

RANCANG BANGUN APLIKASI SMART CASHIER BERBASIS QT*Design Smart Cashier Application Based on QT***Ihksan Hidayat Masrian Noor¹, Unang Sunarya S.T.,M.T², Suci Aulia S.T.,M.T³**¹²³ Prodi D3 Teknik Telekomunikasi, Fakultas Ilmu Terapan, Universitas Telkom¹ikhshan297@gmail.com ²unangsunarya@telkomuniversity.ac.id ³sucia@telkomuniversity.ac.id

Abstrak

Proses perekonomian masyarakat sebagian besar ditopang dalam sebuah proses jual beli dan hal ini terjadi dalam suatu pasar, semakin pesatnya perkembangan zaman pasar pun berkembang dari pasar tradisional menuju pasar yang lebih modern orang-orang lebih memilih pasar modern seperti minimarket, supermarket dan swalayan yang cenderung memberikan fasilitas antara lain tempat yang nyaman, lingkungan yang kondusif, adanya troli belanja dan harga yang pasti. Namun tingkat kenyamanan konsumen akan lebih puas apabila adanya sistem yang mendukung terciptanya waktu yang lebih efisien dalam proses transaksi. Dikarenakan pada sistem kasir pasar modern yang sekarang sering terjadi eror pada saat melakukan *scan barcode* barang saat melakukan pembayaran di kasir.

Proyek tugas akhir ini telah dilakukan pengembangan sistem pada proses transaksi dikasir pasar modern. Dimana pada proyek ini berhasil mengefisienkan waktu saat melakukan pembayaran di kasir. Dengan cara menggunkan teknologi RFID untuk mengidentifikasi setiap barang pada troli belanja konsumen sehingga tidak perlu lagi melakukan *scan barcode* pada kasir. Dan aplikasi kasir yang mendukung teknologi RFID tersebut.

Hasil dari pengujian adalah proses pembayaran dengan aplikasi *Smart Cashier* lebih cepat 42,94 detik dibandingkan aplikasi kasir biasa. Pada pengujian tersebut digunakan sepuluh barang untuk melakukan transaksi. Dan juga karena tidak melakukan *scan barcode* dikasir maka kejadian eror apabila melakukan *scan barcode* dikasir sudah dapat dihindari dan dapat mengurangi panjang antrian.

Kata Kunci: kasir, pasar modern, teknologi, dan aplikasi kasir

Abstract

Process of the community's economy is largely supported in a process of buying and selling and this is happening in a market, the more rapid development of time of any market evolved from traditional markets towards a more modern market people prefer modern market such as supermarkets and convenience stores, which tend to provide facilities among other convenient place, a conducive environment, the existence of the shopping cart and the price for sure. But the comfort level of consumers would be more satisfied if the existence of a system that supports the creation of a more efficient transaction process. Due to the modern market cashier system now common error at the time of the scan the barcode of items when making payment at the cashier.

Project task has been performed on the transaction process system development at the cashier modern market. Where on this project managed to streamline the time when making payment at the cashier. Either way the RFID technology to identify each item on the shopping cart so that consumers no longer need to route the scan the barcode on the cash register. And cashier applications that support the RFID technology.

The result of the test is the payment process with application of the Smart Cashier 42.94 seconds faster than the regular cashier application. On the test used ten items to make a transaction. And also because it does not perform a scan barcode at the cashier then error events when doing a scan of the barcode at the cashier already be avoided and can reduce the length of the queuee.

Key Words: cashier, modern market, technology, application cashier.

1. Pendahuluan

Pada dasarnya pasar modern mempunyai kesamaan seperti pasar tradisional yang membedakannya adalah pasar modern mempunyai konsep yang mengedepankan teknologi termaksud dalam proses pembayaran dimana transaksi menggunakan sistem komputer kasir[1]. Walaupun teknologi kasir yang digunakan sudah sangat cukup namun terkadang pada proses itu sering terjadi kendala seperti kerusakan pada sistem *barcode* dan terkadang keinginan konsumen untuk mengurangi atau menambahkan jumlah barang belanjaan dapat menyebabkan lamanya proses antri dikasir.

Untuk mengetahui seberapa besar dampak yang di rasakan dari kurang maksimalnya teknologi penggunaan kasir saya membuat survei berupa kuisisioner yang telah diisi oleh para konsumen pasar modern dari seluruh responder 79% responden pernah mengalami kejadian eror saat melakukan *scan barcode* dikasir, dan hanya 21% yang belum pernah, juga dengan adanya eror di kasir yang mengakibatkan proses antrian lebih lama 20,3% responder cukup terganggu, 41,9% terganggu, 31,1% sangat terganggu dengan demikian hanya 6,7% responder yang tidak terganggu dapat di tarik kesimpulan dari survei yang saya buat sebagian besar orang terganggu dan belum merasa puas dengan teknologi yang ada pada kasir saat ini.

