itu akan muncul *pop up* form tambah/edit/hapus sesuai opsi yang dipilih, kemudian admin akan mengeksekusinya, dan akan masuk ke *decision* apakah data yang dieksekusi sudah benar dan valid, jika sudah sistem akan menyimpan perubahan data dalam database dan sistem akan memberi *notice* bahwa berhasil tambah/edit/hapus barang, sebaliknya jika belum valid maka sistem akan memberikan *notice* sesuai opsi yang dipilih diawal.



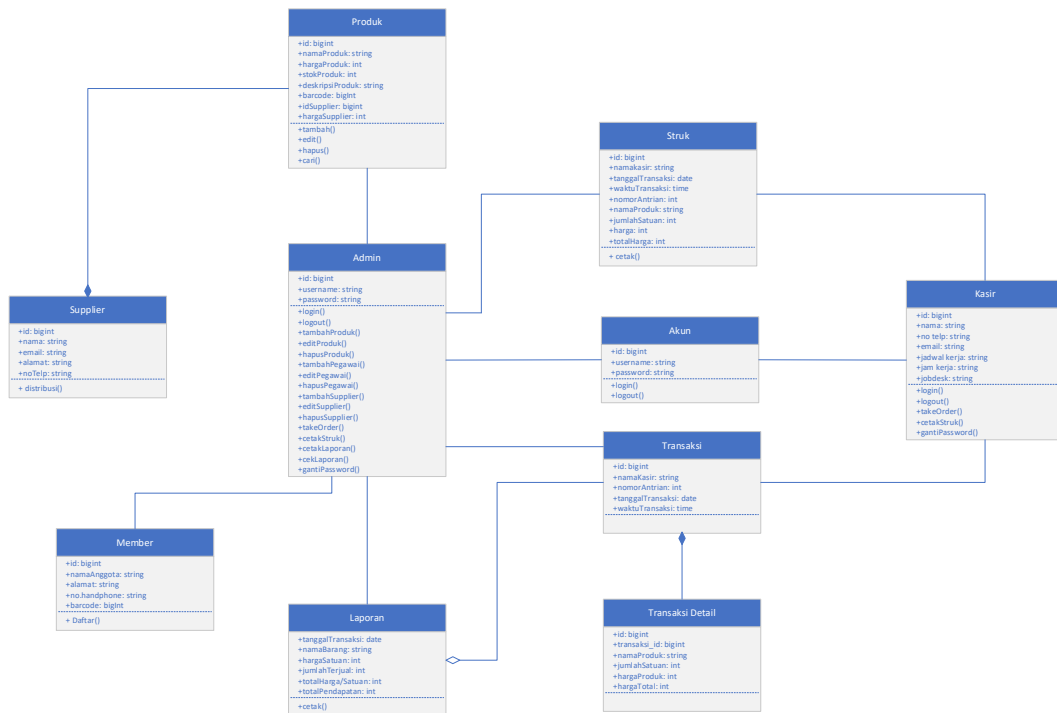
Gambar 6. Activity Kelola Member

Pada gambar 6 dapat dilihat saat admin ingin mengelola *member*, admin terlebih dahulu masuk ke menu data *member*, lalu menekan tombol tambah/edit/hapus, setelah itu akan muncul *pop up* form tambah/edit/hapus sesuai opsi yang dipilih, kemudian admin akan mengeksekusinya, dan akan masuk ke *decision* apakah data yang dieksekusi sudah benar dan valid, jika sudah sistem akan menyimpan perubahan data dalam database dan sistem akan memberi *notice* bahwa berhasil tambah/edit/hapus *member*, sebaliknya jika belum valid maka sistem akan memberikan *notice* sesuai opsi yang dipilih diawal.



