

Rancang Bangun Perangkat Lunak Mobile Front End Virtual Shopping Cart dan Infrastruktur Jaringan Untuk Bisnis Retail Hypermarket

Christian Hendranata Tajudin, Teknik informatika, UC Town, Citraland, Surabaya 60219

ABSTRAK

Dalam proses membeli produk di toko-toko besar yang hanya menampilkan contoh barang. Pelanggan kebingungan untuk melihat detail barang yang akan dibeli. Kurangnya informasi membuat pelanggan sibuk mencari pegawai untuk bertanya mengenai detail barang. Tujuan dari penelitian ini adalah membuat rancangan bangun *front end* dan infrastruktur jaringan untuk sistem yang meningkatkan kenyamanan *customer* berbelanja di bisnis retail hypermarket. Aplikasi ini mempunyai fitur *scan barcode* dimana terdapat *qr-code* di masing-masing barang untuk di *scan* dan dapat melihat detail barang seperti id, nama, dimensi, jumlah, warna, harga, dan gambar barang. Aplikasi ini juga mempunyai fitur *checkout* yang cepat. Desain infrastruktur menggunakan jenis topologi star dimana setiap pelanggan dan program kasir harus terkoneksi dengan *router* yang telah disediakan oleh *hypermarket*. Aplikasi ini bernama *Virtual Shopping Cart*. Perangkat lunak ini telah di implementasikan dan di uji terhadap 20 responden yang mengatakan bahwa responden tidak mengalami kesulitan dengan menggunakan perangkat lunak ini.

Kata kunci: Perangkat lunak mobile, sistem belanja virtual, desain infrastruktur, hypermarket, pemidai kode

1. Pendahuluan

Dalam proses membeli produk di toko-toko besar yang hanya menampilkan contoh barang, seringkali pelanggan kebingungan mencari informasi detail barang. Hal tersebut terjadi karena umumnya informasi yang ditampilkan pada barang terbatas hanya harga, merk, dan ukuran. Pada saat seperti itu pelanggan biasanya mencari pegawai untuk menanyakan hal lain seperti varian warna, ukuran, dan jumlah barang yang tersedia.

Permasalahan lain terjadi ketika pelanggan berniat membeli suatu barang pelanggan harus membawa sendiri barang tersebut ke kasir. Kerepotan akan dirasakan pelanggan apabila barang tersebut berukuran besar atau dalam jumlah banyak. Hal tersebut tentu mengurangi kenyamanan pelanggan untuk berbelanja. Seperti yang kita ketahui di beberapa hypermarket proses berbelanja menggunakan kereta belanja yang besar dan kadang kala pelanggan harus memanggil pegawai untuk bertanya lebih jelas mengenai informasi detail barang yang ingin di beli.

Penelitian ini menawarkan *Virtual Shopping Cart* sebagai solusi dari permasalahan di atas. Dengan *Virtual Shopping Cart* pelanggan hanya membawa gadget yang berukuran kecil tanpa harus membawa kereta belanja yang besar untuk berbelanja pada toko tersebut. Adapun fitur-

fitur yang didapatkan dari *Virtual Shopping Cart* ini yaitu *scan barcode* yang dapat digunakan pelanggan untuk mendapatkan detail barang, pada *virtual shopping cart* ini pelanggan juga dapat melihat list belanjaan yang sudah di beli serta pelanggan dapat melakukan *checkout* yang cepat pada saat di kasir.

Ruang lingkup dari karya ilmiah ini adalah

- a. *Aplikasi front end mobile* ini dibangun dalam sistem operasional android.
- b. Sistem yang digunakan pada aplikasi ini adalah QR-Code untuk pemidai kode barang.
- c. *Virtual Shopping Cart* mampu menampilkan detail dari barang seperti jenis barang, stok, warna, dimensi, dan harga barang.
- d. *Virtual shopping cart* mempunyai list barang-barang yang ingin dibeli, dan dilengkapi dengan biaya total belanja.

Yang bertujuan untuk membuat rancang bangun *front end* dan infrastruktur jaringan untuk system retail *hypermarket*. Adapun manfaat dari pembuatan aplikasi ini bagi pengguna adalah memberikan kenyamanan pengguna berbelanja di bisnis retail *hypermarket*. Manfaat bagi perusahaan adalah perusahaan dapat menekan biaya untuk investasi kereta belanja. Manfaat bagi pembuat adalah menerapkan ilmu

yang didapat kedalam perangkat lunak yang dapat diimplementasikan.