Maka dari hasil kuisisioner yang telah saya amati membuat fokus penulisan tugas akhir ini adalah teknologi yang ada pada pasar modern dimana penulis ingin mengembangkan lagi teknologi yang digunakan di pasar modern agar dapat lebih menarik minat masyarakat selaku konsumen. Dengan bantuan teknologi *smart market* masalah masalah-masalah yang banyak konsumen keluhkan para konsumen di pasar modern akan teratasi, berdasar pada hasil dari kuisisioner tersebut penulis yakin *smart market* akan sangat membantu pelayanan pada pasar modern.

Smart Market adalah konsep teknologi yang akan lebih memudahkan proses pada sistem pembayaran di pasar modern. Dengan konsep ini proses transaksi dipasar akan lebih mudah dengan bantuan teknologi yang lebih baru. Dengan teknologi ini pekerjaan manusia akan sangat dipermudah dan akan mengurangi kesalahan pada proses transaksi, dan juga akan lebih memudahkan konsumen dalam beberapa hal salah satunya adalah dalam 2 mengefisienkan waktu dan anggarannya dalam belanja. Konsep *smart market* yang ini berdasar pada pembuatan program *smart cashier* menggunakan *rfid* sebelumnya penelitian sejenis yang membuat konsep *smart casier* dengan *rfid* ini sudah ada namun digunakan untuk proses pembayaran sedangkan konsep yang penulis ciptakan ini menggunakan *rfid* untuk identitas barang yang akan lebih bisa mengatasi keluhan konsumen soal antran. Maka dari itu perpaduan teknologi ini pada kasir di pasar modern akan membuat proses pembayaran menjadi lebih mudah, terarah, efektif dan efisien dari segi penggunaan sumber daya manusia dan juga akan meminimalisir lamanya saat melakukan antri.

2. Dasar Teori

2.1 C++

Bahasa C++ adalah Bahasa yang kuat untuk mendukung banyak gaya pemrograman yang berbeda. C++ telah digunakan selama bertahun-tahun sebelum itu standar pada tahun 1989, yang membuat Bahasa matang dan relative dibandingkan dengan yang populer digunakan saat ini. C++ sangat penting untuk membuat dengan cepat, efisien, dan misi-kritis sistem. C++ juga merupakan Bahasa pemograman yang sangat fleksibel dan banyak pengembangnya sehingga dapat digunakan pada tingkat tinggi GUI kode serta *lowlevel device driver*[2].

2.2 QT



Gambar 2.1 Logo Qt

Qt adalah sebuah sistem modular kelas dan alat yang dapat membuat lebih mudah. Qt sebetulnya bukan hanya pustaka, melainkan sudah merupakan *framework* untuk GUI. Qt itu sangat lengkap dan Qt adalah *framework* yang membangun KDE, *Unity*, *Texmaker*, dan lain-lain[3].

Qt merupakan IDE yang menyediakan *tools* untuk mendesain dan mengembangkan aplikasi menggunakan *framework* aplikasi Qt. QT merancang tampilan dan mengembangkan aplikasi sekali kemudian menyebarkan aplikasi tersebut ke berbagai *platform desktop* dan *platform mobile*. QT menyediakan *tools* menyeluruh dalam mengembangkan aplikasi dimulai dari memulai projek dan menyebarkan aplikasi ke berbagai target *platform*[4].

2.3 Database

Database adalah kumpulan informasi yang disimpan didalam computer secara sistematika sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. *Database* memiliki dua sifat yaitu *internal* dan terbagi (*share*) [7].

2.4 SQL

SQL (*Structured Query Language*) adalah sebuah bahasa yang digunakan untuk mengakses data dalam basis data. SQL sering disebut juga sebagai *query*. Saat ini hampir semua server basis data yang ada mendukung bahasa ini untuk melakukan manajemen datanya[8].

2.5 DBMS

DBMS merupakan suatu sistem perangkat lunak yang memungkinkan pengguna untuk membuat, memelihara, mengontrol, dan mengakses *database* secara praktis dan efisien. Ada begitu banyak DBMS yang populer digunakan di dunia saat ini mulai dari yang *opensource*, gratis, sampai yang lisensi berbayar.

2.6 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi diseluruh dunia. MySQL juga sangat mudah untuk digunakan para pemula. Mysql adalah sebuah server *database open source* yang terkenal yang digunakan berbagai aplikasi terutama untuk server atau membuat WEB. Mysql berfungsi sebagai SQL (*Structured Query Language*) yang dimiliki sendiri dan sudah diperluas oleh Mysql umumnya digunakan bersamaan dengan PHP untuk membuat aplikasi server yang dinamis dan memiliki banyak fungsi[10].

2.7 Pasar Modern

Pasar modern adalah tempat terjadinya transaksi antara pembeli dengan penjual dengan harga yang pas sehingga tidak akan terjadi tawar menawar. Pada pasar modern kualitas barang sangat baik dan kondisi pasar sangat bersih dan nyaman. Biasanya pasar modern berada di perkotaan. Contoh pasar modern adalah mall, swalayan dan *minimarket*.