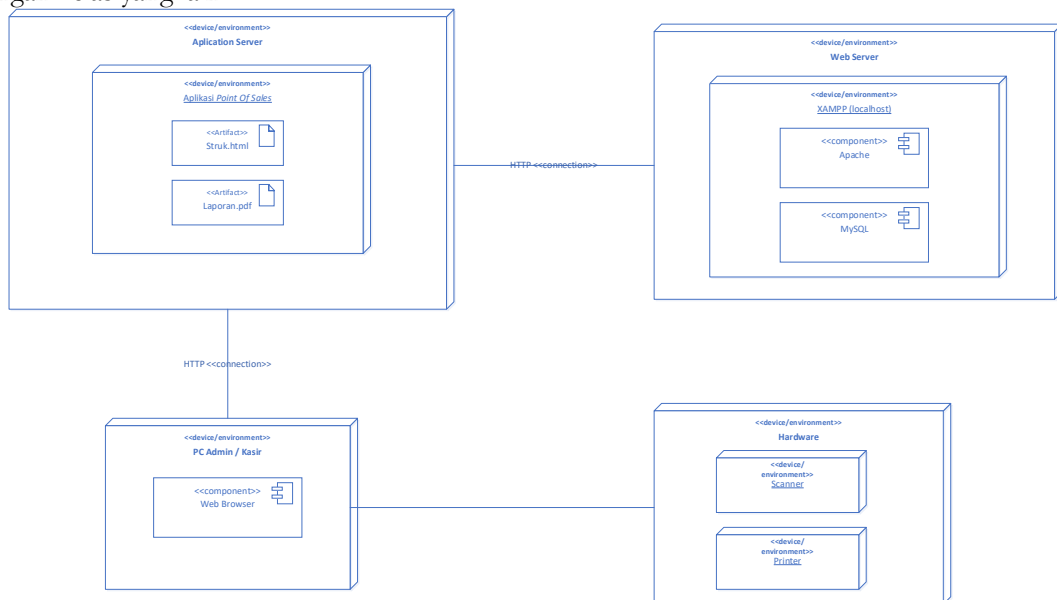
Gambar 7. *Activity* Transaksi

Pada gambar 7 dapat dilihat saat admin / kasir hendak melakukann transaksi dengan pembeli, admin terlebih dahulu masuk ke menu transaksi, lalu menginput barang-barang yang dibeli, setelah itu menekan tombol bayar dan akan masuk ke *decision* apakah data yang dimasukkan sudah benar dan valid, jika sudah sistem akan menyimpan data transaksi dalam database dan sistem akan memberi *notice* bahwa transaksi berhasil, sebaliknya jika belum valid maka sistem akan memberikan *notice* bahwa ada data yang belum di isi atau *error*.



Gambar 8. Class Diagram

Pada gambar 8 menjelaskan tentang *class diagram* dari aplikasi yang dibangun. *Class diagram* disini lebih menjelaskan kepada perancangan database nantinya, disini terdapat beberapa kelas yaitu admin, kasir, barang, akun, supplier, member, transaksi, transaksi detail, laporan, dan struk, dimana masing kelas memiliki atribut dan operasinya masing-masing serta satu kelas memiliki hubungan dengan kelas yang lain.



Gambar 9. Deployment Diagram

Pada gambar 9 dapat dilihat, *deployment diagram* dari aplikasi *point of sales* yang dibangun. Terdapat perangkat keras yang digunakan yaitu *Personal Computer (PC)*, *Scanner*, *Printer* dan beberapa perangkat lunak yang tersedia yakni *web browser*, *web server (Apache dan MySQL)* serta aplikasi *point of sales* itu sendiri.

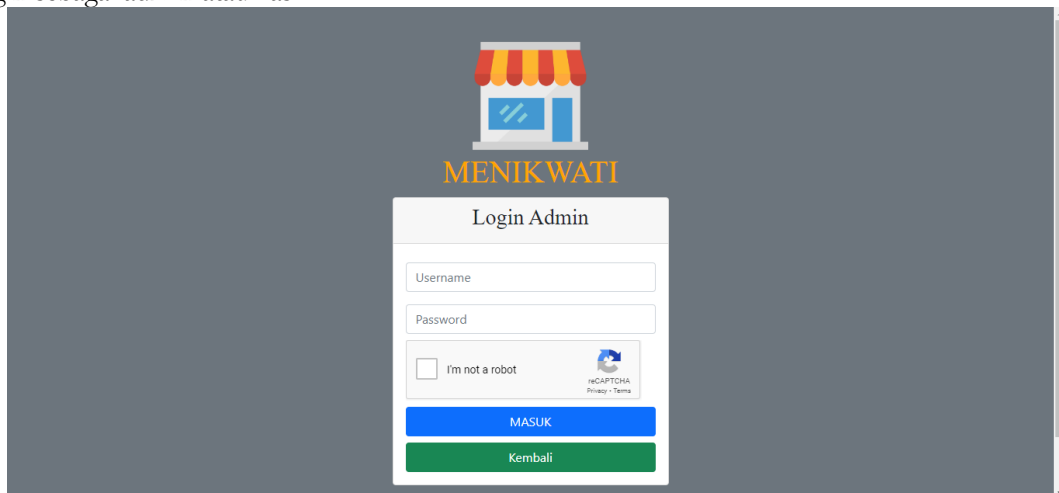
3. Hasil dan Pembahasan

Berikut merupakan hasil implementasi beserta hasil pengujian yang dilakukan menggunakan UAT (*User Acceptance Test*) dengan metode *black box testing*.