2. Landasan Teori

Pada bab ini dijelaskan tentang teori yang melandasi karya ilmiah ini antara lain :

2.1. Barcode

Menurut (Querini, Grillo, Lentini, & Italiano, 2011).

Barcode adalah representasi optik yang dapat dibaca mesin data, mampu menyimpan informasi digital tentang obyek fisik yang mereka pasang. Disebabkan oleh kecepatan mereka membaca, akurasi, dan fungsi karakterisi, barcode telah ada di mana-mana dalam berbagai perangkat lunak, termasuk penggunaan mereka di *department store* dan rantai ritel untuk harga barang, untuk melacak item dan mengidentifikasi member card, dan lain-lain. Barcode dibagi menjadi 2 yaitu *barcode* tradisional (1D), mewakili data dengan memvariasikan lebar dan jarak dari garis paralel. Jumlah digital informasi yang disimpan dalam barcode 1D terbatas dan hanya meningkat dengan meningkatnya jumlah digit barcode atau dengan meletakkan beberapa barcode. Dan ini memiliki banyak efek negative. Teknologi barcode telah menyebarkan pola geometris (seperti kotak, titik, segitiga, segi enam) yang disebut barcode 2D (QR-Code) Seperti pada Gambar 2.1 di bawah.



Gambar 2.1 Dimensi penyimpanan data (a) 2D, dan (b)1D

QR Code(Quick Response Code)

Menurut (Querini, Grillo, Lentini, & Italiano, 2011) QR-Code adalah code 2D yang dikembangkan oleh Denso Wave, Sebuah divisi dari Denso Corporation dengan waktu, dan dirilis pada tahun 1994 dengan tujuan utama yang dengan mudah ditafsirkan oleh peralatan *scanner*. Qr-Code adalah merek dagang terdaftar dari Denso Wave Incorporated di Jepang dan Negara-negara lainnya. Qr-Code mampu menyimpan lebih dari 7000 desimal digit.

2.2. Android SDK(Software Development Kit)

Android SDK adalah perangkat API(Application Programming Interface) yang diperlukan untuk mendesain dan merancang perangkat lunak pada platform *Android* (Reto, 2010) menggunakan Bahasa pemrograman java. Android adalah subset perangkat lunak untuk ponsel yang meliputi system operasi, *middleware*, dan perangkat lunak kunci yang telah dikeluarkan oleh Google. Android SDK mempunyai isi yang terdiri dari :

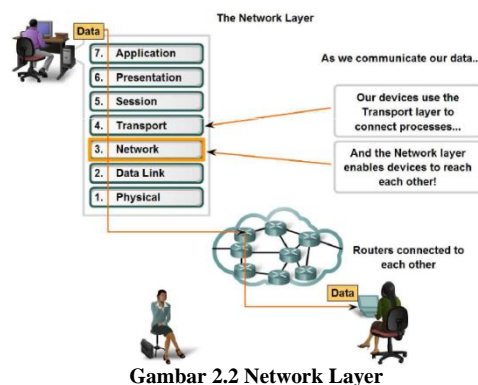
1. Android API(Application Programming Interface) yang merupakan inti dari SDK android API

merupakan kumpulan folder untuk mengakses kelas-kelas dan kode milik android.

2. Development Tools berguna untuk melakukan *compile* dan *debug* perangkat lunak
3. *Android Virtual Device Manager and Emulator* merupakan emulator android interaktif dengan berbagai macam *skin*. Emulator yang merupakan emulator android berjalan di dalam AVD(*Android Virtual Device*) yang menyimulasikan konfigurasi perangkat keras. Emulator ini berguna untuk melihat bagaimana tampilan dan kegunaan perangkat lunak android yang sedang dibangun di dalam peralatan *mobile*. Perangkat lunak ini bergerak dalam *dalvik virtual machine*.