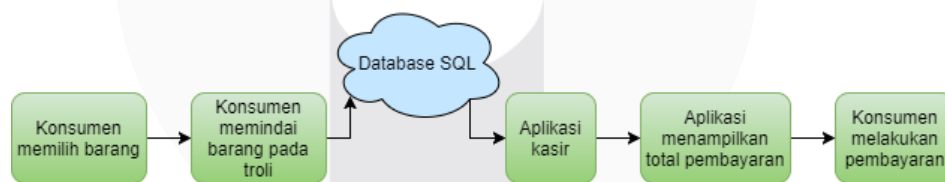
2.8 Kasir

Kasir menurut kamus besar bahasa Indonesia adalah seorang yang memegang kas (uang) atau orang yang bertugas layanin proses transaksi. Setelah istilah kasir ada juga istilah kassa, kassa adalah ada tempatnya sedangkan kasir adalah orangnya. Selain itu tugas dari kasir juga melakukan penghitungan jumlah total penjualan barang yang terjual. Kasir jugamemiliki tugas dan tanggung jawab berikut tugas dan tanggung jawab kasir[12].

3. Perancangan Sistem

3.1 Blok Diagram Sistem

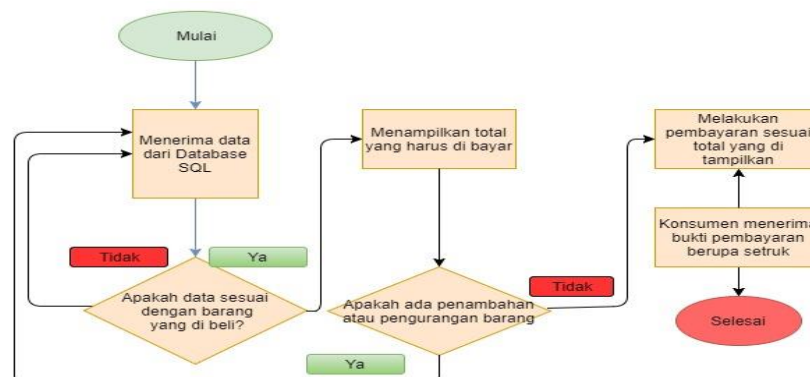
Pada bab ini menjelaskan rancangan sistem yang akan dilakukan aplikasi *Smart Cashier* ini. Berikut diagram blok alur kerja sistem pada aplikasi kasir:



Gambar 3.1 Blok Diagram Sistem

Pada gambar diatas menunjukkan cara kerja sistem, pada awalnya aplikasi kasir mendapatkan data atau informasi dari *database SQL*, selanjutnya aplikasi *Smart Cashier* menampilkan total jumlah harga yang harus dibayar oleh pembeli, kemudian pembeli membayar dengan jumlah yang sudah disetujui.

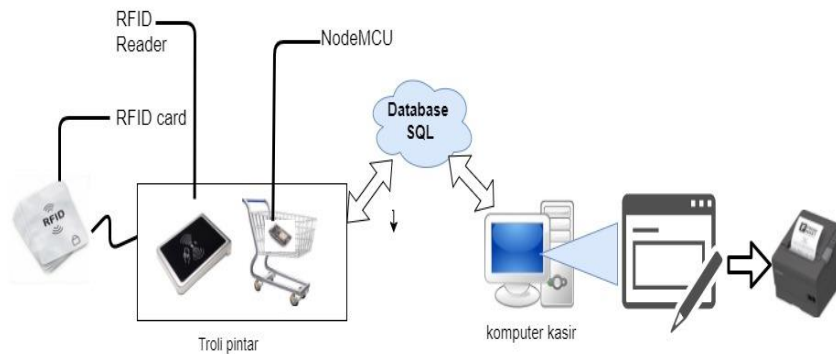
3.2 FlowChart Sistem



Gambar 3.2 Flowchart Sistem

Pada proses aplikasi ini, aplikasi menerima data dari *database* yang telah dikirimkan ke aplikasi kasir. Jika data sesuai dengan pelanggan maka aplikasi akan menampilkan total pembayaran, dan jika tidak maka aplikasi akan mengirimkan kembali data tersebut ke *database* dan meminta data yang benar. Aplikasi telah menampilkan total pembayaran, jika pelanggan ingin melakukan perubahan pada total pembayaran maka pelanggan melakukan pindai ulang pada troli belanjaan. Jika benar maka pelanggan membayar total belanjaan yang tertera pada aplikasi kasir. Setelah selesai membayar maka pelanggan akan menerima setruk belanjaan dari kasir sebagai bukti pembayaran.

3.3 Arsitektur Aplikasi Smart Cashier



Gambar 3.3 FlowChart Sistem GPS

Saat melakukan proses belanja dipasar modern, konsumen memilih barang yang ingin dibeli. Lalu disetiap barang memiliki RFID Card, kemudian konsumen menempelkan RFID Card ke RFID Reader yang berada pada troli. Lalu pada nodeMCU informasi diterima dari RFID Reader dan diproses, kemudian data dikirim ke database SQL. Lalu aplikasi Smart Cashier menerima data belanjaan konsumen dari database SQL. Data tersebut ditampilkan pada antarmuka aplikasi Smart Cashier. Total yang harus dibayar oleh konsumen pun tampil dilayar kasir, lalu konsumen membayar sesuai yang ditampilkan oleh aplikasi Smart Cashier. Setelah semua proses transaksi selesai maka konsumen akan mendapatkan bukti pembayaran berupa setruk pembayaran.