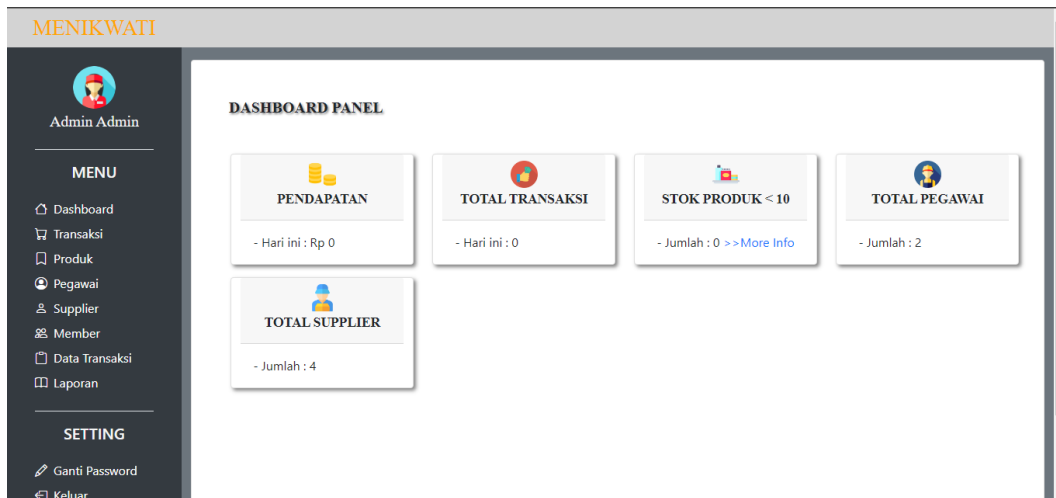
Gambar 10. Tampilan halaman login awal

Gambar 10 merupakan tampilan awal sebelum masuk ke menu utama. Pada tampilan ini *user* memilih login sebagai admin atau kasir.



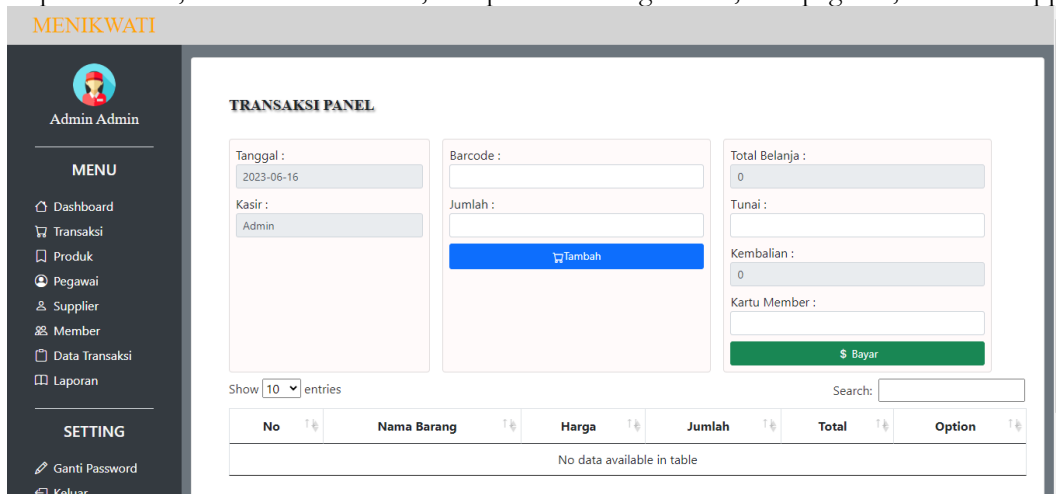
Gambar 11. Tampilan halaman login admin

Gambar 11 merupakan tampilan halaman login admin. Untuk halaman login kasir memiliki tampilan yang sama dengan login admin. Pada tampilan ini admin akan memasukkan *username* dan *password* serta mencentang *captcha* yang tersedia. Jika *username* dan *password* yang dimasukkan valid maka akan masuk ke halaman menu utama, sebaliknya jika salah akan tetap berada pada halaman yang sama.