2.3. Open System Interconnection (OSI) Layer

Menurut Paul Simoneau (2010), OSI model adalah alat referensi untuk memahami komunikasi data antara dua system jaringan. Ini membagi proses komunikasi menjadi tujuh lapisan. Setiap lapisan melakukan fungsi spesifik untuk mendukung lapisan di atasnya dan menawarkan layanan ke lapisan di bawahnya. Tiga lapisan terendah fokus pada lalu lintas yang lewat melalui jaringan untuk sistem akhir. Bagian atas empat lapisan berperan dalam system akhir untuk menyelesaikan proses. Model OSI Layer dibagi menjadi dua grup yaitu “Lower Layer” dan “Upper Layer”. Lower layer adalah physical, data link, network, dan transport. Upper Layer adalah session, presentation, dan application. Upper layer fokus pada aplikasi pengguna dan bagaimana dipresentasikan di komputer. Untuk *network engineer*, bagian utama yang menjadi perhatiannya adalah pada “Lower Layer”. Lower Layer adalah intisari komunikasi data melalui jaringan aktual. Network Layer dapat dilihat seperti Gambar 2.2 dibawah.



Gambar 2.2 Network Layer

2.4. Router

Menurut (Wirawan & Sumarianta, 2011), Router adalah salah satu komponen pada jaringan komputer yang mampu melewatkan data melalui sebuah jaringan atau internet menuju sasarannya, melalui sebuah proses yang dikenal sebagai *routing*. Proses *routing* dilakukan dengan memasukkan informasi suatu alamat jaringan secara manual kedalam table *routing* ataupun dengan bantuan *protocol routing*. Sama seperti computer lainnya, termasuk PC.

2.5. Topologi Jaringan

Terdapat tiga jenis topologi dalam dunia jaringan komputer antara lain :

1. Topologi Cincin

Pada topologi ini, data atau message berjalan mengelilingi jaringan dengan satu arah pengiriman ke komputer selanjutnya, terus hingga mencapai komputer yang dituju

2. Topologi Bus

Topologi jenis ini menggunakan sebuah kabel pusat yang merupakan media utama dari jaringan. Terminal-terminal yang akan membangun jaringan dihubungkan dengan kabel utama yang merupakan inti dari jaringan. Data yang dikirimkan akan langsung menuju terminal yang dimaksud tanpa harus melewati terminal-terminal dalam jaringan, atau yang akan diroutingkan ke *head end controller*.

3. Topologi Bintang

Topologi jaringan ini menggunakan satu terminal sebagai terminal sentral yang menghubungkan ke semua terminal *client*. Terminal sentral ini akan mengarahkan setiap data yang dikirimkan ke komputer yang dituju. Apabila ada satu client yang tidak berfungsi atau media transmisi yang putus maka tidak akan mempengaruhi kerja dari jaringan karena gangguan tersebut hanya mempengaruhi terminal yang bersangkutan.

2.6. ZXing(Zebra Crossing)

ZXing ("zebra cross") adalah sebuah open-source, multi-format 1D / 2D perpustakaan pengolahan gambar barcode dilaksanakan di Java, dengan port untuk bahasa lain. ZXing dapat digunakan untuk meng-encode dan decode barcode untuk desktop dan server juga. Untuk saat ini format yang dukung oleh ZXing adalah UPC-A dan UPC-E, EAN-8 dan EAN-13, Code 39, Code 128, QR Code, Data matrix, PDF 417, ITF, RSS-14, RSS-Expanded.

3. Analisa dan Desain Sistem

Pada bab ini dijelaskan tentang analisa yang akan dilakukan serta desain dari system yang akan di implementasi.

3.1. Analisa Kebutuhan

Berikut ini akan dibahas analisis kebutuhan dari perangkat lunak yang akan dibuat yaitu kebutuhan aplikasi front end yang membahas mengenai analisa fitur dari aplikasi yang di buat oleh penulis yang melakukan survey dengan menggunakan kuisioner melalui google form yang disebarakan melalui media sosial di www.facebook.com yang bertujuan untuk mengetahui tingkat kemudahan dan kenyamanan pelanggan dalam berbelanja di suatu hypermarket. Dengan pertanyaan yang mencakup

1. Keseringan pelanggan berbelanja barang-barang besar di hypermarket

2. Ketertarikan pelanggan pada sebuah produk di hypermarket dan detail apa saja yang ingin diketahui oleh pelanggan
3. Ketertarikan pelanggan jika aplikasi mobile menyediakan informasi detail barang tanpa harus memanggil pegawai
4. Kesiapan pelanggan memberikan informasi apa saja untuk aplikasi.