3.4 Rencana Pengujian

Pengujian pada aplikasi Smart Cashier ini akan dilakukan simulasi dengan sepuluh contoh barang dan satu teman penulis sendiri sebagai konsumen. Pada sepuluh barang tersebut akan dicoba melalui proses yang sesuai dengan arsitektur aplikasi Smart Cashier. Sepuluh barang itu adalah sebagai berikut,

Tabel 3. 1 Daftar perencanaan barang yang akan digunakan

| Nama Barang | Harga Barang(Rp) |
|-----------------|------------------|
| Indomie Goreng | 5000 |
| Mayonaise | 7000 |
| Biskuit | 7000 |
| Keripik Kentang | 8000 |
| Roti Coklat | 12000 |
| Air Mineral | 2500 |
| Teh Hijau | 4000 |
| Indomie Kuah | 3000 |
| Susu | 7000 |
| Kola Kola | 8000 |

Sepuluh barang tersebut akan didaftarkan kedalam database yang digunakan. Lalu setiap barang akan dikumpulkan didalam database troli. Setelah itu konsumen pergi ke kasir untuk membayar barang yang telah

dipilih dan menyebutkan kode dari troli yang digunakan. Pada proses pengujian ini akan diukur efisiensi waktu yang digunakan dan dibandingkan dengan proses transaksi aplikasi kasir biasa. Dan dukungan dari konsumen melalui kuisioner untuk menilai apakah aplikasi ini dapat mengurangi panjang antrian atau tidak.

4. Hasil dan Pengujian





4.1 Pengujian Aplikasi Smart Cashier

Untuk mengetahui hasil pengujian aplikasi *Smart Cashier*, maka dilakukan perbandingan waktu pada saat melakukan transaksi dengan aplikasi kasir biasa. Setiap transaksi sama-sama dilakukan dengan melakukan pembayaran dengan *sepuluh barang*.

4.1.1 Perbandingan Aplikasi Smart Cashier dengan Aplikasi Kasir Biasa

Pada pengujian ini akan diambil sepuluh barang yang sudah direncanakan pada bab 3. Pengujian akan dilakukan dengan membandingkan waktu tercepat pada saat melakukan transaksi antara konsumen dengan kasir. Berikut hasil pengujian,

Tabel 4.1 Hasil pengujian dengan aplikasi kasir biasa dan aplikasi Smart Cashier

| Pengujian Waktu Lama Transaksi | | | |
|--------------------------------|---|--|---|
| Nama barang | Gambar Barang | Waktu Transaksi Kasir Biasa | Waktu Transaksi Smart Cashier |
| Indomie Goreng |  | Waktu yang digunakan untuk melakukan transaksi dengan sepuluh barang adalah 55,32 detik menggunakan Aplikasi kasir biasa | Waktu yang digunakan untuk melakukan transaksi dengan sepuluh barang adalah 12,38 detik menggunakan Aplikasi <i>Smart Cashier</i> |
| Mayonaise |  | | |
| Biskuit |  | | |
| Keripik Kentang |  | | |

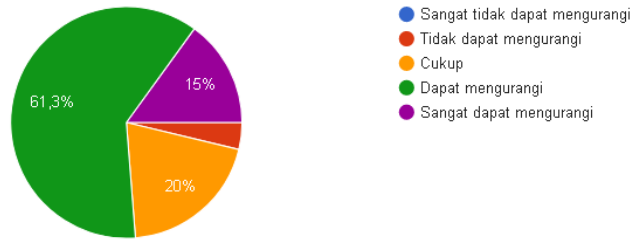
| | | | |
|--------------|---|--|---|
| Roti Coklat |  | Waktu yang digunakan untuk melakukan transaksi dengan sepuluh barang adalah 55,32 detik menggunakan Aplikasi kasir biasa | Waktu yang digunakan untuk melakukan transaksi dengan sepuluh barang adalah 12,38 detik menggunakan Aplikasi <i>Smart Cashier</i> |
| Air Mineral |  | | |
| Teh Hijau |  | | |
| Indomie Kuah |  | | |
| Susu |  | | |
| Kola Kola |  | | |

Pengujian ini dilakukan didalam satu ruangan saja dengan menggunakan jaringan lokal dan menggunakan jaringan tethering dari ponsel genggam. Dari troli mengirim sepuluh barang kedalam *database* troli itu sendiri dengan waktu 2,21 detik. Kemudian aplikasi *Smart Cashier* menerima data konsumen dari *database* yang telah menerima data dari troli sebanyak sepuluh barang dan akan tampil seperti pada gambar 4.3 apabila telah dimasukkan kode troli pelanggan. Dari Tabel 4.1 terlihat hasil perbandingan waktu melakukan transaksi dengan aplikasi kasir biasa dan aplikasi *Smart Cashier*. Terlihat perbedaan diantaranya yaitu 42,94 detik dan aplikasi *Smart Cashier* memiliki waktu transaksi yang lebih cepat dibandingkan dengan aplikasi kasir biasa.