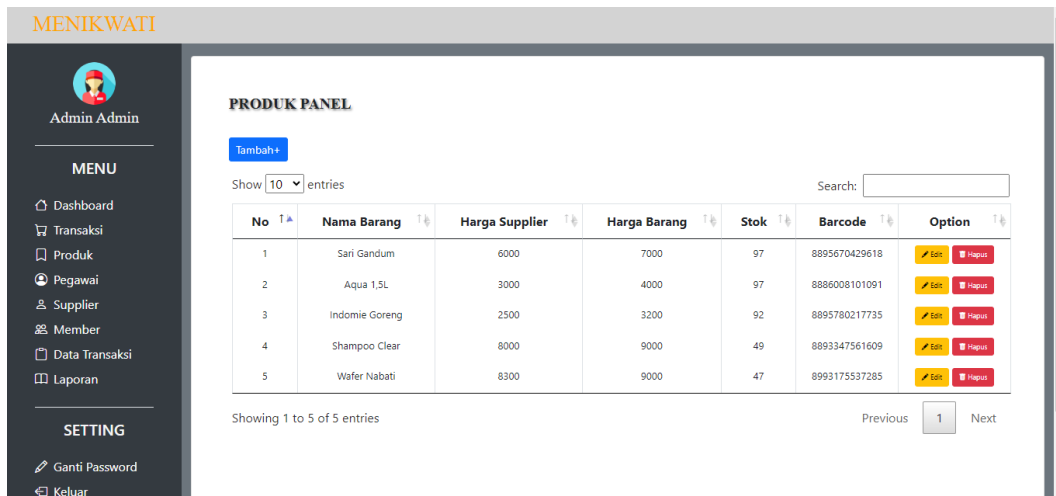
Gambar 12. Tampilan halaman dashboard

Gambar 12 merupakan tampilan halaman dashboard. Pada tampilan ini terdapat informasi yang berisi pendapatan harian, total transaksi harian, stok produk kurang dari 10, total pegawai, dan total supplier.



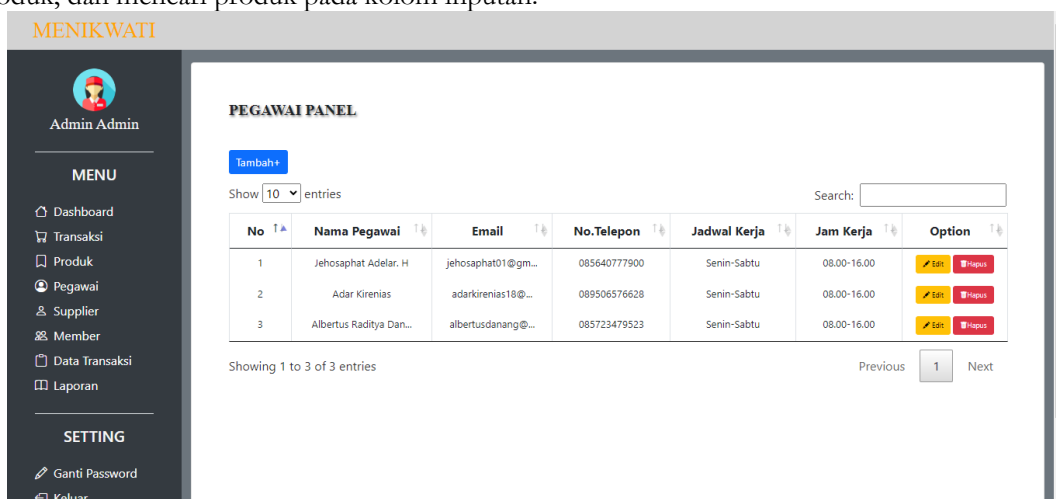
Gambar 13. Tampilan halaman transaksi

Gambar 13 merupakan tampilan halaman transaksi. Pada tampilan ini admin akan menginputkan produk yang dibeli dengan mengscan *barcode* produk yang menempel pada produk lalu menginputkan jumlah yang dibeli. Setelah semua produk masuk dalam tabel, admin akan menginput jumlah uang yang dibayar pembeli. Jika pembeli memiliki kartu *member* yang terdaftar sebagai penerima bantuan sosial PKH maka pembeli akan mendapatkan potongan harga atas total pembelian. Setelah itu admin menekan tombol bayar dan mencetak struk transaksi pada halaman data transaksi.



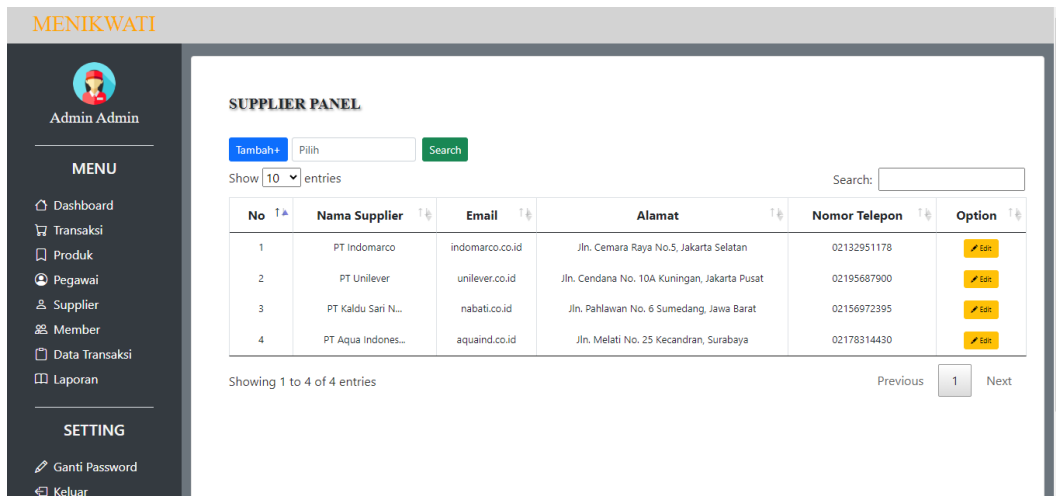
Gambar 14. Tampilan halaman produk

Gambar 14 merupakan tampilan halaman produk. Pada tampilan ini berisi daftar produk yang ada, selain itu admin dapat menambah produk pada tombol tambah diatas, mengedit produk, menghapus produk, dan mencari produk pada kolom inputan.