Hasil dari perhitungan pertanyaan yang diberikan kepada responden adalah pelanggan ingin memperoleh informasi produk terutama tentang gambar, warna, stok, dimensi, dan harga barang. Pelanggan mengharapkan mendapat detail produk yang jelas tanpa harus memanggil pegawai toko dan juga pelanggan bersedia memberikan informasi nama dan email sebagai data dari aplikasi.

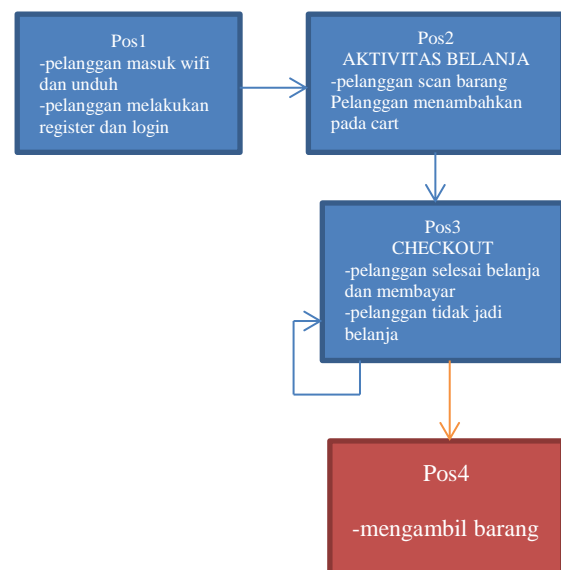
3.2. Desain Fitur

Berdasarkan analisa kebutuhan tersebut maka fitur-fitur *virtual shopping cart* adalah Scan Barcode, Detail Produk, View Cart, dan Checkout.

1. fungsi detail produk yang didahului dengan fitur scan barcode. Fungsi ini menampilkan detail produk seperti nama, harga, warna, dimensi, dan stok produk.
2. Fitur *View Cart* yang adalah tempat menyimpan barang yang akan di beli. Fitur *shopping cart* ini mempunyai fitur menghapus, mengganti, dan menambah stok dan jumlah barang.
3. Fungsi *Shopping cart* juga dilengkapi dengan fitur *checkout*. Fungsi ini mempercepat proses di kasir.

3.2.1. Aliran aktivitas pelanggan

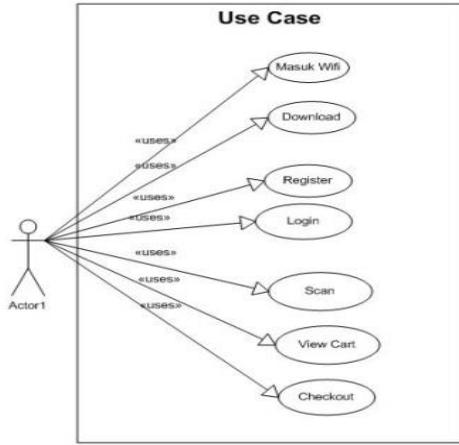
Aktivitas pelanggan sebuah toko, terkait dengan penggunaan aplikasi shopping cart terbagi menjadi empat pos. Pos pertama berhubungan dengan user masuk wifi, melakukan register dan login. Pos ke dua adalah aktivitas belanja. Pos ke tiga adalah proses checkout dan pos ke empat adalah proses pengambilan barang. Seperti pada Gambar 3.1.



Gambar 3.1 Aliran aktivitas pelanggan

3.2.2. Use Case Diagram

Yang dapat dilakukan oleh pelanggan pada saat berbelanja pada hypermarket adalah mencari wifi toko kemudian pelanggan baru mengunduh aplikasi melalui *website* aplikasi dan kemudian pelanggan membuat akun baru lalu masuk ke *login*, kemudian pelanggan dapat memidai qr code dari barang sehingga bisa menampilkan fitur dari detail barang. Setelah pelanggan cocok dengan barang maka barang akan masuk ke shopping list, hingga proses transaksi di kasir berlangsung. Pada Gambar 3.2 di bawah adalah fitur yang dijabarkan dengan diagram use case.