4.1.2 Hasil Survei Konsumen

Apakah sistem kasir pintar dapat mengurangi panjang antrian?

80 tanggapan

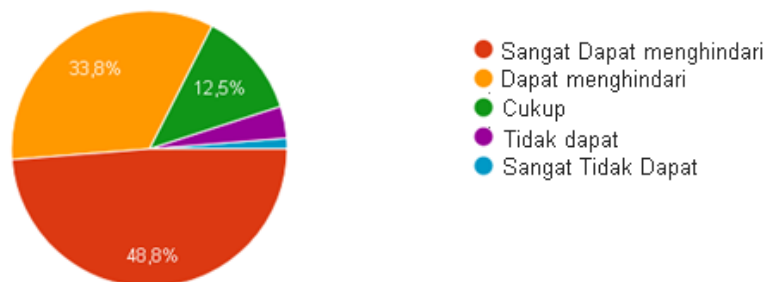


Gambar 4. 1 Hasil survei konsumen pada *Smart Cashier*

Dari Gambar 4.1 terlihat hasil survei yang dilakukan oleh 80 responder dengan hasil 96,3% berpendapat bahwa aplikasi *Smart Cashier* dapat mengurangi panjang antrian. Aplikasi *Smart Cashier* juga dapat menghindari adanya resiko sistem eror pada kasir karena sudah tidak lagi melakukan *scan barcode* dikasir. Dan menurut survei yang saya lakukan kepada 80 responder 95,1% berpendapat bahwa aplikasi *Smart Cashier* dapat menghindari kerusakan *scan barcode* pada sistem kasir biasa, dan hasil survei dapat dilihat pada gambar 4.2.

Apakah kasir pintar dapat menghindari eror pada sistem scan barcode dikasir biasa? Dikarena proses scan barcode dikasir sudah ditiadakan dan diganti RFID pada troli

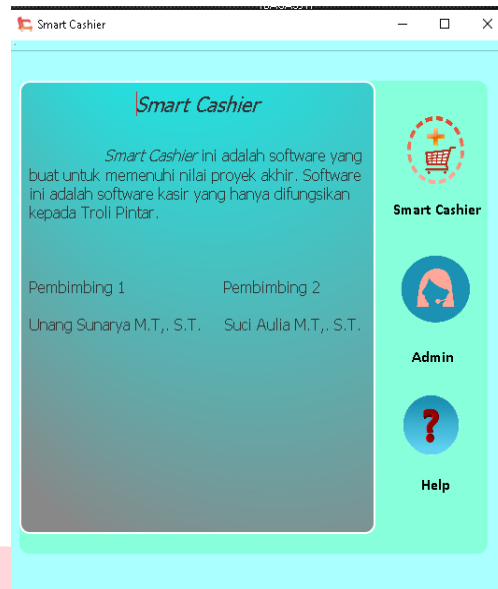
80 tanggapan



Gambar 4. 2 Hasil survei konsumen pada *Smart Cashier*

4.2 Hasil Aplikasi Yang Dibuat

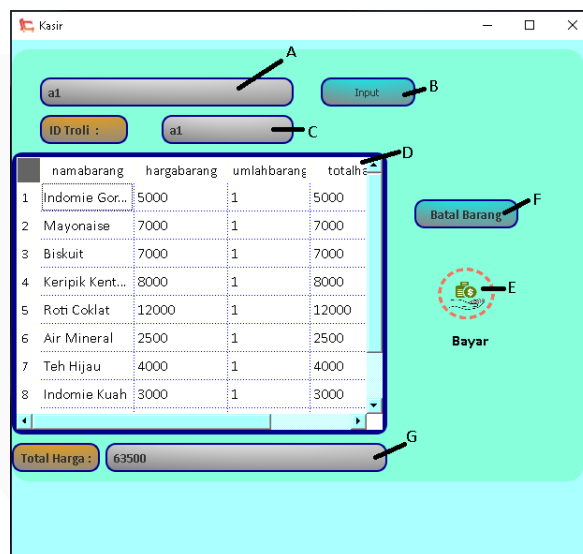
4.2.1 Menu Utama



Gambar 4. 3 Markup menu utama aplikasi Smart Cashier

Pada gambar 4.3 terlihat *markup* menu utama dari aplikasi *Smart Cashier*. Pada gambar tersebut terlihat di menu utama ada tombol , *admin*, dan *help*. Ada juga sedikit pengenalan terhadap aplikasi *Smart Cashier*.