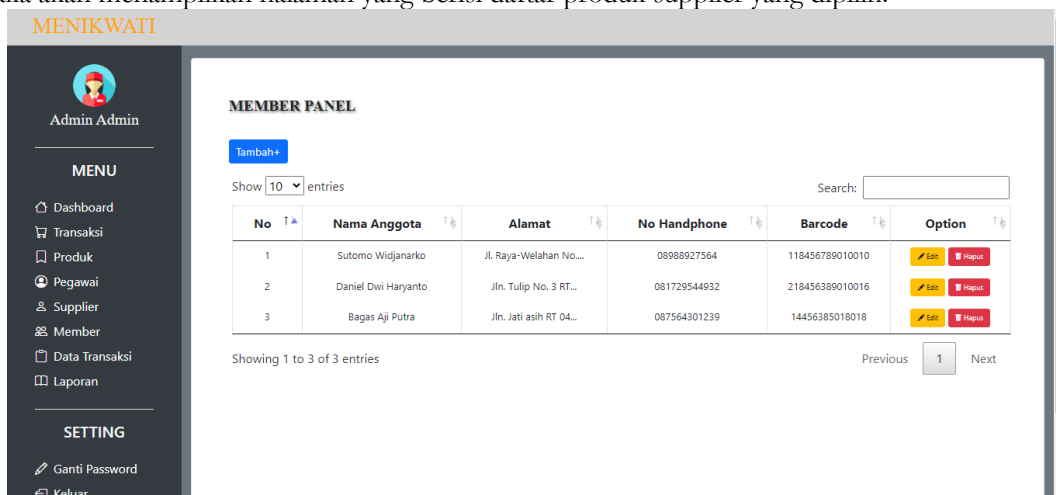
Gambar 15. Tampilan halaman pegawai

Gambar 15 merupakan tampilan halaman pegawai. Pada tampilan ini berisi daftar pegawai, selain itu admin dapat menambah pegawai pada tombol tambah diatas, mengedit pegawai, menghapus pegawai, dan mencari pegawai pada kolom inputan.



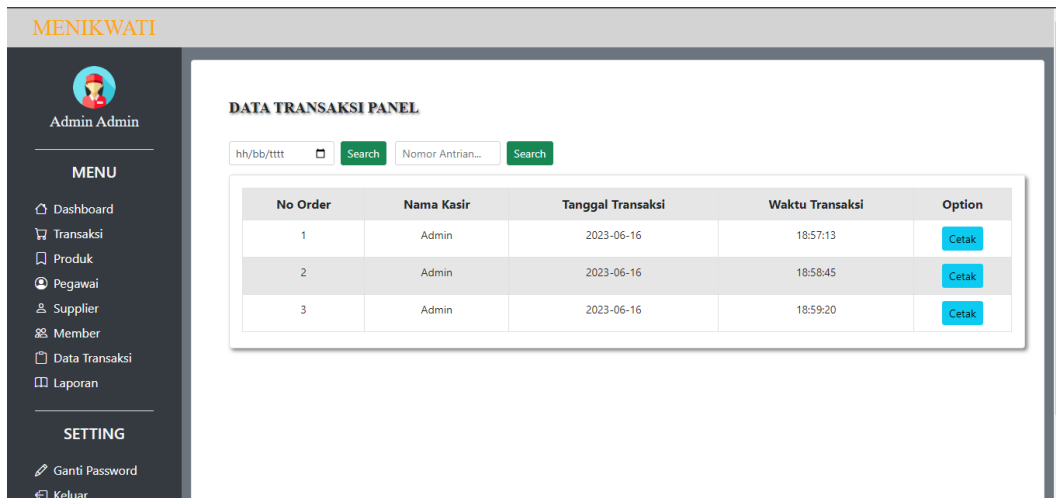
Gambar 16. Tampilan halaman supplier

Gambar 16 merupakan tampilan halaman supplier. Pada tampilan ini berisi daftar supplier, selain itu admin dapat menambah supplier pada tombol tambah diatas, mengedit supplier, dan mencari supplier pada kolom inputan. Admin juga dapat mengecek produk apa saja dari supplier dengan menekan tombol *dropdown* disebelah tombol tambah setelah itu memilih supplier lalu menekan tombol “*search*” maka akan menampilkan halaman yang berisi daftar produk supplier yang dipilih.