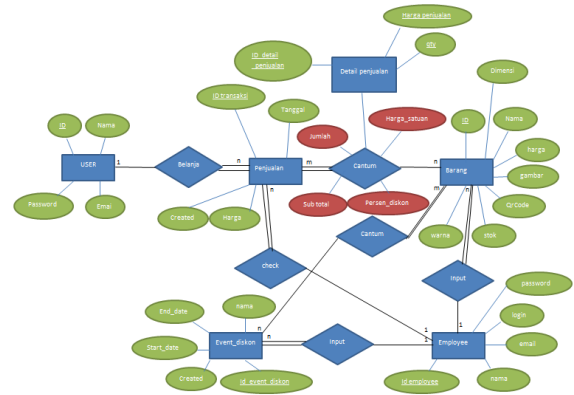
Gambar 5.2 Use Case Diagram

Gambar 3.2 di atas adalah Use case diagram yang mempunyai fitur-fitur sebagai berikut :

- Masuk wifi, Pelanggan harus masuk wifi untuk dapat menjalankan dan mendownload aplikasi *virtual shopping cart*.
- Download*, Untuk dapat menggunakan aplikasi user harus mendownload terlebih dahulu.
- zRegister*, ini adalah halaman untuk membuat akun baru jika pelanggan belum pernah menggunakan aplikasi ini.
- Login*, halaman ini adalah halaman untuk dapat menggunakan aplikasi dan memulai proses belanja di *hypermarket*.
- Scan*, ini adalah fitur scan barcode yang digunakan pelanggan untuk berbelanja. Dengan menggunakan scan barcode pelanggan dapat melihat detail barang.
- View Cart* adalah halaman dimana semua barang yang telah di scan dan di masukan ke cart akan tampil beserta stok dan harga. Dalam halaman ini pelanggan bisa mengganti ataupun menghapus produk dan detail produk.
- Checkout*, fitur ini adalah fitur untuk mengakhiri transaksi. Pelanggan dapat melakukan checkout juga keluar tanpa pembelian.

3.2.3. Diagram ER

Berdasarkan kebutuhan data yang dijelaskan dari diagram aktifitas, maka rancangan fisikal databasenya seperti pada Gambar 3.4 di bawah.



Gambar 4.3 ER Diagram

3.2.4. Pengambilan ID menggunakan barcode

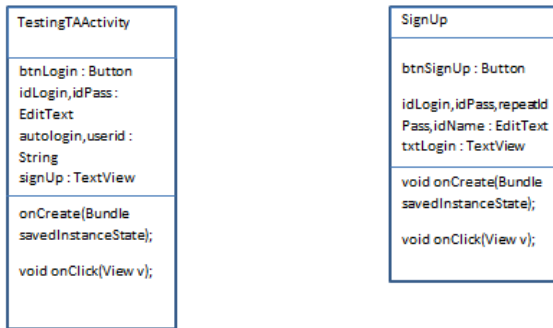
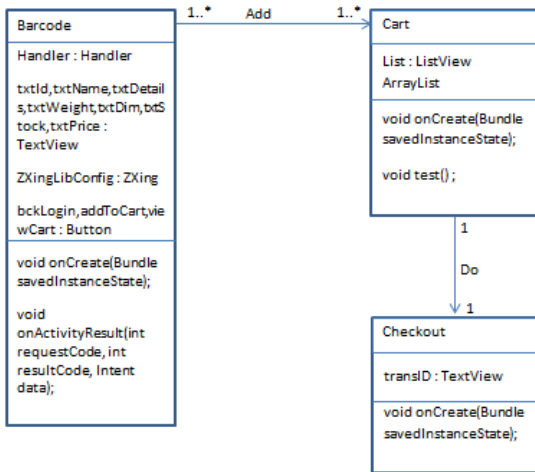
ID barang yang digunakan adalah berupa Qr Code. Qr Code yang secara otomatis tergenerate dengan adanya kode yang telah dibuat oleh programmer mengubah gambar menjadi angka. Ketika pelanggan memidai gambar yang terbuat dari angka (sudah tergenerate menjadi Qr Code) yang ada di masing-masing produk, aplikasi secara otomatis memidai gambar dan mengubah gambar menjadi angka setelah itu angka mencocokkan dengan id barang yang ada di database. Setelah aplikasi mencocokkan id barang, aplikasi kemudian mengambil detail barang dan menampilkannya di tampilan handphone pelanggan.

3.2.5. Pengubahan stok barang

Pengubahan stok barang terjadi ketika pelanggan ada di fitur *view cart*. Pelanggan dapat mengubah stok dengan mengganti jumlah stok yang tersedia. Untuk mengganti stok ataupun barang dengan yang baru pelanggan harus menekan tombol *save* agar stok maupun barang yang telah diperbarui yang ada di detail penjualan bisa terganti dengan yang baru. Menghapus barang juga dapat dilakukan di fitur *view cart* ini, dan apapun yang dilakukan di fitur *view cart* pelanggan harus menekan tombol *save*.