4.1.1 Menu Smart Cashier

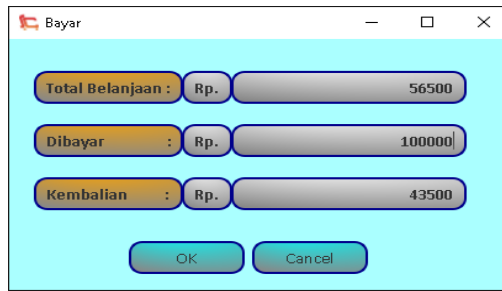


Gambar 4. 4 Markup menu Smart Cashier

Keterangan gambar :

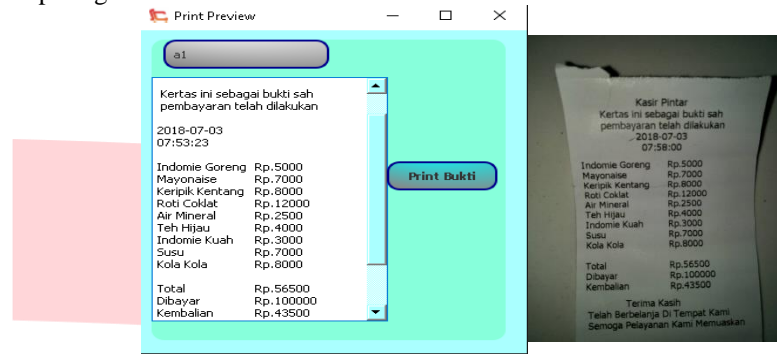
- Tempat untuk memasukkan kode troli.
- Tombol untuk memasukkan kode troli.
- Setelah kode troli berhasil ditemukan maka akan muncul di sini.
- Tabel untuk menunjukkan hasil belanjaan dari konsumen yang terdiri dari nama barang, harga barang, jumlah barang, dan total harga.
- Tombol untuk melakukan pembayaran apabila isi table sesuai dengan belanjaan konsumen.
- Tombol untuk melakukan pembatalan barang.
- Menampilkan total dari seluruh harga belanjaan konsumen.

Pada gambar 4.4 sudah terlihat tampilan dari menu *Smart Cashier*, di menu inilah dilakukannya transaksi antara penjual dengan konsumen. Yaitu dengan cara memasukkan kode troli yang digunakan konsumen ke tanda A pada gambar 4.4, lalu klik tombol input maka aplikasi akan mencari *didatabase* untuk kode troli tersebut. Kemudian jika berhasil maka akan muncul daftar belanjaan konsumen pada tanda D yang ada pada gambar 4.4. Jika semua benar maka klik tombol bayar yang ada pada tanda E yang ada pada gambar 4.4 dan akan muncul seperti gambar 4.5.



Gambar 4. 5 Markup bayar

Jika uang konsumen cukup atau lebih maka akan bisa melakukan pembayaran dan akan mendapatkan struk seperti gambar 4.6.



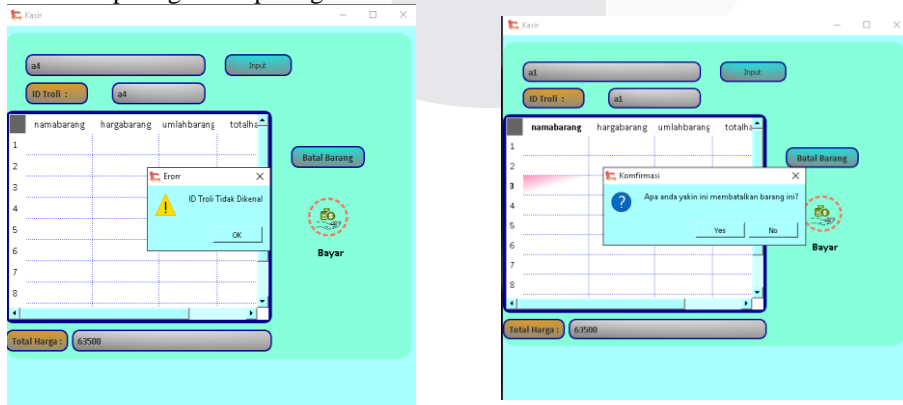
Gambar 4. 6 Markup print preview dan struk dari belanjaan konsumen

Jika uang konsumen kurang maka tidak dapat melakukan pembayaran dan akan muncul peringatan seperti gambar 4.7.



Gambar 4. 7 Peringatan jika uang dari konsumen kurang

Apabila kode troli tidak terdeteksi dan apabila ingin melakukan pembatalan pada barang maka akan ada peringatan seperti gambar 4.8.



Gambar 4. 8 Peringatan jika kode troli tidak dikenal dan konfirmasi pembatalan barang

Jika konsumen menyetujui bahwa barang tersebut batal untuk dibeli maka tekan “Yes”. Setelah itu barang yang telah dipilih untuk dibatalkan akan menghilang, dan total dari harga pun ikut berubah menyesuaikan jumlah barang yang telah dipilih.

4.2.3 Menu Admin



| | id | rfidcarditem | namabarang | hargabarang | jumlahbarang |
|---|----|--------------|-----------------|-------------|--------------|
| 1 | 1 | D0 55 81 A6 | Indomie Gor... | 5000 | 10 |
| 2 | 2 | 70 CF E2 A4 | Mayonaise | 7000 | 10 |
| 3 | 3 | A0 48 7A A6 | Biskuit | 7000 | 10 |
| 4 | 4 | E0 2C E1 A4 | Tolak Angin | 3000 | 10 |
| 5 | 5 | F0 06 83 A6 | Keripik Kent... | 8000 | 10 |
| 6 | 6 | 60 BA 84 A6 | Roti Coklat | 12000 | 10 |
| 7 | 7 | C0 69 84 A6 | Air Mineral | 2500 | 10 |
| 8 | 8 | C0 67 E0 A4 | Teh Hijau | 4000 | 10 |
| 9 | 9 | 25 BA 7A 11 | Indomie Kuah | 3000 | 10 |