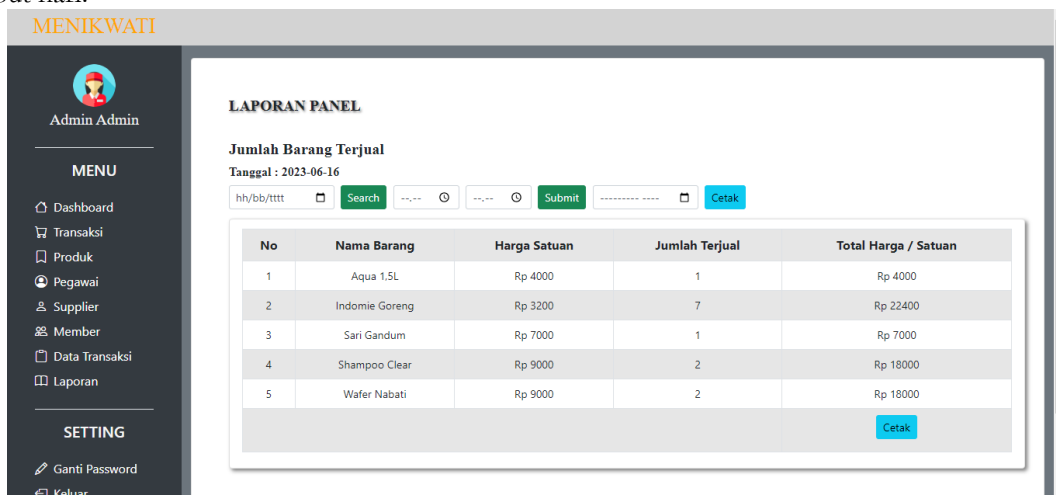
Gambar 17. Tampilan halaman member

Gambar 17 merupakan tampilan halaman member. Pada tampilan ini berisi daftar member yang terdaftar sebagai penerima bantuan sosial program PKH, selain itu admin dapat menambah member yang sebelumnya telah mengumpulkan berkas yang dibutuhkan dan mendapatkan surat keterangan tidak mampu dari kelurahan, mengedit data member, menghapus member, dan mencari member pada kolom inputan.



Gambar 18. Tampilan halaman data transaksi

Gambar 18 merupakan tampilan halaman data transaksi. Pada tampilan ini berisi struk transaksi atas transaksi yang terjadi pada hari itu, dan mencetak struk transaksi. Admin juga dapat melacak struk transaksi pada hari sebelumnya dengan memasukan data hari pada kolom inputan diatas, serta mencari salah satu struk transaksi di hari itu dengan memasukkan nomor antrian disebelah kolom input hari.



Gambar 19. Tampilan halaman laporan

Gambar 19 merupakan tampilan halaman laporan. Pada tampilan ini berisi jumlah barang yang terjual pada hari itu serta total pendapatan dari total jual per produk serta mencetaknya dalam bentuk pdf. Admin juga dapat mengecek laporan penjualan jumlah barang pada hari sebelumnya dengan menginputkan data hari pada kolom inputan hari diatas. Selain itu admin dapat melihat laporan penjualan barang secara *real time* dengan memasukkan rentang waktu dari jam berapa sampai jam berapa pada kolom inputan waktu. Dan yang terakhir admin juga dapat mencetak laporan penjualan barang bulanan dengan memasukkan data bulan dan tahun pada kolom inputan bulan dan tahun di sebelah kanan kolom inputan waktu.

Kode Program 1 Fungsi ajax *javascript* untuk mengirim data kartu member.

```
function kartuMember(nilai) {
    var brcd = nilai.value;
    var ttl = parseInt($('#ttl').val());
    if (ttl == 0) {
        alert("Harap Masukkan Barang Belanja!");
    }
    else {
        $.ajax({
            type: "GET",
            url: "/kartu/member/diskon",
            data: {
                _token: "{{ csrf_token() }}",
                kartuMember: brcd,
            },
            dataType: "json",
            success: function(response) {
                if (response.error) {
                    alert(response.message);
                }else {
                    cek = true;
                    var ttlFix = ttl-(ttl*1/4);
                    $('#ttl').val(ttlFix);
                    $('#ttlBlnj').val(ttl);
                    $('#diskon').val("25%");
                    $('#potongan').val(ttl*1/4);
                }
            },
            error: function(jqXHR, textStatus, errorMessage) {
                alert(errorMessage);
            }
        });
    }
}
```

Pada kode program 1 menjelaskan bagaimana mengirim data kartu member dari hasil mengscan *barcode* kartu *member*. Dengan menggunakan method GET, jika belum ada barang yang di input atau total belanja masih 0 maka sistem akan memberi *notice* “Harap Masukkan Barang Belanja!”, sebaliknya data akan dikirim, data tersebut berupa *barcode* kartu dan dikirim dalam bentuk data json. Setelah itu dilakukan pengecekan di controller apakah kartu member valid atau tidak. Jika tidak maka sistem akan memberi *notice error* berupa *alert*. Sebaliknya jika kartu member valid, maka total belanja keseluruhan akan mendapat potongan harga.