3.2.6. Diagram class

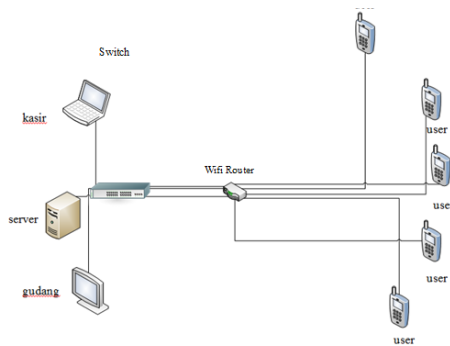
Diagram class menjelaskan scan barcode dijelaskan *scan barcode* dapat masuk atau tidak masuk ke dalam kereta belanja, barang dapat masuk kedalam kereta belanja sebanyak yang disediakan oleh gudang dan selama persediaan masih ada. Setiap yang ada di kereta belanja hanya dapat melakukan satu kali *checkout* dan setelah pelanggan menekan *checkout* maka nomer id penjualan akan muncul dan harus di konfirmasi ke kasir untuk melakukan *checkout* lebih lanjut dan proses belanja selesai. Seperti pada Gambar 3.4 dibawah.



Gambar 9.4 Class diagram

3.3. Desain Infrastruktur

Infrastruktur jaringan ini menggunakan topologi star dengan desain seperti pada Gambar 3.5 dibawah. Pelanggan yang ingin berbelanja harus mengkoneksikan gadget nya dengan otentikasi ke wifi toko yang telah disediakan untuk melakukan proses belanja. Komputer kasir, gudang, dan server juga harus terkoneksi ke switch yang berguna untuk mengatur tujuan dan sumber paket data.



Gambar 7.5 Desain Infrastruktur

4. Implementasi dan pengujian

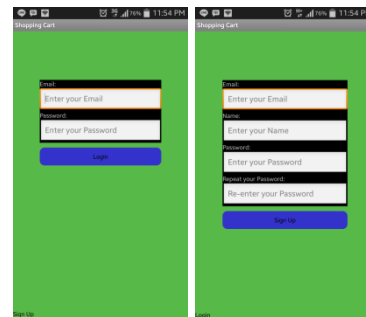
4.1. Konfigurasi infrastruktur jaringan

Pada bab ini dijelaskan tentang konfigurasi infrastruktur jaringan yang akan digunakan oleh pelanggan toko yang akan berbelanja di *hypermarket*. Konfigurasi dari router dengan memasukkan IP dari router dan masuk ke dalam UI dari router setting. Kemudian melakukan konfigurasi dengan masuk ke dalam halaman wireless dan mensett wireless secara manual dan memberikan password di bagian wireless security agar jaringan lebih tersecuritas.

4.2. Implementasi aplikasi front end

Bab ini menjelaskan tentang implementasi dari hasil analisa dan desain yang telah di buat pada bab sebelumnya. Selain implementasi bab ini juga menjelaskan hasil dari pengujian aplikasi yang telah dilakukan oleh penulis. Berikut adalah tampilan yang telah di buat :

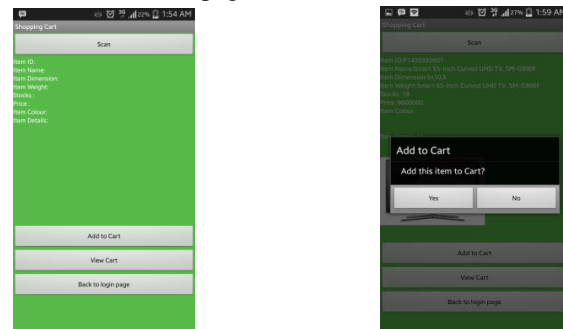
- i. Login Page dan Register page



Gambar 8.6 (a)Login Page (b)Register page

Halaman login ketika pelanggan selesai download aplikasi dan membuka aplikasi. Kemudian apabila pelanggan baru dan belum mempunyai id dan password maka pelanggan harus ke tombol sign up untuk melakukan registrasi.