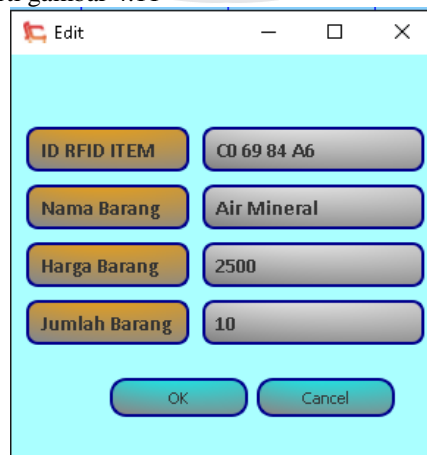
Gambar 4. 9 Markup Menu Admin

Pada menu ini digunakan untuk melihat stok barang yang tersedia ditoko dan dapat melakukan perubahan terhadap barang yang tersedia. Jika ingin menambahkan barang pada stok ditoko dengan klik tombol *Add Item* maka akan muncul seperti gambar 4.10



Gambar 4. 10 Markup add item

Apabila ingin melakukan penghapusan barang maka klik barang yang ingin dihapus kemudian klik *Delete* dan jika ingin melakukan perubahan pada data yang sudah ada maka pilih barang yang ingin di ubah kemudian klik tombol *Edit*, maka akan muncul seperti gambar 4.11



Gambar 4. 11 Markup Edit

4.2.4 Database Yang Telah Dibuat

| + Options | | id | rfidcarditem | namabarang | hargabarang | jumlahbarang |
|--------------------------|------------------|----|--------------|-----------------|-------------|--------------|
| <input type="checkbox"/> | Edit Copy Delete | 1 | D0 55 81 A6 | Indomie Goreng | 5000 | 10 |
| <input type="checkbox"/> | Edit Copy Delete | 2 | 70 CF E2 A4 | Mayonaise | 7000 | 10 |
| <input type="checkbox"/> | Edit Copy Delete | 3 | A0 4B 7A A6 | Biskuit | 7000 | 10 |
| <input type="checkbox"/> | Edit Copy Delete | 4 | A2 B3 22 H2 | Susu | 7000 | 10 |
| <input type="checkbox"/> | Edit Copy Delete | 5 | F0 0B 83 A6 | Keripik Kentang | 8000 | 10 |
| <input type="checkbox"/> | Edit Copy Delete | 6 | 60 BA 84 A6 | Roti Coklat | 12000 | 10 |
| <input type="checkbox"/> | Edit Copy Delete | 7 | C0 69 84 A6 | Air Mineral | 2500 | 10 |
| <input type="checkbox"/> | Edit Copy Delete | 8 | C0 67 E0 A4 | Teh Hijau | 4000 | 10 |
| <input type="checkbox"/> | Edit Copy Delete | 9 | 25 BA 7A 11 | Indomie Kuah | 3000 | 10 |
| <input type="checkbox"/> | Edit Copy Delete | 10 | B2 H1 21 K2 | Kola Kola | 8000 | 10 |
| <input type="checkbox"/> | Edit Copy Delete | 11 | B2 32 55 C2 | Kertas HVS | 1000 | 10 |

Gambar 4. 12 Daftar barang yang ada di toko pada database

Pada gambar 4.12 terlihat daftar barang yang tersedia. Pada *database* tercantumkan kode RFID setiap barang, nama barang, harga barang, dan jumlah barang yang tersedia ditoko.

| + Options | | id | namabarang | hargabarang | jumlahbarang | totalharga |
|--------------------------|------------------|----|-----------------|-------------|--------------|------------|
| <input type="checkbox"/> | Edit Copy Delete | 1 | Indomie Goreng | 5000 | 1 | 5000 |
| <input type="checkbox"/> | Edit Copy Delete | 2 | Mayonaise | 7000 | 1 | 7000 |
| <input type="checkbox"/> | Edit Copy Delete | 4 | Keripik Kentang | 8000 | 1 | 8000 |
| <input type="checkbox"/> | Edit Copy Delete | 5 | Roti Coklat | 12000 | 1 | 12000 |
| <input type="checkbox"/> | Edit Copy Delete | 6 | Air Mineral | 2500 | 1 | 2500 |
| <input type="checkbox"/> | Edit Copy Delete | 7 | Teh Hijau | 4000 | 1 | 4000 |
| <input type="checkbox"/> | Edit Copy Delete | 8 | Indomie Kuah | 3000 | 1 | 3000 |
| <input type="checkbox"/> | Edit Copy Delete | 9 | Susu | 7000 | 1 | 7000 |
| <input type="checkbox"/> | Edit Copy Delete | 10 | Kola Kola | 8000 | 1 | 8000 |

Gambar 4. 13 Daftar barang yang sudah masuk kedalam table troli

Pada gambar diatas adalah kondisi apabila troli telah mengirim data belanjaan konsumen ke *database*. Setelah semua data sudah diterima maka akan dapat diproses pada aplikasi *Smart Cashier* dan konsumen dapat meneruskan proses transaksi ke kasir untuk melakukan pembayaran dengan menyebutkan kode troli yang sudah tertera pada troli yang digunakan. Tetapi apabila troli belum mengirimkan data belanjaan konsumen makan *database* akan tampil seperti gambar 4.14.