Kode Program 2 Fungsi untuk mengecek kartu *member*.

```
public function kartu_member(Request $request) {
    $kartu = $request->has('kartuMember') ? trim($request->kartuMember) : '';
    $getMember = DB::table('kartu_anggota')->where('barcode_kartu', $kartu)->first();
    if ($getMember) {
        $responseData = ['error' => false, 'message' => 'Kartu Telah Terdaftar!'];
        return response()->json($responseData);
    }
    else {
        $responseData = ['error' => true, 'message' => 'Maaf Kartu Member Tidak Ditemukan!'];
        return response()->json($responseData);
    }
}
```

Pada kode program 2 menjelaskan bagaimana mengecek kartu member apakah valid atau tidak. Dimulai dari mengecek apakah data *barcode* kartu masuk ke dalam fungsi. Setelah itu dilakukan pengecekan kevalidan kartu member dengan menggunakan fungsi *eloquent laravel*. Jika kartu *member* ditemukan atau valid maka sistem akan merespon atau mengembalikan nilai *error = false*. Sebaliknya jika kartu member tidak ditemukan atau tidak valid maka sistem akan merespon atau mengembalikan nilai *error = true*.

Pada pengujian sistem menggunakan metode *User Acceptance Test (UAT)* dengan teknik pengujian *black box testing*, sehingga semua modul dan sub sistem dapat berfungsi sesuai yang diharapkan atau masih ada *bug*. Berikut merupakan hasil pengujian sistem pada setiap sub sistem dan modul pada aplikasi *point of sales*.

Tabel 1. Tabel Pengujian Aplikasi

Fungsi yang diuji	Kondisi	Hasil Uji	Hasil
Login Admin / Kasir	<i>Username</i> dan <i>Password</i> benar	Login berhasil (<i>UI Dashboard</i>)	Valid
	<i>Username</i> dan <i>Password</i> salah	Login gagal (<i>UI login admin / kasir</i>)	Valid
Transaksi	Semua data di isi dengan benar	Data transaksi berhasil di simpan, sistem memberi <i>notice</i> transaksi sukses	Valid
	Data di isi sebagian atau ada data yang kosong	Sistem memberi <i>notice</i> sesuai <i>error</i> yang ada	Valid
Tambah Barang	Form tambah barang di isi semua dengan benar	Data Barang berhasil di simpan, sistem memberi <i>notice</i> barang berhasil di simpan	Valid
	Data di isi sebagian atau ada data yang kosong	Sistem memberi <i>notice</i> sesuai <i>error</i> yang ada	Valid
Edit Barang	Isi form edit barang di edit dengan benar	Data barang berhasil di <i>update</i> , sistem memberi <i>notice</i> barang berhasil di hapus	Valid
	Isi form edit barang hanya di isi sebagian atau ada data yang kosong	Sistem memberi <i>notice</i> sesuai <i>error</i> yang ada	Valid
Hapus Barang	<i>Button</i> hapus di klik, kemudian menekan <i>button</i> hapus pada jendela <i>pop up</i>	Data barang berhasil di hapus, sistem memberi <i>notice</i> data barang berhasil di hapus	Valid
	<i>Button</i> hapus di klik, kemudian menekan <i>button</i> batal pada jendela <i>pop up</i>	Kembali ke <i>UI</i> data barang	Valid

Tambah <i>Member</i>	Form tambah <i>member</i> di isi semua dengan benar	Data <i>member</i> berhasil di simpan, sistem memberi <i>notice</i> barang berhasil di simpan	Valid
	Data di isi sebagian atau ada data yang kosong	Sistem memberi <i>notice</i> sesuai <i>error</i> yang ada	Valid
Edit <i>Member</i>	Isi form edit <i>member</i> di edit dengan benar	Data barang berhasil di <i>update</i> , sistem memberi <i>notice</i> barang berhasil di hapus	Valid
	Isi form edit <i>member</i> hanya di isi sebagian atau ada data yang kosong	Sistem memberi <i>notice</i> sesuai <i>error</i> yang ada	Valid
Hapus <i>Member</i>	<i>Button</i> hapus di klik, kemudian menekan <i>button</i> hapus pada jendela <i>pop up</i>	Data <i>member</i> berhasil di hapus, sistem memberi <i>notice</i> data <i>member</i> berhasil di hapus	Valid
	<i>Button</i> hapus di klik, kemudian menekan <i>button</i> batal pada jendela <i>pop up</i>	Kembali ke <i>UI</i> data <i>member</i>	Valid

Dapat dilihat pada Tabel 1 semua fungsi yang diuji berjalan dengan baik dan hasil yang diharapkan telah sesuai, sehingga aplikasi siap untuk digunakan.