- ii. Scan barcode page



Gambar 6.7 Halaman Scan Barcode

Pada bagian ini pelanggan akan memulai proses berbelanja pada suatu *hypermarket*. Pelanggan juga dapat menekan tombol *scan* barcode untuk melihat detail barang. Berikut adalah *source code* dari *barcode scanner* yang telah dibuat dengan menggunakan library *Zebra Crossing*.


```

zxingLibConfig = new ZXingLibConfig();
zxingLibConfig.useFrontLight = true;

View btnScan = findViewById(R.id.scan_button);
bckLogin= (Button)findViewById(R.id.back_button);
addToCart= (Button)findViewById(R.id.addToCart);
viewCart= (Button)findViewById(R.id.viewCart);
btnScan.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View v) {
        /*
        * Manggil barcode scanner
        */
        IntentIntegrator.initiateScan(Barcode.this, zxingLibConfig);
    }
});

```

Gambar 14.8 Code snippet Barcode scanner

iii. Detail barang page



Gambar 11.8 Detail page

Pada Gambar 3.8 di atas adalah tampilan ketika pelanggan melakukan scan barcode, kemudian detail barang akan tampil. Dengan menggunakan code request item seperti Gambar 3.9 di bawah.

```

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    switch (requestCode) {
        case IntentIntegrator.REQUEST_CODE:
            IntentResult scanResult = IntentIntegrator.parseActivityResult(requestCode,
                resultCode, data);
            if (scanResult == null) {
                return;
            }
            final String result = scanResult.getContents();
            if (result != null) {
                handler.post(new Runnable() {
                    @Override
                    public void run() {
                        tvId.setText(getString(R.string.itemid)+result);
                        /* Manggil request getItem
                        */
                        new retItem().execute(result);
                    }
                });
            }
            break;
        default:
    }
}

```

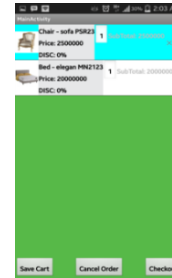
Gambar 12.9 code snippet getItem

iv. Add to cart page

Diatas adalah Gambar 3.10. Tampilan Add To Cart ketika pelanggan ingin menaruh atau meletakkan barang yang sudah dipilih ke kereta belanja atau *cart* . Pada bagian ini barang akan langsung masuk ke dalam *cart list* yang bisa di lihat ketika pelanggan menekan tombol *View Cart*

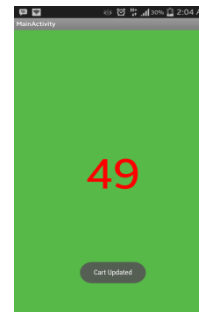
v. View cart page

Untuk melihat barang apa saja yang ingin di beli pelanggan menekan *View Cart*. Dalam *page View Cart* ini pelanggan dapat mengubah stok barang dan menghapus barang yang tidak jadi di beli dan harus menekan tombol *save* untuk mengupdate *cart* yang telah di susun atau yang telah di ubah. *Cancel order* adalah tombol yang digunakan untuk keluar tanpa pembelian. Seperti pada Gambar 3.11 di bawah.



Gambar 13.11 view cart

vi. Checkout page



Gambar 15.12 Checkout page

Pada Gambar 3.12 di atas adalah proses dimana *Checkout* telah ditekan dan aplikasi mengeluarkan id transaksi. Ketika id transaksi muncul di gadget maka kasir akan menuliskan ke program kasir setelah pelanggan melakukan pembayaran barulah proses *checkout* benar-benar terjadi.

4.3. Pengujian

Tujuan pengujian adalah untuk mengetahui tingkat kenyamanan pengguna aplikasi yang telah dibuat. Pengujian dilakukan pada 20 responden dengan umur 22-49 tahun. Router yang digunakan untuk pengujian adalah router cisco Lynksys X1000.