Gambar 4. 14 Kondisi jika troli belum menerima data konsumen

Pada *database* dapat dilakukan menggunakan aplikasi *Smart Cashier* karena pada aplikasi *Smart Cashier* sudah terdapat fungsi-fungsi yang dapat mengubah, menghapus dan menambahkan *database*. Jadi admin tidak perlu repot untuk melakukan perubahan pada database langsung karena tidak semua pengguna memahami hal tersebut. Jadi sudah dimudahkan oleh aplikasi *Smart Cashier*.

5 Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan paparan yang telah dijelaskan pada bab-bab sebelumnya, bahwa aplikasi *Smart Cashier*:

1. Menurut hasil pengujian untuk melakukan transaksi dengan 10 barang aplikasi *Smart Cashier* memiliki waktu 12,38 detik dan aplikasi kasir biasa 55,32 detik, dengan kesimpulan bahwa aplikasi *Smart Cashier* lebih cepat 42,94 detik daripada aplikasi kasir biasa.
2. Dari hasil survei yang dilakukan kepada 80 responder menyatakan bahwa aplikasi *Smart Cashier* dapat mengurangi panjang antrian dengan rincian 20% cukup, 61,3% dapat mengurangi, 15% sangat dapat mengurangi.
3. Dari hasil pengujian diatas tidak adanya proses *scan barcode* dikasir maka akan mengurangi resiko eror pada sistem kasir, serta hasil survei yang dilakukan kepada 80 responder sebagian besar berpendapat sama dengan rincian 48,8% sangat dapat menghindari, 33,8% dapat menghindari, 12,% cukup.
4. Proyek akhir ini telah membuat aplikasi yang dapat membaca data dari *database SQL*.
5. Aplikasi *Smart Cashier* mampu mengurangi jumlah barang belanjaan konsumen, dapat melakukan transaksi dengan perhitungan yang tepat, memberikan keluaran berupa struk pembayaran ke konsumen
6. Aplikasi *Smart Cashier* memiliki menu admin yang dapat mengelola stok barang, mengubah informasi barang, menambahkan barang, dan menghapus barang yang ada didalam stok(*database*)..

5.2 Saran

1. Untuk aplikasi disarankan agar dapat menambahkan fitur tambah barang pada aplikasi.
2. Untuk fungsi admin dibuat batasan pengguna agar orang yang tidak memiliki wewenang tidak dapat mengubah informasi barang.

Daftar Pustaka

- [1] Belshaw S. Cyril, 1981, "TUKAR MENUKAR TRADIONAL DAN PASAR MODERN," Jakarta, PT Gramedia.
- [2] Raharjo Budi, 2016 "PEMOGRAMAN GUI DENGAN C++ DAN QT," Bandung, Informatika.
- [3] Akbar, A. M. Ade Malsasa Akbar., "Pengantar Pemrograman Aplikasi Gui Di Linux Pustaka Framework Dan Ide," 31 Oktober 2013(www.malsasa.wordpress.com diakses 5 Januari 2018))
- [4] Iwan, "Mengenal Qt Creator". 28 Mei 2015(www.proweb.co.id diakses 21 Maret 2018)
- [5] Ezust, A d dan Ezust P. *An Introduction to Design Patterns in C++ with Qt 4*. New York, Safari.
- [6] Prabowo, B.A, Michrandi, S. dan Osmond A.B, 2016. Sistem Informasi Akuntansi Dan Sistem Transaksi Menggunakan RFID Dengan Sistem Saldo Pada Pembeli Untuk Tokok Pintar Tanpa Kasir. *e-Proceeding of Engineering*, 2(2),p. 3540.
- [7] Wikipedia, "Pangkalan Data" 15 Juni 2018(id.wikipedia.org diakses 4 Juli 2018)
- [8] Sukanto, Kholid Muhammad, "Pengertian SQL, DDL, DML, DCL" 16 September 2015(kholidz.weebly.com diakses 28 April 2018)
- [9] Hindro, "Pengertian *Database*" 12 Desember 2017(www.termasmedia.com diakses 29 Januari 2018)
- [10] Solichin, A. 2010, "MySQL 5 Dari Pemula Hingga Mahir" Jakarta, Universitas Budi Luhur.
- [11] Suryono, "Pengertian SQL" 9 Januari 2015(<http://www.mandalamaya.com> diakses 29 Juni 2018)
- [12] Rohmah, "Pengertian Kasir" 25 Mei 2018 (www.scribd.com diakses 2 Juli 2018)
- [13] Wangandowo, "Pengertian dan Penjelasan Fitur-Fitur Yang Dimiliki C++" 13 Januari 2017(<http://www.materidosen.com> diakses 2 Juli 2018)