4. Kesimpulan

Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa implementasi kartu *member* pada aplikasi *point of sales* berbasis *web* dalam mendukung program PKH (Studi Kasus : Toko Menikwati) berhasil dibangun. Adanya fitur kartu *member* membuat perekonomian khususnya keluarga yang menyandang status keluarga miskin lebih makmur dan sejahtera sehingga keluarga miskin dapat terpenuhi kebutuhannya, dan secara langsung membantu mensukseskan program PKH (Program Keluarga Harapan) serta mengurangi dan memutus angka kemiskinan.

5. Daftar Pustaka

- [1] Sugihartono, J., Satoto, K. I., & Widiyanto, E. D. (2015). Pembuatan Aplikasi Point of Sale Toko Cabang Perusahaan Torani Menggunakan Framework CodeIgniter. *Jurnal Teknologi dan Sistem Komputer*, 3(4), 445-455.
- [2] Arman, M., & Maberur, R. (2022). Perancangan Aplikasi Point Of Sales Pada Toko Cahaya Purnama Soppeng. *Jurnal Ilmiah Sistem Informasi dan Teknik Informatika (JISTI)*, 5(1), 43-50. DOI: <https://doi.org/10.57093/jisti.v5i1.108>.
- [3] Saragi, S., Batoebara, M. U., & Arma, N. A. (2021). Analisis Pelaksanaan Program Keluarga Harapan (Pkh) Di Desa Kota Rantang Kecamatan Hamparan Perak. *Publik: Jurnal Manajemen Sumber Daya Manusia, Administrasi Dan Pelayanan Publik*, 8(1), 1-10. DOI: <https://doi.org/10.37606/publik.v8i1.150>.

- [4] Permatasari, A., & Suhendi, S. (2020). Rancang Bangun Sistem Informasi Pengelolaan Talent Film berbasis Aplikasi Web. *Jurnal Informatika Terpadu*, 6(1), 29-37. DOI: <https://doi.org/10.54914/jit.v6i1.255>.
- [5] Hidayat, A. T. (2019). Perancangan Sistem Informasi Tingkat Kemiskinan Di Kota Lubuklinggau Berbasis Web Menggunakan Framework Codeigniter. *JUSIM (Jurnal Sistem Informasi Musirawas)*, 4(02), 97-108.
- [6] Yudanto, A. L., Tolle, H., & Brata, A. H. (2017). Rancang Bangun Aplikasi Sistem Informasi Manajemen Laboratorium Biomedik Fakultas Kedokteran Universitas Brawijaya. *Jurnal pengembangan teknologi informasi dan ilmu komputer*, 1(8), 628-634.
- [7] Handika, I. G. (2018). *Pemanfaatan Framework Laravel Dalam Pembangunan Aplikasi E-Travel Berbasis Website* (Doctoral dissertation, Fakultas Teknik Unpas).
- [8] Parlaungan, S., Faritcan, T., & Wisnu, D. (2020). Rancang Bangun Sistem Pengidentifikasi Travel Bag Pada Kelompok Biro Perjalanan Umroh/Haji Berbasis Web. *Jurnal Teknologi Informasi dan Komunikasi*, 15(1), 26-40.
- [9] Ramadhan, M. G. (2020). Perancangan Sistem Informasi Pos (Point of Sales) Berbasis Web Dengan Menggunakan Framework Codeigniter Pada Pasar Swalayan. *Electrician: Jurnal Rekayasa dan Teknologi Elektro*, 14(3), 76-83.
- [10] Wiguna, P. D. A., Swastika, I. P. A., & Satwika, I. P. (2018). Rancang bangun aplikasi point of sales distro management system dengan menggunakan framework react native. *Jurnal Nasional Teknologi Dan Sistem Informasi*, 4(3), 149-159.
- [11] Amilia, N. A., Nurkamid, M., & Listyorini, T. (2021). Perancangan Dan Penerapan Sistem Penjualan Berbasis Web Pada Restoran Tengkleng Mbah Ninik. *Jurnal Dialektika Informatika (Detika)*, 2(1), 34-39. DOI: <https://doi.org/10.24176/detika.v2i1.6890>.
- [12] Putra, I. G. A. K., Sudana, A. A. K. O., & Raharja, I. M. S. Sistem Informasi Manajemen Bengkel Modul Point of Sales Berbasis Web. *Jurnal Ilmiah Teknologi dan Komputer*, 2(3), 595-606.
- [13] Wahid, A. A. (2020). Analisis metode waterfall untuk pengembangan sistem informasi. *J. Ilmu-ilmu Inform. dan Manaj. STMIK*, no. November, 1-5.
- [14] Simatupang, J., & Sianturi, S. (2019). Perancangan sistem informasi pemesanan tiket bus pada po. Handoyo berbasis online. *Jurnal Intra Tech*, 3(2), 11-25. DOI: <https://doi.org/10.37030/jit.v3i2.56>.