Tabel 4.1

Pertanyaan	Indikator				
	SS	S	B	TS	STS
Saya tidak mengalami kesulitan dalam melakukan koneksi dengan wifi	11	5	3	1	0
Saya tidak mengalami kesulitan dalam melakukan download aplikasi	6	10	3	1	0
Saya tidak mengalami kesulitan dalam melakukan register pada aplikasi	7	5	5	3	0
Saya merasa nyaman dengan adanya barcode scan untuk melihat informasi produk	6	9	5	0	0
Menurut saya informasi produk yang diberikan jelas	5	13	2	0	0
Saya tidak mengalami kesulitan untuk menambah/mengubah/menghapus barang di shopping cart	1	9	10	0	0
Saya merasa proses checkout untuk pembayaran dikasir mudah dan cepat	9	10	2	0	0

Pengujian yang diujikan bersifat pengujian experimental dengan scenario sebagai berikut :

1. responden masuk wifi yang telah disediakan oleh tester
2. responden membuka halaman website dan mendownload aplikasi di 192.168.1.99/barcodescan/app.apk
3. responden membuka aplikasi dan register di halaman sign up
4. responden menggunakan aplikasi untuk berbelanja dan mengeksplor fitur-fitur yang ada
5. responden mengisi kuisioner yang telah disediakan.

Berikut adalah pertanyaan dari kuisioner :

- i. Pelanggan tidak mengalami kesulitan dalam melakukan koneksi dengan wifi.
- ii. Pelanggan tidak mengalami kesulitan dalam melakukan download aplikasi.
- iii. Pelanggan tidak mengalami kesulitan dalam melakukan register pada aplikasi.
- iv. Pelanggan merasa nyaman dengan adanya barcode scan untuk melihat informasi detail barang.
- v. Menurut pelanggan informasi yang diberikan sudah jelas.
- vi. Pelanggan tidak mengalami kesulitan untuk menambah/mengubah/menghapus barang di shopping list.
- vii. Pelanggan merasa proses checkout untuk pembayaran di kasir mudah dan cepat.

4.4. Hasil Pengujian

Hasil dari pengujian yang di dapat dari pertanyaan di atas adalah

1. pelanggan tidak kesulitan dalam melakukan koneksi dengan wifi (86%)
2. pelanggan tidak mengalami kesulitan dalam melakukan download aplikasi (81%)

3. pelanggan tidak melakukan kesulitan dalam membuat akun (76%)
4. pelanggan merasa nyaman dengan adanya barcode scan untuk melihat informasi barang (81%)
5. informasi produk yang diberikan jelas (83%)
6. Pelanggan tidak memiliki kesulitan untuk menambah, mengubah, dan menghapus barang di *cart* (71%)
7. pelanggan merasa proses checkout untuk pembayaran di kasir mudah dan cepat (91%).

Hasil dari 20 responden yang telah di ujikan menyatakan bahwa responden tidak mengalami kesulitan dalam penggunaan aplikasi ini. Dan dari hasil pengujian terdapat masalah dan kritik mengenai tombol *cancel order*.

5. Kesimpulan

Berdasarkan hasil implementasi dan pengujian pada bab 4 sebelumnya dapat disimpulkan bahwa Aplikasi front end *virtual shopping cart* berhasil dibangun dan dinilai nyaman untuk digunakan dalam proses belanja di hypermarket. Topologi jaringan yang digunakan adalah jaringan topologi bintang dengan cara mengkoneksikan gadget pelanggan ke router dan komputer kasir ke router.

6. Saran

- Saran untuk pengembangan penelitian ini adalah
- i. Pengembangan obyek penelitian tentang adopsi teknologi *Virtual Shopping Cart* yang kepada sebuah hypermarket.
 - ii. Perancangan desain infrastruktur yang lebih tersecuritas.

DAFTAR PUSTAKA

- Android Developers. 2010 *Managing Android Projects*. <http://developer.android.com/tools/projects/index.html>. Diakses 22 mei 2014
- Meier, Reto, 2010. *Professional Android 2 Application Development*. United States of America : Wiley Publishing, Inc.
- QUERINI, M., GRILLO, A., LENTINI, A., & ITALIANO, G. F. 2011. *2D COLOR BARCODES FOR MOBILE PHONES*
- ZXing(zebra crossing), 2011, <https://github.com/zxing/zxing>, Diakses 05 Febuari 2015
- Joefrie, Y.Y, 2013. Perancangan Program Simulasi Perintah Dasar Jaringan Komputer. *Jurnal Ilmiah Foristek* Vol.3, 296.
- Wirawan & Sumarianta, 2011. Implementasi Load Balance Pada Jaringan Multithoming Menggunakan Router Dengan Metode Round Robin.
- Alqathani, Abdullah H.,Iftikhiar, Mohnis. 2013. *TCP/IP Attacks, Defenses and Security Tool* Vol 1